Autotuning OpenCL Workgroup Sizes

Tuning GPU Stencils with machine learning outperforms human experts

Chris Cummins Pavlos Petoumenos Michel Steuwer Hugh Leather
c.cummins@ed.ac.uk

3.79x speedup!

Predicting OpenCl workgroup sizes of 429 stencil programs, execution devices, and datasets.

Hand tuning programs is expensive and time consuming

We automate this tuning using collaborative machine learning

Stencil Skeletons

Stencil Skeletons are a common data parallel pattern with a range of applications from image processing to partial differential equations and cellular automata.
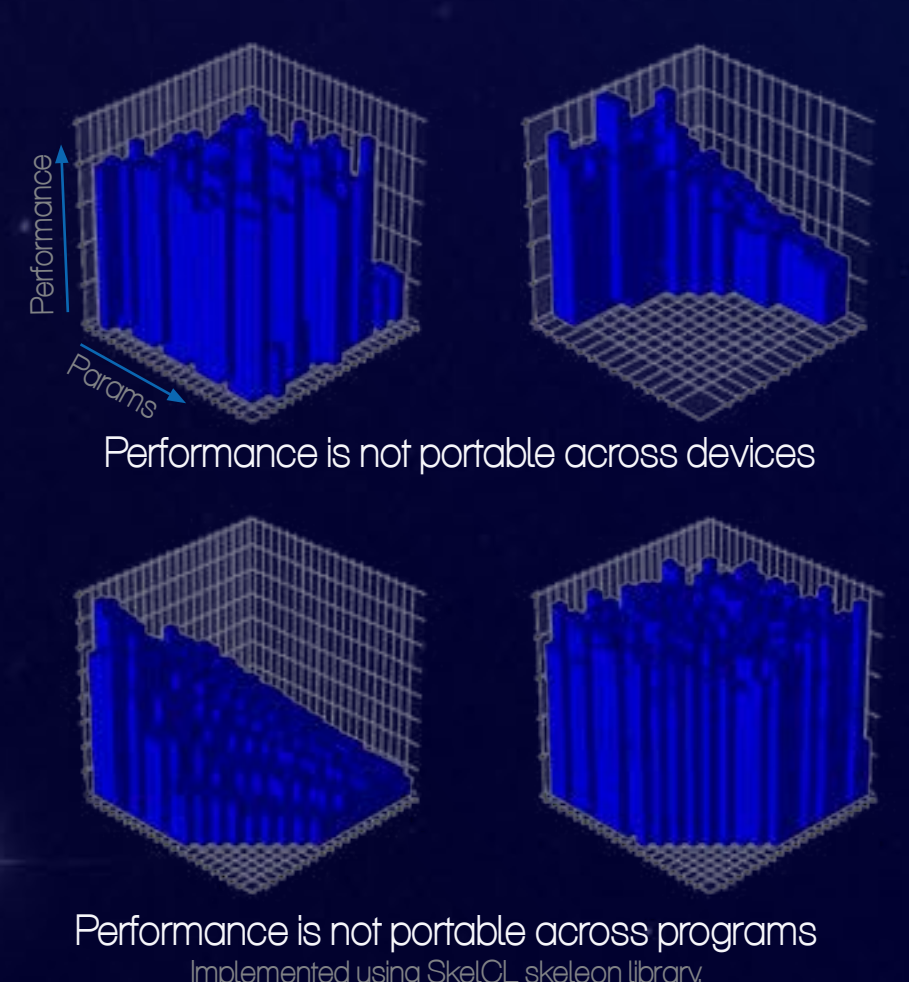
Input → Stencil → Output
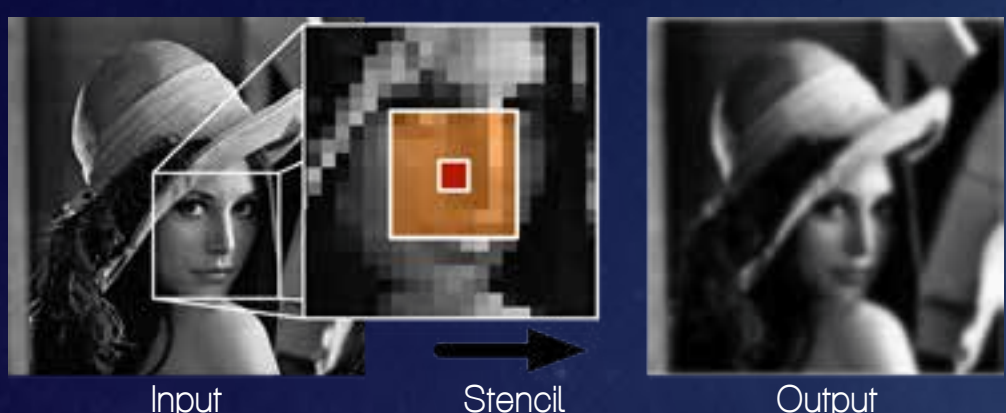
OpenCL Workgroup Size

OpenCL stencil skeletons are parameterised with a workgroup size, which controls grouping of hardware threads and local memory utilisation.

