



PROGRAMAÇÃO LINEAR INTEIRA MISTA E ALGORITMO GENÉTICO APLICADOS AO PROBLEMA DE TRANSFERÊNCIA E ESTOCAGEM DE PRODUTOS EM UMA INDÚSTRIA PETROLÍFERA

Angela Olandoski Barboza^a, Flavio Neves Junior^a, Silvana Ligia Vincenzi Bortolotti^a, Rosely Antunes de Souza^a

^a Universidade Tecnológica Federal do Paraná (UTFP)

Resumo

O crescimento do comércio internacional de produtos e serviços, a troca constante de informações vêm desafiando os administradores a definir novos rumos para suas empresas. Desta forma, buscam-se novas tecnologias para conseguir-se a melhoria da eficiência operacional. Em especial, a indústria petrolífera brasileira tem investido na pesquisa aplicada, desenvolvimento e capacitação tecnológica para manter-se competitiva no mercado internacional. Muitos são os problemas que ainda devem ser estudados neste setor produtivo. Dentre estes, pode-se destacar os problemas de transferência e estocagem de produtos. Este trabalho aborda um problema de programação da produção (*scheduling*) envolvendo estocagem e distribuição de diesel em uma refinaria de petróleo. Para solucionar este problema, primeiramente foi utilizado um modelo de Programação Linear Inteira Mista (PLIM) com representação do tempo discreta. Este modelo foi resolvido com o aplicativo computacional LINGO 8.0. Em seguida, desenvolveu-se uma metodologia aplicando Algoritmo Genético de Estado Estacionário Híbrido integrado à Programação Linear para a resolução do mesmo modelo. Após a realização de testes com o modelo em PLIM e com a nova metodologia, foi possível após a análise dos resultados, concluir que a nova abordagem obteve desempenho satisfatório em termos de qualidade de solução e tempo computacional quando comparada à modelagem PLIM.

Palavras-chave: Scheduling, Programação Linear Inteira Mista, Algoritmo Genético.

1. INTRODUÇÃO

O mercado mundial vem se modificando em decorrência dos avanços tecnológicos, da globalização, das grandes fusões de empresas e da maior conscientização ecológica. As transformações econômicas, a dinâmica dos mercados e a crescente competitividade fazem parte da globalização mundial, a qual intensifica o comércio internacional de produtos e serviços, promove intercâmbio cultural acentuado e a constante troca de informações. Tais mudanças implicam em um aumento da competitividade, obrigando as organizações a criarem soluções inovadoras para se manterem no mercado. Diante deste perfil, as empresas buscam um novo modelo de gestão baseado, principalmente, na redução dos custos dos produtos e das margens de lucratividade. Almejam, ainda, uma melhoria substancial do nível de serviços relacionados à distribuição para poderem competir com outras empresas. De maneira geral, os custos produtivos e a qualidade dos produtos ten-

dem a se equiparar, independentemente da empresa que os produz e, por isso, o grande diferencial está na otimização das operações, ou seja, na capacidade dos produtos chegarem ao cliente final na quantidade certa, no tempo esperado e a um preço justo.

No contexto das indústrias, um dos problemas é o de otimização do sistema de produção, no qual o planejamento e a programação de produção são de fundamental importância. Esta otimização é hoje essencial para a competitividade das indústrias e minimização de custos de operação e envolve a determinação da taxa de produção, a política de manutenção de equipamentos, as estratégias de regulagem, a alocação e designação de produtos às máquinas, a alocação de produtos finais e, finalmente, a política de entrega a clientes.

Hoje, as empresas brasileiras vivem uma realidade em que ganhos de produtividade significam sobrevivência. Os sistemas de produção *just in time* (enxuta), isto é, siste-



mas preparados para atender a demanda por produtos a qualquer momento e com estoques reduzidos, apresentam vantagens com relação à produtividade, eficiência e qualidade. Desta forma, torna-se útil a criação de técnicas que proporcionem aos empresários e gerentes a certeza de que estão utilizando um sistema de produção que, se bem administrado, pode gerar vantagens competitivas.

Entre os vários setores produtivos existentes, que buscam melhorias em termos de desempenho, têm-se as refinarias de petróleo. Um dos problemas a serem otimizados neste setor, é o da programação de produção (*scheduling*). Parte desse problema envolve estocagem e distribuição de produtos, cuja solução ótima é difícil de ser encontrada devido à sua característica combinatorial. Por esse motivo, tem sido objeto de muitos estudos, tais como os trabalhos de Jia *et al.* (2003), Joly Moro *et al.* (2002) e Wu *et al.* (2005). Moro (2000) salienta que muitos problemas envolvendo refinaria de petróleo são modelados como problema de Programação Linear Inteira Mista (PLIM) e resolvido com auxílio de pacotes computacionais comerciais. Contudo, pela sua natureza, o aumento do número de variáveis inteiras torna impraticável o uso destes aplicativos, por exigirem um tempo computacional excessivo. Em geral, diante deste cenário, justifica-se a procura por outras abordagens para a resolução deste problema.

Desta forma, este trabalho foi desenvolvido com o objetivo de se encontrar uma metodologia mais eficiente em termos computacionais, para a resolução de um problema de PLIM, do que os métodos clássicos, entre estes, o método *Branch and Bound*. Especificamente, buscou-se desenvolver um modelo em PLIM para um problema de transferência e estocagem de diesel em uma refinaria de petróleo, com representação de tempo discreto e resolver aplicando o método *Branch and Bound*. Em seguida, desenvolveu-se uma nova metodologia para resolver o mesmo modelo, aplicando Algoritmo Genético integrado à Programação Linear (PL). A escolha da aplicação do Algoritmo Genético para a busca de soluções adequadas para o problema abordado em um tempo computacional aceitável foi feita em razão da sua comprovada eficiência, versatilidade e robustez. Dentre os algoritmos existentes que poderiam também ser utilizados, vale citar: Busca Tabu, *Simulated Annealing* e Evolução Diferencial para problemas discretos.

A PLIM e o Algoritmo Genético têm sido utilizados na solução de muitos problemas de otimização, tanto em refinarias de petróleo como também em outras áreas. Alguns trabalhos como os de Thunberg *et al.* (2011), Subbaraj, Rengaraj e Salivahanan (2011), Wang *et al.* (2011), Zhang *et al.* (2009), Floudas *et al.* (2005), Duan *et al.* (2011), Morabito *et al.* (2008), Carrano *et al.* (2005), Almeida *et al.* (2003) e Liu *et al.* (2011) foram desenvolvidos utilizando estas técnicas da pesquisa operacional.

2. FUNDAMENTAÇÃO TEÓRICA

2.1 Programação Linear (PL)

A PL é uma das mais importantes e mais utilizadas técnicas de pesquisa operacional. A simplicidade do modelo envolvido e a disponibilidade de uma técnica de solução programável em computador como o método Simplex, descrito por Dantzig (1963), facilitam sua aplicação. Esta técnica é amplamente utilizada, pois possui habilidade para modelar importantes e complexos problemas de decisão. A descrição do método Simplex pode ser encontrada em Zions (1974).

Um problema de PL é composto por:

1. Uma função linear formada com as variáveis de decisão chamada de função objetivo, cujo valor deve ser otimizado;
2. Relações de interdependência entre as variáveis de decisão que se expressam por um conjunto de equações ou inequações lineares, chamadas de restrições do modelo;
3. Variáveis de decisão que devem ser positivas ou nulas.

A equação (1) mostra a formulação para um problema de PL:

$$\begin{aligned}
 & \text{maximize (ou minimize)} \\
 & z = \sum_{j \in N} c_j x_j, N = \{1, \dots, n\} \\
 & \text{sujeito a} \\
 & \sum_{j \in N} a_{ij} x_j (\leq, = \text{ ou } \geq) b_i, i \in M = \{1, 2, \dots, m\} \\
 & x_j \geq 0, j \in N
 \end{aligned} \tag{1}$$

no qual c_j , a_{ij} e b_i são constantes conhecidas para todo i e j ; x_j são variáveis não negativas.

As restrições do problema podem ser transformadas em equações adicionando-se uma variável de folga (não negativa) x_{n+p} se a i -ésima desigualdade é do tipo \leq e subtraindo uma variável de folga (não negativa), x_{n+k} se a k -ésima desigualdade é do tipo \geq . Considerando que ao serem acrescentadas as variáveis de folga, obtém-se um total de $m + n$ variáveis, pode-se escrever o problema na forma matricial como mostra a equação (2):

$$\begin{aligned}
 & \text{maximize (ou minimize)} \\
 & z = cx \\
 & \text{sujeito a} \\
 & Ax = b \\
 & x \geq 0
 \end{aligned} \tag{2}$$



no qual, c é um vetor linha de ordem $(n + m)$, A é uma matriz $m \times (m + n)$, x é um vetor coluna de ordem $(m + n)$ e b é um vetor coluna de ordem m .

