



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΕΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

ΕΡΓΑΣΤΗΡΙΟ ΥΠΟΛΟΓΙΣΤΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

<http://www.cs-lab.ece.ntua.gr>

Διπλωματικές Εργασίες

Ακαδημαϊκό έτος 2018-19

1 Μελέτη και προγραμματιστική υλοποίηση αλγορίθμων χρονοδρομολόγησης σε υπάρχων χρονοδρομολογητή εργασιών

Τα υπολογιστικά συστήματα μεγάλης κλίμακας και συγκεκριμένα τα υπολογιστικά συστήματα υψηλής επίδοσης (High Performance Computing clusters) είναι ευρέως διαδεδομένα και συχνά χρησιμοποιούνται για την επίλυση πολύπλοκων προβλημάτων από ποικίλες ερευνητικές περιοχές όπως η πρόγνωση και η μοντελοποίηση των καιρικών φαινομένων καθώς και η διερεύνηση της ακολουθίας του ανθρώπινου γονιδιώματος. Όσο εξαπλώνεται η εφαρμογή τους, η κατάλληλη χρονοδρομολόγηση των εργασιών [1] σ' αυτά, αποτελεί καθοριστικό παράγοντα για την αποτελεσματική χρήση αλλά και την εξοικονόμηση στην κατανάλωση ενέργειας. Κύριο λογισμικό που συνθέτει ένα τέτοιο υπολογιστικό σύστημα, είναι κι ο διαχειριστής πόρων (resource manager), ο οποίος περιλαμβάνει έναν χρονοδρομολογητή εργασιών (job scheduler). Ο resource manager αναλαμβάνει να διαμοιράσει τους υπολογιστικούς πόρους στις αντίστοιχες εργασίες. Ο job scheduler επικοινωνεί με τον resource manager προκειμένου να πληροφορηθεί για τις ουρές (queues), τα φορτία των υπολογιστικών κόμβων (nodes) και την διαθεσιμότητα των πόρων, ώστε να πάρει αποφάσεις για τη χρονοδρομολόγηση εργασιών. Σχεδόν όλοι οι resource managers διαθέτουν έναν ενσωματωμένο job scheduler, αλλά οι περισσότεροι διαχειριστές συστημάτων τον αντικαθιστούν συνήθως με έναν εξωτερικό ώστε να βελτιώσουν την λειτουργικότητα του συστήματος. Είναι απαραίτητη η δημιουργία προηγμένων χρονοδρομολογητών, που θα βοηθήσουν στην καλύτερη αξιοποίηση των πόρων (resource utilization) και την ποιότητα των παρεχόμενων υπηρεσιών (Quality of Service). Σκοπός της παρούσας διπλωματικής εργασίας είναι η κατανόηση του τρόπου λειτουργίας του ενσωματωμένου job scheduler του resource manager Slurm [2] ή του εξωτερικού job scheduler Maui [3] του resource manager Torque [4], η μελέτη υπαρχόντων (αλλά μη υλοποιημένων) αλγορίθμων χρονοδρομολόγησης και η ενσωμάτωσή τους προγραμματιστικά (γλώσσα C) στο job scheduler του Slurm [5] ή στο Maui [6] job scheduler.

Σχετικά Μαθήματα: Συστήματα Παράλληλης Επεξεργασίας

Σχετική Βιβλιογραφία:

1. http://dspace.lib.ntua.gr/bitstream/handle/123456789/38155/glykantzisiv_hpc.pdf
2. https://en.wikipedia.org/wiki/Slurm_Workload_Manager
3. https://en.wikipedia.org/wiki/Maui_Cluster_Scheduler
4. <https://en.wikipedia.org/wiki/TORQUE>
5. <https://github.com/SchedMD/slurm>
6. <https://github.com/LabAdvComp/maui>

Επικοινωνία: Νικόλαος Τριανταφύλλης, ntriantafyl@cslab.ece.ntua.gr
Γεώργιος Γκούμας, goumas@cslab.ece.ntua.gr, 210-772-2402

2 Ανάλυση της επίδοσης εφαρμογών σε αρχιτεκτονικές με ανομοιομορφη πρόσβαση μνήμης (Non Uniform Memory Access - NUMA) και υλοποίηση αποτελεσματικής κατανομής και χρονοδρομολόγησης

Ένας από τους τρόπους αύξησης της κλιμακωσιμότητας των υπολογιστικών συστημάτων που έχουν πολλαπλούς επεξεργαστές στο ίδιο σύστημα είναι μέσω των Αρχιτεκτονικών με Ανομοιομορφη Πρόσβαση Μνήμης ή αλλιώς Non Uniform Memory Access (NUMA) systems. Η NUMA αρχιτεκτονική αποτελεί έναν σχεδιασμό συστήματος μνήμης πολυεπεξεργαστικών υπολογιστικών συστημάτων, στα οποία ο χρόνος πρόσβασης της κύριας μνήμης (RAM) εξαρτάται από την απόσταση της θέσης μνήμης που προσπελάει ο επεξεργαστής. Σε μία NUMA αρχιτεκτονική, ένας επεξεργαστής έχει γρηγορότερη πρόσβαση σε μία τοπική μνήμη από ότι σε μία απομακρυσμένη, η οποία όμως είναι τοπική για κάποιον άλλον επεξεργαστή. Στόχος της παρούσας διπλωματικής είναι η μελέτη και η ανάλυση της επίδοσης που προκαλεί η NUMA αρχιτεκτονική κατά την εκτέλεση σύγχρονων εφαρμογών, καθώς επίσης και η αναγνώριση προτύπων και συμπεριφορών και η κατηγοριοποίηση των εφαρμογών σε ευαίσθητες και μη-ευαίσθητες, ως προς την συμπεριφορά τους λόγω της NUMA αρχιτεκτονικής. Επιπλέον, θα υλοποιηθεί εργαλείο και πολιτικές αποτελεσματικής χρονοδρομολόγησης πολλαπλών εφαρμογών, τόσο μέσω της κατάλληλης κατανομής των κόμβων μνήμης (memory node allocation) όσο και μέσω της κατάλληλης επιλογής των υπολογιστικών κόμβων (compute node allocation).

Σχετικά Μαθήματα: Προηγμένα Θέματα Αρχιτεκτονικής Υπολογιστών, Λειτουργικά Συστήματα, Εργαστήριο Λειτουργικών Συστημάτων

Επικοινωνία: Βασίλης Καρακώστας, vkarakos@cslab.ece.ntua.gr, 210-772-4133
Νικέλα Παπαδοπούλου, nikela@cslab.ece.ntua.gr, 210-772-2495
Κωστής Νίκας, knikas@cslab.ece.ntua.gr, 210-772-4159

3 Σχεδιασμός και υλοποίηση δυναμικών μηχανισμών διαμοιρασμού κρυφής μνήμης και εύρους ζώνης μνήμης σε σύγχρονα πολυπύρρηνα υπολογιστικά συστήματα (cache and bandwidth partitioning)

Η κρυφή μνήμη τελευταίου επιπέδου (last-level cache) και το διαθέσιμο εύρος ζώνης πρόσβασης στην μνήμη (memory bandwidth) αποτελούν δύο πολύ σημαντικούς πόρους του υπολογιστικού συστήματος που επηρεάζουν την επίδοση των εφαρμογών. Επιπλέον, αυτοί οι πόροι είναι κοινοί και διαμοιραζόμενοι για όλες τις διεργασίες που εκτελούνται στο σύστημα. Αυτό σημαίνει ότι οι συνεκτελούμενες διεργασίες ανταγωνίζονται για πρόσβαση στους πόρους αυτούς, με αποτέλεσμα συχνά μια διεργασία να επηρεάζει την απόδοση άλλων. Οι πλέον σύγχρονοι επεξεργαστές παρέχουν υποστήριξη σε επίπεδο υλικού (hardware) και λογισμικού (runtime) για την επίβλεψη της χρησιμοποίησης (monitoring) των πόρων αυτών. Επιπλέον, παρέχουν υποστήριξη για τον διαμοιρασμό των πόρων αυτών (partitioning). Στόχος της παρούσας διπλωματικής είναι η μελέτη, ο σχεδιασμός και η υλοποίηση μηχανισμών και πολιτικών διαμοιρασμού της κρυφής μνήμης και του εύρους ζώνης μνήμης σε επίπεδο λογισμικού (runtime), ώστε να ελαχιστοποιηθεί η επίπτωση του ανταγωνισμού για αυτούς τους κοινούς πόρους, έχοντας ως στόχο (i) την αύξηση του συνολικού throughput του συστήματος, ή (ii) την διατήρηση υψηλής επίδοσης για εφαρμογές υψηλής προτεραιότητας (quality of service) ενώ βελτιώνεται ταυτόχρονα το συνολικό throughput του συστήματος.

