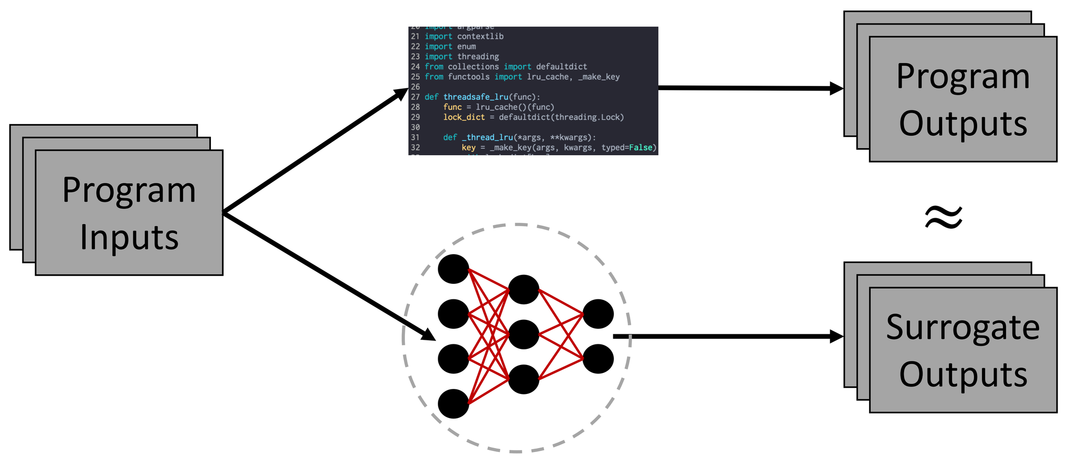
Programming with Neural Surrogates of Programs

Alex Renda, Yi Ding, Carbin
{renda, ding1, mcarbin}@csail.mit.edu

Surrogates of Programs

- Models of the behavior of program
- Implemented with machine learning models (e.g., neural networks)
- Trained with input-output examples of the program

