

# MINICURSO DE SIMULADOR EMSO

## Dia 2: Otimização de Processos

Rodolfo Rodrigues, Eng MSc

Doutorando em Engenharia Química  
PPGEQ/UFRGS  
rodolfo@enq.ufrgs.br

XVI COREEQ  
Curitiba, Paraná  
17–18 de julho de 2011



## Introdução

Otimização é uma classe de problema matemático que envolve a busca por um máximo ou mínimo de uma função, sujeita ou não a restrições de igualdade ou desigualdade.

$$\left\{ \begin{array}{l} \min_{x_n} f(x_n) \\ 0 \leq x_n \leq x_n^{max} \\ \text{sujeito a:} \\ h(x_n) = 0 \\ g(x_n) \leq 0 \end{array} \right.$$

onde:

$f(x_n)$ : Função Objetivo

$h(x_n)$ : Restrições de Igualdade

$g(x_n)$ : Restrições de Desigualdade



## Introdução

Otimização é uma classe de problema matemático que envolve a busca por um máximo ou mínimo de uma função, sujeita ou não a restrições de igualdade ou desigualdade.

$$\left\{ \begin{array}{l} \min_{x_n} f(x_n) \\ 0 \leq x_n \leq x_n^{max} \\ \text{sujeito a:} \\ h(x_n) = 0 \\ g(x_n) \leq 0 \end{array} \right.$$

onde:

$f(x_n)$ : Função Objetivo

$h(x_n)$ : Restrições de Igualdade

$g(x_n)$ : Restrições de Desigualdade



## Introdução

- A busca pelo máximo ou mínimo envolve a variação de algumas variáveis (variáveis de decisão).
- Em aplicações industriais, esse máximo ou mínimo pode estar associado a objetivos econômicos ou operacionais.
- As restrições devem sempre ser respeitadas.
- Em casos industriais as restrições estão associadas a leis físicas e químicas ou operacionais.



## Introdução

- A busca pelo máximo ou mínimo envolve a variação de algumas variáveis (variáveis de decisão).
- Em aplicações industriais, esse máximo ou mínimo pode estar associado a objetivos econômicos ou operacionais.
- As restrições devem sempre ser respeitadas.
- Em casos industriais as restrições estão associadas a leis físicas e químicas ou operacionais.



## Introdução

- A busca pelo máximo ou mínimo envolve a variação de algumas variáveis (variáveis de decisão).
- Em aplicações industriais, esse máximo ou mínimo pode estar associado a objetivos econômicos ou operacionais.
- As restrições devem sempre ser respeitadas.
- Em casos industriais as restrições estão associadas a leis físicas e químicas ou operacionais.



## Introdução

- A busca pelo máximo ou mínimo envolve a variação de algumas variáveis (variáveis de decisão).
- Em aplicações industriais, esse máximo ou mínimo pode estar associado a objetivos econômicos ou operacionais.
- As restrições devem sempre ser respeitadas.
- Em casos industriais as restrições estão associadas a leis físicas e químicas ou operacionais.



# Introdução

## Exemplos de otimização aplicada a processos industriais:

- Variar a razão de refluxo para atingir a pureza de uma torre.
- Aumentar a produção de leves em um separador *flash*.
- Reduzir a perda do produto principal em uma purga.
- Escolher o melhor compromisso entre conversão e seletividade de um processo.





## Introdução

Exemplos de otimização aplicada a processos industriais:

- Variar a razão de refluxo para atingir a pureza de uma torre.
- Aumentar a produção de leves em um separador *flash*.
- Reduzir a perda do produto principal em uma purga.
- Escolher o melhor compromisso entre conversão e seletividade de um processo.



## Introdução

Exemplos de otimização aplicada a processos industriais:

- Variar a razão de refluxo para atingir a pureza de uma torre.
- Aumentar a produção de leves em um separador *flash*.
- Reduzir a perda do produto principal em uma purga.
- Escolher o melhor compromisso entre conversão e seletividade de um processo.



## Introdução

Exemplos de otimização aplicada a processos industriais:

- Variar a razão de refluxo para atingir a pureza de uma torre.
- Aumentar a produção de leves em um separador *flash*.
- Reduzir a perda do produto principal em uma purga.
- Escolher o melhor compromisso entre conversão e seletividade de um processo.



## Introdução

Exemplos de otimização aplicada a processos industriais:

- Variar a razão de refluxo para atingir a pureza de uma torre.
- Aumentar a produção de leves em um separador *flash*.
- Reduzir a perda do produto principal em uma purga.
- Escolher o melhor compromisso entre conversão e seletividade de um processo.



## Introdução

Além disso nas etapas de projeto a necessidade de otimização é frequente, exemplo:

- Projeto de redes de trocador de calor visando o reaproveitamento energético.
- Projeto de redes de reaproveitamento de água.
- Dimensionamento ótimo de unidades que sofrem regeneração frequente.
- Projeto ótimo de unidades de reação e separação.
- Escolha de catalizadores.



## Introdução

Além disso nas etapas de projeto a necessidade de otimização é frequente, exemplo:

- Projeto de redes de trocador de calor visando o reaproveitamento energético.
- Projeto de redes de reaproveitamento de água.
- Dimensionamento ótimo de unidades que sofrem regeneração frequente.
- Projeto ótimo de unidades de reação e separação.
- Escolha de catalizadores.



## Introdução

Além disso nas etapas de projeto a necessidade de otimização é frequente, exemplo:

- Projeto de redes de trocador de calor visando o reaproveitamento energético.
- Projeto de redes de reaproveitamento de água.
- Dimensionamento ótimo de unidades que sofrem regeneração frequente.
- Projeto ótimo de unidades de reação e separação.
- Escolha de catalizadores.



## Introdução

Além disso nas etapas de projeto a necessidade de otimização é frequente, exemplo:

- Projeto de redes de trocador de calor visando o reaproveitamento energético.
- Projeto de redes de reaproveitamento de água.
- Dimensionamento ótimo de unidades que sofrem regeneração frequente.
- Projeto ótimo de unidades de reação e separação.
- Escolha de catalizadores.





## Introdução

Além disso nas etapas de projeto a necessidade de otimização é frequente, exemplo:

- Projeto de redes de trocador de calor visando o reaproveitamento energético.
- Projeto de redes de reaproveitamento de água.
- Dimensionamento ótimo de unidades que sofrem regeneração frequente.
- Projeto ótimo de unidades de reação e separação.
- Escolha de catalizadores.



## Introdução

Além disso nas etapas de projeto a necessidade de otimização é frequente, exemplo:

- Projeto de redes de trocador de calor visando o reaproveitamento energético.
- Projeto de redes de reaproveitamento de água.
- Dimensionamento ótimo de unidades que sofrem regeneração frequente.
- Projeto ótimo de unidades de reação e separação.
- Escolha de catalizadores.



