

# SuperComputação

Aula 07 – Busca local

2021 – Engenharia

André Filipe de Moraes Batista <[andrefmb@insper.edu.br](mailto:andrefmb@insper.edu.br)>

# Exploration x Exploitation



**Exploration**

## **Exploration:**

- decisão não localmente ótima feita "de propósito"
- visa adicionar variabilidade nas soluções geradas



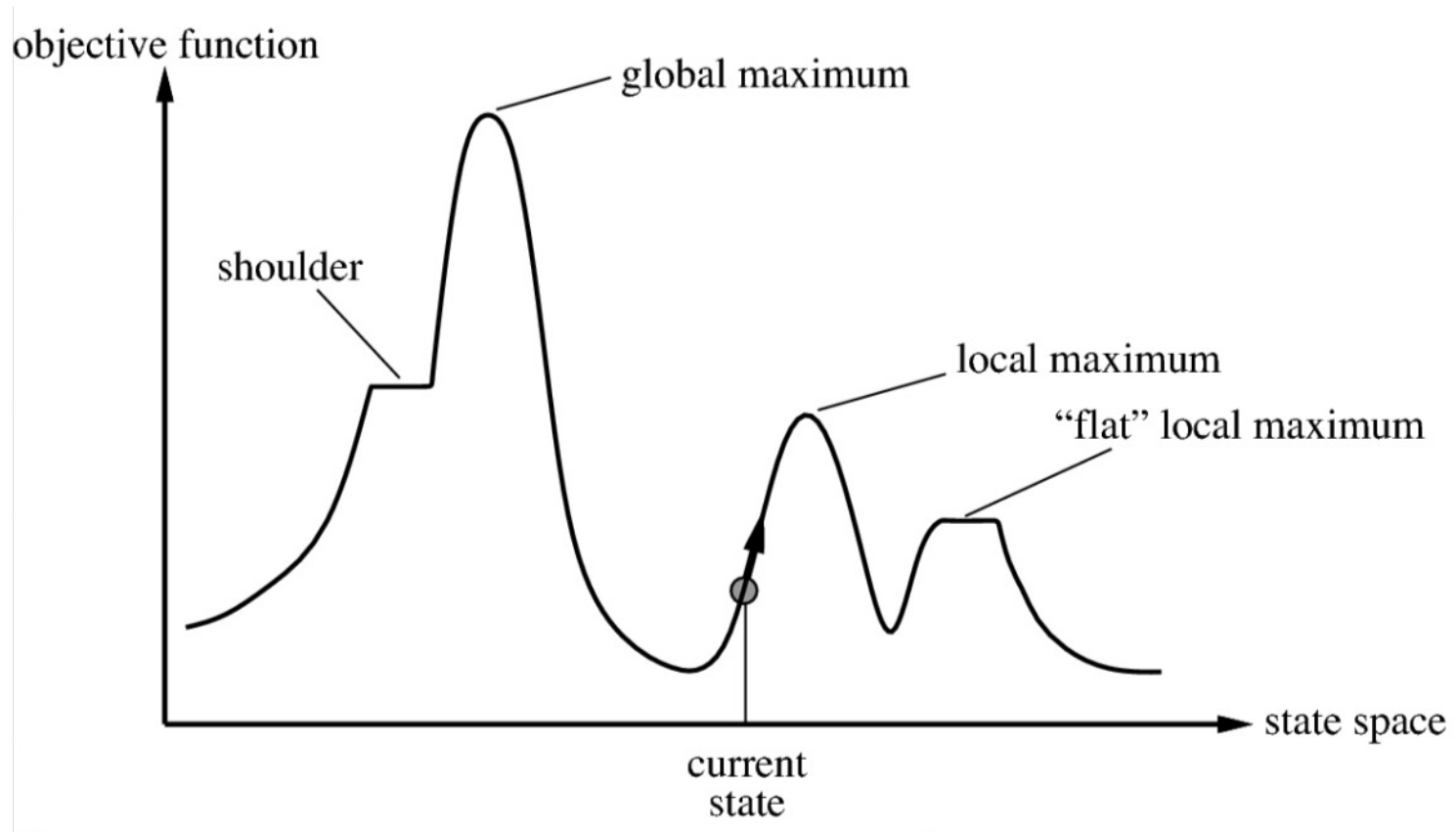
**Exploitation**

## **Exploitation:**

- explorar alguma **propriedade do problema**
- pode ser uma intuição que leve a bons resultados em curto prazo

