Evolutionary-based Optimization of Hardware Configurations for DNN on Edge GPUs

Halima Bouzidi, Hamza Ouarnoughi, El-Ghazali Talbi, Abdessamad Ait El Cadi, Smail Niar

Evolutionary-based Optimization of Hardware Configurations for DNN on Edge GPUs

H. Bouzidi[1], H. Ouarnoughi[1], E-G. Talbi[2], A. Ait El Cadi and[1] S. Niar[1]

Université Polytechnique Hauts-de-France, LAMIH/CNRS, Valenciennes, France[1]
Université de Lille, CNRS/CRIStAL INRIA Lille Nord Europe[2]

Abstract. Performance and power consumption are major concerns for Deep Learning (DL) deployment on Edge hardware platforms. On the one hand, software-level optimization techniques such as pruning and quantization provide promising solutions to minimize power consumption while maintaining reasonable performance for Deep Neural Network (DNN). On the other hand, hardware-level optimization is an important solution to balance performance and power efficiency without changing the DNN application. In this context, many Edge hardware vendors offer the possibility to manually configure the Hardware parameters for a given application. However, this could be a complicated and a tedious task given the large size of the search space and the complexity of the evaluation process. This paper proposes a surrogate-assisted evolutionary algorithm to optimize the hardware parameters for DNNs on heterogeneous Edge GPU platforms. Our method combines both metaheuristics and Machine Learning (ML) to estimate the Pareto-front set of Hardware configurations that achieve the best trade-off between performance and power consumption. We demonstrate that our solution improves upon the default hardware configurations by 21% and 24% with respect to performance and power consumption, respectively.

1 Introduction and related works

Deep Neural Networks (DNN) are known for their intensive computations and memory operations. Thus, they need a careful tuning of both software and hardware, especially for resource-constrained Edge platforms. Modern Edge Graphical Processing Unit (GPU) accelerators provide outstanding performances for Deep Learning (DL) applications [1]. Nevertheless, this comes at the cost of considerable power consumption. Adjusting hardware parameters such as processing cores and operating frequencies according to the DNN execution requirements, represents a different way to improve performance and power efficiency. However, it is hard to decide the best Hardware configuration because of the heterogeneous complexity of the GPU architecture and the wide range of possible configurations. The contradictory nature of the two objectives, increasing performance and decreasing power consumption, makes the optimization even more complex. Hence, this issue can be formulated as a multi-objective optimization problem where we search for an optimal Pareto set of hardware configurations that achieve the best trade-off between the two objectives for a given DNN application. This paper proposes a surrogate-assisted multi-objective optimization that incorporates both Machine Learning (ML) and metaheuristics to approximate an optimal Pareto set of hardware operating frequencies for DNNs on Edge GPU accelerators. The resulted Pareto set will help the user to choose adequate operating frequencies according to the application requirements and system budget constraints.

Some works have been proposed in the literature to address the hardware tuning issue in heterogeneous GPUs. Authors in [2] propose a prediction model based on Support Vector Regression (SVR) for power consumption of GPU kernels for different GPU core and memory frequencies. In [3] and [4], the authors propose a cross-domain modeling approach for power consumption that models both the application and the GPU micro-architecture under variable GPU core and memory frequencies. [5] conducts an empirical study on the impact of frequency scaling on performance and energy consumption of DNNs training and inference on high-performance GPUs. This study shows that GPU DVFS has a significant improvement on both performance and energy consumption of DNNs. [6] proposes a ML based prediction methodology for performance and power consumption of OpenCL kernels on GPU platforms. They combine the two prediction models to approximate a Pareto-set of frequency configurations on GPUs. Where the works mentioned above only focus on tuning GPUs, [7] considers both CPU and GPU tuning in heterogeneous devices. However,

the authors use neither prediction models nor optimization algorithms. They rely on empirical observations of profiling results, which may lead to sub-optimal solutions.

2 Problem formulation

Given a fixed DNN application and Edge GPU platform, adjusting the hardware parameters can be formulated as a multi-objective optimization problem where we search for the optimal hardware configurations that provide the best trade-off between performance and power consumption. Let $X = \{x_1, x_2, \ldots, x_n\}$ be a set of hardware configurations, where each x_i represents one instance of the hardware operating frequencies. For instance, a x_i can represent the frequency value of a CPU, GPU cores or memory. Let $F = (f_1, f_2)$ be a vector of objectives to minimize, where $f_i \in \{\text{execution\_time, power\_consumption}\}$. A real evaluation of these objectives is a tedious and time-consuming task. Thus, instead of directly measure F on the Hardware platform, we rely on prediction functions as surrogate-models for F that we denote $\hat{F}$. Our problem is defined as follows:

$$MOP = \begin{cases} \min \hat{F(x)} = (\hat{f}_1(x), \hat{f}_2(x)) \\ s.t. \quad x \in X \end{cases} \tag{1}$$

In this paper, we study the case of optimizing the hardware configurations of a modern Edge GP-GPU platform: NVIDIA Jetson AGX Xavier [8] for a state-of-the-art DNN: AlexNet [9]. We tune four hardware parameters: CPU, GPU, PVA, and memory frequencies. We set the lower and upper bounds for each parameter according to the reported minimum and maximum values in the configuration file of Jetson Tegra system [10].

