

INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA
Área Departamental de Engenharia de Sistemas de Potência e Automação



AFECTAÇÃO DE UNIDADES TÉRMICAS CONSIDERANDO AS EMISSÕES POLUENTES

RUI JORGE RIBEIRO LAIA
(Bacharel em Energia e Sistemas de Potência)

Dissertação para a obtenção do grau de Mestre em
Engenharia Electrotécnica – ramo de Energia

Júri:

Presidente:

Professor Doutor Jorge Alberto Mendes de Sousa

Vogais:

Professor Doutor Victor Manuel Fernandes Mendes

Professor Doutor Mário Rui Melício da Conceição

Setembro de 2011

Tese realizada sob orientação de
Professor Doutor Victor Manuel Fernandes Mendes
Professor Coordenador com Agregação da Área Departamental de
Engenharia de
Sistemas de Potencia e Automação
INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

Resumo

Este trabalho incide sobre o planeamento de produção de unidades térmicas de geração de energia eléctrica num horizonte temporal de uma semana.

O objectivo deste trabalho é realizar a afectação das unidades térmicas considerando simultaneamente o custo de operação e o custo das emissões poluentes.

Abstract

This work focus on the planning production of thermal power plants for one week time horizon.

The goal of this work is to schedule the operation of thermal power plants, unit commitment, considering simultaneously the operation costs and poluent emissions costs.

Índice

1. Introdução	3
1.1 Motivação	3
1.2 Objectivos	6
1.3 Apresentação do trabalho	7
2. Estado da Arte.....	13
2.1 Preâmbulo	13
2.2 Dados Cronológicos	14
2.3 Evolução	48
2.4 Conclusões	56
3. Casos de Estudo	60
3.1 Afecção de uma Unidade Térmica Considerando o Custo do Combustível	64
3.2 Afecção de uma Unidade Térmica Considerando o Custo do Combustível e das Emissões Poluentes	74
3.3 Afecção de Unidades Térmicas Sujeitas a Acordo de Fornecimento Bilateral	80
4. Conclusões.....	95
Referências	98
AN1. Fundamentação Teórica.....	104
AN1.1 Programação Dinâmica	105

AN1.2 Programação Não Linear	111
AN2. Formulação do Problema	150
AN2.1 Função objectivo	151
AN2.2 Restrições	154
AN3. Exemplos Programação Dinâmica	158
AN4. Listagem do programa desenvolvido para os casos de estudo 1 e 2.....	174
AN5. Listagem do programa desenvolvido para os casos de estudo 3.....	185

Lista de Figuras e Tabelas

Figura 3.1: Diagrama de transição da Máquina de Estados.....	61
Figura 3.2: Custo de arranque de unidade térmica	63
Tabela 3.1: Tabela de estados da Máquina de Estados.....	64
Figura 3.3: Preço da energia e energia consumida em Portugal de 11 a 17 Abril de 2011	65
Figura 3.4: Estado e défice operacional da unidade térmica G1	68
Figura 3.5: Custo e facturação horária da unidade térmica G1	69
Figura 3.6: Preço da energia e potência da unidade térmica G1	70
Figura 3.7: Estado e custo da unidade térmica G1 com alteração do preço da energia	71
Figura 3.8: Custo e facturação horária da unidade térmica G1 com alteração do preço da energia.....	72
Figura 3.9: Preço da energia e potência da unidade térmica G1 com alteração do preço da energia.....	73
Figura 3.10: Energia fornecida e custo total em função de λ	76
Figura 3.11: Défice operacional e custo das emissões da unidade térmica G1 .	77
Figura 3.12: Défice operacional ponderado e custo ponderado atribuído às emissões da unidade térmica G1	78
Figura 3.13: Défice operacional em função do custo atribuído às emissões da unidade térmica G1	79

Tabela 3.2: Restrições das unidades térmicas.....	82
Tabela 3.3: Parâmetros das funções das unidades térmicas.....	83
Tabela 3.4: Estados iniciais das unidades térmicas.....	83
Tabela 3.5: Custo e energias fornecidas e contratada em função de λ	84
Figura 3.15: Energia contratada e energias fornecidas pelas unidades térmicas	84
Figura 3.16: Déficit operacional em função de λ	85
Figura 3.17: Potência contratada, potência fornecida e déficit operacional para $\lambda = 0$	87
Figura 3.18: Potência gerada por cada unidade térmica para $\lambda = 0$	88
Figura 3.19: Potência contratada, potência fornecida e déficit operacional para $\lambda = 0,5$	89
Figura 3.20: Potência gerada por cada unidade térmica para $\lambda = 0,5$	90
Figura 3.21: Potência contratada, potência fornecida e déficit operacional para $\lambda = 1$	91
Figura 3.22: Potência gerada por cada unidade térmica para $\lambda = 1$	92

Lista de Símbolos

I	Conjunto unidades térmicas
H	Conjunto centrais onde estão instaladas as unidades térmicas
u_{it}	Estado da unidade térmica i no instante t ; 1 significa ligada 0 significa desligada
NM	Número máximo permitido de unidade térmicas ligadas em simultâneo
RP_h	Número máximo permitido de unidade térmicas na Central h que podem ser ligadas (<i>crew constraint</i>)
p_{it}	Potência gerada pela unidade térmica i no instante t
\underline{p}_i	Potência mínima permitida da unidade térmica i
\bar{p}_i	Potência máxima permitida da unidade térmica i
D_t	Potência consumida no instante t
R_t	Reserva girante no instante t
UR_i	Taxa máxima de variação positiva da potência da unidade i
DR_i	Taxa máxima de variação negativa da potência da unidade i
$uminup_i$	Tempo mínimo de funcionamento da unidade i
$umindo_i$	Tempo mínimo de paragem da unidade i
$utcool_i$	Tempo arrefecimento total da unidade i
$uccool_i$	Custo arranque após $utcool_i$ da unidade i

$C_i(u_i, p_i)$	Custo do combustível da unidade i
a, b, c	Coeficientes da função custo do combustível
$E_i(u_i, p_i)$	Emissões poluentes da unidade i
α, β, γ	Coeficientes da função emissões poluentes
ω	Custo unitário das emissões poluentes
λ	Coeficiente ponderação entre custo do combustível e custo das emissões poluentes

Capítulo 1

Introdução

Resumo

É apresentada a relevância das unidades térmicas no abastecimento de energia eléctrica, o custo de produção e o impacte ambiental destas unidades. É também apresentada a forma como este trabalho está organizado.

1. Introdução

1.1 Motivação

1.2 Objectivos

1.3 Apresentação do Trabalho

1. Introdução

1.1 Motivação

As unidades térmicas de geração de energia eléctrica são actualmente responsáveis, segundo os dados da International Energy Agency para 2008, por 52,7 % do total de energia eléctrica produzida na Europa e por 67,5 % da energia produzida a nível mundial [44].

Ainda segundo os dados da International Energy Agency para 2008, os combustíveis mais utilizados nestas unidades são o carvão e o gás natural, sendo que as unidades que utilizam carvão asseguraram 25,8 % do total da energia produzida na Europa e 40,8 % no mundo e as unidades que utilizam gás asseguraram 24,0 % do total da energia produzida na Europa e 21,2 % no mundo. As unidades que utilizam como combustível os produtos derivados do petróleo asseguraram 2,9 % do total da energia produzida na Europa e 5,5 % no mundo, sendo a sua utilização cada vez mais reduzida devido ao custo crescente deste combustível.

Os custos de produção, também de acordo com a International Energy Agency, para uma taxa de desconto de 5 %, variam entre 25 e 50 USD/MWh para centrais que utilizam como combustível o carvão e entre 37 e 60 USD/MWh nas centrais a gás, sendo este custo tipicamente inferior a 55 USD/MWh.

Não são tomadas em consideração as unidades térmicas que utilizam como combustível a biomassa ou os resíduos porque estas unidades operam tipicamente de acordo com regras próprias que visam a promoção da produção de energia eléctrica a partir de fontes renováveis, não estando assim sujeitas à afectação. Em Portugal, são classificados como Produtores em Regime Especial e têm garantido o escoamento da totalidade da sua produção a preços definidos na legislação.

Mais uma vez segundo os dados da International Energy Agency para 2008, nos 27 estados da União Europeia, as emissões de dióxido de carbono por kWh gerado em unidades que utilizam como combustível o carvão foram 820 gCO₂/kWh enquanto que para as que utilizam gás este valor foi de 324 g CO₂/kWh [45].

As emissões de CO₂ resultantes da geração de energia eléctrica em centrais térmicas deverão ser limitadas devido às consequências ambientais, em particular devido ao efeito de estufa. Estas preocupações ambientais deram origem ao protocolo de Kyoto, tendo sido definidos valores limite para as emissões de CO₂ em 37 países industrializados, tendo como objectivo atingir uma redução de 5% destas emissões entre 2008 e 2012 tomando 1990 como ano de referência.

Foi instituído um mecanismo de comércio destas emissões, o que permite atribuir um custo e conseqüentemente reduzir o valor das emissões. De acordo

com os dados obtidos no Sistema Electrónico de Negociação de Direitos de Emissão de Dióxido de Carbono, o valor médio destas emissões, no primeiro semestre de 2011, foi de 12,3 €/MgCO₂ [46].

Há assim um conflito na geração da energia eléctrica a partir de centrais térmicas: por um lado procura-se obter energia a baixo custo, por outro lado, procura-se reduzir as emissões de CO₂, o que limita a utilização de algumas destas centrais ou condiciona a sua utilização e fonte de combustível.

O problema de afectação de unidades é a selecção das unidades térmicas que devem entrar em serviço e quando, por forma a atingir um ou mais objectivos pré-definidos, tal como a minimização do custo de operação ou das emissões poluentes, ou a maximização do lucro, garantindo simultaneamente a satisfação de alguns requisitos.

Têm sido, ao longo dos últimos 40 anos, desenvolvidas e ensaiadas técnicas que têm como objectivo a optimização da afectação das unidades térmicas.

Numa primeira fase, ainda sem recursos computacionais e não sujeitas a restrições dos mercados, esta afectação era realizada através de listas de prioridades baseadas no mérito das unidades, tipicamente o custo de produção.

A complexidade deste problema foi crescendo ao longo dos anos, com a contínua inclusão de novas restrições: pelo lado das unidades, temos como exemplos os tempos mínimos de paragem e arranque das centrais, restrições de pessoal, taxas de aumento de potência e outras; externamente às unidades,

temos, como exemplo, os limites impostos pelas redes de distribuição de energia e as restrições no fornecimento de combustível. O próprio objectivo também se alterou nos locais onde houve uma liberalização dos mercados, tendo passado de minimização dos custos para maximização dos resultados.

As restrições impostas pelas questões ambientais vêm adicionar complexidade ao problema, pois poderemos estar perante um problema com objectivos conflituosos: a maximização dos resultados e a minimização das emissões.

As técnicas desenvolvidas nos últimos quarenta anos aproveitam os crescentes recursos computacionais.

Esta dissertação é uma contribuição para os estudos de aplicação destas técnicas, tendo como motivação dar resposta ao problema da afectação de unidades térmicas.

1.2 Objectivos

É o objectivo deste trabalho entender a evolução das técnicas utilizadas no problema de afectação de unidades e a sua motivação, a ligação entre as técnicas utilizadas e os novos desafios levantados tanto pela introdução de novas limitações, que têm como origem uma cada vez mais detalhada descrição do funcionamento das centrais térmicas, conduzindo a estimativa de custos mais próximos da realidade, como pela consideração de factores externos, tais como limitações nas redes de transporte e distribuição de energia, limitações no

abastecimento de combustíveis ou novas regras de mercado e considerações ambientais.

1.3 Apresentação do trabalho

O trabalho é desenvolvido, para além desta introdução, em 2 capítulos e 4 anexos.

Capítulo 2 – Estado da Arte

São apresentados, de forma cronológica, uma colecção de artigos científicos sobre o problema de afectação de unidades, que permitem avaliar a evolução das técnicas utilizadas e das restrições impostas ao problema ao longo dos anos, assim como dos próprios objectivos, com a transição para um mercado liberalizado de energia, e a introdução de objectivos conflitantes, com a consideração das emissões poluentes.

Capítulo 3 – Casos de Estudo

São simulados três casos de estudo, ilustrando a aplicabilidade das técnicas referenciadas no trabalho explorando as mesmas pela alteração de algumas variáveis.

No primeiro caso de estudo é simulada a afectação de uma unidade térmica num regime de mercado liberalizado, considerando apenas o custo do combustível e o preço da energia.

No segundo caso de estudo é simulada a afectação da mesma unidade térmica do primeiro caso de estudo, incluído o custo das emissões poluentes.

No terceiro caso de estudo é simulada a afectação de um conjunto de três unidades térmicas, sujeitas a um acordo bilateral de fornecimento de energia, considerando o custo do combustível e o custo das emissões poluentes.

Capítulo 4 – Conclusões

São apresentadas as conclusões fundamentadas pelos artigos científicos referenciados no capítulo Estado da Arte e as conclusões da técnica de afectação de unidades utilizada no capítulo Casos de Estudo.

Anexo I – Fundamentação Teórica

São apresentadas duas técnicas fundamentais no problema de afectação de unidades, a programação dinâmica e a programação não linear.

A programação dinâmica permite planear a afectação das unidades ao longo do tempo de forma óptima, pela avaliação do resultado final dos diferentes caminhos, desde a origem ao final do período considerado. O caminho seleccionado será o que apresenta melhor resultado final.

Esta técnica está limitada na sua aplicação pela dimensão dos problemas, o que é conhecido como maldição da dimensionalidade.

A programação não linear permite, para cada possível combinação (estádio) de unidades em funcionamento dentro de cada intervalo de tempo, determinar o regime de funcionamento de cada unidade térmica, de forma a obter uma combinação de regimes de funcionamento que será o regime global óptimo dessa opção.

Anexo II – Formulação do Problema

É apresentada a função objectivo dum problema de Afectação de Unidades, onde se procura minimizar o custo do combustível e o impacte das emissões poluentes.

São também expressas as restrições a que estão sujeitas as unidades e introduzidas as variáveis que afectam o custo de operação devido aos custos de arranque.

Anexo III – Listagem de Programa de Programação Dinâmica

Listagem do programa utilizado nos dois primeiros casos de estudo.

Anexo IV – Listagem de Programa de Programação Dinâmica

Listagem do programa utilizado no terceiro caso de estudo.

Capítulo 2

Estado da Arte

Resumo

É apresentada a evolução do problema de Afectação de Unidades, em termos de utilização das técnicas utilizadas, do incremento da complexidade do problema devido ao aumento dos factores incluídos na sua análise e dos objectivos, devido à liberalização do mercado e da consideração do impacte ambiental.

2. Estado da Arte

2.1 Preâmbulo

2.2 Dados Cronológicos

2.3 Evolução

2.4 Conclusões

2. Estado da Arte

2.1 Preâmbulo

Na maioria dos sistemas de energia eléctrica, a satisfação dos consumos é satisfeita principalmente por unidades geração térmica. Várias estratégias de funcionamento são possíveis para satisfazer os consumos, que variam de hora a hora ao longo do dia. De entre essas estratégias que identificam a afectação das unidades é necessário individualizar as mais convenientes tendo em consideração quer factores económicos quer ambientais.

As técnicas utilizadas para a resolução dos problemas de planeamento da Afectação de Unidades produtoras de energia eléctrica evoluíram ao longo dos anos, atendendo aos desenvolvimentos significativos nos domínios da programação matemática e aos recursos computacionais, como o crescente aumento da capacidade de processamento.

Inicialmente este problema era resolvido utilizando listas de prioridades, obtidas de forma heurística. Segundo C.K. Pang [2], 1981, a lista de prioridades é uma das mais antigas e simples formas de planear a Afectação de Unidades, sendo ainda largamente utilizada no início dos anos 80 do século XX. Actualmente, este problema é resolvido utilizando as técnicas de relaxação Lagrangeana, programação dinâmica e até técnicas de inteligência artificial, como os algoritmos genéticos.

2.2 Dados Cronológicos

Em 1976, C.K. Pang [1] apresentou um método de programação dinâmica “truncated” destinado à Afectação de Unidades Térmicas para um período de até 48 horas. Esta afectação de Unidades Térmicas é realizada de forma a minimizar o custo total, que inclui o custo de produção e o custo de arranque e paragem. O custo de arranque de cada unidade é dependente do tempo em que a unidade esteve fora de serviço. São estudados vários condicionalismos de reserva girante e limitações de pessoal de operação nas operações de arranque e paragem de duas ou mais unidades na mesma instalação. A operação das unidades individuais deve também satisfazer as especificações de tempos mínimos de colocação em serviço e paragem das unidades. Esta afectação de Unidades utiliza uma lista de prioridades de forma a limitar (“truncate”) a procura de planeamentos economicamente satisfatórios. É possível aplicar um método de programação dinâmica se a cada passo forem retiradas as opções potencialmente não satisfatórias.

Em consequência, a afectação obtida para um sistema de teste com 17 unidades usa recursos computacionais admissíveis, tendo o autor indicado que obteve um custo total de produção mínimo. O método foi programado para ser utilizado num computador de controlo. O estudo dos resultados obtidos revela que podem ser atingidas reduções significativas no custo do combustível.

Em 1981, C.K. Pang [2] comparou o desempenho de quatro métodos de Afectação de Unidades Térmicas, três dos quais baseados em programação

dinâmica. É também apresentada a modelização das restrições na transferência de energia entre diferentes áreas por redes de fluxo linear, que tem em conta as limitações no transporte de energia.

Nesta comparação, para assegurar resultados realísticos foram utilizados dois cenários, tendo no máximo 96 unidades: o Cenário “System C” do EPRI (Electric Power Research Institute) e uma rede da região central dos EUA. Os quatro métodos foram utilizados para estabelecer o potencial na redução de custos e os necessários recursos computacionais.

O primeiro método, “Complete Priority Order (CPO)”, baseia-se inteiramente numa lista de prioridades. Este processo simula o planeamento típico realizado por um operador. Os dois métodos seguintes, “Dynamic Programming – Sequential Combinations (DP-SC)” e “Dynamic Programming – Truncated Combinations (DP-TC)” são métodos de programação dinâmica baseados em listas de prioridades. O último método, “Dynamic Programming – Sequential/ Truncated Combinations (DP-STC)” é uma combinação dos métodos DP-SC e DP-TC.

Esta comparação demonstrou que a utilização destes métodos de programação dinâmica permitem atingir reduções significativas nos custos, mas não garantem que se obtenha o planeamento óptimo.

A determinação da lista de prioridades (ordem de mérito) não é crítica quando se utiliza o método DP-TC. No entanto, os recursos computacionais necessários poderão ser excessivos.

Para problemas de Despacho, o melhor método é o DP-SC. Para problemas de Afecção de Unidades, o DP-TC é uma boa alternativa, podendo mesmo ser usada para gerar a lista de prioridades. Independentemente da técnica utilizada, o planeamento deverá ser actualizado à medida que as condições mudam.

A modelização das restrições na transferência de energia entre diferentes áreas traz um considerável valor acrescentado no âmbito de se obter afecções mais realistas. A utilização dos recursos computacionais é facilmente justificável quando existem problemas de contingências ou funcionamento em ilhas.

Ainda o autor desta comparação conclui que existe a necessidade de pesquisas adicionais para desenvolver metodologias para geração de listas de prioridades. A definição desta metodologia deve incluir as restrições intangíveis que são incluídas com base em processos heurísticos.

Em 1983, A.I. Cohen [3] apresentou um novo método para resolução do problema de Afecção de Unidades, baseado em técnicas de partição e avaliação (branch-and-bound).

Este método incorpora custos de arranque dependentes do tempo anterior à paragem, procura e restrições de reserva girante e restrições de tempos mínimos de arranque e paragem. Não requer uma lista de prioridades e pode ser expandido de forma a poder incluir restrições de reserva probabilística.

O autor faz um ponto de situação sobre as técnicas utilizadas até 1983:

- Lista de prioridades e métodos heurísticos
- Programação dinâmica

- Programação Inteiros
- Relaxação Lagrangeana

A lista de prioridade especifica a ordem de arranque e paragem das unidades térmicas. A decisão de colocar uma unidade em serviço num dado instante era baseada em métodos heurísticos. A programação dinâmica era a técnica mais usada por diversos autores nesta época. Devido às limitações computacionais, estes métodos assumiam parcialmente listas de prioridades. Foi também utilizada a programação inteira, mas a formulação dos problemas de Afectação de Unidades para sistemas típicos é muito complexa, limitando a sua aplicação a pequenos sistemas. Na altura a técnica de Relaxação Lagrangeana começava a ser utilizada, tendo sido anunciado a resolução de um sistema com 250 unidades utilizando esta técnica.

O novo algoritmo é baseado em técnicas de partição e avaliação e a sua metodologia diferia da maioria das técnicas então usadas pelo facto de:

- Evitar o recurso à listas de prioridade
- Permitir custos dependentes do tempo
- Permitir considerar a variação aleatória do consumo e paragens por avarias das unidades

O método baseia-se na suposição que as unidades com custos de arranque terão no máximo um único arranque num período de 24 horas. Esta hipótese aplicava-se à maioria das operadoras dos EUA pois a sua carga diária tem um forte pico ou dois grandes picos próximos um do outro. São permitidas vários

arranques diários às unidades com custos de arranque nulos ou desprezáveis. As variáveis de decisão são as horas de arranque e paragem das unidades e o seu nível de produção.

O autor demonstrou que o algoritmo pode incluir uma restrição para reserva probabilística que inclui as consequências na reserva das paragens aleatórias das unidades e a incerteza no consumo de energia previsto.

O algoritmo, com uma restrição para reserva determinística, foi utilizado para resolver uma rede com 18 unidades num período de 24 horas, garantindo um desvio máximo do óptimo de 0,7%, tendo demorado a sua execução 67 segundos num sistema VAX11/780.

Em 1983, A. Merlin [4] apresentou o novo método utilizado pela EDF (Électricité de France), a Relaxação Lagrangeana.

Em relação à anterior utilização deste método, onde estava limitada a fornecer inputs para os métodos de partição e avaliação, o autor demonstrou que, utilizando alguns desenvolvimentos deste método, este pode resolver problemas de Afectação de Unidades de grande escala.

Em 1989, F.N. Lee [5] apresentou um método desenvolvido para a OG&E (Oklahoma Gas & Electric Co.), baseado em restrições de consumo de combustível, para um problema de Despacho. A OG&E tinha restrições significativas nesta área, incluindo contratos de fornecimento de gás do tipo “take or pay” e limitações associadas com o sistema de distribuição de gás, que

impunha limites máximos e mínimos no fornecimento às unidades térmicas. Estas restrições são directamente consideradas no problema de Afectação de Unidades. O método utilizado é semelhante à Relaxação Lagrangeana, tendo sido adicionado um procedimento de afectação sequencial que verifica continuamente a carga pedida e força uma solução admissível.

O método provou ser eficiente em termos de convergência e necessidade de recursos computacionais, tendo produzido consistentemente boas soluções.

Em 1991, C.C. Su [6] propôs um método baseado em “fuzzy logic” para resolver o problema de Afectação de Unidades, aplicável a um período entre um dia e uma semana.

Este método tem como benefício incluir a incerteza na previsão horária dos consumos no modelo, sendo esta incerteza tratada como uma regra “fuzzy”. São também tratados como regras “fuzzy” os factores não precisos, tais como as necessidades de reserva girante. Estas regras fuzzy, tanto para o objectivo como para as restrições, são expressas em funções de pertença.

As outras restrições que afectam o problema, tais como tempos mínimos de serviço e paragem e limitações de pessoal, não havendo incerteza na sua determinação, são tratados segundo os métodos da programação dinâmica convencional.

O método foi testado utilizando a rede de Taiwan, constituída por 6 unidades nucleares, 48 unidades térmicas e 44 unidades hídras. O sistema destina-se à Afectação de Unidades térmicas e assume que a optimização das unidades

hídricas foi realizada separadamente. Verificou-se que o método é muito eficiente, necessitando no entanto de mais recursos computacionais do que os métodos de programação dinâmica.

Em 1992, F.N. Lee [7] apresentou um método de Afectação de Unidades para sistemas interligados, que era uma extensão do eficiente “commitment utilization factor (CUF)”, aplicável a sistemas isolados.

A Afectação de Unidades em sistemas interligados pode ser realizada a vários níveis. No primeiro nível, é realizada uma Afectação de Unidades para cada área e o despacho é depois realizado para estas áreas interligadas. Num segundo nível, tanto a Afectação de Unidades como o despacho são determinados de forma conjunta para as diferentes áreas interligadas. O método é aplicado a esta segunda situação.

A Afectação de Unidades para áreas interligadas era anteriormente baseada em índices de custos médios de plena carga. Os estudos mostraram que este método conduzia a soluções longes das soluções óptimas para cada área, sendo que este desvio é incrementado quando são requeridas restrições no transporte de energia.

As restrições no transporte são consideradas no método proposto por F.N. Lee. O método demonstrou ter como vantagens o facto de oferecer uma implementação única para redes isoladas ou interligadas, produzir resultados eficientes com baixos recursos computacionais e é de fácil implementação,

podendo produzir listas de prioridades que podem ser utilizados pelos programas baseados neste método.

Em 1993, C. Wang [8] propôs um método que incluía os limites nas taxas de variação da carga dos geradores no arranque e paragem na determinação da Afectação de Unidades.

Os estudos produzidos até 1993 consideravam a energia gerada pelos geradores como uma função escada e que a energia produzida podia ser ajustada instantaneamente. Embora estas premissas permitissem simplificar significativamente o problema, elas não reflectem o processo de geração de energia. Consequentemente, introduzem um afastamento das decisões óptimas. Com a implementação de restrições nas taxas de variação de potência o problema tem uma modelização dinâmica, obrigando ao uso da programação dinâmica, que é a ferramenta adequada para este problema. Para contornar o maior problema da programação dinâmica, a maldição da dimensionalidade, exigindo significativos recursos computacionais, são numa primeira fase utilizadas técnicas de inteligência artificial, no caso redes neuronais, para produzir soluções que incluem todas as restrições, excluindo as das taxas de variação de potência. Na segunda fase é utilizada programação dinâmica para incluir as restrições dessas taxas de variação.

O processo foi ensaiado numa rede composta por 26 unidades térmicas e provou ser muito rápido e capaz de gerar resultados satisfatórios.

Em 1994, D. Dasgupta [9] aplica os algoritmos genéticos à determinação da Afecção de Unidades para o período de um dia.

Segundo o autor, os métodos com maior sucesso na altura eram os baseados em programação dinâmica, mas requeriam grandes recursos computacionais. Alguns pesquisadores tinham sugerido técnicas que combinavam a programação dinâmica com inteligência artificial, no entanto estes novos métodos requeriam uma grande interação com o operador, o que é um problema e um grande consumo de tempo mesmo em redes de média dimensão.

O método, ainda em avaliação, demonstrou ser eficiente embora tivessem faltado ensaios em sistemas de grandes dimensões. Apresenta como desvantagem os recursos computacionais necessários, mas como este método pode ser implementado em execução paralela, esta desvantagem foi considerada como podendo ser eliminada.

Em 1994, D. Zhai [10] descreveu um método para analisar o risco resultante da incerteza da carga na Afecção de Unidades, que se traduz na probabilidade de afectar uma capacidade insuficiente para compensar falhas nas unidades e/ou variações não previstas na carga.

Este risco é avaliado utilizando o modelo de Markov para as falhas nas unidades e um modelo de carga Gauss-Markov para a incerteza da carga. É essencial, para uma avaliação estocástica do risco, uma política de replaneamento da Afecção de Unidades, tendo o autor apresentado uma dessas políticas.

Dado um conjunto de unidades de produção e uma curva de carga determinística, o objectivo dos métodos para determinação da Afectação de Unidades é minimizar o custo operacional, sendo que a potência total afectada deve ser maior ou igual à carga mais um valor para reserva girante. Esta reserva girante tem como finalidade a compensação de aumentos bruscos da carga ou paragens intempestivas das unidades de produção. Na prática, o valor da reserva girante era determinada pelos operadores por métodos heurísticos. Pressões do mercado encorajaram os operadores a operar os seus sistemas de forma mais económica e como resultado alguns operadores começaram a operar os seus sistemas próximos do “limite”, realçando o tradicional compromisso entre maior risco/menor custo de operação e menor risco/maior custo de operação. Para se tomar uma decisão fundamentada neste compromisso é necessário determinar o risco associado a um determinado valor de reserva girante.

Historicamente, o principal critério para determinar o valor da reserva girante era de que este valor deveria ser maior ou igual que a capacidade da maior unidade afectada. Este método não tomava em conta as interligações entre redes e outros parâmetros do sistema, pelo que foram desenvolvidos métodos probabilísticos para produzir índices mais quantitativos do risco.

O método descrito permite definir o valor da reserva girante para não se ultrapassar um risco máximo pré-determinado, função das características de falha das unidades e da incerteza da variação da carga.

Em 1994, A. A. El-Keib [11] avaliou o impacto das restrições ambientais impostas em 1990 nos EUA, através do “Clean Air Act of 1990” que impunha limites nas emissões de SO_2 e NO_x geradas pelas unidades térmicas, no problema de Despacho.

De forma geral, a principal preocupação dos operadores era, nos problemas de despacho, de cariz económico. Devido às limitações impostas pelos novos limites, os operadores foram obrigados à tomada de decisões difíceis, pois tinham que cumprir estes limites de emissões mantendo a melhor opção exequível que permita a minimização do custo operação.

Tinham sido propostas várias estratégias de Despacho para um problema simultaneamente económico e ambiental, mas nenhuma das estratégias apresentadas até à altura reflectia os requisitos exactos da legislação.

Este autor apresentou nesta avaliação duas formulações de despacho económico. A primeira, baseada em Relaxação Lagrangeana, satisfaz as restrições ambientais em cada intervalo de despacho de forma a garantir os valores limite anuais. A segunda tira partido duma reserva de geração de compensação, denominada “compensating generation provision”, a qual está relacionada com aspectos específicos da legislação que são descritos pelo autor.

De acordo com o autor, ambas as soluções apresentadas são simples e utilizam as técnicas de despacho económico convencional apenas com ligeiras alterações.

Em 1995, S.J. Wang [12] propôs um novo método de Despacho baseado em Relaxação Lagrangeana que incluía restrições na transmissão de energia e ambientais.

Neste método, as restrições do sistema, tais como a carga, reserva girante, capacidade de transmissão e as restrições ambientais são relaxadas utilizando multiplicadores Lagrangeanos e são adicionados à função objectivo, que se denomina então de função Lagrangeana, factores quadráticos de penalização associados com a carga.

Os testes realizados demonstraram que o algoritmo desenvolvido é rápido, eficiente e robusto, produzindo resultados razoáveis em sistemas de dimensão normal.

Em 1996, S.A. Kazarlis [13] apresentou uma solução baseada em Algoritmos Genéticos para o problema de Afectação de Unidades.

A implementação simples desta técnica permite obter soluções algo próximas da óptima, mas geralmente não permite obter uma convergência para a solução óptima. No entanto, o autor relata resultados satisfatórios obtidos utilizando a técnica “Varying Quality Function” e incluindo operadores específicos no problema.

O autor apresenta também uma comparação entre os Algoritmos Genéticos com a Programação Dinâmica e a Relaxação Lagrangeana. Para um sistema com 10 unidades, as soluções obtidas pelas três técnicas, Algoritmos Genéticos, Programação Dinâmica e Relaxação Lagrangeana, foram idênticas. Segundo o

autor, em alguns testes não reportados no artigo, os Algoritmos Genéticos obtiveram melhores soluções que a Programação Dinâmica devido às dificuldades apresentadas por esta técnica em modelizar restrições dependentes do tempo e restrições do sistema de acoplamento “coupling constraints”, tais como restrições de reserva e balanço da rede. Para sistemas de maior dimensão, a comparação foi feita apenas com as soluções obtidas através de Relaxação Lagrangeana, devido aos excessivos recursos necessários para apresentar soluções completas utilizando Programação Dinâmica. Foi verificado que para sistemas com mais de 60 unidades, as soluções obtidas através de Algoritmos Genéticos são melhores que as produzidas utilizando a Relaxação Lagrangeana. Para sistemas com dimensões de 20 e 40 unidades as soluções obtidas pela Relaxação Lagrangeana encontram-se entre o limite inferior e limite superior das soluções obtidas por Algoritmos Genéticos.

Segundo o autor, os Algoritmos Genéticos têm como vantagem a facilidade de modelizar sistemas com restrições dependentes do tempo e restrições de sistema, e como desvantagem o não ser possível garantir que se obtém a melhor solução. No entanto, a maior diferença entre as soluções produzidas pelos Algoritmos Genéticos foi de 0,74% para o sistema com 10 unidades. Outra desvantagem é o elevado tempo de execução. Foi antevisto que estas duas desvantagens viessem a ser eliminadas pelo progresso na computação paralela.

Em 1997, C. Li [14] apresentou um novo método para a resolução do problema de planeamento da Afectação de Unidades produtoras de energia eléctrica, baseado na desafectação das unidades. Partindo duma situação inicial em que todas as unidades disponíveis estão afectas, é atingida a desafectação de uma unidade de cada vez utilizando programação dinâmica de acordo com os critérios económicos especificados. O processo de desafectação continua até que não é possível uma maior redução do custo total ou a afectação das unidades permanece inalterável em duas interacções consecutivas sem qualquer violação da restrição da reserva girante.

O método foi comparado com a relaxação Lagrangeana e com o método sequencial apresentado em 1989 por F.N. Lee [5], tendo-se verificado que apresenta excelentes resultados em termos de redução de custos, rápida computação e convergência estável.

Em 1998, A.H. Mantawy [15] apresentou o “Simulated Annealing Algorithm – SAA” para o problema de planeamento de Afectação de Unidades.

Em termos físicos, “annealing” refere-se ao processo de aquecer um sólido a temperaturas elevadas seguido de arrefecimento suave, decrescendo a temperatura ambiente em passos. Em cada passo a temperatura é mantida constante o tempo suficiente para que o sólido atinja o equilíbrio térmico.

Segundo o autor, as técnicas utilizadas até então podem ser divididas em três categorias:

- Métodos clássicos de optimização, tais como programação dinâmica, programação inteiros, relaxação lagrangeana e partição e avaliação
- Lista de prioridades e métodos heurísticos
- Métodos de inteligência artificial, tais como redes neuronais, algoritmos genéticos, “expert systems”, pesquisa tabu e “simulated annealing”

Os métodos clássicos estão bem documentados na literatura como meios directos de resolução do problema de planeamento de Afectação de Unidades. Alguns destes métodos dão bons resultados, tal como a relaxação lagrangeana, enquanto outros, como a programação dinâmica, apresentam problemas de dimensão, maldição da dimensionalidade, especialmente em sistemas de grande escala.

Os métodos heurísticos conduzem à obtenção de soluções não óptimas devido à pesquisa incompleta do universo de soluções e à falta da prova matemática de que se encontrou a solução óptima.

Os métodos de inteligência artificial pareciam promissores e estavam ainda em evolução. As redes neuronais e algoritmos genéticos eram inspirados em processos biológicos enquanto que o “simulated annealing (SA)” deriva das ciências dos materiais.

O SAA é uma boa técnica para resolver problemas de optimização combinatória. Embora o SAA tenha como desvantagem necessitar de longos tempos de CPU, tem algumas vantagens, tais como:

- Não depende fortemente da escolha da solução inicial para encontrar uma boa solução
- Não necessita dum modelo matemático complexo do problema em estudo
- Começa com qualquer solução inicial e tenta melhorá-la. Esta característica pode ser utilizada para otimizar uma solução não óptima resultante de métodos heurísticos ou outros métodos que produzam soluções não óptimas
- Está teoricamente provado que converge para a solução óptima
- Não necessita de grandes quantidades de memória para proceder à computação

A técnica “Simulated Annealing” introduzida em 1982 foi aplicada ao problema de planeamento de Afectação de Unidades térmicas em 1990, por F. Zhuang.

O artigo introduz uma metodologia para determinar o parâmetro inicial de controlo segundo o qual todas as soluções são virtualmente aceites e implementava novas regras para a geração de soluções exequíveis, o que resulta em considerável menor necessidade de recursos computacionais.

São testados três sistemas e os resultados são comparados com os obtidos por relaxação lagrangeana, programação inteiros e pela SAA introduzida por F. Zhuang. Os resultados demonstram que se obtêm ganhos entre 0,79 e 2,1%.

