Modelagem de Problemas em Programação Linear Inteira

Josemir Araújo Neves

Doutorando em Matemática Computacional – UFPE – PE.

Empresa de Pesquisa Agropecuária do Rio Grande do Norte - EMPARN

Av. Jaguarari, 2192 - CEP 59062-500- Natal – RN - Email: josemiremparn@rn.gov.br

Maria Angela Caldas Didier

Doutoranda em Matemática Computacional – UFPE - PE.

Departamento de Matemática, UFRPE.

Rua Dom Manuel de Medeiros S/N Dois Irmãos – CEP: 22171-030-Recife -PE

Resumo: O homem sempre buscou formas aproximativas para representar a complexidade do mundo ao seu redor. A este processo de interpretação do mundo dá-se o nome de modelagem. Quando essa interpretação pode ser expressa matematicamente de forma quantitativa temos então os modelos de programação matemática e otimização que constituem uma parte importante da área da Pesquisa Operacional. Neste trabalho são abordados diversos aspectos destes modelos, como se dá o processo de modelagem e por fim são apresentados problemas práticos resultantes de trabalhos e pesquisas desenvolvidas.

Palavras-chave: Modelagem de Problemas, Programação Matemática, Programação Linear Inteira.

Introdução

Na impossibilidade de lidar com a complexidade dos problemas do mundo o homem sempre buscou formas aproximativas de interpretação, e este processo de busca da realidade se constitui um fenômeno de modelagem (GOLDBARG & LUNA, 2005).

Os problemas que podem ser expressos matematicamente de forma quantitativa constituem os chamados Modelos de Programação Matemática (GOLDBARG & LUNA, 2005). Neste trabalho estaremos descrevendo alguns aspectos tocantes à modelagem matemática de problemas desta natureza e exemplificando a sua aplicabilidade a partir de exemplos práticos que foram resultados de trabalhos e pesquisas desenvolvidas.

Modelagem – Conceitos Iniciais

Define-se como modelagem o processo que engloba todas as atividades de interpretação e simulação da realidade. Os modelos resultantes deste processo constituem representações simplificadas da natureza dada à alta complexidade dos fenômenos modelados, mas que preservam para determinadas situações e enfoques uma equivalência adequada (GOLDBARG & LUNA, 2005).

Para se conseguir modelos eficientes o domínio de três habilidades se faz necessário: 1) Foco holístico; 2) Tratamento eclético da dimensão de análise e 3) Tradução adequada (Figura 1) (GOLDBARG & LUNA, 2005).

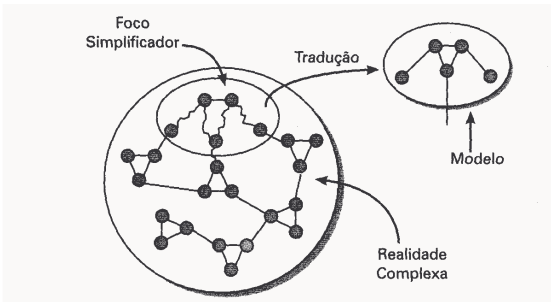


Figura 1-O processo de tradução

Fonte: (GOLDBARG & LUNA, 2005).

A Figura 2 abaixo ilustra as dimensões de complexidade de muitos modelos encontrados na prática inclusive os sócio-econômicos. Note que o espaço viável para a atuação dos modelos de programação matemática e otimização se resume às áreas limitadas pelas regiões, o que nos dá uma idéia da dificuldade de se modelar e simular problemas práticos da nossa realidade (GOLDBARG & LUNA, 2005).

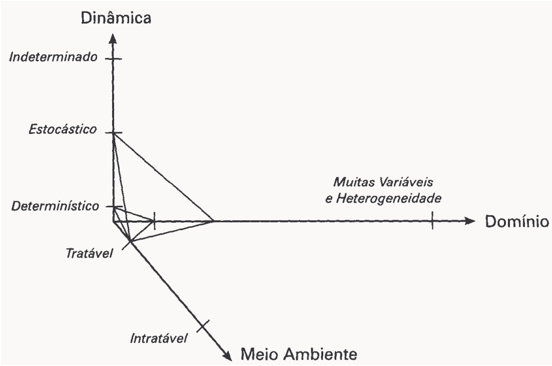


Figura 2-Dimensões da complexidade de modelos

Fonte: (GOLDBARG & LUNA, 2005).