2.1.1 Programação Inteira (PI) e Programação Inteira Mista (PIM)

Alguns problemas reais requerem o uso de variáveis que assumem somente valores inteiros. Quando isto acontece tem-se um problema de Programação Inteira (PI). Este problema está definido na equação (3) a seguir:

$$\begin{aligned} &\text{maximize (ou minimize)} \\ & z = g_0(x_1, x_2, \dots, x_n) \\ &\text{sujeito a} \\ & g_i(x_1, x_2, \dots, x_n) (\leq \text{ou} = \text{ou} \geq) b_i, i \in M = \{1, 2, \dots, m\} \\ & x_j \geq 0, j \in N = \{1, 2, \dots, n\} \\ & x_j \text{ inteira}, j \in I \subseteq N \end{aligned} \quad (3)$$

no qual, $x_j, j \in N$ são as variáveis, $g_i, i \in M \cup \{0\}$ são funções das variáveis x_1, x_2, \dots, x_n , e $b_i, i \in M$ são constantes conhecidas. Se $I = N$, isto é, todas as variáveis são inteiras, então o problema é dito de PI. Caso contrário, se $I \subset N$, então chama-se de problema de Programação Inteira Mista (PIM).

2.1.2 Programação Linear Inteira (PLI) e Programação Linear Inteira Mista (PLIM)

Em muitos dos problemas abordados em PI, as funções $g_i, i \in \{0\} \cup M$ na equação (3) são lineares e o modelo pode então ser descrito como mostra a equação (4) a seguir:

$$\begin{aligned} &\text{maximize (ou minimize)} \\ & z = \sum_{j \in N} c_j x_j, N = \{1, \dots, n\} \\ &\text{sujeito a} \\ & \sum_{j \in N} a_{ij} x_j (\leq, \geq \text{ou} =) b_i, i \in M = \{1, \dots, m\} \\ & x_j \geq 0, j \in N \\ & x_j \text{ inteira}, j \in I \subseteq N \end{aligned} \quad (4)$$

onde, c_j, a_{ij} e b_i são constantes conhecidas para todo i e j , e x_j são variáveis não negativas. Se $I = N$, isto é, todas as variáveis são inteiras, então temos um problema Programação Linear Inteira (PLI). Se $I \subset N$, então o problema é de Programação Linear Inteira Mista (PLIM).

Muitos modelos práticos de PLI restringem algumas das variáveis inteiras para valores "0" ou "1" e, neste caso, tem-se um problema de Programação Linear Inteira Binária (PLIB). Estas variáveis são usadas para decisão: sim ("1") e não ("0").

Existem muitos problemas de programação de produção (*scheduling*) que podem ser colocados como problemas de PLIM, pois os modelos matemáticos de otimização correspondentes envolvem variáveis contínuas e discretas que devem satisfazer um conjunto de restrições lineares de igualdade e desigualdade (Moro, 2000). A resolução para problemas de otimização linear inteira mista, entendida como a obtenção de uma solução ótima, pode ser difícil, pela sua natureza combinatorial. Considera-se que o espaço de soluções inteiras é constituído de um número finito de pontos. Mesmo no caso misto, o espaço de busca é primeiramente controlado pelas variáveis inteiras. Na forma mais simples, os métodos de enumeração analisam todos os pontos. É o que se chama de busca exaustiva. Um simples método de busca exaustiva pode se tornar mais eficiente se enumerar apenas uma parte das soluções candidatas enquanto descarta pontos que não são promissores. A eficiência de um algoritmo de busca depende de sua capacidade em descartar pontos de solução não promissores. Dentre estas técnicas podem ser citadas: separação e avaliação progressiva ou *Branch and Bound* (Zionts, 1974) e enumeração implícita (Taha, 1975). Estas técnicas podem utilizar uma relaxação do problema para obter em tempo razoável uma estimativa para o valor da melhor solução que pode ser encontrada em cada ramo da enumeração.

2.2 Ferramentas computacionais para problemas de otimização

Existem vários aplicativos computacionais desenvolvidos para a resolução de problemas de busca e otimização, incorporando diversas técnicas. O progresso na solução destes tipos de problemas tem sido impulsionado pelos avanços obtidos na PL.

Os aplicativos que resolvem problemas de PL utilizam o método Simplex e/ou o método de busca por ponto interior. Já para problemas de PLI e PLIM, a maior parte dos aplicativos utiliza o método *branch and bound*.

Pinto (2000) listou em tabelas vários aplicativos com especificações, tais como: dados da empresa que desenvolveu ou produto, plataformas, formatos de entrada, tipos de problemas que resolve e outros.

Dentre estes aplicativos, vale destacar:

- LINGO 8.0 (LINDO Systems Inc., 2002): é uma ferramenta de simples utilização e serve para resolver e analisar soluções para problemas de grande porte em PL e não linear. Os modelos inteiros são resolvidos através do método *branch and bound*.



- ILOG CPLEX 8.0 (ILOG CPLEX, 2002): foi desenvolvido para resolver problemas de PL. Resolve também problemas de Fluxos em Rede, Programação Quadrática e PIM.

2.3 Algoritmo Genético (AG)

O AG foi inicialmente proposto por Holland (1975) que se inspirou no mecanismo de evolução das espécies, tendo como base os trabalhos de Darwin sobre a origem das espécies e a genética natural devida principalmente a Mendel. Dentre as definições de AG encontradas na literatura, tem-se a de Tanomaru (1995) que o define como métodos computacionais de busca baseados nos mecanismos de evolução natural e na genética. Em AG, uma população de possíveis soluções para o problema em questão evolui de acordo com operadores probabilísticos concebidos a partir de metáforas biológicas, de modo que há uma tendência de que, na média, os indivíduos representem soluções cada vez melhores à medida que o processo evolutivo continua. O AG é considerado eficiente para busca de soluções ótimas, ou quase ótimas em uma grande variedade de problemas, pois não impõe muitas das limitações encontradas nos métodos de busca tradicionais. Além disso, em muitos casos em que outras estratégias de otimização falham na busca de uma solução, o AG converge.

Segundo Goldberg (1989), o AG difere dos métodos tradicionais de busca e otimização, principalmente em quatro aspectos:

- opera numa população de pontos e não a partir de um ponto isolado;
- opera num espaço de soluções codificadas e não no espaço de busca diretamente;
- necessita somente de informação sobre o valor de uma função objetivo para cada membro da população e não requerem derivadas ou qualquer outro tipo de conhecimento;
- usa transições probabilísticas e não regras determinísticas.

No AG trabalha-se com um conjunto de indivíduos (população) no qual cada elemento é candidato a ser a solução desejada. A função a ser otimizada representa o ambiente no qual a população inicial se insere. Espera-se então que, através dos mecanismos de evolução das espécies e da genética natural, os indivíduos mais aptos tenham maior probabilidade de se reproduzirem e que a cada nova geração estejam mais aptos ao ambiente. A próxima geração será uma evolução da anterior e para que isso ocorra, os mais aptos deverão possuir maior probabilidade de serem

selecionados para dar origem à nova geração. Contudo, alguns poucos não muito aptos também poderão ser selecionados. O mecanismo responsável que faz a escolha seletiva de indivíduos denomina-se seleção. Após a seleção, o próximo passo é a aplicação dos operadores genéticos que atuam sobre os genótipos produzindo novos indivíduos, também denominados de mecanismos de busca (Holland, 1975). Dentre estes mecanismos, os mais comumente empregados são o cruzamento ou recombinação e a mutação. Se as operações de seleção, recombinação e mutação forem bem conduzidas, espera-se que a nova geração seja, em média, melhor do que a que lhe deu origem.

3. MODELAGEM DO PROBLEMA

Neste trabalho foi desenvolvida uma técnica para solução de um problema de *scheduling* de curto termo em um sub-sistema de uma refinaria de petróleo. A busca por técnicas mais eficientes se justifica, principalmente, pela complexidade das operações de programação de produção que geram problemas combinatórios de grande porte e também pelas limitações computacionais encontradas no processamento. Em geral, problemas de *scheduling* em refinarias de petróleo são formulados como modelos de otimização inteira mista e resolvidos utilizando-se técnicas exatas, mas o tempo despendido para chegar à solução é inviável para a implementação dos mesmos. O sub-sistema estudado neste trabalho envolve as operações de transferência e estocagem de diesel. São considerados o desenvolvimento e a solução de modelos de otimização para o *scheduling* de um conjunto de operações que incluem: o recebimento do diesel nos tanques, armazenamento e envio deste produto a clientes.