Em 1998, S. Sen [16] publicou um estudo no qual é apresentado o problema do planeamento de Afectação de Unidades e discute várias considerações, funções

objectivo e diferentes técnicas para resolução do problema, fazendo um ponto de situação das técnicas utilizadas. Discute também as tendências na pesquisa de técnicas para o de Afectação de Unidades.

Define como funções objectivo a minimização do custo total de operação, a minimização de emissões e a maximização da fiabilidade e segurança, embora geralmente este terceiro objectivo seja considerado uma restrição. O segundo objectivo pode também ser considerado como uma restrição, mas caso seja considerado um objectivo, então o planeamento de Afectação de Unidades é um problema de optimização multiobjectivo, com funções objectivo conflituosas.

O estudo constata que os métodos mais eficientes são a programação dinâmica com busca restrita, a relaxação lagrangeana aumentada e o “Simulated Annealing”, sendo que na altura se dava um ênfase particular aos algoritmos genéticos.

O estudo conclui também que a maioria dos operadores ainda usavam processos heurísticos devido à sua simplicidade e facilidade de implementação.

Em 1998, A.H. Mantawy [17] utilizou o método pesquisa tabu no planeamento de Afectação de Unidades.

Este método procura evitar o perigo de cair em soluções óptimas locais incorporando uma estrutura de memória que proíbe ou penaliza alguns movimentos que levariam a soluções já visitadas. Introduce também novas regras para geração de soluções exequíveis. Os pontos fortes deste método são:

- Encontra uma solução, eventualmente boa, que não depende fortemente da escolha da solução inicial
- Não necessita dum modelo matemático complexo do problema em estudo

Foram resolvidos vários exemplos para testar o método desenvolvido, tendo-se obtido resultados melhores do que os reportados na literatura.

Em 1999, A. Rudolf [18] propôs os algoritmos genéticos para o planeamento de Afectação de Unidades num sistema com unidades hídricas e térmicas.

O método previa a utilização dos algoritmos genéticos para o problema da Afectação de Unidades e a relaxação lagrangeana para o problema de despacho económico.

As restrições temporais das unidades térmicas e as restrições das turbinas e bombas das centrais hidroeléctricas foram embebidas em código binário que representam os estados das unidades. A fim de poupar recursos computacionais, o problema de despacho apenas é executado se a solução de planeamento de Afectação de Unidades encontrada satisfaz todas as restrições.

Os resultados das simulações mostram que para garantir resultados satisfatórios e uma rápida convergência é necessário otimizar os parâmetros dos algoritmos genéticos, o que é difícil e depende do sistema em estudo, mas é possível obter um método de simples implementação e que permite atingir soluções satisfatórias num curto tempo de computação.

Em 1999, B. Hobbs [19] analisou o impacto da optimização na previsão da carga no problema de Afectação de Unidades.

O erro na previsão da carga conduz a planeamentos não óptimos de Afectação de Unidades.

Se a carga for inferior à previsão, haverá um aumento do custo devido a:

- Eventual afectação desnecessária de unidades, aumentando custos de combustível e eventualmente custos de manutenção.
- Possível aquisição de energia não necessária ou possível perda de oportunidade de venda de energia.
- Eventual geração de energia a partir de centrais hídricas, energia esta que poderia ter um maior valor mais tarde.
- Eventuais interrupções de fornecimento ou chamada de controlos de carga, com consequências nos consumidores e redução da facturação.
- Eventual cotação elevada do preço da energia, com efeitos negativos nas vendas.

Se, por outro lado, a carga for superior à previsão, haverá um aumento do custo devido a:

- Recursos disponibilizados poderão não ser suficientes para fazer face à restrições de reserva, tais como a reserva girante, pondo em risco o sistema.
- Eventual necessidade de geração energia de forma não económica ou aquisição de energia a preços elevados para fazer face ao aumento de carga. Como alternativa, poderão ser necessárias eventuais interrupções

de fornecimento ou chamadas de controlos de carga, com consequências nos consumidores e redução da facturação.

- Podem ter sido tomados compromissos de venda de energia a um preço inferior ao do custo para o operador
- Eventual cotação baixa do preço da energia, com aproximação aos valor de custo.

A partir das simulações efectuadas com os dados disponibilizados por dois operadores, o estudo conclui que uma redução de 1% do MAPE (mean absolute percentage error - percentagem do módulo do erro médio) leva a uma redução de custos entre 0,3 e 0,5 % quando o MAPE se encontra entre os 3 e 5%.

Em 1999, A.H. Mantawy [20] integrou várias técnicas utilizadas para o planeamento de Afectação de Unidades num único algoritmo.

O núcleo do algoritmo é baseado nos algoritmos genéticos. A técnica de pesquisa tabu é utilizada para gerar novos membros da população na fase de reprodução dos algoritmos genéticos e o “Simulated Annealing” é utilizado para acelerar a convergência do algoritmo genético pela sua aplicação no teste a todos os membros da população.

A implementação do algoritmo genético difere em três aspectos das outras implementações de algoritmos genéticos: é codificado como uma combinação entre representação binária e decimal, reduzindo assim a necessidade de recursos computacionais, tempo e memória; a função objectivo resume-se ao custo total não sendo incluídas quaisquer penalizações; para melhorar a

pesquisa fina de soluções locais é introduzido um operador de mutação baseado em pesquisa tabu.

Para comparar com outros métodos, foram resolvidos três exemplos retirados da literatura. Os resultados são superiores aos obtidos recorrendo à relaxação lagrangeana e à programação de inteiros, sendo também superiores aos obtidos pela utilização individual de algoritmos genéticos, “Simulated Annealing” e pesquisa tabu.

Em 2000, C.P.Cheng [21] combinou os algoritmos genéticos com a relaxação lagrangeana.

A relaxação lagrangeana produz uma solução rápida mas pode ter como desvantagens a convergência e a qualidade das soluções apresentadas. O método proposto incorpora os algoritmos genéticos na relaxação lagrangeana para actualizar os multiplicadores lagrangeanos, melhorando o desempenho da relaxação lagrangeana na resolução de problemas de optimização combinatorios, tais como o planeamento de Afectação de Unidades.

Os resultados obtidos em dois casos estudados, um dos quais com 100 unidades, e a sua comparação com os resultados obtidos com relaxação lagrangeana e algoritmos genéticos, mostram que a técnica de incorporação dos algoritmos genéticos na relaxação lagrangeana apresenta melhores resultados, sendo uma técnica particularmente atractiva para sistemas de grande dimensão.

Em 2000, C. Richter [22] apresentou uma formulação para o problema do planeamento de Afectação de Unidades baseada em aspectos económicos.

A reestruturação do mercado da energia eléctrica nos EUA, que tinha como objectivo promover a competitividade, levava à necessidade de actualização dos algoritmos para planeamento de Afectação de Unidades que procuravam minimizar custos enquanto satisfaziam toda a carga.

Deixando de haver garantia, para as empresas produtoras de energia, de serem as únicas fornecedoras do mercado, a previsão da carga a satisfazer torna-se mais difícil. O desaparecimento da obrigação de garantir a produção de energia que satisfaça as necessidades atenua a restrição da procura.

São utilizados algoritmos genéticos para a resolução deste problema, sendo que não se procura minimizar o custo mas sim maximizar o lucro. O algoritmo proposto permite também prever os preços mais prováveis da energia, sendo que a execução repetida do algoritmo permite ao operador verificar a sensibilidade do planeamento a variações das entradas.

Em 2002, K. S. Swarup [23] utilizou os algoritmos genéticos para resolver um problema de Afectação de Unidades e de despacho económico.

A decisão de Afectação de Unidades determina quais as unidades que estarão em serviço a cada hora durante o horizonte temporal considerando a capacidade necessária, incluindo a reserva girante, e as restrições de arranque e paragem das unidades. O despacho económico trata da distribuição da carga e da reserva girante pelas unidades em serviço durante cada hora específica de operação.

Estas duas tarefas estão inter-relacionadas, pelo que ambas devem ser incluídas na formulação do problema, o qual procura obter a solução de menor custo para o horizonte temporal em questão. Adicionalmente, as decisões individuais da Afectação de Unidades e de despacho económico são conflituosas, pelo que a satisfação de uma restrição resulta na violação de uma outra. O objectivo é encontrar, se não a melhor, um conjunto de soluções próximas da melhor.

O problema inclui as restrições de tempos mínimos de paragem e arranque e empregou uma nova estratégia para a representação dos cromossomas e codificação do espaço de procura de soluções, que se revelou eficiente e capaz de tratar sistema de grande escala. O balanço de potência é verificado antes da operação dos algoritmos genéticos, o que garante soluções exequíveis durante cada passo da simulação.

Em 2002, C. Palanichamy [35] abordou o problema de Despacho considerando as emissões poluentes, em particular a inversão da temperatura, tendo proposto uma abordagem denominada “day-night weather-based” que minimiza o custo total da geração de energia tendo em consideração as restrições ambientais, não sendo necessário desenvolver um modelo climático.

Segundo o autor, as tecnologias alternativas de geração e armazenamento de energia consideradas suficientemente desenvolvidas são amplamente utilizadas nos sistemas de potência de forma económica e ambientalmente correcta. As fontes renováveis de energia permitem comercializar a energia com reduzidos

impactes ambientais, melhoria da segurança do abastecimento e outros benefícios directos e indirectos. As tecnologias de armazenamento reduzem a necessidade de utilizar centrais térmicas com baixo rendimento em períodos de pico do consumo. A energia armazenada pode substituir as unidades de baixo rendimento afectas a reserva girante que utilizam combustível fóssil para responder aos picos na carga. Para carregar os sistemas de armazenagem pode ser utilizada energia gerada em unidades com alto rendimento ou turbinas eólicas, sendo possível obter uma redução substancial no consumo de combustível e das emissões para a atmosfera.

As fontes renováveis de energia e os sistemas de armazenagem são integrados com as centrais que utilizam combustíveis fósseis para responder à carga variável diária. O despacho é baseado nas condições climáticas “dia-noite” obtidas dos sistemas de informação meteorológicos, o que resulta numa concentração controlada de poluentes na vizinhança das fontes poluidoras, com um custo atractivo. A vantagem desta abordagem é de não incluir na função objectivo um modelo climático, sendo seleccionados algoritmos de despacho de acordo com as condições climatéricas, evitando uma excessiva complexidade matemática na formulação e na metodologia de resolução do problema.

Esta abordagem “day-night weather-based” ao problema de despacho foi aplicada a vários sistemas de teste tendo apresentado resultados que mostram ser viável a sua implementação em sistemas reais.

Em 2003, A. Motto [24] abordou o problema de Afectação de Unidades sob o ponto de vista do mercado de energia, considerando também, para além das restrições técnicas comuns às outras abordagens, restrições financeiras.

A solução óptima é aquela que maximiza a margem monetária das unidades aceites, ao mesmo tempo que minimiza a soma das margens das unidades não seleccionadas que atingiriam margens positivas, caso não haja alteração do preço da energia no mercado.

A resolução consistiu na formulação do novo modelo de afectação de unidades num problema de programação matemática linear que pode ser resolvido utilizando ferramentas de optimização disponíveis.

Em 2003, C.C. Rajan [25] recorreu ao método pesquisa tabu para resolver o problema de Afectação de Unidades, comparando os resultados obtidos com os resultados obtidos utilizando programação dinâmica, relaxação Lagrangeana e “Simulated Annealing”.

Para o custo de operação são considerados os custos de produção das unidades, dados por uma função quadrática, e o custo de arranque, dependente do tempo em que a unidade esteve fora de serviço. Como restrições foram considerados o balanço de potência, as necessidades de reserva girante, os tempos mínimos de paragem e arranque e a obrigatoriedade de colocação em serviço de algumas unidades para controlo da tensão na rede. Não foi tido em conta a taxa de variação da carga dos geradores no arranque porque a sua inclusão aumentaria significativamente os recursos computacionais necessários.

O algoritmo assume que numa primeira fase todas as unidades estão em serviço, calculando o valor da função objectivo à medida que retira unidades. A verificação das restrições é feita recorrendo à pesquisa tabu, validando ou não as soluções testadas.

Os resultados obtidos quando comparados com os de outros métodos são satisfatórios, não existindo limite para a dimensão da rede e não obriga à relaxação de restrições. Permite obter soluções quase-optimas em problemas com múltiplas restrições e funções objectivo conflituosas num tempo razoável.

Em 2003, U. Ozturk [26] considerou a Afecção de Unidades como um problema estocástico, utilizando a programação “Chance Constrained” na Afecção de Unidades.

A programação “Chance Constrained” é um dos métodos possíveis para a resolução de problemas de optimização em sistemas com parâmetros aleatórios, tal como a carga horária num problema de Afecção de Unidades.

O problema de Afecção de Unidades é inicialmente formulado como uma optimização “chance constrained” em que a incerteza da carga ao longo do tempo é baixa. A solução consiste na resolução duma sequência de problemas determinísticos (ausência de incerteza na carga), que convergem para a solução do programa de “chance constrained”. Utilizou-se a relaxação Lagrangeana para resolução dos problemas determinísticos.

Este método permitirá não utilizar a reserva girante como uma restrição, pois a ocorrência duma situação adversa foi já considerada no problema. A aplicação

deste método apenas pode ser avaliada após comparações extensivas com os métodos determinísticos.

Em 2004, T. Saksornchai [27] avaliou a influência da previsão da carga no problema de Afectação de Unidades e a sua optimização recorrendo a redes neuronais para previsão de cargas num horizonte temporal curto.

Num mercado desregulado, erros nas previsões das cargas podem resultar na necessidade de aquisição de energia no mercado spot, a preços elevados, ou na desnecessária colocação em serviço de unidades produtoras. A melhoria da previsão da carga será o primeiro passo para melhorar o desempenho do problema de afectação de unidade.

A utilização de redes neuronais na previsão de cargas para horizontes temporais horários e diários permitiu atingir reduções de custos diários superiores a 3 %.

Em 2005, J. Catalão [28] considerou o impacte ambiental no problema de Afectação de Unidades.

Pretendeu considerar simultaneamente o custo do combustível resultante da afectação das unidades e as considerações ambientais devido ao mercado de emissões, sendo assim um problema multiobjectivo.

A minimização do custo do combustível é inadequada quando as emissões poluentes são incluídas na operação das centrais térmicas. A minimização do custo do combustível e das emissões são objectivos contraditórios, sendo apresentada uma abordagem prática baseada em optimização multiobjectivo,

também conhecido por soluções “Pareto-optimal”, representadas graficamente pelas curvas de compromisso entre os critérios de custo do combustível e emissões poluentes.

A abordagem prática baseou-se na modelização dum problema de programação matemática não-linear inteiro-misto, combinado conhecimentos técnicos e económicos com considerações ambientais.

A abordagem foi testada utilizando um sistema com 11 centrais térmicas num horizonte temporal de 168 horas, sendo que a estratégia adoptada foi de otimizar numa primeira fase o custo do combustível e as emissões poluentes de forma independente e posteriormente procedeu-se à combinação dos dois objectivos.

Esta abordagem permite ao decisor optar, recorrendo às curvas de compromisso, por uma afectação de unidades que tem em conta também o comércio de emissões poluentes.

Em 2005, Q. Zhou [29] utilizou o método SCUC (Security-Constrained Unit Commitment) para minimizar a perda de eficiência resultante da metodologia de Afectação de Unidades utilizada na Califórnia.

O mercado da energia eléctrica na Califórnia sofreu uma grave crise em 2000 e em 2001, devido à incapacidade de gerar energia quando esta era necessária. Como consequência, quase todas as centrais não hídricas são actualmente designadas pelo FERC (Federal Energy Regulator Commission) como “Must Offer Units”, o que significa que têm que estar disponíveis para o Despacho do

operador independente da Califórnia (CAISO - Califórnia Independent System Operator).

É dada garantia de retorno dos custos de arranque e de carga mínima às unidades sujeitas a “Must Offer”. Dependendo das condições do sistema, algumas destas unidades poderão não ser necessárias, pelo que o CAISO pode desistir da sua utilização e designar outras unidades para garantir a fiabilidade do sistema. Esta desistência de unidades é designada por MOW (Must Offer Waiver).

A utilização desta metodologia conduziu um processo de afectação de unidades que não é nem transparente nem óptimo e causou um aumento significativo nos custos.

O novo procedimento baseia-se no método “SCUC” para que a afectação de unidades “Must Offer” seja realizada de forma óptima em número suficiente para garantir os requisitos de fiabilidade ao mesmo tempo que minimiza os custos de arranque e de carga mínima durante o período de tempo especificado.

O autor utilizou a Programação Mista Inteira Linear (MILP – Mixed Integer Linear Programming) para a formulação SCUC do problema MOW.

As simulações de mercado utilizando dados operacionais reais demonstraram que este método permite reduzir significativamente os custos da afectação de unidades “Must Offer”.

Em 2006, J. Raglend [30] formulou o problema de Afectação de Unidades como um problema multiobjectivo, que integra simultaneamente as vertentes económicas e ambientais.

O problema bi-objectivo de despacho económico e despacho ambiental é convertido numa única função objectivo pela introdução de um factor de penalização do preço.

As soluções de Afectação de Unidades não são exequíveis devido à operação segura do sistema se não considerarem as restrições operacionais e de trânsito de energia. O algoritmo proposto introduz uma eficiente Afectação de Unidades que tem em conta restrições ambientais e de trânsito de energia, obtendo-se os custos mínimos de operação que satisfazem tanto as restrições das unidades como da rede.

Segundo o autor, desde a primeira abordagem realizada em 1971 por Gent e Lamont ao problema de minimização de emissões, que têm surgido várias abordagens ao problema de Afectação de Unidades considerando tanto as vertentes económicas e ambientais. O autor propõe um novo método de penalização do preço que determina o custo exacto das emissões e do combustível em função da carga, considerando tanto o problema económico como o de emissões sujeitos às restrições operacionais e de trânsito de energia. O método foi testado no sistema de 30 barramentos do IEEE e num sistema de 75 barramentos de uma rede Indiana, tendo os resultados validado o método proposto.

Em 2006, J. Catalão [31] apresentou uma metodologia baseada em otimização multiobjectivo para o problema da otimização da exploração de recursos térmicos tendo em consideração a restrição de emissões.

As consequências ambientais resultante da utilização das centrais térmicas, em particular a questão das emissões de gases com efeito de estufa, assumiu particular importância devido à entrada em vigor a 16 de Fevereiro de 2005 do Protocolo de Quioto, que impõe uma redução das emissões para os países desenvolvidos de 5% de 2008 a 2012 comparativamente às emissões em 1990.

A programação dinâmica foi uma das primeiras metodologias utilizadas para a resolução do problema de afectação de unidades, assegurando a obtenção da solução óptima. A desvantagem da utilização da programação dinâmica advém da necessidade de se proceder a uma discretização, sendo exigente em termos de capacidade de memória e tempo de computação. Esta desvantagem evolui de forma exponencial com a dimensão do problema. A aplicação de heurísticas à programação dinâmica permite reduzir o tempo de computação, deixando contudo de assegurar a obtenção da solução óptima. Consequentemente, a utilização da programação dinâmica só é possível para sistemas termoeléctricos de reduzida dimensão.

Para sistemas termoeléctricos de dimensão considerável é usada a metodologia baseada na relaxação Lagrangeana. A vantagem mais relevante que resulta da utilização desta metodologia reside na decomposição do problema em subproblemas. Assim, a afectação de unidades é obtida pela resolução dos subproblemas associados a cada uma das unidades, sendo as unidades

otimizadas individualmente. O termo relaxação está relacionado com o princípio básico da metodologia baseada na relaxação Lagrangeana, que consiste em ignorar as restrições do tipo global e penalizar a sua violação na função objectivo de forma linear recorrendo a multiplicadores, denominados de multiplicadores de Lagrange. A metodologia baseada na relaxação Lagrangeana não apresenta a desvantagem da dimensionalidade típica da programação dinâmica, visto que, as exigências em termos de capacidade de memória e tempo de computação evoluem em proporcionalidade directa com a dimensão do problema.

Para o problema de afectação de unidades, se não é considerada a restrição de emissões no problema de afectação de unidades, pode ser privilegiada a afectação uma unidade de custo reduzido, mas excessivamente poluente. É assim apresentada uma metodologia baseada em optimização multiobjectivo para a resolução deste problema, em que as funções objectivo consideradas são de natureza conflituosa, visto que, podem existir soluções admissíveis que não minimizem simultaneamente as funções objectivo, sendo contudo possível identificar soluções que definem compromissos entre as funções objectivo — soluções não dominadas, não inferiores ou óptimas-Pareto.

A metodologia apresentada foi ensaiada num sistema constituído por onze centrais térmicas, num horizonte temporal de 168 horas, tendo os resultados numéricos obtidos confirmado o bom desempenho desta metodologia, visto que fornece a curva de Pareto e a melhor solução de compromisso, do conjunto de soluções não dominadas (solução admissível é não dominada se e só se não existe outra solução admissível que melhore simultaneamente todas as funções

objectivo, i.e. a melhoria num objectivo só pode ser alcançada à custa da degradação do valor do outro objectivo), com um tempo de computação reduzido.

Em 2006, D. Simopoulos [32] abordou os efeitos das restrições das emissões poluentes no problema de Despacho de Unidades.

Propôs um novo método para incorporar as restrições das emissões de dióxido de carbono na solução do problema de Despacho, baseado num algoritmo “simulated annealing”.

As restrições das emissões são incluídas no problema integrando na função objectivo o custo de aquisição de licenças adicionais de emissões sempre que o total das emissões do sistema excede um valor pré-definido.

Segundo o autor, a técnica de SA, introduzida em 1983, tem a vantagem de resolver problemas difíceis de optimização combinatória.

O método foi testado num sistema com 26 unidades produtoras num espaço temporal de 24 horas, tendo-se provado eficiente e evidenciado que o preço das licenças de emissões de CO₂ têm um grande impacte na solução final do problema de Despacho.

Em 2007, L. Wu [33] apresentou um modelo para calcular o custo da fiabilidade dum sistema baseado numa optimização estocastica de afectação de unidades a longo prazo com restrições de segurança (SCUC – security-constrained unit commitment).

Paragens aleatórias das centrais e problemas nas linhas de transmissão, assim como uma pouca precisa previsão de carga são tipicamente modelizadas, segundo o autor, como árvores de cenários na simulação de Monte Carlo. Os métodos tipicamente descritos na literatura procuram a resolução do problema do despacho económico e assumiam restrições pré-definidas de reserva girante para uma solução determinística da afectação de unidades.

O autor apresenta um modelo que resolve o problema de afectação de unidades numa base temporal horária, que tem em conta o carácter aleatório das paragens das centrais e dos problemas nas linhas de transmissão assim como da incerteza na previsão da carga na questão da fiabilidade do sistema. Considera também, no cálculo da reserva girante óptima, as possíveis incertezas como um compromisso entre a minimização dos custos de operação e o cumprimento dos requisitos de fiabilidade do sistema.

É incluída a previsão da perda de carga (LOLE – loss of load expectation) como uma restrição na afectação estocástica de unidades, como forma de determinar o custo da reserva girante.

Segundo o autor o método proposto pode tanto ser utilizado por empresas verticais ou por operadores independentes (ISO – Independent System Operators). No primeiro caso, o método permitirá considerar nos estudos de fiabilidade o impacte a longo prazo da afectação de combustíveis e emissões. No segundo caso, as restrições de combustível e de emissões são consideradas como restrições de energia na resolução do problema de fiabilidade dos sistemas.

Em 2009, F. Aminifar [34] apresentou uma formulação de Afectação de Unidades que integra um critério probabilístico, considerando uma reserva girante probabilística e perdas de carga.

Tanto a reserva girante como a perda de carga são consideradas como reservas de operação e incluiu-se o tempo de aviso prévio de perda de carga na formulação do problema. A energia planeada não fornecida (EENS – Expected Energy Not Supplied) é considerada como um critério probabilístico e é proposto um novo método para calcular este índice na presença de carga não interruptíveis.

A formulação deste problema de afectação de unidades com restrições de fiabilidade foi baseada no método de Programação Inteira Mista. Para ultrapassar as dificuldades associadas à dimensão do problema, as equações foram limitadas de forma apropriada com limites inferiores e superiores.

Segundo o autor, o conjunto de estudos numéricos realizados demonstraram a eficiência da formulação proposta.

2.3 Evolução

Os artigos produzidos ao longo dos últimos 30 anos mostram um aumento da complexidade do problema de Afectação de Unidades, que tem como objectivo minimizar o custo de operação das unidades, através da consideração de um número crescente de restrições que reflectem a realidade das unidades.

A crescente complexidade do problema foi acompanhada pelo desenvolvimento das técnicas utilizadas para a sua resolução, tanto através do desenvolvimento de técnicas conhecidas como pela introdução de novas técnicas. São também desenvolvidos métodos que combinam diferentes técnicas num mesmo problema.

É também evidenciado a inclusão de novas considerações no problema, não directamente relacionadas com a operação das unidades térmicas. Estas restrições são tanto de ordem técnica, tais como as restrições nas redes de transporte de energia ou de capacidade de abastecimento de combustível, como de ordem económica, em que a liberalização dos mercados leva a uma alteração no objectivo, passando de minimização de custos para a maximização do resultado financeiro, como de ordem ambiental, tendo sido introduzidas limitações ou penalizações nas emissões das unidades térmicas.

As restrições nas emissões poluentes são consequência da nova legislação que foi introduzida nos Estados Unidos [11] e nos países signatários do Protocolo de Quioto [31].

Em Portugal, o Decreto-Lei nº 178/2003 estabelece limitações às emissões para a atmosfera de SO₂ e de NO_x provenientes de grandes instalações de combustão, transpondo para a ordem jurídica nacional a Directiva n.º 2001/80/CE, do Parlamento Europeu e do Conselho, de 23 de Outubro.

Numa primeira fase, o problema de Afectação de Unidades era resolvido utilizando listas de prioridades obtidas de forma heurística. C. K. Pang [2] afirma que no início dos anos 80 do século XX esta era uma metodologia ainda largamente utilizada. Em 1998 a maioria dos operadores ainda usavam processos heurísticos devido à sua simplicidade e facilidade de implementação [16].

A crescente complexidade do problema da Afectação de Unidades é já ilustrada em 1976 por C.K: Pang [1], ao formular um problema que tem em consideração não só o custo de produção mas também o custo de arranque, dependente do tempo em que a unidade esteve fora de serviço. São também tidas em consideração as limitações associadas à reserva girante, e as restrições resultantes das limitações de pessoal de operação nos arranques e paragens das unidades e os tempos mínimos para arranque e paragem. O autor recorreu à programação dinâmica tendo como ponto de partida uma lista de prioridades, limitando assim a pesquisa de soluções e contornando desta forma a maldição da dimensionalidade. Embora o autor afirme que com este método possa atingir reduções significativas de combustível, não fica garantida a solução óptima pois as soluções estão limitadas pela lista de prioridades inicial.

A complexidade do problema foi crescendo pela introdução de cada vez mais restrições relacionadas com as próprias unidades, tais como o limite na variação da carga nos geradores [8]. A consideração da potência gerada como sendo

uma função escada permite simplificar significativamente o problema. Para a resolução incluindo o limite na variação da carga nos geradores, C. Wang utilizou em 1993 uma combinação de dois métodos: redes neuronais e a programação dinâmica.

A incerteza na previsão da carga pode conduzir a uma sobre-afecção de unidades, o que se reflecte no aumento do custo [18], ou, em sentido oposto, numa sub-afecção, tendo como consequência o aumento do risco na operação da rede [10][19] e eventualmente também maiores custos de operação pela necessidade de aquisição de energia no mercado spot, a preços elevados [27] ou gerar energia de forma não económica [19]. A partir de simulações efectuadas com os dados disponibilizados por dois operadores, B. Hobbs [19] conclui em 1999 que uma redução de 1% do MAPE (mean absolute percentage error - percentagem do módulo do erro médio) leva a uma redução de custos entre 0,3 e 0,5 % quando o MAPE se encontra entre os 3 e 5%.

Historicamente, o principal critério para determinar o valor da reserva girante era de que este valor deveria ser maior ou igual que a capacidade da maior unidade afectado [10]. A redução de custos motivada pela liberalização dos mercados conduziu à introdução de novos métodos para determinação da reserva girante, tomando-se também em conta o risco, limitando-se este a um valor máximo pré-definido [10]. Em 2004, T. Saksornchai [27] conseguiu uma redução de custos superiores a 3 % pela utilização de redes neuronais na previsão de cargas para horizontes temporais horários e diários.

A liberalização dos mercados conduz também a uma nova abordagem ao problema de Afecção de Unidades, tendo C. Richter [22] apresentado em 2000 uma formulação para o problema do planeamento de Afecção de Unidades baseada em aspectos económicos, em que utilizando algoritmos genéticos, procura maximizar o lucro em vez de minimizar os custos.

São utilizadas diversas técnicas ao longo dos anos, sendo no entanto recorrente a utilização de algumas delas. Segundo A. I. Cohen [3], as técnicas utilizadas até 1983 podiam ser agrupadas da seguinte forma:

- Lista de prioridades e métodos heurísticos
- Programação dinâmica
- Programação Inteiros
- Relaxação Lagrangeana

Ainda segundo A. I. Cohen, a programação dinâmica era a técnica mais utilizada por diversos autores nesta época, tendo A. I. Cohen apresentado nesta data um método baseado em técnicas de partição e avaliação (branch-and-bound). Segundo A. Merlin [4], a relaxação lagrangeana era já utilizada pela EDF em 1983.

Em 1998 A.H. Mantawy [15] faz um novo ponto de situação sobre as técnicas utilizadas, agrupando estas em três categorias:

- Técnicas clássicas de optimização, tais como programação dinâmica, programação inteiros, relaxação lagrangeana e partição e avaliação

- Lista de prioridades e métodos heurísticos
- Técnicas de inteligência artificial, tais como redes neuronais, algoritmos genéticos, “expert systems”, pesquisa tabu e “simulated annealing”.

Foi também explorada por diversos autores a combinação de diferentes técnicas num mesmo algoritmo. Exemplos desta aproximação ao problema de Afectação de Unidades são A. Merlin [4] que combinou em 1983 a relaxação lagrangeana com a técnica de partição e avaliação, A.H. Mantawy [20] que em 1998 integrou num único algoritmo algoritmos genéticos, pesquisa tabu e “Simulated Annealing” e C.P.Cheng [21], que combinou em 2000 a relaxação lagrangeana com os algoritmos genéticos.

A combinação de diferentes técnicas, quando bem sucedida, permite obter melhores resultados do que os obtidos isoladamente por cada técnica [20] [21]. Pretendia-se com esta aproximação ao problema eliminar os pontos fracos de cada técnica.

As restrições ambientais vêm impor novas abordagens ao problema de Afectação de Unidades, passando a ser um problema multiobjectivo.

Em 1994, A. A. El-Keib [11] avaliou o impacte das restrições ambientais impostas em 1990 nos EUA, através do “Clean Air Act of 1990” no problema de Despacho. Segundo A. A. El-Keib, tinham sido propostas várias estratégias de Despacho para um problema simultaneamente económico e ambiental, mas nenhuma das estratégias apresentadas até à altura reflectia os requisitos exactos da legislação, tendo A. A. El-Keib apresentdo duas formulações de

despacho económico. A primeira baseada em Relaxação Lagrangeana, satisfaz as restrições ambientais e a segunda tira partido duma reserva de geração de compensação, denominada “compensating generation provision”, a qual está relacionada com aspectos específicos da legislação que são descritos pelo autor. De acordo com o autor, ambas as soluções apresentadas são simples e utilizam as técnicas de despacho económico convencional apenas com ligeiras alterações.

Em 1998, S. Sen [16] define como funções objectivo a minimização do custo total de operação, a minimização de emissões e a maximização da fiabilidade e segurança, embora geralmente este terceiro objectivo seja considerado uma restrição. O segundo objectivo pode também ser considerado como uma restrição, mas caso seja considerado um objectivo, então o planeamento de Afectação de Unidades é um problema de optimização multiobjectivo, com funções objectivo conflituosas. O estudo constata que os métodos mais eficientes para este problema são a programação dinâmica com busca restrita, a relaxação lagrangeana aumentada e o “Simulated Annealing”.

Em 2002, C. Palanichamy [35] aborda a questão da inversão da temperatura. A inversão da temperatura é a subida da temperatura com a altitude, em oposição ao normal, devido à radiação de calor no espaço, durante a noite calma e límpida e pode originar o afundamento de ar mais denso, frio, para os vales e bacias a uma cota inferior [36], originando uma corrente descendente de ar. O ar quente na camada superior à do ar frio actua como cobertura, suprimindo o

movimento ascendente vertical, aprisionando o ar frio junto à superfície. Consequentemente, os poluentes emitidos são mantidos junto à superfície devido à inversão da temperatura, originando uma diminuição da qualidade do ar [37].

Este fenómeno foi responsável pela morte de vinte pessoas em Donora, no estado da Pensilvânia nos EUA, em Outubro de 1948. Devido à não dispersão dos poluentes das emissões atmosféricas de várias indústrias, estes foram continuamente respirados durante vários dias [38].

C. Palanichamy propôs a integração de fontes renováveis de energia e das novas tecnologias de armazenagem de energia com as centrais que utilizam combustíveis fósseis no problema de despacho, atingindo uma concentração controlada de poluentes na vizinhança das fontes poluidoras, com um custo atractivo, não sendo necessário incluir na função objectivo um modelo climático. Os algoritmos de despacho seleccionados dependem das condições climatéricas.

Em 2005 J. Catalão [28] considerou simultaneamente o custo do combustível resultante da afectação das unidades e as considerações ambientais devido ao mercado de emissões, sendo assim o problema de Afectação de Unidades um problema multiobjectivo com objectivos conflituosos, tendo apresentada uma abordagem prática baseada em optimização multiobjectivo, também conhecido por soluções “Pareto-optimal”, representadas graficamente pelas curvas de compromisso entre os critérios de custo do combustível e emissões poluentes

que permitem ao decisor optar por uma afectação de unidades que tem em conta também o comércio de emissões poluentes.

Em 2006, J. Raglend [30] formulou o problema de Afectação de Unidades como um problema multiobjectivo, que integra simultaneamente as vertentes económicas e ambientais, sendo o problema bi-objectivo de despacho económico e despacho ambiental é convertido numa única função objectivo pela introdução de um factor de penalização do preço.

J. Catalão volta, em 2006 [28], a abordar o problema de optimização da exploração de recursos térmicos tendo em consideração simultaneamente o objectivo económico e o objectivo ambiental, resolvendo o problema através de uma metodologia baseada em optimização multiobjectivo, que tem em consideração o conflito entre os objectivos. A melhor solução de compromisso, entre a solução correspondente ao custo total mínimo e a solução correspondente à emissão total mínima, foi determinada em função de um novo parâmetro apresentado, taxa de variação, e do correspondente ângulo de declive.

2.4 Conclusões

As técnicas utilizadas na resolução do problema de Afectação de Unidades acompanharam o aumento da complexidade do problema, quer seja devido a um maior rigor e detalhe na descrição do funcionamento das unidades, quer seja devido à introdução de condicionantes externas ou mesmo devido à alteração do objectivo, passando de um problema de minimização de custos para

maximização do resultado financeiro, devido à liberalização do mercado. Finalmente, devido às preocupações ambientais, evoluiu para um problema multiobjectivo, adicionando mais uma função objectivo associada com considerações ambientais, em que há um conflito entre o objectivo custo e o objectivo das emissões poluentes.

As principais técnicas referenciadas, são a programação dinâmica, referenciada desde 1976 [1] e a relaxação lagrangeana, sendo um marco importante a adopção desta técnica pela EDF em 1983 [4].

São também referenciadas outras técnicas, tais como a avaliação e partição, referenciada desde 1983 [3] e algoritmos genéticos, referenciados desde 1994 [9].

Capítulo 3

Casos de Estudo

Resumo

São avaliados três Casos de Estudo, em que no primeiro o objectivo é minimizar o custo de operação de uma unidade térmica considerando o custo do combustível, no segundo é adicionado o custo atribuído às emissões poluentes e finalmente no terceiro é simulado um conjunto de três unidades térmicas sujeitas a um acordo de fornecimento bilateral e em que se considera o custo do combustível e das emissões poluentes.

