



Synapse meets Slurm

Proposta de um middleware para
paralelização de algoritmos de otimização populacionais

Arthur M. P. Gabardo¹ Thiago P. Tancred^{1,2} Pablo A. Jaskowiak¹

¹Núcleo de Simulação e Otimização de Sistemas Dinâmicos (NSO) — Universidade Federal de Santa Catarina (UFSC)

²Laboratório de Simulação Naval (LaSiN) — Universidade Federal de Santa Catarina (UFSC)

- Em sistemas complexos de engenharia, diferentes disciplinas podem ter objetivos e restrições conflitantes;
- A Otimização Multidisciplinar (MDO) proporciona uma abordagem holística e integrada, buscando o aprimoramento global de novos projetos;
- As meta-heurísticas (e.g. algoritmos genéticos) permitem explorar espaços de solução de maneira eficiente, mesmo quando abordagens analíticas tradicionais falham.

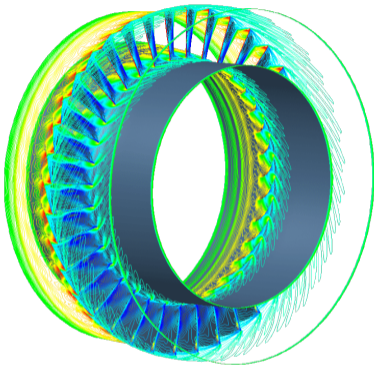


Figura 1: Simulação DFC do rotor de um compressor axial (Enteknograte, 2023)

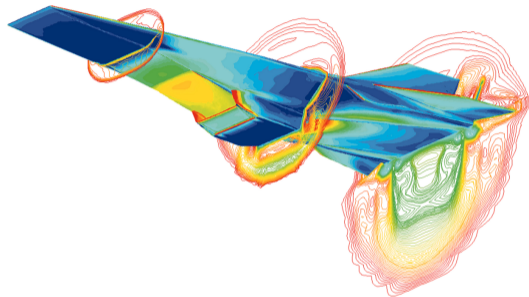


Figura 2: Escoamento e distribuição de pressões em torno do X-43 a Mach 7 (NASA, 1997)

- O *Synapse Engenharia Multidisciplinar* é um software de otimização mono e multiobjetivo;
- Ambiente colaborativo e amigável ao usuário;
- Integração com ferramentas de planejamento de experimentos e análise de "*fitness*";
- Diversas meta-heurísticas baseadas em população e gradiente.

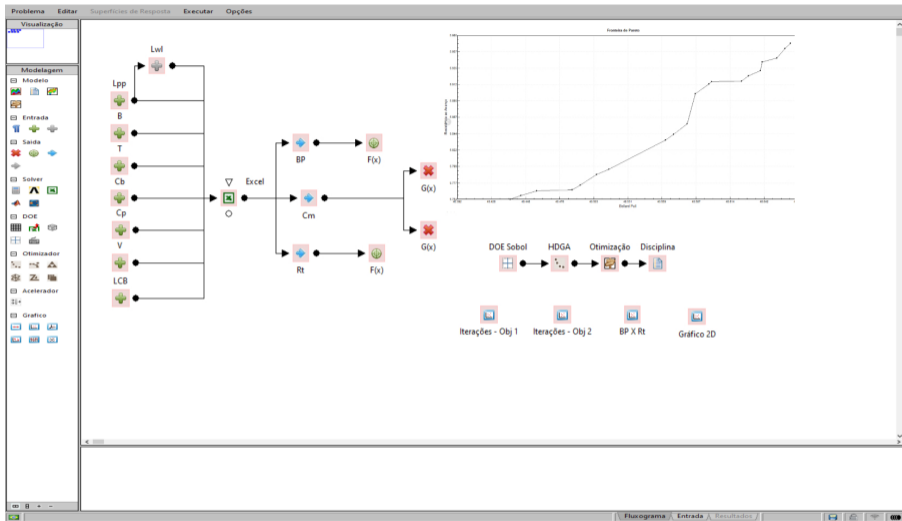
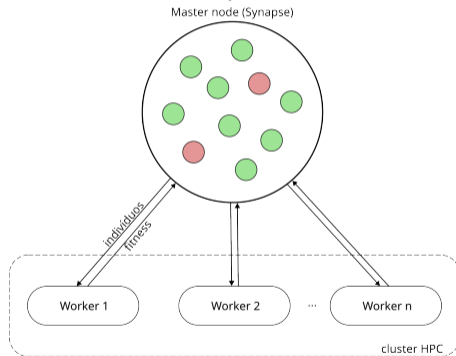


Figura 3: Modelagem de problema de otimização por fluxogramas no Synapse

- Processo de busca é limitado por recursos computacionais;
- Synapse permite o processamento paralelo, mas não o distribuído;
- Alto nível de maturidade do software dificulta grandes mudanças.

