

Εργαστηριακή Άσκηση 2020-2021

Μελέτη Κλιμάκωσης Παράλληλων Προγραμμάτων MPI, hybrid-MPI, CUDA

ΠΡΟΣΟΧΗ Για να δώσετε γραπτά ή/και εργασία χρειάζεται να έχετε περάσει το εργαστήριο

Στόχος η εξοικείωση σας με εργαλεία απαραίτητα στην κύρια εργασία:

- Η σύνδεση με ΑΡΓΩ (RSA keys, VPN)
- Μεταγλώττιση προγράμματος MPI, MPI+OpenMp, CUDA
- εκτέλεση με PBS (εντολές qsub, qstat, qdel)
- ορισμός πυρήνων και διεργασιών στο PBS
- χρονομέτρηση και profiling προγράμματος
- μικρή μελέτη κλιμάκωσης των προγραμμάτων mpi_trapDemo.c και mpiH_trapDemo.c που υπολογίζουν το ορισμένο ολοκλήρωμα [0,3] της συνάρτησης $f(x)=x^2$ με την μέθοδο των τραπεζίων.

Βαθμοί: Μελέτη Κλιμάκωσης για καθαρά MPI (10%), Υβριδικά MPI+OpenMp (5%) και CUDA (5%). Σύνολο 2/10.

Εκφώνηση με Οδηγίες για καθαρά MPI (και αναφορές σε επιμέρους οδηγίες)

1. Δημιουργήστε κλειδί RSA (δείτε οδηγίες στο eclass)
2. Σύνδεση με VPN (δείτε οδηγίες στο eclass)
3. Ακολουθήστε Βήματα 1,2, 3 από Argo-InstructionsMPIInPBS 20-21.pdf και κατόπιν να εντοπίσετε στο πρόγραμμα mpi_trapDemo.c:
 - a. το `#define nTraps` και να το ορίσετε σε $(1024*1024*16)$ (νέα τιμή 2^{24})
 - b. την κλήση απενεργοποίησης του mpiP με `MPI_Pcontrol(0)`
 - c. τις κλήσεις `t1=MPI_Wtime()` και αμέσως μετά την ενεργοποίηση mpiP `MPI_Pcontrol(1)` (δες manual MpiP, στο τμήμα «Controlling the Scope of mpiP Profiling in your Application»)
 - d. αντίστοιχες κλήσεις `t2=MPI_Wtime()` και απενεργοποίησης του mpiP με `MPI_Pcontrol(0)`. Επομένως χρόνος εκτέλεσης είναι η διαφορά $t2-t1$, ο οποίος εκτυπώνεται και θα χρησιμοποιήσετε στα επόμενα βήματα.
4. Να μεταγλωττίσετε (Βήμα 4 Argo-InstructionsMPIInPBS 20-21.pdf) το mpi_trapDemo.c με το mpiP στο mpi_trapDemo.x
5. Στο myPBSscript.sh το όνομα του εκτελέσιμου είναι mpi_trapDEMO.x
6. Να εκτελέσετε (Βήμα 5 του Argo-InstructionsMPIInPBS 20-21.pdf) με qsub, δοκιμάστε το qstat
7. Θα πάρετε 3 αρχεία εξόδου, 2 από την εκτέλεση του προγράμματος (myJob.oxxxx, myJob.exxxx) και ένα από το mpiP. Κάνετε μια μικρή διερεύνηση των περιεχομένων τους. Για το mpiP δείτε το MpiP Manual (<https://software.llnl.gov/mpiP/>). Στο myJob.oxxxx (stdout) έχει τον αριθμό των τραπεζίων, την τιμή του ολοκληρώματος και τον χρόνο εκτέλεσης του τμήματος του προγράμματος που χρονομετρήσατε. Κρατήστε τον χρόνο.
8. Έχοντας κατακτήσει αυτά τα εργαλεία (σύνδεση, edit, compile, PBS script, qsub, αρχεία εξόδου) ξεκινήστε μικρή μελέτη κλιμάκωσης του mpi_trapDemo.c
9. Αλλάζετε διαδοχικά την τιμή select N στο script myPBSscript.sh, (N=1,2,4,8,10)
`#PBS -l select=N:ncpus=8:mpiprocs=8` (N nodes, 8 processes/node)
τρέχετε (qsub) και συμπληρώνετε τους χρόνους στα αντίστοιχα κελιά του πιο κάτω πίνακα, όπου οι στήλες αναφέρονται σε αριθμό διεργασιών 1,2,4,8,16,32,64,80 και οι γραμμές σε μέγεθος προβλήματος, δηλαδή σε πόσα τραπέζια έχετε χωρίσει το ολοκλήρωμα.

