PolyTOPS: Configurable and Flexible Polyhedral Scheduler

Gianpietro Consolaro[1,2], Corinne Ancourt[2], Zhen Zhang[1], Cedric Bastoul

1 Huawei Technologies France, Paris.

2 MINES Paris - PSL University

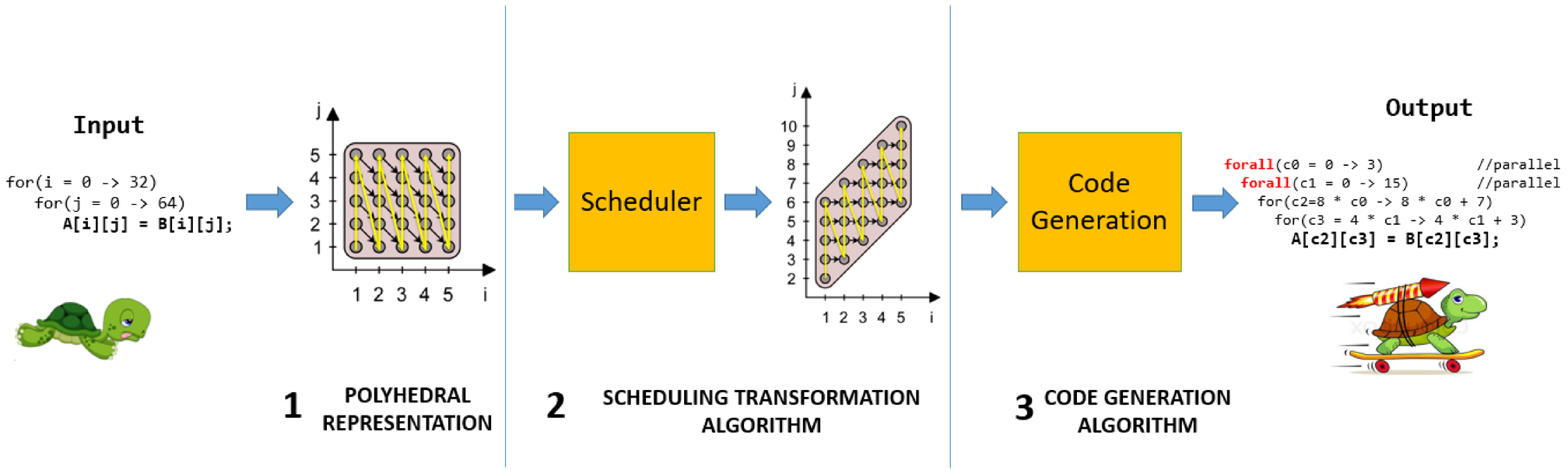
gianpietro.consolaro@mines-paristech.fr

Introduction

Polyhedral Optimization is used to apply loop transformations automatically, **minimizing the execution time** of the input program. It has become increasingly important in **AI applications** to automatically generate more efficient applications, and is widely used across various domains.