## Introdução

E também nas etapas de pós-produção diversos problemas podem ser resolvido por otimização, exemplo:

- Problemas de transporte.
- Mistura de hidrocarbonetos para formar um produto final baseado nos custos de produção e vendas.
- Alocação de recursos (funcionários, equipamentos, etc).
- Planejamento ótimo de produção.
- Venda e compra de energia.



## Introdução

E também nas etapas de pós-produção diversos problemas podem ser resolvido por otimização, exemplo:

- Problemas de transporte.
- Mistura de hidrocarbonetos para formar um produto final baseado nos custos de produção e vendas.
- Alocação de recursos (funcionários, equipamentos, etc).
- Planejamento ótimo de produção.
- Venda e compra de energia.



## Introdução

E também nas etapas de pós-produção diversos problemas podem ser resolvido por otimização, exemplo:

- Problemas de transporte.
- Mistura de hidrocarbonetos para formar um produto final baseado nos custos de produção e vendas.
- Alocação de recursos (funcionários, equipamentos, etc).
- Planejamento ótimo de produção.
- Venda e compra de energia.



## Introdução

E também nas etapas de pós-produção diversos problemas podem ser resolvido por otimização, exemplo:

- Problemas de transporte.
- Mistura de hidrocarbonetos para formar um produto final baseado nos custos de produção e vendas.
- Alocação de recursos (funcionários, equipamentos, etc).
- Planejamento ótimo de produção.
- Venda e compra de energia.



## Introdução

E também nas etapas de pós-produção diversos problemas podem ser resolvido por otimização, exemplo:

- Problemas de transporte.
- Mistura de hidrocarbonetos para formar um produto final baseado nos custos de produção e vendas.
- Alocação de recursos (funcionários, equipamentos, etc).
- Planejamento ótimo de produção.
- Venda e compra de energia.



## Introdução

E também nas etapas de pós-produção diversos problemas podem ser resolvido por otimização, exemplo:

- Problemas de transporte.
- Mistura de hidrocarbonetos para formar um produto final baseado nos custos de produção e vendas.
- Alocação de recursos (funcionários, equipamentos, etc).
- Planejamento ótimo de produção.
- Venda e compra de energia.





## Otimização no EMSO

- Atualmente o simulador EMSO é capaz de resolver problemas de otimização estacionária.
- Estes problemas são descritos na forma de problemas de otimização não-linear (NLP).
- Recentemente foi disponibilizado também o suporte para problemas envolvendo variáveis inteiras (MINLP).
- O simulador EMSO pode resolver problemas de otimização através da comunicação com **solvers** externos.



## Otimização no EMSO

- Atualmente o simulador EMSO é capaz de resolver problemas de otimização estacionária.
- Estes problemas são descritos na forma de problemas de otimização não-linear (NLP).
- Recentemente foi disponibilizado também o suporte para problemas envolvendo variáveis inteiras (MINLP).
- O simulador EMSO pode resolver problemas de otimização através da comunicação com **solvers** externos.



## Otimização no EMSO

- Atualmente o simulador EMSO é capaz de resolver problemas de otimização estacionária.
- Estes problemas são descritos na forma de problemas de otimização não-linear (NLP).
- Recentemente foi disponibilizado também o suporte para problemas envolvendo variáveis inteiras (MINLP).
- O simulador EMSO pode resolver problemas de otimização através da comunicação com *solvers* externos.



## Otimização no EMSO

- Atualmente o simulador EMSO é capaz de resolver problemas de otimização estacionária.
- Estes problemas são descritos na forma de problemas de otimização não-linear (NLP).
- Recentemente foi disponibilizado também o suporte para problemas envolvendo variáveis inteiras (MINLP).
- O simulador EMSO pode resolver problemas de otimização através da comunicação com **solvers** externos.



## Otimização Estática

- Otimizações estacionárias permitem determinar condições operacionais ótimas para se atingir objetivos:
  - 1 Econômicos;
  - 2 de Segurança;
  - 3 Ambientais.
- Respeitando um determinado modelo e restrições operacionais.



## Otimização Estática

- Otimizações estacionárias permitem determinar condições operacionais ótimas para se atingir objetivos:
  - 1 Econômicos;
  - 2 de Segurança;
  - 3 Ambientais.
- Respeitando um determinado modelo e restrições operacionais.



## Otimização Estática

- Otimizações estacionárias permitem determinar condições operacionais ótimas para se atingir objetivos:
  - 1 Econômicos;
  - 2 de Segurança;
  - 3 Ambientais.
- Respeitando um determinado modelo e restrições operacionais.



## Otimização Estática

- Otimizações estacionárias permitem determinar condições operacionais ótimas para se atingir objetivos:
  - 1 Econômicos;
  - 2 de Segurança;
  - 3 Ambientais.
- Respeitando um determinado modelo e restrições operacionais.





## Otimização Estática

- Otimizações estacionárias permitem determinar condições operacionais ótimas para se atingir objetivos:
  - 1 Econômicos;
  - 2 de Segurança;
  - 3 Ambientais.
- Respeitando um determinado modelo e restrições operacionais.



## Otimização Estática

- Um problema de otimização é composto por:
  - 1 Uma função objetivo.
  - 2 O modelo da planta.
  - 3 Restrições operacionais.
- O ponto inicial de uma otimização no EMSO é um modelo de planta *FlowSheet*.
- Porém um *FlowSheet* consistente não tem graus de liberdade para otimizar.
- Então é necessário *criar* graus de liberdade para otimizar.
- Estes graus de liberdade são *criados* liberando especificações do *FlowSheet*.



## Otimização Estática

- Um problema de otimização é composto por:
  - 1 Uma função objetivo.
  - 2 O modelo da planta.
  - 3 Restrições operacionais.
- O ponto inicial de uma otimização no EMSO é um modelo de planta *FlowSheet*.
- Porém um *FlowSheet* consistente não tem graus de liberdade para otimizar.
- Então é necessário *criar* graus de liberdade para otimizar.
- Estes graus de liberdade são *criados* liberando especificações do *FlowSheet*.



## Otimização Estática

- Um problema de otimização é composto por:
  - 1 Uma função objetivo.
  - 2 O modelo da planta.
  - 3 Restrições operacionais.
- O ponto inicial de uma otimização no EMSO é um modelo de planta *FlowSheet*.
- Porém um *FlowSheet* consistente não tem graus de liberdade para otimizar.
- Então é necessário *criar* graus de liberdade para otimizar.
- Estes graus de liberdade são *criados* liberando especificações do *FlowSheet*.