O poder de representatividade e a capacidade de simplificação é a característica do modelo que o torna desejável e atrativo para a sua implementação computacional (GOLDBARG & LUNA, 2005).

Portanto são características principais de um modelo simples:

- A pouca influência exercida pelas variações em seu meio ambiente;

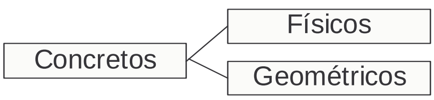
- A estabilidade estrutural, homogeneidade e poucas variáveis;

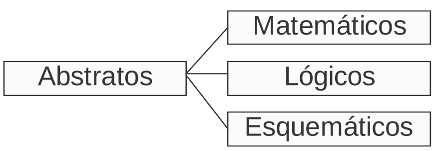
- Previsibilidade de comportamento.

Classificação dos modelos

Como são vários os enfoques para abordar o processo de construção de modelos temos a seguinte classificação:

- Quanto a sua natureza:





- Quanto às propriedades:

Icônicos - As propriedades relevantes dos objetos reais são representadas como tais, e diferem entre si na escala. Ex: fotografias, mapas.

Analógicos - Usam um conjunto de propriedades inerentes ao modelo para representar o conjunto de propriedades da realidade. Ex.: substituição de sistemas hidráulicos por elétricos.

Simbólicos - Usam letras, números e outros símbolos para representar as variáveis e suas relações. Ex.: fluxogramas e DFD.

- Quanto as variáveis controladas

Explanatórios - possuem variáveis controladas.

Descritivos - não possuem variáveis controladas.

- Classificação de Emshoff (1970)

Descritivos - são expressos em linguagem corrente.

Físicos - possuem alto custo e são muito específicos. Exemplo: miniaturas.

Procedurais - são também denominados de simulação.

O processo de modelagem

De uma maneira geral pode-se resumir o processo de modelagem sob uma ótica operacional (Figura 3), satisfazendo elementos palpáveis tais como: objetivos, variáveis de decisão e controle e níveis de detalhe.

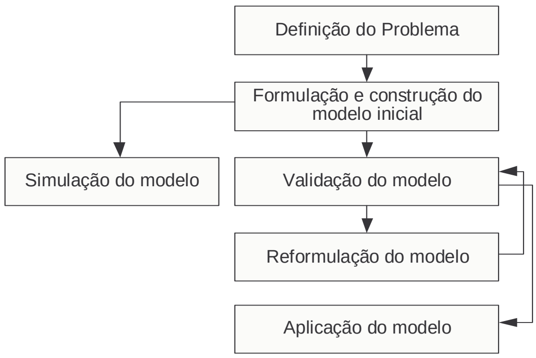


Figura 3 – Processo de Modelagem

Fonte: (GOLDBARG & LUNA, 2005)

As fórmulas e equações que compõem um modelo não estão prontas e envolvem aspectos tais como: intuição, experiência, criatividade e poder de síntese, por parte dos pesquisadores e/ou equipe responsável pela resolução do problema (GOLDBARG & LUNA, 2005).

Programação Matemática

Os problemas que podem ser expressos matematicamente de forma quantitativa constituem os chamados Modelos de Programação Matemática, que fazem parte da Pesquisa Operacional, tradicional disciplina que engloba várias das mais consagradas técnicas de modelagem matemática (GOLDBARG & LUNA, 2005).

Dentre as diversas áreas que compõe a Programação Matemática destacam-se:

- Programação Linear – nas quais os modelos de otimização que a compõem apresentam variáveis contínuas com comportamento linear, tanto em relação às restrições como em relação à função objetivo.

- Programação Não-Linear – os modelos apresentam algum tipo de não-linearidade seja na função objetivo ou nas restrições.

- Programação Inteira – quando qualquer de suas variáveis assumirem valores inteiros.

