# Problema de optimização de Gestão Empresarial utilizando PLR

FEUP-PLOG, Mário Esteves (up201607940@fe.up.pt), Turma 3MIEIC01

Faculdade de Engenharia da Universidade do Porto
`https://sigarra.up.pt/feup/pt/web_page.inicial`

**Resumo** Dado um conjunto de critérios com diversas prioridades, e um conjunto de medidas que afetam esses mesmos critérios, devem ser escolhidas as medidas que em conjunto, maximizem o retorno sob cada critério, enquanto dentro do orcamento disponível. Para a resolução deste problema, foi implementada uma solução em Prolog, utilizando a biblioteca CLPFD.

**Keywords:** CLPFD, Prolog, Problema, Optimização, Gestão

## 1 Introdução

No âmbito da unidade curricular de Programação Lógica da Faculdade de Engenharia da Universidade do Porto, foi proposto um problema de optimização que lida com uma situação de uma empresa que tem acesso a dados do ambiente, que correspondem critérios a obedecer, e um conjunto de medidas a adotar perante estes mesmos critérios. Essa empresa deve adotar um conjunto de medidas das existentes, que maximizem o retorno sob as prioridades dos critérios a obedecer. Para isso foi implementada uma solução em Prolog, utilizando a biblioteca de programação com restrições em domínios finitos CLPFD.
Neste artigo é abordada a modelação dos dados de entrada e saída no programa, bem como as restrições implementadas para a resolução deste problema.

## 2 Descrição do Problema

Este problema lida com os seguintes fatores:

- um conjunto de critérios, que têm prioridades entre 0 e 1, e cuja soma dessas prioridades tem de ser obrigatoriamente 1;
- um conjunto de medidas, que são definidas pelo seu custo de implementação (custo no orçamento), e pela afetação em cada um dos critérios, sendo que cada afetação pode ser positiva, negativa, ou nula;
- e finalmente, o orçamento disponível à empresa, para a implementação dessas medidas.

Tendo em conta estes fatores, a representação interna de cada um destes requesitos tem as seguintes restrições:

- cada critério têm uma representação em ponto flutuante;
- cada medida é representada por um custo de implementação sendo esse mesmo um número inteiro, e uma lista de tamanho sempre equivalente ao número de critérios utilizados, em que cada elemento é um número inteiro.
- o orçamento é representado apenas por um número inteiro.

Para se saber se uma medida deve ser adotada, deve ser verificado se o somatório dos custos dessa medida com os custos das medidas já adotadas ultrapassa o orçamento disponível.
Para cada medida deve ser calculado o *"custo relativo"*, que consiste do produto escalar entre todas as prioridades de cada critério (em percentagem), e as afetações nos respetivos critérios. O somatório dos custos relativos das medidas escolhidas deve ser maximizado para aumentar o sucesso da empresa.

## 3 Abordagem de implementação do problema

Nesta secção descreve-se com detalhe a implementação do problema em prolog.

### 3.1 Variáveis de decisão

Para as variáveis de decisão temos uma lista de valores que podem ser apenas 0 ou 1, que corresponde à utilização (valor a 1), ou não da medida correspondente ao índice desse valor na lista (sendo que o indíce começa por 1 e não por 0). Esta decisão permite que essa lista de valores atue como multiplicador dos custos, e dos custos relativos.
Como variáveis do problema, temos os vários critérios impostos pelo ambiente em que a empresa se encontra, caracterizados pelas suas prioridades, cada custo de cada medida bem como cada uma afeta os critérios existentes, e o orçamento disponível.
Tendo isto em conta, foram implementados factos na base de conhecimento do prolog da seguinte forma:

```
orcamento_disponivel(30).
criterio(ID, Peso).
medida(ID, Custo, [AfetaçõesAosCritérios]).
```

São de notar dois fatores: a) ambos os IDs dos critérios e das medidas, não são utilizados na etiquetagem dos valores da variável de domínio, e b) o peso dos critérios é representado por um número de vírgula flutuante, que no que diz respeito à biblioteca, tem de ser convertido num número inteiro antes de poder ser incluido nas restrições impostas pelo problema.