Σχετικά Μαθήματα: Προηγμένα Θέματα Αρχιτεκτονικής Υπολογιστών, Λειτουργικά Συστήματα, Εργαστήριο Λειτουργικών Συστημάτων

Επικοινωνία: Κωστής Νίκας, knikas@cslab.ece.ntua.gr, 210-772-4159

Νικέλα Παπαδοπούλου, nikela@cslab.ece.ntua.gr, 210-772-2495

Βασίλης Καρακώστας, vkarakos@cslab.ece.ntua.gr, 210-772-4133

4 Μελέτη και αξιολόγηση των τεχνολογιών διαμοιρασμού κρυφής μνήμης και εύρους ζώνης μνήμης (cache and bandwidth partitioning) σε σύγχρονα πολυπύρρηνα υπολογιστικά συστήματα για τη βελτίωση της επίδοσης πολυνηματικών εφαρμογών

Το διαθέσιμο εύρος ζώνης μνήμης και το τελευταίο επίπεδο κρυφής μνήμης είναι διαμοιραζόμενοι πόροι του συστήματος ανάμεσα στις συνεκτελούμενες εφαρμογές. Όταν πολλές διαφορετικές εφαρμογές εκτελούνται ταυτόχρονα σε ένα σύστημα, αυτές ανταγωνίζονται μεταξύ τους για την πρόσβαση σε κοινούς πόρους και συγκεκριμένα στο περιορισμένο εύρος ζώνης μνήμης (memory bandwidth) και στο τελευταίο επίπεδο κρυφής μνήμης (last level cache). Αυτός ο ανταγωνισμός καθώς και η κακή διαχείριση του περιορισμένου εύρους ζώνης μνήμης και της κρυφής μνήμης, έχει ως αποτέλεσμα οι εφαρμογές να επηρεάζουν η μία την άλλη και να υποβαθμίζεται σημαντικά η συνολική απόδοση του συστήματος αλλά και των ίδιων των εφαρμογών (μία εφαρμογή μπορεί να επηρεάζει την απόδοση μιας άλλης). Παρόμοιο φαινόμενο παρατηρείται ανάμεσα σε παράλληλα νήματα μιας πολυνηματικής εφαρμογής. Τα παράλληλα νήματα μιας παράλληλης εφαρμογής ανταγωνίζονται για αυτούς τους διαμοιραζόμενους πόρους επηρεάζοντας την συνολική απόδοση της παράλληλης εφαρμογής. Πρόσφατα, η διαχείριση του εύρους ζώνης μνήμης (bandwidth) και του τελευταίου επιπέδου της κρυφής μνήμης, δηλαδή η παρακολούθηση της χρήσης τους (monitoring) και ο διαμοιρισμός τους (partitioning) στις εφαρμογές του συστήματος έγινε εφικτός μέσω των τεχνολογιών Memory Bandwidth Monitoring (MBM),

Memory Bandwidth Allocation (MBA) και Cache Monitoring Technology (CMT) και Cache Allocation Technology, αντίστοιχα, που παρέχονται από την Intel RDT. Οι τεχνολογίες MBA και CMT παρέχουν ένα προσεγγιστικό και έμμεσο έλεγχο επιπέδου πυρήνα στο εύρος ζώνης μνήμης και στο τελευταίο επίπεδο της κρυφής μνήμης, αντίστοιχα, που διατίθενται στους πυρήνες της CPU. Στα πλαίσια αυτής της διπλωματικής θα μελετήσουμε τις τεχνολογίες MBA και CAT, τις πολιτικές που διατίθενται και τα χαρακτηριστικά αυτών των τεχνολογιών σε πολυνηματικές εφαρμογές και θα τις αξιοποιήσουμε για τη βελτίωση της επίδοσής τους.

Σχετικά Μαθήματα: Προηγμένα Θέματα Αρχιτεκτονικής Υπολογιστών

Επικοινωνία: Αθηνά Ελαφρού, athena@cslab.ece.ntua.gr, 210-772-2133

Χριστίνα Γιαννούλα, cgiannoula@cslab.ece.ntua.gr,

Κωστής Νίκας, knikas@cslab.ece.ntua.gr, 210-772-4159

5 Μελέτη, σχεδιασμός, και υλοποίηση επεκτάσεων των μηχανισμών εικονικής μνήμης

Η εικονική μνήμη αποτελεί θεμελιώδη “αφαίρεση” ενός υπολογιστικού συστήματος. Κάθε διεργασία χρησιμοποιεί το δικό της συνεχή χώρο εικονικών διευθύνσεων για να προσπελάσει τη μνήμη, ενώ το λειτουργικό σύστημα είναι υπεύθυνο για να αντιστοιχίσει τις εικονικές διευθύνσεις σε φυσικές διευθύνσεις μηχανήματος και να αποθηκεύσει/διατηρήσει/διαχειριστεί αυτές τις αντιστοιχίσεις σε πίνακες σελίδων (page tables) για κάθε διεργασία. Για να εξυπηρετήσει μια εντολή προσπέλασης μνήμης ο επεξεργαστής πρέπει να μεταφράσει την εικονική διεύθυνση (virtual address) σε φυσική (physical address). Για λόγους επίδοσης, οι επεξεργαστές χρησιμοποιούν εξειδικευμένες κρυφές μνήμες (Translation Lookaside Buffers, TLBs) για την αποθήκευση των πιο πρόσφατων μεταφράσεων. Επιπλέον, η εικονικοποίηση (virtualization) είναι μια τεχνική που επιτρέπει την ταυτόχρονη εκτέλεση πολλών εικονικών μηχανών (virtual machines ή guest VMs) σε ένα φυσικό μηχάνημα (host). Κάθε guest τρέχει το δικό του λειτουργικό σύστημα, έχοντας τη ψευδαίσθηση ότι έχει πλήρη και αποκλειστική πρόσβαση σε εικονικούς πόρους. Ειδικό λογισμικό (Virtual Machine Monitor) που τρέχει στο host είναι υπεύθυνο για την ασφαλή πρόσβαση του VM στους πραγματικούς φυσικούς πόρους του συστήματος (π.χ μνήμη).

Στο πλαίσιο των μηχανισμών εικονικής μνήμης, θα δοθούν τρεις διπλωματικές εργασίες:

- **Αποδοτικές πολιτικές διαχείρισης μεγάλων virtual-to-physical mappings**

Ένας τρόπος μείωσης του κόστους της εικονικής μνήμης είναι η αύξηση του TLB reach, δηλαδή του εύρους των εικονικών διευθύνσεων που το TLB μπορεί να μεταφράσει. Ένας τρόπος αύξησης του TLB reach είναι η χρήση μεγάλων virtual-to-physical mappings. Ωστόσο η δημιουργία και η διαχείριση μεγάλων mappings, επιφέρει δυσκολίες/προκλήσεις στο memory management (π.χ fragmentation, memory bloat, unfairness, page sharing). Σκοπός της παρούσας εργασίας είναι η μελέτη αυτών των προβλημάτων και η διερεύνηση πολιτικών διαχείρισης, αρχικά με χρήση των huge pages και στη συνέχεια με χρήση των range translations.

- **Μεγάλες σελίδες και σύστημα αρχείων (page cache)**

Σκοπός αυτής της διπλωματικής εργασίας είναι η μελέτη και κατανόηση της δέσμευσης μεγάλων σελίδων από το λειτουργικό σύστημα για σελίδες που αποθηκεύουν δεδομένα από το δίσκο. Στα πλαίσια της εργασίας αυτής θα πειραματιστούμε με patches του πυρήνα του λειτουργικού συστήματος Linux που στόχο έχουν να υποστηρίξουν τη συγκεκριμένη περίπτωση αλλά δεν έχουν ακόμα ενσωματωθεί στο mainline.

- **Nested Paging vs Shadow Paging**

Σε virtualized περιβάλλοντα υπάρχουν δύο τεχνικές εικονικοποίησης της μνήμης: το shadow paging και τα nested page tables. Η software τεχνική του shadow paging μειώνει το κόστος μετάφρασης αλλά αυξάνει το κόστος διαχείρισης της μνήμης του guest προκαλώντας συχνά το σταμάτημα της λειτουργίας του guest (VM exits). Αντίθετα, η υποβοηθούμενη από το hardware τεχνική του nested paging αυξάνει το κόστος μετάφρασης αλλά ελαχιστοποιεί το κόστος διαχείρισης της μνήμης του guest. Έχει παρατηρηθεί στην ερευνητική βιβλιογραφία ότι ένας πιθανός συνδυασμός των δυο τεχνικών μπορεί να πετύχει το καλύτερο συμβιβασμό πλεονεκτημάτων των δυο τεχνικών. Σκοπός αυτής της εργασίας είναι η μελέτη των δύο τεχνικών, η μελέτη της αλληλεπίδρασης τους με το memory management του guest και του host μηχανήματος και ο πειραματισμός με το συνδυασμό τους.

Σχετικά Μαθήματα: Εργαστήριο Λειτουργικών Συστημάτων, Προηγμένα Θέματα Αρχιτεκτονικής Υπολογιστών

Επικοινωνία: Χλόη Αλβέρτη, xalverti@cslab.ece.ntua.gr, 210-772-2279

Βασίλης Καρακώστας, vkarakos@cslab.ece.ntua.gr, 210-772-4133

6 Σχεδιασμός και υλοποίηση μηχανισμών σε επίπεδο αρχιτεκτονικής για την εκτίμηση της ενεργά χρησιμοποιούμενης μνήμης (working set estimation)

Η έννοια της “ενεργά χρησιμοποιούμενης μνήμης” (ή αλλιώς active working set) ορίστηκε από τον Peter Denning το 1968 ως το σύνολο των δεδομένων που η κάθε διεργασία προσπελαύνει σε ένα συγκεκριμένο χρονικό παράθυρο. Το active working set αποτελεί σημαντική μετρική στην οποία βασίζονται πολιτικές διαμοιρασμού πόρων (π.χ. μνήμης). Ο υπολογισμός του active working set πραγματοποιείται συνήθως στο λογισμικό, χρησιμοποιώντας υπάρχοντες μηχανισμούς του λειτουργικού συστήματος και της αρχιτεκτονικής, όπως για παράδειγμα τα access και dirty bits του πίνακα σελίδων (page table). Στόχος της διπλωματικής αυτής είναι η μελέτη, ο σχεδιασμός και η υλοποίηση νέων μηχανισμών εκτίμησης του working set σε επίπεδο αρχιτεκτονικής/υλικού.

