Universidade do Minho
Mestrado Integrado em Engenharia Informática
3ºano - 1º Semestre

Modelos Determinísticos de Investigação Operacional

Relatório sobre o Trabalho 3

Grupo de trabalho prático experimental 34




a83732 – Gonçalo Rodrigues Pinto
a84197 – João Pedro Araújo Parente
a84829 – José Nuno Martins da Costa
a85059 – Diogo Paulo Lopes de Vasconcelos

8 de Janeiro de 2020

Conteúdo

Lista de Figuras

1 Introdução

No 1º semestre do 3º ano do Curso de Engenharia Informática da Universidade do Minho, existe uma Unidade Curricular denominada por Modelos Determinísticos de Investigação Operacional, que tem como objectivo ajudar os estudantes a desenvolver a sua capacidade de resolução de problemas com ênfase em problemas de engenharia de sistemas, dar a conhecer as técnicas e os métodos de Investigação Operacional, e consequentemente permitir que os alunos sejam capazes de aplicar estas técnicas e métodos na resolução de instâncias de problemas de pequena dimensão. Tem também como meta desenvolver a capacidade dos estudantes na análise de sistemas complexos, de criar modelos para os descrever, de obter soluções para esses modelos utilizando programas computacionais adequados, de validar os modelos obtidos, de interpretar as soluções obtidas, de elaborar recomendações para o sistema em análise e por fim, permitir que os alunos sejam capazes de compreender a importância da avaliação das soluções e de realizar análises de sensibilidade.

O presente trabalho pretende-se estudar o método do caminho mais crítico que constitui uma ferramenta muito importante em gestão de projectos O método do caminho crítico é aplicado a projectos que podem ser decompostos num conjunto de actividades, que se considera terem durações determinísticas, entre as quais existem relações de precedência. As restrições de precedência traduzem o facto de o instante em que se pode dar início a uma dada actividade ter de ser posterior aos instantes em que terminam as actividades que lhe são precedentes.

2 Projecto

No método do caminho crítico, a rede que representa o projecto pode ser representada de duas formas alternativas: uma, em que as actividades do projecto são representadas por arcos do grafo, e a outra, em que são representadas por nós. Considerou-se a segunda representação. O projecto que se baseia este trabalho é o abaixo apresentado com as actividades e as relações de precedência a seguir indicadas.

Actividade	Duração	Precedências
0	4	–
1	6	0
2	7	1,4
3	2	2,5
4	9	0,7
5	4	4,8
6	5	–
7	6	6
8	4	7,10
9	2	8,11
10	8	6
11	7	10

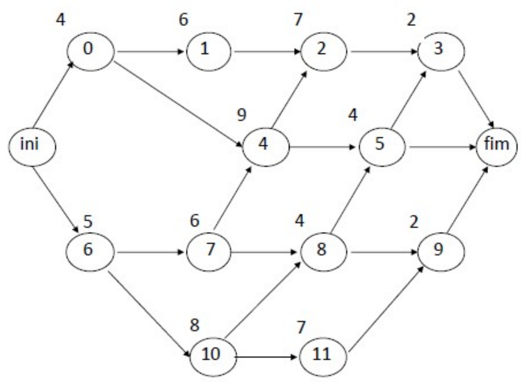

Figura 1: Actividades e as relações de precedência.

No problema em análise, o caminho crítico corresponde às actividades 6, 7, 4, 2 e 3, com uma duração de 29 unidades de tempo, que é também o menor tempo necessário para completar a execução de todo o projecto.

O problema a solucionar, é determinar os instantes em que devem ser começados um conjunto de actividades, sendo que estas podem ser apenas iniciadas se as suas actividades de que dependem já tiverem sido terminadas. A solução óptima é aquela que minimiza o tempo final, sendo o instante em que todas as actividades já se encontram finalizadas.

3 Parte 0 - Apresentação do projecto

3.1 Determinação da lista de actividades

Conforme o enunciado do trabalho considerou-se ABCDE o número de inscrição do aluno do grupo com maior número de inscrição, no nosso grupo esse aluno é o Diogo Vasconcelos com o número 85059. Tal como foi solicitado removido da lista de actividades as actividades D e E, o que equivale à actividade 5 e 9, respectivamente. As precedências passam a ser estabelecidas da seguinte forma os sucessores da actividade D passam a ter como novas precedências os antecessores da actividade D e mesmo para E. Originado o novo grafo abaixo apresentado.

