

# Algoritmo identificador de vértices viáveis para solução do PLDN

Érik da Silva Oliveira (UNIG) - eriksilvaoliveira@yahoo.com.br



## **Resumo**

*Devido à grande competitividade mundial, as organizações estão preocupadas com a otimização de seus processos produtivos inseridos em ambientes sinérgicos, mutáveis e hierarquizados. A pesquisa operacional é uma metodologia decisória com algoritmos que buscam a otimização desses processos. A programação linear em dois níveis (PLDN) é um modelo de Pesquisa Operacional que representa bem operações de produção que dependem de dois níveis hierárquicos de decisão. Assim, com o modelo de PLDN encontra-se uma solução compatível com os interesses de dois níveis decisórios distintos, sendo cada um deles governado por uma parcela de variáveis que interagem nas restrições do modelo. O algoritmo identificador de vértices viáveis é um método que reconhece todos os pontos extremos viáveis do PLDN, sendo sua implementação em MATLAB bastante eficiente no entendimento dos resultados teóricos do PLDN.*

**Palavras-chave:** Programação linear em dois níveis, sistemas de decisão descentralizados e condições de KKT e algoritmo identificador de vértices viáveis.

## ALGORITHM FOR IDENTIFYING VIABLE VERTICES FOR PLDN SOLUTION

### **Abstract**

*Due to great worldwide competitiveness, organizations are focused on optimization of their productive processes in synergistic, changeable and hierarchical environments. Operational research is a decisive methodology with algorithms searching optimization of these processes. Bi-level linear programming (BLP) is an Operational Research model that well represents production operations depending on two hierarchic levels of decision. Thus, the PLDN model provides a solution compatible with the interests of two distinct decisive levels, each being governed by a number of variables interacting in the restrictions of the model. The algorithm for identifying viable vertices is a method that recognizes all the BLP viable extreme points, with its implementation in MATLAB being sufficiently efficient in understanding the theoretical BLP results.*

**Keywords:** Bilevel linear programming, decentralized decision systems, KKT conditions and algorithm for identifying viable vertices.

# 1. INTRODUÇÃO

Em decorrência do processo de globalização do mundo atual, as empresas estão preocupadas com seus produtos, serviços e processos produtivos, propiciando, assim, uma busca incessante por melhorias, almejando minimizar os custos e aumentar os lucros. Esse esforço pela otimização faz com as empresas possam estar mais preparadas e competitivas a cada dia, com o objetivo de se estabelecer no mercado e manter a aceitação de seu público consumidor em relação a seus produtos e serviços, além de buscar liderança em relação a seus concorrentes.

A pesquisa operacional é considerada como um ramo da ciência que busca a otimização dos processos empresariais, de modo a atingir um patamar ótimo no processo produtivo como um todo, isto é, sem deixar de considerar o fator humano. No entanto, deve-se ter em mente que em um ambiente mutável como o atual, com possibilidade de influência mútua entre as empresas, se deve buscar procedimentos de otimização que tratem não somente do ambiente interno da empresa, mas também que considerem o ambiente externo onde ela estiver inserida, e como se pode interagir com esse ambiente, de forma que se possa obter o melhor dele, visando um desenvolvimento sustentável.

## 2. HISTÓRICO DA PESQUISA OPERACIONAL

A origem da pesquisa operacional formalmente data dos anos de 1940 com os primeiros intentos de usar o método científico na administração de recursos escassos. Com a chegada da revolução industrial, as organizações, os negócios e os órgãos governamentais se depararam com a necessidade de trabalhar os recursos da melhor forma possível, pois estes são limitados. Segundo Hillier e Lieberman (1998) apud Gavira, (2003, p. 25):

A revolução industrial trouxe ao mundo um notável crescimento no tamanho e complexidade das organizações. Essa revolução proporcionou um aumento na divisão do trabalho e das responsabilidades das empresas. Os resultados foram excelentes, mas junto com eles surgiram problemas. Dentre esses problemas encontra-se por parte dos segmentos organizacionais, a perda da visão do objetivo organizacional e de como as atividades das organizações devem interagir para atingí-lo. Outro problema relacionado é a alocação de recursos disponíveis entre as várias atividades de maneira eficaz. Esses problemas, bem como a necessidades de solucioná-los, proporcionaram um incentivo a estudos científicos que hoje podemos relacionar com a Pesquisa Operacional.

A Pesquisa Operacional como ciência surgiu para resolver, de uma forma mais eficiente, a distribuição ótima de recursos, os problemas de administração nas organizações, etc., originados pelo acelerado desenvolvimento provocado pela revolução industrial (JORDÁN, 2002, p.1).

A experiência de sucesso na solução de problemas bélicos e o reconhecimento de que, em essência, uma organização industrial tem restrições e objetivos análogos aos de um procedimento militar em época de guerra, o progresso alcançado no conhecimento de conceitos e técnicas matemáticas na solução de problemas e a aparição do computador aceleraram o desenvolvimento da pesquisa operacional. Assim, a necessidade de resolução dos problemas, de forma mais eficiente, na área de administração dos processos de produção, buscando uma solução ótima, conduziram para a afirmação da importância da pesquisa operacional como ciência utilizada em problemas relacionados à condução e coordenação de operações (ou atividades) dentro de uma organização.

Atualmente, é possível classificar as áreas envolvidas na pesquisa operacional, segundo o Quadro 1. Nesse quadro foram incorporados o Problema da Mochila, a Programação em Redes, a Programação Multi-Objetivo, a Programação Quadrática, a Programação Multinível e as Heurísticas, adaptando-se o quadro proposto por Jordán (2002, p. 4), à visão do autor deste trabalho.

### 2.1. A Programação Multinível

Muitos problemas de tomada de decisão requerem acordos entre as partes envolvidas no processo de negociação ou de resposta às decisões tomadas por uma das partes, pois os objetivos dessas partes, ou seja, os diversos indivíduos ou entidades, se encontram em contínua interação.

De forma geral, tais processos de decisão se encontram dispostos dentro de uma estrutura hierárquica, cujos objetivos independentes são, na grande maioria das vezes, antagônicos e, de certa forma, podem-se opor. Cada unidade ou nível está associado a certa hierarquia, ao qual as partes estão dispostas e interagindo entre si e desejam a otimização de seu objetivo individual, de forma que exerça certo controle sobre os outros níveis ou hierarquias. A Programação Multinível representa melhor a otimização de organizações hierárquicas que buscam um valor ótimo para o nível principal, de forma que seja compatível ou conciliável com os vários níveis aos quais estão ligados e dos quais ele depende.

Programação Matemática	Outros Ramos
<ul style="list-style-type: none"> <li>+ Programação Linear (<i>Linear Programming</i>);               <ul style="list-style-type: none"> <li>• Problemas de Distribuição de Recursos, Pessoal, etc.</li> <li>• Problemas de Transporte</li> <li>• Problemas de Planejamento da Produção</li> <li>• Problemas de Cortes de Material</li> <li>• Problema da Mochila</li> </ul> </li> <li>+ Programação Não-Linear</li> <li>+ Programação em Redes</li> <li>+ Programação Dinâmica</li> <li>+ Programação Multi-Objetivo</li> <li>+ Programação Inteira</li> <li>+ Programação Quadrática</li> <li>+ Programação Multinível</li> <li>+ Otimização Global</li> </ul>	<ul style="list-style-type: none"> <li>+ Análise Estatística</li> <li>+ Programação Estocástica</li> <li>+ Teoria de Jogos</li> <li>+ Teoria de Filas               <ul style="list-style-type: none"> <li>• Organização de Tráfego Aéreo</li> <li>• Congestão do Trafeg</li> <li>• Construção de Barragens</li> </ul> </li> <li>+ Simulação</li> <li>+ Gestão de Estoques</li> <li>+ Heurísticas</li> <li>+ Etc.</li> </ul>

Quadro 1 - Os ramos mais importantes desenvolvidos pela pesquisa operacional.

