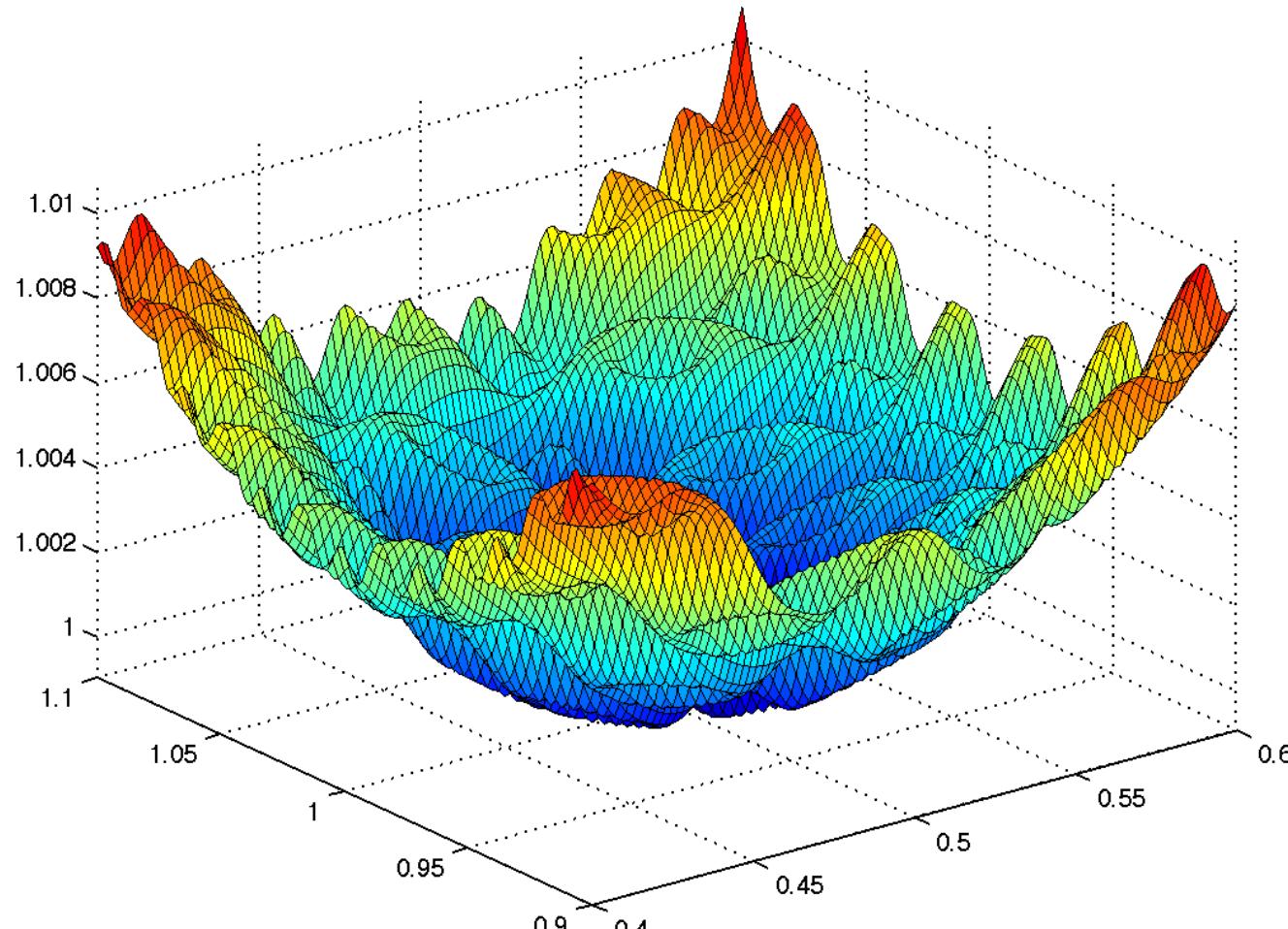


Otimização Automática de Programas

Prof. Dr. Alfredo Goldman, Pedro Bruel – {gold, phrb}@ime.usp.br



IME-USP

Índice

- 1) Contextualização;**
- 2) Auto-tuning;**
- 3) Exemplo.**

Características de Arquiteturas e Problemas...

- Domínio do Problema;
- Multi-core;
- Paralelismo;
- Computação em Nuvem;
- GPU;
- Memória Transacional;
- Modelos de Acesso a Memória;
- ...



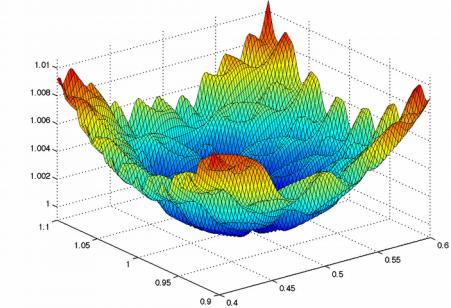
Características de Arquiteturas e Problemas... ...definem possíveis otimizações.

- Domínio do Problema;
 - Multi-core;
 - Paralelismo;
 - Computação em Nuvem;
 - GPU;
 - Memória Transacional;
 - Modelos de Acesso a Memória;
 - ...
-
- Escolha de Algoritmos e Heurísticas;
 - Parâmetros de Compilação;
 - Otimizações Algorítmicas:
 - Loop Unroling;
 - Alinhamento de Memória;
 - ...



Características de Arquiteturas e Problemas... ...definem possíveis otimizações.

- Domínio do Problema;
- Multi-core;
- Paralelismo;
- Computação em Nuvem;
- GPU;
- Memória Transacional;
- Modelos de Acesso a Memória;
- ...
- Escolha de Algoritmos e Heurísticas;
- Parâmetros de Compilação;
- Otimizações Algorítmicas:
 - Loop Unroling;
 - Alinhamento de Memória;
 - ...
- **Espaços de Busca!**



Contexto

Exemplo: Merge sort vs. Insertion sort em `std::sort` (C++)

Implementação de `std::sort` em C++:

- Merge Sort, se $n \geq 15$;
- Insertion Sort, se $n < 15$.

Contexto

Exemplo: Merge sort vs. Insertion sort em `std::sort` (C++)

Implementação de `std::sort` em C++:

- Merge Sort, se $n \geq 15$;
- Insertion Sort, se $n < 15$.

Ponto de troca fixo para todas as arquiteturas!

