
Autotuning e Profiling de Aplicações Paralelas

Pedro Bruel

phrb@ime.usp.br

MAC5742 - Computação Paralela e Distribuída

1 de Julho de 2015



Motivações

Arquiteturas multi-core e co-processadores:

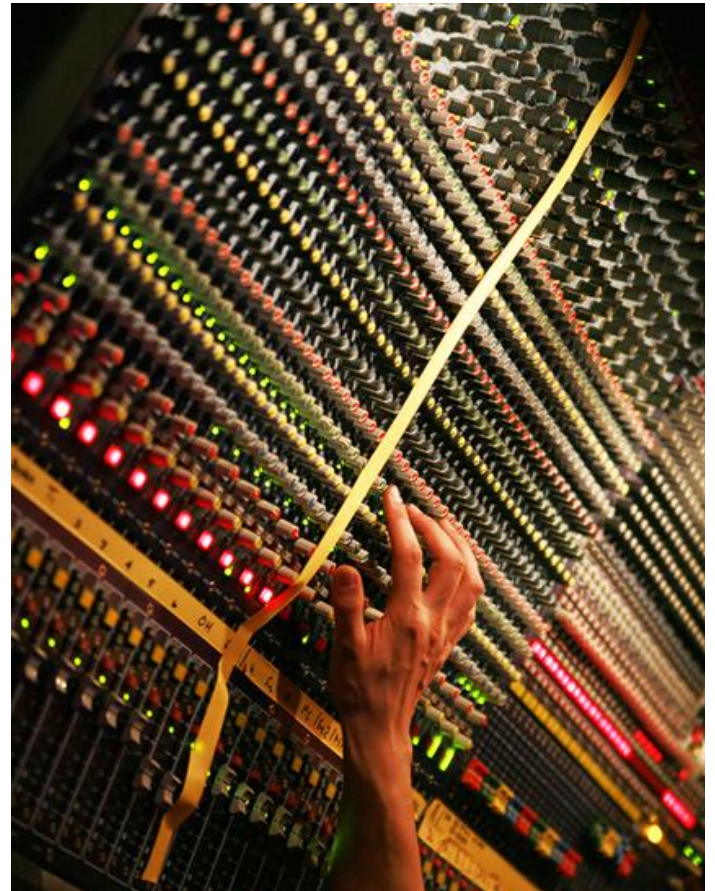
- Grande variedade
- Cada vez mais disponíveis
- Potencial para alto desempenho



Motivações

Como aproveitar esse potencial?

- OpenMP, MPI, CUDA...
- Bibliotecas para domínios específicos
- Otimização



Profiling

- Quais os objetivos da otimização?
 - Consumo de energia
 - Distribuição de recursos ou tarefas
 - Tempo de compilação
 - Tempo de execução
 - ...
- Como medir esses valores?