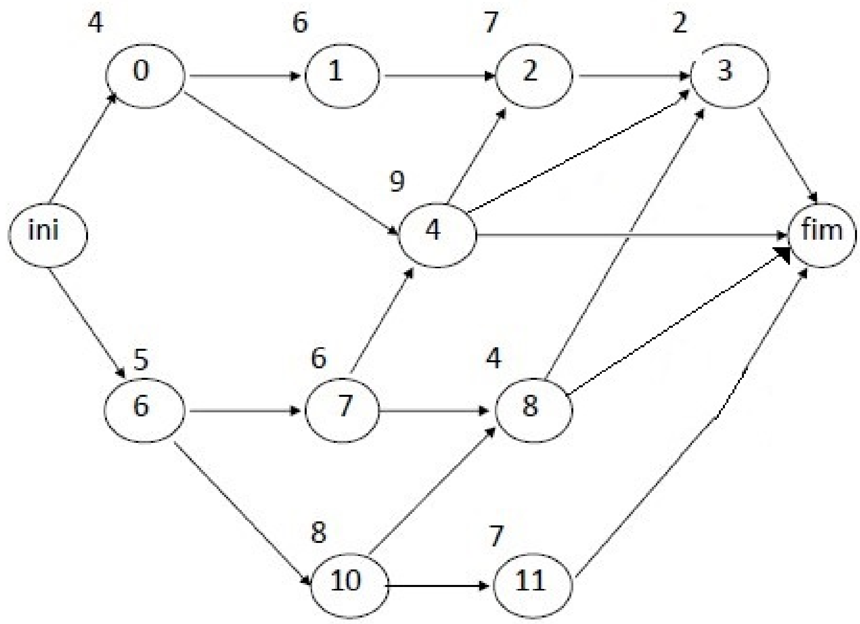

Figura 2: Novo grafo após determinação da lista de actividades.

3.2 Diagrama de Gantt e duração do projecto

Considerando cada variável de decisão ti, $\forall i$, representa o tempo de início da actividade i. As restrições do problema, relativas a cada um dos arcos do grafo, traduzem as relações de precedência entre as actividades. Para uma dada actividade j, o tempo de início da actividade j deve ser posterior ao tempo de conclusão de cada uma das actividades i que precedem j. Dado que ti designa o tempo de início da actividade i, a função $ti + di$ designa o tempo de conclusão da actividade i. O projecto termina no instante de tempo tf, quando todas as actividades predecessoras imediatas da actividade fictícia fim estiverem concluídas.

Função Objectivo: min : tf ;

arco_01: t1>=t0 + 4;
arco_12: t2>=t1 + 6;
arco_23: t3>=t2 + 7;
arco_i0: t0>=ti + 0;
arco_04: t4>=t0 + 4;
arco_42: t2>=t4 + 9;
arco_3f: tf>=t3 + 2;
arco_i6: t6>=ti + 0;
arco_74: t4>=t7 + 6;
arco_67: t7>=t6 + 5;
arco_78: t8>=t7 + 6;
arco_610: t10>=t6 + 5;
arco_108: t8>=t10 + 8;
arco_1011: t11>=t10 + 8;
arco_43: t3>=t4 + 9;
arco_4f: tf>=t4 + 9;
arco_83: t3>=t5 + 4;
arco_8f: tf>=t8 + 4;
arco_11f: tf>=11 + 7;

O valor da variável de decisão ti 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 projecto é executado num tempo com a duração do valor da solução óptima. O Diagrama de Gantt do projecto em análise é apresentado na Figura 3.

