



Universidade do Minho
Escola de Engenharia

MODELOS DETERMINÍSTICO DE INVESTIGAÇÃO OPERACIONAL

RELATÓRIO DE PROJETO 1

DRONE - NOVEMBRO 2020



Luís Miguel Pinto A89506

Ana Luísa Carneiro A89533



Pedro Almeida Fernandes A89574

Ana Rita Peixoto A89612



ÍNDICE

INTRODUÇÃO	2
REMOÇÃO DE ARESTAS – C, E	2
FORMULAÇÃO DO PROBLEMA	3
DESCRIÇÃO DO PROBLEMA E OBJETIVO	3
ESCOLHA DE VARIÁVEIS DE DECISÃO	4
RESTRIÇÕES E FUNÇÃO OBJETIVO	4
MODELO DO PROBLEMA	5
VARIÁVEIS DE DECISÃO	5
PARAMETROS	5
FUNÇÃO OBJETIVO	5
RESTRIÇÕES	6
INTERPRETAÇÃO DA SOLUÇÃO ÓTIMA	7
SOLUÇÃO ÓTIMA	7
PERCURSO	7
DISTÂNCIA TOTAL PERCORRIDA	7
VALIDAÇÃO DO MODELO	8
TESTES DE VALIDAÇÃO E ADMISSIBILIDADE DA SOLUÇÃO	8
ADEQUAÇÃO AO SISTEMA REAL	9
CONCLUSÃO	10

INTRODUÇÃO

No trabalho prático 1 é proposto a formulação e modelação de um problema que consiste em permitir que um drone consiga inspecionar todas as linhas de alta tensão, iniciando e finalizando o seu percurso no ponto estrela. O objetivo deste projeto visa encontrar um percurso ótimo que permita ao drone realizar a sua função de forma a que a distância percorrida seja o mínimo possível.

O desenvolvimento deste trabalho focou-se em diversos conceitos que foram a base da solução apresentada, tais como: grafo euleriano, circuito euleriano, duplicação de arestas e emparelhamento de vértices.

REMOÇÃO DE ARESTAS – C, E

Dado o contexto do problema, e de modo a personalizar o caminho efetuado pelo drone, foi proposto retirar determinadas arestas tendo em conta o maior número de inscrição do aluno do grupo. No nosso caso, o número selecionado foi o 89612. Assim, foram removidas as arestas C (8-13) e E (5-13), originando o seguinte grafo:

