



Instituto Superior de Economia e Gestão

UNIVERSIDADE TÉCNICA DE LISBOA

DESDE 1911

MESTRADO

DECISÃO ECONÓMICA E EMPRESARIAL

TRABALHO FINAL DE MESTRADO

RELATÓRIO DE ESTÁGIO

**GESTÃO DE PROJETOS COM RECURSOS LIMITADOS - UMA
HEURÍSTICA**

NELSON DIOGO NETO GONÇALVES

SETEMBRO - 2012



Instituto Superior de Economia e Gestão

UNIVERSIDADE TÉCNICA DE LISBOA

DESDE 1911

MESTRADO EM DECISÃO ECONÓMICA E EMPRESARIAL

TRABALHO FINAL DE MESTRADO RELATÓRIO DE ESTÁGIO

**GESTÃO DE PROJETOS COM RECURSOS LIMITADOS - UMA
HEURÍSTICA**

NELSON DIOGO NETO GONÇALVES

ORIENTAÇÃO:

**PROFESSORA DOUTORA LEONOR SANTIAGO PINTO
DOUTORA ANNABELLE LE ROHELLEC**

SETEMBRO - 2012

Agradecimentos

Desejo prestar os meus mais sinceros agradecimentos a várias pessoas e entidades que de forma direta ou indireta contribuíram para a realização deste relatório de estágio.

Pela oportunidade que me foi dada na realização do estágio, quero agradecer à empresa Quidgest S.A., ao Doutor João Paulo, à Doutora Annabelle e a toda a equipa de colaboradores pela prestabilidade com que sempre me presentearam.

À Professora Doutora Leonor Santiago Pinto, por ter sido essencial no decorrer da elaboração do presente relatório. Pela orientação, compreensão, ajuda e acompanhamento que teve para comigo, obrigado Professora.

A toda a minha família, que sempre acreditou em mim, o meu mais sincero obrigado.

Resumo

O desafio proposto pela Quidgest S.A. foi o desenvolvimento de uma ferramenta que permitisse o planeamento de projetos, incluindo explicitamente o facto de os recursos disponíveis serem limitados. As empresas solicitam frequentemente a resolução deste tipo de problemas, uma vez que eles se ajustam à realidade.

Uma das formas de modelizar a gestão de projetos com recursos limitados recorre ao *Multi-Mode Resource Constrained Project Scheduling Problem* (MRCPSP). Para cada uma das atividades do projeto são considerados vários modos (*multi-mode*) de execução, que identificam uma duração e correspondente consumo de recursos. O termo modo designa uma alternativa.

O MRCPSP consiste em determinar o instante de tempo em que cada atividade deve ser iniciada e o modo que deve ser utilizado para realizar cada uma das atividades, respeitando as relações de precedência e as restrições de recursos, com o objetivo de minimizar a duração total do projeto.

Com esta opção de formulação selecionou-se uma heurística sugerida por Favez Boctor em 1996, que se programou em Visual Basic para Excel.

Palavras chave: Escalonamento de Projetos, Recursos limitados, Heurísticas.

Abstract

The challenge proposed by Quidgest S.A. was the development of a tool that would allow the planning of projects, including explicitly the fact that the resources available are limited. Companies often seek to solve this kind of problems once they adjust to reality.

One way to model the project management with limited resources, refers to the Multi-Mode Resource Constrained Project Scheduling Problem (MRCPSP). For each activity of the project are considered various modes of execution, which identify duration and the corresponding resource consumption. The term mode designates an alternative.

The MRCPSP sets the time that each activity should be started and the mode that should be performed, respecting the precedence relations and resource constraints, in order to minimize the total duration of the project.

With this formulation, the heuristic suggested by Fayed Boctor in 1996 was selected and programmed in Visual Basic for Excel.

Keywords: Scheduling Projects, Limited Resources, Heuristics.

Índice

Agradecimentos	1
Resumo	2
Abstract.....	3
Índice de Figuras.....	5
Índice de Tabelas	6
1 Introdução	7
2 Gestão de Projetos	9
2.1 Planeamento de Projetos	11
2.2 Escalonamento de Projetos com Recursos Limitados	16
3 MRCPSP	21
3.1 Métodos de Resolução.....	22
3.2 Heurística	23
3.2.1 Procedimento.....	24
3.2.2 Exemplo.....	34
4 Conclusões	36
5 Referências Bibliográficas.....	38
Anexo	41

Índice de Figuras

Figura 1 - Processo da gestão de projetos.....	10
Figura 2 - Diagrama em rede AoA.....	13
Figura 3 - Diagrama em rede AoN	13
Figura 4 - Caminho crítico do projeto exemplo.....	14
Figura 5 - Mapa de Gantt	15
Figura 6 - Procedimento CPM	27
Figura 7 - Fluxograma da heurística de Boctor	30
Figura 8 - Iteração 1 para o projeto exemplo.....	34
Figura 9 - Iteração 2 para o projeto exemplo.....	35
Figura 10 - Iteração 3 para o projeto exemplo.....	35

Índice de Tabelas

Tabela 1 - Projeto exemplo	12
Tabela 2 - Projeto exemplo com recursos	16
Tabela 3 - Projeto exemplo com 2 modos de execução para cada atividade... 21	
Tabela 4 - Projeto exemplo - numeração de atividades	25
Tabela 5 - Projeto exemplo com notação	34

1 Introdução

Com o presente relatório pretende-se descrever o estágio de 3 meses decorrido na Quidgest S.A..

A Quidgest é uma empresa tecnológica 100% portuguesa, que desde 1988 se dedica à consultoria e desenvolvimento de sistemas de informação de gestão, com forte aposta na investigação em engenharia de software.

Entre outros prémios e distinções é Microsoft Gold Certified partner, PME Excelência 2009, 2010 e 2011 e integra a rede de PME inovadoras da COTEC Portugal - Associação empresarial para a inovação.

Pela variedade de soluções produzidas nos seus departamentos de I&D, com mais de 200 sistemas de informação diferentes em produção, a Quidgest é um dos maiores produtores de software empresarial de origem nacional.

Com empresas já constituídas em Espanha, Reino Unido, Timor-Leste, Moçambique e Macau, para além de parcerias em Angola e Lituânia, a Quidgest tem investido fortemente e com sucesso na sua internacionalização.

Em Portugal, a empresa foi pioneira na informatização da Administração Pública, e o seu sistema de gestão SINGAP é, hoje em dia, utilizado por mais de metade dos organismos públicos da Administração Central do Estado.

Foi também pioneira na racionalização em prestação de cuidados de saúde com a solução PRATICS, um sistema que faz a gestão dos beneficiários e prestadores de todos os planos de saúde disponibilizados pela Portugal Telecom - Associação de Cuidados de Saúde (PT-ACS).

O desafio proposto pela empresa foi o de otimizar o escalonamento de projetos informáticos. Para isso foi sugerida a elaboração de uma ferramenta de trabalho que conseguisse gerir a utilização dos recursos afetos ao projeto no escalonamento de atividades, com a finalidade de minimizar o tempo de execução dos projetos.

No capítulo seguinte faz-se uma breve apresentação da gestão de projetos, destacando a situação em que a escassez de recursos é relevante.

No capítulo 3 é abordado o problema MRCPSP e apresentada a heurística de Boctor, que foi aplicada para resolver este problema.

Por fim, são alinhadas as principais conclusões do trabalho.

2 Gestão de Projetos

A gestão de projetos, tem tido cada vez mais um importante papel na sociedade. No mundo atual em que a competitividade é uma característica fundamental, é determinante cumprir os prazos de entrega previstos, respeitando o orçamento e com a qualidade pretendida. Não é por isso surpreendente que a gestão de projetos seja um tema bastante relevante, ao qual se dedique cada vez mais atenção.

De acordo com o PMI ou *Project Management Institute* (instituição com cerca de 650.000 membros certificada pela ISO (*International Organization for Standardization*)) um projeto é definido como sendo um esforço temporário empreendido para criar um único produto, serviço ou resultado (PMI, 2004).

A mesma instituição define a gestão de projetos como a aplicação de conhecimentos, competências, ferramentas e técnicas às atividades do projeto com vista a alcançar um conjunto de metas/objetivos e cumprir requisitos preestabelecidos do produto.

