Introdução à pesquisa operacional

Fundamentos em Pesquisa Operacional

Marcelo Antonio Marotta

Departamento de Ciência da Computação
Universidade de Brasília

O que é pesquisa operacional?

- Conjunto de técnicas direcionadas a problemas complexos
- Voltado a tomada de decisões
- O ponto chave
  - Construção de modelos matemáticos
  - Seleção de técnica adequada para resolução
- Exemplos de problemas
  - Custo mínimo para produção, maximização de lucros, maximização de utilização de equipamentos, redução de desperdícios de produtos, problemas de corte, empacotamento, transporte, rotas, entre outros.

História

Pesquisa Operacional

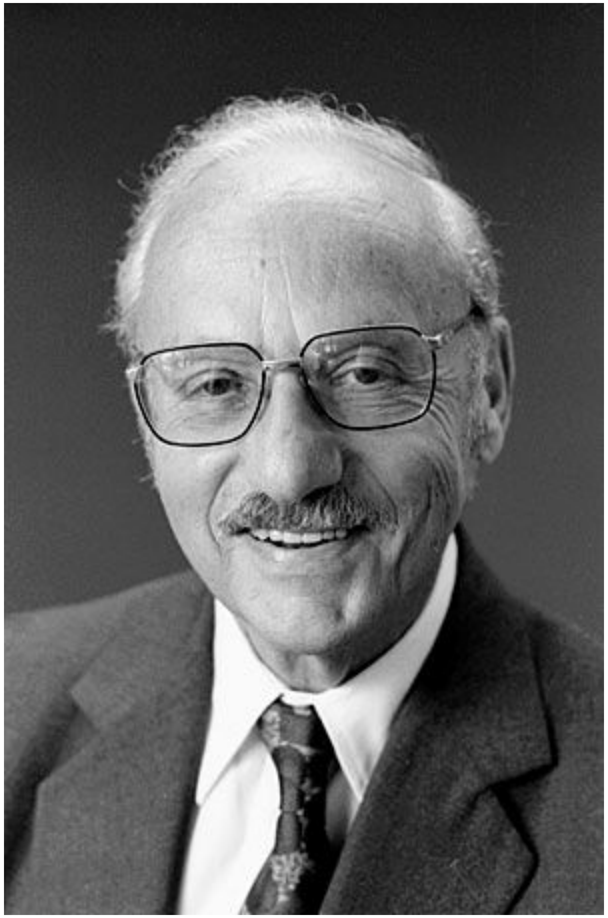
- Operational Research na Inglaterra, Operations Research nos EUA, Investigação Operacional em Portugal e Investigación Operativa em países hispânicos
- 1938
  - Designar o estudo sistemático de problemas estratégicos e táticos decorrentes de operações militares

Um grupo de especialistas (entre eles: Patrick Blackett, Cecil Gordon, C. H. Waddington, Owen Wansbrough-Jones and Frank Yates) foi designado para avaliar e posicionar adequadamente os radares do sistema de defesa aérea da Grã-Bretanha antes e durante a Segunda Guerra Mundial. Outras aplicações militares incluíram o planejamento de operações de comboios, bombardeios e de guerra anti-submarina

História

1947 - Método mais importante do período pós-guerra

- Método Simplex
  - George Dantzig
  - Problemas de Programação Linear
    - Problemas de planejamento nos quais são utilizados modelos de otimização lineares



História

1960 - Brasil, a PO se iniciou

1968 - Primeiro Simpósio Brasileiro de Pesquisa Operacional (SBPO) - ITA

Em seguida, foi criada a Sociedade Brasileira de Pesquisa Operacional (SOBRAPO)

Conceitos básicos

$$Otimizar\ Z = f(x_1, x_2, \dots x_n)$$

$$sujeito\ a: \left\{ \begin{pmatrix} g_1(x_1, x_2, \dots x_n) \\ g_2(x_1, x_2, \dots x_n) \\ \cdot \\ \cdot \\ \cdot \\ g_m(x_1, x_2, \dots x_n) \end{pmatrix} \right\} \le, =, \ge \left\{ \begin{pmatrix} b_1 \\ b_2 \\ \cdot \\ \cdot \\ \cdot \\ b_m \end{pmatrix} \right\}$$

