



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ
ΕΡΓΑΣΤΗΡΙΟ ΥΠΟΛΟΓΙΣΤΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

www.cslab.ece.ntua.gr

Διπλωματικές Εργασίες
Ακ. Έτος 2011 – 2012

Παράλληλα προγραμματιστικά μοντέλα βασισμένα σε εργασίες: μελέτη και βελτιστοποιήσεις

1 Εισαγωγή

Οι σύγχρονες αρχιτεκτονικές υπολογιστών περιλαμβάνουν πλέον μεγάλο αριθμό από πυρήνες/νήματα, προσφέροντας τη δυνατότητα στους προγραμματιστές να βελτιώσουν την απόδοση των εφαρμογών τους εκτελώντας κομμάτια του κώδικα παράλληλα. Παρόλα αυτά, ο παράλληλος προγραμματισμός δεν είναι εύκολος. Για την υλοποίηση παράλληλων εφαρμογών έχουν προταθεί διαφορετικά προγραμματιστικά μοντέλα, τα οποία προσπαθούν να εκμεταλλευτούν συγκεκριμένα χαρακτηριστικά των αλγορίθμων/εφαρμογών.

Στην κατεύθυνση της απλοποίησης του παράλληλου προγραμματισμού, ένα δημοφιλές μοντέλο που έχει προταθεί από την ερευνητική κοινότητα είναι αυτό του παράλληλου προγραμματισμού βασισμένου σε εργασίες (task programming model). Σύμφωνα με το μοντέλο αυτό, ο προγραμματιστής εκφράζει τον παραλληλισμό δηλώνοντας *ποιες* εργασίες μπορούν να εκτελεστούν παράλληλα, αφήνοντας στον δρομολογητή του runtime συστήματος να αποφασίσει *πώς* οι εργασίες αυτές θα απεικονιστούν στα νήματα του συστήματος με τον πλέον βέλτιστο τρόπο. Το μοντέλο αυτό υλοποιείται από νέες γλώσσες ή βιβλιοθήκες όπως η Cilk, τα Threading Building Blocks (TBB) της Intel, η Parallel Patterns Library (PPL) της Microsoft, η Chapel της Cray, κ.λπ..

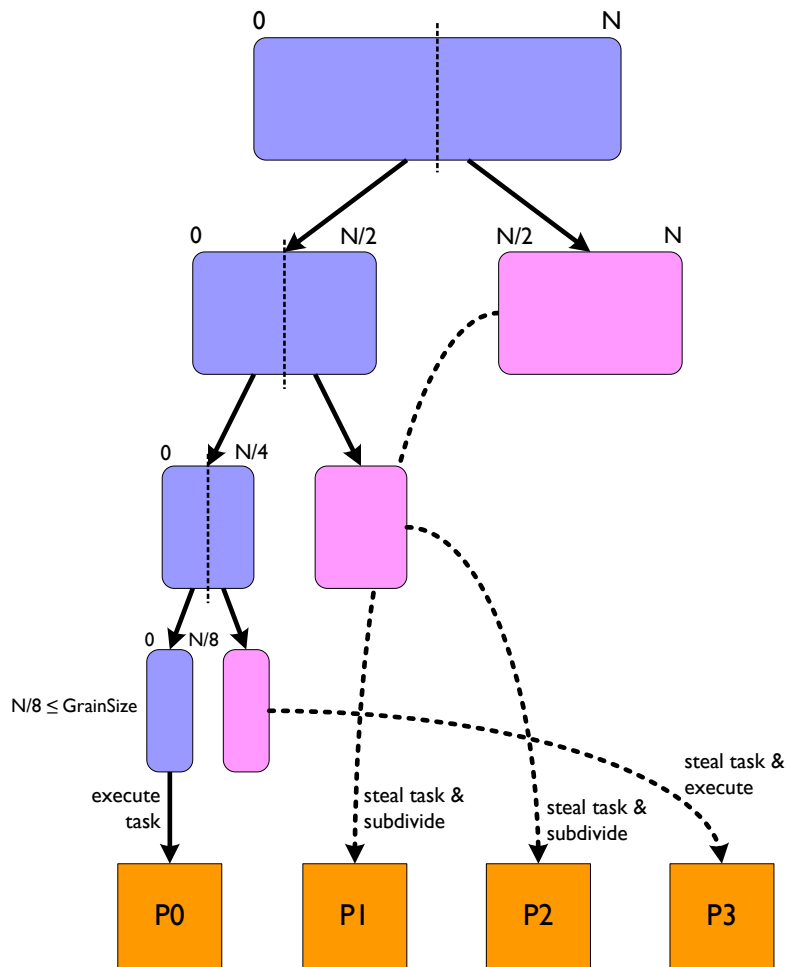
Δύο βασικοί μηχανισμοί στους οποίους στηρίζεται η εκτέλεση παράλληλων προγραμματιστικών δομών σε αρκετές από τις παραπάνω βιβλιοθήκες είναι η **αναδρομική διάσπαση** (recursive splitting) και η **κλοπή εργασιών** (task stealing). Στην πρώτη περίπτωση, το αρχικό πρόβλημα διασπάται αναδρομικά σε μικρότερα υπο-προβλήματα, τα οποία πρακτικά συνιστούν εν δυνάμει παράλληλες εργασίες. Η διάσπαση γίνεται μέχρι το μέγεθος των εργασιών να ικανοποιεί συγκεκριμένες απαιτήσεις ως προς την ισοκατανομή του φορτίου εργασίας (load balancing) ή την εκμετάλλευση της τοπικότητας των αναφορών. Στη δεύτερη περίπτωση, οι άεργοι επεξεργαστές προσπαθούν να κλέψουν εργασίες που προκύπτουν από την αναδρομική διάσπαση, και οι οποίες τοποθετούνται συνήθως στην τοπική ουρά του εκάστοτε επεξεργαστή όπου γίνεται η διάσπαση. Με αυτόν τον τρόπο επιτυγχάνεται τελικά η ισοκατανομή του συνολικού φορτίου εργασίας ανάμεσα σε όλους τους επεξεργαστές του συστήματος. Στο Σχήμα 1 παρουσιάζεται ένα παράδειγμα παράλληλης εκτέλεσης στα Intel TBBs, όπου φαίνεται ο τρόπος αναδρομικής διάσπασης του χώρου επαναλήψεων ενός for-loop και η κλοπή των αντίστοιχων εργασιών που προκύπτουν.

