



Marco Angioli

ISTRUZIONE E FORMAZIONE

Dottorato in Tecnologie dell'Informazione e delle Comunicazioni (ICT), Curriculum: Ingegneria Elettronica

Sapienza University Of Rome [01/11/2022 – Attuale]

Città: Roma

Paese: Italia

Sito web: https://phd.uniroma1.it/web/MARCO-ANGIOLI_nP1773057_IT.aspx

Campi di studio: Digital Hardware Design for Machine Learning

Laurea Magistrale in Ingegneria Elettronica

Sapienza Università di Roma [09/2019 – 15/07/2022]

Voto finale: 110/110L

Tesi: Analisi ed implementazione di algoritmi Contextual Bandits su processori RISC-V dotati di accelerazione vettoriale riconfigurabile

Laurea Triennale in Ingegneria Elettronica

Sapienza Università di Roma [09/2016 – 10/12/2019]

Paese: Italia

Voto finale: 100/110

Tesi: Stato dell'arte nella computazione neuromorfica (neuromorphic computing)

COMPETENZE LINGUISTICHE

Lingua madre: **Italiano**

Altre lingue:

Inglese

ASCOLTO B2 LETTURA B2 SCRITTURA B2

PRODUZIONE ORALE B1 INTERAZIONE ORALE B1

Livelli: A1 e A2: Livello elementare B1 e B2: Livello intermedio C1 e C2: Livello avanzato

COMPETENZE DIGITALI

Linguaggi di programmazione

Programmazione VHDL / Programmazione C++ / Programmazione Bash / Matlab / Programmazione in C / Programmazione Python / System Verilog

Software

Padronanza del Pacchetto Office (Word Excel PowerPoint ecc) / ModelSim / Vivado / Matlab e Simulink / Autocad / Synopsis Verdi / Synopsis Fusion Compiler

Altre competenze

Machine Learning / AI and Deep Neural Networks / DIGILENT ZYBO / Embedded Linux (Petalinux, Buildroot, Yocto), SoC (FPGA) / Programmazione FPGA / Reinforcement Learning / Conoscenza del Data Mining e capacità di analisi di Classificazione, Clustering o Regressione / Windows / Elaborazione delle informazioni / Utilizzo del browser

Sistemi Operativi

Risoluzione dei problemi / Unix Linux

PROGETTI

Design VHDL di un'unità FMA (Fused Multiply Add) ad 8 bit

Progetto sviluppato durante il corso Digital Integrated System Architectures della Laurea Magistrale in Ingegneria Elettronica. Design dell'unità in VHDL, sintesi su Vivado, valutazione delle performance ed ottimizzazioni.

Implementazione della funzione di Motion Detection sulla board Zybo Z7-20 ed ottimizzazione tramite accelerazione hardware

Progetto svolto durante il corso Digital System Programming della Laurea Magistrale in Ingegneria Elettronica. Utilizzo dell'evaluation board Zybo Z7-20 e della telecamera esterna Pcam 5C per la realizzazione di una funzione di motion detection ad alte performance grazie all'accelerazione hardware.

Manutenzione predittiva tramite reti neurali ricorrenti

Progetto svolto durante il corso di Machine Learning for Signal Processing della Laurea Magistrale in Ingegneria Elettronica. Utilizzo di reti neurali ricorrenti su Python per la predizione di possibili malfunzionamenti delle turbine degli aerei. Utilizzo di LSTM, Bi-LSTM e GRU.

Manutenzione predittiva sui motori dei treni

Progetto sviluppato durante il corso di Computational Intelligence della Laurea Magistrale in Ingegneria Elettronica. Realizzazione di una rete che, a partire dall'analisi di campioni di pressione raccolti dai compressori dei treni, permette di prevedere guasti o possibili condizioni fallimentari del motore stesso. Utilizzo di un classificatore SVM con kernel RBF ed algoritmo di ottimizzazione genetico.

Valutazione del grado rotazionale di cifre scritte a mano tramite rete neurale convolutiva

Progetto svolto durante il corso di Circuiti e algoritmi per il Machine Learning della Laurea Magistrale in Ingegneria Elettronica. Utilizzo di una rete neurale convolutiva per la risoluzione di un problema di regressione mirato alla valutazione del grado rotazionale delle cifre scritte a mano contenute nel dataset MNIST.

