

UM MODELO HÍBRIDO (CLP-MILP) PARA *SCHEDULING* DE OPERAÇÕES EM POLIDUTOS

Leandro Magatão

Lúcia Valéria Ramos de Arruda*

Flávio Neves-Jr

Progr. Pós-Grad. Eng. Elétrica e Inform. Industrial (CPGEI)

Universidade Tecnológica Federal do Paraná (UTFPR)

Curitiba – PR

magatao@utfpr.edu.br

arruda@cpgei.cefetpr.br

neves@utfpr.edu.br

* *Corresponding author* / autor para quem as correspondências devem ser encaminhadas

Recebido em 09/2007; aceito em 07/2008 após 1 revisão

Received September 2007; accepted July 2008 after one revision

Resumo

A eficácia na transferência de derivados de petróleo através de dutos motiva a execução deste trabalho. O objetivo principal é a modelagem do *scheduling* de um poliduto, isto é, um sistema de dutos que transporta diferentes derivados de petróleo. O poliduto em estudo com 93,5 km de extensão conecta uma refinaria a um terminal portuário. Foi desenvolvido um modelo de otimização baseado na união de *Constraint Logic Programming* (CLP) e *Mixed Integer Linear Programming* (MILP). O modelo utiliza uma abordagem de decomposição do problema, com representação temporal contínua e calcula janelas de tempo (restrições temporais) que devem ser respeitadas. A abordagem híbrida CLP-MILP proporcionou a solução de cenários reais em tempo computacional da ordem de segundos. A resolução computacional do modelo proposto evidenciou novos pontos de operação para o poliduto, proporcionando ganhos operacionais significativos. O modelo implementado configura uma ferramenta de auxílio para tomada de decisões operacionais no cenário estudado.

Palavras-chave: busca orientada por restrições (CLP); programação linear inteira mista (MILP); programação operacional de polidutos.

Abstract

This work is motivated by the need of optimization in the pipeline-oil distribution scenario. The considered problem involves the short-term scheduling of activities in a specific pipeline. The pipeline is 93.5 km in length, and it connects refinery and harbor tankfarms, conveying different types of commodities (gasoline, diesel, kerosene, etc). An optimization model was developed to determine the pipeline scheduling with improved efficiency. Such model combines Constraint Logic Programming (CLP) and Mixed Integer Linear Programming (MILP) in an integrated CLP-MILP framework. The proposed model uses decomposition strategies, continuous time representation, and intervals that represent time constraints (time windows). Real cases were solved in a reduced computational time (order of seconds). The computational results have demonstrated that the model is able to define new operational points to the pipeline, providing significant cost saving. Indeed the CLP-MILP model is an efficient tool to aid the operational decision-making within this real-world pipeline scenario.

Keywords: constraint logic programming (CLP); mixed integer linear programming (MILP); pipeline scheduling.

1. Introdução

A diminuição de custos de produção e a melhoria em produtos e serviços são objetivos comuns a diversos setores industriais. Contudo, a tomada de decisões operacionais ainda é conduzida pelo emprego de critérios experimentais. A complexidade do planejamento (*planning*) e programação (*scheduling*) da produção é contornada pela adoção de políticas operacionais conservativas, que não utilizam a capacidade máxima de operação do sistema produtivo. Motivado pela necessidade industrial, o desenvolvimento de modelos, em especial os que empregam técnicas de otimização, tem possibilitado que procedimentos operacionais complexos sejam avaliados de forma criteriosa, fazendo com que recursos críticos sejam utilizados da melhor maneira possível. Operadores e gerentes de diversas áreas industriais têm de enfrentar decisões diárias que concernem o gerenciamento de matéria prima, de pessoal, alternativas de configuração de equipamentos, seqüências de produção, etc. Via de regra, estas decisões podem ser encaradas como sendo a resolução de problemas combinatoriais de otimização. Desta forma, os problemas combinatoriais estão presentes em nosso dia a dia. Vários trabalhos encontrados na literatura abordam estes tipos de problemas e demonstram que mesmo pequenos exemplos são de difícil solução, podendo demandar horas de processamento, ou até mesmo não serem computacionalmente tratáveis (Garey & Johnson, 1979).

A área de Pesquisa Operacional (PO) tem endereçado a resolução de problemas combinatoriais a pelo menos 50 anos, e uma das abordagens de sucesso da PO são modelos desenvolvidos através de Programação Linear Inteira Mista (*Mixed Integer Linear Programming – MILP*). Muito sucintamente, uma abordagem MILP é caracterizada pela existência de um sistema algébrico de equações, destacando-se a existência de uma função objetivo (tipicamente um lucro a ser maximizado ou um custo a ser minimizado) e uma série de restrições que interligam as variáveis do modelo. Algumas destas variáveis só podem assumir valores discretos (variáveis inteiras). Contudo, um novo paradigma para a resolução de problemas combinatoriais tem evoluído dos campos da Ciência da Computação (CC) e da Inteligência Artificial (IA). Este paradigma recebe o nome de Busca Orientada por Restrições (*Constraint Logic Programming – CLP*). Basicamente, um modelo em CLP é constituído por um conjunto de variáveis, um conjunto de valores possíveis para cada variável (domínios) e um conjunto de restrições envolvendo as variáveis. CLP utiliza dois mecanismos base para reduzir o espaço de soluções válidas: propagação de restrições e redução de domínios (Rossi, 1999). Associado aos mecanismos de base, CLP também utiliza um mecanismo de busca, tipicamente um método de busca em árvore do tipo *branch-and-bound* (Land & Doig, 1960).

Desde o trabalho pioneiro de Williams (1995), que faz um levantamento das conexões entre lógica computacional e programação inteira, um crescente número de pesquisadores vem investigando a possibilidade de integrar métodos de CLP e MILP na solução de problemas combinatoriais. De fato, as técnicas de CLP e MILP possuem similaridades, são utilizadas para a resolução dos mesmos problemas e, sobretudo possuem características complementares. A sinergia entre estas duas técnicas origina uma terceira (CLP-MILP) que pode endereçar problemas combinatoriais até então intratáveis ou de difícil resolução (Hooker, 2000).

Neste trabalho, uma abordagem envolvendo CLP-MILP é desenvolvida e aplicada na elaboração de um modelo de auxílio à tomada de decisões operacionais em um cenário específico do sistema dutoviário brasileiro. Este cenário envolve o poliduto que interliga a Refinaria Presidente Getúlio Vargas, localizada em Araucária - PR, ao terminal portuário de Paranaguá, no litoral paranaense. Significativos resultados foram obtidos com o modelo, que

forneceu soluções operacionais, provando a otimalidade destas soluções, em tempos computacionais de poucos segundos para cenários reais do problema.

Na próxima seção as premissas, hipóteses e restrições descrevendo o problema real do poliduto são detalhadas. Em seguida um modelo matemático híbrido baseado numa integração das abordagens MILP e CLP é proposto e dois problemas-exemplos são apresentados. Esses problemas são solucionados utilizando o modelo proposto, implementado no *solver* CPLEX/OPL Studio (ILOG, 2002), e os resultados computacionais são analisados. Por fim, as conclusões e perspectivas futuras são apresentadas.

2. Descrição do Problema

No cenário econômico mundial, a indústria do petróleo tem vital importância, englobando atividades como a exploração, o refino e a distribuição dos produtos finais. As melhorias de eficiência e qualidade nos processos industriais ligados ao petróleo são facilmente traduzidas em ganhos financeiros, podendo também contribuir para a minimização de impactos ambientais. Inserida no contexto de distribuição dos derivados de petróleo está uma vasta rede que compreende elementos do transporte rodoviário, ferroviário, naval e dutoviário. Esta rede realiza a conexão entre terminais portuários, refinarias e centros consumidor-distribuidores. De acordo com os estudos realizados por Kennedy (1993), sistemas de transporte de óleo e gás através de dutos são eficientes, apresentam baixo custo e alto grau de segurança ambiental. A utilização eficaz do sistema dutoviário representa a adição significativa de ganhos financeiros e ambientais para a indústria do petróleo.

Recentemente, tem havido na literatura um aumento do número de trabalhos que resolvem problemas de escalonamento de polidutos através de, em sua maioria, modelos MILP. Este aumento é fruto da importância que a eficácia no transporte de petróleo e derivados a longas distâncias e baixos custos tem recebido (Souza-Filho *et al.*, 2006). Alguns destes trabalhos são destacados em seguida.

Shah (1996) estudou o problema de gerenciamento de estoques de petróleo por meio de programação matemática, tratando de um sistema composto por um porto, um oleoduto e uma refinaria. O trabalho engloba dois modelos distintos envolvendo o transporte de óleo cru, para um horizonte de tempo de aproximadamente um mês. O primeiro modelo determina a política de fornecimento da refinaria pelo oleoduto, gerenciando a alocação de tanques. O segundo, determina como os tanques do porto alimentam o oleoduto e como a descarga de navios deve ser conduzida.

Sasikumar *et al.* (1997) trataram um problema de geração do *scheduling* de bombeio de um poliduto ligando uma única fonte a múltiplos destinos. A obtenção do *scheduling* era baseada em uma busca heurística, adaptada ao cenário de estudo. Foram consideradas restrições de disponibilidade de produtos, tancagem, seqüenciamento, controle de qualidade e prazos de entrega.

Um sistema composto por uma refinaria de petróleo, um duto e diversos depósitos conectados a mercados consumidores em localizações geográficas distintas são considerados no trabalho de Rejowski & Pinto (2003). Os modelos MILP desenvolvidos incluem condições de balanço de massa, restrições de demanda e distribuição dos produtos a serem armazenados e transportados. Uma representação uniforme de tempo com formulação Big-M e relações lógicas são utilizados na obtenção da solução do problema. Com o objetivo de diminuir a carga computacional deste modelo, os autores propuseram em outro trabalho (Rejowski & Pinto, 2004) a inserção de restrições de corte, visando melhorar a eficiência da formulação.

Em Magatão *et al.* (2004) são apresentados modelos que abordam a programação do transporte dutoviário de derivados de petróleo. O trabalho trata de um poliduto que transporta diferentes derivados de petróleo e interliga os parques de tancagem de uma refinaria e de um terminal portuário. O gerenciamento das operações do poliduto envolve fatores como a disponibilidade de produtos, restrições de armazenamento, seqüenciamento de eventos, possibilidade de reversão do sentido de bombeio do duto, tratamento formal de interfaces de bombeio, além de uma variedade de restrições operacionais. Foram desenvolvidos modelos baseados em MILP com discretização uniforme do tempo para auxiliar o processo de tomada de decisões operacionais presentes nesse cenário específico. O poliduto modelado no presente artigo é o mesmo considerado nessa referência (Magatão *et al.*, 2004) e uma descrição do cenário de estudo será detalhada na seção 2.1.

Neiro & Pinto (2004) desenvolvem um modelo para uma cadeia de suprimentos petrolífera. Tal cadeia envolve refinarias, terminais e redes de polidutos. O modelo é baseado em MINLP com representação discreta do tempo. Foram aplicadas técnicas de decomposição baseadas em relaxação Lagrangeana, apresentando resultados com desempenho computacional promissor.

Cafaro & Cerdá (2004) apresentam um modelo MILP com representação contínua do tempo para o problema de escalonamento de um poliduto. O sistema completo é composto por uma refinaria, um poliduto e cinco bases de distribuição conectadas a mercados consumidores. O modelo gerencia os níveis de inventário da refinaria e os parques de tancagem em cada área de destino. Em um trabalho posterior (Cafaro & Cerdá, 2008), estes autores abordam o mesmo problema considerando um horizonte de escalonamento multiperíodo sendo que programações cíclicas são efetuadas em intervalos pré-determinados, atualizando periodicamente as operações de transporte do poliduto.

Relvas *et al.* (2006) tratam um problema de transporte de derivados de petróleo em um sistema composto por uma refinaria e um centro de distribuição, interligados através de um poliduto. O modelo MILP desenvolvido combina uma representação contínua do tempo e dos volumes envolvidos. O modelo usa uma estratégia de decomposição temporal, dividindo o horizonte de escalonamento em duas partes. Condições finais do primeiro período servem de condições iniciais para o segundo período, reduzindo o tempo total de resolução. Em um trabalho posterior (Relvas *et al.*, 2007), este modelo é aumentado, incluindo situações práticas tais que parada de dutos, vazões variáveis e outras. Além disso, uma estratégia de *re-scheduling* é apresentada.

Rejowski & Pinto (2008) apresentam um modelo MINLP baseado em tempo contínuo para endereçar o *scheduling* de um poliduto que deve suprir vários pontos consumidores descentralizados. O trabalho considera que o duto opera de forma intermitente e analisa a influência de diferentes vazões de bombeio no custo de operação do sistema. O modelo desenvolvido é comparado com uma abordagem anteriormente proposta pelos mesmos autores (Rejowski & Pinto, 2004).

A listagem de trabalhos supracitados não é uma revisão exaustiva dos esforços para melhorar a eficácia no transporte de petróleo e derivados através de dutos. Contudo, é notório que os estudos de problemas sobre transferência e estocagem no setor dutoviário ainda são escassos. Desta forma, este trabalho tem por objetivo contribuir com o estudo de um cenário específico do sistema dutoviário brasileiro, evidenciando possibilidades de melhoria através de uma abordagem baseada na união de MILP com *Constraint Logic Programming* (CLP), a qual será detalhada *a posteriori* (vide seção 3.1).

2.1 O Cenário Considerado

O problema considerado envolve a determinação das atividades operacionais de curto prazo (*short-term scheduling*) em um poliduto específico. A figura 1 ilustra o cenário em estudo. O poliduto apresenta 93,5 km de extensão (Δl), armazena um volume total de 7314 m³ (Δv) e conecta um parque de tancagem de uma refinaria ao parque de tancagem de um porto, interligando regiões com um desnível de 900 m (Δh). Existe a possibilidade de “reversão” no sentido de bombeio, isto é, os produtos podem ser bombeados tanto da refinaria para o porto (operação denominada “fluxo”), quanto do porto para a refinaria (operação denominada “refluxo”). Esta operação de reversão é bastante particular sendo formalmente tratada na formulação do modelo matemático apresentado na seção 4.