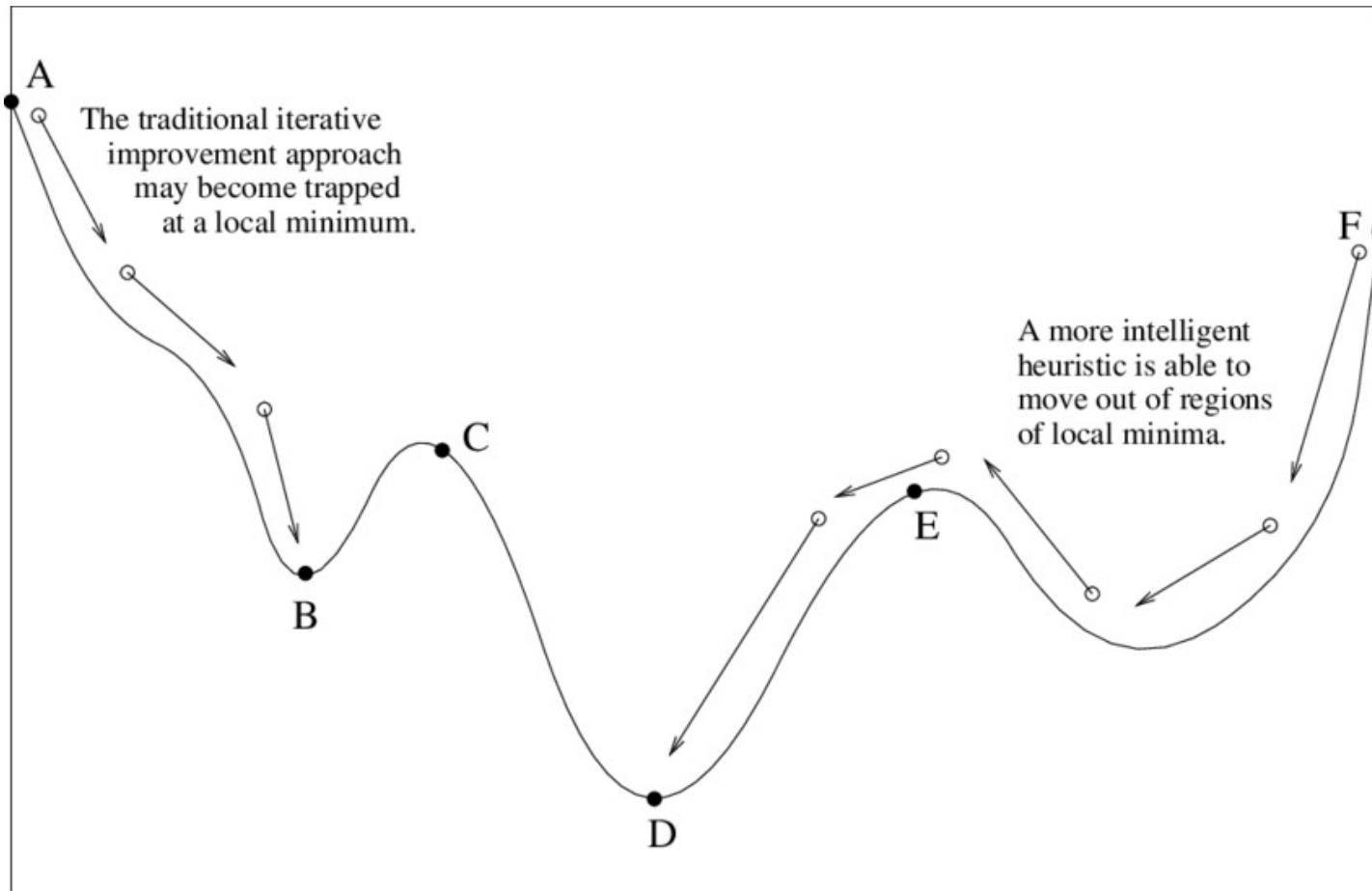
# **Solução aleatorizada**

# Características de uma otimização

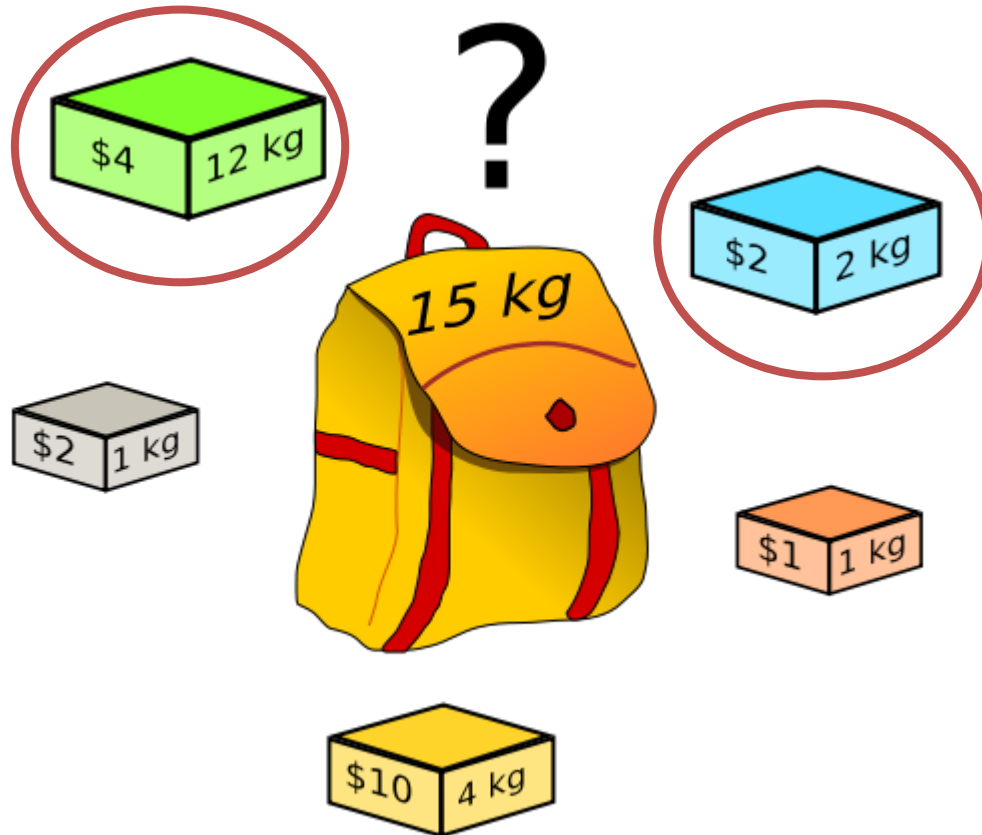


Em muitos problemas de otimização, o caminho para se atingir um objetivo é irrelevante, o objetivo é gerar uma solução para o problema em si.