2 Σκοπός

Σκοπός των προτεινόμενων εργασιών είναι η μελέτη, αξιολόγηση και επέκταση του runtime συστήματος των Threading Building Blocks, όσον αφορά τους μηχανισμούς κλοπής εργασιών και αναδρομικής διάσπασης. Η υλοποίηση της εργασίας περιλαμβάνει:

- Μελέτη σχετικής βιβλιογραφίας και του runtime συστήματος των Intel TBBs

```
parallel_for ( blocked_range<int>(0,N,GrainSize), Body() )
```



Σχήμα 1: Παράδειγμα εκτέλεσης ενός παράλληλου for-loop στα Threading Building Blocks. Ο χώρος επαναλήψεων διασπάται αναδρομικά σε μικρά κομμάτια μέχρι ενός ελάχιστου ορίου (GrainSize). Τα τμήματα που απομένουν από κάθε διάσπαση διατίθενται για κλοπή από τους άργουους επεξεργαστές, και ανάλογα με το μέγεθός τους είτε εκτελούνται άμεσα είτε διασπώνται περαιτέρω.

- Μετρήσεις επίδοσης και καταγραφή προφίλ λειτουργίας των μηχανισμών
- Βελτιστοποίηση/τροποποίηση των μηχανισμών με στόχο την ικανοποίηση διαφορετικών κριτηρίων (π.χ. καλύτερη κατανομή φορτίου εργασίας, καλύτερη εκμετάλλευση της τοπολογίας της ιεραρχίας μνήμης, μείωση κατανάλωσης ενέργειας, κ.λπ.)
- Τελικές μετρήσεις και εξαγωγή συμπερασμάτων

Ενδεικτικά θέματα για το Ακ. Έτος 2011-2012:

1. “Αξιολόγηση και βελτίωση του μηχανισμού κλοπής εργασιών που υλοποιείται στη βιβλιοθήκη Intel TBB”.
2. “Αξιολόγηση και βελτίωση του μηχανισμού αναδρομικής διάσπασης που υλοποιείται στη βιβλιοθήκη Intel TBB”.

3 Προαπαιτούμενες γνώσεις και γνώση που θα αποκτηθεί

- Καλή γνώση C/C++
- Καλή γνώση αρχιτεκτονικής υπολογιστών

Σχετικά μαθήματα: Συστήματα Παράλληλης Επεξεργασίας, Αρχιτεκτονική Υπολογιστών, Προηγμένα Θέματα Αρχιτεκτονικής Υπολογιστών.

Κατά τη διάρκεια της διπλωματικής εργασίας θα αποκτηθεί γνώση στον παράλληλο προγραμματισμό, και ειδικότερα στους μηχανισμούς λειτουργίας των σύγχρονων παράλληλων runtime συστημάτων και τον τρόπο που αυτά αλληλεπιδρούν με την υποκείμενη αρχιτεκτονική.

4 Επικοινωνία

- Νεκτάριος Κοζύρης, Αναπ. Καθηγητής (nkoziris@cslab.ece.ntua.gr)
- Νίκος Αναστόπουλος, Μεταδιδ. ερευνητής (anastop@cslab.ece.ntua.gr)
- Κωστής Νίκας, Μεταδιδ. ερευνητής (knikas@cslab.ece.ntua.gr)

Βιβλιογραφία

- [1] Intel threading building blocks. threadingbuildingblocks.org, 2011.
- [2] Intel threading building blocks documentation. threadingbuildingblocks.org/documentation.php, 2011.
- [3] Abhishek Bhattacharjee, Gilberto Contreras, and Margaret Martonosi. Parallelization libraries: Characterizing and reducing overheads. *ACM Trans. Archit. Code Optim.*, 8:5:1–5:29, February 2011.
- [4] Alexey Kukanov and Michael Voss. The foundations for scalable multi-core software in intel threading building blocks. *Intel Technology Journal*, Nov 2007.
- [5] Arch Robison, Michael Voss, and Alexey Kukanov. Optimization via reflection on work stealing in tbb. *Parallel and Distributed Processing Symposium, International*, 0:1–8, 2008.
- [6] Alexandros Tzannes, George C. Caragea, Rajeev Barua, and Uzi Vishkin. Lazy binary-splitting: a run-time adaptive work-stealing scheduler. In *Proceedings of the 15th ACM SIGPLAN symposium on Principles and practice of parallel programming*, PPOPP '10, pages 179–190, New York, NY, USA, 2010. ACM.