



Universidade do Minho

Métodos Determinísticos de Investigação Operacional

MIEI - 3º ANO - 1º SEMESTRE
UNIVERSIDADE DO MINHO

TRABALHO PRÁTICO EXPERIMENTAL II



André Ferreira
A64296

Tiago Gomes
A69853

Sébastião Freitas
A71074

Gonçalo Almeida
A84610

Angélica Cunha
A84398

Braga, 12 de dezembro de 2020

Conteúdo

1	Introdução	3
2	Modelação	4
2.1	Objetivo	4
2.2	Formulação do Problema	5
2.2.1	Variáveis de Decisão	5
2.2.2	Restrições	5
2.2.3	Função Objetivo	5
2.3	Rede do Problema de Fluxo Máximo	5
2.3.1	Input Relax4	7
2.3.2	Output Relax4	8
2.3.3	Interpretação da Solução Óptima Relax4	9
2.4	Validação do modelo	11
2.4.1	Variáveis de Decisão	11
2.4.2	Restrições	11
2.4.3	Função Objetivo	12
2.4.4	Output LPSolve	12
2.4.5	Interpretação da Solução Óptima LPSolve	13
3	Conclusão	14

Lista de Figuras

2.1	Novo quadro de inventário	4
2.2	Novo quadro de lucros	4
2.3	Exemplo de Extracção	4
2.4	Grafo Proposto	6
2.5	Output Relax4	8
2.6	Grafo Bipartido	9
2.7	Grafo Solução	10
2.8	Output LPSolve	12
2.9	Segmento do output LPSolve	12
2.10	Grafo Solução LPSolve	13

1. Introdução

O presente relatório irá abordar a elaboração do projeto desenvolvido no âmbito da unidade curricular de Modelos Determinísticos de Investigação Operacional. Neste projeto é apresentado um problema comum de maximização, em que se pretende descobrir quais os blocos a extrair, de modo a maximizar o lucro.

Mais especificamente, temos o caso de uma exploração de minas a céu aberto; na fase da exploração, o terreno é escavado e são extraídos blocos. Para extrair um bloco, é necessário extrair três blocos, o que está imediatamente por cima do bloco a retirar e os dois que lhe são adjacentes. Para a realização deste projeto, foi-nos fornecido um quadro de inventário de minério onde nos diz o valor estimado do inventário de minério em cada bloco.

De modo a apresentar uma solução a este problema, procedemos a modelá-lo, de modo a poder descrevê-lo de um modo mais facilmente comprehensível e objetivo, permitindo-nos seguidamente aplicar métodos de programação linear, com o auxílio de software, para encontrar uma solução ótima para o problema.

2. Modelação

De forma a respondermos às necessidades do enunciado proposto, começamos por realizar as alterações necessárias ao quadro do inventário de minério, tendo em conta o número mecanográfico 84610, seguido da formulação do modelo para atingir o objectivo proposto do problema de fluxo máximo.

nível -1							10	8			
nível -2	—					12	14	15	40		—
nível -3	—	—			16				20	—	—
nível -4	—	—	—	3	18	4			6	—	—
nível -5	—	—	—	—	20	1		0	—	—	—

Figura 2.1: Novo quadro de inventário

nível -1	-1	-1	-1	-1	-1	-1	9	7	-1	-1	-1
nível -2	—	-2	-2	-2	-2	10	12	13	38	-2	-2
nível -3	—	—	-3	-3	13	-3	-3	-3	17	-3	—
nível -4	—	—	—	-1	14	0	-4	-4	2	—	—
nível -5	—	—	—	—	15	-4	-5	-5	—	—	—

Figura 2.2: Novo quadro de lucros

2.1 Objetivo

O problema consiste em saber quais os blocos que devem ser extraídos de modo a maximizar o lucro. Para extrair um bloco de um certo nível, é necessário extraír os blocos da seguinte forma:

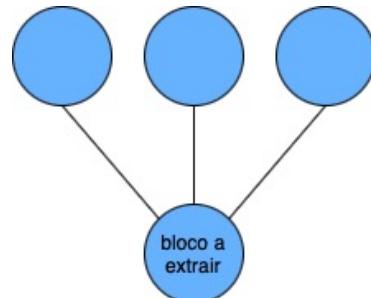


Figura 2.3: Exemplo de Extracção

2.2 Formulação do Problema

Estando perante um problema de fecho máximo com maximização do fluxo, assumimos um grafo orientado $G = (V, A)$, em que cada vértice $j \in V$ tem associado um peso c_j .

