Bayesian Optimization for Efficient Heterogeneous MPSoC based DNN Accelerator Runtime Tuning

Xuqi Zhu, Cong Gao, Sangeet Saha, Xiaojun Zhai and Klaus D McDonald-Maier
University of Essex, Colchester, United Kingdom
{xz18173, cg21670, sangeet.saha, xzhai, kdm}@essex.ac.uk

***Abstract*—With the explosive growth of Internet of Things (IoT) devices and applications, deploying Deep Neural Networks (DNNs) on resource-constrained embedded edge devices has become a popular research trend. Because such systems have limited resources, they need to rely on optimising resource utilisation to meet performance requirements. However, for scenarios where the DNN application and workloads are dynamically changing, the offline system optimisation technique cannot achieve optimal runtime performance in practical environments. Hence, in this PhD project, we propose a Bayesian Optimisation (BO)-based runtime tuning scheme for improving energy efficiency of heterogeneous MPSoC-based DNN accelerator in the context of DNN applications. By seeking suitable hardware configurations of the accelerator for dynamic DNN inference workloads ranging from 200 M to 600 M FLOPs (floating-point operations) at runtime, the recommended configuration can averagely save up to 15.33% energy consumption from a random configuration setting.**

***Index Terms*—Heterogeneous MPSoC, Bayesian optimization (BO), Partial Reconfiguration**

I. Introduction

In recent years, Deep Neural Networks (DNN) have become one of the most commonly used machine learning (ML) algorithms in a wide range of AI applications. The development towards deeper neural networks results in higher computational requirements, which make it difficult to implement large models on a resource-constrained edge device with a satisfying computing speed, memory, and energy cost requirements [1]. To overcome these challenges, pruning network structure and quantizing floating-point precision and various pruning techniques have become one of the most promising offline solutions for reducing the computation consumption of inference with acceptable accuracy loss [2].

Instead of optimising the network to meet hardware resource requirements, the other researchers consider optimising both hardware architecture and the DNN model as a "co-design" problem [3]. These techniques provide the designed hardware and DNN model pair with a trade-off considerations on performance and cost. However, this may not be an ideal solution for the edge computing systems that are desired to handle multiple DNN applications or workloads, since the optimised hardware configuration is tied to a specific DNN model, and it is difficult to flexibly alter the hardware architecture for variable workloads to achieve the optimal energy efficiency on a prepared-in-advance edge device at runtime. Hence, in this PhD project, we propose an adaptive energy efficiency searching strategy aided by partial reconfiguration technique to maximise energy efficiency for dynamic workloads on heterogeneous MPSoC-based DNN accelerators at runtime.

II. Problem Description

We aim to achieve the optimal energy consumption E by altering DNN accelerator (D) for workloads $\mathcal{W}$ that changed dynamically to optimise overall energy efficiency at runtime. Mathematically, this optimization problem is represented as:

$$\begin{cases} D_{best} = \underset{D \in \boldsymbol{D}}{\arg\min}\ E(D|\mathcal{W}) \\ \mathcal{W} \in \langle \mathcal{W}_1, \mathcal{W}_2, ..., \mathcal{W}_s \rangle \end{cases} \tag{1}$$

The Eq.(1) is to identify the optimal DNN accelerator D_{best} that can achieve the lowest energy consumption from the searching space $\boldsymbol{D}$, where the vector D_k represents the combination of $\langle area_k, freq_k \rangle$, where, $area_k$ is a tuple representing the area, including the number of FPGA resources in terms of LUT, Register, BRAM, and DSP required for D_k and $freq_k$ is the DNN accelerator input clock frequency for a given $\mathcal{W}$.

III. Proposed methodology

Fig.1 presents the experiment setup for validating the proposed runtime energy efficiency searching methodology. The main idea is to obtain the runtime performance metrics, namely execution time (i.e., latency) and power consumption donated as $\boldsymbol{t}$ and $\boldsymbol{P}$, respectively, for a range of hardware accelerators $\boldsymbol{D}$ operating on the current workload $\boldsymbol{\mathcal{W}}$, and through a closed-loop runtime searching algorithm in Processing System (PS), an optimal hardware configuration will be used for partial runtime reconfiguration of DNN accelerators in Programmable Logic (PL).

The BO algorithm adopts a surrogate function [4], $f_s(x)$, to fit the black box function $E(D|\mathcal{W}_i)$, while employing an Acquisition Function (AF), such as Upper Confidence Bound (UCB) and Thompson Sampling (TS), to guide the exploration of the search space. With each iteration, the $f_s(x)$ is updated by absorbing a new sampling points $\langle D, E(D|\mathcal{W}_i) \rangle$ to gradually approximate the true function underlying the black box system, and AF will suggest the next sampling point according to the latest $f_s(x)$.