## Otimização Estática

- Um problema de otimização é composto por:
  - 1 Uma função objetivo.
  - 2 O modelo da planta.
  - 3 Restrições operacionais.
- O ponto inicial de uma otimização no EMSO é um modelo de planta *FlowSheet*.
- Porém um *FlowSheet* consistente não tem graus de liberdade para otimizar.
- Então é necessário *criar* graus de liberdade para otimizar.
- Estes graus de liberdade são *criados* liberando especificações do *FlowSheet*.



## Otimização Estática

- Um problema de otimização é composto por:
  - 1 Uma função objetivo.
  - 2 O modelo da planta.
  - 3 Restrições operacionais.
- O ponto inicial de uma otimização no EMSO é um modelo de planta *FlowSheet*.
- Porém um *FlowSheet* consistente não tem graus de liberdade para otimizar.
- Então é necessário *criar* graus de liberdade para otimizar.
- Estes graus de liberdade são *criados* liberando especificações do *FlowSheet*.



## Otimização Estática

- Um problema de otimização é composto por:
  - 1 Uma função objetivo.
  - 2 O modelo da planta.
  - 3 Restrições operacionais.
- O ponto inicial de uma otimização no EMSO é um modelo de planta *FlowSheet*.
- Porém um *FlowSheet* consistente não tem graus de liberdade para otimizar.
- Então é necessário *criar* graus de liberdade para otimizar.
- Estes graus de liberdade são *criados* liberando especificações do *FlowSheet*.



## Otimização Estática

- Um problema de otimização é composto por:
  - 1 Uma função objetivo.
  - 2 O modelo da planta.
  - 3 Restrições operacionais.
- O ponto inicial de uma otimização no EMSO é um modelo de planta *FlowSheet*.
- Porém um *FlowSheet* consistente não tem graus de liberdade para otimizar.
- Então é necessário *criar* graus de liberdade para otimizar.
- Estes graus de liberdade são *criados* liberando especificações do *FlowSheet*.





## Otimização Estática

- Um problema de otimização é composto por:
  - 1 Uma função objetivo.
  - 2 O modelo da planta.
  - 3 Restrições operacionais.
- O ponto inicial de uma otimização no EMSO é um modelo de planta *FlowSheet*.
- Porém um *FlowSheet* consistente não tem graus de liberdade para otimizar.
- Então é necessário *criar* graus de liberdade para otimizar.
- Estes graus de liberdade são *criados* liberando especificações do *FlowSheet*.



## Solvers disponíveis

- **IPOPT:** Ponto interior (utiliza derivadas), pacote de código aberto (<http://projects.coin-or.org/Ipopt>).
- **Complex:** Poliedros flexíveis (não utiliza derivadas).
- **DiRect:** Divisão de retângulos (otimização global sem a utilização de derivadas).



# Linguagem

```
Optimization name [as base]
PARAMETERS #Opcional: Declaracao de parametros
VARIABLES #Opcional: Declaracao de variaveis
MINIMIZE #Funcao a ser minimizada ou maximizada
expression1;
expression2;

MAXIMIZE #Funcao a ser minimizada ou maximizada
expression3;
expression4;

EQUATIONS #Restricoes de igualdade ou desigualdade
expression5 = expression6;
expression7 > expression8;

FREE #Opcional Variaveis a serem liberadas (caso importado
de um FlowSheet)
variable1;
variable2;

OPTIONS
end
```



## Linguagem: Exemplo

---

```
Optimization hs71
  VARIABLES   #Declaracao das variaveis
  x1 as Real(Default=2, Lower=1, Upper=5);
  x2 as Real(Default=5, Lower=1, Upper=5);
  x3 as Real(Default=5, Lower=1, Upper=5);
  x4 as Real(Default=1, Lower=1, Upper=5);

  MINIMIZE   #Secao minimize
  x1*x4*(x1+x2+x3) + x3;

  EQUATIONS  #Restricoes de igualdade e desigualdade
  x1*x2*x3*x4 > 25;
  x1*x1 + x2*x2 + x3*x3 + x4*x4 = 40;

  OPTIONS    #Opcoes
  Dynamic = false;
end
```

---



## Linguagem: Exemplo

---

```
Optimization hs71
  VARIABLES   # Declaracao das variaveis
  x1 as Real(Default=2, Lower=1, Upper=5);
  x2 as Real(Default=5, Lower=1, Upper=5);
  x3 as Real(Default=5, Lower=1, Upper=5);
  x4 as Real(Default=1, Lower=1, Upper=5);
```

---

**Lower:** Limite inferior

**Upper:** Limite superior

**Default:** Chute inicial

### Observação:

A configuração apropriada dos limites da variável e da estimativa inicial melhora o desempenho do algoritmo.



## Resultados das Otimizações

- Ao longo da otimização, são apresentados resultados no `Console`, de acordo com o nível de detalhamento escolhido.
- Sempre ao finalizar a otimização é necessário ver o status final no `Console`, certificando-se que a mesma foi finalizado com sucesso.



## Resultados das Otimizações

- Ao longo da otimização, são apresentados resultados no `Console`, de acordo com o nível de detalhamento escolhido.
- Sempre ao finalizar a otimização é necessário ver o status final no `Console`, certificando-se que a mesma foi finalizado com sucesso.



## Exercício 1: Descrição

Uma refinaria produz 2 tipos de gasolinas: Aditivada (A) e Normal (N), que são vendidas a \$1,1 e \$1,0, respectivamente. A refinaria pode comprar 3 tipos de óleos de 3 diferentes fontes com os seguintes constituintes e preços:

| Tipo Óleo | Constituintes |      |     | Preço \$/litro |
|-----------|---------------|------|-----|----------------|
|           | A             | B    | C   |                |
| 1         | 0,70          | 0,2  | 0,1 | 0,60           |
| 2         | 0,25          | 0,25 | 0,5 | 0,50           |
| 3         | 0,60          | 0,1  | 0,3 | 0,45           |

- A gasolina A deve ter no mínimo 50% de A e não mais que 35% de C.
- A gasolina N **não** deve ter mais que 30% de C.

Determine como o óleo deve ser misturado para maximizar o lucro.