### 3.2 Restrições

Como restrições do problema temos as seguintes:

- a soma dos pesos dos critérios tem de ser igual a 1;
- o custo total das medidas implementadas tem de ser menor ou igual ao orçamento disponível.
- finalmente, a afetação das medidas escolhidas sob os critérios existentes deve ser maximizado.

Devido a isso as seguintes restrições foram impostas ao solver da biblioteca CLPFD, no predicado de resolução do problema:

```
sum(Prioridades, #=, 100). % Soma das prioridades igual a 100.
assegurar_custo_medidas_menor_que_orcamento(ListaMedidas,
CustoMedidas, Orcamento).
% Assegura que perante um dado Orçamento, as medidas escolhidas
não ultrapassam o orçamento.
labeling([maximize(CustoRelativo)], ListaMedidas). % Assegura
a maximização do CustoRelativo total perante as medidas
escolhidas.
```

O predicado sum/3, assegura que a soma da lista contendo por ordem as prioridades dos diversos critérios existentes, seja 100. Isto porque a biblioteca CLPFD, lida com domínios finitos, e para que os valores das prioridades dos critérios sejam compativeis com esta convenção, simplemente multiplicamos todos os valores de prioridade por 100.
Para este fim, foi implementado o predicado obter_percentagens_de_criterios/2, que dada uma lista de prioridades em números reais, devolve uma lista com esses mesmos valores multiplicados por 100 e sem qualquer vírgula, recorrendo ao predicado integer/1.

### 3.3 Função de avaliação

Para a função de avaliação dos dados deste problema temos o predicado gest_empr/0, efectua as seguintes operações:

1. impõe o domínio entre 0 e 1 às variáveis de decisão (ListaMedidas);
2. mantém o tamanho da lista anteriormente referida, igual ao número de medidas existentes;
3. obtém os dados do problema (orçamento, custos das medidas, prioridades dos diversos critérios, e as afetações de cada medida);
4. através das listas de prioridades e das listas de afectações de cada medida calcula-se o custo relativo de cada medida, e obtém-se uma lista dos custos relativos das medidas respetivas;
5. finalmente começam-se por se impor as restrições ao problema, começando pela restrição da soma das prioridades dos critérios ter que ser igual a 1;

6. a restrição seguinte impõe que o produto escalar entre a lista de variáveis de decisão e a lista dos custos de cada medida;
7. antes da etiquetagem dos valores, calcula-se o custo relativo total perante as variáveis de decisão escolhidas;
8. executa-se o predicado labeling/2, com a opção de maximização para a variável representativa do custo relativo total;
9. finalmente escrevem-se os valores para o ecrã utilizando o predicado escrever_medidas_adotar/2.

Através desta aproximação ao problema, vê-se imediatamente o poder em utilizar este domínio: devido ao facto de todas as listas serem obtidas e manterem a ordem dos factos existentes na base de conhecimento, podemos fazer operações de produtos escalares para obtenção da maior parte dos dados.
Claramente também se vê as duas maiores limitações: a) as listas de afetações têm de ser sempre de tamanho igual ao número de critérios existentes, e b) os IDs presentes em ambos as medidas e os critérios têm de estar sempre em ordem crescente de cima para baixo na base de conhecimento para que os indíces coincidam com os IDs das próprias medidas.

### 3.4 Estratégia de pesquisa

Para a estratégia de etiquetagem dos valores foi apenas utilizadas a opção de maximização para maximizar o somatório dos custos relativos das medidas escolhidas.
Esta foi a única opção utilizada devido a duas razões predominantes:

- um dos requesitos do problema é a maximização do custo relativo (melhoria obtida), perante os critérios existentes;
- as outras opções não têm grande relevância, pois o domínio das varíaveis deste problema é binário, e todas as outras opções afetam a pesquisa em termos de escolha das variáveis. Isto é, se de dois valores só temos de escolher um, não importa a ordem em como fazemos a verificação das mesmas.