3 Proposed Approach

We propose a surrogate-assisted evolutionary algorithm that leverages both metaheuristics and Machine Learning. We speed up the optimization process using ML-based prediction models to estimate $\hat{F}$. Our proposed methodology is composed of two main steps:

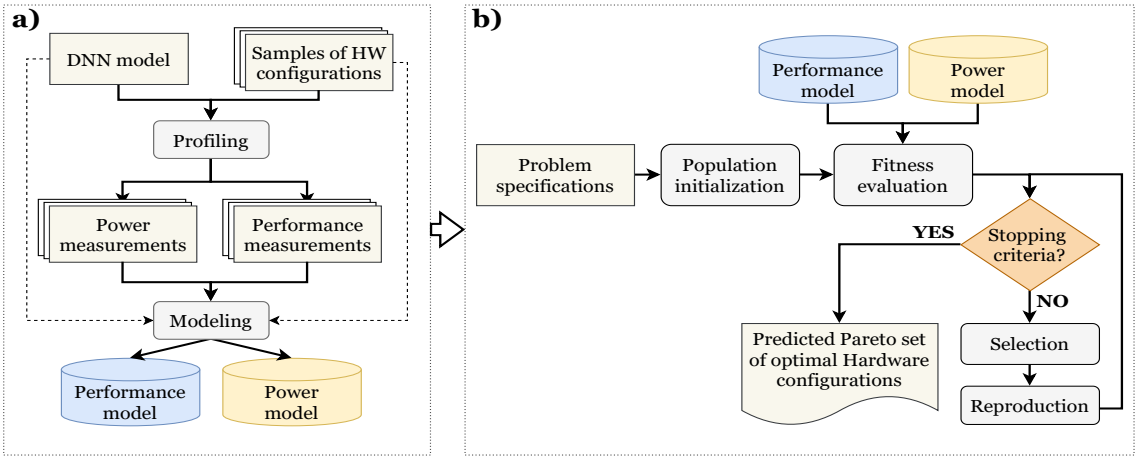

Fig. 1: Overview of the proposed methodology: **a)** corresponds to the training phase of the prediction models for performance and power consumption. **b)** depicts the optimization phase of the hardware parameters using both the trained prediction models and evolutionary-based multi-objective metaheuristic

a. Prediction models training: The training phase is illustrated in figure 1.a. First, we collect training data by profiling the DNN application on randomly sampled Hardware configurations. We denote the resulting training datasets for performance by $D_l = \{(x_1, l1), (x_2, l_2), \ldots, (x_n, l_n)\}$ and for power consumption by $D_p = \{(x_1, p1), (x_2, p_2), \ldots, (x_n, p_n)\}$, where l_i and p_i refer to the measured values of performance and power consumption, respectively, under the hardware configuration x_i. Second, we train SVR-based prediction models for performance and power consumption

on D_l and D_p. The trained prediction models are used in the optimization step for a rapid evaluation. The following prediction models are defined:

$$\begin{cases} M_{performance}(D_l) = \sum_{j=1}^{n}(\alpha_j - \hat{\alpha_j})K_{RBF}(d_{jl}, D_l) + b \\ M_{power}(D_p) = \sum_{j=1}^{n}(\alpha_j - \hat{\alpha_j})K_{RBF}(d_{jp}, D_p) + b \end{cases} \quad (2)$$

where b, α_j, and $\hat{\alpha_j}$ refer to the bias and training coefficients of the trained instance of SVR. K_{RBF} is a *radial basis kernel function*. D_l, D_p are the datasets used to train the prediction models for performance and power consumption, respectively.

b. Optimization: Figure 1.b gives an overview of the optimization phase. To efficiently explore the search space of the hardware configurations, we implement MOEA/D, a decomposition-problem-based metaheuristic, as a multi-objective evolutionary optimization algorithm. It uses different evolutionary operators to combine good solutions of neighboring problems, resulting in quick and accurate convergence. We adapt MOEA/D for our problem by leveraging the normalization technique as both performance and power consumption have different scales. We choose the Tchebycheff method as a problem decomposition technique. To generate an ensemble of uniformly distributed weight vectors, we use the Das and Dennis technique. The trained prediction models from figure 1.a are used to evaluate the fitness in the MOEA/D algorithm.

4 Experimental Results

Figures 2 and 3 provide an overview of the estimated Pareto front and set by our proposed method. In figure 2, the blue points represent the predicted Pareto front, while orange ones report the measured values of performance and power consumption of the Pareto front. The seven default hardware configurations of NVIDIA Jetson AGX GPU are marked with the other point types and colors.

