

# Escalonamento de Tarefas usando Implementações Híbridas GPU/CPU das Heurísticas Min-min e Max-min no Cloudsim

Rafael Schmid, Edson Cáceres

*rafaelfschmid@gmail.com, edson@facom.ufms.br*

2 de Outubro de 2018

1 Introdução

2 Algoritmos

3 Experimentos e Resultados

- Cenário 1: Relação entre Tarefas e Máquinas Fixa
- Cenário 2: Poder Computacional e Quantidade de Tarefas Fixos

4 Conclusões

## 1 Introdução

## 2 Algoritmos

## 3 Experimentos e Resultados

- Cenário 1: Relação entre Tarefas e Máquinas Fixa
- Cenário 2: Poder Computacional e Quantidade de Tarefas Fixos

## 4 Conclusões

- Em geral, uma grande **aplicação** pode ser **decomposta** em um conjunto de **tarefas menores** e executadas em **múltiplos processadores**.
- Em ambientes de computação heterogênea esses **processadores possuem diferentes capacidades** computacionais.
- A **forma como essas tarefas serão distribuídas** entre eles é o ponto chave para atingir alto desempenho.
- Um exemplo é o ambiente de computação em nuvem, onde os recursos são dinamicamente alocados, de acordo com a necessidade do usuário.

Esse problema é conhecido como o **Problema do Escalonamento em Computação Heterogênea (HCSP)**.

- A métrica mais comum do HCSP é o **makespan**.
  - **Tempo** gasto a partir do momento em que a **primeira** tarefa começa até o momento em que a **última** tarefa é **completada**.
  - *Definido pela máquina que termina por último.*

# Introdução

	T1	T2	T3	T4	T5	T6	T7	T8
M1	23	41	12	39	17	26	57	82
M2	32	43	21	31	71	62	39	28
M3	24	14	13	93	73	29	75	11

# Introdução

	T1	T2	T3	T4	T5	T6	T7	T8	
M1	23	41	12	39	17	26	57	82	$23 + 39 + 26 = 88$
M2	32	43	21	31	71	62	39	28	$71 + 39 = 110$
M3	24	14	13	93	73	29	75	11	$14 + 13 + 11 = 38$

# Introdução

	T1	T2	T3	T4	T5	T6	T7	T8	
M1	23	41	12	39	17	26	57	82	$23 + 39 + 26 = 88$
M2	32	43	21	31	71	62	39	28	$71 + 39 = 110$
M3	24	14	13	93	73	29	75	11	$14 + 13 + 11 = 38$
									<b>makespan = 110</b>

O nosso objetivo é minimizar o *makespan*.



1 Introdução

2 Algoritmos

3 Experimentos e Resultados

- Cenário 1: Relação entre Tarefas e Máquinas Fixa
- Cenário 2: Poder Computacional e Quantidade de Tarefas Fixos

4 Conclusões

# Heurística Min-Min Sort

	T1	T2	T3	T4	T5	T6	T7	T8
M1	23	41	12	39	17	26	57	82
M2	32	43	21	31	71	62	39	28
M3	24	14	13	93	73	29	75	11

- O primeiro passo é realizar uma ordenação segmentada (crescente).

M1	<b>12</b> (T3)	17(T5)	23(T1)	26(T6)	39(T4)	41(T2)	57(T7)	82(T8)
M2	<b>21</b> (T3)	28(T8)	31(T4)	32(T1)	39(T7)	43(T2)	62(T6)	71(T5)
M3	<b>11</b> (T8)	13(T3)	14(T2)	24(T1)	29(T6)	73(T5)	75(T7)	93(T4)

- Depois, escolhe a tarefa que gera o menor *makespan*.

# Heurística Min-Min Sort

- O escalonamento da tarefa T8 na máquina M3 é o que gera o menor *makespan*.

M1	<b>12</b> (T3)	17(T5)	23(T1)	26(T6)	39(T4)	41(T2)	57(T7)	82(T8)
M2	<b>21</b> (T3)	28(T8)	31(T4)	32(T1)	39(T7)	43(T2)	62(T6)	71(T5)
M3	<b>11</b> (T8)	13(T3)	14(T2)	24(T1)	29(T6)	73(T5)	75(T7)	93(T4)

$$M1 = 0$$

$$M2 = 0$$

$$M3 = 11$$

# Heurística Min-Min Sort

- Como a tarefa T8 já foi escalonada, ela é removida da lista de escalonamento.
- A tarefa que gera o menor *makespan* na próxima iteração é a tarefa T3 na máquina M2.