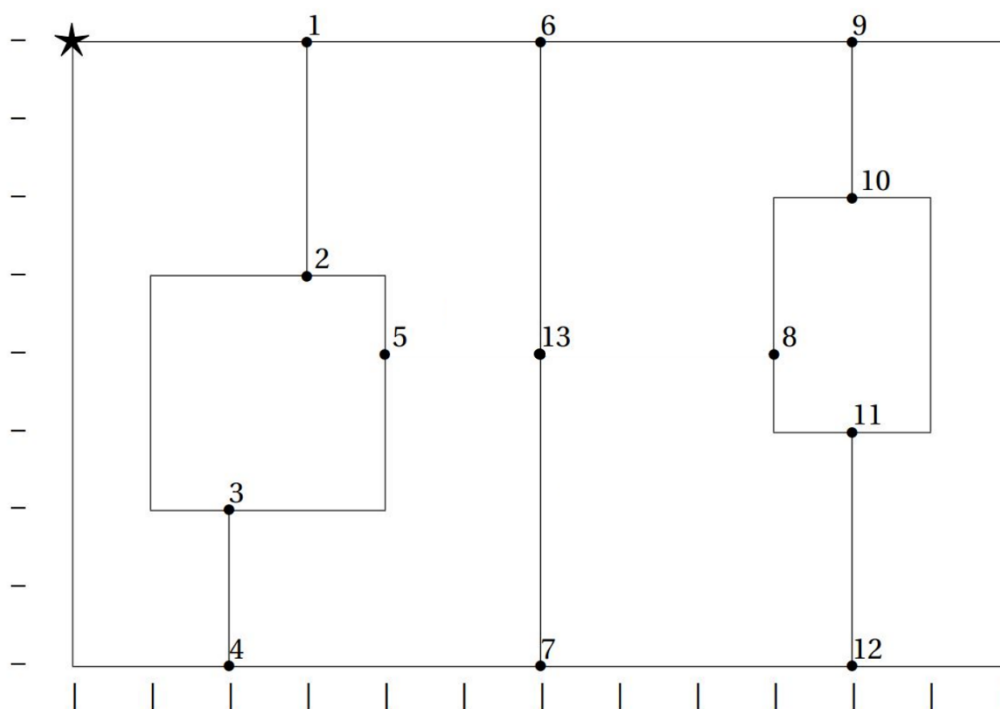


Figura 1 : Rede obtida a partir de 89612

FORMULAÇÃO DO PROBLEMA

DESCRIÇÃO DO PROBLEMA E OBJETIVO

O projeto I tem como objetivo determinar o percurso, que começa e acaba no mesmo vértice, em que todas as arestas da rede são percorridas pelo menos uma vez, minimizando a distância total percorrida.

Neste problema pretende-se que um drone realize um percurso de menor custo em que seja possível inspecionar todas as linhas de alta tensão (arestas da figura 1), começando e acabando no ponto estrela. Para realizar essa função, o drone pode percorrer pelo menos uma vez cada aresta ou então reposicionar-se pelo ar, por exemplo ir de 13 a 8 sem passar nas arestas.

Para a resolução deste problema e de forma a cumprir com o seu objetivo é necessário ter em consideração o conceito de circuito euleriano, isto é, um percurso que comece e termine no mesmo ponto, percorrendo todas as arestas apenas uma vez. Para que isso aconteça é necessário que todos os vértices desse circuito sejam de grau par (número de arestas que entram e/ou saem de um vértice tem de ser par). Caso contrário, não existe circuito euleriano.

Se a rede do drone fosse euleriana, qualquer circuito que percorresse todas as arestas ia constituir uma solução ótima, pois só se ia passar uma vez em cada aresta. Contudo, como a rede da figura 1 não é euleriana (por exemplo vértice 1 tem grau ímpar) então é necessário que o drone repita arestas de forma a cumprir com o proposto. Podemos, assim, concluir que a existência de um circuito de Euler é condição suficiente para a obtenção de uma solução ótima.

Através da rede (figura 1) e das distâncias euclidianas (figura 2), formulou-se um problema que visa selecionar quais as arestas a percorrer mais do que uma vez, atendendo ao seu custo. Essa seleção, em contexto teórico, é vista como duplicação de arestas, ou seja, na transformação da rede da figura 1 num circuito de Euler, cada “nova” aresta a adicionar a essa rede é vista como repetição de arestas ou reposicionamento do drone pelo ar no circuito original. Assim, após essa duplicação, a rede resultante terá todos os vértices com grau par e consequentemente conseguimos obter uma solução ótima.

		x	3	3	2	2	4	6	6	9	10	10	10	10	6
		y	8	5	2	0	4	8	0	4	8	6	3	0	4
			1	2	3	4	5	6	7	8	9	10	11	12	13
3	8	1	0,00	3,00	6,08	8,06	4,12	3,00	8,54	7,21	7,00	7,28	8,60	10,63	5,00
3	5	2	3,00	0,00	3,16	5,10	1,41	4,24	5,83	6,08	7,62	7,07	7,28	8,60	3,16
2	2	3	6,08	3,16	0,00	2,00	2,83	7,21	4,47	7,28	10,00	8,94	8,06	8,25	4,47
2	0	4	8,06	5,10	2,00	0,00	4,47	8,94	4,00	8,06	11,31	10,00	8,54	8,00	5,66
4	4	5	4,12	1,41	2,83	4,47	0,00	4,47	4,47	5,00	7,21	6,32	6,08	7,21	2,00
6	8	6	3,00	4,24	7,21	8,94	4,47	0,00	8,00	5,00	4,00	4,47	6,40	8,94	4,00
6	0	7	8,54	5,83	4,47	4,00	4,47	8,00	0,00	5,00	8,94	7,21	5,00	4,00	4,00
9	4	8	7,21	6,08	7,28	8,06	5,00	5,00	5,00	0,00	4,12	2,24	1,41	4,12	3,00
10	8	9	7,00	7,62	10,00	11,31	7,21	4,00	8,94	4,12	0,00	2,00	5,00	8,00	5,66
10	6	10	7,28	7,07	8,94	10,00	6,32	4,47	7,21	2,24	2,00	0,00	3,00	6,00	4,47
10	3	11	8,60	7,28	8,06	8,54	6,08	6,40	5,00	1,41	5,00	3,00	0,00	3,00	4,12
10	0	12	10,63	8,60	8,25	8,00	7,21	8,94	4,00	4,12	8,00	6,00	3,00	0,00	5,66
6	4	13	5,00	3,16	4,47	5,66	2,00	4,00	4,00	3,00	5,66	4,47	4,12	5,66	0,00