2.2.1 Variáveis de Decisão

Quanto a variáveis de decisão, temos que:

x_i : variável binária que determina se o vértice i pertence ao fecho de um grafo ou não. Assume o valor de 1 caso pertença e o valor de 0 caso contrário.

2.2.2 Restrições

Temos restrições lógicas de implicação, que traduzem que, se o vértice i pertencer ao fecho de um grafo, então o vértice j também deve pertencer.

$$\begin{aligned} x_i - x_j &\leq 0, \forall (i, j) \in A \\ x_j &\text{ binário, } \forall j \in V \end{aligned}$$

No nosso caso, de modo a extrair um bloco, devemos extrair o bloco imediatamente acima e os dois adjacentes a este.

2.2.3 Função Objetivo

A função objetivo deste problema é dada por:

$$\max \sum_{j \in V} c_j x_j$$

O lucro da operação é dado pela diferença entre o valor da soma do lucro de todos os blocos e o valor da capacidade do corte.

$$\begin{aligned} \text{lucro de operação} &= \sum_{j \in S} l_j - \sum_{i \in I} c_i \\ &= \sum_{j \in S} l_j + \sum_{j \in \bar{S}} l_j - (\sum_{j \in \bar{S}} l_j + \sum_{i \in I} c_i) \\ &= \sum_{j \in (S \cup \bar{S})} l_j - \text{capacidade do corte} \end{aligned}$$

2.3 Rede do Problema de Fluxo Máximo

Para a resolução do problema de fecho máximo, desenvolvemos um grafo auxiliar com dois novos vértices, representantes da fonte (s) e do terminal (t) deste, que representamos de forma arbitrária por 41 e 42 respectivamente.