Conceitos Básicos

PO é determinada por vários fatores

- Complexidade, grau, tipo de variáveis
  - Problemas com grau 1 são chamadas lineares
    - Uma técnica adotada é a programação linear que é aplicada a modelos cujas funções objetivo e restrições são lineares
- Outras técnicas são: programação inteira, programação dinâmica, otimização em redes, programação não-linear, programação multiobjetivo, teoria de jogos, entre outras

A maioria das técnicas de PO são baseadas em algoritmos

Soluções podem precisar da adoção de heurísticas a fim de obter soluções em tempo viável

Conceitos básicos

Os modelos de PO são elaborados para “otimizar” um critério objetivo específico sujeito a um conjunto de restrições

- A qualidade da solução resultante depende de quanto o modelo representa o sistema real
- Solução viável
  - Satisfaz todas as restrições do modelo
- Solução ótima
  - Solução viável
  - Resultar no melhor valor (máximo ou mínimo) para o modelo especificado

Fase: 1 - Definição do Problema

Define o escopo do problema sob investigação

Três elementos primordiais

- Determinação do objetivo do estudo
  - Maximizar
  - Minimizar
- Descrição das alternativas de decisão
  - Quantidade, lucro, custo, associação
- Especificação das limitações do sistema
  - Limites máximos, limites mínimos

Fase 2: Construção do modelo

Definição das componentes/primitivas do modelo, bem como suas relações, restrições e limites

- Parametrização
  - Adoção de uma notação apropriada para as principais quantidades presentes na definição do problema
    - Valores pré-alocados e normalmente constantes
      - Constantes numéricas, vetores, matrizes (c,N, A_n, T_{nm})
- Definição das variáveis de decisão
  - Hipóteses - Valores manipulados no problema
    - É comum denotar por x1, x2, ... , xn (por hipótese)
- Definição das relações entre variáveis de decisão, parâmetros e o objetivo do problema
  - Formulação da função-objetivo
  - Limites inferiores e superiores
  - Restrições do problema

Fase 1 : Exemplo

O lucro por cada Kg do produto A vendido por uma empresa é de R$3,00. Já, o produto B, gera um lucro de R$ 5,00 por Kg vendido. Todavia, o estoque só comporta 4 Kg do produto A. Para o produto B, o estoque apresenta 12 potes que comportam ½ Kg deste produto por pote. Os produtos são extremamente especializados e difíceis de serem obtidos. Dessa forma, o fornecedor limita suas vendas a no máximo R$ 18 dos produtos A e B por cliente, sendo R$ 3 cada Kg do produto A e R$ 2 para o produto B. Por exemplo, se ele vender 6 Kg do produto A, ele venderá 0 Kg do produto B. Finalmente, quais as proporções ideais de cada produto devem ser obtidas para atingir o máximo lucro da empresa?

Fase 2: Construção do modelo : Parametrização

Definição da notação

- 3 e 5 = Lucro com produtos ‘A’ e ‘B’
- 4 e 12 = Estoque do produto ‘A’ em Kg e ‘B’ em potes
- 18 = Venda total em reais dos dois produtos
- 3 e 2 = Total cobrado em reais pelo produto ‘A’ e ‘B’

Fase 2: Construção do modelo : Variáveis de decisão

Definição das variáveis de decisão

- x1 = Quantidade a ser adquirida do produto A em Kg
- x2 = Quantidade a ser adquirida do produto B em Kg

Fase 2 : Função objetivo e restrições

Função objetivo:

… “quais as proporções ideais de cada produto devem ser obtidas para atingir o **máximo lucro** da empresa?”