Πίνακας Χρόνων (Δεδομένα-Διεργασίες)

Δεδομένα / Διεργασίες	1	2	4	8	16	32	64	80
2^{24}								
2^{26}								
2^{28}								
2^{30}								

10. Έχετε ήδη τρέξει το πρόγραμμα για 8,16,32,64 και 80 διεργασίες με 2^{24} τραπέζια (με το #define στο πρόγραμμα). Μπορείτε να συμπληρώσετε τα 5 αντίστοιχα κελιά της πρώτης γραμμής (χρήση spreadsheet θα σας βοηθήσει). Οι χρόνοι σε δευτερόλεπτα με 3 δεκαδικά, δεν έχει νόημα μεγαλύτερη ακρίβεια.
11. Για να τρέξετε με d διεργασίες, d= 1,2,4 για τους χρόνους των τριών πρώτων στηλών χρειάζεται να ορίσετε (δείτε το Argo-InstructionsMPIinPBS-Dec20.pdf)
`#PBS -l select=1:ncpus=d:mpiprocs=d`
και να τρέξετε το mpiRun στο myPBSScript με παράμετρο `--bind-to none`
`mpirun --bind-to none mpi_trapDemo.x`, συμπληρώνοντας τους χρόνους της πρώτης γραμμής για 2^{24} .
12. Να επαναλάβετε το 9,10,11 (8 μετρήσεις), αλλάζοντας διαδοχικά στον κώδικα (#define nTraps) το 2^{24} , σε 2^{26} , 2^{28} , 2^{30} (και μεταγλώττιση!), και το script (select, ncpus, mpiprocs) ώστε να συμπληρώσετε όλον τον πίνακα (συνολικά $4 \times 8 = 32$ σύντομα τρεξίματα).
Απλός έλεγχος Χρόνων Στηλών: Σε κάθε γραμμή >1 το μέγεθος του προβλήματος τετραπλασιάζεται από την προηγούμενη, επομένως περιμένουμε οι χρόνοι της 1^{ης} στήλης (ακολουθιακό πρόγραμμα) περίπου να τετραπλασιάζονται σε κάθε σειρά. Οι χρόνοι των άλλων στηλών θα πρέπει επίσης να αυξάνονται, όχι όμως υποχρεωτικά $\times 4$.
Απλός έλεγχος Χρόνων Σειρών: Οι χρόνοι είναι λογικό να αυξομειώνονται, όμως να υπάρχει μόνο ένας (σημαντικά) ελάχιστος χρόνος, .π.χ, όχι αρχικά μείωση, μετά αύξηση και μείωση πάλι.
Αν οι χρόνοι αποκλίνουν σημαντικά από αυτούς τους εμπειρικούς «κανόνες», να ελέγξετε το script (select, ncpus, mpiprocs) και πρόγραμμα (ntraps) για λάθη, και ξανατρέχετε.
13. Να υπολογίσετε την Επιτάχυνση $-speedup S(n,p)=T_s(n)/T(n,p)$ - και την αποδοτικότητα - Efficiency $E(n,p)=S(n,p)/p$ - σε άλλους δυο πίνακες της ίδιας μορφής με τον πίνακα των χρόνων, με ακρίβεια δύο δεκαδικών. Προφανώς η στήλη 1 θα περιέχει 1 και στους δυο πίνακες.
14. Έχετε ολοκληρώσει την μελέτη κλιμάκωσης. Είναι χρήσιμο να δείτε την έξοδο (αρχείο) του mpiP, ιδιαίτερα εκεί που παίρνετε κακά αποτελέσματα (χαμηλά S, E) για να δείτε που ξοδεύει χρόνο το πρόγραμμά σας και να εντοπίσετε πιθανά θέματα βελτίωσης και να σχολιάσετε.

Κατάθεση Μελέτης καθαρών MPI (Βαθμός: Συνεισφορά στον τελικό 1 /10)

Καταθέτετε αρχείο σε μορφή <ΑριθμόΜητρώου>.pdf στο eclass με τους 3 πίνακες (χρόνων, επιτάχυνσης, αποδοτικότητας) και σχόλια για την κλιμάκωση αντλώντας στοιχεία από το mpiP για να εξηγήσετε χαμηλές επιδόσεις. Σχολιάστε την ακρίβεια του ολοκληρώματος με την αύξηση των διεργασιών. Επίσης να συμπεριλάβετε στο κείμενο την έξοδο της εντολής που εμφανίζει την χρήση σας `qstat -H -u argoXXX` (προφανώς XXX ο δικός σας αριθμός).

