

# Um Algoritmo de Escalonamento para Redução do Consumo de Energia em Computação em Nuvem

Pedro Paulo Vezza Campos

MACo499 – Trabalho de Formatura Supervisionado  
Instituto de Matemática e Estatística  
Universidade de São Paulo, São Paulo, Brasil  
`pedro@vezza.com.br`

3 de novembro de 2013

# Agenda

## Motivação & Conceitos

- Consumo energético
- Escalonamento de fluxos de trabalho
- Um algoritmo clássico: *Heterogeneous Earliest Finish Time*

## Experimentos

## Conclusões

# Agenda

## Motivação & Conceitos

- Consumo energético
  - Escalonamento de fluxos de trabalho
  - Um algoritmo clássico: *Heterogeneous Earliest Finish Time*

## Experimentos

## Conclusões

# Agenda

## Motivação & Conceitos

- Consumo energético
- Escalonamento de fluxos de trabalho
- Um algoritmo clássico: *Heterogeneous Earliest Finish Time*

## Experimentos

- O desenvolvimento de um novo algoritmo: Exatidão e Interações

## Conclusões

# Agenda

## Motivação & Conceitos

- Consumo energético
- Escalonamento de fluxos de trabalho
- Um algoritmo clássico: *Heterogeneous Earliest Finish Time*

## Experimentos

- O desenvolvimento de um novo algoritmo: Êxitos e frustrações

## Conclusões

# Agenda

## Motivação & Conceitos

- Consumo energético
- Escalonamento de fluxos de trabalho
- Um algoritmo clássico: *Heterogeneous Earliest Finish Time*

## Experimentos

- O desenvolvimento de um novo algoritmo: Êxitos e frustrações

## Conclusões

Reflexões das contribuições e resultados obtidos

# Agenda

## Motivação & Conceitos

- Consumo energético
- Escalonamento de fluxos de trabalho
- Um algoritmo clássico: *Heterogeneous Earliest Finish Time*

## Experimentos

- O desenvolvimento de um novo algoritmo: Êxitos e frustrações

## Conclusões

- Análise das contribuições e resultados obtidos

# Agenda

## Motivação & Conceitos

- Consumo energético
- Escalonamento de fluxos de trabalho
- Um algoritmo clássico: *Heterogeneous Earliest Finish Time*

## Experimentos

- O desenvolvimento de um novo algoritmo: Êxitos e frustrações

## Conclusões

- Análise das contribuições e resultados obtidos



# Agenda

## Motivação & Conceitos

- Consumo energético
- Escalonamento de fluxos de trabalho
- Um algoritmo clássico: *Heterogeneous Earliest Finish Time*

## Experimentos

- O desenvolvimento de um novo algoritmo: Êxitos e frustrações

## Conclusões

- Análise das contribuições e resultados obtidos

# Agenda

**1** Motivação

2 Conceitos

3 Experimentos

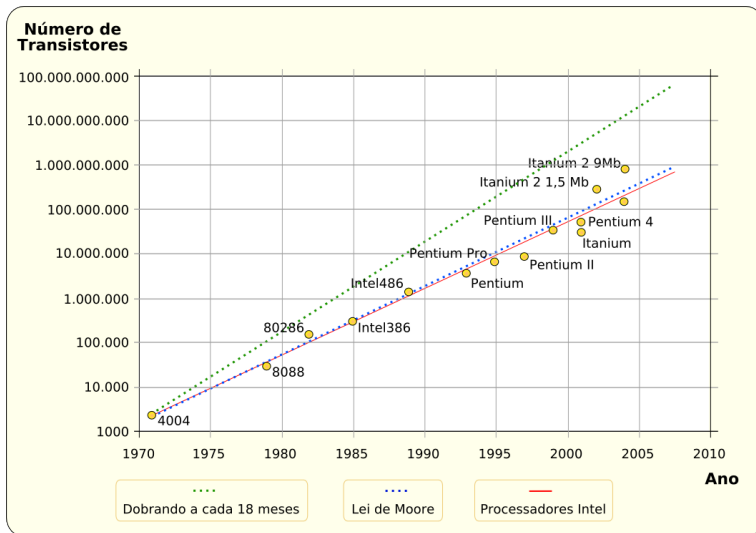


Figura: Lei de Moore <sup>1</sup>

<sup>1</sup> Fonte: Wikipédia, [http://pt.wikipedia.org/wiki/Ficheiro:Lei\\_de\\_moore\\_2006.svg.png](http://pt.wikipedia.org/wiki/Ficheiro:Lei_de_moore_2006.svg.png), em domínio público