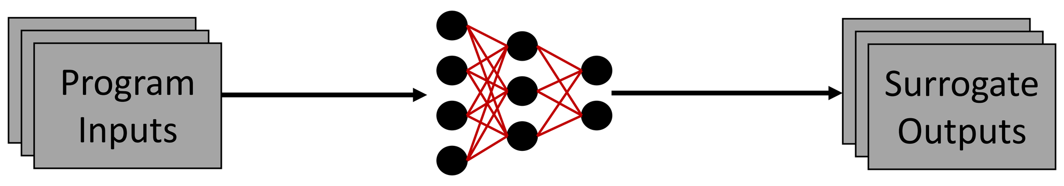


Surrogate Construction: developing a surrogate of a program

Surrogate-Based Design Patterns

Surrogate Compilation

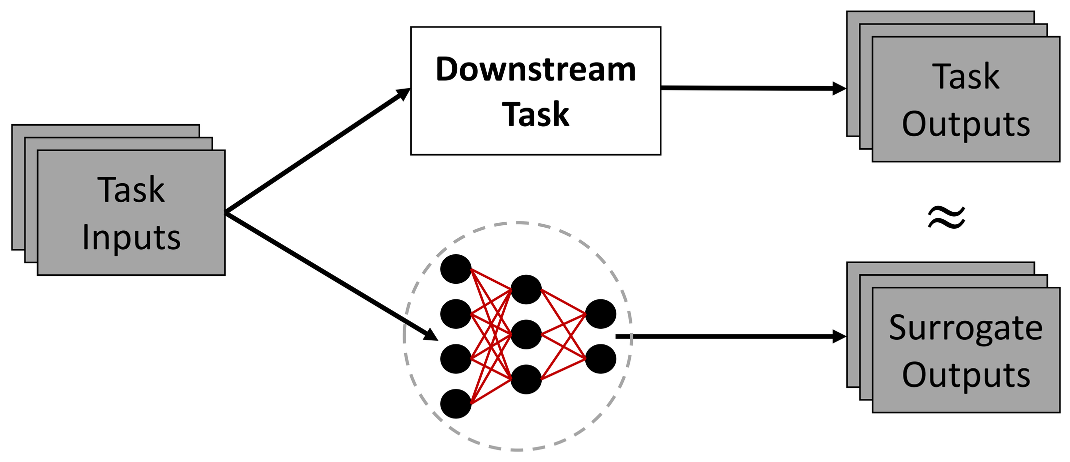
1. Deploy the surrogate to end-users



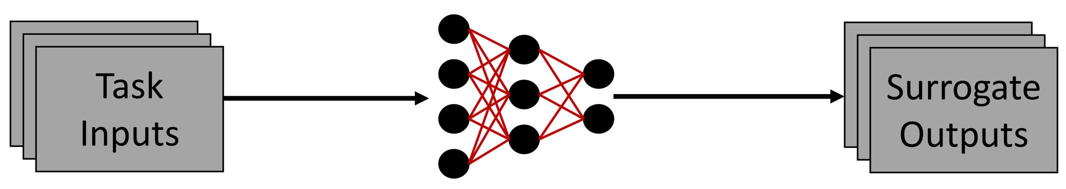
Surrogate executes faster than the program:
- Surrogate implementation can be more optimized
- Surrogate can execute on different hardware
- Surrogate can have different complexity

Surrogate Adaptation

1. Fine-tune the surrogate on a downstream task



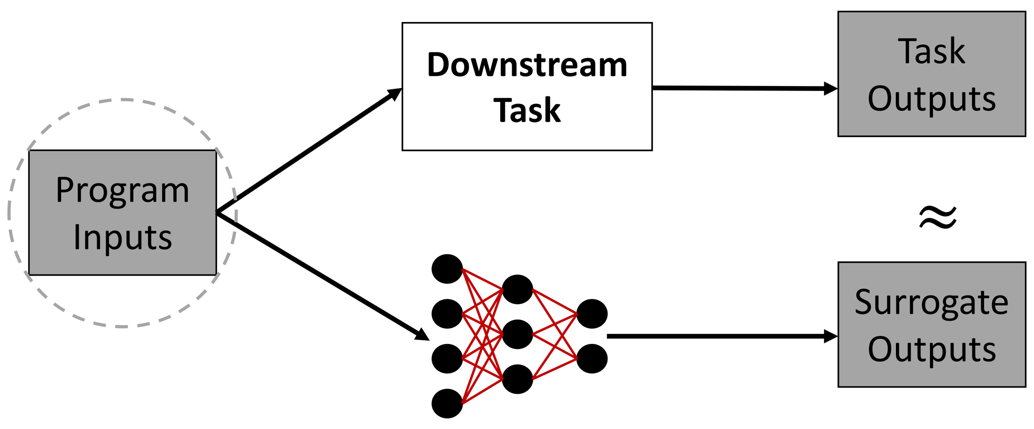
2. Deploy the surrogate to end-users



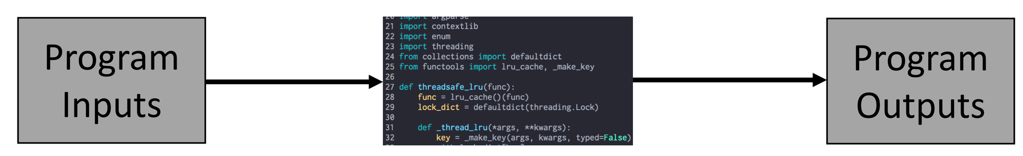
Changes the semantics of the program to accomplish a downstream task:
- Surrogate is more accurate than the program on the downstream task
- Requires less data than training a network from scratch on the downstream task

