

Εργαστήριο Τεχνολογίας & Αρχιτεκτονικής Υπολογιστών

Τομέας Υλικού & Αρχιτεκτονικής των Υπολογιστών

Τμήμα Μηχανικών Η/Υ & Πληροφορικής

Πολυτεχνική Σχολή – Πανεπιστήμιο Πατρών

<https://tcal.ceid.upatras.gr>

Διπλωματικές Εργασίες

Ακαδημαϊκό Έτος 2025-2026

Διπλωματική Εργασία 1	Ανάλυση απόδοσης και ενεργειακής κατανάλωσης σε σύγχρονους πολυπύρηνους επεξεργαστές με χρήση των εργαλείων AMD uProf και Intel VTune (Performance and Energy Efficiency Analysis on Modern Multicore Processors using AMD uProf and Intel VTune)
Περιγραφή/ Αντικείμενο	Η εργασία έχει ως στόχο τη μελέτη και ανάλυση της απόδοσης, και της ενεργειακής κατανάλωσης σε σύγχρονα πολυπύρηνια συστήματα, βασισμένα σε επεξεργαστές AMD και Intel. Ο φοιτητής/Η φοιτήτρια θα χρησιμοποιήσει εργαλεία ανάλυσης όπως τα AMD uProf [4] και Intel VTune [3], προκειμένου να εξετάσει πως διαφορετικοί τύποι εφαρμογών (π.χ. CPU-bound, memory-bound, ή I/O-bound) αξιοποιούν τους πόρους του συστήματος και πνς παράγοντες όπως η κατανάλωση ενέργειας, η θερμοκρασία και η χρήση της μνήμης επηρεάζουν τη συνολική απόδοση. Η μελέτη θα βασιστεί σε πειραματικές μετρήσεις, όπου ο φοιτητής/η φοιτήτρια θα συλλέξει και θα αναλύσει μικροαρχιτεκτονικές μετρικές (όπως CPU utilization, cache misses, memory bandwidth, power consumption) και θα διερευνήσει σχέσεις και αλληλεπιδράσεις ανάμεσά τους. Σκοπός είναι η κατανόηση των παραγόντων που περιορίζουν την απόδοση ή τη σταθερότητα του συστήματος και η διατύπωση πιθανών προτάσεων βελτιστοποίησης, τόσο σε επίπεδο ρυθμίσεων συστήματος (system tuning) όσο και εφαρμογών (application optimization).
Στόχοι	- Εξοικείωση με τα εργαλεία AMD uProf και Intel VTune και τη συλλογή μετρήσεων. - Εκτέλεση και ανάλυση διαφορετικών τύπων εφαρμογών (compute-bound, memory-bound, I/O-bound). - Εξαγωγή μετρικών απόδοσης, κατανάλωσης ενέργειας και καθυστέρησης (latency). - Διατύπωση προτάσεων για βελτιστοποίηση και ενεργειακά αποδοτική λειτουργία χωρίς απώλεια επιδόσεων.
Ενδεικτική Βιβλιογραφία	[1] D. Ustiugov, T. Amariuca and B. Grot, "Analyzing Tail Latency in Serverless Clouds with STeLLAR," 2021 IEEE International Symposium on Workload Characterization (IISWC), Storrs, CT, USA, 2021, pp. 51-62, doi: 10.1109/IISWC53511.2021.00016. [2] The Polyhedral Benchmark Suite, https://cavazos-lab.github.io/PolyBench-ACC/ [3] Intel VTune Profiler Documentation: https://www.intel.com/vtune [4] AMD uProf Documentation: https://developer.amd.com/amd-uprof/ [5] J. L. Hennessy, D. A. Patterson, Computer Architecture: A Quantitative Approach, 6th Ed, Morgan Kaufmann, 2017.
