

Um Algoritmo Evolutivo Híbrido para o Escalonamento de *Workflows* Científicos em Ambientes de Nuvem

Luan Teylo¹, Ubiratam de Paula², Yuri Frota¹, Daniel de Oliveira¹,
Lúcia M. A. Drummond¹

¹Instituto de Computação
Universidade Federal Fluminense

²COPPE
Universidade Federal do Rio de Janeiro

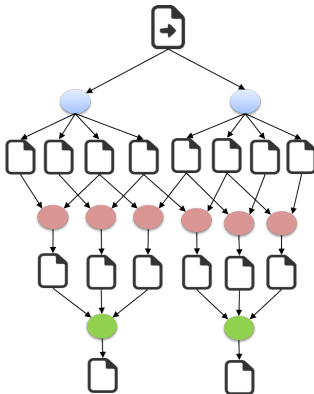
II Escola Regional de Alto Desempenho do Rio de Janeiro, 2016

Agenda

- 1 Definição do Problema
- 2 Algoritmo Evolutivo Híbrido
- 3 Resultados Experimentais
- 4 Conclusão e Trabalhos Futuros

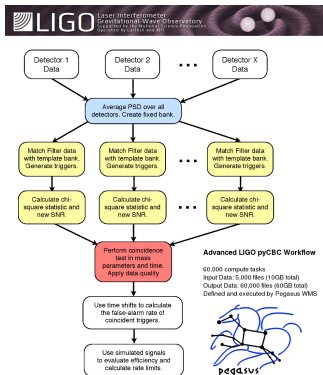
Workflows Científicos

- Os chamados experimentos *in silico* são comumente representados como uma **cadeia de aplicações**, na qual a **saída** de um programa é a **entrada** para o outro.
- Essas aplicações são modeladas na forma de grafo acíclico dirigido, chamados de *workflows* científicos.



Workflows Científicos

- Um exemplo recente do uso de *workflows* científicos foi o utilizado nas detecções de ondas gravitacionais feitas pelo observatório americano LIGO.

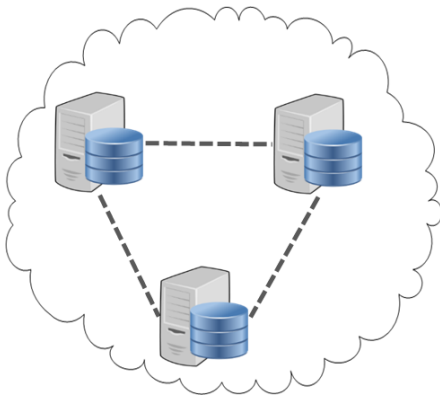


O LIGO pyCBC contém 60.000 tarefas e processa mais de 65.000 arquivos (70 GB).

Figura: *LIGO pyCBC workflow*.

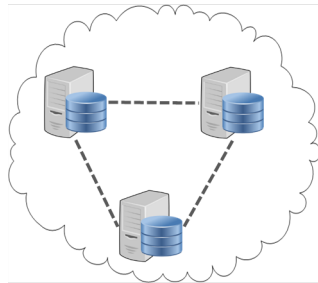
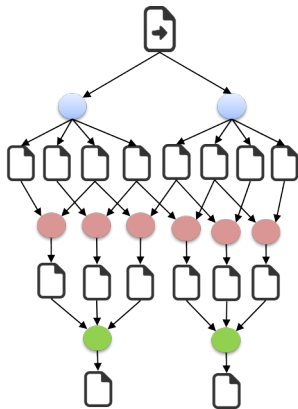
Ambiente

- Os *Workflows* científicos têm uma alta demanda computacional e, portanto, é necessário utilizar ambientes computacionais de alto desempenho.
- As nuvens computacionais surgem como uma alternativa viável, tanto do ponto de vista financeiro como da facilidade técnica.



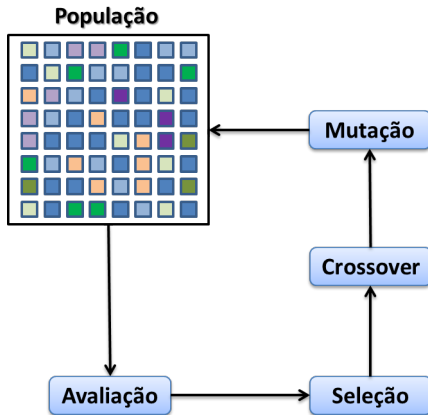
O Problema de Escalonamento

- Neste trabalho, o escalonamento de tarefas e a distribuição de dados são abordados de forma conjunta. O objetivo é reduzir o tempo total de execução de *workflows* em ambientes de nuvem.