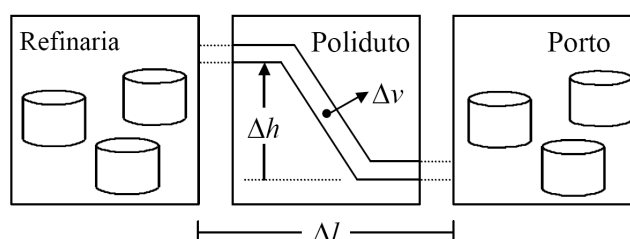


Figura 1 – Cenário em Estudo: Estrutura Simplificada.

O poliduto transporta diferentes derivados de petróleo (produtos claros) e opera sempre totalmente preenchido. Não existe nenhuma separação entre elementos que são transportados em seqüência. Desta forma, quando dois elementos são transportados em série pelo mesmo duto, existe entre eles uma área de mistura ou contaminação, a qual recebe o nome de interface. Essa mistura, que apresenta características dos dois produtos, é um terceiro produto. A interface possui um valor comercial inferior a pelo menos um dos produtos transportados em seqüência. Não são raras as situações em que interfaces de bombeio são simplesmente descartadas ou misturadas a outros produtos de menor valor agregado. A maneira mais eficiente de evitar interfaces é inviável do ponto de vista econômico, pois requer a existência de dutos independentes para cada produto. Aliado a isto, o sistema dutoviário é muito versátil, tornando o transporte seqüencial de vários produtos (poliduto) uma realidade, e as interfaces um mal necessário. Alternativas para minimizar os efeitos negativos introduzidos pelas interfaces são incessantemente procuradas, mesmo porque algumas interfaces são operacionalmente não recomendadas. No poliduto em questão, é prática operacional a utilização de um “selo” (pequeno volume de um produto) entre lotes de produtos “incompatíveis”, evitando a formação de interfaces não recomendadas. Contudo, o uso de selos eleva os custos de bombeio. A minimização do número de selos utilizados durante as operações de bombeio, ao mesmo tempo em que interfaces de bombeio operacionalmente adequadas são geradas, é um fato formalmente tratado na formulação do modelo matemático apresentado na seção 4.

Paralelamente a esta preocupação operacional com a interface, o programador do sistema recebe uma carteira de pedido de produtos, definida *a priori* pelo planejamento da produção. Ele deve realizar a programação (*scheduling*) das atividades envolvidas no bombeio dos produtos requisitados. A escolha da seqüência em que os produtos devem ser bombeados, a avaliação de interfaces, o uso de selos, a determinação do início e do final dos procedimentos

de bombeio, a determinação da vazão de transporte ideal são algumas das variáveis envolvidas no processo de obtenção das atividades operacionais de curto prazo. A diversidade de aspectos a serem considerados torna a tomada de decisões uma prática complexa, requerendo um bom domínio do sistema. O programador implementa soluções de *scheduling* válidas, porém, a obtenção da seqüência de atividades caracterizada como ideal ou ótima passa a ser um objetivo secundário. A criação de um sistema computacional capaz de auxiliar a tomada de decisões operacionais é de fundamental importância.

A partir desta necessidade, este trabalho propõe um modelo de apoio ao processo de tomada de decisões operacionais para este cenário específico. Procurou-se considerar restrições de ordem prática do sistema, de forma a se obter um modelo matemático que forneça condições de operação válidas e otimizadas para cenários reais do poliduto.

3. Modelagem do Problema

Em modelos de *scheduling* a representação da grandeza temporal (representação do tempo) assume um papel importante, determinando o detalhamento das respostas do modelo. Classicamente, a representação do tempo pode ser categorizada em duas principais abordagens: modelos que utilizam uma representação temporal discreta e modelos que utilizam uma representação temporal contínua (Kondili *et al.*, 1993).

Na abordagem discreta, o horizonte de *scheduling* é dividido em um número fixo de intervalos de igual duração (subintervalos de discretização) e todos os eventos devem ocorrer nas bordas destes subintervalos. Uma limitação de abordagens discretas é que a duração do subintervalo deve ser especificada de acordo com o máximo divisor comum da duração de todos os eventos. Por exemplo, se os tempos de processamento de diferentes tarefas variam de dez horas a quinze minutos, o menor valor deve ser adotado como o subintervalo de discretização. Conseqüentemente, o número de variáveis/restrições do modelo tende a aumentar com a diminuição do valor do subintervalo. Se os dados do problema são arredondados, o valor do subintervalo pode aumentar, mas a solução obtida pode ser muito conservativa ou infactível (Reklaitis, 1992). Em uma abordagem temporal contínua, a ocorrência de eventos reais do sistema modelado é que determina a subdivisão temporal. Não há necessidade de discretização do horizonte de *scheduling* em subintervalos de duração fixa. Desta forma, para um mesmo cenário de estudo, as representações contínuas tendem a produzir modelos menores que os criados através de representações discretas. Contudo, formulações contínuas tendem a ser mais difíceis de serem criadas (Kondili *et al.*, 1993). No presente artigo, o modelo de *scheduling* desenvolvido utiliza uma abordagem temporal contínua. Os tempos de início e final de bombeio dos produtos determinam a ocorrência de eventos no modelo.

Adicionalmente, a busca por modelos de *scheduling* que possam ser utilizados na prática, considerando uma carga computacional aceitável, é um grande desafio. Desta forma, uma abordagem de decomposição do problema é proposta neste artigo. Esta abordagem torna viável a obtenção de soluções operacionais para cenários reais do problema em tempo computacional adequado, no caso, poucos segundos. Esta subdivisão será baseada nos três elementos chave do *scheduling*: a determinação e a alocação dos recursos a serem utilizados (*assignment*), o seqüenciamento de atividades (*sequencing*) e a temporização (*timing*) do uso dos recursos pelas atividades (Reklaitis, 1992).

A figura 2 descreve a estrutura do modelo proposta para a solução do problema. Este modelo é composto por um modelo principal, um módulo para cálculo de janelas de tempo

(Pré-Processamento), um modelo auxiliar (Módulo de Tancagem) e uma Base de Dados, a qual recebe os dados de entrada e as informações vindas de outros blocos e as disponibiliza para o Modelo Principal.

O módulo de *Tancagem* é responsável pela alocação de alguns recursos (tanques) para uma dada atividade (o bombeamento dos produtos demandados). Seus dados de entrada são os requisitos de demanda, o estoque (disponibilidade) de produtos e as restrições de tancagem. Esse módulo fornece como resultado a lista de tanques a serem utilizados nos procedimentos operacionais. O modelo utilizado neste módulo é o mesmo apresentado em Magatão *et al.* (2004) e por isso não será detalhado neste artigo.

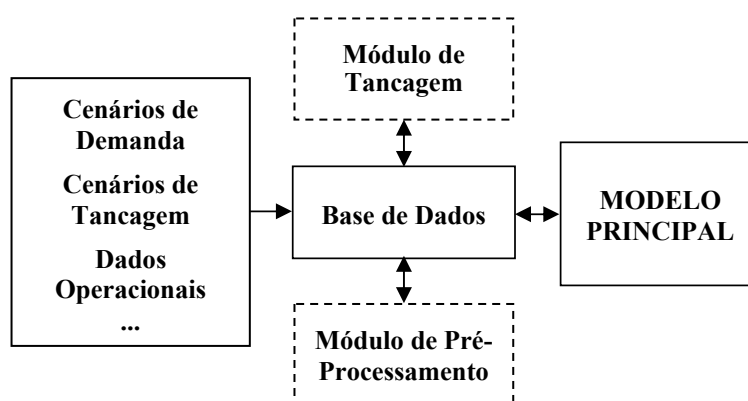


Figura 2 – Estrutura do Modelo de Otimização.

O módulo de *Pré-Processamento* leva em conta o horizonte de tempo disponível para o bombeio, as faixas de variações possíveis para o bombeio de cada produto e a demanda a ser atendida, a fim de calcular janelas de tempo (restrições temporais) que devem ser respeitadas no modelo principal.

O *Modelo Principal* determina a seqüência de bombeio de produtos e estabelece o instante de tempo inicial e final para o bombeio de cada produto (*scheduling* de saída). O Modelo Principal é formulado em uma abordagem híbrida a qual combina técnicas de CLP e MILP, conforme apresentado nas seções seguintes. O *scheduling* final é obtido resolvendo primeiro o módulo de Tancagem, em seguida o módulo de Pré-Processamento e por fim o Modelo Principal.

3.1 Abordagem híbrida CLP-MILP

Como citado anteriormente, alguns pesquisadores tem investigado a possibilidade de integrar métodos oriundos da Pesquisa Operacional com outros provenientes da Lógica Computacional, gerando técnicas híbridas para a solução de problemas combinatoriais. Vários estudos computacionais têm investigado as potencialidades destas técnicas híbridas. Estes estudos envolvem, por exemplo, processos químicos (Raman & Grossmann, 1994; Türkay & Grossmann, 1996; Hooker & Osório, 1999; Harjunkski & Grossmann, 2002), projetos de estruturas em engenharia civil (Bollapragada *et al.*, 2001), problema do caixeiro viajante (Focacci, 2000) e roteamento de veículos (Corréa *et al.*, 2004), planejamento e programação

da produção (Heipcke, 1999; Thorsteinsson, 2001; Hooker, 2007). Um levantamento detalhado de outras aplicações pode ser encontrado em Magatão (2005).

Apesar do sucesso obtido em várias aplicações, ainda é difícil dar uma recomendação geral sobre o uso, em determinados problemas, de técnicas baseadas em CLP, MILP ou uma combinação de ambas (CLP-MILP). Isto porque a abordagem combinada pode herdar algumas desvantagens de cada uma das técnicas originais (*inherited drawbacks*) e, assim, apresentar um desempenho computacional inferior, se comparada a abordagens que utilizam as técnicas originais separadamente (Heipcke, 1999; Hooker, 2002). Desta forma, estudos que evidenciem a eficácia computacional de uma abordagem combinada CLP-MILP em relação às técnicas raízes são de fato relevantes.

Segundo Hooker & Osório (1999) o sucesso da integração destas duas áreas depende ainda de dois fatores primordiais:

- o desenvolvimento de um esquema de integração robusto e largamente aceitável em ambas as comunidades (PO e CC-AI);
- o desenvolvimento de linguagens de modelagem genéricas que permitam a construção de modelos integrados.

Um grande problema relacionado com o desenvolvimento de um esquema de integração CLP-MILP diz respeito ao vocabulário de modelagem. Enquanto a abordagem CLP possui um vocabulário rico (Williams & Wilson, 1998) que permite envolver de forma direta, por exemplo, implicações e equivalências, o vocabulário de modelagem MILP se restringe a equações e inequações, tornando a tarefa de modelagem mais árdua que em CLP. Assim, a existência de estruturas que permitam uma tradução imediata de restrições matemáticas de MILP em restrições lógicas do contexto de CLP (e vice versa), é um facilitador no desenvolvimento de modelos no contexto CLP-MILP.

Tais estruturas foram apresentadas por Magatão (2005) e permitem a formulação de modelos híbridos CLP-MILP. Estas estruturas são baseadas em paradigmas da lógica inferencial, os quais permitem, a partir de declarações de “alto-nível”, a formulação de modelos em MILP e em CLP. Algumas destas estruturas são conhecidas na literatura (Mitra *et al.*, 1994; Floudas, 1995; Hürlimann, 1998; Williams, 1999), outras foram expandidas e/ou criadas por Magatão (2005). De posse destas estruturas, para a obtenção de um modelo, é necessário apenas estabelecer declarações de “alto-nível” sobre o problema e, em seguida, derivar automaticamente as expressões equivalentes em MILP e CLP com o auxílio das estruturas. A principal vantagem do esquema de integração proposto é a abordagem unificada, robusta e de fácil utilização.

Como exemplo, a figura 3 ilustra uma proposição de “alto-nível” do tipo “se somente se” que é expressa nas abordagens MILP e CLP. Nesta proposição, uma variável binária δ_k deve ser feita igual a 1 se somente se $\sum_j a_{kj}x_j < b_k$, onde J é o conjunto de variáveis ($\forall j \in J$), K é o conjunto de restrições ($\forall k \in K$), a_{kj} são os coeficientes das restrições das variáveis contínuas x_j , e b_k são os requisitos. Além disto, L_k e U_k são, respectivamente, os limites inferiores e superiores tais que $L_k \leq \sum_j a_{kj}x_j - b_k \leq U_k \quad \forall k \in K$; ε é um valor pequeno de tolerância ($\varepsilon > 0$). A formulação CLP utiliza diretamente o operador lógico de equivalência (\leftrightarrow) e em MILP as desigualdades são explicitadas (inequações entre chaves na figura 3).

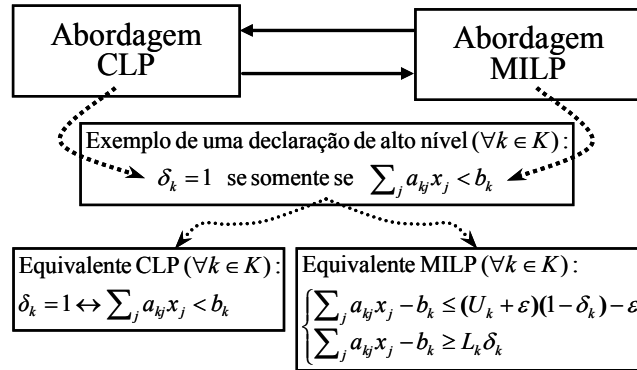


Figura 3 – Formulação da Abordagem Híbrida.

Esta abordagem foi utilizada para a construção do modelo CLP-MILP proposto neste artigo. As hipóteses e restrições do modelo foram inicialmente escritas em uma linguagem de alto-nível e em seguida foram expressas na abordagem de solução mais conveniente. Em termos práticos, três modelos diferentes podem ser implementados: um modelo somente em CLP, um modelo somente em MILP e, por fim, um modelo CLP-MILP formado pelos modelos anteriormente criados. Note que a obtenção do modelo CLP-MILP é trivial devido à equivalência entre os modelos CLP e MILP concebidos através de estruturas de modelagem de alto-nível.

