

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

GABRIEL BRONZATTI MORO

Plano de Ensino e Pesquisa

Orientador: Prof. Dr. Lucas Mello Schnorr

Porto Alegre
2016

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. Carlos Alexandre Netto

Vice-Reitor: Prof. Rui Vicente Oppermann

Pró-Reitor de Graduação: Prof. Sérgio Roberto Kieling Franco

Diretor do Instituto de Informática: Prof. Luis da Cunha Lamb

Coordenador do Curso de Ciência de Computação: Prof. Carlos Arthur Lang
Lisbôa

Bibliotecária-chefe do Instituto de Informática: Beatriz Regina Bastos Haro

RESUMO

Abstract ...

Palavras-chave: .

SUMÁRIO

1 INTRODUÇÃO	5
2 CONCEITOS BÁSICOS	6
3 TRABALHOS RELACIONADOS	7
4 EXPERIMENTOS	9
APÊNDICE A — APENDICE 1	10

1 INTRODUÇÃO

2 CONCEITOS BÁSICOS

3 TRABALHOS RELACIONADOS

??) apresentam uma ferramenta chamada Power-Sleuth que é capaz de fornecer uma descrição detalhada do comportamento de uma aplicação quando executada em determinada frequência. Esse trabalho utiliza três técnicas fundamentais para compreender o comportamento de um programa, detecção de fases, modelo DVFS (*Dynamic Voltage and Frequency Scaling*) e modelos de correlação. A abordagem desenvolvida pelos autores identificam as fases da aplicação utilizando uma biblioteca chamada ScarPhase, a qual utiliza como base um histórico de execução da aplicação, agrupando em fases, funções do programa que possuem um comportamento similar (acessos a memória, taxa de misses, entre outros). Esse artigo investiga apenas aplicações sequenciais, nessa perspectiva a identificação de áreas de regiões memory-bound podem ser obtidas em uma granularidade mais grosseira no intervalo entre amostras. Já para aplicações paralelas, como são executadas sobre diferentes fluxos de processamento, cada fluxo pode possuir um comportamento distinto de acordo como o balanceamento da carga da aplicação, até mesmo para mesmas regiões de código.

??) definem uma abordagem automatizada que permite selecionar a frequência mais adequada de processador para determinado laço do programa. A frequência do processador é escolhida utilizando como base uma análise estática (realizada antes da execução) e outra análise realizada durante o tempo de execução da aplicação, utilizando os rastros obtidos. Os autores utilizaram vários benchmarks, tendo como base de execução o framework chamado pcubed (*PMaC's Performance and Power benchmark*) que permite explorar diferentes comportamentos de laços de interações a fim de definir uma caracterização para a máquina alvo. A caracterização da máquina define valores como consumo de potência, desempenho, padrões de execução e frequências de processador. Os resultados obtidos no experimento podem ser utilizados posteriormente como base de conhecimento, assim é possível visualizar o comportamento do consumo de energia quando se ajusta os fatores de caracterização da máquina. Dentre os resultados obtidos pelo trabalho, o melhor foi a redução de até 10,6\

Diferente de ??), ??) apresentam uma abordagem voltada à memória distribuída para aplicações MPI. Essa abordagem encontra a melhor frequência para cada nó, a frequência é definida por uma heurística chamada “gear”, a qual de-

fine um ganho entre consumo de energia e desempenho. Com o trace obtido a partir de uma pré-execução, a abordagem define blocos (*Basic blocks*) que realizam operações comuns, depois dessa classificação é obtido as fases da aplicação que correspondem a junção desses blocos. Para cada bloco é definido o ganho desejado. O ganho é a melhor configuração encontrada (frequência de processador) entre consumo de energia e desempenho para determinada fase da aplicação. Os resultados apresentam um ganho considerável para mais da metade das aplicações executadas, o melhor resultado obtido foi a redução do consumo de energia em 9% e do tempo de execução em 1%.

Para aplicações paralelas escritas com OpenMP, ??) apresentam uma abordagem que analisa as regiões paralelas de um programa, utilizando uma análise rigorosa usando a técnica Design of Experiments e Screening Design. Os autores realizaram experimentos com sete benchmarks, através das execuções eles concluíram que é possível obter um ganho considerável de energia e desempenho com a utilização da abordagem, dependendo das características comportamentais da aplicação. A técnica consiste na instrumentação manual de código para assinalar as regiões paralelas no código fonte. Diferente disso, o foco desse trabalho é direcionado na identificação automática dessas regiões paralelas, baseando-se em contadores de hardware específicos.

4 EXPERIMENTOS

APÊNDICE A — APENDICE 1

asdsaodk set of optional flags. An example, is the flag `STATIC`, which means that there is only one parser for the JVM when set to true.