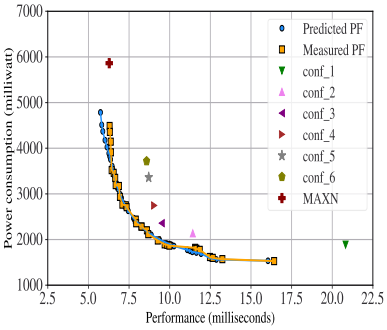

Fig. 2: Predicted vs measured PF

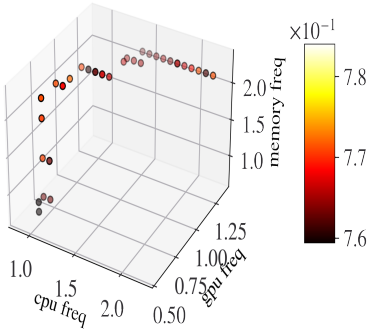

Fig. 3: Obtained Pareto set

Metric	Performance	Power
MAPE	4.21%	5.75%
RMSPE	4.83%	6.69%
Kendall's tau	0.971	0.970

Table 1: Prediction errors and rank order persevering

Figure 2 shows that in addition to the small gap between predictions and measurements, our approach gives configurations that dominate the default suggested NVIDIA configurations. Details of the prediction errors are given in table 1. Obtained MAPE and RMSPE values are small for both performance and power predictions. Moreover, the rank order is highly respected between predicted and measured metrics according to the reported Kendall's τ coefficients in table 1. For configurations that give high performance and low power consumption, we have obtained a configuration with the same performance as the MAXN power mode of NVIDIA with a power-saving of 24%. Similarly, for performance, we have obtained a configuration that gives similar power consumption as the minimum power mode suggested by NVIDIA (i.e., conf_1), with a performance gain of 21%. Figure 3 presents the Pareto set in the decision space. We notice that most configurations maximize the memory frequency. This is explained by the architecture of AlexNet that holds a large number of parameters, which results a high memory activities. This also corroborate our motivation to adjust the hardware configuration according to the DNN requirements.

5 Conclusion

In this paper we introduced a multi-objective optimization approach that leverages both metaheuristics and Machine Learning to optimize the Hardware configurations for Deep Neural Networks on GPU heterogeneous accelerators. The optimization approach incorporates prediction models for approximating the fitness functions to speed up the evaluation of the sampled configurations by the optimization algorithm. Experimental results on AlexNet and Jetson AGX Xavier GPU demonstrated that a higher accurate prediction and a more energy-efficient configurations that outperform the predefined ones can be obtained. As a future work, we plan to develop a cross-surrogate-based multi-objective optimization approach that models both DNN architecture and Hardware configuration. We also propose to enhance the optimization process by injecting the knowledge on the execution requirements of the DNN.

References

1. Hassan Halawa, Hazem A Abdelhafez, Andrew Boktor, and Matei Ripeanu. Nvidia jetson platform characterization. In *European Conference on Parallel Processing*, pages 92–105. Springer, 2017.
2. Qiang Wang and Xiaowen Chu. Gpgpu power estimation with core and memory frequency scaling. *ACM SIGMETRICS Performance Evaluation Review*, 45(2):73–78, 2017.
3. Joao Guerreiro, Aleksandar Ilic, Nuno Roma, and Pedro Tomas. Gpgpu power modeling for multi-domain voltage-frequency scaling. In *IEEE International Symposium on High Performance Computer Architecture*, pages 789–800. IEEE, 2018.
4. João Guerreiro, Aleksandar Ilic, Nuno Roma, and Pedro Tomás. Modeling and decoupling the gpu power consumption for cross-domain dvfs. *IEEE Transactions on Parallel and Distributed Systems*, 30(11):2494–2506, 2019.
5. Zhenheng Tang, Yuxin Wang, Qiang Wang, and Xiaowen Chu. The impact of gpu dvfs on the energy and performance of deep learning: An empirical study. In *10th ACM International Conference on Future Energy Systems*, pages 315–325, 2019.
6. Kaijie Fan, Biagio Cosenza, and Ben Juurlink. Predictable gpus frequency scaling for energy and performance. In *48th International Conference on Parallel Processing*, pages 1–10, 2019.
7. Ourania Spantidi, Ioannis Galanis, and Iraklis Anagnostopoulos. Frequency-based power efficiency improvement of cnns on heterogeneous iot computing systems. In *2020 IEEE 6th World Forum on Internet of Things (WF-IoT)*, pages 1–6. IEEE, 2020.
8. Jetson AGX xavier developer kit. https://developer.nvidia.com/embedded/jetson-agx-xavier-developer-kit. Accessed: 2021-02-01.
9. Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.
10. Jetson developer kits and modules. https://docs.nvidia.com/jetson/l4t/. Accessed: 2021-02-01.