Σχετικά Μαθήματα: Προηγμένα Θέματα Αρχιτεκτονικής Υπολογιστών, Εργαστήριο Λειτουργικών Συστημάτων

Επικοινωνία: Βασίλης Καρακώστας, vkarakos@cslab.ece.ntua.gr, 210-772-4133

Χλόη Αλβέρτη, xalverti@cslab.ece.ntua.gr, 210-772-2279

7 Εφαρμογή αλγορίθμων μηχανικής μάθησης στην αρχιτεκτονική υπολογιστών

Αλγόριθμοι μηχανικής μάθησης ταξινόμησης (classification) και πρόβλεψης (prediction) εφαρμόζονται πλέον κατά κανόνα σε τομείς όπως η όραση υπολογιστών, η επεξεργασία φυσικής γλώσσας κ.α, πετυχαίνοντας εντυπωσιακά αποτελέσματα. Και ενώ συχνά σχεδιάζεται εξειδικευμένο hardware για την επιτάχυνση τους, λίγες είναι προς το παρόν οι περιπτώσεις εφαρμογής/χρήσης τους για τη βελτίωση της ίδιας της επίδοσης ενός υπολογιστικού συστήματος.

Οι σύγχρονες αρχιτεκτονικές συχνά εμπλέκουν ευριστικές μεθόδους, μεθόδους πρόβλεψης/υποθετικής εκτέλεσης για τη μεγιστοποίηση της επίδοσης ενός συστήματος. Παράδειγμα μπορεί να θεωρηθεί η χρήση προανάκλησης (prefetching), που χρησιμοποιείται για την αντιμετώπιση ενός σημαντικού σημείου συμφόρησης (bottleneck) επίδοσης των σύγχρονων αρχιτεκτονικών, του κόστους προσπέλασης της κύριας μνήμης. Σκοπός της συγκεκριμένης διπλωματικής είναι η διερεύνηση της δυνατότητας εφαρμογής αλγορίθμων μηχανικής μάθησης πρόβλεψης ακολουθιών (sequence prediction) κ.α στην αρχιτεκτονική για τη βελτιστοποίηση της επίδοσης του υποσυστήματος μνήμης. Αυτό μπορεί να περιλαμβάνει βελτιστοποιήσεις στη χρήση των κρυφών μνημών (caches-prefetching), στο μηχανισμό ει-κονικής μνήμης (TLBs) κ.α.

Στόχος είναι αρχικά να χρησιμοποιηθεί λογισμικό μηχανικής μάθησης (π.χ pytorch) με πραγματικά δεδομένα από σύγχρονα μηχανήματα για τη μελέτη διαφορετικών μοντέλων (π.χ LSTMs). Στη συνέχεια, ανάλογα με τα συμπεράσματα του πρώτου βήματος, θα επιχειρήσουμε να αξιολογήσουμε τη δυνατότητα εφαρμογής τους σε επίπεδο μικροαρχιτεκτονικής λαμβάνοντας υπόψη την πολυπλοκότητα, το χρόνο απόκρισης και την κατανάλωση χώρου και ενέργειας.

Σχετικά Μαθήματα: Προηγμένα Θέματα Αρχιτεκτονικής Υπολογιστών

Σχετική Βιβλιογραφία:

1. “Learning Memory Access Patterns”, <https://arxiv.org/pdf/1803.02329.pdf>

Επικοινωνία: Χλόη Αλβέρτη, xalverti@cslab.ece.ntua.gr, 210-772-2279

Βασίλης Καρακώστας, vkarakos@cslab.ece.ntua.gr, 210-772-4133

Κωστής Νίκας, knikas@cslab.ece.ntua.gr, 210-772-4159

8 Βελτιστοποίηση της σουίτας Graph500 σε αρχιτεκτονικές NUMA

Η σουίτα προγραμμάτων Graph500 (www.graph500.com) δημιουργήθηκε για την αξιολόγηση της επίδοσης υπολογιστικών συστημάτων στην εκτέλεση εφαρμογών με ένταση στα δεδομένα (data-intensive applications). Σε αυτή την κατηγορία ανήκουν και οι αλγόριθμοι επεξεργασίας δεδομένων οργανωμένων σε γράφους (graph analytics). Η τρέχουσα έκδοση της σουίτας Graph500 υλοποιεί τρεις υπολογιστικούς πυρήνες: ο πρώτος κατασκευάζει έναν μη-κατευθυνόμενο γράφο, ο δεύτερος διασχίζει το γράφο κατά πλάτος (breadth-first search / BFS) και ο τρίτος υπολογίζει όλα τα ελάχιστα μονοπάτια από έναν κόμβο-αφετηρία (single-source shortest path / SSSP). Στην παρούσα διπλωματική, θα μελετήσουμε την επίδοση της βασικής έκδοσης της σουίτας Graph500 σε συστήματα μεγάλης κλίμακας, με κοινή μνήμη, σε αρχιτεκτονική NUMA, και θα βελτιστοποιήσουμε τους τρεις υπολογιστικούς πυρήνες, αξιοποιώντας τεχνικές βελτιστοποίησης που έχουν προταθεί στη βιβλιογραφία, εναλλακτικούς τρόπους αναπαράστασης των γράφων, ταυτόχρονες δομές δεδομένων, διαφορετικά σχήματα συγχρονισμού και επικοινωνίας.

Σχετικά Μαθήματα: Συστήματα Παράλληλης Επεξεργασίας

Επικοινωνία: Νικέλα Παπαδοπούλου, nikela@cslab.ece.ntua.gr, 210-772-2495

Γεώργιος Γκούμας, goumas@cslab.ece.ntua.gr, 210-772-2402

9 Υλοποίηση και μελέτη επίδοσης εφαρμογών υψηλής έντασης σε ετερογενή συστήματα (CPU , GPU, FPGA) με τη χρήση του προγραμματιστικού μοντέλου OpenCL

Την τελευταία δεκαετία η αύξηση της πολυπλοκότητας μιας ευρείας κατηγορίας υπολογιστικών εφαρμογών και η κατάρρευση της κλιμάκωσης της ενεργειακής επίδοσης των επεξεργαστών γενικού σκοπού προκάλεσε μια στροφή προς τα ετερογενή υπολογιστικά συστήματα (heterogeneous computing). Η ενορχηστρωμένη χρήση επεξεργαστών γενικού σκοπού (CPU) και προγραμματιζόμενων επιταχυντών ειδικού σκοπού (HW accelerators) για υπολογιστικές εφαρμογές υψηλής έντασης εφαρμόζεται ήδη σε υπολογιστικά συστήματα κέντρων δεδομένων (data centers), αλλά και σε συστήματα υψηλών επιδόσεων (High Performance Computing -HPC). Στόχος η αύξηση της απόδοσης με όρους επίδοσης (performance) αλλά και κατανάλωσης ισχύος (power efficiency). Οι επαναδιαμορφούμενες αρχιτεκτονικές, γνωστές και με τον αγγλικό όρο FPGAs (Field Programmable Gate Arrays), είναι ένα ισχυρό υπολογιστικό υλικό το οποίο επιτρέπει στους σχεδιαστές τη δημιουργία συστημάτων εξειδικευμένων εφαρμογών (application-specific). Παρά τη μεγάλη ευελιξία και τη συχνή υπεροχή τους σε ενεργειακή κατανάλωση/επίδοση συγκριτικά με τους επεξεργαστές γενικού σκοπού αλλά και με άλλους επιταχυντές (π.χ GPU), ο προγραμματισμός τους με μοντέλα επιπέδου μεταφοράς καταχωρητή (RTL-based programming model) δρούσε ανασταλτικά στην ευρεία χρήση τους. Τα τελευταία χρόνια όμως αναπτύχθηκαν προηγμένα εργαλεία σύνθεσης (HLS) με στόχο την αυτοματοποίηση τη διαδικασίας σχεδίασης, και πρόσφατα άρχισε να υποστηρίζεται/ χρησιμοποιείται το προγραμματιστικό μοντέλο της OpenCL που στοχεύει στον παράλληλο προγραμματισμό ετερογενών συστημάτων (CPU, GPU, FPGA, DSP κ.α) διευκολύνοντας παράλληλα τη φορητότητα μεταξύ διαφορετικών πλατφορμών. Η εξέλιξη αυτή σε συνδυασμό με την εξέλιξη των αρχιτεκτονικών FPGA, καθιστά τις τελευταίες μια ελκυστική τεχνολογία επιταχυντών ευρείας χρήσης. Ανακοινώσεις μεγάλων εταιριών (Intel, IBM, Microsoft) για λύσεις που συνδέουν CPUs και FPGAs είναι στοιχεία που καταδεικνύουν τη στροφή αυτή.

Η παρούσα διπλωματική περιλαμβάνει εξοικείωση με το προγραμματιστικό μοντέλο της OpenCL, υλοποίηση αλγορίθμων υψηλής έντασης σε ετερογενή συστήματα (FPGA και GPU) και εφαρμογή τεχνικών βελτιστοποίησης στηριζόμενων στις ιδιαιτερότητες της κάθε αρχιτεκτονικής. Τέλος περιλαμβάνει τη σύγκριση των διάφορων υλοποιήσεων με όρους επίδοσης και κατανάλωσης ισχύος.