A modelagem desenvolvida para este sistema tem como objetivo a obtenção da sequência para o fluxo e armazenamento do diesel de forma a atender às restrições de operações e de demandas a um custo mínimo. Os parâmetros envolvidos na modelagem deste problema incluem a configuração do parque de tancagem, as restrições de operação e a demanda de diesel como dados de entrada e o gerenciamento de atividades como dados de saída. A Figura 1 ilustra o relacionamento destes parâmetros com o modelo matemático.

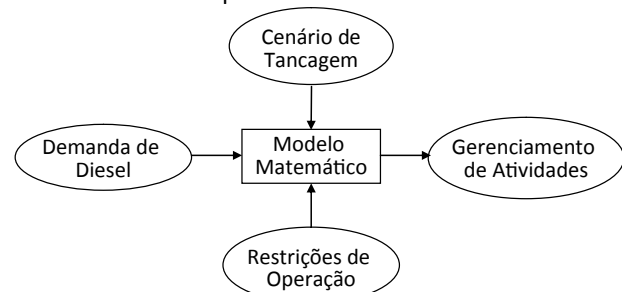


Figura 1. Modelo Matemático – Entradas e Saída



Fonte: Barboza (2005)

As seguintes premissas e restrições operacionais foram consideradas na modelagem do problema:

- Os volumes de diesel estocados nos tanques são conhecidos;
- Os tanques são dedicados, ou seja, armazenam um único tipo de diesel;
- Todos os tanques devem estar disponíveis durante o horizonte de planejamento, ou seja, não devem sofrer manutenção ou qualquer operação que impossibilite seu uso;
- Os tempos de transição entre tarefas não são considerados, por serem desprezíveis em relação às demais operações;
- Cada um dos tanques não pode efetuar operações de carga e descarga simultaneamente;
- Cada tanque pode receber de uma única origem num determinado intervalo de tempo;
- Cada tanque pode enviar para um único destino num determinado intervalo de tempo;
- As restrições operacionais de vazão mínima/máxima de fluxo de recebimento e envio de produtos devem ser respeitadas;
- Os tanques não poderão estar com volume abaixo do mínimo e nem acima do máximo estipulados;
- Quando um tanque iniciar o recebimento de carga, deverá fazê-lo até o enchimento total do mesmo;
- Após o enchimento, o tanque deverá ficar parado por um período mínimo estipulado para repouso e análise do produto estocado. Após este período o tanque estará disponível para envio;
- Todo o lote requerido pelo cliente deve ser bombeado de forma ininterrupta;
- As demandas de recebimento e envio podem ser satisfeitas em qualquer instante do horizonte de planejamento.
- Para a resolução do modelo foram utilizadas as seguintes técnicas: *Branch and Bound*, Método Simplex e AG.

4. MODELO EM PLIM

O primeiro modelo matemático para o problema de transferência e estocagem utiliza a modelagem PLIM com discretização uniforme no tempo. São utilizadas variáveis contínuas, inteiras e binárias. As variáveis binárias representam as decisões a serem tomadas. Eventos de qualquer tipo devem começar e terminar no início de cada intervalo de tempo. Como já citado, o objetivo deste modelo é o de minimizar custos operacionais. As restrições foram construídas respeitando-se os procedimentos operacionais, as restrições físicas do processo e a demanda.

Nomenclatura utilizada no Modelo

Índices:

tq	-	representa o número do tanque: $tq = 1, 2, \dots, TQ$
c	-	representa o número do cliente: $c = 1, 2, \dots, C$
t	-	representa o número do intervalo de tempo: $t = 1, 2, \dots, T$

Conjuntos:

CTQ	-	conjunto formado pelos tanques tq
CC	-	conjunto formado pelos clientes c
CT	-	conjunto formado pelos intervalos de tempo t

Vale observar o uso de índices subscritos, para representar os tanques, os clientes ou os intervalos de tempo envolvidos nos parâmetros e variáveis.

Parâmetros:

CBC	-	custo de bombeio para o cliente c (unidades monetárias/mil m ³);
CAtq	-	custo de armazenamento no tanque tq (unidades monetárias/mil m ³);
CTRtq	-	custo de troca do tanque tq que recebe da produção (unidades monetárias);
QRmin	-	vazão mínima de recebimento pelos tanques tq (mil m ³ /h);
QRmax	-	vazão máxima de recebimento pelos tanques tq (mil m ³ /h);
QEminc	-	vazão mínima de envio ao cliente c (mil m ³ /h);
QEmaxc	-	vazão máxima de envio ao cliente c (mil m ³ /h);
Volmintq	-	volume mínimo permitido no tanque tq (mil m ³);
Volmaxtq	-	volume máximo permitido no tanque tq (mil m ³);
Volinitq	-	volume no tanque tq no início do processo (mil m ³);
DEMc	-	demanda de diesel do cliente c (mil m ³).



Variáveis binárias:

$$Rec_{tq,t} = \begin{cases} 1, \text{ se o tanque } tq \text{ recebe diesel da produção no intervalo de tempo } t \\ 0, \text{ caso contrário} \end{cases}$$

$$Env_{tq,c,t} = \begin{cases} 1, \text{ se o tanque } tq \text{ envia diesel para o cliente } c \text{ no intervalo de tempo } t \\ 0, \text{ caso contrário} \end{cases}$$

$$XI_{c,t} = \begin{cases} 1, \text{ se o cliente } c \text{ começa a receber diesel no intervalo de tempo } t \\ 0, \text{ caso contrário} \end{cases}$$

$$XF_{c,t} = \begin{cases} 1, \text{ se o cliente } c \text{ termina de receber diesel no intervalo de tempo } t \\ 0, \text{ caso contrário} \end{cases}$$

$$TR_{tq,tq',t} = \begin{cases} 1, \text{ se o tanque } tq \text{ termina de receber e o tanque } tq' \text{ começa a receber no intervalo de tempo } t \\ 0, \text{ caso contrário} \end{cases}$$

Variáveis Contínuas:

$QR_{tq,t}$ volume de diesel recebido pelo tanque tq no intervalo de tempo t;

$QE_{tq,c,t}$ volume de diesel enviado pelo tanque tq ao cliente c no intervalo de tempo t;

$Vol_{tq,t}$ volume de diesel estocado no tanque tq no tempo t;

TI_c marca o momento de início do recebimento do cliente c;

TF_c marca o momento de término do recebimento do cliente c;

Função objetivo:

Para este modelo assume-se que a operação ótima do problema de transferência e estocagem é aquela que minimiza os custos operacionais do processo, definidos pelos custos de bombeamento de envio adicionado ao custo de armazenamento de produto nos tanques e, finalmente, adicionado ao custo de transição por troca de tanques na operação de recebimento do processo.

$$(5) \quad Custos = \sum_{tq \in CTQ} \sum_{c \in CC} CB_{tq,c,t} \cdot QE_{tq,c,t} + \sum_{tq \in CTQ} CA_{tq} \cdot Vol_{tq,t} + \sum_{tq \in CTQ} \sum_{tq' \neq tq \in CTQ} CTR_{tq,tq',t} \quad (5)$$

Restrições:

Um tanque não pode receber e enviar diesel ao mesmo tempo. Portanto, são três os estados possíveis para o tanque: recebendo, enviando ou ocioso.

$$Rec_{tq,t} + \sum_{c \in CC} Env_{tq,c,t} \leq 1 \quad \forall tq = 1, \dots, TQ; t = 1, \dots, T \quad (6)$$

Como o fluxo de produto do processo é contínuo, haverá sempre um tanque recebendo diesel em cada intervalo de tempo.

$$\sum_{tq \in CTQ} Rec_{tq,t} = 1 \quad \forall t = 1, \dots, T \quad (7)$$

Somente um tanque poderá enviar diesel para um determinado cliente em cada intervalo de tempo.

$$\sum_{tq \in CTQ} Env_{tq,c,t} \leq 1 \quad \forall c = 1, \dots, C; t = 1, \dots, T \quad (8)$$

Quando um cliente iniciar seu recebimento de diesel, deverá fazê-lo sem interrupção. Portanto, haverá apenas uma variável de início e de término igual a 1.

$$\sum_{t \in CT} XI_{c,t} \leq 1 \quad \forall c = 1, \dots, C \quad (9)$$

$$\sum_{t \in CT} (XI_{c,t} - XF_{c,t}) = 0 \quad \forall c = 1, \dots, C \quad (10)$$

O momento de início e término de bombeio são armazenados nas variáveis auxiliares TI_c e TF_c .

$$\sum_{t \in CT} XI_{c,t} \leq 1 \quad \forall c = 1, \dots, C \quad (11)$$

$$\sum_{t \in CT} (XI_{c,t} - XF_{c,t}) = 0 \quad \forall c = 1, \dots, C \quad (12)$$