A Programação Inteira ainda subdivide-se em:

- Programação Inteira Mista – com variáveis inteiras e contínuas;

- Programação Inteira Binária - onde todas as variáveis assumem somente os valores 0-1.

Entre as diversas aplicações deste tipo de programação destacam-se: programação de trens, planejamento de produção, planejamento da geração elétrica, telecomunicações, planejamento do número e rotas de aviões, otimização do padrão de corte de matéria-prima nas indústrias, entre outras (WOLSEY, 1998).

Os Problemas de Programação Linear (PPL), de uma maneira geral podem ser formulados como:

$$\begin{aligned}
&Otimizar \quad z = cx \\
&sujeito \quad a: \\
&\qquad Ax \le b \quad ou \quad Ax \ge b \\
&\qquad x \ge 0
\end{aligned}$$

Passos p/ formulação de um PPL

√ Definição das atividades – Envolve a definição de todas as atividades e as unidades de medidas a elas associadas.

√ Definição dos Recursos - Considerando-se os insumos disponíveis por cada atividade, determinam-se os recursos que estão sendo usados e produzidos em cada atividade.

√ Cálculo dos Coeficientes de Insumo/produção: Definição do relacionamento das atividades e recursos em termos de recursos necessários por unidade de atividade produzida.

√ Determinação das condições externas: É a determinação da quantidade de cada recurso disponível para o processo modelado.

√ Formalização do Modelo: Consiste em associar quantidades não negativas x_1, x_2, ... ,x_n a cada uma das atividades, escrever as equações de balanceamento e indicar o uso de cada recurso.

Modelagem de Problemas-Exemplos

SUDOKO

O SUDOKU é um quebra-cabeça ("puzzle") baseado na colocação lógica de números. Seu objetivo é a colocação de números de 1 a 9 em cada uma das células vazias numa grade de 9x9, subdividida por subgrades de tamanho 3x3 chamadas regiões. Em cada coluna, linha e/ou região não poderá haver repetições de números.

O SUDOKU foi projetado anonimamente por Howard Garns, um arquiteto aposentado de 74 anos de idade e construtor independente de quebra-cabeças ("puzzles"), que o publicou pela primeira vez em 1979. Embora tenha se inspirado provavelmente no quadrado latino, invenção do século XVIII do matemático suíço Leonhard Euler, ele adicionou uma terceira dimensão (a limitação regional) a construção matemática (Figura 4).

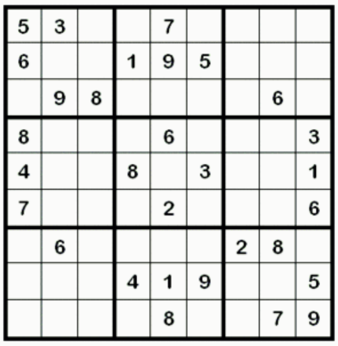


Figura 4 – Grade 9x9 do Sudoku.

Existem diversas variações do jogo, para regiões regulares: grades de 4x4 com regiões 2x2, grades de 12x12 com regiões 3x4, 16x16 com regiões 4x4, 20x20 com regiões 4x5, 25x25 com regiões 5x5, este último é publicado com o nome de Sudoku Gigante; e para regiões irregulares (Figura 5) com grades 5x5 regiões pentaminó (chamadas Logi-5); 6x6 com regiões 2x3, 7x7 com 6 regiões heptominó (sete células) e com regiões desconexas. São encontradas ainda variações do jogo circulares, em 3 dimensões e com restrições a mais como regiões diagonais, pares e ímpares e ainda também o Sudoku sem a limitação regional.

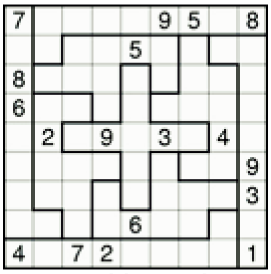


Figura 5 – Sudoku Irregular (9x9).

