

APLICAÇÃO DO ALGORITMO SIMULATED ANNEALING AO PROBLEMA
DE LOCALIZAÇÃO CAPACITADO, USANDO TESTES DE REDUÇÃO

HUGO ALEJANDRO BRAVO AZLAN

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS PROGRAMAS DE
POS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE
JANEIRO COMO PARTE DOS REQUISITOS NECESSARIOS PARA A OBTENÇÃO DO
GRAU DE DOUTOR EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E
COMPUTAÇÃO.

Aprovada por:



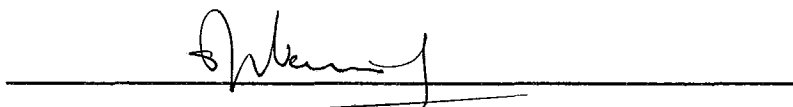
Prof. CLAUDIO THOMAS BORNSTEIN, Dr. Rer. Nat



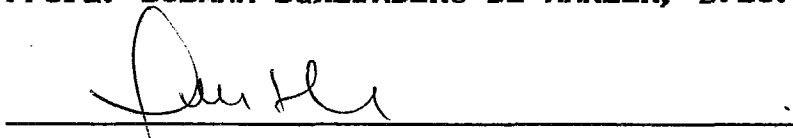
Prof. JAYME LUIZ SZWARCFITER. Ph. D.



Prof. NAIR MARIA MAIA DE ABREU, D. Sc.



Prof. SUSANA SCHEIMBERG DE MAKLER, D. Sc.



Prof. LUIZ SATORU OCHI, D. Sc.

RIO DE JANEIRO, RJ-BRASIL

ABRIL DE 1995

BRAVO AZLAN, HUGO ALEJANDRO

APLICAÇÃO DO ALGORITMO SIMULATED ANNEALING AO PROBLEMA DE LOCALIZAÇÃO CAPACITADO, USANDO TESTES DE REDUÇÃO (Rio de Janeiro), 1995.

VI, 151p. 29.7 cm (COPPE/UFRJ, D.Sc., Engenharia de Sistemas e Computação, 1995)

Tese- Universidade Federal do Rio de Janeiro, COPPE.

1. Problema de Localização Capacitado, Testes de redução, Algoritmo Simulated Annealing.

I. COPPE/UFRJ II. Título (Série)

AOS MEUS PAIS E IRMAOS

AOS MEUS FILHOS PIA E CHRISTIAN

AGRADECIMENTOS

Ao Professor CLAUDIO T. BORNSTEIN pela sua orientação e estímulo durante o desenvolvimento desta tese e nas atividades do curso.

Aos Professores do Programa de Engenharia de Sistemas e Computação da COPPE, por seu valioso ensino.

Ao Professor NELSON MACULAN F. por seu constante apoio e preocupação por seus alunos.

Aos Funcionários Administrativos da COPPE, pela constante colaboração no meu curso.

A CAPES pelo seu apoio financeiro durante meus estudos.

Ao Departamento de Matemática da Universidade de Tarapacá, pelas facilidades dadas para a conclusão da tese.

RESUMO DA TESE APRESENTADA À COPPE/UFRJ COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS (D.Sc.)

APLICAÇÃO DO ALGORITMO SIMULATED ANNEALING AO PROBLEMA DE LOCALIZAÇÃO CAPACITADO, USANDO TESTES DE REDUÇÃO

Hugo Alejandro Bravo Azlan

Abril de 1995

ORIENTADOR: Prof. Claudio Thomas Bornstein

PROGRAMA : Engenharia de Sistemas e Computação

Nesta tese se apresentam novos resultados dos testes de redução do Problema de Localização Capacitado, baseados em relaxações lagrangeanas.

Adicionalmente se apresenta a aplicação do algoritmo Simulated Annealing ao problema e uma interessante aproximação que permite uma considerável redução no tempo computacional envolvido.

Também são apresentados alguns resultados computacionais tanto na fase de redução como nas próprias heurísticas geradas.

ABSTRACT OF THESIS PRESENTED TO COPPE/UFRJ AS PARTIAL
FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF
SCIENCES (D.Sc.)

AN APPLICATION OF THE SIMULATED ANNEALING ALGORITHM TO THE
CAPACITATED WAREHOUSE LOCATION PROBLEM, USING REDUCTION TESTS

Hugo Alejandro Bravo Azlan

April, 1995

THESIS SUPERVISOR: Prof. Claudio Thomas Bornstein
DEPARTMENT: Systems Engineering and Computer Science

In this thesis we present new results of reduction tests for the Capacitated Warehouse Location Problem, based on lagrangean relaxation.

Furthermore, we present an application of the Simulated Annealing algorithm to the problem and a new approximation scheme that allows an important reduction on computational times.

We also present some computational results of our improvements on reduction tests and those related to the generated heuristics.

INDICE

	Pág
CAPÍTULO I - INTRODUÇÃO	1
CAPÍTULO II - TESTES DE REDUÇÃO PARA O PROBLEMA DE LOCALIZAÇÃO CAPACITADO.....	8
II.1 - Limites inferiores e relaxações lagrangeanas.....	10
II.2 - Limites Superiores.....	29
II.3 - Problemas Testes.....	34
II.4 - Experiência Computacional.....	35
CAPÍTULO III - ELEMENTOS DE CADEIAS DE MARKOV.....	39
CAPÍTULO IV - ASPECTOS GERAIS DO ALGORITMO SIMULATED ANNEALING	
IV.1 - Introdução.....	86
IV.2 - Fundamentos.....	88
IV.3 - Formulação Matemática.....	92
IV.4 - Convergência.....	98
IV.5 - Resumo.....	112
CAPÍTULO V - APLICAÇÃO DO ALGORITMO SIMULATED ANNEALING AO PROBLEMA DE LOCALIZAÇÃO CAPACITADO.....	113
V.1 - O Modelo Matemático.....	113
V.2 - Estrutura de Vizinhança.....	114
V.3 - Irredutibilidade.....	115
V.4 - Criterio de Aceitação.....	116
V.5 - Algoritmo S.A. aplicado ao CWLP.....	119
V.6 - A Temperatura Inicial.....	121
V.7.- Condição de Equilíbrio.....	127
V.8.- Esquema de Redução de Temperatura.....	128
V.9.- Criterio de Parada.....	128
V.10 - Implementação e Resultados Computacionais.....	129
CAPÍTULO VI - CONCLUSÕES.....	136
REFERENCIAS BIBLIOGRAFICAS.....	138
APÊNDICE.....	144

CAPITULO 1

INTRODUÇÃO

Nesta tese trabalhamos em cima do PROBLEMA DE LOCALIZAÇÃO CAPACITADO, que simbolizaremos por CWLP e que pode ser definido da seguinte maneira: sejam $I = \{1, 2, \dots, m\}$ o conjunto de localizações potenciais para a instalação de armazens (ou plantas ou facilidades), e $J = \{1, 2, \dots, n\}$ o conjunto de centros de demanda ou clientes. Cada armazém $i \in I$ tem associados uma capacidade (de fornecimento) máxima a_i e um custo fixo f_i de instalação ou ativação, enquanto que cada centro $j \in J$ tem associada uma demanda b_j relativa a um determinado produto. Seja também, c_{ij} o custo unitário de transporte desde o armazém i ao centro de demanda j .

Considerando que a demanda total $\sum_{j \in J} b_j$ deverá ser totalmente satisfeita, o CWLP consiste em determinar quantas e quais facilidades serão instaladas, bem como quais centros de demanda e a forma que serão atendidos por cada facilidade, de modo a minimizar o custo total que é a soma dos custos fixos de instalação e dos custos de atendimento (transporte) a cada centro de demanda.

Matematicamente, o CWLP pode ser formulado como:

$$(P) \left\{ \begin{array}{l} \text{minimizar} \quad \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} + \sum_{i \in I} f_i y_i \\ \text{sujeito a} \quad \sum_{j \in J} x_{ij} \leq a_i y_i \quad , \quad \forall i \in I \quad (R1) \\ \sum_{i \in I} x_{ij} = b_j \quad , \quad \forall j \in J \quad (R2) \\ x_{ij} \geq 0 \quad , \quad \forall i \in I, \forall j \in J \quad (R3) \\ y_i \in \{0, 1\} \quad , \quad \forall i \in I \quad (R4) \end{array} \right.$$

Trata-se de minimizar uma função linear (côncava) sujeito a um conjunto de restrições lineares.

A restrição (R1) limita a quantidade com que cada armazém $i \in I$ pode fornecer aos clientes $j \in J$, caso o armazém seja aberto ou instalado.

A restrição (R2) exige o fornecimento total de cada centro de demanda $j \in J$.

A variável não negativa x_{ij} representa a quantidade de unidades do produto com que o armazém i fornece o centro de demanda j e a variável de decisão y_i tem relação com o fato de abrir ($y_i=1$) ou não ($y_i=0$) o armazém i .

No instante que fixamos as variáveis y_i , CWLP se transforma num problema de transporte capacitado.

Observando a função objetivo no problema (P), podemos notar que existe uma situação de conflito entre os custos fixos e variáveis. Por uma parte, a minimização dos custos variáveis (de transporte) leva a localizar um grande número de pequenos (em capacidade) armazens, e por outra, a minimização dos custos fixos leva a localizar um pequeno número de grandes armazens.

O CWLP é um conhecido problema de otimização combinatória da classe NP-Hard (Cornuejols [19]), que no caso que $a_i \geq \sum_{j \in J} b_j$, $\forall i \in I$ se transforma no Problema de Localização de Plantas Não Capacitado, também NP-Hard. Cornuejols [19] prova que se as capacidades a_i e as demandas b_j assumem valores em $\{1,2\}$ então CWLP pode ser resolvido em tempo polinomial, mas se as capacidades assumem valores no conjunto $\{1, \dots, p\}$ com $p \geq 3$ (ainda se todas as demandas são unitárias), então o problema resulta NP-Hard.

As referências do CWLP datam de aproximadamente 30 anos e incluem modelos, limites inferiores e superiores no valor ótimo do problema, métodos de solução, estudos sobre a estrutura poliédrica, testes de redução e problemas testes.

Os modelos podem variar na sua função objetivo que pode ou não acrescentar a parcela variável com custos operacionais como funções (geralmente côncavas e ligadas ao conceito de economia

de escala) da dimensão ou dos fluxos em cada facilidade (Bitran [10], Kelly [34], Monteroso [45], Feldman [25], Soland [54]). Também os modelos variam nas restrições, as quais podem incluir limites superiores ou inferiores na quantidade de facilidades a serem abertas (Christofides [16], Beasley [9], Barceló [8]). Em Leung [39], se apresenta uma formulação de "empacotamento de vértices" (Vertex Packing) para o caso em que as demandas são iguais e indivisíveis entre as plantas. Também existem formulações para modelos multi-capacitados (Pedrosa [47]) que se referem além do mais, à determinação de um nível ótimo de capacidade escolhido dentre um número finito de possibilidades por cada armazém.

No entanto, nós trabalhamos o modelo (P) anterior, que é o mesmo utilizado por Akinc [2], Bartezzaghi [6], Sá [53], Davis [20], Van Roy [55], Naus [46], Robson [50,51], Dromschke [21] e Jacobsen [31], dentre outros.

A respeito dos limites inferiores do valor ótimo do CWLP, varios tipos de relaxações tem sido desenvolvidas, tanto aquelas que eliminam alguma restrição (Sá [53], Akinc [2], Davis [20], Baker [5]) como aquelas do tipo lagrangeana (Geoffrion [26], Naus [46], Van Roy [55], Christofides [16]). Estes limites tem sido usados na construção de heurísticas ou incorporados em algoritmos exatos tipo separação e avaliação (Branch&Bound, B&B).

Os métodos ou algoritmos de solução de todo problema de otimização combinatorial da classe NP-Hard, podem ser classificados em exatos (que atingem a solução ótima global do problema) ou aproximados (que atingem uma solução viável não necessariamente ótima global). Em qualquer caso eles produzem soluções viáveis, ou seja, limites superiores para o valor ótimo do problema.

Dentre o primeiro grupo e de acordo com a estratégia básica de solução, tem-se classificado os algoritmos como de decomposição ou de enumeração.

Como as formulações do CWLP envolvem variáveis inteiras e reais, o método de decomposição separa a parte real da inteira, seguindo o método de decomposição o particionamento (primal) de Benders, acompanhado de um esquema de enumeração pois o problema mestre (que conecta problemas menores produzidos na decomposição) se reduz num problema de programação inteiro 0-1. Van Roy [55], desenvolve um método chamado de Decomposição Cruzada (Cross Decomposition) que combina os métodos de decomposição de Benders e de relaxação lagrangeana. Quando aplicado ao CWLP, envolve sucessivas soluções de um problema de transporte e de um problema de localização não capacitado, conseguindo explorar simultaneamente as estruturas primal e dual do problema. O algoritmo proposto por Van Roy inclui um procedimento B&B e sua experiencia computacional faz referência a instâncias com 100 facilidades e 200 centros de demanda como máximo.

Os métodos de enumeração procuram dividir o problema em subproblemas menores em tamanho e complexidade. Estes algoritmos são basicamente do tipo separação e avaliação (B&B) que é uma técnica de busca em árvore onde cada nó, obtido pela fixação de alguma variável, representa um subproblema cuja solução ótima será uma solução viável no problema original. Importantes aportes neste sentido e relacionados com o CWLP, tem sido dados por Efroymsen & Ray [23], Sá [53], Soland [54], Akinc & Khumawala [2], Christofides & Beasley [16], Nauss [46], Beasley [9], Bartezzaghi, Colomi & Palermo [6], Davis & Ray [20].

Os métodos aproximados ou heurísticos têm obtido bastante êxito na obtenção de boas soluções viáveis em variados problemas de otimização combinatorial. Particularmente no caso do CWLP, Jacobsen [31] tem generalizado heurísticas do problema de localização não capacitado e do problema da p-mediana. Estas heurísticas baseam-se nos procedimentos gulosos ADD (Kuehn & Hamburger [38]) e DROP (Feldman [25]). O procedimento ADD adiciona em cada iteração aquele armazém que produz a maior diferença entre o decréscimo nos custos variáveis e o custo fixo produzidos pela abertura do armazém, enquanto que o

procedimento DROP elimina em cada iteração aquele armazém que produz a maior diferença entre o custo fixo e o incremento nos custos variáveis produzidos pela eliminação do armazém. Dentre os métodos, alguns não permitem revisar ou recompor decisões anteriores, enquanto que outros baseiam-se na idéia de melhorar uma solução conhecida, ou seja de permitir mudanças em decisões prévias. Estes últimos consideram os seguintes passos: obter uma solução viável inicial (por exemplo mediante algum método construtivo), induzir uma perturbação na solução, reotimizar a solução e repetir os últimos dois passos até que a solução não tenha mais mudanças.

A heurística ALA (Alternate Location Allocation) se caracteriza por fazer perturbação tipo DROP e reotimização tipo ADD, enquanto que a heurística VSM (Vertex Substitution Method) utiliza perturbação tipo ADD e reotimização tipo DROP.

SÁ [53] propõe uma heurística em duas etapas: a primeira faz uma ciclagem dos procedimentos ADD e DROP até que não se tenha mais melhoras e a segunda, analisa todas as combinações dentre as facilidades que estão abertas e as fechadas aceitando-se uma troca caso se produza melhora. Em tal caso voltamos à primeira etapa e assim sucessivamente. Esta heurística é de um alto custo computacional.

Para o caso em que as facilidades tenham diferentes capacidades, Domschke & Drexl [21] mostram que as heurísticas que usam o procedimento ADD podem levar a soluções de baixa qualidade. Isto se dá devido à seleção da facilidade ser feita em ordem decrescente de capacidade até que a viabilidade (soma das capacidades de facilidades abertas maior ou igual a soma das demandas) seja atingida. Neste caso, os autores propõem regras de prioridade para selecionar as facilidades a serem abertas; numa se escolhe aquela que tenha o menor custo fixo por unidade de capacidade (a_h/f_h), outra prefere a facilidade cujo recíproco do custo medio de transporte para fornecer todos os clientes, multiplicado pela capacidade e diminuído no custo fixo seja mínimo, e a última regra é similar a segunda mas considerando na

soma dos custos de transporte só o terço de clientes mais próximos.

No CWLP restrito ao caso em que cada cliente seja fornecido por um único armazém, com todas as variáveis inteiras 0 ou 1, Klincewicz & Luss [37] propõem uma heurística que combina um procedimento ADD para obter uma solução viável inicial, com a relaxação lagrangeana nas capacidades. Para o mesmo modelo acrescentado um limite para as facilidades abertas, Barceló & Casanovas [8] apresentam outra heurística em duas etapas; a primeira consiste na seleção de plantas ou facilidades, baseada na solução de um subproblema lagrangeano, e a segunda que aloca a cada cliente uma única facilidade como resultado de um subproblema de alocação.

Rangel [49], Robson & Bornstein [50], Robson [51] e Campêlo [13] exploram também o procedimento ADD em combinação com testes de redução do problema, inicialmente desenvolvidos por Akinc & Khumawala [2]. Os resultados obtidos com estas heurísticas são muito bons e de fato na maioria dos casos se atingem soluções ótimas globais dos problemas testes. Nestes trabalhos se procuram com êxito boas estimativas para o incremento (decremento) dos custos variáveis, quando uma facilidade é fechada (aberta).

Cornuejols [19] apresenta uma análise comparativa de diferentes relaxações lagrangeanas do CWLP e propõe uma heurística lagrangeana que em cada passo do algoritmo subgradiente na resolução do dual lagrangeano, encontra uma solução do subproblema correspondente. No caso que ela seja viável no problema original, ela é a solução ótima e se for inviável, considera o conjunto das facilidades abertas ($y_i=1$) nessa solução e resolve o problema de transporte associado, achando deste modo sempre uma solução viável do problema original. A melhor solução encontrada durante estas iterações do método de subgradientes é escolhida como a solução da heurística.

Outro método exato de solução aplicado ao CWLP foi

apresentado no trabalho de Bitran et al [10] e é chamado de Otimização Inversa. Na verdade, trata-se de um algoritmo exato só porque na fase final do algoritmo proposto, se utiliza um esquema de enumeração B&B. A parte heurística desse algoritmo explora o seguinte resultado devido a Everett (1963):

Se $P(d)$ é um problema primal com vetor de demandas d , então cada vez que calculamos a função lagrangeana, encontramos uma solução ótima de $P(d)$ para algum valor de d . Devido à escassa experiência computacional apresentada por Bitran et al., não é possível medir o êxito deste algoritmo no CWLP.

Com respeito a estrutura poliédrica do CWLP, não se encontra uma grande quantidade de publicações. Contudo, Leung & Magnanti [39] conseguem identificar uma família de facetas e outras inequações válidas para uma versão do problema em que as capacidades são idênticas em todos os armazens.

São duas as principais contribuições desta tese. A primeira tem a ver com os testes de redução, em cima dos quais se apresenta um resumo dos resultados anteriores acrescentados por uma proposta de melhoria (Capítulo 2). A segunda é a geração e a implementação de uma heurística para o CWLP baseada no algoritmo Simulated Annealing (Capítulo 5) junto com a experiência computacional e a comparação dos resultados com aqueles obtidos por outros autores. No Capítulo 3 se mostram resultados relativos a Cadeias de Markov. No Capítulo 4 se apresenta a estrutura geral do algoritmo Simulated Annealing e se prova sua convergência assintótica a um ótimo global, baseada num modelo de Cadeias de Markov segundo o trabalho de Romeo et al. [52]. No Capítulo 6 se apresentam as conclusões do trabalho e no Apêndice, alguns resultados matemáticos de interesse nos aspectos de convergência do algoritmo.

CAPITULO 2

TESTES DE REDUÇÃO PARA O PROBLEMA DE LOCALIZAÇÃO CAPACITADO

Neste capítulo serão apresentados alguns complementos a respeito dos testes de redução para o Problema de Localização Capacitado (CWLP), inicialmente desenvolvidos por Akinc-Khumawala [2] e também trabalhados por Robson [51] e Robson & Bornstein [50].

Usaremos a seguinte formulação do CWLP:

$$\left\{ \begin{array}{l}
 \text{minimizar} \quad \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} + \sum_{i \in I} f_i y_i \\
 \text{s.a} \quad \sum_{j \in J} x_{ij} \leq a_i y_i \quad ; \forall i \in I = \{1, 2, \dots, m\} \\
 \sum_{i \in I} x_{ij} = b_j \quad ; \forall j \in J = \{1, 2, \dots, n\} \\
 x_{ij} \geq 0 \quad ; \forall i \in I, \forall j \in J \\
 y_i \in \{0, 1\} \quad ; \forall i \in I
 \end{array} \right.$$

Seja $W(S) = \min \{ \sum_{i \in S} \sum_{j \in J} c_{ij} x_{ij} / S \subseteq K, x \in X^S \}$ e $f(S) = \sum_{i \in S} f_i$ onde $S \subseteq K \subseteq I$ é o conjunto das facilidades abertas e X^S definido por $X^S = \{ x \geq 0 / \sum_{j \in J} x_{ij} \leq a_i ; \forall i \in S; \sum_{i \in S} x_{ij} = b_j ; \forall j \in J \}$ é o conjunto das soluções viáveis $x = (x_{ij})$.

O CWLP é o problema de encontrar $S \neq \emptyset$ com $X^S \neq \emptyset$ tal que a função $Z(S) = W(S) + f(S)$ é minimizada.

As funções $Z(S)$, $W(S)$, $f(S)$ satisfazem a seguinte relação conhecida como "supermodularidade":

$$Z(S) - Z(S \cup k) \leq Z(R) - Z(R \cup k); \forall R \subseteq S \subseteq I; \forall k \in I - S \quad (1)$$

ou equivalentemente

$$Z(S' - k) - Z(S') \leq Z(R' - k) - Z(R'); \forall R' \subseteq S' \subseteq I; \forall k \in R' \quad (2)$$

A ideia de um teste de redução exato para uma facilidade $i \in I$, consiste em determinar a priori se a facilidade i aparece

aberta em toda solução ótima ($y_i=1$) ou aparece fechada em toda solução ótima ($y_i=0$). Estes testes baseam-se em criterios de dominância entre custos fixos (dados pelos f_i) e custos variáveis (dados pelos c_{ij}).

Definimos $K_0 = \{ i / y_i = 0 \text{ em toda solução ótima} \}$
 $K_1 = \{ i / y_i = 1 \text{ em toda solução ótima} \}$
 $K_2 = I - K_0 - K_1 = \text{conjunto das variáveis livres}$

Inicialmente teremos $K_0 = K_1 = \phi$, $K_2 = I$

Akinc-khumawala [2] definem:

$$\Delta_i = W[K_1 \cup K_2 - i] - W[K_1 \cup K_2]; \quad \forall i \in K_2 \quad (3)$$

que corresponde ao incremento em custo de transporte ao fechar a facilidade i .

Ao examinar os armazens, um a um, testando se eles podem ser abertos ou fechados, os conjuntos K_0 e K_1 resultam não decrescentes e K_2 resultará não crescente. Logo $K_1 \cup K_2 = I - K_0$ será não crescente e assim Δ_i será não decrescente devido a (2).

Akinc-Khumawala [2] aproveitam este fato para dar o seguinte teste para abrir a facilidade i :

TESTE 1 Se $\Delta_i \geq f_i$ então $y_i = 1$.

O correspondente teste exato para fechar a facilidade i é:

TESTE 2 Se $\Omega_i \leq f_i$ então $y_i = 0$

onde $\Omega_i = W[K_1] - W[K_1 \cup i]$; $\forall i \in K_2$ representa o decrescimento no custo de transporte ao abrir a facilidade i .

O teste de redução completo implica na aplicação alternada dos testes 1 e 2. Começando por teste 1 aplicado a cada $i \in K_2$ (que implica na resolução de $|K_2|+1$ problemas de transporte) obtemos novos conjuntos K_1 e K_2 . Se $|K_1|$ é suficientemente grande como para que $W[K_1]$ seja viável, aplicamos teste 2 para cada $i \in K_2$ (que implica na resolução de $|K_2|+1$ novos problemas de transporte), produzindo outros conjuntos K_0 e K_2 caso o teste tenha tido êxito para pelo menos algum $i \in K_2$. Em tal caso voltamos ao teste 1 e assim sucessivamente até que não possa mais ser aplicado nenhum dos testes.

A análise a seguir corresponde à procura de limites

inferiores para Δ_i e de limites superiores para Ω_i , de modo tal que possamos com eles aplicar testes similares aos anteriores. Especificamente, se denotamos por l.i. ao limite inferior para Δ_i e por l.s. ao limite superior para Ω_i , então os testes se reduzem a:

TESTE 1' : Se l.i. $\geq f_i$ então $\Delta_i \geq f_i$ (que implica em $y_i = 1$)

TESTE 2' : Se l.s. $\leq f_i$ então $\Omega_i \leq f_i$ (que implica em $y_i = 0$)

2.1 LIMITES INFERIORES PARA Δ_i

2.1.1 Limite de Akinc-Khumawala [2]

Se define $\bar{\Delta}_i = W[K_1 U K_2 UR - i] - W[K_1 U K_2 UR]$

O teste resultante de usar $\bar{\Delta}_i$ em vez de Δ_i (teste 1') perde poder de poda, já que pode fixar uma quantidade inferior de variáveis que o teste 1, mas tem a vantagem de sua simplicidade que se traduz na resolução de um problema de mochila simples. Este comentário é válido para todos os testes que resultam de usar aproximações de Δ ou Ω .

Observamos que $\bar{\Delta}_i$ corresponde a relaxar as restrições de capacidade para cada armazém em $K_1 U K_2$ exeto no armazém i , ou equivalentemente para cada $l \in (K_1 U K_2 - i)$, adicionar outro armazém l' com capacidade não limitada (∞) ou com capacidade

$$a_{l'} = \sum_{i \in J} b_j - a_l$$

Neste caso definimos $c_{l'j} = c_{lj}$; $\forall j \in J$.

Então R é definido como o conjunto dos armazens l' .

Segundo (2), temos

$$\bar{\Delta}_i = W[K_1 U K_2 UR - i] - W[K_1 U K_2 UR] \leq W[K_1 U K_2 - i] - W[K_1 U K_2] = \Delta_i$$

resultando $\bar{\Delta}_i$ um limite inferior para Δ_i .

A seguir obteremos um método de cálculo de $\bar{\Delta}_i$.

Consideremos o problema $T[K_1 U K_2 UR - i]$ dado por :

$$T[K_1 UK_2 UR-i] = \begin{cases} \text{minimizar} & \sum_{l \in (K_1 UK_2 UR)} \sum_{j \in J} c_{lj} x_{lj} \\ & l \neq i \\ \text{s.a} & \sum_{l \in (K_1 UK_2 UR)} x_{lj} = b_j \quad ; \quad \forall j \in J \\ & l \neq i \\ & x_{lj} \geq 0 \quad ; \quad \forall l \in K_1 UK_2 UR-i \quad ; \quad \forall j \in J \end{cases}$$

A solução deste problema é dada por

$$\begin{cases} x_{kj} = b_j & \text{se} \quad c_{kj} = \min_{l \in K_1 UK_2-i} c_{lj} \\ \text{senão} & x_{lj} = 0 \end{cases}$$

para cada $j \in J$.

O valor desta solução é $W[K_1 UK_2 UR-i] = \sum_{j \in J} b_j c_{kj}$.

A solução para $T[K_1 UK_2 UR]$ consiste em adicionar o armazém i . Em relação a $T[K_1 UK_2 UR-i]$, temos a possibilidade de reduzir o custo de transporte para cada cliente j .

Seja Δ_{ij} = decréscimo em custo de transporte para j ao incorporar o armazém i

Se $c_{ij} \geq \min_{l \in K_1 UK_2-i} c_{lj}$ então $\Delta_{ij} = 0$
(o cliente j não reduz seu custo de transporte)

Se $c_{ij} < \min_{l \in K_1 UK_2-i} c_{lj}$ então $\Delta_{ij} > 0$
(o cliente j será fornecido pelo armazém i)

Logo $\Delta_{ij} = \max \left\{ \min_{l \in K_1 UK_2-i} c_{lj} - c_{ij}; 0 \right\}$
= decréscimo em custo de transporte unitário ao fornecer o centro de demanda j desde armazém i .

