**Projeto Prático**

Problemas 4, 6 e 8.

Modelos Matemáticos de Análise e de Apoio à Decisão

2020/2021

Jorge Silva

Bacar Braima Baldé

Iduino Indeque

**ÍNDICE**

[INTRODUÇÃO](#_mncmeav4k218) **3**

[PROGRAMAÇÃO LINEAR](#_bxbkq2gfavle) **4**

[LINGUAGEM GAMS](#_ktyy57n0h44r) **5**

[APRESENTAÇÃO E RESOLUÇÃO DE PROBLEMAS](#_tl02dc9j117e) **6**

[Problema 4](#_j3upomvl22r6) 6

[Problema 6](#_tsvhdywe4kvi) 15

# **INTRODUÇÃO**

No âmbito da disciplina de Modelos Matemáticos de Análise e de Apoio à Decisão foi-nos proposto a realizar um projeto prático, onde vamos resolver alguns problemas, problemas esses que serão descritos ao longo desse projeto. No caso do nosso grupo vamos resolver os problemas 4, 6 e 8, usando a linguagem de modelagem GAMS.

O problema 4 consiste em estabelecer uma afetação empregado-máquina de forma a ter o menor custo possível, e para isso usamos o algoritmo de húngaro e de seguida usamos o sistema GAMS[[1]](#footnote-0) para obter o resultado mais otimizado possível.

Já o problema 6, passa por reduzir os custos associado à ligação das redes informáticas de 5 agências à rede da sua Sede, e essa ligação pode ser direta ou indiretamente, desde que os custos sejam mínimos.

E por último o problema 8, que tem como objetivo minimizar os custos de um professor que está sempre a viajar entre cidades para proferir seminários.

# **PROGRAMAÇÃO LINEAR**

Segundo Bernardo (2016), a programação linear é um ramo da programação matemática que está incluso nos métodos quantitativos que auxilia na tomada de decisão. Segundo o autor, os problemas de programação matemática normalmente estão ligados à afetação de recursos escassos a usos alternativos, de forma a satisfazer um objetivo, que está sujeito a um conjunto, de condições ou restrições, a solução que satisfaz simultaneamente a função objetivo e as restrições é chamada a solução ótima do problema.

Martins (2017), afirma que a programação linear consiste em um problema de optimização em que a função que se pretende optimizar (função objectivo) é linear e está sujeita a restrições (condições) que redefinem o seu domínio.

**Onde se aplica?**

A programação linear pode ser utilizada na resolução de problemas reais, das mais diversas áreas, nomeadamente: economia, gestão, física e engenharia, etc.

**A quem se destina?**

Destina-se, essencialmente, a administradores, engenheiros, técnicos, …, com o objectivo de, por exemplo, minimizar custos ou maximizar lucros.

# **LINGUAGEM GAMS**

Para Silva (2010), o GAMS é uma Linguagens de Modelagem e foi projetada para o desenvolvimento e solução de modelos de programação matemática complexa.

Nesse projeto o GAMS destaca-se pela sua utilização para resolução da otimização dos problemas apresentados (4, 6 e 8). O autor supracitado, destaca a significância das Linguagens de Modelagem (nesse caso o GAMS), visto que os problemas analisados estão se tornando cada vez mais complexos, e para os solucionar, temos de usar sistemas que propiciam aos modeladores a possibilidade de obter resultados em curto espaço de tempo.

A figura 1 mostra como está estruturado o GAMS.

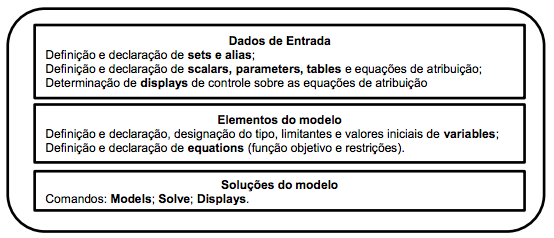


Figura 1 - Estrutura geral do modelo GAMS

fonte: Silva (2010)