Fonte: Adaptado de JORDÁN, 2002, p. 4.

Como descreveu Bialas (2004, p. 1), "As técnicas de otimização Multinível dividem o controle sobre as variáveis de decisão de um problema de otimização entre os responsáveis pelas decisões". Assim, um problema de Programação Multinível, de forma geral, é um conjunto de problemas de otimização encadeadas, embutidas uma nas outras, cuja composição fornece a região viável onde se vai ser procurado um ótimo para o Problema de Programação Multinível:

Uma característica importante de problemas de otimização Multinível é que um planejador em um nível da hierarquia pode ter sua função objetivo determinada, em parte, pelas variáveis controladas por outros níveis. Entretanto, seus instrumentos do controle podem lhe dar permissão para que influencie nas políticas dos outros níveis e melhorem, através disso, sua própria função objetivo. Tais políticas podem incluir o controle de alocação e de uso dos recursos nos níveis mais baixos, e o controle dos benefícios conferidos sob os níveis subordinados (BIALAS; KARWAN, 1979, p. 2).

Observa-se, dessa forma, que o controle sobre as variáveis de decisão é dividido entre os diversos níveis hierárquicos, ressaltando-se que uma variável de decisão pode impactar a função-objetivo de diversos outros níveis hierárquicos ou, então, de todos os níveis. Segundo Bialas (2005),

O paradigma da otimização multinível é útil em muitos campos incluindo: Economia, Pesquisa Operacional, Estatísticas, e a Teoria de Controle, com aplicações nos Sistemas do Transporte, Projeto de Engenharia do Sistema, Engenharia ambien-

tal, Psicologia, Fatores Humanos, Comportamento da Organização (elementos que influenciam a organização), Sociologia e Ciências Políticas.

A aplicação de modelos de Programação Multinível é, assim, útil em Teoria de Controle Ótimo, Economia, Sistema de Transporte, Engenharia Ambiental, Sociologia e Ciências Políticas etc. Dessa forma, pode-se dizer que os problemas de Programação Multinível apresentam basicamente as seguintes características:

- i) O sistema-modelado possui unidades de tomada de decisão que interagem dentro de uma estrutura predominantemente hierárquica ou disposta de forma escalonada.
- ii) Cada unidade de tomada de decisão busca maximizar seus lucros, ou minimizar seus custos, independentemente de outras unidades, mas, no entanto, suas decisões são afetadas pelas ações de outras unidades externas.
- iii) O efeito externo relacionado à tomada de decisão, pelo responsável de uma unidade de nível mais alto, é refletido na função-objetivo e em seu conjunto de decisões viáveis das unidades de níveis mais baixos.

Assim, no processo de tomada de decisão, temos que ter em mente que todas as decisões estão embutidas em um processo sinérgico ditado pelo dinamismo da interação entre as variáveis dos diversos níveis hierárquicos.



$$\begin{aligned} (1) Ax + w &= b, \quad x \geq 0, \quad w \geq 0; \\ (2) A^T u &\geq c, \quad u \geq 0; \\ (3) wu &= 0; \quad e \\ x(A^T u - c) &= 0 \end{aligned}$$

$$\left. \begin{aligned} u^T w &= 0 \\ v^T y &= 0 \end{aligned} \right\} \text{Condições de Complementaridade das Folgas}$$

em que:

$x$ : variável do problema primal;  
 $u$ : variável do problema dual;  
 $w$ : vetor m-dimensional das variáveis de folga do primal; e

$(A^T u - c)$  corresponde ao vetor n-dimensional das variáveis de excesso do dual, representado por  $v = A^T u - c$

Determinando uma solução para as **variáveis** ( $x, u, w, v$ ) relacionadas ao sistema (1), (2) e (3) resolvem os problemas primal e dual originais. Fazendo uma leitura das condições de **KKT**, o sistema representado anteriormente significará a relação:

- (1) corresponde às condições de viabilidade primal do problema;
- (2) corresponde às condições de viabilidade dual do problema;
- (3) corresponde às condições de complementaridade das folgas, ou seja, o vetor das variáveis de folga do problema primal está associado à variável do dual, e o vetor das variáveis de excesso do dual está associado à variável do primal.

A seguir é mostrado o problema equivalente ao Problema de Programação Linear em Dois Níveis (LBP), denotado por (LBP-KKT), como se segue:

$$\begin{aligned} (LBP) \quad \text{Max}_{x,y} \quad & f_1(x,y) = c_1^T x + c_2^T y \\ \text{s.a.} \quad & x = 0, \text{ onde } y \text{ é solução de:} \\ \text{Max}_y \quad & f_2(x,y) = d^T y \\ \text{s.a.} \quad & A_1 x + A_2 y = a \\ & y \geq 0 \end{aligned}$$

em que os vetores  $c_1, x \in \mathbb{R}^n$ ;  $c_2, d, y \in \mathbb{R}^{n_2}$ ;  $a \in \mathbb{R}^m$ ; as matrizes  $A_1 \in \mathbb{R}^{m \times n_1}$ ; e  $A_2 \in \mathbb{R}^{m \times n_2}$ . Baseando-se nas condições de otimalidade de **Karush-Kuhn-Tucker (KKT)**, o Problema de Programação Linear em Dois Níveis pode, então, ser reescrito, substituindo o segundo nível do problema, como se pode observar:

$$\begin{aligned} (LBP-KKT) \quad \text{Max}_{x,y} \quad & c_1^T x + c_2^T y \\ \text{s.a.} \quad & A_1 x + A_2 y + w = a \\ & x \geq 0, \quad y \geq 0, \quad w \geq 0 \end{aligned} \left. \vphantom{\begin{aligned} \text{Max}_{x,y} \\ \text{s.a.} \end{aligned}} \right\} \text{Condições de Viabilidade Primal}$$

$$\begin{aligned} & A_2^T u - v = b \\ & u \geq 0, \quad v \geq 0 \end{aligned} \left. \vphantom{\begin{aligned} A_2^T u - v = b \\ u \geq 0, v \geq 0 \end{aligned}} \right\} \text{Condições de Viabilidade Dual}$$

em que:

$w \in \mathbb{R}^m$  é o vetor de folga da variável primal;  
 $u \in \mathbb{R}^m$  é o vetor da variável do dual; e  
 $v \in \mathbb{R}^{n_2}$  é o vetor de excesso da variável dual.

O problema (**LBP - KKT**), no entanto, transformado nas condições de otimalidade de **KKT**, apresenta uma coleção de restrições não-lineares, correspondente às condições de complementaridade das folgas. Esse fator, no entanto, não se torna interessante no nível de implementação, pois aumentará a complexidade de resolução do problema, uma vez que a restrição não-linear levaria a usar algoritmos diferentes ao Simplex. Com esse intuito, utiliza-se um artifício comum na programação matemática, como abordagem alternativa para o problema (**LBP-KKT**), que seria penalizar essas restrições não-lineares. Assim, as restrições não-lineares são incrementadas para função objetivo, penalizadas com um valor M se o problema for de minimização e - M se o problema for de maximização, sendo este um parâmetro de penalidade, em que  $M \geq 0$ , de forma a buscar uma solução ótima para o problema. Portanto, observa-se que o problema (**LBP - KKT**) dependerá do parâmetro M e irá ter o seguinte aspecto:

$$\begin{aligned} (LBP-KKT(M)) \quad \text{Max}_{x,y} \quad & c_1^T x + c_2^T y - M (u^T w + v^T y) \\ \text{s.a.} \quad & A_1 x + A_2 y + w = a \\ & x \geq 0, \quad y \geq 0, \quad w \geq 0 \\ & A_2^T u - v = b \\ & u \geq 0, \quad v \geq 0 \end{aligned}$$