Fase 2 : Função objetivo e restrições

Função objetivo:

… “quais as proporções ideais de cada produto devem ser obtidas para atingir o **máximo lucro** da empresa?”

Maximizar = max

$$\mathbf{max}_{R\$}\ \mathbf{3x1 + 5x2}$$

Fase 2 : Função objetivo e restrições

Restrições:

“Todavia, o estoque só comporta 4 Kg do produto A.”

- Ao se associar com a variável de decisão x1

$$x1 \leq 4$$

Fase 2 : Função objetivo e restrições

Restrições:

“Para o produto B, o estoque apresenta 12 potes que comportam ½ Kg deste produto por pote.”

- Ao se associar com a variável de decisão x2

$$x2 \leq 12$$

Fase 2 : Função objetivo e restrições

Restrições:

“Para o produto B, o estoque apresenta 12 potes que comportam ½ Kg deste produto por pote.”

- Ao se associar com a variável de decisão x2

$$x2 \leq 12$$

ERRADO!!!!!

Fase 2 : Função objetivo e restrições

Restrições:

“Para o produto B, o estoque apresenta 12 potes que comportam ½ Kg deste produto por pote.”

- Ao se associar com a variável de decisão x2

$$\mathbf{2}.x2 \leq 12$$

O valor de x2 é dado em Kg!!!!!!

Correto!

CIC

Fase 2 : Função objetivo e restrições

Restrições:

“Dessa forma, o fornecedor limita suas vendas a no máximo R$ 18 dos produtos A e B por cliente, sendo R$ 3 cada Kg do produto A e R$ 2 para o produto B. Por exemplo, se ele vender 6 Kg do produto A, ele venderá 0 Kg do produto B.”

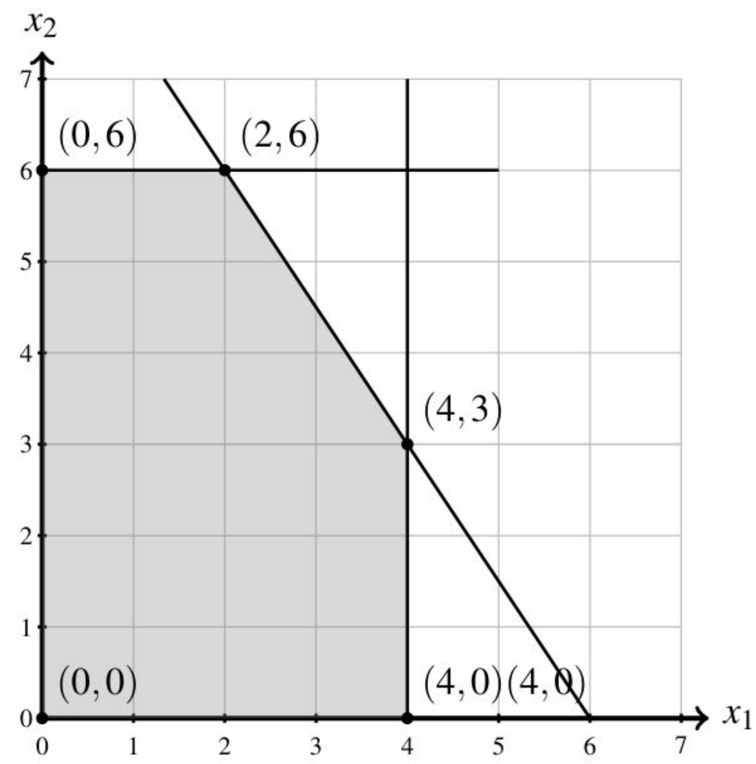
- Ao se associar com as variáveis de decisão x1 e x2

$$3.x1 + 2.x2 \leq 18$$

CIC

Espaço de solução

$$
\begin{array}{llcl}
\text{Max} & 3x_1 + 5x_2 & & \\
\text{sujeito a} & x_1 & \leq & 4 \\
& 2x_2 & \leq & 12 \\
& 3x_1 + 2x_2 & \leq & 18 \\
& x_1, x_2 & \geq & 0
\end{array}
$$