3. Casos de Estudo

3.1 Afecção de uma Unidade Térmica

Considerando o Custo do Combustível

3.2 Afecção de uma Unidade Térmica

Considerando o Custo do Combustível e
das Emissões Poluentes

3.3 Afecção de Unidades Térmicas Sujeitas

a Acordo de Fornecimento Bilateral

3. Casos de Estudo

Os preços de mercado da energia eléctrica considerados foram retirados do site do OMEL [39], que é o operador do mercado responsável pela gestão do sistema de ofertas de compra e venda de energia eléctrica no mercado diário.

Será utilizada a programação dinâmica para resolver os problemas de afectação de unidades térmicas, sendo estas unidades codificadas recorrendo a um modelo de autómato finito. Um autómato finito é abstractamente definido por quintuplas da forma

$$D = (Q, I, \delta, q_0, F)$$

onde

Q é um conjunto finito não vazio de estados

I é um conjunto de símbolos, denominado alfabeto de entrada do autómato

$\delta = Q \times I$, em que Q é função de transição directa. Esta função fornece para cada par "estado e símbolo de entrada" um novo estado para onde o autómato deverá mover-se

$q_0 \in Q$ é o estado inicial

$F \subseteq Q$ é o conjunto de estados finais.

No capítulo de Formulação do Problema são referidos os conceitos que serão requeridos nos casos de estudo. Assim, para uma unidade térmica com o tempo

mínimo de funcionamento de quatro horas, o tempo mínimo de paragem de duas horas e o tempo de arrefecimento total de quatro horas, a representação por um autómato finito pode ser ilustrada por uma Máquina de Estados cujo diagrama de transição de estados é o seguinte

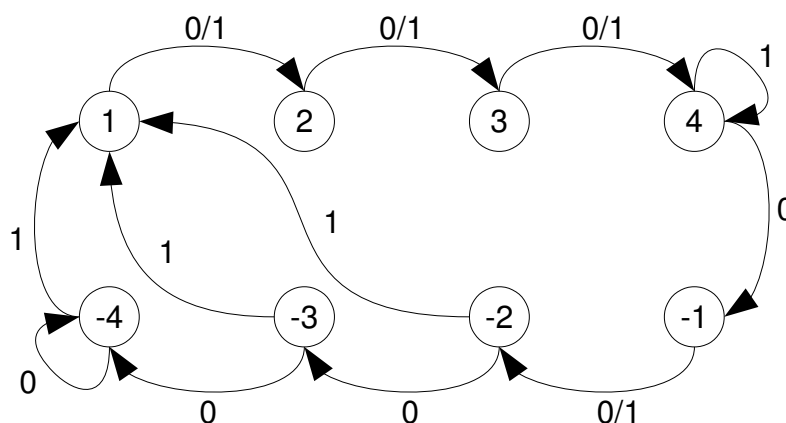


Figura 3.1: Diagrama de transição da Máquina de Estados.

Na Figura 3.1, o estado 1 corresponde ao arranque da Unidade Térmica. Como o tempo mínimo de funcionamento é de quatro horas, a unidade tem que transitar em funcionamento até ao estado 4, correspondente a quatro horas ou mais em funcionamento contínuo.

No estado 4 a unidade pode parar, transitando para o estado -1, que corresponde ao primeiro estado em paragem, ou pode continuar em funcionamento. Não são representados mais estados para além do estado de tempo de funcionamento mínimo pois no fim do período correspondente a este

estado não existe diferença a acrescentar aos futuros períodos de funcionamento do sistema.

Após paragem, a Unidade tem que permanecer parada pelo menos durante o tempo de mínimo de paragem, estado -2. Atingido este estado, a Unidade pode arrancar, transitando para o estado 1, ou permanecer parada, transitando para estados sucessivos, aonde eventualmente poderá arrancar, ou caso não arranque, transitar até ser atingido o tempo de arrefecimento total, a partir do qual a transição em caso de paragem se faz para o estado -4. Não são representados mais estados para além do estado de tempo de arrefecimento total pois no fim deste período a energia armazenada foi já totalmente dissipada, não havendo diferença a acrescentar ao sistema neste estado.

Pode ser dada ordem de arranque à Unidade em qualquer estado entre o estado -2, correspondente ao tempo de mínimo de paragem e o estado -4, correspondente ao tempo de arrefecimento total. O custo de arranque é tanto menor quanto mais cedo for dada esta ordem porque a energia dissipada é crescente com o tempo de acordo com a expressão

$$uccool(t) = uccool_0 \left(1 - e^{-\frac{5 \times t}{utcool}} \right)$$

onde

$uccool(t)$ é o custo de arranque correspondente a um estado -t de paragem

$uccool_0$ é o custo de arranque correspondente ao estado -t em que é atingido o tempo de arrefecimento total
 $utcool$ é o tempo de arrefecimento total.

A representação gráfica deste custo é do tipo indicado na figura seguinte

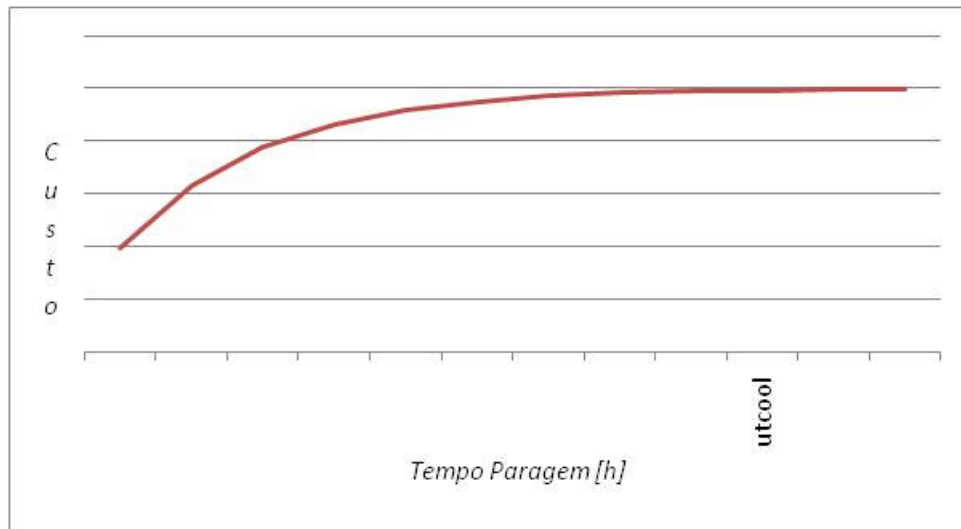


Figura 3.2: Custo de arranque de unidade térmica.

Como alternativa ao diagrama de transição de estados para a Máquina de Estados, pode ser utilizada uma representação em tabela de estados, como indicado na tabela seguinte

ESTADO	Origem	COMANDO	
		0	1
	Destino		
1	2	2	2
2	3	3	3
3	4	4	4
4	-1	4	4
-1	-2	-2	-2
-2	-3	1	1
-3	-4	1	1
-4	-4	1	1

Tabela 3.1: Tabela de estados da Máquina de Estados.

3.1 Afectação de uma Unidade Térmica Considerando o Custo do Combustível

É apresentada a simulação da Afectação de uma Unidade Térmica para um horizonte temporal de uma semana com períodos de uma hora, considerando os preços de mercado da energia eléctrica em Portugal, entre 11 e 17 de Abril de 2011.

O objectivo deste caso de estudo é maximizar o resultado operacional desta unidade, que representa a diferença entre os proveitos e os custos relativos à actividade principal da empresa [41], significando neste caso a diferença entre a facturação e o custo de operação ao longo do período considerado.

A facturação é o produto da energia produzida pelo preço oferecido pelo mercado da energia eléctrica. Em cada hora k , a *facturação*(k) = $ul(k)p(k)$, em que $ul(k)$ é o preço na hora k em €/MWh.

Os dados utilizados para os preços e os consumos da energia em Portugal estão ilustrados na Figura 3.4.

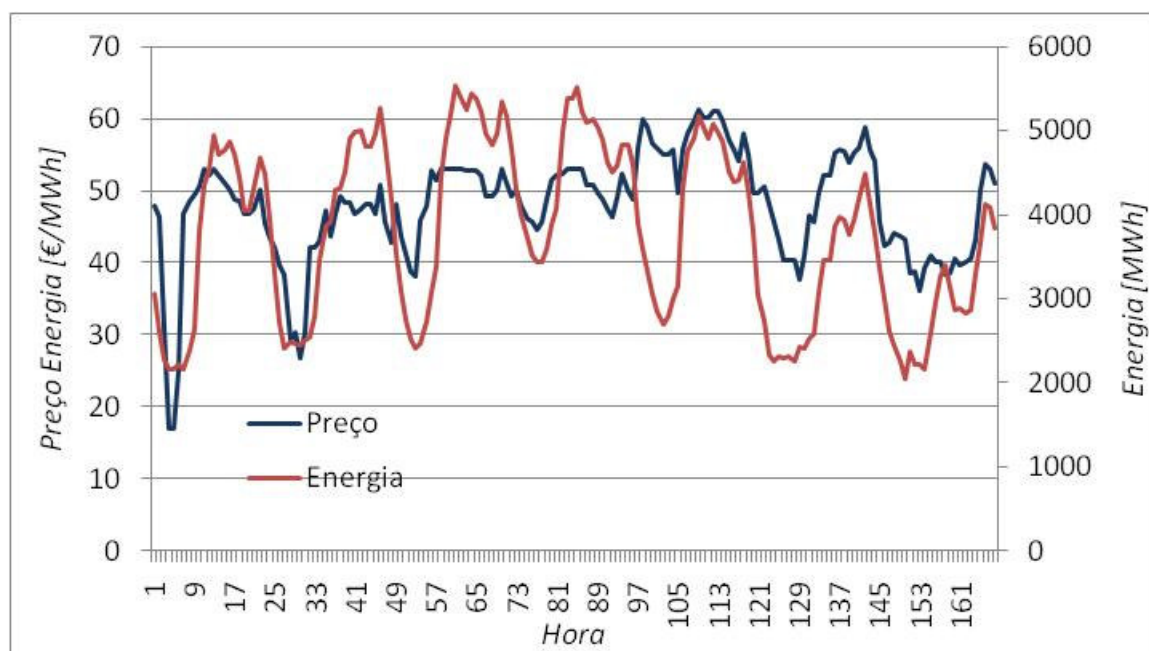


Figura 3.3: Preço da energia e energia consumida em Portugal de 11 a 17 Abril de 2011.

O mercado da energia eléctrica é um mercado liberalizado, pelo que a relação entre o preço da energia e a sua procura é explicada pela Lei da Oferta e da Procura, que não é considerada neste estudo. Ainda, é considerado que não existe poder de mercado para o caso em estudo.

O custo horário de operação da Unidade Térmica é determinado pela função convexa dada por

$$C(u, p) = (a + bp + \frac{1}{2}cp^2)u$$

sendo os coeficientes considerados $a = 277$, $b = 26,5$ e $c = 0,045$. O resultado operacional é dado por

$$RO(k) = ul(k)p(k) - (a + bp(k) + \frac{1}{2}cp(k)^2)$$

Pelo que, quando $u = 1$, o resultado operacional, facturação subtraída do custo, deve ser maximizado em cada hora. A função que traduz o resultado financeiro é uma função côncava, ou seja, a segunda derivada é negativa o que garante a existência de máximo. Em cada hora, caso não sejam violadas os limites de operação, a energia que garante o valor máximo do resultado financeiro é aquela que torna a derivada do resultado financeiro nulo, sendo esta derivada dada por

$$\frac{dRO}{dp}(k) = ul(k) - b - cp(k)$$

pelo que, atendendo a que o período é de uma hora, o valor óptimo da potência eléctrica associada a energia entregue ao mercado é dada por

$$p^*(k) = \frac{ul(k) - b}{c}$$

Este valor é limitado pelos limites de operação e pelas taxas máximas de variação da potência. Assim, caso o valor da potência eléctrica associada à energia entregue seja superior ao valor mínimo entre o máximo admissível e à

adição da potência na hora anterior com a taxa máxima de variação positiva da potência, o valor óptimo da potência eléctrica associada à energia entregue será esse mínimo; caso o valor da potência eléctrica associada à energia entregue seja inferior ao valor máximo entre o mínimo admissível e à potência na hora anterior subtraída da taxa máxima de variação negativa da potência, o valor óptimo da potência eléctrica associada a energia entregue será esse máximo.

A solução escolhida é aquela que conduz à maximização do resultado operacional total, ao longo do horizonte temporal, dado por

$$F.O. = \max \left\{ \sum_{k=1}^{168} u(k) RO(k) \right\}$$

Assim, o melhor resultado operacional é escolhido no fim do horizonte temporal, determinando qual o estado admissível com o maior resultado operacional.

Para a Unidade Térmica G1 simulada, são consideradas as seguintes restrições:

Potência Mínima, $\underline{p}_i = 40$ MW

Potência Máxima, $\bar{p}_i = 120$ MW

Taxa de máxima de variação positiva de potência, $UR_i = 20$ MW/h

Taxa de máxima de variação negativa de potência, $DR_i = 40$ MW/h

Tempo mínimo de funcionamento, $uminup_i = 4$ horas

Tempo mínimo de paragem, $umindo_i = 3$ horas

Tempo de arrefecimento total, $utcool_i = 4$ horas

Custo de arranque após tempo de arrefecimento total, $uccool_j = 2200 \text{ €}$

O resultado é apresentado em termos de défice de resultado operacional, pelo que um valor negativo representa um resultado operacional positivo. No gráfico seguinte é apresentado o estado da unidade e o défice operacional acumulado ao longo do período considerado

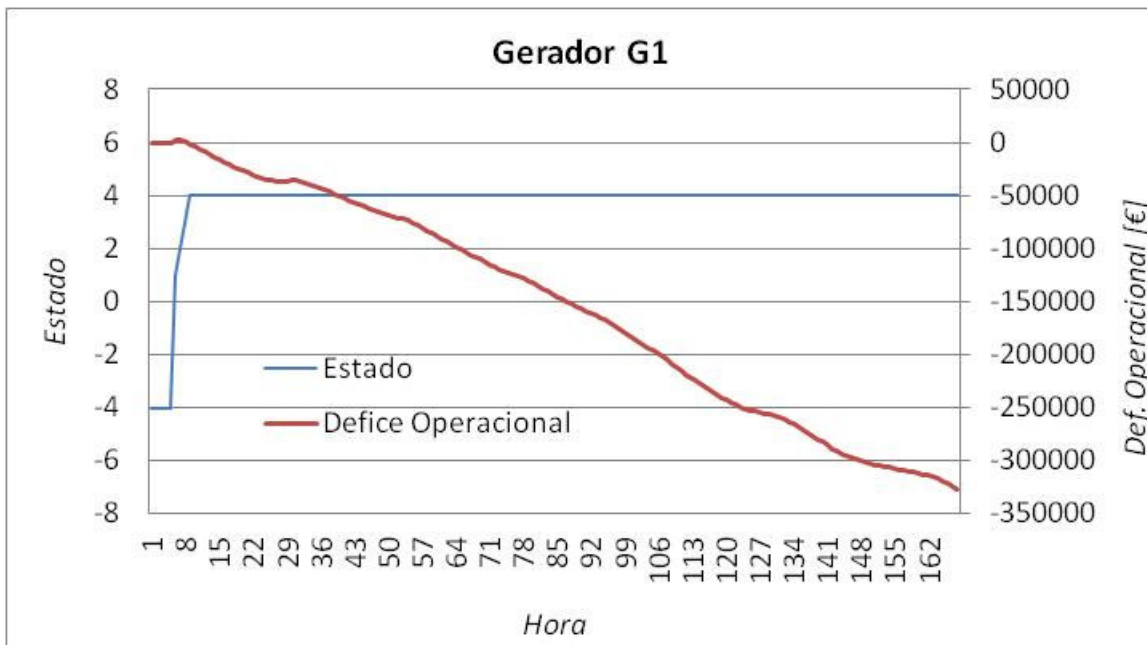


Figura 3.4: Estado e défice operacional da unidade térmica G1.

O arranque da unidade térmica ocorre na hora 6 devido ao baixo preço da energia nas primeiras horas, em particular o valor de 17,1 €/MWh nas horas 4 e 5, que tornam a facturação inferior ao custo qualquer que seja o valor da potência.

No gráfico seguinte são apresentados os valores da facturação e custo ao longo do período considerado

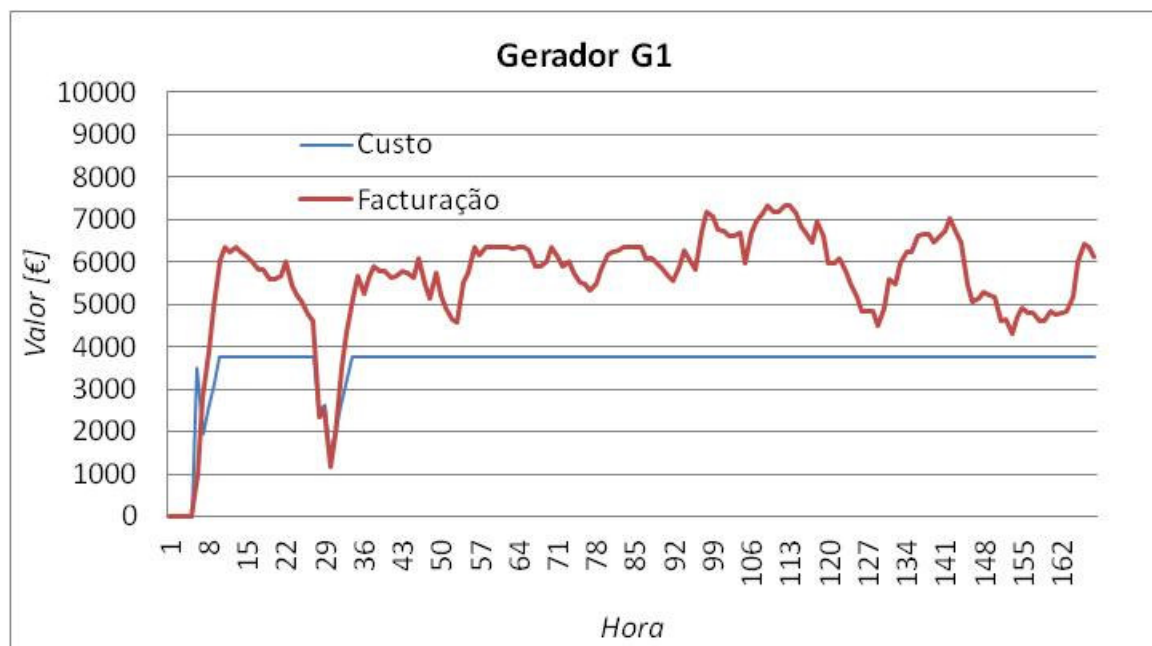


Figura 3.5: Custo e facturação horária da unidade térmica G1.

No arranque, na hora 6, a facturação é inferior ao custo devido ao custo de arranque. Há um decréscimo da potência entre as horas 28 e 33 devido ao decréscimo do preço da energia neste período, o que pode ser verificado no gráfico seguinte

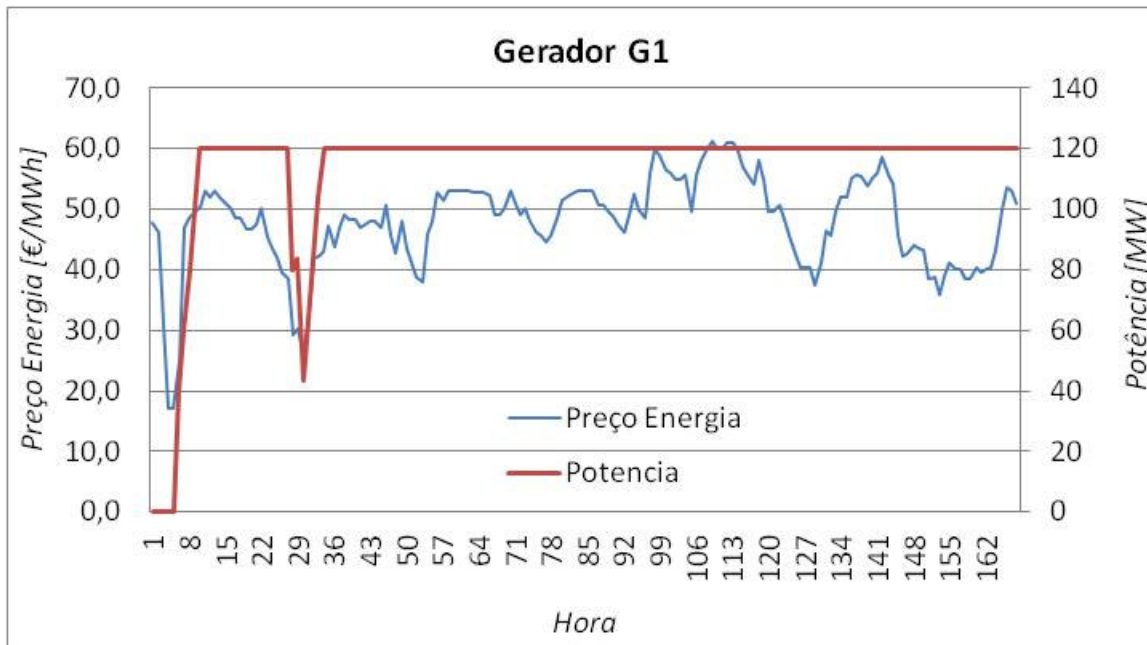


Figura 3.6: Preço da energia e potência da unidade térmica G1.

Para efeitos de avaliação da operação da unidade, é simulado um cenário, em que é alterado o preço da energia. Nesta simulação, o preço da energia eléctrica tem uma redução de 50 % nos intervalos entre as horas 40 a 42 (3 horas), 62 a 68 (7 horas) e 120 a 129 (10 horas).

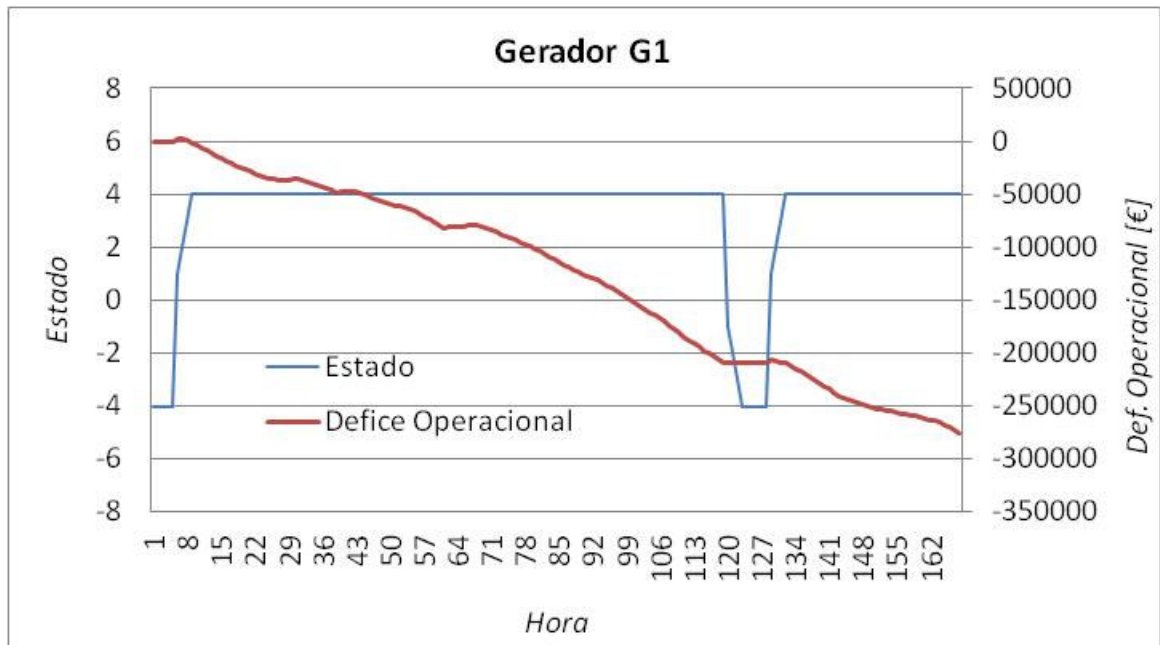


Figura 3.7: Estado e custo da unidade térmica G1 com alteração do preço da energia.

O decréscimo do preço de energia conduz à paragem da unidade entre as horas 120 e 129, período mais longo com redução do preço da energia.

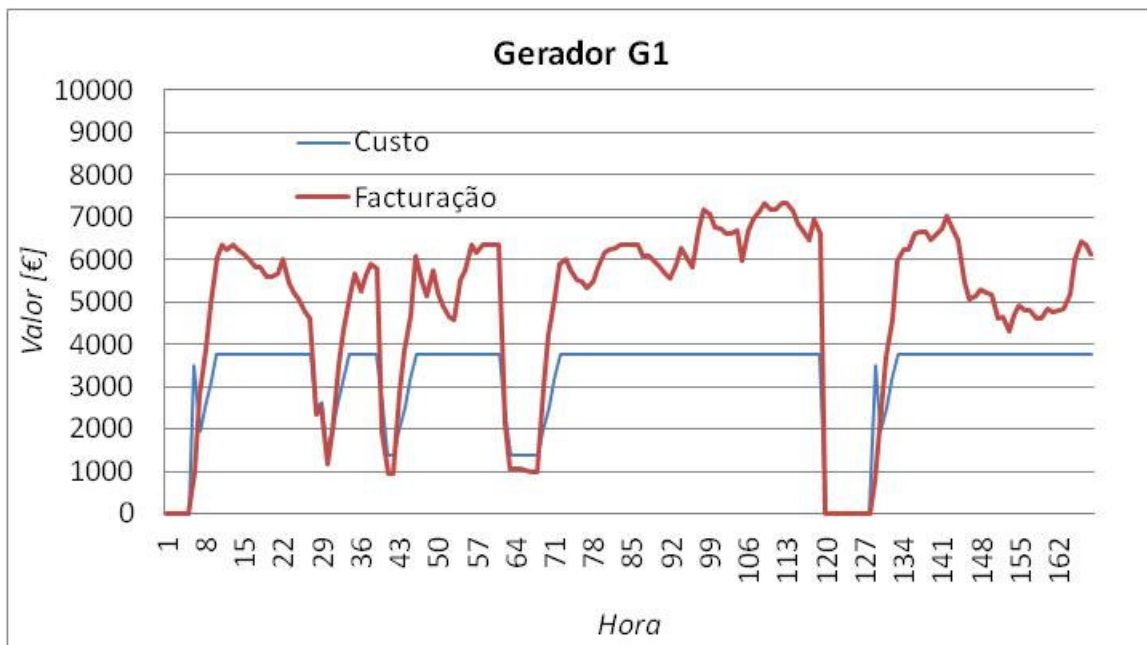


Figura 3.8: Custo e facturação horária da unidade térmica G1 com alteração do preço da energia.

Apesar do custo ser inferior à facturação entre as horas 40 e 42 e entre as horas 62 a 68, a unidade não pára nestes períodos.

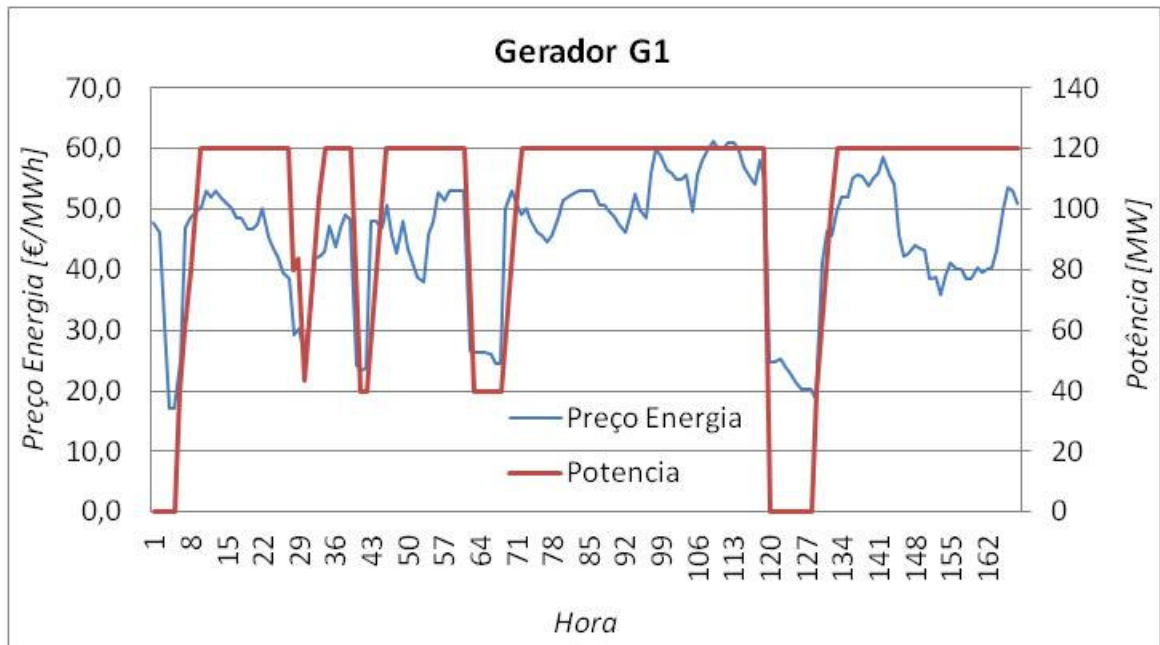


Figura 3.9: Preço da energia e potência da unidade térmica G1 com alteração do preço da energia.

Nos dois primeiros intervalos com redução do preço da energia a unidade reduz potência e o resultado é negativo nesses intervalos. Tal é devido ao facto do custo de arranque da unidade ter um peso maior do que o resultado negativo acumulado em períodos relativamente curtos devido à redução do preço da energia.

O caminho óptimo força à paragem da Unidade Térmica na 120.^a hora apesar do resultado negativo no arranque da unidade na 128.^a hora. A redução durante o terceiro intervalo, 10 horas, do preço de energia origina um resultado negativo superior em valor ao custo de arranque caso a unidade não pare.

3.2 Afectação de uma Unidade Térmica Considerando o Custo do Combustível e das Emissões Poluentes

Enquanto que no caso anterior o objectivo é minimizar o somatório da diferença entre o custo de operação e a venda de energia ao longo de uma semana, neste caso o objectivo tem em consideração as emissões poluentes.

As funções objectivo para a minimização do custo do combustível e das emissões são conflituosos [28], pelo que é necessário utilizar programação matemática multiobjectivo.

A unidade térmica é agora descrita à custa de duas funções. A primeira, tal como no caso anterior, é o custo horário tendo em conta o combustível, sendo dado por

$$C(u,p) = (a + bp + \frac{1}{2}cp^2)u$$

com $a = 277$, $b = 26,5$ e $c = 0,045$.

A segunda é o nível das emissões poluentes, dado por

$$E(u,p) = (\alpha + \beta p + \frac{1}{2}\gamma p^2)u$$

com $\alpha = 40$, $\beta = -0,0016$ e $\gamma = 0,008$.

O resultado vem expresso em MgCO_2/h . O valor ω utilizado para as emissões é de $12,3 \text{ €/MgCO}_2$, sendo este o valor médio destas emissões no primeiro

semestre de 2011, segundo o Sistema Electrónico de Negociação de Direitos de Emissão de Dióxido de Carbono [45].

Foi utilizado o método das somas ponderadas para resolver o problema de programação matemática bi-objectivo que resulta deste caso de estudo, sendo a função objectivo dada por

$$(1 - \lambda) \sum_{k=1}^{168} \left((a + b p(k) + \frac{1}{2} c p(k)^2) - ul(k) p(k) \right) + \lambda \omega \sum_{k=1}^{168} \left((\alpha + \beta p(k) + \frac{1}{2} \gamma p(k)^2) \right)$$

O factor λ varia entre 0 e 1, correspondendo $\lambda = 0$ ao caso de estudo anterior, pois é anulado o termo das emissões e $\lambda = 1$ à situação em que as emissões são a função objectivo, sendo anulado o termo correspondente ao custo do combustível.

A potência que em cada hora que garante o valor mínimo da função é aquela que torna a derivada do resultado financeiro nulo, sendo esta potência dada por

$$p^*(k) = \frac{ul(k) - b(1 - \lambda) - \lambda \omega \beta}{\lambda \omega \gamma + c(1 - \lambda)}$$

Para $\lambda = 0$ o valor do défice operacional coincide com o valor do caso de estudo anterior, pois é anulado o termo do custo das emissões. Com o incremento de λ o valor cresce devido ao maior peso do custo das emissões no valor final.

A afectação da unidade térmica toma os seguintes valores em função de λ .

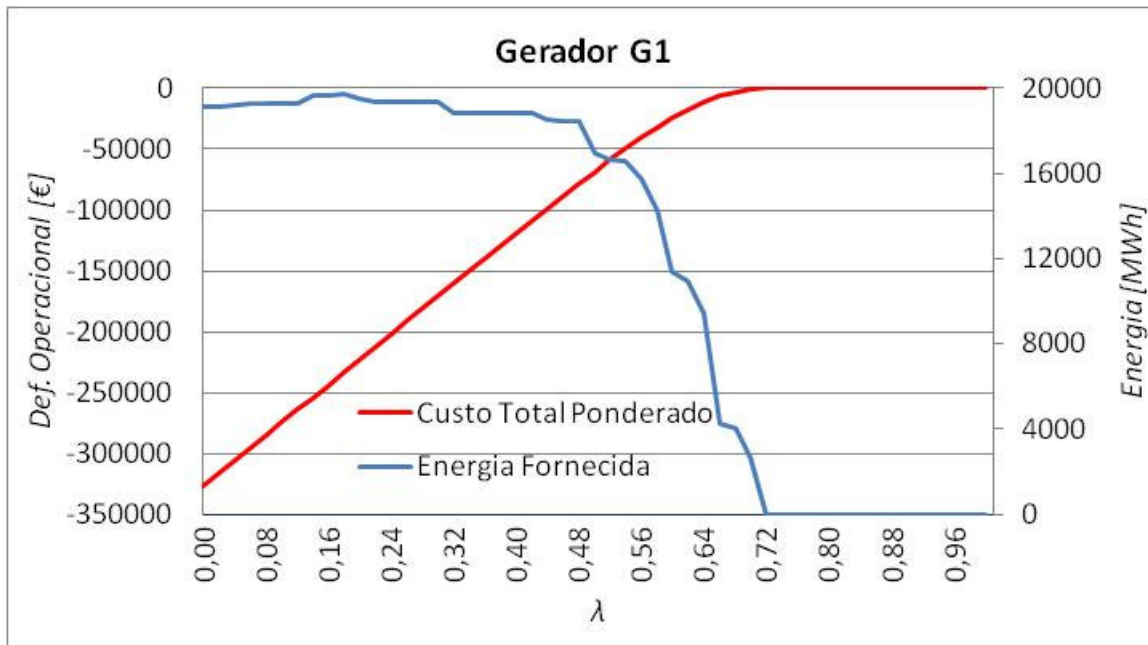


Figura 3.10: Energia fornecida e custo total em função de λ .

O custo total ponderado é crescente devido ao custo das emissões e ao decréscimo do peso do valor da facturação, que é multiplicada por $1 - \lambda$. Para $\lambda > 0,70$ o aumento do peso do custo das emissões é de tal forma preponderante que força a paragem da unidade.