$\left. \begin{aligned} \text{Condições de} \\ \text{Complementaridade} \\ \text{das Folgas} \end{aligned} \right\}$   
 $\left. \begin{aligned} \text{Condições de} \\ \text{Viabilidade Primal} \end{aligned} \right\}$   
 $\left. \begin{aligned} \text{Condições de} \\ \text{Viabilidade Dual} \end{aligned} \right\}$

Segundo Santos (2002, p. 13),

Os principais algoritmos existentes na literatura para resolver o PLDN (Problema Linear de Dois Níveis), [e] agrupados de acordo com sua idéia principal ou estratégia em comum (...) [são]: (i) Enumeração de Pontos Extremos - [K-ésimo Melhor Vértice; Pesquisa de Bases Ótimas do Seguidor], (ii) Métodos Baseados nas Condições de Otimalidade - [Condições de KKT, MLCP (Maximum Linear Complementarity Problem)], (iii) Métodos de Penalidade - [uso da função do Primeiro Nível para penalizar o gap primal-dual do Segundo Nível], e (vi) Banch-and-Bound- [Fixação de Variáveis Complementares, Eliminação de Variáveis do Seguidor].

Com a finalidade de reduzir a formulação de **LBP-KKT(M)** será utilizada a notação a seguir, conforme Campêlo e Scheimberg (2005, p. 3):

! **As Matrizes em Bloco**

$$A = [A_1 \ A_2 \ I_m] \in \mathbb{R}^{m \times n};$$

$$D = [0 \ -I_{n_2} \ A_2^T] \in \mathbb{R}^{n_2 \times n};$$

$$n = n_1 + n_2 + m;$$

$I_p$  é uma **matriz identidade** ( $p \times p$ ),

$0$  é uma **matriz nula** com dimensão apropriada para cada caso.

! **Os Vetores Fixos e Variáveis:**

$c' = (c_1^T, c_2^T, 0) \in \mathbb{R}^n$  – o vetor da função-objetivo do líder; e

$z' = (x', y', w') \in \mathbb{R}^n$  – o vetor de todas as variáveis relacionadas ao Problema de Programação Linear em Dois Níveis e associadas com a formulação primal padrão do seguidor. Assim, a variável  $x$  está associada ao líder, a variável  $y$  associada ao seguidor e a variável  $w$  pertence às variáveis de folga do seguidor;

$s' = (0, v', u') \in \mathbb{R}^m$  – representa as variáveis ligadas ao problema dual do seguidor, sendo  $u$  a variável da formulação dual do seguidor e  $v$ , a variável de excesso do dual.

! **Os Conjuntos Viáveis**

$$Z = \{ z \in \mathbb{R}^n : Az = a, z \geq 0 \}$$

$$S = \{ s \in \mathbb{R}^m : Ds = d, s \geq 0 \}$$

Assim, o Problema de Programação Linear em Dois Níveis Penalizado **LBP-KKT(M)**, utilizando-se a notação anterior, será escrito como:

$$P(M) \quad \text{Max } F_M(z, s) = c'z - Ms^Tz$$

$$\text{s.a. } z \in Z, s \in S$$

Baseando-se na definição dos vetores  $z$  e  $s$  e nas matrizes dispostas em blocos, é definido o que é um Ponto de Equilíbrio. Campêlo e Scheimberg (2003) definiram um Ponto de Equilíbrio do Problema Penalizado **P(M)** como um ponto  $(z', s')$  para o qual existe  $M' \geq 0$  tal que para cada  $M \geq M'$  se verifica a seguinte relação:

$$\text{Max} \{ F_M(z', s) : s \in S \} = F_M(z', s') = \text{Max} \{ F_M(z, s') : z \in Z \}$$

em que:

$$F_M(z, s) = c'z - Ms^Tz \text{ – Função-Objetivo do Problema } P(M).$$

Da definição de Ponto de Equilíbrio, podem-se identificar dois Problemas de Programação Linear:

$$\left. \begin{array}{l} P(M; s') \quad \text{Max } c'z - Ms^Tz \\ \text{s.a. } z \in Z = \\ \{ z \in \mathbb{R}^n, z \geq 0 : Az = a \} \end{array} \right\} \begin{array}{l} \text{Fixado} \\ (0, v', u') = s' \end{array}$$

$$\left. \begin{array}{l} P(M; z') \quad \text{Max } c'z' - Mz^T s \\ \text{s.a. } s \in S = \\ \{ s \in \mathbb{R}^m, s \geq 0 : Ds = d \} \end{array} \right\} \begin{array}{l} \text{Fixado} \\ (x', y', w') = z' \end{array}$$

Para  $x', y'$  e  $w'$  fixos, tem-se, assim, a função linear  $c_1^T x' + c_2^T y' - M(u^T w' + v^T y')$  nas variáveis  $u$  e  $v$ . E, de forma análoga, fixando  $u'$  e  $v'$ , tem-se a função  $c_1^T x + c_2^T y - M(u^T w + v^T y)$  linear em  $x, y$  e  $w$ . Observa-se também que, em relação a **P(M; s')** e quando o valor de  $c'z$  é máximo, o valor de  $-Ms^Tz$  é mínimo, e de forma análoga em relação a **P(M; z')**, quando o valor de  $c'z'$  é máximo e o valor de  $-Mz^T s$  é mínimo. Dessa forma, se  $(z', s')$  é um Ponto de Equilíbrio do Problema Penalizado **P(M)**, então se verifica também:

$$0 = \text{Min} \{ z^T s : s \in S \} = \text{Min} \{ s^T z : z \in Z \} = s^T z'$$

Em consequência, observa-se que, se  $(z', s')$  é um Ponto de Equilíbrio do Problema de Programação Linear em Dois Níveis Penalizado **P(M)**, a condição de complementaridade  $s^T z' = 0$  será satisfeita.

Baseando-se nas observações expostas anteriormente, o algoritmo proposto por Campêlo e Scheimberg (2005, p. 5) para encontrar um Ponto de Equilíbrio do Problema Penalizado **P(M)** é demonstrado como se segue:

**Passo 0:** Se  $Z \times S = \emptyset$ , então o Problema de Programação Linear é impossível. Caso contrário, se  $Z \times S \neq \emptyset$ , escolher um  $z^0 \in Z$ .

**Passo 1:** Resolva o problema **P(M, z<sup>0</sup>)** para encontrar uma solução  $s^*$ , através do Método Simplex.

**Passo 2:** Resolva o problema **P(M, s\*)**. Se este problema paramétrico é ilimitado, então o **P(M)** é ilimitado para todos os  $M \geq 0$ , portanto, o Problema de Programação em Dois Níveis também é ilimitado. Em caso contrário, calcular a solução  $z^*$ . O ponto  $(z^*, s^*)$  é um Ponto de Equilíbrio.