Estamos interessados em maximizar este decréscimo em custo, ou seja,

$$\begin{aligned} \bar{\Delta}_i &= W[K_1 UK_2 UR-i] - W[K_1 UK_2 UR] = \\ &= \max \left\{ \sum_{j \in J} \Delta_{ij} x'_{ij} / \sum_{j \in J} x'_{ij} \leq a_i; 0 \leq x'_{ij} \leq b_j; \quad \forall j \in J \right\} \end{aligned} \quad (4)$$

Assim, $\bar{\Delta}_i$ é o valor de um problema de mochila resolvido com a ordenação não crescente dos Δ_{ij} e atribuindo aos x'_{ij} os valores b_j , até completar a mochila. Podemos ter $0 \leq x'_{ik} < b_k$ para algum k .

2.1.2 Limite inferior de Robson [51].

A idéia é obter Δ_i^D tal que $\bar{\Delta}_i \leq \Delta_i^D \leq \Delta_i$, de modo que se $\Delta_i^D \geq f_i$ então $\Delta_i \geq f_i$ e assim $y_i=1$.

Lembrando que $\Delta_i = W[K_1UK_2-i] - W[K_1UK_2]$, Δ_i^D é obtido mediante um limite inferior (l.i.) para $W[K_1UK_2-i]$ e mantendo invariante o valor $W[K_1UK_2]$. Assim teremos:

$$\Delta_i = W[K_1UK_2-i] - W[K_1UK_2] \geq \text{l.i.} - W[K_1UK_2] = \Delta_i^D$$

Sejam x_{lj}^* , u_l^* , v_j^* ($l \in K_1UK_2$; $j \in J$) as soluções ótimas para $T[K_1UK_2]$ e seu dual $T^D[K_1UK_2]$ dados por:

$$T[K_1UK_2] = \left\{ \begin{array}{l} \text{minimizar} \quad \sum_{l \in K_1UK_2} \sum_{j \in J} c_{lj} x_{lj} \\ \text{s.a} \quad \sum_{j \in J} x_{lj} \leq a_l \quad ; \quad \forall l \in K_1UK_2 \quad (\longleftrightarrow u_l) \\ \sum_{l \in K_1UK_2} x_{lj} = b_j \quad ; \quad \forall j \in J \quad (\longleftrightarrow v_j) \\ x_{lj} \geq 0 \quad ; \quad \forall l \in K_1UK_2 ; \forall j \in J \end{array} \right.$$

$$T^D[K_1UK_2] = \left\{ \begin{array}{l} \text{maximizar} \quad \sum_{l \in K_1UK_2} a_l u_l + \sum_{j \in J} b_j v_j \\ \text{s.a} \quad u_l + v_j \leq c_{lj} \quad ; \quad \forall l \in K_1UK_2 ; \forall j \in J \\ u_l \leq 0 \quad ; \quad \forall l \in K_1UK_2 \end{array} \right.$$

Temos $v_j^* = \min_{l \in K_1UK_2} \{c_{lj} - u_l^*\}$ para cada $j \in J$

O limite inferior para $W[K_1UK_2-i]$ é obtido usando uma solução viável de $T^D[K_1UK_2-i]$, onde

$$T^D[K_1 U K_2 - i] = \begin{cases} \text{maximizar} & \sum_{l \in K_1 U K_2 - i} a_l u_l + \sum_{j \in J} b_j v_j \\ \text{s.a.} & u_l + v_j \leq c_{lj} \quad ; \quad \forall l \in K_1 U K_2 - i; \quad \forall j \in J \\ & u_l \leq 0 \quad ; \quad \forall l \in K_1 U K_2 - i \end{cases}$$

Sejam $u_l' = u_l^*$; $\forall l \in K_1 U K_2 - i$

$$v_j' = \min_{l \in K_1 U K_2 - i} \{ c_{lj} - u_l^* \}; \quad \forall j \in J$$

Podemos provar que $\{ (u_l', v_j') / l \in K_1 U K_2 - i; j \in J \}$ é solução viável de $T^D[K_1 U K_2 - i]$.

$$\text{Efetivamente, } u_p' + v_j' = u_p^* + \min_{l \in K_1 U K_2 - i} \{ c_{lj} - u_l^* \} \leq u_p^* + c_{pj} - u_p^* = c_{pj}$$

e $u_p' = u_p^* \leq 0 \quad ; \quad \forall p \in K_1 U K_2 - i$.

O valor desta solução viável na função objetivo de $T^D[K_1 U K_2 - i]$ é

$$\sum_{l \in K_1 U K_2 - i} a_l u_l' + \sum_{j \in J} b_j v_j' = \sum_{l \in K_1 U K_2 - i} a_l u_l^* + \sum_{j \in J} b_j v_j'$$

Logo $\Delta_i = W[K_1 U K_2 - i] - W[K_1 U K_2]$

$$\begin{aligned} &\geq \sum_{l \in K_1 U K_2 - i} a_l u_l^* + \sum_{j \in J} b_j v_j' - \left(\sum_{l \in K_1 U K_2} a_l u_l^* + \sum_{j \in J} b_j v_j^* \right) \\ &= \sum_{j \in J} b_j (v_j' - v_j^*) - a_i u_i^* \end{aligned}$$

$$\text{Definimos } \Delta_i^D = \sum_{j \in J} b_j (v_j' - v_j^*) - a_i u_i^* \quad (5)$$

A seguir, usamos condições das folgas complementares para obter outra forma de Δ_i^D .

$$(i) \quad x_{ij}^* = 0 \Rightarrow v_j^* = v_j'$$

Efetivamente, supondo que $b_j > 0$, deve existir $x_{pj}^* > 0$ com $p \neq i$

$$\Rightarrow u_p^* + v_j^* = c_{pj} \Rightarrow v_j^* = c_{pj} - u_p^*$$

$$\Rightarrow \min_{l \in K_1 U K_2} \{ c_{lj} - u_l^* \} = c_{pj} - u_p^*$$

$$\Rightarrow \min_{l \in K_1 U K_2} \{ c_{lj} - u_l^* \} = \min_{l \in K_1 U K_2 - i} \{ c_{lj} - u_l^* \} \Rightarrow v_j^* = v_j'$$

$$(ii) \quad x_{ij}^* > 0 \Rightarrow u_i^* + v_j^* = c_{ij} \Rightarrow v_j^* = c_{ij} - u_i^*$$

$$(iii) u_i^* \neq 0 \Rightarrow \sum_{j \in J} x_{ij}^* = a_i = \sum_{\substack{j \in J \\ x_{ij}^* \neq 0}} x_{ij}^*$$

$$\begin{aligned} \text{Logo } \Delta_i &\geq \Delta_i^D = \sum_{j \in J} b_j (v'_j - v_j^*) - a_i u_i^* \stackrel{(*)}{=} \sum_{j \in J} x_{ij}^* (v'_j - v_j^*) - a_i u_i^* = \\ &= \sum_{\substack{j \in J \\ x_{ij}^* \neq 0}} x_{ij}^* (v'_j - v_j^*) - \left[\sum_{\substack{j \in J \\ x_{ij}^* \neq 0}} x_{ij}^* \right] \cdot u_i^* \\ &= \sum_{\substack{j \in J \\ x_{ij}^* \neq 0}} x_{ij}^* (v'_j - c_{ij} + u_i^* - u_i^*) = \sum_{\substack{j \in J \\ x_{ij}^* \neq 0}} x_{ij}^* (v'_j - c_{ij}) \end{aligned}$$

Verifiquemos agora a igualdade (*) acima.

Da primeira consequência das folgas complementares temos:

$$x_{ij}^* = 0 \Rightarrow v_j^* = v'_j \Rightarrow b_j (v'_j - v_j^*) = x_{ij}^* (v'_j - v_j^*)$$

Se $x_{ij}^* \neq 0$ então podemos ter :

$$(a) \quad x_{ij}^* = b_j \Rightarrow b_j (v'_j - v_j^*) = x_{ij}^* (v'_j - v_j^*) \quad \text{ou}$$

$$(b) \quad 0 < x_{ij}^* < b_j. \quad \text{Neste caso } \exists p \neq i \text{ tal que } x_{pj}^* > 0 \text{ e } x_{pj}^* + x_{ij}^* \leq b_j$$

De $x_{pj}^* > 0$ resulta $v'_j = v_j^*$ como foi provado na primeira consequência das folgas complementares.

$$\text{Assim, } x_{ij}^* < b_j \Rightarrow v'_j = v_j^* \quad \text{e} \quad b_j (v'_j - v_j^*) = x_{ij}^* (v'_j - v_j^*)$$

Em consequência,

$$\begin{aligned} \sum_{j \in J} x_{ij}^* (v'_j - v_j^*) &= \sum_{\substack{j \in J \\ x_{ij}^* = 0}} x_{ij}^* (v'_j - v_j^*) + \sum_{\substack{j \in J \\ 0 < x_{ij}^* < b_j}} x_{ij}^* (v'_j - v_j^*) + \sum_{\substack{j \in J \\ x_{ij}^* = b_j}} x_{ij}^* (v'_j - v_j^*) \\ &= \sum_{\substack{j \in J \\ x_{ij}^* = 0}} b_j (v'_j - v_j^*) + \sum_{\substack{j \in J \\ 0 < x_{ij}^* < b_j}} b_j (v'_j - v_j^*) + \sum_{\substack{j \in J \\ x_{ij}^* = b_j}} b_j (v'_j - v_j^*) \\ &= \sum_{j \in J} b_j (v'_j - v_j^*) \end{aligned}$$

$$\text{Assim, } \Delta_i^D = \sum_{\substack{j \in J \\ x_{ij}^* \neq 0}} x_{ij}^* (v'_j - c_{ij}) \leq \Delta_i$$

Provaremos agora que $\bar{\Delta}_i \leq \Delta_i^D$

Uma prova detalhada se encontra em [51].

A idéia ali é obter um limite superior $\hat{\Delta}_i$ para $\bar{\Delta}_i$, ou seja, $\bar{\Delta}_i \leq \hat{\Delta}_i$.

Depois encontrar um limite superior $\hat{\Delta}_i^U$ para $\hat{\Delta}_i$, ou seja, $\hat{\Delta}_i \leq \hat{\Delta}_i^U$ e finalmente mostrar que $\hat{\Delta}_i^U = \Delta_i^D$.

Aqui é apresentada uma outra prova mais simples.

Seja

$$(P) : \begin{cases} \text{maximizar } \sum_{j \in J} \Delta_{ij} x'_{ij} \\ \text{s.a. } \sum_{j \in J} x'_{ij} \leq a_i \quad (\leftrightarrow u_i) \text{ \u00fanica restri\u00e7\u00e3o} \\ x'_{ij} \leq b_j \quad ; \quad \forall j \in J \quad (\leftrightarrow v_j) \\ x'_{ij} \geq 0 \quad ; \quad \forall j \in J \end{cases}$$

onde $\Delta_{ij} = \max \{ \min_{l \in K_1, K_2 - i} c_{lj} - c_{ij} ; 0 \}$ com valor \u00f3timo $\bar{\Delta}_i$.

O problema dual de (P) \u00e9:

$$(D) : \begin{cases} \text{minimizar } \sum_{j \in J} b_j v_j + a_i u_i \\ \text{s.a. } u_i + v_j \geq \Delta_{ij} \quad ; \quad \forall j \in J \\ u_i \geq 0 \\ v_j \geq 0 \quad ; \quad \forall j \in J \end{cases}$$

Consideremos a seguinte solu\u00e7\u00e3o vi\u00e1vel de (D): $\begin{cases} \hat{u}_i = -u_i^* \\ \hat{v}_j = v_j' - v_j^*, \forall j \in J \end{cases}$

Verifiquemos a viabilidade:

(a) $\hat{u}_i = -u_i^* \geq 0$, pois $u_i^* \leq 0$

(b) $\hat{v}_j = v_j' - v_j^* \geq 0$, pela defini\u00e7\u00e3o de v_j' e v_j^*

(c) $\hat{u}_i + \hat{v}_j = v_j' - v_j^* - u_i^* = v_j' - (u_i^* + v_j^*) \geq v_j' - c_{ij} =$

$$= \min_{l \in K_1, K_2 - i} \{ c_{lj} - u_l^* \} - c_{ij}$$

$$= c_{pj} - u_p^* - c_{ij} \quad \text{para } p \neq i$$

$$\geq \min_{l \in K_1, K_2 - i} c_{lj} - u_p^* - c_{ij}$$

$$\geq \min_{l \in K_1, K_2 - i} c_{lj} - c_{ij} \quad \text{pois } u_p^* \leq 0$$

Logo $\hat{u}_i + \hat{v}_j \geq \min_{l \in K_1, K_2 - i} c_{lj} - c_{ij}$ e tamb\u00e9m $\hat{u}_i + \hat{v}_j \geq 0$

$$\text{Assim, } \hat{u}_i + \hat{v}_j \geq \max \{ 0 ; \min_{l \in K_1 U_{K_2} - i} c_{lj} - c_{ij} \} = \Delta_{ij}$$

O valor desta solução viável de (D) é:

$$\sum_{j \in J} b_j \hat{v}_j + a_i \hat{u}_i = \sum_{j \in J} b_j (v'_j - v_j^*) - a_i u_i^* = \sum_{j \in J} x_{ij}^* (v'_j - c_{ij}) = \Delta_i^D$$

$x_{ij}^* \geq 0$

$$\text{Logo } \Delta_i^D \geq \bar{\Delta}_i$$

2.1.3 Limites obtidos mediante relaxações lagrangeanas

Seguindo com a ideia de encontrar limites inferiores para Δ_i através de limites inferiores para $W[K_1 U_{K_2} - i]$, estudamos algumas relaxações lagrangeanas do problema $T[K_1 U_{K_2} - i]$.

$$T[K_1 U_{K_2} - i] = \left\{ \begin{array}{l} \text{minimizar } \sum_{l \in K_1 U_{K_2} - i} \sum_{j \in J} c_{lj} x_{lj} \\ \text{s.a. } \sum_{j \in J} x_{lj} \leq a_l ; \forall l \in K_1 U_{K_2} - i \quad (\leftrightarrow u_l) \quad (a) \\ \sum_{l \in K_1 U_{K_2} - i} x_{lj} = b_j ; \forall j \in J \quad (\leftrightarrow v_j) \quad (b) \\ x_{lj} \geq 0 ; \forall l \in K_1 U_{K_2} - i ; \forall j \in J \end{array} \right.$$

Tentaremos as seguintes relaxações:

(Rel 1) relaxar restrições (a) e (b) usando multiplicadores u_l^*, v_j^*

(Rel 2) relaxar restrições (b) usando multiplicadores v_j^*

(Rel 3) relaxar restrições (a) usando multiplicadores u_l^*

(Rel 4) relaxar restrições (b) usando multiplicadores v_j^*

Em cada caso, u^*, v^* representará a solução ótima dual de $T[K_1 U_{K_2} - i]$.

(Rel 1):

Seja $LT[K_1 U_{K_2} - i]$ o seguinte problema:

Podemos decompor este problema em $|K_1 \cup K_2 - i|$ problemas de mochila, um para cada $l \in K_1 \cup K_2 - i$.

$$\text{Seja } LT^l[K_1 \cup K_2 - i] = \begin{cases} \text{minimizar } \sum_{j \in J} (c_{lj} - v_j^*) x_{lj} \\ \text{s.a. } \sum_{j \in J} x_{lj} \leq a_l \\ x_{lj} \geq 0 \quad ; \forall j \in J \end{cases}$$

Se $c_{lj} - v_j^* \geq 0$ então $x_{lj} = 0$. Seja $J_l^* = \{j \in J / c_{lj} - v_j^* < 0\}$.

Para cada $l \in K_1 \cup K_2 - i$, seja $c_l^* = c_{l, t(l)} - v_{t(l)}^* = \min\{c_{lj} - v_j^* / j \in J\}$

Se $c_l^* \geq 0$ então $x_{lj} = 0$; $\forall j \in J$.

Se $c_l^* < 0$ então $x_{l, t(l)} = a_l$ e $x_{lj} = 0 \quad \forall j \neq t(l)$.

Assim, a solução ótima de $LT^l[K_1 \cup K_2 - i]$ é dada por:

$$\hat{x}_{lj} = \begin{cases} 0 & \text{se } j \in J - J_l^* \\ a_l & \text{se } j = t(l) \\ 0 & \text{se } j \neq t(l), j \in J_l^* \end{cases}$$

$$\text{com valor } \begin{cases} c_l^* \cdot a_l & \text{se } c_l^* \leq 0 \\ 0 & \text{se } c_l^* \geq 0 \end{cases} \quad \text{ou } \min\{c_l^*; 0\} \cdot a_l$$

Com isto, o valor ótimo de $LT[K_1 \cup K_2 - i]$ resulta:

$$\sum_{l \in K_1 \cup K_2 - i} \min\{c_l^*; 0\} \cdot a_l + \sum_{j \in J} b_j v_j^*$$

Logo, $\Delta_i = W[K_1 \cup K_2 - i] - W[K_1 \cup K_2]$

$$\geq \sum_{l \in K_1 \cup K_2 - i} \min\{c_l^*; 0\} a_l + \sum_{j \in J} b_j v_j^* - \left[\sum_{l \in K_1 \cup K_2} u_l^* a_l + \sum_{j \in J} b_j v_j^* \right]$$

$$= \sum_{l \in K_1 \cup K_2 - i} \left[\min\{c_l^*; 0\} - u_l^* \right] a_l - a_i u_i^* = \Delta_i^{(2)}$$

O seguinte exemplo mostra que não é necessariamente $\Delta_i^{(2)} \geq \Delta_i^D$

EXEMPLO 2.1Tabela de custos variáveis ou de transporte

		centros de demanda			capacidades
		1	2	3	a_i
centros de ofertas	1	1	3	5	100
	2	3	6	7	100
	3	7	5	1	60
demandas		b_j	70	70	60

Aquí vamos considerar $K_1 \cup K_2 = \{1, 2, 3\}$ e $i=3$

A solução ótima de $T[K_1 \cup K_2]$ é $x_{11}^* = 30$; $x_{12}^* = 70$; $x_{21}^* = 40$; $x_{33}^* = 60$ (quando o valor de x_{lj}^* não seja especificado, entenderemos que é 0)

A solução ótima dual é $u_1^* = -2$; $u_2^* = 0$; $u_3^* = 0$; $v_1^* = 3$; $v_2^* = 5$; $v_3^* = 1$

O valor ótimo é 420.

$$v_1' = \min_{l=1,2} \{c_{l,1} - u_l^*\} = \min \{c_{11} - u_1^*; c_{21} - u_2^*\} = \min \{3, 3\} = 3 = v_1^*$$

$$v_2' = \min_{l=1,2} \{c_{l,2} - u_l^*\} = \min \{c_{12} - u_1^*; c_{22} - u_2^*\} = \min \{5, 6\} = 5 = v_2^*$$

$$v_3' = \min_{l=1,2} \{c_{l,3} - u_l^*\} = \min \{c_{13} - u_1^*; c_{23} - u_2^*\} = \min \{7, 7\} = 7$$

$$\text{Assim, } \Delta_3^D = \sum_{j \in J} (v_j' - v_j^*) b_j - a_3 u_3^* = 0 \cdot 70 + 0 \cdot 70 + 6 \cdot 60 - 60 \cdot 0 = 360$$

$$\text{Agora, } c_1' = \min_{j=1,2,3} \{c_{1,j} - v_j^*\} = \min \{-2, -2, 4\} = -2 \quad (l=1 \Rightarrow t(l)=1,2)$$

$$c_2' = \min_{j=1,2,3} \{c_{2,j} - v_j^*\} = \min \{0, 1, 6\} = 0 \quad (l=2 \Rightarrow t(l)=1)$$

$$\text{Logo } \Delta_3^{(2)} = \sum_{l=1,2} (\min \{c_l'; 0\} - u_l^*) \cdot a_l - a_3 u_3^* = 0 \cdot 100 + 0 \cdot 100 - 60 \cdot 0 = 0$$

Em consequência, não há vantagem em se considerar $\Delta_i^{(2)}$.

(Re13)

Seja agora $LT[K_1 \cup K_2 - i]$ o seguinte problema:

$$\left\{ \begin{array}{l} \text{minimizar } \sum_{l \in K_1 U_{K_2-i}} \sum_{j \in J} c_{lj} x_{lj} + \sum_{l \in K_1 U_{K_2-i}} u_l^* \left(a_l - \sum_{j \in J} x_{lj} \right) \\ \text{s.a. } \sum_{l \in K_1 U_{K_2-i}} x_{lj} = b_j \quad ; \forall j \in J \quad (R1) \\ x_{lj} \geq 0 \quad ; \forall l \in K_1 U_{K_2-i} ; \forall j \in J \quad (R2) \end{array} \right.$$

ou equivalentemente

$$\left\{ \begin{array}{l} \text{minimizar } \sum_{j \in J} \sum_{l \in K_1 U_{K_2-i}} (c_{lj} - u_l^*) x_{lj} + \sum_{l \in K_1 U_{K_2-i}} a_l u_l^* \\ \text{s.a. } (R1) \text{ e } (R2) \end{array} \right.$$

Podemos decompor este problema em $|J|$ problemas de mochila, um para cada $j \in J$.

Seja

$$LT^j[K_1 U_{K_2-i}] = \left\{ \begin{array}{l} \text{minimizar } \sum_{l \in K_1 U_{K_2-i}} (c_{lj} - u_l^*) x_{lj} \\ \text{s.a. } \sum_{l \in K_1 U_{K_2-i}} x_{lj} = b_j \\ x_{lj} \geq 0 \quad ; \forall l \in K_1 U_{K_2-i} \end{array} \right.$$

Definamos para cada $j \in J$, $v'_j = \min_{l \in K_1 U_{K_2-i}} \{c_{lj} - u_l^*\} = c_{k(j),j} - u_{k(j)}^*$

Então a solução ótima de $LT^j[K_1 U_{K_2-i}]$ é dada por

$$\hat{x}_{lj} = \begin{cases} b_j & \text{se } l = k(j) \\ 0 & \text{se } l \neq k(j) \end{cases}$$

com valor $(c_{k(j),j} - u_{k(j)}^*) \cdot b_j = v'_j b_j$.

Assim, a solução ótima para $LT[K_1 U_{K_2-i}]$ tem valor

$$\sum_{j \in J} v'_j b_j + \sum_{l \in K_1 U_{K_2-i}} a_l u_l^*$$

Com isso, $\Delta_i = W[K_1 U_{K_2-i}] - W[K_1 U_{K_2}]$

$$\begin{aligned} &\geq \sum_{j \in J} v'_j b_j + \sum_{l \in K_1 U_{K_2-i}} a_l u_l^* - \sum_{j \in J} v_j^* b_j - \sum_{l \in K_1 U_{K_2}} a_l u_l^* \\ &= \sum_{j \in J} (v'_j - v_j^*) b_j - \sum_{i} a_i u_i^* = \sum_{\substack{j \in J \\ x_{ij} \neq 0}} x_{ij}^* (v'_j - c_{ij}) = \Delta_i^D \end{aligned}$$

Logo, usando este tipo de relaxação obtemos o mesmo limite Δ_i^D para Δ_i , apresentado em 2.1.2.

(Rel4)

Seja agora $LT[K_1UK_2-i]$ o seguinte problema:

$$\left\{ \begin{array}{l} \text{minimizar} \quad \sum_{l \in K_1UK_2-i} \sum_{j \in J} c_{lj} x_{lj} + \sum_{j \in J} v'_j \left(b_j - \sum_{l \in K_1UK_2-i} x_{lj} \right) \\ \text{s.a} \quad \sum_{j \in J} x_{lj} \leq a_l \quad ; \quad \forall l \in K_1UK_2-i \quad (R1) \\ x_{lj} \geq 0 \quad ; \quad \forall l \in K_1UK_2-i ; \quad \forall j \in J \quad (R2) \end{array} \right.$$

ou equivalentemente

$$\left\{ \begin{array}{l} \text{minimizar} \quad \sum_{l \in K_1UK_2-i} \sum_{j \in J} (c_{lj} - v'_j) x_{lj} + \sum_{j \in J} b_j v'_j \\ \text{s.a} \quad (R1) \text{ e } (R2) \end{array} \right.$$

onde $v'_j = \min_{l \in K_1UK_2-i} \{c_{lj} - u_l^*\}$ para cada $j \in J$.Podemos decompor este problema em $|K_1UK_2-i|$ problemas de mochila.

$$\text{Seja } LT^l[K_1UK_2-i] = \left\{ \begin{array}{l} \text{minimizar} \quad \sum_{j \in J} (c_{lj} - v'_j) x_{lj} \\ \text{s.a} \quad \sum_{j \in J} x_{lj} \leq a_l \\ x_{lj} \geq 0 \quad ; \quad \forall j \in J \end{array} \right.$$

Definamos para cada $l \in K_1UK_2-i$, $c_l^* = c_{l,j(\alpha)} - v'_{j(\alpha)} = \min_{j \in J} \{c_{lj} - v'_j\}$

Temos duas situações:

- (a) Se $c_l^* \geq 0$ então a solução ótima para $LT^l[K_1UK_2-i]$ é $x_{lj} = 0, \forall j \in J$. Neste caso, o valor na função objetivo é zero.
- (b) Se $c_l^* < 0$ então a solução ótima de $LT^l[K_1UK_2-i]$ é dada por

$$x_{lj} = \begin{cases} a_l & \text{se } j=j(\alpha) \\ 0 & \text{se } j \neq j(\alpha) \end{cases}$$

Neste caso, o valor na função objetivo de $LT^l[K_1UK_2-i]$ é $c_l^* \cdot a_l$ Em consequência, o valor ótimo de $LT^l[K_1UK_2-i]$ é $\min\{c_l^*; 0\} \cdot a_l$

Assim, a solução ótima para $LT[K_1 \cup K_2 - i]$ tem custo

$$\sum_{l \in K_1 \cup K_2 - i} \min\{c_l^*; 0\} \cdot a_l + \sum_{j \in J} b_j v'_j$$

Logo, $\Delta_i = W[K_1 \cup K_2 - i] - W[K_1 \cup K_2]$

$$\begin{aligned} &\geq \sum_{l \in K_1 \cup K_2 - i} \min\{c_l^*; 0\} \cdot a_l + \sum_{j \in J} b_j v'_j - \sum_{l \in K_1 \cup K_2} a_l u_l^* - \sum_{j \in J} b_j v_j^* \\ &= \sum_{l \in K_1 \cup K_2 - i} \left[\min\{c_l^*; 0\} - u_l^* \right] \cdot a_l - a_l u_l^* + \sum_{j \in J} b_j (v'_j - v_j^*) \\ &= \sum_{l \in K_1 \cup K_2 - i} \left[\min\{c_l^*; 0\} - u_l^* \right] \cdot a_l + \Delta_i^D \end{aligned}$$

Definimos $\Delta_i^{(4)} = \Delta_i^D + \sum_{l \in K_1 \cup K_2 - i} \left[\min\{c_l^*; 0\} - u_l^* \right] \cdot a_l$

Logo $\Delta_i^{(4)} \leq \Delta_i$ é outro limite inferior para Δ_i .

Para provar que $\Delta_i^{(4)} \geq \Delta_i^D$, devemos provar que $\min\{c_l^*; 0\} - u_l^* \geq 0$ para cada $l \in K_1 \cup K_2 - i$. Dividimos em dois casos:

(i) Se $c_l^* < 0$, basta provar que $c_l^* - u_l^* \geq 0$.

Efetivamente, $c_l^* = \min_{j \in J} \{c_{lj} - v'_j\} = c_{l, j_0} - v'_{j_0}$; $\forall l \in K_1 \cup K_2 - i$

$$\text{e } v'_{j_0} = \min_{p \in K_1 \cup K_2 - i} \{c_{p, j_0} - u_p^*\} \leq c_{l, j_0} - u_l^*$$

$$\text{Logo } 0 \leq c_{l, j_0} - u_l^* - v'_{j_0} \Rightarrow 0 \leq c_l^* - u_l^*$$

(ii) Se $c_l^* \geq 0$, devemos provar que $-u_l^* \geq 0$ o que acontece pois $u_l^* \leq 0$, para cada $l \in K_1 \cup K_2 - i$.