Durante a resolução computacional, o processo de busca de CLP e o de MILP são iniciados e limiares (*bounds*) são trocados entre as buscas. As funcionalidades da ferramenta comercial ILOG OPL Studio 3.6.1 (ILOG, 2002) são utilizadas para a implementação proposta. Essa ferramenta oferece a possibilidade, por exemplo, de se adicionar informação obtida com relaxações lineares do modelo MILP (Wolsey, 1998) para auxiliar o processo de busca definido no contexto de CLP. Assim, existe a possibilidade de um mapeamento entre variáveis nos contextos MILP e CLP. Se, por exemplo, um limiar (*bound*) é obtido para uma variável no processo de busca de MILP, então, este limiar pode ser utilizado para “diminuir o domínio” (*domain reduction*) que esta variável apresenta no contexto de CLP. Com isto, a sinergia entre as duas técnicas contribui de forma decisiva para a eficácia do procedimento de solução. Estudos conduzidos por J.N. Hooker (Hooker, 2000; Hooker, 2002; Hooker, 2007) vêm demonstrando o potencial de se unir esforços destas duas técnicas, que possuem diversas similaridades, mas que utilizam procedimentos de solução oriundos de áreas distintas (Pesquisa Operacional x Ciência da Computação – Inteligência Artificial).

4. Formulação Matemática

As hipóteses assumidas para a elaboração do modelo matemático são sumarizadas nos itens 1 a 13. Ressalta-se que tais hipóteses refletem condições operacionais do poliduto em estudo.

1. O volume de produto requisitado deve ser bombeado em um único lote.
2. Não deve haver descontinuidade entre o término do envio de um lote e o início de envio de um outro lote de produto. Existe uma região de contaminação ou interface entre os produtos que estão sendo bombeados no poliduto. Em particular, a seção 2 apresenta um descritivo sobre as principais características e potenciais prejuízos de interfaces de bombeio a um sistema dutoviário.

3. Existe a possibilidade de se bombear um lote pequeno de um produto entre lotes de elementos que não são compatíveis. Tal produto é denominado *selo*. O uso de selos é prática operacional, mas eleva o custo de bombeio.
4. Os produtos podem ser bombeados de duas origens diferentes: refinaria ou porto. A refinaria é a origem para a operação denominada *fluxo*; o porto é a origem para a operação denominada *refluxo*. O conjunto O é utilizado para representar estas duas diferentes origens. Por convenção $o = 1$ indica a operação de *fluxo* e $o = 2$ indica a operação de *refluxo*. Desta forma, $o \in O$, $O = \{1, 2\}$.
5. Um subconjunto de produtos é bombeado de cada diferente origem ($p \in Po$) durante o horizonte de *scheduling* (H). Uma operação de bombeio completa cobre um procedimento de *fluxo* seguido por um de *refluxo*, ou um procedimento de *refluxo* seguido por um de *fluxo*. Desta forma, a “reversão” no sentido de bombeio deve ser contemplada no modelo.
6. A transição de bombeio que ocorre quando o sistema inverte o sentido de envio de produtos (transição de reversão) deve ser considerada, sendo possível o uso de selos.
7. Em condições normais de operação o poliduto encontra-se preenchido por um produto. O modelo deve considerar a transição entre o produto que inicia a seqüência de bombeio e o produto que preenche o duto (transição inicial), sendo possível o uso de selos.
8. A vazão de bombeio deve permanecer dentro de uma faixa operacional. Desta forma, existe um intervalo de tempo mínimo e máximo para cada operação de bombeio.
9. Um horizonte de *scheduling* mínimo (H^{min}) deve estar disponível para que as operações de bombeio possam ser realizadas. Caso contrário ($H < H^{min}$), o problema é infactível. Para o caso $H = H^{min}$, cada produto deve ser bombeado em sua máxima vazão de bombeio.
10. O poliduto possui um volume interno de 7314 m^3 e sempre opera completamente preenchido. Desta forma, uma quantidade extra de produto deve ser bombeada depois do último produto seqüenciado nas operações de *fluxo* e *refluxo*. Esta quantidade extra de produto é utilizada para manter o poliduto preenchido. A figura 8 ilustra este fato.
11. O volume do produto selo ($\sim 50 \text{ m}^3$) é muito menor que o volume demandado de qualquer outro produto. Então, o seu tempo de bombeio não é considerado.
12. As operações de bombeio no poliduto podem ser paradas (interrompidas) durante o horizonte de *scheduling*. Contudo, existe um custo operacional para se manter o poliduto sem bombear (pressurizado).
13. O custo da energia elétrica sofre variações devido a horários de pico de consumo (tarifação horo-sazonal). Tipicamente, das 18 h às 21 h (pico de demanda) o custo é mais elevado que no resto do dia (COPEL, 2008). Esta variação de custo é considerada na formulação matemática.

Assim, foram consideradas restrições de ordem prática do sistema, de forma a se obter um modelo matemático que forneça condições de operação válidas e otimizadas.

4.1 Notação utilizada

A notação utilizada para a elaboração do modelo, que envolve definições de conjuntos, índices, parâmetros e variáveis, é descrita nesta seção. A consulta a esta notação é indispensável para o entendimento da formulação matemática a seguir apresentada.

4.1.1 Conjuntos

- E Conjunto de intervalos de pico de demanda de energia elétrica que ocorrem dentro do horizonte de *scheduling*; $E = \{1, 2, \dots, ne\}$ (o parâmetro ne é definido na seção 4.1.3).
- O Conjunto de origens, $O = \{1, 2\}$.
- P_o Conjunto de produtos demandados de cada origem o , $P_1 = \{1, 2, \dots, np_1\}$ e $P_2 = \{np_1+1, np_1+2, \dots, np_1 + np_2\}$ (np_1 e np_2 são definidos na seção 4.1.3 – veja o parâmetro np_o).

4.1.2 Índices

- e Índice do intervalo de pico de demanda ($e \in E$, $E = \{1, 2, \dots, ne\}$); cada intervalo possui um custo elétrico associado.
- o, \bar{o} Índices das origens de bombeio, sendo $o \in O$, $\bar{o} \in O$, $o, \bar{o} = \{1, 2\}$. Os produtos podem ser bombeados de duas origens diferentes: refinaria ou porto. A refinaria é a origem para a operação denominada *fluxo*; o porto é a origem para a operação denominada *refluxo*. Os índices o, \bar{o} são utilizados para auxiliar a representação destas duas diferentes origens. Por convenção $o = 1$ indica a operação de *fluxo* e $o = 2$ indica a operação de *refluxo*. Se o bombeio de um produto p da origem o é seguido pelo bombeio do produto pa da origem \bar{o} e $o \neq \bar{o}$, isto indica que haverá reversão no sentido do fluxo após o bombeio de p .
- p, pa Índice dos produtos ($p \in P_o$, $pa \in P_o$).

4.1.3 Parâmetros

- C_o^{low} A energia elétrica recebe uma tarifação horo-sazonal; o parâmetro C_o^{low} indica o custo elétrico básico (fora do horário de pico) quando os produtos são bombeados da origem o (\$/h).
- C_o^{high} A energia elétrica recebe uma tarifação horo-sazonal; o parâmetro C_o^{high} indica a tarifação adicional sobre o valor de custo elétrico básico (C_o^{low}), no caso de bombeio durante um intervalo de pico de consumo de energia; C_o^{high} é aplicado para produtos bombeados da origem o (\$/h).
- C_o^{plug} Custo médio para se bombear um produto selo da origem o (\$).
- C^{pres} Custo médio (\$/h) para manter o duto sem bombear (pressurizado).
- d Define a origem (refinaria ou porto) que inicia o procedimento de bombeio ($d \in \{1, 2\}$, $d + \bar{d} = 3$); se $d = 1$ e $\bar{d} = 2$ então a operação de *fluxo* inicia o procedimento de bombeio, caso contrário ($d = 2$ e $\bar{d} = 1$) a operação de *refluxo* inicia o bombeio.
- \bar{d} Define a origem (porto ou refinaria) que termina o procedimento de bombeio ($\bar{d} \in \{1, 2\}$, $d + \bar{d} = 3$); se $d = 1$ e $\bar{d} = 2$ então a operação de *refluxo* termina o procedimento de bombeio, caso contrário ($d = 2$ e $\bar{d} = 1$) a operação de *fluxo* termina o bombeio.

- $dem_{p,o}$ Volume demandado de um produto p da origem o (m^3).
- ε Parâmetro adimensional “pequeno” (e.g. $\varepsilon = 10^{-3}$, $\varepsilon > 0$) utilizado para expressar declarações *if-then* e *if-then-else* no vocabulário de MILP (Magatão, 2005).
- $\varphi_{p,o}^{\max}$ Vazão máxima de um produto p quando está sendo bombeado da origem o (m^3/h).
- $\varphi_{p,o}^{\min}$ Vazão mínima de um produto p quando está sendo bombeado da origem o (m^3/h). De fato, quando o poliduto está parado (sem bombear), a vazão é igual a zero. Contudo $\varphi_{p,o}^{\min}$ é a mínima vazão permitida de um produto p na origem o quando p está sendo efetivamente bombeado.
- fh_e Durante o horizonte de *scheduling* a energia elétrica recebe uma tarifação horazonal, ou seja, há intervalos de tempo que recebem uma tarifação diferenciada (veja a figura 5, intervalos $e = 1, e = 2, \dots, e = ne$). O parâmetro fh_e indica a hora exata em que o intervalo de pico de demanda e acaba (h).
- H Horizonte de *scheduling* (h).
- L Parâmetro adimensional (limite inferior) utilizado para expressar declarações *if-then* e *if-then-else* no vocabulário de MILP (Magatão, 2005).
- ne Número total de intervalos de pico de consumo de energia que ocorrem durante o horizonte de *scheduling* (veja a figura 6).
- np_o Número de produtos demandados em cada origem o .
- pin Produto que preenche o poliduto no instante de tempo inicial.
- $plug_{p,pa}$ 1 se a interface entre os produtos p e pa demanda o uso de um selo, 0 caso contrário.
- p_v Volume interno do poliduto ($7314 m^3$).
- sh_e Durante o horizonte de *scheduling* a energia elétrica recebe uma tarifação horazonal, ou seja, há intervalos de tempo que recebem uma tarifação diferenciada (veja a figura 5, intervalos $e = 1, e = 2, \dots, e = ne$). O parâmetro sh_e indica a hora exata em que o intervalo de pico de demanda e inicia (h).
- U Parâmetro adimensional (limite superior) utilizado para expressar declarações *if-then* e *if-then-else* no vocabulário de MILP (Magatão, 2005).

4.1.4 Parâmetros de Pré-Processamento

- $d_{p,o}^{\max}$ Máxima duração de bombeio de um produto p da origem o (h), quando p é bombeado de forma ininterrupta.
- $d_{p,o}^{\min}$ Mínima duração de bombeio de um produto p da origem o (h).
- fd Tempo disponível (h) para completar as operações de *fluxo* ($d = 1, \bar{d} = 2$) ou *refluxo* ($d = 2, \bar{d} = 1$) – veja a figura 4.
- fd^{\min} Tempo mínimo necessário (h) para completar as operações de *fluxo* ($d = 1, \bar{d} = 2$) ou *refluxo* ($d = 2, \bar{d} = 1$) – veja a figura 4.

- H^{\min} Horizonte de *scheduling* mínimo (h) para completar todas as operações de bombeio (*fluxo e refluxo*) – veja a figura 4.
- $td_{p,o}^{\max}$ Duração de bombeio máxima para preencher o duto de forma ininterrupta com o produto p da origem o (h).
- $td_{p,o}^{\min}$ Duração de bombeio mínima para preencher o duto de forma ininterrupta com o produto p da origem o (h).

4.1.5 Variáveis

- $dpres$ Variável contínua que indica o tempo (h) que o duto permanece sem bombear (pressurizado).
- $f_{p,o}$ Variável contínua que indica o tempo de fim de bombeio de um produto p da origem o (h).
- $fgh_{p,o,e}$ Variável binária que indica se $f_{p,o}$ é maior ou igual a fh_e ($fgh_{p,o,e} = 1$) ou, caso contrário, $fgh_{p,o,e} = 0$ (veja a figura 5).
- $first_{p,o}$ Variável binária que indica se p é o primeiro produto bombeado da origem o ($first_{p,o} = 1$) ou, caso contrário, $first_{p,o} = 0$.
- $gt_{p,pa,o}$ Variável binária que indica se o produto p é bombeado antes do produto pa ($gt_{p,pa,o} = 1$) ou, caso contrário, $gt_{p,pa,o} = 0$; ambos produtos são bombeados da origem o .
- $inh_{p,o,e}$ Variável contínua que indica, juntamente com $stop_{p,o,e}$ a duração de bombeio de um produto p da origem o no intervalo e (h).
- $last_{p,o}$ Variável binária que indica se p é o último produto a ser bombeado da origem o ($last_{p,o} = 1$) ou, caso contrário, $last_{p,o} = 0$.
- $s_{p,o}$ Variável contínua que indica o tempo de início de bombeio de um produto p da origem o (h).
- $seq_{p,o}$ Variável contínua que indica a posição de um produto p da origem o no seqüenciamento de bombeio; de fato, $seq_{p,o}$ somente assume valores inteiros.
- $sgh_{p,o,e}$ Variável binária que indica se $s_{p,o}$ é maior ou igual a sh_e ($sgh_{p,o,e} = 1$) ou, caso contrário, $sgh_{p,o,e} = 0$ (veja a figura 5).
- $stop_{p,o,e}$ Variável contínua que indica o período de tempo em que um produto p da origem o é mantido pressurizado, durante o intervalo e (h).
- $t_{p,pa,o}$ Variável binária que indica se o bombeio do produto p é imediatamente seguido pelo bombeio do produto pa e, desta forma, a interface $p - pa$ é estabelecida ($t_{p,pa,o} = 1$) ou, caso contrário, $t_{p,pa,o} = 0$; ambos produtos são bombeados da origem o .
- $tw_{p,pa}$ A transição de bombeio que ocorre quando o sistema inverte o sentido de envio de produtos (transição de reversão) é considerada, sendo possível o uso de selos. A variável binária $tw_{p,pa}$ indica se o bombeio do produto p da origem o é seguido pelo bombeio do produto pa da origem \bar{o} ($tw_{p,pa} = 1$) ou, caso contrário, $tw_{p,pa} = 0$. Se ($tw_{p,pa} = 1$) haverá reversão no sentido do fluxo no poliduto após o bombeio de p .