PUBBLICAZIONI

[Contextual Bandits Algorithms for Reconfigurable Hardware Accelerators](#)

[2022]

ApplePies, Genova, Italy, 26-27 september 2022, published by Springer, indexed in Scopus

Authors: Marco Angioli, Marcello Barbirotta, Abdallah Cheikh, Antonio Mastrandrea, Francesco Menichelli, Saeid Jamili, and Mauro Olivieri

Abstract: Reconfigurable processing cores for IoT and edge computing applications are emerging topics to calibrate costs, energy consumption and area occupation with performance and reliability on Commercial Off the Shelf (COTS) devices. This work analyzes how to take advantage of Machine Learning to potentially automate the reconfiguration process of an Hardware accelerator inside the Klessydra Vector Coprocessor Unit (VCU) [1][2][3], choosing the best configuration according to the workload. The problem is modeled with a contextual bandits approach using the Linear UCB algorithms and validated with offline Python simulations.

[Implementation of Dynamic Acceleration Unit Exchange on a RISC-V Soft-Processor](#)

[2022]

ApplePies, Genova, Italy, 26-27 september 2022, published by Springer, indexed in Scopus

Authors: Saeid Jamili, Abdallah Cheikh, Antonio Mastrandrea, Marcello Barbirotta, Francesco Menichelli, Marco Angioli, Mauro Olivieri

Abstract: Reconfigurable computing, also known as adaptive computing, exploits the reconfigurability of reprogrammable logic devices like FPGAs to perform runtime hardware reconfigurations, enabling the system to better adapt to the underlying application. By using reconfigurable computing, parts of the logic implemented on

an FPGA can be dynamically changed according to the task demands during runtime. The underlying hardware can be changed to trade off performance/power or can be modified to perform functional reconfiguration, reprogramming the behavior of a functional unit. By exploiting this flexibility, we can significantly scale the performance and power efficiency of a system. We present a dynamic acceleration unit exchange on a RISC-V soft-processor, based on the open-source Klessydra-T13 RISC-V core. We demonstrate reconfiguration for functional versatility or for improving the hardware accelerator performance, providing, as a case study, an example of how a deep neural network like VGG16 can be accelerated by using runtime reconfiguration techniques.

Automatic Hardware Accelerators Reconfiguration through LinearUCB Algorithms on a RISC-V Processor

[2023]

2023 18th Conference on Ph. D Research in Microelectronics and Electronics (PRIME)

Authors: Marco Angioli, Marcello Barbirotta, Antonio Mastrandrea, Saeid Jamili, Mauro Olivieri

Abstract: Reconfigurable processors are hardware architectures that allow for the dynamic configuration of processing resources to optimize performance and power consumption, using partial reconfiguration to modify a portion of the design or update it without affecting the entire system. In this work, we present an automatic reconfiguration technique that leverages machine learning (ML) algorithms to automatically select the optimal configuration of a general-purpose hardware accelerator according to the workload and reconfigure the architecture at run-time. The problem is formulated as a Contextual Bandit (CB) case using the Linear Upper Confidence Bound (LinearUCB) algorithms and verified using the RISC-V Klessydra family cores as a case of study.

Fault-Tolerant Hardware Acceleration for High-Performance Edge-Computing Nodes

[2023]

Electronics 2023

Authors: Marcello Barbirotta, Abdallah Cheikh, Antonio Mastrandrea, Francesco Menichelli, Marco Angioli, Saeid Jamili, Mauro Olivieri

Abstract: High-performance embedded systems with powerful processors, specialized hardware accelerators, and advanced software techniques are all key technologies driving the growth of the IoT. By combining hardware and software techniques, it is possible to increase the overall reliability and safety of these systems by designing embedded architectures that can continue to function correctly in the event of a failure or malfunction. In this work, we fully investigate the integration of a configurable hardware vector acceleration unit in the fault-tolerant RISC-V Klessydra-ft03 soft core, introducing two different redundant vector co-processors coupled with the Interleaved-Multi-Threading paradigm on which the microprocessor is based. We then illustrate the pros and cons of both approaches, comparing their impacts on performance and hardware utilization with their vulnerability, presenting a quantitative large-fault-injection simulation analysis on typical vector computing benchmarks, and comparing and classifying the obtained results. The results demonstrate, under specific conditions, that it is possible to add a hardware co-processor to a fault-tolerant microprocessor, improving performance without degrading safety and reliability.