Figura 2: Distâncias Euclidianas

ESCOLHA DE VARIÁVEIS DE DECISÃO

As incógnitas do problema devem corresponder aos vértices que devemos emparelhar de forma a implementar um circuito Euleriano. Assim, cada variável $X_{i,j}$ representa uma possível aresta a adicionar à rede, sendo i o vértice de menor cardinalidade (origem) e j o de maior (destino). A escolha de variáveis binárias refletiu-se na necessidade de sinalizar a adição ou não de uma nova aresta.

Existem no total 10 vértices de grau ímpar (1,2,3,4,6,7,9,10,11,12) que podem ser representados como um grafo completo K_{10} com 45 ligações entre si (figura 3), ou seja, será necessário implementar 45 variáveis de decisão. Devido ao número de vértices de grau ímpar no nosso problema, serão necessários apenas 5 emparelhamentos para que esses vértices assumam grau par.

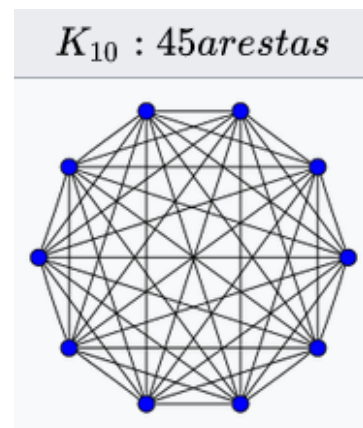


Figura 3: Grafo K_{10}

RESTRIÇÕES E FUNÇÃO OBJETIVO

As restrições devem ser implementadas de forma a que cada vértice seja emparelhado com apenas um outro vértice. Assim, assegura-se que todos os vértices têm grau par e que o número de emparelhamentos é o menor possível, de forma a atingir a solução ótima. Para além disso, o emparelhamento escolhido também deve ter em consideração o custo associado à duplicação de cada aresta. Para isso, é necessário que a função objetivo traduza o relacionamento entre uma possível aresta a duplicar e o seu custo, permitindo minimizar a distância percorrida pelo drone. Consequentemente, através dos emparelhamentos de custo mínimo conseguiremos deduzir qual o caminho a efetuar e a distância total percorrida.

MODELO DO PROBLEMA

VARIAVEIS DE DECISAO

As variáveis de decisão, tal como foi referido anteriormente, denotam a possibilidade de adicionar uma aresta. Assim, o método selecionado para denotar esta possibilidade foi o uso de variáveis binárias (figura 4). Deste modo, caso a aresta seja selecionada, o valor da variável correspondente será igual a 1. Caso contrário, obterá o valor 0.

$$X_{i,j} = \begin{cases} 1, & \text{caso seja duplicada a aresta } X_{i,j} \\ 0, & \text{caso não seja duplicada a aresta } X_{i,j} \end{cases}$$

$X_{i,j}$: aresta que une os vértices i e j , $i, j \in \{1,2,3,4,6,7,9,10,11,12\} \wedge i < j$

```
/* Variaveis de Decisao*/  
bin  
x1_2 x1_3 x1_4 x1_6 x1_7 x1_9 x1_10 x1_11 x1_12  
x2_3 x2_4 x2_6 x2_7 x2_9 x2_10 x2_11 x2_12  
x3_4 x3_6 x3_7 x3_9 x3_10 x3_11 x3_12  
x4_6 x4_7 x4_9 x4_10 x4_11 x4_12  
x6_7 x6_9 x6_10 x6_11 x6_12  
x7_9 x7_10 x7_11 x7_12  
x9_10 x9_11 x9_12  
x10_11 x10_12  
x11_12;
```

Figura 4: Variáveis usadas LP Solve

PARAMETROS

Os parâmetros do problema, ou seja, os dados do sistema que não podem ser alterados são: a tabela que contém as distâncias euclidianas, $C_{i,j}$, entre todos os vértices (figura 2) e a rede que o drone tem de percorrer (figura 1), que foi obtida através do maior número de aluno pertencente ao grupo.