O processo de gestão de projetos é concretizado através de cinco grandes fases que interagem: **Início, Planeamento, Execução, Controlo e Encerramento.**

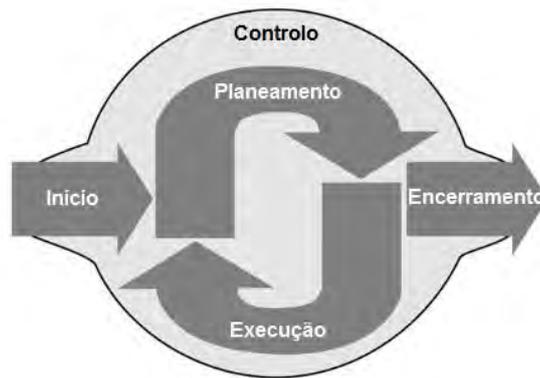


Figura 1 - Processo da gestão de projetos

- **Início** – A ideia do projeto é delineada, assim como os seus objetivos. Nesta fase não existe uma noção bem clara do que se pretende fazer nem como o atingir, daí a necessidade de se proceder a uma análise de viabilidade do projeto, conjuntamente com uma análise económica e de risco. O resultado final é a decisão sobre a implementação do projeto;
- **Planeamento** – São desenvolvidos planos detalhados, com identificação e caracterização das tarefas necessárias e respetivas durações, assim como, das relações de precedência entre atividades. São definidos os custos e os recursos indispensáveis para executar cada tarefa. Nesta fase, são também estabelecidas metas que irão servir para controlar a execução do projeto. Com todas estas informações é então possível construir um plano de execução.
- **Execução** – Consiste basicamente em realizar o trabalho e comunicar resultados. A execução do trabalho deve seguir o plano preestabelecido, sendo necessário garantir isso mesmo através de uma monitorização constante que esteja atenta às metas predefinidas, aos custos, bem como à qualidade do produto que vai sendo obtido;

- **Controlo** – Acompanha o progresso do projeto com vista a identificar os potenciais problemas a tempo de tomar medidas de correção necessárias, para que os objetivos finais do projeto sejam atingidos. Caso haja um desvio do plano, a informação disponível será usada para ajustar o plano de forma a repor o projeto no seu curso normal;
- **Encerramento** – Avalia o resultado final do projeto. A informação mais importante a reter será a razão para a discrepância dos dados reais face ao inicialmente planeado, nomeadamente na duração real do projeto, no custo das atividades e na utilização dos recursos, assim como, o grau de satisfação do cliente perante o resultado final. Esta informação deve ser registada para referência em eventuais futuros projetos.

2.1 Planeamento de Projetos

O planeamento é uma das competências mais importantes de qualquer gestor de projeto. No entanto, na prática encontramos muitos profissionais que conduzem os seus projetos sem reconhecer a vantagem de estabelecerem um plano prévio, alegando desperdício de tempo. Contudo, é precisamente quando existem mais restrições temporais para a execução do projeto que o planeamento é mais importante. Na ausência de plano, torna-se impossível controlar a execução, uma vez que não existe termo de comparação para reconhecer a necessidade de tomar medidas corretivas.

É na fase de planeamento que se especifica em pormenor tudo o que deve ser feito e como deve ser feito. Numa primeira fase, o projeto é estruturado, ou seja, dividido em atividades. De seguida, essas atividades são ordenadas

tendo em atenção relações de precedências que têm origem em questões processuais ou tecnológicas e é avaliada a duração das atividades.

A estrutura de um projeto conduz-nos a informação que está sintetizada na tabela seguinte:

Atividade	Precedente(s)	Duração
A	-	3
B	A	4
C	A	6
D	B,C	2

Tabela 1 - Projeto exemplo

Far-se-á uso deste projeto exemplo ao longo do relatório, sempre que se julgue necessário esclarecer algum tópico.

Depois de estar definida a estrutura do projeto, é necessário proceder ao seu escalonamento.

O escalonamento de um projeto consiste em encadear as atividades no tempo, conseguindo desta forma estabelecer datas de início (e de fim) de cada atividade e conseqüentemente a duração prevista do projeto.

Normalmente, o encadeamento temporal das atividades é representado por um diagrama em rede, constituído por um conjunto de nós e arcos. Dois tipos de representação podem ser adotadas:

- **Activity on arc (AoA):** cada atividade é representada por um arco. Um nó é utilizado para separar uma atividade da sua predecessora imediata. A sequência de arcos representa as precedências entre atividades.

Nesta abordagem, por vezes, há necessidade de considerar atividades fictícias, normalmente desenhadas a tracejado, como na Figura 2.

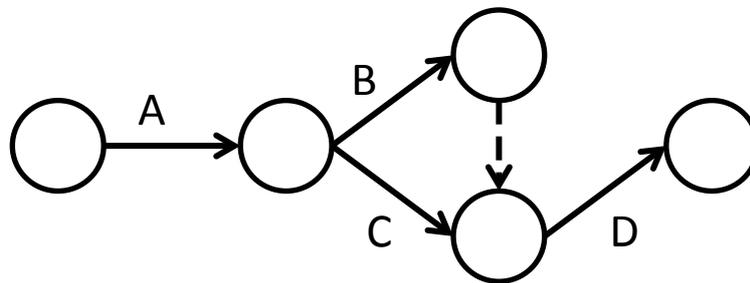


Figura 2 - Diagrama em rede AoA

- **Activity on node (AoN):** cada atividade é representada por um nó. Os arcos são utilizados para representar as relações de precedência entre atividades, como mostra a Figura 3.

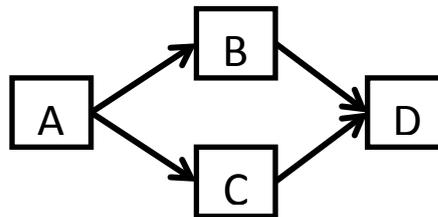


Figura 3 - Diagrama em rede AoN

A representação adotada neste trabalho é *Activity on node*.

Após a identificação da rede do projeto, pode aplicar-se o método do caminho crítico (CPM ou *Critical Path Method*).

Este método foi introduzido (Kelley, et al., 1959) com o intuito de melhorar o processo de escalonamento de projetos nas suas empresas (DuPont e Remington Rand), nas quais, os tempos das atividades são perfeitamente conhecidos.

James Kelley e Morgan Walker descrevem um algoritmo para determinar o conjunto de atividades que não podem sofrer atrasos na sua realização, sob pena de atrasarem o projeto. Essas atividades são chamadas críticas, e o seu conjunto define um (ou vários) caminho crítico (caminhos críticos). Normalmente, as atividades críticas representam uma pequena percentagem do conjunto de atividades do projeto.

As atividades que não são críticas, ou seja, em que o tempo em que podem ser realizadas é maior que a sua duração efetiva, são chamadas atividades com folga. A folga é o tempo que a atividade se pode atrasar sem atrasar o projeto. Assim, as atividades críticas, podem ser reconhecidas por exibirem folga nula.

Assumi-se que o tempo de cada atividade é conhecido, no entanto, por vezes é necessário estimar a sua duração. Quando assim acontece, aplica-se o PERT (*Program Evaluation and Review Technique*), que adota uma forma estatística para a medição e previsão do tempo de duração das atividades de um projeto.

O caminho crítico é determinado segundo um procedimento que mais à frente será detalhado. Contudo, retomando o projeto exemplo, é possível reconhecer com facilidade que será constituído pelas atividades A, C e D, uma vez que se trata do caminho mais longo na rede, como vemos na Figura 4.

Atividade	Precedente(s)	Duração
A	-	3
B	A	4
C	A	6
D	B,C	2

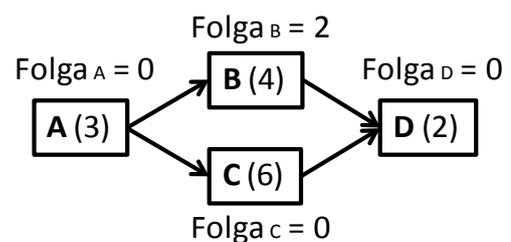


Figura 4 - Caminho crítico do projeto exemplo

Uma ferramenta de controlo muito útil para relacionar as atividades com o tempo, o que permite uma leitura mais fácil de todo o projeto, é o mapa de Gantt.

O mapa consiste num gráfico de barras, no qual o eixo vertical representa todas as atividades do projeto e o eixo horizontal representa o tempo. À frente de cada atividade é representada uma barra com a sua localização temporal, o que permite visualizar claramente quando deve ser iniciada e quando deve ser terminada.