# **APRESENTAÇÃO E RESOLUÇÃO DE PROBLEMAS**

## **Problema 4**

Numa fábrica foram instaladas 4 novas máquinas dispondo para a sua laboração de 5 empregados. A direção da fábrica tem por objetivo estabelecer uma afetação empregado– máquina. Com esta finalidade, e após vários testes, estimou-se esta matriz de custos:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| máquina/empregado | empregado 1 | empregado 2 | empregado 3 | empregado 4 | empregado 5 |
| máquina 1 | 35 | 95 | 87 | - | - |
| máquina 2 | 53 | 90 | 85 | 65 | 69 |
| máquina 3 | 91 | 93 | 73 | 75 | 73 |
| máquina 4 | 15 | 20 | 22 | 30 | 42 |

onde o símbolo “—” significa que o empregado não tem formação suficiente para operar a máquina respetiva. Formule matematicamente o problema, e use o GAMS para o resolver.

Para resolver o problema 4, será usado o problema de afetação conforme descrito a seguir.

**Resolução do problema 4**

**Problema de afetação**

Assim como o problema de transporte, o problema de afetação também é um caso particular do problema de Programação Linear. Segundo Fogliatto (2015), o problema de afetação consiste em afetar n indivíduos a n máquina/tarefas, tendo como objetivo a minimização do custo total envolvido neste processo, de modo que cada empregado seja afetada uma e uma só máquina. Este problema pode ser entendido como um problema de otimização combinatória, sendo n! o número total de afetações possíveis. No entanto, esta abordagem do problema é muito trabalhosa. Para o nosso problema em particular foi feito o levantamento das variáveis de decisão a fim de formular o modelo. Nesse caso em particular tendo em conta que o modelo possui 4 máquinas e 5 empregados, o que não satisfaz as exigências do problema de afetação, foi adicionada mais uma máquina fictícia de forma a equilibrar o sistema. Sendo assim segue a tabela onde pode-se ver isso.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| máquina/empregado | empregado 1 | empregado 2 | empregado 3 | empregado 4 | empregado 5 |
| máquina 1 | 35 | 95 | 87 | - | - |
| máquina 2 | 53 | 90 | 85 | 65 | 69 |
| máquina 3 | 91 | 93 | 73 | 75 | 73 |
| máquina 4 | 15 | 20 | 22 | 30 | 42 |
| máquina 5 | 0 | 0 | 0 | 0 | 0 |

Tabela 1 - Tabela do sistema equilibrado.

Baseando na tabela 1, vamos formular o problema.

**Variáveis de decisão:**

X*ij* = máquina *i* afetada ao empregado *j*; *i* = 1,..., 5 e *j* = 1,...,5.

Onde X*ij* = 1 se empregado *j* for afetada a máquina *i*, caso contrário, 0 se empregado *j* não for afetada a máquina *i*.

**Função objetivo:**

Min z = 35x11 + 95x12 + 87x13 + 0x14 + 0x25 + 53x21 + 90x22 + 85x23 + 65x24 + 69x25 + 91x31 + 93x32 + 73x33 + 75x34 + 73x35 + 15x41 + 20x42 + 22x43 + 30x44 + 42x45 + 0x51 + 0x52 + 0x53 + 0x54 + 0x55

Restrição das máquinas:

x11 +x12 +x13 +x14 + x15 = 1

x21 +x22 +x23 +x24 + x25 = 1

x31 +x32 +x33 +x34 + x35 = 1

x41 +x42 +x43 +x44 + x45 = 1

x51 +x52 +x53 +x54 + x55 = 1

Restrição Empregados:

x11 +x21 +x31 +x41 +x51= 1

x12 +x22 +x32 +x42 +x52= 1

x13 +x23 +x33 +x43 +x53= 1

x14 +x24 +x34 +x44 +x54= 1

x15 +x25 +x35 +x45 +x55= 1

Restrição negatividade dos funcionários não afetadas a máquina:

x14 = 0

x15 = 0

Tendo o problemas de alocação formulado, usaremos o Algoritmo Húngaro para resolvê-la.

De acordo com Kolinski (2013), o algoritmo de húngaro, consiste em resolver o problema usando os passos a seguir:

**Passo 1 -** Determinar o elemento de custo mínimo em cada linha da tabela. Construir uma nova tabela subtraindo de cada custo o custo mínimo da linha correspondente. De seguida determinar o custo mínimo em cada coluna. Construir uma nova tabela subtraindo de cada custo o custo mínimo da coluna correspondente.