$C_{i,j}$: custo associado à aresta que une os vértices i e j

FUNCAO OBJETIVO

A função objetivo, tal como foi supramencionado, retrata um problema de **minimização** representado através do emparelhamento de custo mínimo. Assim, será necessário relacionar cada um dos emparelhamentos (variáveis de decisão) com o seu custo, $C_{i,j}$, (figura 2), de forma a que esse emparelhamento tenham o menor custo possível (figura 5).

```
/* Objective function */  
min: 3 x1_2 + 6.08 x1_3 + 8.06 x1_4 + 3 x1_6 + 8.54 x1_7 + 7 x1_9 + 7.28 x1_10 + 8.60 x1_11 + 10.63 x1_12  
+ 3.16 x2_3 + 5.10 x2_4 + 4.24 x2_6 + 5.83 x2_7 + 7.62 x2_9 + 7.07 x2_10 + 7.28 x2_11 + 8.60 x2_12  
+ 2 x3_4 + 7.21 x3_6 + 4.47 x3_7 + 10 x3_9 + 8.94 x3_10 + 8.06 x3_11 + 8.25 x3_12  
+ 8.94 x4_6 + 4 x4_7 + 11.31 x4_9 + 10 x4_10 + 8.54 x4_11 + 8 x4_12  
+ 8 x6_7 + 4 x6_9 + 4.47 x6_10 + 6.40 x6_11 + 8.94 x6_12  
+ 8.94 x7_9 + 7.21 x7_10 + 5 x7_11 + 4 x7_12  
+ 2 x9_10 + 5 x9_11 + 8 x9_12  
+ 3 x10_11 + 6 x10_12  
+ 3 x11_12;
```

Figura 5: Função Objetivo LP Solve

RESTRIÇÕES

As restrições deste problema devem ser implementadas de forma a que cada vértice seja só emparelhado com um outro vértice. De forma a implementar esta condição, é preciso relembrar que, por exemplo, se o vértice 1 for emparelhando com o vértice 2 ($X_{1_2} = 1$) então os restantes emparelhamentos (variáveis de decisão) relativos aos vértices 1 e 2 tem de ser iguais a 0 ($X_{i_j} = 0$, com $i = 1, 2$ e $j = 3, 4, 6, 7, 9, 10, 11, 12$). Assim, por exemplo, a soma das variáveis de decisão para o vértice 1 tem de ser igual a 1, pois só é possível que o vértice 1 tenha um emparelhamento. Analogamente, esta condição deve ser aplicada aos restantes vértices de grau ímpar (figura 6).

```
/* Variable bounds */
// Cada vertice pode ser apenas emparelhado com um outro vertice

x1_2 + x1_3 + x1_4 + x1_6 + x1_7 + x1_9 + x1_10 + x1_11 + x1_12 = 1; // vertice 1
x1_2 + x2_3 + x2_4 + x2_6 + x2_7 + x2_9 + x2_10 + x2_11 + x2_12 = 1; // vertice 2
x1_3 + x2_3 + x3_4 + x3_6 + x3_7 + x3_9 + x3_10 + x3_11 + x3_12 = 1; // vertice 3
x1_4 + x2_4 + x3_4 + x4_6 + x4_7 + x4_9 + x4_10 + x4_11 + x4_12 = 1; // vertice 4
x1_6 + x2_6 + x3_6 + x4_6 + x6_7 + x6_9 + x6_10 + x6_11 + x6_12 = 1; // vertice 6
x1_7 + x2_7 + x3_7 + x4_7 + x6_7 + x7_9 + x7_10 + x7_11 + x7_12 = 1; // vertice 7
x1_9 + x2_9 + x3_9 + x4_9 + x6_9 + x7_9 + x9_10 + x9_11 + x9_12 = 1; // vertice 9
x1_10 + x2_10 + x3_10 + x4_10 + x6_10 + x7_10 + x9_10 + x10_11 + x10_12 = 1; // vertice 10
x1_11 + x2_11 + x3_11 + x4_11 + x6_11 + x7_11 + x9_11 + x10_11 + x11_12 = 1; // vertice 11
x1_12 + x2_12 + x3_12 + x4_12 + x6_12 + x7_12 + x9_12 + x10_12 + x11_12 = 1; // vertice 12
```

Figura 6: Restrições usadas LP Solve

INTERPRETAÇÃO DA SOLUÇÃO ÓTIMA

SOLUÇÃO ÓTIMA

Com o uso do LP Solve chegou-se á tabela (figura 7), onde estão representados todos os valores das variáveis de decisão de forma a obtermos a solução ótima. Assim, as variáveis iguais 1 serão as ligações cuja soma tem custo mínimo. Por outro lado, as restantes variáveis iguais a 0 não fazem parte da solução ótima.

