Modelos Determinísticos de Investigação Operacional

8 de Janeiro de 2020

Grupo nr.	15
a83899	André Morais
a84577	José Pedro Silva
a85954	Luís Ribeiro
a84783	Pedro Rodrigues

Mestrado Integrado em Engenharia Informática
Universidade do Minho

Conteúdo

1 Introdução

O presente relatório irá abordar a elaboração do projeto realizado no âmbito da Unidade Curricular **Modelos Determinísticos de Investigação Operacional**. Este trabalho está dividido em 3 partes. Na **parte 0** baseia-se a construir um novo grafo, com algumas restrições. Na **parte 1** e **parte 2** foram nos proposto dois problemas a partir do grafo obtido anteriormente.

Para tal, foi-nos fornecido um gráfico (Figura 12) que inicialmente já era definido o caminho crítico que correspondia às actividades 6,7,4,2 e 3, com uma duração de 29 unidades, que é também o tempo mínimo necessário para a realização de todo o projeto.

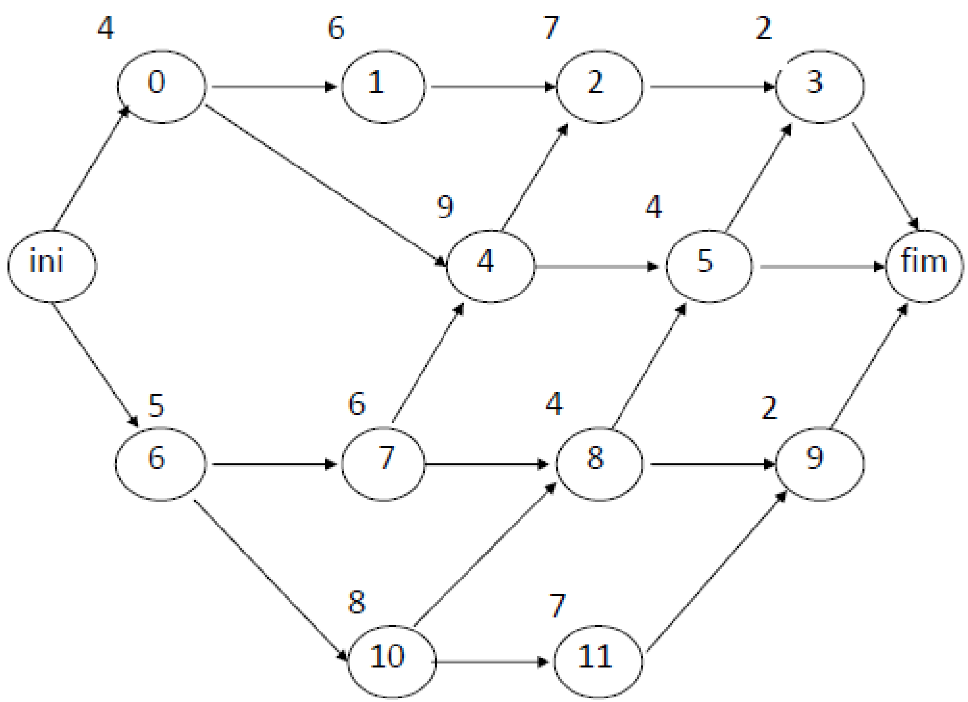

Figura 1: Gráfico Inicial

2 Questões

2.1 Parte 0

2.1.1 Questão 1

"Apresente a rede que representa o projecto depois de eliminar as actividades indicadas na secção "Determinação da lista de actividades", no final do texto, identificando os vértices da rede e os arcos e respectivos custos.".

Sendo ABCDE o maior número mecanográfico do grupo, analisando os mesmos concluímos que *a85954* é o maior. Através do conjunto de regras definidas no enunciado do trabalho prático completamos o gráfico, obtendo o seguinte gráfico (Figura 2).