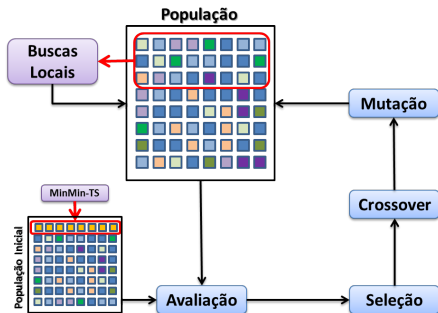
Algoritmo Evolutivo (EA)

Algoritmos Evolutivos são métodos de otimização inspirados nos mecanismos de evolução biológica observados na natureza.



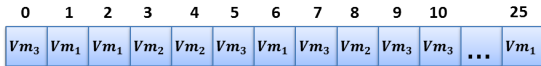
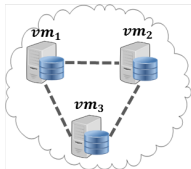
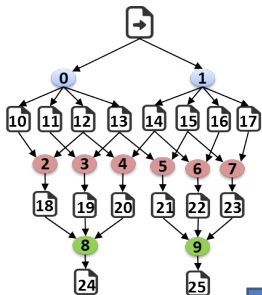
Algoritmo Evolutivo Híbrido (HEA)

- A população inicial contém uma solução gerada com a heurística MinMin-TS.
- Buscas locais são efetuadas nas melhores soluções.



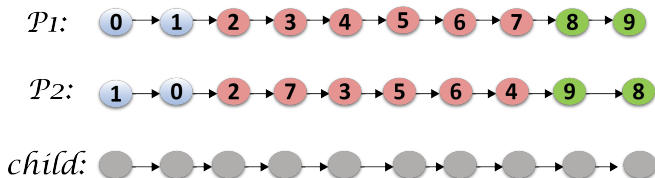
Representação

- O cromossomo é composto por duas estruturas, que representam: (i) a ordem de execução das tarefas, e (ii) a distribuição das tarefas e dos dados.



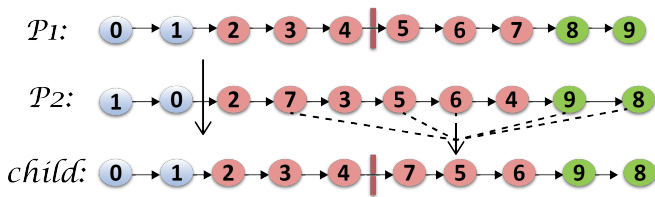
Operador de Crossover

Ordem de execução



Operador de Crossover

Ordem de execução



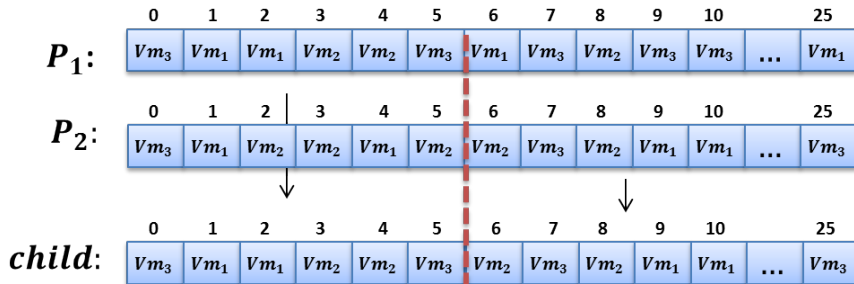
Operador de Crossover

Vetor de Alocação



Operador de Crossover

Vetor de Alocação

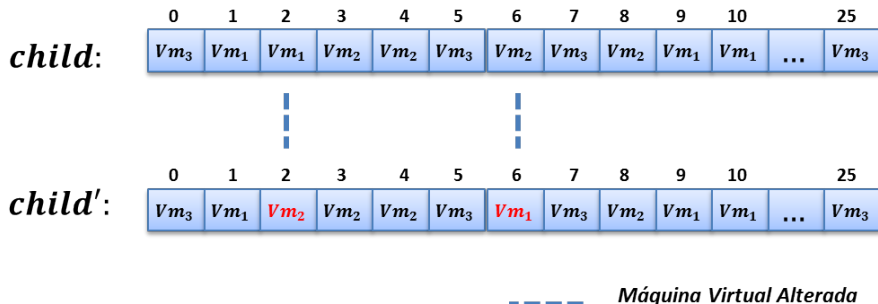


Operador de Mutação

- O operador de mutação é aplicado apenas no vetor de alocação.