Com base nesta tabela, obteve-se uma rede (figura 8) em que o drone consegue percorrer todas linhas de tensão (arestas a preto), tendo em certas circunstâncias de se reposicionar (arestas a vermelho - representadas pelas variáveis a 1 na tabela da figura 7).

Variables	MILP ...	MILP ...	result
	16	15,16	15,16
X1_6	0	1	1
X2_3	0	1	1
X4_7	0	1	1
X9_10	0	1	1
X11_12	0	1	1
X1_2	1	0	0
X1_3	0	0	0
X1_4	0	0	0
X1_7	0	0	0
X1_9	0	0	0
X1_10	0	0	0
X1_11	0	0	0
X1_12	0	0	0
X2_4	0	0	0

Figura 8: Solução Ótima LP Solve

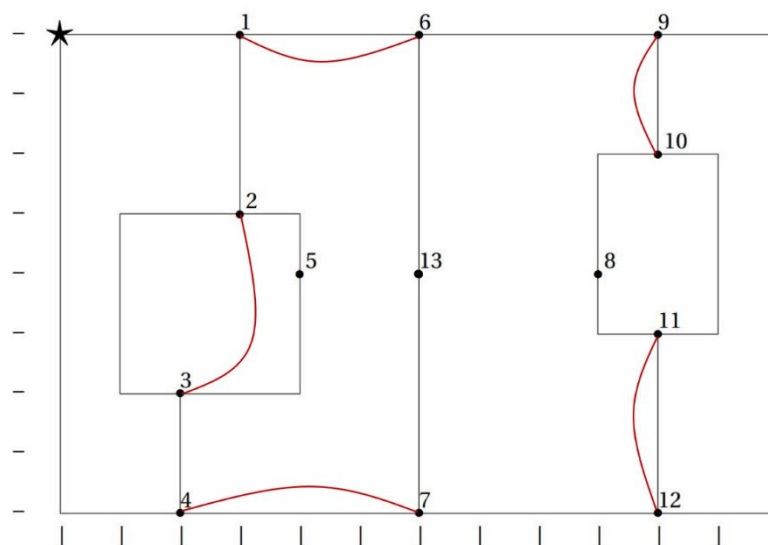


Figura 7: Rede ótima

PERCURSO

Dada a solução ótima obtida, é possível efetuar vários percursos que percorram todas as arestas. A título de exemplo, foi selecionado o percurso da figura 9.

* -> 1 -> 2 -> 5 -> 3 -> 2 -> 3 -> 4 -> 7 -> 13 -> 6 -> 1 -> 6 -> 9 -> 12 -> 11 -> 10 -> 9 -> 10 -> 8 -> 11 -> 12 -> 7 -> 4 -> *

Figura 9: Exemplo de percurso possível

DISTÂNCIA TOTAL PERCORRIDA

Através da observação da tabela da figura 7 conseguimos inferir o custo de percorrer as **arestas duplicadas**. Para obtermos a **distância total**, é necessário ter em consideração a distância percorrida nas linhas de **alta tensão** (calculado a partir de observação gráfica, 80) e o custo anteriormente referido. Deste modo, a distância total percorrida foi:

$$80 + 15,16 = 95,16$$

VALIDAÇÃO DO MODELO

De modo a validar o modelo, é importante focar a análise em 2 pontos: garantir que a solução obtida é admissível e correta e que é uma solução adequada ao sistema real.

TESTES DE VALIDAÇÃO E ADMISSIBILIDADE DA SOLUÇÃO

Uma forma de validar o resultado e consequentemente garantir que a solução ótima é admissível é através de testes. Um teste feito foi a substituição da solução ótima nas restrições e função objetivo, tal como está descrito na figura 10 e 11.