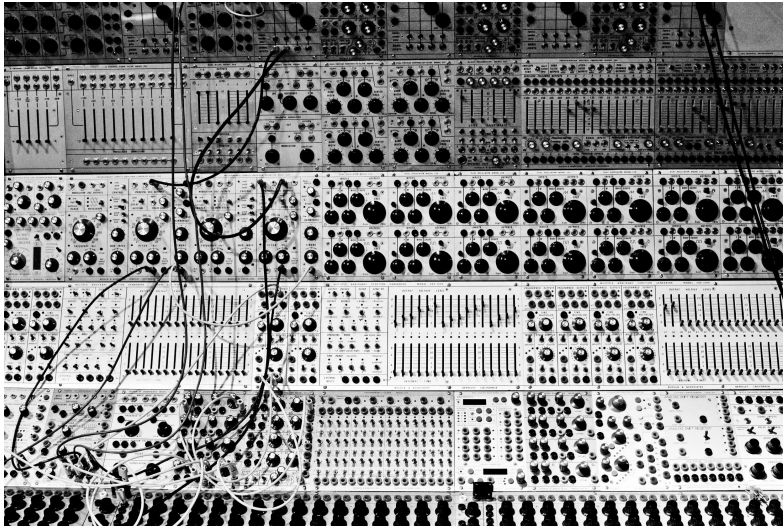
Profiling: Algumas Ferramentas

System	Features	License
perf tools	Sampling profiler supporting hardware events on several architectures.	GPL
LTTng	Collects data on processes blocking, context switches, and execution time. Helps identify performance problems over multiple processes or threads.	GPL
gprof	Several tools with combined sampling and call-graph profiling. A set of visualization tools, VCG tools, uses the Call Graph Drawing Interface (CGDI) to interface with gprof.	BSD
Allinea MAP	I/O, communication, floating point operation usage, memory access costs, MPI and OpenMP support.	Proprietary
Streamline	Graphical performance visualization of hardware and software for ARM CPUs, Mali GPUs, OpenCL, power consumption metrics.	Proprietary
VTune	Serial and threaded performance analysis. Hotspot, call tree and threading analysis on both Intel and AMD x86 processors. Hardware event sampling that uses the on chip performance monitoring unit requires an Intel processor.	Proprietary
CodeXL	GPU and CPU profilers, GPU debugger and static kernel analyzer.	Proprietary
NVIDIA Visual Profiler	Performance profiling for optimizing CUDA C/C++ applications.	Proprietary

Figure 5: Some profilers for parallel applications and their features.