O fluxo de recebimento deve estar entre o fluxo mínimo e máximo estipulados se a variável $Rectq,t$ for igual a "1". Se o valor da variável $Rectq,t$ for "0", o fluxo $QR_{tq,t}$ também será igual a "0".

$$QR_{min} \cdot Rec_{tq,t} \leq QR_{tq,t} \leq QR_{max} \cdot Rec_{tq,t} \quad \forall tq = 1, \dots, TQ; t = 1, \dots, T \quad (13)$$

O fluxo de envio deve estar entre o fluxo mínimo e máximo estipulados se a variável $Env_{tq,c,t}$ for igual a "1". Se o valor da variável $Env_{tq,c,t}$ for "0", o fluxo $QE_{tq,c,t}$ também será igual a "0".

$$QE_{min,c} \cdot Env_{tq,c,t} \leq QE_{tq,c,t} \leq QE_{max,c} \cdot Env_{tq,c,t} \quad \forall tq = 1, \dots, TQ; c = 1, \dots, C; t = 1, \dots, T \quad (14)$$

O volume de um tanque num certo intervalo de tempo deve ser igual ao volume inicial, mais o volume recebido do processo menos o volume enviado a clientes até este intervalo de tempo.



$$Vol_{iq,t} = Volini_{iq} + \sum_{\substack{t' \in CT \\ t' \leq t}} \left(QR_{iq,t'} - \sum_{\substack{c \in CC \\ t' \leq t}} QE_{iq,c,t'} \right) \quad \forall tq = 1, \dots, TQ; t = 1, \dots, T \quad (15)$$

O volume dos tanques deve estar sempre entre o volume mínimo e o máximo determinados.

$$Vol\ min_{iq} \leq Vol_{iq,t} \leq Vol\ max_{iq} \quad \forall tq = 1, \dots, TQ; t = 1, \dots, T \quad (16)$$

A demanda dos clientes deverá ser cumprida em sua totalidade.

$$\sum_{tq \in CTQ} \sum_{t \in CT} QE_{tq,c,t} = DEM_c \quad \forall c = 1, \dots, C \quad (17)$$

A transição ocorre quando num intervalo de tempo um certo tanque está recebendo do processo e no intervalo seguinte um outro tanque é que recebe.

$$TR_{iq,tq',t} \leq Rec_{iq,t-1} \quad \forall t = 2, \dots, T; tq, tq' = 1, \dots, TQ, tq \neq tq' \quad (18)$$

$$TR_{iq,tq',t} \leq Rec_{tq',t} \quad \forall t = 2, \dots, T; tq, tq' = 1, \dots, TQ, tq \neq tq' \quad (19)$$

$$TR_{iq,tq',t} \geq Rec_{iq,t-1} + Rec_{tq',t} - 1 \quad \forall t = 2, \dots, T; tq, tq' = 1, \dots, TQ, tq \neq tq' \quad (20)$$

As restrições a seguir fazem com que a variável $XI_{c,t}$ assumam valor "1", se no intervalo $(t-1)$ o cliente não está recebendo e passa a receber no intervalo t e assume valor "0" para qualquer outra situação.

$$XI_{c,t} \leq \sum_{tq \in CTQ} Env_{tq,c,t} \quad \forall t = 2, \dots, T; c = 1, \dots, C \quad (21)$$

$$XI_{c,t} \leq 1 - \sum_{tq \in CTQ} Env_{tq,c,t-1} \quad \forall t = 2, \dots, T; c = 1, \dots, C \quad (22)$$

$$XI_{c,t} \geq \sum_{tq \in CTQ} Env_{tq,c,t} - \sum_{tq \in CTQ} Env_{tq,c,t-1} \quad \forall t = 2, \dots, T; c = 1, \dots, C \quad (23)$$

As restrições, a seguir, fazem com que a variável $XF_{c,t}$ assumam valor "1", se no intervalo $(t-1)$ o cliente está recebendo e passa a não receber no intervalo t e assume valor "0" para qualquer outra situação.

$$XF_{c,t} \leq \sum_{tq \in CTQ} Env_{tq,c,t-1} \quad \forall t = 2, \dots, T; c = 1, \dots, C \quad (24)$$

$$XF_{c,t} \leq 1 - \sum_{tq \in CTQ} Env_{tq,c,t} \quad \forall t = 2, \dots, T; c = 1, \dots, C \quad (25)$$

$$XF_{c,t} \geq \sum_{tq \in CTQ} Env_{tq,c,t-1} - \sum_{tq \in CTQ} Env_{tq,c,t} \quad \forall t = 2, \dots, T; c = 1, \dots, C \quad (26)$$

As restrições (21) a (26) não contemplam o primeiro intervalo de tempo. Se no início do período algum tanque estiver bombeando, este será considerado o início do bombeio. Nenhum bombeio poderá encerrar no primeiro intervalo.

$$XI_{c,1} = \sum_{tq \in CTQ} Env_{tq,c,1} \quad \forall c = 1, \dots, C \quad (27)$$

$$XF_{c,1} = 0 \quad \forall c = 1, \dots, C \quad (28)$$

5. MODELAGEM E METODOLOGIA DO AGEEH – PLIM

O modelo utilizado para o problema é gerado a partir do modelo de PLIM com discretização uniforme do tempo (Seção 4). Para a resolução, este modelo em PLIM é modificado para ser resolvido por PL. Esta modificação inclui a retirada de algumas restrições e variáveis binárias, e relaxação linear das variáveis binárias e inteiras restantes. Algumas das variáveis retiradas do modelo são tratadas por um Algoritmo Genético de Estado Estacionário Híbrido (AGEEH) e outras por meio de um procedimento. Os valores encontrados são inseridos como dados no modelo em PL, que é então resolvido pelo aplicativo LINGO 8.0. O valor da função objetivo resultante é utilizado como valor da função de aptidão para o AGEEH. Depois de terminado o processo do AGEEH, o resultado encontrado é inserido como dados no modelo em PLIM. Este modelo é então resolvido pelo LINGO 8.0 usando a técnica de *branch and bound*. A Figura 2 mostra um fluxograma simplificado desta modelagem.

As variáveis $Rectq,t$, $tq = 1, \dots, TQ$, $t = 1, \dots, T$ foram tratadas pelo AGEEH. Como esta variável é binária, os indivíduos da população do AGEEH foram formados gerando-se aleatoriamente vetores compostos pelos dígitos binários "0" e "1". Um procedimento foi desenvolvido para gerar estes cromossomos de forma a satisfazer a restrição (7) que impõe que haverá sempre um, e somente um tanque recebendo da produção em cada intervalo de tempo. Logo, para a construção de um indivíduo deve-se ter em cada intervalo de tempo $t \in \{1, \dots, T\}$, um e somente um dos tanques $tq \in \{1, \dots, TQ\}$ com valor da variável binária igual a "1". Como as variáveis binárias $TRtq,tq',t$ com $tq, tq' \in \{1, \dots, TQ\}$ e $t \in \{1, \dots, T\}$ dependem exclusivamente das variáveis $Rectq,t$, um procedimento foi implementado para encontrá-las, satisfazendo as restrições (18), (19) e (20). O procedimento verifica, para todas as combinações duas a duas dos tanques, se as variáveis $Rectq,t$ no intervalo de tempo t e $(t-1)$ são iguais a "1". Em caso afirmativo, a variável $TRtq,tq',t$ assumirá valor "1" no intervalo t . Para qualquer outro caso, a variável assume valor "0". Esta verificação é feita para os intervalos de tempo $t \in \{2, \dots, T\}$.

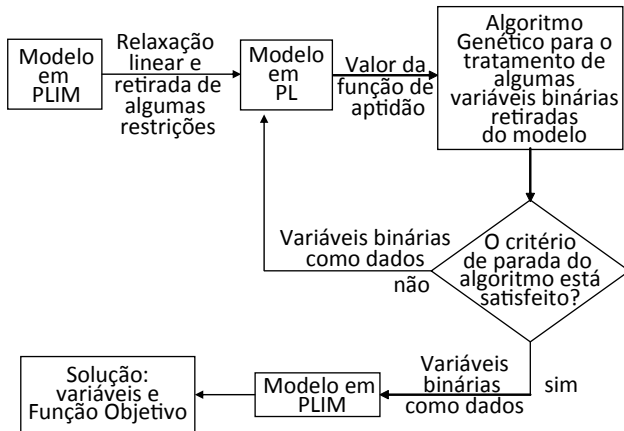


Figura 2. Fluxograma de PLIM – Computação Evolucionária
Fonte: Barboza (2005)

Para encontrar o valor da função de aptidão, seguem-se as seguintes etapas: extração dos dados do cromossomo (variáveis $Rectq,t$, $tq = 1, \dots, TQ$, $t = 1, \dots, T$); aplicação do procedimento para gerar as variáveis $TRtq,tq',t$; inserção dos dados obtidos no modelo do LINGO 8.0 em PL; resolução deste modelo usando o Método Simplex e obtenção do valor da função objetivo para ser utilizado como valor da função aptidão.