M1	<b>12</b> (T3)	17(T5)	23(T1)	26(T6)	39(T4)	41(T2)	57(T7)	
M2	<b>21</b> (T3)		31(T4)	32(T1)	39(T7)	43(T2)	62(T6)	71(T5)
M3		<b>13</b> (T3)	14(T2)	24(T1)	29(T6)	73(T5)	75(T7)	93(T4)

$$M1 = 0$$

$$M2 = 0$$

$$M3 = 11$$

# Heurística Min-Min Sort

- T3 é removida da lista de escalonamento.
- Na próxima iteração a tarefa T2 é escalonada na máquina M3.

M1	<b>17</b> (T5)	23(T1)	26(T6)	39(T4)	41(T2)	57(T7)	
M2		<b>31</b> (T4)	32(T1)	39(T7)	43(T2)	62(T6)	71(T5)
M3		<b>14</b> (T2)	24(T1)	29(T6)	73(T5)	75(T7)	93(T4)

$$M1 = 12$$

$$M2 = 0$$

$$M3 = 11 + 14 = 25$$

# Heurística Max-Min Sort

	T1	T2	T3	T4	T5	T6	T7	T8
M1	23	41	12	39	17	26	57	82
M2	32	43	21	31	71	62	39	28
M3	24	14	13	93	73	29	75	11

- O primeiro passo é realizar uma ordenação segmentada (decrecente).

M1	<b>82</b> (T8)	57(T7)	41(T2)	39(T4)	26(T6)	23(T1)	17(T5)	12(T3)
M2	<b>71</b> (T5)	62(T6)	43(T2)	39(T7)	32(T1)	31(T4)	28(T8)	21(T3)
M3	<b>93</b> (T4)	75(T7)	73(T5)	29(T6)	24(T1)	14(T2)	13(T3)	11(T8)

# Heurística Max-Min Sort

- Depois, escolhe a tarefa maior.

M1	<b>82</b> (T8)	57(T7)	41(T2)	39(T4)	26(T6)	23(T1)	17(T5)	12(T3)
M2	<b>71</b> (T5)	62(T6)	43(T2)	39(T7)	32(T1)	31(T4)	28(T8)	21(T3)
M3	<b>93</b> (T4)	75(T7)	73(T5)	29(T6)	24(T1)	14(T2)	13(T3)	11(T8)

$$M1 = 0$$

$$M2 = 0$$

$$M3 = 0$$

# Heurística Max-Min Sort

- E escalona na máquina que gera o menor *makespan*.

M1	82(T8)	57(T7)	41(T2)	<b>39(T4)</b>	26(T6)	23(T1)	17(T5)	12(T3)
M2	71(T5)	62(T6)	43(T2)	39(T7)	32(T1)	<b>31(T4)</b>	28(T8)	21(T3)
M3	<b>93(T4)</b>	75(T7)	73(T5)	29(T6)	24(T1)	14(T2)	13(T3)	11(T8)

$$M1 = 0$$

$$M2 = 31$$

$$M3 = 0$$



# Heurística Max-Min Sort

M1	<b>82</b> (T8)	57(T7)	41(T2)		26(T6)	23(T1)	17(T5)	12(T3)
M2	<b>71</b> (T5)	62(T6)	43(T2)	39(T7)	32(T1)		28(T8)	21(T3)
M3		<b>75</b> (T7)	73(T5)	29(T6)	24(T1)	14(T2)	13(T3)	11(T8)

$$M1 = 0$$

$$M2 = 31$$

$$M3 = 0$$

# Heurística Max-Min Sort

M1	<b>82(T8)</b>	57(T7)	41(T2)		26(T6)	23(T1)	17(T5)	12(T3)
M2	71(T5)	62(T6)	43(T2)	39(T7)	32(T1)		<b>28(T8)</b>	21(T3)
M3		75(T7)	73(T5)	29(T6)	24(T1)	14(T2)	13(T3)	<b>11(T8)</b>

$$M1 = 0$$

$$M2 = 31$$

$$M3 = 11$$

	T1	T2	T3	T4	T5	T6	T7	T8
M1	23	41	12	39	17	26	57	82
M2	32	43	21	31	71	62	39	28
M3	24	14	13	93	73	29	75	11

- As tarefas são distribuídas entre as máquinas sem levar em conta seus tamanhos.