## 2.4. O Problema Linear de Dois Níveis Abordado

O Problema de Programação Linear de Dois Níveis que será utilizado para a aplicação do algoritmo proposto é um caso especial de PLDN que não apresenta restrições apenas no segundo nível ao qual irá denominar (PLDNP), cuja formulação é a seguinte:

$$\text{(PLDNP) } \text{Min}_{x,y} f_1(x,y) = c_1^T x + c_2^T y$$

**s.a.**  $x \geq 0, y$  solução de

$$\text{Min}_y f_2(x,y) = d_2^T y$$

$$\text{s.a. } A_1 x + A_2 y \leq a$$

$$y \geq 0$$

Como todo PLDN, sua resolução passa por definir um problema penalizado equivalente de um nível, ao qual é obtida aplicando as condições de KKT ao problema de PL do seguidor, cuja formulação é a que se segue:

$$P_M(x,y,u) \text{ Min}_{x,y} f_1(x,y) = c_1^T x + c_2^T y + M(d_2^T y - (-a + A_1 x)^T u)$$

$$\text{s.a. } A_1 x + A_2 y \leq a$$

$$A^T u \geq -d_2$$

$$x,y,u \geq 0$$

Definidas as restrições no formato-padrão, obtêm-se as variáveis de folga do problema de PL do seguidor, do primal e do dual. Substituindo as variáveis  $(x^T, y^T, w^T)$  por  $z^T$  e  $(0, v^T, u^T)$  por  $s^T$ , o problema  $P_M(x, y, u)$  é escrito desta forma:

$$\text{Min } P_M(z,s) = c^T z + M s^T z$$

$$\text{s.a. } Az = a,$$

$$Ds = d,$$

$$z^T \geq 0, s^T \geq 0,$$

em que  $A = [A_1 \ A_2 \ I_{m_2}] \in \mathcal{R}^{m \times n}$ ,  $D = [0 \ -I_{n_2} \ A_2^T] \in \mathcal{R}^{n \times n}$ ,  $c^T = (c_1^T, c_2^T, 0) \in \mathcal{R}^n$ , respectivamente.

### 3. O ALGORITMO IDENTIFICADOR DE VÉRTICES VIÁVEIS PARA PROBLEMA LINEAR DE DOIS NÍVEIS (AIVV- PLDNP)

O algoritmo implementado está inspirado no Algoritmo de Pontos de Equilíbrio, proposto por Campêlo e Scheimberg (2005, p. 5). Segundo Campêlo (1999, p. 5), "este é um procedimento usado para encontrar candidatos à uma solução de um Problema Linear de Dois Níveis, sem restrições no primeiro nível do problema".

Em síntese, o procedimento do algoritmo consiste

em formular o Problema de Programação Linear em Dois Níveis, como um problema penalizado equivalente, depois de aplicadas as condições otimalidade de Karush-Kuhn-Tucker (KKT) no problema do segundo nível. Nesse problema penalizado,  $\text{Min PM}(z^o, s)$ , sua função-objetivo está formada pela função-objetivo do líder e a parcela penalizada da complementaridade das folgas, sendo suas restrições as condições KKT do problema do seguidor. Fixadas as variáveis do problema primal do seguidor ou as variáveis do problema dual do seguidor, são gerados dois problemas penalizados lineares que definem um Ponto de Equilíbrio para  $M > M^o$ , quando se verifica que:

$$\text{Min PM}(z^o, s) = \text{Min PM}(z, s^o) = \text{PM}(z^o, s^o)$$

O algoritmo AIVV, que é apresentado neste trabalho, não completa a penalização, pois se trabalha com um valor de penalidade fixa e se repete a rotina até encontrar cada um dos pontos extremos da região viável do problema do seguidor, satisfazendo a seguinte condição:

$$z^o = \text{argmin} \{ \text{PM}(z, s^o) \}, \text{ em que}$$

$$s^o = \text{argmin} \{ \text{PM}(z^o, s) \}$$

A implementação desse algoritmo é um procedimento de grande complexidade computacional pelo fato de enumerar todos os vértices viáveis da região viável do problema do seguidor e, posteriormente, faz-se um teste, para verificação, se é ou não um ponto da região viável do PLDNP. O algoritmo se sustenta nos resultados colocados nos Teoremas 1.2.3 e 1.2.4 propostos por Campêlo (1999, p. 17-18):

Teorema 1.2.3: Se o PLDN tem solução, pelo menos uma delas é atingida em um ponto extremo do conjunto viável do Problema de Dois Níveis (...)

Teorema 1.2.4: Todo ponto extremo do conjunto viável do Problema de Dois Níveis é um vértice do conjunto viável relaxado (...). Problema relaxado é o problema definido pela função objetivo do líder e todas as restrições do problema PLDN.

Corolário 1.2.2: Se PLDN tem solução, pelo menos uma delas é atingida em um vértice do conjunto viável relaxado.

Assim, pode-se observar que o Algoritmo Identificador de Vértices Viáveis (AIVV) tem convergência finita, uma vez que o conjunto viável do problema relaxado tem definição de finitos números de restrições lineares. Porém, pode ser um número muito grande, aumentando, assim, a complexidade do problema. No entanto, para problemas em duas dimensões fornece grande ajuda para a compreensão das dificuldades de uma resolução do problema de PLDN. O fluxograma do Algoritmo Identificador de Vértices Viáveis para o Problema de Programação Linear em Dois Níveis (AIVV), proposto por Oliveira (2005, p. 146), é descrito a seguir:

Algoritmo Identificador de Vértices Viáveis para Solução do PLDN

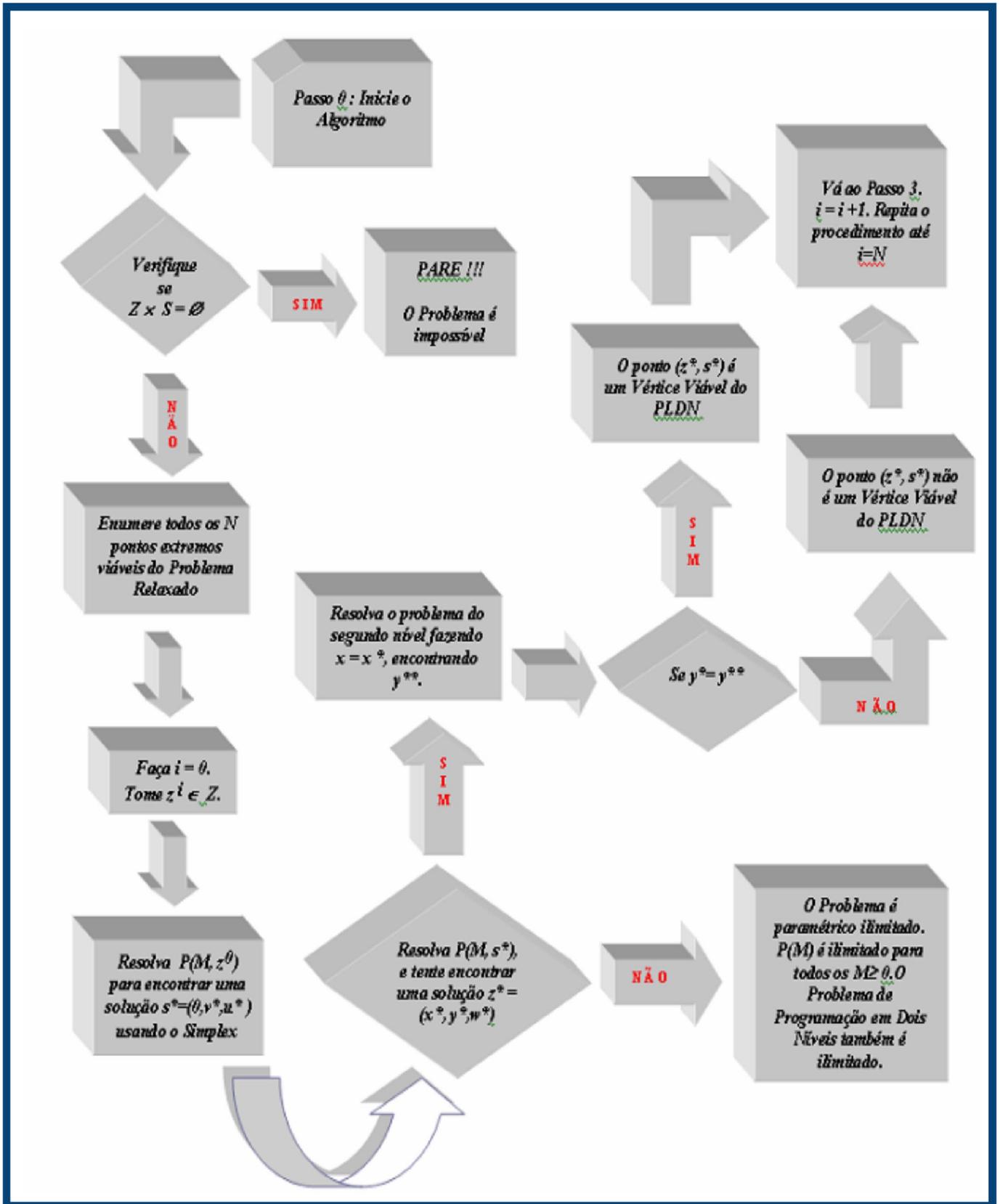


Figura 1 - Fluxograma do Algoritmo Identificador de Vértices Viáveis (AIVV).