No gráfico seguinte são apresentados os valores do défice operacional e do custo das emissões

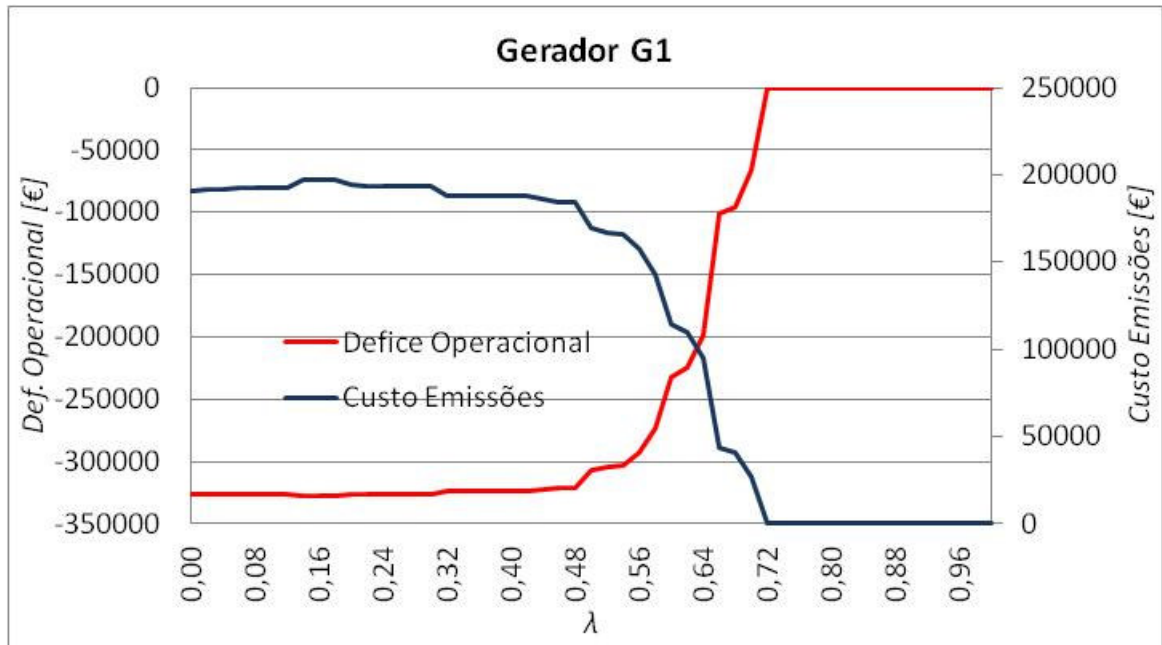


Figura 3.11: Déficit operacional e custo das emissões da unidade térmica G1.

Tanto o déficit operacional como o custo das emissões tendem para zero pois dependem da potência, que se anula com o aumento de λ .

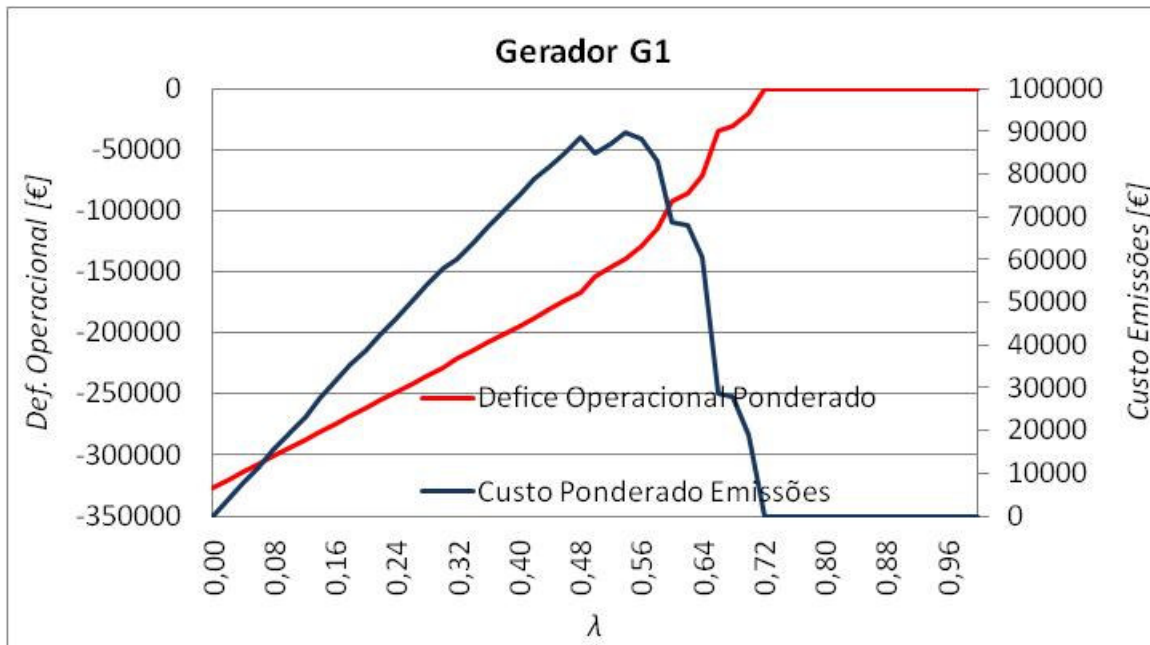


Figura 3.12: Déficit operacional ponderado e custo ponderado atribuído às emissões da unidade térmica G1.

O déficit operacional ponderado nunca se torna positivo. A paragem da unidade tem um custo nulo, pelo que esta é a melhor opção quando as alternativas apresentam custos positivos.

Os valores obtidos do déficit operacional e custo das emissões permitem desenhar uma curva de Pareto, tal como representada no gráfico seguinte

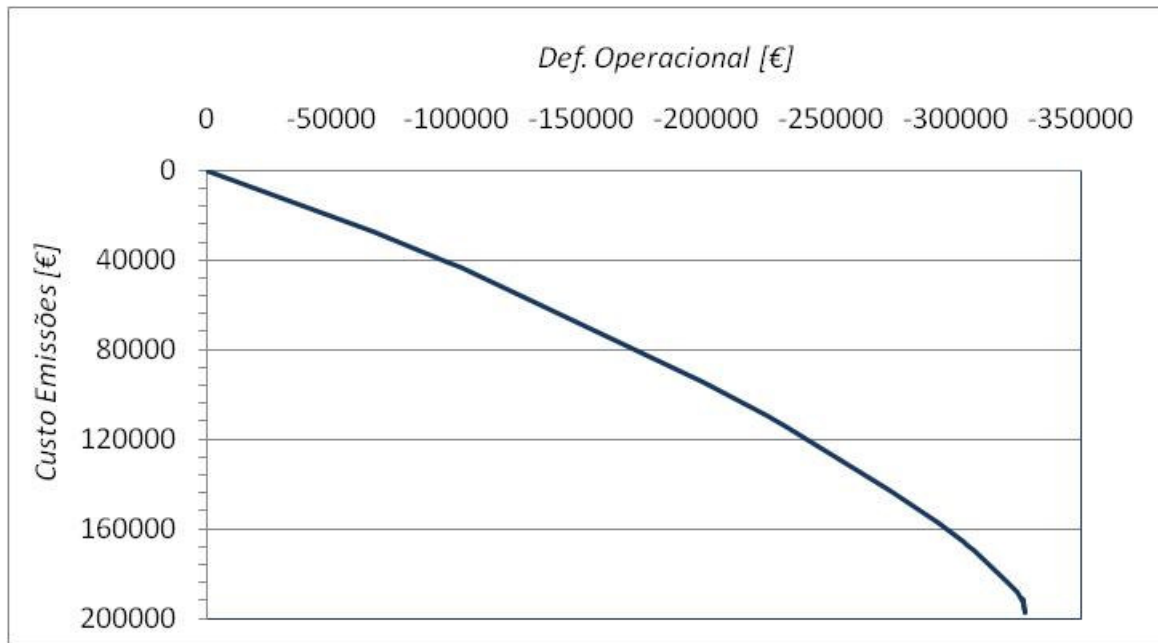


Figura 3.13: Déficit operacional em função do custo atribuído às emissões da unidade térmica G1.

A curva de Pareto representa um compromisso entre o déficit operacional e o custo das emissões, servindo de suporte à decisão da afectação de unidades em mercado de emissões [28].

3.3 Afectação de Unidades Térmicas Sujeitas a Acordo de Fornecimento

Bilateral

Neste caso é simulado um acordo bilateral, em que há o compromisso de fornecer uma energia contratada para um horizonte temporal de uma semana, sendo esta energia fornecida por três unidades térmicas.

Havendo capacidade disponível, as unidades térmicas poderão fornecer energia para além do valor acordado, sujeitas ao preço de mercado utilizado nos casos anteriores, sendo considerado o custo do combustível e o custo das emissões.

O valor ω utilizado para as emissões é de 12,3 €/MgCO₂.

O valor da potência contratada e do preço da energia ao contratado estão representados no gráfico seguinte

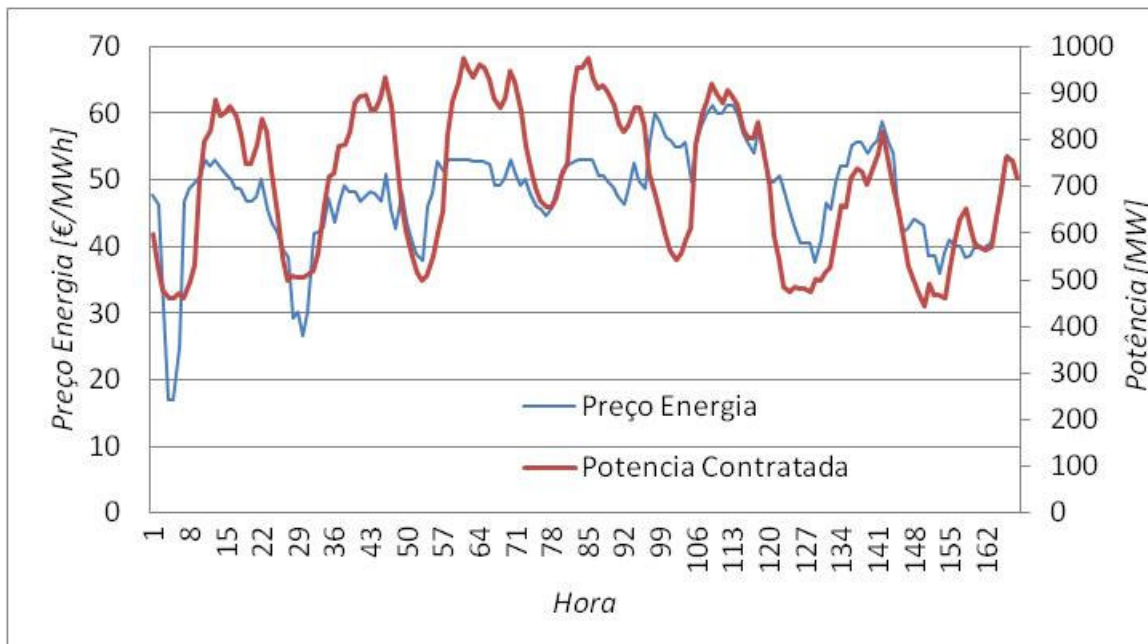


Figura 3.14: Potência contratada e preço da energia.

Foi utilizado o método das somas ponderadas para resolver o problema de programação matemática bi-objectivo que resulta deste caso de estudo, sendo a função objectivo dada por

$$(1 - \lambda) \sum_{k=1}^{168} \left((a + b p(k) + \frac{1}{2} c p(k)^2) - ul(k)(p(k) - d(k)) \right) + \lambda \omega \sum_{k=1}^{168} \left((\alpha + \beta p(k) + \frac{1}{2} \gamma p(k)^2) \right)$$

onde $d(k)$ é a potência contratada. A diferença $p(k) - d(k)$ nunca é negativa porque $p(k) - d(k) \geq 0$ é considerada uma restrição.

Não é considerado o valor a facturar resultante do acordo bilateral.

Foi utilizada a programação dinâmica para a resolução do problema. A potência de cada unidade térmica para os estados de cada hora (estádio) foi calculada recorrendo ao Solver do Excel, sempre que verificada a exequibilidade da solução.

As restrições consideradas para cada unidade são as seguintes

	G1	G2	G3
\underline{p}_i [MW]	40	120	240
\bar{p}_i [MW]	120	400	700
UR_i [MW/h]	40	120	240
DR_i [MW/h]	20	80	160
$uminup_i$ [h]	4	5	7
$umindp$ [h]	3	3	5
$utcool_i$ [h]	4	5	7
$uccool_i$ [€]	2200	2400	3000

Tabela 3.2: Restrições das unidades térmicas.

Em cada estádio, o número total de estados é a combinação dos estados das unidades térmicas. Para a unidade térmica G1 o número de estados são 8, para G2 são 10 e para G3 são 14, o que resulta num total de 1120 estados a serem processados em cada hora, o que evidencia o problema da maldição da dimensionalidade da programação dinâmica.

Os parâmetros das funções do custo do combustível e do custo das emissões são os seguintes

	G1	G2	G3
a	277	300	320
b	26,5	26,3	26,1
c	0,045	0,051	0,031
α	40	43	38
β	-0,0016	-0,0013	-0,0019
γ	0,008	0,009	0,006

Tabela 3.3: Parâmetros das funções das unidades térmicas.

Foram considerados as seguintes condições iniciais

	G1	G2	G3
Estado inicial	4	5	6
P_0 [MW]	90	350	200

Tabela 3.4: Estados iniciais das unidades térmicas.

Os valores obtidos estão representados na tabela seguinte

λ	Custo Total Ponderado [€]	Energia Contratada [MWh]	Energia Fornecida [MWh]	Energia Gerada Unidades Térmicas [MWh]		
				G1	G2	G3
0	4117780	119096	138084	4430	24909	108745
0,2	4092690	119096	120329	3270	28418	88641
0,4	3923691	119096	119135	3270	31800	84065
0,6	3729017	119096	119096	2990	34956	81150
0,8	3568313	119096	119096	2870	35578	80649
1	3408946	119096	119096	2870	35799	80427

Tabela 3.5: Custo e energias fornecidas e contratada em função de λ .

As energias fornecidas e a energia contratada em função de λ estão representadas no gráfico seguinte

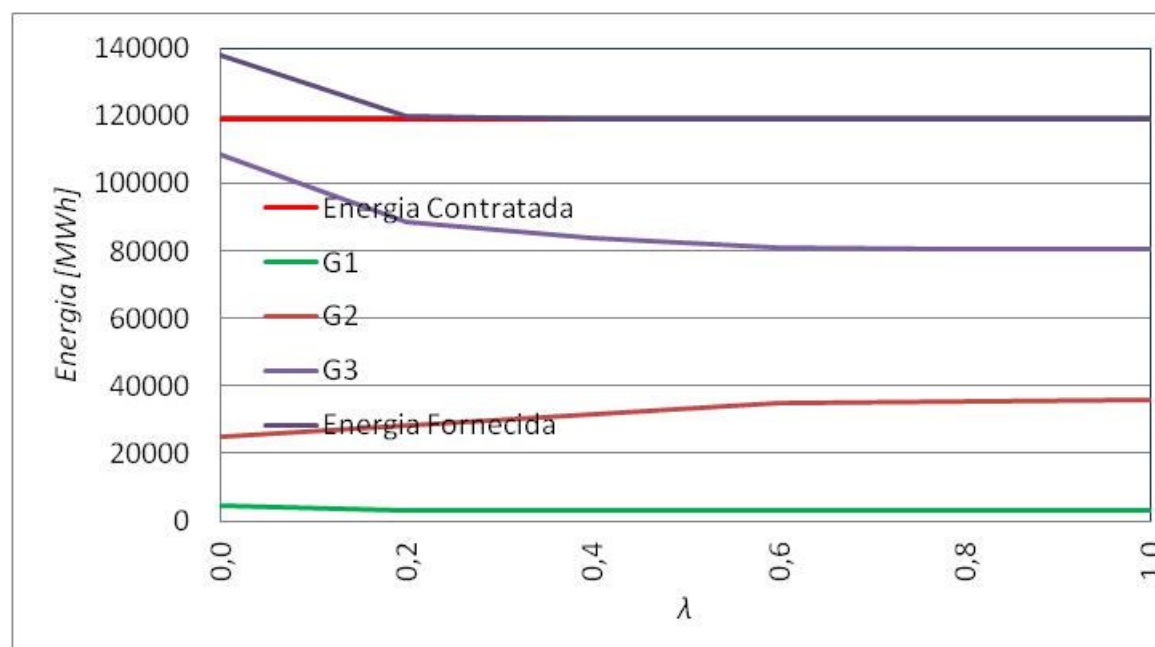


Figura 3.15: Energia contratada e energias fornecidas pelas unidades térmicas.

A maior diferença entre a energia contratada e a energia fornecida ocorre para $\lambda = 0$. Nesta situação, é nulo o custo das emissões.

Para $\lambda \geq 0,6$, a energia fornecida pelo conjunto das 3 unidades térmicas é igual ao da energia contratada. No entanto, a distribuição da energia pelas três unidades não é constante, o que é devido à diferença nos parâmetros do custo do combustível e das emissões das três unidades.

O défice operacional está representado no gráfico seguinte

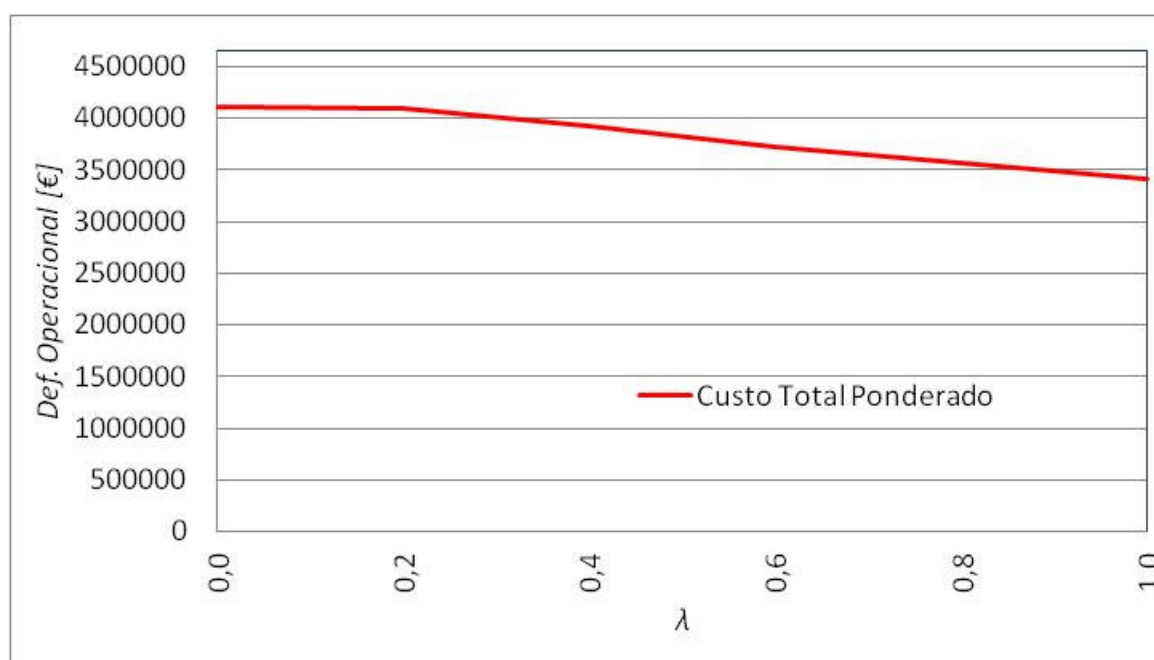


Figura 3.16: Déficit operacional em função de λ .

O défice operacional é decrescente com o aumento do valor de λ devido ao decréscimo do peso do custo do combustível e da facturação e aumento do peso do custo das emissões. Para $\lambda = 0$, o valor do défice operacional seria 4214556

€ caso a potência gerada fosse igual à fornecida, pois nesta situação a facturação seria nula.

Para valores superiores de λ , o valor a facturar perde peso perante o custo das emissões, pelo que a potência gerada tende a igualar o valor mínimo permitido, que é o valor da potência contratada.

Foram registados os valores da potência contratada, da potência fornecida, das potências geradas por cada unidade térmica e do défice operacional para três diferentes valores de λ .

Para $\lambda = 0$, os valores da potência contratada, da potência fornecida e do défice operacional são os seguintes

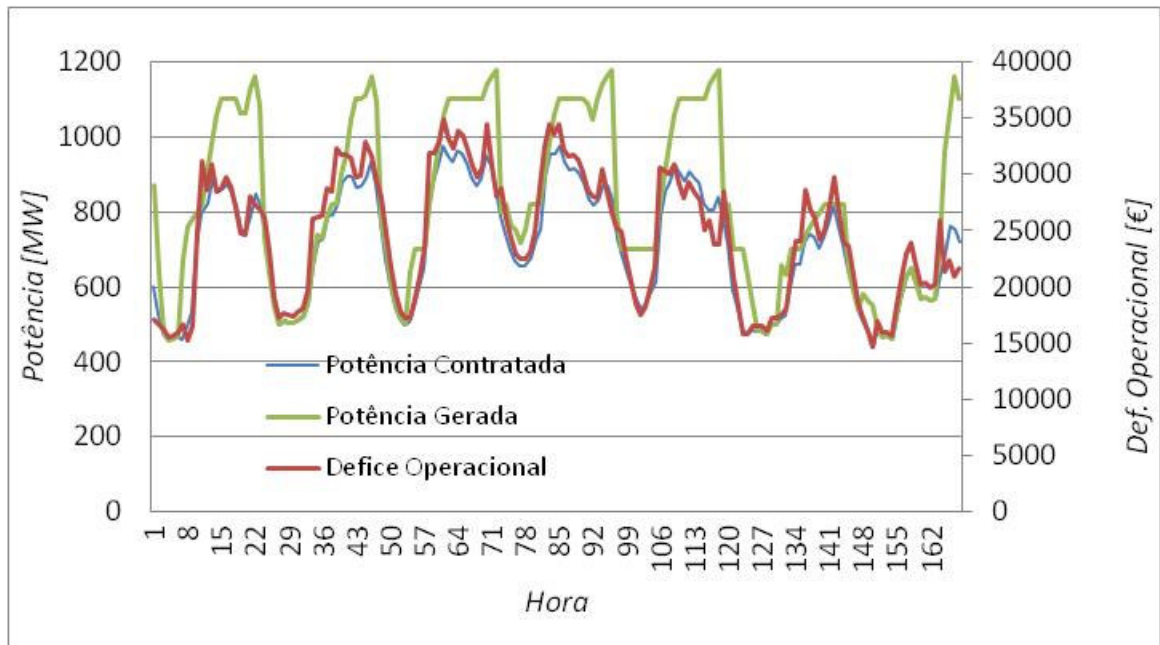


Figura 3.17: Potência contratada, potência fornecida e défice operacional para

$$\lambda = 0 .$$

Para os valores mais elevados do preço da energia eléctrica, a potência gerada é superior ao valor da potência contratada. Estes valores da potência aproximam-se para preços mais baixos da energia.

A potência gerada nesta situação por cada unidade térmica é a seguinte

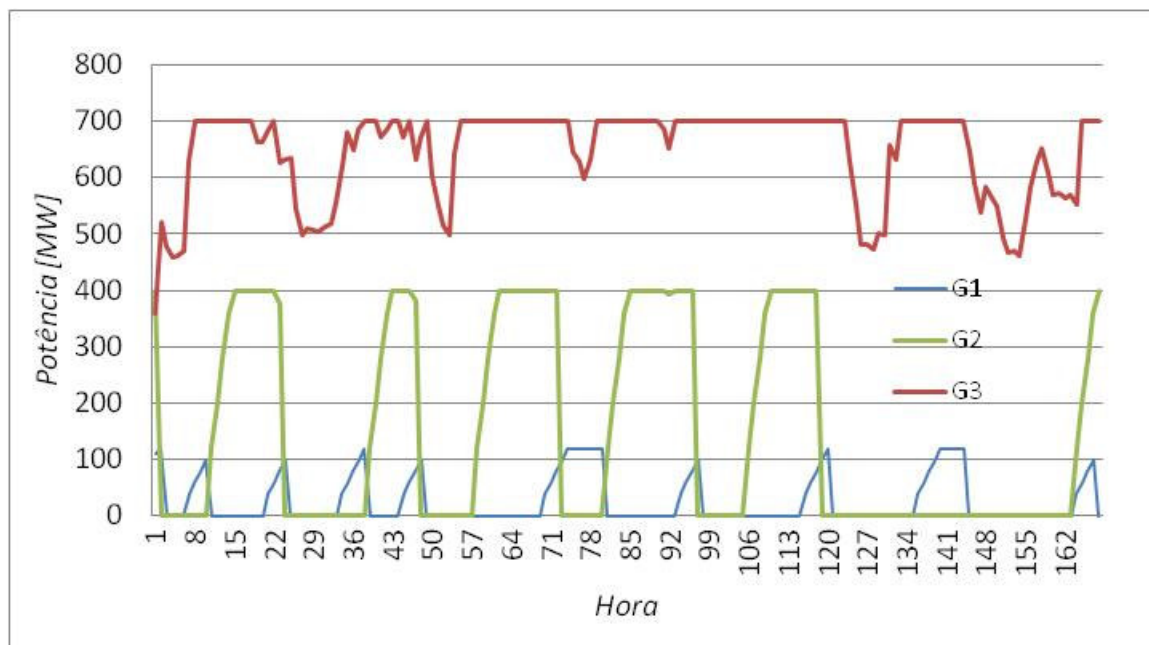


Figura 3.18: Potência gerada por cada unidade térmica para $\lambda = 0$.

Devido à diferença dos custos de arranque, tanto as unidades térmicas G1 como G2 páram quando o preço da energia é menor, o que não acontece com a unidade G3, que tem um custo de arranque superior ao das unidades G1 e G2.

Para $\lambda = 0,5$, os valores da potência contratada, da potência fornecida e do déficit operacional são os seguintes

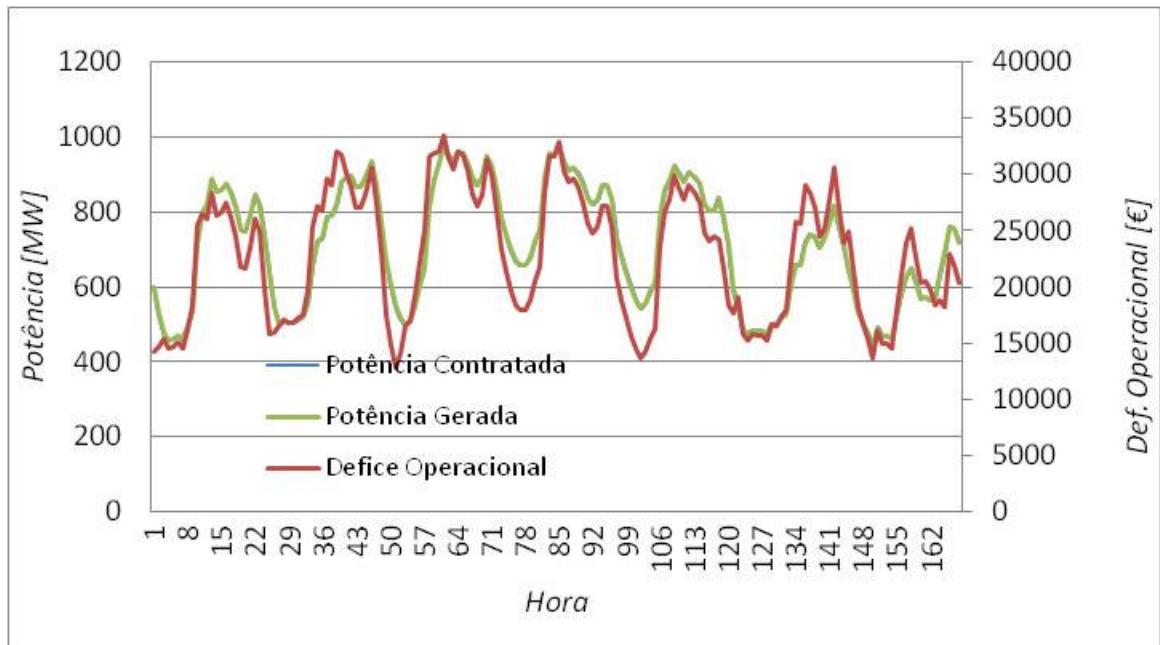


Figura 3.19: Potência contratada, potência fornecida e déficit operacional para $\lambda = 0,5$.

Com o aumento de λ , maior peso do custo das emissões e menor peso do custo do combustível e da facturação, a potência fornecida é igual à potência contratada. O valor da energia fornecida foi 119096 MWh e o valor do déficit operacional foi 3822524 €.

No caso de estudo anterior a unidade térmica G1 parava quando $\lambda > 0,70$. Neste caso a potência gerada iguala a contratada por força do contrato bilateral, pelo que há sempre geração de energia.

A potência gerada nesta situação por cada unidade térmica é a seguinte

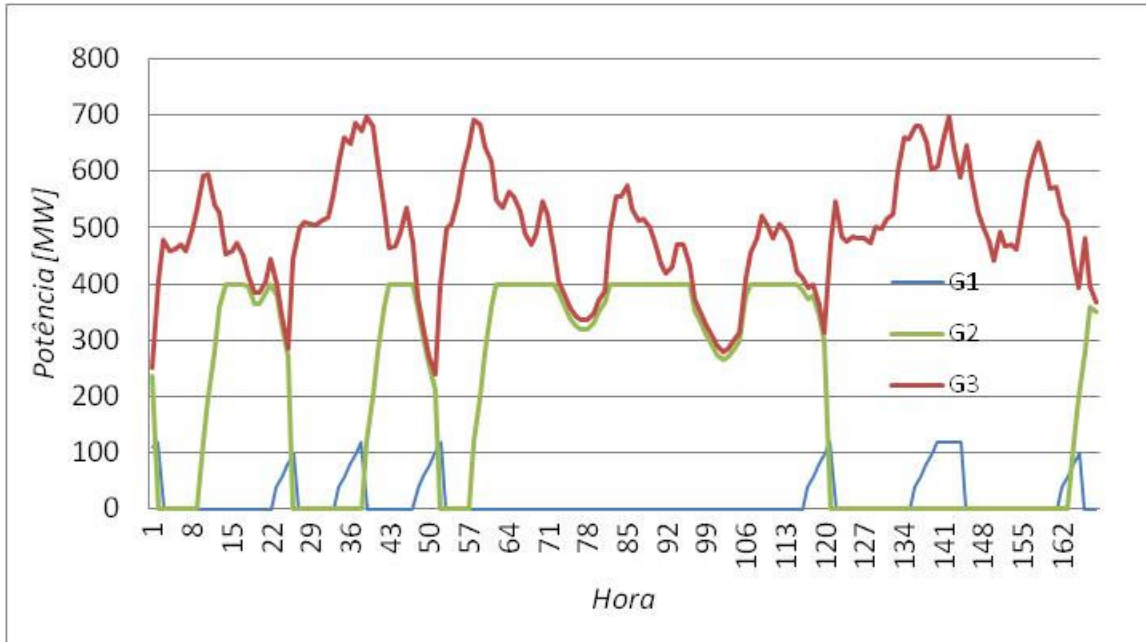


Figura 3.20: Potência gerada por cada unidade térmica para $\lambda = 0,5$.

Tal como para $\lambda = 0$, a unidade térmica G3 não pára devido a ter um custo de arranque superior ao das unidades térmicas G1 e G2.

Para $\lambda = 1$, os valores da potência contratada, da potência fornecida e do défice operacional são os seguintes

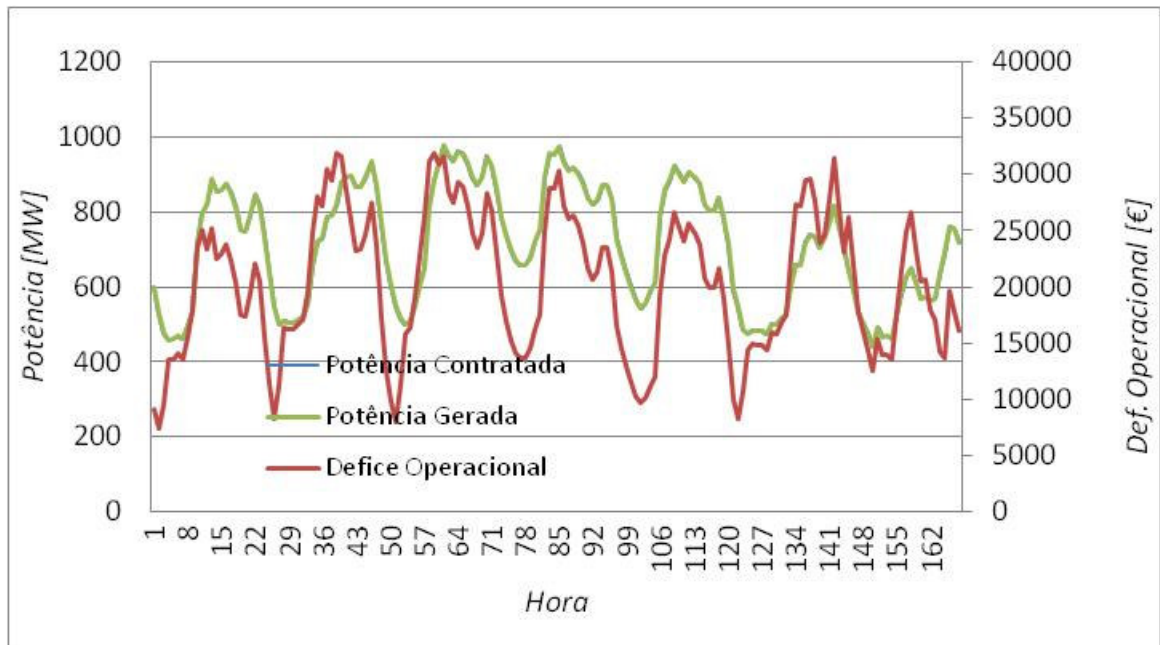


Figura 3.21: Potência contratada, potência fornecida e déficit operacional para

$$\lambda = 1.$$

Tal como para $\lambda = 0,5$, o valor da energia fornecida foi 119096 MWh, mas o valor do déficit operacional é inferior.

A potência gerada nesta situação por cada unidade térmica é a seguinte

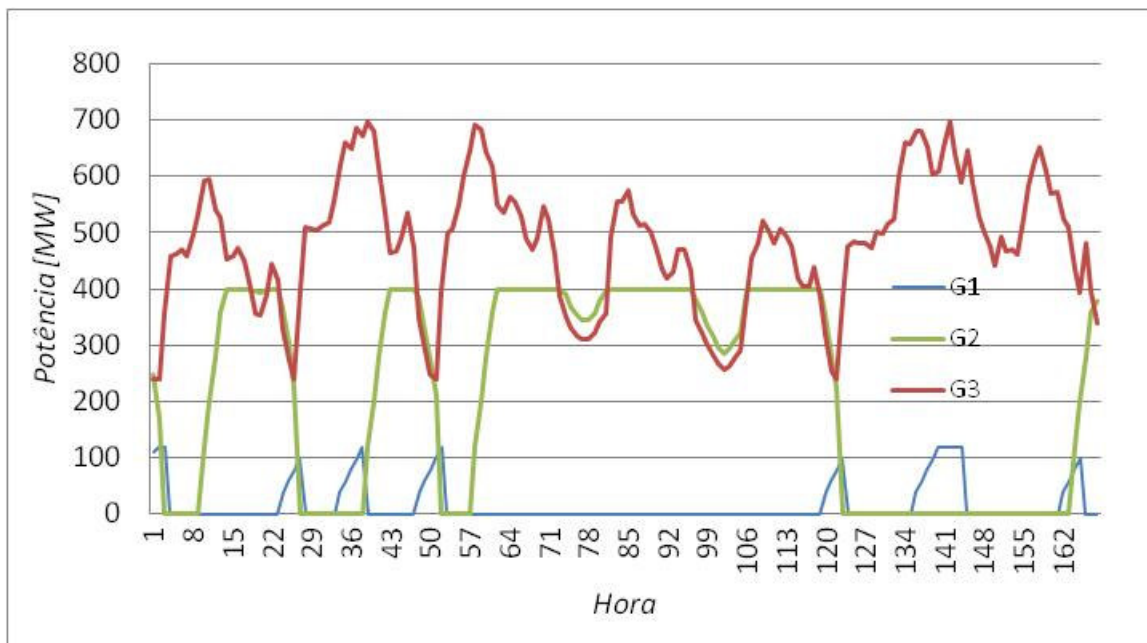


Figura 3.22: Potência gerada por cada unidade térmica para $\lambda = 1$.

Apesar do valor total da energia gerada para $\lambda = 1$ ser igual ao valor da energia gerada para $\lambda = 0,5$, a sua distribuição pelas unidades térmicas G2 e G3 é diferente devido aos diferentes parâmetros das funções custo do combustível e custo das emissões. Para estes dois valores de λ a unidade térmica G1 apenas entra em funcionamento para os menores valores da potência contratada.

Capítulo 4

Conclusões

Resumo

São apresentadas as conclusões sobre as técnicas utilizadas ao longo dos anos para resolver o problema de afectação das unidades térmicas e a evolução do problema de afectação das unidades térmicas.