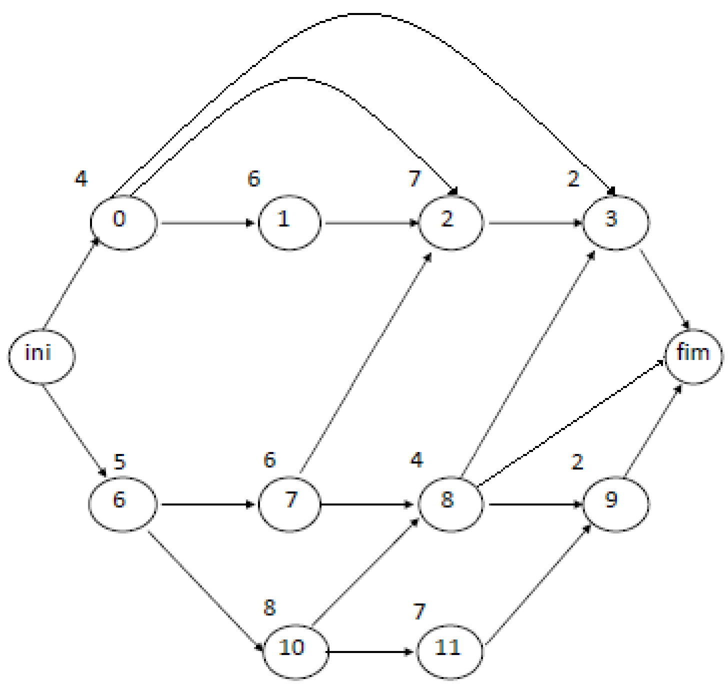

Figura 2: Grafo com direções

A rede anteriormente apresentada tem as seguintes características:

Vértices da Rede

- Só houve alteração nos vértices 4 e 5, sendo estes removidos.Por isso temos um conjunto de 12 vértices, que posteriormente vão nos ajudar os arcos deste grafo.

t_i, t_0, t_1, t_2, t_3, t_6, t_7, t_8, t_9, t_{10}, t_{11}, t_f

Arcos da Rede

- Seja t_{ij} o número de vezes que um arco é percorrido, onde i corresponde ao vértice de origem e j ao vértice de chegada. Isto é, o arco t_{01} corresponde ao número que o arco que liga o *vértice 0* ao *vértice 1* é percorrido.

Arcos com Duração 0 - t_{i0}, t_{i6}
Arcos com Duração 2 - t_{23} , t_{9f}
Arcos com Duração 3 - t_{3f}
Arcos com Duração 4 - t_{01}, t_{02}, t_{03}, t_{83}, t_{8f}, t_{89}
Arcos com Duração 5 - t_{67}, t_{610}
Arcos com Duração 6 - t_{12}, t_{72}, t_{78}
Arcos com Duração 7 - t_{119}
Arcos com Duração 8 - t_{108}, t_{1011}

Custos dos Arcos

- O custo de cada arco, ou melhor dizendo, a duração deste, é por consequente aprensentado no vértice. Considerando um exemplo aleatório, a atividade 10 tem duração de 8, isto é, todos os arcos t_{10j} vão ter um custo de 8, como é o caso de t_{1011} e t_{108} no nosso caso.

Restrições

- As restrições são baseada na soma das durações do caminho que fomos percorrendo, como podemos verificar na figura abaixo evidenciada.

  Sendo p_i a duração da atividade i, a variável t_i, é uma variável de decisão, que é igual ao instante de início da execução da tarefa i.Depois temos restrições disjuntivas: o instante do fim da execução de uma tarefa é anterior ao instante de início da outra, isto é

  $$t_i + p_i \leq \mathrm{t}_j$$

Função objetivo

- Como o objetivo visa calcular o caminho crítico, então é necessário minimizar a duração de t_f,obtendo então a seguinte função objetivo:

  $$\min = t_f$$

```
/* Objective function */
min:tf ;

/* Variable bounds */
arco_i0: t0 >= ti + 0 ;
arco_i6: t6 >= ti + 0 ;
arco_01: t1 >= t0 + 4 ;
arco_02: t2 >= t0 + 4 ;
arco_03: t3 >= t0 + 4 ;
arco_12: t2 >= t1 + 6 ;
arco_23: t3 >= t2 + 7 ;
arco_67: t7 >= t6 + 5 ;
arco_78: t8 >= t7 + 6 ;
arco_72: t2 >= t7 + 6 ;
arco_89: t9 >= t8 + 4 ;
arco_83: t3 >= t8 + 4 ;
arco_610: t10 >= t6 + 5 ;
arco_108: t8 >= t10 + 8 ;
arco_119: t9 >= t11 + 7 ;
arco_1011: t11 >= t10 + 8;
arco_3f: tf >= t3 + 2 ;
arco_9f: tf >= t9 + 2 ;
arco_8f: tf >= t8 + 4;
```

Figura 3: Gráfico com as restrições

2.1.2 Questão 2

Apresente o diagrama de Gantt (que resulta de resolver o modelo com as variáveis de decisão t_i ,$\forall_i$), e indique a duração do projecto.

