ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ

ΤΜΗΜΑ ΕΠΙΣΤΗΜΗΣ ΥΠΟΛΟΓΙΣΤΩΝ

ΠΑΡΟΥΣΙΑΣΗ / ΕΞΕΤΑΣΗ ΜΕΤΑΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

Ντεγιάννης Δημήτριος

Μεταπτυχιακός Φοιτητής

Τμήμα Επιστήμης Υπολογιστών, Πανεπιστήμιο Κρήτης

Επόπτης Μεταπτ. Εργασίας: Καθηγητής Ε. Μαρκάτος

Δευτέρα, 20/2/2017, 17:00

Αίθουσα Τηλεδιάσκεψης Κ206, Τμήμα Επιστήμης Υπολογιστών, Πανεπιστήμιο Κρήτης

" Μια παράλληλη Μηχανή Αναζήτησης Κανονικών Εκφράσεων και Αλφαριθμητικών για Ευρέως Διαθέσιμο Υπολογιστικό Υλικό"

Το ταίριασμα αλφαριθμητικών προτύπων (string pattern matching) είναι ένα πεδίο έρευνας στο οποίο έχει αφιερώσει σημαντικό ποσοστό έρευνας η επιστημονική κοινότητα ανά τα χρόνια. Ήδη, από το 1970, επιστήμονες από διάφορους φορείς και ερευνητικά πεδία, προσπαθούν συνεχώς να αναπτύξουν αλγορίθμους, τόσο έξυπνους όσο και αποδοτικούς. Ακόμα όμως, το πρόβλημα της αντιστοίχισης αλφαριθμητικών προτύπων αποτελεί ανοιχτό πεδίο σκέψης και μελέτης. Ο λόγος της ραγδαίας αυτής δημοτικότητας της αντιστοίχισης αλφαριθμητικών προτύπων στην επιστημονική κοινότητα, είναι η ευρεία χρήση και εφαρμογή της σε πολλές και ποικίλες περιοχές, όπως για παράδειγμα στην πληροφορική, στη βιοπληροφορική, στην υπολογιστική βιοϊατρική και άλλες.

Πρόσφατα, καθώς η τεχνολογία συνεχώς εξελίσσεται, η χρήση των παράλληλων επεξεργαστών έχει αποτελέσει σημαντικό παράγοντα για την ανάπτυξη όλο και πιο

γρήγορων και αποδοτικών συστημάτων. Ο προγραμματισμός αυτών των παράλληλων επεξεργαστών –είτε αυτοί είναι πολυπύρηνοι επεξεργαστές (CPUs) είτε είναι επεξεργαστές γραφικών γενικού σκοπού (GPGPUs)– βασίζεται σε πλατφόρμες που επιτρέπουν στο χρήστη την εποπτεία και τον προγραμματισμό τους. Σε αυτή τη δουλειά, οι πλατφόρμες που χρησιμοποιούνται ονομάζονται CUDA και OpenCL. Συγκεκριμένα, η CUDA απευθύνεται σε επεξεργαστές γραφικών γενικού σκοπού της εταιρίας NVIDIA, σε αντίθεση με την OpenCL, η οποία επιτρέπει τον προγραμματισμό οποιουδήποτε είδους επεξεργαστή.

Σε αυτή τη δουλειά, παρουσιάζουμε μία βιβλιοθήκη για αντιστοίχιση αλφαριθμητικών προτύπων που μέσω μιας αφηρημένης προγραμματιστικής διεπαφής, επιτρέπει την χρήση της σε κάθε είδους πολυπύρηνο επεξεργαστή. Πέρα από την αντιστοίχιση απλών αλφαριθμητικών προτύπων, η βιβλιοθήκη αυτή επιτρέπει τον εντοπισμό και το ταίριασμα προτύπων που προκύπτουν από κανονικές γραμματικές. Για αυτόν το σκοπό, αναπτύξαμε μία μηχανή παράλληλης αναζήτησης αλφαριθμητικών και κανονικών εκφράσεων με την χρήση πολυπύρηνων επεξεργαστών και καρτών γραφικών. Επιπλέον, η μηχανή μπορεί να πετύχει ταυτόχρονη αναζήτηση πολλών αλφαριθμητικών και κανονικών εκφράσεων σε είσοδο πολλαπλών δεδομένων με μία μόνο προσπέλαση αυτών.

Τέλος, η αξιολόγηση της απόδοσης του συστήματος αυτού, μέσω της βιβλιοθήκης που παρέχουμε, έδειξε ότι μπορεί να επιτύχει μέχρι και 21 φορές μεγαλύτερη απόδοση στην αναζήτηση απλών αλφαριθμητικών, καθώς και μέχρι 15 φορές μεγαλύτερη απόδοση στην αναζήτηση κανονικών εκφράσεων, σε σχέση με τις αντίστοιχες εκδόσεις των αλγορίθμων για κεντρικούς επεξεργαστές. Συγκεκριμένα, το σύστημά μας μπορεί να επιτύχει απόδοση έως και 65Gbits/s στην αναζήτηση αλφαριθμητικών και έως 50Gbits/s στην αναζήτηση κανονικών εκφράσεων.

Deyiannis Dimitrios

M.Sc. Thesis

Computer Science Department

University of Crete

Master's Thesis Supervisor: Professor E. Markatos

Monday, 20/2/2017, 17:00
Room K206, Computer Science dept., University of Crete

“A Massively Parallel Regular Expression and String Matching Engine for Commodity Hardware"

ABSTRACT

String pattern matching is one of the most studied fields in the research community, mainly due to the fact that it can be used and applied in various and diverse fields, such as computer science, computational biology, chemistry and others. Since 1970, researchers aim to develop algorithms for efficient string searching and until today, the problem of pattern matching remains a popular area for studying.

Recently, in order to cope with the ever advancing technology, parallel computing platforms –such as CUDA and OpenCL– offer general purpose programming using commodity CPUs, hardware accelerators and GPUs.

In this work, we propose a framework for string pattern matching on parallel hardware architectures. Using CUDA and OpenCL, our framework offers uniform execution on any processor available in a system. The framework provides an abstraction layer to the user –without penalizing the performance– and it is provided as either a C- or Java-like API. Except for simple string matching, our engine supports the use of multiple regular expressions that comply with the POSIX ERE standard. Specifically, we achieve the simultaneous matching of multiple simple strings and binary patterns against multiple data streams as input. Finally, the framework manages to simultaneously match large sets of regular expressions against multiple data streams.

The performance evaluation shows that our massively parallel engine can achieve up to 21 times performance increase when processing simple strings and up to 15 times when processing regular expressions, compared to the CPU versions of both matching algorithms. Specifically, the engine can sustain simple string matching throughput up to 65 Gbits/s and regular expression matching throughput up to 50 Gbits/s.