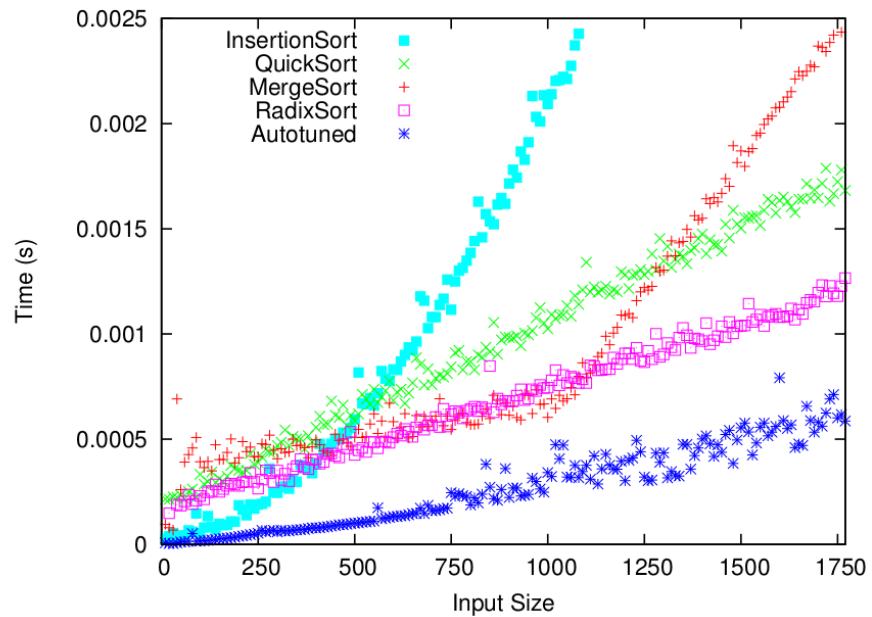
Contexto

Exemplo: Merge sort vs. Insertion sort em `std::sort` (C++)

Implementação de `std::sort` em C++:

- Merge Sort, se $n \geq 15$;
- Insertion Sort, se $n < 15$.

Ponto de troca fixo para todas as arquiteturas!

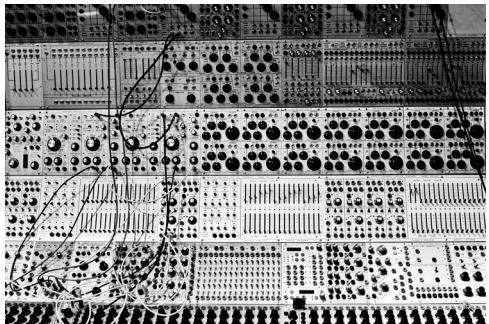


Desempenho de Algoritmos de Ordenação
em Arquitetura com 8 núcleos. [ANSEL09]

Auto-tuning

Navegando em Espaços de Busca

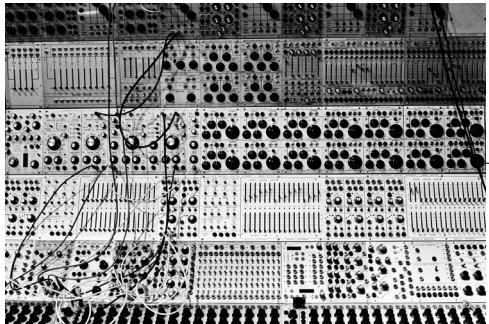
Programa



Auto-tuning

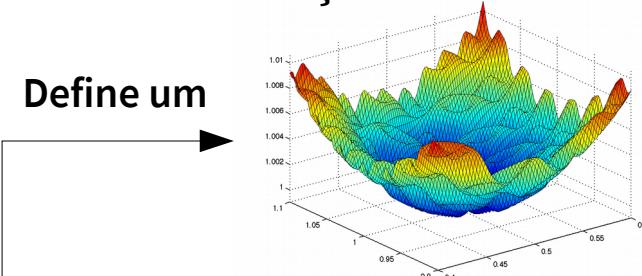
Navegando em Espaços de Busca

Programa



Define um

Espaço de Configurações
e Otimizações



Auto-tuning

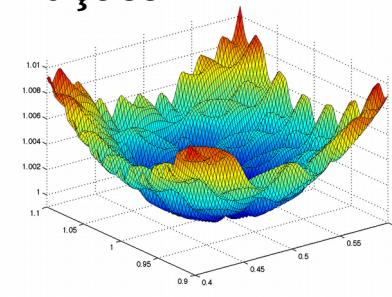
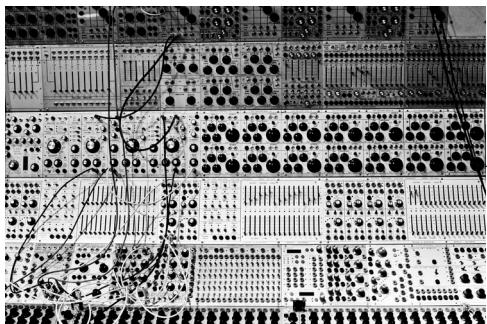
Navegando em Espaços de Busca