**Passo 2 -** Determinar o número mínimo de linhas (horizontais e/ou verticais) necessárias para cobrir todos os zeros na tabela de custos reduzidos. Se forem necessárias n linhas, a resposta ótima é dada pelos zeros cobertos por linhas na tabela. Se menos de n linhas forem necessárias, siga para o passo 3.

**Passo 3 -** Determinar o menor elemento ≠ 0 dentre os elementos não cobertos por linhas na tabela; seja k o valor deste elemento. Subtrair k de cada elemento não-coberto da tabela e adicionar k a cada elemento coberto por duas linhas na tabela. Volte ao passo 2.

Os passos a seguir encontram-se nas tabelas a seguir com as respectivas descrições.

Passo 1: Subtração de cada custo o custo mínimo da linha correspondente. No extremo direito de cada linha estão os mínimos. Quanto às colunas, é irrelevante identificar os mínimos tendo em conta que todas as colunas possuem como custo mínimo zero (0).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 35 | 95 | 87 | 0 | 0 | 0 |
| 53 | 90 | 85 | 65 | 69 | -53 |
| 91 | 93 | 73 | 75 | 73 | -73 |
| 15 | 20 | 22 | 30 | 42 | -15 |
| 0 | 0 | 0 | 0 | 0 | 0 |

Após a operação no passo 1 temos a tabela resultante a seguir.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 35 | 95 | 87 | 0 | 0 |
| 0 | 37 | 12 | 12 | 16 |
| 18 | 20 | 0 | 2 | 0 |
| 0 | 5 | 7 | 15 | 27 |
| 0 | 0 | 0 | 0 | 0 |

Passo 2: Cobrir todos os zeros na tabela de custos reduzidos, com número mínimo de linhas (horizontais e/ou verticais) necessárias. A seguir segue a tabela resultante da operação.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 35 | 95 | 87 | 0 | 0 |
| 0 | 37 | 12 | 12 | 16 |
| 18 | 20 | 0 | 2 | 0 |
| 0 | 5 | 7 | 15 | 27 |
| 0 | 0 | 0 | 0 | 0 |

Passo 3: Determinar o menor número de entre os não cobertos/riscados (nesse caso é o nº 5). Subtrair o nº 5 por cada elemento não riscado. A seguir somar o nº 5 a cada um dos custos cobertos por duas linhas (nesse casos os nº das variáveis x11=35; x31=18; x51=0).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 40 | 95 | 87 | 0 | 0 |
| 0 | 32 | 7 | 7 | 11 |
| 23 | 20 | 0 | 2 | 0 |
| 0 | 0 | 2 | 10 | 22 |
| 5 | 0 | 0 | 0 | 0 |

Uma vez transformada a matriz de custos numa matriz com n zeros enquadrados (nesse caso 5), esses zeros correspondem à afectação óptima. Sendo assim temos o quadro ótimo onde os zeros enquadrados correspondem a x14, x21, x33, x42, x55, onde:

x14 = 0;

x21 = 53;

x33 = 73;

x42 = 20;

x55 = 0;

Com isso: z = x14 + x21 + x33 + x42 + x55 z = 146.

O método acima foi usado para confirmar o resultado obtido usando o GAMS, que por sua vez é uma linguagem de modelagem (LM), orientada para a construção de modelos matemáticos, que por vezes são complexas.

Na figura 2 pode-se ver o quadro inicial do problema inserida no GAMS:

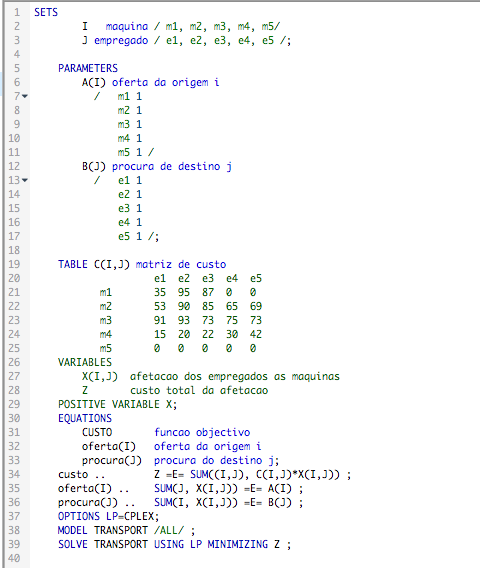


Figura 2 - Input no GAMS problema 4.