O valor da variável de decisão t_i na solução óptima define o tempo de início de execução da actividade i , permitindo construir um plano de execução do projecto, designado por diagrama de Gantt. O Diagrama de Gantt do projecto em análise é apresentado na Figura 5, com uma solução ótima de 23

Variables	result
	22
tf	22
t0	0
ti	0
t6	0
t1	5
t2	11
t3	20
t7	5
t8	16
t9	20
t10	5
t11	13

Figura 4: Resultado LPSolve

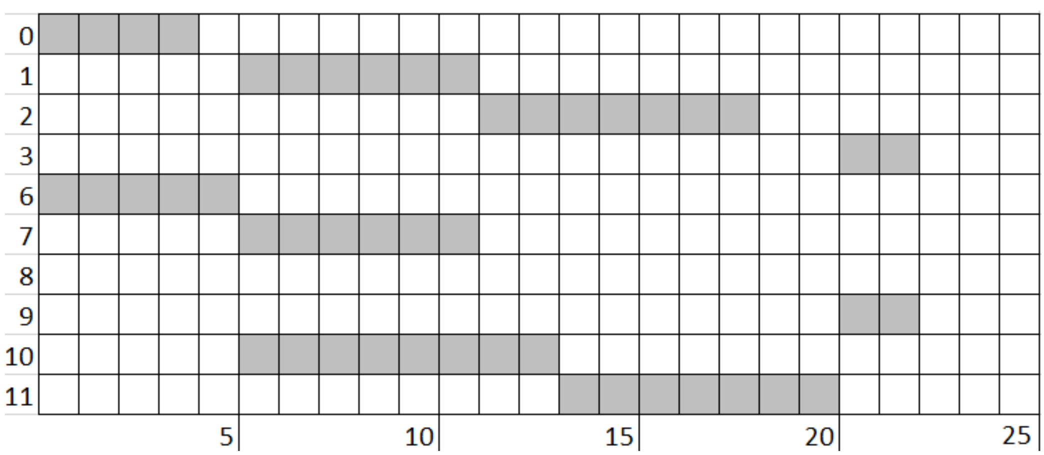

Figura 5: Diagrama de Gantt

2.2 Parte 1

Para este trabalho, identifique 3 actividades que, no diagrama de Gantt da PARTE 0, decorrem em paralelo; uma das actividades que seleccionar deve pertencer ao caminho crítico. Considere agora que existe apenas um equipamento para realizar as três actividades. O objectivo continua a ser realizar o projecto na menor duração possível.

2.2.1 Questão 1

Explique a forma que escolheu para formular este problema. Identifique claramente o significado das novas restrições e da função objectivo do novo modelo de programação linear inteira mista. Teça todos os comentários que considere adequados.

Após a observação do diagrama obtido na Parte 0, podemos concluir que apenas 3 atividades ocorrem em paralelo sendo estas as atividades 1, 7 e 10. Como no enunciado desceve que apenas uma atividade se pode realizar de cada vez, tivemos que acrescentar algumas restrições à nossa solução. Primeiramente criamos variáveis binárias, y_{ij}, que tomam o valor de 1 se a atividade i precede a atividade j ou 0 se a atividade j precede a atividade i
Restrições de não-simultaneidade

t_i + p_i $\leq$t$_j$ + M(1-y_{ij})
t_i + p_i $\leq$t$_j$ + My_{ij}

2.2.2 Questão 2

Apresente o ficheiro de input (cut-and-paste).

```
/* Objective function */
min:tf ;

/* Variable bounds */
arco_i0: t0 >= ti + 0 ;
arco_i6: t6 >= ti + 0 ;
arco_01: t1 >= t0 + 4 ;
arco_02: t2 >= t0 + 4 ;
arco_03: t3 >= t0 + 4 ;
arco_12: t2 >= t1 + 6 ;
arco_23: t3 >= t2 + 7 ;
arco_67: t7 >= t6 + 5 ;
arco_78: t8 >= t7 + 6 ;
arco_72: t2 >= t7 + 6 ;
arco_89: t9 >= t8 + 4 ;
arco_83: t3 >= t8 + 4 ;
arco_610: t10 >= t6 + 5 ;
arco_108: t8 >= t10 + 8 ;
arco_119: t9 >= t11 + 7 ;
arco_1011: t11 >= t10 + 8;
arco_3f: tf >= t3 + 2 ;
arco_9f: tf >= t9 + 2 ;
arco_8f: tf >= t8 + 4;

t1 + 6 <= t7 + 1000000 - 1000000y17;
t7 + 6 <= t1 + 1000000y17;

t1 + 6 <= t10 + 1000000 - 1000000y110;
t10 + 8 <= t1 + 1000000y110;

t7 + 6 <= t10 + 1000000 - 1000000y710;
t10 + 8 <= t7 + 1000000y710;

bin y17,y110,y710;
```