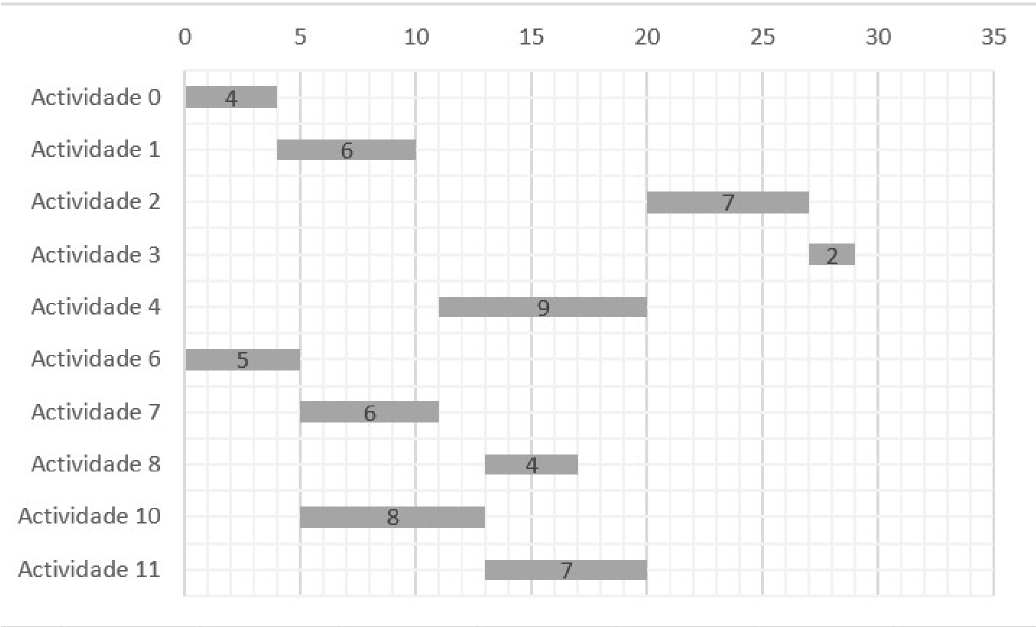

Figura 3: Diagrama de Gantt do projecto em análise.

4 Parte I

O método do caminho crítico assume que qualquer actividade pode ser realizada em paralelo com qualquer outra. Há, no entanto, situações em que os recursos são limitados, e a realização das actividades depende dos recursos disponíveis. Claramente, pode haver casos em que a duração global do projecto aumenta em consequência disso.

Considerando que existe apenas um equipamento para realizar as três actividades. O objectivo continua a ser realizar o projecto na menor duração possível mas garantir que não pode haver nenhum instante em que essas actividades estejam a acontecer ao mesmo tempo.

Analisando o Diagrama de Gantt presente na figura 3, podemos observar que existem 2 trios de actividades que têm instantes que coincidem, sendo estas as actividades: 1-7-10 e 4-8-11;

Todas respeitam a condição de ter alguma actividade pertencerem ao CPM, sendo assim escolhemos uma ao acaso, a sequência escolhida foi a 1-7-10.

4.1 Formulação do problema

Considerando na mesma cada variável de decisão ti, $\forall i$, representar o tempo de início da actividade i. Adicionamos uma variável binária yi o que vai permitir distinguir todas as possíveis sequências entre as 3 actividades em causa, isto é, se a variável yi for igual a um significa que a sequência i ocorre, se for igual a zero significa que esta não ocorre.

Sequências possíveis yi:

$$i = 1, t_1 t_7 t_{10}$$

$$i = 2, t_1 t_{10} t_7$$

$$i = 3, t_7 t_1 t_{10}$$

$$i = 4, t_7 t_{10} t_1$$

$$i = 5, t_{10} t_1 t_7$$

$$i = 6, t_{10} t_7 t_1$$

Agora além das restrições apresentadas na parte 0, iremos ter um conjunto de restrições para assegurar que estas 3 actividades não ocorrem em simultâneo.

Para cada sequência iremos ter duas condições, uma para indicar que o tempo entre a actividade em segundo lugar apenas acontece após a primeira actividade ter começado e consequentemente ter terminado e a segunda condição para que a terceira actividade só aconteça após a segunda actividade e a sua respectiva duração. Por exemplo para a sequência 1:

$$t_1 + 6 <= t_7 + (1 - y_1) * 100$$

$$t_7 + 6 <= t_{10} + (1 - y_1) * 100$$

,onde na primeira condição o valor 6 representa a duração da actividade 1 somando ao instante onde começa tem de ser menor que o instante que a actividade 7 é começada, o (1-y1)*100 é uma função relativamente grande ao problema para que caso esta sequência não aconteça a restrição continue a ser verdadeira. A segunda condição apresentada representa que o instante em que actividade 7 é começada mais a sua duração tem que ser menor que o instante em que actividade 10 é começada.