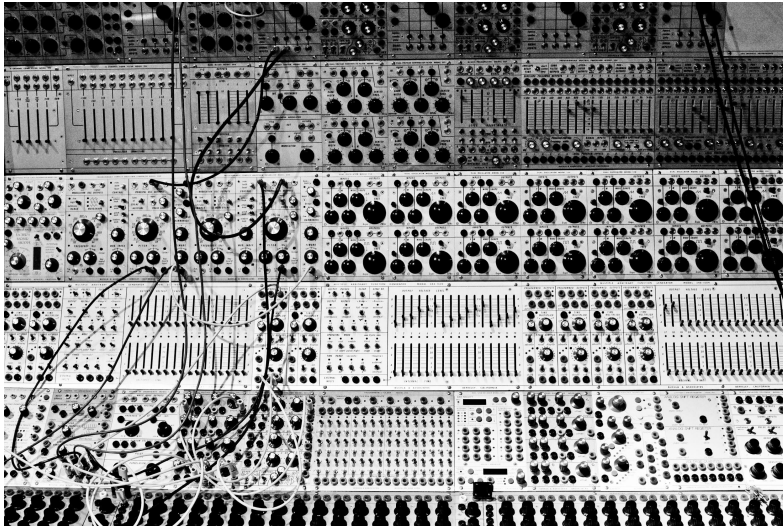
Autotuning

Configurações e Otimizações

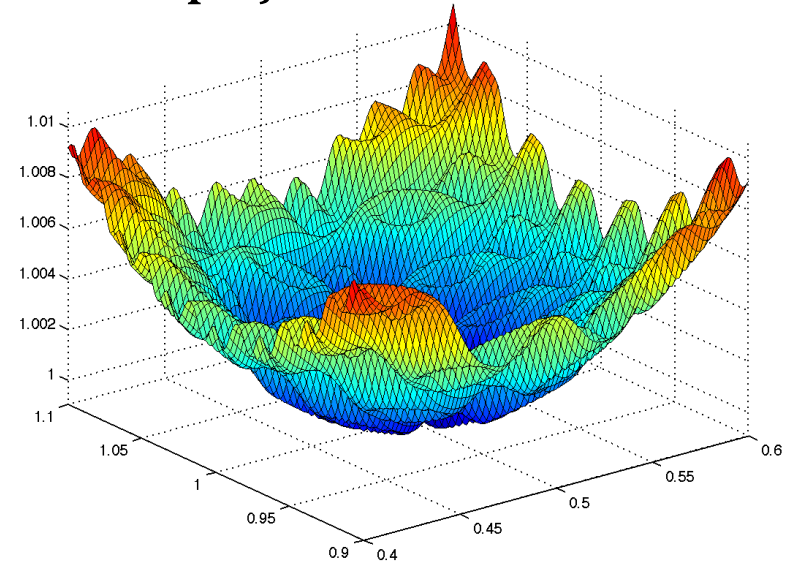


Autotuning

Configurações e Otimizações



Espaço de Busca



Autotuning: Alguns Sistemas

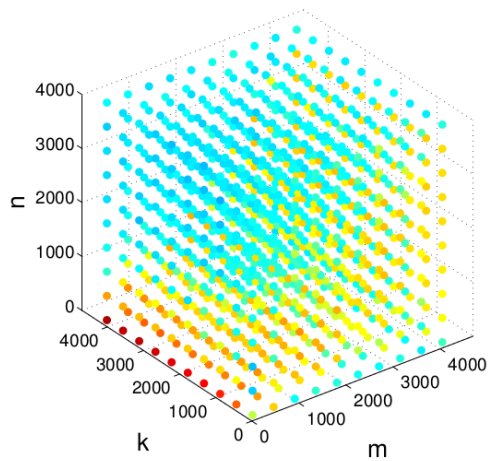
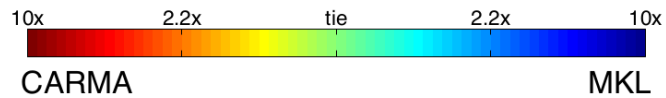
Projeto	Domínio	Método de Busca
ATLAS	Álgebra Linear (Matrizes Densas)	Exaustivo
Code Perforation	Compilador	Exaustivo + Simulated Annealing
FFTW	Transformada Rápida de Fourier	Exaustivo / Prog. Dinâmica
OSKI	Álgebra Linear (Matrizes Esparsas)	Exaustivo + Heurísticas
Active Harmony	Ambiente de Execução	Nelder-Mead
PATUS	Stencil Computations	Nelder-Mead ou Evolucionário
SPIRAL	Algoritmos para DSP	Pareto Active Learning
Dynamic Knobs	Ambiente de Execução	Control Theory
Milepost GCC / cTuning	Compilador	IID Model + Central DB
SEEC / Heartbeats	Ambiente de Execução	Control Theory
Insieme	Compilador	Differential Evolution
StreamJIT	“Compilador Comensal” (Dataflow)	OpenTuner
PetaBricks	Linguagem de Programação	Bottom-Up Evolutionary
OpenTuner	Framework	Vários

Autotuning: Espaço de Busca

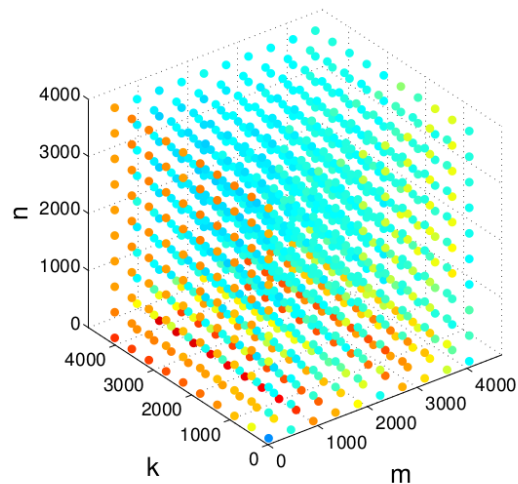
- **Espaço de Busca:** Duas estratégias para *multiplicação de matrizes* (CARMA e MKL)
- **Autotuner:** *Support Vector Machines*
- **Hardware:**

Machine	Cores	Threads	CPU Type
Hopper	24	24	AMD 'MagnyCours'
Emerald	32	64	Intel Xeon X7560
Boxboro	40	80	Intel Xeon E7-4860