	T1	T2	T3	T4	T5	T6	T7	T8
M1	23	41	12	39	17	26	57	82
M2	32	43	21	31	71	62	39	28
M3	24	14	13	93	73	29	75	11

	T1	T2	T3	T4	T5	T6	T7	T8	
M1	23	41	12	39	17	26	57	82	$23 + 39 + 57 = 119$
M2	32	43	21	31	71	62	39	28	$43 + 71 + 28 = 142$
M3	24	14	13	93	73	29	75	11	$13 + 29 = 42$
									<b>makespan = 142</b>

1 Introdução

2 Algoritmos

3 Experimentos e Resultados

- Cenário 1: Relação entre Tarefas e Máquinas Fixa
- Cenário 2: Poder Computacional e Quantidade de Tarefas Fixos

4 Conclusões

# Experimentos

- O número de instruções das tarefas foram gerados aleatoriamente
- O ambiente de teste utilizado é consistente.

## 1 Introdução

## 2 Algoritmos

## 3 Experimentos e Resultados

- Cenário 1: Relação entre Tarefas e Máquinas Fixa
- Cenário 2: Poder Computacional e Quantidade de Tarefas Fixos

## 4 Conclusões

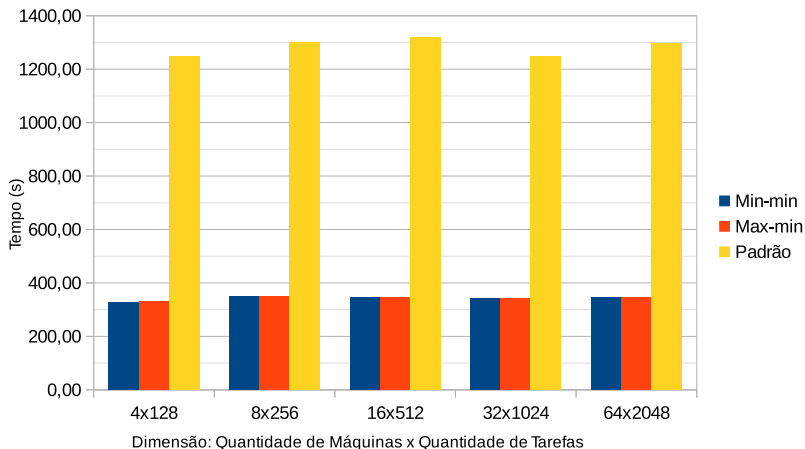
# Cenário 1: Relação entre Tarefas e Máquinas Fixa

Quantidade de Máquinas Virtuais				Total de Máquinas Virtuais	Total de Tarefas	Total de Datacenter
MIPS 1000	MIPS 500	MIPS 250	MIPS 125			
1	1	1	1	4	128	1
2	2	2	2	8	256	1
4	4	4	4	16	512	2
8	8	8	8	32	1024	3
16	16	16	16	64	2048	6

**Tabela:** Cenário 1: Relação entre Tarefas/Máquinas igual em todos os testes.



# Cenário 1: Relação entre Tarefas e Máquinas Fixa



**Figura:** Média do Tempo de Execução de Cada Heurística para as Dimensões do Problema.

# Cenário 1: Relação entre Tarefas e Máquinas Fixa

- Ao manter as mesmas proporções entre tarefas e máquinas os resultados das heurísticas min-min e max-min foram semelhantes em todas as dimensões do problema.
- Então, surgiu a seguinte dúvida:
  - **As heurísticas min-min e max-min são muito afetadas se aumentarmos a quantidade de máquinas?**

## 1 Introdução

## 2 Algoritmos

## 3 Experimentos e Resultados

- Cenário 1: Relação entre Tarefas e Máquinas Fixa
- Cenário 2: Poder Computacional e Quantidade de Tarefas Fixos

## 4 Conclusões

## Cenário 2: Poder Computacional do Ambiente e Quantidade de Tarefas Fixos

Quantidade de Datacenter	Quantidade de VMs	MIPS de cada VM	MIPS de cada Host	Total de Tarefas
1	2	1100, 900	1100	1024
2	4	550, 525, 475, 450	550	1024
3	8	265, 262, 257, 253, 248, 244, 239, 232	265	1024
5	16	160, 158, 153, 148, 144, 137, 132, 129, 126, 117, 113, 107, 101, 94, 93, 88	160	1024