Assim, temos provado que $\Delta_i^D \leq \Delta_i^{(4)} \leq \Delta_i$

Provemos agora que $\Delta_i^D = \Delta_i^{(4)}$, não resultando $\Delta_i^{(4)}$ uma melhoria com respeito a Δ_i^D (pelo menos na teoria).

Observamos que $\Delta_i^{(4)} > \Delta_i^D$ sse

$$\exists l_0 \in K_1 \cup K_2 - i \text{ tal que } \min\{c_{l_0}^*; 0\} - u_{l_0}^* > 0$$

Segundo foi provado em (i) acima, $0 \leq c_{l_0}^* - u_{l_0}^*$ ou $c_{l_0}^* \geq u_{l_0}^*$, relação válida independentemente se $c_{l_0}^* < 0$ ou se $c_{l_0}^* \geq 0$.

Se $u_{l_0}^* = 0$ então $c_{l_0}^* \geq 0$ e logo $\min\{c_{l_0}^*; 0\} - u_{l_0}^* = 0$ (A)

Então só pode ser $u_{l_0}^* < 0$. Suponhamos que $u_{l_0}^* + v'_{j_0} < c_{l_0, j_0}$; $\forall j \in J$

e consideremos

$$\hat{u}_l = \begin{cases} u_l^* & \text{para } l \neq l_0 \\ u_l^* + \varepsilon & \text{para } l = l_0 \\ u_i^* & \text{para } l = i \end{cases} ; \forall l \in K_1 \cup K_2$$

$$\hat{v}_j = v_j^* ; \forall j \in J,$$

sendo $\varepsilon > 0$ tal que $u_{l_0}^* + \varepsilon \leq 0$ e $(u_{l_0}^* + \varepsilon) + v_j^* \leq c_{l_0, j} ; \forall j \in J$

Por exemplo, $\varepsilon = \min \{-u_{l_0}^* ; \min_{j \in J} \{c_{l_0, j} - v_j^* - u_{l_0}^*\}\}$

$$= \min \{-u_{l_0}^* ; c_{l_0, j_0} - v_{j_0}^* - u_{l_0}^*\} > 0$$

Para esse valor de ε temos, $u_{l_0}^* + \varepsilon \leq u_{l_0}^* + (-u_{l_0}^*) = 0$ (*)

$$\text{e } (u_{l_0}^* + \varepsilon) + v_j^* \leq u_{l_0}^* + v_j^* + c_{l_0, j} - v_j^* - u_{l_0}^* = c_{l_0, j} ; \forall j \in J \quad (**)$$

Afirmamos então que $\{(\hat{u}_l, \hat{v}_j) / l \in K_1 \cup K_2 ; j \in J\}$ representa uma solução viável de $T^D[K_1 \cup K_2]$. Efetivamente:

- (-) $\hat{u}_l \leq 0$ segundo a definição de \hat{u}_l e o resultado (*)
- (-) $\hat{u}_l + \hat{v}_j = u_l^* + v_j^* \leq c_{l, j} ; \forall l \in K_1 \cup K_2 - \{l_0\} ; \forall j \in J$
- $\hat{u}_{l_0} + \hat{v}_j = (u_{l_0}^* + \varepsilon) + v_j^* \leq c_{l_0, j} ; \forall j \in J$ segundo (**).

O valor desta solução na função objetivo de $T^D[K_1 \cup K_2]$ é

$$\begin{aligned} \sum_{l \in K_1 \cup K_2} a_l \hat{u}_l + \sum_{j \in J} b_j \hat{v}_j &= \sum_{l \in K_1 \cup K_2 - l_0} a_l u_l^* + a_{l_0} \cdot (u_{l_0}^* + \varepsilon) + \sum_{j \in J} b_j v_j^* \\ &= \sum_{l \in K_1 \cup K_2} a_l u_l^* + \sum_{j \in J} b_j v_j^* + a_{l_0} \cdot \varepsilon \\ &> W[K_1 \cup K_2] \text{ o que não é possível.} \end{aligned}$$

A contradição vem de supor que $u_{l_0}^* + v_j^* < c_{l_0, j} ; \forall j \in J$

Assim, deve existir $j_0 \in J$ tal que $u_{l_0}^* + v_{j_0}^* = c_{l_0, j_0}$

Isto implica em $v'_{j_0} \geq v_{j_0}^* = c_{l_0, j_0} - u_{l_0}^*$

$$\Rightarrow v'_{j_0} = \min_{l \in K_1 \cup K_2 - i} \{c_{l, j_0} - u_l^*\} \leq c_{l_0, j_0} - u_{l_0}^* \leq v'_{j_0}$$

$$\Rightarrow v'_{j_0} = c_{l_0, j_0} - u_{l_0}^* \Rightarrow u_{l_0}^* = c_{l_0, j_0} - v'_{j_0}$$

$$\Rightarrow c_{l_0}^* = \min_{j \in J} \{c_{l_0, j} - v_j^*\} \leq c_{l_0, j_0} - v'_{j_0} = u_{l_0}^*$$

$$\rightarrow c_{l_0}^* \leq u_{l_0}^*$$

e como também temos $c_{l_0}^* \geq u_{l_0}^*$ então $c_{l_0}^* = u_{l_0}^* < 0$ o que implica

$$\text{em } \min\{c_{l_0}^*; 0\} - u_{l_0}^* = 0 \quad (B)$$

Assim, considerando (A) e (B) temos que $\min\{c_l^*; 0\} - u_l^* = 0$ para cada $l \in K_1 \cup K_2 - 1$.

Com isso fica provado que

$$\Delta_i^{(4)} = \Delta_i^D + \sum_{l \in K_1 \cup K_2 - 1} \left(\min\{c_l^*; 0\} - u_l^* \right) \cdot a_l = \Delta_i^D$$

(Rel5)

Consideremos agora uma outra alternativa similar a (Rel3), mas acrescentando uma restrição redundante em $T[K_1 \cup K_2 - 1]$; escrevemos

$$T[K_1 \cup K_2 - 1] = \left\{ \begin{array}{l} \text{minimizar} \quad \sum_{l \in K_1 \cup K_2 - 1} \sum_{j \in J} c_{lj} x_{lj} \\ \\ \text{s.a} \quad \sum_{j \in J} x_{lj} \leq a_l \quad ; \forall l \in K_1 \cup K_2 - 1 \quad (R1) \\ \\ \sum_{l \in K_1 \cup K_2 - 1} x_{lj} = b_j \quad ; \forall j \in J \quad (R2) \\ \\ x_{lj} \geq 0 \quad ; \forall l \in K_1 \cup K_2 - 1; \forall j \in J \quad (R3) \\ \\ x_{lj} \leq \min\{a_l, b_j\} \quad ; \forall l \in K_1 \cup K_2 - 1; \forall j \in J \quad (R4) \end{array} \right.$$

Seja $LT[K_1 \cup K_2 - 1]$ a seguinte relaxação lagrangeana de $T[K_1 \cup K_2 - 1]$:

$$\left\{ \begin{array}{l} \text{minimizar} \quad \sum_{l \in K_1 \cup K_2 - 1} \sum_{j \in J} c_{lj} x_{lj} + \sum_{l \in K_1 \cup K_2 - 1} u_l^* \left[a_l - \sum_{j \in J} x_{lj} \right] \\ \\ \text{s.a} \quad (R2), (R3) \text{ e } (R4) \end{array} \right.$$

ou equivalentemente

$$\left\{ \begin{array}{l} \text{minimizar} \quad \sum_{j \in J} \sum_{l \in K_1 \cup K_2 - 1} (c_{lj} - u_l^*) x_{lj} + \sum_{l \in K_1 \cup K_2 - 1} u_l^* a_l \\ \\ \text{s.a} \quad (R2), (R3) \text{ e } (R4) \end{array} \right.$$

Podemos decompor este problema em $|J|$ problemas de mochila :

$$LT_1^j[K_1, K_2 - i] = \begin{cases} \text{minimizar} & \sum_{l \in K_1 \cup K_2 - i} (c_{lj} - u_l^*) x_{lj} \\ \text{s.a} & \sum_{l \in K_1 \cup K_2 - i} x_{lj} = b_j \\ & 0 \leq x_{lj} \leq \min\{a_l, b_j\}; \forall l \in K_1 \cup K_2 - i; \end{cases}$$

Comparando este $LT_1^j[K_1, K_2 - i]$ com $LT^j[K_1, K_2 - i]$ obtido em (Rel3), temos $v(LT_1^j) \geq v(LT^j)$, onde $v(\cdot)$ representa o valor ótimo do problema (\cdot) . Isto é devido à incorporação da restrição (R4) em LT_1^j . A seguir calculamos $v(LT_1^j)$.

Para isso, ordenamos os valores $v_{lj} = c_{lj} - u_l^*$ em ordem crescente, ou seja,

$$v_j^* = v_{l_1, j} < v_{l_2, j} < v_{l_3, j} < \dots \quad (j \text{ é fixo}) \quad (C)$$

(a) Se $a_{l_1} \geq b_j$ então a solução ótima de LT_1^j é $\hat{x}_{lj} = \begin{cases} b_j & \text{se } l=l_1 \\ 0 & \text{se } l \neq l_1 \end{cases}$

com valor $v(LT_1^j) = v_{l_1, j} \cdot b_j = v_j^* \cdot b_j$

(b) Se $a_{l_1} < b_j$ então podemos supor $a_{l_1} + a_{l_2} + \dots + a_{l_p} + \alpha a_{l_{p+1}} = b_j$ onde $0 \leq \alpha < 1$.

Neste caso, $v(LT_1^j) = v_{l_1, j} \cdot a_{l_1} + v_{l_2, j} \cdot a_{l_2} + \dots + \alpha \cdot v_{l_{p+1}, j} \cdot a_{l_{p+1}}$

Em ambos os casos, a solução ótima de $LT_1^j[K_1, K_2 - i]$ assume o seguinte valor:

$$v(LT_1^j) = \sum_{j \in J} v(LT_1^j) + \sum_{l \in K_1 \cup K_2 - i} u_l^* a_l$$

Definamos $\Delta_i^{(5)} = \sum_{j \in J} \left(v(LT_1^j) - b_j v_j^* \right) - a_i u_i^*$

Assim temos $\Delta_i = W[K_1, K_2 - i] - W[K_1, K_2]$

$$\begin{aligned} & \geq v(LT_1^j) - \left(\sum_{j \in J} b_j v_j^* + \sum_{l \in K_1 \cup K_2} a_l u_l^* \right) \\ & = \sum_{j \in J} \left(v(LT_1^j) - b_j v_j^* \right) - a_i u_i^* = \Delta_i^{(5)} \end{aligned}$$

$$\begin{aligned} &\geq \sum_{j \in J} \left[v(\text{LT}_1^j) - b_j \cdot v_j^* \right] - a_i \cdot u_i^* \\ &= \sum_{j \in J} (v_j' - v_j^*) b_j - a_i \cdot u_i^* = \Delta_i^D \end{aligned}$$

Observamos que no cálculo de $\Delta_i^{(5)}$ é necessário resolver $|J|$ problemas de mochila simples. Se isto for considerado muito caro (na prática não acontece assim), podemos simplificar o cálculo considerando a seguinte variante que só leva em conta os primeiros dois v_{l_j} em (C), ou seja,

$$\begin{aligned} v_j' &= \min_{l \in K_1 \cup K_2 - i} \{c_{lj} - u_l^*\} = c_{l_1, j} - u_{l_1}^* = v_{l_1, j} \\ v_j'' &= \min_{\substack{l \in K_1 \cup K_2 - i \\ l \geq l_1}} \{c_{lj} - u_l^*\} = v_{l_2, j} \end{aligned} \quad (D)$$

Cabe notar que l_1 e l_2 dependem de j .

Os casos (a) e (b) anteriores ficam assim:

(a) Se $a_{l_1} \geq b_j$, definindo $w_j := b_j \cdot v_j'$ teremos $v(\text{LT}_1^j) = w_j$.

(b) Se $a_{l_1} < b_j$, definindo $w_j := v_j' \cdot a_{l_1} + v_j''(b_j - a_{l_1})$ teremos

$$\begin{aligned} v(\text{LT}_1^j) &= v_j' \cdot a_{l_1} + v_j'' \cdot a_{l_2} + v_{l_3, j} \cdot a_{l_3} + \dots + \alpha \cdot v_{l_{p+1}, j} \cdot a_{l_{p+1}} \\ &\geq v_j' \cdot a_{l_1} + v_j''(a_{l_2} + a_{l_3} + \dots + \alpha \cdot a_{l_{p+1}}) \\ &= v_j' \cdot a_{l_1} + v_j''(b_j - a_{l_1}) = w_j \end{aligned}$$

$$\begin{aligned} \text{Logo } v(\text{LT}_1) &= \sum_{j \in J} v(\text{LT}_1^j) + \sum_{l \in K_1 \cup K_2 - i} u_l^* \cdot a_l \\ &\geq \sum_{j \in J} w_j + \sum_{l \in K_1 \cup K_2 - i} u_l^* \cdot a_l \end{aligned}$$

Assim, $\Delta_i = W[K_1 \cup K_2 - i] - W[K_1 \cup K_2] \geq v(\text{LT}_1) - W[K_1 \cup K_2] \geq$

$$\begin{aligned} &\geq \left(\sum_{j \in J} w_j + \sum_{l \in K_1 \cup K_2 - i} u_l^* \cdot a_l \right) - \left(\sum_{l \in K_1 \cup K_2} u_l^* \cdot a_l + \sum_{j \in J} b_j \cdot v_j^* \right) \\ &= \sum_{j \in J} (w_j - b_j \cdot v_j^*) - a_i \cdot u_i^* \end{aligned}$$

Definimos

$$\Delta_i^{(\phi)} := \sum_{j \in J} (w_j - b_j \cdot v_j^*) - a_i \cdot u_i^* \quad (6)$$

TEOREMA 2.1

(i) $\Delta_i^{(\phi)} \geq \Delta_i^D$

(ii) Se existe $j_0 \in J$ tal que $a_{l_1} < b_{j_0}$ e $v'_{j_0} < v''_{j_0}$ então $\Delta_i^{(\phi)} > \Delta_i^D$

PROVA

(i) Claramente se só dá-se o caso (a), ou seja $a_{l_1} \geq b_j, \forall j \in J$ então

$$\begin{aligned} \Delta_i^{(\phi)} &= \sum_{j \in J} (w_j - b_j \cdot v_j^*) - a_i \cdot u_i^* = \sum_{j \in J} (b_j v'_j - b_j v_j^*) - a_i \cdot u_i^* \\ &= \sum_{j \in J} (v'_j - v_j^*) \cdot b_j - a_i \cdot u_i^* = \Delta_i^D. \end{aligned}$$

$$\begin{aligned} \text{Agora, } \Delta_i^{(\phi)} \geq \Delta_i^D &\Leftrightarrow \sum_{j \in J} w_j - \sum_{j \in J} b_j v_j^* \geq \sum_{j \in J} v'_j \cdot b_j - \sum_{j \in J} b_j \cdot v_j^* \\ &\Leftrightarrow \sum_{j \in J} w_j \geq \sum_{j \in J} b_j \cdot v'_j \end{aligned}$$

Logo é só provar que $w_j \geq b_j \cdot v'_j, \forall j \in J$.

Se j satisfaz (a) então $w_j = b_j \cdot v'_j$

$$\begin{aligned} \text{Se } j \text{ satisfaz (b) então } w_j &= v'_j \cdot a_{l_1} + v''_j (b_j - a_{l_1}) \\ &\geq v'_j (a_{l_1} + b_j - a_{l_1}) = b_j \cdot v'_j \end{aligned}$$

Logo $w_j \geq b_j \cdot v'_j, \forall j \in J$

(ii) $w_{j_0} = v'_{j_0} \cdot a_{l_1} + v''_{j_0} (b_{j_0} - a_{l_1})$ pois j_0 satisfaz (b)

$$\begin{aligned} &> v'_{j_0} \cdot a_{l_1} + v'_{j_0} (b_{j_0} - a_{l_1}) \\ &= v'_{j_0} (a_{l_1} + b_{j_0} - a_{l_1}) \\ &= v'_{j_0} \cdot b_{j_0} \end{aligned}$$

$$\text{Logo, } \sum_{j \in J} w_j > \sum_{j \in J} b_j \cdot v'_j \Rightarrow \Delta_i^{(\phi)} > \Delta_i^D \quad \blacksquare$$

Observação

A simplificação proposta em (D), que só usa v'_j e v''_j pode-se estender de maneira natural para outros casos em que sejam usados v', v'', v''' ou v', v'', v''', v'''' , etc. O caso limite é

a resolução total da mochila, como foi proposto no cálculo de $\Delta_l^{(5)}$. Nos problemas testes apresentados em 2.3, os limites $\Delta_l^{(5)}$ e $\Delta_l^{(6)}$ se mostram absolutamente equivalentes, razão pela qual só será considerado no que segue, o último deles.

O cálculo de $\Delta_l^{(6)}$ requer diferentemente de Δ_l^D , o cálculo de v_j' , para cada $j \in J$. Porém, este incremento no custo computacional resulta pouco significativo na prática.

O seguinte exemplo mostra um caso onde acontece $\Delta_l^{(6)} > \Delta_l^D$

EXEMPLO 2.2

Tabela de custos variáveis ou de transporte

		centros de demanda			capacidades
		1	2	3	a_i
centros de oferta	1	1	3	5	100
	2	8	6	2	100
	3	5	4	1	60
demandas		b_j	110	30	60

Consideremos $K_1 \cup K_2 = \{1, 2, 3\}$ e $i=3$.

A solução ótima de $T[K_1 \cup K_2]$ é $x_{11}^* = 100$; $x_{23}^* = 40$; $x_{31}^* = 10$; $x_{32}^* = 30$; $x_{33}^* = 20$; entanto que a solução ótima dual é $u_1^* = -5$; $u_2^* = 0$; $u_3^* = -1$; $v_1^* = 6$; $v_2^* = 5$; $v_3^* = 2$. O valor ótimo deste problema é $W[K_1 \cup K_2] = 370$.

Para o problema $T[K_1 \cup K_2 - 3]$, a solução ótima primal é $\hat{x}_{11} = 100$; $\hat{x}_{21} = 10$; $\hat{x}_{22} = 30$; $\hat{x}_{23} = 60$ com valor ótimo $W[K_1 \cup K_2 - 3] = 480$.

Assim, $\Delta_3 = W[K_1 \cup K_2 - 3] - W[K_1 \cup K_2] = 480 - 370 = 110$.

Cálculo de Δ_3^D

Temos $v_1' = \min_{l=1,2} \{c_{l,1} - u_l^*\} = \min\{6, 8\} = 6$. Aqui $l_1 = 1$. Também $v_1'' = 8$

$v_2' = \min_{l=1,2} \{c_{l,2} - u_l^*\} = \min\{8, 6\} = 6$. Aqui $l_1 = 2$. Também $v_2'' = 8$

$v_3' = \min_{l=1,2} \{c_{l,3} - u_l^*\} = \min\{10, 2\} = 2$. Aqui $l_1 = 2$. Também $v_3'' = 10$

Logo $\Delta_3^D = \sum_{j=1}^3 (v_j' - v_j^*) \cdot b_j - a_3 u_3^* = 0 \cdot 100 + 1 \cdot 30 + 0 \cdot 60 - 60(-1) = 90$

Cálculo de $\Delta_3^{(\infty)}$

$$\begin{aligned} \text{Para } j=1, \text{ temos } a_{l_1} = a_1 = 100 < b_1 = 110 &\Rightarrow w_1 = v_1' a_1 + v_1'' (b_1 - a_1) \\ &\Rightarrow w_1 = 6 \cdot 100 + 8 \cdot 10 = 680 \end{aligned}$$

$$\text{Para } j=2, \text{ temos } a_{l_1} = a_2 = 100 \geq b_2 = 30 \Rightarrow w_2 = b_2 v_2' = 30 \cdot 6 = 180$$

$$\text{Para } j=3, \text{ temos } a_{l_1} = a_2 = 100 \geq b_3 = 60 \Rightarrow w_3 = b_3 v_3' = 60 \cdot 2 = 120$$

$$\begin{aligned} \text{Logo } \Delta_3^{(\infty)} &= \sum_{j=1}^3 (w_j - b_j v_j^*) - a_3 u_3^* \\ &= (680 - 110 \cdot 6) + (180 - 30 \cdot 5) + (120 - 60 \cdot 2) - 60(-1) = 110. \end{aligned}$$

Em consequência, $\Delta_3 = \Delta_3^{(\infty)} > \Delta_3^D$. ■

2.2 LIMITES SUPERIORES PARA Ω_i

Passamos a analisar o teste para fechar facilidades :

Teste 2 : Se $\Omega_i \leq f_i$ então $y_i = 0$, onde $\Omega_i := W[K_1] - W[K_1 U_i]$

Como já foi antecipado, o objetivo é encontrar limites superiores l.s. para Ω_i , de modo tal que ele possa ser usado num teste alternativo:

Teste 2' : Se l.s. $\leq f_i$ então $\Omega_i \leq f_i$ (e logo $y_i = 0$)

Neles serão usadas x^*, u^*, v^* , as soluções ótimas de $T[K_1]$ e seu dual.

2.2.1 Limite Superior de Akinc-Khumawala([2])

Aquí os autores obtêm um limite superior para Ω_i através de um limite inferior para o valor $W[K_1 U_i]$ do problema $T[K_1 U_i]$ dado por

$$T[K_1 U_i] = \left\{ \begin{array}{l} \text{minimizar } \sum_{l \in K_1 U_i} \sum_{j \in J} c_{lj} x_{lj} \\ \text{s.a. } \sum_{j \in J} x_{ij} \leq a_i \quad (R1) \\ \sum_{j \in J} x_{lj} \leq a_l ; \forall l \in K_1 \quad (R2) \\ \sum_{j \in J} x_{lj} = b_j ; \forall j \in J \quad (R3) \\ x_{lj} \geq 0 ; \forall l \in K_1 U_i ; \forall j \in J \quad (R4) \end{array} \right.$$

Esse limite inferior resulta de resolver o problema em que temos relaxado as restrições (R2) e (R3) de $T[K_1 U_i]$.

O limite superior para Ω_i resultante é:

$$\bar{\Omega}_i = \max \left\{ \sum_{j \in J} (v_j^* - c_{ij}) \cdot \hat{x}_{ij} / \sum_{j \in J} \hat{x}_{ij} \leq a_i; 0 \leq \hat{x}_{ij} \leq b_j \right\}$$

2.2.2 Limite superior de Mateus e Bornstein ([50], [51])

Aquí, o limite inferior para $T[K_1 U_i]$ é uma solução dual viável desse problema. Com isso obtém-se:

$$\Omega_i^D = \sum_{l \in K_1} \sum_{j \in J} x_{lj}^* \cdot \max \{ 0; v_j^* - c_{lj} \}$$

Nas referências citadas, foi provado que $\bar{\Omega}_i \geq \Omega_i$ e que $\Omega_i^D \geq \Omega_i$.

O que não foi provado nas referências é a desigualdade $\Omega_i^D \leq \bar{\Omega}_i$ que faria de Ω_i^D um limite superior melhor do que $\bar{\Omega}_i$, pelo menos do ponto de vista teórico.

O exemplo a seguir mostra que pode acontecer $\Omega_i^{(D)} > \bar{\Omega}_i$

EXEMPLO 2.3

Tabela de custos variáveis ou de transporte

		centros de demanda			capacidades
		1	2	3	
centros de oferta	1	1	3	4	100
	2	3	6	5	100
	3	7	5	2	50
demandas	b_j	70	70	60	

Aquí consideramos $K_1 = \{1, 2\}$ e $i=3$.

A solução ótima de $T[K_1]$ é : $x_{11}^* = 30$; $x_{12}^* = 70$; $x_{21}^* = 40$; $x_{23}^* = 60$.

A solução ótima dual é : $u_1^* = -2$; $u_2^* = 0$; $v_1^* = 3$; $v_2^* = 5$; $v_3^* = 5$.

O valor ótimo de $T[K_1]$ é $W[K_1] = 660$.

Para o problema $T[K_1 \cup 3]$, a solução ótima é : $\hat{x}_{11} = 30$; $\hat{x}_{12} = 70$;

$$\hat{x}_{21} = 40; \hat{x}_{23} = 10; \hat{x}_{33} = 50; \text{ com valor } W[K_1 \cup 3] = 510.$$

$$\text{Logo, } \Omega_3 = W[K_1] - W[K_1 \cup 3] = 660 - 510 = 150.$$

Cálculo de $\bar{\Omega}_3$:

$$\bar{\Omega}_3 = \max \left\{ \sum_{j=1}^3 (v_j^* - c_{3j}) \hat{x}_{3j} / \sum_{j=1}^3 \hat{x}_{3j} \leq 50; 0 \leq \hat{x}_{31} \leq 70; 0 \leq \hat{x}_{32} \leq 70; 0 \leq \hat{x}_{33} \leq 60 \right\}$$

Temos:

j	v_j^*	c_{3j}	$v_j^* - c_{3j}$
1	3	7	-4
2	5	5	0
3	5	2	3

$$\text{Logo } \bar{\Omega}_3 = \max \left\{ (-4) \hat{x}_{31} + 0 \cdot \hat{x}_{32} + 3 \cdot \hat{x}_{33} / \hat{x}_{31} + \hat{x}_{32} + \hat{x}_{33} \leq 50; 0 \leq \hat{x}_{32} \leq 70; 0 \leq \hat{x}_{31} \leq 70; 0 \leq \hat{x}_{33} \leq 60 \right\}$$

$$= (-4) \cdot 0 + 0 \cdot 0 + 3 \cdot 50 = 150.$$

Cálculo de Ω_3^D :

$$\begin{aligned} \Omega_3^D &= \sum_{l=1}^2 \sum_{j=1}^3 x_{lj}^* \cdot \max\{0; v_j^* - c_{3j}\} \\ &= x_{11}^* \cdot \max\{0; -4\} + x_{12}^* \cdot \max\{0; 0\} + x_{13}^* \cdot \max\{0; 3\} + x_{21}^* \cdot \max\{0; -4\} \\ &\quad + x_{22}^* \cdot \max\{0; 0\} + x_{23}^* \cdot \max\{0; 3\} \end{aligned}$$

$$= 30 \cdot 0 + 70 \cdot 0 + 0 \cdot 3 + 40 \cdot 0 + 0 \cdot 0 + 60 \cdot 3 = 180$$

$$\text{Logo } \Omega_3 = \bar{\Omega}_3 = 150 < \Omega_3^D = 180. \quad \blacksquare$$

A seguir provaremos que $\Omega_i^D \geq \bar{\Omega}_i$, resultando Ω_i^D um limite superior pior do que $\bar{\Omega}_i$, pelo menos do ponto de vista teórico.

$$\begin{aligned} \Omega_i^D &= \sum_{l \in K_1} \sum_{j \in J} x_{lj}^* \cdot \max\{0; v_j^* - c_{lj}\} = \sum_{j \in J} \left[\max\{0; v_j^* - c_{lj}\} \cdot \sum_{l \in K_1} x_{lj}^* \right] \\ &= \sum_{j \in J} b_j \cdot \max\{0; v_j^* - c_{lj}\} \end{aligned}$$

$$\bar{\Omega}_i = \max \left\{ \sum_{j \in J} (v_j^* - c_{lj}) \cdot \hat{x}_{lj} / \sum_{j \in J} \hat{x}_{lj} \leq a_i; 0 \leq \hat{x}_{lj} \leq b_j \right\}$$

Se $v_j^* - c_{ij} \leq 0$ então $\begin{cases} \hat{x}_{ij} = 0, \text{ no cálculo de } \bar{\Omega}_i \\ \max\{0, v_j^* - c_{ij}\} = 0, \text{ no cálculo de } \Omega_i^D \end{cases}$

Logo só consideraremos $J^* = \{j \in J / v_j^* - c_{ij} > 0\}$

Neste caso, $\max\{0, v_j^* - c_{ij}\} = v_j^* - c_{ij}, \forall j \in J^*$.