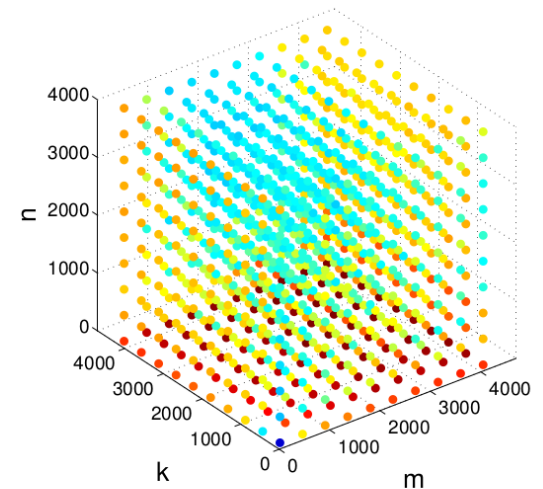
Autotuning: Espaço de Busca



(a) Hopper

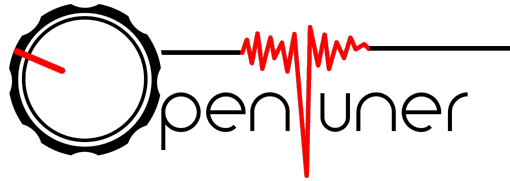


(b) Emerald



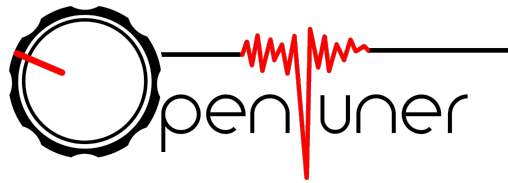
(c) Boxboro

Autotuning:

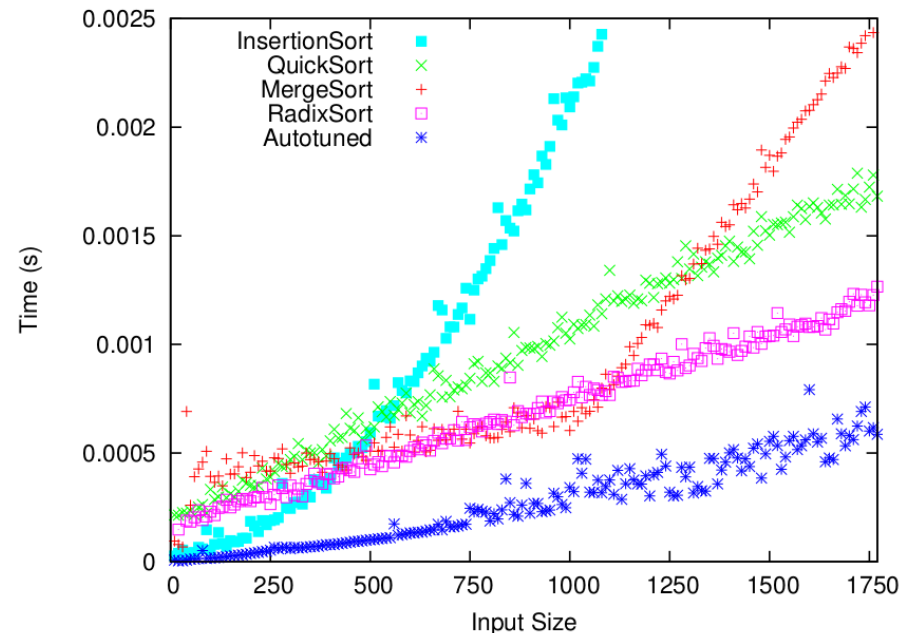


- *Framework* para implementação de *autotuners*
- Conjuntos de técnicas de busca
- Compartilhamento de resultados através de um banco de dados

Autotuning:

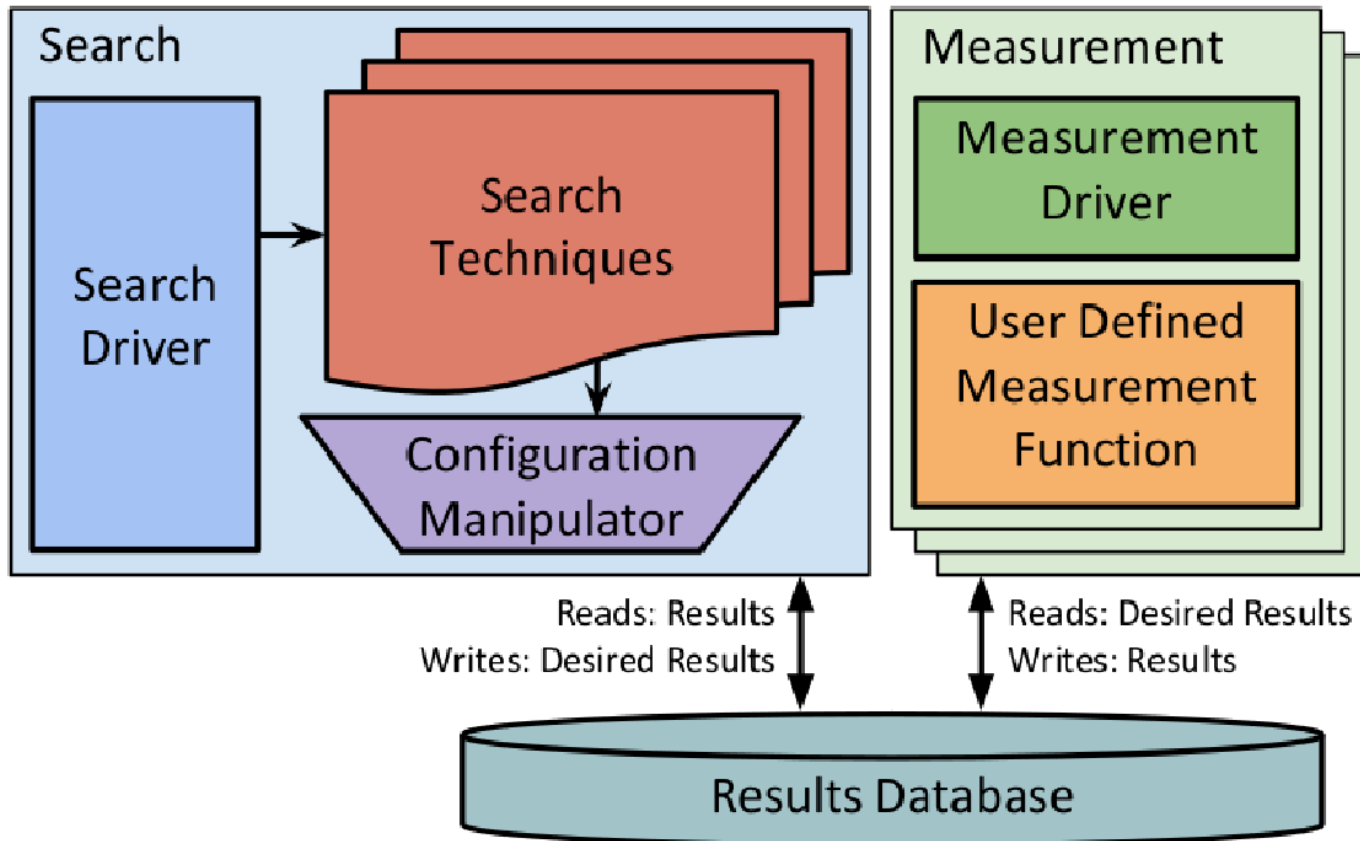
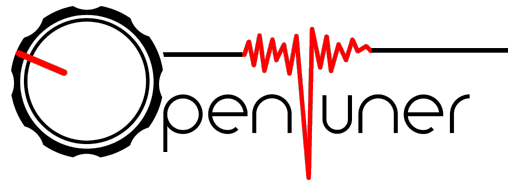


- *Framework* para implementação de *autotuners*
- Conjuntos de técnicas de busca
- Compartilhamento de resultados através de um banco de dados



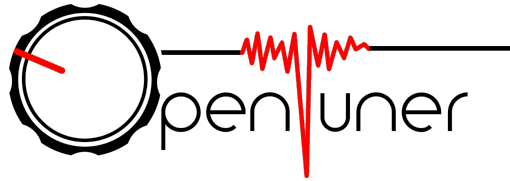
Ansel, Jason, et al. "Opentuner: An extensible framework for program autotuning." *Proceedings of the 23rd international conference on Parallel architectures and compilation*. ACM, 2014.

Autotuning:



Ansel, Jason, et al. "Opentuner: An extensible framework for program autotuning." *Proceedings of the 23rd international conference on Parallel architectures and compilation*. ACM, 2014.