O problema geral de solucionar puzzles SUDOKU em tabuleiros n^2 x n^2 de blocos n x n é conhecido como *NP*-completo (YATO & SETA, 2002). Isto dá algumas indicações de porque o SUDOKU é difícil de ser resolvido, contudo em tabuleiros de tamanhos finitos o problema é finito e pode ser solucionado através de um autômato finito probabilístico que conhece toda a árvore do jogo. O problema também pode ser modelado como um Problema de Satisfabilidade de Expressões Booleanas (*SAT Problem*) e ser resolvido utilizando-se as técnicas de resoluções para estes tipos de problemas (SIMONIS, 2005; LYNCE & OUAKNINE, 2006), ou ser modelado como um Problema de Programação Linear Inteira (BARTLETT & LANGVILLE, 2006).

O número de soluções do SUDOKU para uma grade padrão de 9 x 9 foi calculado em 2005 por Bertram Felgenhauer(http://www.afjarvis.staff.shef.ac.uk/sudoku/), como sendo 6.670.903.752.021.072.936.960.

Formulação para o SUDOKU Tradicional e Irregular

Considere o SUDOKU com regiões regulares (paralelogramos m x n), em uma grade de tamanho mn x mn.

Seja uma matriz $A_{mn\times mn} = \{a_{ij}\}$, onde

$a_{ij} \in \{1,\dots,mn\}; \quad i, j, k \in \{1,\dots,mn\}$.

- Variáveis de Decisão:

$$x_{ijk} = \begin{cases} 1, & se\ a_{ij} = k \\ 0, & se\ a_{ij} \neq k \end{cases}$$

- Função Objetivo:

$$\max \quad z = \sum_{0 \le i,j,k \le mn} (1 + \varepsilon_{ijk}) x_{ijk}$$

- Restrições:

a) Viabilidade das linhas:

$$\sum_{j=1}^{mn} x_{ijk} = 1, \quad \forall \quad i, k \in \{1,\dots,mn\}$$

b) Viabilidade das colunas:

$$\sum_{i=1}^{mn} x_{ijk} = 1, \quad \forall \quad j, k \in \{1,\dots,mn\}$$

c) Viabilidade das células:

$$\sum_{k=1}^{mn} x_{ijk} = 1, \quad \forall \quad i, j \in \{1,\dots,mn\}$$

d) Viabilidade das regiões:

$$\sum_{i=1}^{n} \sum_{j=1}^{m} x_{n\sigma_i + i, n\sigma_j + j, k} = 1,$$

$$\forall \quad k, \quad 0 \le \sigma_i \le m-1; \quad \forall \quad 0 \le \sigma_j \le n-1.$$

Formulação Completa (Sudoku Tradicional):

$$\max \quad z = \sum_{0 \le i,j,k \le mn} (1 + \varepsilon_{ijk}) x_{ijk}$$

s.a.:

$$\sum_{j=1}^{mn} x_{ijk} = 1, \quad \forall \quad i, k \in \{1,\dots,mn\}$$

$$\sum_{i=1}^{mn} x_{ijk} = 1, \quad \forall \quad j,k \in \{1,\ldots,mn\}$$

$$\sum_{k=1}^{mn} x_{ijk} = 1, \quad \forall \quad i,j \in \{1,\ldots,mn\}$$

$$\sum_{i=1}^{n}\sum_{j=1}^{m} x_{n\sigma_i+i,n\sigma_j+j,k} = 1,$$
$$\forall \quad k, \quad 0 \le \sigma_i \le m-1; \quad \forall \quad 0 \le \sigma_j \le n-1.$$

$$x_{ijk} \in \{0,1\}, \quad \varepsilon_{ijk} \in \Re$$

Para o Sudoku com regiões irregulares, conexas e com n células, a formulação é a seguinte:

Seja uma matriz $A_{n\times n} = \{a_{ij}\}$, onde $a_{ij} \in \{1,\ldots,n\}$; $i,j,k \in I = \{1,\ldots,n\}$, assim teremos:

Formulação Completa (Sudoku Irregular):

$$\max \quad z = \sum_{0\le i,j,k\le n} (1+\varepsilon_{ijk})x_{ijk}$$

s.a.:

$$\sum_{j=1}^{n} x_{ijk} = 1, \quad \forall \quad i,k \in I$$

$$\sum_{i=1}^{n} x_{ijk} = 1, \quad \forall \quad j,k \in I$$

$$\sum_{k=1}^{n} x_{ijk} = 1, \quad \forall \quad i,j \in I$$

$$\sum_{(i,j)\in S_t} x_{ijk} = 1, \quad \forall k \in I,$$

$$S_t = \{(i,j) \mid i,j \in I, \exists\, (i \pm \alpha, j \pm \beta) \in S_t \},$$

$$c/\alpha + \beta = 1, \alpha,\beta \in \{0,1\}, t \in I, \ |S_t| = n, \ \bigcap_{t\in I} S_t = \phi.$$

$$x_{ijk} \in \{0,1\}, \quad \varepsilon_{ijk} \in \Re$$

Note que a segunda formulação é uma generalização da primeira basta, para isso, tomar os conjuntos S_t iguais as sub-regiões definidas na primeira formulação.

Unidade Móvel de Pistoneio

Na bacia petrolífera potiguar terrestre encontra-se uma vasta quantidade de poços terrestres *não surgentes*, ou seja, poços cuja característica principal é a pouca vazão de óleo o que não justifica a utilização dos métodos tradicionais de explotação, tais como: cavalo-de-pau e BCP (Bomba de Cavidade Progressiva). Para estes tipos de poços é empregada uma coleta periódica realizada a partir de uma Unidade Móvel de Pistoneio (UMP), que se trata de um caminhão ou trator acompanhado de tanque e acessórios de elevação do fluido (guincho hidráulico, rolo de cabo de aço, lança, etc.) (Figura 6). Este veículo circula livremente de poço em poço, seguindo uma rota diária pré-estabelecida para o operador. Ao chegar ao poço o conjunto é montado de tal forma que se introduz neste um copo de pistoneio suspenso por um cabo de aço, e que permite a retirada do fluido por levantamento, em ações repetidas, até que se chegue ao limite de capacidade do poço, ou seja, num nível de fluido bastante baixo. Esvaziado um poço, o conjunto é desmontado e, então, desloca-se para um outro mais adiante. O fluido é armazenado no próprio tanque da UMP para posteriormente ser transferido para um tanque de armazenamento.

Figura 6 – Unidade Móvel de Pistoneio.
Fonte: (NEVES, 2000)

O poço esvaziado será realimentado durante, em média, alguns dias, após o que estará em condições de ser novamente pistoneado.

Portanto, o problema consiste em formular matematicamente a programação diária para as UMPs que atendem a cada campo de petróleo terrestre (NEVES, 2000; ALOISE et al, 2000; ALOISE et al, 2000(b); ALOISE et al, 2000(c); ALOISE et al, 2000(d); SANTOS et al, 2001).

O Problema de Otimização do Emprego da Unidade Móvel de Pistoneio acima descrito foi denominado por **Problema da Coleta Orientada** (**PCO**) (ALOISE et al, 2000; ALOISE et al, 2000(c); ALOISE et al, 2000(d)). Este problema é uma variante do Problema de Orientação (PO), classificado como NP-Árduo (GOLDEN et al. 87).

Formulação para o PCO

Dados de entrada (Constantes):

m - número de UMP's,
n - número total de poços,
p_i - volume de óleo que pode ser extraído do poço i,
bsw_i – taxa de água por volume recuperado para o poço i,
m_i - tempo de montagem do equipamento no poço i,
o_i - tempo de operação no poço i,
d_i - tempo de desmontagem do equipamento no poço i,
L_k - duração da jornada diária da UMP k,
c_{ij} - menor tempo de percurso entre os poços i e j

Variáveis de Decisão:

P - Produção diária total de óleo a ser obtida (volume em m³)

$$x_{ik} = \begin{cases} 1, & \text{se o poço i é explotado, } \textit{pela UMP k} \\ & \text{para } i = 1,\ldots,n \quad e \quad k = 1,\ldots,m \\ 0, & \text{caso contrário.} \end{cases}$$

$$r_{ijk} = \begin{cases} 1, & \textit{se os poços i e j são consecutivos no roteamento} \\ & \textit{da UMP k para } i, j = 0,\ldots,n \quad e \quad k = 1,\ldots,m \\ 0, & \text{caso contrário.} \end{cases}$$