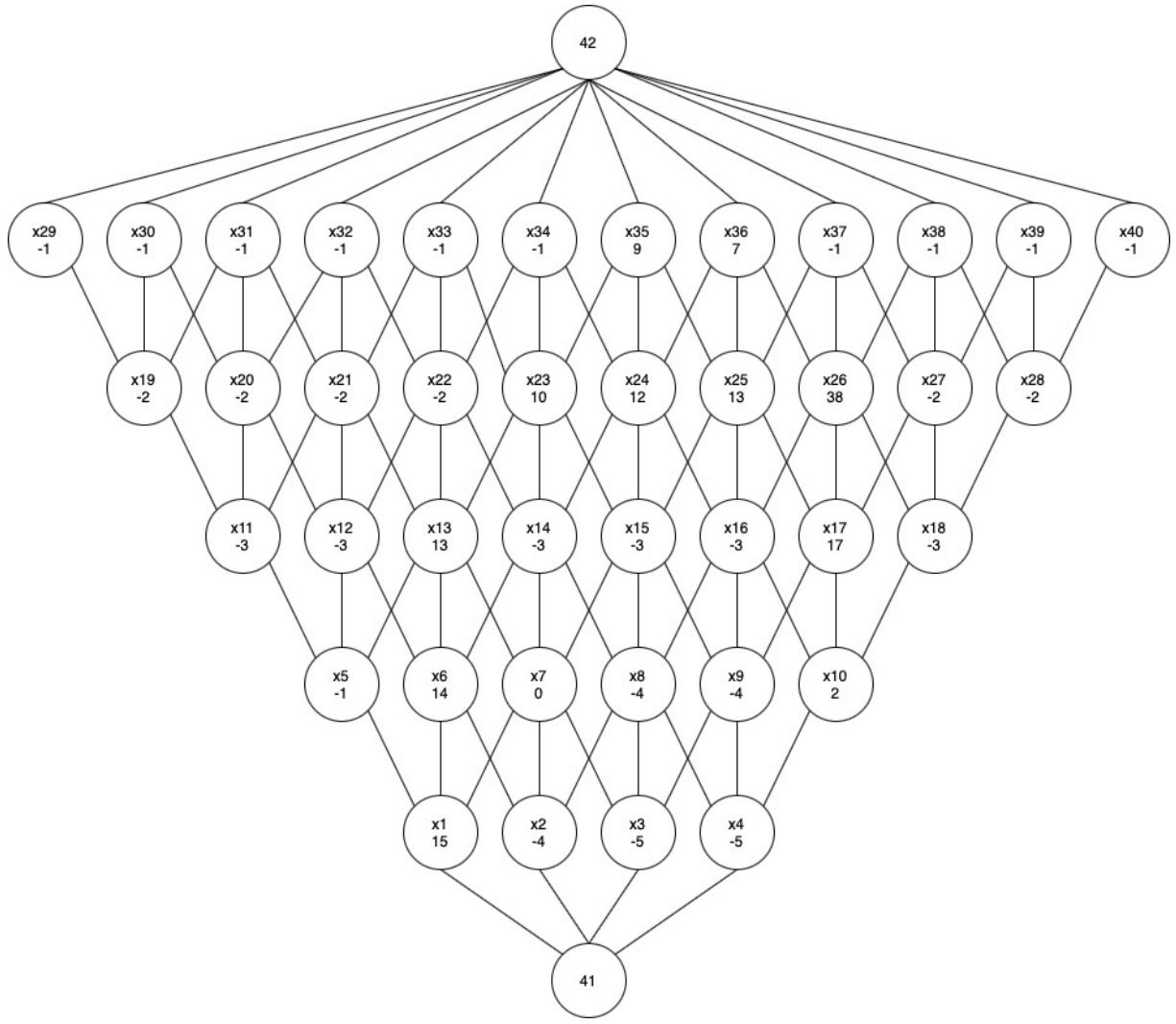


Figura 2.4: Grafo Proposto

Consideremos que:

$$V^+ = \{j \in V : c_j \geq 0\}$$

$$V^- = \{j \in V : c_j < 0\}$$

Para todos os vértices j com lucro (valor ≥ 0) haverá um arco (s, j) , $j \in V^+$. Por outro lado, para todos os vértices j com prejuízo (valor ≤ 0) haverá um arco (j, t) , $j \in V^-$. Também temos que, para todos os vértices com lucro j , haverá um arco (j, i) em que i são todos os vértices com prejuízo para os quais o vértice j pode contribuir, ou seja, incluídos no "cone de desbaste". Por fim, o fluxo máximo no grafo é dado pelo fluxo no arco (t, s) , um arco entre o terminal e a fonte, por onde se faz o retorno do fluxo que atravessa o grafo.

2.3.1 Input Relax4

Para o desenvolvimento do ficheiro de input, com base na formulação anterior, consideramos o seguinte:

42	6 22 0 1000	5 42 0 1	0
92	6 30 0 1000	8 42 0 4	0
41 1 0 15	6 31 0 1000	9 42 0 4	0
41 6 0 14	6 32 0 1000	11 42 0 3	0
41 7 0 0	6 33 0 1000	12 42 0 3	0
41 10 0 2	6 34 0 1000	14 42 0 3	0
41 13 0 13	10 16 0 1000	15 42 0 3	0
41 17 0 17	10 18 0 1000	16 42 0 3	0
41 23 0 10	10 27 0 1000	18 42 0 3	0
41 24 0 12	10 28 0 1000	19 42 0 2	0
41 25 0 13	10 34 0 1000	20 42 0 2	0
41 26 0 38	10 37 0 1000	21 42 0 2	0
41 35 0 9	10 38 0 1000	22 42 0 2	0
41 36 0 7	10 39 0 1000	27 42 0 2	0
1 5 0 1000	10 40 0 1000	28 42 0 2	0
1 11 0 1000	13 21 0 1000	29 42 0 1	0
1 12 0 1000	13 22 0 1000	30 42 0 1	0
1 14 0 1000	13 31 0 1000	31 42 0 1	0
1 15 0 1000	13 32 0 1000	32 42 0 1	0
1 19 0 1000	13 33 0 1000	33 42 0 1	0
1 20 0 1000	13 34 0 1000	34 42 0 1	0
1 21 0 1000	17 27 0 1000	37 42 0 1	0
1 22 0 1000	17 37 0 1000	38 42 0 1	0
1 29 0 1000	17 38 0 1000	39 42 0 1	0
1 30 0 1000	17 39 0 1000	40 42 0 1	0
1 31 0 1000	23 33 0 1000	42 41 -1 1000	0
1 32 0 1000	23 34 0 1000	0	0
1 33 0 1000	24 34 0 1000	0	0
1 34 0 1000	25 37 0 1000	0	0
1 37 0 1000	26 37 0 1000	0	0
6 12 0 1000	26 38 0 1000	0	0
6 14 0 1000	2 42 0 4	0	0
6 20 0 1000	3 42 0 5	0	0
6 21 0 1000	4 42 0 5	0	0