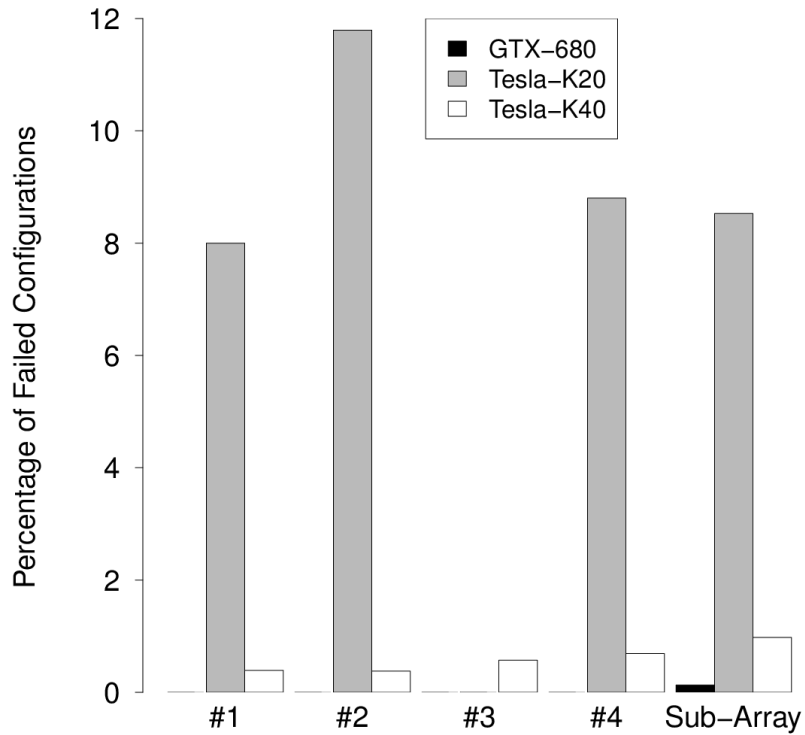
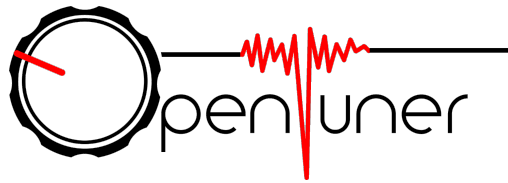
Autotuning:



Exemplo: Quais parâmetros de compilação para aplicações (CUDA) em GPUs produzem programas com melhor desempenho?

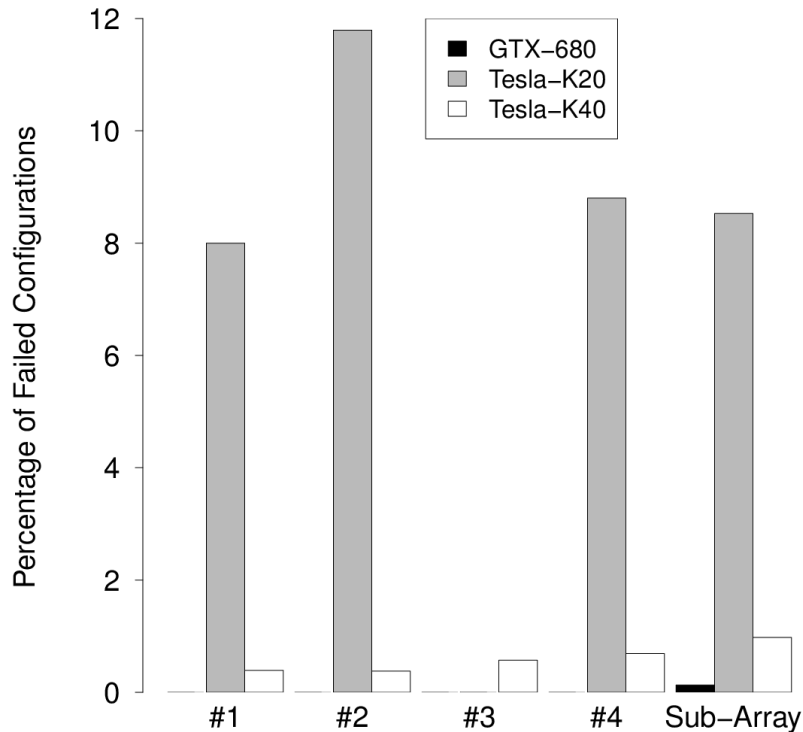
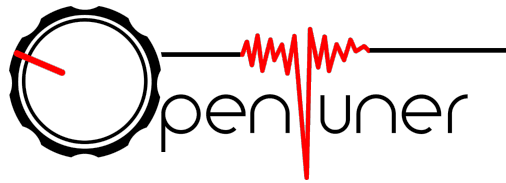
- **Espaço de Busca:** Parâmetros de compilação
- **Aplicações:** Código GPU legado, para multiplicação de matrizes e subsequência máxima
- **Hardware:** Tesla-K40, Tesla-K20, GeForce GTX-680
- **Autotuner:** Implementado utilizando OpenTuner