Na figura 2 foi apresentada o modelo inserida no GAMS contendo os blocos:

* SETS: são os índices;
* SCALAR, PARAMETER ou TABLE: são os parâmetros;
* VARIABLE: variável dependente;
* POSITIVE VARIABLE: variáveis de decisão;
* EQUATIONS: equações;
* MODEL: modelo;
* SOLVE: ordem para resolução;
* MINIMIZING ou MAXIMIZING: direção para a otimização;
* USING LP: Especificação do tipo de PPL em uso;
* DISPLAY: Mostrar o resultado.

Já agora com os inputs apresentados na figura 2, obteve-se o custo mínimo de 146 (z=146), conforme mostra a figura 3.

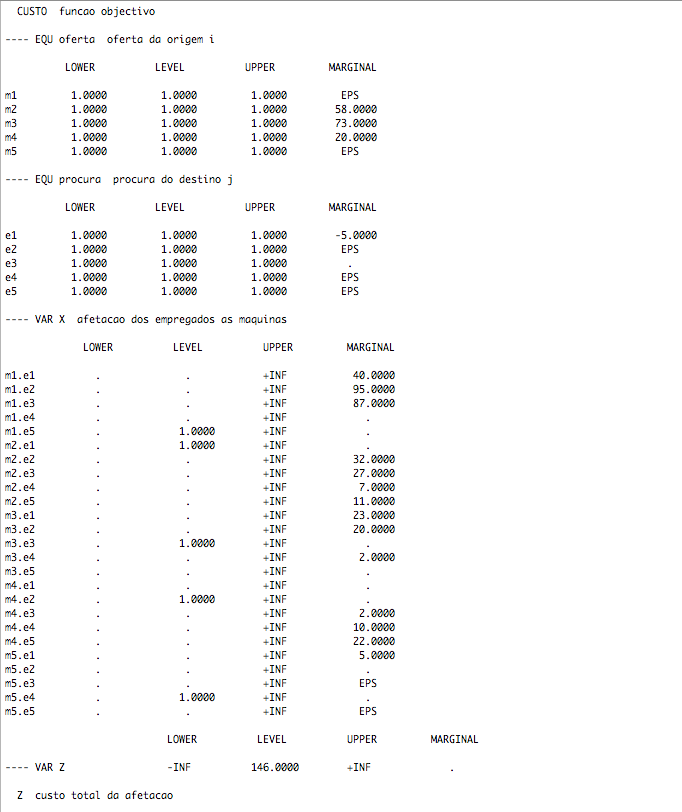


Figura 3 - Output GAMS problema 4.

Na solução, as colunas LOWER e UPPER mostram os limites inferior e superior das variáveis, respectivamente. O valor zero é representado por um ponto. O valor EPS representa um zero (armazenado), ou seja, significa que está muito próximo, ou é mesmo zero. +INF significa infinito. A coluna LEVEL reporta o valor encontrado como solução da execução do modelo. A coluna MARGINAL representa o valor de complementaridade da variável (valor sombra). Deve-se dar uma atenção especial às colunas LEVEL e MARGINAL.

**Validação do modelo**

Segundo Gurgel (2010), a complementaridade implica que, em equilíbrio, o valor da variável será positivo ou o valor marginal será positivo, mas nunca ambos ao mesmo tempo. Se ambos aparecem como positivos, é necessário checar o modelo, nesse caso o nosso modelo é válido porque esse fenômeno não se regista em nenhuma das linhas.

A figura 3, além de nos mostrar a validade do modelo, também nos mostra quais as melhores combinações entre as variáveis de modo a afetar empregado/máquina de forma a ter custo reduzido. Nesse caso as variáveis são: m1e5; m2e1; m3e3; m4e2; m5e4;

Como apresentada na tabela 1, a afeção m1e5 (máquina 1 a empregado 5) teria o custo de 0, m2e1 teria o custo de 53, m3e3 teria o custo de 73, m4e2 teria custo de 20 e por último m5e4, teria o custo de 0. O somatório de m1e5, m2e1, m3e3, m4e2 e m5e4 teria o custo total de 146.