Formulação:

$$Maximizar \quad P = \sum_{k=1}^{m} \sum_{i=1}^{n} (1 - bsw_i)\, p_i\, x_{ik} \qquad (1)$$

sujeito a :

$$\sum_{i=1}^{n} (m_i + o_i + d_i) x_{ik} + \sum_{i=0}^{n} \sum_{j=0}^{n} r_{ijk} c_{ij} \le L_k, \quad para\ k = 1,\ldots,m \qquad (2)$$

$$\sum_{j=0}^{n} r_{jik} - x_{ik} = 0, \quad para \quad i = 1,\ldots,n \quad e \quad k = 1,\ldots,m \qquad (3)$$

$$\sum_{j=0}^{n} r_{ijk} - x_{ik} = 0, \quad para \quad i = 1,\ldots,n \quad e \quad k = 1,\ldots,m \qquad (4)$$

$$\sum_{j=1}^{n} r_{j0k} = 1, \quad para\ k = 1,\ldots,m \qquad (5)$$

$$\sum_{j=1}^{n} r_{0jk} = 1, \quad para\ k = 1,\ldots,m \qquad (6)$$

$$\sum_{k=1}^{m} x_{ik} \le 1, \quad para\ i = 1,\ldots,n \qquad (7)$$

$$\sum_{\substack{j=0 \\ j \ne i}}^{n} y_{ijk} - \sum_{\substack{j=0 \\ j \ne i}}^{n} y_{jik} = x_{ik}, \quad para \quad i = 1,\ldots,n \quad e \quad k = 1,\ldots,m \qquad (8)$$

$$y_{ijk} \le (l+1)\, r_{ijk} \quad para\ i, j = 0,1,\ldots,n \quad e \quad k = 1,\ldots,m \qquad (9)$$

$$x_{ik}, r_{ijk} \in \{0,1\} \quad e \quad y_{ijk} \ge 0 \quad \forall i, j, k \quad com \quad i \ne j \qquad (10)$$

As restrições (2) impõem que o somatório dos tempos de operação mais deslocamento até o poço a partir do depósito não exceda o limite L_k da jornada diária de cada UMP k. As restrições de (3) a (6) garantem a viabilidade da rota definida pelas variáveis r_{ijk} para cada UMP k. As equações (3) e (4) garantem que**:** (*i*) caso x_{ik}=1, há apenas um arco entrando e um arco saindo do poço *i*, pois a decisão é pistoneá-lo; (*ii*) caso $x_{ik} = 0$, não há arco algum entrando ou saindo do poço *i*, pois a decisão é não pistoneá-lo. De maneira análoga, as equações (5) e (6) garantem que há apenas um arco entrando e um arco saindo do depósito. A restrição (7) garante que cada poço somente seja atribuído a uma única UMP. As equações (8) e (9) são equações de fluxo que garantem a viabilidade da rota eliminando loops e sub-rotas. E em (10) é garantido que as variáveis x_{ik} e r_{ijk} sejam binárias, e que as variáveis y_{ijk} sejam não-negativas.

Considerações sobre a implementação computacional e resolução dos modelos apresentados

Diversos são os pacotes de softwares disponíveis no mercado para modelar e resolver problemas desta natureza. Entre os pacotes de softwares comerciais destacamos: AIMMS, XPRESS-MP, ILOG OPL STUDIO, LINDO, etc. e entre os pacotes de software livres: GLPK, COIN-OR e (ZIMPL-SOPLEX-SCIP) (IGNÁCIO & FILHO, 2004; ACHTERBERG, 2006).

Os modelos aqui apresentados foram implementados no Mosel, ambiente de modelagem matemática do software de otimização XPRESS-MP, e solucionados com o solver do software. Para o Sudoku foram testadas grades tradicionais e irregulares de até 25x25, e para o problema da UMP os testes foram feitos em uma malha simulada de 200 poços, com diversas configurações: número de poços, número de UMPs e jornadas iguais e diferentes para cada UMP.