- *Middleware* que permita ao Synapse tirar proveito de sistemas de processamento distribuído de alto desempenho;
- Modelo de paralelização *master-worker*;
- Menor número de dependências possível;
- Comunicação *Synapse-cluster* via ssh interpretada pelo *middleware*.



- (i) Receber arquivos contendo indivíduos que devem ser avaliados;
- (ii) Alocar recursos e distribuir os indivíduos nos nós de computação;
- (iii) Monitorar e gerenciar uma fila interna ao middleware, realizando balanceamento de carga;
- (iv) Compactar arquivos de resultados e transferi-los.

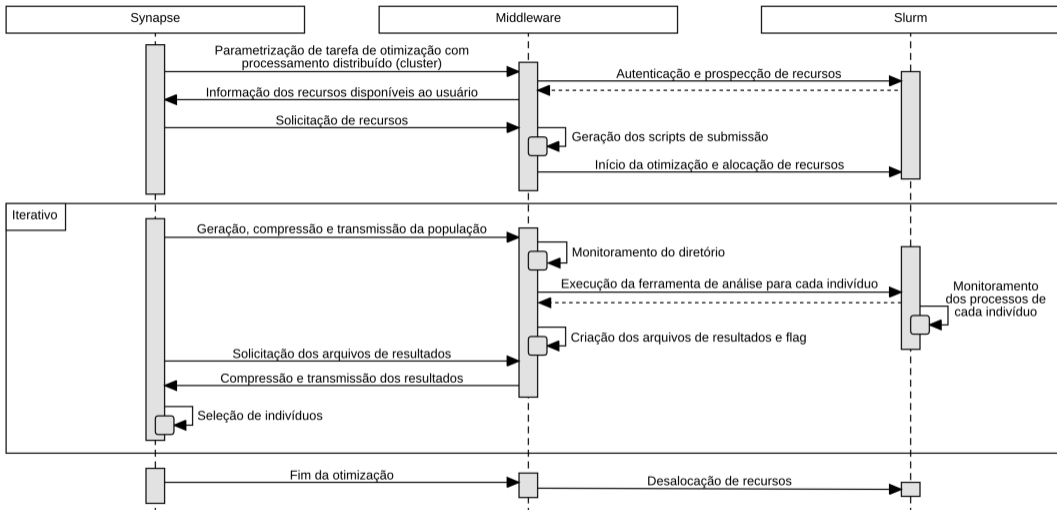


Figura 4: Diagrama de sequência do *middleware* proposto

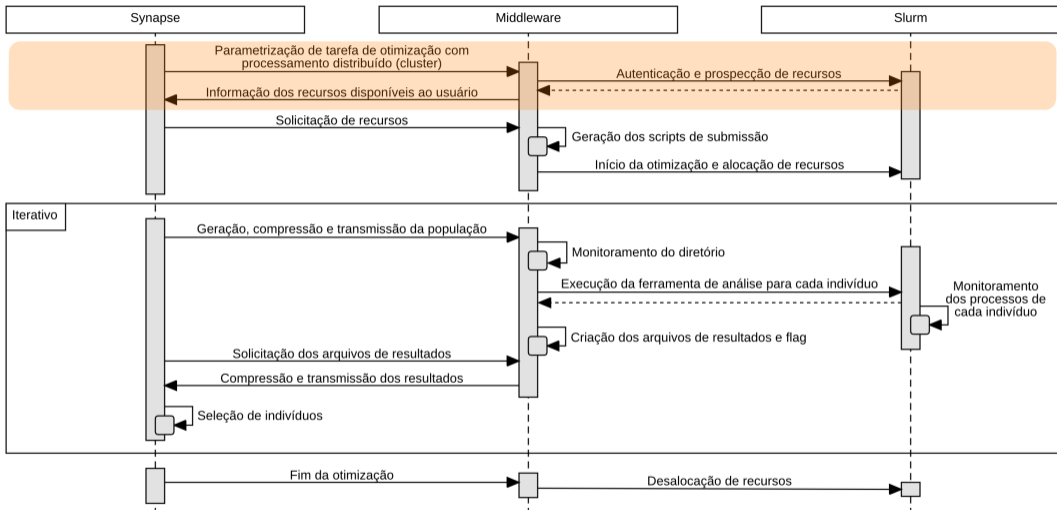


Figura 4: Diagrama de sequência do *middleware* proposto

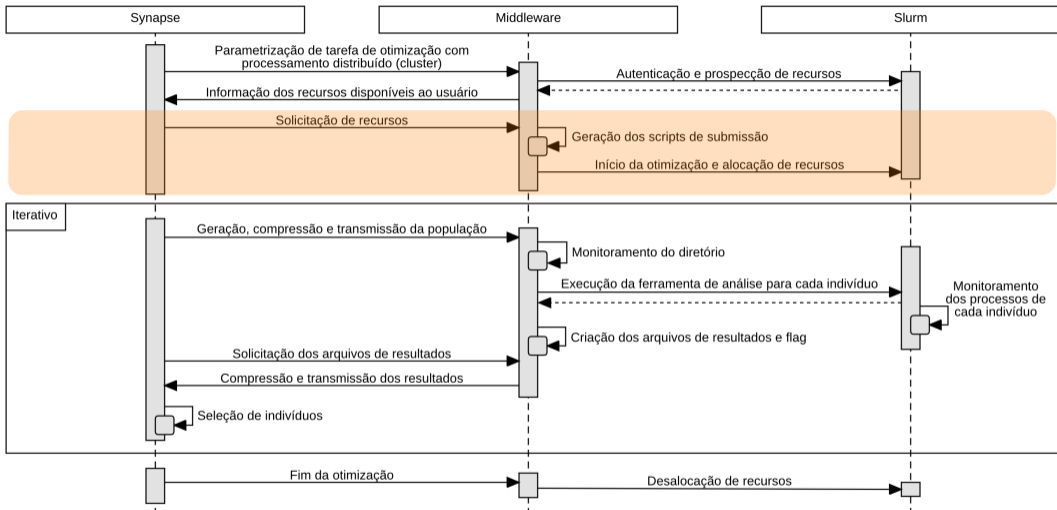


Figura 4: Diagrama de sequência do *middleware* proposto

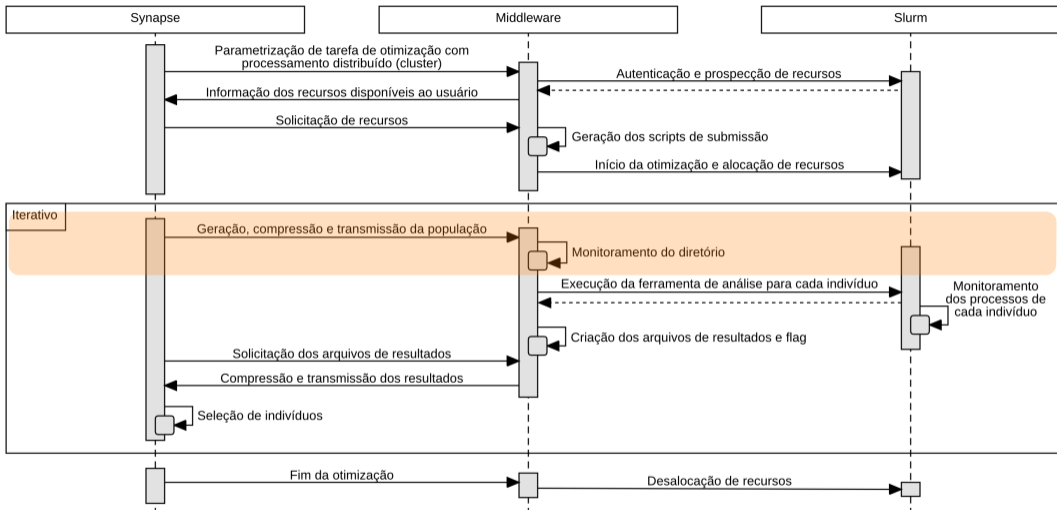


Figura 4: Diagrama de sequência do *middleware* proposto

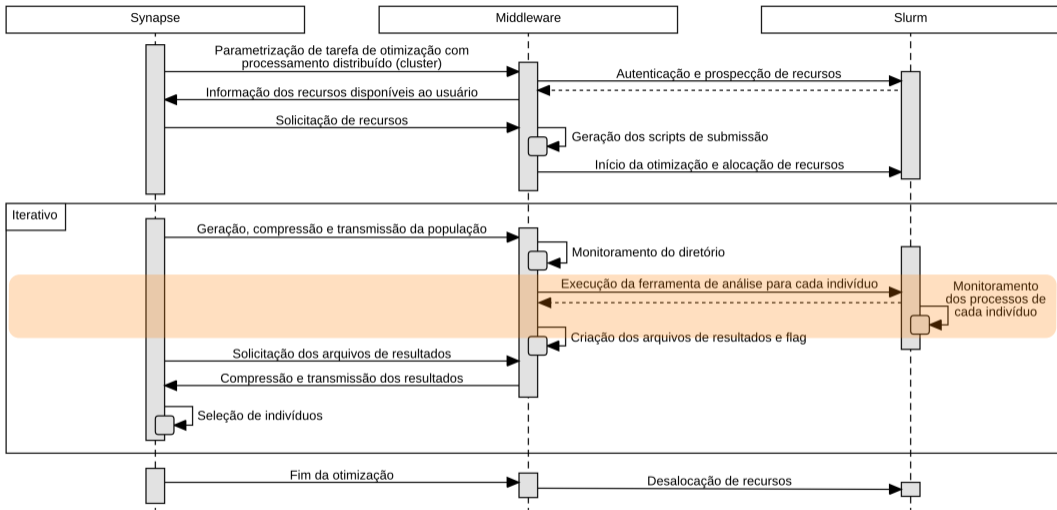


Figura 4: Diagrama de sequência do *middleware* proposto