Σχετικά Μαθήματα	Αρχιτεκτονική Υπολογιστών, Εργαστήριο Αρχιτεκτονικής Υπολογιστών, Προχωρημένα Θέματα Αρχιτεκτονικής Υπολογιστών, Προγραμματισμός Συστημάτων, Λειτουργικά Συστήματα
Επιβλέπων	Γιώργος Παπαδημητρίου, Επίκουρος Καθηγητής

Διπλωματική Εργασία 2	Μελέτη της μικροαρχιτεκτονικής συμπεριφοράς διαφορετικών μοτίβων εκτέλεσης με χρήση της Bringup-Bench σουίτας στον gem5 (Studying the Microarchitectural Behavior of Different Program Patterns Using the Bringup-Bench Benchmark Suite on gem5)
Περιγραφή/ Αντικείμενο	Η κατανόηση της μικροαρχιτεκτονικής συμπεριφοράς των προγραμμάτων είναι ένα από τα πιο σημαντικά βήματα στη μελέτη της απόδοσης των σύγχρονων επεξεργαστών. Κάθε πρόγραμμα, ακόμα και μικρό, ασκεί διαφορετικού είδους «πίεση» στη CPU και στη μνήμη. Άλλα προγράμματα εξαρτώνται από τις προσπελάσεις στη μνήμη (memory-bound), και άλλα από τους αριθμητικούς υπολογισμούς (compute-bound). Για να μπορέσουμε να μελετήσουμε αυτά τα φαινόμενα σε βάθος, η ομάδα του πανεπιστημίου του Michigan παρουσίασε πρόσφατα τη σουίτα Bringup-Bench Benchmark Suite [1], ένα σύνολο από μικρά, προσεκτικά σχεδιασμένα workloads που αναπαριστούν βασικά μοτίβα συμπεριφοράς προγραμμάτων. Στην παρούσα διπλωματική, ο φοιτητής/η φοιτήτρια θα χρησιμοποιήσει τον μικροαρχιτεκτονικό προσομοιωτή gem5 [2], έναν state-of-the-art προσομοιωτή για όλες τις επικρατέστερες αρχιτεκτονικές συνόλου εντολών (instruction set architectures), για να εκτελέσει τα workloads της Bringup-Bench σουίτας και να μελετήσει λεπτομερώς την απόδοση διαφορετικών μοτίβων εκτέλεσης [3]. Θα μάθει να χειρίζεται μικροαρχιτεκτονικές παραμέτρους (όπως το βάθος της διοχέτευσης (pipeline depth), μεγέθη κρυφών μνημών, συσχετιστικότητα, κλπ.) και θα πραγματοποιήσει αναλύσεις των στατιστικών απόδοσης (π.χ. IPC, cache miss rates, branch mispredictions). Στόχος είναι να δημιουργηθεί μια ταξινόμηση των προγραμμάτων ανάλογα με το ποια μικροαρχιτεκτονικά στοιχεία επηρεάζουν περισσότερο την απόδοσή τους — μια γνώση εξαιρετικά χρήσιμη τόσο για ερευνητές όσο και για φοιτητές που θέλουν να κατανοήσουν τη συμπεριφορά των πραγματικών workloads.
Στόχοι	1. Εξοικείωση με την Bringup-Bench σουίτα και τον gem5: εγκατάσταση, εκτέλεση και ανάλυση των βασικών benchmarks. 2. Ομαδοποίηση των προγραμμάτων: προσδιορισμός κατηγοριών (compute-bound, memory-bound, κλπ). 3. Ανάλυση της επίδρασης διαφόρων παραμέτρων: αλλαγές μεγέθους κρυφών μνημών, issue width, branch predictor, κλπ. 4. Δημιουργία οπτικοποιημένων αποτελεσμάτων και αναφοράς: συγκριτικοί πίνακες, διαγράμματα και περιγραφές που να βοηθούν στη γρήγορη εκτίμηση της συμπεριφοράς κάθε workload.
Ενδεικτική Βιβλιογραφία	[1] https://github.com/toddmaustin/bringup-bench [2] The gem5 Simulator, https://www.gem5.org [3] Lieven Eeckhout, Computer Architecture Performance Evaluation Methods, Synthesis Lectures on Computer Architecture (SLCA), 2010. [4] J. L. Hennessy, D. A. Patterson, Computer Architecture: A Quantitative Approach, 6th Ed, Morgan Kaufmann, 2017.