Se $\sum_{j \in J} b_j \leq a_i$ então $\bar{\Omega}_i = \sum_{j \in J} (v_j^* - c_{ij}) \cdot b_j$
 $= \sum_{j \in J} \max\{0, v_j^* - c_{ij}\} \cdot b_j = \Omega_i^D$

Se $\sum_{j \in J} b_j > a_i$ então $\bar{\Omega}_i$ é calculado ordenando os $v_j^* - c_{ij} = k_{ij}$ em ordem decrescente, i.e. $k_{ij_1} \geq k_{ij_2} \geq k_{ij_3} \geq \dots \geq k_{ij_r} > 0$, onde $|J^*| = r$

Podemos supor que existe $p \leq r$ e $0 \leq \alpha < 1$ tal que

$$a_i = b_{j_1} + b_{j_2} + \dots + b_{j_p} + \alpha \cdot b_{j_{p+1}}$$

Então, $\bar{\Omega}_i = k_{ij_1} \cdot b_{j_1} + k_{ij_2} \cdot b_{j_2} + \dots + k_{ij_p} \cdot b_{j_p} + \alpha \cdot k_{ij_{p+1}} \cdot b_{j_{p+1}}$
 $< k_{ij_1} \cdot b_{j_1} + k_{ij_2} \cdot b_{j_2} + \dots + k_{ij_{p+1}} \cdot b_{j_{p+1}} + \dots + k_{ij_r} \cdot b_{j_r}$
 $= \sum_{t=1}^r k_{ij_t} \cdot b_{j_t} = \sum_{t=1}^r (v_{j_t}^* - c_{ij_t}) \cdot b_{j_t} = \Omega_i^D$

Logo, no caso em que $\sum_{j \in J} b_j > a_i$, podemos ter $\bar{\Omega}_i < \Omega_i^D$.

2.2.3 Proposta de outro limite superior para Ω_i

A ideia agora consiste em usar como limite inferior para $W[K_1 U_i]$ o valor $W^{NC}[K_1 U_i]$ que é definido como o valor ótimo do problema de localização não capacitado associado com $T[K_1 U_i]$.

Evidentemente teremos $W[K_1 U_i] \geq W^{NC}[K_1 U_i]$.

Logo, $\Omega_i = W[K_1] - W[K_1 U_i] \leq W[K_1] - W^{NC}[K_1 U_i] := \Omega_i^*$.

As expressões Ω_i^* e $\bar{\Omega}_i$ não são comparáveis no sentido em que existem instâncias nas quais $\Omega_i^* > \bar{\Omega}_i$, e outras em que se verifica a desigualdade contrária. Porém, é bom observar a simplicidade que implica o cálculo de Ω_i^* , ou seja, o cálculo de $|J|$ mínimos. Devido a este fato, é possível implementar o

teste 2', usando como limite superior para Ω_i a expressão :

$$\hat{\Omega}_i = \min\{\bar{\Omega}_i; \Omega_i^*\}$$

EXEMPLO 2.4

Consideremos novamente os dados do exemplo 2.3.

Nele, $K_1 = \{1, 2\}$ e $i=3$. $W[K_1] = 660$, $W[K_1 \cup 3] = 510$, $\Omega_3 = 150$, $\bar{\Omega}_3 = 150$ e $\Omega_3^D = 180$.

Agora, $\Omega_3^* = 660 - W^{NC}[K_1 \cup i] = 660 - (1 \cdot 70 + 3 \cdot 70 + 2 \cdot 60) = 240$

Aqui dá-se $\Omega_3 = \bar{\Omega}_3 < \Omega_3^D < \Omega_3^*$. Também temos $\hat{\Omega}_3 = \min\{150, 240\} = 150$.

EXEMPLO 2.5

Consideremos agora a seguinte instância:

Tabela de custos variáveis ou de transporte

		centros de demanda			capacidades
		1	2	3	a_i
centros de oferta	1	1	5	6	100
	2	5	1	6	100
	3	5	6	1	60
demandas		b_j	70	70	60

De novo fazemos $K_1 = \{1, 2\}$ e $i=3$.

A solução ótima de $T[K_1]$ é $x_{11}^* = 70$; $x_{13}^* = 30$; $x_{22}^* = 70$; $x_{23}^* = 30$, com valor $W[K_1] = 440$.

A solução ótima para $T[K_1 \cup 3]$ é $x_{11} = 70$; $x_{22} = 70$; $x_{33} = 60$, com valor $W[K_1 \cup 3] = W^{NC}[K_1 \cup 3] = 200$.

Assim, $\Omega_3 = 440 - 200 = 240$; $\Omega_3^* = 440 - 200 = 240$; $\bar{\Omega}_3 = 5 \cdot 60 = 300$ e $\Omega_3^D = 30 \cdot 5 + 30 \cdot 5 = 300$. Logo, $\hat{\Omega}_3 = \min\{240; 300\} = 240$.

Neste caso temos $\Omega_3 = \Omega_3^* = 240 < \bar{\Omega}_3 = \Omega_3^D = 300$.

2.3 PROBLEMAS TESTES

Os problemas ou instâncias testes utilizados neste trabalho são aqueles formalizados por Kuehn e Hamburger[38] e outros.

Trata-se de instâncias frequentemente utilizadas pelos pesquisadores para comparar os seus resultados referente ao problema de localização capacitado. Estas são construídas usando as distâncias entre 50 cidades. Os custos variáveis ou de transporte resultam de multiplicar essas distâncias por um custo unitário por unidade de distância que por razões de espaço não são aqui apresentadas. Maior detalhe neste sentido pode-se encontrar em Rangel [49], porém, os dados utilizados em toda nossa experiência computacional, forem obtidos através da OR-Library: distributing tests problems by electronic mail (Beasley-1990).

Cada cidade tem associada uma demanda e cada instância é gerada usando:

- as 50 cidades como centros de demanda
- um número m representando a quantidade de centros de oferta ou de armazenamento (m assume valores 16, 25 ou 50 dependendo da instância)
- custos fixos idênticos para todos os centros de oferta considerados (estes assumem valores 7500, 12500, 17500 ou 25000 dependendo da instância)
- ofertas idênticas para todos os centros de oferta (esta assume os valores 5000, 10000 ou 15000 dependendo da instância).

Deste modo, geram-se 25 instâncias teste, distribuídas em 7 grupos, com diferentes especificações como se indica na tabela 2.1. A denominação do grupo é a dada nos arquivos de Beasley.

O fato de dispor do valor ótimo para cada um destes problemas, faz com que os nossos resultados possam ser devidamente comparados com eles, garantindo assim um bom estudo em termos da qualidade das soluções obtidas pelo algoritmo a ser desenvolvido em capítulos posteriores deste trabalho.

2.4 EXPERIENCIA COMPUTACIONAL

Nesta parte serão apresentadas os resultados obtidos no cálculo dos diferentes limites inferiores para Δ e dos limites superiores para Ω quando aplicados nos problemas testes. No primeiro caso só será necessário considerar um problema por cada grupo (CAP41, CAP42, CAP43, CAP44, CAP51, CAP61, CAP62, CAP63, CAP64, CAP81, CAP82, CAP83, CAP84, CAP91, CAP92, CAP93, CAP94, CAP111, CAP112, CAP113, CAP114, CAP121, CAP122, CAP123, CAP124) devido a que os cálculos não são afetados por mudanças nos custos fixos.

TABELA 2.1

TABELA DE PROBLEMAS TESTES

Grupo	Nome da Instância	Nº centros oferta	oferta	custo oferta	custo fixo	Nº centros demanda	valor ótimo
4	CAP41	16	5000	7500	50	1.040.444,975	1.040.444,975 1.098.000,450 1.153.000,450 1.235.500,450
	CAP42	16	5000	12500	50	1.098.000,450	
	CAP43	16	5000	17500	50	1.153.000,450	
	CAP44	16	5000	25000	50	1.235.500,450	
5	CAP51	16	10000	7500	50	1.025.208,225	1.025.208,225
	CAP61	16	15000	7500	50	932.615,750	
	CAP62	16	15000	12500	50	977.799,400	
	CAP63	16	15000	17500	50	1.014.062,050	
6	CAP64	16	15000	25000	50	1.045.650,250	1.045.650,250
	CAP81	25	5000	7500	50	838.499,288	
	CAP82	25	5000	12500	50	910.889,563	
	CAP83	25	5000	17500	50	975.889,563	
8	CAP84	25	5000	25000	50	1.069.369,525	1.069.369,525
	CAP91	25	15000	7500	50	796.648,438	
	CAP92	25	15000	12500	50	855.733,500	
	CAP93	25	15000	17500	50	896.617,538	
9	CAP94	25	15000	25000	50	946.051,925	946.051,925
	CAP111	50	5000	7500	50	826.124,713	
	CAP112	50	5000	12500	50	901.977,213	
	CAP113	50	5000	17500	50	970.567,750	
11	CAP114	50	5000	25000	50	1.069.356,488	1.069.356,488
	CAP121	50	15000	7500	50	799.439,563	
	CAP122	50	15000	12500	50	852.524,625	
	CAP123	50	15000	17500	50	895.302,925	
12	CAP124	50	15000	25000	50	946.051,925	946.051,925

No caso dos limites superiores para Ω_i , é necessário ter aplicado pelo menos uma vez o teste para abrir facilidades (teste 1) e ter obtido um conjunto K_1 de facilidades abertas, que seja viável. Isto aconteceu somente nos problemas CAP51, CAP61, CAP91 e CAP121.

Na tabela 2.2, Δ_i corresponde ao cálculo exato dado por (3) e $\bar{\Delta}_i$, Δ_i^D e $\Delta_i^{(6)}$, são as aproximações dadas por (4), (5) e (6) respectivamente.

TABELA 2.2

TABELA DE RESULTADOS COMPARATIVOS DE APROXIMAÇÕES DELTAS

		Δ_i	$\bar{\Delta}_i$	Δ_i^D	$\Delta_i^{(6)}$
CAP41	t (seg.)	16.31	0.54	1.04	1.09
	E%		56.19	21.29	20.95
	I%		18.75	25.00	25.00
	AP%		25.00	87.50	100.00
CAP51	t (seg.)	22.18	0.54	1.31	1.31
	E%		7.90	4.77	3.79
	I%		68.75	75.00	75.00
	AP%		75.00	81.25	100.00
CAP61	t (seg.)	22.57	0.49	1.37	1.37
	E%		2.53	2.53	2.53
	I%		81.25	81.25	81.25
	AP%		100.00	100.00	100.00
CAP81	t (seg.)	65.52	0.98	2.52	2.58
	E%		17.87	9.17	7.34
	I%		56.00	64.00	68.00
	AP%		64.00	92.00	100.00
CAP91	t (seg.)	73.60	0.98	2.85	2.91
	E%		0.99	0.99	0.99
	I%		92.00	92.00	92.00
	AP%		100.00	100.00	100.00
CAP111	t (seg.)	479.11	3.02	9.61	9.61
	E%		8.80	6.02	4.87
	I%		82.00	86.00	88.00
	AP%		84.00	96.00	100.00
CAP121	t (seg.)	495.20	3.02	9.83	9.88
	E%		0.43	0.43	0.43
	I%		94.00	94.00	94.00
	AP%		100.00	100.00	100.00

São também usadas as seguintes notações:

- t(seg.) representa o tempo computacional consumido em calcular os valores delta para todas as facilidades.
- E% representa o erro percentual medio relativo do valor aproximado respeito do valor exato Δ_i .
- I% representa a porcentagem das vezes em que o cálculo aproximado coincide com o valor exato Δ_i .
- AP% representa a porcentagem das vezes em que o cálculo aproximado corresponde à melhor aproximação dentre todas

Uma primeira observação nos resultados da tabela 2.2 é a enorme redução no tempo computacional conseguida usando aproximações. Junto com isso, pode-se ver que o menor erro (E%) se produz em $\Delta_i^{(6)}$ alem de que também apresenta AP% de 100%, que se traduz em que ela é a melhor aproximação em todos os casos.

Também, $\Delta_i^{(6)}$ tem o maior I% mostrando-se que este tipo de aproximação é a que mais vezes coincide com o valor exato de Δ_i .

Finalmente, se são comparados os tempos de Δ_i^D com os de $\Delta_i^{(6)}$ pode-se observar que não existe diferença significativa entre eles.

Como consequência destes calculos podemos concluir que $\Delta_i^{(6)}$ é uma ótima aproximação para Δ_i , se comparada com as existentes. Ela será utilizada no Capítulo 5 na implementação do teste de redução bem como na implementação do algoritmo Simulated Annealing para o Problema de Localização Capacitado (CWLP).

Na tabela 2.3 são apresentados os resultados obtidos com as aproximações de Ω_i . Aqui são usadas $\bar{\Omega}_i$ (ver 2.2.1), Ω_i^D (ver 2.2.2), $\hat{\Omega}_i$ (ver 2.2.3), e as mesmas notações que na tabela 2.2.

Os resultados obtidos mostram um comportamento muito similar entre as três aproximações usadas. Só no que respeita ao arquivo CAP121, resulta $\hat{\Omega}_i$ mais eficiente que as outras em termos de E%, I% e AP%, apresentando no entanto um incremento no tempo computacional. Este tempo no entanto é bem menor que o tempo necessario para calcular Ω_i . Em consequência, se consideramos razoável o excesso de tempo consumido no cálculo de $\hat{\Omega}_i$, este

limite pode perfeitamente ser utilizado num teste de redução do CWLP.

TABELA 2.3

TABELA DE RESULTADOS COMPARATIVOS DE APROXIMAÇÕES OMEGA

		Ω_i	$\bar{\Omega}_i$	Ω_i^D	$\hat{\Omega}_i$
CAP51	t(seg.)	11.20	2.58	2.85	3.29
	E%		12.70	12.70	12.70
	I%		85.70	85.70	85.70
	AF%		100.00	100.00	100.00
CAP61	t(seg.)	11.80	2.63	2.85	3.29
	E%		0.00	0.00	0.00
	I%		100.00	100.00	100.00
	AF%		100.00	100.00	100.00
CAP91	t(seg.)	44.98	5.65	6.15	7.52
	E%		0.00	0.00	0.00
	I%		100.00	100.00	100.00
	AF%		100.00	100.00	100.00
CAP121	t(seg.)	395.79	18.61	19.99	27.57
	E%		34.66	25.90	8.13
	I%		46.50	46.50	53.50
	AF%		81.40	81.40	100.00

CAPÍTULO 3

ELEMENTOS DE CADEIAS DE MARKOV

Neste capítulo serão usados os seguintes conceitos e notações probabilísticas :

$P[A]$: a probabilidade do evento A

$P[A, B]$ ou $P[A \cap B]$: a probabilidade de que A e B aconteçam

$P[A \cup B]$: a probabilidade de que A ou B aconteçam

O seguinte resultado relaciona as probabilidades acima :

$$P[A \cup B] = P[A] + P[B] - P[A \cap B]$$

Dois eventos são ditos " mutuamente exclusivos" se $A \cap B = \emptyset$. Neste caso, $P[A \cup B] = P[A] + P[B]$, pois $P[A \cap B] = P[\emptyset] = 0$.

Uma família $\{ A_i \}_{i=1}^n$ de eventos é "mutuamente exclusiva" se

$$A_i \cap A_j = \emptyset \quad , \quad \forall i, \forall j, \quad i \neq j$$

Neste caso resulta $P\left[\bigcup_{i=1}^n A_i \right] = \sum_{i=1}^n P[A_i]$

Se $P[B] > 0$, a probabilidade de A dado que B tem acontecido (conhecida como a probabilidade condicional de A dado B) é definida por $P[A/B] = \frac{P[A \cap B]}{P[B]}$

De aqui temos $P[A \cap B] = P[A/B] \cdot P[B]$, sendo esta relação válida também no caso $P[B] = 0$.

Dado que $A \cap B = B \cap A$, então também se verifica

$$P[A \cap B] = P[A] \cdot P[B/A]$$

Usando repetidamente esta última relação obtemos

$$\begin{aligned} P[A_1 \cap A_2 \cap \dots \cap A_n] &= \\ &= P[A_1] \cdot P[A_2/A_1] \cdot P[A_3/A_1, A_2] \cdot \dots \cdot P[A_n/A_1, A_2, \dots, A_{n-1}] \end{aligned} \quad (1)$$

Seja Ω o espaço amostral sobre o qual é definida a função

de probabilidade P e seja $\{B_i\}_{i=1}^n$ uma partição de Ω , ou seja $\bigcup_{i=1}^n B_i = \Omega$ e $\{B_i\}_{i=1}^n$ é mutuamente exclusiva.

Então

$$\begin{aligned} P[A] &= P[A \cap \Omega] = P\left[A \cap \left(\bigcup_{i=1}^n B_i\right)\right] \\ &= P\left[\bigcup_{i=1}^n (A \cap B_i)\right] \\ &= \sum_{i=1}^n P[A \cap B_i] \quad , \text{ pois } \{A \cap B_i\}_{i=1}^n \text{ é mutuamente exclusiva} \\ &= \sum_{i=1}^n P[B_i] \cdot P[A/B_i] \end{aligned}$$

O Teorema das Probabilidades Totais estabelece esse resultado, ou seja, $P[A] = \sum_{i=1}^n P[B_i] \cdot P[A/B_i]$ (2)

Uma Variável Aleatória definida sobre um espaço amostral Ω é uma função $X: \Omega \rightarrow \mathbb{R}$. Se $i \in \mathbb{R}$, a notação $P[X=i]$ representa $P[\{\omega \in \Omega / X(\omega)=i\}]$.

Um PROCESSO ESTOCASTICO (P.E) é uma familia de variáveis aleatórias $\{X_t / t \in T\}$ definidas sobre o mesmo espaço amostral Ω .

Elementos fundamentais num P.E. são o espaço de estados, o conjunto de índices T e as relações de dependência entre as variáveis aleatórias X .

O Espaço de Estados S é o conjunto dos possíveis valores das variáveis aleatórias X_t .

Se S é finito ($S = \{0,1,2,\dots,n\}$) ou numerável ($S = \{0,1,2,\dots\}$) o processo se diz "de estado discreto" e se chama CADEIA.

Se o conjunto de índices T é numerável, o processo é chamado de TEMPO DISCRETO e é representado por $\{X_n\}$. Caso T seja não numerável, o processo é dito de TEMPO CONTINUO e é representado por $\{X_t / t \geq 0\}$.

Um processo estocástico $\{X_n\}; n=1,2,3,\dots$ com espaço de estados $S = \{0,1,2,\dots\}$, ou seja uma cadeia de tempo discreto,

satisfaz a PROPRIEDADE DE MARKOV se para cada n e quaisquer estados i_1, i_2, \dots, i_n , é verificada a seguinte igualdade :

$$P[X_n = i_n / X_{n-1} = i_{n-1}, X_{n-2} = i_{n-2}, \dots, X_1 = i_1] = P[X_n = i_n / X_{n-1} = i_{n-1}] \quad [3]$$

A propriedade markoviana estabelece que a probabilidade de qualquer comportamento futuro do processo, conhecido o seu comportamento presente e passado, depende unicamente do estado presente e é independente de estados anteriores a ele. (O processo não tem memória).

Uma CADEIA DE MARKOV DE TEMPO DISCRETO é uma cadeia de tempo discreto que satisfaz a propriedade de Markov. Só este tipo de processos seram estudados a seguir.

Quando for utilizada a notação $X_n = i$, será para representar o fato de que a cadeia está (ou se encontra, ou atinge) o estado i no tempo n . Suponhamos que uma cadeia de Markov se encontra no estado i no tempo $n-1$, ou seja, $X_{n-1} = i$. Será possível conhecer o estado que a cadeia atinge no tempo n ? Uma resposta exata não é sempre possível, no máximo poderemos ter a distribuição de probabilidades sobre os diferentes estados em S , condicionado à locação no tempo $n-1$. Esta distribuição condicional é dada por

$$P[X_n = j / X_{n-1} = i], \forall j \in S$$

DEFINIÇÃO 3.1

Para cada estado i e j definimos a PROBABILIDADE DE TRANSIÇÃO NUM PASSO por $P_{ij}^{n,n+1} = P[X_{n+1} = j / X_n = i]$ e representa a probabilidade de que a cadeia se encontre no estado j no tempo $n+1$ se é conhecido que no tempo n a cadeia está no estado i . ■

Estas probabilidades dependem do estado inicial i , do estado final j e do tempo da transição n .

DEFINIÇÃO 3.2

Uma cadeia de Markov de tempo discreto é ESTACIONARIA ou HOMOGENEA no tempo, se a probabilidade de passar de um estado para outro é independente do instante de tempo no qual o passo é feito.

Ou seja, se para quaisquer estados i, j se satisfaz

$$P[X_n = j / X_{n-1} = i] = P[X_{n+k} = j / X_{n+k-1} = i]$$

para $k = -(n-1), -(n-2), \dots, -1, 0, 1, 2, \dots$ ■

Neste caso $P_{ij}^{n, n+1} = P[X_n = j / X_{n-1} = i]$ é denotada por P_{ij} e representa a probabilidade de passar do estado i ao estado j num passo. No que segue, só serão consideradas cadeias de Markov homogêneas.

Denotamos por $P = (P_{ij}) = \begin{pmatrix} P_{00} & P_{01} & P_{02} & \dots \\ P_{10} & P_{11} & P_{12} & \dots \\ \vdots & \vdots & \vdots & \vdots \\ P_{i0} & P_{i1} & P_{i2} & \dots \\ \vdots & \vdots & \vdots & \vdots \end{pmatrix}$ a matriz das

probabilidades de transição num passo, também chamada MATRIZ MARKOVIANA.

Claramente, no caso que S for finito, a matriz P é uma matriz quadrada de ordem $|S| \times |S|$.

A matriz P satisfaz as seguintes propriedades:

(a) $P_{ij} \geq 0, \forall i, j \in S$

(b) $\sum_{j=0}^{\infty} P_{ij} = 1, \forall i \in S$

Estas propriedades caracterizam as chamadas MATRIZES ESTOCÁSTICAS.

A condição (b) expressa o fato de que com probabilidade 1, alguma transição acontece em cada passo.

Toda cadeia de Markov de tempo discreto fica totalmente especificada mediante a matriz de probabilidades de transição P e a distribuição de probabilidades de X_0 . (Aqui, $T = \{0, 1, 2, \dots\}$) Efetivamente, seja $P[X_0 = i] = p_i$

$$\begin{aligned}
 &P[X_0=i_0, X_1=i_1, X_2=i_2, \dots, X_n=i_n] = \\
 &= P[X_0=i_0] \cdot P[X_1=i_1/X_0=i_0] \cdot \dots \cdot P[X_n=i_n/X_0=i_0, X_1=i_1, \dots, X_{n-1}=i_{n-1}] \\
 & \hspace{20em} \text{(devido a (1))}
 \end{aligned}$$

$$\begin{aligned}
 &= p_{i_0} \cdot P_{i_0, i_1} \cdot \dots \cdot P[X_n=i_n/X_{n-1}=i_{n-1}] \hspace{10em} \text{(devido a (3))} \\
 &= p_{i_0} \cdot P_{i_0, i_1} \cdot P_{i_1, i_2} \cdot \dots \cdot P_{i_{n-1}, i_n}
 \end{aligned}$$

Alem disto, toda probabilidade que envolve $X_{j_1}, X_{j_2}, \dots, X_{j_k}$ sendo $j_1 < j_2 < \dots < j_k$, pode ser obtida por somas de termos do tipo $P[X_0=i_0, \dots, X_n=i_n]$ usando o teorema das probabilidades totais (2).

Assim, por exemplo,

$$\begin{aligned}
 P[X_1=i] &= P[X_1=i/X_0=0] \cdot P[X_0=0] + P[X_1=i/X_0=1] \cdot P[X_0=1] + \dots \\
 &= P_{i,0} \cdot p_0 + P_{i,1} \cdot p_1 + \dots \\
 &= \sum_{j=0}^{\infty} P_{ij} \cdot p_j
 \end{aligned}$$

EXEMPLO 3.1 (Caminho Aleatório Unidimensional)

O caminho aleatório unidimensional é uma cadeia de Markov onde o espaço de estados S é associado à posição de uma partícula móvel, que estando no estado i , numa transição pode permanecer em i ou atingir os estados $i-1$ ou $i+1$. Se S é considerado como o conjunto de inteiros não negativos, a matriz de transição P é dada por:

$$P = \begin{pmatrix} r_0 & p_0 & 0 & 0 & 0 & \dots \\ q_1 & r_1 & p_1 & 0 & 0 & \dots \\ 0 & q_2 & r_2 & p_2 & 0 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}$$

onde $p_i > 0$, $q_i > 0$, $r_i \geq 0$, $q_i + r_i + p_i = 1$ para $i=1, 2, \dots$

$$p_0 \geq 0, r_0 \geq 0, r_0 + p_0 = 1$$

e para $i \geq 1$ temos $P[X_{n+1}=i+1/X_n=i] = p_i$

$$P[X_{n+1}=i-1/X_n=i] = q_i$$

$$P[X_{n+1}=i/X_n=i] = r_i$$

e também $P[X_1=1/X_0=0] = p_0$ e $P[X_1=0/X_0=0] = r_0$

No caso em que S é o conjunto dos números inteiros e a matriz P está dada por

$$P_{ij} = \begin{cases} q & \text{se } i=j+1 \\ p & \text{se } i=j-1 \\ 0 & \text{em outro caso} \end{cases}$$

obtemos outro exemplo de caminho aleatório unidimensional onde obrigadamente a partir de um estado j só é possível atingir os estados $j-1$ ou $j+1$. ■

A análise de uma cadeia de Markov diz respeito principalmente ao cálculo das probabilidades das possíveis realizações do processo. Por exemplo, qual é a probabilidade de que tendo começado no estado i , o processo se encontre no estado j depois de 3 transições? Sempre considerando o caso de cadeias homogêneas, a pergunta corresponde ao cálculo de $P[X_3=j/X_0=i]$.

Isto motiva a seguinte

DEFINIÇÃO 3.3

A probabilidade de que uma cadeia de Markov passe do estado i ao estado j em n transições, é definida por

$$P_{ij}^{(n)} = P[X_{n+m}=j / X_m=i]$$

Denotamos por P a matriz de entradas $P_{ij}^{(n)}$ ■

Da definição anterior temos que $P_{ij}^{(1)} = P_{ij}$

TEOREMA 3.1 (Equação de Chapman-Kolmogorov)

$$P_{ij}^{(n)} = \sum_{k \in S} P_{ik}^{(r)} \cdot P_{kj}^{(s)}$$

sendo r e s inteiros não negativos tais que $r+s=n$

Aqui é definido $P_{ij}^{(0)} = \begin{cases} 1 & \text{se } i=j \\ 0 & \text{se } i \neq j \end{cases}$

Prova:

O evento "passar de i para j em n transições" pode ser realizado de maneira mutuamente exclusiva indo de i para algum estado intermediário k ($k=0,1,2,\dots$) em exatamente r transições

(com probabilidade $P_{ik}^{(r)}$) e depois de k para j em exatamente s transições (com probabilidade $P_{kj}^{(s)}$). Usando o teorema das probabilidades totais (2), temos o resultado. ■

OBSERVAÇÃO

Denotemos por $P_{ij}^{(n)}$ o elemento (i,j) da matriz P^n ($n=2,3,\dots$)

Para $n=1$ temos $P_{ij}^{(1)} = P_{ij}$ (1)

Para $n=2$ temos $P_{ij}^{(2)} = \sum_{k \in S} P_{ik}^{(1)} \cdot P_{kj}^{(1)} = \sum_{k \in S} P_{ik} \cdot P_{kj} = P_{ij}^{(2)}$ (2)

Para $n=3$ temos $P_{ij}^{(3)} = \sum_{k \in S} P_{ik}^{(1)} \cdot P_{kj}^{(2)} = \sum_{k \in S} P_{ik} \cdot P_{kj}^{(2)} = P_{ij}^{(3)}$ (3)

e indutivamente temos

$$P_{ij}^{(n+1)} = \sum_{k \in S} P_{ik}^{(1)} \cdot P_{kj}^{(n)} = \sum_{k \in S} P_{ik} \cdot P_{kj}^{(n)} = P_{ij}^{(n+1)}$$

Conclui-se que a matriz $P^{(n)}$ das transições em n passos é igual à n -ésima potência da matriz P de transições num passo, ou seja $P^{(n)} = P^n$ ■

De aqui em diante, usaremos a notação P^n para representar a matriz de transições em n passos.