Espaço de Configurações
e Otimizações

Define um

Realiza Buscas em um

Programa

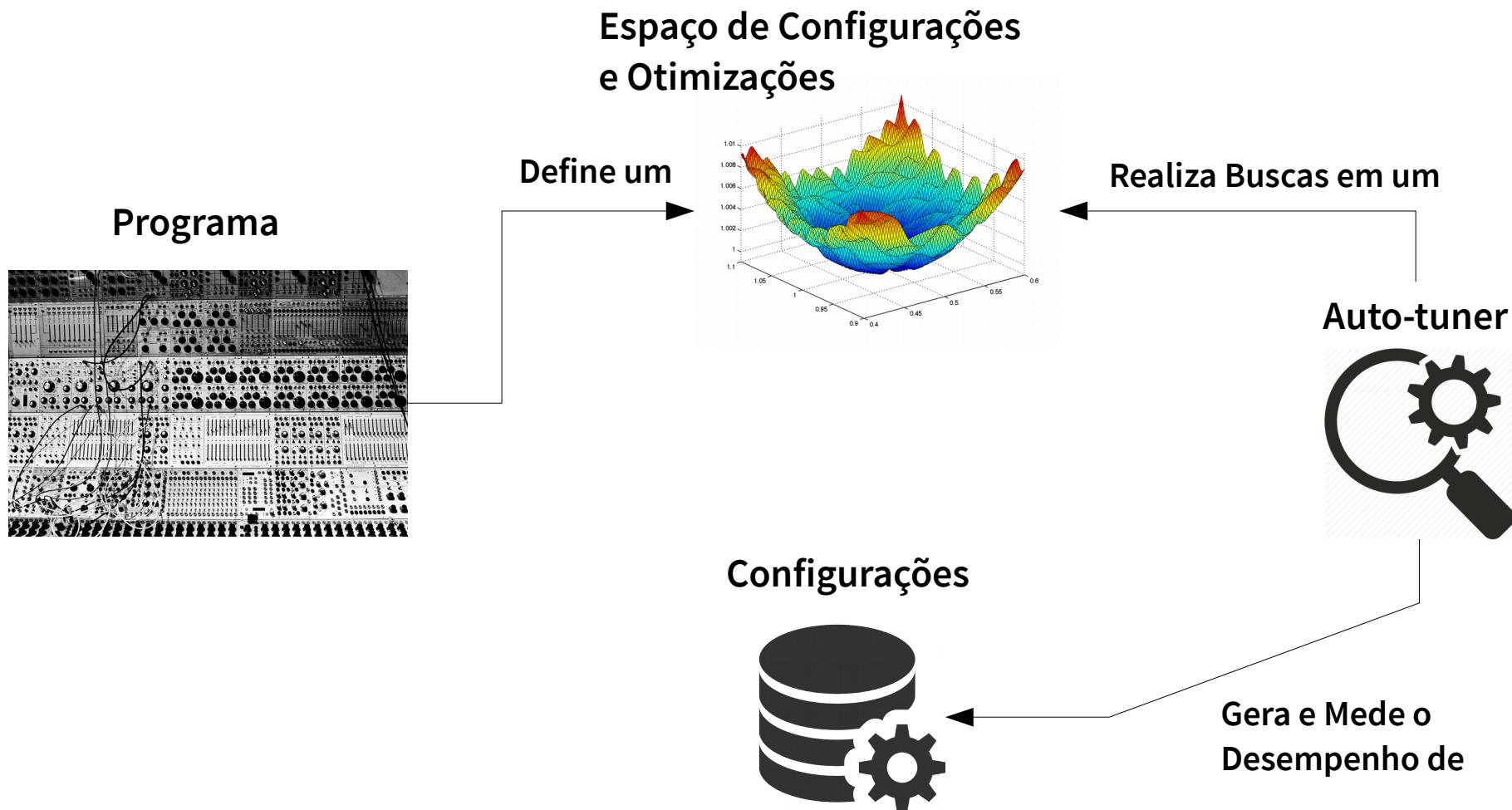


Auto-tuner



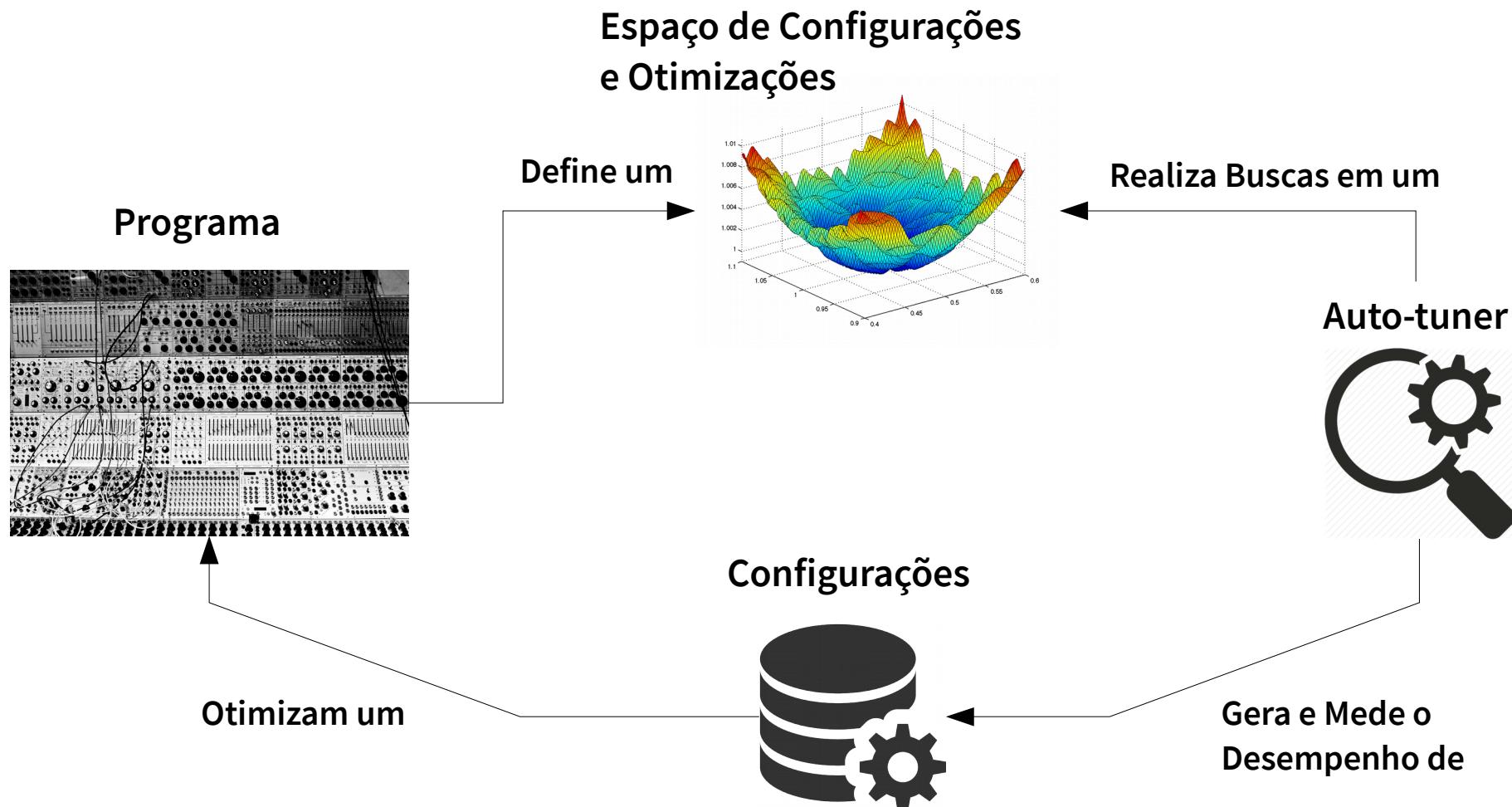
Auto-tuning

Navegando em Espaços de Busca



Auto-tuning

Navegando em Espaços de Busca



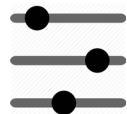
Auto-tuning

Trabalhos Relacionados

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

Exemplo

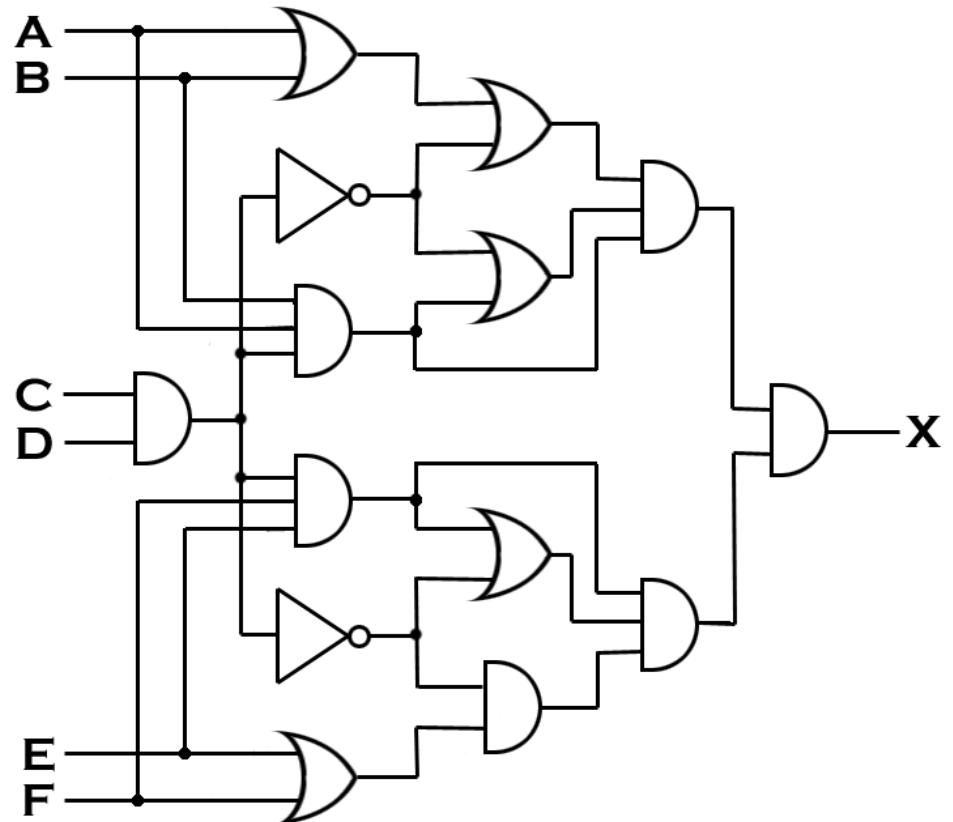
Conjuntos de Instâncias de SAT



Programa

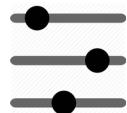
Resolução de um Conjunto de Instâncias do Problema da Satisfabilidade Booleana (SAT).