Conclusão

Cada vez mais a programação matemática desempenha papel essencial nas atividades das empresas principalmente para garantir a própria sobrevivência destas, dado ao alto nível de exigências por eficiência dos mercados no mundo atual. Este trabalho demonstra um pouco da factibilidade da utilização desta ferramenta em aplicações do nosso cotidiano sejam elas de natureza distintas.

Referências

[1] ALOISE, D.J., BARROS, C.A., NEVES, J.A., MOURA, L.S.S., ASSMANN, B.W., SOUZA, M.C. Otimização do Emprego da Unidade Móvel de Pistoneio através de GRASP. In: XXXII Simpósio Brasileiro de Pesquisa Operacional, Viçosa, MG, 2000. (c)

[2] ALOISE, D.J., BARROS, C.A., NEVES, J.A., MOURA, L.S.S., SOUZA, M.C. Um Algoritmo Genético para uma Variante do Problema de Orientação. In: X Latinamerican Conference On Operational Research - CLAIO'2000, México, 2000.

[3] ALOISE, D.J., MOURA, L.S.S., ASSMANN, B.W., BARROS, C.A., NEVES, J.A. Otimização do Emprego da Unidade Móvel de Pistoneio na Explotação de Campos Petrolíferos de Poços não surgentes. In: Rio Oil & Gás Expo and Conference, Riocentro, Rio de Janeiro, 2000. (b)

[4] ALOISE, D.J., NEVES, J.A., BARROS, C.A., MOURA, L.S.S., ASSMANN, B.W. Um Algoritmo Genético na Otimização do Emprego da Unidade Móvel de Pistoneio. In: XXXII Simpósio Brasileiro de Pesquisa Operacional, Viçosa, MG, 2000. (d)

[5] ARCHTERBERG, T., GROTSCHEL, M., KOCH, T. Software for teaching modeling of integer programming problems. ZIP-Report 06-23. Zuse Institute Berlin, Berlin, 2006.

[6] BARTLETT, A.C., LANGVILLE, A.N. An Integer Programming Model for the Sudoku Problem. Preprint. March 2006. (http://math.cofc.edu/langvillea/).

[7] GOLDBARG, M.C., LUNA, H.P.L. Otimização combinatória e programação linear: modelos e algoritmos – 2º ed. - Rio de Janeiro: Elsevier, 2005.

[8] GOLDEN, B., WANG, Q., LIU, L.. A Multifaced Heuristics for the Orienteering Problem. Naval Research Logistics. 35 : 359-366 (1987).

[9] IGNÁCIO, A.A.V., FILHO, V.J.M.F. O uso do software de modelagem AIMMS na solução de problemas de programação matemática. Pesquisa Operacional, v.24, n.1, p.197-210, 2004.

[10] LYNCE, I., OUAKNINE, J. Sudoku as a SAT Problem, *Proceedings of the Ninth International Symposium on Artificial Intelligence and Mathematics (AIMATH 2006)*, Jan. 2006.

[11] NEVES, J.A. Uma Aplicação de Algoritmo Genético na Otimização do Emprego da Unidade Móvel de Pistoneio. Dissertação (Mestrado). Área de Concentração: Matemática Aplicada. Departamento de Informática e Matemática Aplicada, UFRN, Natal, 2000.

[12] SANTOS, A.C.; ALOISE, D.J.; BARROS, C.A.; NEVES, J.A., ASSMANN,B.W.;MOURA, L.S.S. - Development of a Georeferenced Computational System to Support Decision in order to Optimize the use of Oil Recovery Mobile Systems (ORS). Fourth International Conference on Optimization - Optimization 2001, July 23th - 25th 2001 - Portugal, Aveiro University.

[13] SIMONIS, H. Sudoku as a constraint problem. In CP Workshop on Modeling and Reformulating Constraint Satisfaction Problems, pages 13-27, October 2005.

[14] WOLSEY, L.A. Integer Programming. Wiley - Interscience publications. John Wiley & Sons, INC, 1998.

[15] YATO, T., SETA, T. Complexity and completeness of ending another solution and its application to puzzles.In Proceedings of the National Meeting of the Information Processing Society of Japan (IPSJ), 2002.