$\delta_{p,o,e}^{00}$; $\delta_{p,o,e}^{01}$; $\delta_{p,o,e}^{10}$; $\delta_{p,o,e}^{11}$ Variáveis contínuas utilizadas para indicar a posição temporal do bombeio de um produto p da origem o em relação a um intervalo e . De fato, as variáveis citadas somente assumem valores binários. O entendimento destas variáveis é facilitado pela observação da figura 5. Tal figura ilustra as seis possíveis posições de um lote de bombeio p (bloco temporal p) em relação a um intervalo e . Na primeira posição (1), o bloco p precede totalmente o intervalo e ($f_{p,o} < sh_e$). Na segunda posição (2), o bloco p inicia antes do intervalo e ($s_{p,o} < sh_e$), mas acaba dentro de e ($sh_e \leq f_{p,o} < fh_e$). Nas posições (1) e (2) a variável $\delta_{p,o,e}^{00}$ é feita igual a 1. Na terceira posição (3), o bloco p inicia e acaba dentro do intervalo e ($s_{p,o} \geq sh_e$ e $f_{p,o} < fh_e$). Na posição (3) a variável $\delta_{p,o,e}^{01}$ é feita igual a um. Na quarta posição (4), o bloco p inicia antes de e ($s_{p,o} < sh_e$) e acaba depois de e ($f_{p,o} \geq fh_e$). Na posição (4) a variável $\delta_{p,o,e}^{10}$ é feita igual a 1. Na quinta posição (5), o bloco p inicia antes de e ($sh_e \leq s_{p,o} < fh_e$) e acaba depois do intervalo e ($f_{p,o} \geq fh_e$). Na sexta posição (6), o bloco p sucede totalmente o intervalo e ($s_{p,o} \geq fh_e$). Nas posições (5) e (6) a variável $\delta_{p,o,e}^{11}$ é feita igual a um.

4.2 Decomposição do modelo

4.2.1 Pré-Processamento

As equações 1 a 7 determinam os parâmetros obtidos no módulo de pré-processamento dos dados. Estes parâmetros estabelecem janelas de tempo que devem ser respeitadas durante a execução das atividades de bombeio. A figura 4 ilustra estas janelas de tempo.

$$d_{p,o}^{\min} = dem_{p,o} / \varphi_{p,o}^{\max} \quad \forall o \in O, p \in P_o \quad (1)$$

$$d_{p,o}^{\max} = dem_{p,o} / \varphi_{p,o}^{\min} \quad \forall o \in O, p \in P_o \quad (2)$$

$$td_{p,o}^{\min} = pv / \varphi_{p,o}^{\max} \quad \forall o \in O, p \in P_o \quad (3)$$

$$td_{p,o}^{\max} = pv / \varphi_{p,o}^{\min} \quad \forall o \in O, p \in P_o \quad (4)$$

$$fd^{\min} = \sum_{\substack{o=d \\ p \in P_o}} d_{p,o}^{\min} + \min_{\substack{o=d \\ p \in P_o}} (td_{p,o}^{\min}) \quad (5)$$

$$H^{\min} = \sum_{\substack{o=d \\ p \in P_o}} d_{p,o}^{\min} + \sum_{\substack{o=d \\ p \in P_o^-}} d_{pa,o}^{\min} + \min_{\substack{o=d \\ p \in P_o}} (td_{p,o}^{\min}) + \min_{\substack{o=d \\ p \in P_o^-}} (td_{pa,o}^{\min}) \quad (6)$$

$$fd = fd^{\min} + H - H^{\min} \quad (7)$$

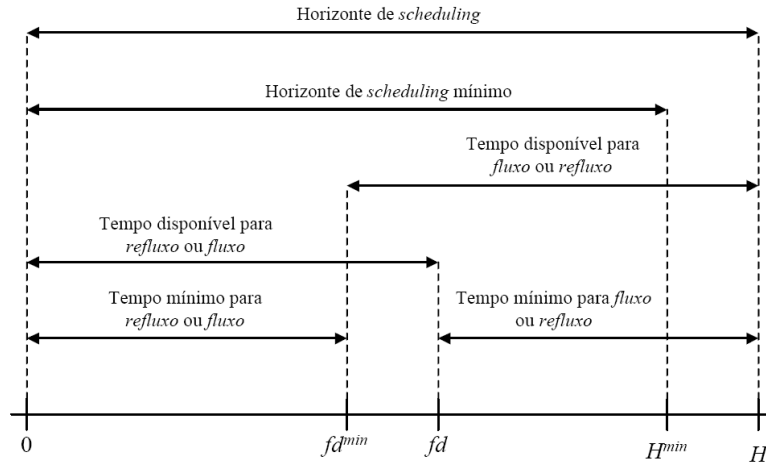


Figura 4 – Janelas de Tempo Obtidas com o Pré-Processamento.

4.2.2 Função Objetivo

A função objetivo do modelo, que indica a minimização do custo operacional, é demonstrada na expressão (8).

$$\begin{aligned}
 \text{minimize } & \sum_{o \in O} \sum_{p \in P_o} C_o^{low} \cdot (f_{p,o} - s_{p,o}) \\
 & + \sum_{o \in O} \sum_{p \in P_o} \sum_{e \in E} C_o^{high} \cdot (inh_{p,o,e} - stop_{p,o,e}) \\
 & + \sum_{o \in O} \sum_{p \neq pa \in P_o} C_o^{plug} \cdot t_{p,pa,o} \cdot plug_{p,pa} \\
 & + \sum_{\substack{o=d \\ p \in P_o}} \sum_{\substack{o=d \\ pa \in P_o}} C_o^{plug} \cdot tw_{p,pa,o} \cdot plug_{p,pa} \\
 & + \sum_{\substack{p=pin \\ o=d \\ pa \in P_o}} C_o^{plug} \cdot first_{p,o} \cdot plug_{p,pa} \\
 & + C^{pres} \cdot dpres
 \end{aligned} \tag{8}$$

Tal função é ponderada pelos seguintes fatores de custo:

- (i) Variações de custo elétrico (C_o^{low} , $C_o^{high} \forall o \in O$);
- (ii) Custo de se utilizar selos ($C_o^{plug} \forall o \in O$); e,
- (iii) Custo para manter o poliduto sem bombear (C^{pres}).

Simplificadamente, a expressão (8) indica que o período de tempo em que um produto é bombeado durante intervalos de pico de demanda é detectado pelo modelo ($inh_{p,o,e} - stop_{p,o,e}$), recebendo uma tarifação adicional (C_o^{high}). Desta forma, o método procura soluções de *scheduling* que evitam o bombeio de produtos durante horários de pico de demanda. Neste caso, alternativamente, o duto pode ser mantido sem bombear ($stop_{p,o,e} > 0$), mas esta condição

também influencia o custo operacional através do termo “ $C^{pres} \cdot dpres$ ” (veja a equação 28). Além disto, a inclusão de selos eleva o custo operacional e o método deve procurar soluções que minimizem o uso de selos. Assim, uma série de fatores deve ser considerada para se determinar a condição de *scheduling* que acarreta o menor custo operacional.

4.2.3 Restrições

O modelo matemático está sujeito a uma série de restrições, indicadas nas expressões 9 a 35. Nestas expressões, as inequações escritas entre chaves são, de fato, o conjunto de restrições implementadas em MILP para as declarações de “alto-nível” *if-then* (se – então) e / ou *if-then-else* (se – então – caso contrário). Como a abordagem CLP possui um vocabulário de modelagem mais rico que o de MILP (Williams & Wilson, 1998) e uma declaração do tipo *if-then-else* é diretamente modelada no contexto de CLP; não há necessidade das inequações apresentadas entre chaves para construção do modelo CLP. Em contrapartida, o vocabulário de modelagem MILP permite somente o uso de igualdades e inequações, tornando a tarefa de modelagem mais árdua que em CLP. Notoriamente é mais simples modelar declarações de alto-nível *if-then* e / ou *if-then-else* em CLP do que formular o conjunto de inequações que representam o mesmo significado lógico de um *if-then* e / ou *if-then-else* em um contexto de modelagem de MILP.

Na declaração *if-then-else* da equação (9), a validade da variável binária $gt_{p,pa,o}$ ($gt_{p,pa,o} = 1$) indica que o produto p é bombeado antes do produto pa ($f_{p,o} \leq s_{pa,o}$); se esta mesma variável assumir um valor falso ($gt_{p,pa,o} = 0$), então ($f_{p,o} > s_{pa,o}$). Ambos os produtos são bombeados da origem o .

$$\begin{aligned} \text{If } gt_{p,pa,o} \text{ Then } f_{p,o} \leq s_{pa,o} \text{ Else } f_{p,o} > s_{pa,o} \quad \forall o \in O, p \neq pa \in P_o \\ \left\{ \begin{array}{l} f_{p,o} - s_{pa,o} \leq U \cdot (1 - gt_{p,pa,o}) \\ f_{p,o} - s_{pa,o} \geq (L - \varepsilon) \cdot gt_{p,pa,o} + \varepsilon \end{array} \right. \quad \forall o \in O, p \neq pa \in P_o \end{aligned} \quad (9)$$

As equações (10) e (11) assinalam a seqüência de bombeio na variável $seq_{p,o}$. Por exemplo, $seq_{1,1}=2$ indica que o produto um ($p=1$) é o segundo a ser bombeado durante a operação de fluxo ($o=1$).

$$\sum_{p \neq pa \in P_o} gt_{p,pa,o} = \sum_{k=1}^{k=np_o} (k-1) \quad \forall o \in O \quad (10)$$

$$seq_{p,o} = np_o - \sum_{p \neq pa \in P_o} gt_{p,pa,o} \quad \forall o \in O, p \in P_o \quad (11)$$

Na declaração *if-then* da equação (12), se a variável binária $t_{p,pa,o}$ assumir um valor verdadeiro, isto indica que o bombeio do produto p é imediatamente seguido pelo bombeio do produto pa . Ambos os produtos são bombeados da origem o . Como anteriormente explicado, as inequações escritas entre chaves são, de fato, o conjunto de restrições implementadas em MILP para a declaração de “alto-nível” *if-then*, que é diretamente modelada em CLP.

$$\begin{aligned} \text{If } t_{p,pa,o} \text{ Then } seq_{pa,o} - seq_{p,o} = 1 \quad \forall o \in O, p \neq pa \in P_o \\ \left\{ \begin{array}{l} seq_{pa,o} - seq_{p,o} - 1 \leq U \cdot (1 - t_{p,pa,o}) \\ seq_{pa,o} - seq_{p,o} - 1 \geq L \cdot (1 - t_{p,pa,o}) \end{array} \right. \quad \forall o \in O, p \neq pa \in P_o \end{aligned} \quad (12)$$

A equação (13) e as inequações (14) e (15) estabelecem que um produto deve ser bombeado somente uma vez durante o horizonte de *scheduling*.

$$\sum_{p \neq pa \in P_o} t_{p,pa,o} = np_o - 1 \quad \forall o \in O, p \in P_o \quad (13)$$

$$\sum_{p \neq pa \in P_o} t_{p,pa,o} \leq 1 \quad \forall o \in O, p \in P_o \quad (14)$$

$$\sum_{p \neq pa \in P_o} t_{pa,p,o} \leq 1 \quad \forall o \in O, p \in P_o \quad (15)$$

A declaração *if-then* na equação (16) e a equação (17) determinam que, se a variável binária $first_{p,o}$ assumir um valor verdadeiro, isto indica que p é o primeiro produto a ser bombeado da origem o .

$$\begin{aligned} & \text{If } first_{p,o} \text{ Then } seq_{p,o} = 1 \quad \forall o \in O, p \neq pa \in P_o \\ & \left\{ \begin{array}{l} seq_{p,o} - 1 \leq U.(1 - first_{p,o}) \quad \forall o \in O, p \in P_o \\ seq_{p,o} - 1 \geq L.(1 - first_{p,o}) \quad \forall o \in O, p \in P_o \end{array} \right. \quad (16) \end{aligned}$$

$$\sum_{p \in P_o} first_{p,o} = 1 \quad \forall o \in O \quad (17)$$

A declaração *if-then* (18) e a equação (19) determinam que, se a variável binária $last_{p,o}$ assumir um valor verdadeiro, isto indica que p é o último produto a ser bombeado da origem o .

$$\begin{aligned} & \text{If } last_{p,o} \text{ Then } seq_{p,o} = np_o \quad \forall o \in O, p \neq pa \in P_o \\ & \left\{ \begin{array}{l} seq_{p,o} - np_o \leq U.(1 - last_{p,o}) \quad \forall o \in O, p \in P_o \\ seq_{p,o} - np_o \geq L.(1 - last_{p,o}) \quad \forall o \in O, p \in P_o \end{array} \right. \quad (18) \end{aligned}$$

$$\sum_{p \in P_o} last_{p,o} = 1 \quad \forall o \in O \quad (19)$$

A expressão (20) estabelece que a variável binária $tw_{p,pa}$ é assume o valor um se o bombeio de um produto p da origem o é seguido pelo bombeio de um produto pa da origem \bar{o} , caso contrário $tw_{p,pa}$ assume o valor zero. As inequações entre chaves são, de fato, a formulação MILP para o argumento lógico *and* (\wedge), que é diretamente modelado em CLP. A expressão de declarações lógicas envolvendo variáveis binárias é um tema conhecido dentro da literatura de MILP. Considere-se, por exemplo, que $c = a \wedge b$, com a e b sendo variáveis binárias. Então, a formulação MILP equivalente para a declaração *and* é indicada pelas seguintes inequações: $c \leq a$, $c \leq b$ e $c \geq a + b - 1$ (Schrage, 2000). Esta equivalência é utilizada na simplificação do modelo CLP-MILP proposto.

$$\begin{aligned} & tw_{p,pa} = (last_{p,o}) \wedge (first_{pa,\bar{o}}) \quad \forall p \in P_o, pa \in P_o, o = d, \bar{o} = \bar{d} \\ & \left\{ \begin{array}{l} tw_{p,pa} \leq last_{p,o} \quad \forall p \in P_o, pa \in P_o, o = d, \bar{o} = \bar{d} \\ tw_{p,pa} \leq first_{pa,\bar{o}} \quad \forall p \in P_o, pa \in P_o, o = d, \bar{o} = \bar{d} \\ tw_{p,pa} \geq last_{p,o} + first_{pa,\bar{o}} - 1 \quad \forall p \in P_o, pa \in P_o, o = d, \bar{o} = \bar{d} \end{array} \right. \quad (20) \end{aligned}$$

As declarações *if-then-else* (21) e (22) indicam que, se as variáveis binárias $fgh_{p,o,e}$ e $sg_{p,o,e}$ recebem o valor um, então, respectivamente, as condições $f_{p,o} \geq fh_e$ e $s_{p,o} \geq sh_e$ são estabelecidas; caso contrário, as condições $f_{p,o} < fh_e$ e $s_{p,o} < sh_e$ é que são estabelecidas. Estas variáveis binárias são utilizadas para identificar a posição do lote de bombeio de um produto p em relação ao horário de ponta de consumo de energia elétrica e .