Fonte: OLIVEIRA, 2005, p. 146.

## 4. IMPLEMENTAÇÃO COMPUTACIONAL DO ALGORITMO IDENTIFICADOR DE VÉRTICES VIÁVEIS (AIVV) EM $\mathcal{R}^2$

A abrangência ou amplitude da implementação do Algoritmo Identificador de Vértices Viáveis (AIVV) foca alguns aspectos, ao qual limitarão tanto nossos problemas-testes quanto suas características. Os aspectos são os seguintes:

- i) Serão testados problemas que contenham apenas duas variáveis ( $\mathcal{R}^2$ ).
- ii) Os problemas abordados serão lineares, tanto sua função objetivo quanto as suas restrições, e terão apenas dois níveis hierárquicos; assim, o foco de otimização será a Programação Linear em Dois Níveis.
- iii) Os exemplos trabalhados terão uma solução finita, ou seja, terão a região viável possível determinada e limitada.
- iv) A região viável das restrições do problema relaxado será convexa e não terá soluções básicas degeneradas.
- v) Os problemas abordados serão de pequeno porte, apenas para efeitos didático e acadêmico.
- vi) O código-fonte da implementação do Algoritmo Identificado de Vértices Viáveis irá focar apenas problemas de duas variáveis, para efeito de representação geométrica visível.

Baseando-se assim em tais preceitos, a implementação desenvolvida sobre os seguintes problemas-teste é descrita como se segue:

┆ Problema-Teste 1 - Proposto por Dantas (1998, p. 13):

$$\text{Min}_{x,y} \quad x + 4y$$

$$\text{s.a: Min}_y - y$$

$$\text{s.a} \quad -x - y = -8$$

$$-3x + 2y = 6$$

$$3x + 4y = 48$$

$$2x - 5y = 9$$

$$3x + 2y = 36$$

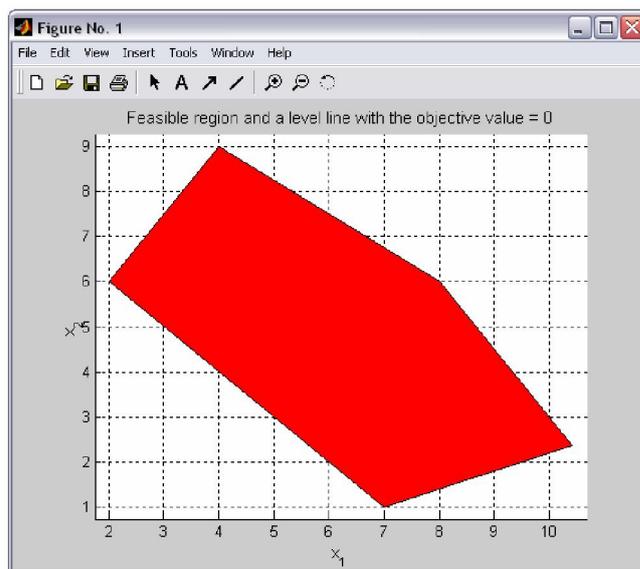


Figura 2 - Região viável do problema relaxado.

Fonte: OLIVEIRA, 2005, p. 149.

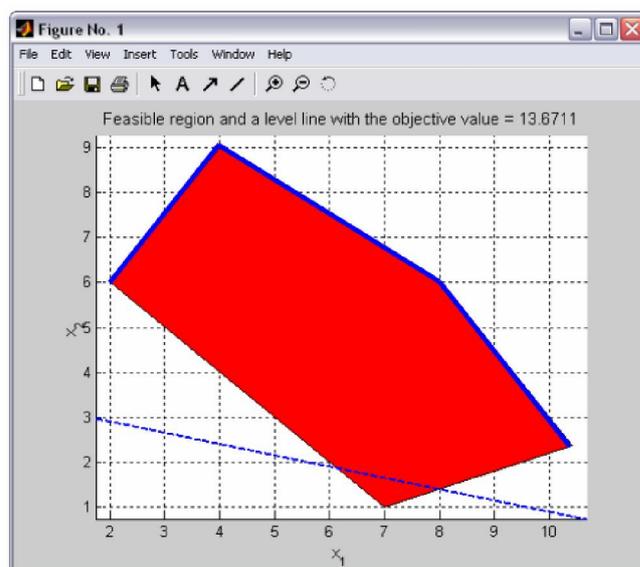


Figura 3 - Região viável do PLDNP.

Fonte: OLIVEIRA, 2005, p. 150

==> A Matriz das Soluções Possíveis do Problema de Dois Níveis é:

vert =

2.0000	4.0000	7.0000	8.0000	10.4211
6.0000	9.0000	1.0000	6.0000	2.3684
0	5.0000	0	6.0000	4.7895
0	0	25.0000	18.0000	32.5263
18.0000	0	23.0000	0	7.2632
35.0000	46.0000	0	23.0000	0
18.0000	6.0000	13.0000	0	0

ITERAÇÃO “1”

==> Não se está em um vértice viável!!!

ans = Parte-se do ponto: (2, 6)

ans = Chega-se ao ponto (7, 1), que não é um vértice viável

ans = O valor da função-objetivo no primeiro nível é: 11

ans = O valor da função-objetivo no segundo nível é: -1

ITERAÇÃO “2”

==> Está-se em um vértice viável!!!

ans = Parte-se do ponto: (4, 9)

ans = Chega-se ao ponto (2, 6), que é um vértice viável

ans = O valor da função-objetivo no primeiro nível é: 26

ans = O valor da função-objetivo no segundo nível é: -6

ITERAÇÃO “3”

==> Está-se em um vértice viável!!!

ans = Parte-se do ponto: (7, 1)

ans = Chega-se ao ponto (8, 6), que é um vértice viável:

ans = O valor da função-objetivo no primeiro nível é: 32

ans = O valor da função-objetivo no segundo nível é: -6

ITERAÇÃO “4”

==> Está-se em um vértice viável!!!

ans = Parte-se do ponto: (8.000000e+000, 6)

ans = Chega-se ao ponto (8, 6), que é um vértice viável:

ans = O valor da função-objetivo no primeiro nível é: 32

ans = O valor da função-objetivo no segundo nível é: -6

ITERAÇÃO “5”

==> Está-se em um vértice viável!!!

ans = Parte-se do ponto: (10.4211, 2.3684)

ans = Chega-se ao ponto (10.4211, 2.3684), que é um vértice viável

ans = O valor da função-objetivo no primeiro nível é: 19.8947

ans = O valor da função-objetivo no segundo nível é: -2.3684

Dessa forma, observa-se que esse algoritmo obtém todos os pontos extremos da região viável do problema estendido em primeira mão, com a ressalva de que, após as rodadas e fazendo uso de seus procedimentos, ele obtém os pontos da região viável do PLDNP, os quais, depois de uma análise refinada, possa servir de ponto de partida para achar uma solução para o Problema Linear de Dois Níveis. Verifica-se que o ponto (10,4211; 2,3684), após uma análise detalhada do problema, é o ótimo global do problema de dois níveis, pois as funções-objetivo do primeiro e do segundo nível encontram seu valor mínimo com esse respectivo ponto.