## Exercício 1: Resolução

Organizando o problema:

- $X_{i,j}$ : Quantidade do óleo  $i$  (1,2,3) usado para fazer a gasolina  $j$  (A,N)
- Função Objetivo:

- $\text{Lucro} = \text{Preço de venda da gasolina} - \text{Custo do óleo}$   
 $1,1 (X_{1,A} + X_{2,A} + X_{3,A}) + 1,0 (X_{1,N} + X_{2,N} + X_{3,N}) -$   
 $0,6 (X_{1,A} + X_{1,N}) - 0,5 (X_{2,A} + X_{2,N}) - 0,45 (X_{3,A} + X_{3,N})$

- Restrições de igualdade:

- A gasolina A deve ter no mínimo 50% de A e não mais que 35% de C.

$$0,7X_{1,A} + 0,25X_{2,A} + 0,6X_{3,A} \geq 0,5 (X_{1,A} + X_{2,A} + X_{3,A})$$

$$0,1X_{1,A} + 0,5X_{2,A} + 0,3X_{3,A} \leq 0,35 (X_{1,A} + X_{2,A} + X_{3,A})$$

- A gasolina N **não** deve ter mais que 30% de C.

$$0,1X_{1,N} + 0,5X_{2,N} + 0,3X_{3,N} \leq 0,3 (X_{1,N} + X_{2,N} + X_{3,N})$$



## Exercício 1: Resolução

Organizando o problema:

- $X_{i,j}$ : Quantidade do óleo  $i$  (1,2,3) usado para fazer a gasolina  $j$  (A,N)

- Função Objetivo:

- $\text{Lucro} = \text{Preço de venda da gasolina} - \text{Custo do óleo}$   
 $1,1 (X_{1,A} + X_{2,A} + X_{3,A}) + 1,0 (X_{1,N} + X_{2,N} + X_{3,N}) -$   
 $0,6 (X_{1,A} + X_{1,N}) - 0,5 (X_{2,A} + X_{2,N}) - 0,45 (X_{3,A} + X_{3,N})$

- Restrições de igualdade:

- A gasolina A deve ter no mínimo 50% de A e não mais que 35% de C.

$$0,7X_{1,A} + 0,25X_{2,A} + 0,6X_{3,A} \geq 0,5 (X_{1,A} + X_{2,A} + X_{3,A})$$

$$0,1X_{1,A} + 0,5X_{2,A} + 0,3X_{3,A} \leq 0,35 (X_{1,A} + X_{2,A} + X_{3,A})$$

- A gasolina N **não** deve ter mais que 30% de C.

$$0,1X_{1,N} + 0,5X_{2,N} + 0,3X_{3,N} \leq 0,3 (X_{1,N} + X_{2,N} + X_{3,N})$$



## Exercício 1: Resolução

Organizando o problema:

- $X_{i,j}$ : Quantidade do óleo  $i$  (1,2,3) usado para fazer a gasolina  $j$  (A,N)
- Função Objetivo:

$$\text{Lucro} = \text{Preço de venda da gasolina} - \text{Custo do óleo}$$

$$1,1(X_{1,A} + X_{2,A} + X_{3,A}) + 1,0(X_{1,N} + X_{2,N} + X_{3,N}) - 0,6(X_{1,A} + X_{1,N}) - 0,5(X_{2,A} + X_{2,N}) - 0,45(X_{3,A} + X_{3,N})$$

- Restrições de igualdade:

- A gasolina A deve ter no mínimo 50% de A e não mais que 35% de C.

$$0,7X_{1,A} + 0,25X_{2,A} + 0,6X_{3,A} \geq 0,5(X_{1,A} + X_{2,A} + X_{3,A})$$

$$0,1X_{1,A} + 0,5X_{2,A} + 0,3X_{3,A} \leq 0,35(X_{1,A} + X_{2,A} + X_{3,A})$$

- A gasolina N não deve ter mais que 30% de C.

$$0,1X_{1,N} + 0,5X_{2,N} + 0,3X_{3,N} \leq 0,3(X_{1,N} + X_{2,N} + X_{3,N})$$



## Exercício 1: Resolução

Organizando o problema:

- $X_{i,j}$ : Quantidade do óleo  $i$  (1,2,3) usado para fazer a gasolina  $j$  (A,N)
- Função Objetivo:

$$\text{Lucro} = \text{Preço de venda da gasolina} - \text{Custo do óleo}$$

$$1,1 (X_{1,A} + X_{2,A} + X_{3,A}) + 1,0 (X_{1,N} + X_{2,N} + X_{3,N}) - 0,6 (X_{1,A} + X_{1,N}) - 0,5 (X_{2,A} + X_{2,N}) - 0,45 (X_{3,A} + X_{3,N})$$

- Restrições de igualdade:

- A gasolina A deve ter no mínimo 50% de A e não mais que 35% de C.

$$0,7X_{1,A} + 0,25X_{2,A} + 0,6X_{3,A} \geq 0,5 (X_{1,A} + X_{2,A} + X_{3,A})$$

$$0,1X_{1,A} + 0,5X_{2,A} + 0,3X_{3,A} \leq 0,35 (X_{1,A} + X_{2,A} + X_{3,A})$$

- A gasolina N não deve ter mais que 30% de C.

$$0,1X_{1,N} + 0,5X_{2,N} + 0,3X_{3,N} \leq 0,3 (X_{1,N} + X_{2,N} + X_{3,N})$$



## Exercício 1: Resolução

Organizando o problema:

- $X_{i,j}$ : Quantidade do óleo  $i$  (1,2,3) usado para fazer a gasolina  $j$  (A,N)
- Função Objetivo:

$$\text{Lucro} = \text{Preço de venda da gasolina} - \text{Custo do óleo}$$

$$1,1(X_{1,A} + X_{2,A} + X_{3,A}) + 1,0(X_{1,N} + X_{2,N} + X_{3,N}) - 0,6(X_{1,A} + X_{1,N}) - 0,5(X_{2,A} + X_{2,N}) - 0,45(X_{3,A} + X_{3,N})$$

- Restrições de igualdade:
  - A gasolina A deve ter no mínimo 50% de A e não mais que 35% de C.

$$0,7X_{1,A} + 0,25X_{2,A} + 0,6X_{3,A} \geq 0,5(X_{1,A} + X_{2,A} + X_{3,A})$$

$$0,1X_{1,A} + 0,5X_{2,A} + 0,3X_{3,A} \leq 0,35(X_{1,A} + X_{2,A} + X_{3,A})$$

- A gasolina N não deve ter mais que 30% de C.

$$0,1X_{1,N} + 0,5X_{2,N} + 0,3X_{3,N} \leq 0,3(X_{1,N} + X_{2,N} + X_{3,N})$$



## Exercício 1: Resolução

Organizando o problema:

- $X_{i,j}$ : Quantidade do óleo  $i$  (1,2,3) usado para fazer a gasolina  $j$  (A,N)
- Função Objetivo:

$$\text{Lucro} = \text{Preço de venda da gasolina} - \text{Custo do óleo}$$

$$1,1 (X_{1,A} + X_{2,A} + X_{3,A}) + 1,0 (X_{1,N} + X_{2,N} + X_{3,N}) - 0,6 (X_{1,A} + X_{1,N}) - 0,5 (X_{2,A} + X_{2,N}) - 0,45 (X_{3,A} + X_{3,N})$$

- Restrições de igualdade:

- A gasolina A deve ter no mínimo 50% de A e não mais que 35% de C.

$$0,7X_{1,A} + 0,25X_{2,A} + 0,6X_{3,A} \geq 0,5 (X_{1,A} + X_{2,A} + X_{3,A})$$

$$0,1X_{1,A} + 0,5X_{2,A} + 0,3X_{3,A} \leq 0,35 (X_{1,A} + X_{2,A} + X_{3,A})$$

- A gasolina N **não** deve ter mais que 30% de C.