The BO algorithm and the runtime monitoring system for reading power/latency sensors, as well as system clock configuration functions are implemented in the PS, and DNN accelerators are implemented in the PL using Deep-learning Processor Unit (i.e., DPU) IPs. To maximise the runtime

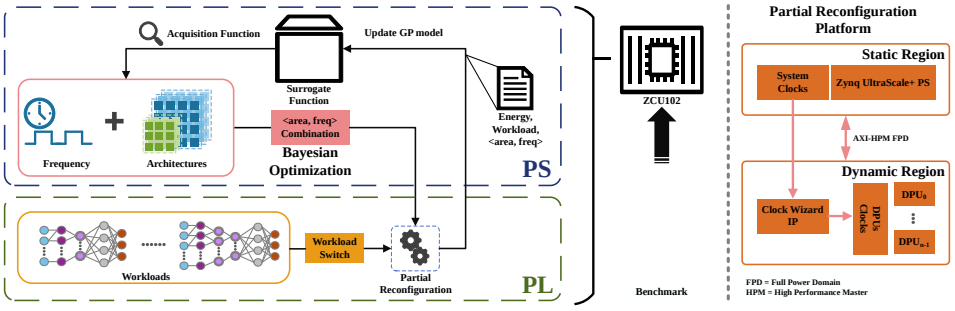

Fig. 1. Runtime adaptive energy efficiency searching strategy on MPSoC based DNN accelerator

energy efficiency of PL for the current workload, we introduced a partial reconfiguration platform to support runtime reconfiguration for DNN accelerators.

IV. Experiments and Results

Currently, 8 DPU configurations (i.e., $area$) [5] and 23 system input clock frequency (i.e., $freq$) settings ranging from 60 MHz to 108 MHz are available to construct the searching space $\boldsymbol{D}$. In order to validate the exploration efficiency of the proposed strategy, we use 3 different AFs: 'UCB', 'TS' and 'Greedy', ZCU102 board runs 200 images classification tasks on a batch of DNN models with 200 M to 600 M FLOPs workloads generated by the OFA (once for all) framework [6]. The Fig.2 illustrates that the proposed strategy can suggest a D which has $\leq 4\%$ average relative difference to the global optimal energy consumption for an unexplored new workload using only 2 iterations. This can be achieved by initializing BO with 72 sampling points $\langle D, E(D|\mathcal{W}_i)\rangle$ from the 3 explored workloads $\mathcal{W}$ and using 'Greedy' AF. In addition, Table I provides a comparison of energy consumption for examples of the unexplored workloads ranging from 200 M to 600 M FLOPs with and without applying the proposed strategy. The results demonstrate that the proposed searching strategy can averagely save energy consumption up to 15.33% from a random configuration setting.

TABLE I
Comparison of energy cost on different workloads

	Average energy cost (J) of W_i				
	283 M	**289 M**	**407 M**	**436 M**	**456 M**
Without Optimisation	6.47	6.34	7.52	9.01	8.85
With Optimisation	5.63	5.68	6.65	7.84	7.67
Average Improvement	14.86%	11.46%	13.00%	14.84%	15.33%

V. Future Work

In this PhD project, we propose a runtime adaptive strategy to improve the energy efficiency and resource utilisation of heterogeneous MPSoC-based DNNs acceleration for dynamic workloads. The experiments can preliminarily prove that the proposed strategy can efficiently optimize energy efficiency. We foresee that the proposed approach can be employed on a larger search space $\boldsymbol{D}$ with more adjustable parameters (e.g., chip voltages, fine tune network parameters) and optimisation targets such as the accuracy of DNN models and resource utilisation. Thus, we will focus on extending the search space and try to further validate the proposed strategy with more DNN models with a wider range of workloads in the future.

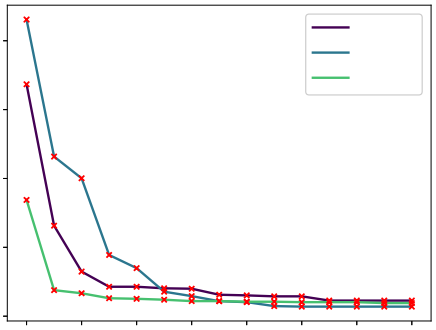

Fig. 2. Proposed strategy exploration efficiency on an unexplored workload for different AF when BO initialised by 3 explored workload.

Acknowledgement

This work is supported by the UK Engineering and Physical Sciences Research Council through grants EP/V034111/1, EP/X015955/1 and EP/V000462/1. For the purpose of open access, the author has applied a CC BY licence to any Author Accepted Manuscript version arising.

References

[1] G. Premsankar and B. Ghaddar, "Energy-efficient service placement for latency-sensitive applications in edge computing," *IEEE internet of things journal*, vol. 9, no. 18, pp. 17 926–17 937, 2022.

[2] K. S. Zaman, M. B. I. Reaz, S. H. M. Ali, A. A. A. Bakar, and M. E. H. Chowdhury, "Custom Hardware Architectures for Deep Learning on Portable Devices: A Review," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–21, 2021.

[3] C. Hao, X. Zhang, Y. Li, S. Huang, J. Xiong, K. Rupnow, W.-m. Hwu, and D. Chen, "Fpga/dnn co-design: An efficient design methodology for 1ot intelligence on the edge," in *DAC*, 2019, pp. 1–6.

[4] P. I. Frazier, "A Tutorial on Bayesian Optimization," no. Section 5, pp. 1–22, 2018.

[5] C. Gao, X. Zhu, S. Saha, K. D. Mcdonald-maier, and X. Zhai, "Modelling and Analysis of FPGA-based MPSoC System with Multiple DNN Accelerators," in *the 21st IEEE Interregional NEWCAS Conference*, 2023.

[6] H. Cai, C. Gan, T. Wang, Z. Zhang, and S. Han, "Once-for-All: Train One Network and Specialize it for Efficient Deployment," pp. 1–15, 2019.