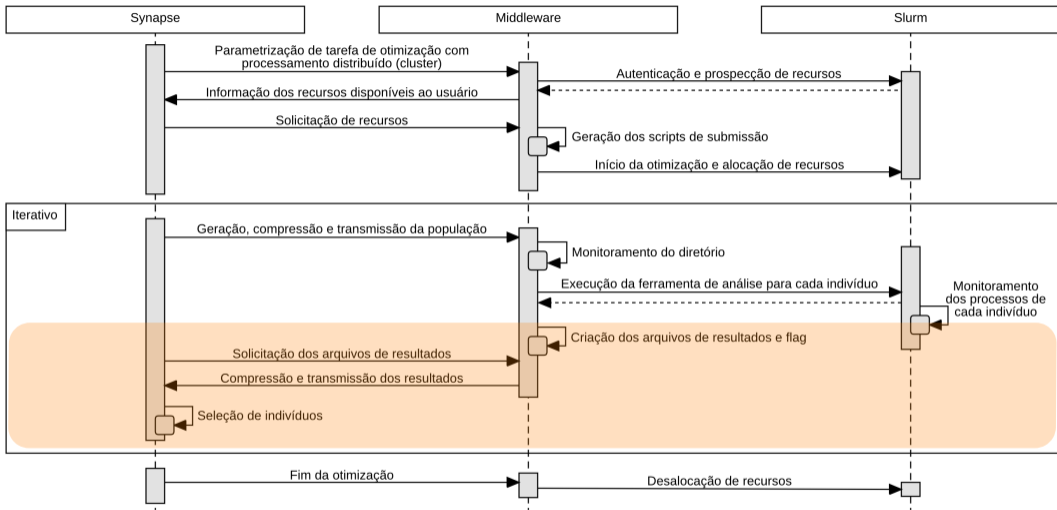


Figura 4: Diagrama de sequência do *middleware* proposto

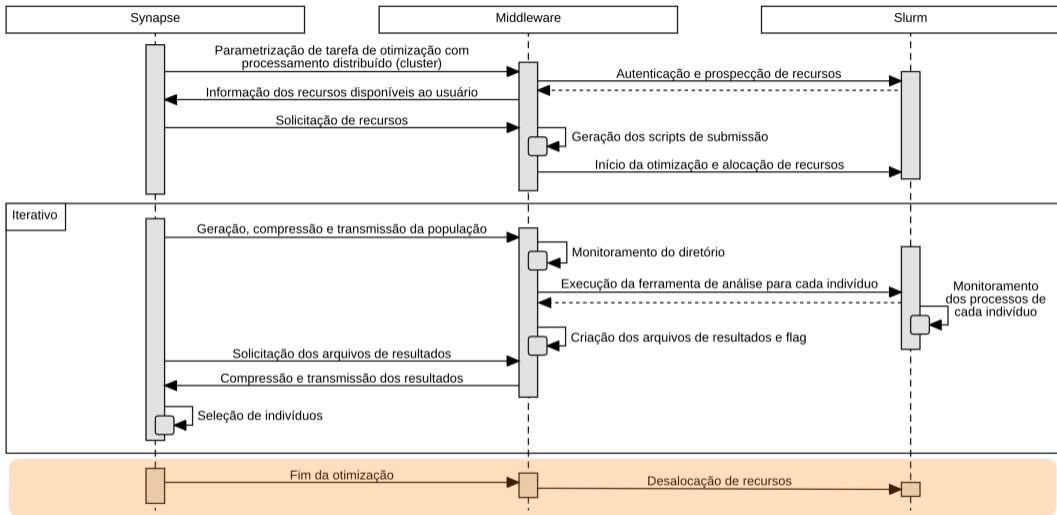


Figura 4: Diagrama de sequência do *middleware* proposto

- Implementação e validação em um cluster heterogêneo*;
- O *middleware* foi integrado ao Synapse Offshore;
- Testes preliminares apontaram um *speedup* de até dez vezes**.

*Três nós com 2 processadores Xeon E5-2630v3 e um nó com 2 processadores Xeon E5-2670v3.