## **Problema 6**

Um banco deseja ligar as redes informáticas de cada uma das suas agências (A1, A2, A3,

A4 e A5) à rede da sua Sede, usando linhas de fibra óptica. A ligação de qualquer agência

com a sede pode ser direta ou indireta através de uma outra agência. No entanto é exigido

que cada agência fique por alguma via ligada à sede. O custo das linhas é diretamente

proporcional ao comprimento total utilizado, onde a distância entre os vários pares de

agências é dada pela tabela seguinte:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Sede | A1 | A2 | A3 | A4 | A5 |
| Sede | - | 190 | 70 | 115 | 270 | 160 |
| A1 | - | - | 100 | 240 | 215 | 50 |
| A2 | - | - | - | 140 | 120 | 220 |
| A3 | - | - | - | - | 175 | 80 |
| A4 | - | - | - | - | - | 310 |

Pretende-se apurar a forma de fazer a ligação das agências através das linhas por forma

a ligar cada agência (direta ou indiretamente) à sede, de modo a que o custo total seja

mínimo. Formule matematicamente o problema, e use o GAMS para obter uma solução.

**Resolução do problema 6**

**Problema 8**

Um certo professor a viver em Coimbra foi convidado a proferir três seminários durante

cada um dos três dias de um fim-de-semana alargado (sexta-feira, sábado e domingo).

Cada seminário terá lugar numa das cidades de Coimbra, Porto, Braga, Lisboa e Évora,

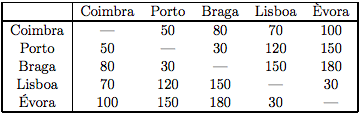
recebendo para o efeito um subsídio de 400, 450, 440, 500 e 550 euros respectivamente. O

seminário de Braga não poderá decorrer no sábado e o professor tem de estar em Lisboa

na segunda-feira a seguir ao fim-de-semana.

As deslocações serão suportadas pelo professor, com custos (em euros) definidos na

seguinte tabela:



Pretende-se saber os seminários e respetivos locais que o professor deve ministrar de

modo a maximizar o seu lucro. Formule matematicamente o problema, e obtenha uma

solução recorrendo ao GAMS.

**Resolução problema 8**

Para resolver o problema em causa comecemos por definir as variáveis de decisão.

**Variáveis de decisão:**

X*ij* = local de origem *i* local de destino *j*; *i* = 1,..., 5 e *j* = 1,...,5.

Onde X*ij* = 1 se o professor tiver o custo de deslocamento do local i para o local *j*, caso contrário, 0 se ele não tiver um custo de viagem do local *i* para o local *j*.

Com isso as **variáveis de decisão** são:

x11, x12, x13, x14, x15,

x21, x22, x23, x24, x25,

x31, x32, x33, x34, x35,

x41, x42, x43, x44, x45,

x51, x52, x53, x54, x55;

**Função objetivo:**

Nesse problema de programação linear, o objetivo passa por minimizar os custos de deslocamento do professor do local *i* para o local *j.* Ao minimizar os custos automaticamente ele consegue maximizar o seu lucro.

Min z = 400\*x11 + (450 - 50)\*x12 + (440 - 80)\*x13 + (500 - 70)\*x14 + (550 - 100)\*x15 +

(400 - 50)\*x21 + 450\*x22 + (440 - 30)\*x23 + (500 - 120)\*x24 + (550 - 150)\*x25 +

(400 - 80)\*x31 + (450 - 30)\*x32 + 440\*x33 + (500 - 150)\*x34 + (550 - 180)\*x35 +

(400 - 70)\*x41 + (450 - 120)\*x42 + (440 - 150)\*x43 + 500\*x44 + (550 - 30)\*x45 +

(400 - 100)\*x51 + (450 - 150)\*x52 + (440 - 180)\*x53 + (440 - 30)\*x54 + 550\*x55

**Restrição geral:**

x11 + x12 + x13 + x14 + x15 = 1

x21 + x22 + x23 + x24 + x25 = 1

x31 + x32 + x33 + x34 + x35 = 1

x41 + x42 + x43 + x44 + x45 = 1

x51 + x52 + x53 + x54 + x55 = 1

**Restrição negatividade:**

x11 = x22 = x33 = x44 = x55 = 0;

Isso porque a deslocação Xij nesse caso particular não teve custos associados, como mostra a tabela de custos desse problema.