Σχετικά Μαθήματα: Συστήματα Παράλληλης Επεξεργασίας, Ψηφιακά Συστήματα VLSI

Επικοινωνία: Χλόη Αλβέρτη, xalverti@cslab.ece.ntua.gr, 210-772-2279

Νικέλα Παπαδοπούλου, nikela@cslab.ece.ntua.gr, 210-772-2495

Αθηνά Ελαφρού, athena@cslab.ece.ntua.gr, 210-772-2133

Γεώργιος Γκούμας, goumas@cslab.ece.ntua.gr, 210-772-2402

10 Ανάλυση ιστορικών δεδομένων υπερυπολογιστικών συστημάτων

Οι διαχειριστές των σύγχρονων υπερυπολογιστικών συστημάτων έχουν τη δυνατότητα συλλογής σημαντικής πληροφορίας για τη συμπεριφορά των εφαρμογών που εκτελούνται στο σύστημα, τα αιτήματα των χρηστών και την ποιότητα υπηρεσίας που λαμβάνουν, αλλά και τη συνολική κατάσταση του συστήματος. Η συλλογή ιστορικών δεδομένων του συστήματος δίνει δυνατότητες για τη βελτίωση της χρήσης του συστήματος, αφού η επεξεργασία της μπορεί να συμβάλλει στη μείωση του χρόνου αναμονής από την πλευρά των χρηστών, στη βελτίωση των πολιτικών δέσμευσης πόρων και χρονοδρομολόγησης των εργασιών, αλλά και στην ανθεκτικότητα του συστήματος σε σφάλματα. Η πα-

ρούσα διπλωματική περιλαμβάνει την ανάλυση δεδομένων που έχουν συλλεχθεί κατά τη λειτουργία του εθνικού υπερυπολογιστικού συστήματος ARIS (<https://hpc.grnet.gr/>) και την εξόρυξη σχέσεων μεταξύ συμβάντων του συστήματος, αιτημάτων των χρηστών και συμπεριφοράς των εφαρμογών, με στόχο τη βελτίωση της λειτουργίας του συστήματος και της ικανοποίησης των χρηστών, καθώς και την ανάπτυξη μίας μεθοδολογίας συστηματικής συλλογής και επεξεργασίας αντίστοιχων δεδομένων σε υπερυπολογιστικά συστήματα.

Σχετικά Μαθήματα: Συστήματα Παράλληλης Επεξεργασίας, Τεχνητά Νευρωνικά Δίκτυα και Μηχανική Μάθηση

Επικοινωνία: Νικέλα Παπαδοπούλου, nikela@cslab.ece.ntua.gr, 210-772-2495

Κωστής Νίκας, knikas@cslab.ece.ntua.gr, 210-772-4159

11 Τεχνικές πρόβλεψης και προβολής της επίδοσης παράλληλων εφαρμογών σε υπολογιστικά συστήματα μεγάλης κλίμακας

Η μοντελοποίηση και πρόβλεψη της επίδοσης των παράλληλων εφαρμογών που εκτελούνται σε υπερυπολογιστές είναι κρίσιμη για το σχεδιασμό των συστημάτων επόμενης γενιάς. Ένα από τα σημαντικότερα ερωτήματα που καλούνται να απαντήσουν τα διάφορα μοντέλα πρόβλεψης είναι η επίδοση των εφαρμογών σε συστήματα μεγαλύτερης κλίμακας από τα υπάρχοντα, δηλαδή η προβολή της επίδοσης των εφαρμογών (performance extrapolation). Το συγκεκριμένο ερώτημα γίνεται επιτακτικό καθώς βρισκόμαστε στη φάση της μετάβασης από την εποχή των επιδόσεων της τάξης των PetaFLOPS στην εποχή των επιδόσεων της τάξης των ExaFLOPS, που συνοδεύεται από σημαντική αύξηση του πλήθους των πυρήνων και των κόμβων των υπερυπολογιστών. Στη βιβλιογραφία έχουν προταθεί πολλές τεχνικές μοντελοποίησης και πρόβλεψης της επίδοσης, που στηρίζονται είτε σε αναλυτικά μοντέλα είτε σε εμπειρική μοντελοποίηση. Επιπλέον, από τις τεχνικές αυτές, ορισμένες έχουν επίγνωση της αρχιτεκτονικής ή/και της εφαρμογής, ενώ άλλες αντιμετωπίζουν με γενικό τρόπο την αρχιτεκτονική ή τις εφαρμογές. Για την εφαρμογή τους, απαιτούν τη συλλογή κάποιας πληροφορίας, στο χρόνο μεταγλώττισης ή στο χρόνο εκτέλεσης, και χρησιμοποιούν πολλαπλά εργαλεία για την εξαγωγή και τη χρήση τέτοιας πληροφορίας, όπως προσομοιωτές (simulators) και εργαλεία συλλογής ίχνους (tracing tools). Η παρούσα διπλωματική έχει δύο μέρη. Στο πρώτο μέρος, θα μελετηθεί η σχετική βιβλιογραφία και τα υπάρχοντα εργαλεία προσομοίωσης, για την επιλογή της κατάλληλης μεθοδολογίας για την προβολή της επίδοσης σε συστήματα μεγαλύτερης κλίμακας. Στο δεύτερο μέρος, θα αναπτυχθούν μοντέλα πρόβλεψης επίδοσης παράλληλων εφαρμογών, θα ενσωματωθούν σε εργαλεία προσομοίωσης και θα εξεταστούν πειραματικά ως προς την αποτελεσματικότητά τους να προβάλουν την επίδοση εφαρμογών σε συστήματα μεγαλύτερης κλίμακας.

Σχετικά Μαθήματα: Συστήματα Παράλληλης Επεξεργασίας

Επικοινωνία: Νικέλα Παπαδοπούλου, nikela@cslab.ece.ntua.gr, 210-772-2495

Γεώργιος Γκούμας, goumas@cslab.ece.ntua.gr, 210-772-2402

12 Παράλληλες Δομές Δεδομένων

Με την επικράτηση των πολυπύρηνων επεξεργαστών, οι παράλληλες δομές δεδομένων έχουν γίνει πολύ σημαντικό κομμάτι των σύγχρονων πολυνηματικών εφαρμογών και αποτελούν πολύ ενεργό ερευνητικό πεδίο. Στις παράλληλες δομές δεδομένων είναι σημαντικό να συγχρονιστούν τα πολλαπλά

νήματα εκτέλεσης που επιτελούν λειτουργίες πάνω σε αυτές ώστε πέρα από την εξασφάλιση της σωστής λειτουργίας να έχουν και υψηλές επιδόσεις. Οι πιο διαδεδομένοι τρόποι συγχρονισμού είναι τα κλειδώματα (locks) και οι ατομικές εντολές που παρέχονται στους σύγχρονους επεξεργαστές [2]. Τα κλειδώματα και οι ατομικές εντολές μπορούν να εφαρμοστούν σχετικά εύκολα σε απλές δομές δεδομένων όπως συνδεδεμένες λίστες, ουρές κλπ. Ωστόσο σε πιο σύνθετες δομές όπως είναι τα δέντρα οι lock-based και lock-free υλοποιήσεις είναι αρκετά πολύπλοκες. Για αυτό το λόγο υπάρχει έρευνα γύρω από δύο τεχνικές συγχρονισμού που στόχος τους είναι να διευκολύνουν την υλοποίηση σύνθετων παράλληλων δομών δεδομένων. Οι τεχνικές αυτές είναι το Read-Copy-Update [4,5] και το Transactional Memory [6] το οποίο από το 2013 και έπειτα υποστηρίζεται και από το υλικό των επεξεργαστών της Intel [7]. Διπλωματικές σχετικά με παράλληλες δομές δεδομένων θα ασχοληθούν με την ανάλυση και αξιολόγηση των διαφόρων τεχνικών συγχρονισμού σε διαφορετικές δομές δεδομένων.

Σχετικά Μαθήματα: Συστήματα Παράλληλης Επεξεργασίας

Σχετική Βιβλιογραφία:

1. “Asynchronized Concurrency: The Secret to Scaling Concurrent Search Data Structures”,
<https://dl.acm.org/citation.cfm?id=2694359>
2. “Atomic operations”, <https://software.intel.com/en-us/node/506090>
3. “The Art of Multiprocessor Programming”, https://www.researchgate.net/publication/213876653_The_Art_of_Multiprocessor_Programming
4. “Read-Copy-Update”, <https://en.wikipedia.org/wiki/Read-copy-update>
5. “Concurrent updates with RCU: search tree as an example”,
<https://dl.acm.org/citation.cfm?id=2611471>
6. “Transactional Memory”, https://en.wikipedia.org/wiki/Transactional_memory
7. “Transactional Synchronization Extensions”,
https://en.wikipedia.org/wiki/Transactional_Synchronization_Extensions

Επικοινωνία: Δημήτρης Σιακαβάρας, jimsiak@cslab.ece.ntua.gr, 210-772-2495

13 Δυναμική Τροποποίηση της Τοπολογίας NUMA στο Λειτουργικό Σύστημα Linux για Αξιοποίηση σε Εικονικές Μηχανές (VMs)

Στα συστήματα ανομοιομορφης πρόσβασης μνήμης (NUMA), ο χρόνος πρόσβασης στη μνήμη εξαρτάται από την τοποθεσία της μνήμης σε σχέση με τον επεξεργαστή. Ένας επεξεργαστής προσπελάει πολύ γρηγορότερα τα δεδομένα που είναι αποθηκευμένα στην τοπική μνήμη από ότι τα δεδομένα που βρίσκονται σε μία απομακρυσμένη. Το λειτουργικό σύστημα γνωρίζει την τοπολογία NUMA, δηλαδή το χρόνο που απαιτείται από κάθε πυρήνα για την πρόσβαση σε κάθε NUMA island. Ωστόσο, καθώς στα πραγματικά συστήματα αυτή η τοπολογία δεν μεταβάλλεται, δεν υπάρχει και η αντίστοιχη πρόβλεψη στο λειτουργικό σύστημα. Ωστόσο, σε εικονικές μηχανές (Virtual Machines, VMs) η τοπολογία NUMA μπορεί να μεταβάλλεται δυναμικά με το χρόνο καθώς τα virtual cpus των VMs εκτελούνται σε διαφορετικά physical cpus του πραγματικού μηχανήματος. Στόχος αυτής της διπλωματικής είναι η

υλοποίηση και αξιολόγηση ενός μηχανισμού μέσω του οποίου το λειτουργικό σύστημα θα ενημερώνεται για αλλαγές στην τοπολογία NUMA έτσι ώστε να δίνει την ευκαιρία σε εφαρμογές που μπορούν να επωφεληθούν από αυτό να προσαρμοστούν κατάλληλα και να αυξήσουν την επίδοσή τους.