Σχετικά Μαθήματα	Αρχιτεκτονική Υπολογιστών, Εργαστήριο Αρχιτεκτονικής Υπολογιστών, Προχωρημένα Θέματα Αρχιτεκτονικής Υπολογιστών, Προγραμματισμός Συστημάτων, Λειτουργικά Συστήματα
Επιβλέπων	Γιώργος Παπαδημητρίου, Επίκουρος Καθηγητής

Διπλωματική Εργασία 3	Μελέτη της απόδοσης και της μικροαρχιτεκτονικής συμπεριφοράς AMD GPU μοντέλων με χρήση του προσομοιωτή gem5 (Performance and Microarchitectural Behavior Analysis of AMD GPU Models Using the gem5 Simulator)
Περιγραφή/ Αντικείμενο	Οι σύγχρονες μονάδες επεξεργασίας γραφικών (Graphics Processing Units – GPUs) αποτελούν τον βασικό υπολογιστικό πυρήνα σε πληθώρα εφαρμογών, από υπολογιστικά εντατικά workloads (HPC, machine learning, scientific computing) έως ενσωματωμένα συστήματα και edge AI πλατφόρμες. Η κατανόηση της μικροαρχιτεκτονικής συμπεριφοράς και των παραμέτρων απόδοσης των GPUs είναι κρίσιμη τόσο για τη βελτιστοποίηση των συστημάτων αυτών όσο και για την ανάπτυξη νέων αρχιτεκτονικών και πολιτικών διαχείρισης πόρων. Ο gem5 αποτελεί έναν από τους πιο διαδεδομένους μικροαρχιτεκτονικούς προσομοιωτές, παρέχοντας υποστήριξη για CPU και GPU μοντέλα. Στο GPU κομμάτι, ενσωματώνει ένα μοντέλο αρχιτεκτονικής AMD GCN (Graphics Core Next), το οποίο προσφέρει τη δυνατότητα λεπτομερούς μελέτης θεμάτων όπως: - ο βαθμός παραλληλίας και αξιοποίησης των μονάδων επεξεργασίας, - η συμπεριφορά των ιεραρχιών μνήμης, - η απόδοση σε διαφορετικού τύπου workloads (compute-bound, memory-bound). Η εργασία αυτή έχει ως στόχο τη συστηματική μελέτη της απόδοσης και συμπεριφοράς των AMD GPU μοντέλων του gem5, μέσα από διαφορετικά configuration setups (π.χ. αριθμός compute units, cache sizes, memory latency) και GPU workloads. Ο φοιτητής/Η φοιτήτρια θα εκτελέσει και θα αναλύσει προγράμματα από γνωστές σουίτες μετροπρογραμμάτων, όπως η Rodinia, καθώς και απλά microbenchmarks (π.χ. vector addition, matrix multiplication, stencil), ώστε να εξεταστεί η συμπεριφορά της GPU με διαφορετικού τύπου προγράμματα. Μέσα από την εργασία, ο φοιτητής/η φοιτήτρια θα αποκτήσει εις βάθος γνώση της αρχιτεκτονικής GCN, θα μάθει να χρησιμοποιεί και να τροποποιεί gem5 GPU configurations, και θα εξαγάγει ποσοτικές μετρικές. Τα αποτελέσματα θα αναλυθούν συγκριτικά ώστε να αναδειχθούν: - οι κύριοι παράγοντες που επηρεάζουν την απόδοση, - η σχέση ανάμεσα σε μοτίβα εκτέλεσης και σημεία συμφόρησης, - και πιθανές μικροαρχιτεκτονικές βελτιώσεις ή βέλτιστες ρυθμίσεις.
Στόχοι	- Μελέτη και κατανόηση της αρχιτεκτονικής AMD GCN που υποστηρίζεται από το GPU μοντέλο του gem5. - Εκτέλεση και αξιολόγηση GPU workloads (Rodinia, microbenchmarks) σε διαφορετικές παραμετροποιήσεις (π.χ. compute units, cache sizes). - Ανάλυση μετρικών απόδοσης, όπως occupancy, IPC, cache hit/miss rates, κλπ. - Αναγνώριση και κατηγοριοποίηση performance bottlenecks ανά workload (compute-bound, memory-bound). - Σύνταξη συγκριτικής μελέτης με γραφήματα και ανάλυση των trade-offs μεταξύ επιδόσεων και χρονικών καθυστερήσεων για διαφορετικές ρυθμίσεις.
Ενδεικτική Βιβλιογραφία	[1] M. Poremba, V. Ramadas, M. Sinclair, "Running ML Workloads in PyTorch/TensorFlow in gem5," https://www.gem5.org/assets/files/isca2024-tutorial/05-gpu.pdf. [2] D Ramadas et al., "Simulation Support for Fast and Accurate Large-Scale GPGPU Workloads in gem5," OSCAR 2024, https://pages.cs.wisc.edu/~sinclair/papers/vramadas-oscar24-gem5GPUFS.pdf. [3] The gem5 Simulator, https://www.gem5.org [4] AMD Graphics Cores Next (GCN) Architecture, https://www.techpowerup.com/gpu-specs/docs/amd-gcn1-architecture.pdf [5] J. L. Hennessy, D. A. Patterson, Computer Architecture: A Quantitative Approach, 6th Ed, Morgan Kaufmann, 2017.