$$\begin{aligned} &\text{If } fgh_{p,o,e} \text{ Then } f_{p,o} \geq fh_e \text{ Else } f_{p,o} < fh_e \quad \forall o \in O, p \in P_o, e \in E \\ &\begin{cases} f_{p,o} - fh_e \leq L(1 - fgh_{p,o,e}) & \forall o \in O, p \in P_o, e \in E \\ f_{p,o} - fh_e \geq (U + \varepsilon).fgh_{p,o,e} - \varepsilon & \forall o \in O, p \in P_o, e \in E \end{cases} \end{aligned} \quad (21)$$

$$\begin{aligned} &\text{If } sg_{p,o,e} \text{ Then } s_{p,o} \geq sh_e \text{ Else } s_{p,o} < sh_e \quad \forall o \in O, p \in P_o, e \in E \\ &\begin{cases} s_{p,o} - sh_e \leq L(1 - sg_{p,o,e}) & \forall o \in O, p \in P_o, e \in E \\ s_{p,o} - sh_e \geq (U + \varepsilon).sg_{p,o,e} - \varepsilon & \forall o \in O, p \in P_o, e \in E \end{cases} \end{aligned} \quad (22)$$

A figura 5 ilustra as seis possíveis posições de um lote de bombeio p (bloco temporal p) em relação a um intervalo e . Na primeira posição (1), o bloco p precede totalmente o intervalo e ($f_{p,o} < sh_e$). Na segunda posição (2), o bloco p inicia antes do intervalo e ($s_{p,o} < sh_e$), mas acaba dentro de e ($sh_e \leq f_{p,o} < fh_e$). Na terceira posição (3), o bloco p inicia e acaba dentro do intervalo e ($s_{p,o} \geq sh_e$ e $f_{p,o} < fh_e$). Na quarta posição (4), o bloco p inicia antes de e ($s_{p,o} < sh_e$) e acaba depois de e ($f_{p,o} \geq fh_e$). Na quinta posição (5), o bloco p inicia antes de e ($sh_e \leq s_{p,o} < fh_e$) e acaba depois do intervalo e ($f_{p,o} \geq fh_e$). Na sexta posição (6), o bloco p sucede totalmente o intervalo e ($s_{p,o} \geq fh_e$). A duração de um bloco temporal p dentro de um intervalo é determinada pelas expressões (23) a (26).

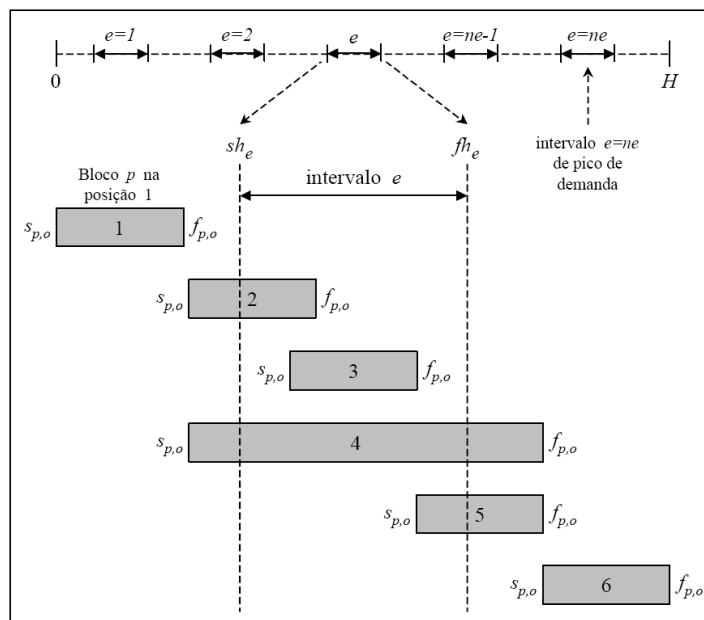


Figura 5 – Custo Elétrico: Intervalos de Pico de Demanda.

A declaração *if-then* (23) indica que, se a expressão lógica $(\neg fgh_{p,o,e}) \wedge (\neg sgh_{p,o,e})$ for verdadeira, então a inequação $inh_{p,o,e} \geq f_{p,o} - sh_e$ deve ser imposta no modelo. A variável auxiliar $\delta_{p,o,e}^{00}$ assume os valores zero ou um de acordo com a expressão lógica $(\neg fgh_{p,o,e}) \wedge (\neg sgh_{p,o,e})$. Em particular, se esta expressão lógica for verdadeira, isto indica que o bloco temporal p está localizado nas posições 1 ou 2 da figura 5.

$$\begin{aligned} & \mathbf{If} (\neg fgh_{p,o,e}) \wedge (\neg sgh_{p,o,e}) \quad \mathbf{Then} \quad inh_{p,o,e} \geq f_{p,o} - sh_e \quad \forall o \in O, p \in P_o, e \in E \\ & \left\{ \begin{array}{ll} \delta_{p,o,e}^{00} \leq 1 - fgh_{p,o,e} & \forall o \in O, p \in P_o, e \in E \\ \delta_{p,o,e}^{00} \leq 1 - sgh_{p,o,e} & \forall o \in O, p \in P_o, e \in E \\ \delta_{p,o,e}^{00} \geq 1 - fgh_{p,o,e} - sgh_{p,o,e} & \forall o \in O, p \in P_o, e \in E \\ inh_{p,o,e} - f_{p,o} + sh_e \geq L \cdot (1 - \delta_{p,o,e}^{00}) & \forall o \in O, p \in P_o, e \in E \end{array} \right. \quad (23) \end{aligned}$$

A declaração *if-then* (24) indica que, se a expressão lógica $(\neg fgh_{p,o,e}) \wedge (sgh_{p,o,e})$ for verdadeira, então a inequação $inh_{p,o,e} = f_{p,o} - s_{p,o}$ deve ser imposta no modelo. A variável auxiliar $\delta_{p,o,e}^{01}$ assume os valores zero ou um de acordo com a expressão lógica $(\neg fgh_{p,o,e}) \wedge (sgh_{p,o,e})$. Em particular, se esta expressão lógica for verdadeira, isto indica que o bloco temporal p está localizado na posição 3 da figura 5.

$$\begin{aligned} & \mathbf{If} (\neg fgh_{p,o,e}) \wedge (sgh_{p,o,e}) \quad \mathbf{Then} \quad inh_{p,o,e} = f_{p,o} - s_{p,o} \quad \forall o \in O, p \in P_o, e \in E \\ & \left\{ \begin{array}{ll} \delta_{p,o,e}^{01} \leq 1 - fgh_{p,o,e} & \forall o \in O, p \in P_o, e \in E \\ \delta_{p,o,e}^{01} \leq sgh_{p,o,e} & \forall o \in O, p \in P_o, e \in E \\ \delta_{p,o,e}^{01} \geq sgh_{p,o,e} - fgh_{p,o,e} & \forall o \in O, p \in P_o, e \in E \\ inh_{p,o,e} - f_{p,o} + s_{p,o} \leq U \cdot (1 - \delta_{p,o,e}^{01}) & \forall o \in O, p \in P_o, e \in E \\ inh_{p,o,e} - f_{p,o} + s_{p,o} \geq L \cdot (1 - \delta_{p,o,e}^{01}) & \forall o \in O, p \in P_o, e \in E \end{array} \right. \quad (24) \end{aligned}$$

A declaração *if-then* (25) indica que, se a expressão lógica $(fgh_{p,o,e}) \wedge (\neg sgh_{p,o,e})$ for verdadeira, então a inequação $inh_{p,o,e} = fh_e - sh_e$ deve ser imposta no modelo. A variável auxiliar $\delta_{p,o,e}^{10}$ assume os valores zero ou um de acordo com a expressão lógica $(fgh_{p,o,e}) \wedge (\neg sgh_{p,o,e})$. Em particular, se esta expressão lógica for verdadeira, isto indica que o bloco temporal p está localizado na posição 4 da figura 5.

$$\begin{aligned} & \mathbf{If} (fgh_{p,o,e}) \wedge (\neg sgh_{p,o,e}) \quad \mathbf{Then} \quad inh_{p,o,e} = fh_e - sh_e \quad \forall o \in O, p \in P_o, e \in E \\ & \left\{ \begin{array}{ll} \delta_{p,o,e}^{10} \leq fgh_{p,o,e} & \forall o \in O, p \in P_o, e \in E \\ \delta_{p,o,e}^{10} \leq 1 - sgh_{p,o,e} & \forall o \in O, p \in P_o, e \in E \\ \delta_{p,o,e}^{10} \geq fgh_{p,o,e} - sgh_{p,o,e} & \forall o \in O, p \in P_o, e \in E \\ inh_{p,o,e} - fh_e + sh_e \leq U \cdot (1 - \delta_{p,o,e}^{10}) & \forall o \in O, p \in P_o, e \in E \\ inh_{p,o,e} - fh_e + sh_e \geq L \cdot (1 - \delta_{p,o,e}^{10}) & \forall o \in O, p \in P_o, e \in E \end{array} \right. \quad (25) \end{aligned}$$

A declaração *if-then* (26) indica que, se a expressão lógica $(fgh_{p,o,e}) \wedge (sgh_{p,o,e})$ for verdadeira, então a inequação $inh_{p,o,e} \geq fh_e - s_{p,o}$ deve ser imposta no modelo. A variável auxiliar $\delta_{p,o,e}^{11}$ assume os valores zero ou um de acordo com a expressão lógica $(fgh_{p,o,e}) \wedge (sgh_{p,o,e})$. Em particular, se esta expressão lógica for verdadeira, isto indica que o bloco temporal p está localizado nas posições 5 ou 6 da figura 5.

$$\begin{aligned} & \mathbf{If} (fgh_{p,o,e}) \wedge (sgh_{p,o,e}) \quad \mathbf{Then} \quad inh_{p,o,e} \geq fh_e - s_{p,o} \quad \forall o \in O, p \in P_o, e \in E \\ & \left\{ \begin{array}{ll} \delta_{p,o,e}^{11} \leq fgh_{p,o,e} & \forall o \in O, p \in P_o, e \in E \\ \delta_{p,o,e}^{11} \leq sgh_{p,o,e} & \forall o \in O, p \in P_o, e \in E \\ \delta_{p,o,e}^{11} \geq fgh_{p,o,e} + sgh_{p,o,e} - 1 & \forall o \in O, p \in P_o, e \in E \\ inh_{p,o,e} - fh_e + sp,o \geq L \cdot (1 - \delta_{p,o,e}^{11}) & \forall o \in O, p \in P_o, e \in E \end{array} \right. \quad (26) \end{aligned}$$

A inequação (27) indica um limite para o tempo em que um produto p da origem o pode ser mantido parado (pressurizado), durante o intervalo e .

$$stop_{p,o,e} \leq inh_{p,o,e} \quad \forall o \in O, p \in P_o, e \in E \quad (27)$$

A equação (28) indica que a variável contínua $dpres$ assume o valor do tempo em que o poliduto permanece sem bombear (pressurizado).

$$dpres = H - \sum_{o \in O} \sum_{p \in P_o} (f_{p,o} - s_{p,o}) + \sum_{o \in O} \sum_{p \in P_o} \sum_{e \in E} stop_{p,o,e} \quad (28)$$

As inequações (29) e (30) estabelecem, respectivamente, o mínimo e o máximo intervalo de tempo de bombeio de cada produto p da origem o .

$$f_{p,o} - s_{p,o} \geq d_{p,o}^{\min} + \sum_{e \in E} stop_{p,o,e} + last_{p,o} \cdot td_{p,o}^{\min} \quad \forall o \in O, p \in P_o \quad (29)$$

$$f_{p,o} - s_{p,o} \leq d_{p,o}^{\max} + \sum_{e \in E} stop_{p,o,e} + last_{p,o} \cdot td_{p,o}^{\max} \quad \forall o \in O, p \in P_o \quad (30)$$

A inequação (31) evita sobreposições entre blocos temporais.

$$f_{p,o} \leq s_{p_a,\bar{o}} \quad \forall p \in P_o, p_a \in P_{\bar{o}}, o = d, \bar{o} = \bar{d} \quad (31)$$

As inequações (32) a (35) estabelecem que os parâmetros temporais (janelas de tempo) determinados no pré-processamento (equações 1 a 7) devem ser respeitados.

$$s_{p,o} \geq 0 \quad \forall p \in P_o, o = d \quad (32)$$

$$s_{p_a,\bar{o}} \geq fd^{\min} \quad \forall p_a \in P_{\bar{o}}, \bar{o} = \bar{d} \quad (33)$$

$$f_{p,o} \geq fd \quad \forall p \in P_o, o = d \quad (34)$$

$$f_{p_a,\bar{o}} \geq H \quad \forall p_a \in P_{\bar{o}}, \bar{o} = \bar{d} \quad (35)$$

Desta forma, as expressões (1) a (35) estabelecem a formulação matemática do modelo CLP-MILP. É importante salientar que os modelos base CLP e MILP podem ser derivados a

partir da formulação matemática descrita. O modelo base CLP é formado pelas expressões (1) a (35), entretanto, as declarações *if-then* e *if-then-else* são diretamente expressas no contexto de CLP, não havendo necessidade das inequações apresentadas entre chaves. De forma análoga, o modelo base MILP é formado por todas as expressões (1) a (35). Contudo, MILP possui um vocabulário de modelagem mais restrito que o de CLP; as declarações *if-then* e *if-then-else* são equivalentemente expressas através dos conjuntos de inequações apresentados entre chaves. Note-se que o número de variáveis/restrições apresentadas pelos modelos base CLP e MILP diferem. MILP utiliza variáveis auxiliares (e.g. $\delta_{p,o,e}^{00}; \delta_{p,o,e}^{01}; \delta_{p,o,e}^{10}; \delta_{p,o,e}^{11}$) e restrições auxiliares (as restrições entre chaves nas declarações *if-then* e *if-then-else*) que não são utilizadas no contexto da formulação CLP. De uma forma simplificada, a abordagem CLP-MILP utiliza a troca de informações entre CLP e MILP para prover uma redução do tempo computacional em relação às formulações de base. Estudos computacionais comparativos entre CLP, MILP e CLP-MILP são realizados na seção 5.

