ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ

ΤΜΗΜΑ ΕΠΙΣΤΗΜΗΣ ΥΠΟΛΟΓΙΣΤΩΝ

ΠΑΡΟΥΣΙΑΣΗ / ΕΞΕΤΑΣΗ ΜΕΤΑΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

Τσατσαράκης Μύρων

Μεταπτυχιακός Φοιτητής

Τμήμα Επιστήμης Υπολογιστών, Πανεπιστήμιο Κρήτης

Επόπτης Μεταπτυχιακής Εργασίας: Αναπλ. Καθηγητής, Π. Πρατικάκης

Μ. Δευτέρα, 18 Απριλίου 2022, ώρα 14:00 μ.μ.

Join Zoom Meeting
https://us02web.zoom.us/j/85426410315

" Σχεδιαστική βελτιστοποίηση εφαρμογής στατικών κτηριακών μελετών μέσω μεθόδων υπομνηματισμού και κατανεμημένου υπολογισμού "

ΠΕΡΙΛΗΨΗ

Η απόδοση μια εφαρμογής είναι σημαντικός παράγοντας για τον τελικό χρήστη. Καθυστερήσεις, εντολές χωρίς ανταπόκριση και μεγάλες περίοδοι αναμονής υποβαθμίζουν την εμπειρία χρήσης μιας εφαρμογής. Μερικοί από τους λόγους που εμποδίζουν τους μηχανικούς λογισμικού από το να βελτιώσουνε την απόδοση των εφαρμογών είναι η ύπαρξη παλαιωμένου κώδικα και η ανεπαρκής εκμετάλλευση πόρων υλισμικού.

Ο παλαιωμένος κώδικας αποτελεί εμπόδιο στην επεκτασιμότητα των δυνατοτήτων μιας εφαρμογής και στην αναδόμηση του κώδικά της με σκοπό τη βελτιστοποίηση. Η έλλειψη τεκμηρίωσής του εμποδίζει συζητήσεις σχετικά με τη λειτουργία και τις παρενέργειες του στο σύστημα. Από την άλλη, πόροι υλισμικού όπως οι επεξεργαστικοί πυρήνες μπορούν να εκμεταλλευτούν κατάλληλα ώστε να μειωθεί ο χρόνος των υπολογιστικά ακριβών λειτουργιών, μέσω μεθόδων κατανεμημένου υπολογισμού.

Εφαρμόζουμε τη δουλειά μας σε μία εμπορική εφαρμογή στατικών κτηριακών μελετών η οποία πάσχει από αυτά τα προβλήματα. Περιέχει μία μεγάλη βάση κώδικα με αρκετές παλαιωμένες δομικές ενότητες. Επιπλέον, η επαρκής εκμετάλλευση των πόρων υλισμικού δεν ήταν δυνατή κατά τη συγγραφή της εφαρμογής, καθώς η τεχνολογία σε εργαλεία κατανεμημένου υπολογισμού δεν ήταν αρκετά αναπτυγμένη.

Στο πρώτο μέρος της εργασίας αυτής παρουσιάζουμε ένα τρόπο υπομνηματισμού τιμών υπολογιστικά ακριβών και παλαιωμένων λειτουργιών, σχεδιάζοντας ένα επεκτάσιμο μοντέλο υπομνηματισμού. Ανακαλύπτουμε τις υπολογιστικά ακριβές λειτουργίες αναλύοντας την απόδοση της εφαρμογής και εφαρμόζουμε το μοντέλο υπομνηματισμού σε παλαιωμένο κώδικα με μη παρεμβατικό τρόπο. Παρουσιάζουμε τις απαιτήσεις, τη σχεδίαση και την

υλοποίηση του μοντέλου υπομνηματισμού σε γλώσσα C++. Η αξιολόγηση της απόδοσης της δουλειάς μας δείχνει πως η εφαρμογή του μοντέλου υπομνηματισμού προσφέρει μέχρι και 100% αύξηση της απόδοσης. Η εφαρμογή του μοντέλου έχει προγραμματιστεί για εμπορική χρήση μέσω της ένταξής του σε μελλοντική αναβάθμισή της εφαρμογής.

Στο δεύτερο μέρος της εργασίας αυτής παρουσιάζουμε ένα τρόπο κατανεμημένου υπολογισμού υπολογιστικά ακριβών ιδιοτήτων κατασκευαστικών στοιχείων. Επεκτείνουμε τη σχεδίαση του στοιχείου διαχείρισης υπολογισμού των ιδιοτήτων αυτών εφαρμόζοντας μεθόδους κατανεμημένου υπολογισμού. Χρησιμοποιούμε τις έννοιές ασύγχρονων διεργασιών και υλοποιούμε μια ομάδα νημάτων ως βοηθητικά εργαλεία της σχεδίασής μας. Παρουσιάζουμε τις απαιτήσεις, τη σχεδίαση και την υλοποίηση του κατανεμημένου στοιχείου διαχείρισης σε γλώσσα C++. Η αξιολόγηση της απόδοσης της δουλειάς μας δείχνει πως το κατανεμημένο στοιχείο διαχείρισης προσφέρει μέχρι και 420% αύξηση της απόδοσης. Το κατανεμημένο στοιχείο διαχείρισης έχει προγραμματιστεί για εμπορική χρήση μέσω της ένταξής του σε μελλοντική αναβάθμισή της εφαρμογής.

University of Crete

Computer Science Department

M.Sc. Thesis

Tsatsarakis Miron

Master's Thesis Supervisor: Associate Professor, P. Pratikakis

Monday, 18 April 2022, 14:00p.m.

Join Zoom Meeting
https://us02web.zoom.us/j/85426410315

“Design level software optimization of structural analysis tool through memoization and application of concurrent computing methods”

ABSTRACT

Performance is an important concern for the end user of an application. Latency, unresponsiveness of commands and long wait times can make or break the experience. Some of the factors that prevent software engineers from achieving their performance goals are the presence of legacy code and the inability to exploit all the available hardware resources.

Legacy code acts as a barrier that prevents both feature extensibility and optimization-oriented refactoring. It is considered to be a no-man's-land, preventing any fruitful discussion about its detailed functionality and side effects in performance. On the other hand, hardware resources, such as CPU cores, can be exploited to reduce the time cost of computationally expensive operations by the application of concurrency. Data-oriented design for concurrency can provide an increase in performance by distributing workload among cores, fully utilizing the available hardware.

Our work is applied on a commercial structural design and analysis application which suffers from the aforementioned setbacks. It comes with a large code-base, guaranteeing a sizeable number of legacy modules. Furthermore, efficient hardware utilization, was not possible during the early development times of the application since the technological research regarding concurrent computation was not sufficiently developed.

In the first part of this thesis we present a way to memoize the return values of computationally expensive legacy code operations by designing an extensible Memoization Model. We identify the computationally expensive operations by conducting performance analysis using an external profiling tool and apply our Memoization Model to the legacy code base in a non-intrusive way. We present the requirements, design and implementations of our Memoization Model using C++. Our evaluation indicates that the application of our Memoization Model, yields up to 100% increase in performance and has been scheduled for a commercial release, in a future update of the application.

In the second part of this thesis we present a way to concurrently calculate computationally expensive properties of structural elements. We extend the design of the manager component, responsible for calculating these expensive properties, by applying concurrency semantics. We use the concept of asynchronous tasks and our own implementation of a thread pool to aid our design. We present the requirements, design and implementation of our concurrent manager component using C++. Our evaluation indicates that our concurrent manager component, yields up to 420% increase in performance and has been scheduled for a commercial release, in a future update of the application.