Σχετικά Μαθήματα: Λειτουργικά Συστήματα, Εργαστήριο Λειτουργικών Συστημάτων

Επικοινωνία: Δημήτρης Σιακαβάρας, jimsiak@cslab.ece.ntua.gr, 210-772-2495

14 Μελέτη λειτουργικών συστημάτων για manycore αρχιτεκτονικές επιταχυντών

Την τελευταία δεκαετία παρατηρείται επιβράδυνση στην κλιμάκωση των πολυπύρηνων (multicore) επεξεργαστών γενικού σκοπού για λόγους που σχετίζονται με την ενεργειακή πυκνότητα στα ολοένα και μικρότερου μεγέθους transistors. Μία απάντηση σε αυτήν την τάση είναι η χρησιμοποίηση ετερογενών συστημάτων και πιο συγκεκριμένα επιταχυντών. Τα λειτουργικά συστήματα γενικού σκοπού δεν είναι τα καταλληλότερα για τέτοιους είδους περιβάλλοντα, καθώς με την πληθώρα των λειτουργιών που προσφέρουν, προσθέτουν αχρείαστο θόρυβο κατά την εκτέλεση των εφαρμογών. Σε αυτό το πλαίσιο, έχουν αναπτυχθεί ελαφρύτερα συστήματα για manycore αρχιτεκτονικές με στόχο την καλύτερη κλιμακωσιμότητα των εφαρμογών. Σκοπός αυτής της διπλωματικής είναι η μελέτη λειτουργικών συστημάτων εξειδικευμένων για συσκευές επιταχυντών, η αξιολόγηση κάποιου από αυτά σε συγκεκριμένο επιταχυντή και η πιθανή επέκτασή του με βάση τα αποτελέσματα της προαναφερθείσας αξιολόγησης.

Σχετικά Μαθήματα: Λειτουργικά Συστήματα, Εργαστήριο Λειτουργικών Συστημάτων

Επικοινωνία: Στέφανος Γεράγγελος, sgerag@cslab.ece.ntua.gr

Κωστής Παπαζαφειρόπουλος, krapazaf@cslab.ece.ntua.gr

Στράτος Ψωμαδάκης, psomas@cslab.ece.ntua.gr

15 Επιτάχυνση υλικού για αποδοτική εκτέλεση εφαρμογών ως uni-kernels

Μια ενδιαφέρουσα προσέγγιση στη μείωση του θορύβου του λειτουργικού συστήματος και περιττών εξαρτήσεων στο περιβάλλον εκτέλεσης μιας εφαρμογής είναι η δημιουργία ενός λεπτού στρώματος εξαρτήσεων (βιβλιοθήκες, λειτουργικό σύστημα) και η σύνθεση ενός ενιαίου εκτελέσιμου αρχείου της εφαρμογής (unikernel), που θα μπορεί να εκτελεστεί αυτόνομα, όπως σε ένα κοινό λειτουργικό σύστημα. Ταυτόχρονα, η διεύρυνση της χρήσης επιταχυντών υλικού για υπολογιστικά απαιτητικά κομμάτια εφαρμογών καθιστά το υλικό περισσότερο προσβάσιμο, και άρα διαθέσιμο σε περιβάλλοντα cloud (Amazon AWS, Azure, κλπ). Στόχος της παρούσας εργασίας είναι η σχεδίαση και υλοποίηση ενός συστήματος που θα συνδυάζει την απάλειψη περιττών εξαρτήσεων της εφαρμογής από το περιβάλλον εκτέλεσης (unikernel) καθώς και την ένταξη επιτάχυνσης συγκεκριμένων υπολογιστικά απαιτητικών κομματιών της εφαρμογής. Συγκεκριμένα, η εργασία περιλαμβάνει: (α) μελέτη των διαθέσιμων frameworks για unikernels, (β) αποδελτίωση εφαρμογών που αξιοποιούν την επιτάχυνση υλικού σε GPUs/FPGAs, (γ) υλοποίηση του συστήματος που παράγει unikernels με αυτή την υποστήριξη. και (δ) πειραματική αποτίμηση του συστήματος.

Σχετικά Μαθήματα: Λειτουργικά Συστήματα, Εργαστήριο Λειτουργικών Συστημάτων

Σχετική Βιβλιογραφία:

- Unikernel frameworks:

1. <https://github.com/cloudius-systems/osv>
2. <http://rumpkernel.org/>
3. <https://github.com/libos-nuse/lkl-linux>
4. <http://cnp.necslab.eu/clickos/>
5. <https://wiki.xenproject.org/wiki/Mini-OS>

- Acceleration:

1. <https://www.khronos.org/onecl/>
2. <https://www.xilinx.com/products/design-tools/software-zone/sdaccel.html>

Επικοινωνία: Στράτος Ψωμαδάκης, psomas@cslab.ece.ntua.gr
Κωστής Παπαζαφειρόπουλος, kparazaf@cslab.ece.ntua.gr
Στέφανος Γεράγγελος, sgerag@cslab.ece.ntua.gr

16 Benchmarking Αλγορίθμων Consensus σε Ethereum/Hyperledger

Η τεχνολογία blockchain, που αρχικά δημιουργήθηκε για να αποτελέσει τη βάση λειτουργίας του δικτύου Bitcoin, λειτουργεί ως ένα κοινόχρηστο δημόσιο λογιστικό βιβλίο στο οποίο εγγράφονται όλες οι επιβεβαιωμένες συναλλαγές – ένα σύνολο συναλλαγών αποτελούν ένα block και το κάθε block αναφέρεται στο προηγούμενο του δημιουργώντας μια αλυσίδα [1]. Η επιβεβαίωση των συναλλαγών και η συμφωνία για τη σειρά εκτέλεσής τους γίνεται με καταναμημένο τρόπο με χρήση αλγορίθμων consensus. Οι δύο βασικές κατηγορίες τέτοιων αλγορίθμων είναι οι lottery-based (π.χ., ο Proof-of-Work του Bitcoin) και οι voting-based (π.χ., ο Byzantine Fault Tolerance του Hyperledger) [2, 3]. Καθένας από του αλγορίθμους αυτού έχουν πλεονεκτήματα και μειονεκτήματα σε σχέση με το transaction throughput που επιτυγχάνουν, την κλιμακωσιμότητά τους, την ασφάλεια που παρέχουν, ακόμα και την ενέργεια που σπαταλούν. Στόχος της διπλωματικής είναι η μελέτη διαφορετικών αλγορίθμων consensus που χρησιμοποιούνται στις πιο δημοφιλείς υλοποιήσεις Blockchain, είτε public (Ethereum [4]) είτε permissioned (Hyperledger [5]) με τη χρήση benchmarks (Blockbench [6] ή Hyperledger Caliper [7] αντίστοιχα) ως προς τις παραπάνω ιδιότητες.

Σχετικά Μαθήματα: Καταναμημένα Συστήματα

Σχετική Βιβλιογραφία:

1. Nakamoto, Satoshi. "Bitcoin: A peer-to-peer electronic cash system." (2008): 28.
2. Bano, S., Sonnino, A., Al-Bassam, M., Azouvi, S., McCorry, P., Meiklejohn, S., Danezis, G. (2017). Consensus in the age of blockchains. arXiv preprint arXiv:1711.03936.
3. Hyperledger Architecture, Volume 1, https://www.hyperledger.org/wp-content/uploads/2017/08/Hyperledger_Arch_WG_Paper_1_Consensus.pdf
4. Ethereum, <https://www.ethereum.org/>
5. Hyperledger, <https://www.hyperledger.org/>

6. Dinh, Tien Tuan Anh, et al. "Blockbench: A framework for analyzing private blockchains." Proceedings of the 2017 ACM International Conference on Management of Data. ACM, 2017.