O AGEEH implementado mostrou a necessidade da inclusão de uma busca local mais agressiva, após um certo número de iterações, com o objetivo de sair de mínimos locais. Esta busca ocorre após um número de iterações determinado, considerado como parâmetro variável do AGEEH. Esta busca consiste na seleção de um indivíduo da população e troca exaustiva dos valores da variável $Rectq,t$. O indivíduo gerado pela mudança que obtiver melhor resultado para a função de aptidão será o inserido na população.

Os operadores de seleção, cruzamento e mutação utilizados no AGEEH são:

Operador de seleção: proposto por Mayerle (1994) através da seguinte fórmula:

$$Select(R) = \left\{ r_j \in R / j = m + 1 - \left\lceil \frac{-1 + \sqrt{1 + 4 \cdot rnd \cdot (m^2 + m)}}{2} \right\rceil \right\} \quad (29)$$

onde: R é o conjunto dos m cromossomos; r_j é o j -ésimo cromossomo; $rnd \in [0,1)$ é um número aleatório uniformemente distribuído; $\lceil x \rceil$ é a função que retorna o menor inteiro maior que x .

Cruzamento: de dois pontos. Este cruzamento foi especializado para a configuração dos cromossomos em

questão, de forma a gerar novos indivíduos viáveis face à restrição (7).

Mutação: mutação simples. Um intervalo de tempo é escolhido aleatoriamente para que sua cadeia binária seja modificada. Sorteia-se então um tanque, com a restrição de que o seu valor deve ser "0". Troca-se este valor para "1" e o tanque que estava com valor "1" passa a ser "0".

Foram aplicadas as taxas adaptativas de Srinivas e Patnaik (1994) para a probabilidade de recombinação (cruzamento) p_c e a probabilidade de mutação p_m , com o objetivo de se evitar uma convergência prematura do AG para um ótimo local. Para problemas de minimização, considera-se f' como o menor valor de aptidão entre os indivíduos escolhidos para pais, f_{min} , o menor valor de aptidão e f_{max} o maior valor de aptidão, respectivamente, entre todos os indivíduos da população. As fórmulas (30) e (31) calculam as probabilidades de cruzamento e mutação, respectivamente.

$$p_c = \left[(k_2 - k_1) \cdot f' + f_{max} \cdot k_1 - f_{min} \cdot k_2 \right] / (f_{max} - f_{min}) \quad (30)$$

$$p_m = \left[(k_4 - k_3) \cdot f' + f_{max} \cdot k_3 - f_{min} \cdot k_4 \right] / (f_{max} - f_{min}) \quad (31)$$

Os parâmetros k_1 , k_2 , k_3 e k_4 são números reais no intervalo $[0,1]$ com $k_1 < k_2$ e $k_3 < k_4$.

O pseudo-código do algoritmo genético utilizado nesta metodologia é mostrado a seguir:

Algoritmo AGEEH

Inicialize a população P com N cromossomos

Avalie indivíduos na população P

Ordene a população P em ordem crescente pelo valor de aptidão

Repita

Se a aplicação do operador de recombinação com probabilidade p_c é verdadeira **então**

Selecione dois indivíduos em P

Aplique o operador de recombinação

Avalie os indivíduos gerados

Insira estes indivíduos em P de acordo com sua aptidão

Se a aplicação do operador de mutação com probabilidade p_m é verdadeira **então**

Selecione um indivíduo em P

Aplique o operador de mutação

Avalie o indivíduo gerado

Insira este indivíduo em P de acordo com sua aptidão

Se o número de iterações para a hibridização foi alcançado **então**

Faça busca local

Até máximo de gerações

Fim



6. IMPLEMENTAÇÃO E RESULTADOS

Os resultados são apresentados para os modelos com representação de tempo discreto com a implementação das metodologias: algoritmo *Branch and Bound* com o uso do aplicativo LINGO 8.0 e AGEEH – PLIM. O aplicativo computacional LINGO 8.0 utilizado foi cedido pelo Programa de Pós-Graduação em Métodos Numéricos em Engenharia da Universidade Federal do Paraná. Os programas computacionais implementados foram desenvolvidos com o aplicativo Visual Basic 6.0 (VB 6.0) na versão Enterprise Edition. Na implementação da abordagem AGEEH – PLIM foram utilizados dois módulos em VB, fornecidos pela LINDO Systems Inc., para integração entre VB 6.0 e LINGO 8.0. Para o modelo descrito, foram efetuados dois grupos de testes distintos para a obtenção de resultados: resolução do modelo em PLIM utilizando o aplicativo LINGO 8.0 e aplicação da metodologia AGEEH – PLIM.

Os dados utilizados para estes grupos de testes foram:

- Número de tanques: $TQ = 4$;
- Número de clientes: $C = 2$;
- Número de intervalos de tempo: $\tau = 24$;
- Custo de bombeio para os clientes: $CB1 = 0,15$ unidades monetárias/mil m3 e $CB2 = 0,2$ unidades monetárias /mil m3 de produto enviado;
- Custo de armazenamento nos tanques: $CAtq = 0,01$ unidades monetárias/mil m3 de produto estocado em qualquer tanque;
- Custo de troca: $CTRtq = 2,00$ unidades monetárias para cada troca efetuada entre tanques na operação de recebimento;
- Vazão mínima de recebimento pelos tanques: $QR_{min} = 0,6$ mil m3/h;
- Vazão máxima de recebimento pelos tanques: $QR_{max} = 0,7$ mil m3/h;
- Vazão mínima de envio ao cliente c: $QEmin1 = 0,5$ mil m3/h e $QEmin2 = 0,9$ mil m3/h;
- Vazão máxima de envio ao cliente c: $QEmax1 = 0,6$ mil m3/h e $QEmax2 = 1$ mil m3/h;
- Volume mínimo permitido nos tanques: $Volmintq = 1$ mil m3 para todos os tanques;

- Volume máximo permitido nos tanques: $Volmaxtq = 16$ mil m3 para todos os tanques;
- Volume no tanque tq no início do processo: $Volini1 = 7$ mil m3, $Volini2 = 1$ mil m3, $Volini3 = 1$ mil m3 e $Volini4 = 1$ mil m3;
- Demanda de diesel para cada cliente c: $DEM1 = 5$ mil m3 e $DEM2 = 6$ mil m3;

6.1 Resultados para a modelagem PLIM com representação de tempo discreto

Os dados acima aplicados ao modelo PLIM resultaram em 390 variáveis contínuas, 766 inteiras e um total de 2.149 restrições. O modelo foi executado 25 vezes pelo LINGO 8.0, até encontrar-se o valor ótimo igual a 6,285 unidades monetárias. A Tabela 1 mostra os valores para a média e o desvio padrão para o número de iterações efetuadas pelo LINGO 8.0 e o tempo despendido no processo.

Tabela 1. Resultados para o modelo PLIM

Estatística	Nº de iterações	Tempo computacional (s)
Média	743124,8	1359,4
Desvio padrão	273218,7	480,5

Fonte: Barboza (2005)

6.2 Resultados para a metodologia AGEEH – PLIM

Para a obtenção dos resultados para o modelo AGEEH-PLIM foram efetuadas 135 rodadas de testes, variando-se os parâmetros: tamanho da população e o número necessário de iterações do AGEEH para realizar a busca local (hibridização). Os dados referentes ao número de tanques, clientes e intervalos de tempo devem estar em conformidade com o modelo em PL do LINGO 8.0, interligado ao programa. Em cada teste foram salvos todos os indivíduos com valor da função de aptidão menor que 6,9, sendo este valor 10% maior que o valor ótimo (resolução em PL) de 6,266523 unidades monetárias.

Nas rodadas de testes foram combinados os seguintes valores para os parâmetros:

- Tamanho da população: 40, 45 e 50;
- Número de iterações para a busca local (hibridização): 100, 115 e 125.
- Para as probabilidades adaptativas foram utilizados os valores de $k1 = 0,5$, $k2 = 1$, $k3 = 0,1$ e $k4 = 0,5$.



Na Tabela 2 são apresentadas estatísticas descritivas do número de iterações do AGEEH, do número de iterações do LINGO 8.0, do tempo computacional e da função de aptidão obtidos a partir dos resultados obtidos, separados por intervalos para o valor da função de aptidão f .