Προθεσμία 06/12/2020 ή/και μαζί με το υπόλοιπο εργαστήριο στις 10/1/2021. Όσοι καταθέσουν το εργαστήριο στις 6/12/2020 θα μπορούν να το ξανακαταθέσουν για βελτίωση βαθμού (κάνοντας τις απαραίτητες αλλαγές) στις 10/1/2021.

Εκφώνηση με Οδηγίες για το υβριδικό MPI+OpenMp

Να εκτελέσετε παρόμοια μελέτη κλιμάκωσης για το υβριδικό (MPI+OpenMp) πρόγραμμα mpiH_trapDemo.c με τις μισές διεργασίες (d=1,2,4,8,16,32), όμως με 2 νήματα η καθεμία, δηλαδή σε κάθε κόμβο να δημιουργούνται 4 διεργασίες αντί για 8. Οι στήλες του πίνακα χρόνων αναφέρονται τώρα σε νήματα. Η στήλη 1 παραμένει ίδια. Για d=4,8,16,32,40
`#PBS -l select=<d/4>:ncpus=8:mpiprocs=4:ompthreads=2`

Και για $d=1,2$ (αντίστοιχα 2,4 νήματα)

```
#PBS -l select=1:ncpus=8:mpiprocs=d:ompthreads=2
```

Τα υβριδικά τα τρέχετε πάντα με την

```
mpirun --bind-to none mpiH_trapDemo.x
```

Αλλιώς οι διεργασίες τρέχουν όλες στον έναν από τους δυο επεξεργαστές των κόμβων (στους 4 πυρήνες) ενώ ο άλλος είναι αδρανής.

Για 1 διεργασία και 1 νήμα να μεταφέρετε τις μετρήσεις του καθαρού MPI για τον υπολογισμό speedup και efficiency.

Κατάθεση Μελέτης του υβριδικού MPI+OpenMp (Βαθμός: Συνεισφορά στον τελικό 0,5/10)

Καταθέτετε αρχείο σε μορφή <ΑριθμόΜητρώου>.pdf στο eclass με τους 3 αντίστοιχους πίνακες και σχόλια για την κλιμάκωση και επιπλέον την σύγκριση με το καθαρό MPI πρόγραμμα. Επίσης να συμπεριλάβετε στο κείμενο την έξοδο της εντολής που εμφανίζει την χρήση σας `qstat -H -u argoXXX`.

Προθεσμία 13/12/2020 ή/και μαζί με το υπόλοιπο εργαστήριο στις 10/1/2021. Όσοι καταθέσουν το εργαστήριο στις 13/12/2020 θα μπορούν να το ξανακαταθέσουν για βελτίωση βαθμού (κάνοντας τις απαραίτητες αλλαγές) στις 10/1/2021.

Εκφώνηση με Οδηγίες για OpenMp, CUDA και υβριδικά CUDA+OpenMp

Να συγκρίνετε χρόνους εκτέλεσης του προγράμματος `simple_add1GPUOmp.cpp` και `simple_add2GPUOmp.cpp` (ακολουθιακό με 1 νήμα, OpenMp με 10 νήματα σε 10 πυρήνες και με 2 GPUs) για μέγεθος στοιχείων $16 \cdot 10^6$ (αρχική τιμή) στο διπλασιάζοντας το μέγεθος, $32 \cdot 10^6$, $64 \cdot 10^6$, $128 \cdot 10^6$, έως το μέγιστο της δυνατότητας μνήμης.

Πίνακας Χρόνων

	1 νήμα	10 νήματα/10 πυρ.	1 GPU	2 GPUs
$16 \cdot 10^6$				
$32 \cdot 10^6$				
$64 \cdot 10^6$				
$128 \cdot 10^6$				
....				

Κατάθεση Μελέτης CUDA και CUDA+OpenMp (Βαθμός: Συνεισφορά στον τελικό 0,5/10)

Καταθέτετε κείμενο σε μορφή pdf στο eclass με τον πίνακα συμπληρωμένο και σχόλια για την κλιμάκωση. Επίσης να συμπεριλάβετε στο κείμενο την έξοδο της εντολής που εμφανίζει την χρήση σας `qstat -H -u argoXXX`.

Προθεσμία 20/12/2020 ή/και μαζί με το υπόλοιπο εργαστήριο στις 10/1/2021. Όσοι καταθέσουν το εργαστήριο στις 20/12/2020 θα μπορούν να το ξανακαταθέσουν για βελτίωση βαθμού (κάνοντας τις απαραίτητες αλλαγές) στις 10/1/2021

ΤΕΛΙΚΗ ΠΡΟΘΕΣΜΙΑ 10/1/2021. Επανακατάθεση ΜΟΝΟ για βελτίωση βαθμού μετά από βελτιώσεις τις οποίες αναφέρετε ρητά.