Σχετικά Μαθήματα	Αρχιτεκτονική Υπολογιστών, Εργαστήριο Αρχιτεκτονικής Υπολογιστών, Προχωρημένα Θέματα Αρχιτεκτονικής Υπολογιστών, Προγραμματισμός Συστημάτων, Λειτουργικά Συστήματα
Επιβλέπων	Γιώργος Παπαδημητρίου, Επίκουρος Καθηγητής

Διπλωματική Εργασία 4	Ανάλυση των επιπτώσεων της διαφύλλωσης κλήσεων σε Serverless εφαρμογές με τη χρήση των vSwarm-u (Analysis of Interleaving Effects in Serverless Workloads using vSwarm-u)
Περιγραφή/ Αντικείμενο	Η ανάπτυξη serverless εφαρμογών έχει καταστήσει την έννοια της λειτουργίας (function) ως υπηρεσία κυρίαρχη στα σύγχρονα cloud περιβάλλοντα. Ωστόσο, σε αντίθεση με παραδοσιακές μακροχρόνιες εφαρμογές, οι serverless λειτουργίες χαρακτηρίζονται από πολύ μικρό χρόνο εκτέλεσης, περιοδικές ενεργοποιήσεις και έντονη διαφύλλωση (interleaving) μεταξύ πολλαπλών κλήσεων (invocations). Ένα από τα πιο κρίσιμα προβλήματα που αναδύονται σε αυτό το σενάριο είναι οι καθυστερήσεις στο υποσύστημα προσκόμισης εντολών (instruction fetch frontend): όταν οι λειτουργίες είναι “lukewarm” (δηλαδή, όχι πλήρως ψυχρές (cold) αλλά και όχι διαρκώς ενεργές (active)), η μονάδα πρόβλεψης διακλαδώσεων (branch predictor), η κρυφή μνήμη εντολών (instruction cache) και οι διάφορες ενδιάμεσες μνήμες (buffers) τους αδειάζουν συχνά, με αποτέλεσμα επιπλέον καθυστερήσεις (stalls) στη διοχέτευση (pipeline) και μεγάλες ποινές αστοχίας (miss penalties). Στη βιβλιογραφία έχει παρατηρηθεί το φαινόμενο αυτό μελετώντας το σε πραγματικό σύστημα, αλλά η μικροαρχιτεκτονική διάστασή του δεν έχει μελετηθεί επαρκώς σε περιβάλλοντα προσομοίωσης. Στην παρούσα διπλωματική θα χρησιμοποιηθεί η σουίτα vSwarm-u, μια παραλλαγή του vSwarm suite που εκτελεί serverless κλήσεις στον gem5, για να αναλυθεί η επίδραση της διαφύλλωσης (interleaving) σε λειτουργίες διαφορετικού τύπου (CPU-bound, memory-bound, branch-heavy). Θα σχεδιαστούν σενάρια με back-to-back invocations και σενάρια με interleaved invocations με διαφορετικούς χρόνους άφιξης. Οι μετρικές που θα μετρηθούν περιλαμβάνουν IPC, instruction cache miss rates, branch misprediction rate και frontend stalls. Επιπλέον, θα εξεταστούν μικροαρχιτεκτονικές τροποποιήσεις όπως αύξηση μεγέθους της instruction cache, ενεργοποίηση prefetchers κλπ. Συνολικά, ο φοιτητής/η φοιτήτρια θα στήσει το περιβάλλον του vSwarm-u στον gem5, θα εκτελέσει functions με διαφορετικά invocation patterns, θα συλλέξει μετρικές και θα τις αναλύσει συγκριτικά. Στη συνέχεια, θα επαναλάβει τα ίδια πειράματα με παραλλαγές μικροαρχιτεκτονικής (π.χ. μεγαλύτερες caches, ενεργοποίηση prefetching, κλπ.) ώστε να εκτιμηθεί το μέγεθος της βελτίωσης. Η ανάλυση αυτή θα δώσει τη δυνατότητα να καταγραφούν τα σημεία συμφόρησης του «μπροστινού» υποσυστήματος προσκόμισης εντολών και να διατυπωθούν προτάσεις για στοχευμένες βελτιώσεις στους επεξεργαστές που εκτελούν serverless workloads.
Στόχοι	1. Επανεκτέλεση των vSwarm standalone functions στον gem5 μέσω vSwarm-u με ελεγχόμενη διαφύλλωση. 2. Καταγραφή και ανάλυση διάφορων μετρικών, όπως frontend stalls, I-cache/L2/L3 misses, CPI. 3. Διερεύνηση επιδράσεων αλλαγών (π.χ. διπλασιασμός I-cache, instruction prefetch). 4. Διατύπωση συμπερασμάτων και προτάσεων για βελτιστοποιήσεις σε επίπεδο μικροαρχιτεκτονικής για “lukewarm” μοτίβα κλήσεων.