Tabela 2. Estatísticas descritivas dos testes da metodologia AGEEH – PLIM (Média \pm desvio padrão)

Intervalos da função f	Nº de iterações do AGEEH	Nº de iterações do LINGO	Tempo computacional (s)	Valor da função de aptidão
$f = 6,285$	908 \pm 502	1229718 \pm 623219	1219 \pm 621	6,285 \pm 0
$6,285 < f \leq 6,47$	679 \pm 261	928961 \pm 347005	921 \pm 352	6,353 \pm 0,056
$6,47 < f \leq 6,66$	634 \pm 241	860666 \pm 307462	856 \pm 313	6,530 \pm 0,025
$6,66 < f \leq 6,9$	601 \pm 271	837905 \pm 365698	834 \pm 372	6,790 \pm 0,071

Fonte: Barboza (2005)

O resultado para a média geral do tempo computacional expresso em minutos foi de 16 minutos e 59 segundos com desvio padrão de 8 minutos e 19 segundos. Através das médias do número de iterações do LINGO e do tempo computacional do programa desenvolvido, calculou-se o desempenho de aproximadamente 1.006 iterações do LINGO por segundo.

A Tabela 3 mostra as médias do número de iterações do AG, do número de iterações do LINGO e do tempo computacional para cada combinação dos parâmetros tamanho da população e número de iterações para hibridização, quando o resultado ótimo de 6,285 foi obtido. O melhor desempenho pode ser observado para tamanho da população igual a 45 e número de iterações para hibridização igual a 125.

Tabela 3. Média para o resultado ótimo da metodologia AG – PLIM

Tamanho População	Nº de Iterações para Hibridização	Nº de Iterações do AG	Nº de Iterações do LINGO	Tempo Computacional (segundos)
40	100	1211	1626152	1609,0
40	115	971	1306102	1291,7
40	125	940	1219458	1207,2
45	100	786	1100281	1091,5
45	115	845	1152047	1185,3
45	125	732	993706	978,6
50	100	735	1056174	1032,9
50	115	1079	1440459	1428,9
50	125	873	1173083	1149,5
Médias	-	908	1229718	1219,4

Fonte: Barboza (2005)

A Figura 3 ilustra o comportamento do número de iterações do AGEEH em relação ao valor da função de aptidão.

Para ilustrar o número de iterações do LINGO 8.0, de acordo com o valor da função de aptidão, construiu-se o gráfico da Figura 4.

Finalmente, a Figura 5 ilustra o tempo computacional (em segundos) em relação ao valor de aptidão.

Observando-se as Figuras 3, 4 e 5, percebe-se uma semelhança no comportamento dos resultados para número de iterações do AGEEH, número de iterações do LINGO e tempo computacional em relação ao valor da função de aptidão. Ainda, estes gráficos mostram uma acentuada variabilidade e são observados pontos de máximo local, próximos de 6,76; 6,55 e 6,28. Nestes pontos, quando o algoritmo alcançou mínimos locais, um maior número de iterações do AGEEH, maior número

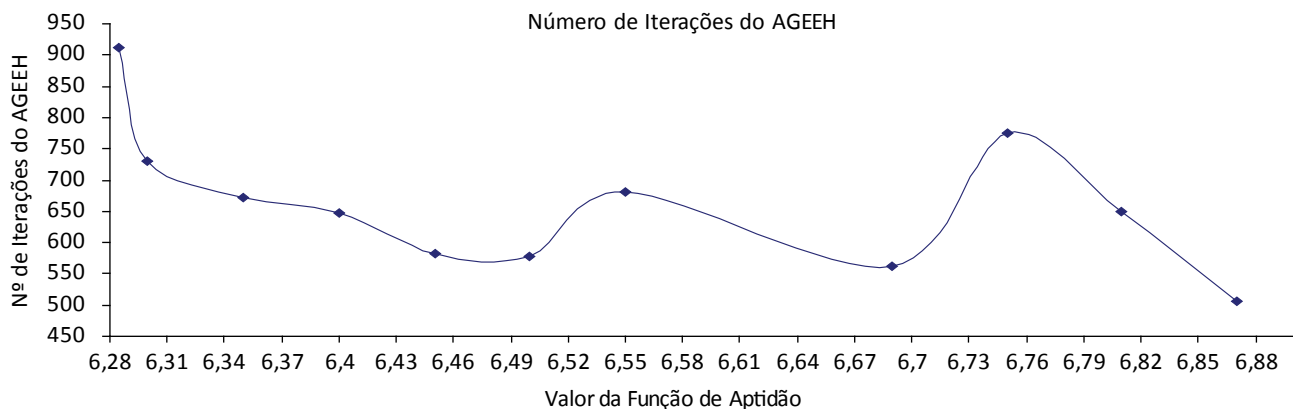


Figura 3. Número de iterações do AGEEH de acordo com o valor da função de aptidão

Fonte: Barboza (2005)

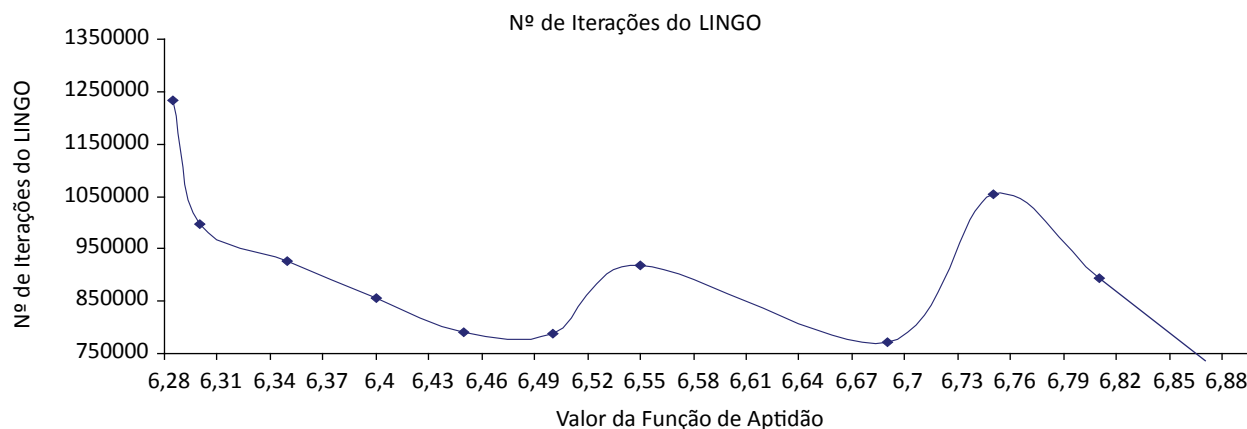


Figura 4. Número de iterações do LINGO de acordo com o valor da função de aptidão

Fonte: Barboza (2005)

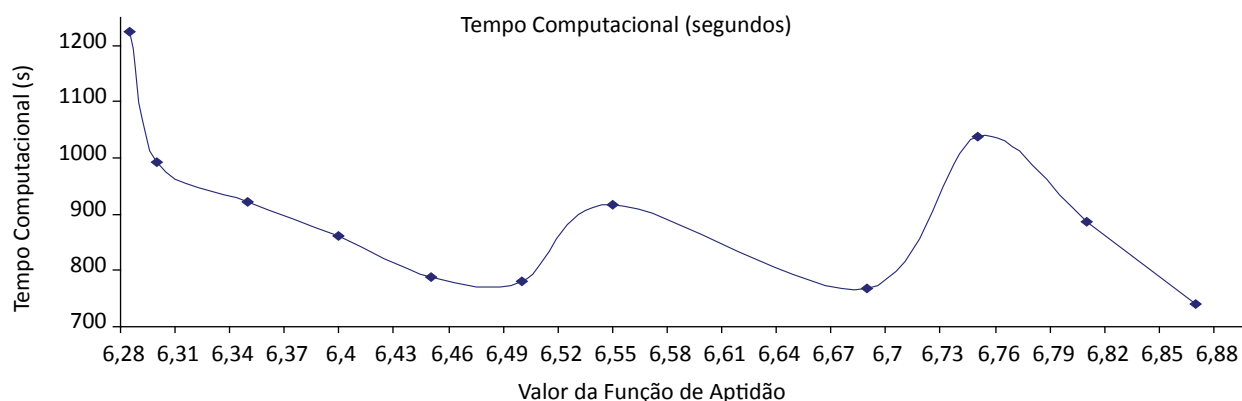


Figura 5. Tempo computacional de acordo com o valor da função de aptidão

Fonte: Barboza (2005)

ro de iterações do LINGO e maior tempo computacional foram despendidos, gerando um esforço computacional maior para prosseguir a busca.

Uma análise foi efetuada com o objetivo de investigar a influência do tamanho da população e o número de iterações para hibridização sobre os resultados do número de iterações do AG, número de iterações do LINGO e tempo computacional quando se atinge o resultado ótimo.