**Comparação entre uma *workstation* com processador i9 12900k e o *cluster*.

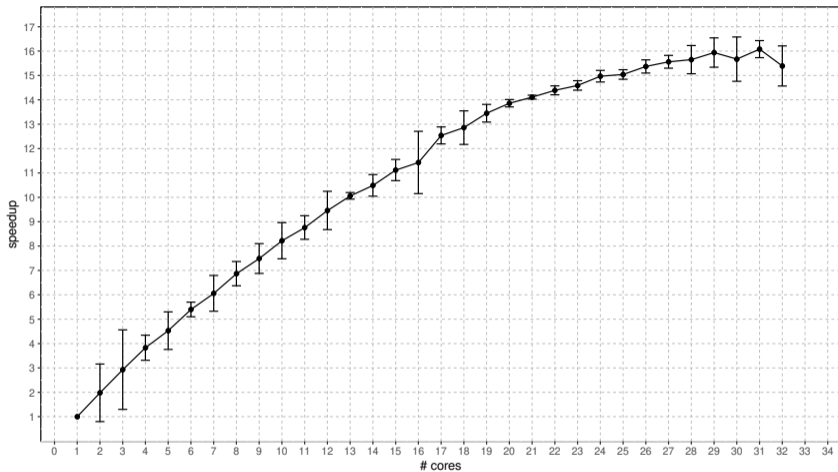


Figura 5: Speedup na otimização de um sistema de ancoragem de uma plataforma *offshore*

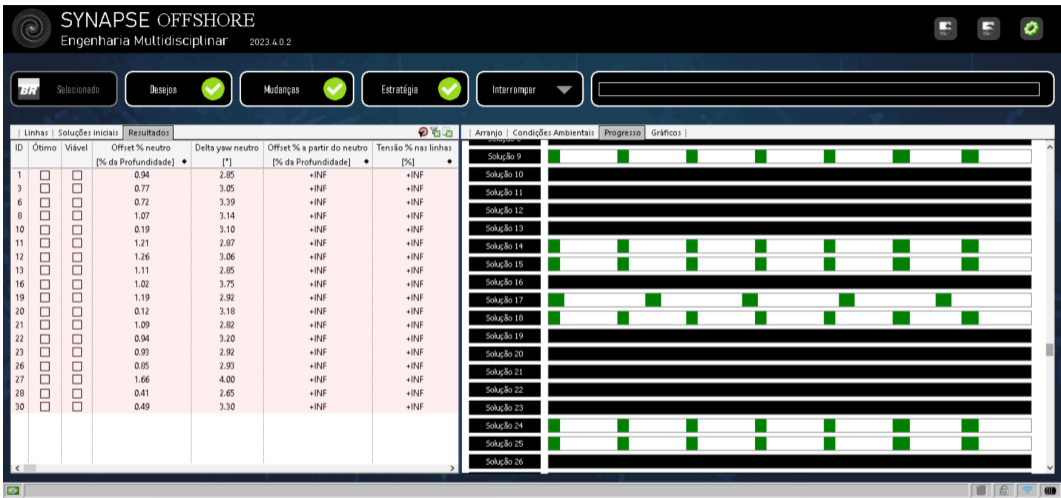


Figura 6: Tela de monitoramento da otimização no Synapse Offshore

- *Benchmarks* e medidas de *speedup* seguindo o rigor estatístico necessário;
- Portar um algoritmo genético como um todo para o cluster;
- Implementação e comparação de diferentes modelos de paralelismo.

Agradecimentos



Uni**SENAI**