Ενδεικτική Βιβλιογραφία	[1] D. Schall, A. Margaritov, D. Ustiugov, A. Sandberg, and B. Grot, “Lukewarm serverless functions: characterization and optimization,” In Proceedings of the 49th Annual International Symposium on Computer Architecture (ISCA '22), 757–770, 2022, https://doi.org/10.1145/3470496.3527390. [2] G. Pournaras, V. Karakostas, G. Papadimitriou, D. Gizopoulos, “Benchmarking Support for RISC-V CPUs in Serverless Computing”, IEEE International Symposium on Workload Characterization, 2025. [3] vSwarm - Serverless Benchmarking Suite, https://github.com/vhive-serverless/vSwarm [4] vSwarm-RISCV – Serverless Benchmarking Suite, https://github.com/vitamin-v-software/vSwarm-RISCV [5] The gem5 Simulator, https://www.gem5.org [6] J. L. Hennessy, D. A. Patterson, Computer Architecture: A Quantitative Approach, 6th Ed, Morgan Kaufmann, 2017.
Σχετικά Μαθήματα	Αρχιτεκτονική Υπολογιστών, Εργαστήριο Αρχιτεκτονικής Υπολογιστών, Προχωρημένα Θέματα Αρχιτεκτονικής Υπολογιστών, Προγραμματισμός Συστημάτων, Λειτουργικά Συστήματα
Επιβλέπων	Γιώργος Παπαδημητρίου, Επίκουρος Καθηγητής

Διπλωματική Εργασία 5	Ανάλυση του φαινομένου των ψυχρών εκκινήσεων σε Serverless εφαρμογές με τεχνικές προφόρτωσης χρησιμοποιώντας τη σουίτα vSwarm-u (Analysis of Cold Start Phenomena in Serverless Workloads with Preloading Techniques using vSwarm-u)
Περιγραφή/ Αντικείμενο	Ένα από τα σημαντικότερα προβλήματα στα serverless συστήματα είναι το φαινόμενο των ψυχρών εκκινήσεων (cold start): ο χρόνος που απαιτείται όταν μια λειτουργία καλείται για πρώτη φορά, πριν από την προθέρμανση (warm-up) του επεξεργαστή, δηλαδή πριν γεμίσουν οι κρυφές μνήμες, οι πίνακες μετάφρασης διευθύνσεων (TLBs) και οι μονάδες πρόβλεψης διακλαδώσεων. Στη βιβλιογραφία, το φαινόμενο των ψυχρών εκκινήσεων έχει μελετηθεί κυρίως σε επίπεδο περιβάλλοντος εκτέλεσης (runtime) ή περιβάλλοντος κοντέινερ (container), όπως στην περίπτωση του AWS Lambda. Ωστόσο, η μικροαρχιτεκτονική διάσταση του cold start, δηλαδή το πως οι πρώτες εντολές μιας serverless λειτουργίας αλληλεπιδρούν με το «μπροστινό» υποσύστημα προσκόμισης εντολών (instruction fetch frontend), τις κρυφές μνήμες και τον branch predictor, δεν έχει μελετηθεί επαρκώς. Η παρούσα διπλωματική εργασία θα αξιοποιήσει τη σουίτα vSwarm-u για την μικροαρχιτεκτονική αξιολόγηση του φαινομένου των ψυχρών εκκινήσεων σε διάφορες Serverless λειτουργίες (όπως οι AES, Fibonacci κλπ.) χρησιμοποιώντας τον μικροαρχιτεκτονικό προσομοιωτή gem5. Θα διεξαχθούν πειράματα όπου μια λειτουργία εκτελείται σε πλήρως ψυχρή κατάσταση (με επαναφορά – reset του gem5), και θα συγκριθούν με σενάρια προθέρμανσης όπου εκτελείται μια προκαταρκτική κλήση της λειτουργίας, ώστε να «γεμίσουν» οι κρυφές μνήμες και οι TLB πριν την κύρια μέτρηση. Οι μετρικές που θα καταγραφούν περιλαμβάνουν το συνολικό χρόνο απόκρισης (latency), τις εντολές ανά κύκλο (IPC), τα ποσοστά αστοχίας (miss rates) και τα ποσοστά λανθασμένων προβλέψεων διακλαδώσεων. Στη συνέχεια, θα εξεταστεί η επίδραση διαφορετικών μεγεθών κρυφών μνημών, διαφορετικών πολιτικών αντικατάστασης (replacement policies) και της ενεργοποίησης των μονάδων εκ των προτέρων προσκόμισης (prefetchers) αναφορικά με το κόστος των ψυχρών εκκινήσεων. Με τον τρόπο αυτό, η εργασία θα παραγάγει μια μικροαρχιτεκτονική χαρτογράφηση του κόστους των ψυχρών εκκινήσεων, θα ποσοτικοποιήσει το όφελος που προκύπτει από την προφόρτωση (preloading) ή την προθέρμανση (warm-up), και θα καταγράψει ποιες αλλαγές έχουν τον μεγαλύτερο αντίκτυπο στη μείωση του χρόνου απόκρισης. Τελικός στόχος είναι η διατύπωση προτάσεων για συστήματα εκτέλεσης (runtimes) που αξιοποιούν μικροαρχιτεκτονικές ιδιαιτερότητες, ώστε να περιορίζεται το κόστος των ψυχρών εκκινήσεων σε πραγματικές serverless εφαρμογές.