$$0,1X_{1,N} + 0,5X_{2,N} + 0,3X_{3,N} \leq 0,3 (X_{1,N} + X_{2,N} + X_{3,N})$$



## Exercício 1: Resolução

O problema de otimização final fica:

- Maximizar:

$$1,1 (X_{1,A} + X_{2,A} + X_{3,A}) + 1,0 (X_{1,N} + X_{2,N} + X_{3,N}) - 0,6 (X_{1,A} + X_{1,N}) - 0,5 (X_{2,A} + X_{2,N}) - 0,45 (X_{3,A} + X_{3,N})$$

- Sujeito à:

$$0,7X_{1,A} + 0,25X_{2,A} + 0,6X_{1,A} \geq 0,5 (X_{1,A} + X_{2,A} + X_{1,A})$$

$$0,1X_{1,A} + 0,5X_{2,A} + 0,3X_{1,A} \leq 0,35 (X_{1,A} + X_{2,A} + X_{1,A})$$

$$0,1X_{1,N} + 0,5X_{2,N} + 0,3X_{1,N} \leq 0,3 (X_{1,N} + X_{2,N} + X_{1,N})$$



## Exercício 2

Exemplo de *flash* estacionário:

- Abrir o exemplo  
sample/optimization/flash\_opt.mso.
- Rodar o `FlowSheet FlashSteadyTest` (sem otimização), veja o valor da variável `leves`.
- Em seguida, rodar a otimização e inspecionar a mesma variável.





## Exercício 2

Exemplo de *flash* estacionário:

- Abrir o exemplo  
sample/optimization/flash\_opt.mso.
- Rodar o `FlowSheet` `FlashSteadyTest` (sem otimização), veja o valor da variável `leves`.
- Em seguida, rodar a otimização e inspecionar a mesma variável.



## Exercício 2

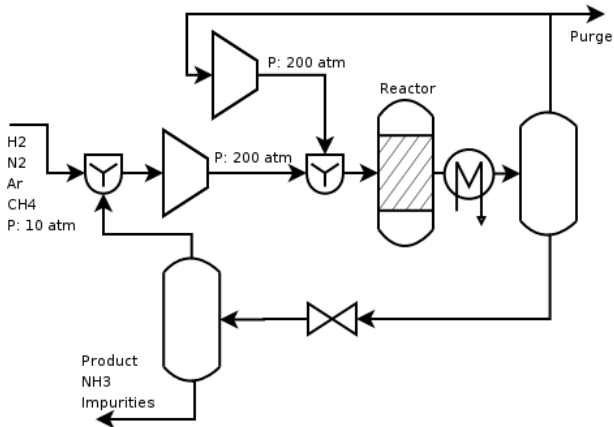
Exemplo de *flash* estacionário:

- Abrir o exemplo  
sample/optimization/flash\_opt.mso.
- Rodar o `FlowSheet` `FlashSteadyTest` (sem otimização), veja o valor da variável `leves`.
- Em seguida, rodar a otimização e inspecionar a mesma variável.



## Exercício 3

Exemplo de uma planta de amônia:



## Exercício 3

- **Abrir o exemplo**  
`sample/optimization/ammonia_opt.mso.`
- Conferir o status do problema no `Console`.
- Rodar a planta de produção de amônia sem otimização (`sample/optimization/ammonia.mso`).
- Comparar o resultado da função objetivo e das restrições antes e depois da otimização.



## Exercício 3

- **Abrir o exemplo**  
`sample/optimization/ammonia_opt.mso.`
- **Conferir o status do problema no Console.**
- Rodar a planta de produção de amônia sem otimização  
(`sample/optimization/ammonia.mso`).
- Comparar o resultado da função objetivo e das restrições antes e depois da otimização.



## Exercício 3

- Abrir o exemplo  
`sample/optimization/ammonia_opt.mso`.
- Conferir o status do problema no `Console`.
- Rodar a planta de produção de amônia sem otimização  
(`sample/optimization/ammonia.mso`).
- Comparar o resultado da função objetivo e das restrições antes e depois da otimização.



## Exercício 3

- Abrir o exemplo  
`sample/optimization/ammonia_opt.mso`.
- Conferir o status do problema no `Console`.
- Rodar a planta de produção de amônia sem otimização  
(`sample/optimization/ammonia.mso`).
- Comparar o resultado da função objetivo e das restrições antes e depois da otimização.



## Exercício 3: Opções

```
1   Dynamic = false;  
2   NLPsolveNLA = true;  
3   NLPsolver(File = "ipopt_emso",  
4             MaxIterations = 300,  
5             Hessian_approximation = "exact",  
6             RelativeAccuracy = 1e-6);  
7   end
```

**Linha 2** Resolve primeiro o `FlowSheet` e dá ao otimizador esse valor como estimativa inicial.

**Linha 3** Escolha do *solver*.

**Linha 4** Número máximo de iterações.

**Linha 5** Aproximação da Hessiana (derivada segunda), afeta o desempenho mas melhora o resultado.

**Linha 6** Critério de convergência.





## Otimização Inteira-Mista (MINLP)

- Em muitos casos industriais, as variáveis não se comportam de maneira contínua, entre elas:
  - Existência ou não de um equipamento.
  - Existência ou não de uma conexão (linha).
  - Número de equipamentos que devem ser usados para atingir uma determinada produção.
  - Prato de alimentação de uma coluna.
  - Número de peças produzidas.
  - Existência/disponibilidade ou não de um equipamento em um instante de tempo ou etapa do processo (sequenciamento de produção).



## Otimização Inteira-Mista (MINLP)

- Em muitos casos industriais, as variáveis não se comportam de maneira contínua, entre elas:
  - Existência ou não de um equipamento.
  - Existência ou não de uma conexão (linha).
  - Número de equipamentos que devem ser usados para atingir uma determinada produção.
  - Prato de alimentação de uma coluna.
  - Número de peças produzidas.
  - Existência/disponibilidade ou não de um equipamento em um instante de tempo ou etapa do processo (sequenciamento de produção).



## Otimização Inteira-Mista (MINLP)

- Em muitos casos industriais, as variáveis não se comportam de maneira contínua, entre elas:
  - Existência ou não de um equipamento.
  - Existência ou não de uma conexão (linha).
  - Número de equipamentos que devem ser usados para atingir uma determinada produção.
  - Prato de alimentação de uma coluna.
  - Número de peças produzidas.
  - Existência/disponibilidade ou não de um equipamento em um instante de tempo ou etapa do processo (sequenciamento de produção).