Ressalta-se que as expressões (1) a (35) estabelecem uma formulação matemática criada para as características particulares do poliduto em estudo. Contudo, o procedimento utilizado na geração desta formulação, que faz uso freqüente de declarações *if-then* e *if-then-else* para expressar condições operacionais (Magatão, 2005), pode ser aplicado a outros problemas do setor dutoviário, e/ou outros problemas combinatórios. Em particular, a modelagem detalhada da existência de tarifação horo-sazonal de energia elétrica (vide expressões 21 a 26 e figura 5) em um contexto MILP com tempo contínuo demonstrou-se não trivial e pode ser adequada a outros cenários correlatos.

5. Resultados

5.1 O Cenário Operacional

Um cenário operacional típico do poliduto modelado é a seguir detalhado. Tal cenário é utilizado para investigar os resultados computacionais apresentados pela formulação matemática da seção 4.

- Considera-se um exemplo envolvendo o bombeio de quatro produtos do porto para a refinaria, e em seguida o bombeio de outros quatro produtos da refinaria para o porto. Desta forma, $np_1 = 4$, $np_2 = 4$, $d = 2$, e $\bar{d} = 1$.
- O exemplo cobre simulações desde um horizonte de *scheduling* mínimo (114 h), determinado pela equação (6), até um horizonte de 144 h (6 dias).
- Os parâmetros $C^{plug} \forall o \in O$, $C^{low} \forall o \in O$ e C^{pres} são considerados unitários. O valor de $C_o^{high} \forall o \in O$ é considerado cinco vezes maior que o valor de $C_o^{low} \forall o \in O$, conforme tarifação horo-sazonal apresentada em COPEL (2008).
- Os intervalos de pico de demanda de energia foram considerados entre 18 h – 21 h, 42 h – 45 h, 66 h – 69 h, 90 h – 93 h, 114 h – 117 h e 138 h – 141 h, como ilustrado na figura 6. Desta forma, se $H \leq 114$ h então $ne = 4$; se $114 < H \leq 138$ h então $ne = 5$; se $138 < H \leq 144$ h então $ne = 6$. Em particular, a figura 6 indica seis intervalos de pico ($e=1, e=2, \dots, e=6$).
- No instante inicial o poliduto está preenchido com um produto denominado *pin*, que não requer o uso de selos (*plugs*) com nenhum outro produto ($plug_{pin,p} = 0 \forall o \in O, p \in P_o$).

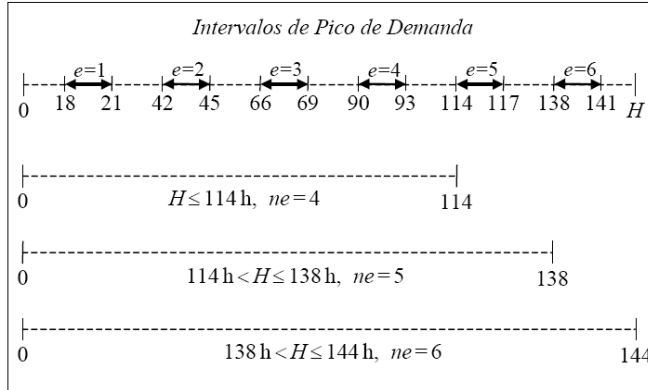


Figura 6 – O Valor de *ne* de acordo com o Horizonte de *Scheduling*.

- A tabela 1 apresenta uma série de parâmetros de entrada para o cenário considerado. Além disto,

$$\varphi_{p,o}^{\min} = 150\text{m}^3/\text{h} \quad \forall o \in O, p \in P_o; \quad \varphi_{p,o}^{\max} = 300\text{m}^3/\text{h} \quad \forall o \in O, p \in P_o;$$

$$td_{p,o}^{\min} = 24\text{h} \quad \forall o \in O, p \in P_o; \quad td_{p,o}^{\max} = 48\text{h} \quad \forall o \in O, p \in P_o;$$

$$U = +145, \quad L = -145 \quad \text{e} \quad \varepsilon = 10^{-3}.$$

Tabela 1 – Parâmetros de Entrada para o Cenário Considerado.

Operação	Produto	$dem_{p,o}(\text{m}^3)$	Necessidade de Selos ($plug_{p,pa}$)
<i>Fluxo</i>	<i>p1</i>	$dem_{1,1} = 1800$	$plug_{1,pa} = 1, \quad pa = 2,4,6,8$
	<i>p2</i>	$dem_{2,1} = 3600$	$plug_{2,pa} = 1, \quad pa = 1,5$
	<i>p3</i>	$dem_{3,1} = 1800$	$plug_{3,pa} = 1, \quad pa = 4,8$
	<i>p4</i>	$dem_{4,1} = 1800$	$plug_{4,pa} = 1, \quad pa = 1,3,5,7$
<i>Refluxo</i>	<i>p5</i>	$dem_{5,2} = 1800$	$plug_{5,pa} = 1, \quad pa = 2,4,6,8$
	<i>p6</i>	$dem_{6,2} = 1800$	$plug_{6,pa} = 1, \quad pa = 1,5$
	<i>p7</i>	$dem_{7,2} = 3600$	$plug_{7,pa} = 1, \quad pa = 4,8$
	<i>p8</i>	$dem_{8,2} = 3600$	$plug_{8,pa} = 1, \quad pa = 1,3,5,7$

5.2 Primeiro Estudo de Caso

Este primeiro estudo de caso considera o cenário de bombeio previamente descrito na seção 5.1. Comparações numéricas entre os modelos base CLP e MILP em relação ao modelo CLP-MILP são realizadas.

A tabela 2 traz informações sobre a simulação dos modelos em uma plataforma computacional Pentium 4, 2.4 GHz, 1 Gbyte RAM. Nesta tabela, *H* é o horizonte de *scheduling*, *ne* é o número de intervalos de pico de demanda de energia elétrica durante *H*, o rótulo MILP refere-se aos resultados obtidos com a formulação base MILP, CLP refere-se aos resultados obtidos com a formulação base CLP, e CLP-MILP refere-se aos resultados

obtidos com a formulação CLP-MILP. Para cada horizonte de *scheduling* (H), um custo operacional específico é obtido (valor da função objetivo calculado pela equação 8). As simulações não consideram incertezas e o modelo é resolvido para demandas fixas, conforme tabela 1. No total, resultados de 31 simulações (problemas-teste) são reportados nesta tabela.

As colunas denominadas “Número de Variáveis” e “Número de Restrições” indicam o número de variáveis e restrições apresentadas pelas abordagens MILP, CLP e CLP-MILP. A coluna “Ótimo encontrado em (s)” indica o tempo de processamento (em segundos) gasto por cada abordagem para encontrar o valor da solução ótima, de acordo com H . Contudo, neste momento, a otimalidade da solução ainda não foi provada. Por outro lado, a coluna “Otimalidade provada em (s)” indica o tempo de processamento (em segundos) gasto por cada abordagem para provar a otimalidade da solução, de acordo com H . A coluna “Nós Explorados” indica o número total de nós visitados pelo algoritmo de busca para provar a otimalidade da solução, de acordo com o modelo e o valor de H . Nestas três colunas, um traço (-) indica que o modelo não foi simulado até a otimalidade (para o problema-teste específico) e os dados computacionais são omitidos. De fato, a simulação foi interrompida após 2 h (7200 s). A coluna “Custo” indica o valor da função objetivo, de acordo com H .

5.2.1 Resultados: estudo comparativo entre as abordagens

A tabela 2 indica que o esforço computacional demandado pela abordagem CLP é maior que o demandado pelas abordagens MILP e CLP-MILP. Em alguns problemas-teste ($121 \text{ h} \leq H \leq 144 \text{ h}$), a solução ótima não pode ser obtida pelo modelo CLP. No entanto, é necessário ressaltar, que soluções (sub)ótimas viáveis foram encontradas pelo modelo CLP em poucos segundos de processamento. Este fato foi observado durante as simulações, mas não é detalhadamente reportado no presente artigo. Ao contrário da formulação CLP, as abordagens MILP e CLP-MILP demandaram um esforço computacional razoável (segundos a poucos minutos). Em particular, a tabela 2 apresenta os resultados computacionais do modelo MILP obtidos com o *solver* CPLEX, imerso no ambiente OPL Studio 3.6.1. (ILOG, 2002).

Tomando-se a coluna “Ótimo encontrado em (s)”, os tempos médios de processamento das abordagens MILP e CLP-MILP foram, respectivamente, 27,3 s e 5,9 s. Desta forma, a abordagem CLP-MILP encontrou a solução ótima, em média, 4,6 vezes mais rápido que o modelo MILP. Agora, tomando-se a coluna “Otimalidade provada em (s)”, os tempos médios das abordagens MILP e CLP-MILP foram, respectivamente, 61,8 s e 28,9 s. Então, em média, a abordagem CLP-MILP forneceu a resposta final de *scheduling* 2,1 vezes mais rápido que a abordagem MILP. Desta forma, a tabela 2 indica que a abordagem CLP-MILP tende a encontrar a solução ótima, e provar a sua otimalidade, mais rápido que o modelo MILP. Ressalta-se que estas foram simulações realizadas em cenários reais do problema em estudo.

Tomando-se, agora, a coluna “Nós Explorados”, percebe-se que a abordagem CLP-MILP tende a visitar um número menor de nós que a abordagem CLP. Em essência, os mecanismos de busca das duas abordagens são similares, à exceção que a abordagem MILP fornece relaxações que auxiliam no processo de busca de CLP-MILP. Desta forma, fica evidente que a troca de informações entre as abordagens (*e.g.* valores de relaxação obtidos com MILP) pode contribuir decisivamente para a redução do espaço de busca. Em particular, a literatura tem relatado técnicas para obter boas relaxações de modelos desde a década de 60 (Dantzig, 1963). A integração destas técnicas, que são amplamente estudadas, para auxiliar os mecanismos de busca de CLP tem, obviamente, um grande potencial.

A bibliografia vem apontando para potenciais da união entre CLP e MILP, destacando resultados promissores oriundos da junção das técnicas (*e.g.* Williams & Wilson, 1998; Heipcke, 1999; Rossi, 1999; Hooker, 2000; Thorsteinsson, 2001; Harjunoski & Grossmann, 2002; Hooker, 2002; Corrêa *et al.*, 2004; Hooker, 2007). De acordo com Hooker & Osorio (1999), é provável que técnicas “tradicionais” de otimização tais como MILP, CLP e Busca Local sejam combinadas de diferentes formas num futuro próximo. Eventualmente, a fronteira entre as técnicas irá se tornar difusa e os procedimentos de busca serão interconectados através de, por exemplo, relaxações e inferências, objetivando-se a eficácia na resolução dos modelos.

5.2.2 Resultados: Redução de Custos Operacionais

A figura 7 é baseada nos resultados apresentados na tabela 2. Esta figura demonstra a variação do custo operacional (valor da função objetivo calculado pela equação 8) como uma função do horizonte de *scheduling* disponível para realizar as operações de bombeio (H).

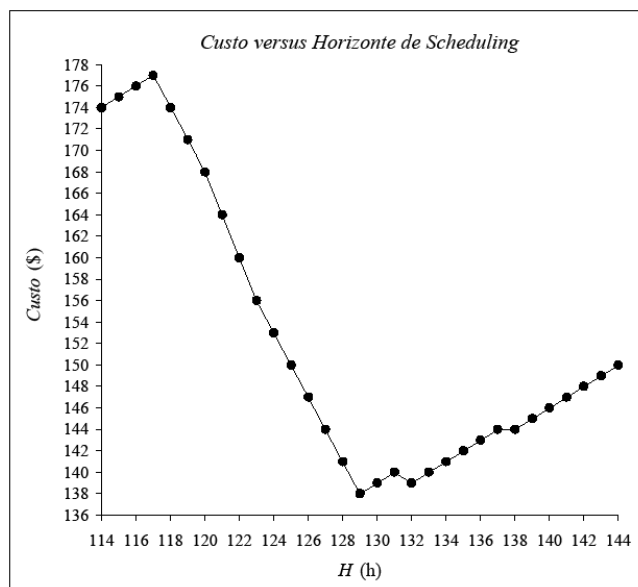


Figura 7 – Custo versus Horizonte de *Scheduling*.

A figura 7 indica a existência de horizontes específicos (*e.g.* $128 \text{ h} \leq H \leq 134 \text{ h}$) que proporcionam baixos custos de operação para o sistema de bombeio. A diversidade de aspectos a serem considerados (*e.g.* sequenciamento de produtos, avaliação de interfaces, uso de selos, temporização do início e final de bombeios) faz com que o programador do sistema, via de regra, estabeleça uma condição operacional válida, mas deixe de operar o poliduto em uma região de baixo custo. A determinação das atividades operacionais de curto prazo não é uma tarefa trivial. O modelo implementado sugere novos pontos de operação para o sistema de bombeio.

5.2.3 Resultados: Visualização das Operações de Bombeio

O modelo fornece o *scheduling* otimizado das atividades operacionais. A figura 8 é um exemplo de visualização de saída operacional a partir da solução do modelo proposto. Esta figura ilustra a operação de bombeio estabelecida pelo modelo para $H = 129$ h. Pela figura 8, a seqüência de bombeio determinada é $p8, p6, p7$ e $p5$ (operação de refluxo); $p1, p3, p2$ e $p4$ (operação de fluxo). De acordo com a tabela 1, esta seqüência de bombeio não demanda o uso de selos:

$$plug_{pin,8} = plug_{8,6} = plug_{6,7} = plug_{7,5} = plug_{5,1} = plug_{1,3} = plug_{3,2} = plug_{2,4} = 0.$$

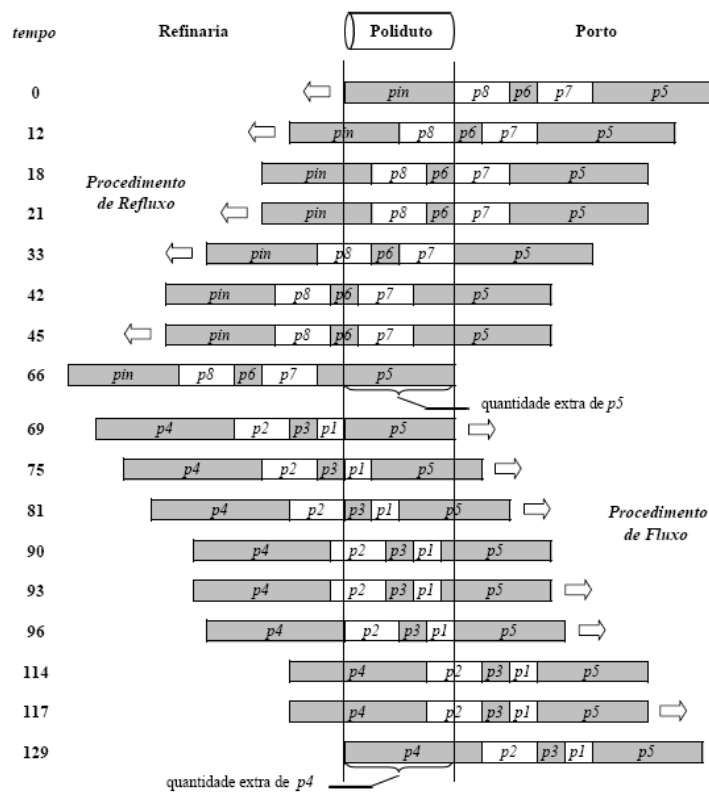


Figura 8 – Procedimento de Bombeio: Um Caso Ilustrativo.