Após a inserção desses *inputs,* no GAMS obteve-se o seguinte resultado como mostra a figura 4.

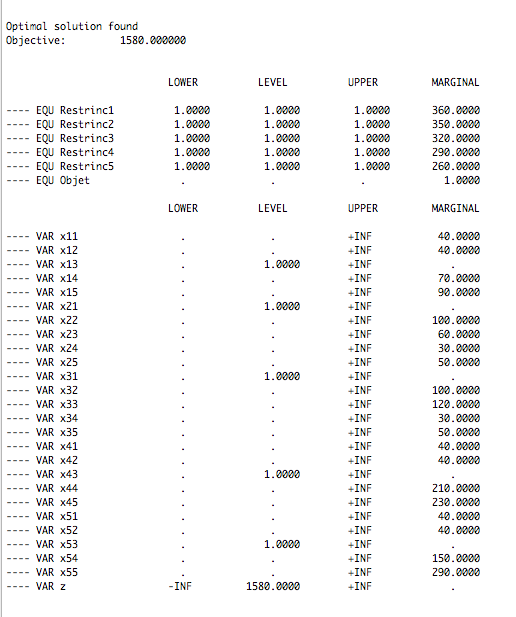


Figura 4 - Output GAMS do problema 8.

Na figura 4, é apresentado o valor do z (1580), e é apresentada a combinação das variáveis de forma obter o melhor resultado. E as variáveis a combinar nesse caso seriam: x13, x21, x31, x43, x53. E o somatório dessas variáveis teria o valor de 1580 inerente ao menor custo.

Igual procedimento foi aplicado no problema 4.

Para obter esse resultado supracitado foram inseridos os *inputs,* conforme mostra a figura 5.

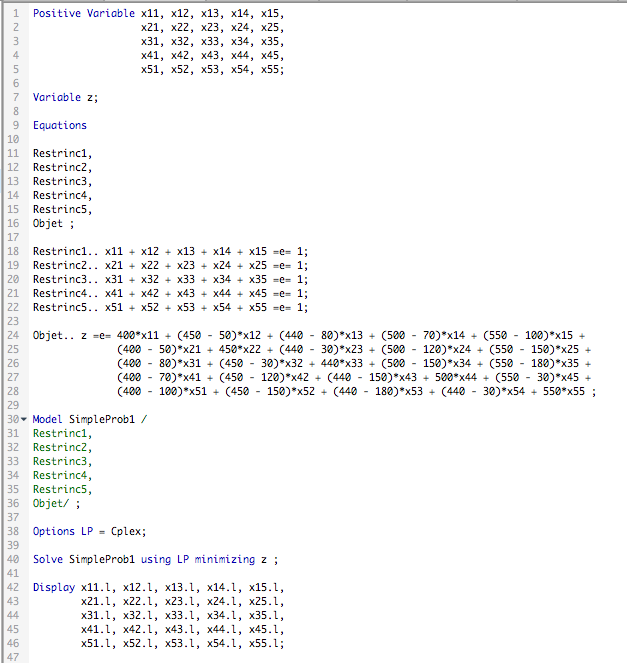


Figura 5 - Input GAMS problema 8.

**REFERÊNCIAS**

SILVA, A. (2010) Pesquisa Operacional: Desenvolvimento e Otimização de Modelos

Matemáticos por meio da Linguagem Gams. Universidade Estadual Paulista.

BERNARDO, M. (2016) Programação Linear. Universidade Aberta.

MARTINS, M. (2017). Programação Linear: Fundamentos e Ensino de Álgebra. Faculdade

de Ciências e Tecnologia da Universidade de Coimbra.

FOGLIATTO, F. (2015). Pesquisa Operacional. DEPROT / UFRGS.

KOLINSKI, A. (2013). Innovations in Management and Production Engineering: The Use Of

Hungarian Method In The Evaluation Of Production Efficiency

GURGEL, A. (2010). Introdução ao MPSGE e GAMS.

1. general algebraic modeling system [↑](#footnote-ref-0)