Vértices	1	2	3	4	6	7	9	10	11	12	Restrições
1	↳	0	0	0	1	0	0	0	0	0	R1 = 1
2	-	↳	1	0	0	0	0	0	0	0	R2 = 1
3	-	-	↳	0	0	0	0	0	0	0	R3 = 1
4	-	-	-	↳	0	1	0	0	0	0	R4 = 1
6	-	-	-	-	↳	0	0	0	0	0	R5 = 1
7	-	-	-	-	-	↳	0	0	0	0	R6 = 1
9	-	-	-	-	-	-	↳	1	0	0	R9 = 1
10	-	-	-	-	-	-	-	↳	0	0	R10 = 1
11	-	-	-	-	-	-	-	-	↳	1	R11 = 1
12	-	-	-	-	-	-	-	-	-	↳	R12 = 1

Figura 10: Confirmação das restrições do LP Solve

$$\text{min: } 3 X1_2 + 6.08 X1_3 + 8.06 X1_4 + 3 X1_6 + 8.54 X1_7 + 7 X1_9 + 7.28 X1_10 + 8.60 X1_11 + 10.63 X1_12$$

$$+ 3.16 X2_3 + 5.10 X2_4 + 4.24 X2_6 + 5.83 X2_7 + 7.62 X2_9 + 7.07 X2_10 + 7.28 X2_11 + 8.60 X2_12$$

$$+ 2 X3_4 + 7.21 X3_6 + 4.47 X3_7 + 10 X3_9 + 8.94 X3_10 + 8.06 X3_11 + 8.25 X3_12$$

$$+ 8.94 X4_6 + 4 X4_7 + 11.31 X4_9 + 10 X4_10 + 8.54 X4_11 + 8 X4_12$$

$$+ 8 X6_7 + 4 X6_9 + 4.47 X6_10 + 6.40 X6_11 + 8.94 X6_12$$

$$+ 8.94 X7_9 + 7.21 X7_10 + 5 X7_11 + 4 X7_12$$

$$+ 2 X9_10 + 5 X9_11 + 8 X9_12$$

$$+ 3 X10_11 + 6 X10_12$$

$$+ 3 X11_12;$$

$$3 * 1 (X1_6) + 3.16 * 1 (X2_3) + 4 * 1 (X4_7) + 2 * 1 (X9_10) + 3 * 1 (X11_12) = 15.16$$

Figura 11: Confirmação da função objetivo no LP Solve

Na figura 10 podemos ver que todas as restrições foram consideradas, pois para cada vértice só existe um emparelhamento com um outro vértice. Assim, a soma de todas as variáveis de decisão para cada vértice é igual a 1, tal como estava expresso na modelação. Para além disso, na figura 11 podemos ver que se substituímos os valores da solução ótima (sombreados) por 1 e os restantes por 0, conseguimos obter o custo mínimo de **15,16**, que foi o valor obtido na solução ótima. Por conseguinte, podemos garantir que a solução ótima é válida e admissível no modelo do problema.

ADEQUAÇÃO AO SISTEMA REAL

O outro ponto referido averigua se a solução é adequada à realidade, isto é, se é uma solução que faz sentido no contexto do problema. Por um lado, a solução enquadra-se porque permite que um drone realize um percurso contínuo e fluido. Por outro, ao observarmos atentamente a rede com as "novas" arestas (figura 7), verificamos que as arestas adicionadas (em tom vermelho) estão próximas umas das outras mostrando, em princípio, respeitar o critério da minimização da distância percorrida pelo drone.

CONCLUSÃO

Concluindo o 1º trabalho prático, consideramos que a realização deste projeto foi fundamental para melhor formular e modelar problemas otimização linear, nomeadamente de minimização.

Durante a realização deste trabalho surgiram algumas dificuldades. Apesar de termos consciência que o problema devia ser formulado com o intuito de “criar” um grafo euleriano, inicialmente tivemos dificuldade na implementação das variáveis de decisão e consequentemente nas restrições. Após uma análise mais aprofundada ao problema do “Carteiro Chines” e do estudo de problemas semelhantes, chegamos à conclusão de que a solução implementada foi a mais eficiente e viável para o problema em questão.

Numa primeira fase de implementação foram usadas 90 variáveis de decisão. Contudo, após uma análise mais detalhada concluímos que havia uma redundância pois, por exemplo, a variável X_{1_6} representaria o mesmo que a variável X_{6_1} . Assim, foi possível reduzir para metade o número de variáveis utilizadas, tornando a implementação mais simples sem perder nenhuma especificação.

Com base no que foi exposto, consideramos que as dificuldades foram ultrapassadas de forma eficaz e que por isso o balanço do resultado foi positivo.