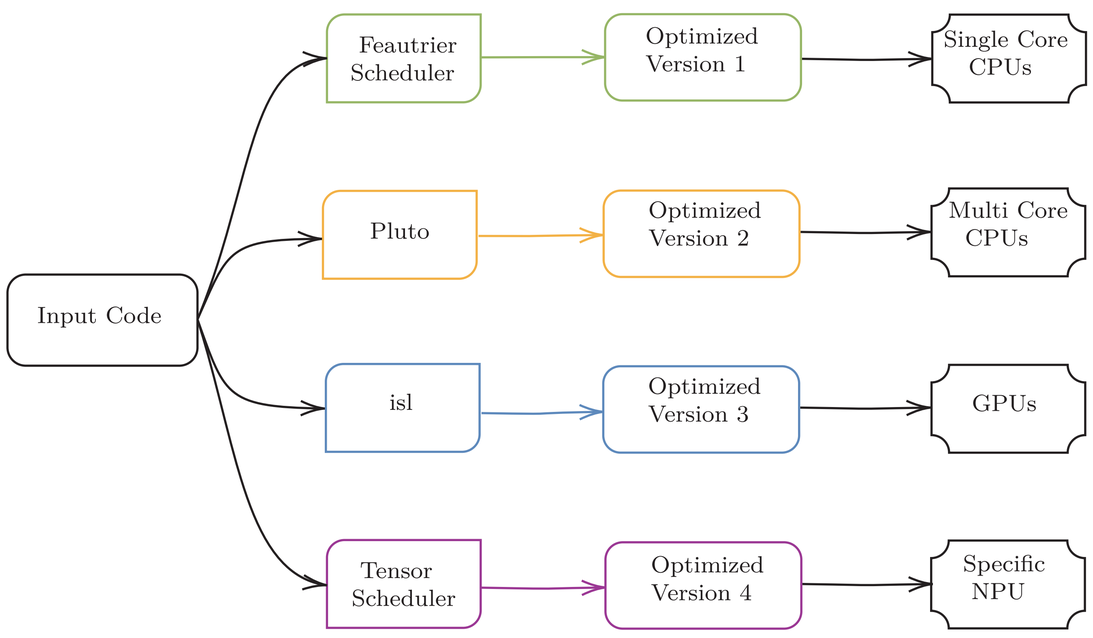
Polyhedral Optimization

Polyhedral Optimization is a compiler technique used for automatic **loop transformation**, taking care of **semantic preservation**, optimizing **cache locality** and extracting **parallelism** to reach better performances. Here we can see an example of a polyhedral optimization process.



State-of-the-Art

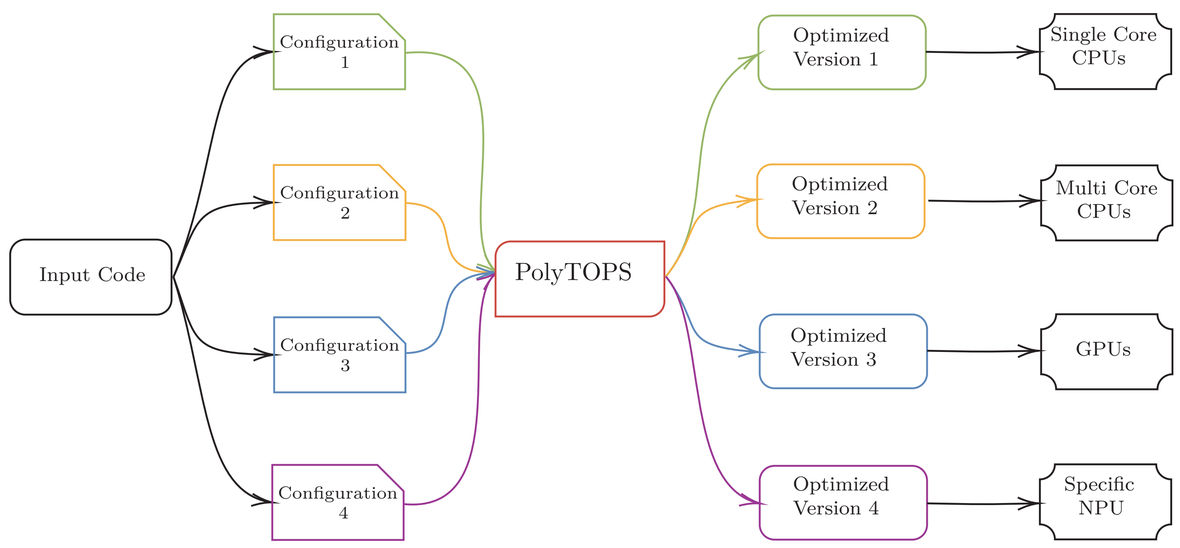
State-of-the-art polyhedral schedulers are **designed** with a specific strategy, **for specific scenarios and architectures**. They are seen as **black-box** schedulers because the optimization criteria entirely rely on an **internal heuristic** that cannot be changed.



For each different scenario, we have a specific scheduler.