Στόχοι	1. Μέτρηση ψυχρών εκκινήσεων (cold start latency) στον gem5 με τη σουίτα vSwarm-u (benchmark που εκτελείται πρώτη φορά). 2. Καταγραφή πως αλλάζουν σημαντικές μετρικές (metrics) όταν ακολουθεί run-once warm-up (π.χ. cache, TLB). 3. Εκτίμηση οφέλους σε latency και CPI 4. Προτάσεις για runtime μηχανισμούς (warm-up pooling, prefetching, snapshotting) με μικροαρχιτεκτονικές επιπτώσεις.
Ενδεικτική Βιβλιογραφία	[1] A. Bhattacharya and T. Wen, “Understanding and Remediating Cold Starts: An AWS Lambda Perspective,” https://aws.amazon.com/blogs/compute/understanding-and-remediating-cold-starts-an-aws-lambda-perspective. [2] M. Golec, G. Kaur Walia, M. Kumar, F. Cuadrado, S. Singh Gill, and S. Uhlig, “Cold Start Latency in Serverless Computing: A Systematic Review, Taxonomy, and Future Directions,” ACM Comput. Surv. 57, 3, Article 65 (March 2025), https://doi.org/10.1145/3700875. [3] G. Pournaras, V. Karakostas, G. Papadimitriou, D. Gizopoulos, “Benchmarking Support for RISC-V CPUs in Serverless Computing”, IEEE International Symposium on Workload Characterization, 2025. [4] vSwarm - Serverless Benchmarking Suite, https://github.com/vhive-serverless/vSwarm [5] vSwarm-RISCV – Serverless Benchmarking Suite, https://github.com/vitamin-v-software/vSwarm-RISCV [6] The gem5 Simulator, https://www.gem5.org [7] J. L. Hennessy, D. A. Patterson, Computer Architecture: A Quantitative Approach, 6th Ed, Morgan Kaufmann, 2017.
Σχετικά Μαθήματα	Αρχιτεκτονική Υπολογιστών, Εργαστήριο Αρχιτεκτονικής Υπολογιστών, Προχωρημένα Θέματα Αρχιτεκτονικής Υπολογιστών, Προγραμματισμός Συστημάτων, Λειτουργικά Συστήματα
Επιβλέπων	Γιώργος Παπαδημητρίου, Επίκουρος Καθηγητής

Διπλωματική Εργασία 6	Μελέτη της αξιοπιστίας μέσω στατιστικής εισαγωγής σφαλμάτων σε Serverless εφαρμογές με χρήστη της σουίτας vSwarm-u (Reliability Assessment via Statistical Fault Injection on Serverless Functions using vSwarm-u)
Περιγραφή/ Αντικείμενο	Η εξάπλωση των serverless εφαρμογών σε κρίσιμα περιβάλλοντα, όπως οι οικονομικές συναλλαγές, τα συστήματα εφοδιαστικής αλυσίδας (logistics) και οι αναλύσεις σε πραγματικό χρόνο (real-time analytics), καθιστά την αξιοπιστία εξίσου σημαντική με την απόδοση. Παρά τη διαρκώς αυξανόμενη σημασία της, η πλειονότητα των μελετών που αξιολογούν τις serverless λειτουργίες επικεντρώνεται αποκλειστικά σε μετρικές απόδοσης και δεν εξετάζει την ανθεκτικότητα σε σφάλματα σε επίπεδο μικροαρχιτεκτονικής. Η συγκεκριμένη διπλωματική εργασία στοχεύει στην ενσωμάτωση μηχανισμών στατιστικής εισαγωγής σφαλμάτων (statistical fault injection) στο περιβάλλον του μικροαρχιτεκτονικού προσομοιωτή gem5 και τη χρήση της σουίτας vSwarm-u, με σκοπό τη μελέτη της συμπεριφοράς των serverless λειτουργιών υπό την παρουσία σφαλμάτων. Η διπλωματική εργασία στοχεύει να επεκτείνει τον gem5 με μονάδες εισαγωγής σφαλμάτων για την μελέτη διαφορετικών τύπων σφαλμάτων, όπως τα παροδικά σφάλματα (transient faults) και μόνιμα σφάλματα (permanent faults), σε κρίσιμες δομές του επεξεργαστή, όπως τα αρχεία καταχωρητών (register files), οι κρυφές μνήμες εντολών και