COROLARIO 3.1

Se $P[X_0=k] = p_k$ para $k=0,1,2,\dots$

então $P[X_n=j] = \sum_{k \in S} P_{kj}^{(n)} \cdot p_k$

PROVA

$$\begin{aligned} P[X_n=j] &= \sum_{k \in S} P[X_n=j / X_0=k] \cdot P[X_0=k] && \text{devido a (2)} \\ &= \sum_{k \in S} P_{kj}^{(n)} \cdot p_k \end{aligned}$$

A seguir estudaremos condições para a existencia de $\lim_{n \rightarrow \infty} P_{ij}^{(n)}$. Poder-se-ia esperar que a influência do estado inicial i vai diminuindo no tempo, e consequentemente $P_{ij}^{(n)}$ se aproxima de um limite (quando $n \rightarrow \infty$) que é independente de i .

Certamente isto pode efetivamente acontecer, mas para

estabelecer as condições que deve satisfazer a cadeia, precisam-se de outros conceitos que permitem fazer classificações dos estados.

DEFINIÇÃO 3.4

Um estado j é **ACESSIVEL** a partir de um estado i ($i \rightarrow j$) se existe $n \geq 0$ tal que $P_{ij}^{(n)} > 0$, ou seja, se temos probabilidade positiva de que num número finito de transições a cadeia alcança o estado j tendo começado no estado i .
Os estados i e j que são acessíveis mutuamente serão chamados de **COMUNICADOS** ($i \longleftrightarrow j$). ■

TEOREMA 3.2

A relação de comunicação é uma relação de equivalência

PROVA

a) Reflexividade:

$i \longleftrightarrow i$, pois existe $n=0$ tal que $P_{ii}^{(0)} = 1 > 0, \forall i \in S$

b) Simetria:

$i \longleftrightarrow j$ quer dizer que i e j estão comunicados mutuamente, ou seja que também temos $j \longleftrightarrow i$.

c) Transitividade:

Suponhamos $i \longleftrightarrow j$ e $j \longleftrightarrow k$.

Logo $\exists n, m \geq 0$ tais que $P_{ij}^{(n)} > 0$ e $P_{ji}^{(m)} > 0$

$\exists r, s \geq 0$ tais que $P_{jk}^{(r)} > 0$ e $P_{kj}^{(s)} > 0$

Logo $P_{ik}^{(n+r)} = \sum_{l \in S} P_{il}^{(n)} \cdot P_{lk}^{(r)} \geq P_{ij}^{(n)} \cdot P_{jk}^{(r)} > 0$ (devido ao Teor 3.1)

e $P_{ki}^{(m+s)} = \sum_{l \in S} P_{kl}^{(s)} \cdot P_{li}^{(m)} \geq P_{kj}^{(s)} \cdot P_{ji}^{(m)} > 0$.

Logo $\exists n+r, m+s$ tais que $P_{ik}^{(n+r)} > 0$ e $P_{ki}^{(m+s)} > 0$.

$\therefore i \longleftrightarrow k$. ■

A relação de comunicação define classes de equivalência. Uma classe C de equivalência é formada por todos os estados que se comunicam entre si, ou seja, $i, j \in C$ sse $i \longleftrightarrow j$.

Observamos que sendo possível começar num estado de uma classe C e passar para outra classe com probabilidade positiva, o retorno só será possível se ambas classes são idênticas.

DEFINIÇÃO 3.5

Uma cadeia de Markov é IRREDUTIVEL se a relação de comunicação induz uma única classe de equivalência, ou seja, cada estado está comunicado com todo outro. ■

EXEMPLO 3.2

Num modelo de caminho aleatório com conjunto de estados $S = \{0, 1, 2, \dots, a-1, a\}$ e matriz de transição

$$P = \begin{pmatrix} 1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ q & 0 & p & 0 & \dots & 0 & 0 & 0 \\ 0 & q & 0 & p & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & q & 0 & p \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 & 1 \end{pmatrix} \quad (p+q=1)$$

temos 3 classes de equivalência : $\{0\}$, $\{1, 2, \dots, a-1\}$, $\{a\}$. ■

EXEMPLO 3.3

Consideremos a cadeia de Markov, onde $S = \{0, 1, 2, \dots\}$

$$e \quad P = \begin{pmatrix} q_0 & p_0 & 0 & 0 & 0 & \dots \\ q_1 & 0 & p_1 & 0 & 0 & \dots \\ q_2 & 0 & 0 & p_2 & 0 & \dots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix} \quad \text{com } p_i > 0, q_i > 0, p_i + q_i = 1, \forall i \in S$$

Sejam i e j dois estados quaisquer. Sem perda de generalidade podemos supor $0 \leq i < j$. Então a probabilidade de passar de i para j em $j-i$ passos é

$$\begin{aligned} P_{i,j}^{j-i} &\geq P_{i,i+1} \cdot P_{i+1,i+2} \cdot \dots \cdot P_{j-1,j} \\ &= p_i \cdot p_{i+1} \cdot \dots \cdot p_{j-1} > 0 \end{aligned}$$

(a relação \geq é consequência do teorema 3.1)

Logo $i \rightarrow j$.

Agora, se $i=0$ temos $P_{j,i} = P_{j,0} = q_j > 0$

$$\text{se } i=1 \text{ temos } P_{j,i}^2 \geq P_{j,1}^2 \geq P_{j,0} \cdot P_{0,1} = q_j \cdot p_0$$

$$\text{se } i>1 \text{ temos } P_{j,i}^{i+1} \geq P_{j,0} \cdot P_{0,1} \cdot P_{1,2} \cdot \dots \cdot P_{i-1,i} = q_0 p_0 p_1 \dots p_{i-1}$$

Assim, $j \rightarrow i$ e a cadeia é irredutível. ■

O seguinte conceito a ser definido diz respeito ao fato de

que os movimentos da cadeia nos diferentes estados, podem acontecer de modo tal que a cada intervalo de tempo, a cadeia possa retornar ao estado inicial.

DEFINIÇÃO 3.6

Um estado j tem PERÍODO $d(j)$, se $d(j)$ é o maior inteiro positivo tal que

$$P_{jj}^n > 0 \Rightarrow n = m \cdot d(j) \text{ para } m > 0 \text{ e inteiro}$$

Caso $d(j)=1$ se diz que o estado j é APERIÓDICO

Uma cadeia é APERIÓDICA se todos os seus estados são aperiódicos. ■

EXEMPLO 3.4

Seja $\{X_n\}$ uma cadeia de Markov de tempo discreto com espaço de estados $S = \{1, 2, 3, 4\}$ e matriz de transição

$$P = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

A ideia é obter $d(1)$, o período do estado 1.

Verifica-se que esta matriz P goza das seguintes propriedades:

$$P^{4n-3} = P; \quad P^{4n-2} = P^2; \quad P^{4n-1} = P^3; \quad P^{4n} = P^4; \quad \forall n \geq 1.$$

Dado que $P_{11}^4 = 1 > 0$ então $d(1)$ não pode ser 3, 5, 6, 7, 8, ... pois 4 não é múltiplo deles.

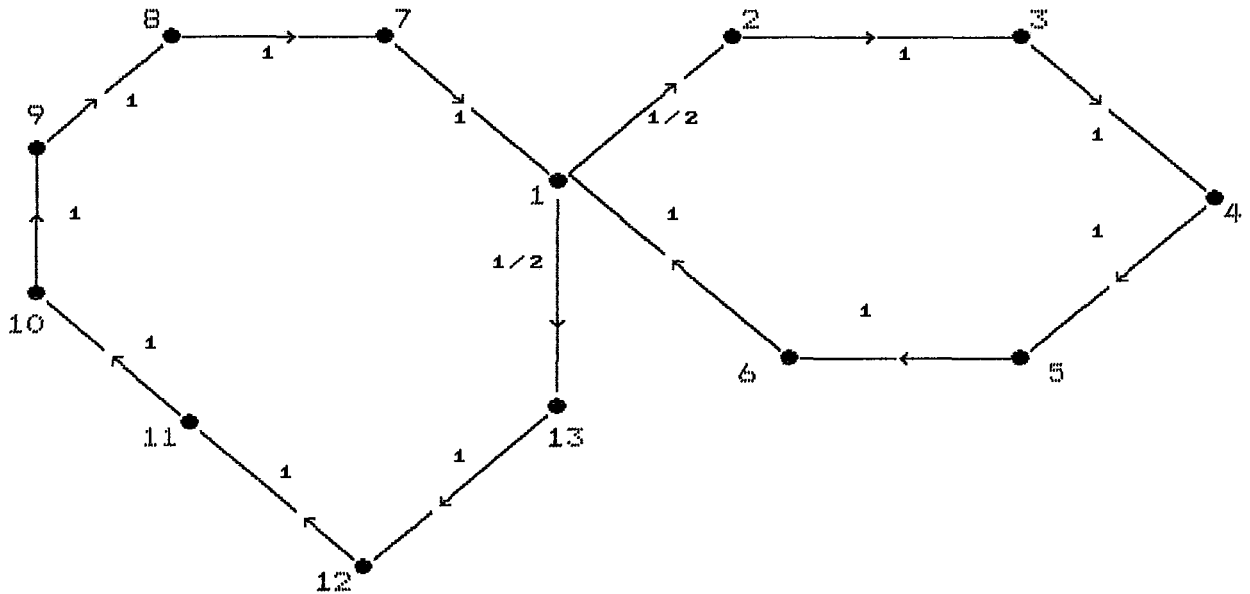
Assim, as únicas possibilidades de período para o estado 1 são 2 e 4, porém 4 é maior.

Em consequência $d(1)=4$. ■

EXEMPLO 3.5

Neste exemplo, temos uma cadeia de Markov com espaço de estados $S = \{1, 2, \dots, 13\}$ e onde as probabilidades de transição num passo estão dadas como pesos nos arcos do seguinte dígrafo.

(os arcos que não aparecem correspondem a aqueles com peso ou probabilidade de transição zero)



Para calcular as probabilidades de transição em n passos, devemos seguir todos os caminhos no dígrafo que levam de i para j em n passos.

Assim, por exemplo, para calcular P_{13}^2 devemos seguir o único caminho $1 \rightarrow 2 \rightarrow 3$ com probabilidade $1/2 \cdot 1 = 1/2$.

No exemplo temos :

$$P_{11}^1 = P_{11}^2 = P_{11}^3 = P_{11}^4 = P_{11}^5 = 0; \quad P_{11}^6 = 1/2; \quad P_{11}^7 = 0; \quad P_{11}^8 \geq 0$$

$$P_{11}^9 = P_{11}^{10} = P_{11}^{11} = 0; \quad P_{11}^{12} > 0; \quad P_{11}^{13} = 0; \quad P_{11}^{14} > 0; \dots$$

$$\text{Logo } \{ n / P_{11}^n > 0 \} = \{ 6, 8, 12, 14, 16, 18, 24, 28, \dots \}$$

Assim o período do estado 1 não pode ser 4, 5, 7, 8, 9, 10, ... pois $P_{11}^6 > 0$ e 6 não é múltiplo deles.

A propriedade também não é satisfeita por $d=6$, pois existe $n=8$ tal que $P_{11}^8 > 0$ e 8 não é múltiplo de 6. Mesma coisa acontece com $d=3$.

Porém, $d=2$ satisfaz a condição e sendo o único resulta $d(1)=2$. ■

Observemos que não necessariamente todo múltiplo k de $d(j)$ satisfaz $P_{jj}^k > 0$, como no caso dos múltiplos 4, 10, 20 do período $d(1)=2$.

Acontece que 2 é precisamente o máximo divisor comum do conjunto $\{ n / P_{11}^n > 0 \}$. Isto fica estabelecido no seguinte resultado.

TEOREMA 3.3

O estado j tem período d sse d é o máximo divisor comum (MDC) do conjunto $\{n / P_{jj}^n > 0\}$.

PROVA

Suponha que j tem período d . Logo $P_{jj}^n = 0$ exceto para n múltiplo de d . Assim, aqueles n tais que $P_{jj}^n > 0$ devem ser múltiplos de d .

Se $\text{MCD}\{n / P_{jj}^n > 0\} = k \cdot d$, então $P_{jj}^n = 0$ exceto para $n = m(kd)$

Por definição de período, d é o maior com tal propriedade

Assim $k=1$.

Reciprocamente, se $d = \text{MCD}\{n / P_{jj}^n > 0\}$ então d divide ao conjunto $\{n / P_{jj}^n > 0\}$ e $P_{jj}^n = 0$, exceto para $n = md$.

Suponhamos que $d' > d$ e que $P_{jj}^n = 0$ exceto para $n = kd'$.

Então d' divide $\{n / P_{jj}^n > 0\}$ o que contradiz o fato de que d fosse o maior divisor comum de $\{n / P_{jj}^n > 0\}$. ■

OBSERVAÇÃO

Da definição de período ou do teorema 3.3 segue que se o período do estado j é d , então um retorno ao estado j só é possível nos tempos $d, 2d, 3d, \dots$ (não necessariamente todos eles). ■

O seguinte resultado estabelece que estados numa mesma classe de equivalência (com respeito à relação de comunicação) tem o mesmo período.

TEOREMA 3.4

Se $i \longleftrightarrow j$ então $d(i) = d(j)$

PROVA

Seja s tal que $P_{ii}^s > 0$.

$$\text{Logo } P_{ii}^{2s} = \sum_{k \in S} P_{ik}^s \cdot P_{kj}^s \geq P_{ii}^s \cdot P_{ii}^s > 0$$

Como $i \longleftrightarrow j$ então existem $n, m > 0$ tais que $P_{ij}^n > 0$ e $P_{ji}^m > 0$.

$$\text{Assim, } P_{jj}^{n+s+m} \geq P_{ji}^m \cdot P_{ii}^s \cdot P_{ij}^n > 0$$

$$\text{e } P_{jj}^{n+2s+m} \geq P_{ji}^m \cdot P_{ii}^{2s} \cdot P_{ij}^n > 0$$

Logo $d(j)$ divide a $(n+s+m)$ e também a $(n+2s+m)$

$\therefore d(j)$ divide a $(n+2s+m)-(n+s+m) = s$

Como s é qualquer inteiro tal que $P_{ii}^s > 0$ então $d(j)$ divide a $\{n / P_{ii}^n > 0\}$. Sendo $d(i)$ o maior divisor comum de $\{n / P_{ii}^n > 0\}$, resulta $d(j) \leq d(i)$.

Um raciocínio análogo conduz a $d(i) \leq d(j)$ e assim $d(i) = d(j)$. ■

COROLARIO 3.2

Se numa cadeia de Markov irredutível existe um estado i tal que $P_{ii} > 0$ então todos os estados tem período 1 (ou seja que a cadeia é aperiódica).

PROVA

Se $P_{ii} > 0$ então $d(i) = 1$. Como a cadeia é irredutível $i \leftrightarrow j$, $\forall j \in S$ Por teorema anterior, $d(i) = d(j) = 1$. ■

EXEMPLO 3.6

Consideremos o seguinte modelo de caminho aleatório onde $S = \{0, 1, 2, \dots\}$, $p_0 > 0$, $r_0 > 0$, $r_0 + p_0 = 1$, para $i = 1, 2, 3, \dots$, $q_i > 0$, $r_i \geq 0$, $p_i > 0$, $q_i + r_i + p_i = 1$, e a matriz P é dada por

$$P = \begin{pmatrix} r_0 & p_0 & 0 & 0 & 0 & \dots \\ q_1 & r_1 & p_1 & 0 & 0 & \dots \\ 0 & q_2 & r_2 & p_2 & 0 & \dots \\ \cdot & \cdot & \cdot & \cdot & \dots & \dots \\ \cdot & \cdot & \cdot & \cdot & \dots & \dots \end{pmatrix}$$

Como $P_{00} = r_0 > 0$ então $d(0) = 1$. A cadeia é irredutível pois se $0 \leq i < j$,

$$P_{ij}^{j-i} \geq p_{i,i+1} \cdot p_{i+1,i+2} \cdot \dots \cdot p_{j-1,i} = p_i \cdot p_{i+1} \cdot \dots \cdot p_{j-1} > 0$$

e

$$P_{ij}^{j-i} \geq p_{j,j-1} \cdot p_{j-1,j-2} \cdot \dots \cdot p_{i+1,i} = q_j \cdot q_{j-1} \cdot \dots \cdot q_{i+1} > 0$$

De acordo com Corolario 3.2, a cadeia é aperiódica. ■

O período de um estado j diz respeito ao tempo no qual a cadeia pode retornar ao estado j . Porém, será que a cadeia alguma vez retorna ao estado inicial? Estamos interessados no tempo (possivelmente infinito) que deve passar antes de que a cadeia retorne ao estado inicial.

O seguinte conceito permite classificar estados de acordo com tal situação.

DEFINIÇÃO 3.7

Para cada estado i , e para cada $n \geq 1$, definimos f_{ii}^n como a probabilidade de que o primeiro retorno ao estado i (tendo começado no estado i) aconteça na n -ésima iteração, ou seja,

$$f_{ii}^n = P[X_n = i, X_m \neq i, \text{ para } m = 1, 2, \dots, n-1 / X_0 = i]$$

Definimos também $f_{ii}^0 = 0$.

Da mesma maneira definimos f_{ij}^n como a probabilidade de que tendo começado no estado i , a primeira visita ao estado j aconteça na n -ésima iteração, ou seja,

$$f_{ij}^n = P[X_n = j, X_m \neq j, \text{ para } m = 1, 2, \dots, n-1 / X_0 = i]$$

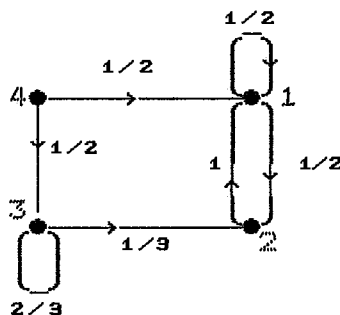
Completamos a definição com $f_{ij}^0 = 0$. ■

EXEMPLO 3.7

Seja $\{X_n\}$ uma cadeia de Markov com espaço de estados

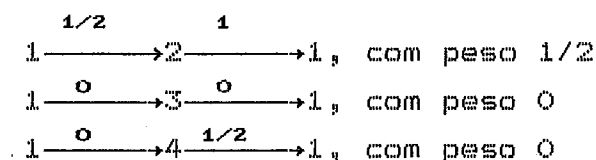
$S = \{1, 2, 3, 4\}$ e matriz de transição $P = \begin{pmatrix} 1/2 & 1/2 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1/3 & 2/3 & 0 \\ 1/2 & 0 & 1/2 & 0 \end{pmatrix}$

Representemos P em termos de um dígrafo, tal como foi explicado no exemplo 3.5.



Para encontrar f_{ii}^n , devemos considerar todos os caminhos que começando no estado i , retornem ao estado i em n passos, sem usar i como nó intermediário.

Por exemplo, para encontrar $f_{11}^2 = P[X_2=1, X_1 \neq 1, X_0=1]$ devemos considerar os caminhos:



Usando novamente o teorema das probabilidades totais (2), temos

$$f_{11}^2 = 1/2 + 0 + 0 = 1/2.$$

$$\begin{aligned}
 \text{Porém, } P_{11}^2 &= \sum_{j=1,2,3,4} P_{1j} \cdot P_{j1} = P_{11} \cdot P_{11} + P_{12} \cdot P_{21} + P_{13} \cdot P_{31} + P_{14} \cdot P_{41} \\
 &= 1/2 \cdot 1/2 + 1/2 \cdot 1 + 0 \cdot 0 + 0 \cdot 1/2 = 3/4
 \end{aligned}$$

Observamos que no cálculo de P_{11}^2 devemos acrescentar mais um caminho ($1 \rightarrow 1 \rightarrow 1$) do que o considerado no cálculo de f_{11}^2 .

Aquí, o nó 1 é usado como intermediário. Intuitivamente podemos observar que $P_{ii}^n \geq f_{ii}^n$, propriedade a ser provada depois.

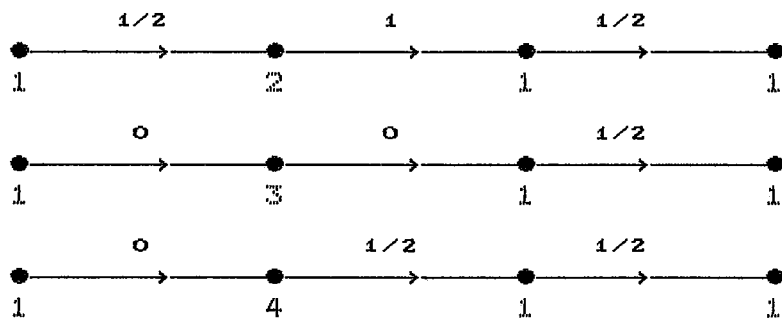
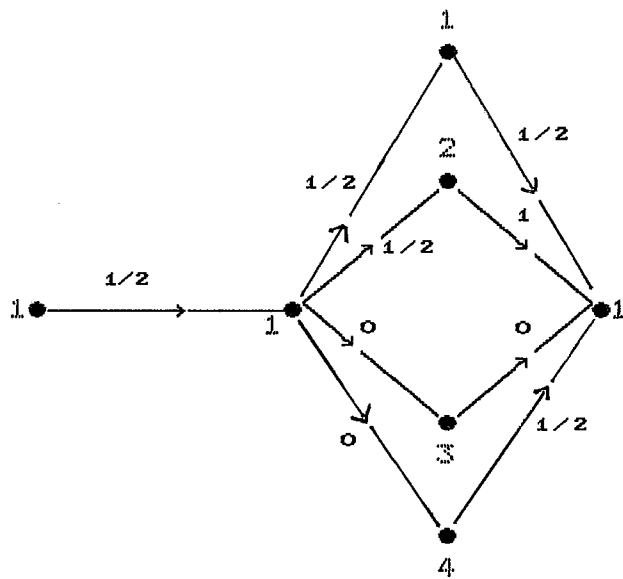
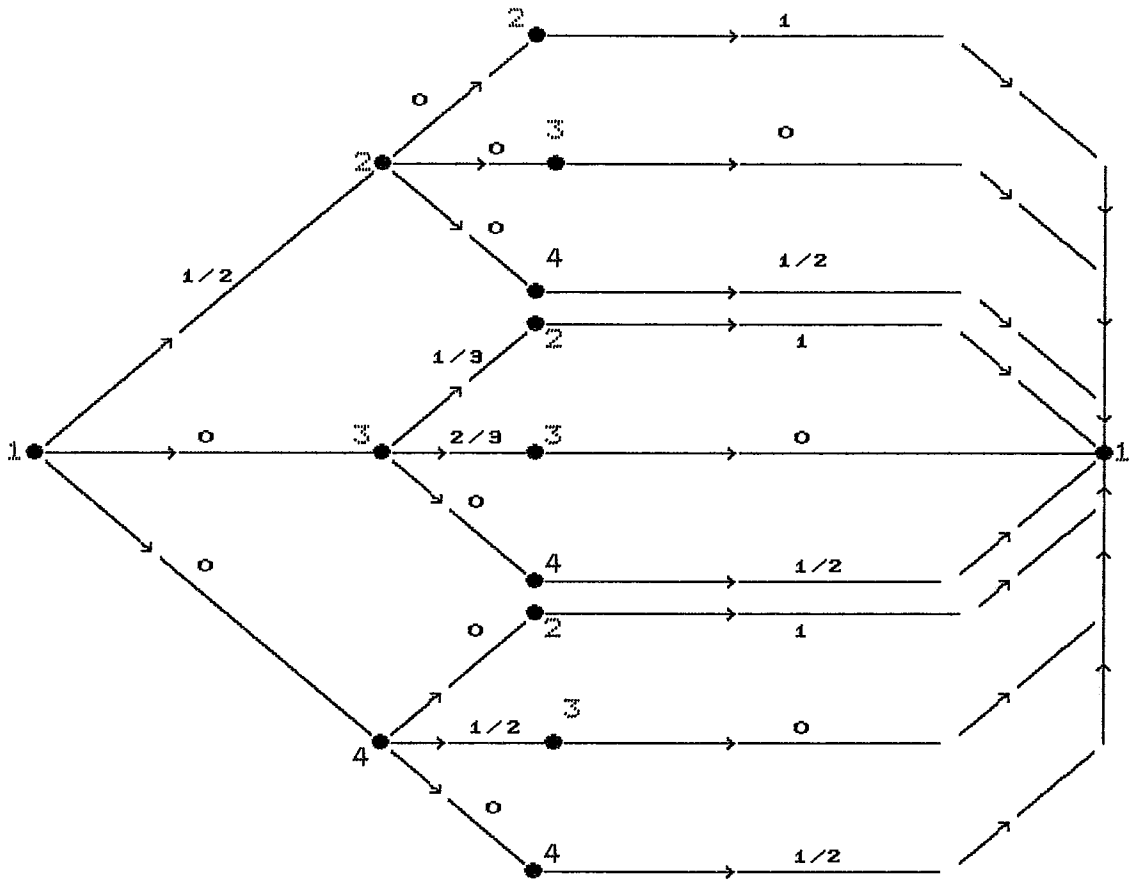
No caso $f_{11}^3 = P[X_3=1, X_2 \neq 1, X_1 \neq 1, X_0=1]$ devemos considerar todos os caminhos que começando em 1, acabem em 1, podendo usar os nós 2, 3 ou 4 como nós intermediários.

O esquema que segue pode ser utilizado para tais efeitos.

Observemos que todos esses caminhos tem probabilidade zero.

$$\text{Assim, } f_{11}^3 = 0.$$

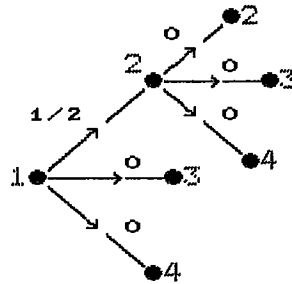
No entanto para o cálculo de P_{11}^3 consideramos 7 caminhos, indicados nos outros desenhos que seguem.



$$\text{Assim, } P_{11}^3 = \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} + \frac{1}{2} \cdot \frac{1}{2} \cdot 1 + \frac{1}{2} \cdot 1 \cdot \frac{1}{2} = \frac{5}{8}$$

Os grafos anteriores que mostram todos os caminhos possíveis do nó i até nó j , sem usar i como nó intermediário (no caso de f_{ii}^n) ou passando pelo nó i (no caso de P_{ii}^n) poderiam ser simplificados podendo todo caminho em que aparece um arco de peso zero.

Deste modo, para o caso de f_{11}^3 , o grafo dos caminhos pode ser representado por



Como não foi atingido o nó 1, teremos $f_{11}^3 = 0$.