Figura 6: Resultado LPSolve

2.2.3 Questão 3

Apresente o ficheiro de output produzido pelo programa (cut-and-paste).

Variables	MILP ...	result
	33,00...	33,00...
tf	33,00...	33,00...
t0	0	0
ti	0	0
t6	0	0
t1	4	4
t2	24,00...	24,00...
t3	31,00...	31,00...
t7	10,00...	10,00...
t8	24,00...	24,00...
t9	31,00...	31,00...
t10	16,00...	16,00...
t11	24,00...	24,00...
y17	1	1
y110	1	1
y710	1	1

Figura 7: Resultado LPSolve

2.2.4 Questão 4

Apresente o plano de execução (diagrama de Gantt) do projecto. Verifique sumariamente que o plano obedece às novas restrições.

$t_1 + 6 \leq \mathrm{t}_7 + 1000000 - 1000000 y_{17}$
$t_7 + 6 \leq \mathrm{t}_1 + 1000000 y_{17}$

$t_1 + 6 \leq \mathrm{t}_{10} + 1000000 - 1000000 y_{110}$
$t_{10} + 8 \leq \mathrm{t}_1 + 1000000 y_{110}$

$t_7 + 6 \leq \mathrm{t}_{10} + 1000000 - 1000000 y_{710}$
$t_{10} + 8 \leq \mathrm{t}_7 + 1000000 y_{710}$

Observando a Figura 7, podemos retirar de lá os valores para validar as nossas restrições

$4 + 6 \leq 10 + 1000000 - 1000000 * 1 \Leftrightarrow 10 \leq 10$
$10 + 6 \leq 4 + 1000000 * 1 \Leftrightarrow 16 \leq 1000004$

$4 + 6 \leq 16 + 1000000 - 1000000 * 1 \Leftrightarrow 10 \leq 16$
$16 + 8 \leq 4 + 1000000 * 1 \Leftrightarrow 24 \leq 100004$

$10 + 6 \leq 16 + 1000000 - 1000000 * 1 \Leftrightarrow 16 \leq 16$
$16 + 8 \leq 10 + 1000000 * 1 \Leftrightarrow 24 \leq 1000010$

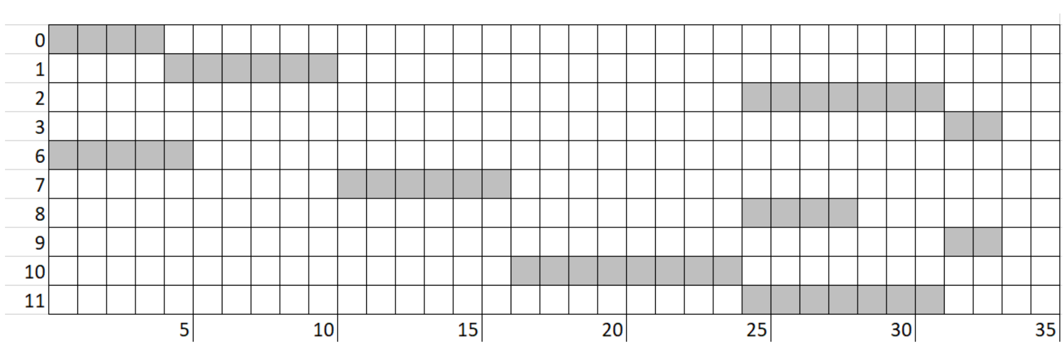

Figura 8: Diagrama de Gantt

2.3 Parte 2

Considere que é possível, aumentando os recursos aplicados e com custos suplementares, reduzir a duração de uma actividade, num caso em que o custo da redução é não-linear. Um modelo com funções não-lineares pode ser aproximado por um modelo em que cada uma dessas funções é aproximada por uma função contínua linear por partes. Neste trabalho, para aproximar a função não-linear, vamos usar uma função linear com 2 partes. Esses valores estão apresentados na seguinte Tabela (considere apenas aqueles que dizem respeito às actividades da lista do grupo):