## Otimização Inteira-Mista (MINLP)

- Em muitos casos industriais, as variáveis não se comportam de maneira contínua, entre elas:
  - Existência ou não de um equipamento.
  - Existência ou não de uma conexão (linha).
  - Número de equipamentos que devem ser usados para atingir uma determinada produção.
  - Prato de alimentação de uma coluna.
  - Número de peças produzidas.
  - Existência/disponibilidade ou não de um equipamento em um instante de tempo ou etapa do processo (sequenciamento de produção).



## Otimização Inteira-Mista (MINLP)

- Em muitos casos industriais, as variáveis não se comportam de maneira contínua, entre elas:
  - Existência ou não de um equipamento.
  - Existência ou não de uma conexão (linha).
  - Número de equipamentos que devem ser usados para atingir uma determinada produção.
  - Prato de alimentação de uma coluna.
  - Número de peças produzidas.
  - Existência/disponibilidade ou não de um equipamento em um instante de tempo ou etapa do processo (sequenciamento de produção).



## Otimização Inteira-Mista (MINLP)

- Em muitos casos industriais, as variáveis não se comportam de maneira contínua, entre elas:
  - Existência ou não de um equipamento.
  - Existência ou não de uma conexão (linha).
  - Número de equipamentos que devem ser usados para atingir uma determinada produção.
  - Prato de alimentação de uma coluna.
  - Número de peças produzidas.
  - Existência/disponibilidade ou não de um equipamento em um instante de tempo ou etapa do processo (sequenciamento de produção).



## Otimização Inteira-Mista (MINLP)

- Em muitos casos industriais, as variáveis não se comportam de maneira contínua, entre elas:
  - Existência ou não de um equipamento.
  - Existência ou não de uma conexão (linha).
  - Número de equipamentos que devem ser usados para atingir uma determinada produção.
  - Prato de alimentação de uma coluna.
  - Número de peças produzidas.
  - Existência/disponibilidade ou não de um equipamento em um instante de tempo ou etapa do processo (sequenciamento de produção).



## Otimização Inteira-Mista (MINLP)

- A solução desse tipo de problema não é simplesmente o “arredondamento” da solução contínua mais próxima, pois, ao realizar o arredondamento, o problema pode vir a se tornar inviável em relação à satisfação das restrições.
- Esse tipo de problema é resolvido por outra classe de *solvers* específica.
- Esses *solvers* encontram o ótimo através de uma busca sistemática nas variáveis inteiras, conhecida como **busca em árvore**, sendo essa busca, não-combinatorial.





## Otimização Inteira-Mista (MINLP)

- A solução desse tipo de problema não é simplesmente o “arredondamento” da solução contínua mais próxima, pois, ao realizar o arredondamento, o problema pode vir a se tornar inviável em relação à satisfação das restrições.
- Esse tipo de problema é resolvido por outra classe de *solvers* específica.
- Esses *solvers* encontram o ótimo através de uma busca sistemática nas variáveis inteiras, conhecida como **busca em árvore**, sendo essa busca, não-combinatorial.



## Otimização Inteira-Mista (MINLP)

- A solução desse tipo de problema não é simplesmente o “arredondamento” da solução contínua mais próxima, pois, ao realizar o arredondamento, o problema pode vir a se tornar inviável em relação à satisfação das restrições.
- Esse tipo de problema é resolvido por outra classe de *solvers* específica.
- Esses *solvers* encontram o ótimo através de uma busca sistemática nas variáveis inteiras, conhecida como **busca em árvore**, sendo essa busca, não-combinatorial.



## Escrevendo o Problema

---

```
Optimization name [as base]
PARAMETERS      # Opcional: Declaracao de parametros
VARIABLES       # Opcional: Declaracao de variaveis
MINIMIZE        # Funcao a ser minimizada ou maximizada
expression1;
expression2;

MAXIMIZE        # Funcao a ser minimizada ou maximizada
expression3;
expression4;

EQUATIONS       # Restrioes de igualdade ou desigualdade
expression5 = expression6;
expression7 > expression8;

FREE            # Opcional Variaveis a serem liberadas (caso
                # importado de um FlowSheet)
variable1;
variable2;

OPTIONS

end
```

---



## Escrevendo o Problema

A diferença básica está na declaração das variáveis, que passa a ter o atributo de inteira (*Integer*).

---

```

Optimization minlp1
VARIABLES
x0 as Integer(Default=0, Lower=0, Upper=1);
x1 as Real(Default=0, Lower=0, Upper=1e10);
x2 as Real(Default=0, Lower=0, Upper=1e10);
x3 as Integer(Default=0, Lower=0, Upper=5);

MINIMIZE
-1*(x0 + x1 + x2);

EQUATIONS
(x1 - 0.5)^2 + (x2 - 0.5)^2 <= 0.25;
x0 - x1 <= 0;
x0 + x2 + x3 <= 2;

OPTIONS
Dynamic = false;
NLPsolveNLA = true;

NLPsolver(File="minlp_emso", derivative_test="second-order",
print_level=5);
end
    
```

---



## Escrevendo o Problema

```
Optimization minlp1
VARIABLES
x0 as Integer(Default=0, Lower=0, Upper=1);
x1 as Real(Default=0, Lower=0, Upper=1e10);
x2 as Real(Default=0, Lower=0, Upper=1e10);
x3 as Integer(Default=0, Lower=0, Upper=5);
```

- O limite superior da variável inteira é importante.
- Se a variável tem um comportamento **binário** (existe ou não-existe), essa deve ter seu limite superior fixado em “1”.
- Isso é importante, pois o *solver* trata de forma diferente as variáveis **inteiras** e **binárias** no seu processo interno de busca.
- O EMSO informa o *solver* que a variável é binária e o *solver* passa a tratá-la de uma forma diferente de uma variável inteira normal.



## Exercício 4

- Rodar as otimizações inteiras-mistas do arquivo `sample/optimization/sample_minlp` e ver os resultados.

### Dicas

- A configuração apropriada dos limites da variável e da estimativa inicial melhora o desempenho do algoritmo.
- A escolha dos parâmetros de configuração do *solver* pode levar ao sucesso ou insucesso da otimização.



## Exercício 4

- Rodar as otimizações inteiras-mistas do arquivo `sample/optimization/sample_minlp` e ver os resultados.

### Dicas

- A configuração apropriada dos limites da variável e da estimativa inicial melhora o desempenho do algoritmo.
- A escolha dos parâmetros de configuração do *solver* pode levar ao sucesso ou insucesso da otimização.



## Teses/Dissertações

- 2003** SOARES, R. P. *Desenvolvimento de um simulador genérico de processos dinâmicos*. Dissertação (Mestrado). UFRGS, Porto Alegre.
- 2006** BICCA, G. B. *Modelagem hierárquica de trocadores de calor casco e tubos*. Dissertação (Mestrado). UFRGS, Porto Alegre.
- 2007** STAUDT, P. B. *Modelagem e simulação dinâmica de colunas de destilação*. Dissertação (Mestrado). UFRGS, Porto Alegre.