What combination of the inputs A, B, C, D, E, F makes X true?



Exemplo

Conjuntos de Instâncias de SAT



Programa

Resolução de um Conjunto de Instâncias do Problema da Satisfabilidade Booleana (SAT).

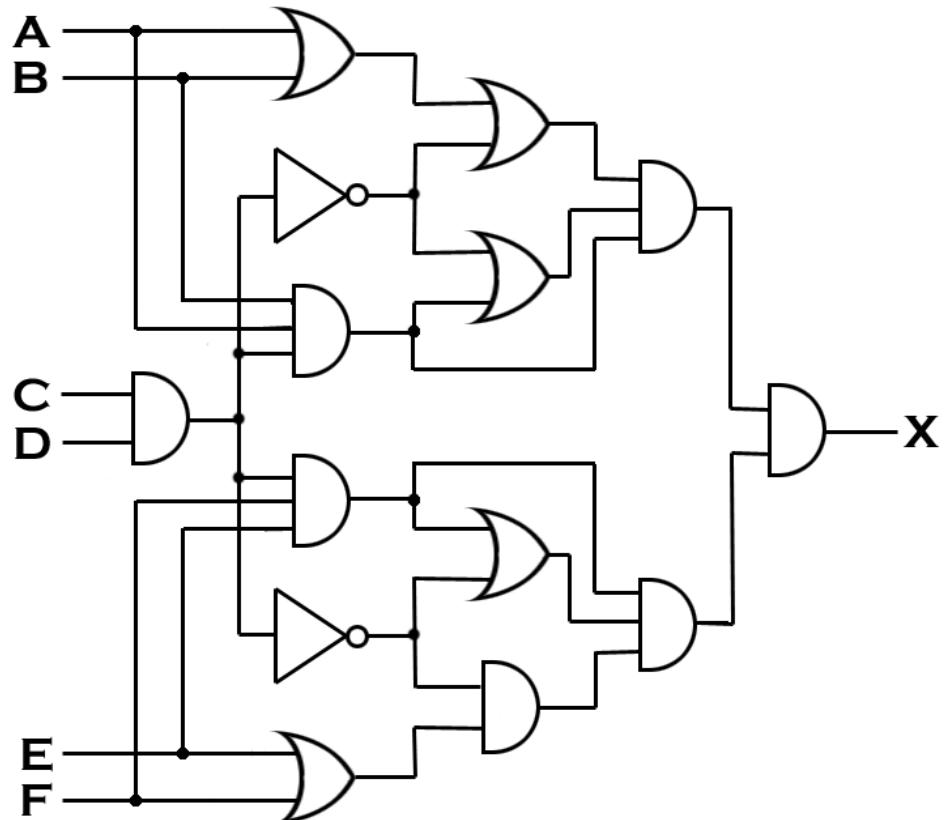
- 100 Instâncias¹;
- 7 Resolvedores².

Objetivo: Distribuir as instâncias entre os resolvedores para diminuir o tempo de solução do conjunto.

[1] cs.ubc.ca/~hoos/SATLIB/benchm.html

[2] satcompetition.org/

What combination of the inputs A, B, C, D, E, F makes X true?



Exemplo

Conjuntos de Instâncias de SAT



Auto-tuner



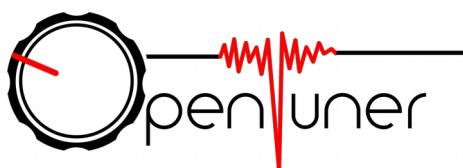
- Arcabouço para Implementação de Auto-tuners;
- Conjuntos de Técnicas de Busca;
- Divisão de Testes entre Técnicas (Meta-técnicas).

Exemplo

Conjuntos de Instâncias de SAT



Auto-tuner



- Arcabouço para Implementação de Auto-tuners;
- Conjuntos de Técnicas de Busca;
- Divisão de Testes entre Técnicas (Meta-técnicas).

Differential Evolution
Genetic Algorithms
Greedy Mutation
Multi-armed Bandit
Nelder Mead
Partial Swarm Optimization
Pattern Search
Pseudo Annealing
Torczon
...

Exemplo

Conjuntos de Instâncias de SAT



Auto-tuner



- Arcabouço para Implementação de Auto-tuners;
- Conjuntos de Técnicas de Busca;
- Divisão de Testes entre Técnicas (Meta-técnicas).

Differential Evolution
Genetic Algorithms
Greedy Mutation
Multi-armed Bandit
Nelder Mead
Partial Swarm Optimization
Pattern Search
Pseudo Annealing
Torczon
...

AUC MAB
...

Exemplo

Conjuntos de Instâncias de SAT



Configurações

$$C = [s_0, \dots, s_{99}], s_i \in [0,6]$$

Uma configuração C para o programa é um vetor de inteiros, e especifica um resolvedor para cada instância do conjunto.

Exemplo

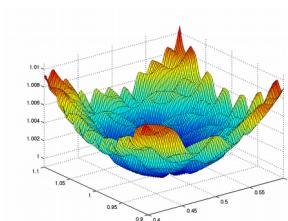
Conjuntos de Instâncias de SAT



Configurações

$$C = [s_0, \dots, s_{99}], s_i \in [0, 1]$$

Uma configuração C para o programa é um vetor de inteiros, e especifica um resolvedor para cada instância do conjunto.



Espaço de Busca

- $7^{100} \approx 10^{84}$ configurações possíveis;
- $\approx 10^{80}$ átomos no universo observável.

Exemplo

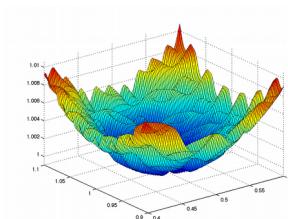
Conjuntos de Instâncias de SAT



Configurações

$$C = [s_0, \dots, s_{99}], s_i \in [0, 1]$$

Uma configuração C para o programa é um vetor de inteiros, e especifica um resolvedor para cada instância do conjunto.