2.3.2 Output Relax4

```
NUMBER OF NODES = 42, NUMBER OF ARCS = 92
CONSTRUCT LINKED LISTS FOR THE PROBLEM
CALLING RELAX4 TO SOLVE THE PROBLEM
*****
TOTAL SOLUTION TIME = 0. SECS.
TIME IN INITIALIZATION = 0. SECS.

41 1 13.
41 6 6.
41 10 2.
41 13 6.
41 17 3.
41 23 1.
41 24 1.
41 26 2.
1 5 1.
1 11 3.
1 15 3.
1 19 2.
1 20 2.
1 29 1.
1 30 1.
6 12 3.
6 14 3.
10 16 2.
13 21 2.
13 22 2.
13 31 1.
13 32 1.
17 27 2.
17 39 1.
23 33 1.
24 34 1.
26 37 1.
26 38 1.
5 42 1.
11 42 3.
12 42 3.
14 42 3.
15 42 3.
16 42 2.
19 42 2.
20 42 2.
21 42 2.
22 42 2.
27 42 2.
29 42 1.
30 42 1.
31 42 1.
32 42 1.
33 42 1.
34 42 1.
37 42 1.
38 42 1.
39 42 1.
42 41 34.

OPTIMAL COST = -34.
NUMBER OF AUCTION/SHORTEST PATH ITERATIONS = 48
NUMBER OF ITERATIONS = 75
NUMBER OF MULTINODE ITERATIONS = 14
NUMBER OF MULTINODE ASCENT STEPS = 0
NUMBER OF REGULAR AUGMENTATIONS = 14
*****
```

Figura 2.5: Output Relax4

2.3.3 Interpretação da Solução Óptima Relax4

Atendendo tanto o input como o output desenvolvemos o seguinte grafo bipartido:

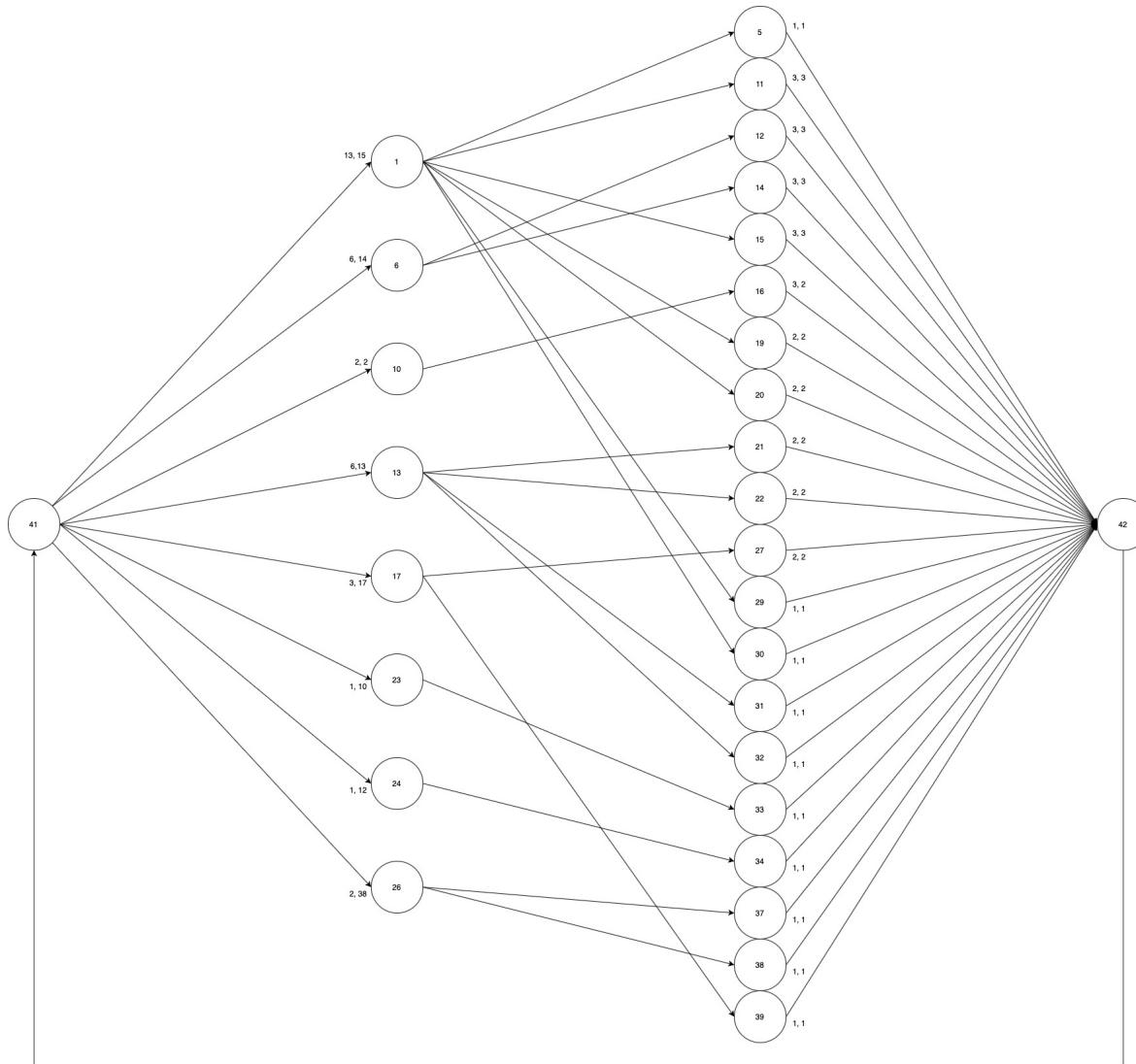


Figura 2.6: Grafo Bipartido

Com a solução ótima obtida podemos determinar quais os blocos a serem minerados e ver que o valor do corte mínimo é 34. Assim, ficamos com o seguinte plano de escavação:

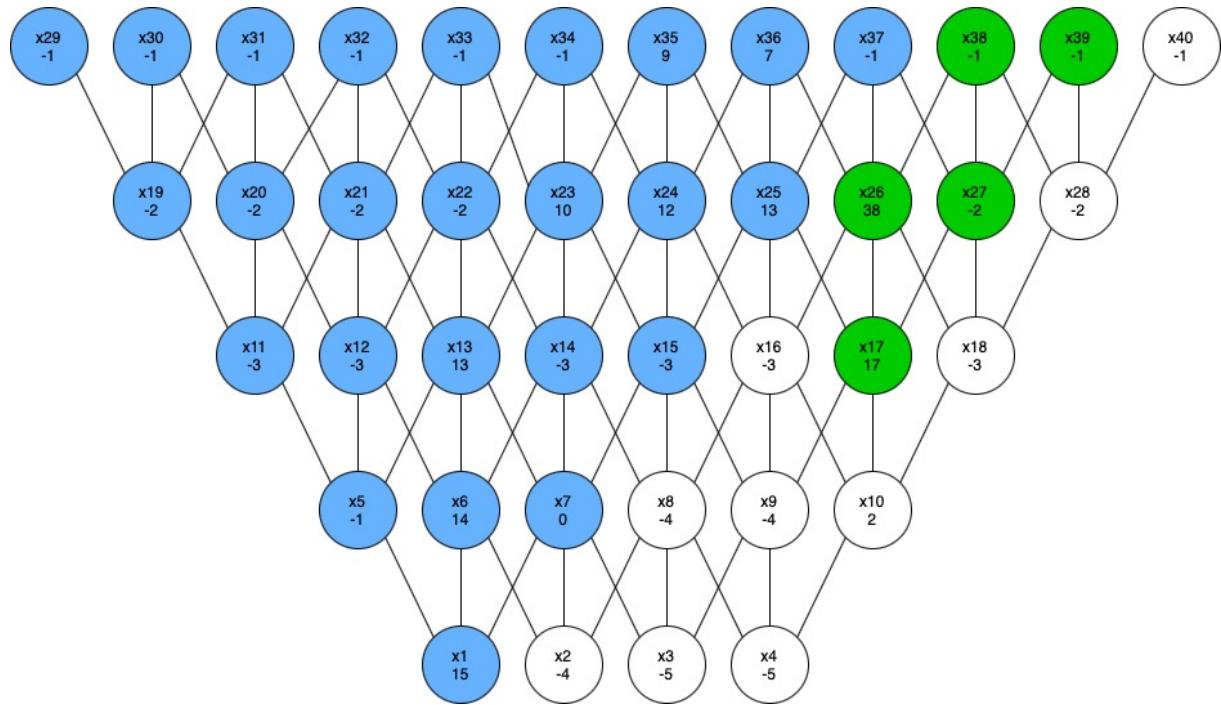


Figura 2.7: Grafo Solução

$$\text{Proveito} = 150$$

$$\text{Custo de operação} = 34$$

$$\text{Lucro} = 150 - 34 = 116$$

2.4 Validação do modelo

De forma a validar os resultados obtidos, desenvolvemos o modelo recorrendo à ferramenta LPSolve.

2.4.1 Variáveis de Decisão

Para alcançar o objectivo proposto, optamos por definir as seguintes variáveis de decisão:

$$\begin{aligned} x_j &\in \mathbb{N}: j \text{ corresponde ao bloco a extraer} \\ l_j &\in \mathbb{N}: \text{lucro/custo da extração do bloco } j \end{aligned}$$

2.4.2 Restrições

Para responder às necessidades propostas pelo enunciado, necessitamos de estipular as restrições correspondentes à extração de cada bloco. Para garantirmos que aquando da extração de um dado bloco, é necessário extraír os blocos (associados ao vale) do nível superior, elaboramos as seguintes restrições correspondentes a cada bloco (com excepção do primeiro nível):

$$\begin{array}{llll} x_1 \leq x_5; & x_8 \leq x_{14}; & x_{15} \leq x_{23}; & x_{22} \leq x_{32}; \\ x_1 \leq x_6; & x_8 \leq x_{15}; & x_{15} \leq x_{24}; & x_{22} \leq x_{33}; \\ x_1 \leq x_7; & x_8 \leq x_{16}; & x_{15} \leq x_{25}; & x_{22} \leq x_{34}; \\ \\ x_2 \leq x_6; & x_9 \leq x_{15}; & x_{16} \leq x_{24}; & x_{23} \leq x_{33}; \\ x_2 \leq x_7; & x_9 \leq x_{16}; & x_{16} \leq x_{25}; & x_{23} \leq x_{34}; \\ x_2 \leq x_8; & x_9 \leq x_{17}; & x_{16} \leq x_{26}; & x_{23} \leq x_{35}; \\ \\ x_3 \leq x_7; & x_{10} \leq x_{16}; & x_{17} \leq x_{25}; & x_{24} \leq x_{34}; \\ x_3 \leq x_8; & x_{10} \leq x_{17}; & x_{17} \leq x_{26}; & x_{24} \leq x_{35}; \\ x_3 \leq x_9; & x_{10} \leq x_{18}; & x_{17} \leq x_{27}; & x_{24} \leq x_{36}; \\ \\ x_4 \leq x_8; & x_{11} \leq x_{19}; & x_{18} \leq x_{26}; & x_{25} \leq x_{35}; \\ x_4 \leq x_9; & x_{11} \leq x_{20}; & x_{18} \leq x_{27}; & x_{25} \leq x_{36}; \\ x_4 \leq x_{10}; & x_{11} \leq x_{21}; & x_{18} \leq x_{28}; & x_{25} \leq x_{37}; \\ \\ x_5 \leq x_{11}; & x_{12} \leq x_{20}; & x_{19} \leq x_{29}; & x_{26} \leq x_{36}; \\ x_5 \leq x_{12}; & x_{12} \leq x_{21}; & x_{19} \leq x_{30}; & x_{26} \leq x_{37}; \\ x_5 \leq x_{13}; & x_{12} \leq x_{22}; & x_{19} \leq x_{31}; & x_{26} \leq x_{38}; \\ \\ x_6 \leq x_{12}; & x_{13} \leq x_{21}; & x_{20} \leq x_{30}; & x_{27} \leq x_{37}; \\ x_6 \leq x_{13}; & x_{13} \leq x_{22}; & x_{20} \leq x_{31}; & x_{27} \leq x_{38}; \\ x_6 \leq x_{14}; & x_{13} \leq x_{34}; & x_{20} \leq x_{32}; & x_{27} \leq x_{39}; \\ \\ x_7 \leq x_{13}; & x_{14} \leq x_{22}; & x_{21} \leq x_{31}; & x_{28} \leq x_{38}; \\ x_7 \leq x_{14}; & x_{14} \leq x_{23}; & x_{21} \leq x_{32}; & x_{28} \leq x_{39}; \\ x_7 \leq x_{15}; & x_{14} \leq x_{24}; & x_{21} \leq x_{33}; & x_{28} \leq x_{40}; \end{array}$$