Actividade	Custo Normal	c_1	Máx. red. a custo c_1	c_2	Máx. red. a custo c_2
0	400	200	0,5	100	0,5
1	1000	600	1	300	1
2	1400	1000	3	500	1
3	300	200	0,5	100	0,5
6	800	180	1	90	1
7	900	–	0	–	0
8	600	200	0,5	100	0,5
9	300	–	0	–	0
10	1600	1000	0,5	500	0,5
11	1400	600	1	300	1

Figura 9: Tabela de Custos

Pretende-se que o tempo de execução do projecto encontrado na PARTE 0 seja reduzidoem 3 U.T. .O objectivo do problema é decidir como devem ser reduzidas as durações das actividades, de modo a realizar o projecto na nova duração desejada, com um custo suplementar mínimo.

2.3.1 Questão 1

Explique a forma que escolheu para formular este problema. Identifique claramente o significado das novas restrições e da função objectivo do novo modelo de programação linear inteira mista. Teça todos os comentários que considere adequados. Primeiramente tivemos que definir a nossa função obje-

tivo que era minimizar o custo, e definir que t_f menor que 19 , pois tinhamos de subtrair 3 unidades ao nosso antigo valor que é pedido no enunciado. A função min é a soma de todos os custos, mas tendo em conta que as reduções variam e por isso tivemos de criar duas variáveis, r_i e z_i.

As primeiras restrições mantinham se iguais as da parte 0 e parte 1 , mas com o acrescente que agora iriamos ter de retirar o custo de r e o custo de z, como por exemplo:

$arco_{01}$: $t_1 \geq$t$_0$ - r_0 - z_0 + 4;

Seguidamente temos de impôr restrições nos r_i e nos z_i em que os r_i não podem ser maiores que o Máx.red. a custo c_1 (tendo que depois verificar que se o caminho está a ser usado, ou seja, verificar se y_{ij} é 0 ou 1). Nos z_i também não podem ser maior do que o Máx.red. a custo c_2 verificando se a atividade está a decorrer ou não.

2.3.2 Questão 2

Apresente o ficheiro de input (cut-and-paste).

```
/* Objective function */
min: 200r0 + 600r1 + 1000r2 + 200r3 + 180r6 + 0r7 + 200r8 + 0r9 + 1000r10 + 600r11 + 100z0 + 300z1 + 500z2 + 100z3 + 90z6 + 0z7 + 100z8 + 0z9 + 500z10 + 300z11;

/* Variable bounds */
tf <=19;

arco_i0: t0 >= ti + 0;
arco_i6: t6 >= ti + 0;
arco_01: t1 >= t0-r0-z0 + 4;
arco_02: t2 >= t0-r0-z0 + 4 ;
arco_03: t3 >= t0-r0-z0 + 4 ;
arco_12: t2 >= t1-r1-z1 + 6 ;
arco_23: t3 >= t2-r2-z2 + 7 ;
arco_67: t7 >= t6-r6-z6 + 5 ;
arco_78: t8 >= t7-r7-z7 + 6 ;
arco_72: t2 >= t7-r7-z7 + 6 ;
arco_89: t9 >= t8-r8-z8 + 4 ;
arco_83: t3 >= t8-r8-z8 + 4 ;
arco_610: t10 >= t6-r6-z6 + 5 ;
arco_108: t8 >= t10-r10-z10 + 8 ;
arco_119: t9 >= t11-r11-z11 + 7 ;
arco_1011: t11 >= t10-r10-z10 + 8;
arco_3f: tf >= t3-r3-z3 + 2 ;
arco_9f: tf >= t9-r9-z9 + 2 ;
arco_8f: tf >= t8-r8-z8 + 4;


r0<= 0.5;
r1<= 1;
r2<= 3;
r6<= 1;
r7<= 0;
r8<= 0.5;
r9<= 0;
r10<= 0.5;
r11<= 1;

0.5y0 <= r0;
1y1   <= r1;
3y2   <= r2;
0.5y3 <= r3;
1y6   <= r6;
0y7   <= r7;
0.5y8 <= r8;
0y9   <= r9;
0.5y10<= r10;
1y11  <= r11;


z0 <= 0.5*y0;
z1 <= 1*y1;
z2 <= 1*y2;
z3 <= 0.5*y3;
z6 <= 1*y6;
z7 <= 0*y7;
z8 <= 0.5*y8;
z9 <= 0*y9;
z10<= 0.5*y10;
z11<= 1*y11;

bin y0,y1,y2,y3,y6,y7,y8,y9,y10,y11;
```

Figura 10: Tabela de Custos

2.3.3 Questão 3

Apresente o ficheiro de output produzido pelo programa (cut-and-paste).

