Chariot: Compiler-Aware Heterogeneous Graph Representation Learning for Automated HLS Optimization

Yanlong Huang*, Jierui LIU*, Yuhan She*, Rongliang Fu†, Tsung-Yi Ho†, Hong Yan*, Ray C.C. Cheung*
*City University of Hong Kong †The Chinese University of Hong Kong
Emails: {yanlhuang5-c, jieruiliu2-c, yuhanshe3-c}@my.cityu.edu.hk,
{h.yan, r.cheung}@cityu.edu.hk, {rlfu, tyho}@cse.cuhk.edu.hk

***Abstract*—High-level synthesis (HLS) design space exploration (DSE) aims to find Pareto-optimal designs but is hindered by slow synthesis evaluations. Existing graph neural network (GNN) surrogates struggle with homogeneous-style graph representations (causing signal over-squashing) and imprecise source-level heuristics for pragma mapping. We propose Chariot, an automated HLS optimization framework. Chariot leverages LLVM-based static analysis for high-fidelity Use-Def chain tracking, modeling HLS designs as semantic-rich heterogeneous graphs that explicitly map directives to true hardware targets. Our framework achieves state-of-the-art QoR prediction, identifying Pareto-optimal solutions with drastically reduced ranking regret while delivering orders-of-magnitude DSE speedup.**

I. INTRODUCTION

HLS Design Space Exploration (DSE) identifies Pareto-optimal hardware designs by tuning compiler directives (pragmas), but exhaustive synthesis-in-the-loop evaluation is prohibitively expensive.

While Graph Neural Networks (GNNs) [1], [2] serve as fast Quality of Result (QoR) surrogates, existing methods predominantly use homogeneous-style graph representations. This obscures the semantic differences between instructions, variables, and pragmas, causing signal "over-squashing". Furthermore, they rely on imprecise source-level heuristics (e.g., line-number matching) to map pragmas to hardware entities. This fails for memory optimizations (e.g., `array_partition`) whose hardware effects are realized solely through compiler intermediate representations (IR). To bridge this gap, we propose **Chariot**, a framework constructing compiler-aware heterogeneous graphs for highly accurate QoR prediction and ultra-fast DSE.

II. METHODOLOGY

Compiler-Aware Graph Construction: To overcome source-level imprecision, Chariot compiles C/C++ into LLVM IR and performs deep static analysis via a custom Pass. By tracking Use-Def chains across intermediate address calculations, it precisely identifies the downstream `load`/`store` instructions accessing specific memories. This tracking explicitly links pragmas to their verified hardware targets via dedicated `applies_to` edges. Consequently, Chariot builds a heterogeneous graph with three primary node types (`Instruction`, `Variable`, `Pragma`). Crucially, `applies_to` edges encode pragma attributes (category, factor, style) as dense numerical features to prevent semantic dilution.

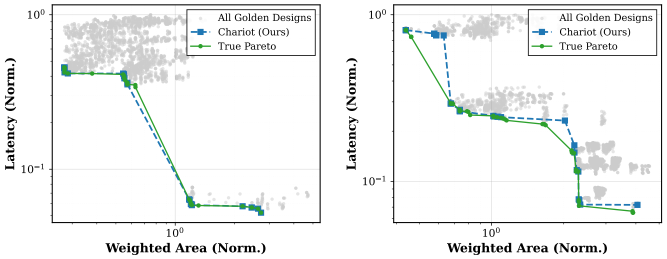

Fig. 1. Pareto frontier approximation on unseen kernels *atax* (left) and *viterbi* (right). Chariot closely tracks the true Pareto optimal sets.

Heterogeneous GNN Architecture: Chariot employs a multi-task Heterogeneous GNN using Graph Isomorphism Network with Edge features (GINEConv) layers. This native edge-feature integration allows the model to differentiate compiler dependencies from optimization-specific relations, modulating messages via the rich numerical semantics stored on the `applies_to` edges.

Surrogate-Driven DSE: The trained H-GNN acts as a fast surrogate for unseen kernels. Chariot reuses a baseline compiler graph and dynamically injects pragma semantics via lightweight in-memory graph mutation. This strategy avoids repeated compilation overheads, transforming expensive HLS evaluations into a rapid candidate ranking process.

III. CONCLUSION AND ONGOING WORK

Chariot elevates directive-target interactions into a first-class heterogeneous graph relation guided by LLVM analysis. Ongoing evaluations on multi-suite datasets aim to validate its superior predictive fidelity and its ability to drastically reduce DSE search time.

REFERENCES

[1] A. Sohrabizadeh, Y. Bai, Y. Sun, and J. Cong, "Automated accelerator optimization aided by graph neural networks," in *ACM/IEEE DAC*, 2022.
[2] H. Kuang, X. Cao, J. Li, and L. Wang, "HGBO-DSE: Hierarchical GNN and bayesian optimization based HLS design space exploration," in *IEEE ICFPT*, 2023.