# Características de uma otimização

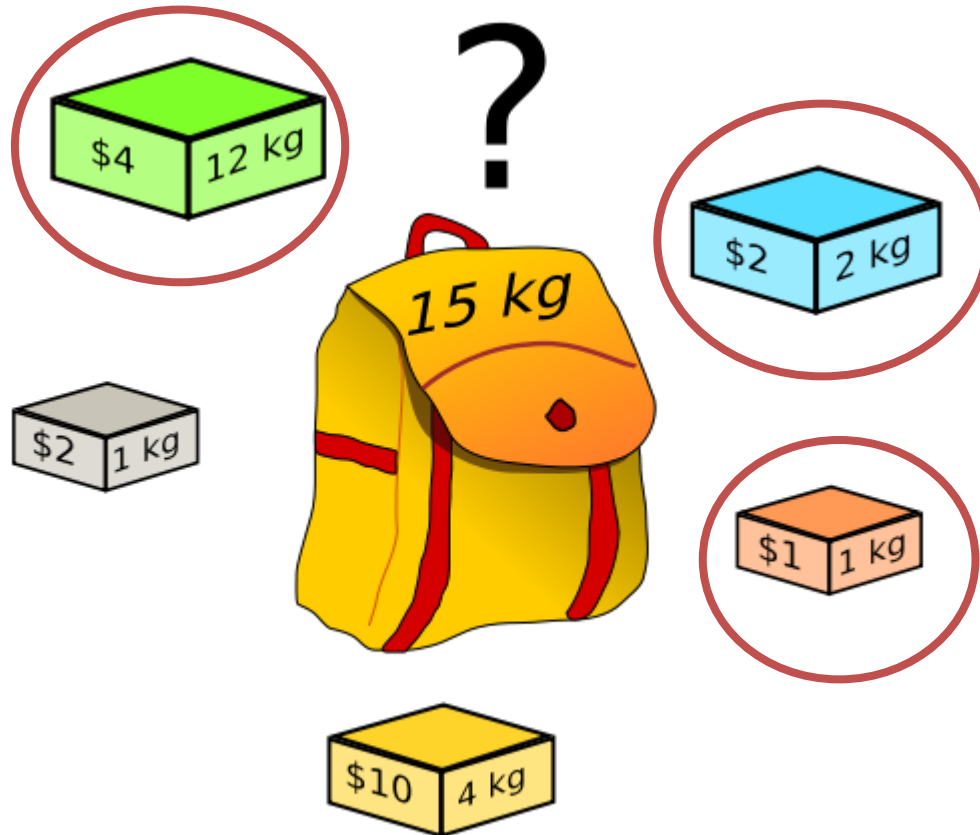


# Uma solução para a mochila



Peso: 14kg  
Valor: \$6

# Uma solução para a mochila (II)



Peso: 15kg  
Valor: \$7

# Melhorando uma solução

1. **Encher a mochila:** verificar se algum objeto não selecionado cabe na mochila
2. **Trocar dois objetos:** verificar se é possível substituir um objeto selecionado por outro de melhor valor que foi deixado de fora



# Solução ótima global

1. **Encher a mochila:** não é possível
2. **Trocar dois objetos:** não é possível

# Solução ótima global

1. **Encher a mochila:** não é possível
2. **Trocar dois objetos:** não é possível

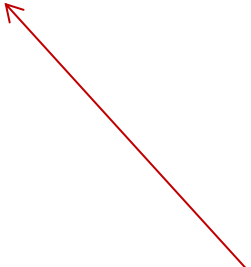
**Ambas são condições  
necessárias, mas não  
suficientes, para  
otimalidade**

# Busca local

1. Cria uma solução
2. Aplicar, sucessivamente, uma operação que melhora esta solução.
3. Parar quando não for mais possível

# Busca local

1. Cria uma solução
2. Aplicar, sucessivamente, uma operação que melhora esta solução.
3. Parar quando não for mais possível



**Aplicar alguma condição necessária mas não suficiente**

# Busca local

1. Cria uma solução
2. Aplicar, sucessivamente, uma operação que melhora esta solução.
3. Parar quando não for mais possível

**Não tem nada  
melhor por perto!**

**Aplicar alguma condição  
necessária mas não  
suficiente**

# Busca local

1. Cria uma solução
2. Aplicar, sucessivamente, uma operação que melhora esta solução.
3. Parar quando não for mais possível

**Depende de onde começou!**

**Não tem nada melhor por perto!**

**Aplicar alguma condição necessária mas não suficiente**

# Busca local

1. Repetir N vezes:

1. Cria uma solução

2. Aplicar, sucessivamente, uma operação que melhora esta solução.

3. Parar quando não for mais possível

2. Retorne a melhor solução

# Busca local (vantagens)

1. Rápida
2. Resultados bons para  $N$  grande
3. Oferece "garantia" (fraca) de qualidade
4. Não ficou bom? Roda mais vezes!



# Busca local (desvantagens)

1. Depende de gerar soluções iniciais
2. Aleatorizado (pode não ser problema)
3. Oferece "garantia" (**fraca**) de qualidade



# Atividade prática

## Busca local e aleatoriedade

1. Criar algoritmo que seleciona a melhor de N soluções
2. Implementar "Mochila cheia"
3. Implementar "Substitui objeto"

# Fechamento

As soluções ficaram melhores que as heurísticas?

E o tempo de execução?

7

# Insper

[www.insper.edu.br](http://www.insper.edu.br)