δεδομένων, οι πίνακες μετάφρασης διευθύνσεων (TLBs) και οι μονάδες πρόβλεψης διακλαδώσεων (branch predictors). Κατά την εκτέλεση των serverless λειτουργιών θα μετρηθεί το ποσοστό εμφάνισης σιωπηλών καταστροφών δεδομένων (silent data corruptions – SDCs), καθώς και η υποβάθμιση της απόδοσης. Θα πραγματοποιηθεί επίσης στατιστική ανάλυση για τον εντοπισμό των πιο ευάλωτων δομών και των λειτουργιών που παρουσιάζουν τη μεγαλύτερη ευαισθησία για αυτές τις λειτουργίες. Το πλάνο περιλαμβάνει την επιλογή αντιπροσωπευτικών λειτουργιών από το vSwarm-u, την εκτέλεση με διαφορετικούς ρυθμούς εισαγωγής σφαλμάτων, καθώς και τη συλλογή μετρικών σε πολλαπλές επαναλήψεις για στατιστική εγκυρότητα. Στη συνέχεια θα γίνει κατηγοριοποίηση των αποτελεσμάτων, θα παρασταθούν γραφικά και θα εξαχθούν χάρτες ανθεκτικότητας (resilience maps) ανά δομή και τύπο λειτουργίας. Τελικός στόχος είναι η διατύπωση προτάσεων για στοχευμένους μηχανισμούς προστασίας, όπως κωδικοποίηση διόρθωσης σφαλμάτων (ECC) στις κρυφές μνήμες, πλεονασμός (redundancy), κλπ., οι οποίοι θα μειώνουν τον ρυθμό σφαλμάτων με περιορισμένο κόστος σε απόδοση.
Στόχοι	1. Υλοποίηση μονάδας στατιστικής εισαγωγής σφαλμάτων (statistical fault injector) στον gem5 και εκτέλεση των Serverless λειτουργιών της σουίτας vSwarm-u. 2. Εκτέλεση Serverless λειτουργιών με/χωρίς σφάλματα, καταγραφή αποτελεσμάτων (crash, silent data corruption, performance degradation). 3. Ανάλυση ποια σφάλματα εμφανίζονται πιο συχνά, ανά workload και δομή. 4. Προτάσεις για μηχανισμούς ανοχής σφαλμάτων (ECC, redundancy, κλπ.) για αυτή τη μορφή λειτουργιών / προγραμμάτων.
Ενδεικτική Βιβλιογραφία	[1] G. Papadimitriou and D. Gizopoulos, "Demystifying the System Vulnerability Stack: Transient Fault Effects Across the Layers," 2021 ACM/IEEE 48th Annual International Symposium on Computer Architecture (ISCA), Valencia, Spain, 2021, pp. 902-915, doi: 10.1109/ISCA52012.2021.00075. [2] G. Pournaras, V. Karakostas, G. Papadimitriou, D. Gizopoulos, "Benchmarking Support for RISC-V CPUs in Serverless Computing", IEEE International Symposium on Workload Characterization, 2025. [3] J. L. Hennessy, D. A. Patterson, Computer Architecture: A Quantitative Approach, 6th Ed, Morgan Kaufmann, 2017. [4] vSwarm - Serverless Benchmarking Suite, https://github.com/vhive-serverless/vSwarm [5] vSwarm-RISCV – Serverless Benchmarking Suite, https://github.com/vitamin-v-software/vSwarm-RISCV [6] The gem5 Simulator, https://www.gem5.org
Σχετικά Μαθήματα	Αρχιτεκτονική Υπολογιστών, Εργαστήριο Αρχιτεκτονικής Υπολογιστών, Προχωρημένα Θέματα Αρχιτεκτονικής Υπολογιστών, Προγραμματισμός Συστημάτων, Λειτουργικά Συστήματα
Επιβλέπων	Γιώργος Παπαδημητρίου, Επίκουρος Καθηγητής

Διπλωματική Εργασία 7	Μελέτη της αξιοπιστίας μέσω στατιστικής εισαγωγής σφαλμάτων σε NVDLA μοντέλα με τη χρήση του gem5 (Reliability Evaluation through Statistical Fault Injection in NVDLA Accelerator Models using gem5 Simulator)