Operador de Mutação

- O operador de mutação é aplicado apenas no vetor de alocação.



Buscas Locais

Foram definidas três buscas locais e uma heurística de busca:

- *Swap-vm*
- *Swap-position*
- *Move-element*
- Heurística *Path-relinking*

Todas as buscas locais são *first-improvement*.

Buscas Locais

Swap-vm



vm_1



vm_1



...



vm_2



vm_2



...



vm_3



vm_3

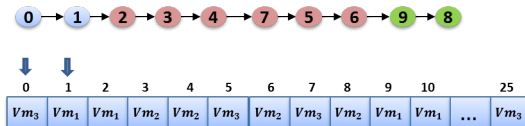


...



Buscas Locais

Swap-vm



vm_1



vm_1



...



vm_2



vm_2



...



vm_3



vm_3

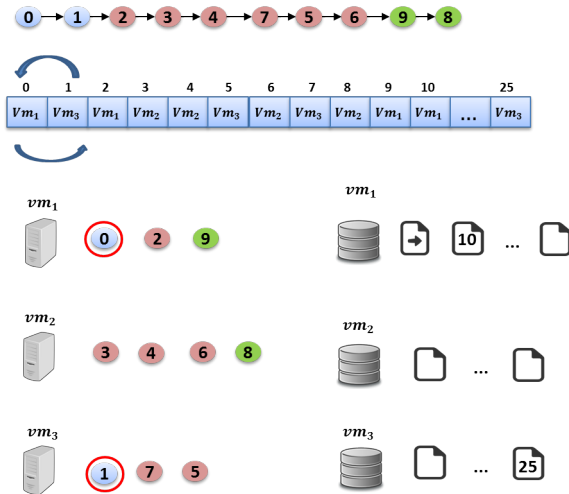


...



Buscas Locais

Swap-vm



Buscas Locais

Swap-position

Buscas Locais

Swap-position



vm_1



vm_1



...



vm_2



vm_2



...



vm_3



vm_3

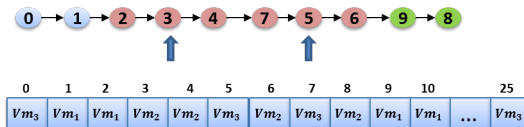


...



Buscas Locais

Swap-position



vm_1



1 2 9

vm_1



10

...



vm_2



3 4 6 8

vm_2



...



vm_3



0 7 5

vm_3

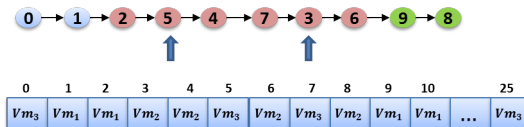


...

25

Buscas Locais

Swap-position



vm_1



vm_1



...



vm_2



vm_2



...



vm_3



vm_3



...