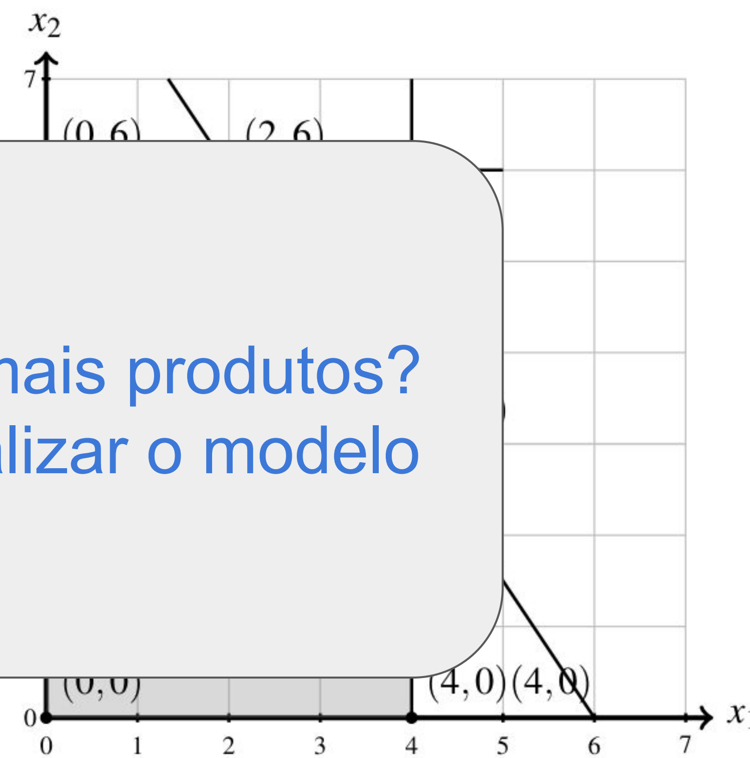
Espaço de solução

Max $3x$

sujeito a

$3x$

$x_1, x_2 \geq 0$



E se eu tivesse mais produtos?
Podemos generalizar o modelo

Generalização do modelo

Definição da notação

- C = conjunto de produtos C={‘A’,’B’}
- L_i = Lucro com produto i | i ∈ C
    - L_i = {3,5}
- E_i = Estoque total por produto i | i ∈ C
    - E_i={4,6} -> Normalização do valor 12 para a unidade da variável
- T_i = Total cobrado por produto i | i ∈ C
    - T_i = {3, 2}
- T = Venda total dos produtos
        - T = 18

Generalização do modelo

Variáveis de decisão

- x_i= Variáveis de decisão i | i $\in$ C = {'A', 'B'}
  - $x_i \in \mathbb{R}$ | i $\in$ C = {'A', 'B'}

Função objetivo

$$Lucro = L_A.x1 + L_B.x2$$

se torna

$$Lucro = \sum_C L_i x_i$$

$$\max \sum_i^C L_i x_i$$

Generalização do modelo

Restrições

- Estoque

$$x1 \leq 4$$

$$2.x2 \leq 12$$

Normalizando

$E_i = \{4, \mathbf{6}\}$

$$x_A \leq E_A$$

$$x_B \leq E_B$$

$$\text{Logo :} \quad x_i \leq E_i$$

Generalização do modelo

Restrições

- Estoque

$$x1 \leq 4$$

$$2.x2 \leq 12$$

Normalizando

$E_i = \{4, \mathbf{6}\}$

$$x_A \leq E_A$$

$$x_B \leq E_B$$

$$x_i \leq E_i, \ \forall i \in C$$

Generalização do modelo

Restrições

- Total fornecido

$$3.x1 + 2.x2 \leq 18$$

$$T_A.x_A + T_B.x_B \leq T$$

$$\sum_i^C T_i x_i \leq T$$

CIC

Modelo generalizado

$$\max \sum_{i}^{C} L_i x_i$$

$$s.t.$$

$$x_i \leq E_i, \qquad \forall i \in C$$

$$\sum_{i}^{C} T_i x_i \leq T$$

$$x_i \geq 0 \qquad \forall i \in C$$

Modelo generalizado

Mas, como solucionar esse tipo de problema?
Existe uma forma de se fazer isso computacionalmente?