Autotuning:

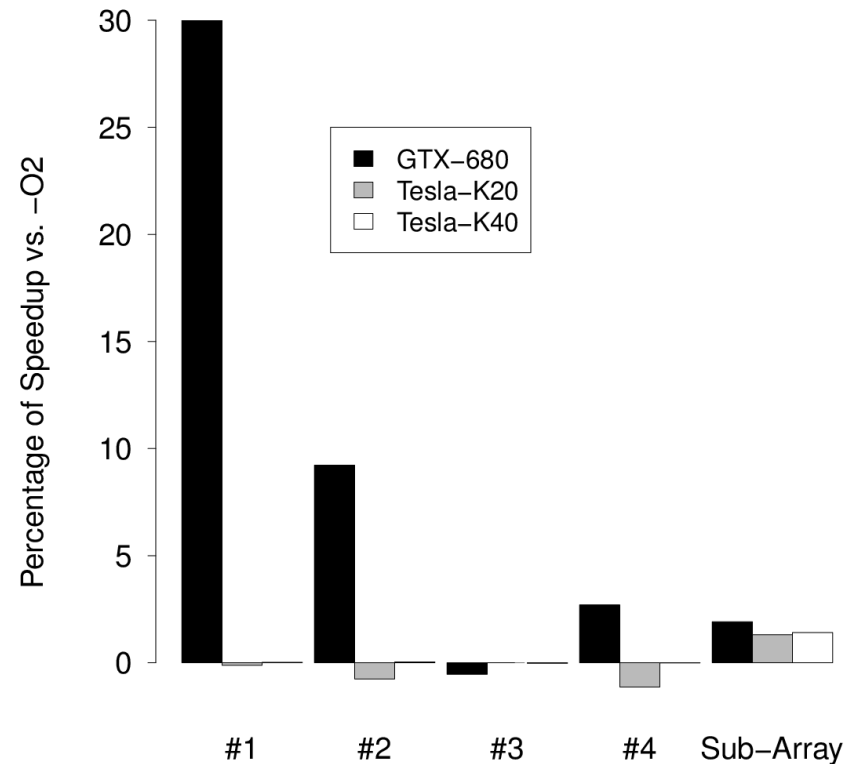


Conjuntos de opções que compilam mas produzem resultados incorretos.

Autotuning:



Conjuntos de opções que compilam mas produzem resultados incorretos.