Inicialmente, estimou-se os coeficientes de correlação de Pearson entre as variáveis número de iterações do AG, número de iterações do LINGO e tempo computacional, duas a duas, e testou-se a hipótese nula de ausência de correlação *versus* a hipótese alternativa de existência de correlação. Os resultados indicaram significância estatística no nível de 0,05. Os coeficientes de correlação *r* evidenciaram uma forte associação entre número de iterações do AG e número de iterações do LINGO ($r = 0,9913$), entre número de iterações do AG e tempo computacional ($r = 0,9875$) e entre número de iterações do LINGO e tempo computacional ($r = 0,9978$).

Sendo assim, a análise foi efetuada somente para o número de iterações do AG.

Considerando-se os níveis 40, 45 e 50 para o fator tamanho da população e os níveis 100, 115 e 125 para o fator número de iterações para hibridização, efetuou-se uma análise de variância para avaliar a influência dos fatores sobre o número médio de iterações do AG para atingir o resultado ótimo. Inicialmente, testou-se a hipótese nula de inexistência de interação entre o tamanho da população e o número de iterações para hibridização. O resultado indicou que não há esta interação ($p = 0,2307$). Em seguida, ao testar-se a hipótese nula de médias iguais para os três níveis de iterações para hibridização, verificou-se que não existe diferença significativa entre eles ($p = 0,5992$). Da mesma forma, para os três níveis da população, não houve diferença significativa quanto ao número de iterações do AG ($p = 0,0553$). Entretanto, para o nível de significância de 0,05, pode-se afirmar que, para o tamanho da população há uma tendência à diferença estatisticamente significativa entre os níveis 40, 45 e 50 da população, em relação ao número médio de iterações do AG para atingir o resultado ótimo.



O comportamento semelhante do número de iterações do AG, número de iterações do LINGO e tempo computacional, para as combinações de tamanho da população e número de iterações para hibridização pode ser observado nos gráficos das Figuras 6, 7 e 8, construídos a partir das Tabelas 4, 5 e 6, das médias obtidas com todos os resultados para cada uma destas variáveis. Os gráficos confirmam o resultado do coeficiente de correlação encontrado. O melhor desempenho pode ser observado para tamanho da população igual a 45 e número de iterações para hibridização igual a 100 (Tabelas 4, 5 e 6 e Figuras 6, 7 e 8).

Tabela 4. Médias para o número de iterações do AGEEH de acordo com o número de iterações para hibridização e tamanho da população

Número de iterações para hibridização			
Tamanho da População	100	115	125
40	834	805	710
45	640	746	696
50	661	797	901

Fonte: Barboza (2005)

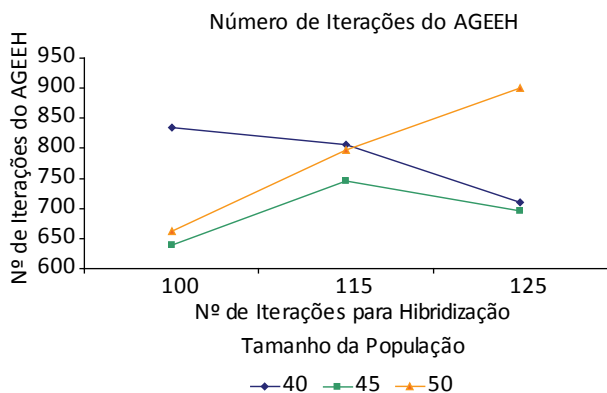


Figura 6. Número de iterações do AGEEH de acordo com o número de iterações para hibridização e tamanho da população
Fonte: Barboza (2005)

Tabela 5. Médias para o número de iterações do LINGO de acordo com o número de iterações para hibridização e tamanho da população

Número de iterações para hibridização			
Tamanho da População	100	115	125
40	1142776	1094358	941442
45	907578	1013426	944055
50	941944	1085100	1200496

Fonte: Barboza (2005)

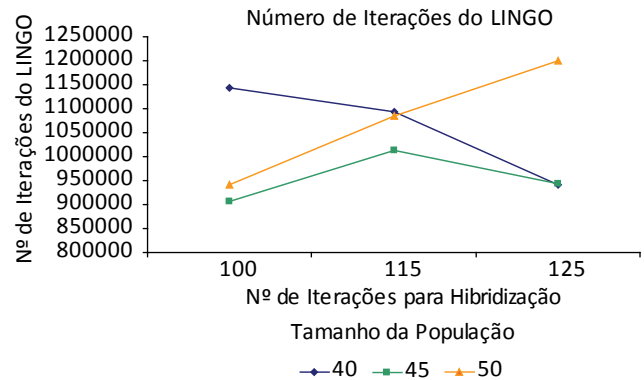


Figura 7. Número de iterações do LINGO de acordo com o número de iterações para hibridização e tamanho da população

Fonte: Barboza (2005)

Número de iterações para hibridização			
Tamanho da População	100	115	125
40	1134	1079	932
45	901	1039	939
50	918	1069	1184

Tabela 6. Médias para o tempo computacional de acordo com o número de iterações para hibridização e tamanho da população
Fonte: Barboza (2005)

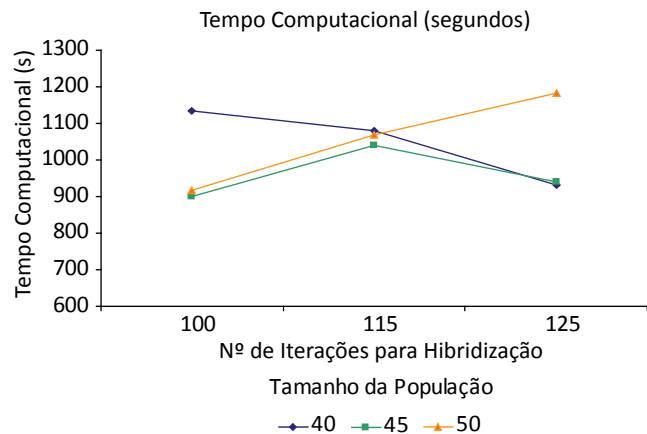


Figura 8. Tempo computacional de acordo com o número de iterações para hibridização e tamanho da população

Fonte: Barboza (2005)

7. ANÁLISE DOS RESULTADOS E CONCLUSÕES

Neste trabalho foram abordadas aplicações para otimização da programação de produção (*scheduling*) em uma refinaria de petróleo. O problema tratado foi o de transferência e estocagem de diesel, que inclui o envio de diesel aos tanques e posterior distribuição deste produto a clientes. O desenvolvimento e implementação de novas



metodologias visaram à obtenção de soluções eficientes que priorizassem o retorno financeiro, de forma a tornar economicamente viáveis seus resultados práticos.

O processo de transferência e estocagem de óleo diesel da refinaria de petróleo estudado foi analisado de forma a obter-se um modelo com representação discreta no tempo, com características lineares. Por ser um modelo em PLIM com discretização do tempo, um problema que surge na resolução através do aplicativo LINGO 8.0 é que a redução do tamanho do intervalo de tempo implica em um número de intervalos (T) maior e conseqüentemente um número maior de variáveis binárias, fazendo com que o tempo computacional aumente. Este acréscimo no tempo se deve ao fato do problema ser do tipo combinatorial. Neste caso, aumentando-se o número de variáveis inteiras, aumenta-se o número de combinações possíveis para o vetor solução, o que pode tornar o problema inviável, em termos de tempo computacional, na resolução com o algoritmo *branch and bound*.

Para o modelo em PLIM, com a aplicação do LINGO 8.0, foram efetuados 25 testes que resultaram no valor ótimo de 6,285 unidades monetárias para a função objetivo. A Tabela 4 mostra os resultados obtidos para o número de iterações e o tempo computacional (segundos). A média para o número de iterações foi 743125, com um desvio padrão de 273218,69. Para o tempo, a média foi de 22 minutos e 39 segundos, com desvio padrão de 8 minutos e 52 segundos.

A Tabela 7 exhibe as médias das iterações do LINGO 8.0 e do tempo computacional das metodologias PLIM e AGEEH-PLIM de acordo com os intervalos da função de aptidão (f). Com base nestes dados foram efetuadas comparações entre as duas metodologias em relação ao desempenho.