O procedimento de bombeio é iniciado no tempo igual a zero, com a operação de refluxo. No *tempo* igual a 12, a quantidade demandada de $p8$ já foi totalmente bombeada para o duto, mas a refinaria somente receberá este produto mais tarde. Conseqüentemente, existe um tempo entre o envio e o recebimento de um produto. Durante o intervalo $18 < tempo \leq 21$ o duto permanece pressurizado. A condição de manter o duto pressurizado também ocorre durante os intervalos $42 < tempo \leq 45$, $66 < tempo \leq 69$, $90 < tempo \leq 93$ e $114 < tempo \leq 117$. No *tempo* igual a 66, a refinaria já recebeu os produtos enviados do porto ($p8, p6, p7$ e $p5$) e uma quantidade extra de $p5$ preencheu o duto. Então, a operação de fluxo é iniciada com o bombeio de $p1$ (de fato, o bombeio é iniciado somente após o tempo 69). No *tempo* igual a

129 o porto já recebeu os produtos demandados ($p1$, $p3$, $p2$ e $p4$) e um volume extra de $p4$ mantém o duto preenchido. Desta forma, o programador pode antever o *scheduling* otimizado das operações e o modelo funciona como uma ferramenta de auxílio à tomada de decisões operacionais. Ressalta-se que a figura 8 é apenas um exemplo de visualização de saída operacional a partir da solução do modelo proposto. Maiores detalhes podem ser extraídos a partir dos valores encontrados para as variáveis do modelo. Além disso, conforme necessidade, interfaces ser humano computador específicas podem ser construídas a partir da saída do modelo.

5.3 Segundo Estudo de Caso

No primeiro estudo de caso (seção 5.2), as abordagens MILP, CLP e CLP-MILP foram simuladas em problemas-teste que representavam um cenário real do poliduto em estudo. As simulações demandaram de segundos a poucos minutos, ao menos para as abordagens MILP e CLP-MILP, conforme análise apresentada na seção 5.2.1.

No segundo estudo de caso, as três abordagens serão testadas em cenários hipotéticos. Estes cenários não necessariamente representam uma condição operacional real do poliduto. De fato, o objetivo principal é expandir as investigações sobre o esforço computacional demandado pelas três abordagens. Consideram-se, então, cenários teoricamente mais difíceis de serem resolvidos. Os resultados deste experimento são reportados na tabela 3. Os principais parâmetros utilizados no segundo estudo de caso são a seguir declarados nos itens (i) a (v):

- (i) O número de produtos demandados varia entre 8 e 12 $\left(8 \leq \sum_{o \in O} np_o \leq 12\right)$;
- (ii) O modelo é resolvido para demandas fixas: $dem_{p,o} = 1800 \text{ m}^3 \forall o \in O, p \in P_o$;
- (iii) Para cada conjunto de produtos demandados $\left(\sum_{o \in O} np_o = 8, 9, \dots, 12\right)$, o horizonte de *scheduling* varia no seguinte intervalo: $H^{min} \leq H \leq H^{min} + 6 \text{ h}$;
- (iv) A necessidade de selos (*plugs*) segue a seguinte regra:
 $plug_{p,pa} = 1$ se $p + pa$ é um número ímpar, e
 $plug_{p,pa} = 0$ se $p + pa$ é um número par.
 Por exemplo, se $np_o = np_{\bar{o}} = 6$, então $plug_{1,pa} = 1$ para $pa = 2, 4, 6, 8, 10, 12$; e, se $plug_{1,pa} = 0$ para $pa = 1, 3, 5, 7, 9, 11$;
- (v) Os parâmetros de simulação que não foram mencionados nos itens (i) a (iv) são mantidos como originalmente apresentados na seção 5.1.

A tabela 3 traz informações sobre a simulação dos modelos em uma plataforma computacional Pentium 4, 2.4 GHz, 1 Gbyte RAM. Nesta tabela, H é o horizonte de *scheduling*, ne é o número de intervalos de pico de demanda de energia elétrica durante H , np_o e $np_{\bar{o}}$ indicam o número de produtos demandados nas origens o e \bar{o} , enquanto $\sum_{o \in O} np_o$

representa o número total de produtos demandados. O rótulo MILP refere-se aos resultados obtidos com a formulação base MILP, CLP refere-se aos resultados obtidos com a formulação base CLP e CLP-MILP refere-se aos resultados obtidos com a formulação CLP-MILP. Para cada conjunto de valores H , ne , np_o e $np_{\bar{o}}$ os modelos são simulados e um custo operacional específico é obtido (valor da função objetivo na equação 8). As simulações

não consideram incertezas. No total, resultados de 49 simulações (problemas-testes) são reportados nesta tabela.

As colunas denominadas “Número de Variáveis” e “Número de Restrições” indicam o número de variáveis e restrições apresentadas pelas abordagens MILP, CLP e CLP-MILP. A coluna “Otimidade provada em(s)” indica o tempo de processamento (em segundos) gasto por cada abordagem para provar a otimalidade da solução. Nesta coluna, um traço (-) indica que o modelo não foi simulado até a otimalidade (para o problema-teste específico) e os dados computacionais são omitidos. De fato, a simulação foi interrompida após 72 h (259200 s). A coluna “Tempo Médio” indica o tempo computacional médio de simulação, mostrado no formato horas : minutos : segundos (hh:mm:ss). Este tempo é calculado com base nas sete diferentes condições simuladas quando $\sum_{o \in O} np_o$ permanece constante. Por exemplo, para

$\sum_{o \in O} np_o = 8$, $H = 96, 97, 98, 99, 100, 101$ e 102 h, a coluna “Otimidade provada em (s)” indica que os tempos da abordagem CLP-MILP foram, respectivamente, 5,2s, 7,3s, 7,1s, 14s, 12s, 12s e 23s. Então, o tempo médio da abordagem CLP-MILP para $\sum_{o \in O} np_o = 8$ pode ser calculado como sendo $(5,2+7,3+7,1+14+12+12+23) / 7 \approx 12$ s. A coluna “Custo” indica o valor da função objetivo na equação (8).

5.3.1 Resultados: avaliação do esforço computacional

Em essência, os experimentos conduzidos no segundo estudo de caso objetivam avaliar MILP, CLP e CLP-MILP em relação ao esforço computacional demandado. Em particular a coluna “Tempo Médio” da tabela 3 indica a tendência de esforço computacional apresentado pelas três abordagens. A figura 9 permite visualizar esta tendência computacional. Esta figura mostra o tempo computacional médio como uma função do número total de produtos demandados $\sum_{o \in O} np_o$, ou seja, do problema-teste avaliado. Curvas de tendência são inseridas.

Estas curvas são aproximadas por funções exponenciais do tipo $y = ce^{bx}$, onde c e b são constantes; e é o valor da base do logaritmo natural ($e \approx 2,7182818...$). A figura 9 indica que, em média, a abordagem CLP-MILP tende a ter um tempo computacional menor que as abordagens MILP e CLP, principalmente para os problemas-testes em que o número de variáveis e restrições do modelo tratado é significativamente elevado em função de mudanças (hipotéticas) nos parâmetros de entrada. Obviamente, mesmo a abordagem CLP-MILP apresenta uma tendência de crescimento do tempo de processamento. Contudo, CLP-MILP pôde ir “um passo a frente” em relação aos modelos de base, configurando uma alternativa para a diminuição do tempo médio de processamento.

6. Considerações Finais

O aumento da capacidade e da confiabilidade na transferência de derivados de petróleo através de dutos motivou a execução deste trabalho. Foi realizada uma modelagem da programação operacional (*scheduling*) de um poliduto específico do sistema dutoviário brasileiro. O cenário de estudo envolve o poliduto de 93,5 km de extensão que interliga a Refinaria Presidente Getúlio Vargas, localizada em Araucária – PR, ao terminal portuário de Paranaguá, no litoral paranaense.

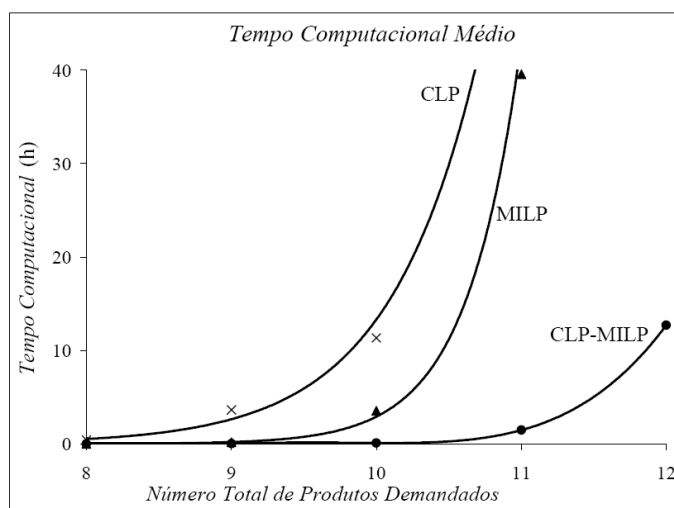


Figura 9 – Tendência do Tempo Computacional para as Abordagens MILP, CLP e CLP-MILP.

Foi desenvolvido um modelo de otimização através de uma abordagem baseada na união de Busca Orientada por Restrições (*Constraint Logic Programming* – CLP) com Programação Linear Inteira Mista (*Mixed Integer Linear Programming* – MILP). O modelo utiliza uma representação temporal contínua e trata detalhes de modelagem como, por exemplo, a existência de tarifação horo-sazonal de energia elétrica. Em particular, a modelagem detalhada da existência de tarifação horo-sazonal (vide expressões 21 a 26 e figura 5) em um contexto MILP com tempo contínuo demonstrou-se não trivial e pode ser adequada a outros cenários correlatos. Assim, foram consideradas restrições de ordem prática do sistema, de forma a se obter um modelo matemático que fornece condições de operação válidas e otimizadas.

A abordagem CLP-MILP proporcionou a resolução de condições reais do problema em tempos computacionais de segundos (vide tabela 2). Além disto, simulações adicionais evidenciaram a viabilidade de resolução através da abordagem CLP-MILP, de problemas-testes em que o número de variáveis e restrições do modelo tratado é significativamente elevado em função de mudanças nos parâmetros de entrada (vide tabela 3 e figura 9). Em um estágio final, a implementação e a resolução computacionais do modelo proposto demonstraram novos pontos de operação para o sistema de bombeio, evidenciando a possibilidade de ganhos operacionais significativos (vide figura 7).

Do ponto de vista teórico, o presente artigo apresenta uma abordagem para a elaboração de um modelo CLP-MILP através de estruturas de modelagem de alto-nível, desenvolvidas por Magatão (2005). Esta abordagem é avaliada para a geração de um modelo representativo de um cenário real do sistema dutoviário brasileiro. O estudo de caso realizado demonstrou uma eficácia computacional da abordagem CLP-MILP em relação às técnicas raízes MILP e CLP. As funcionalidades da ferramenta ILOG OPL Studio 3.6.1 (ILOG, 2002) foram utilizadas para a implementação proposta. Adicionou-se informação obtida com relaxações lineares do modelo MILP para auxiliar o processo de busca definido no contexto de CLP. No caso, foi feito um mapeamento entre variáveis nos contextos MILP e CLP. Quando um limiar (*bound*) era obtido para uma variável no processo de busca de MILP, então, este limiar era utilizado para “diminuir o domínio” (*domain reduction*) que esta variável apresentava no contexto de

CLP. Ou seja, realizou-se uma interferência no processo de busca de CLP. A melhor eficácia computacional da implementação proposta em relação à MILP e CLP atuando em separado é um fato relevante (vide tabela 3 e figura 9), visto que uma abordagem combinada pode herdar deficiências das técnicas raízes (*inherited drawbacks* – Heipcke, 1999; Hooker, 2002). A combinação MILP-CLP via relaxação linear e redução de domínios é uma contribuição dessa abordagem.

Em termos práticos, o modelo desenvolvido configura uma ferramenta de auxílio ao processo de tomada de decisões operacionais. O programador pode antever detalhes do *scheduling* otimizado das operações (vide figura 8). Atualmente a maior parte dos polidutos opera somente baseado na experiência de programadores; via de regra, políticas operacionais que não utilizam a máxima capacidade de operação dos polidutos são adotadas. O desenvolvimento de um modelo de otimização como o apresentado neste artigo possibilita que procedimentos operacionais complexos sejam avaliados de forma criteriosa, fazendo com que o sistema opere da melhor maneira possível.

Agradecimentos

Apoio financeiro da Agência Nacional do Petróleo – ANP – e da Financiadora de Estudos e Projetos – FINEP – por meio do Programa de Recursos Humanos da ANP para o Setor Petróleo e Gás PRH-ANP/MCT (PRH10 – UTFPR) e CNPq (processo 303517/2004-2).