E por final temos a restrição:

$$\sum_{i=1}^{6} y_i = 1$$

, que garante que apenas uma sequência é usada.

4.2 Ficheiro de Input

```
min: tf ;
/*restrições*/
arco_01: t1>= t0 + 4 ;
arco_12: t2>= t1 + 6 ;
arco_23: t3>= t2 + 7 ;
arco_i0: t0>= ti + 0 ;
arco_04: t4>= t0 + 4 ;
arco_42: t2>= t4 + 9 ;

arco_3f: tf>= t3 + 2 ;


arco_i6: t6>= ti + 0 ;
arco_74: t4>= t7 + 6 ;

arco_67: t7>= t6 + 5 ;
arco_78: t8>= t7 + 6 ;

arco_610: t10>= t6 + 5 ;
arco_108: t8>= t10 + 8 ;

arco_1011: t11>= t10 + 8;

// parte 0
arco_43: t3>= t4 + 9;
arco_4f: tf>= t4 +9;
arco_83: t3>= t8 +4;
arco_8f: tf>= t8 +4;
arco_11f: tf>= t11 + 7;

// parte 1

seq_1_res1: t1 + 6 <= t7 +100 -100*y1;
seq_1_res2: t7 + 6 <= t10 +100 -100*y1;

seq_2_res1: t1 + 6 <= t10 +100 -100*y2;
seq_2_res2: t10 + 8 <= t7 +100 -100*y2;

seq_3_res1: t7 + 6 <= t1 +100 -100*y3;
seq_3_res2: t1 + 6 <= t10 +100 -100*y3;
seq_3_res2: t1 + 6 <= t10 +100 -100*y3;

seq_4_res1: t7 + 6 <= t10 +100 -100*y4;
seq_4_res2: t10 + 8 <= t1 +100 -100*y4;

seq_5_res1: t10 + 8 <= t1 +100 -100*y5;
seq_5_res2: t1 + 6 <= t7 +100 -100*y5;

seq_6_res1: t10 + 8 <= t7 +100 -100*y6;
seq_6_res2: t7 + 6 <= t1 +100 -100*y6;

so_1seq: y1 + y2 +y3 + y4 + y5 + y6 = 1;

bin y1,y2,y3,y4,y5,y6;
```

4.3 Ficheiro de Output

Source | Matrix | Options | Result

Objective | Constraints | Sensitivity

Variables	MILP ...	MILP ...	result
	34	32	32
tf	34	32	32
t1	4	11	11
t0	0	0	0
t2	25	23	23
t3	32	30	30
ti	0	0	0
t4	16	14	14
t6	0	0	0
t7	10	5	5
t8	24	25	25
t10	16	17	17
t11	24	25	25
y1	1	0	0
y2	0	0	0
y3	0	1	1
y4	0	0	0
y5	0	0	0
y6	0	0	0