- Border region
- Work-item
- Workgroup
- Tile
- Matrix
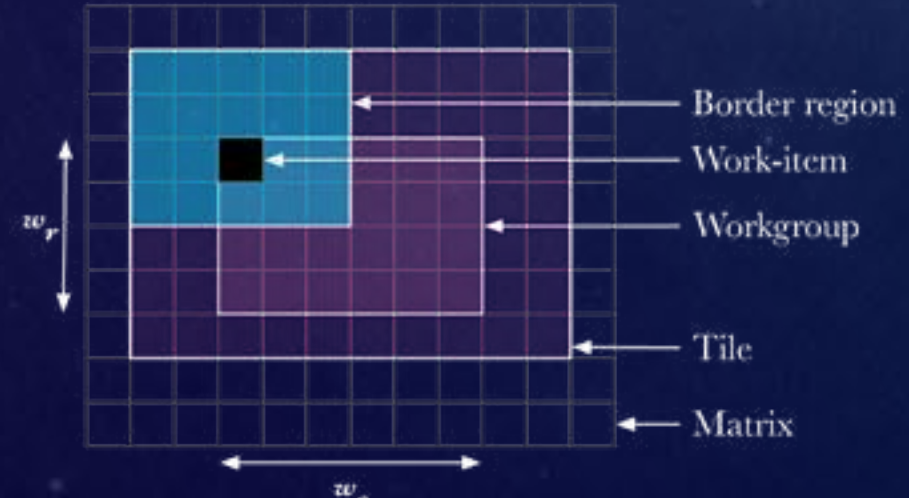
Optimization Space

Choosing the right OpenCL workgroup size for stencil kernels depends on the program, device and dataset:

Performance is not portable across devices

Performance is not portable across programs
Implemented using SkelCL skeleton library
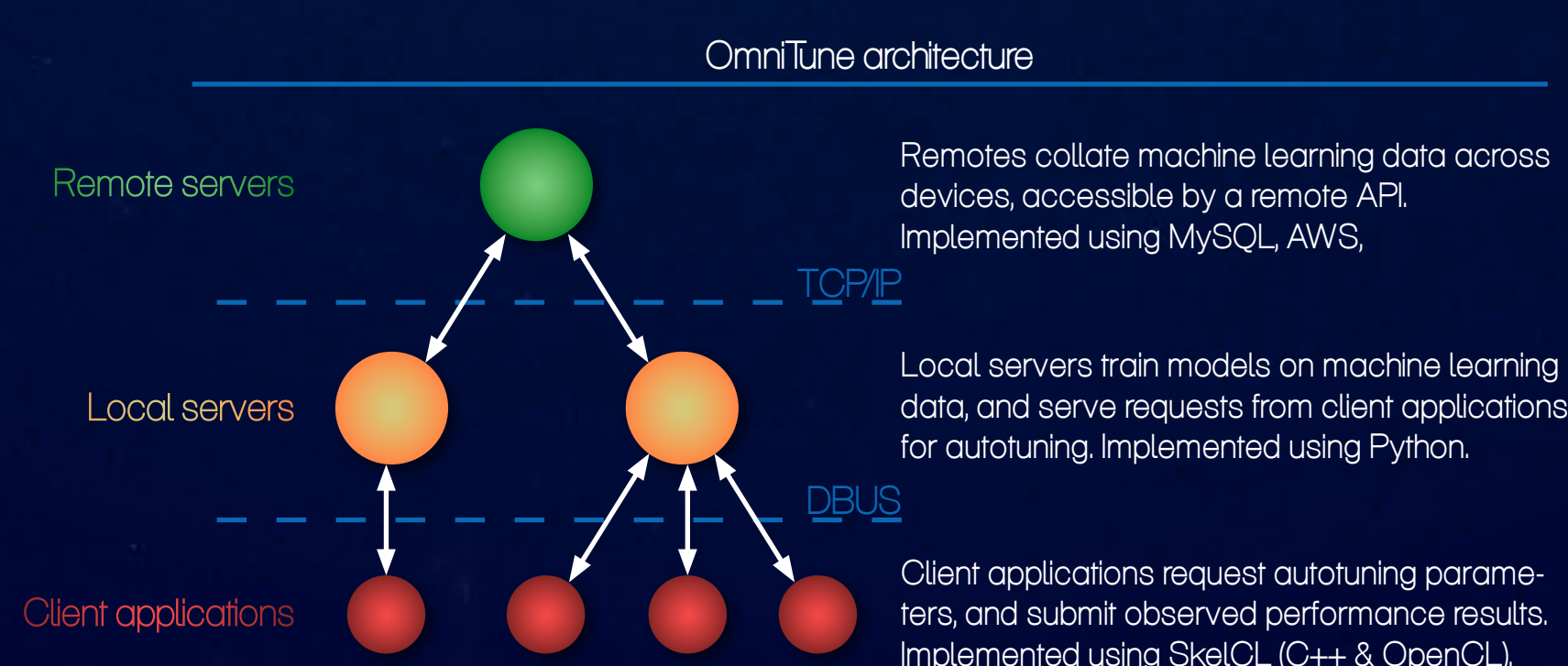
Introducing OmniTune ...