O mapa de Gantt para o projeto exemplo consta na Figura 5.

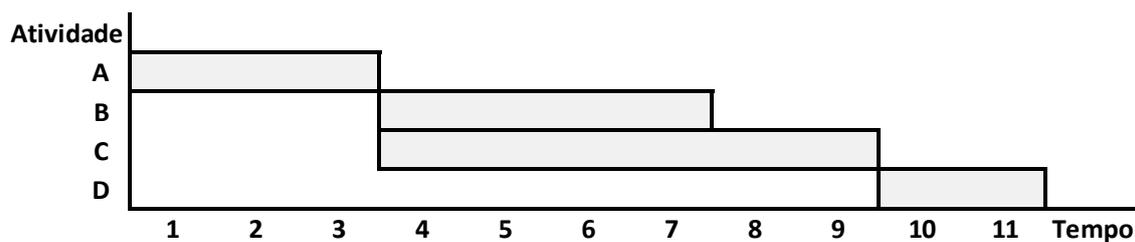


Figura 5 - Mapa de Gantt

Não obstante, estes métodos admitem que os recursos afetos ao projeto são ilimitados, o que não corresponde à realidade.

O problema típico no escalonamento de projetos com recursos limitados, consiste em escalonar um conjunto de atividades sujeitas a restrições de precedência e de recursos, com o objetivo de minimizar a duração total do projeto (*Resource-Constrained Project Scheduling Problem* ou RCPSP).

Neste problema, assume-se que existe apenas uma forma de realizar cada uma das atividades do projeto, portanto, existe apenas uma duração possível e um único valor de recursos necessários para que a atividade seja executada.

Cada um desses valores é perfeitamente conhecido. O problema é por isso do tipo determinístico.

Será importante referir que não é viável o escalonamento de atividades em paralelo que carecem de um recurso que não possa ser partilhado. Seguindo o projeto exemplo, é fácil perceber que se houver 6 unidades de recurso disponíveis e as atividades B e C necessitarem de 3 e 5 unidades de recurso respetivamente para serem executadas, estas não poderão ser realizadas em simultâneo.

Unidades de Recurso disponíveis = 6			
Atividade	Precedente(s)	Duração	Unidades de Recurso necessárias
A	-	3	2
B	A	4	3
C	A	6	5
D	B,C	2	4

Tabela 2 - Projeto exemplo com recursos

Na secção seguinte faz-se uma resenha de trabalhos publicados no âmbito do escalonamento de projetos com recursos limitados.

2.2 Escalonamento de Projetos com Recursos Limitados

Segundo (Slowinski, 1980) os recursos podem ser classificados em três categorias:

- **Recursos renováveis:** limitados em cada período do projeto; depois de utilizados, estes recursos voltam a estar disponíveis nos períodos seguintes. Exemplo: recursos humanos;

- **Recursos não renováveis:** limitados em todo o horizonte do projeto; restringidos por uma quantidade limitada para todo o projeto, que uma vez consumida não volta a estar disponível. Exemplo: orçamento;
- **Recursos duplamente restritos:** recursos limitados num dado período e no horizonte global do projeto; estes recursos podem sempre ser desdobrados em pelo menos um recurso renovável e um recurso não renovável. Exemplo: recursos financeiros.

Neste trabalho são considerados apenas recursos renováveis.

Em (Blazewicz, et al., 1983) é demonstrado que o problema típico de escalonamento de projetos com recursos escassos (RCPSP) é uma generalização do problema JSP¹ (*Job Shop Problem*), o qual pertence à classe de problemas NP-difíceis. Justificando-se assim, a utilização de heurísticas para a obtenção de soluções em tempo útil para problemas de grande dimensão. No entanto, também foram desenvolvidos vários métodos exatos, recorrendo principalmente a métodos de partições e avaliações sucessivas.

O RCPSP, é um problema com pressupostos demasiado restritivos para muitas situações reais. Existem portanto, várias extensões que podem ser consideradas.

No artigo publicado por Hartmann (Hartmann, et al., 2010), são apresentadas de forma genérica as principais extensões determinísticas do RCPSP,

¹ O problema consiste em executar um conjunto de atividades no menor período de tempo possível. Cada atividade consiste no uso de uma ou mais máquinas por um determinado período de tempo. Nenhuma atividade pode ser executada ao mesmo tempo na mesma máquina e uma atividade depois de iniciada não pode ser interrompida.

enquanto que nos artigos (Neumann, 1990) e (Herroelen, et al., 2005) são aprofundados os problemas do tipo estocástico.

➤ **Vários modos de execução (*multi-mode*)**

Cada atividade pode ser realizada em um de vários modos de execução. Cada modo de execução é uma combinação de recursos necessários e respectiva duração. Estes problemas são designados por *Multi-Mode Resource-Constrained Project Scheduling Problem* (MRCPSP).

O MRCPSP foi introduzido por Elmaghraby (Elmaghraby, 1977) e desde então tem merecido grande atenção por parte da comunidade científica, uma vez que se ajusta à realidade.

Teremos oportunidade de aprofundar esta extensão do RCPSP posteriormente, dado que é o problema principal focado neste trabalho.

➤ **Compromisso tempo-custo (*time-cost trade-off*)**

Poderá existir uma relação alternativa para cada modo de execução, nomeadamente, entre o custo de execução e a sua duração. Assim, quanto maior for o investimento em materiais, recursos humanos ou outros, maior a rapidez com que se conseguirão realizar as atividades desse projeto. Essa relação define um problema que é designado por problema de compromisso tempo-custo (*time-cost trade-off problem*).

Nos problemas de compromisso tempo-custo, não existe uma referência explícita à existência de recursos. Procura-se antes determinar o melhor

escalonamento das atividades de modo a respeitar uma restrição orçamental (*budget problem*), ou encontrar o escalonamento que minimize o custo de execução do projeto (*deadline problem*) mediante um prazo fixado.

Este problema foi discutido, entre outros, por (Demeulemeester, et al., 2000), (Ranjbar, et al., 2007) e (Ranjbar, et al., 2009).

➤ **Intervalos máximos e mínimos entre atividades (*minimum and maximum time lags*)**

Pode haver necessidade de considerar intervalos de tempo máximo e/ou mínimo entre o arranque de cada atividade. Estes desfasamentos temporais são muitas vezes necessários na prática, por exemplo, se o material depois de utilizado numa atividade tem a necessidade de repousar por um certo tempo para que fique em condições de ser trabalhado na atividade seguinte, ou se é necessária a execução simultânea de atividade.

Entre os trabalhos sobre o problema estão os de (Klein, et al., 2000) e (Neumann, et al., 2002).

➤ **Interrupção de atividades (*preemptive*)**

O problema padrão RCPSA assume que uma atividade depois de começada, não pode ser interrompida até estar finalizada. Nesse caso, estamos num problema do tipo *non preemptive*. Se por outro lado, as atividades pudessem ser interrompidas em qualquer momento da sua realização e reiniciadas mais tarde, estaríamos perante um problema do tipo *preemptive*.

Esta extensão foi estudada por (Demeulemeester, et al., 1996) e (Brucker, et al., 2001).

Em particular, no que se refere ao tipo de função objetivo, que representa o propósito que se pretende atingir ao resolver um determinado problema, têm sido estudados principalmente dois diferentes tipos de função:

- Minimização da duração total do projeto (*makespan minimization*) tem sido o mais popular até à data (Kolisch, et al., 2006). Em problemas com este tipo de função objetivo, procura-se escalonar as atividades do projeto de modo a garantir que o projeto termine o mais cedo possível.
- Maximização do valor atual líquido do projeto (*net present value maximization*). Este objetivo surge se os fluxos de caixa ocorrem enquanto o projeto é realizado. As saídas de caixa são normalmente induzidas pela execução das atividades e pela utilização de recursos. No entanto, os fluxos de caixa podem também resultar de pagamentos devidos à conclusão de determinadas partes do projeto. Este problema foi introduzido por Russell (Russell, 1970), e mais recentemente estudado por (Kimms, 2001), (Vanhoucke, et al., 2001) e (Padman, et al., 2006).

3 MRCPSP

De entre as várias extensões do problema de escalonamento de projetos com recursos limitados, o que se aprofunda neste trabalho é o problema de escalonamento de projetos com recursos limitados em que as atividades podem ser executadas segundo diferentes modos (MRCPSP).