$$x_i \geq 0 \qquad \forall i \in C$$

Solvers

Solvers podem implementar diferentes técnicas para solucionar problemas de otimização

- Simplex, Branch & Bound e Branch & Cut

Muito dos solvers são proprietários

- CPLEX, Optimization Toolbox Matlab

Existem solvers gratuitos

- GLPK e OrTools da Google

Utilizaremos o ORTools durante a disciplina

ORTools

https://developers.google.com/optimization/install

- OR-Tools C++
- OR-Tools Python
- OR-Tools Java
- OR-Tools C# (.Net)

Home > Products > OR-Tools > Installation

Install OR-Tools

Google created OR-Tools in C++, but you can also use it with Python, Java, or C# (on the .NET platform).

Install OR-Tools for Python

The fastest way to get OR-Tools is to install the Python binary version. If you already have Python (version 3.6+ on Linux, or 3.6+ on Mac OS or Windows), and the Python package manager PIP, you can install OR-Tools as follows:

```
$ python -m pip install --upgrade --user ortools
```

CIC

Implementando uma solução com ORTools

Dado o problema:

max $3x_1 + x_2$

s.t.

$0 <= x_1 <= 1$

$0 <= x_2 <= 2$

$x_1 + x_2 <= 2$

Implementando uma solução com ORTools

Dado o problema:

max $3x_1 + x_2$

s.t.

$0 <= x_1 <= 1$

$0 <= x_2 <= 2$

$x_1 + x_2 <= 2$

Como mapeá-lo para o ORTools?

Programando com ORTools

Estruture as declarações de acordo com a seguinte lógica:

1. Importação de bibliotecas e definição do ORTools solver
2. Instanciamento de Parâmetros
3. Definição das variáveis de decisão
4. Criação das restrições do problema
5. Definição da função objetivo
6. Executar o solver

Importação de bibliotecas e definição do ORTools solver

Em um novo documento .py

Comece importando a biblioteca do ORTools

```
#Vamos importar o solver linear do ORTOOLS
from __future__ import print_function
from ortools.linear_solver import pywraplp
```

Declare o solver e seu tipo

```
#Vamos declarar o solver considerando o Google's Linear Optimization Programming system (GLOP)
solver = pywraplp.Solver('simple_lp_program', pywraplp.Solver.GLOP_LINEAR_PROGRAMMING)
```

Instanciamento de Parâmetros

Os parâmetros do problema dado são relacionados com os valores que acompanham x1 e x2 na função objetivo e nas restrições

Ou seja

F. Obj:

- f1*x1 + f2*x2;
  - f1 = 3; e f2 = 1

Rest de limites

- lbi<=xi<=ubi
  - lb1=0 e lb1 = 1; e lb2=0 e ub2=2

Rest de relacionamento

- a1*x1+a2*x2 <= b1
  - a1 = 1; a2=1; e b1=2

Instanciamento de Parâmetros

Os parâmetros do problema dado são relacionados com os valores que acompanham x1 e x2 na função objetivo e nas restrições

Ou seja

F. Obj:

- f1*x1 + f2*x2;
  - f1 = 3; e f2 = 1

Rest de limites

- lbi<=xi<=ubi
  - lb1=0 e lb1 = 1; e lb2=0 e ub2=2

Rest de relacionamento

- a1*x1+a2*x2 <= b1
  - a1 = 1; a2=1; e b1=2

```
#Definição dos parametros
lb = [0,0]
ub = [1,2]
f = [3,1]
a = [1,1]
b = [2]
```