7. Hyperledger Caliper, <https://www.hyperledger.org/projects/caliper>

Επικοινωνία: Κατερίνα Δόκα, katerina@cslab.ece.ntua.gr, 210-772-1175

17 Κλιμακωσιμότητα Blockchain

Η διαφάνεια συναλλαγών, η έλλειψη κεντρικού ελέγχου, η μη αντιστρέψιμη φύση της και η ψευδό-ανωνυμία που προσφέρει η τεχνολογία blockchain έχουν οδηγήσει στην υιοθέτησή της σε πληθώρα εφαρμογών πέρα από το νόμισμα (διαχείριση ψηφιακών περιουσιακών στοιχείων, από τίτλους ιδιοκτησίας και μετοχές μέχρι ταξιδιωτικά μίλια, δημιουργία ψηφιακών ταυτοτήτων που χρησιμοποιούνται σε ψηφιακές υπογραφές, δημιουργία επαληθεύσιμης καταγραφής για κάθε είδους δεδομένο, αρχείο ή διεργασία). Ωστόσο, σε πολλές περιπτώσεις υπάρχουσες υλοποιήσεις blockchain υστερούν σε ρυθμικό πόδοση (throughput), χρόνο απόκρισης (latency) και απαιτήσεις σε αποθηκευτικό χώρο. Η αδυναμία κλιμακωσιμότητας (scalability) επιτείνει το πρόβλημα [1]. Στόχος της διπλωματικής είναι να αντιμετωπίσει αυτούς τους περιορισμούς του blockchain υιοθετώντας διαφορετικές σχεδιαστικές επιλογές που θα το καταστήσουν καταλληλότερο για τη χρήση του σε εφαρμογές που απαιτούν υψηλό throughput και χαμηλό latency. Οι επιλογές αυτές αφορούν σε αντικατάσταση του υποκείμενου αδόμητου δικτύου P2P στο οποίο βασίζεται η τεχνολογία blockchain με κατακερματισμένο δίκτυο κατακερματισμού (DHT). Τα DHTs είναι δομημένα P2P δίκτυα τα οποία κατανέμουν δεδομένα και φόρτο δρομολόγησης στους εμπλεκόμενους κόμβους σύμφωνα με συγκεκριμένους, αυστηρούς κανόνες [2].

Σχετικά Μαθήματα: Κατανεμημένα Συστήματα

Σχετική Βιβλιογραφία:

1. McConaghy, Trent, et al. "BigchainDB: a scalable blockchain database.", BigChainDB (2016).
2. Androutsellis-Theotokis, Stephanos, and Diomidis Spinellis. "A survey of peer-to-peer content distribution technologies." ACM computing surveys (CSUR) 36.4 (2004): 335-371.

Επικοινωνία: Κατερίνα Δόκα, katerina@cslab.ece.ntua.gr, 210-772-1175

18 Βελτιστοποιητής για εφαρμογές μηχανικής μάθησης

Η μηχανική μάθηση (machine learning) χρησιμοποιείται όλο και περισσότερο σε πολλούς και διαφορετικούς τομείς εφαρμογών. Η πληθώρα αλγορίθμων μηχανικής μάθησης που υπάρχουν υλοποιημένοι σε διαφορετικές πλατφόρμες δυσχεραίνουν την απόφαση του προγραμματιστή ως προς το ποια υλοποίηση ποιου αλγορίθμου είναι η πιο κατάλληλη για μια συγκεκριμένη εφαρμογή ανάλογα με τις απαιτήσεις της (π.χ., επιθυμητή ακρίβεια και χρόνος που απαιτείται για να επιτευχθεί), τα δεδομένα που επεξεργάζεται (π.χ., streaming ή batch) και τον διαθέσιμο εξοπλισμό (π.χ., μέγεθος cluster, ύπαρξη GPUs). Ιδανικά, ο χρήστης θα ήθελε να ορίσει με δηλωτικό τρόπο το machine learning task του και κάποιες απαιτήσεις της εφαρμογής του και ένας βελτιστοποιητής (optimizer) να αποφασίζει αυτόματα τον καλύτερο δυνατό αλγόριθμο μαζί με τις παραμέτρους του συστήματος όπου θα εκτελεστεί για να καλυφθούν οι απαιτήσεις του χρήστη [5]. Η βελτιστοποίηση αυτή θα στηρίζεται σε εκτιμήσεις για το κόστος και την ακρίβεια του κάθε αλγορίθμου μηχανικής μάθησης για συγκεκριμένες παραμέτρους

και υποδομή. Στην παρούσα διπλωματική θα υλοποιηθεί ένας τέτοιος βελτιστοποιητής στο πρότυπο του συστήματος IReS που έχει αναπτυχθεί στο εργαστήριο [1]. Θα χρησιμοποιηθούν αλγόριθμοι μηχανικής μάθησης υλοποιημένοι στα πιο γνωστά κατανεμημένα Big Data frameworks, όπως Spark MLlib [2], Tensorflow [3], FlinkML [4].

Σχετικά Μαθήματα: Κατανεμημένα Συστήματα

Σχετική Βιβλιογραφία:

1. K. Doka, N. Papailiou, V. Giannakouris, D. Tsoumakos and N. Koziris: Mix 'n' Match Multi-Engine Analytics. In Proceedings of the 2016 IEEE International Conference on Big Data (BigData 2016), December 5-8, 2016, Washington DC, USA.
2. <https://spark.apache.org/mllib/>
3. <https://www.tensorflow.org/>
4. <https://ci.apache.org/projects/flink/flink-docs-release-1.2/dev/libs/ml/index.html>
5. Kaoudi, Zoi, et al. "A Cost-based Optimizer for Gradient Descent Optimization." Proceedings of the 2017 ACM International Conference on Management of Data. ACM, 2017.

Επικοινωνία: Κατερίνα Δόκα, katerina@cslab.ece.ntua.gr, 210-772-1175

19 Βελτιστοποίηση Επεξεργασίας Big Data σε Ετερογενείς Υποδομές

Σύγχρονα συστήματα επεξεργασίας δεδομένων, χρονοδρομολογητές και υποδομές cloud πλέον χρησιμοποιούν, εκτός από CPUs, hardware accelerators όπως GPUs, FPGAs, TPUs, ASICs, κλπ για να επιταχύνουν την εκτέλεση εργασιών που επεξεργάζονται πολλά δεδομένα, όπως αλγορίθμων μηχανικής μάθησης ή εργασιών ανάλυσης δεδομένων (Data Analytics). Ωστόσο αυτό συνήθως απαιτεί εξειδικευμένες γνώσεις από τον προγραμματιστή, ώστε να διαλέξει τον κατάλληλο τύπο hardware για την εφαρμογή του και να γράψει κώδικα ειδικά για το hardware που επέλεξε. Ιδανικά θα θέλαμε να υπάρχει βελτιστοποιητής, ο οποίος ανάλογα με την εφαρμογή, τα δεδομένα που επεξεργάζεται και τις απαιτήσεις του χρήστη σε χρόνο εκτέλεσης, κόστος υποδομής και κατανάλωση ενέργειας να αποφασίζει το είδος του hardware που θα χρησιμοποιηθεί και να δρομολογήσει την εφαρμογή προς εκτέλεση με διαφανή προς τον χρήστη τρόπο []. Για τον σκοπό αυτό θα χρησιμοποιηθούν είτε πλατφόρμες επεξεργασίας Big Data που υποστηρίζουν εκτέλεση σε ειδικό hardware (π.χ., Tensorflow [2], Flink [3]) είτε Βάσεις Δεδομένων που τρέχουν πάνω από accelerators [4].

Σχετικά Μαθήματα: Κατανεμημένα Συστήματα

Σχετική Βιβλιογραφία:

1. Rossbach, Christopher J., et al. "Dandelion: a compiler and runtime for heterogeneous systems." Proceedings of the Twenty-Fourth ACM Symposium on Operating Systems Principles. ACM, 2013.
2. <https://www.tensorflow.org/>
3. Chen, Cen, et al. "Gflink: An in-memory computing architecture on heterogeneous CPU-GPU clusters for big data." IEEE Transactions on Parallel and Distributed Systems 29.6 (2018): 1275-1288.

4. Breß, Sebastian, et al. "Gpu-accelerated database systems: Survey and open challenges." Transactions on Large-Scale Data-and Knowledge-Centered Systems XV. Springer, Berlin, Heidelberg, 2014. 1-35.

Επικοινωνία: Κατερίνα Δόκα, katerina@cslab.ece.ntua.gr, 210-772-1175
Ιωάννης Κωνσταντίνου, ikons@cslab.ece.ntua.gr

20 Αυτοματοποίηση χρονοδρομολόγησης και δέσμευσης πόρων για εκτέλεση εφαρμογών μεγάλης κλίμακας σε περιβάλλοντα υπολογιστικών νεφών

Τα τελευταία χρόνια η έκρηξη που σημειώνεται στην παραγωγή δεδομένων έχει δημιουργήσει αντίστοιχες ανάγκες επεξεργασίας και αξιοποίησης αυτών. Μια πληθώρα συστημάτων επεξεργασίας δεδομένων σε μεγάλη κλίμακα έχει κάνει την εμφάνισή της – συστήματα τα οποία συχνά υλοποιούν διαφορετικά προγραμματιστικά μοντέλα και παρουσιάζουν διαφορετικά χαρακτηριστικά. Οι τελευταίες εξελίξεις οδηγούν σε ένα μοντέλο όπου η υπολογιστική και προγραμματιστική πολυπλοκότητα «κρύβονται» από τον τελικό χρήστη και οι δυνατότητες των σύγχρονων αυτών συστημάτων προσφέρονται με τη μορφή υπηρεσιών που εκτελούνται πάνω από υποδομές υπολογιστικών νεφών. Με αυτό τον τρόπο, εργαλεία επεξεργασίας μεγάλου όγκου δεδομένων και μηχανικής μάθησης γίνονται προσιτά και σε χρήστες οι οποίοι πιθανώς δεν έχουν τις τεχνικές γνώσεις ώστε να τα διαχειριστούν – καθώς η διαχείριση γίνεται από τον πάροχο των υπηρεσιών αυτών. Από την πλευρά ενός οργανισμού που παίζει αυτό το ρόλο του παρόχου τέτοιου είδους υπηρεσιών, είναι επιθυμητή η δυνατότητα βέλτιστης διαχείρισης των υπολογιστικών πόρων με διπλό στόχο. Από τη μια πλευρά την ελαχιστοποίηση του κόστους και από την άλλη της τήρησης των service-level agreements (SLAs) – διμερών συμφωνιών που συνάπτονται με τους χρήστες και αφορούν το επίπεδο των παρεχόμενων υπηρεσιών.