Buscas Locais

Move-Element

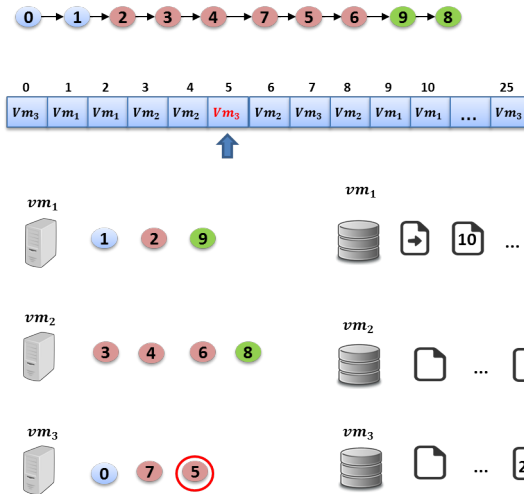
Buscas Locais

Move-Element



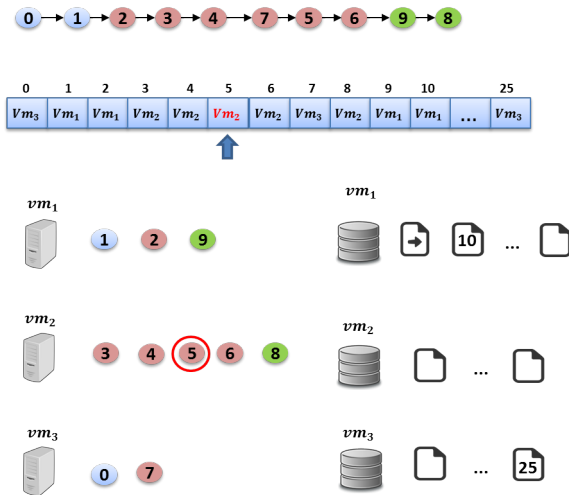
Buscas Locais

Move-Element



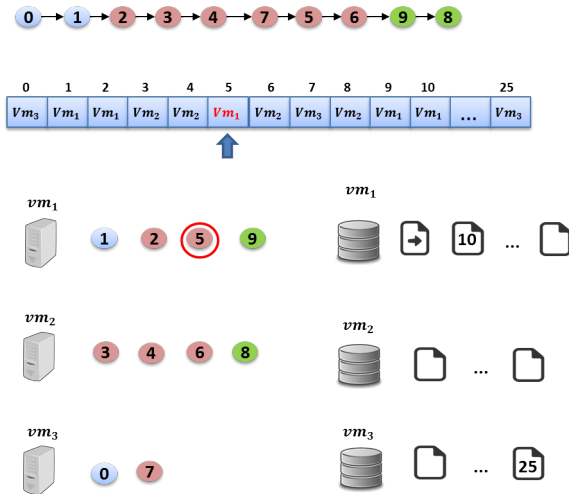
Buscas Locais

Move-Element



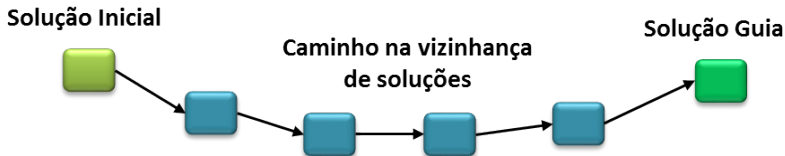
Buscas Locais

Move-Element



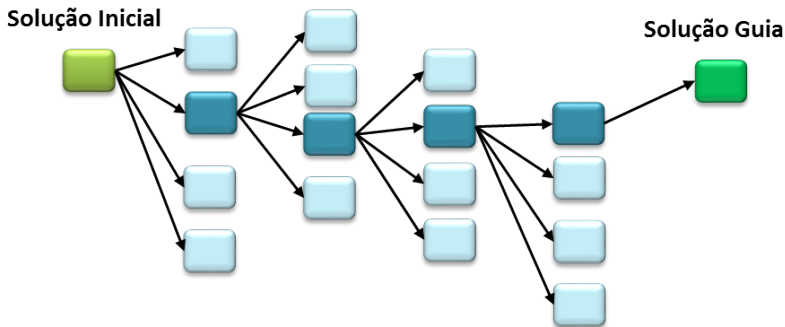
Path-Relinking

- As melhores soluções encontradas pela metaheurística são guardadas em um **conjunto elite**
- A estratégia de *path-relinking* explora soluções intermediárias ao longo da trajetória que conecta duas soluções pertencentes a esse conjunto



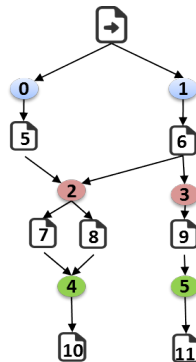
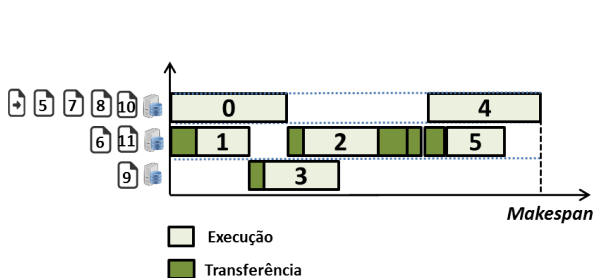
Path-Relinking

- O caminho é gerado selecionando movimentos que introduzem características da **Solução Guia** na **Solução Inicial**.



Função de *Fitness*

A qualidade de um indivíduo é estimada conforme o *makespan* de sua solução.



Resultados Experimentais

Foram realizados os seguintes experimentos:

- **Avaliação teórica** comparando duas propostas encontradas na literatura, a heurística *MinMin-Task Schedule* [Mahesswaran *et al.* 1999] e o Algoritmo Evolutivo (EA) [Szabo *et al.* 2013].
- **Avaliação prática** com o *workflow* SciPhy [Ocaña *et al.* 2011], utilizando a máquina de execução *SciCumulus* [Oliveira *et al.* 2010].