Variables	MILP ...	result
	870	870
r0	0	0
r1	0	0
r2	0	0
r3	0	0
r6	1	1
r7	0	0
r8	0	0
r9	0	0
r10	0	0
r11	1	1
z0	0	0
z1	0	0
z2	0	0
z3	0	0
z6	1	1
z7	0	0
z8	0	0
z9	0	0
z10	0	0
z11	0	0
tf	19	19
t0	0	0
ti	0	0
t6	0	0
t1	4	4
t2	10	10
t3	17	17
t7	4	4
t8	11	11
t9	17	17
t10	3	3
t11	11	11
y0	0	0
y1	0	0
y2	0	0
y3	0	0
y6	1	1
y8	0	0
y10	0	0
y11	1	1
y7	0	0
y9	0	0

Figura 11: Resultado LPSolve

2.3.4 Questão 4

Apresente o plano de execução (diagrama de Gantt) do projecto representando as actividades com as durações que elas têm após a respectiva redução.

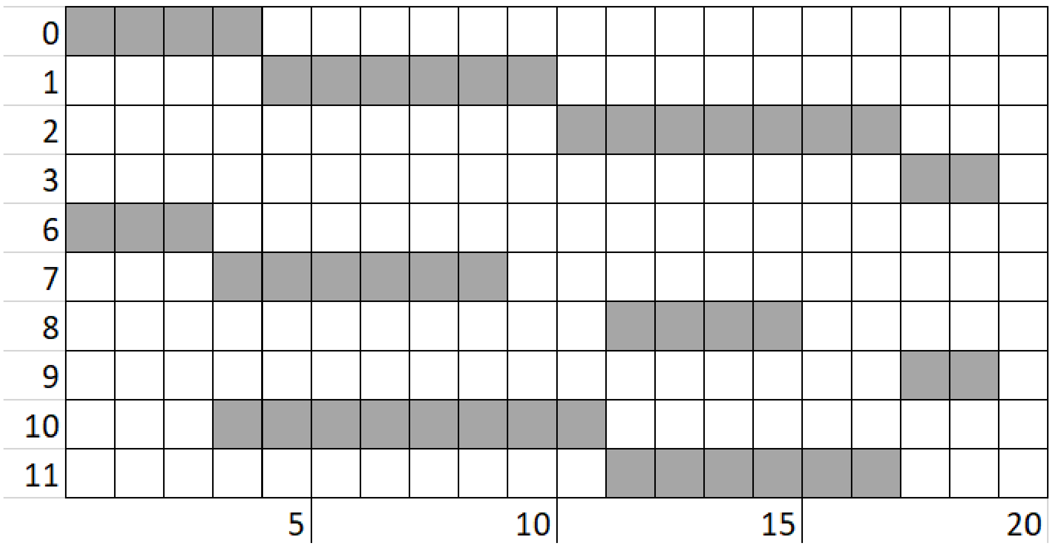

Figura 12: Diagrama de Gantt

2.3.5 Questão 5

Verifique que o custo da solução está correcto.

min $= 200r_0 + 600r_1 + 1000r_2 + 200r_3 + 180r_6 + 0r_7 + 200r_8 + 0r_9 + 1000r_{10} + 600r_{11} + 100z_0 + 300z_1 + 500z_2 + 100z_3 + 90z_6 + 0z_7 + 100z_8 + 0z_9 + 500z_{10} + 300z_{11}$

se substituirmos os valores obtidos no LPSolve, verificamos que o min = 870

min= 200 * 0 + 600 * 0 + 1000 * 0 + 200 * 0 + 180 * 1 + 0 + 200 * 0 + 0 + 1000 * 0 + 600 * 1 + 100 * 0 + 300 * 0 + 500 * 0 + 100 * 0 + 90 * 1 + 0 + 100 * 0 + 0 + 500 * 0 + 300 * 0

Fazendo as contas:

min= 180 + 600 + 90 $\Leftrightarrow min = 870$