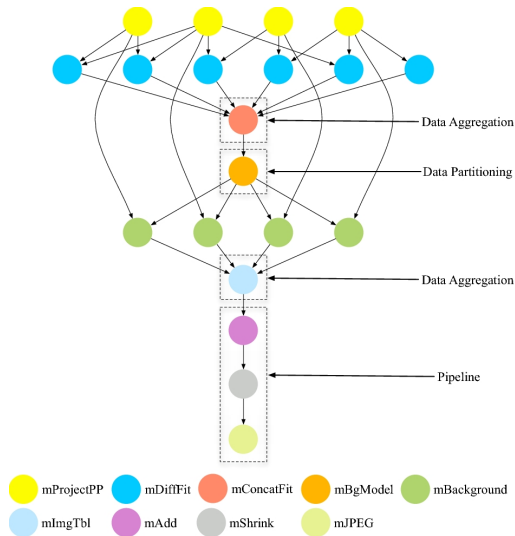


Figura: Montage: Gerador de mosaicos astronômicos <sup>2</sup>

<sup>2</sup>

Fonte: Projeto Pegasus,

<https://confluence.pegasus.isi.edu/display/pegasus/WorkflowGenerator>, sob a licença Apache V2

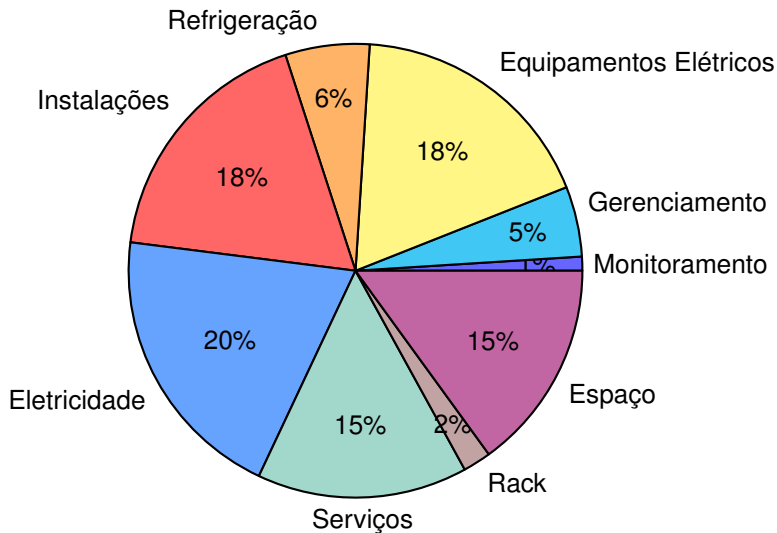
# Agenda

1 Motivação

2 Conceitos

3 Experimentos

# Computação em nuvem



# Estratégias para economia de energia

- **DVFS**: *Dynamic Voltage and Frequency Scaling*
- Migração de máquinas virtuais
- Algoritmos de escalonamento energeticamente eficientes



# Estratégias para economia de energia

- **DVFS**: *Dynamic Voltage and Frequency Scaling*
- **Migração** de máquinas virtuais
- **Algoritmos de escalonamento** energeticamente eficientes

# Estratégias para economia de energia

- **DVFS**: *Dynamic Voltage and Frequency Scaling*
- **Migração** de máquinas virtuais
- **Algoritmos de escalonamento** energeticamente eficientes

# HEFT: *Heterogeneous Earliest Finish Time*

- Publicado em **2002**
- Bastante aceito na comunidade científica (Quase **mil citações**)
- Duas fases: **priorização** e **seleção**

## HEFT: *Heterogeneous Earliest Finish Time*

- Publicado em **2002**
- Bastante aceito na comunidade científica (Quase **mil citações**)
- Duas fases: **priorização** e **seleção**

## HEFT: *Heterogeneous Earliest Finish Time*

- Publicado em **2002**
- Bastante aceito na comunidade científica (Quase **mil citações**)
- Duas fases: **priorização** e **seleção**

# Fase de priorização

- **Qual** tarefa **escalonar primeiro?**
- Algoritmo **offline**
- Ordenação topológica:

$$rank_u(n_i) = \overline{w_i} + \max_{n_j \in succ(n_i)} (\overline{c_{i,j}} + rank_u(n_j))$$

# Fase de priorização

- Qual tarefa **escalonar primeiro**?
- Algoritmo **offline**
- Ordenação topológica:

$$rank_u(n_i) = \overline{w}_i + \max_{n_j \in succ(n_i)} (\overline{c}_{i,j} + rank_u(n_j))$$

# Fase de priorização

- Qual tarefa **escalonar primeiro**?
- Algoritmo **offline**
- Ordenação topológica:

$$rank_u(n_i) = \overline{w}_i + \max_{n_j \in succ(n_i)} (\overline{c}_{i,j} + rank_u(n_j))$$