Um modo pode ser definido como sendo uma combinação entre duração e respectivo consumo de recursos, na execução de uma atividade.

Na Tabela 3, foi adicionado ao nosso projeto exemplo um modo distinto de realizar cada atividade. Assume-se que no 1º modo a duração é sempre mais curta que no 2º, no entanto são necessárias sempre mais unidades de recurso para este ser executado.

Unidades de Recurso disponíveis = 6					
Atividade	Precedente(s)	Duração Modo 1	Unidades de Recurso necessárias Modo 1	Duração Modo 2	Unidades de Recurso necessárias Modo 2
A	-	3	2	5	1
B	A	4	3	7	1
C	A	6	5	8	3
D	B,C	2	4	3	3

Tabela 3 - Projeto exemplo com 2 modos de execução para cada atividade

O MRCPSP determina o instante de tempo em que cada atividade deve ser iniciada, e o modo que deve ser utilizado para realizar cada uma das atividades, respeitando as relações de precedência e as restrições de recursos, com o objetivo de minimizar a duração total do projeto.

3.1 Métodos de Resolução

No artigo (Peteghem, et al., 2010), é apresentada uma visão geral dos métodos de resolução exatos, heurísticos e meta-heurísticos disponíveis para o MRCPSP.

Entre os métodos exatos mais recentes estão os de (Sprecher, et al., 1997), (Sprecher, et al., 1998), (Hartmann, et al., 1998) e (Zhu, et al., 2006). Nos três primeiros trabalhos são apresentados algoritmos de partições e avaliações sucessivas (*branch-and-bound*), enquanto que em (Zhu, et al., 2006) é descrito um algoritmo de partição e corte (*branch-and-cut*).

No entanto, nenhum destes algoritmos garante a obtenção de uma solução ótima num tempo computacional razoável, sobretudo quando se trata de projetos de grande escala. Portanto, diferentes procedimentos heurísticos e meta-heurísticos foram naturalmente desenvolvidos.

Os trabalhos de investigação no campo dos métodos heurísticos para o MRCPSP são bastante mais numerosos do que aqueles relacionados com os métodos exatos. Essa realidade justifica-se quer pela complexidade inerente ao MRCPSP (foi provado em (Kolisch, et al., 1997) que o problema é NP-Completo), quer pelo facto de muitas vezes as heurísticas conduzirem em tempos computacionais aceitáveis, a soluções aproximadas de boa qualidade, que podem mesmo em alguns casos coincidir com a solução ótima.

No artigo (Peteghem, et al., 2010), são destacados os trabalhos de (Boctor F, 1993), (Boctor F., 1996), (Özdamar, et al., 1994), (Kolisch, et al., 1997) e (Lova, et al., 2006).

Veremos (Boctor F, 1993) e (Boctor F., 1996) em detalhe na secção seguinte, dado que a heurística aplicada no estágio pertence a este autor.

Em (Özdamar, et al., 1994) a abordagem heurística proposta para lidar com o problema é chamada *Local Constraint Based Analysis* (LCBA), onde a seleção de atividades e os seus respetivos modos é feita individualmente em cada ponto de decisão.

Kolish e Drexl (Kolisch, et al., 1997) desenvolveram uma heurística baseada em pesquisa local, que primeiro tenta encontrar uma solução e seguidamente procura na “vizinhança” o conjunto de modos que podem ser associados a cada atividade.

Mais recentemente, (Lova, et al., 2006) apresenta várias heurísticas *multi-pass*, baseadas em regras de prioridade para resolver o MRCPSP.

No campo das meta-heurísticas destacam-se os métodos de pesquisa tabu e os algoritmos genéticos. (Özdamar, 1999), (Hartmann, 2001) e (Alcaraz, et al., 2003) apresentam algoritmos genéticos, enquanto que o trabalho de (Nonobe, et al., 2001) se destaca nos métodos de pesquisa tabu.

3.2 Heurística

Boctor (Boctor F, 1993) desenvolveu um estudo em que envolveu 7 regras de prioridade para a seleção das atividades e 3 regras para os modos, numa combinação de 21 diferentes regras de seleção para efetuar o escalonamento. Nesse procedimento, as atividades são analisadas separadamente e apenas uma atividade é escalonada em cada iteração.

Mais tarde, o mesmo autor publica um outro estudo (Boctor F., 1996) onde apresenta uma heurística em que o escalonamento das atividades não é feito de forma individual, mas considerando conjuntos de atividades. O conjunto de atividades escalonado será aquele que tiver o melhor valor segundo um determinado critério.

Baseado em 240 problemas gerados aleatoriamente, (Boctor F., 1996) apresenta testes computacionais que exibem resultados competitivos, quando comparados com as melhores heurísticas propostas até à data da sua publicação.

Esta foi a heurística selecionada para resolver o problema MRCPSP durante o estágio na Quidgest e programada em VBA / Excel.

O manual de utilização do programa apresenta-se em anexo.

3.2.1 Procedimento

Na heurística, a estratégia consiste, num primeiro passo, em escalonar as atividades sem atender à restrição de recursos, isto é, assume-se que todas as atividades podem ser executadas no seu modo mais rápido, e portanto, mais consumidor de recursos.

Tal consiste na aplicação do método do caminho crítico, sendo as atividades consideradas nos nós (AoN).

Para isso, assume-se que as atividades estão ordenadas de tal forma que, se i é predecessor de j , então $i < j$. Se tal não for possível, é sinal que a rede do projeto tem um ciclo, e portanto, não está bem formulada.

No projeto exemplo, substituindo as letras pela numeração ordenada, consiste em aplicar o método ao seguinte projeto:

Atividade	Precedente(s)	Duração mais curta
1	-	3
2	1	4
3	1	6
4	2,3	2

Tabela 4 - Projeto exemplo – numeração de atividades

Assim começaremos por detalhar este método.

➤ CPM

Definições

Projeto - Conjunto de atividades a executar, com vista a alcançar um fim definido.

Atividade - Tarefa incluída no projeto. Pode exigir a conclusão de outras tarefas.

Predecessora imediata - A atividade i é considerada predecessora imediata da atividade j, se e só se, for necessário que a atividade i termine para a atividade j poder começar.

Sucessora imediata - A atividade i é considerada sucessora imediata da atividade j, se e só se, a atividade j for predecessora imediata da atividade i.

Dados - Input

A - conjunto de atividades do projeto ($A = \{0,1,2,\dots,N,N+1\}$);

P_i - conjunto das predecessoras imediatas da atividade *i*;

S_i - conjunto das sucessoras imediatas da atividade *i*;

N - número de atividades do projeto original ($|A| = N+2$);

d_i - duração da atividade *i*.

Por simplificação na apresentação dos algoritmos, assume-se que o projeto inclui sempre duas tarefas fictícias (com duração zero, e sem consumo de recursos). Uma inicial, 0, com $P_0=\emptyset$ e S_0 composto por todas as atividades sem predecessoras imediatas, e outra final, $N+1$, com $S_{N+1}=\emptyset$ e P_{N+1} composto por todas as atividades sem sucessoras imediatas.

Resultados - Output

ES_i - data mais cedo em que a atividade *i* pode começar;

EF_i - data mais cedo em que a atividade *i* pode acabar;

LS_i - data mais tarde em que a atividade *i* pode começar;

LF_i - data mais tarde em que a atividade *i* pode acabar;

F_i - folga total da atividade *i*.

Pseudocódigo

$$ES_0 = 0$$

$$EF_0 = 0$$

Para a atividade $i = 1, \dots, N+1$

$$ES_i = \max \{ EF_j : j \in P_i \}$$

$$EF_i = ES_i + d_i$$

$$LS_{N+1} = ES_{N+1}$$

$$LF_{N+1} = EF_{N+1}$$

Para a atividade $i = N, N-1, \dots, 0$

$$LF_i = \max \{ LF_j : j \in S_i \}$$

$$LS_i = LF_i - d_i$$

Para $i = 0, 1, \dots, N, N+1$

$$F_i = LF_i - EF_i$$

Na Figura 6 é exemplificado o procedimento CPM, para o projeto exemplo.