Surrogate Optimization

1. Optimize inputs of the surrogate



2. Plug the inputs back into the original program



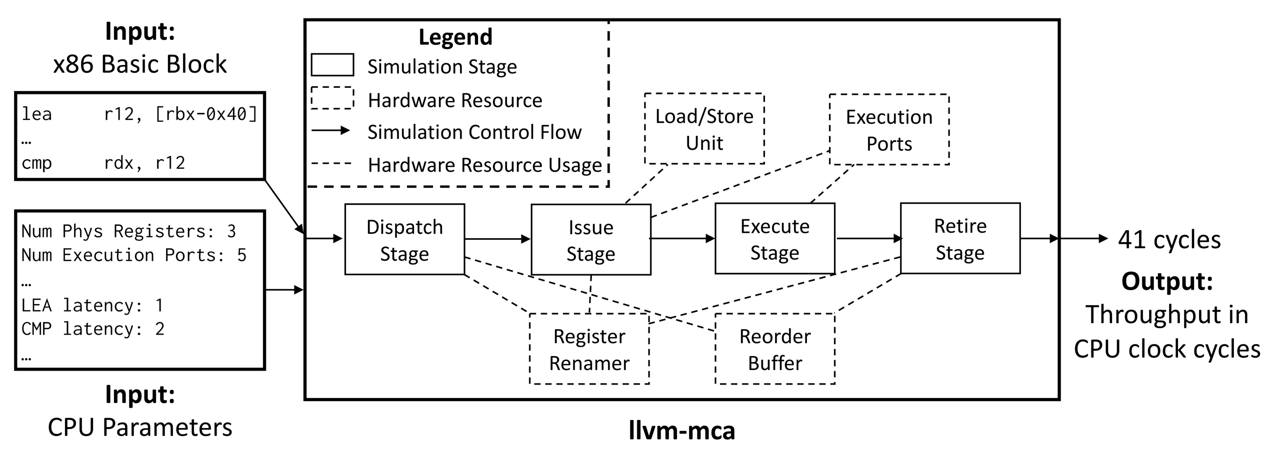
Optimizes inputs faster than optimizing against the original program:
- Surrogate is differentiable allowing for using gradient descent
- Surrogate can execute faster than the program

Case Study: llvm-mca

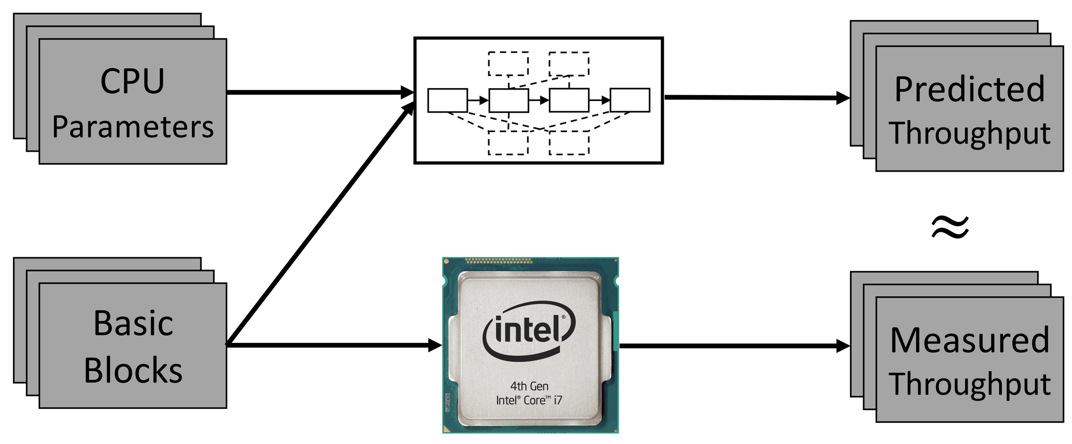
CPU simulator included with the LLVM compiler.