2.4.3 Função Objetivo

Dado que o objectivo é maximizar o lucro, encontrando os blocos mais rentáveis a extrair, definimos a seguinte função objectivo, onde como coeficientes das variáveis de decisão temos o balanço monetário associado a cada um, isto é, a diferença entre o custo de extração (relacionado com o nível de profundidade) do bloco j , e o seu valor de retorno.

max: $15 x_1 - 4 x_2 - 5 x_3 - 5 x_4 - x_5 + 14 x_6 - 4 x_8 - 4 x_9 + 2 x_{10} - 3 x_{11} - 3 x_{12} + 13 x_{13} - 3 x_{14} - 3 x_{15} - 3 x_{16} + 17 x_{17} - 3 x_{18} - 2 x_{19} - 2 x_{20} - 2 x_{21} - 2 x_{22} + 10 x_{23} + 12 x_{24} + 13 x_{25} + 38 x_{26} - 2 x_{27} - 2 x_{28} - x_{29} - x_{30} - x_{31} - x_{32} - x_{33} - x_{34} + 9 x_{35} + 7 x_{36} - x_{37} - x_{38} - x_{39} - x_{40}$;

2.4.4 Output LPSolve

Variables	MILP Feasible	result
x1	1	1
x5	1	1
x6	1	1
x11	1	1
x12	1	1
x13	1	1
x14	1	1
x15	1	1
x17	1	1
x19	1	1
x20	1	1
x21	1	1
x22	1	1
x23	1	1
x24	1	1
x25	1	1
x26	1	1
x27	1	1
x29	1	1
x30	1	1
x31	1	1
x32	1	1
x33	1	1
x34	1	1
x35	1	1
x36	1	1
x37	1	1
x38	1	1
x39	1	1
x7	1	1

Figura 2.8: Output LPSolve

```
C:\Program Files (x86)\LPSolve IDE\lpsolveIDE.exe
Model name: 'LPSolver' - run #1
Objective: Maximize(R0)

SUBMITTED
Model size: 81 constraints,      40 variables,          162 non-zeros.
Sets:           0 GUB,             0 SOS.

Using DUAL simplex for phase 1 and PRIMAL simplex for phase 2.
The primal and dual simplex pricing strategy set to 'Devex'.

Relaxed solution           116 after      21 iter is B&B base.
Feasible solution          116 after      21 iter,        0 nodes (gap 0.0%)
Optimal solution            116 after      21 iter,        0 nodes (gap 0.0%).
Relative numeric accuracy ||*|| = 5.55112e-017

MEMO: lp_solve version 5.5.2.5 for 32 bit OS, with 64 bit REAL variables.
In the total iteration count 21, 0 (0.0%) were bound flips.
There were 1 refactorizations, 0 triggered by time and 1 by density.
... on average 21.0 major pivots per refactorization.
The largest [LUSOL v2.2.1.0] fact(B) had 165 NZ entries, 1.0x largest basis.
The maximum B&B level was 1, 0.0x MIP order, 1 at the optimal solution.
The constraint matrix re-normal is 1, with a dynamic range of 1.
Time to load data was 0.002 seconds, presolve used 0.003 seconds,
... 0.007 seconds in simplex solver, in total 0.012 seconds.
```

Figura 2.9: Segmento do output LPSolve

2.4.5 Interpretação da Solução Óptima LPSolve

Analisando o resultado obtido, vemos que a solução corresponde a minerar as variáveis de decisão com valor correspondente a 1, alcançando o proveito de 116 unidades.