OmniTune generates synthetic benchmark programs to use for empirical testing

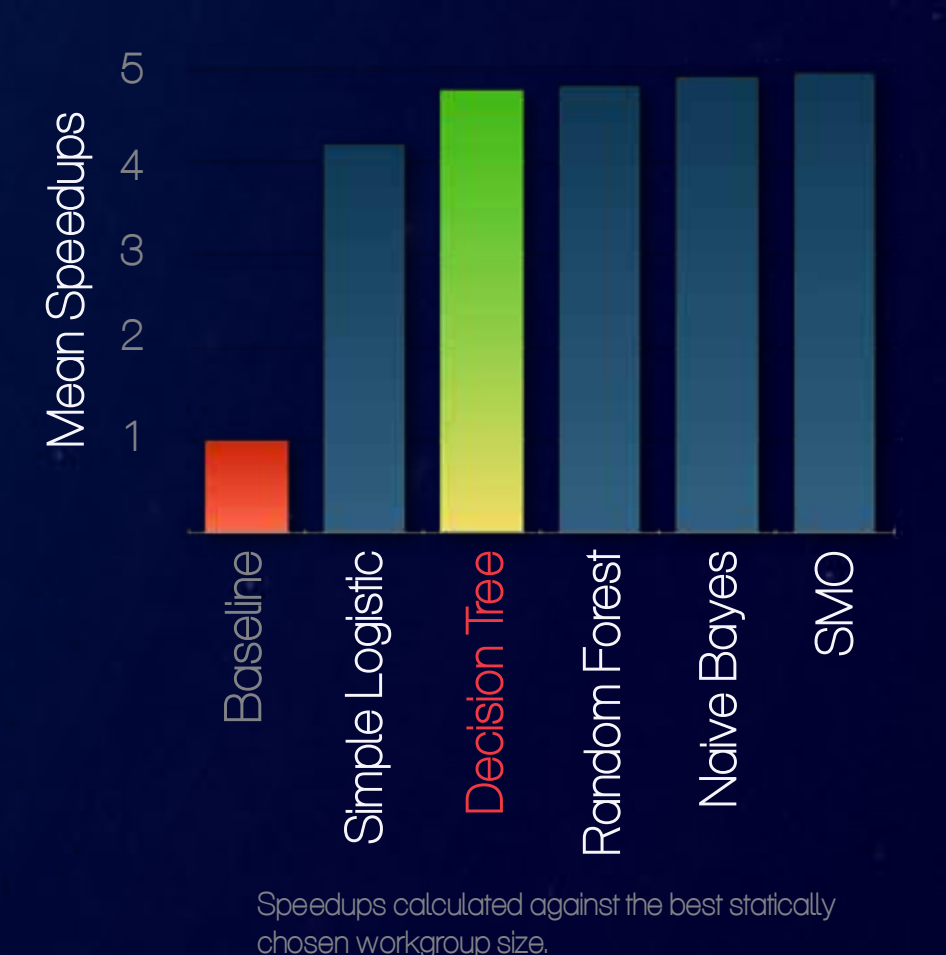OmniTune collaboratively gathers performance data by testing different parameter values

OmniTune uses machine learning to predict parameters for unseen programs at runtime

OmniTune architecture

Remote servers

Remotes collate machine learning data across devices, accessible by a remote API. Implemented using MySQL, AWS.

TCP/IP

Local servers

Local servers train models on machine learning data, and serve requests from client applications for autotuning. Implemented using Python.

DBUS

Client applications

Client applications request autotuning parameters, and submit observed performance results. Implemented using SkelCL (C++ & OpenCL)

Machine Learning features

Program	Device	Dataset
LLVM bytecode	OpenCL API	SkelCL container
- Instruction densities	- Compute devices	- Container width
- Num. basic blocks	- Memory sizes	- Container height
- StencIl region size	- Cache types	- Input data type
- etc.	- etc.	- etc.

Results

Mean Speedups

Baseline, Simple Logistic, Decision Tree, Random Forest, Naive Bayes, SMO

Speedups calculated against the best statically chosen workgroup size.

Read more ...

C. Cummins, P. Petoumenos, M. Steuwer, H. Leather "Autotuning OpenCL Workgroup Size for Stencil Patterns" ADAPT 2016.

C. Cummins, P. Petoumenos, M. Steuwer, H. Leather "Towards Collaborative Performance Tuning of Algorithmic Skeletons" HLPGPU 2016.

http://chriscummins.cc