Resultados Experimentais

Configuração:

- Intel Core i7-3770 CPU 3.40GHz com 12Gb memória e sistema operacional Ubuntu 14.04.
- Os dados sintéticos utilizados no experimento teórico foram gerados pelo *Workflow Generator* [Silva et al. 2014] e são baseados em *workflows* reais.
- A avaliação prática foi realizada no serviço de nuvem EC2 da Amazon.

Resultados Experimentais

HEA vs MinMin-TS e EA

Tabela: Avaliação teórica entre MinMin-TS, EA e HEA.

Workflow	MinMin-TS		EA		HEA		
	Makespan(min)	Makespan(min)	Std. dev	Tempo(s)	Makespan(min)	Std. dev	time(s)
CyberShake_30	54.52	56.15	1.31	11.2	50.45	0.35	124.64
CyberShake_50	65.25	79.13	6.21	15.66	62.54	1.41	125.9
CyberShake_100	83.87	127.71	2.89	37.31	78.73	2.22	300.17
Epigenomics_24	1123.53	1027.89	22.33	9.23	998.97	1.64	67.96
Epigenomics_46	2310.08	1993.54	77.91	27.89	1856.45	37.73	54.56
Epigenomics_100	17140.17	17899.87	2256.66	49.5	13018.03	406.02	291.45
Inspiral_30	374.62	414.23	21.3	14.33	313.18	0	75.06
Inspiral_50	610.02	741.59	25.55	24.07	458.58	9.75	136.37

Resultados Experimentais

HEA vs MinMin-TS e EA

- Foram obtidos resultados mais significativos em *workflows* com uma maior quantidade de dados.

Tabela: Avaliação teórica entre MinMin-TS, EA e HEA.

Workflow	MinMin-TS		EA		HEA		
	Makespan(min)	Makespan(min)	Std. dev	Tempo(s)	Makespan(min)	Std. dev	time(s)
CyberShake_30	54.52	56.15	1.31	11.2	50.45	0.35	124.64
CyberShake_50	65.25	79.13	6.21	15.66	62.54	1.41	125.9
CyberShake_100	83.87	127.71	2.89	37.31	78.73	2.22	300.17
Epigenomics_24	1123.53	1027.89	22.33	9.23	998.97	1.64	67.96
Epigenomics_46	2310.08	1993.54	77.91	27.89	1856.45	37.73	54.56
Epigenomics_100	17140.17	17899.87	2256.66	49.5	13018.03	406.02	291.45
Inspiral_30	374.62	414.23	21.3	14.33	313.18	0	75.06
Inspiral_50	610.02	741.59	25.55	24.07	458.58	9.75	136.37

Resultados Experimentais

HEA vs MinMin-TS e EA

- Foram obtidos resultados mais significativos em *workflows* com uma maior quantidade de dados.

Tabela: Avaliação teórica entre MinMin-TS, EA e HEA.

Workflow	MinMin-TS		EA		HEA		
	Makespan(min)	Makespan(min)	Std. dev	Tempo(s)	Makespan(min)	Std. dev	time(s)
CyberShake_30	54.52	56.15	1.31	11.2	50.45	0.35	124.64
CyberShake_50	65.25	79.13	6.21	15.66	62.54	1.41	125.9
CyberShake_100	83.87	127.71	2.89	37.31	78.73	2.22	300.17
Epigenomics_24	1123.53	1027.89	22.33	9.23	998.97	1.64	67.96
Epigenomics_46	2310.08	1993.54	77.91	27.89	1856.45	37.73	54.56
Epigenomics_100	17140.17	17899.87	2256.66	49.5	13018.03	406.02	291.45
Inspiral_30	374.62	414.23	21.3	14.33	313.18	0	75.06
Inspiral_50	610.02	741.59	25.55	24.07	458.58	9.75	136.37

Resultados Experimentais

Avaliação prática

Tabela: Avaliação prática entre Greedy (*Scicumulus*) e HEA.

Algoritmo	Média	Mediana	Min	Max
Greedy	268.99	266.39	250.26	294.45
HEA	210.01	207.08	198.54	221.05

Conclusão

- A abordagem proposta demonstrou resultados satisfatórios e tempo de execução aceitável (média de 2 minutos).
- Houve um ganho médio de 20% em relação ao MinMin-TS, 36% em relação ao EA e 21,93% em relação ao *Greedy* (avaliação prática).

Conclusão

- Foram observados resultados mais significativos em *workflows* com muita movimentação de dados (GB).
- Em trabalhos futuros serão incluídas técnicas multi-objetivo e alocação dinâmica de recursos.

Obrigado pela atenção!

Perguntas?