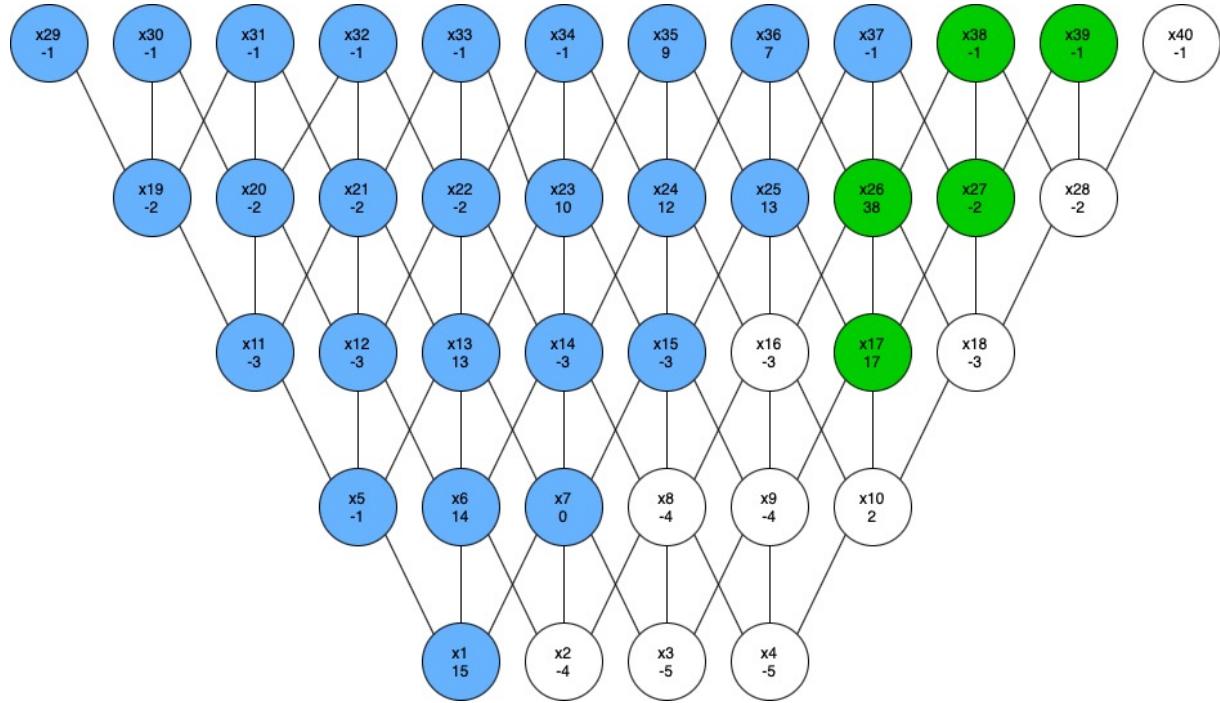


Figura 2.10: Grafo Solução LPSolve

De facto, podemos verificar que a solução vai de encontro às necessidades propostas, bem como respeita as regras de extracção propostas pelo enunciado. Como tal, podemos considerar esta como uma solução admissível, o que conseguimos verificar pela representação gráfica da solução quando aplicada ao grafo relativo ao problema, sendo esta idealmente uma solução ótima para o mesmo. Como esta solução coincide com a solução obtida pelo Relax4, podemos concluir que esta modelação serve de validação para o modelo anterior.

3. Conclusão

Para a resolução deste problema de investigação operacional, começamos por modular o problema, como um problema de programação linear, para o poder tratar e resolver com auxílio de software com tal propósito.

Em particular, utilizamos o Relax4, como nos era proposto, na tentativa de obter uma solução ótima para este problema. Sendo a nossa primeira experiência com o mesmo, permitiu-nos aprender o seu uso, como o aplicar para resolver este tipo de problemas, e como interpretar os seus resultados.

Utilizamos também novamente o LPSolve para obter uma solução ao problema, de modo a poder comparar e validar as soluções obtidas, sendo que o uso deste software era-nos já familiar do trabalho prático anterior.

Este trabalho introduziu-nos a um novo *solver* para problemas de programação linear, e permitiu-nos melhorar as nossas capacidades de traduzir problemas concretos em modelações apropriadas que permitam a sua resolução, de um modo mais fácil e perceptível.