1 Problema Teste 2 - Proposto por Oliveira (2005, p. 164):

$$\begin{aligned} \text{Min}_{x,y} \quad & 2x - y \\ \text{s.a.} \quad & \text{Min}_y \quad x - 10y \\ & x - 2y = -6 \\ & 3x - 4y = 18 \\ & 4x + y = 43 \\ & x + 2y = 23 \\ & -x + 2y = 13 \\ & -3x + 2y = 7 \\ & -3x - y = -8 \end{aligned}$$

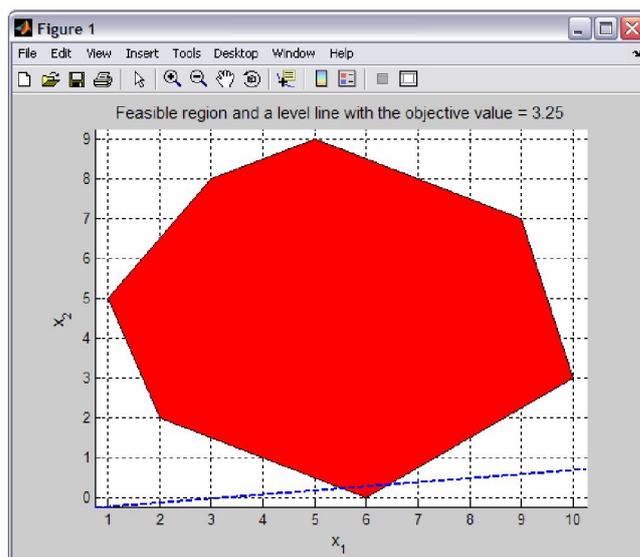


Figura 4 - Região viável do problema relaxado

Fonte: OLIVEIRA, 2005, p. 164.

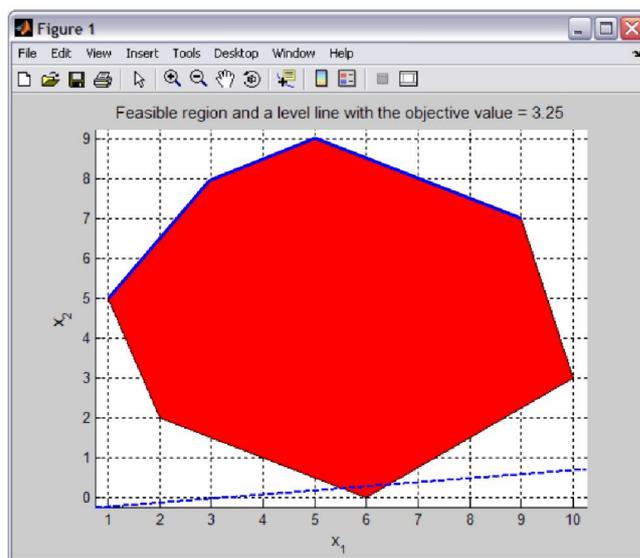


Figura 5 - Região viável do PLDNP

Fonte: OLIVEIRA, 2005, p. 165.

==> A Matriz das Soluções Possíveis do Problema de Dois Níveis é:

vert =

1.0000	2.0000	3.0000	5.0000	6.0000	9.0000	10.0000
5.0000	2.0000	8.0000	9.0000	0	7.0000	3.0000
5.0000	0	13.0000	17.0000	0	17.0000	10.0000
35.0000	20.0000	41.0000	39.0000	0	19.0000	0
34.0000	33.0000	23.0000	14.0000	19.0000	0	0
12.0000	17.0000	4.0000	0	17.0000	0	7.0000
4.0000	11.0000	0	0	19.0000	8.0000	17.0000
0	9.0000	0	4.0000	25.0000	20.0000	31.0000
0	0	9.0000	16.0000	10.0000	26.0000	25.0000

ITERAÇÃO “1”

==> Está-se em um vértice viável!!!

ans = Parte-se do ponto: (1 , 5)

ans = Chega-se ao ponto (1 , 5), que é um vértice viável:

ans = O valor da função-objetivo no primeiro nível é: -3

ans = O valor da função-objetivo no segundo nível é: -49

ITERAÇÃO “2”

==> Está-se em um vértice viável!!!

ans = Parte-se do ponto: (2 , 2)

ans = Chega-se ao ponto (1 , 5), que é um vértice viável:

ans = O valor da função-objetivo no primeiro nível é: -3

ans = O valor da função-objetivo no segundo nível é: -49

ITERAÇÃO “3”

==> Está-se em um vértice viável!!!

ans = Parte-se do ponto: (3 , 8)

ans = Chega-se ao ponto (3 , 8), que é um vértice viável:

ans = O valor da função-objetivo no primeiro nível é: -2

ans = O valor da função-objetivo no segundo nível é: -77

ITERAÇÃO “4”

==> Está-se em um vértice viável!!!

ans = Parte-se do ponto: (5 , 9)

ans = Chega-se ao ponto (5 , 9), que é um vértice viável:

ans = O valor da função-objetivo no primeiro nível é: 1

ans = O valor da função-objetivo no segundo nível é: -85

ITERAÇÃO “5”

==> Está-se em um vértice viável!!!

ans = Parte-se do ponto: (6 , 0)

ans = Chega-se ao ponto (5 , 9), que é um vértice viável:

ans = O valor da função-objetivo no primeiro nível é: 1

ans = O valor da função-objetivo no segundo nível é: -85

ITERAÇÃO “6”

==> Está-se em um vértice viável!!!

ans = Parte-se do ponto: (9 , 7)

ans = Chega-se ao ponto (9 , 7), que é um vértice viável:

ans = O valor da função-objetivo no primeiro nível é: 11

ans = O valor da função-objetivo no segundo nível é: -61

ITERAÇÃO “7”

==> Está-se em um vértice viável!!!

ans = Parte-se do ponto: (10 , 3)

ans = Chega-se ao ponto (9 , 7), que é um vértice viável:

ans = O valor da função-objetivo no primeiro nível é: 11

ans = O valor da função-objetivo no segundo nível é: -61

## 5. UM MODELO DE APLICAÇÃO DO PROBLEMA DE PROGRAMAÇÃO LINEAR EM DOIS NÍVEIS

Modelos de dois níveis têm surgido com o intuito de representar os diversos níveis hierárquicos dos processos de produção. Cada nível desses processos possui suas necessidades de uma otimização e controla apenas algumas variáveis específicas, mas que têm seu espaço de decisão total ou parcialmente determinado por outras variáveis de outros níveis. Por exemplo, as cadeias de distribuição de estoques estão dispostas em níveis hierárquicos, formando o conhecido esquema de estoques em múltiplos escalões. Assim, conforme o modelo exposto em Oliveira (2005, p. 134-137), pode-se observar um exemplo de aplicação do Modelo de Programação Linear em Dois Níveis.

### 5.1. Uma Aplicação do Problema Linear de Dois Níveis

Uma organização tem um problema de distribuição de peças. Ela atua num mercado de área geográfica de cobertura extensa e, portanto, deve estabelecer a seguinte estrutura de estoques. Os estoques do Depósito Central (Atacadista) e dos dois Distribuidores (Atacadistas

Regionais) são geridos, atendendo aos clientes com diferenciação cultural por regiões, e, portanto, tem independência em algumas decisões, porém pertencem a uma só organização. Assim, o depósito e os dois distribuidores possuem os respectivos gerentes.