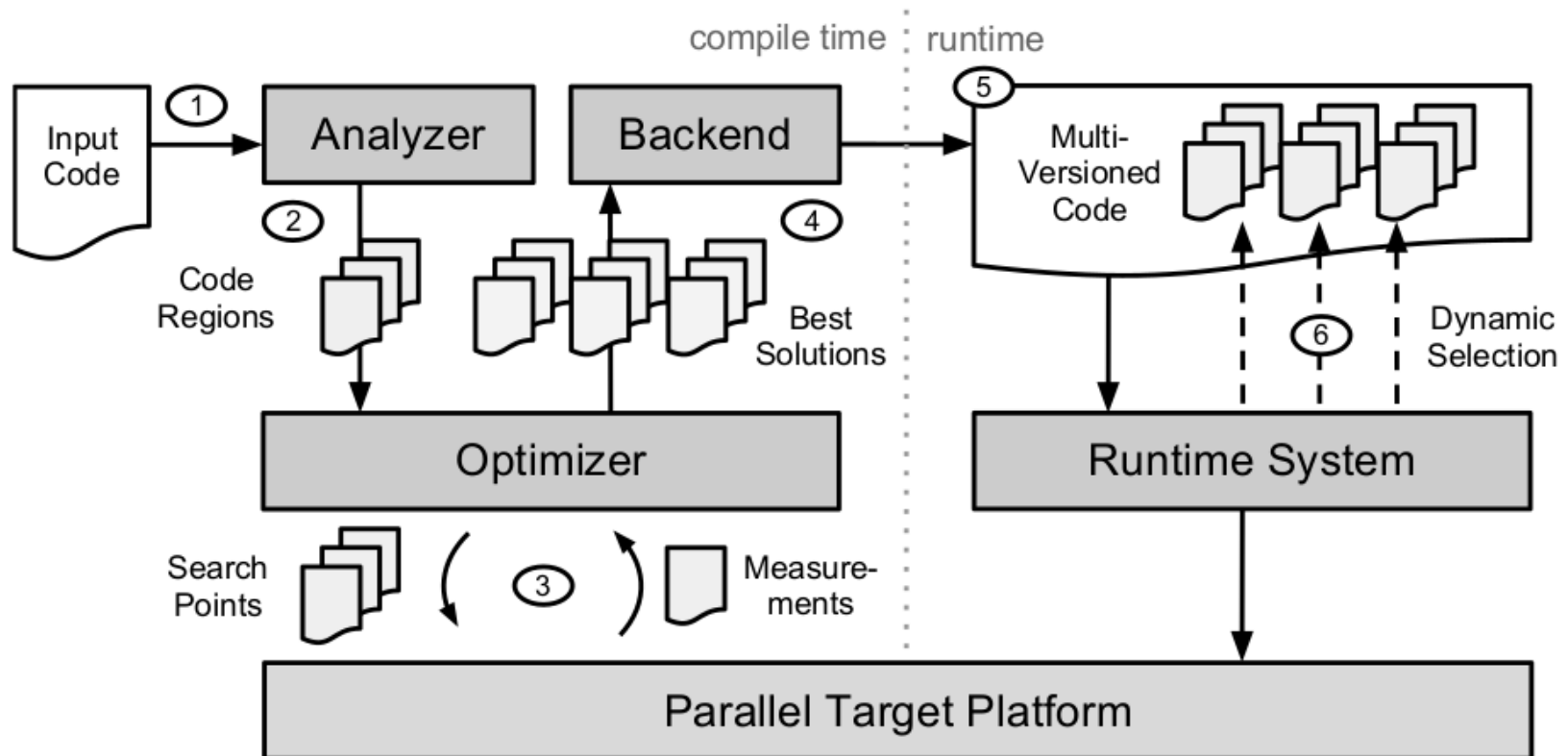
Desempenho das opções encontradas por *autotuning*. Ganho de 30% em um dos casos.

Autotuning:



- Compilador C/C++ fonte-para-fonte
- Representação comum para abstrações de OpenCL, OpenMP e MPI
- Suporte a arquiteturas heterogêneas

Autotuning:



Jordan, Herbert, et al. "A multi-objective auto-tuning framework for parallel codes." *High Performance Computing, Networking, Storage and Analysis (SC)*, 2012 International Conference for. IEEE, 2012.

Conclusão

- Otimização manual de aplicações paralelas fica cada dia mais difícil

Conclusão

- Otimização manual de aplicações paralelas fica cada dia mais difícil
- *Profilers* fornecem métricas para a otimização

Conclusão

- Otimização manual de aplicações paralelas fica cada dia mais difícil
- *Profilers* fornecem métricas para a otimização
- Estratégias automatizadas de otimização atingem bons resultados

Conclusão

- Otimização manual de aplicações paralelas fica cada dia mais difícil
- *Profilers* fornecem métricas para a otimização
- Estratégias automatizadas de otimização atingem bons resultados
- Disponibilidade de ferramentas independentes de domínio para *autotuning* de aplicações paralelas

Referências

1. Ansel, Jason, et al. "Opentuner: An extensible framework for program autotuning." *Proceedings of the 23rd international conference on Parallel architectures and compilation*. ACM, 2014.
2. Jordan, Herbert, et al. "A multi-objective auto-tuning framework for parallel codes." *High Performance Computing, Networking, Storage and Analysis (SC), 2012 International Conference for*. IEEE, 2012.
3. Spillinger, Omer, et al. "*Matrix Multiplication Algorithm Selection with Support Vector Machines.*" (2015).
4. Kotthoff, Lars. "Algorithm selection for combinatorial search problems: A survey." (2012).
5. Smith-Miles, Kate A. "Cross-disciplinary perspectives on meta-learning for algorithm selection." *ACM Computing Surveys (CSUR)* 41.1 (2008): 6.

Autotuning e Profiling de Aplicações Paralelas

Pedro Bruel

phrb@ime.usp.br

MAC5742 - Computação Paralela e Distribuída

1 de Julho de 2015