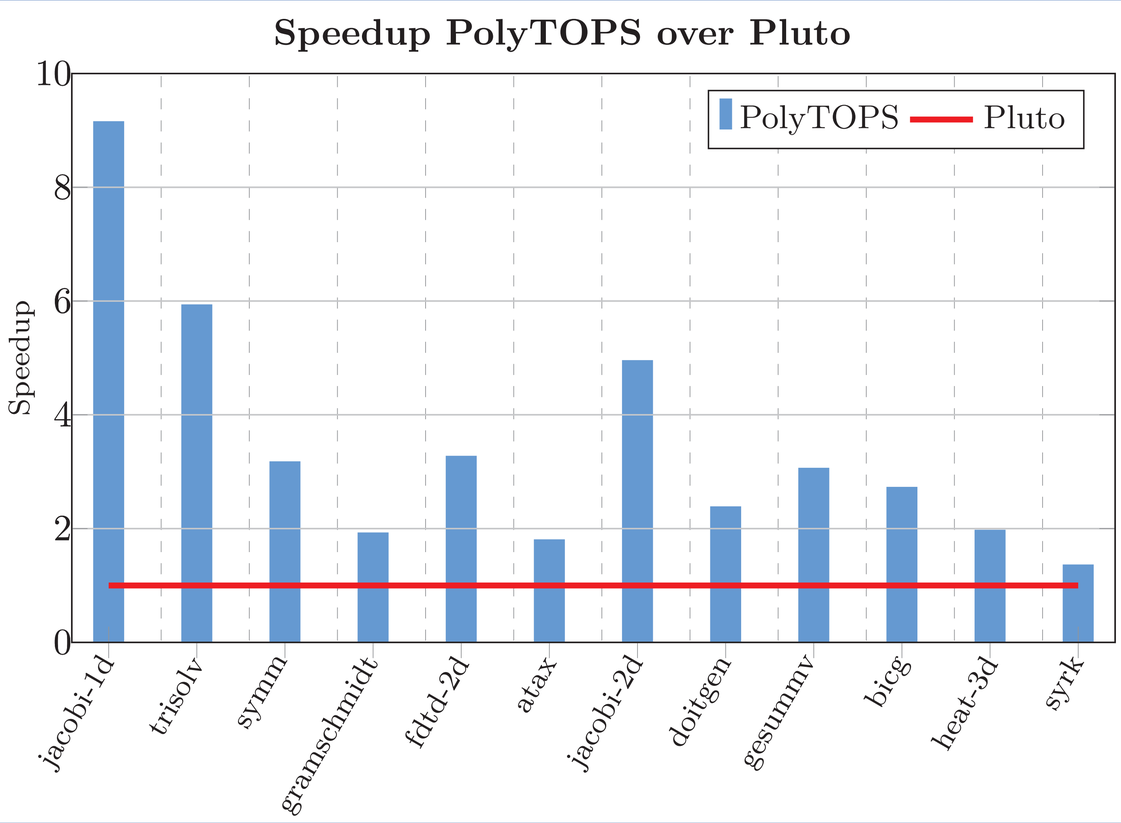
PolyTOPS

PolyTOPS is a polyhedral scheduler configurable according to a simple design (around 20 lines of JSON code). Through it, PolyTOPS can **achieve the same results** as previous **state-of-the-art schedulers**, adding the ability to **design new strategies**.

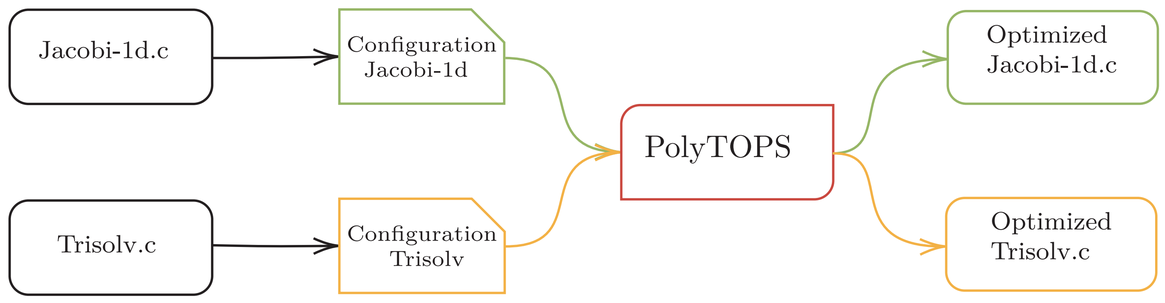


PolyTOPS is **configurable**, and **flexible** enough to adapt to possible **new scenarios**!

Speedup Comparison



Our experiments focus on the **Polybench** benchmark. The results show the speedup of PolyTOPS loop transformation compared to the Pluto ones. In PolyTOPS we designed a **kernel-specific configuration for each case**, while Pluto uses a generic strategy.



PolyTOPS obtained a **geomean speedup** of **1.49** for the Polybench cases over Pluto. The tests have been run on an Intel machine (**Intel Xeon E5-2683, 2 sockets, 16 cores each, 2 threads per core**) using **gcc-10.5** as compiler.

PolyTOPS configurability can be used to easily design new generic strategies (targeting new architectures) or to apply kernel-specific optimizations.

PolyTOPS: Configurable and Flexible Polyhedral Scheduler, submitted to CGO24