As projeções de demanda dos produtos em cada região são primeiramente feitas pelo gerente dos distribuidores e repassadas para o gerente do Depósito Central. Cabe a esse gerente cobrir a oferta da demanda estipulada pelos dois distribuidores. Assim, o modelo matemático do problema é feito colocando no primeiro nível a decisão dos distribuidores regionais e no segundo nível, a decisão do Depósito Central. É de interesse do Depósito Central maximizar sua quantidade de estoque em relação à quantidade demandada nos distribuidores, enquanto estes visam à minimização dos custos de transporte e dos custos de estoque.

Como informação adicional, o pagamento do custo de transporte do Depósito ao Distribuidor  $j$  é de responsabilidade do próprio distribuidor, além de existirem dois tipos de transportes, um de custo normal e outro de custo emergencial, o qual tomará menos tempo.

### 5.1.1. Variáveis para Calcular os Custos de Distribuição

à  $x_{DDj}$ : quantidade de mercadorias em  $m^3$  transportada do Depósito para o Distribuidor  $j$ , para  $j = 1, 2$ . Há duas categorias de transporte:

à  $xn_{DDj}$ : quantidade de mercadorias em  $m^3$  transportada do Depósito no modo normal; e

à  $xe_{DDj}$ : quantidade de mercadorias em  $m^3$  transportada do Depósito no modo emergencial. Daí, têm:

$$x_{DDj} = xn_{DDj} + xe_{DDj}$$

$CL_j$ : custo logístico unitário (em reais/ $m^3$ ) de transporte do Depósito para o Distribuidor  $j$ , para  $j = 1, 2$ . O distribuidor é quem paga esses custos logísticos. Os custos logísticos são constituídos pela mão-de-obra de embarque, mão-de-obra no decorrer do transporte, armazenamento da carga durante o transporte e as despesas de entrega. Há duas categorias de custo logístico unitário de transporte do Depósito para o Distribuidor  $j$ :  $CLN_j$ , o custo Logístico Normal e o  $CLE_j$ , o custo Logístico Especializado. Daí, tem-se:  $CL_j = CLN_j + CLE_j$

### 5.1.2. Variáveis para calcular Custos de Armazenagem, Quantidade e Demanda de Estoque no Depósito e no Distribuidor

à  $y_{DIj}$ : estoque em  $m^3$  existente no Distribuidor  $j$ , para  $j = 1, 2$

$CDI_j$ : custo unitário de manutenção do estoque no Distribuidor  $j$ , para  $j = 1, 2$ , no período considerado (em reais/ $m^3$ ).

à  $y_{DE}$ : estoque em  $m^3$  existente no Depósito.

$CDE$ : custo unitário de manutenção do estoque no Depósito, no período considerado (reais/ $m^3$ ).

$CEDE$ : custo máximo de estoque aceitável no Depósito, no período considerado.

$QDE$ : quantidade máxima de estoque aceitável no Depósito.

$CEDI_j$ : custo máximo de estoque aceitável no Distribuidor  $j$ , para  $j = 1, 2$ , no período considerado.

$QDI_j$ : quantidade máxima de estoque aceitável no Distribuidor  $j$ ,  $j = 1, 2$ .

$QDEDI_j$ : quantidade máxima de mercadorias que pode ser transportada do Depósito para o Distribuidor  $j$  para  $j = 1, 2$ .

$CDD_j$ : custo máximo de transporte do Depósito para o Distribuidor  $j$  para  $j = 1, 2$ .

$Di_j$ : demanda existente no Distribuidor  $j$ , para  $j = 1, 2$ .

$De$ : demanda existente no Depósito.

### 5.1.3. Funções-Objetivo do Modelo

Considerando que as seguintes funções

$X = \sum_{j=1,2} [CLN_j * xn_{DDj} + CLE_j * xe_{DDj}]$ : Custos logísticos,

$Y = \sum_{j=1,2} [CDI_j * y_{DIj}]$ : Custos de armazenagem e

$y_{DE}$ : Quantidade de estoque no Depósito

A formulação das funções-objetivo de 1º e 2º níveis do modelo para o problema são, respectivamente:

$$\begin{aligned} & \text{(Distribuidores)} \quad \text{Min } X + Y \\ & \text{(Depósito)} \quad \text{Max } y_{DE} \end{aligned}$$

## 5.1.4. Restrições do Problema

**1. Custo de Estoques no Depósito:** O custo de estoques no Depósito é igual ao estoque no Depósito, mais o que será expedido do Depósito para o Distribuidor  $j$  (quantidade de mercadorias transportada), nas duas modalidades de transporte, caso haja as duas:

$$CDE * y_{DE} + [CLN_j * x_{DDj} + CLE_j * x_{e_{DDj}}] \leq CEDE; \text{ para } j = 1,2$$

**2 Custo de Transporte (Logística):** A quantidade de mercadorias transportada do Depósito para cada Distribuidor  $j$ , vezes o seu custo logístico, nas duas modalidades de transporte, caso haja, não pode ser maior que a capacidade de transporte para esse Distribuidor  $j$ :

$$CLN_j * x_{DDj} + CLE_j * x_{e_{DDj}} \leq CDD_j; \text{ para } j = 1,2$$

**3 Custo de Estoques nos Distribuidores:** O custo de estoque em cada Distribuidor  $j$  é igual ao estoque no Distribuidor  $j$ , vezes o seu custo, mais a quantidade que foi expedida pelo Depósito para o Distribuidor  $j$ , nas duas modalidades de transporte, caso haja, vezes o seu custo:

$$CDI_j * y_{DIj} + CLN_j * x_{DDj} + CLE_j * x_{e_{DDj}} \leq CEDI_j; \text{ para } j = 1,2$$

**4 Quantidade Demandada nos Distribuidores:** A quantidade demandada no Distribuidor  $j$  deve ser inferior ou igual à quantidade de estoque no Distribuidor  $j$ , ou seja, a quantidade no Distribuidor  $j$ , mais a quantidade de mercadoria transportada do Depósito para o Distribuidor  $j$ , nas duas modalidades de transporte, caso haja as duas:

$$DI_j \leq y_{DIj} + x_{DDj} + x_{e_{DDj}}; \text{ para } j = 1,2$$

**5 Quantidade Demandada no Depósito:** A quantidade demandada do Depósito deve ser inferior ou igual ao estoque no Depósito, mais o que será expedido do Depósito para o Distribuidor  $j$  (quantidade de mercadorias transportada), nas duas modalidades de transporte, caso haja as duas:

$$DE \leq y_{DE} + x_{DDj} + x_{e_{DDj}}; \text{ para } j = 1,2$$

## 5.1.5. Formulação Completa do Modelo

$$\begin{aligned} & \text{Min } X + Y \\ & \text{s.a. } \text{Max } y_{DE} \\ & \text{s.a.} \end{aligned}$$

$$CDE * y_{DE} + [CLN_j * x_{DDj} + CLE_j * x_{e_{DDj}}] \leq CEDE; j = 1,2$$

$$CLN_j * x_{DDj} + CLE_j * x_{e_{DDj}} \leq CDD_j; j = 1,2$$

$$CDI_j * y_{DIj} + CLN_j * x_{DDj} + CLE_j * x_{e_{DDj}} \leq CEDI_j; j = 1,2$$

$$DI_j \leq y_{DIj} + x_{DDj} + x_{e_{DDj}}; j = 1,2$$

$$DE \leq y_{DE} + x_{DDj} + x_{e_{DDj}}; j = 1,2$$

O problema é modelado por sete variáveis e 10 restrições, em que somam os custos:

$$X = \sum_{j=1,2} [CLN_j * x_{DDj} + CLE_j * x_{e_{DDj}}] \text{ e } Y = \sum_{j=1,2} [CDI_j * y_{DIj}].$$

Dessa forma, pode-se observar que o Modelo de Programação Linear em Dois Níveis se torna um modelo bastante representativo quando se trata de processo de decisões em que são baseados na hierarquia para determinação de um ótimo global de um sistema interligado. E maiores ganhos que podem ser obtidos através da integração efetiva dos elementos da cadeia produtiva, com otimização global dos custos e de desempenho, sendo estes mais expressivos do que a soma dos possíveis ganhos individuais de cada participante quando atuando separadamente, focalizando, assim, uma visão sinérgica de todos os componentes da estrutura hierárquica da cadeia de distribuição.