## Teses/Dissertações

- 2003** SOARES, R. P. *Desenvolvimento de um simulador genérico de processos dinâmicos*. Dissertação (Mestrado). UFRGS, Porto Alegre.
- 2006** BICCA, G. B. *Modelagem hierárquica de trocadores de calor casco e tubos*. Dissertação (Mestrado). UFRGS, Porto Alegre.
- 2007** STAUDT, P. B. *Modelagem e simulação dinâmica de colunas de destilação*. Dissertação (Mestrado). UFRGS, Porto Alegre.



## Teses/Dissertações

- 2003** SOARES, R. P. *Desenvolvimento de um simulador genérico de processos dinâmicos*. Dissertação (Mestrado). UFRGS, Porto Alegre.
- 2006** BICCA, G. B. *Modelagem hierárquica de trocadores de calor casco e tubos*. Dissertação (Mestrado). UFRGS, Porto Alegre.
- 2007** STAUDT, P. B. *Modelagem e simulação dinâmica de colunas de destilação*. Dissertação (Mestrado). UFRGS, Porto Alegre.



## Teses/Dissertações

- 2007** SOARES, R. P. *Depuração para simuladores de processos baseados em equações*. Tese (Doutorado). UFRGS, Porto Alegre.
- 2008** RODRIGUES, R. *Modelagem e simulação de um gaseificador em leito fixo para o tratamento térmico de resíduos sólidos da indústria calçadista*. Dissertação (Mestrado). UFRGS, Porto Alegre.
- 2009** SARTOR, R. B. *Modelagem, simulação e otimização de uma unidade industrial de extração de óleos essenciais por arraste a vapor*. Dissertação (Mestrado). UFRGS, Porto Alegre.



## Teses/Dissertações

- 2007** SOARES, R. P. *Depuração para simuladores de processos baseados em equações*. Tese (Doutorado). UFRGS, Porto Alegre.
- 2008** RODRIGUES, R. *Modelagem e simulação de um gaseificador em leito fixo para o tratamento térmico de resíduos sólidos da indústria calçadista*. Dissertação (Mestrado). UFRGS, Porto Alegre.
- 2009** SARTOR, R. B. *Modelagem, simulação e otimização de uma unidade industrial de extração de óleos essenciais por arraste a vapor*. Dissertação (Mestrado). UFRGS, Porto Alegre.



## Teses/Dissertações

- 2007** SOARES, R. P. *Depuração para simuladores de processos baseados em equações*. Tese (Doutorado). UFRGS, Porto Alegre.
- 2008** RODRIGUES, R. *Modelagem e simulação de um gaseificador em leito fixo para o tratamento térmico de resíduos sólidos da indústria calçadista*. Dissertação (Mestrado). UFRGS, Porto Alegre.
- 2009** SARTOR, R. B. *Modelagem, simulação e otimização de uma unidade industrial de extração de óleos essenciais por arraste a vapor*. Dissertação (Mestrado). UFRGS, Porto Alegre.



## Artigos

**2008** RODRIGUES, R.; SOARES, R. P.; SECCHI, A. R. Coleção de 10 problemas numéricos típicos em Engenharia Química resolvidos com o simulador EMSO. In *XVII Congresso Brasileiro de Engenharia Química*. Recife, Brasil.

- Conteúdo relacionado disponível em [sample/miscellaneous/tenprobs](#)

**2010** RODRIGUES, R.; SOARES, R. P.; SECCHI, A. R. Teaching chemical reaction engineering using EMSO simulator. *Computer Application in Engineering Education*, v. 18, n. 4, p. 607–618. DOI: [10.1002/cae.20255](#)

- Conteúdo relacionado disponível em [sample/reactors/fogler](#)

### Recomendação:

Como citar o EMSO em publicações:

SOARES, R. P., SECCHI, A. R. EMSO: A new environment for modelling, simulation and optimisation. *Computer Aided Chemical Engineering*. v. 14, n. C, p. 947–952, 2003. DOI: [10.1016/S1570-7946\(03\)80239-0](#)



## Artigos

**2008** RODRIGUES, R.; SOARES, R. P.; SECCHI, A. R. Coleção de 10 problemas numéricos típicos em Engenharia Química resolvidos com o simulador EMSO. In *XVII Congresso Brasileiro de Engenharia Química*. Recife, Brasil.

- Conteúdo relacionado disponível em [sample/miscellaneous/tenprobs](#)

**2010** RODRIGUES, R.; SOARES, R. P.; SECCHI, A. R. Teaching chemical reaction engineering using EMSO simulator. *Computer Application in Engineering Education*, v. 18, n. 4, p. 607–618. DOI: [10.1002/cae.20255](#)

- Conteúdo relacionado disponível em [sample/reactors/fogler](#)

### Recomendação:

Como citar o EMSO em publicações:

SOARES, R. P., SECCHI, A. R. EMSO: A new environment for modelling, simulation and optimisation. *Computer Aided Chemical Engineering*. v. 14, n. C, p. 947–952, 2003. DOI: [10.1016/S1570-7946\(03\)80239-0](#)



## Artigos

**2008** RODRIGUES, R.; SOARES, R. P.; SECCHI, A. R. Coleção de 10 problemas numéricos típicos em Engenharia Química resolvidos com o simulador EMSO. In *XVII Congresso Brasileiro de Engenharia Química*. Recife, Brasil.

- Conteúdo relacionado disponível em [sample/miscellaneous/tenprobs](#)

**2010** RODRIGUES, R.; SOARES, R. P.; SECCHI, A. R. Teaching chemical reaction engineering using EMSO simulator. *Computer Application in Engineering Education*, v. 18, n. 4, p. 607–618. DOI: [10.1002/cae.20255](#)

- Conteúdo relacionado disponível em [sample/reactors/fogler](#)

### Recomendação:

Como citar o EMSO em publicações:

SOARES, R. P., SECCHI, A. R. EMSO: A new environment for modelling, simulation and optimisation. *Computer Aided Chemical Engineering*. v. 14, n. C, p. 947–952, 2003. DOI: [10.1016/S1570-7946\(03\)80239-0](#)





## Artigos

**2008** RODRIGUES, R.; SOARES, R. P.; SECCHI, A. R. Coleção de 10 problemas numéricos típicos em Engenharia Química resolvidos com o simulador EMSO. In *XVII Congresso Brasileiro de Engenharia Química*. Recife, Brasil.