4. Conclusões

4. Conclusões

As unidades de geração térmica são actualmente indispensáveis para assegurar o abastecimento de energia eléctrica. Como anteriormente mencionado, são responsáveis por 52,7 % do total de energia eléctrica produzida na Europa e por 67,5 % da energia produzida a nível mundial.

A energia eléctrica é um bem essencial, tanto para utilização doméstica como para utilização empresarial, sendo o seu custo determinante para a economia das empresas e dos países. Importa pois minimizar o seu custo, sendo um dos factores de custo a afectação das unidades térmicas.

O problema de afectação das unidades térmicas evoluiu em termos de técnicas utilizadas, do âmbito do problema e do próprio objectivo.

Numa primeira fase foram utilizadas listas de prioridades obtidas de forma heurística, sendo esta técnica ainda largamente utilizada no início da década de 80 do século XX [2].

O aumento da complexidade do problema, devido um maior detalhe do funcionamento das unidades térmicas e à inclusão de factores externos às unidades, obrigou à utilização de técnicas capazes de lidar com esta complexidade e capazes de fornecer soluções próximas do óptimo. Foram várias as técnicas utilizadas e ensaiadas ao longo dos anos, sendo frequentemente utilizadas as técnicas de avaliação e partição, técnicas de inteligência artificial e

a relaxação lagrangeana, sendo um marco importante a adopção desta técnica pela EDF em 1983 [4]. A utilização destas técnicas é possível devido ao aumento dos recursos computacionais disponíveis e embora possam produzir soluções próximas do óptimo, não garantem a solução óptima.

Com a liberalização dos mercados de energia, o objectivo do problema passou da minimização do custo para a maximização do resultado financeiro, sendo o preço da energia um novo factor a considerar na formulação do problema.

Os aspectos ambientais vieram acrescentar uma maior complexidade ao problema, tendo sido abordados nos artigos científicos de diferentes formas na formulação do problema: o valor da massa das emissões poluentes pode ser considerado como uma restrição ou pode ser atribuído um valor monetário às emissões, sendo assim um custo que é adicionado aos restantes custos de exploração das unidades térmicas. Se o problema for formulado com o objectivo de minimizar os custos de operação e simultaneamente minimizar o valor das emissões poluentes, então o planeamento de Afectação de Unidades é um problema de optimização multiobjectivo, com funções objectivo conflituosas.

Neste trabalho foram realizadas três simulações, recorrendo à programação dinâmica. Esta técnica garante a solução óptima, mas mesmo para o problema do terceiro caso de estudo, que inclui apenas três unidades térmicas, ficou

evidenciado a denominada maldição da dimensionalidade referenciada nos artigos científicos.

O tempo de processamento para determinar a solução ótima de afectação das três unidades num horizonte temporal de uma semana é de aproximadamente 40 minutos, num computador com 2 GB de memória RAM e microprocessador AMD Turion 64x2, com uma frequência de 1,9 GHz. O software utilizado foi o “MS-Excel”, com utilização do “Solver” para determinação da solução ótima de cada estádio a cada hora (estado). O exemplo citado tem 1120 estados, que resulta da combinação dos estádios das três unidades térmicas.

Referências

- [1]. C. K. Pang, H. C. Chen, "Optimal short-term thermal unit commitment", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-95, nº 4, July/August 1976
- [2]. C. K. Pang, G. B. Sheble, F. Albuyeh, "Evaluation of dynamic programming based methods and multiple area representation for thermal unit commitments", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-100, nº 3, March 1981
- [3]. A. I. Cohen, M. Yoshimura, "A branch-and-bound algorithm for unit commitment", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-102, nº 2, February 1983
- [4]. A. Merlin, P. Sandrin, "A new method for unit commitment at Electricite de France", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-102, nº 5, May 1983
- [5]. F. N. Lee, "A fuel-constrained unit commitment method", Optimization-based methods for operations scheduling", IEEE Transactions on Power Systems, Vol. 4, nº 3, August 1989
- [6]. C. C. Su, Y. Y. Hsu, "Fuzzy Dynamic Programming: an application to unit commitment", IEEE Transactions on Power Systems, Vol. 6, nº 3, August 1991
- [7]. F. N. Lee, Q. Feng, "Multi-area unit commitment", IEEE Transactions on Power Systems, Vol. 7, nº 2, May 1992
- [8]. C. Wang, S. M. Shahidehpour, "Effects on ramp-rate limits on unit commitment and economic dispatch", IEEE Transactions on Power Systems, Vol. 8, nº 3, August 1993
- [9]. D. Dasgupta, D. R. McGregor, "Thermal unit commitment using genetic algorithms", IEEE Transactions on Power Systems, Vol. 141, nº 5, September 1994
- [10]. D. Zhai, A. M. Breipohl, F.N. Lee, R. Adapa, "The effect of load uncertainty on unit commitment risk", IEEE Transactions on Power Systems, Vol. 9, nº 1, February 1994

- [11]. A. A. El-Keib, H. Ma, J. L. Hart, "Economic dispatch in view of the clean air act of 1990", IEEE Transactions on Power Systems, Vol. 9, No. 2, May 1994
- [12]. S. J. Wang, S. M. Shahidehpour, D. S. Kirschen, S. Mokhtari, G. D. Irisarri, "Short-term generation scheduling with transmission and environmental constraints using an augmented lagrangian relaxation", IEEE Transactions on Power Systems, Vol. 10, No. 3, August 1995
- [13]. S. A. Kazarlis, A. G. Bakirtzis, V. Petridis, "A genetic algorithm solution to the unit commitment problem", IEEE Transactions on Power Systems, Vol. 11, No. 1, February 1996
- [14]. C. Li, R. B. Johnson, A. J. Svoboda, "A new unit commitment method", IEEE Transactions on Power Systems, Vol. 12, No. 1, February 1997
- [15]. A. H. Mantawy, Y. L. Abdel-Magid, S. Z. Seliin, "A simulated annealing algorithm for unit commitment", IEEE Transactions on Power Systems, Vol. 13, No. 1, February 1998
- [16]. S. Sen D. P. Kothari, "Optimal thermal generating unit commitment: a review", Electrical Power & Energy Systems, Vol. 20, No. 7, pp. 443-451, 1998
- [17]. A. H. Mantawy, Y. L. Abde-Magid, S.Z. Selim, "Unit commitment by tabu search", IEEE Transactions on Power Systems, Vol. 145, No. 1, January 1998
- [18]. A. Rudolf, R. Bayrleithner, "A Genetic Algorithm for Solving the Unit Commitment Problem of a Hydro-Thermal Power System", IEEE Transactions on Power Systems, Vol. 14, No. 4, November 1999
- [19]. B. F. Hobbs, S. Jitprapaikularn, D. J. Maratukulam, "Analysis of the value for unit commitment of improved load forecasts", IEEE Transactions on Power Systems, Vol. 14, No. 4, November 1999
- [20]. A.H.Mantawy, Y. L. Abde -Magid, S.Z. Selim, "Integrating genetic algorithms, tabu search, and simulated annealing for the unit commitment problem", IEEE Transactions on Power Systems, Vol. 14, No. 3, August 1999
- [21]. C. P. Cheng, C. W. Liu, C. C. Liu, "Unit commitment by lagrangian relaxation and genetic algorithms", IEEE Transactions on Power Systems, Vol. 15, No. 2, May 2000

- [22]. C. W. Richter, G. B. Sheblé, "A profit-based unit commitment ga for the competitive environment", IEEE Transactions on Power Systems, Vol. 15, No. 2, May 2000
- [23]. K. S. Swarup, S. Yamashiro, "Unit commitment solution methodology using genetic algorithm", IEEE Transactions on Power Systems, Vol. 17, no. 1, February 2002
- [24]. A. L. Motto, F. D. Galiana, "Unit commitment with dual variable constraints", IEEE Transactions on Power Systems, Vol. 19, no. 1, February 2004
- [25]. C. A. Rajan, M. R. Mohan, "An evolutionary programming-based tabu search method for solving the unit commitment problem", IEEE Transactions on Power Systems, Vol. 19, no. 1, February 2004
- [26]. U. A. Ozturk, M. Mazumdar, B. A. Norman, "A solution to the stochastic unit commitment problem using chance constrained programming", IEEE Transactions on Power Systems, Vol. 19, no. 1, February 2004
- [27]. T. Saksornchai, W. J. Lee, K. Methaprayoon, J. R. Liao, R. J. Ross, "Improve the unit commitment scheduling by using the neural-network-based short-term load forecasting", IEEE Transactions on Power Systems, Vol. 41, no. 1, January/February 2005
- [28]. J. Catalão, S. Mariano, V. Mendes, L. Ferreira, "Unit commitment with environmental considerations: a practical approach", 15th PSCC, Liege, 22-26 August 2005
- [29]. Q. Zhou, R. Frowd, A. Papalexopoulos, D. Lamb, E. Ledesma, "Minimizing market operation costs using a security-constrained unit commitment approach", 2005 IEEE/PES Transmission and Distribution Conference & Exhibition: Asia and Pacific, Dalian, China
- [30]. I. J. Raglend, N. P. Padhy, "Solutions to practical unit commitment problems with operational, power flow and environmental constraints", Power Engineering Society General Meeting, 2006, IEEE
- [31]. J. Catalão, S. Mariano, V. Mendes, L. Ferreira, "Optimização da exploração de recursos térmicos considerando a restrição de emissões", J. Catalão et al. / Investigação Operacional, 26 (2006) 147-161
- [32]. D. N. Simopoulos, Y. S. Giannakopoulos, S. D. Kavatza, C. D. Vournas, "Effects of emission constraints on short-term unit commitment", IEEE Melecon 2006, May 16-19, Málaga, Spain

- [33]. L. Wu, M. Shahidehpour, T. Li, “Cost of reliability analysis based on stochastic unit commitment”, IEEE Transactions on Power Systems, Vol. 23, no. 3, August 2008
- [34]. F. Aminifar, M. F. Firuzabad, M. Shahidehpour, “Unit commitment with probabilistic spinning reserve and interruptible load considerations”, IEEE Transactions on Power Systems, Vol. 24, no. 1, February 2009
- [35]. C. Palanichamy, Natarajan Sundar Babu, “Day–night weather-based economic power dispatch”, IEEE Transactions on Power Systems, Vol. 17, no. 2, Maio 2002
- [36]. inversão térmica. In Infopédia [Em linha], Porto Editora, 2003-2011. [Consultado em 2011-04-11]; Disponível na www: <URL: [http://www.infopedia.pt/\\$inversao-termica,19](http://www.infopedia.pt/$inversao-termica,19)>.
- [37]. inversão térmica. In National Weather Service Forecast [Em linha], <http://www.wrh.noaa.gov/slc/climate/TemperatureInversions.php>
- [38]. The Nashua Telegraph (New Hampshire), “Donora: A temperature inversion sets the stage for the Donora Smog of 1948”, 26 de Outubro de 2009
- [39]. Preço e consumo de energia em Portugal [Em linha], OMEL, Mercado de Electricidad. [Consultado em 2011-05-14]; Disponível na www: <URL: <http://www.omel.es/pt/inicio>>
- [40]. John W. Chinneck, “Practical Optimization: a Gentle Introduction”, Systems and Computer Engineering, Carleton University, Ottawa, Canada
- [41]. Instituto de Apoio às Pequenas e Médias Empresas e à Inovação [Em linha], [Consultado em 2011-05-30]; Disponível na www: <URL: <http://www.iapmei.pt>>
- [42]. V. Mendes, Texto de Apoio à Unidade Curricular de Programação Matemática e Engenharia Industrial, Departamento de Engenharia Electrotécnica e Automação, Instituto Superior de Engenharia de Lisboa
- [43]. H I G H L I G H T SC. Palanichamy, Natarajan Sundar Babu, “Day–night weather-based economic power dispatch”, IEEE Transactions on Power Systems, Vol. 17, no. 2, Maio 2002

- [44]. Electricity Information; International Energy Agency, 2010
- [45]. CO₂ Emissions From Fuel Combustion Highlights, International Energy Agency, 2010
- [46]. Sistema Electrónico de Negociação de Direitos de Emissão de Dióxido de Carbono [Em linha]; Disponível na www: <URL: <http://www.sendeco2.com> >

Anexo I

Fundamentação Teórica

Resumo

São apresentadas duas técnicas relevantes para a resolução dos problemas de optimização da afectação de unidades térmicas: programação dinâmica e programação não linear.

AN1. Fundamentação Teórica

AN1.1 Programação Dinâmica

AN1.2 Programação Não Linear

AN1. Fundamentação Teórica

AN1.1 Programação Dinâmica [40]

A Programação Dinâmica é uma abordagem geral para tomar, de uma forma óptima, uma sequência de decisões interligadas. Enquanto podemos descrever as características gerais, os detalhes dependem da aplicação em questão. Fundamentalmente, o método é recursivo, como uma rotina de programação que se chama a si própria, adicionando em cada chamada informação a uma pilha, até que certas condições de paragem são satisfeitas. Atingido este ponto, a solução é revelada retirando, pela sequência correcta, a informação armazenada na pilha.

De forma geral o método é resumido por:

1. Definir uma pequena parte do problema e encontrar uma solução óptima para esta parte.
2. Alargar ligeiramente esta pequena parte do problema e encontrar a solução óptima para o novo problema utilizando a solução óptima encontrada previamente.
3. Continuar o passo 2 até que os sucessivos alargamentos do problema conduzam a que o problema actual englobe o problema original. Com este problema resolvido, estão satisfeitas as condições de paragem.

4. A solução do problema é construída a partir das soluções óptimas encontradas para os pequenos problemas resolvidos ao longo do processo.

O algoritmo de Dijkstra para procura do caminho mais curto é um caso clássico de programação dinâmica. A pequena parte do problema em cada estágio é simplesmente a determinação do nó seguinte mais próximo da origem. Este problema é ligeiramente alargado em cada estado pela inclusão de todos os nós e arcos não solucionados ligados directamente ao nó para o qual já foi encontrado a solução óptima. As condições de paragem são satisfeitas quando o nó seguinte mais próximo é o destino. Neste instante é construída a solução, percorrendo o problema em sentido inverso, traçando os arcos desde o nó de destino até ao de origem.

Há várias características importantes da programação dinâmica, de seguida descritas.

O problema pode ser dividido em estádios. No problema do caminho mais curto, cada estado constitui um novo problema que deve ser resolvido de forma a encontrar o nó seguinte mais próximo da origem. Em algumas aplicações da programação dinâmica, os estádios estão relacionados com o tempo, sendo esta a razão da designação programação dinâmica. Estas aplicações são frequentemente problemas de controlo dinâmicos e, por razões de eficiência, os estádios são resolvidos regredindo no tempo, ou seja, dum ponto no futuro até

ao presente. Isto é devido aos caminhos que levam do estado presente ao estado objectivo no futuro serem sempre apenas um subconjunto da totalidade dos caminhos possíveis a partir do estado actual, pelo que é mais eficiente percorrer este subconjunto de caminhos em sentido contrário.

Cada estágio tem um número de estados. Geralmente esta informação é suficiente para resolver o pequeno problema no estágio. No problema do caminho mais curto, o estado é o conjunto dos nós resolvidos, os arcos do conjunto de arcos e os arcos e nós não resolvidos directamente ligados aos nós resolvidos.

A decisão num estágio actualiza o estado no estágio em o estado para o estágio seguinte. No problema do caminho mais curto a decisão é o arco adicionado ao conjunto de arcos e o correspondente nó não resolvido que é adicionado ao conjunto de nós resolvidos. Isto obviamente afecta o conjunto de nós resolvidos, os nós e arcos não resolvidos e os arcos do conjunto de arcos, o que explica que a decisão actualiza o estágio para o estágio seguinte.

Dado o estado actual, a decisão óptima para os restantes estágios é independente das decisões tomadas nos estágios anteriores. Esta característica é o princípio da Optimalidade. Significa que é possível dividir o problema em problemas mais pequenos e resolver cada um dos problemas pequenos independentemente. Tomando como exemplo o problema do caminho mais

curto, é apenas relevante a distância total entre a origem e um nó resolvido, não sendo importante o caminho entre a origem e este nó resolvido. As decisões posteriores apenas utilizam a informação da distância, não tendo em conta o caminho, ou seja, não tendo em conta as decisões anteriores.

Há uma relação recursiva entre o valor da decisão num estádio e o valor das decisões óptimas nos estádios anteriores. De outra forma, a decisão óptima neste estádio utiliza as decisões óptimas previamente determinadas. Numa relação recursiva, a função aparece em ambos os lados da equação. Por palavras, a relação recursiva do caminho mais curto é escrita como:

$$\begin{aligned} & \text{(distância do caminho mais curto desde a origem até ao nó } i \text{)} \\ & = \min_j (\text{distância do caminho mais curto desde a origem até ao nó} \\ & \text{resolvido } j \text{)} \\ & + \text{(distância do arco mais curto desde o nó resolvido } j \text{ ao nó por} \\ & \text{resolver } i \text{)} \end{aligned}$$

De realçar que a “distância do caminho mais curto” aparece em ambos os lados da equação: é recursivo. Todas as relações recursivas da programação dinâmica mostram a função óptima nos dois lados da equação: o novo óptimo é sempre o resultado do óptimo anterior com um valor local. A relação não necessita ser uma adição tal como aqui mostrado, pode ser qualquer operação, tal como uma multiplicação ou qualquer outra relação abstracta.

Para formular uma solução da programação dinâmica, é necessário responder às questões seguintes:

Quais são os estados na solução?

Como é o estado definido no estágio?

Que tipo de decisão deve ser tomada num estágio?

De que forma a decisão actualiza o e estado para o próximo estágio?

Qual a relação do valor recursivo entre a decisão óptima num estágio e a decisão óptima anterior?

Para os problemas mais simples da programação dinâmica, os discretos e determinísticos, o sentido da resolução do problema é irrelevante. Há no entanto problemas que devem ser resolvidos em sentido inverso.

Tomando como exemplo um problema em que é necessário determinar os empregados necessários em cada mês do próximo ano, é necessário determinar o número óptimo de empregados tendo em conta o volume de trabalho. No entanto, empregados qualificados são difíceis de contratar, pelo que deverá ser evitada a dispensa de empregados, pois acarreta custos e poderá não ser possível contratar os empregados que posteriormente serão necessários; por outro lado, o excesso de empregados tem também custos excessivos. O problema é determinar o número de empregados a contratar e a dispensar em cada mês de forma a minimizar o custo total.

É muito mais eficiente resolver este problema em sentido contrário, começando no final do ano e acabando no início do ano, visto ser conhecido o número de empregados necessários em cada mês. Se este problema é resolvido no sentido do tempo, do primeiro para o último mês do ano, será necessário manter em aberto um número enorme de empregados que poderão ser necessários no futuro em vez dum número menor de empregados cuja necessidade é conhecida.

Modelos contínuos, tais como o nível da água numa barragem, podem ser abordados encontrando pontos de transição na grandeza contínua, tais como vazio, nível mínimo para abastecimento de água, nível mínimo para geração de energia, cheio. São também possíveis versões probabilísticas de programação dinâmica; o objectivo passa a ser a minimização ou maximização do valor esperado da relação do valor recursivo.

A programação dinâmica, embora seja morosa se resolvida manualmente, é muito eficiente se comparada com a força bruta de listar todas as opções possíveis para encontrar a melhor. Por exemplo, a procura de um caminho num grafo com 5 nós possíveis a cada 6 estados, totalmente ligado entre cada estado, obriga à pesquisa do melhor caminho entre $5^5=3125$ caminhos, e para cada um destes caminhos são necessárias 5 operações adicionais, o que resulta em 15625 operações. A utilização do algoritmo de Dijkstra de programação dinâmica obriga à utilização de apenas 105 operações, ou seja, a solução

encontrada através de programação dinâmica necessita apenas de 105 / 15625 $\approx 0,07$ do volume de trabalho.

AN1.2 Programação Não Linear

AN1.2.1 Preâmbulo

A programação não linear é o processo de resolução de problemas de otimização em que a função objectivo ou qualquer função associada com uma restrição não é linear. Por exemplo, é o caso do custo horário da energia gerada por uma unidade térmica, expressa normalmente por uma função não linear dada por

$$C(u, p) = (a + bp + \frac{1}{2}cp^2) u$$

quando $u = 1$, esta função quadrática do custo em função do tempo, expressa em €/h, é uma função convexa se $c \geq 0$ ou côncava se $c \leq 0$, sendo então uma função afim caso $c = 0$. Quando dividida pela potência será obtido um custo por unidade de potência, em €/kW, dado por

$$\frac{C(u, p)}{p} = (\frac{a}{p} + b + \frac{1}{2}cp) u \quad (1)$$

Quando $u = 1$, esta função é uma função convexa seja qual for o valor atribuído a c . Uma propriedade importante das funções convexas é a de que um mínimo local de uma função convexa num conjunto convexo de decisões admissíveis é um mínimo global.

Antes de abordar as técnicas que permitem identificar os pontos óptimos nas funções anteriormente descritas, importa listar um conjunto de definições e teoremas que permitirão definir o âmbito de aplicação destas técnicas [42].

AN1.2.2 Definições

Definição: Um conjunto $C \subseteq R^n$ é convexo, se $\forall x_1, x_2 \in C$ implicar $[x_1, x_2] \subseteq C$.

Definição: Uma combinação linear de um conjunto finito de pontos $\{x_1, \dots, x_m\}$ de R^n , dada por $\sum_{i=1}^m \lambda_i x_i$, é chamada combinação convexa se os escalares $\lambda_i \geq 0$ e $\sum_{i=1}^m \lambda_i = 1$.

Da definição anterior resultam as seguintes afirmações: o invólucro convexo de um conjunto é um conjunto convexo; um conjunto convexo é o invólucro convexo dos seus pontos, ou seja, o conjunto C é convexo sse $C = C_o(C)$.

Definição: Uma função $f : R^n \rightarrow R$ é localmente Lipschitziana no ponto $x \in R^n$, se $\exists L, \delta > 0$ tal que $|f(x_1) - f(x_2)| \leq L \|x_1 - x_2\|$ para $\forall x_1, x_2 \in B(x, \delta)$.

Uma função é diferenciável num ponto quando existe na vizinhança desse ponto uma boa aproximação linear para a função. Mais precisamente, deve existir uma

transformação linear $T: R^n \rightarrow R$ tal que $f(x+d) - f(x) = T(d) + \|d\| \varepsilon(x,d)$ com $\varepsilon(x,d) \rightarrow 0$ quando $\|d\| \rightarrow 0$. Geralmente, a transformação linear é indicada em álgebra matricial por $T(d) = T^T d$. Seja $\sigma(t)$ a função sinal de um número $\sigma(x) = 1$ se $x > 0$, $\sigma(x) = 0$ se $x = 0$ e $\sigma(x) = -1$ se $x < 0$. Caso a função f seja diferenciável num ponto resulta para $t \in R^+ \setminus \{0\}$

$$T(d) = \frac{T(td)}{t} = \frac{f(x+td) - f(x)}{t} - \sigma(t) \|d\| \varepsilon(x,td)$$

passando ao limite é válida a igualdade

$$T^T d = f'_d(x) = \lim_{t \rightarrow 0} \frac{f(x+td) - f(x)}{t}$$

ou seja, o valor da transformação linear $T(d)$ é a derivada direccional da função no ponto segundo o vector d . Ainda, é bem conhecido o seguinte resultado para esta derivada direccional

$$f'_d(x) = \nabla f(x)^T d$$

Logo o operador linear T identifica-se com o gradiente da função no ponto. Por isso a definição de função diferenciável pode ser enunciada como se segue.

Definição: Uma função $f: R^n \rightarrow R$ é diferenciável em $x \in R^n$ se $f(x+d) = f(x) + \nabla f(x)^T d + \|d\| \varepsilon(x,d)$ com $\varepsilon(x,d) \rightarrow 0$ quando $\|d\| \rightarrow 0$.

Se uma função é diferenciável num ponto então existem todas as derivadas direccionais dessa função nesse ponto. No entanto, é facto que a recíproca nem

sempre é verdadeira: a existência das derivadas direccionais de uma função num ponto não é condição suficiente para garantir a diferenciabilidade da função nesse ponto. O exemplo clássico deste facto é a função $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ definida por

$$f(x_1, x_2) = \frac{(x_1)^2 x_2}{(x_1)^2 + (x_2)^2}$$

se $(x_1, x_2) \neq (0, 0)$, $f(0, 0) = 0$.

Se $d = (d_1, d_2)$ então $f'(0, 0) = \frac{(d_1)^2 d_2}{(d_1)^2 + (d_2)^2}$. A função não é diferenciável na

origem, visto que, a derivada direccional não é uma função linear das coordenadas do vector $d = (d_1, d_2)$ no ponto $(0, 0)$. Para uma função real de n variáveis reais, caso se tenha que $n-1$ das derivadas parciais sejam contínuas, então a recíproca é válida.

Definição: Uma função é continuamente diferenciável num ponto, se tem derivadas parciais de primeira ordem contínuas nesse ponto.

Caso a função seja continuamente diferenciável em todos os pontos do seu domínio então diz-se que pertence à classe C^1 .

A menos que seja dito o contrário, as funções mencionadas neste texto são continuamente diferenciáveis.

Definição: Uma função $f: R^n \rightarrow R$ é diferenciável até à segunda ordem em $x \in R^n$ se $f(x+d) - f(x) = f(x) + \nabla f(x)^T d + \frac{1}{2} d^T \nabla^2 f(x) d + \|d\|^2 \varepsilon(x,d)$ com $\varepsilon(x,d) \rightarrow 0$ quando $\|d\| \rightarrow 0$.

Uma função é continuamente diferenciável até a segunda ordem, se tem derivadas parciais de segunda ordem contínuas.

Definição: Uma função $f: R^n \rightarrow R$ é decrescente se para $x_1, x_2 \in R^n$ tais que $x_1 \geq x_2$ implicar $f(x_1) \leq f(x_2)$.

Definição: Uma função $f: R^n \rightarrow R$ é estritamente decrescente se para $x_1, x_2 \in R^n$ tais que $x_1 > x_2$ implicar $f(x_1) < f(x_2)$.

Definição: Uma função $f: R^n \rightarrow R$ é fortemente decrescente se para $x_1, x_2 \in R^n$ tais que $x_1 \geq x_2$ implicar que $f(x_1) < f(x_2)$.

Definição: Uma função $f: C \subseteq R^n \rightarrow R$ num conjunto C convexo é convexa se $\forall x_1, x_2 \in C, f(\lambda x_1 + (1-\lambda)x_2) \leq \lambda f(x_1) + (1-\lambda) f(x_2)$ com $0 \leq \lambda \leq 1$.

Portanto uma função é convexa se para $\forall x_1, x_2 \in C$ o valor que a função toma em qualquer ponto de interpolação $x \in [x_1, x_2]$ nunca for superior ao valor da interpolação para esse ponto.

A definição anterior é equivalente a afirmar: uma função $f : C \subseteq R^n \rightarrow R$ num

conjunto C convexo é convexa se $\forall x_1, x_2, \dots, x_p \in C$, $f\left(\sum_i^p \lambda_i x_i\right) \leq \sum_i^p \lambda_i f(x_i)$

com $\sum_i^p \lambda_i = 1$ e $\lambda_i \geq 0$.

Definição: Uma função num conjunto C convexo, $f : C \subseteq R^n \rightarrow R$, é estritamente convexa se $\forall x_1, x_2 \in C$ com $f(x_1) \neq f(x_2)$, $f(\lambda x_1 + (1-\lambda)x_2) < \lambda f(x_1) + (1-\lambda)f(x_2)$ e $0 \leq \lambda \leq 1$.

Definição: Uma função $f : C \subseteq R^n \rightarrow R$ diferenciável num conjunto C convexo é pseudoconvexa, se $\forall x_1, x_2 \in C$ com $\nabla f(x_1)^T (x_2 - x_1) \geq 0$ implicar $f(x_2) \geq f(x_1)$.

Portanto uma função é pseudoconvexa se sempre que a sua derivada direccional for não negativa a função não decrescer segundo essa direcção.

Definição: Uma função $f : C \subseteq R^n \rightarrow R$ conjunto C convexo é quaseconvexa, se $\forall x_1, x_2 \in C$, $f(\lambda x_1 + (1-\lambda)x_2) \leq \max\{f(x_1), f(x_2)\}$ com $0 \leq \lambda \leq 1$.

Toda a função convexa é uma função quaseconvexa, visto que, pela propriedade transitiva é válida a desigualdade anterior para as funções convexas, ou seja, $f(\lambda x_1 + (1-\lambda)x_2) \leq \lambda f(x_1) + (1-\lambda)f(x_2) \leq \max\{f(x_1), f(x_2)\}$.

Contudo, convém referir que uma combinação linear com coeficientes positivos de funções quaseconvexas não é necessariamente uma função quaseconvexa, como acontece com as funções convexas: uma combinação linear com coeficientes positivos de funções convexas é uma função convexa.

Definição: Uma função $f : C \subseteq R^n \rightarrow R$ conjunto C convexo é estritamente quaseconvexa, se $\forall x_1, x_2 \in C$ com $f(x_1) \neq f(x_2)$, $f(\lambda x_1 + (1-\lambda)x_2) < \max\{f(x_1), f(x_2)\}$ e $0 < \lambda < 1$.

Definição: Uma função $f : R^n \rightarrow R$ é crescente se para $x_1, x_2 \in R^n$ tais que $x_1 \geq x_2$ implicar que $f(x_1) \geq f(x_2)$.

Definição: Uma função $f : R^n \rightarrow R$ é estritamente crescente se para $x_1, x_2 \in R^n$ tais que $x_1 > x_2$ implicar que $f(x_1) > f(x_2)$.

Definição: Uma função $f : R^n \rightarrow R$ é fortemente crescente se para $x_1, x_2 \in R^n$ tais que $x_1 \geq x_2$ implicar que $f(x_1) > f(x_2)$.

Ou seja uma função é respectivamente decrescente, estritamente decrescente, fortemente decrescente se a sua simétrica for respectivamente uma função crescente, estritamente crescente, fortemente crescente. Uma função é monótona se for crescente ou se for decrescente.

Definição: Uma função $f : C \subseteq R^n \rightarrow R$ num conjunto C convexo é côncava se $\forall x_1, x_2 \in C, f(\lambda x_1 + (1-\lambda)x_2) \geq \lambda f(x_1) + (1-\lambda)f(x_2)$ com $0 \leq \lambda \leq 1$.

Portanto uma função é côncava se para $\forall x_1, x_2 \in C$ o valor que a função tem em qualquer ponto $x \in [x_1, x_2]$ nunca for inferior ao valor que a interpolação da função pelos pontos x_1 e x_2 tem para esse ponto.

Esta definição é equivalente a afirmar: uma função $f : C \subseteq R^n \rightarrow R$ num conjunto

C convexo é côncava se $\forall x_1, x_2, \dots, x_p \in C, f\left(\sum_i^p \lambda_i x_i\right) \geq \sum_i^p \lambda_i f(x_i)$ com

$$\sum_i^p \lambda_i = 1 \text{ e } \lambda_i \geq 0.$$

Definição: Uma função $f : C \subseteq R^n \rightarrow R$, num conjunto C convexo é estritamente côncava se $\forall x_1, x_2 \in C$ com $f(x_1) \neq f(x_2)$, $f(\lambda x_1 + (1-\lambda)x_2) > \lambda f(x_1) + (1-\lambda)f(x_2)$ e $0 < \lambda < 1$.

Teorema: Uma função $f : C \subseteq R^n \rightarrow R$ diferenciável num conjunto C convexo é côncava, sse $\forall x, y \in C, f(y) \leq f(x) + \nabla f(x)^T (y - x)$.

Definição: Uma função $f : C \subseteq R^n \rightarrow R$ diferenciável num conjunto C convexo é pseudocôncava, se $\forall x_1, x_2 \in C$, com $\nabla f(x_1)^T (x_2 - x_1) \leq 0$ implicar $f(x_2) \leq f(x_1)$.

Portanto, uma função é pseudocôncava se for diferenciável e sempre que a sua derivada direccional não for positiva a função não cresce segundo essa direcção. A definição e o teorema anteriores permitem concluir que uma função côncava diferenciável é necessariamente pseudocôncava, mas a contrária não é necessariamente verdade, ou seja, existem funções pseudocôncavas que não são funções côncavas. Por exemplo a função pseudocôncava $f : R \rightarrow R$ dada por $f(x) = e^{-x}$ que é uma função convexa. Resultado idêntico para as funções convexas, ou seja, se uma função for convexa e diferenciável, então é pseudoconvexa.

Definição: Uma função $f : C \subseteq R^n \rightarrow R$ num conjunto C convexo é quasecôncava, se $\forall x_1, x_2 \in C$, $f(\lambda x_1 + (1-\lambda)x_2) \geq \min\{f(x_1), f(x_2)\}$ com $0 \leq \lambda \leq 1$.

Portanto, uma função é quasecôncava se para qualquer $[x_1, x_2] \subseteq C$ o valor da função no ponto $x \in [x_1, x_2]$ nunca for inferior ao menor dos valores da função nos pontos extremos x_1 e x_2 do intervalo. Uma função pseudocôncava é necessariamente quasecôncava, mas a contrária não é necessariamente verdade. Basta observar que as funções quasecôncavas não são

necessariamente diferenciáveis. Resultado idêntico para as funções convexas, ou seja, se uma função for pseudoconvexa, então é quaseconvexa.

Definição: Uma função $f : C \subseteq R^n \rightarrow R$ num conjunto C convexo é estritamente quasecôncava, se $\forall x_1, x_2 \in C$ com $f(x_1) \neq f(x_2)$, $f(\lambda x_1 + (1-\lambda)x_2) > \min\{f(x_1), f(x_2)\}$ e $0 < \lambda < 1$.

Portanto, uma função é respectivamente côncava, estritamente côncava, pseudocôncava, quasecôncava ou estritamente quasecôncava se a sua simétrica for respectivamente uma função convexa, estritamente convexa, pseudoconvexa, quaseconvexa ou estritamente quaseconvexa.

Algumas propriedades das funções quaseconvexas (quasecôncavas) são:

- 1 Seja $f : C \subseteq R^n \rightarrow R$ uma função quaseconvexa (quasecôncava) num conjunto C convexo e $g : f(C) \subseteq R \rightarrow R$ uma função monótona não decrescente então a função composta $g \circ f : C \subseteq R^n \rightarrow R$ é uma função quaseconvexa (quasecôncava);
- 2 Seja $f : C \subseteq R^n \rightarrow R$ uma função quaseconvexa (quasecôncava) num conjunto C convexo não negativa ou não positiva então $g : C \subseteq R^n \rightarrow R$ com $g(x) = 1/f(x)$ é uma função quasecôncava (quaseconvexa).

Contudo, para as funções convexas não é válido que se $f : C \subseteq R^n \rightarrow R$ é função convexa não negativa ou não positiva num conjunto C convexo

então $g : C \subseteq R^n \rightarrow R$ com $g(x) = 1/f(x)$ é uma função côncava. Por exemplo, a função real de variável real $f(x) = e^{-x}$ é uma função convexa não negativa e a $g(x) = 1/f(x)$ é também uma função convexa não negativa.

Definição: Um problema de otimização é constituído por

1 - uma função vectorial $f : R^n \rightarrow R^p$ dita vector das funções objectivo

2 - um conjunto $G \subseteq R^n$

3 - um conjunto de condições

$g_1(x), g_2(x), \dots, g_j(x), \dots, g_m(x) \leq 0$ em que $g_j : R^n \rightarrow R$ são funções.