**Tabela:** Cenário 2: Poder computacional do ambiente e quantidade de tarefas iguais em todos os testes.

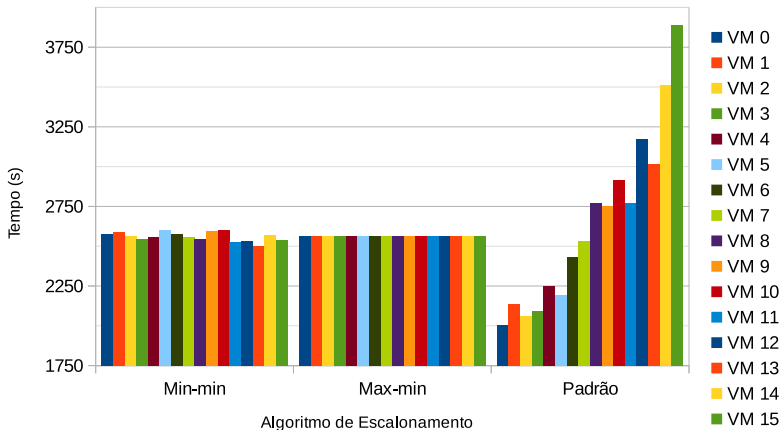
## Cenário 2: Poder Computacional e Quantidade de Tarefas Fixos

	Min-min	Max-min	Padrão do Cloudsim
Dimensão	<i>Makespan</i>	<i>Makespan</i>	<i>Makespan</i>
2x1024	2563,75	2564,93	2826,17
4x1024	2565,15	2564,71	2806,53
8x1024	2559,08	2564,53	2756,42
16x1024	2575,03	2564,54	3429,60

**Tabela:** Média e Desvio Padrão dos *Makespans* para cada Dimensão.

# Cenário 2: Poder Computacional e Quantidade de Tarefas Fixos

A Figura abaixo apresenta o *makespan* de cada VM utilizando uma entrada com 16 máquinas virtuais e 1024 tarefas.



## Cenário 2: Poder Computacional e Quantidade de Tarefas Fixos

- O percentual de utilização dos datacenters foi obtido pela soma do MIPS de todas as VMs alocadas em um *Datacenter* dividido pelo total de MIPS dos hosts existentes nele.
- No *datacenter* 0 foram alocadas 3 VMs, com MIPS igual a: 265, 262 e 257. Esse datacenter possui 3 hosts com 265 MIPS. A utilização desse datacenter foi:  $\frac{265+262+257}{265+265+265} = 0,9862 = 98,62\%$ .

Datacenter 0	60,61%	93,94%	98,62%	98,13%
Datacenter 1		27,27%	93,71%	89,38%
Datacenter 2			59,25%	80,63%
Datacenter 3				88,54%
Datacenter 4				60,00%
Utilização Total	60,61%	121,21%	251,57%	416,67%

**Tabela:** Percentual de Utilização dos Datacenters e Preço Hipotético Sugerido

## Cenário 2: Poder Computacional e Quantidade de Tarefas Fixos

Datacenter 0	60,61%	93,94%	98,62%	98,13%
Datacenter 1		27,27%	93,71%	89,38%
Datacenter 2			59,25%	80,63%
Datacenter 3				88,54%
Datacenter 4				60,00%
Utilização Total	60,61%	121,21%	251,57%	416,67%
Preço Sugerido de cada Datacenter	100,00	50,00	24,09	14,55

**Tabela:** Percentual de Utilização dos Datacenters e Preço Hipotético Sugerido



1 Introdução

2 Algoritmos

3 Experimentos e Resultados

- Cenário 1: Relação entre Tarefas e Máquinas Fixa
- Cenário 2: Poder Computacional e Quantidade de Tarefas Fixos

4 Conclusões

# Conclusões

- Inclusão do Max-min e Min-min como escalonadores nativos do Cloudsim.
- As dimensões do problema não afetou as heurísticas min-min e max-min.
- Isso é importante porque sugere ao usuário escolher o ambiente de computação em nuvem mais barato.
  - Seja ele composto por poucas máquinas potentes ou várias máquinas mais fracas.

- Simular a precificação de ambientes reais de computação em nuvem.
  - Gerar diversos cenários que identifiquem os ambientes mais indicados para cada dimensão do problema.
- Efetuar testes em ambientes:
  - Inconsistentes.
  - Com grande divergência no tamanho das tarefas
  - Com grande divergência no desempenho das máquinas.

Obrigado!