Input: x86 basic blocks
Input: CPU simulation parameters
Output: Throughput of the basic block

- 10,000+ line-of-code C++ program
- 11,265 parameters to set
- >25% error against ground-truth timings



10KLoC C++ program with 11,265 integer-valued parameters



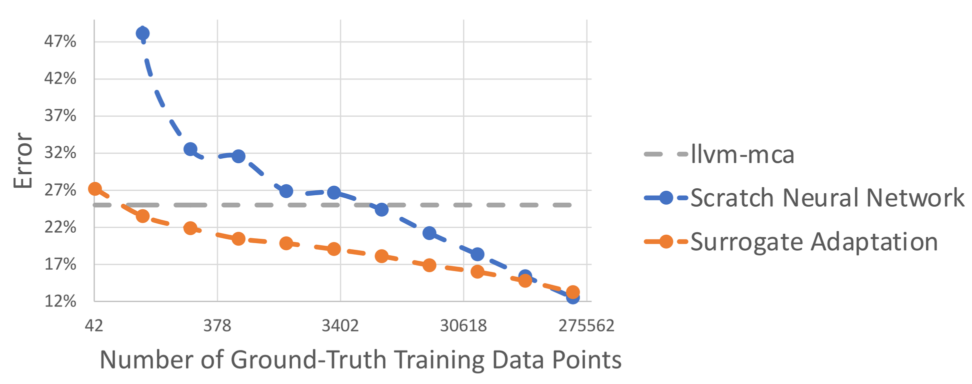
Surrogate Compilation

Accelerates llvm-mca's execution speed by 1.6x with <10% loss in accuracy

Approach	Execution Speed	Error
llvm-mca -O3	1742 BBs/second	25.0%
Surrogate	2820 BBs/second	27.1%

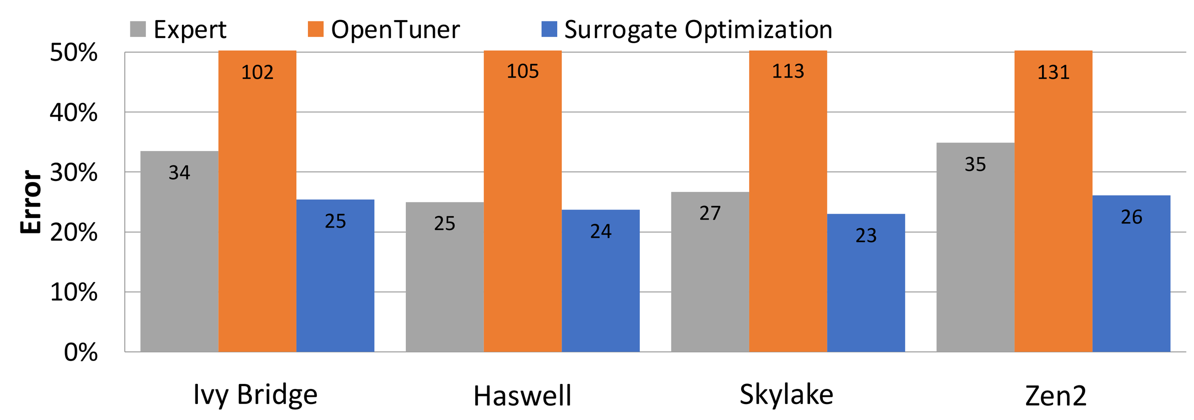
Surrogate Adaptation

Decreases llvm-mca's error by up to 50% with less data than training a network from scratch



Surrogate Optimization

Finds simulation parameters that decrease llvm-mca's error relative to expert-set parameters given the same budget as a surrogate-free technique, Opentuner



Neural Surrogate Programming Methodology

Design

- What neural network architecture topology does the surrogate use?
- How do you scale the surrogate's capacity to represent the original program?

Training

- What training data does the surrogate use?
- What loss function does the surrogate use?
- How long do you train the surrogate?

Deployment

- What hardware does the surrogate use?
- What software execution environment does the surrogate use?