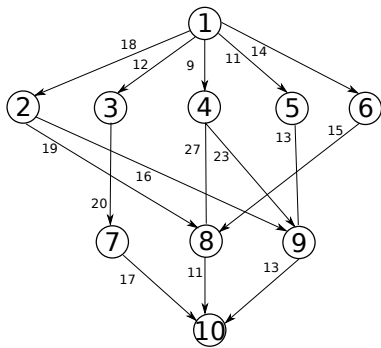


# Fase de seleção

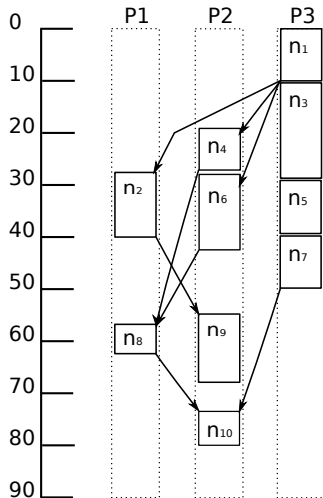
- Minimizar o **tempo mais cedo de conclusão** (*Earliest finish time*)
- Busca por um **espaço vago** grande o suficiente

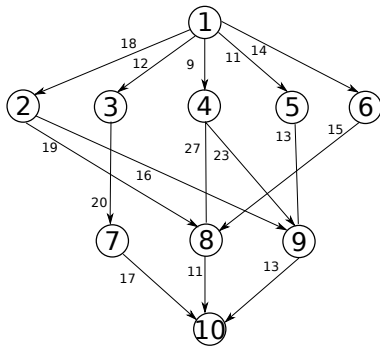
# Fase de seleção

- Minimizar o **tempo mais cedo de conclusão** (*Earliest finish time*)
- Busca por um **espaço vago** grande o suficiente

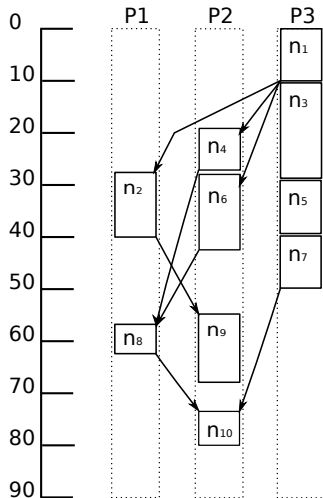


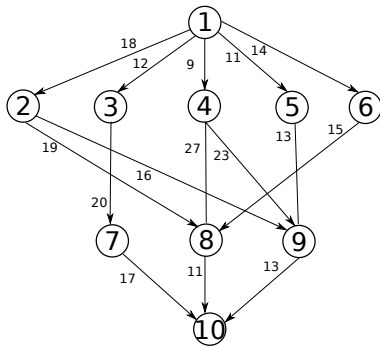
Tarefa	P1	P2	P3	$rank_U(n_i)$
1	14	16	9	108.000
2	13	19	18	77.000
3	11	13	19	80.000
4	13	8	17	80.000
5	12	13	10	69.000
6	13	16	9	63.333
7	7	15	11	42.667
8	5	11	14	35.667
9	18	12	20	44.333
10	21	7	16	14.667



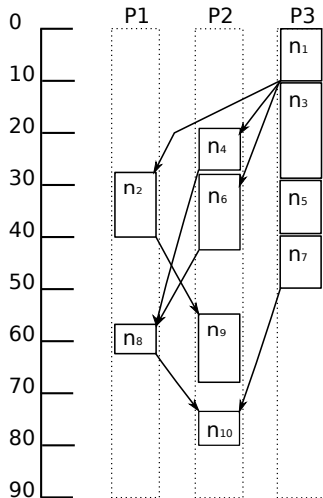


Tarefa	P1	P2	P3	$rank_U(n_i)$
1	14	16	9	108.000
2	13	19	18	77.000
3	11	13	19	80.000
4	13	8	17	80.000
5	12	13	10	69.000
6	13	16	9	63.333
7	7	15	11	42.667
8	5	11	14	35.667
9	18	12	20	44.333
10	21	7	16	14.667





Tarefa	P1	P2	P3	$rank_U(n_i)$
1	14	16	9	108.000
2	13	19	18	77.000
3	11	13	19	80.000
4	13	8	17	80.000
5	12	13	10	69.000
6	13	16	9	63.333
7	7	15	11	42.667
8	5	11	14	35.667
9	18	12	20	44.333
10	21	7	16	14.667



# Agenda

1 Motivação

2 Conceitos

3 Experimentos

# Simuladores

CloudSim lançado em **2010**, já na versão 3, quase 300 citações

WorkflowSim lançado em **abril de 2013**

CloudSim\_DVFS lançado em **junho de 2013** (!!)

# Simuladores

CloudSim lançado em **2010**, já na versão 3, quase 300 citações

WorkflowSim lançado em **abril de 2013**

CloudSim\_DVFS lançado em **junho de 2013** (!!)



# Simuladores

**CloudSim** lançado em **2010**, já na versão 3, quase 300 citações

**WorkflowSim** lançado em **abril de 2013**

**CloudSim\_DVFS** lançado em **junho de 2013** (!!)

# PowerHEFT: Algoritmo proposto

- Variante do HEFT, faz uso de uma estratégia de lookahead