Se $p \neq 1$ o problema é dito multiobjectivo, caso contrário a única função $f : R^n \rightarrow R$ é dita de função objectivo. Os pontos 2 e 3 são redundantes, mas é justificável que ambas possam coexistir. Pode ser difícil caracterizar os pontos do conjunto de condições $g_1(x), g_2(x), \dots, g_j(x), \dots, g_m(x) \leq 0$, enquanto o conjunto $G \subseteq R^n$ pode ter uma fácil caracterização dos seus pontos, podendo interessar a separação para um tratamento particularizado. Quando isso não for importante então o problema de otimização é constituído de acordo com os pontos 1 e 2 da definição. Assim, o problema pode ser formulado como um

problema de minimização de programação matemática multiobjectivo e ser descrito por

$$\min \{ f(x) = (f_1(x), f_2(x), \dots, f_i(x), \dots, f_p(x)) : x \in G \subseteq R^n \} \quad (2)$$

O vector das funções objectivo $f : R^n \rightarrow R^p$ é constituído pelas p coordenadas, funções $f_j : R^n \rightarrow R$ ditas de funções objectivo. O vector x é dito de vector das variáveis de decisão e está restringido ao conjunto G das decisões admissíveis, que será admitido como um conjunto não vazio e caracterizado por m funções restricionais $g_j : R^n \rightarrow R$ componentes da função vectorial $g : R^n \rightarrow R^m$.

$$G = \{ x \in R^n : g_1(x), g_2(x), \dots, g_j(x), \dots, g_m(x) \leq 0 \} \quad (3)$$

Uma decisão a que corresponda um vector que pertence ao conjunto G é dita de admissível. Uma descrição mais conveniente em notação de programação matemática que (2) é a seguinte

$$\min \{ (f_1(x), f_2(x), \dots, f_i(x), \dots, f_p(x)) \}$$

s.a

$$x \in G \subseteq R^n \quad (4)$$

A sigla *min* significa minimizar, sendo lida em (4) como: minimizar simultaneamente as funções objectivo por escolha do vector x sujeita a pertencer ao conjunto G . O significado de minimizar simultaneamente as funções objectivo será caracterizado mais adiante. É admitido neste texto que o problema (4) é um problema com solução.

Em particular se só existir em (4) uma função objectivo o problema será dito só de problema de programação matemática e a sua solução escrita como

$$\begin{aligned} \min f(x) &= f(x^*) \\ \text{s.a} \\ x &\in G \subseteq R^n \end{aligned} \quad (5)$$

sendo lido (5) como: minimizar a função objectivo por escolha do vector x sujeita a pertencer ao conjunto das decisões admissíveis G tem como solução $f(x^*)$.

Através de um processo de solução de (5), obtém-se o valor μ para o mínimo global de (5) que é por definição o maior dos minorantes da função objectivo no conjunto G . Um ponto $x^* \in G$ é dito de solução óptima, no sentido de solução óptima global, se o valor óptimo do problema μ é dado por $f(x^*)$.

Maximizar um vector de funções objectivo por escolha do vector x sujeita a pertencer ao conjunto G é uma transformação imediata de um problema de minimização, visto que,

$$\begin{aligned} \max \{ f(x) = (f_1(x), f_2(x), \dots, f_j(x), \dots, f_p(x)) : x \in G \subseteq R^n \} = \\ - \min \{ -f(x) = (-f_1(x), -f_2(x), \dots, -f_j(x), \dots, -f_p(x)) : x \in G \subseteq R^n \} \end{aligned}$$

Como $g_j(x) \geq 0$ é equivalente a $-g_j(x) \leq 0$ e ainda como $g_j(x) = 0$ é equivalente a $g_j(x) \leq 0$ e $-g_j(x) \leq 0$, então a formulação (3) para o conjunto G inclui também a desigualdade maior ou igual e a igualdade. Consequentemente,

o problema de programação matemática como se indica em (4) é uma formulação suficientemente genérica.

Definição: Uma decisão dada pelo vector $x^* \in G$ é dita de óptima para o problema (5) se não existir $x \in G$ tal que $f(x) \leq f(x^*)$.

Definição: Uma decisão dada pelo vector $x^* \in G$ é dita de ε -óptima para o problema (5) se existir $f(x^*) - \mu \leq \varepsilon$.

Uma decisão é óptima se for ε -óptima para o problema (5), mas a contrária não é necessariamente verdade, ou seja, se uma decisão é ε -óptima não é necessariamente óptima para o problema (5) a não ser que seja 0-óptima.

Definição: Uma decisão dada pelo vector $x^* \in G$ é dita de localmente óptima para o problema (5), se existir $\delta > 0$ e a vizinhança $B(x^*, \delta)$ tal que não exista $x \in B(x^*, \delta) \cap G$ com $f(x) \leq f(x^*)$.

A resolução só com uma das funções objectivo de (4) dos seguintes problemas de programação matemática para $i = 1, 2, \dots, p$

$$\begin{aligned} \min f_i(x) &= f_i(x_i^*) \\ \text{s.a} \\ x &\in G \subseteq R^n \end{aligned} \tag{6}$$

serve para introduzir alguns conceitos importantes, para a programação matemática multiobjectivo, como o de vector objectivo ideal ou o de funções objectivos conflitantes.

Definição: O vector $Z = f(G) = \{z : z = f(x) \text{ com } x \in G\}$ é dito de conjunto dos vectores objectivo admissíveis para o problema (4).

Definição: O conjunto $z^* = (f_1(x_1^*), f_2(x_2^*), \dots, f_i(x_i^*), \dots, f_p(x_p^*))$, cujas coordenadas são determinadas pelas soluções de (5), é dito de vector objectivo ideal para o problema (4).

Definição: Se não existir um vector $x^* = x_1^* = x_2^* = \dots = x_i^* = \dots = x_p^*$, solução de todos os problemas (5), então as funções objectivos para o problema (4) são ditas de conflitantes.

Definição: Se as funções objectivos de (4) têm grandezas com unidades diferentes são ditas de incomensuráveis.

Caso as funções objectivos não sejam conflitantes, ou seja o vector objectivo ideal é admissível, então o problema descrito por (4) é equivalente a qualquer um dos p problemas (5), ou seja, não é necessário o problema multiobjectivo, bastando considerar um só objectivo. Caso o problema tenha só uma função objectivo então Z é um subconjunto de R que é totalmente ordenado pelo que

o conceito de mínimo e de máximo é bem definido matematicamente. Caso contrário, $m \geq 2$, Z é só parcialmente ordenado, logo no problema (4) minimizar simultaneamente as funções objectivos terá que ser caracterizado em conformidade com a relação de ordem parcial. Em Z , alguns vectores objectivo do problema são caracterizados por componentes cujo valor numérico não pode ser melhorado, ou seja, diminuído, sem que haja deterioração, aumento, em pelo menos um dos valores numéricos das outras componentes do vector objectivo.

Em 1881 Edgeworth usou esta caracterização para definir o que hoje é conhecido por solução não inferior. Esta definição sofreu tratamento posterior por Vilfredo Pareto e passou a ser conhecida por condição de óptimo de Pareto ou como Stadler escreve condição de óptimo de Edgeworth-Pareto.

Definição: Uma decisão dada pelo vector $x^* \in G$ é dita de óptima de Pareto para o problema (4) se não existir $x \in G$ tal que $f(x) \leq f(x^*)$. Um vector objectivo $z^* \in Z$ é dito de óptimo de Pareto se não existir $z \in Z$ tal que $z \leq z^*$.

Esta definição usa o conceito de relação de semipositivo: $x \in R^n$ é dito de semipositivo, $x \geq 0$, se $x_i \geq 0$ com $i = 1, 2, \dots, n$ mas $x \neq 0$. A escrita da definição é equivalente a afirmar que: Uma decisão dada pelo vector $x^* \in G$ é dita de óptima de Pareto se não existir $x \in G$ tal que $f_j(x) \leq f_j(x^*)$ para $i = 1, 2, \dots, p$ e $f_j(x) \neq f_j(x^*)$ para pelo menos um índice. Um vector objectivo $z^* \in Z$ é dito de óptimo de Pareto se não existir $z \in Z$ tal que $z_j \leq z_j^*$ para

$i = 1, 2, \dots, p$ e $z_j \neq z_j^*$ para pelo menos um índice. O nome não inferior é utilizado para descrever os vectores nas condições da definição de óptimo de Pareto.

Definição: O conjunto dos vectores objectivos óptimos de Pareto é dito de conjunto dos óptimos de Pareto.

Definição: Uma decisão dada pelo vector $x^* \in G$ é dita de localmente óptima de Pareto para o problema (4), se existir $\delta > 0$ e a vizinhança $B(x^*, \delta)$ tal que não exista $x \in B(x^*, \delta) \cap G$ com $f(x) \leq f(x^*)$. Um vector objectivo $z^* \in Z$ é dito de localmente óptimo de Pareto se corresponder a uma decisão localmente óptima de Pareto.

AN1.2.3 Identificação do ponto óptimo

As condições para a identificação do óptimo do problema de programação matemática têm tratamento diferente caso as função objectivo e as funções restricionais sejam ou não sejam continuamente diferenciáveis. O estudo do problema de programação matemática (5) das funções continuamente diferenciáveis, tal como a função do custo por unidade de potência das unidades de geração térmica (1), aqui introduzido. São apresentados resultados fundamentais derivados do estudo das propriedades do problema de

programação matemática só com uma função objectivo, concluindo condições de óptimo para o problema.

AN1.2.3.1 Teorema de Karush Kuhn Tucker

O teorema de Karush Kuhn Tucker é uma condição necessária, mas não suficiente a menos que sejam verificadas certas condições de suficiência, para a identificação de um ponto óptimo de um problema de programação matemática. Este teorema generaliza o conhecido resultado da condição necessária de num ponto óptimo o gradiente da função objectivo ser nulo, quando não existem restrições no problema (5). Será demonstrado o teorema de Karush Kuhn Tucker.

O problema de programação matemática (5) será considerado escrito como

$$\min f(x) = f(x^*)$$

s.a

$$g_1(x) \leq 0$$

$$g_2(x) \leq 0$$

.

.

.

$$g_m(x) \leq 0$$

$$\text{sendo } G = \{x \in R^n : g_1(x), g_2(x), \dots, g_j(x), \dots, g_m(x) \leq 0\} \quad (7)$$

Definição: O conjunto $A(x^*) = \{j \in \{1, 2, \dots, m\} : g_j(x^*) = 0\}$ é o conjunto das restrições activas no ponto $x \in R^n$.

Definição: Uma direcção $d \in R^n$ é admissível no ponto $x^* \in R^n$ se $\exists \tau > 0$ tal que $x + \alpha d \in G$ para $0 < \alpha \leq \tau$.

Definição: O conjunto $D(x) = \{d \in R^n : \exists \tau > 0 \text{ e } x + \alpha d \in G \text{ para } 0 < \alpha \leq \tau\}$ é o conjunto das direcções admissíveis no ponto $x \in R^n$.

Este conjunto não é necessariamente fechado, visto que, algumas direcções limites podem não estar contidas no conjunto. Por exemplo, seja $G = \{x \in R^2 : g(x) = \|x\| \leq 4\}$ no ponto $x = (0, 2)$ é fácil de verificar que $D(x) = \{(d_1, d_2) \in R^2 : d_1 < 0\}$, ou seja, a restrição é activa e as direcções neste ponto perpendiculares à curva que descreve o lugar geométrico dos pontos em que a restrição é activa não são admissíveis pelo que o conjunto das direcções admissíveis é aberto.

Teorema 1: Seja a função objectivo do problema de programação matemática (7) continuamente diferenciável no ponto x^* . Seja x^* uma solução óptima para o problema (7), então $\nabla f(x^*)^T d \geq 0$ para $d \in D(x^*)$.

Prova: Por contradição. Seja x^* uma solução óptima e seja para um $d \in D(x^*)$ $\nabla f(x^*)^T d < 0$, então por definição de direcção admissível existe um $\exists \alpha > 0$ tal que $x^* + \alpha d \in G$ e $f(x^*) > f(x^* + \alpha d)$ o que contradiz a hipótese.

O teorema anterior é também válido para os vectores pertencentes ao fecho do conjunto das direcções admissíveis como prova o seguinte teorema.

Teorema 2: Seja a função objectivo do problema de programação matemática (7) continuamente diferenciável no ponto x^* . Seja x^* uma solução óptima para o problema (7), então $\nabla f(x^*)^T d \geq 0$ para $d \in \bar{D}(x^*)$.

Prova: Seja $d_k \in D(x^*)$ uma série de direcções admissíveis convergente para $d \in \bar{D}(x^*)$, então a série de números $\nabla f(x^*)^T d_k$, é convergente e como $\nabla f(x^*)^T d_k \geq 0$ será $\nabla f(x^*)^T d \geq 0$.

Teorema 3: O conjunto $\bar{D}(x) \subseteq \{d \in R^n : \nabla g_j(x)^T d \leq 0 \text{ para } j \in A(x)\}$.

Prova: Por contradição. Seja $d \in D(x)$ e $\nabla g_j(x)^T d \leq 0$ para $g_j(x) = 0$, então por definição de direcção admissível existe um $\exists \alpha > 0$ tal que e

$g_j(x) < g_j(x + \alpha d)$ o que contradiz a hipótese, visto que, $g_j(x + \alpha d) > 0$. Logo se $d \in D(x)$ então $d \in \{d \in R^n : \nabla g_j(x)^T d \leq 0 \text{ para } j \in A(x)\}$. Este último conjunto é um conjunto fechado, logo o fecho do conjunto $D(x)$ tem que estar incluído nele.

O que se pretende é demonstrar que $\overline{D}(x) = \{d \in R^n : \nabla g_j(x)^T d \leq 0 \text{ para } j \in A(x)\}$, para tal é agora necessário demonstrar $\{d \in R^n : \nabla g_j(x)^T d \leq 0 \text{ para } j \in A(x)\} \subseteq \overline{D}(x)$. Esta afirmação nem sempre é verdadeira, para ser necessita de condições de regularidade ditas de qualificação restricional. Nos problemas que são estudados no âmbito da aplicação deste texto e se as restrições forem escritas na sua formulação matemática mais simples a qualificação restricional é em geral verificável. Assim, será admitida a seguinte hipótese.

Hipótese da Qualificação Restricional: Num ponto x^* do problema (7) é válida a seguinte afirmação $\{d \in R^n : \nabla g_j(x^*)^T d \leq 0 \text{ para } j \in A(x^*)\} \subseteq \overline{D}(x^*)$.

A igualdade $\overline{D}(x^*) = \{d \in R^n : \nabla g_j(x^*)^T d \leq 0 \text{ para } j \in A(x^*)\}$ é então válida assumida a hipótese de qualificação restricional. O Teorema 4.2 é rescrito como se segue.

Teorema 4: Seja a função objectivo do problema de programação matemática (7) continuamente diferenciável no ponto x^* . Seja x^* uma solução óptima para o problema (7), então $\nabla f(x^*)^T d \geq 0$ para $d \in \{d \in R^n : \nabla g_j(x^*)^T d \leq 0 \text{ para } j \in A(x^*)\}$.

O Teorema 4 afirma que o $\nabla f(x^*)^T d \geq 0$ com o $\nabla g_j(x^*)^T d \leq 0$ para $j \in A(x^*)$, então é aceitável afirmar que $\exists \lambda_j \geq 0$ para $j \in A(x^*)$ tal que

$$\left[\nabla f(x^*) + \sum_{j \in A(x^*)} \lambda_j \nabla g_j(x^*) \right]^T d = 0$$

A expressão anterior permite concluir que o vector entre parêntesis recto é nulo ou perpendicular a d , mas o próximo teorema permite concluir que o vector entre parêntesis recto tem que ser nulo. A demonstração deste teorema recorre ao teorema do hiperplano de separação ou ao lema de Farkas. O lema Farkas é derivado do teorema do hiperplano de separação. Por isso, na demonstração do teorema seguinte é feita opção pelo uso do teorema do hiperplano de separação.

Teorema 5: Sejam a função objectivo e as funções restricionais do problema de programação matemática (7) continuamente diferenciáveis no ponto x^* que satisfaz à condição de qualificação restricional. Então $\nabla f(x^*)^T d \geq 0$ para

$$d \in \{d \in R^n : \nabla g_j(x^*)^T d \leq 0 \text{ para } j \in A(x^*)\} \text{ sse } \exists \lambda_j \geq 0 \text{ tal que}$$

$$\nabla f(x^*) + \sum_{j \in A(x^*)} \lambda_j \nabla g_j(x^*) = 0.$$

Prova: Primeiro será feita a demonstração da necessidade da condição. Seja

$\nabla f(x^*)^T d \geq 0$ para $d \in \{d \in R^n : \nabla g_j(x^*)^T d \leq 0 \text{ para } j \in A(x^*)\}$ e seja o cone

$Y = \{y \in R^n : y = - \sum_{j \in A(x^*)} \lambda_j \nabla g_j(x^*) \text{ com } \lambda_j \geq 0\}$. $\nabla f(x^*) \in Y$ prova que

$\exists \lambda_j \geq 0$ para $j \in A(x^*)$ tal que $\nabla f(x^*) + \sum_{j \in A(x^*)} \lambda_j \nabla g_j(x^*) = 0$. Caso contrário,

$\nabla f(x^*) \notin Y$, então existe pelo teorema da separação um hiperplano de

separação entre $\nabla f(x^*)$ e o cone convexo Y , ou seja, $\exists d \in R^n$,

$\exists \alpha > 0 : \forall y \in Y$ com $y^T d < \alpha \leq \nabla f(x^*)^T d$. Como $0 \in Y$ é válido, então

$\nabla f(x^*)^T d \geq 0$. Como para $y \in Y$ se tem que $\beta y \in Y$ com $\beta \in R^+$, $\forall y \in Y$

$y^T d < 0$. Logo resulta que $\nabla g_j(x^*)^T d > 0$ para $j \in A(x^*)$, o que é uma

contradição. Portanto o $\nabla f(x^*) \in Y$ para se verificar a hipótese. Para a

demonstração da suficiência basta observar que se $\exists \lambda_j \geq 0$ tal que

$\nabla f(x^*) = - \sum_{j \in A(x^*)} \lambda_j \nabla g_j(x^*)$, então para $d \in \{d \in R^n : \nabla g_j(x^*)^T d \leq 0 \text{ com}$

$j \in A(x^*)\}$, $\nabla f(x^*)^T d = - \sum_{j \in A(x^*)} \lambda_j \nabla g_j(x^*)^T d \geq 0$.

Teorema de Karush Kuhn Tucker 6: Sejam a função objectivo e as funções restricionais do problema de programação matemática (7) continuamente diferenciáveis no ponto x^* que satisfaz à condição de qualificação restricional. Se x^* é uma solução optima para o problema, então:

$$1 - x^* \in G \quad (8)$$

$$2 - \exists \lambda_j \geq 0 \text{ tal que } \lambda_j \nabla g_j(x^*) = 0 \text{ para } j = 1, 2, \dots, m \quad (9)$$

$$3 - \nabla f(x^*) + \sum_j^m \lambda_j \nabla g_j(x^*) = 0 \quad (10)$$

Prova: A primeira condição é trivial. Conjuntamente, a segunda e terceira condição são equivalentes a afirmar a tese do teorema anterior.

Conjuntamente, estas três condições constituindo um sistema com equações e inequações são conhecidas por condições de Karush Kuhn Tucker, sendo condições necessárias para a identificação de um ponto óptimo, mas não são suficientes. Os pontos identificáveis pelas condições são ditos de pontos críticos ou de estacionaridade.

Definição: Função de Lagrange para o problema (4.1) é a função

$$L : R^n \times R_+^m \rightarrow R \text{ dada por } L(x, \lambda) = f(x) + \sum_j^m \lambda_j \nabla g_j(x).$$

A terceira condição de Karush Kuhn Tucker pode ser enunciada como $\nabla_x L(x^*, \lambda) = 0$ em que o gradiente da função de Lagrange é calculado

relativamente às variáveis das coordenadas do vector x e o Teorema de Karush Kuhn Tucker pode ser enunciado como

Teorema de Karush Kuhn Tucker 7 (aplicável a problemas só com restrições de desigualdade): Sejam a função objectivo e as funções restricionais do problema de programação matemática (7) continuamente diferenciáveis no ponto x^* que satisfaz à condição de qualificação restricional. Se x^* é uma solução optima para o problema, então:

$$1 - \nabla L(x^*, \lambda) \leq 0 \quad (11)$$

$$2 - \exists \lambda \in R_+^m \cup \{0\} \text{ tal que } \nabla_{\lambda} L(x^*, \lambda)^T \lambda = 0 \quad (12)$$

$$3 - \nabla_x L(x^*, \lambda)^T \lambda = 0 \quad (13)$$

Os números λ_j com $j = 1, 2, \dots, m$ mencionados no Teorema de Karush Kuhn

Tucker são conhecidos por:

1 - multiplicadores de Lagrange

2 - varáveis duais

3 - preços sombra

e são nulos ou positivos de acordo com o teorema de Karush Kuhn Tucker.

É útil para verificar as condições de Karush Kuhn Tucker enumerar as tentativas de solução com base no facto de que ou o multiplicador de Lagrange é nulo ou é positivo, caracterizando dois estados lógicos para os multiplicadores de

Lagrange: 0 se o multiplicador for nulo, 1 se o multiplicador for não nulo. O teorema de Karush Kuhn Tucker nem sempre é condição suficiente para identificar o ótimo.

O problema de programação matemática (4) pode conter restrições que sejam determinadas por igualdades, sendo conveniente particularizar o Teorema de Karush Kuhn Tucker para este caso. Considere o problema escrito como

$$\begin{aligned}
 & \min f(x) \\
 & \text{s.a} \\
 & g_i(x) \leq 0 \text{ para } i = 1, 2, \dots, p \\
 & g_i(x) = 0 \text{ para } i = p+1, p+2, \dots, m \qquad (14)
 \end{aligned}$$

O Teorema de Karush Kuhn Tucker 7 só pode ser aplicado caso o problema só tenha restrições de desigualdade, mas como uma restrição de igualdade pode ser decomposta em duas de desigualdade, o problema (14) pode ser escrito como

$$\begin{aligned}
 & \min f(x) \\
 & \text{s.a} \\
 & g_i(x) \leq 0 \text{ para } i = 1, 2, \dots, p \\
 & g_i^+(x) \leq 0 \text{ para } i = p+1, p+2, \dots, m \\
 & g_i^-(x) \leq 0 \text{ para } i = p+1, p+2, \dots, m \\
 & \text{sendo } g_i^+(x) = g_i(x) \text{ e } g_i^-(x) = -g_i(x) \qquad (15)
 \end{aligned}$$

Cada restrição de igualdade origina duas de desigualdade, podendo então o Teorema de Karush Kuhn Tucker 7 ser aplicado ao problema na formulação equivalente (14). No entanto, visto que existe interesse em considerar restrições de igualdade em diversos problemas, é mais prático obter uma versão do Teorema de Karush Kuhn Tucker considerando restrições de igualdade. O teorema seguinte pode ser então aplicado.

Teorema de Karush Kuhn Tucker 8 (considerando restrições de igualdade):
 Sejam a função objectivo e as funções restricionais do problema de programação matemática (14) continuamente diferenciáveis no ponto x^* que satisfaz à condição de qualificação restricional. Se x^* é uma solução óptima para o problema, então:

$$1 - g_j(x^*) \leq 0 \text{ para } j = 1, 2, \dots, p \text{ e } g_j(x^*) = 0 \text{ para}$$

$$j = p+1, p+2, \dots, m$$

$$2 - \exists \lambda_j \geq 0 \text{ tal que } \lambda_j \nabla g_j(x^*) = 0 \text{ para } j = 1, 2, \dots, p$$

$$3 - \nabla f(x^*) + \sum_j^m \lambda_j \nabla g_j(x^*) = 0$$

Teorema 9: Cada uma das seguintes condições são qualificações restricionais:

1 - $g_j(x) = a_j^t x - b_j$ com $a_j \in R^n \setminus \{0\}$ e $b_j \in R$ para $j = 1, 2, \dots, m$

2 - $g_j(x)$ é uma função convexa para $j = 1, 2, \dots, m$ e $\exists x' \in R^n$ tal

que $g_j(x') < 0$

3 - os vectores $y_j = \nabla g_j(x^*)^T$ com $j \in A(x^*)$ são linearmente

independentes

A primeira condição determina que as restrições sejam funções afins. A segunda é conhecida como condição de Slater. Estas duas condições asseguram condições de regularidade em todo o espaço, enquanto a terceira tem carácter local. Caso as funções restricionais sejam funções afins existe qualificação restricional.

AN1.2.4 Condições de suficiência

Os pontos críticos além de mínimos globais podem ser mínimos locais, máximos globais, máximos locais ou de sela, ou seja, o teorema de Karush Kuhn Tucker não é condição suficiente para a identificação do mínimo global. O mínimo global é o que corresponde ao ponto crítico com o menor valor para a função objectivo. Serão demonstrados resultados que asseguram condições de suficiência para identificar o ponto óptimo do problema (7), primeiramente por argumentação geométrica e em seguida pela teoria da dualidade.

AN1.2.4.1 Condições de suficiência pela argumentação geométrica

A suficiência das condições de Karush Kuhn Tucker pode ser assegurada por argumentação geométrica, ou seja, caso a função objectivo seja pseudoconvexa e o conjunto G dos vectores das decisões admissíveis seja convexo existe suficiência.

Teorema 10: Se a função objectivo do problema de programação matemática (7) for pseudoconvexa e o conjunto G for convexo, então as condições de Karush Kuhn Tucker são suficientes para a identificação do ponto x^* mínimo global do problema de programação matemática (7).

Prova: Sejam para o problema de programação matemática (7) a função objectivo $f : G \subseteq R^n \rightarrow R$ pseudoconvexa e G um convexo, se x^* satisfaz às condições de Karush Kuhn Tucker, então $\forall x \in G \quad \nabla f(x^*)^t (x - x^*) \geq 0$ o que pelo facto da função ser pseudoconvexa implica que $f(x_2) \geq f(x_1)$, ou seja, o ponto x^* é mínimo global.

A prova anterior permite concluir que o teorema é válido mesmo se o conjunto G não for convexo, mas desde que para o ponto x^* e para qualquer $x \in G$ os vectores $x - x^* \in \overline{D}(x^*)$, ou seja, basta que o conjunto G seja estrelado relativamente ao ponto x^* .

Ainda, convém reafirmar que estas condições são suficientes mas não necessárias, visto que, um ponto pode ser óptimo e não as verificar, bastando para isso que a função objectivo não seja pseudoconvexa.

AN1.2.4.2 Problema minimax. Teoria do ponto sela

A teoria da dualidade tem um conteúdo só por si bastante importante em programação matemática. A teoria da dualidade é baseada na teoria do problema minimax e na teoria do ponto de sela. Sejam f , h , e F funções reais definidas respectivamente nos conjuntos $X \subseteq R^n$, $Y \subseteq R^m$ e $X \times Y \subseteq R^n \times R^m$, admitindo que se verifica a desigualdade $\forall (x,y) \in X \times Y \quad f(x) \leq g(y)$ então também se verifica a desigualdade

$$\begin{array}{ll} \max f(x) \leq \min g(y) \\ \text{s.a} & \text{s.a} \\ x \in X & y \in Y \end{array}$$

A teoria da dualidade é o estudo das condições que se têm de verificar para que a desigualdade anterior seja uma igualdade.

Para a função F é válida a desigualdade

$$\max_y \min_x F(x,y) \leq \min_x \max_y F(x,y)$$

A teoria do ponto sela é o estudo das condições que se têm de verificar para que a desigualdade anterior seja uma igualdade. Na escrita anterior e seguinte é convencionado para tornar a escrita menos sobrecarregada que $x \in X$ e $y \in Y$.

Definição: Um ponto $(x^*, y^*) \in X \times Y$ é dito um ponto de sela da função F com respeito à maximização em y e à minimização em x se verifica a seguinte desigualdade

$$F(x^*, y) \leq F(x^*, y^*) \leq F(x, y^*) \quad \forall (x, y) \in X \times Y$$

Fica subentendido para encurtar a escrita que um ponto de sela da função F com respeito à maximização em y e à minimização em x será o objecto do tratamento deste secção e será dito só de ponto de sela no texto que se segue.

Sejam por definição consideradas as seguintes funções: $F^* : X \subseteq R^n \rightarrow R$ e

$F_* : Y \subseteq R^m \rightarrow R$ definidas respectivamente por

$$F^*(x) = \max_y F(x, y) \text{ e } F_*(y) = \min_x F(x, y)$$

Facilmente se concluiu que é válida a desigualdade dupla

$$\forall (x, y) \in X \times Y \quad F_*(y) = \min_x F(x, y) \leq F(x, y) \leq \max_y F(x, y) = F^*(x)$$

ou seja $\forall (x, y) \in X \times Y \quad F_*(y) \leq F^*(x)$

Teorema 11: As três afirmações seguintes são equivalentes:

$$1 - \max_y \min_x F(x, y) \leq \min_x \max_y F(x, y)$$

$$2 - \exists (x^*, y^*) \in X \times Y \quad \max_y F(x^*, y) = F(x^*, y^*) = \min_x F(x, y^*)$$

$$3 - \forall (x, y) \in X \times Y \quad F(x^*, y) \leq F(x^*, y^*) \leq F(x, y^*)$$

Prova: 1 implica em 2, para a função F tem-se sempre a desigualdade

$$F_*(y) = \min_x F(x, y) \leq F(x, y) \leq \max_y F(x, y) = F^*(x)$$

sejam $x^* \in X$ tal que $F^*(x) = \min_x F^*(x)$ e $y^* \in Y$ tal que $F_*(y^*) = \max_y F_*(y)$

então 1 implica $\exists (x^*, y^*) \in X \times Y$ tais que $F_*(y^*) = F^*(x^*)$, ou seja, implica em 2.

2 implica em 3, sendo válido $\max_y F(x^*, y) = F(x^*, y^*) = \min_x F(x, y^*)$. Basta notar

que $\forall (x, y) \in X \times Y \quad F(x^*, y) \leq \max_y F(x^*, y) = F(x^*, y^*) = \min_x F(x, y^*) \leq F(x, y^*)$.

3 implica em 1, sendo válido $\forall (x, y) \in X \times Y \quad F(x^*, y) \leq F(x^*, y^*) \leq F(x, y^*)$

e $F_*(y) = \min_x F(x, y) \leq \max_y \min_x F(x, y) \leq \min_x \max_y F(x, y) = \max_y F(x, y) = F^*(x)$

mas no ponto sela é válida a desigualdade $F^*(x^*) \leq F_*(y^*)$ logo é valido 1.

Este teorema apresenta condições alternativas para definir ponto sela de uma função.

AN1.2.4.3 Condições de suficiência pela teoria da dualidade

A suficiência das condições de Karush Kuhn Tucker pode ser assegurada com base na teoria da dualidade, ou seja, caso a função de Lagrange tenha ponto sela existe suficiência para a identificação do ótimo.

Definição: Função de Lagrange para o problema (7) é a função

$$L : R^n \times R_+^m \rightarrow R \text{ dada por } L(x, \lambda) = f(x) + \sum_j^m \lambda_j g_j(x).$$

Definição: Função primal para o problema (7) é a função $L^* : R^n \rightarrow R$ dada por

$$L^*(X) = \min_{\lambda \in R_+^m} L(x, \lambda).$$

Portanto a função primal será dada por

$$L^*(x) = \max_{\lambda \in R_+^m} f(x) + \sum_j^m \lambda_j g_j(x), \text{ que é igual a } f(x) \text{ se } g_j(x) \leq 0 \text{ e a } +\infty \text{ caso}$$

contrário.

Definição: Função dual para o problema (7) é a função $L_* : R_+^m \rightarrow R$ dada por

$$L_*(\lambda) = \min_{x \in R^n} L(x, \lambda).$$

Teorema 12: A função dual é uma função côncava.

Prova: Seja $0 \leq \alpha \leq 1$, então

$$L_*(\alpha \lambda^1 + (1-\alpha) \lambda^2) = \min_{x \in R^n} f(x) + (\alpha \lambda^1 + (1-\alpha) \lambda^2)^t g(x)$$

recordando que o mínimo de uma soma de funções nunca é inferior a soma dos mínimos das funções

$$L_*(\alpha \lambda^1 + (1-\alpha) \lambda^2) = \min_{x \in R^n} \alpha (f(x) + (\lambda^1)^t g(x)) + (1-\alpha) (f(x) + (\lambda^2)^t g(x)) \geq$$

$$\alpha \min_{x \in R^n} f(x) + (\lambda^1)^t g(x) + (1-\alpha) \min_{x \in R^n} f(x) + (\lambda^2)^t g(x) = \alpha L_*(\lambda^1) + (1-\alpha) L_*(\lambda^2)$$

O que mostra que a função dual é sempre uma função côncava. O vector das variáveis da função dual é chamada de vector das variáveis duais.

Definição: Problema primal para o problema (7) é o problema da minimização da função primal escrito como:

$$\min_x L^*(x) = \min_x \max_{\lambda \in R_+^m} f(x) + \sum_j^m \lambda_j g_j(x), \text{ que é igual a } f(x) \text{ se } g_j(x) \leq 0 \text{ e a } +\infty$$

caso contrário.

Caso haja solução para o problema original (7) o problema primal identifica-se com ele, razão pela qual também se chama ao problema (7) problema primal e ao vector das variáveis de decisão também se chama de vector das variáveis primais.

Definição: Problema dual para o problema (7) é o problema da maximização da

$$\text{função dual escrito como: } \max_{\lambda \in R_+^m} L_*(\lambda) = \max_{\lambda \in R_+^m} \min_x f(x) + \sum_j^m \lambda_j g_j(x).$$

Como a função dual é uma função côncava e o conjunto dos vectores admissíveis para o vector das variáveis duais é um conjunto convexo, o problema dual tem propriedades excelentes no que respeita à disponibilidade de algoritmos para a sua resolução. Caso com base na solução do problema dual seja possível obter a solução do problema primal, pode ser vantajoso recorrer à solução do problema dual.

Definição: Um ponto $(x', \lambda') \in R^n \times R_+^m$ é dito um ponto admissível para o problema dual se satisfaz à seguinte igualdade

$$L_*(\lambda') = L(x', \lambda')$$

Se λ' for um ponto óptimo para o problema dual, o par (x', λ') admissível para o problema dual é também dito de óptimo.

Definição: Salto de dualidade é diferença entre a solução do problema primal e a solução do problema dual.

Teorema 13 (dualidade fraca): Seja x^* um mínimo global do problema (7), então para $\lambda \in R_+^m$, $L_*(\lambda) = L_*(\lambda^*) \leq f(x)$, sendo λ^* o ponto correspondente ao máximo do problema dual.

Um ponto $(x^*, \lambda^*) \in R^n \times R_+^m$ é dito um ponto de sela da função $L : R^n \times R_+^m \rightarrow R$ com respeito à maximização em λ e à minimização em x se verifica a seguinte desigualdade

$$L(x^*, \lambda) \leq L(x^*, \lambda^*) \leq L(x, \lambda^*) \quad \forall (x, \lambda) \in R^n \times R_+^m$$

O Teorema (11), reescrito para a função de Lagrange é seguidamente apresentado

Teorema 14: As três afirmações seguintes são equivalentes:

$$1 - \max_{\lambda} \min_x L(x, \lambda) \leq \min_x \max_{\lambda} L(x, \lambda)$$

$$2 - \exists (x^*, \lambda^*) \in R^n \times R_+^m \quad \max_{\lambda} F(x^*, \lambda) = L(x^*, \lambda^*) = \min_x L(x, \lambda^*)$$

$$3 - \forall (x, \lambda) \in R^n \times R_+^m \quad L(x^*, \lambda) \leq L(x^*, \lambda^*) \leq L(x, \lambda^*)$$

Prova: 1 implica em 2, para a função F tem-se sempre a desigualdade

$$F^*(y) = \min_x F(x, y) \leq F(x, y) \leq \max_y F(x, y) = F^*(x)$$

1 corresponde a afirmar que o problema dual tem o mesmo valor que o problema primal. 2 corresponde a afirmar que existe um ponto de coordenadas $(x^*, \lambda^*) \in R^n \times R_+^m$ no qual a função primal tem valor igual ao valor da função dual. Finalmente, 3 corresponde a afirmar que esse ponto é um ponto de sela da função de Lagrange. Portanto, caso a função de Lagrange tenha um ponto de sela, ou seja existe igualdade dual, então a resolução do problema dual conduz ao mesmo resultado que a do problema primal. Esta é uma das vantagens da teoria da dualidade, ou seja, gerar um problema que se for equivalente pode permitir um código de computação mais eficiente para a resolução do problema

primal. Ainda, a função primal nunca toma valores inferiores aos da função dual, o salto de dualidade é sempre nunca negativo.

Caso não haja ponto de sela função de Lagrange o salto de dualidade é estritamente positivo, mas caso a solução do problema primal seja igual à do problema dual então existe igualdade dual e a primeira coordenada do ponto de sela é uma solução do problema primal (7). Convém então estabelecer condições para a existência de um ponto de sela da função de Lagrange que serão em consequência do Teorema (12) condições suficientes para a identificação de um ponto óptimo do problema (7).