Σε αυτό το πλαίσιο, στο εργαστήριο έχει γίνει η προσπάθεια μοντελοποίησης των εφαρμογών και συστημάτων επεξεργασίας – με τη χρήση decision trees – και χρήσης των παραγόμενων μοντέλων για την αυτοματοποίηση α) της χρονοδρομολόγησης εφαρμογών και β) της εκτίμησης των απαιτούμενων υπολογιστικών πόρων για την εκτέλεσή τους σε χρόνο που δεν παραβιάζει τα SLAs. Το ζητούμενο είναι η επέκταση της υπάρχουσας λύσης στους εξής άξονες:

- ανάπτυξη 'έξυπνης' αυτοματοποιημένης μεθόδου δειγματοληψίας για την κατασκευή μοντέλων με στόχο τη μείωση του κόστους αυτής της διαδικασίας.
- βελτίωση του αλγορίθμου επίλυσης του προβλήματος βελτιστοποίησης που προκύπτει από την ανάγκη μείωσης του κόστους αλλά και του χρόνου εκτέλεσης των εργασιών.
- διερεύνηση χρήσης διαφορετικής μορφής μοντέλων για την προσέγγιση της συμπεριφοράς των εφαρμογών και των συστημάτων – πέραν των δέντρων – καθώς και τον εντοπισμό των παραμέτρων που επηρεάζουν περισσότερο την επίδοσή τους.

Σχετικά Μαθήματα: Κατανεμημένα Συστήματα

Επικοινωνία: Νίκος Χαλβαντζής, nchalv@cslab.ece.ntua.gr
Ιωάννης Κωνσταντίνου, ikons@cslab.ece.ntua.gr

21 Συγκριτική αξιολόγηση αλγορίθμων μηχανικής μάθησης σε συστήματα επεξεργασίας δεδομένων μεγάλης κλίμακας

Σήμερα, ένας τομέας επεξεργασίας δεδομένων που παρουσιάζει μεγάλο ενδιαφέρον είναι ο τομέας της μηχανικής μάθησης. Σε αυτό το πλαίσιο, χρησιμοποιείται ένα σύνολο ιστορικών δεδομένων για την εξαγωγή συμπερασμάτων πάνω σε νέα δεδομένα (κατηγοριοποίηση, πρόβλεψη τιμών, ομαδοποίηση). Ωστόσο, καθώς ο όγκος των δεδομένων συνεχώς αυξάνεται, η διαδικασία δημιουργίας των μοντέλων που είναι ικανά να εξάγουν αυτά συμπεράσματα γίνεται όλο και πιο απαιτητική και χρονοβόρα. Για το λόγο αυτό, έχει αναπτυχθεί πληθώρα κατανεμημένων συστημάτων, εξειδικευμένα και μη σε αυτόν τον τομέα. Ένα από τα πιο διαδεδομένα κατανεμημένα συστήματα γενικού σκοπού είναι το Apache Spark, το οποίο παρέχει βιβλιοθήκες αλγορίθμων μηχανικής μάθησης (MLlib και ο απόγονος της η ML). Ωστόσο, όπως αναφέρθηκε, υπάρχει και ένα σύνολο εξειδικευμένων κατανεμημένων συστημάτων στον τομέα της μηχανικής μάθησης, όπως το Google TensorFlow. Τα διαφορετικά αυτά συστήματα παρουσιάζουν σημαντικές διαφορές μεταξύ τους, τόσο σε επίπεδο αρχιτεκτονικής όσο και σε επίπεδο υλοποίησης. Για το λόγο αυτό, κρίνεται ιδιαίτερα σκόπιμος ο εντοπισμός αυτών των στοιχειωδών διαφορών που παρουσιάζουν αυτά τα συστήματα μέσω πειραματικής ανάλυσης. Η συγκριτική αυτή μελέτη μπορεί να φανεί ιδιαίτερα χρήσιμη τόσο στους προγραμματιστές που αναπτύσσουν το εκάστοτε σύστημα, για την περαιτέρω ανάπτυξη μηχανισμών που είναι ήδη ανεπτυγμένη σε κάποιο άλλο, όσο και στους χρήστες αυτών, παρουσιάζοντας τον βέλτιστο τρόπο ρύθμισης και χρήσης αυτών.

Στο εργαστήριο έχει γίνει συγκριτική μελέτη σε βάθος για την επίδοση βασικών αλγορίθμων μηχανικής μάθησης που χρησιμοποιούν τον αλγόριθμο απότομης καθόδου (Gradient Descent) για την εκπαίδευση των μοντέλων στα συστήματα Apache Spark και Google TensorFlow. Ωστόσο, η σύγκριση αυτή μπορεί να επεκταθεί σε διάφορα επίπεδα, όπως:

- **Επέκταση αυτής της ανάλυσης σε επίπεδο συστημάτων**

Πέρα από αυτά τα δύο συστήματα υπάρχουν και άλλα κατανεμημένα συστήματα μηχανικής μάθησης. Θα ήταν χρήσιμο η ανάλυση αυτού του βασικού τμήματος της μηχανικής μάθησης να επεκταθεί και σε άλλα, όπως για παράδειγμα το GraphLab.

- **Επέκταση της ανάλυσης σε μεγαλύτερη πληθώρα αλγορίθμων**

Όπως αναφέρθηκε, η ανάλυση και η σύγκριση των δύο συστημάτων προς το παρόν περιορίστηκε ανάμεσα σε αλγόριθμους που χρησιμοποιούν τον Gradient Descent ως αλγόριθμο βελτιστοποίησης. Χρήσιμο θα ήταν να εξεταστεί η επίδοση των αλγορίθμων σε άλλους αλγόριθμους μηχανικής μάθησης, όπως ο KMeans και τα SVMs. Τμήμα της εργασίας θα είναι και η υλοποίηση των αλγορίθμων σε όποιο σύστημα αυτοί δεν είναι ήδη διαθέσιμοι.

- **Αξιολόγηση της επίδοσης του Apache Spark σε νευρωνικά δίκτυα**

Το Google Tensorflow παρουσιάζεται ως ένα πάρα πολύ καλό εργαλείο για την υλοποίηση νευρωνικών δικτύων που αποτελούν ένα βασικό εργαλείο του τομέα της βαθιάς μηχανικής μάθησης (Deep Learning). Ωστόσο, το Spark με εξαίρεση το perceptron που αποτελεί ένα απλό νευρωνικό δίκτυο δεν παρέχει την δυνατότητα κατασκευής μοντέλων αυτού του τομέα της μηχανικής μάθησης. Χρήσιμη θα ήταν η υλοποίηση κάποιων αλγορίθμων νευρωνικών δικτύων σε αυτό και σύγκριση τους με τους αντίστοιχους στο Google TensorFlow.

Σχετικά Μαθήματα: Κατανεμημένα Συστήματα

Επικοινωνία: Νίκος Προβατάς, nprov@cslab.ece.ntua.gr

Ιωάννης Κωνσταντίνου, ikons@cslab.ece.ntua.gr

22 Ελαστική διαχείριση πόρων συστημάτων διαχείρισης δεδομένων μεγάλης κλίμακας με τεχνικές μηχανικής μάθησης

Τα καταναμημένα συστήματα επεξεργασίας δεδομένων μεγάλης κλίμακας χρησιμοποιούνται ευρέως για την αποδοτική και κλιμακώσιμη εκτέλεση αλγορίθμων πάνω από δεδομένα μεγάλου όγκου. Συστήματα αναλυτικής επεξεργασίας δεδομένων όπως το Apache Hive και Spark SQL χρησιμοποιούνται για την εκτέλεση batch-SQL ερωτημάτων, βάσεις NoSQL χρησιμοποιούνται για τον αποδοτική εκτέλεση real-time ερωτημάτων ενώ συστήματα όπως το TensorFlow και το Spark MLlib για την εκτέλεση αλγορίθμων μηχανικής μάθησης. Τα συστήματα αυτά έχουν την δυνατότητα με την προσθήκη επιπλέον πόρων να βελτιώσουν την εκτέλεση των αλγορίθμων εκμεταλλευόμενα την παράλληλη φύση της επεξεργασίας, κάτι που τα κάνει ιδανικά για να τρέχουν σε περιβάλλοντα υπολογιστικών νεφών: με την εκμετάλλευση της ιδιότητας της “ελαστικότητας” μπορεί κάποιος να αυξομειώνει τους πόρους της υποδομής ανάλογα με τις ανάγκες των χρηστών.

Το εργαστήριο έχει εργαστεί στον τομέα αυτό με την γενική προσέγγιση της δημιουργίας μοντέλων μηχανικής μάθησης τα οποία προβλέπουν και/ή αποφασίζουν την κατάλληλη δράση (πχ. προσθαφαίρεση υπολογιστικών κόμβων, αλλαγή χαρακτηριστικών εφαρμογής, κλπ) για να ανταπεξέλθει η εφαρμογή σε ένα δοθέν φορτίο εργασίας με βάση ορισμένα κριτήρια επίδοσης που δίνει ο χρήστης.