## 6. CONCLUSÃO

Conclui-se, então, que um modelo de Programação Linear em Dois Níveis tem características que representam muito bem os processos encadeados ou multiescalonados. As soluções calculadas para esses modelos indicam excelentes resultados, além de refletir melhor a presença de uma hierarquia organizacional. Seu foco de otimização é de natureza holística, isto é, busca um valor ótimo que seja a melhor a toda uma Cadeia Sinérgica Empresarial. Assim, buscando o ótimo em uma hierarquia organizacional, faz que as empresas estejam cada vez mais engajadas em seus propósitos, além de motivadas, proporcionando, assim, satisfação mútua, tanto do público fornecedor, do produtor, quanto do consumidor.

O Algoritmo Identificador de Vértices Viáveis (AIVV) apresentado foi inspirado no método de Pontos de Equilíbrio proposto por Campêlo e Scheimberg (2005, p. 5), e é um procedimento computacional que revisa todo o poliedro de pontos viáveis Problema Linear de Dois

Níveis sem Restrições no Primeiro Nível (PLDNP), de forma que mostra o conjunto viável do Problema de Dois Níveis, ou seja, explicita os candidatos à solução ótima global do problema. Deve-se notar que o AIVV é um procedimento que tem complexidade computacional exponencial, uma vez que fornece todos os vértices viáveis do conjunto de restrições do problema PLDNP, fazendo uma revisão à exaustão. Então, sua complexidade cresce juntamente com o tamanho do problema proposto, ficando inviável sua aplicação em problemas de grande porte, porém se torna bastante útil para a apresentação didática do problema PLDNP, bem como o seu detalhamento com respeito à solução ótima do problema.

## 6. REFERÊNCIAS

- BIALAS, Wayne F.; KARWAN, Mark H. **Mathematical methods for multilevel planning**: the operations research program Department of Industrial Engineering State University of New York At Buffalo. Research Report No. 79-2, Buffalo, New York. February, 1979. Disponível em: <<http://www.acsu.buffalo.edu/~bialas/public/pub/Papers/papbk79.pdf>>. Acesso em: 10 jun. 2007.
- BIALAS, Wayne F. University at Buffalo: Industrial Engineering. Multilevel Optimization, and Multilevel Programming and Bilevel Programming. Criado: 10 ago. 2004. Disponível em: <[http://www.acsu.buffalo.edu/~bialas/info/whatis\\_x.html](http://www.acsu.buffalo.edu/~bialas/info/whatis_x.html)>. Acesso em: 11 maio 2007.
- \_\_\_\_\_. Multilevel optimization: some ideas for further study in multilevel optimization. University at Buffalo: Department of Industrial Engineering. 2005. Disponível em: <[http://www.acsu.buffalo.edu/~bialas/info/multilevel\\_ideas.html](http://www.acsu.buffalo.edu/~bialas/info/multilevel_ideas.html)>. Acesso em: 11 maio 2007.
- \_\_\_\_\_. Decision systems group. University at Buffalo: Department of Industrial Engineering. 2005. Disponível em: <[http://www.acsu.buffalo.edu/~bialas/info/decisiongroup\\_x.html](http://www.acsu.buffalo.edu/~bialas/info/decisiongroup_x.html)>. Acesso em: 11 maio 2007.
- CAMPÊLO, Manoel Bezerra Neto. **Programação Linear em Dois Níveis: Uma abordagem teórica e computacional**. Rio de Janeiro: COPPE – UERJ, 1999. Tese (Doutorado em Engenharia de Sistemas e Computação) – Universidade Federal do Rio de Janeiro, Rio de Janeiro.
- CAMPÊLO, Manoel Bezerra Neto; SCHEIMBERG, Suzana Makler. **A simplex approach for finding local solutions of a linear bilevel program by equilibrium points**. Annals of Operations Research, Kluwer, 2005. Disponível em: <<http://www.lia.ufc.br/~mcampelo/mcampeloPublic.htm>>. Acesso em: 18 maio 2007.
- GAVIRA, Muriel de Oliveira. **Simulação computacional como uma ferramenta de aquisição de conhecimento**. São Paulo: USP, 2007. Dissertação (Mestrado em Engenharia de Produção) – Escola de Engenharia de São Carlos. Universidade de São Paulo, 2003. Disponível em: <<http://www.teses.usp.br/teses/disponiveis/18/18140/tde-20052003-004345/publico/Gavira1.pdf>>. Acesso em: 24 maio 2007.
- JORDÁN, Gladys Castillo. Investigaç o operacional e otimizaç o, 2002. Disponível em: <<http://www2.mat.ua.pt/io/acetatos.htm>>. Acesso em: 11 maio 2007.
- LAVAL Silva, Alexandre. **Modelo de programação linear em dois níveis para otimização de estoques de sobressalentes**. Rio de Janeiro: Ministério da Defesa do Exército Brasileiro, 2003. Dissertação (Mestrado em Sistemas e Computação) – Ministério da Defesa Exército Brasileiro – Secretaria de Ciência e Tecnologia, Instituto Militar de Engenharia, Rio de Janeiro.
- MARCOTTE, Patrice. Bilevel programming: algorithms. Universidade de Montreal, Canadá, 1999. Disponível em: <[www.iro.umontreal.ca/~marcotte/ARTIPS/marcot3.ps](http://www.iro.umontreal.ca/~marcotte/ARTIPS/marcot3.ps)> . Acesso em: 09 jun. 2007.
- NEVES, Julio Cezar Silva. **Modelo de programação linear binível para a alocação dinâmica de veículos**. Rio de Janeiro: COOPPE/UFRJ, 2002. Tese (Doutorado em Sistemas e Computação) – Coordenação dos Programas de Pós-Graduação de Engenharia da Universidade Federal do Rio de Janeiro, Rio de Janeiro.
- OLIVEIRA, Érik da Silva. **Uma abordagem da pesquisa operacional aplicada à gestão de Materiais e a Logística**: contribuição para o ensino do Modelo de Programação Linear em Dois Níveis. Campos dos Goytacazes, RJ: UNIOESTE, 2005. Dissertação (Mestrado em Engenharia de Produção) – LEPROD – Laboratório de Engenharia da Produção – Programa de Pós-Graduação em Engenharia de Produção, Universidade Estadual do Norte Fluminense Darcy Ribeiro, Campos dos Goytacazes.
- SANTOS, Carlos André Martins dos. **Programação em dois níveis aplicada ao estudo da oferta ótima em sistemas termelétricos**. Rio de Janeiro: NCE /IM/UFRJ, 2002. Tese (Doutorado em Engenharia de Sistemas de Computação) – Instituto de Matemática do Núcleo de Computação Eletrônica da Universidade Federal do Rio de Janeiro, Rio de Janeiro.
- VICENTE, Luis N. Bilevel programming: introduction, history and overview. In: FLOUDAS, C. A.; PARDALOS, P. M. (Eds.). **Encyclopedia of optimization**. Dordrecht: Kluwer Academic Publishers, 2001. v. 1, p. 178-180. Disponível em: <<http://www.mat.uc.pt/~lnv/papers/papers.html>>; <[www.mat.uc.pt/~lnv/papers/papers.html](http://www.mat.uc.pt/~lnv/papers/EoO.ps)>; <[www.mat.uc.pt/~lnv/papers/EoO.ps](http://www.mat.uc.pt/~lnv/papers/EoO.ps)>. Acesso em: 09 jul. 2007.