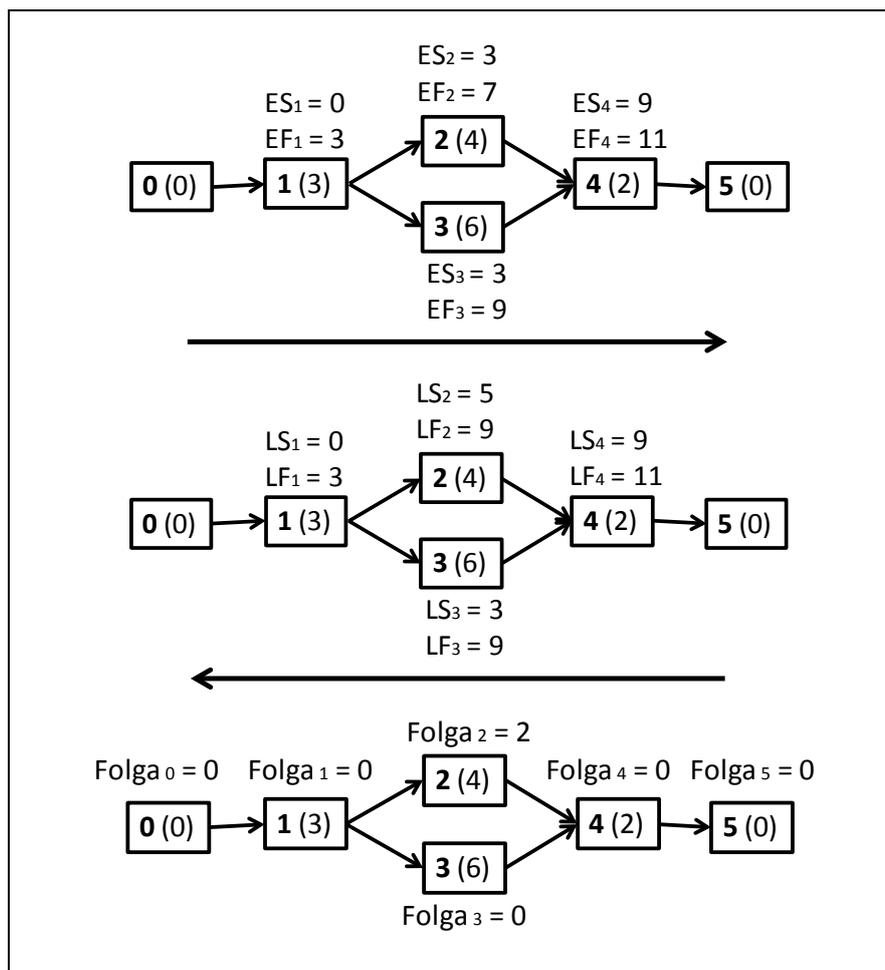


Figura 6 - Procedimento CPM

Após aplicação do CPM, revê-se o plano desde o início. Sempre que se encontre um período em que uma restrição de recursos seja violada altera-se o escalonamento. Opta-se por modos mais lentos, para uma ou várias atividades, ou, caso não seja viável, atrasa-se a sua execução.

Neste processo iterativo, as atividades são avaliadas em blocos (combinações), como descrito na secção seguinte.

O método CPM é utilizado no final de cada iteração, sempre que haja necessidade de atualizar as folgas correspondentes a cada atividade.

➤ Heurística

Definições

Modo - Combinação entre duração e respetivo consumo de recursos, na execução de uma atividade.

Combinação atividade-modo - Subconjunto de atividades, onde cada atividade tem um modo de execução associado.

Combinação atividade-modo dominada - A combinação C_2 é dominada pela combinação C_1 se uma ou ambas das seguintes condições se verificarem:

- C_1 inclui, além dos pares atividade-modo contidos em C_2 , outro par atividade-modo.
- C_1 é idêntica a C_2 , exceto no modo de execução de alguma das atividades, que tem uma duração inferior ao modo selecionado em C_2 .

t - instante de tempo, $t = 0, 1, 2, \dots, T$.

T – majorante para a duração total do projeto, por exemplo, a soma das durações de todas as atividades no modo mais longo.

Atividade escalonada em t - Atividade cujo modo de execução e o momento t em que deve ser iniciada já foram decididos.

Atividade escalonável em t - Atividade não escalonada em t, com todas as suas predecessoras imediatas já escalonadas.

Combinação atividade-modo escalonável em t - Combinação atividade-modo exequível, i.e., com recursos disponíveis em t, e nos momentos seguintes necessários para a execução das atividades, nos modos selecionados.

Dados - Input

J - número de modos possíveis para uma atividade. Por simplicidade assume-se que todas as atividades têm o mesmo número de modos possíveis;

K - número de recursos distintos;

R_k - quantidade de recurso do tipo k disponível para o projeto;

d_{ij} - duração da atividade i no modo j (assume-se que, se $j_1 < j_2$, a duração de i no modo j_1 é mais curta do que no modo j_2);

r_{ijk} - quantidade de recurso k, necessária para executar a atividade i no modo j;

Resultados - Output

t(i) – momento t em que a atividade i é iniciada;

V = {(i, j(i))}, sendo j(i) o modo selecionado para a atividade i.

Atrasos - Para cada Combinação atividade-modo escalonável estão definidos os seguintes atrasos:

- $a_{ij(i)} = \max \{0, d_{ij(i)} - d_{i1} - F_i\}$ - consequente de uma das suas atividades não estar selecionada no seu modo mais curto. $d_{ij(i)}$ é a duração da atividade i no modo selecionado j .
- $b_i = \max \{0, \min_{j \in J} (t_{ij} - t + d_{ij} - d_{i1} - F_i)\}$ - consequente da Combinação atividade-modo conter menos atividades que outra Combinação. t_{ij} é o momento mais próximo de t , em que existem recursos disponíveis para executar a atividade i (não selecionada na Combinação), no modo j .
- $I_c = \max \{0, \max (a_{i,j(i)}), \max (b_i)\}$ - em cada iteração será escalonada a Combinação escalonável não dominada com menor I_c .

Na Figura 7 apresenta-se o fluxograma do método.

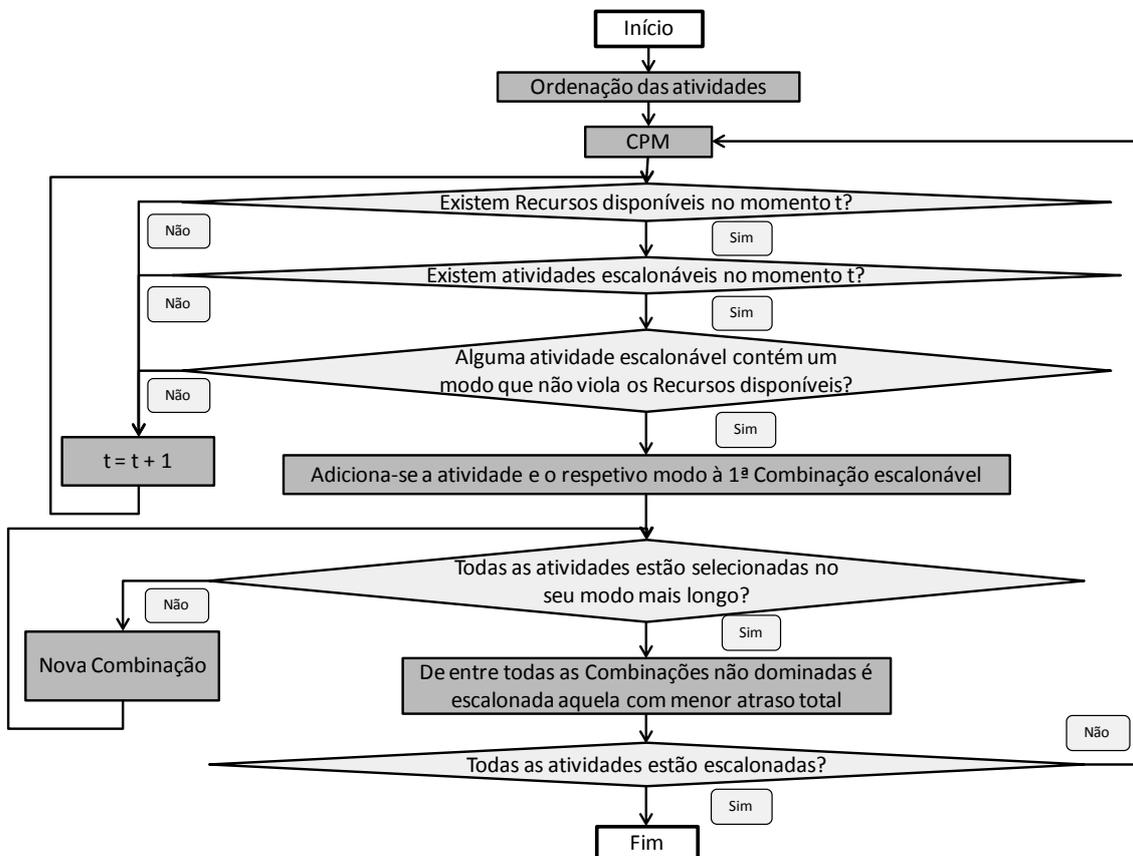


Figura 7 - Fluxograma da heurística de Boctor

Pseudocódigo

Inicialização

Ler: $A, P_i, N, d_{ij}, J, K, R_k, r_{ijk}$

Para o recurso $k = 1, \dots, K$

 Para o momento $t = 0, \dots, T$

$$D_{kt} = R_k$$

// V o conjunto de pares (actividade, modo) para actividades escalonadas

$$V = \{ (0,1) \}^2$$

$$t(0) = 0$$

Iteração: Enquanto existem actividades não escalonadas faz:

I. Determinação de t

Enquanto $D_{kt} = 0$ para todo o recurso k

$$t = t + 1$$

II. Determinação de E - Conjunto de Actividades escalonáveis em t

$$E = \emptyset$$

$E' = \{ i: (i, j(i)) \in V \}$ //conjunto das actividades já consideradas em V .

Para a actividade $i = 1, \dots, N$

 Se $i \notin E'$ e $P_i \subset E'$

$$E = E \cup \{ i \}$$

// no final $E \neq \emptyset$ caso contrário não teria sido possível numerar as actividades, o projecto estaria formulado incorrectamente.

III. Identificação de ConjCombNaoDomin - Combinações actividade-modo escalonáveis não dominadas

a) 1ª Combinação

Para o recurso $k = 1, \dots, K$

 Para o momento $l=t, \dots, T$

$$D'_{kl} = D_{kl}$$

² Assume-se que a actividade fictícia inicial está escalonada no primeiro modo.

ConjCombNaoDomin = \emptyset
C = \emptyset // C é uma Combinação atividade-modo escalonável

Para a atividade $i = 1, \dots, |E|$
 Inclui = True
 j = 0
 Enquanto (j<=J) e (Inclui = True)
 j=j+1
 Para o recurso $k = 1, \dots, K$
 Para o momento $l = t, \dots, t + d_{ij}$
 Se $r_{ijk} > D'_{kl}$
 Inclui = False
 Se Inclui = True
 j(i)' = j
 C = C U { (i, j'(i)) }
 Para o recurso $k = 1, \dots, K$
 Para o momento $l = t, \dots, t + d_{ij}$
 $D'_{kl} = D'_{kl} - r_{ijk}$

Se C = \emptyset
 t = t + 1
 Voltar a III a).
ConjCombNaoDomin = {C}
// no final ConjCombNaoDomin $\neq \emptyset$ caso contrário não haveria solução
admissível para o problema por falta de recursos.

b) Combinações Seguintes

Para a atividade $a = |C|, \dots, 1$
 Se $j'(a) \neq J$
 $j'(a) = j'(a) + 1$
 C' = \emptyset
 Para $p = a+1, \dots, |C|$
 C' = C' U { (p, j'(p)) }
 //calcular D''_{kl} por alteração do modo da atividade a
 Para o recurso $k = 1, \dots, K$
 Para o momento $l = t, \dots, t + d_{aj}$
 $D''_{kl} = D'_{kl} - r_{aj'(a)k} + r_{aj'(a)+1k}$

Para a atividade $i \in E$

```

    Incluir = True
    j = 0
    Enquanto (j <= J) e (Incluir = True)
        j = j + 1
        Para o recurso k = 1, ..., K
            Para o momento l = t, ..., t + dij
                Se rijk > D''kl
                    Incluir = False
        Se Incluir = True
            j'(i) = j
            C' = C' U { (i, j'(i)) }
            Para o recurso k = 1, ..., K
                Para o momento l = t, ..., t + dij
                    D''kl} = D''kl} - rijk
    ConjCombNaoDomin = ConjCombNaoDomin U {C'}

```

IV. Determinação das Atividades escalonadas em t

Para cada Combinação C' em ConjCombNaoDomin
 Determinar o atraso I_c de acordo com as folgas correntes
 Seja C a Combinação do ConjCombNaoDomin com menor I_c
 Para toda a atividade i tal que (i, j'(i)) ∈ C
 j(i) = j'(i)
 t(i) = t
 Para o recurso k = 1, ..., K
 Para o momento l = t, ..., t + d_{ij(i)}}
 D<sub>kl} = D<sub>kl} - r_{ij(i)k}}
 V = V U { C }
 Para toda a atividade i ∈ C
 Se j(i) = 1
 Voltar a I.</sub></sub>

V. Determinação do CPM atualizado

Cálculo do CPM com a lista das atividades escalonadas (em V) e duração correspondente ao modo selecionado d_{ij(i)}}, com vista a atualizar as folgas de cada actividade para determinação dos atrasos.

3.2.2 Exemplo

Retome-se o projeto exemplo em que existem 2 modos de execução para cada atividade e apenas 1 tipo de recurso (com 6 unidades disponíveis).

São ilustradas as iterações da heurística de Boctor para o projeto.

$R_1 = 6$					
Atividade i	P_i	d_{i1}	r_{i11}	d_{i2}	r_{i21}
1	-	3	2	5	1
2	1	4	3	7	1
3	1	6	5	8	3
4	2,3	2	4	3	3

Tabela 5 - Projeto exemplo com notação

Em cada iteração é apresentado o momento de escalonamento (t), os recursos disponíveis em t (D_{kt}), as atividades escalonáveis (E), as combinações não dominadas (ConjCombNaoDomin) e as atividades escalonadas (V).

Iteração 1

t	D_{kt}	E	ConjCombNaoDomin	V
0	6	{1}	{{(1,1)}	{{(1,1)}

$$a_{11} = \max \{0, 3 - 3 - 0\} = 0, I_1 = 0$$

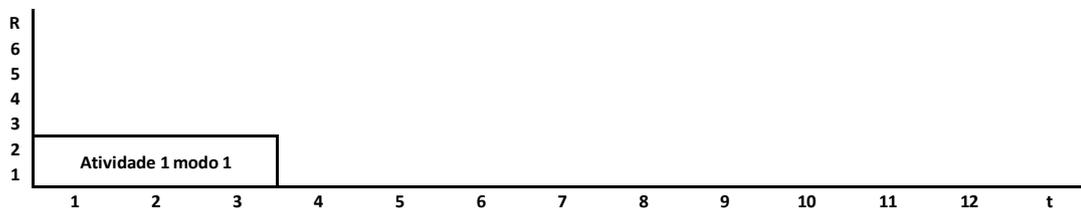


Figura 8 - Iteração 1 para o projeto exemplo

Iteração 2

t	D _{kt}	E	ConjCombNaoDomin	V
3	6	{2,3}	{(2,1),(3,2)}, {(2,2),(3,1)}	{(2,2),(3,1)}

$$a_{21} = \max \{0, 4 - 4 - 2\} = 0, a_{32} = \max \{0, 8 - 6 - 0\} = 2, I_1 = 2$$

$$a_{22} = \max \{0, 7 - 4 - 2\} = 1, a_{31} = \max \{0, 6 - 6 - 0\} = 0, I_2 = 1$$