Referências Bibliográficas

- (1) Bollapragada, S.; Ghattas, O. & Hooker, J.N. (2001). Optimal design of truss structures by mixed logical linear programming. *Operations Research*, **49**, 42-51.
- (2) Cafaro, D.C. & Cerdá, J. (2004). Optimal scheduling of multiproduct pipeline systems using a non-discrete MILP formulation. *Computers & Chemical Engineering*, **28**, 2053-2068.
- (3) Cafaro, D.C. & Cerdá, J. (2008). Dynamic scheduling of multiproduct pipelines with multiple delivery due dates. *Computers & Chemical Engineering*, **32**, 728-753.
- (4) COPEL (2008). Companhia Paranaense de Energia Elétrica – Tarifas de energia elétrica. Disponível em <<http://www.copelsolucoes.com/downloads/downloads.html>>, acessado em abril de 2008.
- (5) Corrêa, A.I.; Langevin, A. & Rousseau, L.M. (2004). Dispatching and conflict-free routing of automated guided vehicles: A hybrid approach combining constraint programming and mixed integer programming. *Lecture Notes in Computer Science*, **3011**, 370-378.
- (6) Dantzig, G.B. (1963). *Linear Programming and Extensions*. Princeton University Press, USA.
- (7) Floudas, C.A. (1995). *Nonlinear and Mixed Integer Optimization: Fundamentals and Applications*. Oxford University Press, New York, USA.
- (8) Focacci, F. (2000). Solving Combinatorial Optimization Problems in Constraint Programming. PhD thesis, Università degli Studi di Ferrara, Ferrara, Itália.

- (9) Garey, M.R. & Johnson, D.S. (1979). *Computers and Intractability – A Guide to the Theory of NP Completeness*. W. H. Freeman and Company, New York, New Jersey, USA.
- (10) Harjunkoski I. & Grossmann, I.E. (2002). Decomposition techniques for multistage scheduling problems using mixed-integer and constraint programming methods. *Computers & Chemical Engineering*, **26**, 1533-1552.
- (11) Heipcke, S. (1999). Combined Modeling and Problem Solving in Mathematical Programming and Constraint Programming. PhD thesis, University of Buckingham, UK.
- (12) Hooker, J.N. (2000). *Logic-Based Methods for Optimization: Combining Optimization and Constraint Satisfaction*. Wiley-Interscience Series in Discrete Mathematics and Optimization, New York, USA.
- (13) Hooker, J.N. (2002). Logic, optimization and constraint programming. *INFORMS Journal on Computing*, **14**, 295-321.
- (14) Hooker, J.N. (2007). Planning and scheduling by logic-based benders decomposition, *Operations Research*, **55**(3), 588-602.
- (15) Hooker, J.N. & Osorio, M.A. (1999). Mixed/logical linear programming. *Discrete Applied Mathematics*, **96-97**(1-3), 395-442.
- (16) Hürlimann, T. (1998). An efficient logic-to-IP translation procedure. *International Conference, Applied Mathematical Programming and Modeling*, Lymassol, Cyprus, 1-26.
- (17) ILOG (2002). ILOG OPL Studio 3.6.1 – User’s Manual. ILOG Corporation, France.
- (18) Kennedy, J.L. (1993). *Oil and Gas Pipeline Fundamentals*. PennWell Publishing Company, Oklahoma.
- (19) Mitra, G.; Lucas C.; Moody, S. & Hadjiconstantinou, E. (1994). Tools for reformulating logical forms into zero-one mixed integer programs. *European Journal of Operational Research*, **72**, 262-276.
- (20) Kondili, E.; Pantelides, C.C. & Sargent, R.W.H. (1993). A general algorithm for short-term scheduling of batch operations – I. MILP formulation. *Computers & Chemical Engineering*, **17**(2), 211-227.
- (21) Land, A.H. & Doig, A.G. (1960). An automatic method for solving discrete programming problems. *Econometrica*, **28**, 497-520.
- (22) Magatão, L. (2005). Constraint Logic Programming and Mixed Integer Linear Programming: Towards a Unified Modeling Framework. PhD Thesis, Universidade Tecnológica Federal do Paraná (UTFPR), Curitiba, Paraná, Brazil.
- (23) Magatão, L.; Arruda, L.V.R. & Neves-Jr, F. (2004). A mixed integer programming approach for scheduling commodities in a pipeline. *Computers & Chemical Engineering*, **28**(1-2), 171-185.
- (24) Neiro, S.M. & Pinto, J.M. (2004). A general modeling framework for the operational planning of petroleum supply chains. *Computers & Chemical Engineering*, **28**, 871-896.
- (25) Raman, R. & Grossmann, I.E. (1994). Modeling and computational techniques for logic based integer programming. *Computers & Chemical Engineering*, **17**(9), 909-927.

- (26) Reklaitis, G.V. (1992). Overview of scheduling and planning of batch process operations. **In: *Proceedings of the NATO Advanced Study Institute on Batch Processing Systems***, Antalya, Turkey, 660-705.
- (27) Rejowski Jr., R. & Pinto, J.M. (2003). Scheduling of a multiproduct pipeline system. *Computers & Chemical Engineering*, **27**, 1229-1246.
- (28) Rejowski Jr., R. & Pinto, J.M. (2004). Efficient MILP formulations and valid cuts for multiproduct pipeline scheduling. *Computers & Chemical Engineering*, **28**, 1511-1528.
- (29) Rejowski Jr., R. & Pinto, J.M. (2008). A novel continuous time representation for the scheduling of pipeline systems with pumping yield rate constraints. *Computers & Chemical Engineering*, **32**, 1042-1066.
- (30) Relvas, S.; Matos, H.A.; Barbosa-Póvoa, A.P.F.D.; Fialho, J. & Pinheiro, A.S. (2006). Pipeline Scheduling and Inventory Management of a Multiproduct Distribution Oil System. *Industrial & Engineering Chemistry Research*, **45**, 7841-7855.
- (31) Relvas, S.; Matos, H.A.; Barbosa-Póvoa, A.P.F.D. & Fialho, J. (2007). Reactive Scheduling Framework for a Multiproduct Pipeline with Inventory Management. *Industrial & Engineering Chemistry Research*, **46**, 5659-5672.
- (32) Rossi, F. (1999). Constraint logic programming: A survey on research and applications. *Lecture Notes in Computer Science*, Springer, **1865**, 40-74.
- (33) Sasikumar, M.; Prakash, P.R.; Patil, S.M. & Ramani, S. (1997). PIPES: A heuristic search model for pipeline schedule generation. *Knowledge-Based Systems*, **10**, 169-175.
- (34) Schrage, L. (2000). *Optimization Modeling with LINGO*. LINDO Systems Inc., Chicago, Illinois.
- (35) Shah, N. (1996). Mathematical Programming Techniques for Crude Oil Scheduling. *Computers and Chemical Engineering*, **20**, suppl., S1227-S1232.
- (36) Souza-Filho, E.M.; Alves, V.R.F.M. & Ferreira-Filho, V.J.M. (2006). Utilização de técnicas de pesquisa operacional em problemas de distribuição dutoviária: uma revisão. *Anais do XXXVIII SBPO*.
- (37) Thorsteinsson, E.S. (2001). Hybrid Approaches to Combinatorial Optimization. PhD Thesis, Carnegie Mellon University, Pittsburg, Pennsylvania, USA.
- (38) Türkay, M. & Grossmann, I.E. (1996). Logic-based MINLP algorithms for the optimal synthesis of process network. *Computers & Chemical Engineering*, **20**, 959-978.
- (39) Williams, H.P. (1995). Logic applied to integer programming and integer programming applied to logic. *European Journal of Operational Research*, **81**, 605-616.
- (40) Williams, H.P. (1999). *Model Building in Mathematical Programming*. John Wiley & Sons, Inc., UK.
- (41) Williams, H.P. & Wilson, J.M. (1998). Connections between integer linear programming and constraint logic programming – An overview and introduction to the cluster of articles. *INFORMS Journal on Computing*, **10**(3), 261–264.
- (42) Wolsey, L.A. (1998). *Integer Programming*. John Wiley & Sons, Inc., New York, USA.

Tabela 2 – Estudo de Caso 1 – Formulações MILP, CLP e CLP-MILP

H (h)	ne	Número de Variáveis			Número de Restrições			Ótimo encontrado em (s)			Otimidade provada em (s)			Nós Explorados			Custo (\$)
		MILP	CLP	CLP-MILP	MILP	CLP	CLP-MILP	MILP	CLP	CLP-MILP	MILP	CLP	CLP-MILP	MILP	CLP	CLP-MILP	
114	4	361	273	417	1132	571	1620	11	0.7	0.8	15	0.8	15	9453	575	575	174
115	5	425	305	481	1324	631	1860	15	7.9	1.1	17	69	17	11573	15610 ³	614	175
116	5	425	305	481	1324	631	1860	2	41	1.1	19	315	19	13492	62710 ³	714	176
117	5	425	305	481	1324	631	1860	2	55	2.3	22	417	22	15708	87010 ³	1655	177
118	5	425	305	481	1324	631	1860	9	86	2.4	23	629	23	15028	1.25 10 ⁶	1642	174
119	5	425	305	481	1324	631	1860	2	151	2.5	34	1075	34	23091	1.90 10 ⁶	1783	171
120	5	425	305	481	1324	631	1860	27	1069	5.1	42	6312	42	28509	13.7 10 ⁶	3596	168
121	5	425	305	481	1324	631	1860	13	-	5.8	41	-	41	25145	-	4073	164
122	5	425	305	481	1324	631	1860	34	-	5.8	46	-	46	32258	-	3976	160
123	5	425	305	481	1324	631	1860	21	-	7.2	75	-	75	49079	-	5046	156
124	5	425	305	481	1324	631	1860	51	-	6.0	85	-	85	56613	-	4509	153
125	5	425	305	481	1324	631	1860	30	-	5.9	61	-	61	37645	-	4480	150
126	5	425	305	481	1324	631	1860	3	-	6.6	64	-	64	44967	-	5911	147
127	5	425	305	481	1324	631	1860	2	-	6.6	91	-	91	61202	-	5419	144
128	5	425	305	481	1324	631	1860	34	-	6.5	78	-	78	48898	-	4685	141
129	5	425	305	481	1324	631	1860	20	-	5.0	45	-	45	25999	-	2749	138
130	5	425	305	481	1324	631	1860	90	-	5.1	90	-	90	50686	-	2879	139
131	5	425	305	481	1324	631	1860	13	-	5.4	36	-	36	22185	-	2988	140
132	5	425	305	481	1324	631	1860	40	-	7.0	60	-	60	31740	-	2888	139
133	5	425	305	481	1324	631	1860	37	-	7.0	57	-	57	31651	-	2900	140
134	5	425	305	481	1324	631	1860	8	-	6.9	50	-	50	27005	-	2783	141
135	5	425	305	481	1324	631	1860	43	-	7.1	91	-	91	47142	-	2878	142
136	5	425	305	481	1324	631	1860	29	-	7.2	84	-	84	48055	-	2883	143
137	5	425	305	481	1324	631	1860	8	-	7.1	64	-	64	34860	-	2844	144
138	5	425	305	481	1324	631	1860	45	-	5.6	73	-	73	36620	-	2547	144
139	6	489	337	545	1516	691	2100	50	-	6.9	52	-	52	27746	-	2835	145
140	6	489	337	545	1516	691	2100	55	-	7.0	88	-	88	43455	-	2873	146
141	6	489	337	545	1516	691	2100	116	-	7.5	170	-	170	62829	-	3067	147
142	6	489	337	545	1516	691	2100	3	-	7.6	113	-	113	60221	-	3091	148
143	6	489	337	545	1516	691	2100	30	-	7.4	73	-	73	32219	-	3116	149
144	6	489	337	545	1516	691	2100	2	-	18	57	-	57	29202	-	8445	150

Tabela 3 – Estudo de Caso 2 – Formulações MILP, CLP e CLP-MILP

H (h)	ne	np _o	np _o	∑ _{o∈O} np _o	Número de Variáveis		Número de Restrições		Otimalidade provada em (s)		Tempo Médio (hh:mm:ss)		Custo (\$)
					MILP	CLP-MILP	MILP	CLP-MILP	MILP	CLP	MILP	CLP	
96	4	4	4	8	361	273	417	1132	571	23	0.7	5.2	158
97	4	4	4	8	361	273	417	1132	571	41	6.7	7.3	154
98	4	4	4	8	361	273	417	1132	571	27	46	7.1	150
99	4	4	4	8	361	273	417	1132	571	61	231	00:00:12	146
100	4	4	4	8	361	273	417	1132	571	52	397	12	143
101	4	4	4	8	361	273	417	1132	571	53	635	12	140
102	4	4	4	8	361	273	417	1132	571	91	9952	23	137
102	4	5	4	9	418	319	481	1302	679	256	3.2	25	164
103	4	5	4	9	418	319	481	1302	679	386	42	36	160
104	4	5	4	9	418	319	481	1302	679	548	332	35	156
105	4	5	4	9	418	319	481	1302	679	498	1570	73	152
106	4	5	4	9	418	319	481	1302	679	717	2789	64	149
107	4	5	4	9	418	319	481	1302	679	721	4514	64	146
108	4	5	4	9	418	319	481	1302	679	1206	82626	157	143
108	4	5	5	10	476	366	546	1479	792	4908	23	166	170
109	4	5	5	10	476	366	546	1479	792	5834	245	222	166
110	4	5	5	10	476	366	546	1479	792	9306	2124	224	162
111	4	5	5	10	476	366	546	1479	792	16752	10819	431	158
112	4	5	5	10	476	366	546	1479	792	16890	19848	378	155
113	4	5	5	10	476	366	546	1479	792	11069	32491	379	152
114	4	5	5	10	476	366	546	1479	792	18202	220110	891	149
114	4	6	5	11	538	417	615	1664	917	103243	356	2186	176
115	5	6	5	11	626	461	703	1925	998	133381	29477	2934	177
116	5	6	5	11	626	461	703	1925	998	2804	111275	186635	178
117	5	6	5	11	626	461	703	1925	998	125112	-	5916	179
118	5	6	5	11	626	461	703	1925	998	169996	-	5907	176
119	5	6	5	11	626	461	703	1925	998	176697	-	5768	173
120	5	6	5	11	626	461	703	1925	998	177174	-	11607	170
120	5	6	6	12	697	517	781	2140	1135	-	-	21324	197
121	5	6	6	12	697	517	781	2140	1135	-	-	24808	193
122	5	6	6	12	697	517	781	2140	1135	-	-	24213	189
123	5	6	6	12	697	517	781	2140	1135	-	-	49254	185
124	5	6	6	12	697	517	781	2140	1135	-	-	50233	182
125	5	6	6	12	697	517	781	2140	1135	-	-	50020	179
126	5	6	6	12	697	517	781	2140	1135	-	-	100530	176