Fica claro que $f_{11}^n = 0$; $\forall n \geq 3$ pois todo caminho com mais de 2 passos deve conter os caminhos antes cortados. O seguinte resultado relaciona P_{ii}^n com f_{ii}^k ($0 \leq k \leq n$)

TEOREMA 3.5

Para cada estado i e para cada $n \geq 1$,

$$P_{ii}^n = \sum_{k=0}^n f_{ii}^k \cdot P_{ii}^{n-k} = \sum_{k=1}^n f_{ii}^k \cdot P_{ii}^{n-k}$$

PROVA

Definamos o evento E_k como

$$\begin{aligned} E_k &= \{ X_n = i \text{ e o primeiro retorno ao estado } i \text{ acontece na} \\ &\quad \text{k-ésima iteração} / X_0 = i \} \\ &= \{ X_n = i, X_1 \neq i, X_2 \neq i, \dots, X_{k-1} \neq i, X_k = i / X_0 = i \} \end{aligned}$$

Os eventos E_k são mutuamente exclusivos e

$$\begin{aligned} P[E_k] &= P[X_n = i, X_1 \neq i, X_2 \neq i, \dots, X_{k-1} \neq i, X_k = i / X_0 = i] \\ &= \frac{P[X_n = i, X_1 \neq i, X_2 \neq i, \dots, X_{k-1} \neq i, X_k = i, X_0 = i]}{P[X_0 = i]} \end{aligned}$$

$$\begin{aligned}
&= \frac{1}{P[X_0=i]} \cdot P[X_0=i] \cdot P[X_1 \neq i, X_2 \neq i, \dots, X_{k-1} \neq i, X_k=i / X_0=i] \\
&\quad \cdot P[X_n=i / X_1 \neq i, X_2 \neq i, \dots, X_{k-1} \neq i, X_k=i, X_0=i] \\
&\hspace{15em} \text{(devido a(1))} \\
&= P[X_n=i / X_0=i, X_1 \neq i, X_2 \neq i, \dots, X_{k-1} \neq i, X_k=i] \cdot \\
&\quad \cdot P[X_k=i, X_{k-1} \neq i, \dots, X_1 \neq i / X_0=i] \\
&= P[X_n=i / X_k=i] \cdot P[X_k=i, X_{k-1} \neq i, \dots, X_1 \neq i / X_0=i] \\
&\hspace{15em} \text{(devido a(3))} \\
&= P_{ii}^{n-k} \cdot f_{ii}^k
\end{aligned}$$

Decompondo $\{X_n=i / X_0=i\}$ respeito do momento do primeiro retorno ao estado i , ou seja, $\{X_n=i / X_0=i\} = \bigcup_{k=1}^n E_k$, temos

$$\begin{aligned}
P_{ii}^n &= P[X_n=i / X_0=i] = P[\bigcup_{k=1}^n E_k] = \sum_{k=1}^n P[E_k] \\
&= \sum_{k=1}^n f_{ii}^k \cdot P_{ii}^{n-k} \\
&= \sum_{k=0}^n f_{ii}^k \cdot P_{ii}^{n-k}
\end{aligned}$$

pois $f_{ii}^0=0$ segundo foi definido. ■

Usando o teorema 3.5 podemos provar que $P_{ii}^n=0 \Rightarrow f_{ii}^n=0$ ou um resultado ainda mais geral : $P_{ii}^n \geq f_{ii}^n, \forall n \geq 0$

Efetivamente, se $n \geq 1$ temos

$$\begin{aligned}
P_{ii}^n &= \sum_{k=1}^n f_{ii}^k \cdot P_{ii}^{n-k} = f_{ii}^1 \cdot P_{ii}^{n-1} + f_{ii}^2 \cdot P_{ii}^{n-2} + \dots + f_{ii}^n \cdot 1 \\
\therefore P_{ii}^n - f_{ii}^n &= \sum_{k=1}^n f_{ii}^k \cdot P_{ii}^{n-k} \geq 0
\end{aligned}$$

No caso $n=0$ a relação é também válida pois $P_{ii}^0=1$ e $f_{ii}^0=0$.

OBSERVAÇÃO

Do teorema 3.5 podemos concluir que

$$P_{ii}^n = \sum_{k=0}^n f_{ii}^{n-k} \cdot P_{ii}^k = \begin{cases} 1 & \text{se } n=0 \\ 0 & \text{se } n>0 \end{cases}$$

TEOREMA 3.6

$$\forall i, j \in S \text{ com } i \neq j \text{ e } \forall n \geq 0, \quad P_{ij}^n = \sum_{k=0}^n f_{ij}^k \cdot P_{jj}^{n-k}$$

PROVA

Análoga ao teorema anterior.

O seguinte resultado será usado na prova do teorema 3.12

TEOREMA 3.7

$$\text{MDC}\{n / P_{ii}^n > 0\} = \text{MDC}\{n / f_{ii}^n > 0\}$$

PROVA

Sejam $d = \text{MDC}\{n / P_{ii}^n > 0\}$ e $f = \text{MDC}\{n / f_{ii}^n > 0\}$

$$\text{Da relação } P_{ii}^n = \sum_{k=1}^n f_{ii}^k \cdot P_{ii}^{n-k} = f_{ii}^n + \sum_{k=1}^{n-1} f_{ii}^k \cdot P_{ii}^{n-k}$$

pode-se deduzir que $f_{ii}^n > 0 \Rightarrow P_{ii}^n > 0$.

$$\text{Logo } \{n / f_{ii}^n > 0\} \subseteq \{n / P_{ii}^n > 0\}$$

Se $f < d$ então d não pode ser divisor de $\{n / f_{ii}^n > 0\}$ pois f é o maior divisor deste conjunto. Assim, d não pode ser divisor de $\{n / P_{ii}^n > 0\}$ o que é uma contradição.

Temos provado que $d \leq f$.

Para estabelecer a igualdade, provaremos que f divide $\{n / P_{ii}^n > 0\}$ e com isso teremos $f \leq d$, concluindo-se $f = d$.

Usaremos a notação f/n para indicar que " f divide a n ".

Devemos provar que f/n , $\forall n$ tal que $P_{ii}^n > 0$.

Sabemos que todo n tal que $P_{ii}^n > 0$ deve ser múltiplo de d , e então pode-se escrever $\{n / P_{ii}^n > 0\} = \{k_1 d, k_2 d, \dots\}$ onde $k_1 < k_2 < \dots$

a) Veremos que $f/k_1 d$.

Temos $P_{ii}^{k_1 d} > 0$. Devido a que $P_{ii}^n = \sum_{k=0}^n f_{ii}^k \cdot P_{ii}^{n-k}$, existe um

inteiro $s \leq k_1 d$ tal que $f_{ii}^s > 0$ e $P_{ii}^{k_1 d - s} > 0$.

Se $s < k_1 d$ temos $P_{ii}^s = 0$ e então

$$P_{ii}^s = f_{ii}^s + \sum_{k=0}^{s-1} f_{ii}^k \cdot P_{ii}^{s-k} = 0 \Rightarrow f_{ii}^s = 0$$

Assim $s = k_1 d \Rightarrow f_{ii}^s = f_{ii}^{k_1 d} > 0 \Rightarrow f / k_1 d$

b) Veremos que $f / k_2 d$

Temos $P_{ii}^{k_1 d} > 0 \Rightarrow \exists s \leq k_2 d$ tal que $f_{ii}^s > 0$ e $P_{ii}^{k_2 d - s} > 0$.

Se $s = k_2 d$ então $f_{ii}^s = f_{ii}^{k_2 d} > 0 \Rightarrow f / k_2 d$

Se $s < k_2 d$ então $s = k_1 d$ (senão $P_{ii}^s = 0 \Rightarrow f_{ii}^s = 0$.)

$$\text{Logo } P_{ii}^{k_2 d - s} = P_{ii}^{k_2 d - k_1 d} = P_{ii}^{(k_2 - k_1) d} > 0$$

$$\Rightarrow k_2 - k_1 = k_1 \text{ ou } k_2 - k_1 = k_2$$

$$\Rightarrow k_2 = 2k_1 \text{ ou } k_1 = 0 \quad (\text{não é possível})$$

Como $f / k_1 d$ então $f / 2k_1 d$ e logo $f / k_2 d$.

c) Vamos supor agora que $f / k_1 d, f / k_2 d, \dots, f / k_p d$ e provaremos que $f / k_{p+1} d$.

$$P_{ii}^{k_{p+1} d} > 0 \Rightarrow \exists s \leq k_{p+1} \text{ tal que } f_{ii}^s > 0 \text{ e } P_{ii}^{k_{p+1} d - s} > 0$$

$$\text{Se } s = k_{p+1} d \text{ então } f_{ii}^s = f_{ii}^{k_{p+1} d} > 0 \Rightarrow f / k_{p+1} d$$

Se $s < k_{p+1} d$ então $s \in \{k_1 d, k_2 d, \dots, k_p d\}$ devido a que

$$f_{ii}^s > 0 \Rightarrow P_{ii}^s > 0. \text{ Logo } s = k_r d \text{ para algum } r \in \{1, \dots, p\}$$

$$\text{Assim, } P_{ii}^{k_{p+1} d - s} = P_{ii}^{(k_{p+1} - k_r) d} > 0$$

$$\Rightarrow k_{p+1} - k_r = k_t \text{ para algum } t \in \{1, \dots, p\}$$

($t = p+1$ não é possível)

$$\Rightarrow k_{p+1} = k_r + k_t$$

Como $f / k_r d$ e $f / k_t d$ então $f / (k_r + k_t) d$

Logo $f / k_{p+1} d$

■

DEFINIÇÃO 3.8

Um estado i é RECORRENTE (ou PERSISTENTE) se

$$\sum_{n=1}^{\infty} f_{ii}^n = 1$$

ou seja, se começando no estado i , a probabilidade de um eventual retorno ao estado i é 1.

Se $\sum_{n=1}^{\infty} f_{ii}^n < 1$, dizemos que o estado i é TRANSITORIO, ou seja começando no estado i , temos probabilidade positiva { dada por $1 - \sum_{n=1}^{\infty} f_{ii}^n$ } de jamais retornar ao estado i .

Uma cadeia é RECORRENTE se todos seus estados são recorrentes.

EXEMPLO 3.8

Consideremos a mesma cadeia de exemplo 3.7

Temos $f_{11}^1 = P_{11} = 1/2$; $f_{11}^2 = 1/2$; $f_{11}^n = 0$ para $n \geq 3$.

$\therefore \sum_{n=1}^{\infty} f_{11}^n = 1$ e o estado 1 é recorrente.

Também se verifica $f_{22}^1 = P_{22} = 0$; $f_{22}^2 = 1/2$;

$$f_{22}^3 = 1/2^2 ; \dots ; f_{22}^n = 1/2^{n-1}.$$

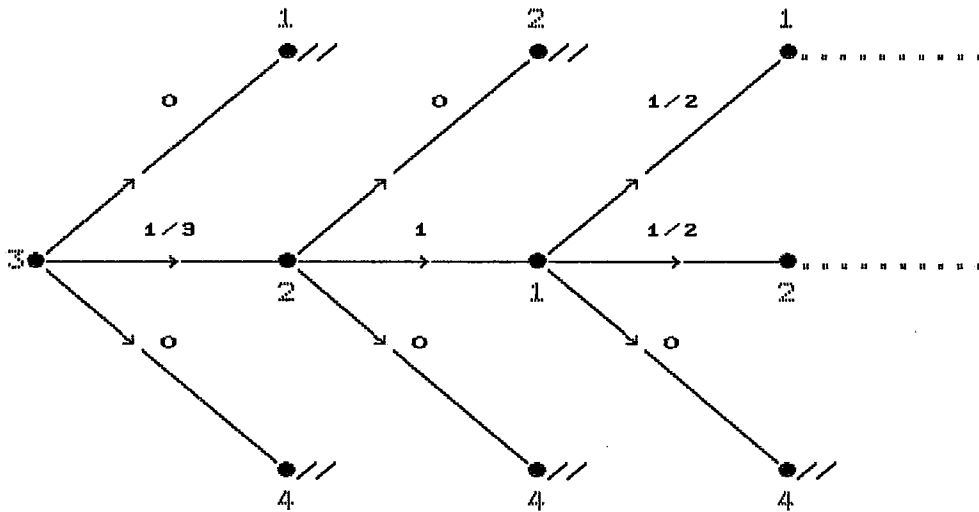
$\therefore \sum_{n=1}^{\infty} f_{22}^n = 1/2 + 1/4 + 1/8 + \dots = 1$

sendo assim 2 um estado recorrente.

Temos $f_{33}^1 = P_{33} = 2/3$; $f_{33}^2 = f_{33}^3 = f_{33}^4 = \dots = 0$

Logo $\sum_{n=1}^{\infty} f_{33}^n = 2/3 < 1$, sendo 3 um estado transitório.

O grafo associado aos cálculos f_{33}^n é



Para $n > 3$, o grafo só poderá considerar os nos 1 e 2, mostrando assim que temos caminhos que nunca retornam ao estado 3, ou seja que existe probabilidade positiva de nunca retornar a esse estado. ■

OBSERVAÇÃO

Na definição 3.8, a probabilidade de um eventual retorno ao estado i é igual a $P[E_1 \cup E_2 \cup E_3 \cup \dots \cup E_k \cup \dots]$, onde E_k é o evento definido por :

E_k = {primeiro retorno ao estado i aconteça na k -ésima iteração}

Os eventos E_k são mutuamente exclusivos e então

$$P[E_1 \cup E_2 \cup \dots \cup E_k \cup \dots] = \sum_{k=1}^{\infty} P[E_k] = \sum_{k=1}^{\infty} f_{ii}^n$$

A seguir vamos provar que a recorrência é (como a periodicidade) uma propriedade da classe de equivalência definida pela relação de comunicação, ou seja, se $i \longleftrightarrow j$ e i é um estado recorrente então j é também recorrente. Assim, os estados comunicados são todos recorrentes ou transitórios.

Para provar esse resultado é conveniente uma boa caracterização da recorrência em termos de P_{ii}^n . O seguinte resultado aponta nesse sentido.

Consideremos
$$P_{ii}(s) = \sum_{n=0}^{\infty} P_{ii}^n \cdot s^n \quad (|s| < 1)$$

$$e \quad F_{ii}(s) = \sum_{n=1}^{\infty} f_{ii}^n \cdot s^n \quad (|s| < 1)$$

(s restrito a valores em módulo menores que 1 é introduzido somente para assegurar a convergência das series).

TEOREMA 3.8

$$P_{ii}(s) = 1 + F_{ii}(s) \cdot P_{ii}(s) \quad \text{para } |s| < 1$$

ou equivalentemente

$$P_{ii}(s) = \frac{1}{1 - F_{ii}(s)} \quad \text{para } |s| < 1$$

PROVA

$$\begin{aligned} P_{ii}(s) \cdot F_{ii}(s) &= \left(\sum_{n=0}^{\infty} P_{ii}^n \cdot s^n \right) \left(\sum_{n=1}^{\infty} f_{ii}^n \cdot s^n \right) \\ &= (1 + P_{ii} \cdot s + P_{ii}^2 \cdot s^2 + \dots) (f_{ii} \cdot s + f_{ii}^2 \cdot s^2 + \dots) \\ &= f_{ii} \cdot s + (f_{ii}^2 + P_{ii} \cdot f_{ii}) s^2 + \dots \\ &= \sum_{k=1}^{\infty} \left(\sum_{l=1}^{\infty} f_{ii}^l \cdot P_{ii}^{k-l} \right) s^k \\ &= \sum_{k=1}^{\infty} P_{ii}^k \cdot s^k = \sum_{k=0}^{\infty} P_{ii}^k \cdot s^k - 1 = P_{ii}(s) - 1 \quad \blacksquare \end{aligned}$$

OBSERVAÇÃO

O teorema 3.8, junto ao teorema de Abel (no apêndice) serão usados na prova da seguinte caracterização de recorrência.

TEOREMA 3.9

Um estado i é recorrente sse $\sum_{n=1}^{\infty} P_{ii}^n = \infty$

PROVA

Suponhamos que o estado i é recorrente, ou seja, $\sum_{k=1}^{\infty} f_{ii}^k = 1$
 Fazendo $a_n = f_{ii}^n$ no teorema de Abel parte (a) resulta:

$$\lim_{s \rightarrow 1^-} \sum_{k=0}^{\infty} f_{ii}^{n-k} \cdot s^k = \lim_{s \rightarrow 1^-} \sum_{k=1}^{\infty} f_{ii}^{n-k} \cdot s^k = \lim_{s \rightarrow 1^-} F_{ii}(s) = 1$$

Logo, pelo teorema 3.8, $\lim_{s \rightarrow 1^-} P_{ii}(s) = \lim_{s \rightarrow 1^-} \frac{1}{1-F_{ii}(s)} = \infty$

Então $\lim_{s \rightarrow 1^-} \sum_{n=0}^{\infty} P_{ii}^{n+1} \cdot s^n = \infty$, pois $P_{ii}(s) = \sum_{n=0}^{\infty} P_{ii}^{n+1} \cdot s^n$

Usando a parte (b) do teorema de Abel, conclui-se que

$$\sum_{n=0}^{\infty} P_{ii}^{n+1} = \infty.$$

Reciprocamente, suponhamos que i não é recorrente, ou seja

$$\sum_{n=1}^{\infty} f_{ii}^{n-1} = a < 1. \text{ Como no caso anterior, temos } \lim_{s \rightarrow 1^-} F_{ii}(s) = a$$

Assim, $\lim_{s \rightarrow 1^-} P_{ii}(s) = \lim_{s \rightarrow 1^-} \frac{1}{1-F_{ii}(s)} = \frac{1}{1-a} < \infty$

$\therefore \lim_{s \rightarrow 1^-} \sum_{n=0}^{\infty} P_{ii}^{n+1} \cdot s^n = \frac{1}{1-a} < \infty$

De novo, usando (b) no teorema de Abel, temos $\sum_{n=0}^{\infty} P_{ii}^{n+1} = \frac{1}{1-a} < \infty$ ■

OBSERVAÇÃO

Um estado j é transitório sse $\sum_{n=0}^{\infty} P_{jj}^{n+1} < \infty$

EXEMPLO 3.9

Consideremos o modelo de caminho aleatório onde S é o conjunto dos números inteiros e a matriz de transição é:

$$P = \begin{pmatrix} \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \end{pmatrix} \begin{matrix} i-1 \\ i \\ i+1 \\ i-1 \\ i \\ i+1 \end{matrix} \quad (p+q=1)$$

Aquí temos $F_{ii}^{2n+1} = 0$ para $n=1,2,\dots$ pois começando no estado i , não conseguimos voltar ao estado i numa quantidade ímpar de passos.

$$\text{Também, } P_{ii}^{2n} = \binom{2n}{n} \cdot p^n \cdot q^n$$

Para obter a relação acima, seja S_{2n} a posição da cadeia depois de $2n$ passos e $S_0 = i$.

Logo $S_{2n} = i$ sse a cadeia realiza n passos a direita e n passos a esquerda.

Isto pode ser feito de $\binom{2n}{n}$ formas diferentes.

A probabilidade de realizar n passos a direita e n passos a esquerda é $p^n \cdot q^n$, devido à independência.

$$\text{Então } P_{ii}^{2n} = P[S_{2n} = i] = \binom{2n}{n} p^n q^n = \frac{(2n)!}{n!n!} p^n q^n$$

Usando a fórmula de Stirling, $n! \approx n^{n+(1/2)} \cdot e^{-n} \cdot \sqrt{2\pi}$ obtemos

$$P_{ii}^{2n} \approx \frac{(pq)^n \frac{(2n)^{2n+(1/2)} e^{-2n}}{\sqrt{2\pi}}}{n^{2n+1} e^{-2n} \sqrt{2\pi}}$$

Agora

$$pq = p(1-p) \leq 1/4 \quad \text{e} \quad p(1-p) = 1/4 \quad \text{sse} \quad p = q = 1/2$$

(i) Se $p = 1/2$,

$$\begin{aligned} \text{então } P_{ii}^{2n} &\approx \frac{1}{\sqrt{\pi n}} \Rightarrow \sum_{n=1}^{\infty} P_{ii}^{2n} = \sum_{n=1}^{\infty} \frac{1}{\sqrt{\pi n}} = \infty \\ &\Rightarrow \sum_{n=0}^{\infty} P_{ii}^{2n} = \sum_{n=1}^{\infty} P_{ii}^{2n} = \infty \end{aligned}$$

(ii) Se $p \neq 1/2$,

$$\begin{aligned} \text{então } pq < 1/4 &\Rightarrow P_{ii}^{2n} \approx \frac{a}{\sqrt{\pi n}} \quad (\text{onde } a = 4pq < 1) \\ &\Rightarrow \sum_{n=1}^{\infty} P_{ii}^{2n} \approx \sum_{n=1}^{\infty} \frac{a}{\sqrt{\pi n}} \end{aligned}$$

A série da direita é convergente pois se $x_n = \frac{a^n}{\sqrt{n}}$ então

$$\lim_{n \rightarrow \infty} \frac{x_{n+1}}{x_n} = a < 1. \quad \text{Logo } \sum_{n=1}^{\infty} P_{ii}^{2n} \text{ e } \sum_{n=0}^{\infty} P_{ii}^{2n} \text{ são convergentes.}$$

Em consequência, $\sum_{n=0}^{\infty} P_{ii}^n = \infty$ sse $p = 1/2$, ou seja que o estado i é recorrente sse $p = 1/2$. Sendo i um estado qualquer, provamos que a cadeia é recorrente sse $p = 1/2$.

TEOREMA 3.10

Se $i \longleftrightarrow j$ e i é recorrente então j é recorrente

PROVA

$i \longleftrightarrow j \rightarrow \exists n, m \geq 1$ tais que $P_{ij}^n > 0$ e $P_{ji}^m > 0$

Logo $P_{jj}^{m+r+n} \geq P_{ji}^m \cdot P_{ii}^r \cdot P_{ij}^n$ para $r=0,1,2,\dots$

como consequência do teorema 3.1.

Assim, $\sum_{r=0}^{\infty} P_{jj}^{m+r+n} \geq \sum_{r=0}^{\infty} P_{ji}^m \cdot P_{ii}^r \cdot P_{ij}^n = P_{ji}^m \cdot P_{ij}^n \cdot \sum_{r=0}^{\infty} P_{ii}^r$

Se i é recorrente, usando teorema 3.9 temos que a série

$\sum_{r=0}^{\infty} P_{ii}^r$ diverge e então $\sum_{r=0}^{\infty} P_{jj}^{m+r+n}$ e $\sum_{r=0}^{\infty} P_{jj}^r$ divergem,

pois $\sum_{r=0}^{\infty} P_{jj}^r \geq \sum_{r=0}^{\infty} P_{jj}^{m+r+n}$

Assim, novamente pelo teorema 3.9, resulta j recorrente. ■

OBSERVAÇÃO

O teorema anterior prova que a recorrência é uma propriedade da classe de equivalência definida pela relação de comunicação.

COROLÁRIO 3.3.

Se uma cadeia de Markov é irredutível e existe um estado recorrente i então a cadeia é recorrente.

PROVA

Como a cadeia é irredutível, temos $i \longleftrightarrow j, \forall j \in S$.

Pelo teorema anterior, j será também recorrente.

EXEMPLO 3.10 Consideremos uma cadeia de Markov com $S=\{1,2,3,4\}$ e

$$\text{matriz de transi\c{c}\~{a}o } P = \begin{pmatrix} 1/4 & 1/4 & 1/4 & 1/4 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

É simples verificar que a cadeia é irredutível.

O estado 1 é recorrente pois $f_{11}^1 = P_{11} = 1/4$; $f_{11}^2 = f_{11}^3 = f_{11}^4 = 1/4$.
Então, usando corolário 3.3, a cadeia é recorrente.

COROLARIO 3.4

Se uma cadeia de Markov é irredutível e o espaço de estados S é finito então a cadeia é recorrente.

PROVA

Como S é finito, depois de pelo menos $|S|$ passos, algum estado j será visitado duas vezes. Então j é recorrente e como a cadeia é irredutível, temos pelo corolário 3.3 que a cadeia é recorrente.

Aplicando este resultado podemos ver de maneira mais simples que a cadeia do exemplo 3.10 é recorrente.

No caso de um estado recorrente j , os números f_{jj}^n formam uma distribuição de probabilidades no tempo do primeiro retorno. Isto é justificado pelo fato de que $f_{jj}^n \geq 0$, $\forall n$ e $\sum_{n=1}^{\infty} f_{jj}^n = 1$.

Assim, faz sentido considerar o tempo esperado até o primeiro retorno a um estado.

DEFINIÇÃO 3.9

O TEMPO MEDIO DE RECORRENCIA do estado i é definido

$$\text{por } \mu_i = \begin{cases} \sum_{n=1}^{\infty} n \cdot f_{ii}^n & \text{se } i \text{ é recorrente} \\ \infty & \text{se } i \text{ é transitório} \end{cases}$$

EXEMPLO 3.11

Consideremos a cadeia de Markov do exemplo 3.7. No exemplo 3.8 foram calculados os f_{ii}^n .

No exemplo 3.8 foram calculados os f_{ii}^n .

Temos então:

$$\mu_1 = \sum_{n=1}^{\infty} n f_{11}^n = 1 \cdot \frac{1}{2} + 2 \cdot \frac{1}{2} = \frac{3}{2}$$

$$\mu_2 = \sum_{n=1}^{\infty} n f_{22}^n = 1 \cdot 0 + 2 \cdot \frac{1}{2} + 3 \cdot \frac{1}{2^2} + 4 \cdot \frac{1}{2^3} + \dots = 3$$

Observamos que $\mu_3 = \infty$ segundo a definição 3.9, dado que 3 é um estado transitório.

$$\text{Porém, } \sum_{n=1}^{\infty} n f_{33}^n = 1 \cdot f_{33}^1 = 2/3. \quad \blacksquare$$

A ideia de definir $\mu_i = \infty$ no caso que i seja transitório faz sentido pois, nesse caso, temos probabilidade positiva de nunca retornar ao estado i .

Seja agora $P = (P_{ij})$ a matriz de transição de uma cadeia de Markov com espaço de estados S . Para cada i, j fixos,

consideremos a sequência $\left\{ P_{ij}^n \right\}_{n=1}^{\infty}$.

A primeira pergunta associada a essa sequência é se existirá $\lim_{n \rightarrow \infty} P_{ij}^n$. Caso exista, podemos perguntar se o limite depende ou não do estado inicial i .

Será que a influência do estado inicial i vai diminuindo no tempo e conseqüentemente P_{ij}^n se aproxime de um limite Π_j (quando $n \rightarrow \infty$) independentemente de i ?

Será que nesse caso, o vetor $\Pi = (\Pi_1, \Pi_2, \dots, \Pi_S)$ é estocástico, ou seja, são satisfeitas as condições:

$$\Pi_j \geq 0, \forall j \in S \quad \text{e} \quad \sum_{j \in S} \Pi_j = 1?$$

DEFINIÇÃO 3.9a

Uma cadeia é dita ERGODICA se $\lim_{n \rightarrow \infty} P_{ij}^n = \Pi_j$ existe para cada $j \in S$, independentemente do estado inicial i , e além do

mais, $\sum_{j \in S} \Pi_j = 1$.

A condição $\Pi_j \geq 0$ é conseqüência do fato que $P_{ij}^n \geq 0, \forall n$.

Duas condições necessárias para garantir a ergodicidade de uma cadeia de Markov são a aperiodicidade e a recorrência da cadeia.

De fato, a periodicidade da cadeia pode fazer com que $\lim_{n \rightarrow \infty} P_{jj}^n$ não

exista (ver exemplo 3.13).

Agora, se a cadeia não for recorrente (tem pelo menos um estado transitório) podemos analisar os seguintes dois casos:

(i) a cadeia é irredutível.

Neste caso, todos os estados serão transitórios, o que se traduz em

$$\forall i \in S, \sum_{n=1}^{\infty} F_{ii}^n < \infty \Rightarrow \lim_{n \rightarrow \infty} P_{ii}^n = \Pi_i = 0, \forall i \in S.$$

Logo, não é possível satisfazer $\sum_{i \in S} \Pi_i = 1$ e a cadeia não é ergódica.

(ii) a cadeia não é irredutível.

Neste caso podemos ter $\lim_{n \rightarrow \infty} P_{ij}^n$ dependente do estado inicial i , como se mostra no seguinte exemplo.

EXEMPLO 3.10a

$$\text{Seja } P = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1/3 & 0 & 2/3 & 0 \\ 1/4 & 1/4 & 0 & 1/2 \end{pmatrix}$$

$$\text{Pode-se provar que } P^n = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 - (\frac{2}{3})^n & 0 & (\frac{2}{3})^n & 0 \\ \frac{1}{2} - (\frac{1}{2})^{n+1} & \frac{1}{2} - (\frac{1}{2})^{n+1} & 0 & (\frac{1}{2})^n \end{pmatrix}$$

Os estados 1 e 2 são recorrentes pois $\sum_{n=1}^{\infty} P_{11}^n = \sum_{n=1}^{\infty} P_{22}^n = \sum_{n=1}^{\infty} 1 = \infty$

e os estados 3 e 4 são transitórios pois $\sum_{n=1}^{\infty} P_{33}^n = \sum_{n=1}^{\infty} (\frac{2}{3})^n = 2$

$$\text{e } \sum_{n=1}^{\infty} P_{44}^n = \sum_{n=1}^{\infty} (\frac{1}{2})^n = 1.$$

Logo a cadeia não é recorrente.

Também a cadeia não é irredutível pois o estado 1 (e também o estado 2) é "absorvente", ou seja que $P_{1j} = 0, \forall j \neq 1$ o que equivale a dizer que nunca podemos sair desse estado uma

vez atingido ele.