## 4 Visualização da Solução

Para a visualização da solução, foram implementados os predicados escrever_medidas_adotar/2 e escrever_medidas_adotar/3, em que o primeiro apenas escreve o cabeçalho da mensagem, e após isso chama o segundo com um contador que indica o ID de medida atual. Isto mostra uma mensagem ao utilizador com o seguinte formato:

```
As medidas a adotar são as seguintes:
|Medida1|Medida2|...|MedidaN|
Estas medidas têm um custo relativo de: CustoRelativo
```

É de notar que o CustoRelativo é formatado na forma de um valor decimal, ou seja, é calculado o valor decimal do custo relativo dividindo o mesmo por 100.

## 5 Resultados obtidos

Para testar a solução implementada, foram criados dois conjuntos de dados, dos quais devem ser comentados aqueles que não vão ser utilizados antes da chamada do solver. De seguida são mostrados em formato de tabela os dados de cada um dos conjuntos do problema, bem como a solução obtida para cada um dos mesmos.
Para o conujunto 1 temos:

**Tabela 1.** Critérios presentes no conjunto 1

| Critério | Prioridade |
|---|---|
| 1 | 0.1 |
| 2 | 0.4 |
| 3 | 0.5 |

**Tabela 2.** Medidas presentes no conjunto 1

| Medida | Custo | Afetação aos diversos critérios | Custo Relativo (Calculado pelo solver) |
|---|---|---|---|
| 1 | 20 | [4, -2, 1] | 0.1 |
| 2 | 5 | [-5, 3, 0] | 0.7 |
| 3 | 7 | [0, 0, 5] | 2.5 |

Neste conjunto o orçamento disponível é de 30 unidades. Na execução do solver temos o seguinte resultado:

```
As medidas a adotar são as seguintes:
|2|3|
Estas medidas têm um custo relativo de: 3.2
```

Como podemos ver o maior somatório de custos relativos é apenas usando a medida 2, e 3. O orçamento também é cumprido pois, o maior somatório é de 3.2 com um custo de 12, se o orçamento disponível fosse maior, então poderíamos implementar todas as medidas existentes (com um custo relativo de 3.3).

Para o próximo conjunto temos os seguintes dados:

**Tabela 3.** Critérios presentes no conjunto 2

| Critério | Prioridade |
|---|---|
| 1 | 0.34 |
| 2 | 0.1 |
| 3 | 0.26 |
| 4 | 0.3 |

**Tabela 4.** Medidas presentes no conjunto 2

| Medida | Custo | Afetação aos diversos critérios | Custo Relativo (Calculado pelo solver) |
|---|---|---|---|
| 1 | 30 | [6, -5, 5, 3] | 3.74 |
| 2 | 10 | [-10, 1, 5, 1] | -1.7 |
| 3 | 5 | [1, -2, -2, 3] | 0.52 |
| 4 | 6 | [5, -2, 0, 0] | 1.18 |
| 5 | 3 | [-1, 3, -1, 3] | 0.6 |

Neste conjunto o orçamento disponível é de 50 unidades. Após a execução do solver temos o seguinte resultado:

```
As medidas a adotar são as seguintes:
|1|3|4|5|
Estas medidas têm um custo relativo de: 6.36
```

Como podemos ver pelo resultado mostrado, apesar de termos orçamento para aplicar-mos todas as medidas, como é maximizado o custo relativo, a medida 2 não é adotada pois o seu custo relativo é negativo e como consequência, diminui o retorno esperado (custo relativo) total.

## 6 Conclusões e Trabalho Futuro

Com este trabalho, conseguiu-se entender o quão poderoso o paradigma da programação que o prolog oferece, em cojunção com a própria utilização da programação com restrições realmente é.
Apesar da utilização das capacidades de resolução ter sido um pouco menos ótima (lista de varíaveis de solução tem apenas domínio binário, apesar de estar a ser usada a livraria de resolução em domínios finitos), acho que a solução chegada é aceitável para o problema posto.
Possíveis melhoramentos futuros será a utilização de valores não binários, mas que realmente representem os valores dos IDs de cada medida, bem como algumas optimizações que sejam necessárias no código.

## Referências

1. Cardoso, H.L.: Programação em Lógica com Restrições (2015)
2. Cardoso, H.L.: Programação em Lógica com Restrições no SICSTUS Prolog (2017)
3. SICSTUS Prolog Documentation, `https://sicstus.sics.se/documentation.html`