Tabela 7. Resumo das médias para as metodologias PLIM e AGEEH – PLIM

Intervalos da função f	Metodologia	Iterações do LINGO	Tempo Computacional (segundos)
$f = 6,285$	PLIM	743125	1359,4
(resultado ótimo)	AGEEH – PLIM	1229718	1219
$6,285 < f \leq 6,47$	AGEEH – PLIM	988961	921
$6,47 < f \leq 6,66$	AGEEH – PLIM	860666	856
$6,66 < f \leq 6,9$	AGEEH – PLIM	837905	834

Fonte: Barboza (2005)

Para o resultado ótimo na metodologia AGEEH – PLIM, tem-se um número de iterações 65,5% maior, com tempo computacional 10,3% menor do que na metodologia PLIM. Se $6,285 < f \leq 6,47$, a metodologia AGEEH – PLIM resulta num número de iterações 33,1% maior, com tempo computacional 32,2% menor. Se $6,47 < f \leq 6,66$, a metodologia AGEEH – PLIM resulta num número de iterações 15,8% maior, com

tempo computacional 37% menor. Ainda, se $6,66 < f \leq 6,9$, a metodologia AGEEH – PLIM resulta num número de iterações 12,8% maior, com tempo computacional 38,6% menor. Em resumo, para a metodologia AGEEH – PLIM, comparada com a metodologia PLIM, quanto mais afastado do valor ótimo, menor é a perda em relação ao número de iterações e maior é o ganho em relação ao tempo computacional. De modo geral, observa-se que o desempenho da metodologia AGEEH – PLIM, com relação ao tempo computacional, foi melhor. Conclui-se, portanto, que na aplicação da metodologia AGEEH – PLIM, para a instância utilizada, deve-se ponderar a vantagem de um tempo computacional menor para a obtenção de um bom resultado, que pode não ser o ótimo.

Os resultados obtidos para a metodologia AGEEH - PLIM evidenciam que o desempenho depende dos parâmetros utilizados. A partir de análise, pode-se afirmar que o número de iterações do AG influencia mais no desempenho do que o número de iterações para hibridização. Os parâmetros utilizados neste trabalho foram escolhidos com base em rodadas de testes prévias.

Deve-se destacar que o uso de probabilidades de recombinação e mutação adaptativas faz com que quanto mais afastado um indivíduo está do primeiro da população, maiores estas probabilidades. Desta forma, este indivíduo acaba sendo induzido a fazer as operações de recombinação e de mutação. Por outro lado, os primeiros indivíduos da população terão sempre probabilidades menores para que a chance de realização destas operações fique reduzida com o objetivo de preservar estes indivíduos.

Para os testes realizados neste trabalho, analisando-se os resultados da Tabela 3, observa-se que o melhor desempenho em termos de número de iterações do AG, número de iterações do LINGO e tempo computacional ao atingir o resultado ótimo, foi observado para tamanho de população igual a 45 e número de iterações para hibridização igual a 125.

A variabilidade dos valores da função de aptidão analisada nas Figuras 3, 4 e 5 apresenta um comportamento semelhante quando se observa o número de iterações do AGEEH, o número de iterações do LINGO e o tempo computacional. À medida que o valor da função se aproxima do valor ótimo (6,285), esta variabilidade diminui e observa-se um crescimento acentuado que gera um custo em relação ao desempenho do algoritmo AG. Isto ocorre porque o algoritmo atinge um mínimo local e acaba por exigir um esforço adicional para sair deste mínimo e se direcionar para o resultado ótimo.

Finalmente, conclui-se que a metodologia AGEEH-PLIM pode contribuir significativamente para o problema de transferência e estocagem de produtos em refinarias de petróleo. Os resultados obtidos mostraram que a metodologia



é adequada, levando-se em conta a redução do tempo computacional, sem grande perda na qualidade da solução.

REFERÊNCIAS

- Almeida, L. F., Pacheco, M. A., Vellasco, M. M. (2003), "Algoritmos Evolucionários na Otimização de Alternativas para o Desenvolvimento de Campos de Petróleo", apresentado no XXIII ENEGEP 2003: Encontro Nacional de Engenharia de Produção, Ouro Preto, MG, 21-24 de Outubro, 2003.
- Barboza, A. O. (2005), Simulação e técnicas da computação evolucionária aplicadas a problemas de programação linear inteira mista. Tese de Doutorado em Engenharia Elétrica e Informática Industrial, Universidade Tecnológica Federal do Paraná, Curitiba, PR.
- Carrano, E. G., Takahashi, R. H. C., Cardoso, E. P., Saldanha, R. R., NETO, O.M. (2005), "Optimal Substation Location and Energy Distribution Network Design using a Hybrid GA-BFGS Algorithm", *Transmission and Distribution*, Vol. 152, No 6, pp. 919-926.
- Dantzig, G. B. (1963), *Linear Programming and Extensions*, Princeton University Press, Princenton, New Jersey, USA.
- Duan, Z. et Wang, L. (2011), "Heuristic algorithms for the inverse mixed integer linear programming problem", *Journal of Global Optimization*, Vol. 51, No. 3, p.463-471.
- Floudas, C. A. et Lin, X. (2005), "Mixed Integer Linear Programming in Process Scheduling: Modeling, Algorithms, and Applications", *Annals of Operations Research*, Vol. 139, No. 1, pp. 131-162.
- Goldberg, D. E. (1989), *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Publishing Company, INC. USA.
- Holland, J. H. (1975), *Adaptation in Natural and Artificial Systems*, 2 ed., The University of Michigan Press, MI.
- ILOG CPLEX 8.0. (2002), *User's Manual ILOG SA*, Gentilly, France.
- Jia, Z. et Ierapetritou, M. (2003), "Mixed-Integer Linear Programming Model for Gasoline Blending and Distribution Scheduling", *Industrial & Engineering Chemistry Research*, Vol. 42, No. 4, p. 825-835.
- Joly, M., Moro, L. F. L., Pinto, J. M. (2002), "Planning and scheduling for Petroleum Refineries using mathematical programming", *Brazilian Journal of Chemical Engineering* Vol. 19, No. 02, pp. 207-228.
- LINDO Systems Inc. (2002), *LINDO Systems Client Models*.
- LINDO Systems, Inc. (2002), "Lingo user's Manual", "LINDO Systems, Chigago, IL.
- Liu, G., Zhang, J., Gao, R., Shang, Y. (2009), "An Improved Parallel Adaptive Genetic Algorithm Based on Pareto Front for Multi-Objective Problems", apresentado no 2nd International Symposium on Knowledge Acquisition and Modeling, Wuban, China, 30 de Novembro – 1 de Dezembro, 2009.
- Mayerle, S. F. (1994), "Um algoritmo genético para solução do problema do caixeiro viajante", Departamento de Engenharia de Produção e sistemas, Universidade Federal de Santa Catarina, SC.
- Morabito, R et Abuabara, A. (2008), "Modelos de programação inteira mista para o planejamento do corte unidimensional de tubos metálicos na indústria aeronáutica agrícola", *Gestão & Produção*, Vol. 15, No. 3, pp. 605-617.
- Moro, L. F. L. (2000), Técnicas de otimização mista-inteira para o planejamento e programação de produção em refinarias de petróleo. Tese de Doutorado, Universidade de São Paulo, SP.
- Pinto, J. M. (2000), *Planejamento e programação de operações de produção e distribuição em refinarias de petróleo*. Tese de Livre Docência, Universidade de São Paulo, SP.
- Srinivas, M. et Patnaik, L. M. (1994), "Adaptative Probabilities of Crossover and Mutation in Genetic Algorithms. IEE Transactions on Systems", *Man an Cybernetics*, Vol. 24, No. 4, pp. 656-666.
- Subbaraj, P., Rengaraj, R., Salivahanan, S. (2011), "Enhancement of Self- Adaptive Real-Coded Genetic Algorithm using Taguchi Method for Economic Dispatch Problem", *Applied Soft Computing*, Vol.11, pp. 83-92.
- Taha, H. A. (1975), *Integer Programming. Theory, Applications, and Computations*. Academic Press, Inc. Orlando, Florida, United States of America.
- Tanomaru, J. (1995), *Motivação, Fundamentos e Aplicações de Algoritmos Genéticos*, apresentado no II Congresso Brasileiro de Redes Neurais e III Escola de Redes Neurais, Curitiba, PR, 29 Outubro-01 Novembro, 1995.
- Thunberg, J. et Ögren, P. (2011), "A Mixed Integer Linear Programming approach to pursuit evasion problems with optional connectivity constraints", *Autonomous Robots*, Vol. 31 No.4, p. 333-343.
- Wang, L. et Tang, D. B. (2011), "An improved adaptive genetic algorithm based on hormone modulation mechanism for job-shop scheduling problem", *Expert Systems with Applications*, Vol. 38, pp. 7243-7250.
- Wu, N., Zhou, M., and Chu, F. (2005), "Short-term Scheduling for Refinery Process: Bridging the Gap between Theory and Applications", *International Journal of Intelligent Control and Systems*, Vol. 10, No. 2, pp. 162-174.
- Zhang, J.; Shang, Y. (2009), "An Improved Multi-Objective Adaptive Niche Genetic Algorithm Based On Pareto Front", apresentado no IACC 2009: International Advance Computing Conference, Patiala, 6-7 March, 2009.
- Zionts, S. (1974), *Linear and Integer Programming*. Prentice-Hall, Inc, Englewood Cliffs, New Jersey, USA.