Agora, $\lim_{n \rightarrow \infty} P_{11}^n = 1$; $\lim_{n \rightarrow \infty} P_{21}^n = 0$; $\lim_{n \rightarrow \infty} P_{31}^n = 1$; $\lim_{n \rightarrow \infty} P_{41}^n = 1/2$
 ou seja que $\lim_{n \rightarrow \infty} P_{i1}^n$ existe para cada i , mas depende do estado inicial i .

Assim, conclui-se que a cadeia não pode ser ergódica. ■

Adicionando irreduzibilidade às condições de aperiodicidade e de recorrência da cadeia, podemos formular e provar o seguinte teorema ergódico.

TEOREMA 3.11

Seja $P = (P_{ij})$ a matriz de transição de uma cadeia de Markov irreduzível, aperiódica e recorrente. Então

a) $\lim_{n \rightarrow \infty} P_{ii}^n$ existe. Seja $\Pi_i = \lim_{n \rightarrow \infty} P_{ii}^n$

$$b) \Pi_i = \lim_{n \rightarrow \infty} P_{ii}^n = \begin{cases} \frac{1}{\sum_{n=1}^{\infty} n \cdot f_{ii}^n} = \frac{1}{\mu_i} & \text{se } \mu_i < \infty \\ 0 & \text{se } \mu_i = \infty \end{cases}$$

c) $\forall j \in S, \lim_{n \rightarrow \infty} P_{ji}^n = \lim_{n \rightarrow \infty} P_{ii}^n = \Pi_i$

ou seja, para n suficientemente grande, a probabilidade de alcançar um estado i na n -ésima iteração, tendo a cadeia começado no estado j , depende unicamente do próprio estado i e não do estado inicial j .

PROVA

Identifiquemos no teorema 2 do Apêndice,

$$a_n = f_{ii}^n; \quad u_n = P_{ii}^n; \quad b_n = \begin{cases} 1 & \text{se } n=0 \\ 0 & \text{se } n>0 \end{cases}$$

Verificação das hipóteses:

$$a_n = f_{ii}^n \geq 0, \quad \forall n$$

$$u_n = P_{ii}^n \geq 0, \quad \forall n$$

$$b_n \geq 0, \quad \forall n$$

$$\sum_{n=0}^{\infty} a_n = \sum_{n=0}^{\infty} f_{ii}^n = 1 \quad (\text{devido à recorrência do estado } i)$$

$$\sum_{k=0}^{\infty} b_k = b_0 = 1 < \infty$$

$$\{u_n\} = \{P_{ii}^n\} \text{ é limitada, pois } 0 \leq P_{ii}^n \leq 1, \forall n$$

$$\begin{aligned} \text{MDC}\langle n / a_n > 0 \rangle &= \text{MDC}\langle n / f_{ii}^n > 0 \rangle \\ &= \text{MDC}\langle n / P_{ii}^n > 0 \rangle \quad (\text{pelo teorema 3.7}) \\ &= 1 \quad (\text{pela aperiodicidade de } i) \end{aligned}$$

A sequência $\{u_n\} = \{P_{ii}^n\}$ deve satisfazer a equação:

$$u_n - \sum_{k=0}^{n-1} a_{n-k} \cdot u_k = b_n$$

Efetivamente temos, na observação ao teorema 3.5

$$P_{ii}^n - \sum_{k=0}^{n-1} f_{ii}^{n-k} \cdot P_{ii}^k = \begin{cases} 1 & \text{se } n=0 \\ 0 & \text{se } n>0 \end{cases}$$

Então $\lim_{n \rightarrow \infty} u_n = \lim_{n \rightarrow \infty} P_{ii}^n$ existe,

$$\lim_{n \rightarrow \infty} P_{ii}^n = \frac{\sum_{n=0}^{\infty} b_n}{\sum_{k=0}^{\infty} k \cdot a_k} = \frac{1}{\sum_{n=0}^{\infty} n \cdot f_{ii}^n} = \frac{1}{\mu_i} \quad \text{se } \mu_i < \infty$$

$$\text{e } \lim_{n \rightarrow \infty} P_{ii}^n = 0 \quad \text{se } \mu_i = \infty$$

Para provar a parte (c) do teorema, consideremos a seguinte relação dada no teorema 3.6:

$$P_{ji}^n = \sum_{k=0}^{n-1} f_{ji}^k \cdot P_{ii}^{n-k} = \sum_{k=0}^{n-1} f_{ji}^{n-k} \cdot P_{ii}^k; \quad n \geq 0; \quad i \neq j$$

Sejam $y_n = P_{ji}^n$; $a_n = f_{ji}^n$; $x_n = P_{ii}^n$; $\forall n \geq 0$.

Com isto, a relação anterior fica: $y_n = \sum_{k=0}^{n-1} a_{n-k} \cdot x_k$

Provaremos a seguir que se $a_n \geq 0$, $\sum_{n=0}^{\infty} a_n = 1$ e $\lim_{n \rightarrow \infty} x_k = c$ ($c \in \mathbb{R}$)
então $\lim_{k \rightarrow \infty} y_k = \lim_{k \rightarrow \infty} x_k = c$.

Verificação destas três hipóteses:

(i) claramente $a_n = f_{ji}^n \geq 0$

(ii) $\lim_{k \rightarrow \infty} x_k = \lim_{n \rightarrow \infty} P_{ii}^n$ existe (parte (a) deste teorema) e

$$\lim_{k \rightarrow \infty} x_k = \lim_{n \rightarrow \infty} P_{ii}^n = \begin{cases} 1/\mu_i & \text{se } \mu_i < \infty \\ 0 & \text{se } \mu_i = \infty \end{cases}$$

(parte (b) deste teorema)

Assim, o valor de c a ser usado no que segue desta prova

$$\text{é } c = \begin{cases} 1/\mu_i & \text{se } \mu_i < \infty \\ 0 & \text{se } \mu_i = \infty \end{cases}$$

(iii) $\sum_{n=0}^{\infty} a_n = \sum_{n=0}^{\infty} f_{ji}^n = 1$

A última igualdade tem a seguinte justificativa:

devido à recorrência do estado i , temos $\sum_{n=0}^{\infty} f_{ii}^n = 1$, ou seja que temos probabilidade 1 de que começando no estado i , alguma vez (ou em alguma quantidade finita de passos) retornemos ao estado i . Agora, se j é outro estado qualquer e devido à irredutibilidade da cadeia, existe $s \geq 1$ tal que $P_{ij}^s > 0$, ou seja que temos algum caminho de i para j . Qualquer que seja a forma em que este caminho continua, ele pode ser pensado como saindo de i e então deverá retornar alguma vez ao estado i .

Assim, começando em j , com certeza todo caminho deve alcançar alguma vez o estado i .

Logo, a probabilidade de que começando em j alguma vez visitemos o estado i é 1.

Agora provaremos que efetivamente $\lim_{n \rightarrow \infty} y_n = \lim_{k \rightarrow \infty} x_k = c$

$$\lim_{k \rightarrow \infty} x_k = c \Leftrightarrow \forall \varepsilon > 0, \exists K \text{ tal que para } k \geq K, |x_k - c| < \varepsilon/3.$$

Seja $n > K$. Então

$$\begin{aligned}
y_n - c &= \sum_{k=0}^n a_{n-k} \cdot x_k - c \left(\sum_{m=0}^{\infty} a_m \right) \\
&= \sum_{k=0}^n a_{n-k} \cdot x_k - c \sum_{m=0}^n a_m - c \sum_{m=n+1}^{\infty} a_m \\
&= \sum_{k=0}^n a_{n-k} \cdot x_k - c \sum_{k=0}^n a_{n-k} - c \sum_{m=n+1}^{\infty} a_m \\
&= \sum_{k=0}^n a_{n-k} \cdot (x_k - c) - c \sum_{m=n+1}^{\infty} a_m \\
&= \sum_{k=0}^K a_{n-k} \cdot (x_k - c) + \sum_{k=K+1}^n a_{n-k} \cdot (x_k - c) - c \sum_{m=n+1}^{\infty} a_m
\end{aligned}$$

Se $M = \max \left\{ |x_k - c| \mid 0 \leq k \leq K \right\}$ então

$$|y_n - c| \leq M \sum_{k=0}^K a_{n-k} + \frac{\varepsilon}{3} \sum_{k=K+1}^n a_{n-k} + |c| \sum_{m=n+1}^{\infty} a_m$$

(No caso $c=0$, não será necessário limitar superiormente $\sum_{m=n+1}^{\infty} a_m$)

$$\sum_{m=0}^{\infty} a_m = 1 \quad \rightarrow \quad \lim_{r \rightarrow \infty} \sum_{m=0}^r a_m = 1$$

$$\Rightarrow \forall \varepsilon > 0, \exists N_1 \text{ tal que para } n \geq N_1, \quad \left| \sum_{m=0}^n a_m - 1 \right| < \frac{\varepsilon}{3|c|}$$

$$\Rightarrow \forall \varepsilon > 0, \exists N_1 \text{ tal que para } n \geq N_1, \quad \sum_{m=n+1}^{\infty} a_m < \frac{\varepsilon}{3|c|}$$

$$\text{Também, } \forall \varepsilon > 0, \exists N_2 \text{ tal que para } n \geq N_2, \quad \left| \sum_{m=0}^{n-K-1} a_m - 1 \right| < \frac{\varepsilon}{3M}$$

$$\rightarrow \forall \varepsilon > 0, \exists N_2 \text{ tal que para } n \geq N_2, \quad \sum_{m=n-K}^{\infty} a_m < \frac{\varepsilon}{3M}$$

$$\text{e como } \sum_{k=0}^K a_{n-k} = \sum_{m=n-K}^n a_m \leq \sum_{m=n-K}^{\infty} a_m, \text{ obtemos que}$$

$$\forall \varepsilon > 0, \exists N_2 \text{ tal que para } n \geq N_2, \sum_{k=0}^K a_{n-k} < \frac{\varepsilon}{3M}$$

Escolhendo $N = \max\{N_1, N_2, K+1\}$ temos:

$$\begin{aligned} |y_n - c| &\leq \frac{\varepsilon}{3} + \frac{\varepsilon}{3} \sum_{k=K+1}^n a_{n-k} + \frac{\varepsilon}{3} \leq \frac{\varepsilon}{3} + \frac{\varepsilon}{3} \sum_{m=0}^{n-K-1} a_m + \frac{\varepsilon}{3} \\ &\leq \frac{\varepsilon}{3} + \frac{\varepsilon}{3} \sum_{m=0}^{\infty} a_m + \frac{\varepsilon}{3} = \frac{\varepsilon}{3} + \frac{\varepsilon}{3} + \frac{\varepsilon}{3} = \varepsilon \end{aligned}$$

$$\text{Logo } \lim_{n \rightarrow \infty} y_n = \lim_{n \rightarrow \infty} x_n = c$$

$$\text{Então } \lim_{n \rightarrow \infty} P_{ji}^n = \lim_{n \rightarrow \infty} P_{ii}^n = c = \begin{cases} 1/\mu_i & \text{se } \mu_i < \infty \\ 0 & \text{se } \mu_i = \infty \end{cases}$$

EXEMPLO 3.12

Seja $\{X_n\}$ uma cadeia de Markov com $S = \{1, 2\}$ e matriz de

$$\text{transição } P = \begin{bmatrix} 1/2 & 1/2 \\ 1/4 & 3/4 \end{bmatrix}$$

A cadeia é aperiódica $\{P_{11}^n > 0, P_{22}^n > 0 \text{ e } \text{MDC}\{n/P_{11}^n\} = \text{MDC}\{n/P_{22}^n\} = 1\}$ é irreduzível $\{P_{12}^n > 0 \text{ e } P_{21}^n > 0\}$ e também recorrente (corolário

3.4). Logo deve existir $\lim_{n \rightarrow \infty} P_{ii}^n$ para $i=1, 2$.

$$\text{Pode-se verificar que } P^n = \begin{bmatrix} \frac{1}{3} + \frac{2}{3 \cdot 2^{2n}} & \frac{2}{3} - \frac{2}{3 \cdot 2^{2n}} \\ \frac{1}{3} - \frac{1}{3 \cdot 2^{2n}} & \frac{2}{3} + \frac{1}{3 \cdot 2^{2n}} \end{bmatrix} \quad \forall n \geq 1.$$

$$\text{Então } \lim_{n \rightarrow \infty} P^n = \begin{bmatrix} 1/3 & 2/3 \\ 1/3 & 2/3 \end{bmatrix}$$

$$\text{Assim, } \lim_{n \rightarrow \infty} P_{11}^n = \lim_{n \rightarrow \infty} P_{21}^n = \pi_1 = 1/3$$

$$\text{e } \lim_{n \rightarrow \infty} P_{12}^n = \lim_{n \rightarrow \infty} P_{22}^n = \pi_2 = 2/3.$$

Pode-se observar que os limites anteriores independem do estado inicial. Equivalentemente podemos escrever :

$$P(X_n=1) \xrightarrow{n \rightarrow \infty} \pi_1 = 1/3 \quad \text{e} \quad P(X_n=2) \xrightarrow{n \rightarrow \infty} \pi_2 = 2/3 \quad \blacksquare$$

Como já foi mencionado, a aperiodicidade e a recorrência são condições necessárias para que uma cadeia de Markov seja ergódica.

O seguinte exemplo mostra o que acontece se não é satisfeita a condição de aperiodicidade da cadeia.

EXEMPLO 3.13

Seja $\{X_n\}$ uma cadeia de Markov com $S=\{1,2\}$ e $P = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$

As potências de P verificam:

$$P^{2n} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}; \forall n \geq 1 \quad \text{e} \quad P^{2n+1} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}; \forall n \geq 0$$

A cadeia é irredutível $\{P_{11}^2 > 0, P_{12}^2 > 0, P_{21}^2 > 0, P_{22}^2 > 0\}$ e

também recorrente $\left[\sum_{n=0}^{\infty} P_{11}^n = \sum_{n=0}^{\infty} P_{22}^n = \sum_{n \text{ par}} 1 = \infty \right]$.

A cadeia é periódica com período 2, pois

$$\text{MDC}\{n/P_{ii}^n > 0\} = \text{MDC}\{2, 4, 6, \dots\} = 2 \quad \text{para } i=1,2$$

Logo pode não ser cumprido o teorema 3.11. Efetivamente

$$P_{11}^n = P_{22}^n = \begin{cases} 0 & \text{se } n \text{ é ímpar} \\ 1 & \text{se } n \text{ é par} \end{cases}$$

e então não existe $\lim_{n \rightarrow \infty} P_{ii}^n$ para $i=1,2$. \blacksquare

OBSERVAÇÕES

1.-Quando $\lim_{n \rightarrow \infty} P_{ij}^n$ existe independentemente do estado i , a matriz

$$P = \lim_{n \rightarrow \infty} P^n \text{ deve ter cada vetor coluna constante e igual a } \pi_i$$

na coluna i -ésima. Ou seja que todas as linhas da matriz P são idênticas e iguais ao vetor $\pi = (\pi_0, \pi_1, \pi_2, \dots)$ onde $S=\{0,1,2,\dots\}$ ou seja que

$$P = \lim_{n \rightarrow \infty} P^n = \begin{pmatrix} \pi_0 & \pi_1 & \pi_2 & \dots \\ \pi_0 & \pi_1 & \pi_2 & \dots \\ \cdot & \cdot & \cdot & \dots \\ \cdot & \cdot & \cdot & \dots \end{pmatrix}$$

2.- O teorema 3.11 assegura a existência de $\lim_{n \rightarrow \infty} P_{ii}^n$, satisfeitas as condições de irredutibilidade, recorrência e aperiodicidade.

Porém, não é simples calcular $\Pi_i, \forall i \in S$. De fato, ao calcular as potências de P , só chegamos a obter uma aproximação de $\lim_{n \rightarrow \infty} P^n$, isso sem levar em conta a dificuldade prática na

obtenção dessas potências. Também não é trivial obter as Π_i

como valores recíprocos de $\mu_i = \sum_{n=1}^{\infty} n \cdot f_{ii}^n$, pois neste caso

precisaríamos dos $f_{ii}^n, \forall n$.

No que segue procuraremos resultados que forneçam formas simples de obter Π_i , pelo menos quando se tenha $\Pi_i > 0$.

3.- Um resultado análogo ao teorema 3.11 pode ser obtido no caso em que a cadeia não seja irredutível. Aqui, o resultado será válido para as classes de equivalência, uma vez mantidas as condições de aperiodicidade e recorrência na classe correspondente.

O teorema 3.11 permite fazer uma classificação dos estados recorrentes, pois para j tal que $\mu_j < \infty$ temos $\lim_{n \rightarrow \infty} P_{jj}^n = 1/\mu_j > 0$

e para $\mu_j = \infty$ temos $\lim_{n \rightarrow \infty} P_{jj}^n = 0$.

De maneira intuitiva, os resultados acima tem a seguinte justificativa: dado um número k de iterações, pode acontecer que antes das k iterações tenhamos voltado uma ou mais vezes ao estado j .

No caso em que o tempo medio de recorrência μ_j é finito, se suposermos que depois de muitas iterações as recorrências ao estado j acontecem exactamente cada μ_j unidades de tempo, então para k suficientemente grande, a probabilidade P_{jj}^k de retornar ao estado j em exactamente k iterações corresponderá à probabilidade de que k seja múltiplo de μ_j , dada por $1/\mu_j > 0$.

No caso em que $\mu_j = \infty$, a medida que o número de iterações aumenta se espera que os tempos de retorno aconteçam cada vez com menor frequência, fazendo com que para k suficientemente grande, P_{jj}^k

seja aproximadamente zero.

DEFINIÇÃO 3.10

Seja j um estado recorrente.

O estado j é RECORRENTE NULO se $\mu_j = \sum_{n=1}^{\infty} n \cdot f_{jj}^n = \infty$.

O estado j é RECORRENTE POSITIVO se $\mu_j < \infty$

Uma cadeia é RECORRENTE NULA (respetivamente RECORRENTE POSITIVA) se todos seus estados são recorrentes nulos (recorrentes positivos).

EXEMPLO 3.14

No exemplo 3.11 foram calculados $\mu_1 = 3/2$ e $\mu_2 = 3$ para os estados 1 e 2 da cadeia do exemplo 3.7. Segundo a definição acima, estes dois estados são recorrente positivos.

Observamos que para a classificação dos estados recorrentes segundo a definição 3.10, não é necessario satisfazer as condições do teorema 3.11, como acontece neste exemplo, ou seja, independentemente desse teorema podemos ter para cada j , $\mu_j = \infty$ ou $\mu_j < \infty$.

Suponhamos agora que $\{X_n\}$ é uma cadeia de Markov com $S = \{1, 2, 3, \dots\}$ e matriz de transição dada por

$$P = \begin{pmatrix} 1/2 & 1/2 & 0 & 0 & 0 & 0 & \dots \\ 1/3 & 0 & 2/3 & 0 & 0 & 0 & \dots \\ 1/4 & 0 & 0 & 3/4 & 0 & 0 & \dots \\ 1/5 & 0 & 0 & 0 & 4/5 & 0 & \dots \\ 1/6 & 0 & 0 & 0 & 0 & 5/6 & \dots \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \dots \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \dots \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \dots \end{pmatrix}$$

Pode-se provar que :

$$f_{11}^1 = P_{11} = 1/2; \quad f_{11}^2 = 1/6; \quad f_{11}^3 = 1/12; \quad f_{11}^4 = 1/20; \dots; \quad f_{11}^n = \frac{1}{n(n+1)}; \dots$$

Logo $\sum_{n=1}^{\infty} f_{11}^n = \sum_{n=1}^{\infty} \frac{1}{n(n+1)} = \sum_{n=1}^{\infty} \left(\frac{1}{n} - \frac{1}{n+1} \right) = 1$ e o estado 1 resulta recorrente.

$$\text{Agora, } \mu_1 = \sum_{n=1}^{\infty} n \cdot f_{11}^n = \sum_{n=1}^{\infty} \frac{1}{n+1} = \infty$$

Em consequência, o estado 1 é recorrente nulo. ■

COROLARIO 3.5

Se numa cadeia de Markov irredutível e aperiódica, um estado j é recorrente positivo então todo estado é recorrente positivo.

PROVA

A recorrência dos outros estados é garantida pelo teorema 3.11. Se j é recorrente positivo, então pelo teorema anterior, temos para cada estado i ,

$$\lim_{n \rightarrow \infty} P_{ij}^{n-r} = \lim_{n \rightarrow \infty} P_{ij}^n = \lim_{n \rightarrow \infty} P_{jj}^n = \Pi_j > 0 \quad (r \text{ fixo e } 1 \leq r \leq n)$$

Como a cadeia é irredutível, $\exists r \geq 1$ tal que $P_{ji}^r > 0$.

Logo, $P_{ii}^n \geq P_{ij}^{n-r} \cdot P_{ji}^r$ (devido ao teorema 3.1)

$$\Rightarrow \lim_{n \rightarrow \infty} P_{ii}^n \geq P_{ji}^r \cdot \lim_{n \rightarrow \infty} P_{ij}^{n-r} = P_{ji}^r \cdot \Pi_j > 0$$

Assim, i é recorrente positivo. ■

OBSERVAÇÃO

A hipótese de aperiodicidade pode ser dispensada. Aqui foi colocada só para obter uma prova simples baseada no teorema anterior.

Um resultado análogo é válido para recorrência nula.

COROLARIO 3.6

Numa cadeia de Markov finita, irredutível e aperiódica todos os estados são recorrentes positivos.

PROVA

A recorrência é garantida pelo corolario 3.4.

Seja $S = \{1, 2, \dots, k\}$ e suponhamos que existe um estado j que é recorrente nulo. De acordo com a observação anterior, todo estado é recorrente nulo.

$$\text{Logo } \lim_{n \rightarrow \infty} P_{ji}^n = \lim_{n \rightarrow \infty} P_{ii}^n = 0 \quad ; \quad \forall i \in S$$

Mas $\sum_{i=1}^k P_{ji}^n = 1 \quad ; \quad \forall n$ (P^n é matriz estocástica)

Assim, $1 = \lim_{n \rightarrow \infty} \sum_{i=1}^k P_{ji}^n = \sum_{i=1}^k \lim_{n \rightarrow \infty} P_{ji}^n = 0$ o que é contradição

O seguinte teorema apresenta uma forma de cálculo dos Π_i quando temos garantida a recorrência positiva dos estados.

TEOREMA 3.12

Seja $\{X_n\}$ uma cadeia de Markov irreduzível, aperiódica e recorrente positiva, com matriz de transição $P = (P_{ij})$ e espaço de estados S .

Seja também $\Pi_j = \lim_{n \rightarrow \infty} P_{ij}^n = \lim_{n \rightarrow \infty} P_{jj}^n \quad ; \quad \forall j \in S$

(a existência dos limites e a igualdade é garantida pelo teorema 3.11)

Então os Π_i 's satisfazem as seguintes condições:

- (a) $\Pi_j > 0 \quad ; \quad \forall j \in S$
- (b) $\sum_{j \in S} \Pi_j = 1$
- (c) $\Pi_j = \sum_{i \in S} \Pi_i \cdot P_{ij} \quad ; \quad \forall j \in S$

PROVA

A condição (a) é verificada trivialmente devido à hipótese de recorrência positiva da cadeia.

Se S for finito, da equação de Chapman-Kolmogorov

$$P_{ij}^{n+1} = \sum_{k \in S} P_{ik}^n \cdot P_{kj}$$

teria-se $\lim_{n \rightarrow \infty} P_{ij}^{n+1} = \Pi_j = \sum_{k \in S} \left(\lim_{n \rightarrow \infty} P_{ik}^n \right) \cdot P_{kj} = \sum_{k \in S} \Pi_k \cdot P_{kj}$

e também $\sum_{j \in S} \Pi_j = \sum_{j \in S} \lim_{n \rightarrow \infty} P_{ij}^n = \lim_{n \rightarrow \infty} \sum_{j \in S} P_{ij}^n = \lim_{n \rightarrow \infty} 1 = 1$.

No caso $|S| = \infty$, obtemos da equação de Chapman-Kolmogorov

$$\Pi_j = \lim_{n \rightarrow \infty} \left(\sum_{k \in S} P_{ik}^n \cdot P_{kj} \right) = \sum_{k \in S} \left(\lim_{n \rightarrow \infty} P_{ik}^n \right) \cdot P_{kj} = \sum_{k \in S} \Pi_k \cdot P_{kj}$$

onde agora a possibilidade de intercâmbio de \lim com \sum é

baseada no teorema de convergência dominada de Lebesgue, se é aceita a hipótese $\sum_{j \in S} P_{kj} < \infty, \forall j \in S$.

No caso em que $|S| = \infty$ e não temos assegurada a hipótese anterior, usamos o seguinte argumento, supondo $S = \{1, 2, 3, \dots\}$

$$\sum_{k=1}^{\infty} P_{ik}^n = 1 \rightarrow \text{para cada inteiro } M, \sum_{k=1}^M P_{ik}^n \leq 1$$

$$\rightarrow \lim_{n \rightarrow \infty} \sum_{k=1}^M P_{ik}^n = \sum_{k=1}^M \lim_{n \rightarrow \infty} P_{ik}^n = \sum_{k=1}^M \Pi_k \leq 1; \forall M \text{ inteiro}$$

$$\rightarrow \sum_{k=1}^M \Pi_k \leq 1$$

$$P_{ij}^{n+1} = \sum_{k=1}^{\infty} P_{ik}^n \cdot P_{kj} \geq \sum_{k=1}^M P_{ik}^n \cdot P_{kj}; \forall M \text{ inteiro}$$

$$\rightarrow \lim_{n \rightarrow \infty} P_{ij}^{n+1} = \Pi_j \geq \lim_{n \rightarrow \infty} \sum_{k=1}^M P_{ik}^n \cdot P_{kj} = \sum_{k=1}^M \Pi_k \cdot P_{kj}; \forall M \text{ inteiro}$$

$$\rightarrow \Pi_j \geq \sum_{k=1}^{\infty} \Pi_k \cdot P_{kj}$$

$$\rightarrow \sum_{j=1}^{\infty} \Pi_j \geq \sum_{j=1}^{\infty} \sum_{k=1}^{\infty} \Pi_k \cdot P_{kj} = \sum_{k=1}^{\infty} \sum_{j=1}^{\infty} \Pi_k \cdot P_{kj} = \sum_{k=1}^{\infty} \Pi_k \left(\sum_{j=1}^{\infty} P_{kj} \right) = \sum_{k=1}^{\infty} \Pi_k$$

onde o intercâmbio das somas é devido à aplicação do teorema de Fubini, garantido por $\Pi_k \cdot P_{kj} \geq 0, \forall k, \forall j$

$$\text{Então } 1 \geq \sum_{j=1}^{\infty} \Pi_j \geq \sum_{j=1}^{\infty} \sum_{k=1}^{\infty} \Pi_k P_{kj} = \sum_{k=1}^{\infty} \Pi_k$$

$$\text{Segue } \sum_{j=1}^{\infty} \Pi_j = \sum_{j=1}^{\infty} \sum_{k=1}^{\infty} \Pi_k P_{kj}$$

Se acontecesse que $\Pi_j > \sum_{k=1}^{\infty} \Pi_k P_{kj}$ para algum j , então a igualdade anterior não seria válida.

Em consequência, $\Pi_j = \sum_{k=1}^{\infty} \Pi_k P_{kj}$, tendo assim provado (c).

Agora provaremos (b), ou seja, $\sum_{j=1}^{\infty} \Pi_j = 1$.

$$\begin{aligned} \text{Temos } \Pi_k &= \sum_{j=1}^{\infty} \Pi_j \cdot P_{jk} = \sum_{j=1}^{\infty} \left(\sum_{i=1}^{\infty} \Pi_i \cdot P_{ij} \right) \cdot P_{jk} \\ &= \sum_{i=1}^{\infty} \Pi_i \cdot \left(\sum_{j=1}^{\infty} P_{ij} \cdot P_{jk} \right) = \sum_{i=1}^{\infty} \Pi_i \cdot P_{ik}^2 \end{aligned}$$

(de novo foi usado o teorema de Fubini para intercâmbio das somas infinitas).

