

# **Programação em Lógica com Restrições na Resolução de Problemas de Decisão e Optimização: “*Arrows with Voids*”**

FEUP-PLOG, Turma 3MIEIC5, *ArrowsWithVoids\_2*

Mário Macedo – ei12105@fe.up.pt

Pedro Faria – ei12097@fe.up.pt

## **Resumo**

Arrows with voids, é um problema cujo objectivo é preencher um tabuleiro, cercado por setas que apontam para o interior deste, com números entre 0 e 8 de forma a maximizar a soma de todos os números no tabuleiro. A tentativa para chegar a uma solução para este problema foi baseada em Programação em Lógica utilizando restrições. Para isso, definiram-se dois conjuntos de variáveis, um para as setas e outro para os números, e aplicaram-se as restrições para cada um deles, restrições estas que foram rígidas e flexíveis. No entanto, não se conseguiu chegar a algum tipo de resultado porque não foi conseguido implementar uma associação viável entre números e setas como seria expectável.

## 1 Introdução

Este trabalho foi realizado no seguimento da matéria leccionada nas aulas de Programação em Lógica tendo como objectivo recorrer ao uso de restrições na resolução de problemas de decisão e/ou optimização. Neste caso concreto, o problema chama-se “Arrows with Voids” e o objectivo é preencher um tabuleiro de  $M$  por  $N$  dimensões, com números de 0 até 8, sendo que não pode haver repetições nas filas e nas colunas de forma a obter o maior somatório possível de todos os números. No sentido de resolver este problema definiu-se um conjunto de variáveis e declarou-se o domínio em que se poderiam expressar. Foram ainda definidas todas as restrições impostas de forma a que a solução fosse a correcta.

No restante documento são explorados em detalhe os métodos adoptados assim como são explorados e discutidos os resultados obtidos e os resultados pretendidos.

## 2 Descrição do problema

O *Arrows with Voids* consiste num problema, simultaneamente, de decisão e de optimização isto porque o objectivo é fazer com que a soma dos números que constituem o tabuleiro ser o maior possível obedecendo a um conjunto de decisões e restrições.

Mais concretamente, o objectivo do problema é: dado um tabuleiro de dimensões  $M$  por  $N$ , em que  $M$  e  $N$  têm de ser menores que 10, ter de se colocar números em cada posição e setas, que apontam para o interior do tabuleiro, na borda do tabuleiro de forma a que a soma dos números seja a maior possível. Claro está que existem algumas condições para o preenchimento do tabuleiro:

- Os números vão de 0 a 8;
- Cada número representa a quantidade de setas que apontam para ele (directamente ou diagonalmente);
- A mesma coluna e a mesma linha não podem conter números repetidos;
- Em vez dos números podem ser deixados os lugares em branco tendo em atenção que lugares em branco não podem estar adjacentes.

## 3 Abordagem

### 3.1 Variáveis de Decisão

Neste problema foram necessárias 2 conjuntos de variáveis de decisão: um conjunto para representar os números e outro para representar as setas.

O conjunto que representa os números tem um domínio entre 0 e 9 em que de 0 a 8 representa a gama de valores que cada posição do tabuleiro pode assumir e o 9 representa o espaço vazio.

O conjunto que representa as setas tem um domínio entre 1 e 3 onde 1 diz respeito à seta horizontal ou vertical e o 2 e o 3 representam as setas diagonais com direcções e sentidos diferentes.

### 3.2 Restrições

Foram aplicadas os dois tipos de restrições, isto é restrições rígidas e flexíveis.

As restrições rígidas implementadas foram:

- O domínio das variáveis, uma vez que os números nunca podem exceder este domínio quando é feito o *labeling*;
- O *all\_distinct* uma vez que em cada linha e coluna não podem haver números repetidos e os espaços em branco não podem partilhar arestas.

A única restrição flexível implementada foi a que corresponde às setas, uma vez que, para cada número no tabuleiro existe mais do que uma configuração possível de setas de forma a que o número corresponda ao número de setas que apontam para a sua posição.

### 3.3 Estratégia de Pesquisa

Uma vez que o objectivo do trabalho era maximizar o somatório dos números no interior do tabuleiro, o *labeling* foi feito tendo em conta o valor da variável Sum. Valor esse calculado através da soma de todos os elementos do tabuleiro de jogo, à excepção do 9, que representa o espaço vazio.

## 4 Visualização da Solução

Como pedido no enunciado, a solução mostra simplesmente o valor do somatório, e os elementos do tabuleiro (sem espaços) separando as linhas por virgulas. No caso de uma célula vazia, é mostrado um hífen.

## 5 Resultados

Como não conseguimos implementar a solução na totalidade foi-nos impossível medir qualquer tipo de resultados, no entanto, achamos que a solução para a qual trabalhamos não seria a mais eficiente no que toca à velocidade de resolução do problema.

## **6 Conclusão**

Os objectivos aos quais nos propusemos não foram totalmente cumpridos uma vez que não conseguimos implementar uma ligação entre os números e as setas, fazendo com que a restrição que diz que o número corresponde ao número de setas que apontam para ele, o que resulta num *labeling* incorrecto. Esta lacuna deve-se principalmente a dificuldades sentidas com a linguagem de programação PROLOG e um pouco com a carga de trabalho ao qual estávamos sujeitos.

No entanto, acreditamos que as restantes restrições foram implementadas com sucesso.

## **Bibliografia**

- Arrows with Voids - <http://logicmastersindia.com/lmitests/dl.asp?v=0&attachmentid=431>
- Slides fornecidos pelos docentes da Unidade Curricular <https://moodle.up.pt/mod/resource/view.php?id=70315>
- [http://www.lel.ed.ac.uk/facilities/howto/prolog/sicstus\\_32.html](http://www.lel.ed.ac.uk/facilities/howto/prolog/sicstus_32.html)
- <http://www.pathwayslms.com/swipltuts/clpfd/clpfd.html>