Espaço de Busca

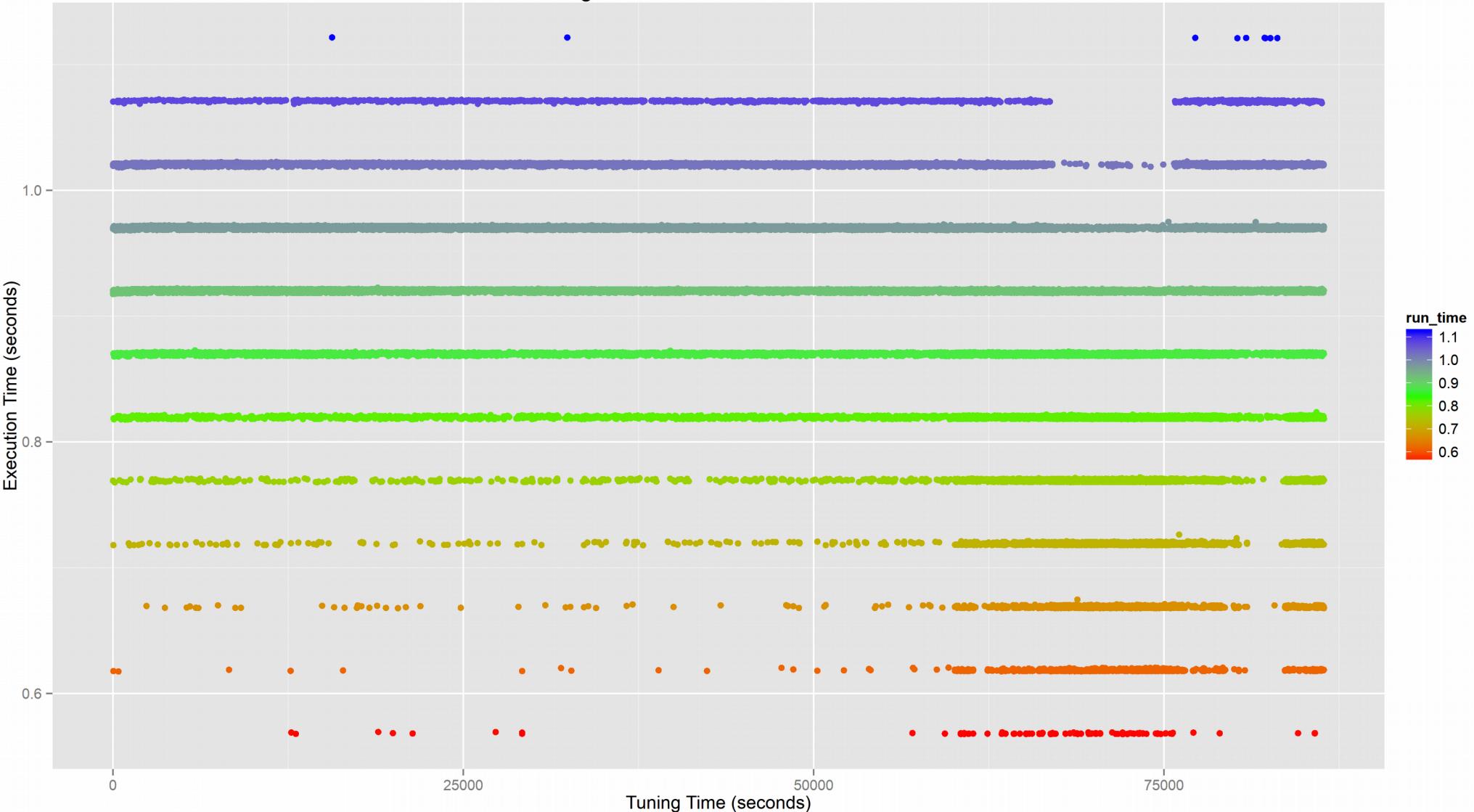
- $7^{100} \approx 10^{84}$ configurações possíveis;
- $\approx 10^{80}$ átomos no universo observável.

No entanto, 700 execuções são suficientes para determinar o melhor resolvedor para cada instância.