4.4 Plano de execução do projecto

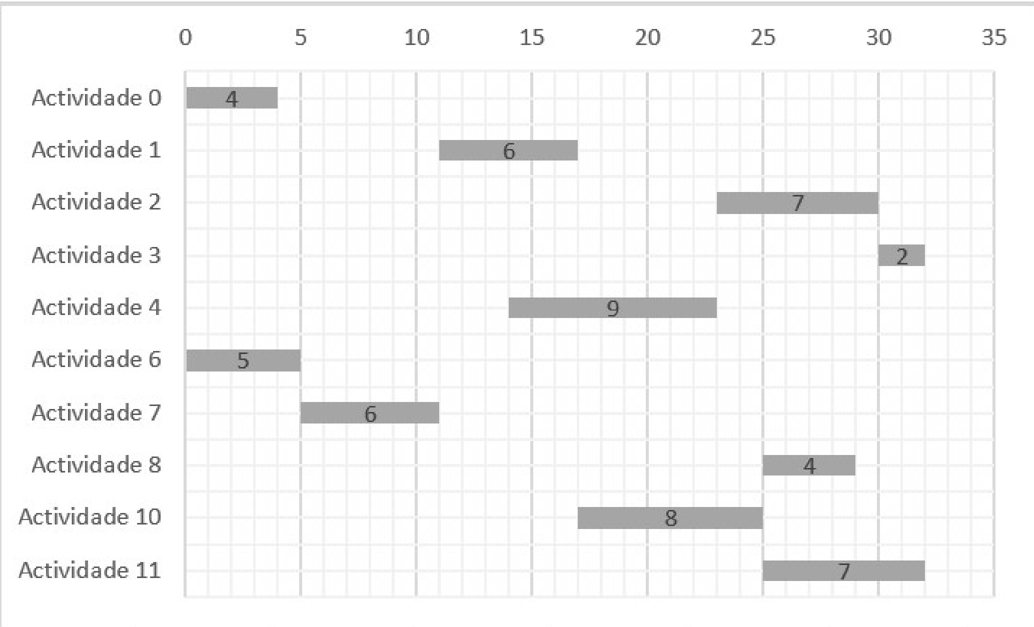

Figura 4: Diagrama de Gantt do projecto considerando que existe apenas um equipamento para realizar as actividades 1-7-10.

5 Parte II

Considerando 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, utiliza-se uma função linear com 2 partes.

Na variante em análise, cada actividade tem cinco parâmetros adicionais: o primeiro é o valor do custo normal, expresso em unidades monetárias [U.M.], o segundo é o valor de c1, o custo suplementar de reduzir a duração da actividade de uma unidade de tempo [U.T.], expresso em [U.M./U.T.], o terceiro é o valor da máxima redução de tempo a um custo c1, o quarto é o valor de c2, o custo suplementar de reduzir a duração da actividade de uma unidade de tempo [U.T.] após ter aplicado a máxima redução a um custo c1, expresso em [U.M./U.T.], e o quinto é o valor da máxima redução de tempo a um custo c2. Esses valores encontra-se presentes na tabela na parte II do enunciado do trabalho.

Pretende-se nesta parte que o tempo de execução do projecto encontrado na parte 0 seja reduzido em 3 U.T. O objectivo 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.

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
4	2000	800	2	400	1
5	1000	1600	0,5	800	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

5.1 Formulação do problema

Dado o problema apresentado decidimos criar as seguintes variáveis de decisão:

rc_{ij} : unidades inteiras de tempo a reduzir da actividade i a partir do tipo de redução j;

$$j \in \{1, 2\}\ , i \in \{0, 1, 2, 3, 4, 6, 7, 8, 10, 11\}$$

Como a redução de tipo 2 só acontece após o limite máximo da redução tipo 1 ser atingido, concluímos que necessitamos de uma variável binária que nos dirá se este limite foi atingido ou não.

y_i : decide se existe, ou não existe, redução do tipo 2 para a actividade j;

$$i \in \{0, 1, 2, 3, 4, 6, 7, 8, 10, 11\}, \mathbf{y}i \in \{0 \text{ se não existir}, 1 \text{ se existir}\}$$

Além destas duas ainda temos as variáveis das outras partes do trabalho nomeadamente a variável *ti*.

Definimos então como função objectivo o mínimo do somatório do custo das reduções multiplicado pelo respectivo número de vezes são feitas essas reduções, ou seja:

$f.o : min z = 200rc_{01} + 100rc_{02} + 600rc_{11} + 300rc_{12} + 1000rc_{21} + 500rc_{22} + 200rc_{21} + 500rc_{22} 200rc_{31} + 100rc_{32} + 800rc_{41} + 400rc_{42} + 180rc_{61} + 90rc_{62} + 0rc_{71} + 0rc_{72} + 200rc_{81} + 100rc_{82} + 1000rc_{101} + 500rc_{102} + 600rc_{111} + 300rc_{112}$

Em termos de restrições definimos antes de tudo que o tempo final teria de ser menor ou igual a 26. De seguida representamos as dependências entre as actividades, ou seja, uma restrição para cada arco, a título ilustrativo para o arco que vai de 0 para 1 temos:

$$t_1 >= t_0 - rc_{01} - rc_{02} + 4$$

, sendo que 4 é a duração da actividade 0 e os *rc* as respectivas reduções da actividade 0, ou seja, o instante que ocorre a actividade 1 tem de ser após a actividade 0, é de notar que subtraímos dois valores à duração fixa da actividade 0 pois são as reduções de unidades de tempo que irão acontecer relativamente a essa actividade.

