REASONING COMPILER: LLM-Guided Optimizations for Efficient Model Serving

Paper Github Author

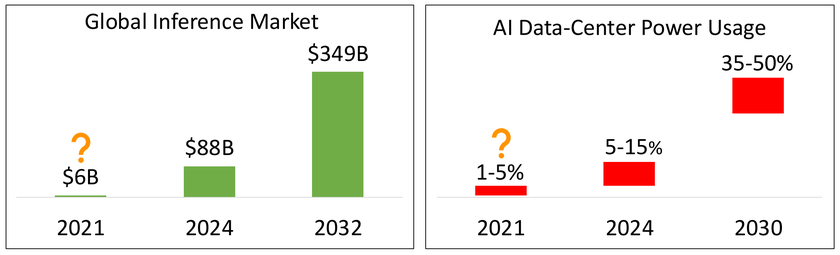
Annabelle Sujun Tang, Christopher Priebe, Rohan Mahapatra, Lianhui Qin, Hadi Esmaeilzadeh

Alternative Computing Technologies **(ACT)** Lab, University of California San Diego

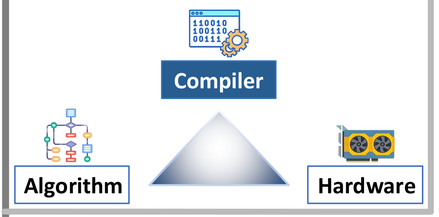
NEURAL INFORMATION PROCESSING SYSTEMS

The Current Landscape

The Shift towards Model Serving

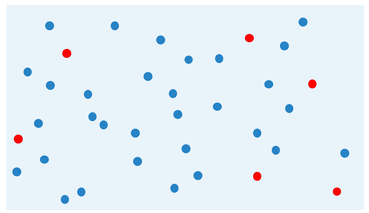


The Triangle of Improvement



Two Categories of Existing Compiler Optimizations

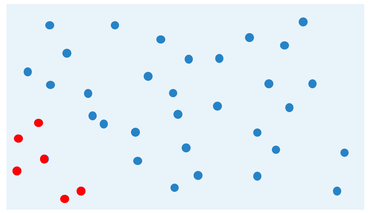
Category II: Stochastic Search for Compilation



1. STOKE (Stochastic Super Optimization)
   - Markov Chain Monte Carlo (MCMC)
   - High quality programs often lie in regions separated by low-probability paths
2. TVM, Ansor, FlexTensor, Tensor Comprehensions
   - Genetic Algorithm
   - Simulated Annealing

+ Find Higher Quality Optimized Programs
– Sample Inefficient
– Cannot leverage context and interdependence

Category I: Rule-Based Compiler Optimization

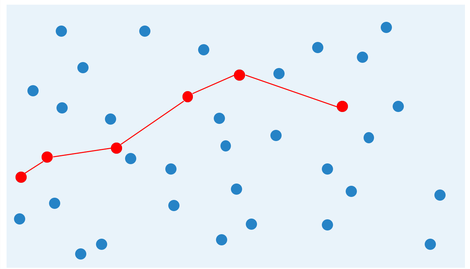


1. Relies on hand-tuning or domain-specific heuristics
2. Often overfit to a specific workload or hardware target

+ Relatively Fast
- Cannot explore the entirety of search space

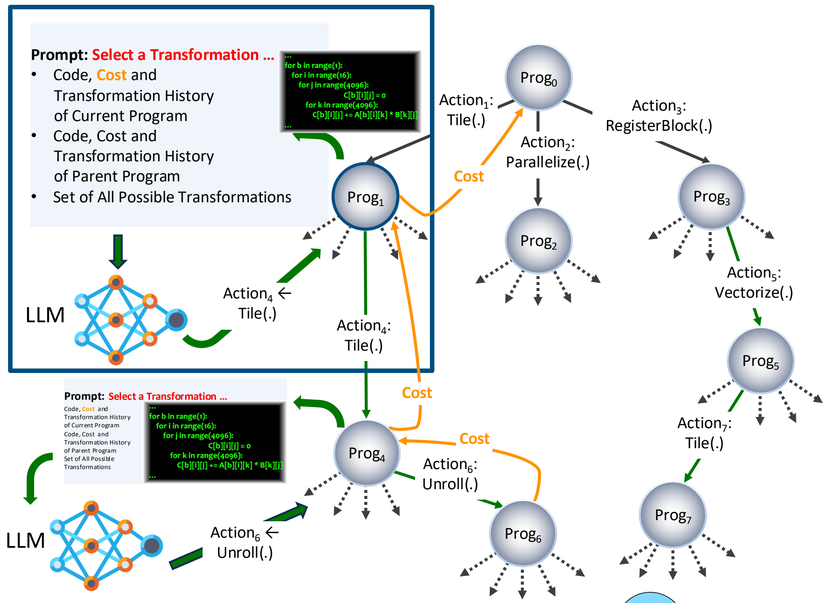
Our Innovation

REASONING COMPILER: Context-Informed, Guided, Structured Search



1. Leaps from fully stochastic to structured optimizations
2. Models optimization as Markov Decision Process
3. Uses Monte Carlo Tree Search (MCTS) as a planner
4. Utilizes LLM reasoning as a guide for Monte-Carlo Tree Search

+ Finds Higher Quality Optimized Programs + Sample Efficient



Empirical Evaluation

Diversity of Workloads and Hardware Platforms

- Self-Attention Layer from Llama-3-8B
- Mixture-of-Experts Layer from DeepSeek-R1
- Self-Attention Layer from FLUX (Stable Diffusion)
- Convolution Layer from FLUX (Stable Diffusion)
- MLP Layer from Llama-4-Scout
- End-to-End Llama-3-8B

- Amazon Graviton2
- AMD EPYC 7R13
- Apple M2 Pro
- Intel Core i9
- Intel Xeon E3

Higher Speedup with Fewer Samples

Across all 25 layer-platform pairs, **REASONING COMPILER** achieves a 5.0× **speedup** with 5.8× **fewer samples**: 10.8× improvement in **sample efficiency** over Evolutionary Search.

Consistent End-To-End Improvements

Hardware Platforms	Evolutionary Search		Reasoning Compiler		Improvement	
	# Samples	Speedup	# Samples	Speedup	Sample Reduction	Sample Efficiency Gain
Amazon Graviton2	4,560	3.7×	1,440	5.1×	3.2×	4.4×
AMD EPYC 7R13	410	2.0×	140	2.2×	2.9×	3.2×
Apple M2 Pro	4,820	2.2×	1,770	3.9×	2.7×	4.8×
Intel Core i9	3,800	2.2×	720	4.9×	5.3×	11.8×
Intel Xeon E3	4,640	5.0×	670	5.0×	6.9×	6.9×
Geomean	–	**2.8×**	–	**4.0×**	**3.9×**	**5.6×**

For Llama-3-8B, **REASONING COMPILER** achieves a **4.0× speedup** using **3.9× fewer samples**, achieving **5.6× sample efficiency** over Evolutionary Search.