Exemplo

Conjuntos de Instâncias de SAT: Resultados

Tuning for 24h: All Combinations Tested

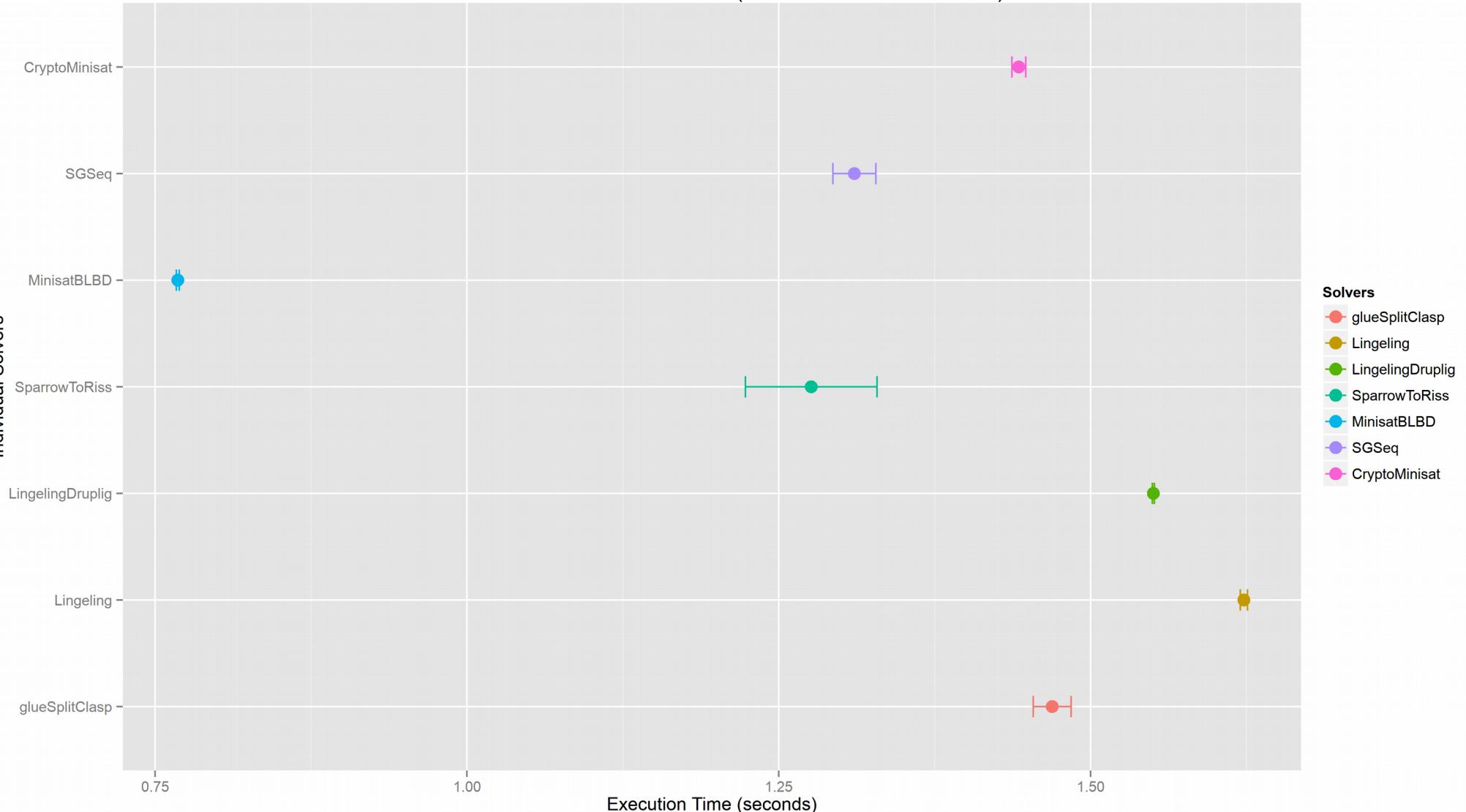


3.16.0-4-amd64 #1 SMP Debian 3.16.7-ckt9-3 (2015-04-23) x86_64 GNU/Linux

Exemplo

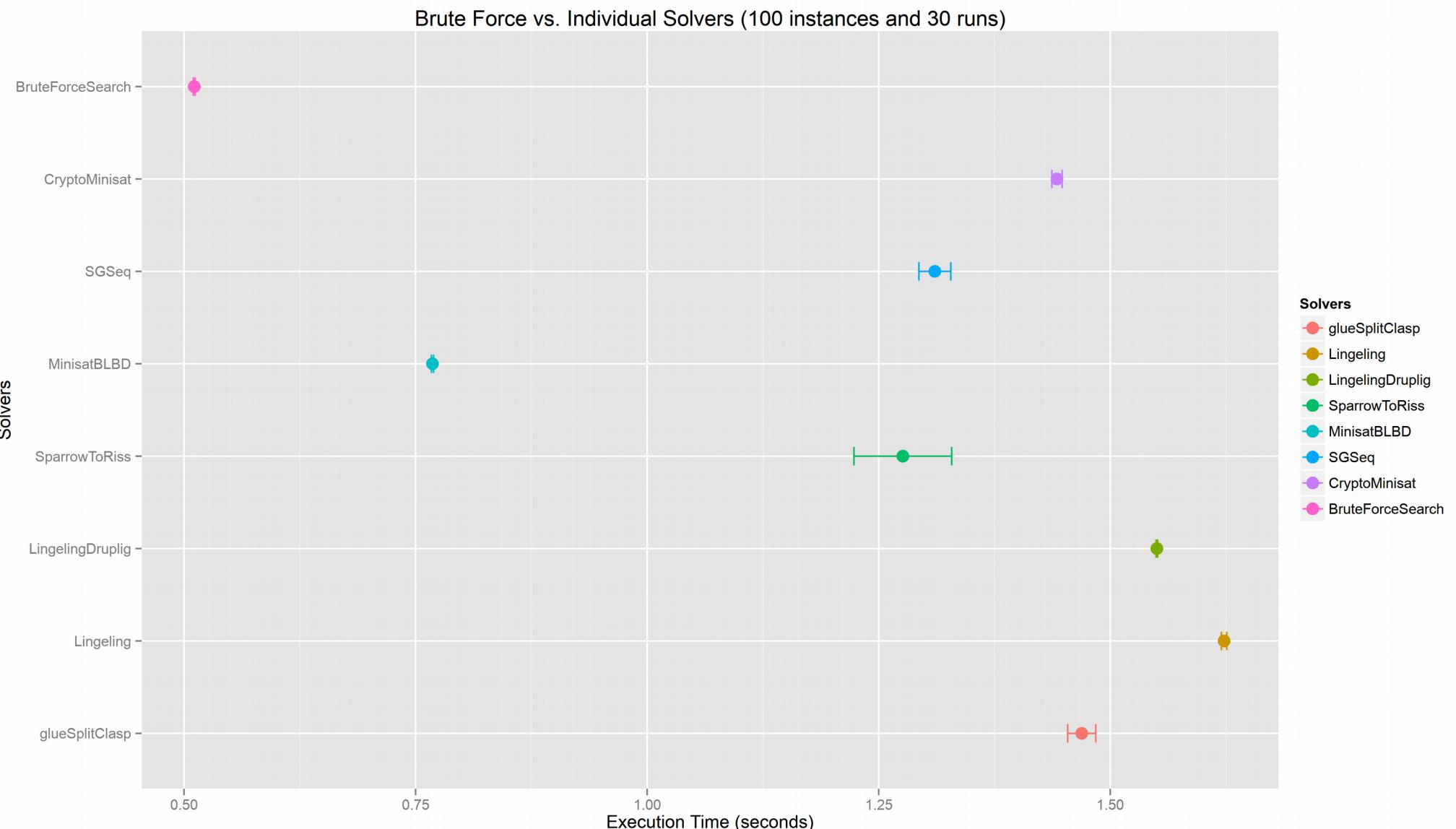
Conjuntos de Instâncias de SAT: Resultados

Benchmark of Individual Solvers (100 instances and 30 runs)