Το εργαστήριο έχει μελετήσει τεχνικές ενισχυτικής μάθησης (reinforcement learning) για την διαχείριση πόρων συστημάτων NoSQL (σύστημα Tiramola [1]). Κατόπιν, μελέτησε μια προσέγγιση με προσρμοστική δημιουργία του χώρου καταστάσεων της μαρκοβιανής αλυσίδας σε περίπτωση που υπήρχαν πολλές διαστάσεις (σύστημα MDP_DT [2]) καθώς και μια προσέγγιση με βαθιά ενισχυτική μάθηση (deep reinforcement learning).

Πιθανές κατευθύνσεις:

1. Επέκταση του μοντέλου ενισχυτικής μάθησης για την εφαρμογή επιπλέον ενεργειών κατά την απόφαση αλλαγής των πόρων μιας εφαρμογής. Προς το παρόν οι αλγόριθμοι προσθαφαιρούν μόνο υπολογιστικούς κόμβους, ενώ υπάρχει η δυνατότητα αλλαγής και επιπλέον χαρακτηριστικών (μέγεθος RAM, αριθμό πυρήνων, κλπ).
2. Εφαρμογή ενός από τους αλγορίθμους των [bigdata, ccgrid] σε ένα νέο σύστημα, πχ στο TensorFlow ή στο Apache Spark με διαφορετικά είδη φορτίου.
3. Πειραματική αποτίμηση επίδοσης διαφορετικών αλγορίθμων ελαστικής διαχείρισης πόρων σε διαφορετικά συστήματα.
4. Εφαρμογή ενός από τους παραπάνω αλγορίθμους στο σύστημα Kubernetes της Google για την διαχείριση ενός καταναμημένου framework όπως το TensorFlow (πχ Kubeflow) [citation]
5. Εφαρμογή διαφορετικών τεχνικών Deep Reinforcement Learning για την εκπαίδευση του μοντέλου. Προτάσεις:
 - Τα recurrent deep neural networks χρησιμοποιούνται ευρέως όταν έχουμε να αντιμετωπίσουμε incomplete datasets, όταν πχ κάποιες μετρήσεις που δομούν ένα state λείπουν ή περιέχουν θόρυβο, όπου δεδομένου της φύσης του προβλήματος (cloud environment), σε πραγματικές συνθήκες θα συναντούμε τέτοια datasets.
 - Στο deep reinforcement learning χρησιμοποιείται η εξίσωση του Belman:
$$Q(s,a)=V(s)+A(a)$$
όπου η αξία του Q δομείται από την αξία του να είσαι σε ένα συγκεκριμένο state, V(s) συν το κέρδος που θα αποκόμιζες αν έπαιρνες το action a. Στην έως τώρα υλοποίησή μας το

δίκτυό μας υπολογίζει τα Q values συνολικά. Στα dueling deep neural networks στόχος είναι δύο διαφορετικά δίκτυα να υπολογίζουν ξεχωριστά τα $V(s)$, $A(a)$ και εν συνεχεία να δομείται η συνολική λύση. Αυτή η τεχνική βοηθάει στο decoupling του προβλήματος και στην καλύτερη εκπαίδευση του δικτύου, ενώ παράλληλα αποφεύγει τη σπατάλη πόρων σε ειδικές περιπτώσεις όπου ένα εκ των $V(s)$, $A(a)$ δεν έχουν μεγάλη σημασία για την πολιτική που έχει επιλέξει ο χρήστης και επομένως ο υπολογισμός τους μπορεί να παραλειφθεί.

Σχετικά Μαθήματα: Προχωρημένα θέματα βάσεων δεδομένων, Κατανεμημένα Συστήματα

Σχετική Βιβλιογραφία:

1. D. Tsoumakos, I. Konstantinou, C. Boumprouka, S. Sioutas and N. Koziris: Automated, Elastic Resource Provisioning for NoSQL Clusters Using TIRAMOLA. In proceedings of the 13th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid), Delft, The Netherlands, May 13-16, 2013. Best Paper Award PDF
2. K. Lolos, I. Konstantinou, V. Kantere and N. Koziris: Elastic Management of Cloud Applications using Adaptive Reinforcement Learning In proceedings of the 2017 IEEE International Conference on Big Data (BigData 2017), Boston, MA, USA December 11-14 2017 PDF

Επικοινωνία: Ιωάννης Κωνσταντίνου, ikons@cslab.ece.ntua.gr

23 Συγκριτική μελέτη τεχνικών της αλγοριθμικής θεωρίας παιχνιδιών για αποδοτικό resource allocation σε υπολογιστικά νέφη και κατασκευή ενός σύγχρονου cloud based resource allocator

Στα σύγχρονα μεγάλα υπολογιστικά νέφη η δυναμική εκχώρηση πόρων σε χρήστες ή tasks χρηστών ανάλογα με την ανάγκη του χρήστη και τη διαθεσιμότητα πόρων στο δίκτυο μια δεδομένη στιγμή εφάπτεται στο πρόβλημα του αποδοτικού resource allocation. Συστήματα όπως το Amazon AWS, το Apache Spark και το TensorFlow χρησιμοποιούν την ιδιότητα της “ελαστικότητας” για να αυξάνουν τους πόρους ενός χρήστη αναλόγως με τον όγκο της πληροφορίας που επεξεργάζεται μια δεδομένη στιγμή ή με το υπολογιστικό βάρος των ερωτημάτων που καλείται να εκτελέσει το task του. Στο εργαστήριο έχουν δημιουργηθεί επιτυχημένα μοντέλα που προβλέπουν τις ανάγκες του χρήστη σε δεδομένο χρόνο και κάνουν αίτηση στον cloud provider για τους ανάλογους πόρους χρησιμοποιώντας τεχνικές reinforcement learning.

Τα τελευταία χρόνια έχουν προταθεί μοντέλα της αλγοριθμικής θεωρίας παιχνιδιών για αποδοτικό task allocation, job scheduling και resource reservation σε υπολογιστικά νέφη, όπου σαν παίκτες λογίζονται από τη μία είτε ο χρήστης, είτε ένα συγκεκριμένο task, είτε μια φυσική μηχανή του νέφους και από την άλλη σαν αντίπαλος ένας agent της υπηρεσίας, ο οποίος είναι είτε ένας resource allocator είτε ένας task allocator είτε ένας job allocator. Διαφορετικές μεταφράσεις μιας παρόμοιας διαδικασίας όπου ένας agent του συστήματος ανταγωνίζεται σε zero-sum mixed strategy παίγνια την ανάγκη για κατάληψη πόρων από χρήστες που τους έχουν ανάγκη μια δεδομένη στιγμή[1][2][3].

Στόχος μας στο εργαστήριο είναι η κατασκευή ενός αυτοματοποιημένου συστήματος και συγκριμένα του resource allocator του συστήματος, που θα λαμβάνει υπόψιν τις αντικειμενικές ανάγκες όλως των χρηστών ενός υπολογιστικού νέφους μια δεδομένη στιγμή (χρησιμοποιώντας τα παραπάνω reinforcement learning μοντέλα) και τις άμεσα μελλοντικές ανάγκες των χρηστών και θα διανέμει αποδοτικά πόρους σε κάθε χρήστη με γνώμονα το χαμηλό κόστος της υπηρεσίας και κυρίως της δίκαιης κατανομής πόρων μεταξύ όλων των χρηστών.

Στόχος της διπλωματικής είναι η συγκριτική μελέτη τουλάχιστον τριών διαφορετικών τεχνικών που έχουν προταθεί για το resource allocation βασισμένες στην αλγοριθμική θεωρία παιγνίων και η κατασκευή της αρχιτεκτονικής όπως και η υλοποίηση ενός σύγχρονου resource allocator agent βασισμένου στην αλγοριθμική θεωρία παιγνίων που θα ενσωματώνεται πάνω σε έναν σύγχρονο cloud orchestrator όπως το Kubernetes[4] ή θα συνεργάζεται με ένα σύγχρονο cloud tool για scheduling ή resource allocating όπως το Apache Mesos[5] ή το Yarn.

Σχετικά Μαθήματα: Προχωρημένα θέματα βάσεων δεδομένων, Κατανεμημένα Συστήματα

Σχετική Βιβλιογραφία:

1. P. S. Pillai and S. Rao, "Resource Allocation in Cloud Computing Using the Uncertainty Principle of Game Theory," in IEEE Systems Journal, vol. 10, no. 2, pp. 637-648, June 2016.
2. Wei, Guiyi & Vasilakos, Athanasios & Zheng, Yao & Xiong, Naixue. (2010). A Game-Theoretic Method of Fair Resource Allocation for Cloud Computing Services. The Journal of Supercomputing. 54. 252-269. 10.1007/s11227-009-0318-1.
3. Liu, Jing, Xing-guo Luo, Xing-ming Zhang, Fan Zhang and Bai-Nan Li. "Job Scheduling Model for Cloud Computing Based on Multi-Objective Genetic Algorithm." (2013).
4. <https://kubernetes.io/>
5. <http://mesos.apache.org/>

Επικοινωνία: Κωνσταντίνος Μπιτσάκος, kbitsak@cslab.ece.ntua.gr
Ιωάννης Κωνσταντίνου, ikons@cslab.ece.ntua.gr