Teorema 15: Um ponto $x^* \in R^n$ é um ponto óptimo para o problema primal se verifica alguma das condições seguintes:

$$1 - \exists \lambda^* \in R_+^m \quad L_*(\lambda^*) = f(x)$$

$$2 - x^* \in R^n \text{ é admissível para o problema (7), o par } (x^*, \lambda^*) \in R^n \times R_+^m \text{ é admissível para o problema dual e } L(x^*, \lambda^*) = f(x)$$

$$3 - x^* \in R^n \text{ é admissível para o problema (7), } \sum_j^m \lambda_j g_j(x^*) = 0 \text{ e}$$

$$L(x^*, \lambda) \leq \min_x L(x, \lambda)$$

$$4 - x^* \in R^n \text{ satisfaz as condições da Karush Kuhn Tucker para o problema (7) e sendo } \lambda^* \text{ o correspondente vector dos multiplicadores de Lagrange}$$

$$\nabla_x L(x^*, \lambda^*) = 0 \text{ implica } L(x^*, \lambda^*) = \min_x L(x, \lambda^*)$$

Prova: A condição 1 implica a existência de ponto de sela da função de Lagrange de acordo com o Teorema (12). A condição 2 é equivalente a 1.

A condição 3 implica que $f(x^*) = f(x^*) + \sum_j^m \lambda_j g_j(x^*) = L(x^*, \lambda) = \min_x L(x, \lambda)$.

Consequentemente, o ponto é admissível para o problema dual e portanto tem-se 2. A condição 4 implica que x^* é admissível para o problema (7) e

$\sum_j^m \lambda_j g_j(x^*) = 0$ com $\lambda_j \geq 0$ e a terceira condição de Karush Kuhn Tucker é

verificada $\nabla_x L(x^*, \lambda^*) = 0$ pelo que tem-se 3.

Das quatro condições do Teorema (13), a que é mais importante para condição de suficiência para o teorema Karush Kuhn Tucker é a quarta, visto que, da aplicação deste teorema resultam os pontos críticos e se satisfizerem a 4 são então ótimos para o problema (7), mas se não a satisfizerem nada poderá ser concluído sobre o ponto com recurso a esta condição. Também aqui, convém reafirmar que as condições do Teorema (13) são suficientes mas não necessárias, visto que um ponto pode ser ótimo e não as verificar.

Anexo II

Formulação do Problema

Resumo

É apresentada uma formulação de um problema de Afectação de Unidades em que se procura minimizar simultaneamente o custo do combustível e o custo associado às emissões poluentes, considerando um conjunto de restrições tipicamente referenciadas na literatura.

AN2. Formulação do Problema

AN2.1 Função objectivo

AN2.2 Restrições

AN2. Formulação do Problema

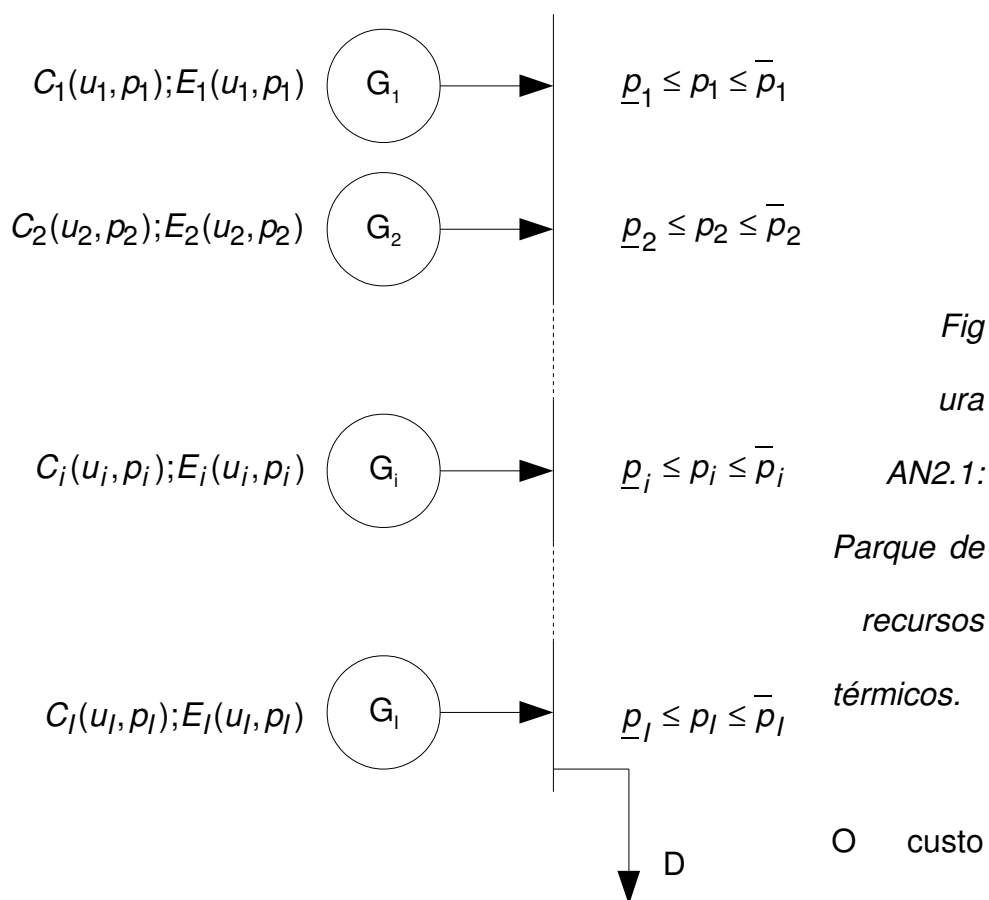
AN2.1 Função objectivo

O problema consiste na minimização do custo total de operação das centrais térmicas, escolhendo a melhor afectação de unidades, sendo entendido por unidade afecta a unidade colocada para operação, tendo como restrição na afectação as emissões poluentes para a atmosfera.

Na formulação do problema, I representa o conjunto unidades térmicas; H o conjunto de centrais onde estão instaladas as unidades térmicas; u_{it} o estado da unidade térmica i no instante t , em que 1 significa ligada e 0 significa desligada; NM o número máximo permitido de unidade térmicas ligadas em simultâneo; RP_h o número máximo permitido de unidade térmicas na Central h que podem ser ligadas (*crew constraint*); p_{it} a potência gerada pela unidade térmica i no instante t ; \underline{p}_i a potência mínima permitida da unidade térmica i ; \bar{p}_i a potência máxima permitida da unidade térmica i ; D_t a potência consumida no instante t ; R_t a reserva girante no instante t ; UR_i a taxa máxima de variação positiva de potência da unidade i ; DR_i a taxa máxima de variação negativa de potência da unidade i ; $umnup_i$ o tempo mínimo de funcionamento da unidade i ; $umndo_i$ o tempo mínimo de paragem da unidade i ; $utcool_i$ o tempo arrefecimento total da unidade i ; $uccool_i$ o custo de arranque da unidade

i , o qual depende do tempo de paragem da unidade, atingindo o máximo após $utcool_i$ e sendo constante após este instante; $C_i(u_i, p_i)$ o custo do combustível; a, b e c , coeficientes da função custo do combustível; $E_i(u_i, p_i)$ a massa de emissões poluentes; α, β e γ , coeficientes da função emissões poluentes; ω o custo unitário das emissões poluentes; λ o coeficiente de ponderação entre o custos do combustível e o custo das emissões

Considere-se a rede seguinte:



de funcionamento de uma unidade térmica, i , em €/h, é dado por

$$u_i \underline{p}_i \leq p_i \leq u_i \bar{p}_i \rightarrow C_i(u_i, p_i) + E_i(u_i, p_i)$$

$C_i(u_i, p_i) = (a_i + b_i p_i + \frac{1}{2} c_i p_i^2) u_i$ e apresenta a seguinte forma:

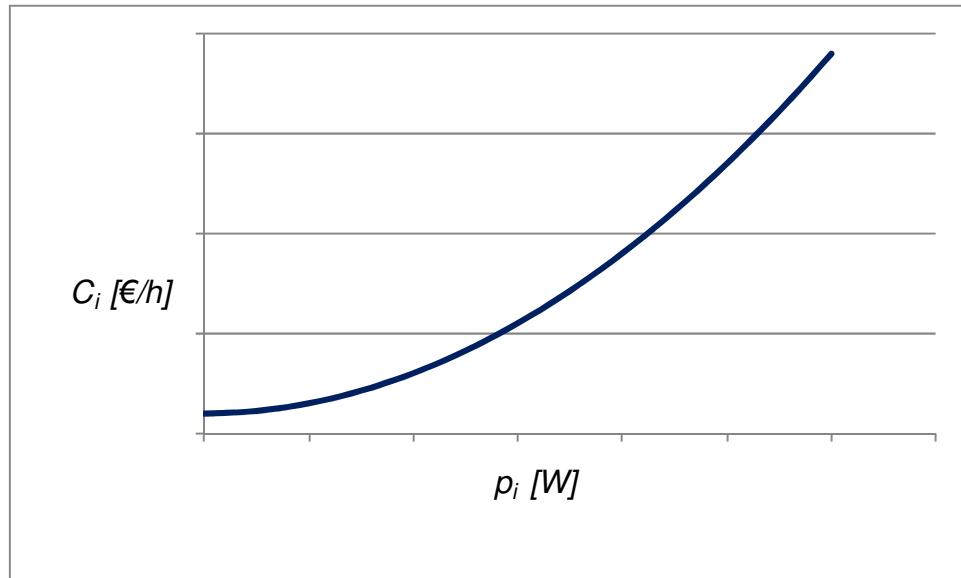


Figura AN2.2: Função custo do combustível em [€/h].

Se considerarmos o custo em função da energia produzida, então

$C_i(u_i, p_i) / p_i = (\frac{a_i}{p_i} + b_i + \frac{1}{2} c_i p_i) u_i$, e a forma da função será:

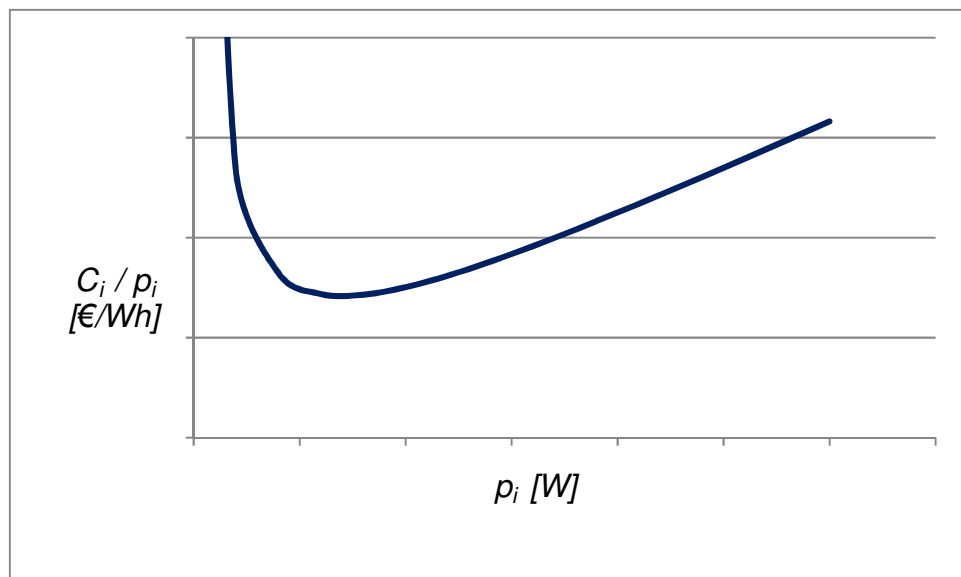


Figura AN2.3: Função custo específico do combustível em [€/Wh].

O valor das emissões poluentes é também uma função quadrática:

$$E_i(u_i, p_i) = (\alpha_i + \beta_i p_i + \frac{1}{2} \gamma_i p_i^2) u_i$$

A função objectivo é dada por

$$\min \left\{ \sum_{t=1}^T \sum_{i=1}^I C_{it}(u_{it}, p_{it}), \sum_{t=1}^T \sum_{i=1}^I \omega E_{it}(u_{it}, p_{it}) \right\}$$

AN2.2 Restrições

São consideradas as seguintes restrições:

1 - O máximo número permitido de unidades ligadas em simultâneo

$$\sum_{i \in I} u_{it} \leq NM$$

2 - O máximo número permitido de unidades que podem ser ligadas na mesma Central. Esta restrição era tipicamente devida às limitações de pessoal (*crew constraint*), perdendo actualmente relevância por esse motivo devido aos automatismos introduzidos na operação das Centrais

$$\sum_{i \in H \subset I} (1 - u_{it}) u_{i(t+1)} \leq RP_h$$

3 - A potência gerada em cada instante tem que ser igual à potência consumida, ou seja, é obrigatório satisfazer o consumo

$$\sum_{i \in I} p_{it} = D_t$$

4 - Garantida uma reserva de potência, reserva girante, seja para satisfazer consumos não previstos, seja para prever eventuais anomalias na produção ou distribuição de energia

$$\sum_{i \in J \subset I} (\bar{p}_i - \bar{p}_{it}) \geq R_t$$

5 - A taxa máxima de subida de potência em cada unidade térmica (*ramp rating*)

$$p_{it} - p_{it-1} \leq UR_i$$

6 - A taxa máxima de descida de potência em cada unidade térmica (*ramp rating*)

$$p_{it-1} - p_{it} \leq DR_i$$

7 - O tempo de mínimo de funcionamento de cada unidade, u_{minup_j} . É necessário para implementar esta restrição determinar o instante de arranque da unidade térmica

$$u_{it+1}(1 - u_{it}) = 1; \text{ se igual a } 1, \text{ arrancou na hora } t + 1$$

8 - O tempo mínimo de paragem de cada unidade, $umindo_j$. É necessário para implementar esta restrição determinar o instante de arranque da unidade térmica

$$u_{it}(1 - u_{it+1}) = 1; \text{ se igual a } 1, \text{ parou na hora } t + 1$$

9 - O tempo de arrefecimento total de cada unidade, $utcool_j$. O tempo de arrefecimento total indica o número de horas de arrefecimento a partir das quais o estado térmico da unidade é considerado nulo, i.e., a partir deste número de horas as perdas térmicas são nulas. O custo de arranque é tanto menor quanto mais cedo for dada esta ordem porque a energia dissipada é crescente com o tempo de acordo com a expressão

$$uccool(t) = uccool_0 \left(1 - e^{-\frac{5 \times t}{utcool}} \right)$$

Anexo III

Exemplos Programação Dinâmica

Resumo

Apresentam-se alguns exemplos de Programação Dinâmica, aplicada em funções objectivo simples tendo em consideração as restrições.

AN3. Exemplos Programação Dinâmica

Introdução

A programação dinâmica consiste em dividir um problema complexo em vários problemas mais pequenos e simples. Baseia-se no Princípio da Optimalidade introduzido por Richard Bellman em 1940: para que um problema possa ser resolvido por Programação Dinâmica, as suas características devem ser tais que em cada estágio da sua resolução, é conhecida toda a informação determinada nos estádios anteriores.

São resolvidos dois exemplos recorrendo a dois diferentes métodos, um gráfico e o segundo em forma de tabela.

Exemplo 1

Função Objectivo: mínimo de $(x_1 + x_2 + x_3^2 + x_4)$

Restrições:

$$x_1 = 2$$

$$x_2, x_3, x_4 \in \{1,2,3\}$$

$$x_2 + x_3 \leq 4$$

$$x_3 + x_4 \leq 4$$

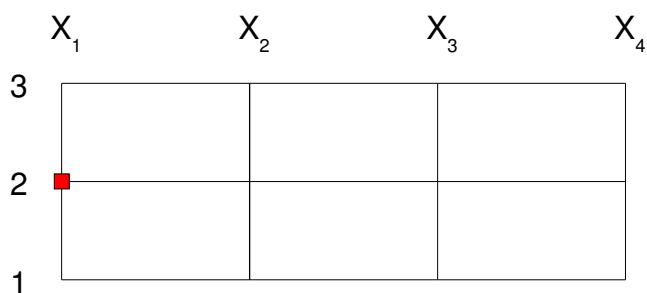
Resolução:

É atribuído em cada estado valores a uma variável, atribuição esta limitada pelas restrições. São registados os valores das variáveis e os valores que a função objectivo vai tomando.

Estado 1: atribuição de valor a x_1 .

O valor atribuído a x_1 está limitado a 2 devido às restrições.

Pelo método gráfico:



Pelo método tabelar, são listados os valores de x_1 , o valor da função objectivo neste estado e a verificação das restrições relevantes neste estado:

x_1	f.o.	$x_1=2$
3	3	inv
2	2	ok
1	1	inv

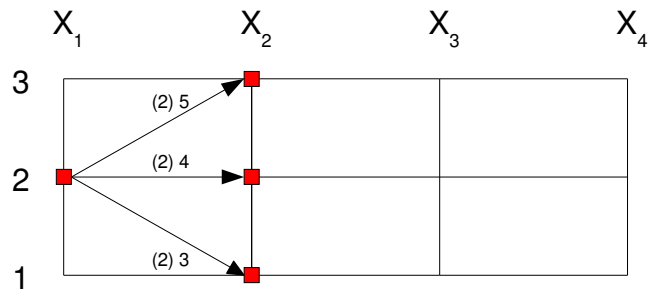
A restrição obriga a que à variável x_1 apenas possa ser atribuído o valor de 2. O valor da função objectivo até este momento é 2:

x_1	f.o.	$x_1=2$
2	2	ok

Estado 2: atribuição de valor a x_2

Neste estado a única restrição a considerar é $x_2, x_3, x_4 \in \{1,2,3\}$.

Pelo método gráfico:



São apresentados dois valores: O valor atribuído a x_1 entre parênteses e o valor da função objectivo neste estado, obtido pela soma de x_1 com x_2 .

Pelo método tabelar:

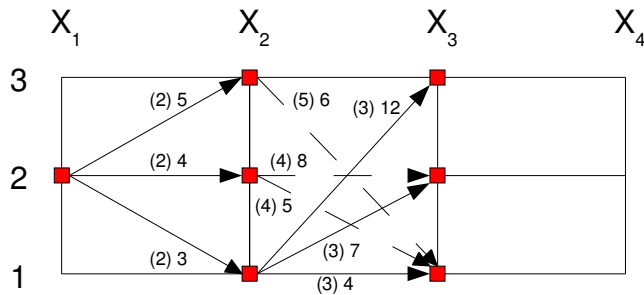
x_1	f.o.	x_2	f.o.
2	2	3	5
		2	4
		1	3

Estado 3: atribuição de valor a x_3

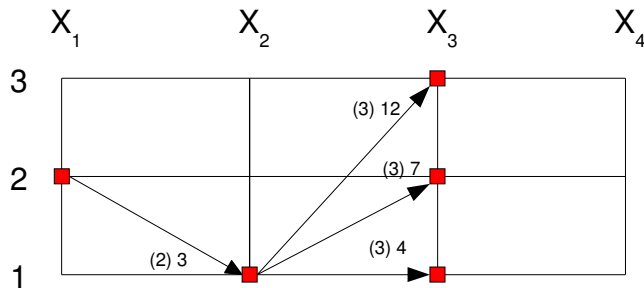
O conhecimento neste passo da informação dos estados anteriores permite calcular o valor da função para cada um dos valores atribuídos a x_3 .

Pelo método gráfico:

São pesquisados os caminhos que cumprem a restrição $x_2 + x_3 \leq 4$:



É novamente mostrado entre parênteses o valor da função objectivo no estado anterior e o valor que a função objectivo toma neste estado. As linhas a tracejado representam soluções não óptimas neste estado, sendo por isso descartadas:



Pelo método tabular:

x_1	f.o.	x_2	f.o.	x_3	f.o.	$x_2+x_3 \leq 4$
2	2	3	5	3	14	inv
				2	9	inv
				1	6	ok
		2	4	3	13	inv
				2	8	ok
				1	5	ok
		1	3	3	12	ok
				2	7	ok
				1	4	ok

Limitando as soluções àquelas que satisfazem as restrições:

x_1	f.o.	x_2	f.o.	x_3	f.o.	$x_2+x_3 \leq 4$
2	2	3	5	1	6	ok
		2	4	2	8	ok
				1	5	ok
		1	3	3	12	ok
				2	7	ok
				1	4	ok

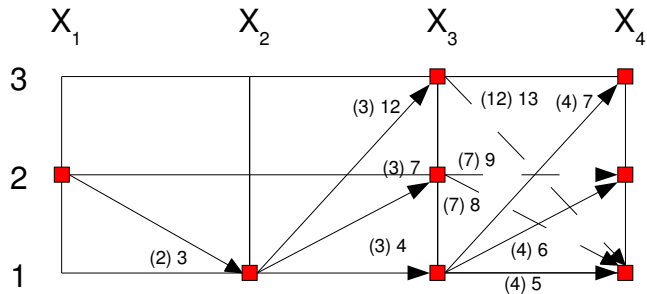
Descartam-se as soluções não ótimas, o que reduz as soluções possíveis até este estado a:

x_1	f.o.	x_2	f.o.	x_3	f.o.	$x_2+x_3 \leq 4$
2	2	1	3	3	12	ok
				2	7	ok
				1	4	ok

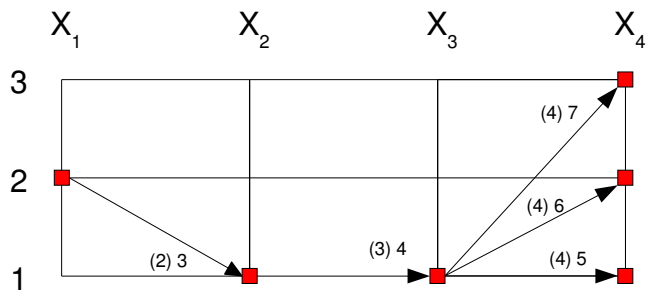
Passo 4: atribuição de valor a x_4

Segue-se um processo similar ao utilizado no passo anterior, desta vez atribuindo valores a x_4 .

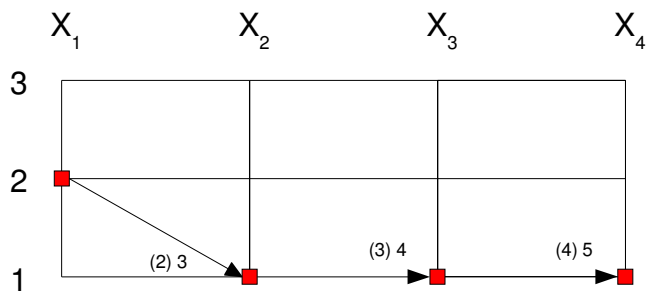
Pelo método gráfico:



Descartando as opções não óptimas:



Sendo o último estado, obtemos:



Pelo método tabelar:

x_1	f.o.	x_2	f.o.	x_3	f.o.	x_4	f.o.	$x_3+x_4 \leq 4$
2	2	1	3	3	12	3	15	inv
						2	14	inv
						1	13	ok
				2	7	3	10	inv
						2	9	ok
						1	8	ok
				1	4	3	7	ok
						2	6	ok
						1	5	ok

Limita-se os valores possíveis de x_4 àqueles que satisfazem a restrição a verificar neste passo, $x_3 + x_4 \leq 4$.

x_1	f.o.	x_2	f.o.	x_3	f.o.	x_4	f.o.	$x_3+x_4 \leq 4$
2	2	1	3	3	12	1	13	ok
						2	9	ok
				2	7	1	8	ok
						3	7	ok
				1	4	2	6	ok
						1	5	ok

Sendo o último estado, obtemos:

x_1	f.o.	x_2	f.o.	x_3	f.o.	x_4	f.o.	$x_3+x_4 \leq 4$
2	2	1	3	1	4	1	5	ok

Solução

O valor mínimo possível é $x_1 + x_2 + x_3^2 + x_4 = 5$, sendo esta solução da função objectivo é obtida com $x_1 = 2$, $x_2 = 1$, $x_3 = 1$ e $x_4 = 1$.

Exemplo 2

Função Objectivo: máximo de $(x_1 + x_2 + x_3^2 + x_4)$

Restrições:

$$x_1 = 2$$

$$x_2, x_3, x_4 \in \{1,2,3\}$$

$$x_2 + x_3 \leq 4$$

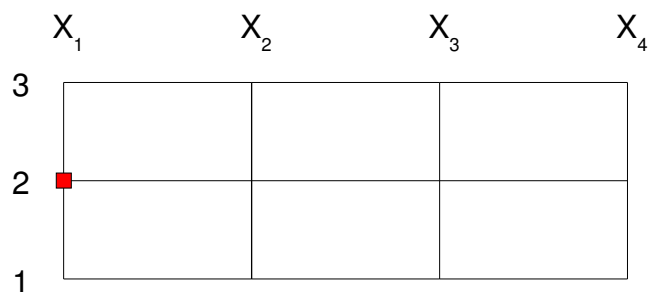
$$x_3 + x_4 \leq 4$$

Resolução:

Tal como no exemplo anterior, determina-se em cada estado o valor de cada variável, de forma a garantir simultaneamente o cumprimento das restrições e a satisfazer a função objectivo.

Estado 1: atribuição de valor a x_1

Pelo método gráfico:



Pelo método tabelar:

x_1	f.o.	$x_1=2$
3	3	inv
2	2	ok
1	1	inv

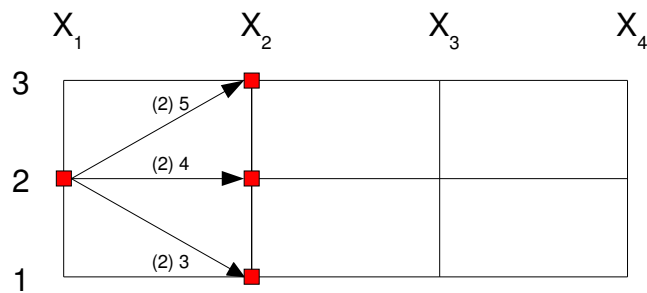
Devido às restrições, ficamos com..

x_1	f.o.	$x_1=2$
2	2	ok

Estado 2: atribuição de valor a x_2

São testadas todas as possibilidades de atribuição de valores à variável x_2 , verificando-se em cada caso o cumprimento das restrições. Por si só, a variável x_2 não está sujeita a qualquer restrição.

Pelo método gráfico:



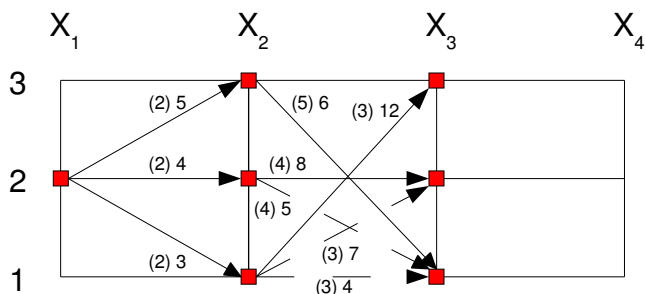
Pelo método tabelar:

x_1	f.o.	x_2	f.o.
2	2	3	5
		2	4
		1	3

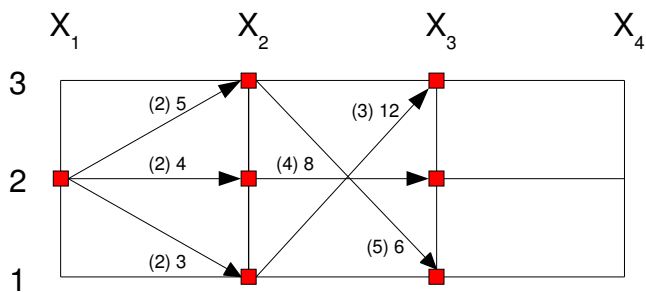
Estado 3: atribuição de valor a x_3

Tal como no estado anterior, são testadas todas as possibilidades de atribuição de valores à variável x_3 . É verificado em cada caso o cumprimento das restrições, neste caso $x_2 + x_3 \leq 4$.

Pelo método gráfico:



As linhas a tracejado representam soluções não ótimas neste estado, sendo por isso descartadas:



Pelo método tabelar:

x_1	f.o.	x_2	f.o.	x_3	f.o.	$x_2+x_3 \leq 4$		
2	2	3	5	3	14	inv		
				2	9	inv		
				1	6	ok		
		2	4	2	4	3	13	inv
						2	8	ok
						1	5	ok
		1	3	1	3	3	12	ok
						2	7	ok
						1	4	ok

São eliminadas as soluções que não cumprem as restrições:

x_1	f.o.	x_2	f.o.	x_3	f.o.	$x_2+x_3 \leq 4$
2	2	3	5	1	6	ok
				2	8	ok
		2	4	1	5	ok
				3	12	ok
		1	3	2	7	ok
				1	4	ok

São descartadas as soluções não óptimas até este estado:

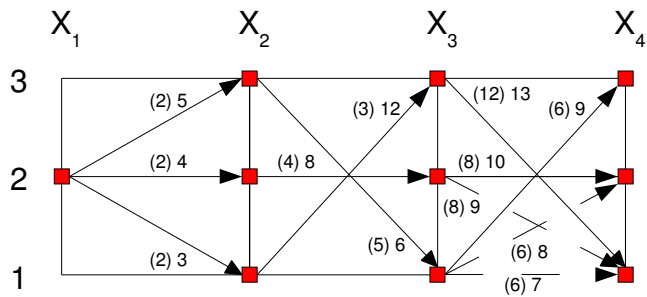
x_1	f.o.	x_2	f.o.	x_3	f.o.	$x_2+x_3 \leq 4$
2	2	3	5	1	6	ok
		2	4	2	8	ok
		1	3	3	12	ok

Estado 4: atribuição de valor a x_4

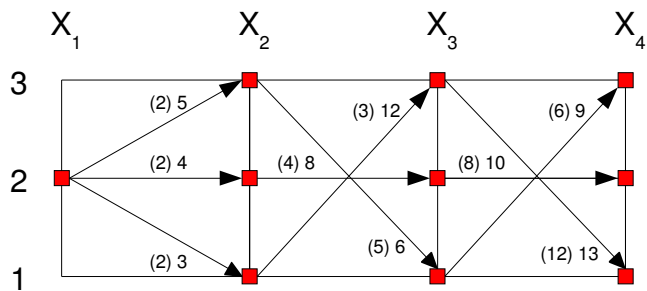
É determinado o valor da função objectivo resultante da atribuição de valores a

x_4 .

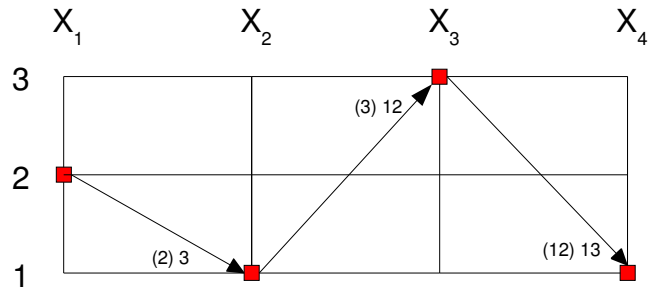
Pelo método gráfico:



Descartando as opções não óptimas:



Sendo o último estado, é seleccionada a solução óptima. Com em cada estado, é conhecida toda a informação determinada nos estados anteriores, é possível, desenhar o caminho óptimo:



Pelo método tabelar:

x_1	f.o.	x_2	f.o.	x_3	f.o.	x_4	f.o.	$x_3+x_4 \leq 4$
2	2	3	5	1	6	3	9	ok
						2	8	ok
						1	7	ok
		2	4	2	8	3	11	inv
						2	10	ok
						1	9	ok
		1	3	3	12	3	15	inv
						2	14	inv
						1	13	ok

A restrição $x_3 + x_4 \leq 4$, limita as opções possíveis às seguintes soluções:

x_1	f.o.	x_2	f.o.	x_3	f.o.	x_4	f.o.	$x_3+x_4 \leq 4$
2	2	3	5	1	6	3	9	ok
						2	8	ok
		2	4	2	8	2	10	ok
						1	9	ok
		1	3	3	12	1	13	ok

Sendo o último estado, obtemos:

x_1	f.o.	x_2	f.o.	x_3	f.o.	x_4	f.o.	$x_3+x_4 \leq 4$
2	2	1	3	3	12	1	13	ok

A solução óptima conduz a um valor da função objectivo de $x_1 + x_2 + x_3^2 + x_4 = 13$. A solução que garante o máximo para a função objectivo é $x_1 = 2$, $x_2 = 1$, $x_3 = 3$ e $x_4 = 1$.

Esta solução era expectável devido às limitações impostas a x_2 , x_3 e x_4 pelas restrições $x_2 + x_3 \leq 4$ e $x_3 + x_4 \leq 4$ conjugado com o facto de na equação a variável x_3 ser elevada ao quadrado.

Anexo IV

Listagem do programa desenvolvido para os casos de estudo 1 e 2

Resumo

Listagem do programa em VBA.