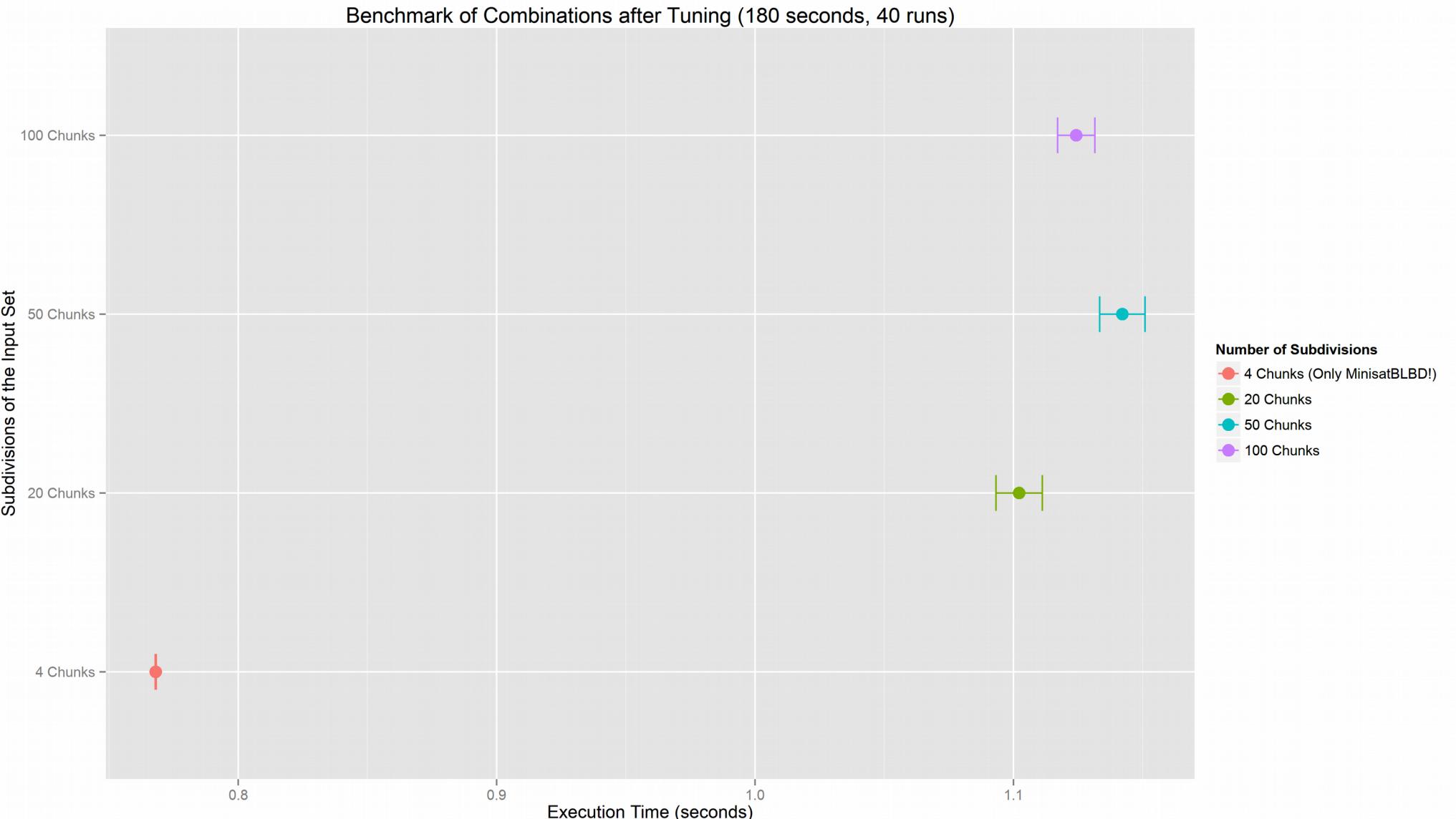
Exemplo

Conjuntos de Instâncias de SAT: Resultados



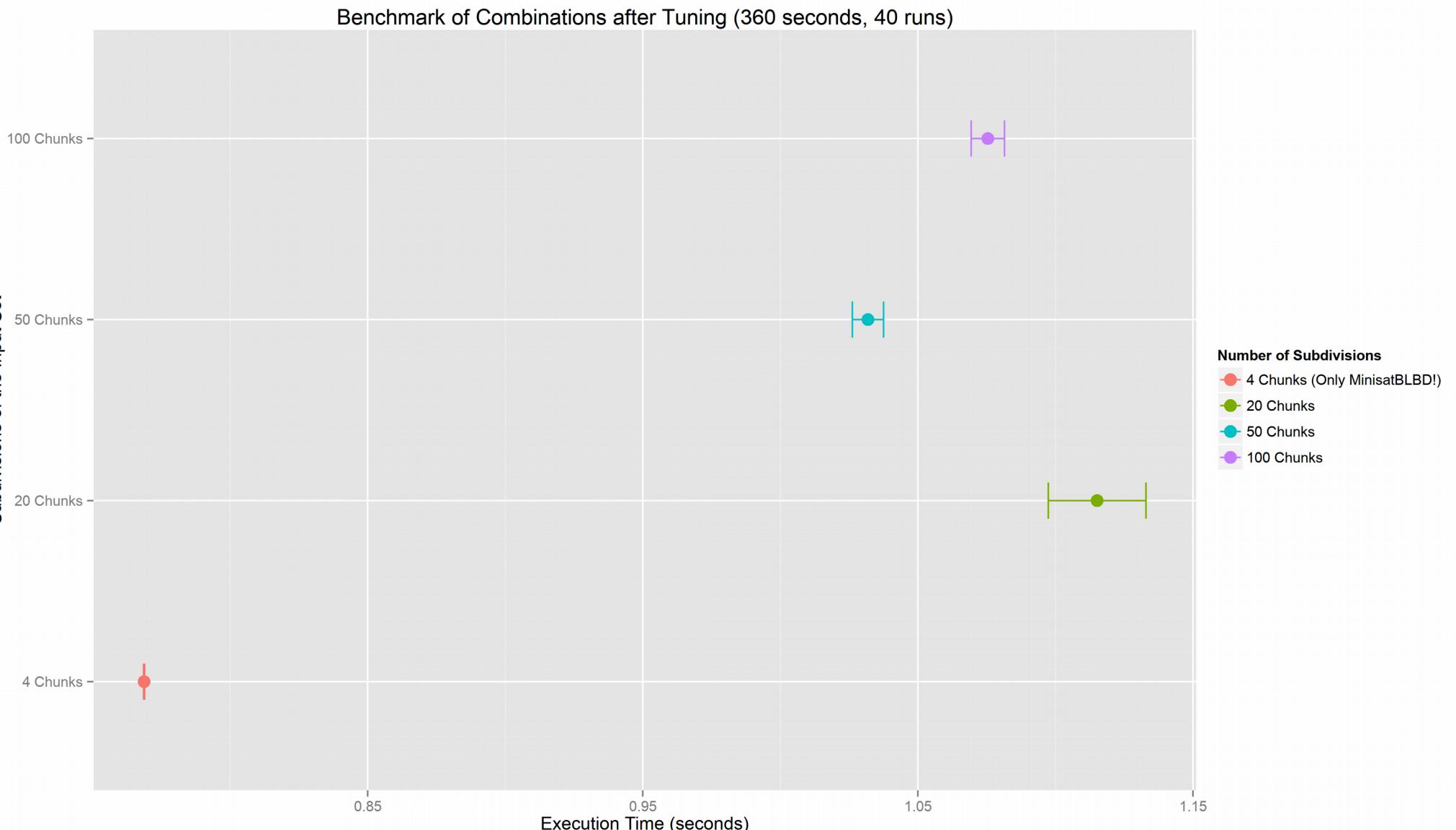
Exemplo

Conjuntos de Instâncias de SAT: Resultados



Exemplo

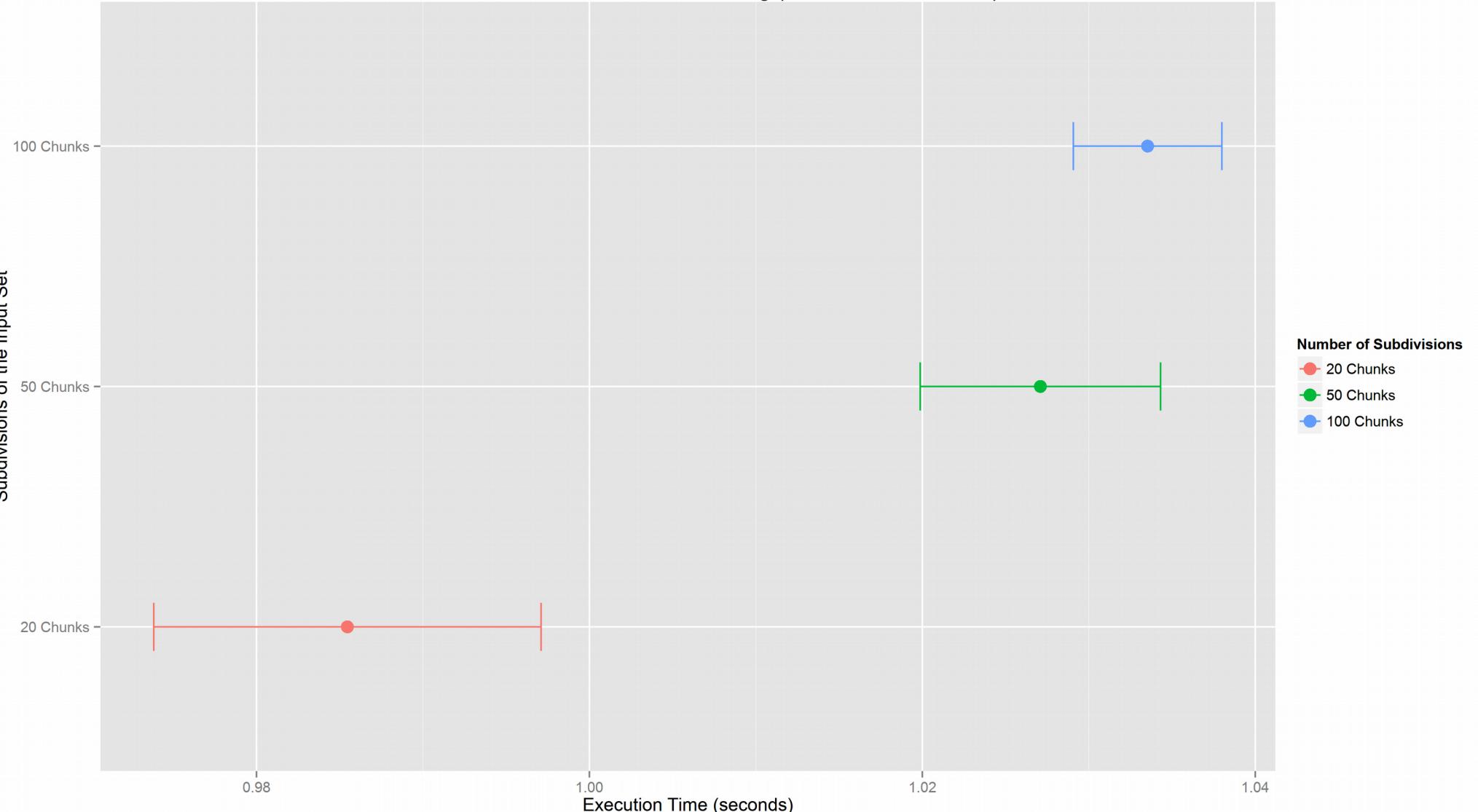
Conjuntos de Instâncias de SAT: Resultados