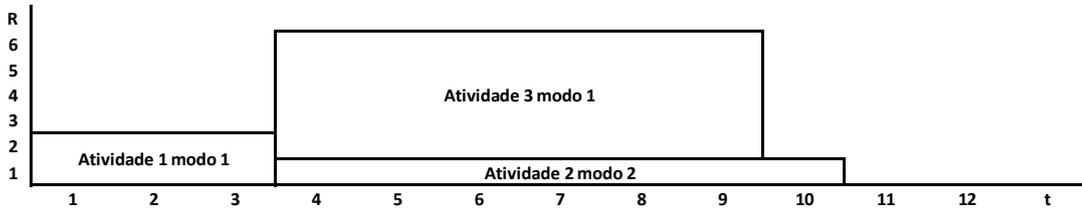


Figura 9 - Iteração 2 para o projeto exemplo

Iteração 3

t	D _{kt}	E	ConjCombNaoDomin	V
10	6	{4}	{(4,1)}	{(4,1)}

$$a_{41} = \max \{0, 2 - 2 - 0\} = 0, I_1 = 0$$

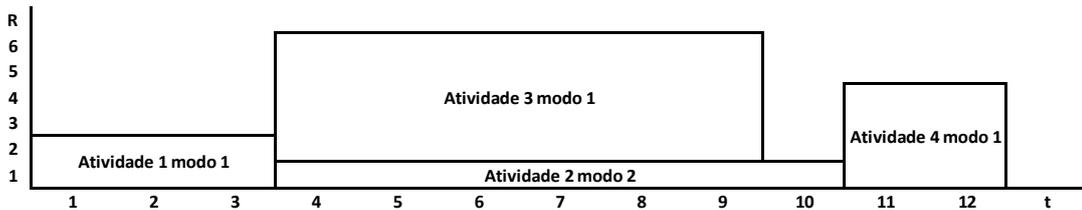


Figura 10 - Iteração 3 para o projeto exemplo

4 Conclusões

O interesse pelos problemas de escalonamento de projetos com recursos escassos e com vários modos de execução para cada atividade tem vindo a crescer significativamente nos últimos anos. Este interesse é fruto da necessidade que as empresas permanentemente têm de melhorar o seu desempenho e eficiência, na busca de uma maior competitividade.

Na prática, o MRCPSP responde a dificuldades comuns no âmbito da gestão de projetos, porém, dada a inerente complexidade associada a este problema, tem sido encarado como um verdadeiro desafio por parte dos investigadores.

Nesse sentido, o método de resolução aplicado durante o período de estágio na Quidgest para resolver o problema foi uma heurística. Os métodos heurísticos apresentam a grande vantagem de produzirem soluções em tempo útil, com sacrifício da garantia de otimalidade é certo, mas por vezes conduzem a soluções com valores aproximados muito razoáveis.

O grande desafio do estágio na Quidgest S.A. foi o de identificar uma necessidade na gestão de projetos da empresa e aplicar os conhecimentos de Investigação Operacional no desenho de uma solução, tendo para tal, um prazo de 3 meses. Foi um trabalho que exigiu igualmente o estudo e compreensão de todos os conceitos envolvidos e por fim a elaboração de uma ferramenta de trabalho em Visual Basic / Excel.

O software desenvolvido pode e deve ser melhorado no sentido de, cada vez mais, ir de encontro às necessidades da empresa. Para potenciar esta ferramenta, a característica estocástica deverá ser explorada, assim como, a

possibilidade de interrupção de atividades a meio da sua execução, o que muitas vezes acontece no âmbito dos seus projetos.

5 Referências Bibliográficas

- Alcaraz, J, Maroto, C e Ruiz, R. 2003. *Solving the multi-mode resource-constrained project scheduling problem with genetic algorithms*. Journal of the Operational Research Society 54, 614–626, 2003.
- Blazewicz, J, Lenstra, J e Rinnooy, A. 1983. *Scheduling subject to resource constraints: classification and complexity*. Discrete Applied Mathematics 5, 11-24, 1983.
- Boctor F. 1993. *Heuristics for scheduling projects with resource restrictions and several resource-duration modes*. International Journal of Production Research 31, 2547-58, 1993.
- Boctor F. 1996. *A new and efficient heuristic for scheduling projects with resource restrictions and multiple execution modes*. European Journal of Operational Research 90, 349–361, 1996.
- Bouleimen, K e Lecocq, H. 2003. *A new efficient simulated annealing algorithm for the resource-constrained project scheduling problem and its multiple mode version*. European Journal of Operational Research 149 (2), 268–281, 2003.
- Brucker, P e Knust, S. 2001. *Resource-constrained project scheduling and timetabling*. Lecture Notes in Computer Science 2079, 277–293, 2001.
- Brucker, P, et al. 1999. *Resource-constrained project scheduling: Notation, classification, models and methods*. European Journal of Operational Research 112, 3-41, 1999.
- Demeulemeester, E e Herroelen, W. 1996. *An efficient optimal solution procedure for the preemptive resource-constrained project scheduling problem*. European Journal of Operational Research 90 (2), 334–348, 1996.
- Demeulemeester, E, Reyck, B e Herroelen, W. 2000. *The discrete time/resource trade-off problem in project networks – A branch-and-bound approach*. IIE Transactions 32, 1059–1069, 2000.
- Elmaghraby. 1977. *Activity networks: Project planning and control by network models*. 1977.
- Hartmann, S e Briskorn, D. 2010. *A survey of variants and extensions of the resource-constrained project scheduling problem*. European Journal of Operational Research, 2010.
- Hartmann, S e Drexl, A. 1998. *Project scheduling with multiple modes: A comparison of exact algorithms*. Networks 32, 283–297, 1998.
- Hartmann, S. 2001. *Project scheduling with multiple modes: A genetic algorithm*. Annals of Operations Research 102, 111–135, 2001.
- Herroelen, W e Leus, R. 2005. *Project scheduling under uncertainty: Survey and research potentials*. European Journal of Operational Research 165 (2), 289–306, 2005.
- Józefowska, J, et al. 2001. *Simulated annealing for multi-mode resource-constrained project scheduling*. Annals of Operations Research 102, 137–155, 2001.

- Kelley, J e Walker, M. 1959.** *Critical-Path Planning and Scheduling*. Proceedings of the Eastern Joint Computer Conference, 1959.
- Kimms, A. 2001.** *Maximizing the net present value of a project under resource constraints using a lagrangian relaxation based heuristic with tight upper bounds*. Annals of Operations Research 102, 221–236, 2001.
- Klein, R e Scholl, A. 2000.** *PROGRESS: Optimally solving the generalized resource-constrained project scheduling problem*. Mathematical Methods of Operations Research 52 (3), 467–488, 2000.
- Kolisch, R e Drexl, A. 1997.** *Local search for nonpreemptive multi-mode resource-constrained project scheduling*. IIE Transactions 29, 987–999, 1997.
- Kolisch, R e Hartmann, S. 2006.** *Experimental investigation of heuristics for resource-constrained project scheduling: An update*. European Journal of Operational Research 174 23–37, 2006.
- Lova, A, Tormos, P e Barber, F. 2006.** *Multi-mode resource constrained project scheduling: Scheduling schemes, priority rules and mode selection rules*. Inteligencia Artificial 30, 69–86, 2006.
- Neumann, K. 1990.** *Stochastic project networks: Temporal analysis, scheduling and cost minimization*. Economics and Mathematical Systems 344, 1990.
- Neumann, K, Schwindt, C e Zimmermann, J. 2002.** *Recent results on resource-constrained project scheduling with time windows: Models, solution methods, and applications*. Central European Journal of Operations Research 10, 113–148, 2002.
- Nonobe, K e Ibaraki, T. 2001.** *Formulation and Tabu Search Algorithm for the Resource Constrained Project Scheduling Problem (RCPSP)*. Technical Report, Kyoto University, 2001.
- Özdamar, L. 1999.** *A genetic algorithm approach to a general category project scheduling problem*. IEEE Transactions on Systems, Applications and Reviews 29, 44–59, 1999.
- Özdamar, L e Ulusoy, G. 1994.** *A local constraint based analysis approach to project scheduling under general resource constraints*. European Journal of Operational Research 79, 287–298, 1994.
- Padman, R e Zhu, D. 2006.** *Knowledge integration using problem spaces: A study in resource-constrained project scheduling*. Journal of Scheduling 9 (2), 133–152, 2006.
- Patterson, J, et al. 1989.** *An algorithm for a general class of precedence and resource constrained scheduling problem*. Advances in Project Scheduling. Elsevier, Amsterdam, 3–28, 1989.
- Peteghem, V e Vanhoucke, M. 2010.** *A genetic algorithm for the preemptive and non-preemptive multi-mode resource-constrained project scheduling problem*. European Journal of Operational Research 201, 409–418, 2010.
- PMI. 2004.** *A Guide to the Project Management Body of Knowledge, Third Edition*. Pennsylvania : Project Management Institute, 2004. ISBN 1-930699-45-X.