- Conteúdo relacionado disponível em [sample/miscellaneous/tenprobs](#)

**2010** RODRIGUES, R.; SOARES, R. P.; SECCHI, A. R. Teaching chemical reaction engineering using EMSO simulator. *Computer Application in Engineering Education*, v. 18, n. 4, p. 607–618. DOI: [10.1002/cae.20255](#)

- Conteúdo relacionado disponível em [sample/reactors/fogler](#)

### Recomendação:

Como citar o EMSO em publicações:

SOARES, R. P., SECCHI, A. R. EMSO: A new environment for modelling, simulation and optimisation. *Computer Aided Chemical Engineering*. v. 14, n. C, p. 947–952, 2003. DOI: [10.1016/S1570-7946\(03\)80239-0](#)



O *blog ChEngineer Space* mantém uma seção sobre o simulador EMSO:

<http://chengineer.com/index.php/tag/emso>

- Simulating dynamic tanks in EMSO (Part 1)
- Simulating dynamic tanks in EMSO (Part 2)
- EMSO Installation on Ubuntu 10.04 32 bits



O *blog ChEngineer Space* mantém uma seção sobre o simulador EMSO:

<http://chengineer.com/index.php/tag/emso>

- Simulating dynamic tanks in EMSO (Part 1)
- Simulating dynamic tanks in EMSO (Part 2)
- EMSO Installation on Ubuntu 10.04 32 bits



O *blog ChEngineer Space* mantém uma seção sobre o simulador EMSO:

<http://chengineer.com/index.php/tag/emso>

- Simulating dynamic tanks in EMSO (Part 1)
- Simulating dynamic tanks in EMSO (Part 2)
- EMSO Installation on Ubuntu 10.04 32 bits



O *blog ChEngineer Space* mantém uma seção sobre o simulador EMSO:

<http://chengineer.com/index.php/tag/emso>

- Simulating dynamic tanks in EMSO (Part 1)
- Simulating dynamic tanks in EMSO (Part 2)
- EMSO Installation on Ubuntu 10.04 32 bits



## Suporte

- Suporte pode ser obtido via sistema de *Ticket* na página do Projeto ALSOC:
  - 1 Acessar a página do Projeto:  
<http://www.enq.ufrgs.br/alsoc>
  - 2 Fazer o registro na seção **Register** no canto superior direito da página.
  - 3 Uma vez que se esteja logado, a opção **New Ticket** ficará disponível.
  - 4 Preencher o formulário dando atenção especial aos campos **Component** (*EMSO, EML, EMSO GUI, EMSO Solvers, Portal* ou *VRTherm*) e **Type** (*defect, enhancement* ou *task*). Também é possível adicionar um arquivo (texto ou imagem) para complementação.

### Importante

Os *tickets* já criados estão disponíveis na seção **View Ticket**. Verifique os *tickets* abertos e fechados antes de abrir o seu para evitar repetições.



## Suporte

- Suporte pode ser obtido via sistema de *Ticket* na página do Projeto ALSOC:
  - 1 Acessar a página do Projeto:  
<http://www.enq.ufrgs.br/alsoc>
  - 2 Fazer o registro na seção **Register** no canto superior direito da página.
  - 3 Uma vez que se esteja logado, a opção **New Ticket** ficará disponível.
  - 4 Preencher o formulário dando atenção especial aos campos **Component** (*EMSO, EML, EMSO GUI, EMSO Solvers, Portal* ou *VRTherm*) e **Type** (*defect, enhancement* ou *task*). Também é possível adicionar um arquivo (texto ou imagem) para complementação.

### Importante

Os *tickets* já criados estão disponíveis na seção **View Ticket**. Verifique os *tickets* abertos e fechados antes de abrir o seu para evitar repetições.



## Suporte

- Suporte pode ser obtido via sistema de *Ticket* na página do Projeto ALSOC:
  - 1 Acessar a página do Projeto:  
<http://www.enq.ufrgs.br/alsoc>
  - 2 Fazer o registro na seção **Register** no canto superior direito da página.
  - 3 Uma vez que se esteja logado, a opção **New Ticket** ficará disponível.
  - 4 Preencher o formulário dando atenção especial aos campos **Component** (*EMSO, EML, EMSO GUI, EMSO Solvers, Portal* ou *VRTherm*) e **Type** (*defect, enhancement* ou *task*). Também é possível adicionar um arquivo (texto ou imagem) para complementação.

### Importante

Os *tickets* já criados estão disponíveis na seção **View Ticket**. Verifique os *tickets* abertos e fechados antes de abrir o seu para evitar repetições.





## Suporte

- Suporte pode ser obtido via sistema de *Ticket* na página do Projeto ALSOC:
  - 1 Acessar a página do Projeto:  
<http://www.enq.ufrgs.br/alsoc>
  - 2 Fazer o registro na seção **Register** no canto superior direito da página.
  - 3 Uma vez que se esteja logado, a opção **New Ticket** ficará disponível.
  - 4 Preencher o formulário dando atenção especial aos campos **Component** (*EMSO, EML, EMSO GUI, EMSO Solvers, Portal* ou *VRTherm*) e **Type** (*defect, enhancement* ou *task*). Também é possível adicionar um arquivo (texto ou imagem) para complementação.

### Importante

Os *tickets* já criados estão disponíveis na seção **View Ticket**. Verifique os *tickets* abertos e fechados antes de abrir o seu para evitar repetições.



## Suporte

- Suporte pode ser obtido via sistema de *Ticket* na página do Projeto ALSOC:
  - 1 Acessar a página do Projeto:  
<http://www.enq.ufrgs.br/alsoc>
  - 2 Fazer o registro na seção **Register** no canto superior direito da página.
  - 3 Uma vez que se esteja logado, a opção **New Ticket** ficará disponível.
  - 4 Preencher o formulário dando atenção especial aos campos **Component** (*EMSO, EML, EMSO GUI, EMSO Solvers, Portal* ou *VRTherm*) e **Type** (*defect, enhancement* ou *task*). Também é possível adicionar um arquivo (texto ou imagem) para complementação.

### Importante

Os *tickets* já criados estão disponíveis na seção **View Ticket**. Verifique os *tickets* abertos e fechados antes de abrir o seu para evitar repetições.



## Suporte

- Suporte pode ser obtido via sistema de *Ticket* na página do Projeto ALSOC:
  - 1 Acessar a página do Projeto:  
<http://www.enq.ufrgs.br/alsoc>
  - 2 Fazer o registro na seção **Register** no canto superior direito da página.
  - 3 Uma vez que se esteja logado, a opção **New Ticket** ficará disponível.
  - 4 Preencher o formulário dando atenção especial aos campos **Component** (*EMSO, EML, EMSO GUI, EMSO Solvers, Portal* ou *VRTherm*) e **Type** (*defect, enhancement* ou *task*). Também é possível adicionar um arquivo (texto ou imagem) para complementação.

### Importante

Os *tickets* já criados estão disponíveis na seção **View Ticket**. Verifique os *tickets* abertos e fechados antes de abrir o seu para evitar repetições.