Exemplo

Conjuntos de Instâncias de SAT: Resultados

Benchmark of Combinations after Tuning (840 seconds, 40 runs)



Exemplo

Conjuntos de Instâncias de SAT: Conclusão

Conclusões:

- O espaço de busca contém muitos mínimos locais.
- Mais tempo de auto-tuning nem sempre produz melhores resultados;
- A representação do problema é muito importante.

Exemplo

Conjuntos de Instâncias de SAT: Conclusão

Conclusões:

- O espaço de busca contém muitos mínimos locais.
- Mais tempo de auto-tuning nem sempre produz melhores resultados;
- A representação do problema é muito importante.

Ideias:

- Implementar técnicas e meta-técnicas do **OpenTuner** que restrinjam a busca;
- Implementar novos tipos de parâmetros do **OpenTuner**;
- Encontrar novas representações;
- Comparar o desempenho e as soluções do **OpenTuner** com as encontradas por uma implementação no **PetaBricks**.

Referências

- [ANSEL09] Jason Ansel, Cy Chan, Yee Lok Wong, Marek Olszewski, Qin Zhao, Alan Edelman, and Saman Amarasinghe. 2009. *PetaBricks: a language and compiler for algorithmic choice*. In Proceedings of the 2009 ACM SIGPLAN conference on Programming language design and implementation (PLDI '09). ACM, New York, NY, USA, 38-49.
- [ANSEL14] Jason Ansel, Shoaib Kamil, Kalyan Veeramachaneni, Jonathan Ragan-Kelley, Jeffrey Bosboom, Una-May O'Reilly, and Saman Amarasinghe. 2014. *OpenTuner: an extensible framework for program autotuning*. In Proceedings of the 23rd international conference on Parallel architectures and compilation (PACT '14). ACM, New York, NY, USA, 303-316.

projects.csail.mit.edu/petabricks/

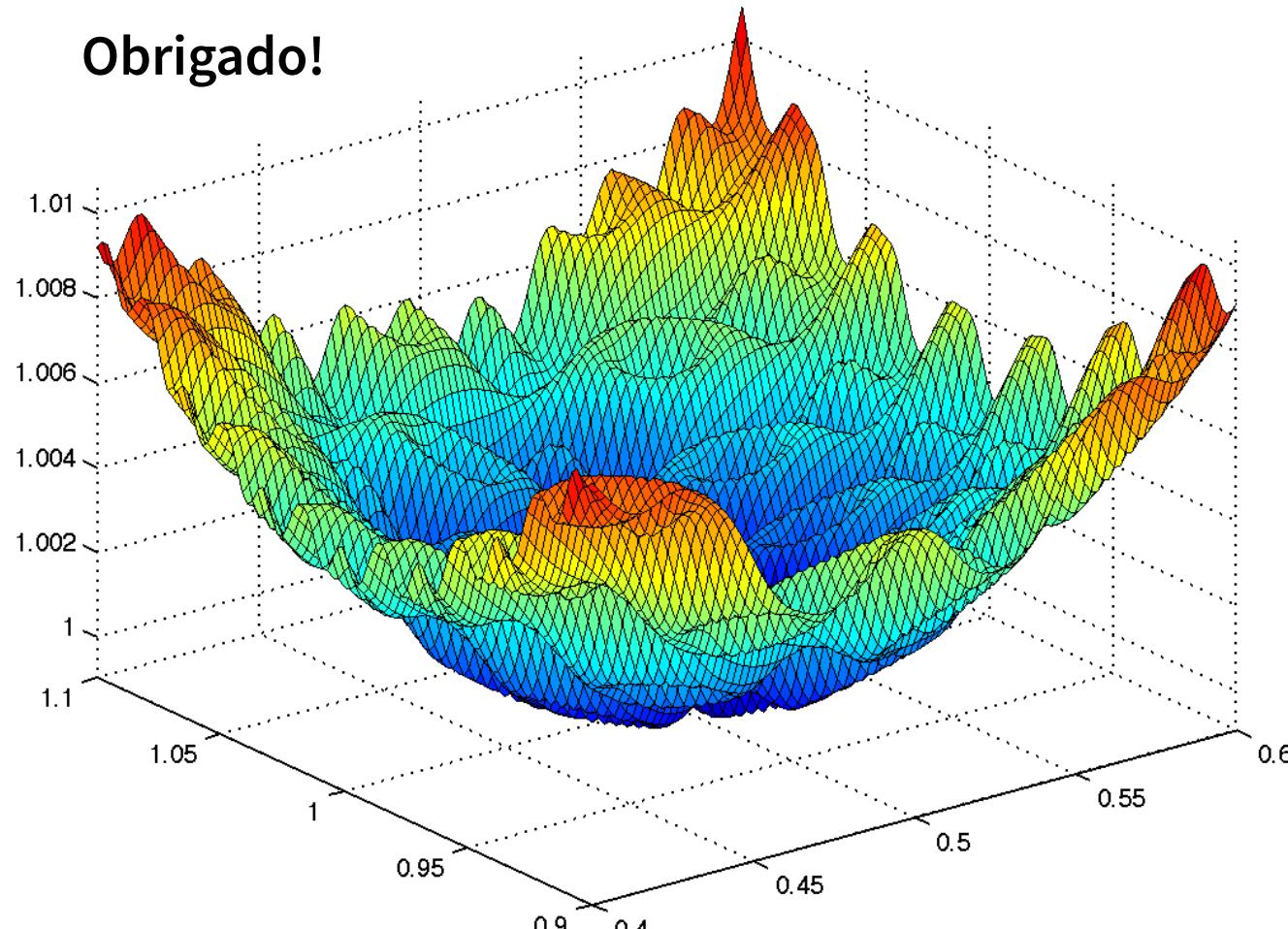
opentuner.org/

github.com/phrb/sat-opentuner

Otimização Automática de Programas

Prof. Dr. Alfredo Goldman, Pedro Bruel – {gold, phrb}@ime.usp.br

Obrigado!



IME-USP



Laboratório
de Sistemas de
Software