Definição das variáveis de decisão

Define-se as variáveis de decisão, bem como seus limites

```
#Vamos criar as variaveis numericas do problema de otimizacao
#Formato: solver.NumVar(limite inferior, limite superior, rotulo)
x1 = solver.NumVar(0, 1, 'x1')
x2 = solver.NumVar(0, 2, 'x2')
```

Utilizando nossa notação para os parâmetros anteriores, fica

```
x1 = solver.NumVar(lb[0],ub[0] , 'x1')
x2 = solver.NumVar(lb[1], ub[1], 'x2')
```

Criação das restrições do problema

Primeiro cria-se a restrição de acordo com os limites

a1*x1+a2*x2 <= b -> x1+x2 <= 2

```
#Vamos criar as restricoes do problema para as variaveis criadas
#Restricao linear x1+x2<=2; para isso, precisamos definir o minimo e o maximo valor
# e reescrever a restricao como: x1+x2 <= 2
ct = solver.Constraint(-solver.infinity(),2,'ct')
```

Agora, precisamos instanciar os valores de a1 e a2

```
#Agora, precisaremos colocar os coeficientes multiplicativos de x1 e x2
ct.SetCoefficient(x1, 1)
ct.SetCoefficient(x2, 1)
```

Criação das restrições do problema

Aplicando-se a notação de parametrização fica:

```
#Vamos criar as restricoes do problema para as variaveis criadas
#Restricao linear x1+x2<=2; para isso, precisamos definir o minimo e o maximo valo
# e reescrever a restricao como: x1+x2 <= 2
ct = solver.Constraint(-solver.infinity(),b[0],'ct')
#Agora, precisaremos colocar os coeficientes multiplicativos de x1 e x2
ct.SetCoefficient(x1, a[0])
ct.SetCoefficient(x2, a[1])
```

Definição da função objetivo

Primeiro cria-se a função objetivo de forma genérica

```
#Definiremos a funcao objetivo 3x1 + x2
# Create the objective function, 3 * x1 + x2.
objective = solver.Objective()
```

A função é mapeada como: f1*x1+f2*x2 ou 3x1+x2

Vamos instanciar f1 e f2

```
#Agora, precisaremos colocar os coeficientes multiplicativos de x1 e x2
objective.SetCoefficient(x1, 3)
objective.SetCoefficient(x2, 1)
```

Para terminar, determinamos o objetivo como uma maximização

```
#Definiremos o problema como uma maximizacao
objective.SetMaximization()
```

Definição da função objetivo

Utilizando nossa notação

```
#Definiremos a funcao objetivo 3x1 + x2
# Create the objective function, 3 * x1 + x2.
objective = solver.Objective()
#Agora, precisaremos colocar os coeficientes multiplicativos de x1 e x2
objective.SetCoefficient(x1, f[0])
objective.SetCoefficient(x2, f[1])
#Definiremos o problema como uma maximizacao
objective.SetMaximization()
```

Executando e imprimindo a resposta

```python
#Executaremos o solver
solver.Solve()

#Imprimiremos a solucao
print('Solucao:')
print('Valor objetivo =', objective.Value())
print('x1 =', x1.solution_value())
print('x2 =', x2.solution_value())
```

Exercício próxima aula

Implementar no ORTools os modelos vistos em aula

$$
\begin{array}{lrcrl}
\text{Max} & 3x_1 & + & 5x_2 & \\
\text{sujeito a} & & & x_1 & \leq 4 \\
 & & & 2x_2 & \leq 12 \\
 & 3x_1 & + & 2x_2 & \leq 18 \\
 & & & x_1, x_2 & \geq 0
\end{array}
$$

Problema A

$$
\begin{aligned}
& \max \textstyle\sum_i^C L_i x_i \\
s.t. & \\
& x_i \leq E_i, \qquad \forall i \in C \\
& \textstyle\sum_i^C T_i x_i \leq T \\
& x_i \geq 0 \qquad \forall i \in C
\end{aligned}
$$

Problema B