Outro grupo de restrições que vamos adicionar, para cada actividade, são restrições para limitar o número de reduções possíveis de cada redução, ou seja, por exemplo:

$$rc_{11} <= 1 \text{ e } rc_{12} <= 1y_1 \ \text{ e } rc_{11} >= 1y_1$$

, onde na primeira restrição limitamos superiormente o número de restrições da actividade da redução de tipo 1 pelo seu respectivo limite para esse tipo de redução que nos é dado na tabela apresentada, relativamente à segunda restrição é o mesmo que na primeira só que agora é para reduções do tipo 2 sendo que o limite desta está está a ser multiplicado por **y**, que é a variável que nos diz que se podemos ou não usar reduções do tipo 2, para que caso não possamos usar este tipo de reduções para que nesta actividade esta restrição continue verdadeira, quanto à terceira restrição estamos a indicar que a variável **y** da respectiva actividade o valor 1 ou 0 dependendo se o limite máximo da restrição que no caso da actividade 1 é 1 foi atingido ou não.

5.2 Ficheiro de Input

```
min: 200 rc01 + 100 rc02
     + 600 rc11 + 300 rc12
     + 1000 rc21 + 500 rc22
     + 200 rc31 + 100 rc32
     + 800 rc41 + 400 rc42
     + 180 rc61 + 90 rc62
     + 0 rc71 + 0 rc72
     + 200 rc81 + 100 rc82
     + 1000 rc101 + 500 rc102
     + 600 rc111 + 300 rc112;
/*restrições*/
tf<=26;
arco_01: t1>= t0 - rc01 -rc02 + 4 ;
arco_12: t2>= t1 - rc11 -rc12 + 6 ;
arco_23: t3>= t2 - rc21 -rc22 + 7 ;
arco_i0: t0>= ti + 0 ;
arco_04: t4>= t0 - rc01 -rc02 + 4 ;
arco_42: t2>= t4 - rc41 -rc42 + 9 ;
arco_3f: tf>= t3 - rc31 -rc32 + 2 ;
arco_i6: t6>= ti + 0 ;
arco_74: t4>= t7 - rc71 -rc72 + 6 ;
arco_67: t7>= t6 - rc61 -rc62 + 5 ;
arco_78: t8>= t7 - rc71 -rc72 + 6 ;
arco_610: t10>= t6 - rc61 -rc62 + 5 ;
arco_108: t8>= t10 - rc101 -rc102 + 8 ;
arco_1011: t11>= t10 - rc101 -rc102 + 8;

arco_43: t3>= t4 - rc41 -rc42 +9;
arco_4f: tf>= t4 - rc41 -rc42 +9;
arco_83: t3>= t8 - rc81 -rc82 +4;
arco_8f: tf>= t8 - rc81 -rc82 +4;
//max reducao 1
rc01<= 0.5;
rc11<= 1;
rc21<= 3;
rc31<= 0.5;
rc41<= 2;
rc61<= 1;
rc71<= 0;
rc81<= 0.5;
rc101<= 0.5;
rc111<= 1;

//max reducao 2
rc02<= 0.5 y0;
rc12<= 1 y1;
rc22<= 1 y2;
rc32<= 0.5 y3;
rc42<= 1 y4;
rc62<= 1 y6;
rc72<= 0 y7;
rc82<= 0.5 y8;
rc102<=0.5 y10;
rc112<= 1 y11;
```

```
// so existe reducao de 2 quando reducao 1 esta no max
rc01>= 0.5 y0;
rc11>= 1 y1;
rc21>= 3 y2;
rc31>= 0.5 y3;
rc41>= 2 y4;
rc61>= 1 y6;
rc71>= 0 y7;
rc81>= 0.5 y8;
rc101>= 0.5 y10;
rc111>= 1 y11;

bin y0,y1,y2,y3,y4,y6,y7,y8,y10,y11;
int rc01,rc02,rc11,rc12,rc21,rc22,rc31,rc32,rc41,rc42,rc61,rc62,rc71,rc72,rc81,rc82,rc101,rc102,rc111,rc112;
```