Περιγραφή/ Αντικείμενο	Οι επιταχυντές τεχνητής νοημοσύνης (AI accelerators) έχουν πλέον εισέλθει σε κάθε σύγχρονο σύστημα, από smartphones μέχρι αυτόνομα οχήματα. Ένας από τους πιο διαδεδομένους ανοικτού κώδικα επιταχυντές είναι ο NVIDIA Deep Learning Accelerator (NVDLA), ο οποίος έχει σχεδιαστεί για inference νευρωνικών δικτύων με χαμηλή κατανάλωση ενέργειας. Ωστόσο, οι μονάδες αυτές περιέχουν πυκνά κυκλώματα αποθήκευσης και υπολογισμού (π.χ. systolic arrays, convolution buffers), τα οποία είναι ευάλωτα σε παροδικά ή/και μόνιμα σφάλματα (transient and/or permanent faults) — σφάλματα στα bit υλικού λόγω ακτινοβολίας, θορύβου τάσης ή γήρανσης (aging), κλπ. Τέτοια σφάλματα μπορεί να προκαλέσουν μικρές ή μεγάλες αποκλίσεις στην τελική έξοδο ενός μοντέλου, επηρεάζοντας την αξιοπιστία ενός συστήματος (π.χ. σε εφαρμογές αυτοκινήτων ή υγείας). Η παρούσα διπλωματική εργασία στοχεύει στη μελέτη της ανθεκτικότητας (resilience) του NVDLA μέσω του gem5, προσθέτοντας έναν μηχανισμό στατιστικής εισαγωγής σφαλμάτων (fault injection) σε επιλεγμένες δομές (όπως buffers, systolic cores ή control registers). Ο φοιτητής/Η φοιτήτρια θα εκτελέσει νευρωνικά δίκτυα (π.χ. LeNet, ResNet, AlexNet) και θα μετρήσει πως η ακρίβεια των αποτελεσμάτων επηρεάζεται από διαφορετικά είδη σφαλμάτων. Η εργασία συνδυάζει σχεδιασμό, προσομοίωση και αξιολόγηση αξιοπιστίας, δίνοντας τη δυνατότητα στον φοιτητή/στη φοιτήτρια να εξοικειωθεί με τεχνικές ανοχής σφαλμάτων (fault tolerance) σε επιταχυντές τεχνητής νοημοσύνης, ένα από τα πιο ανερχόμενα ερευνητικά πεδία στην περιοχή της αρχιτεκτονικής υπολογιστών.
Στόχοι	1. Κατανόηση της αρχιτεκτονικής NVDLA: διερεύνηση των εσωτερικών μονάδων. 2. Υλοποίηση μηχανισμού στατιστικής εισαγωγής σφαλμάτων σε συγκεκριμένες δομές του υλικού 3. Εκτέλεση και αξιολόγηση DNN workloads: μέτρηση επιπτώσεων σε accuracy, latency και throughput για διαφορετικούς ρυθμούς σφαλμάτων. 4. Εντοπισμός κρίσιμων περιοχών που επηρεάζουν περισσότερο την ακρίβεια. 5. Προτάσεις για βελτιωμένες τεχνικές ανθεκτικότητας: όπως selective redundancy..
Ενδεικτική Βιβλιογραφία	[1] C. Lai and W. Zhang, "gem5-NVDLA: A Simulation Framework for Compiling, Scheduling, and Architecture Evaluation on AI System-on-Chips," ACM Trans. Des. Autom. Electron. Syst. 29, 5, Article 84, September 2024, https://doi.org/10.1145/3661997. [2] gem5-NVDLA GitHub repo, https://github.com/suchandler96/gem5-NVDLA [3] J. L. Hennessy, D. A. Patterson, Computer Architecture: A Quantitative Approach, 6th Ed, Morgan Kaufmann, 2017. [4] G. Li et al., "Understanding Error Propagation in Deep Learning Neural Network (DNN) Accelerators and Applications," SC17: International Conference for High Performance Computing, Networking, Storage and Analysis, Denver, CO, USA, 2017, pp. 1-12. [5] Q. Zhang, C. Liu, B. Liu, H. Huang, Y. Wang, H. Li, X. Li, "Cross-Layer Optimization for Fault-Tolerant Deep Learning," 2023, https://arxiv.org/abs/2312.13754 [6] The gem5 Simulator, https://www.gem5.org
Σχετικά Μαθήματα	Αρχιτεκτονική Υπολογιστών, Εργαστήριο Αρχιτεκτονικής Υπολογιστών, Προχωρημένα Θέματα Αρχιτεκτονικής Υπολογιστών, Προγραμματισμός Συστημάτων, Λειτουργικά Συστήματα
Επιβλέπων	Γιώργος Παπαδημητρίου, Επίκουρος Καθηγητής