AN4. Listagem do programa desenvolvido para os casos de estudo 1 e 2

```
Dim Rest(11, 11), Ccost(11, 20, 170), Ecost(11, 20, 170), Tcost(11, 20, 170),  
Traj(11, 20, 170), pow(11, 20, 170), c_hora(11, 20, 170), ul(170) As Double
```

```
Rem Rest(n1,n2); n1->nº Gerador; n2->restrição  
Rem n2=1->Pmin; n2=2->Pmax; n2=3->Uptime; n2=4->downtime; n2=5->utcool;  
n2=6->ramp_down; n2=7->ramp_up; n2=8->reserva; n2=9->custo startup,  
multiplo tempo desligado; n2=10->estado em k=0;n2=11->P em k=0  
Rem Ccost(n1,n2,n3) e Traj(n1,n2,n3); n1->nº Gerador; n2->estado; n3->hora
```

```
Sub Sub_UC()  
Dim Ger, s_opt As Integer  
Dim pmx, pmn As Double  
Dim Energia As Double  
Fich = "P_Dinamica_v8.xls"  
Liv_Rest = "Restr"  
Liv_Estad = "Estados"
```

```
Rem Dynamic programming, k=1 to k=nhour, steam units with  
Rem minimum up time (uminup), minimum down time (umindo), and variable  
Rem startup costs
```

```
N_Ger = 1  
N_Rest = 11  
infini = 10000000000#  
nhours = 7 * 24
```

```
Rem Carrega Receita (preço energia)  
For i = 1 To 24 * 7  
    ul(i) = Workbooks(Fich).Worksheets(Liv_Rest).Cells(i + 28, 3)  
Next
```

```
Rem Carrega Restrições  
For i = 1 To N_Ger  
    For j = 1 To N_Rest  
        Rest(i, j) = Workbooks(Fich).Worksheets(Liv_Rest).Cells(j + 3, i + 3)
```



```

Next
Next

For Ger = 1 To N_Ger
    contador = 0
    For alfa = 0 To 1.02 Step 0.02
        Rem alfa = 0
        contador = contador + 1
        Rem Initialize all states at k=0 to a costs equal to infini
        Rem de -utcool a uminup
        k = 0
        For j = 1 To Rest(Ger, 3) + Rest(Ger, 5) + 1
            Tcost(Ger, j, k) = infini
            Ccost(Ger, j, k) = infini
            Ecost(Ger, j, k) = infini
            pow(Ger, j, k) = 0
        Next
        Rem Initialize initial state of unit to zero cost
        Tcost(Ger, Rest(Ger, 10) + Rest(Ger, 5) + 1, k) = 0
        Ccost(Ger, Rest(Ger, 10) + Rest(Ger, 5) + 1, k) = 0
        Ecost(Ger, Rest(Ger, 10) + Rest(Ger, 5) + 1, k) = 0
        pow(Ger, Rest(Ger, 10) + Rest(Ger, 5) + 1, k) = Rest(Ger, 11)
        Rem Dynamic programming step from k-1 to k hour
        For k = 1 To nhours
            Call dpav(Ger, k, ul(k), alfa)
        Next
        Rem Find optimal final state
        Call os(Ger, s_opt)
        Call caminho(Ger, s_opt, Energia)
        Call sol_final(Ger, s_opt, contador, alfa, Energia)
    Next
Next
End Sub

Sub dpav(ByVal g As Integer, ByVal hora As Integer, ByVal rect As Double,
ByVal af As Double)
    Dim Potn As Double
    Dim good As Double
    Dim cost As Double
    Dim Em_cost As Double
    Dim s, j As Integer

```

Rem sp=1->diminui; sp=2->aumenta

Rem ... does the dynamic programming step

Fich = "P_Dinamica_v8.xls"

Liv_Restr = "Restr"

Liv_Estad = "Estados"

utcool = Rest(g, 5)

uminup = Rest(g, 3)

umindo = Rest(g, 4)

utcool_ar = 1

uminup_ar = utcool + uminup + 1

umindo_ar = utcool - umindo + 1

desl = utcool

lig = utcool + 2

infini = 10000000000#

Rem s=utcool

s1 = utcool_ar

pow(g, s1, hora) = 0

If utcool = 1 Then

Rem só pode vir de uminup ou de utcool

Ccost_utcool = Ccost(g, s1, hora - 1)

Ecost_utcool = Ecost(g, s1, hora - 1)

Tcost_utcool = (1 - af) * Ccost_utcool + af * Ecost_utcool

Ccost_uminup = Ccost(g, uminup_ar, hora - 1)

Ecost_uminup = Ecost(g, uminup_ar, hora - 1)

Tcost_uminup = (1 - af) * Ccost_uminup + af * Ecost_uminup

If Tcost_uminup < Tcost_utcool Then

Ccost(g, s1, hora) = Ccost_uminup

Ecost(g, s1, hora) = Ecost_uminup

Tcost(g, s1, hora) = Tcost_uminup

Traj(g, s1, hora) = uminup_ar

Else

Ccost(g, s1, hora) = Ccost_utcool

Ecost(g, s1, hora) = Ecost_utcool

Tcost(g, s1, hora) = Tcost_utcool

```

    Traj(g, s1, hora) = s1
End If
Else
    Rem só pode vir de utcool+1 ou de utcool
    Ccost_utcool = Ccost(g, s1, hora - 1)
    Ecost_utcool = Ecost(g, s1, hora - 1)
    Tcost_utcool = (1 - af) * Ccost_utcool + af * Ecost_utcool
    Ccost_utcool1 = Ccost(g, s1 + 1, hora - 1)
    Ecost_utcool1 = Ecost(g, s1 + 1, hora - 1)
    Tcost_utcool1 = (1 - af) * Ccost_utcool1 + af * Ecost_utcool1
    If Tcost_utcool < Tcost_utcool1 Then
        Ccost(g, s1, hora) = Ccost_utcool
        Ecost(g, s1, hora) = Ecost_utcool
        Tcost(g, s1, hora) = Tcost_utcool
        Traj(g, s1, hora) = s1
    Else
        Ccost(g, s1, hora) = Ccost_utcool1
        Ecost(g, s1, hora) = Ecost_utcool1
        Tcost(g, s1, hora) = Tcost_utcool1
        Traj(g, s1, hora) = s1 + 1
    End If
End If

```

Rem s = De -utcool+1 a -2

Rem s=-2 caso particular

If utcool >= 2 Then

For s1 = desl - 1 To utcool_ar + 1 Step -1

Tcost(g, s1, hora) = Tcost(g, s1 + 1, hora - 1)

Ccost(g, s1, hora) = Ccost(g, s1 + 1, hora - 1)

Ecost(g, s1, hora) = Ecost(g, s1 + 1, hora - 1)

Traj(g, s1, hora) = s1 + 1

pow(g, s1, hora) = 0

Next

End If

Rem s = -1 => desligar, só pode vir de uminup se utcool>1 (utcool=1 já foi testado)

If utcool > 1 Then

Tcost(g, desl, hora) = Tcost(g, uminup_ar, hora - 1)

```

Ccost(g, desl, hora) = Ccost(g, uminup_ar, hora - 1)
Ecost(g, desl, hora) = Ecost(g, uminup_ar, hora - 1)
Traj(g, desl, hora) = uminup_ar
pow(g, desl, hora) = 0
End If

Rem s = 1=> ligar se uminup>1
amin = 10000000000#
i = 0
If uminup = 1 Then
  Rem pode vir de uminup ou de umindo a utcool
  Rem testa primeiro de umindo a utcool
  For j = umindo_ar To 1 Step -1
    c_arranque_1 = Rest(g, 9) * (1 - Exp(-(desl - j) / (utcool / 5)))
    Call Det_Pot(g, hora - 1, j, rect, af, Potn, cost, good, Em_cost)
    ccostp = c_arranque_1 + cost - good
    ecostp = Em_cost
    Tcost_d = (1 - af) * (Ccost(g, j, hora - 1) + ccostp) + af * (Ecost(g, j, hora - 1)
+ ecostp)
    If (Tcost_d < amin) Then
      amin = Tcost_d
      origem = j
      Ccost_d = ccostp
      Ecost_d = ecostp
      Potn_d = Potn
    End If
  Next
  Rem quando acaba, amin tem o menor custo de arranque em uminup=1
  Rem testa de uminup (já estava ligado)
  Call Det_Pot(g, hora - 1, lig, rect, af, Potn, cost, good, Em_cost)
  ccostp = cost - good
  ecostp = Em_cost
  Tcost_l = (1 - af) * (Ccost(g, lig, hora - 1) + ccostp) + af * (Ecost(g, lig, hora - 1)
+ ecostp)
  If Tcost_l < amin Then
    origem = lig
    amin = Tcost_l
    Ccost_l = ccostp
    Ecost_l = ecostp

```

```

    Potn_l = Potn
End If
Rem amin tem agora o menor valor para uma origem de utcool a ligado
If (amin < infini) Then
    If origem = lig Then
        Ccost(g, lig, hora) = Ccost(g, lig, hora - 1) + Ccost_l
        Ecost(g, lig, hora) = Ecost(g, lig, hora - 1) + Ecost_l
        pow(g, lig, hora) = Potn_l
    Else
        Ccost(g, lig, hora) = Ccost(g, lig, hora - 1) + Ccost_d
        Ecost(g, lig, hora) = Ecost(g, lig, hora - 1) + Ecost_d
        pow(g, lig, hora) = Potn_d
    End If
    Tcost(g, lig, hora) = (1 - af) * Ccost(g, lig, hora) + af * Ecost(g, lig, hora)
    Traj(g, lig, hora) = origem
Else
    Tcost(g, lig, hora) = infini
    Ccost(g, lig, hora) = infini
    Ecost(g, lig, hora) = infini
    Traj(g, lig, hora) = 1
End If
Else
    Rem uminup>1 (estava desligado)
    For j = umindo_ar To 1 Step -1
        c_arranque_1 = Rest(g, 9) * (1 - Exp(-(desl - j) / (utcool / 5)))
        Call Det_Pot(g, hora - 1, j, rect, af, Potn, cost, good, Em_cost)
        ccostp = c_arranque_1 + cost - good
        ecostp = Em_cost
        Tcost_d = (1 - af) * (Ccost(g, j, hora - 1) + ccostp) + af * (Ecost(g, j, hora - 1)
+ ecostp)
        If (Tcost_d < amin) Then
            amin = Tcost_d
            origem = j
            Potn_d = Potn
            Ccost_d = ccostp
            Ecost_d = ecostp
        End If
    Next
    If (amin < infini) Then
        Ccost(g, lig, hora) = Ccost(g, origem, hora - 1) + Ccost_d

```

```

    Ecost(g, lig, hora) = Ecost(g, origem, hora - 1) + Ecost_d
    Tcost(g, lig, hora) = (1 - af) * Ccost(g, lig, hora) + af * Ecost(g, lig, hora)
    pow(g, lig, hora) = Potn_d
    Traj(g, lig, hora) = origem
Else
    Tcost(g, lig, hora) = infini
    Ccost(g, lig, hora) = infini
    Ecost(g, lig, hora) = infini
    Traj(g, lig, hora) = 1
End If
End If
Rem s = 2 até uminup-1 (caso uminup=1 já testado)
If uminup > 1 Then
    For s1 = lig + 1 To uminup_ar - 1
        Call Det_Pot(g, hora - 1, s1 - 1, rect, af, Potn, cost, good, Em_cost)
        Ccost(g, s1, hora) = Ccost(g, s1 - 1, hora - 1) - good + cost
        Ecost(g, s1, hora) = Ecost(g, s1 - 1, hora - 1) + Em_cost
        Tcost(g, s1, hora) = (1 - af) * Ccost(g, s1, hora) + af * Ecost(g, s1, hora)
        Traj(g, s1, hora) = s1 - 1
        pow(g, s1, hora) = Potn
    Next
End If

Rem s = uminup (caso uminup=1 já testado)
Rem só pode vir de uminup ou de uminup-1
If uminup > 1 Then
    s1 = uminup_ar
    Rem de uminup-1
    Call Det_Pot(g, hora - 1, s1 - 1, rect, af, Potn, cost, good, Em_cost)
    Ccost_uminup1 = Ccost(g, s1 - 1, hora - 1) + cost - good
    Ecost_uminup1 = Ecost(g, s1 - 1, hora - 1) + Em_cost
    Tcost_uminup1 = (1 - af) * Ccost_uminup1 + af * Ecost_uminup1
    Potn_uminup1 = Potn
    Rem de uminup
    Call Det_Pot(g, hora - 1, s1, rect, af, Potn, cost, good, Em_cost)
    Ccost_uminup = Ccost(g, s1, hora - 1) + cost - good
    Ecost_uminup = Ecost(g, s1, hora - 1) + Em_cost
    Tcost_uminup = (1 - af) * Ccost_uminup + af * Ecost_uminup
    Potn_uminup = Potn
    If Tcost_uminup < Tcost_uminup1 Then

```

```

        Ccost(g, s1, hora) = Ccost_uminup
        Ecost(g, s1, hora) = Ecost_uminup
        Tcost(g, s1, hora) = Tcost_uminup
        pow(g, s1, hora) = Potn_uminup
        Traj(g, s1, hora) = s1
    Else
        Ccost(g, s1, hora) = Ccost_uminup1
        Ecost(g, s1, hora) = Ecost_uminup1
        Tcost(g, s1, hora) = Tcost_uminup1
        pow(g, s1, hora) = Potn_uminup1
        Traj(g, s1, hora) = s1 - 1
    End If
End If

End Sub
Sub os(ByVal g As Integer, ByRef s_min As Integer)

    Rem     ... finds the optimal final state ...

    Min = 10000000000#
    s_min = 1

    For j = 1 To Rest(g, 3) + Rest(g, 5) + 1
        If Tcost(g, j, 168) < Min Then
            s_min = j
            Min = Tcost(g, j, 168)
        End If
    Next
    If Min = 10000000000# Then s_min = 1

End Sub
Sub caminho(ByVal g As Integer, ByVal s As Integer, ByRef ETot As Double)
    Fich = "P_Dinamica_v8.xls"
    Livro = "Schedule"

    ETot = 0
    For hora = 168 To 1 Step -1
        If hora = 168 Then
            E_act = s

```

```

    custo = Ccost(g, E_act, hora)
    E_ant = Traj(g, E_act, hora)
Else
    E_act = E_ant
    custo = Ccost(g, E_act, hora)
    E_ant = Traj(g, E_act, hora)
End If
c_hr = Ccost(g, E_act, hora) - Ccost(g, E_ant, hora - 1) + pow(g, E_act, hora) *
ul(hora)
Workbooks(Fich).Worksheets(Livro).Cells(hora + 4, 3 * g) = E_act - Rest(g, 5)
- 1
Workbooks(Fich).Worksheets(Livro).Cells(hora + 4, 3 * g + 1) = pow(g, E_act,
hora)
Workbooks(Fich).Worksheets(Livro).Cells(hora + 4, 3 * g + 2) = custo
Workbooks(Fich).Worksheets(Livro).Cells(hora + 4, 4 * g + 14) = c_hr
ETot = pow(g, E_act, hora) + ETot
Rem k=0
Rem Workbooks(Fich).Worksheets(Livro).Cells(hora + 3, 3 * g) = Rest(g, 10)
Rem Workbooks(Fich).Worksheets(Livro).Cells(hora + 3, 3 * g + 1) = Rest(g,
11)
Next
Rem k=0
Workbooks(Fich).Worksheets(Livro).Cells(1 + 3, 3 * g) = Rest(g, 10)
Workbooks(Fich).Worksheets(Livro).Cells(1 + 3, 3 * g + 1) = Rest(g, 11)

End Sub
Sub Det_Pot(ByVal gn As Integer, ByVal hr As Integer, ByVal st As Integer,
ByVal Val_Eng As Double, ByVal alfa As Double, ByRef pot_det As Double,
ByRef custo As Double, ByRef proveito As Double, ByRef Emission_Cost As
Double)

Fich = "P_Dinamica_v8.xls"
Liv_Rest = "Restr"
Liv_Emiss = "Emiss"

Rem Carrega função custo combutivel
uccua = Workbooks(Fich).Worksheets(Liv_Rest).Cells(gn + 18, 3)
uccub = Workbooks(Fich).Worksheets(Liv_Rest).Cells(gn + 18, 4)
uccuc = Workbooks(Fich).Worksheets(Liv_Rest).Cells(gn + 18, 5)
Rem Carrega função emissões

```



```

ucalfa = Workbooks(Fich).Worksheets(Liv_Emiss).Cells(gn + 18, 3)
ucbeta = Workbooks(Fich).Worksheets(Liv_Emiss).Cells(gn + 18, 4)
ucgama = Workbooks(Fich).Worksheets(Liv_Emiss).Cells(gn + 18, 5)
Rem Carrega custo ton CO2
ucomega = Workbooks(Fich).Worksheets(Liv_Emiss).Cells(13, 3)

pot_det = (uccub - alfa * uccub + alfa * ucbeta - Val_Eng) / (uccuc * alfa - uccuc
- alfa * ucgama)
pant = pow(gn, st, hr)
If pant = 0 Then
    pot_det = Rest(gn, 1)
Else
    If pot_det > pant + Rest(gn, 7) Then pot_det = pant + Rest(gn, 7)
    If pot_det > Rest(gn, 2) Then pot_det = Rest(gn, 2)
    If pot_det < pant - Rest(gn, 6) Then pot_det = pant - Rest(gn, 6)
    If pot_det < Rest(gn, 1) Then pot_det = Rest(gn, 1)
End If

custo = uccua + uccub * pot_det + 0.5 * uccuc * pot_det ^ 2
proveito = pot_det * Val_Eng
Emission_Cost = (ucalfa + ucbeta * pot_det + 0.5 * ucgama * pot_det ^ 2) *
ucomega

End Sub
Sub sol_final(ByVal g As Integer, ByVal s As Integer, ByVal cont As Integer,
ByVal af As Double, ByVal enrg As Double)

Fich = "P_Dinamica_v8.xls"
Liv = "E1"

Workbooks(Fich).Worksheets(Liv).Cells(cont + 2, g * 7) = cont
Workbooks(Fich).Worksheets(Liv).Cells(cont + 2, g * 7 + 1) = af
Workbooks(Fich).Worksheets(Liv).Cells(cont + 2, g * 7 + 2) = Tcost(g, s, 168)
Workbooks(Fich).Worksheets(Liv).Cells(cont + 2, g * 7 + 3) = Ccost(g, s, 168)
Workbooks(Fich).Worksheets(Liv).Cells(cont + 2, g * 7 + 4) = Ecost(g, s, 168)
Workbooks(Fich).Worksheets(Liv).Cells(cont + 2, g * 7 + 5) = enrg
Workbooks(Fich).Worksheets(Liv).Cells(cont + 2, g * 7 + 6) = s
End Sub

```

Anexo V

Listagem do programa desenvolvido para os caso de estudo 3

Resumo

Listagem do programa em VBA.

AN5. Listagem do programa desenvolvido para os casos de estudo 3

```
Dim Rest(11, 11), Ccost(11, 5000, 170), Ecost(11, 5000, 170), Tcost(5000, 170),  
Traj(11, 5000, 170), Traj_s(5000, 170), pow(11, 5000, 170), c_hora(11, 20, 170),  
ul(170), Rst_k(11, 20, 4) As Double, perc(11, 20, 21) As Integer, comb(5000, 11)  
As Integer, perc_sg(5000, 5000) As Integer  
Rem Rest(n1,n2); n1->nº Gerador; n2->restrição  
Rem n2=1->Pmin; n2=2->Pmax; n2=3->Uptime; n2=4->downtime; n2=5->utcool;  
n2=6->ramp_down; n2=7->ramp_up; n2=8->reserva; n2=9->custo startup,  
multiplo tempo desligado; n2=10->estado em k=0;n2=11->P em k=0  
Rem Ccost(n1,n2,n3) e Traj(n1,n2,n3); n1->nº Gerador; n2->estado; n3->hora  
Rem Rst_k(n1,n2,n3);n1->nº Gerador; n2->estado; n3=1 -> pmax; n3=2 -> pmin;  
n3=3 -> custo_arranque  
Rem Perc(n1,n2,n3); n1->nº Gerador; n2 -> estado final; n3>1 -> nº estado inicial  
+1; -> n3 =1 -> numero estados iniciais  
Rem Comb(n1,n2); n1->nº combinação; n2 -> nº Gerador  
Rem Perc_sg(n1,n2); n1->estado final sistema; n2>1 -> nº estado inicial +1; ->  
n2 =1 -> numero estados iniciais  
Sub Sub_UC()  
Dim ul(200, 2) As Double  
Dim ger, s_opt As Integer  
Dim num_comb As Integer  
Dim pmx, pmn As Double  
Dim Energia As Double  
  
Fich = "P_Dinamica_solver_v2.xls"  
Liv_Rest = "Restr"  
Liv_Emiss = "Emiss"  
Liv_Estad = "Estados"  
Liv_Rst_Slv = "Rest_Solver"  
Liv_Tst = "ME_Tabela"  
  
Rem Dynamic programming, k=1 to k=nhour, steam units with  
Rem minimum up time (uminup), minimum down time (umindo), and variable  
Rem startup costs  
  
n_ger = 3  
N_Rest = 11
```

```
infini = 10000000000#  
nhours = 24 * 7
```

```
Rem Carrega Receita (preço energia) e potencia contratada
```

```
For i = 1 To nhours  
    Rem preço energia  
    ul(i, 1) = Workbooks(Fich).Worksheets(Liv_Rest).Cells(i + 28, 3)  
    Rem potência contratada  
    ul(i, 2) = Workbooks(Fich).Worksheets(Liv_Rest).Cells(i + 28, 6)  
Next
```

```
Rem Carrega Restrições
```

```
For i = 1 To n_ger  
    For j = 1 To N_Rest  
        Rest(i, j) = Workbooks(Fich).Worksheets(Liv_Rest).Cells(j + 3, i + 3)  
    Next  
Next
```

```
For i = 1 To n_ger  
    For coef = 1 To 3  
        Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(i + 4, coef + 7) =  
Workbooks(Fich).Worksheets(Liv_Rest).Cells(i + 18, coef + 2)  
        Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(i + 4, coef + 11) =  
Workbooks(Fich).Worksheets(Liv_Emiss).Cells(i + 18, coef + 2)  
    Next  
Next
```

```
Rem determina estados do sistema de geradores
```

```
Call comb_st(n_ger, num_comb)  
Rem determina estados anteriores de cada gerador  
Call orig_gi(n_ger)  
Rem determina estados anteriores de cada estado  
Call orig_sg(n_ger, num_comb)
```

```
Rem inicializa estados em k=0
```

```
k = 0  
For i = 1 To num_comb  
    Tcost(i, k) = infini  
    Rem se estado inicial custo nulo  
    cont = 0
```

```

For ger = 1 To n_ger
    pow(ger, i, 0) = 0
    If Rest(ger, 10) < 0 Then si_g = Rest(ger, 10) + Rest(ger, 5) + 1
    If Rest(ger, 10) > 0 Then si_g = Rest(ger, 10) + Rest(ger, 5)
    si_s = comb(i, ger)
    If si_s = si_g Then
        cont = cont + 1
        pow(ger, i, 0) = Rest(ger, 11)
    End If
Next
If cont = n_ger Then Tcost(i, k) = 0
Next
Rem fim inicializa estados em k=0

contador = 0
For lambda = 0 To 1.2 Step 0.2
    contador = contador + 1
    Rem    Dynamic programming step from k-1 to k hour
    For k = 1 To nhours
        Call dpav(n_ger, k, ul(k, 1), ul(k, 2), lambda, num_comb)
    Next
    Rem    Find optimal final state
    Call os(num_comb, s_opt, nhours)
    Call caminho(n_ger, s_opt, nhours, contador, lambda)
    Rem Call sol_final(n_ger, s_opt, contador, lambda, Energia, nhours)
Next

End Sub
Sub dpav(ByVal ng As Integer, ByVal hora As Integer, ByVal rect As Double,
ByVal pot_ctr As Double, ByVal lmbd As Double, ByVal n_comb As Integer)
Dim cst_g(20, 5) As Double
Rem    cst_g(n1,n2);    n1->nº    Gerador;    n2=1->pmin;n2=2->pmax;n2=3-
->custo_arranque
Dim utcool(20), uminup(20), umindo(20), desl(20), lig(20), uminup_ar(20),
utcool_ar(20), st_f(20) As Integer
Dim n_si(20) As Integer
Dim cst As Double

Fich = "P_Dinamica_solver_v2.xls"

```

```
Liv_Rst_Slv = "Rest_Solver"  
Rem Liv_Rest = "Restr"  
Liv_sol = "Sol_Solver"
```

```
Rem Perc(n1,n2,n3); n1->nº Gerador; n2 -> estado final; n3>1 -> nº estado inicial  
+1; -> n3 =1 -> numero estados inicial
```

```
Rem Rest(n1,n2); n1->nº Gerador; n2->restrição
```

```
Rem n2=1->Pmin; n2=2->Pmax; n2=3->Uptime; n2=4->downtime; n2=5->utcool;  
n2=6->ramp_down; n2=7->ramp_up; n2=8->reserva; n2=9->custo startup,  
multiplo tempo desligado; n2=10->estado em k=0;n2=11->P em k=0
```

```
Rem Perc_sg(n1,n2); n1->estado final sistema; n2>1 -> nº estado inicial +1; ->  
n2 =1 -> numero estados iniciais
```

```
sol = 0
```

```
Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(28, 5) = rect
```

```
Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(26, 5) = lmbd
```

```
Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(25, 5) = pot_ctr
```

```
For i = 1 To n_comb
```

```
    infini = 10000000000#
```

```
    Tcost(i, hora) = infini
```

```
    For j = 1 To perc_sg(i, 1)
```

```
        comb_i = perc_sg(i, j + 1)
```

```
        cst = infini
```

```
        If hora = 18 Then
```

```
            a = 5
```

```
        End If
```

```
        If Tcost(comb_i, hora - 1) < infini Then
```

```
            Call rest_solver(ng, i, comb_i, hora)
```

```
            Call rslv_solver(ng, cst)
```

```
        End If
```

```
        If cst < infini Then
```

```
            sol = sol + 1
```

```
            infini = cst
```

```
            Traj_s(i, hora) = comb_i
```

```
            Tcost(i, hora) = cst + Tcost(comb_i, hora - 1)
```

```
            For g = 1 To ng
```

```
                pow(g, i, hora) = Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(g +
```

```
4, 6)
```

```
            Next
```

```

    End If
Next
Rem Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(30, 9) = i
Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(30, 10) = lmbd
Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(30, 11) = hora
Next
Workbooks(Fich).Worksheets(Liv_sol).Cells(hora + 4, 12) = sol
If sol = 0 Then
    a = 5
End If

End Sub

Sub rest_solver(ByVal ng As Integer, ByVal comb_f As Integer, ByVal comb_i As
Integer, ByVal hr As Integer)
Fich = "P_Dinamica_solver_v2.xls"
Liv_Rst_Slv = "Rest_Solver"

For g = 1 To ng
    utcool_ar = 1
    utcool = Rest(g, 5)
    uminup = Rest(g, 3)
    umindo = Rest(g, 4)
    desl = utcool
    lig = utcool + 1
    uminup_ar = utcool + uminup
    umindo_ar = utcool - umindo + 1

    si = comb(comb_i, g)
    sf = comb(comb_f, g)
    c_arranque = 0
    If sf <= desl Then
        pmax = 0
        pmin = 0
        c_arranque = 0
    End If
    If sf = lig Then
        pmax = Rest(g, 1)
        pmin = Rest(g, 1)
        c_arranque = Rest(g, 9) * (1 - Exp(-(desl - si + 1) / (utcool / 5)))
    End If

```

```

If sf > lig Then
    pant = pow(g, comb_i, hr - 1)
    pmax = pant + Rest(g, 7)
    If pmax > Rest(g, 2) Then pmax = Rest(g, 2)
    pmin = pant - Rest(g, 6)
    If pmin < Rest(g, 1) Then pmin = Rest(g, 1)
    c_arranque = 0
End If
Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(g + 4, 3) = pmin
Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(g + 4, 4) = pmax
Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(g + 4, 5) = c_arranque
Next

End Sub
Sub rslv_solver(ByVal ng As Integer, ByRef custo As Double)

Fich = "P_Dinamica_solver_v2.xls"
Liv_Rst_Slv = "Rest_Solver"
Worksheets(Liv_Rst_Slv).Activate
infini = 10000000000#

P_disp_max = Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(35, 4)
P_disp_min = Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(35, 3)
pot_ctr = Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(25, 5)
If P_disp_max >= pot_ctr And P_disp_min <= pot_ctr Then
    SolverReset
    SolverAdd CellRef:="$F$5:$F$7", Relation:=1, FormulaText:="$D$5:$D$7"
    SolverAdd CellRef:="$F$5:$F$7", Relation:=3, FormulaText:="$C$5:$C$7"
    SolverAdd CellRef:="$F$25", Relation:=3, FormulaText:="$E$25"
    SolverOptions      MaxTime:=0,      Iterations:=20,      Precision:=0.01,
Convergence:=0.001 _
    , StepThru:=False, Scaling:=True, AssumeNonNeg:=True, Derivatives:=1
    Rem      SolverOptions      PopulationSize:=100,      RandomSeed:=0,
MutationRate:=0.075, Multistart _
    :=False, RequireBounds:=True, MaxSubproblems:=0, MaxIntegerSols:=0, _
    IntTolerance:=0.1, SolveWithout:=False, MaxTimeNoImp:=30
    Rem      SolverOk      SetCell:="$E$30",      MaxMinVal:=2,      ValueOf:=0,
ByChange:="$F$5:$F$7", Engine:=1, EngineDesc:="GRG Nonlinear"
    SolverOk      SetCell:="$E$30",      MaxMinVal:=2,      ValueOf:=0,
ByChange:="$F$5:$F$7"

```



```

Res_Solver = SolverSolve(True)
custo = Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(30, 5)
If Res_Solver <> 0 And Res_Solver <> 1 Then
    Rem pr_energia = Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(28, 5)
    Rem pot_ctr = Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(25, 5)
    Rem Ptot = Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(25, 6)
    custo = infini
End If
Else
    Rem For g = 1 To ng
        Rem Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(g + 4, 6) =
Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(g + 4, 4)
    Rem Next
    Rem pr_energia = Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(28, 5)
    Rem pot_ctr = Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(25, 5)
    Rem custo = Workbooks(Fich).Worksheets(Liv_Rst_Slv).Cells(30, 5)
    custo = infini
End If

End Sub
Sub comb_st(ByVal ng As Integer, ByRef n_comb As Integer)

Dim st(10) As Integer
Dim sc(10) As Integer
Dim utcool(20), uminup(20), uminup_ar(20), umindo(20), umindo_ar(20),
desl(20), lig(20), utcool_ar(20) As Integer

For g = 1 To ng
    utcool_ar(g) = 1
    utcool(g) = Rest(g, 5)
    uminup(g) = Rest(g, 3)
    umindo(g) = Rest(g, 4)
    desl(g) = utcool(g)
    lig(g) = utcool(g) + 1
    uminup_ar(g) = utcool(g) + uminup(g)
    umindo_ar(g) = utcool(g) - umindo(g) + 1
Next

For g = 1 To ng
    st(g) = uminup_ar(g)

```

```

    sc(g) = 1
Next

i = 1
g = 0
While sc(1) <= st(1)
    g = g + 1
    comb(i, g) = sc(g)
    If g = ng Then
        i = i + 1
        sc(g) = sc(g) + 1
        g1 = ng
        While sc(g1) > st(g1) And g1 > 1
            sc(g1) = 1
            g1 = g1 - 1
            sc(g1) = sc(g1) + 1
        Wend
        g = 0
    End If
Wend
n_comb = i - 1

End Sub
Sub orig_gi(ByVal ng As Integer)
Dim s, j As Integer
Rem Perc(n1,n2,n3); n1->nº Gerador; n2 -> estado final; n3>1 -> nº estado inicial
+1; -> n3 =1 -> numero estados iniciais

For g = 1 To ng
    utcool = Rest(g, 5)
    uminup = Rest(g, 3)
    umindo = Rest(g, 4)
    utcool_ar = 1
    uminup_ar = utcool + uminup
    umindo_ar = utcool - umindo + 1
    desl = utcool
    lig = utcool + 1
    infini = 10000000000#
    For s1 = utcool_ar To uminup_ar
        If s1 = utcool_ar Then

```

```

n_sti = 1
Rem s=utcool
pow(g, s1, hora) = 0
If utcool = 1 Then
    Rem só pode vir de uminup ou de utcool
    n_sti = n_sti + 1
    perc(g, s1, n_sti) = s1
    perc(g, s1, 1) = n_sti - 1
    n_sti = n_sti + 1
    perc(g, s1, n_sti) = uminup_ar
    perc(g, s1, 1) = n_sti - 1
Else
    Rem só pode vir de utcool+1 ou de utcool
    n_sti = n_sti + 1
    perc(g, s1, n_sti) = s1
    perc(g, s1, 1) = n_sti - 1
    n_sti = n_sti + 1
    perc(g, s1, n_sti) = s1 + 1
    perc(g, s1, 1) = n_sti - 1
End If
End If
Rem s = De -utcool+1 a -2
Rem s=-2 caso particular
If s1 <= desl - 1 And s1 >= utcool_ar + 1 Then
    n_sti = 1
    If utcool >= 2 Then
        n_sti = n_sti + 1
        perc(g, s1, n_sti) = s1 + 1
        perc(g, s1, 1) = n_sti - 1
    End If
End If
Rem s = -1 => desligar, só pode vir de uminup se utcool>1 (utcool=1 já foi
testado)
If s1 = desl Then
    n_sti = 1
    If utcool > 1 Then
        n_sti = n_sti + 1
        perc(g, s1, n_sti) = uminup_ar
        perc(g, s1, 1) = n_sti - 1
    End If

```

```

End If
Rem s = 1 => ligar se uminup>1
If s1 = lig Then
  n_sti = 1
  Rem pode vir de uminup (se uminup=1) ou de umindo a utcool
  Rem testa primeiro de umindo a utcool
  For j = umindo_ar To 1 Step -1
    n_sti = n_sti + 1
    perc(g, s1, n_sti) = j
    perc(g, s1, 1) = n_sti - 1
    c_arranque_1 = Rest(g, 9) * (1 - Exp(-(desl - j) / (utcool / 5)))
  Next
  Rem testa de uminup (já estava ligado, uminup=1)
  If uminup = 1 Then
    n_sti = n_sti + 1
    perc(g, s1, n_sti) = lig
    perc(g, s1, 1) = n_sti - 1
  End If
End If
End If
Rem s = 2 até uminup-1
If s1 >= lig + 1 And s1 <= uminup_ar - 1 Then
  n_sti = 1
  Rem caso uminup=1 já testado
  If uminup > 1 Then
    n_sti = n_sti + 1
    perc(g, s1, n_sti) = s1 - 1
    perc(g, s1, 1) = n_sti - 1
  End If
End If
End If
Rem s = uminup (caso uminup=1 já testado)
If s1 = uminup_ar Then
  n_sti = 1
  Rem caso uminup=1 já testado
  If uminup > 1 Then
    Rem de uminup-1
    n_sti = n_sti + 1
    perc(g, s1, n_sti) = s1 - 1
    perc(g, s1, 1) = n_sti - 1
    Rem de uminup
    n_sti = n_sti + 1
  End If
End If

```

```

        perc(g, s1, n_sti) = s1
        perc(g, s1, 1) = n_sti - 1
    End If
End If
Next
Next

End Sub
Sub orig_sg(ByVal ng As Integer, ByVal n_comb As Integer)
Dim comb_or(5000, 5000) As Integer
Dim encontra(20, 5000) As Integer
Rem comb_or(n1,n2) -> n1-> estado final; n2 origem estado inicial, n2=1->
origem possivel; n2=0 -> origem não possivel
Fich = "P_Dinamica_solver_v2.xls"
Liv_Tst = "ME_Tabela"

For i = 1 To n_comb
    For j = 1 To n_comb
        comb_or(i, j) = 1
    Next
Next
For i = 1 To n_comb
    For g = 1 To ng
        encontra(g, i) = 0
    Next
Next

For i = 1 To n_comb
    For g = 1 To ng
        sf = comb(i, g)
        For j = 1 To perc(g, sf, 1)
            si = perc(g, sf, j + 1)
            For ns = 1 To n_comb
                If si = comb(ns, g) Then encontra(g, ns) = 1
            Next
        Next
    Next
Next
For g = 1 To ng
    For ns = 1 To n_comb

```

```

        If encontra(g, ns) = 0 Then comb_or(i, ns) = 0
        encontra(g, ns) = 0
    Next
Next
Next

For i = 1 To n_comb
    contador = 0
    For j = 1 To n_comb
        If comb_or(i, j) = 1 Then
            contador = contador + 1
            perc_sg(i, contador + 1) = j
        End If
    Next
    perc_sg(i, 1) = contador
Next

Fich = "P_Dinamica_solver_v2.xls"
Liv = "ME_Tabela"
For i = 1 To n_comb
    Workbooks(Fich).Worksheets(Liv).Cells(i + 4, 7) = i
    For g = 1 To ng
        Workbooks(Fich).Worksheets(Liv).Cells(i + 4, 7 + g) = comb(i, g)
    Next
    Workbooks(Fich).Worksheets(Liv).Cells(i + 4, ng + 8) = perc_sg(i, 1)
    For j = 1 To perc_sg(i, 1)
        Workbooks(Fich).Worksheets(Liv).Cells(i + 4, ng + 8 + j) = perc_sg(i, j + 1)
    Next
Next

End Sub
Sub os(ByVal n_comb As Integer, ByRef s_min As Integer, ByVal nh As Integer)
Rem     ... finds the optimal final state ...

Min = 10000000000#
s_min = 1

For i = 1 To n_comb
    If Tcost(i, nh) < Min Then

```

```

        s_min = i
        Min = Tcost(i, nh)
    End If
Next

```

```

End Sub

```

```

Sub caminho(ByVal ng As Integer, ByVal s As Integer, ByVal nh As Integer,
ByVal cont As Integer, ByVal lambda As Double)

```

```

    Dim pot_g(11) As Double

```

```

    Fich = "P_Dinamica_solver_v2.xls"

```

```

    Livro = "Sol_Solver"

```

```

    For g = 1 To ng

```

```

        pot_g(g) = 0

```

```

    Next

```

```

    For hora = nh To 1 Step -1

```

```

        If hora = nh Then

```

```

            E_act = s

```

```

        Else

```

```

            E_act = E_ant

```

```

        End If

```

```

        E_ant = Traj_s(E_act, hora)

```

```

        For g = 1 To ng

```

```

            pot_g(g) = pow(g, E_act, hora) + pot_g(g)

```

```

        Next

```

```

    Next

```

```

    Workbooks(Fich).Worksheets(Livro).Cells(cont + 4, 2) = lambda

```

```

    Workbooks(Fich).Worksheets(Livro).Cells(cont + 4, 3) = Tcost(s, nh)

```

```

    For g = 1 To ng

```

```

        Workbooks(Fich).Worksheets(Livro).Cells(cont + 4, g + 3) = pot_g(g)

```

```

    Next

```

```

End Sub

```

```

Sub sol_final(ByVal g As Integer, ByVal s As Integer, ByVal cont As Integer,
ByVal lambda As Double, ByVal enrg As Double, ByVal nh As Integer)

```

```

    Fich = "P_Dinamica_solver_v2.xls"

```

```
Liv = "Sol_Solver"
```

```
Workbooks(Fich).Worksheets(Liv).Cells(cont + 2, g * 7) = cont
```

```
Workbooks(Fich).Worksheets(Liv).Cells(cont + 2, g * 7 + 1) = lambda
```

```
Workbooks(Fich).Worksheets(Liv).Cells(cont + 2, g * 7 + 2) = Tcost(s, nh)
```

```
Workbooks(Fich).Worksheets(Liv).Cells(cont + 2, g * 7 + 3) = s
```

```
End Sub
```