5.3 Ficheiro de Output

Variables	MILP ...	MILP ...	result
	1780	1070	1070
rc01	0	0	0
rc02	0	0	0
rc11	0	0	0
rc12	0	0	0
rc21	0	0	0
rc22	0	0	0
rc31	0	0	0
rc32	0	0	0
rc41	2	1	1
rc42	0	0	0
rc61	1	1	1
rc62	0	1	1
rc71	0	0	0
rc72	0	0	0
rc81	0	0	0
rc82	0	0	0
rc101	0	0	0
rc102	0	0	0
rc111	0	0	0
rc112	0	0	0
tf	26	26	26
t1	4	4	4
t0	0	0	0
t2	17	17	17
t3	24	24	24
ti	0	0	0
t4	10	9	9
t6	0	0	0
t7	4,000...	3	3
t8	12	11	11
t10	4,000...	3	3
t11	12	11	11
y0	0	0	0
y1	0	0	0
y2	0	0	0
y3	0	0	0
y4	1	0	0
y6	1	1	1
y7	0	0	0

5.4 Plano de execução do projecto

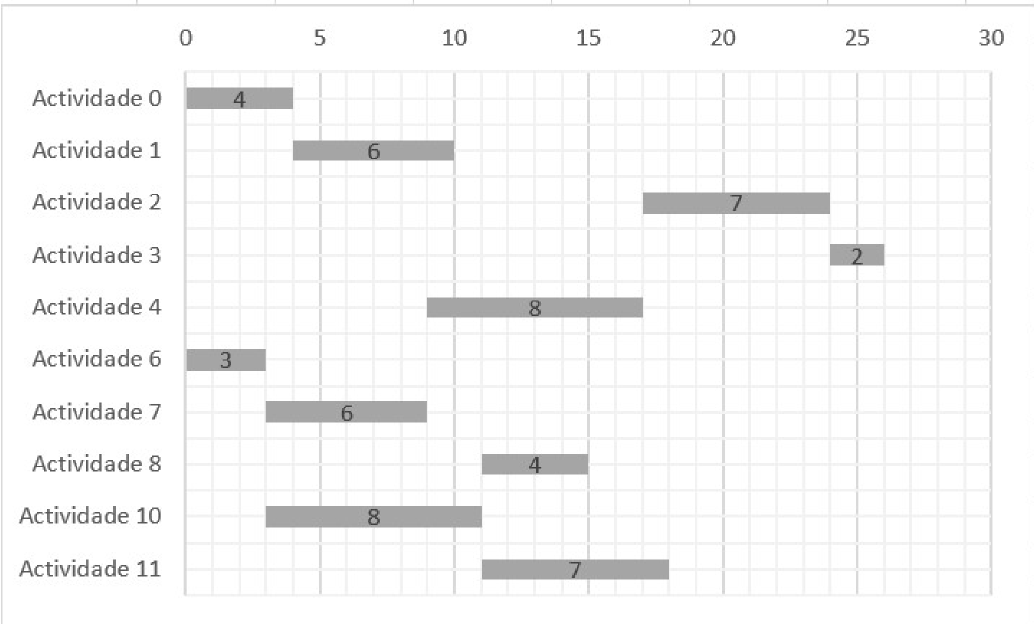

Figura 5: Diagrama de Gantt de modo a ser reduzido em 3 U.T.

5.5 Verificação do custo da solução

O tempo de execução do projecto foi de 26, isto significa que reduziu 3 unidades comparativamente ao tempo de execução encontrado anteriormente como era suposto acontecer o que demonstra que as restrições acima apresentadas foram bem definidas e planeadas de modo a haver uma redução no tempo de execução. Para atingir esse fim temos que reduzir a duração da actividade 4 em 1 U.T, não utilizando o máximo de redução a custo c1 desta actividade e também reduzir a duração da actividade 6 utilizando o máximo de redução a custo c1 e também utilizar o máximo de redução a custo c2 levando a que esta tenha uma redução global de 2 U.T.. Esta redução no tempo de execução tem um custo adicional de 1070 U.M..

6 Conclusão

O presente relatório descreveu, de forma sucinta, a resolução do problema proposto utilizando um Modelo de Programação Linear com diferentes abordagens.

Consideramos que os principais objectivos foram compridos.

Sentimos que a realização deste projecto consolidou os nossos conhecimentos de resolução de problemas através do Modelo de Programação Linear, nomeadamente ajudou-nos a desenvolver a nossa capacidade de analisar sistemas complexos, como também ajudou no processo de criação de modelos de sistemas complexos. Propiciou-nos a experiência de utilizar um programa computacional adequado para obter soluções dos modelos que definimos consequentemente fortaleceu a nossa capacidade de validação, análise e interpretação das soluções obtidas.