- Ranjbar, M e Kianfar, F. 2007.** *Solving the discrete time/resource trade-off problem in project scheduling with genetic algorithms.* Applied Mathematics and Computation 191, 451–456, 2007.
- Ranjbar, M, Reyck, B e Kianfar, F. 2009.** *A hybrid scatter search for the discrete time/resource trade-off problem in project scheduling.* European Journal of Operational Research 193, 35–48, 2009.
- Russell, A. 1970.** *Cash Flows in Networks.* Management Science 16, 357- 373, 1970.
- Slowinski, R. 1980.** *Two Approaches to Problems of Resource Allocation among Project Activities – A Comparative Study.* The Journal of the Operational Research Society, 1980. Vol. 31.
- Sprecher, A e Drexl, A. 1998.** *Solving multi-mode resource-constrained project scheduling problems by a simple, general and powerful sequencing algorithm.* European Journal of Operational Research 107, 431–450, 1998.
- Sprecher, A, Hartmann, S e Drexl, A. 1997.** *An exact algorithm for the project scheduling with multiple modes.* OR Spektrum 19, 195–203, 1997.
- Talbot, F. 1982.** *Resource-constrained project scheduling with time-resource tradeoffs: The nonpreemptive case.* Management Science 28 (10), 1197–1210, 1982.
- Vanhoucke, M, Demeulemeester, E e Herroelen, W. 2001.** *On maximizing the net present value of a project under renewable resource constraints.* Management Science 47, 1113–1121, 2001.
- Verma, S. 2006.** *Exact methods for the preemptive resource constrained project scheduling problem.* Research and publication, 1-17, 2006.
- Zhu, G, Bard, J e Tu, G. 2006.** *A branch-and-cut procedure for the multimode resource-constrained project-scheduling problem.* Journal on Computing 18 (3), 377–390, 2006.

Anexo

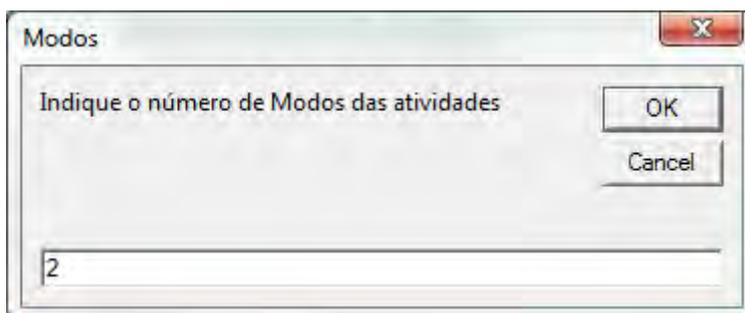
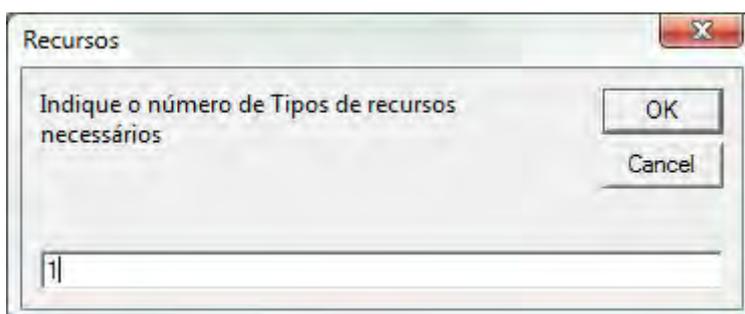
Manual de utilização

Para trabalhar com o software desenvolvido, é indispensável ter em atenção os seguintes procedimentos necessários à sua execução.

Depois de abrir o ficheiro Excel deverá clicar no botão

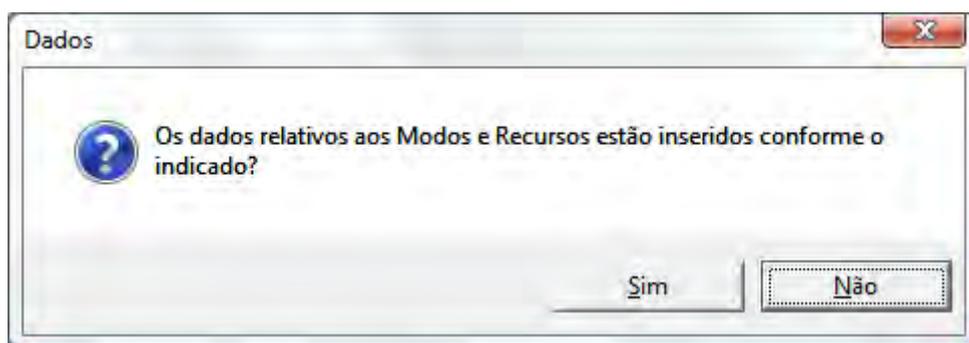
Boctor

Ser-lhe-á pedido para inserir o número de tipos de recursos distintos necessários ao projeto e o número de modos das atividades. Por uma questão de coerência, vamos prosseguir exemplificando com o projeto exemplo referido no relatório.



Será apresentado o seguinte cenário:

Nr Actividade	Descrição	Precedente(s)	Duração M1	Duração M2	Recurso1 M1	Recurso1 M2	ES	EF	LS	LF	Folga	Caminho Crítico
0	Fictícia Inicial						0	0	0	0	0	
1	Fictícia Final	0					0	0	0	0	0	1



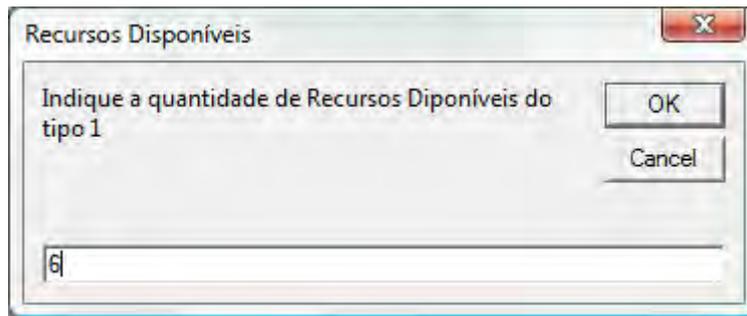
Deverá responder Não à *MsgBox*.

Seguidamente, deverá inserir os dados do projeto sem apagar a linha correspondente à Atividade Fictícia Inicial, ou seja, para o nosso exemplo deverá preencher os dados de forma idêntica à imagem que se segue. Note que se houver mais que uma precedente, estas devem ser divididas por uma vírgula.

Nr Actividade	Descrição	Precedente(s)	Duração M1	Duração M2	Recurso1 M1	Recurso1 M2	ES	EF	LS	LF	Folga	Caminho Crítico
0	Fictícia Inicial											
1		0	3	5	2	1						
2		1	4	7	3	1						
3		1	6	8	5	3						
4		2,3	2	3	4	3						

Clica mais uma vez em , insere o número de tipos de recursos e o número de modos das atividades e responde Sim à pergunta: Os dados relativos aos Modos e Recursos estão inseridos conforme o indicado?

Por fim indica a quantidade de unidades disponíveis para cada tipo de Recurso.



O resultado final é apresentado conforme a última imagem.

Duas linhas abaixo dos dados do projeto podem visualizar-se a solução encontrada pela heurística, isto é, a duração da atividade no modo escolhido, o consumo de recursos correspondentes e o tempo em que deve ser inicializada e finalizada.

O CPM atualizado também é apresentado duas linhas abaixo da solução final.

Nr Actividade	Descrição	Precedente(s)	Duração M1	Duração M2	Recurso1 M1	Recurso1 M2	ES	EF	LS	LF	Folga	Caminho Crítico
0	Fictícia Inicial											
1		0	3	5	2	1	0	3	0	3	0	1
2		1	4	7	3	1	3	7	5	9	2	
3		1	6	8	5	3	3	9	3	9	0	3
4		2,3	2	3	4	3	9	11	9	11	0	4
5	Fictícia Final	4					11	11	11	11	0	5

SOLUÇÃO				
Nr Actividade	Duração	Recursos Tipo 1	Tempo início	Tempo final
1	3	2	0	3
2	7	1	3	10
3	6	5	3	9
4	2	4	10	12

Novo CPM								
Nr Actividade	Descrição	Precedente(s)	Duração	ES	EF	LS	LF	Folga
1		0	3	0	3	0	3	0
2		1	7	3	10	3	10	0
3		1	6	3	9	4	10	1
4		2,3	2	10	12	10	12	0
5	Fictícia Final	4		12	12	12	12	0