Continuando indutivamente, chegamos a

$$\Pi_k = \sum_{i=1}^{\infty} \Pi_i \cdot P_{ik}^n \quad \text{para } n=1, 2, 3, \dots$$

$$\therefore \lim_{n \rightarrow \infty} \Pi_k = \Pi_k = \lim_{n \rightarrow \infty} \sum_{i=1}^{\infty} \Pi_i \cdot P_{ik}^n = \sum_{i=1}^{\infty} \Pi_i \cdot \left(\lim_{n \rightarrow \infty} P_{ik}^n \right) = \sum_{i=1}^{\infty} \Pi_i \cdot \Pi_k$$

De novo usamos o teorema da Convergência Dominada de Lebesgue para intercâmbio de lim com \sum . As condições para a correta aplicação são :

- $|\Pi_i \cdot P_{ik}^n| \leq \Pi_i$ (pois $P_{ik}^n \leq 1$ e $\Pi_i \geq 0$) o que quer dizer que a sequência $\left\{ \Pi_i \cdot P_{ik}^n \right\}_{n=1}^{\infty}$ é limitada
- $\sum_{i=1}^{\infty} \Pi_i < \infty$ (já foi provado que $\sum_{i=1}^{\infty} \Pi_i \leq 1$)

$$\text{Logo } \Pi_k = \sum_{i=1}^{\infty} \Pi_i \cdot \Pi_k = \Pi_k \cdot \sum_{i=1}^{\infty} \Pi_i$$

$$\text{Como } \Pi_k > 0 \text{ então } \sum_{i=1}^{\infty} \Pi_i = 1 \quad \blacksquare$$

OBSERVAÇÕES

1.- O conjunto $\left\{ \Pi_j \right\}_{j \in S}$ satisfazendo as condições (a), (b) e (c) no teorema anterior, é chamado "distribuição de probabilidades estacionária" da cadeia de Markov $\left\{ X_n \right\}$.

As condições (a) e (b) justificam o nome "distribuição de probabilidades", enquanto a condição (c) tem a ver com a parte "estacionária". Essa relação é equivalente com $\Pi = \Pi \cdot P$ em notação matricial, onde $\Pi = (\Pi_1, \Pi_2, \dots, \Pi_s)$ e P é a matriz de transição.

Assim, $\Pi = \Pi \cdot P = (\Pi \cdot P) \cdot P = \Pi \cdot P^2$ e indutivamente conclui-se que $\Pi = \Pi \cdot P^n$; $\forall n \geq 1$.

Logo, se X_0 tem distribuição de probabilidades Π , então a

distribuição de probabilidades de X_n é $\Pi \cdot P^n = \Pi$, ou seja que a distribuição de probabilidades é invariante frente ao tempo.

2.- O teorema 3.13 fornece uma forma de calcular $\Pi_j = \lim_{n \rightarrow \infty} P_{ij}^n$ para cada estado j numa cadeia de Markov onde é garantida a existência desses limites, sem precisar dos cálculos das potências de P nem dos tempos medios de recorrência μ_i .

3.- Os valores Π_j no Teorema 3.12 são univocamente determinados pelas equações:

$$\Pi_j \geq 0, \forall j \in S; \quad \sum_{j \in S} \Pi_j = 1; \quad \Pi_j = \sum_{i \in S} \Pi_i \cdot P_{ij}, \forall j \in S$$

(a condição $\Pi_j \geq 0$ é menos restritiva do que $\Pi_j > 0$, facilitando em termos operacionais o cálculo de Π_j . O fato da cadeia ser recorrente positiva, garante que obteremos $\Pi_j > 0$).

Para provar esta observação, suponhamos que:

$$x_k = \sum_{i=1}^{\infty} x_i \cdot P_{ik}, \forall k; \quad \sum_{i=1}^{\infty} x_i = 1 \quad \text{e} \quad x_i \geq 0, \forall i$$

Obtemos como no teorema, $x_k = \sum_{i=1}^{\infty} x_i \cdot P_{ik}^n; \quad \forall n=1,2,3,\dots$

(nessa prova não foi usado $\Pi_i > 0$, senão $\Pi_i \geq 0$ na aplicação do Teorema de Fubini).

$$\therefore x_k = \lim_{n \rightarrow \infty} x_k = \lim_{n \rightarrow \infty} \sum_{i=1}^{\infty} x_i \cdot P_{ik}^n = \sum_{i=1}^{\infty} x_i \cdot \Pi_k$$

(de novo usamos Teorema da Convergência Dominada de Lebesgue)

$$\therefore x_k = \sum_{i=1}^{\infty} x_i \cdot \Pi_k = \Pi_k \cdot \sum_{i=1}^{\infty} x_i = \Pi_k.$$

O seguinte exemplo mostra o procedimento indicado na observação 2.

EXEMPLO 3.15

Seja $P = \begin{bmatrix} 1/2 & 0 & 1/2 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$ a matriz de transição de uma cadeia

de Markov com espaço de estados $S = \{1,2,3\}$.

$$\text{Temos } P^2 = \begin{bmatrix} 1/4 & 1/2 & 1/4 \\ 1/2 & 0 & 1/2 \\ 1 & 0 & 0 \end{bmatrix}; \quad P^3 = \begin{bmatrix} 5/8 & 1/4 & 1/8 \\ 1/4 & 1/2 & 1/4 \\ 1/2 & 0 & 1/2 \end{bmatrix}; \quad P^4 = \begin{bmatrix} 9/16 & 61/8 & 5/16 \\ 5/8 & 1/4 & 1/8 \\ 1/4 & 1/2 & 1/4 \end{bmatrix}$$

Estas potências só forem calculadas para obter de maneira

simples as condições de irredutibilidade $\left[P_{ij}^4 > 0, \forall i, j \in S \right]$, aperiodicidade $\left[P_{ii}^3 > 0 \text{ e } P_{ii}^4 > 0, \forall i \text{ e também } \text{MDC}\{3,4\}=1 \right]$ e recorrência positiva (devido ao corolário 3.6), que garantem a existência de $\Pi_i = \lim_{n \rightarrow \infty} P_{ii}^n$; $\forall i=1,2,3$

Para achar a distribuição $\Pi = (\Pi_1, \Pi_2, \Pi_3)$ resolvemos o sistema

$$\begin{array}{l} (\Pi_1, \Pi_2, \Pi_3) = (\Pi_1, \Pi_2, \Pi_3) \cdot P \\ \hline \Pi_1 + \Pi_2 + \Pi_3 = 1 \end{array}$$

ou equivalentemente

$$\begin{array}{l} \Pi_1/2 + \Pi_2 = \Pi_1 \\ \Pi_3 = \Pi_2 \\ \Pi_1/2 = \Pi_3 \\ \hline \Pi_1 + \Pi_2 + \Pi_3 = 1 \end{array}$$

onde as primeiras três equações são dependentes, coisa que sempre acontece com $\Pi = \Pi \cdot P$.

A solução (única) do sistema anterior é $\Pi_1=1/2$; $\Pi_2=1/4$; $\Pi_3=1/4$,

ou seja que $\lim_{n \rightarrow \infty} P^n = \begin{pmatrix} 1/2 & 1/4 & 1/4 \\ 1/2 & 1/4 & 1/4 \\ 1/2 & 1/4 & 1/4 \end{pmatrix}$ ou $\begin{cases} P[X_n=1] \xrightarrow{n \rightarrow \infty} 1/2 \\ P[X_n=2] \xrightarrow{n \rightarrow \infty} 1/4 \\ P[X_n=3] \xrightarrow{n \rightarrow \infty} 1/4 \end{cases}$

O seguinte exemplo mostra o que pode acontecer se a hipótese de recorrência positiva é dispensada.

EXEMPLO 3.16

Seja $\{X_n\}$ uma cadeia de Markov com espaço de estados $S = \{1, 2, 3, \dots\}$ e matriz de transição

$$P = \begin{pmatrix} 1/2 & 1/2 & 0 & 0 & 0 & \dots \\ 1/4 & 0 & 3/4 & 0 & 0 & \dots \\ 1/8 & 0 & 0 & 7/8 & 0 & \dots \\ 1/16 & 0 & 0 & 0 & 15/16 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}$$

A cadeia é irredutível $\left(i < j \Rightarrow P_{ij} \geq P_{i,i+1} \cdot P_{i+1,i+2} \cdot \dots \cdot P_{j-1,j} > 0 \right)$ e também $\left(P_{ji} \geq P_{j,1} \cdot P_{1,2} \cdot \dots \cdot P_{i-1,i} > 0 \right)$ e aperiódica $\left[P_{11} > 0 \text{ resultando o estado 1 aperiódico e como a cadeia é irredutível, de acordo ao corolário 3.2, a cadeia é aperiódica} \right]$.

Porém, a cadeia não é recorrente. De acordo com teorema 3.10 é

suficiente provar que o estado 1 não é recorrente. Efetivamente

$$f_{11} = 1/2; \quad f_{11}^2 = (1/2)^2; \quad f_{11}^3 = (1/2)^3 (3/4); \quad f_{11}^4 = (1/2)^4 (3/4)(7/8)$$

$$\text{e} \quad f_{11}^n = \left(\frac{1}{2}\right)^{n+1} \cdot \frac{3}{4} \cdot \frac{7}{8} \cdot \frac{15}{16} \cdot \dots \cdot \frac{2^{n-1}-1}{2^{n-1}}; \quad \forall n \geq 3$$

$$\therefore \sum_{n=1}^{\infty} f_{11}^n = \frac{1}{2} + \left(\frac{1}{2}\right)^2 + \sum_{n=3}^{\infty} f_{11}^n$$

$$= \frac{1}{2} + \left(\frac{1}{2}\right)^2 + \sum_{n=3}^{\infty} \left(\frac{1}{2}\right)^{n+1} \cdot \left(\frac{3}{4} \cdot \frac{7}{8} \cdot \frac{15}{16} \cdot \dots \cdot \frac{2^{n-1}-1}{2^{n-1}}\right)$$

$$< \frac{1}{2} + \left(\frac{1}{2}\right)^2 + \sum_{n=3}^{\infty} \left(\frac{1}{2}\right)^{n+1} \quad \text{pois} \quad \frac{3}{4} \cdot \frac{7}{8} \cdot \frac{15}{16} \cdot \dots \cdot \frac{2^{n-1}-1}{2^{n-1}} < 1$$

$$= \frac{1}{2} + \left(\frac{1}{2}\right)^2 + \frac{2^{-4}}{1-(1/2)} = \frac{3}{4}$$

$$\therefore \sum_{n=1}^{\infty} f_{11}^n = \alpha < 3/4 \text{ e o estado 1 é transitório.}$$

Se tentamos aplicar o teorema 3.12 nesta cadeia que não satisfaz a condição de recorrência positiva, teremos

$$\begin{aligned} \Pi \cdot P &= \Pi \quad \text{e e e} \quad \frac{1}{2} \Pi_1 + \left(\frac{1}{2}\right)^2 \Pi_2 + \left(\frac{1}{2}\right)^3 \Pi_3 + \dots &= \Pi_1 \\ & & \frac{1}{2} \Pi_1 &= \Pi_2 \\ & & \frac{3}{4} \Pi_2 &= \Pi_3 \\ & & \frac{7}{8} \Pi_3 &= \Pi_4 \\ & & & \vdots \\ & & & \vdots \\ & & & \vdots \\ & & & \dots \end{aligned}$$

ou equivalentemente

$$\begin{aligned} \frac{1}{2}\pi_1 + \left(\frac{1}{2}\right)^2 \pi_2 + \left(\frac{1}{2}\right)^3 \pi_3 + \dots &= \pi_1 \\ \pi_2 &= \frac{1}{2} \pi_1 \\ \pi_3 &= \frac{1}{2} \cdot \frac{3}{4} \pi_1 \\ \pi_4 &= \frac{1}{2} \cdot \frac{3}{4} \cdot \frac{7}{8} \pi_1 \\ &\vdots \\ &\vdots \end{aligned}$$

Substituindo na primeira equação, obtemos

$$\begin{aligned} \pi_1 &= \frac{1}{2}\pi_1 + \left(\frac{1}{2}\right)^3 \pi_1 + \left(\frac{1}{2}\right)^4 \cdot \frac{3}{4} \pi_1 + \left(\frac{1}{2}\right)^5 \cdot \frac{3}{4} \cdot \frac{7}{8} \pi_1 + \dots \\ \Leftrightarrow \pi_1 \left[1 - \left[\frac{1}{2} + \left(\frac{1}{2}\right)^3 + \left(\frac{1}{2}\right)^4 \cdot \frac{3}{4} + \left(\frac{1}{2}\right)^5 \cdot \frac{3}{4} \cdot \frac{7}{8} + \left(\frac{1}{2}\right)^{n+1} \cdot \frac{3}{4} \cdot \frac{7}{8} \cdot \dots \cdot \frac{2^{n-1}-1}{2^{n-1}} + \dots \right] \right] &= 0 \\ \Leftrightarrow \pi_1 \left(1 - \sum_{n=1}^{\infty} f_{11}^n \right) &= 0 \end{aligned}$$

e como $\sum_{n=1}^{\infty} f_{11}^n < 1$ então $\pi_1 = 0$. Assim, $\pi = 0$ o que contradiz $\sum_{j \in S} \pi_j = 1$.
O resultado $\pi = 0$ pode também ser obtido pelo fato de que se qualquer estado j é transitório, então $\sum_{n=1}^{\infty} P_{jj}^n < \infty$ (devido à observação no teorema 3.9) o que implica em que $\lim_{n \rightarrow \infty} P_{jj}^n = \pi_j = 0$.

Agora, se j fosse um estado recorrente nulo, então devido à irredutibilidade (ver observação ao corolário 3.5) a cadeia é recorrente nula. Assim, para cada estado j teremos $\mu_j = \infty$ o que leva (teorema 3.11) a $\lim_{n \rightarrow \infty} P_{jj}^n = \pi_j = 0$.

■

RESUMO DO CAPITULO

São apresentados os conceitos e resultados mais relevantes relacionados com as cadeias de Markov e que serão utilizados no capítulo seguinte.

P_{ij} = probabilidade de passar do estado i ao estado j num passo, numa cadeia de Markov homogênea.

$$= P[X_{n+1}=j / X_n=i] = P[X_n=j / X_{n-1}=i] = \dots = P[X_1=j / X_0=i]$$

P_{ij}^n = probabilidade de passar do estado i ao estado j em exatamente n transições, numa cadeia de Markov homogênea.

$$= P[X_{n+m}=j / X_m=i] = P[X_n=j / X_0=i]$$

$$= \sum_{k \in S} P_{ik}^r \cdot P_{kj}^s \quad (r+s=n) \quad \text{Equação de Chapman-Kolmogorov}$$

Dois estados i e j estão COMUNICADOS ($i \leftrightarrow j$) se existem $n, m \geq 0$ tais que $P_{ij}^n > 0$ e $P_{ji}^m > 0$.

A relação de comunicação é uma relação de equivalência. (teorema 3.2).

Uma cadeia de Markov é IRREDUTIVEL se todo estado está comunicado com todo outro estado.

O PERIODO $d(j)$ de um estado j é $d(j) = \text{MDC} \{ n / P_{jj}^n > 0 \}$

Uma cadeia é APERIODICA se todos seus estados tem período 1.

Estados comunicados tem igual período (teorema 3.4).

Se numa cadeia de Markov irredutível existe um estado i tal que $P_{ii}^n > 0$ então a cadeia é aperiódica (corolário 3.2).

f_{ii}^n = probabilidade de que o primeiro retorno ao estado i aconteça na n -ésima iteração.

$$= P[X_n=i, X_{n-1} \neq i, X_{n-2} \neq i, \dots, X_1 \neq i / X_0=i]$$

$$\text{Para cada estado } i, \quad F_{ii}^n = \sum_{k=1}^n f_{ii}^k \cdot P_{ii}^{n-k} \quad (n \geq 1)$$

Um estado i é RECORRENTE sse $\sum_{n=1}^{\infty} f_{ii}^n = 1$ (ou sse $\sum_{n=1}^{\infty} P_{ii}^n = \infty$)

Um estado i é TRANSITORIO sse $\sum_{n=1}^{\infty} f_{ii}^n < 1$ (ou sse $\sum_{n=1}^{\infty} P_{ii}^n < \infty$)

Estados comunicados são todos recorrentes ou todos transitórios (teorema 3.10).

Se numa cadeia de Markov irredutível existe um estado recorrente (transitório) i , então todos os estados são recorrentes (transitórios), ou seja a cadeia é recorrente (transitória).

Se numa cadeia de Markov irredutível o espaço de estados S é finito, então a cadeia é recorrente (corolário 3.4).

O TEMPO MEDIO DE RECORRÊNCIA $\mu_i = \begin{cases} \sum_{n=1}^{\infty} n \cdot f_{ii}^n & \text{se } i \text{ é recorrente} \\ \infty & \text{se } i \text{ é transitório} \end{cases}$

Uma cadeia de Markov é dita ERGÓDICA se $\lim_{n \rightarrow \infty} P_{ij}^n = \pi_j$ existe para cada $j \in S$, independentemente do estado inicial i , e $\sum_{j \in S} \pi_j = 1$.

Teorema 3.11

Se $P = (P_{ij})$ é a matriz de transições de uma cadeia de Markov irredutível, aperiódica e recorrente então

- $\lim_{n \rightarrow \infty} F_{ii}^n$ existe
- $\pi_i = \lim_{n \rightarrow \infty} F_{ii}^n = \begin{cases} 1/\mu_i & \text{se } \mu_i < \infty \\ 0 & \text{se } \mu_i = \infty \end{cases}$
- $\forall j \in S, \lim_{n \rightarrow \infty} P_{ji}^n = \lim_{n \rightarrow \infty} P_{ii}^n = \pi_i$.

Um estado recorrente j é RECORRENTE NULO (recorrente positivo) se $\mu_j = \infty$ ($\mu_j < \infty$). A cadeia é RECORRENTE NULA (POSITIVA) se todos seus estados são recorrentes nulos (positivos).

Numa cadeia de Markov finita, irredutível e aperiódica, todos os estados são recorrentes positivos.

Teorema 3.12

Se $P = (P_{ij})$ é a matriz de transição de uma cadeia de Markov irredutível, aperiódica e recorrente positiva, então

$\pi_j = \lim_{n \rightarrow \infty} P_{ij}^n = \lim_{n \rightarrow \infty} P_{jj}^n, \forall j \in S$ satisfazem as condições:

- $\pi_j > 0, \forall j \in S$
- $\sum_{j \in S} \pi_j = 1$
- $\pi_j = \sum_{i \in S} \pi_i \cdot P_{ij}, \forall j \in S$

O conjunto $\{\pi_j\}_{j \in S}$ satisfazendo (a), (b), e (c) é chamado "distribuição de probabilidades estacionária" da cadeia.

C A P I T U L O 4

ASPECTOS GERAIS DO ALGORITMO SIMULATED ANNEALING

4.1 INTRODUÇÃO

Muitos problemas de otimização combinatória pertencem a uma classe de problemas que são difíceis de resolver, a classe de problemas NP-completos. Para eles não são conhecidos algoritmos exatos cujas complexidades de pior caso estejam limitadas por um polinômio no tamanho da entrada.

Os algoritmos heurísticos são usados para resolver problemas NP-completos de maneira aproximada (são chamados também de algoritmos aproximados), ou seja, algoritmos que fornecem "boas" soluções viáveis, no sentido de que o custo associado esteja próximo do custo ótimo.

Uma classe de heurísticas que tem-se comportado muito bem para alguns problemas de otimização combinatória são as HEURÍSTICAS DE BUSCA LOCAL, ou de busca em vizinhança, como por exemplo a de Lin-Kernigham [40] para o problema do caixeiro viajante, dentre outras. Nessas heurísticas é definido para cada solução viável i do problema, uma estrutura de vizinhança $S(i)$, ou seja, uma regra de geração de soluções viáveis a partir de i , de modo que

$$S(i) = \{j / j \text{ é solução viável que pode ser gerada a partir de } i\}$$

As heurísticas de busca local produzem uma sequência de soluções viáveis de modo tal que a solução seguinte de i é alguma solução $j \in S(i)$ com custo menor do que i (considerando o problema como de minimização). Geralmente pára-se a busca local em $S(i)$ quando encontramos uma solução melhor j ou depois de percorrer a vizinhança toda para ficar com a melhor solução j encontrada, o que a caracteriza como "gulosa". Na seguinte iteração, a busca é feita dentro de $S(j)$ e assim sucessivamente.

O algoritmo pára quando não é encontrada uma solução melhor que a atual k , dentro de $S(k)$. Então k é chamada de SOLUÇÃO LOCALMENTE ÓTIMA.

Este tipo de heurísticas apresentam basicamente duas

deficiências. A primeira é que a solução finalmente encontrada depende fortemente da solução inicial escolhida pelo algoritmo, sendo por esse fato, muito comum a repetição do algoritmo uma certa quantidade de vezes, começando de diferentes soluções e prosseguindo até que um certo limite de tempo computacional seja atingido, ou até que a resposta seja satisfatória. A outra deficiência neste tipo de heurísticas é que ela tem alta probabilidade de ficar "presa" em ótimos locais, longe do ótimo global, especificamente para instâncias com muitos ótimos locais.

Considerando a escolha da solução seguinte pelo algoritmo como um processo de duas etapas, uma de geração e outra de aceitação, os algoritmos de busca local se caracterizam por uma geração aleatória e uma aceitação gulosa em que só são aceitas soluções que melhoram a atual.

Algumas variantes neste tipo de heurísticas são aquelas que apresentam outro tipo de regra de geração (por exemplo usando algum critério de ordenação) ou de aceitação como nos seguintes dois casos:

- (i) aceitar qualquer solução gerada, tanto de melhora ou de piora na solução atual
- (ii) aceitar de acordo com algum critério (que pode ser probabilístico) considerando a função objetivo.

O caso (ii) resulta em uma alternativa intermediária entre a gulosa e (i) pois não é tão cega quanto a gulosa nem tão conformista quanto a (i).

O algoritmo a ser estudado neste capítulo é conhecido como " ALGORITMO SIMULATED ANNEALING " que será referido como algoritmo S.A. Ele foi inicialmente concebido como de geração aleatória e aceitação probabilística e é considerado como uma generalização dos algoritmos de busca local, mais concretamente como um algoritmo de busca local estocástico [36],[41].

A aceitação probabilística consiste em aceitar com probabilidade 1 qualquer solução que melhore a atual e aceitar com probabilidade p ($0 < p < 1$) qualquer solução que piore a atual.

Geralmente p depende da função objetivo ou função de custo

do problema tratado.

O fato de aceitar probabilisticamente soluções que pioram a solução atual, implica na possibilidade de escapar de um ótimo local (mínimo local em nosso caso) ou de caminhar "morro acima", propriedade que caracteriza os algoritmos conhecidos como "hill climbing" [52]. Sendo assim, o algoritmo S.A. é um caso particular dos algoritmos "hill climbing", onde é utilizada uma particular função ou critério de aceitação.

Em Golden [27] é apresentado um exemplo simples do problema de caixeiro viajante, onde um algoritmo de busca local fica preso num mínimo local. Ao utilizar função de aceitação probabilística, se consegue atingir uma solução com custo menor que a solução dada pelo algoritmo de busca local.

4.2 FUNDAMENTOS DO ALGORITMO SIMULATED ANNEALING

A proposta básica do algoritmo S.A. vem de uma analogia com a física dos materiais. É sabido que o resfriamento lento e gradual de um material a partir de uma temperatura alta inicial finalmente leva o material a estruturas de cristalização de mínima energia. Quando se atinge tais estagios ou configurações do sistema de átomos, se diz que o sistema está "congelado".

A este processo cuidadoso de resfriamento dá-se o nome de "annealing".

Metropolis et al [43], apresentam um procedimento computacional que faz uma simulação do movimento atômico. Essa simulação, feita para uma temperatura fixa T , consiste em uma sequência de passos. A cada passo é dado um pequeno deslocamento (ou perturbação) a um dos átomos, obtendo assim uma nova configuração do sistema de átomos. É computada a variação ΔE que a energia do sistema sofre com aquele deslocamento. Se $\Delta E \leq 0$, o deslocamento é incorporado ao estado do sistema (é aceita a nova configuração do sistema), que é então utilizada para o passo seguinte. Caso contrario ($\Delta E > 0$) a aceitação ou não do deslocamento passa a ser uma decisão probabilística. A probabilidade de que a nova configuração seja aceita é

$$P[\Delta E] = \exp(-\Delta E/kT)$$

onde k é a constante de Boltzman e T é a temperatura medida na escala de Kelvin.

Kirkpatrick [36] e independentemente Cerny [14] incorporam o algoritmo de Metropolis num processo análogo ao "annealing", para gerar um algoritmo na resolução de problemas de otimização combinatoria. O algoritmo é chamado Simulated Annealing e consiste em iniciar o processo com um valor apropriadamente alto de temperatura. O algoritmo compõe-se de uma sequência de passos, cada um deles reduzindo a temperatura numa pequena quantidade até que esta seja reduzida às proximidades de zero. Em cada passo, ou seja para uma dada temperatura fixada em um certo patamar, um procedimento semelhante ao utilizado por Metropolis é usado, visando atingir um certo estado que representa, do ponto de vista físico, um estado de equilíbrio do sistema atômico nesse patamar de temperatura.

Assim, o nome "Simulated Annealing" refere-se ao uso dessa técnica de simulação em conjunto com o esquema de "annealing" na redução das temperaturas.

Na aplicação do algoritmo S.A. em problemas de otimização combinatoria, são associados :

- cada configuração do sistema com uma solução viável do problema
- uma nova configuração do sistema produto do deslocamento de um átomo, com uma solução vizinha da atual
- a energia do sistema com a função objetivo ou função de custo
- um estado de mínima energia com uma solução ótima global e
- a temperatura T é considerada como um parâmetro de controle da mobilidade do sistema, ou seja, traduz a flexibilidade de aceitar mudanças que aumentam o valor na função de custo (problema de minimização). Devido a isso, no caso de otimização não é necessario considerar a constante k de Boltzman, resultando a probabilidade

$$P[\Delta E] = \exp(-\Delta E/T)$$

Cabe assinalar que quanto menor a temperatura T , menor será esta probabilidade $P[\Delta E]$ de aceitar soluções que pioram a

função objetivo ($\Delta E > 0$), o que se traduz em rejeitar cada vez mais aquele tipo de soluções.

No caso extremo, ou seja $T=0$, o algoritmo S.A. se comporta como um algoritmo descendente (ou de melhoras sucessivas) pois nesse caso $P[\Delta E]=0$ quando $\Delta E > 0$, que se traduz em não aceitar soluções que pioram a função objetivo.

A seguir apresentamos a estrutura geral do algoritmo Simulated Annealing sobre um problema de otimização combinatoria qualquer, cujo espaço de soluções viáveis seja não vazio, garantindo assim a existência de um ótimo global.

ALGORITMO SIMULATED ANNEALING

- 1 Obter uma solução viável inicial i .
- 2 Obter uma temperatura inicial $T=T_0 > 0$.
- 3 Enquanto (não seja atingida uma condição de parada), fazer
 - 3.1 Enquanto (não seja atingida uma condição de equilíbrio), fazer
 - 3.1.1 Gerar uma solução viável j , vizinha de i
 - 3.1.2 Fazer $\Delta = \text{custo}(j) - \text{custo}(i)$
 - 3.1.3 Se $\Delta \leq 0$, fazer $i:=j$
 - 3.1.4 Se $\Delta > 0$ fazer
 - 3.1.4.1 Gerar um número aleatório x uniformemente distribuído no intervalo $(0,1)$
 - 3.1.4.2 Se $x < P[\Delta] = \exp(-\Delta/T)$ fazer $i:=j$
 - 3.1.4.3 Caso contrario ($x \geq P[\Delta]$) fazer $i:=i$
 - 3.2 Reduzir a temperatura
- 4.-A solução é i .

Algumas observações neste algoritmo:

- O loop interno 3.1 corresponde ao algoritmo de Metropolis e pode implicar num número infinito de passos. Na prática, a condição de equilíbrio é relaxada limitando este loop a uma quantidade finita L de iterações. Porém, na formulação matemática do algoritmo apresentada a seguir, manteremos o conceito de " condição de equilíbrio " que será devidamente

representado em termos de cadeias de Markov.

- O loop externo 3 no algoritmo corresponde ao decrescimento gradual de temperatura, que também pode levar a um número infinito de passos.
- O algoritmo assim formulado não necessariamente é finito devido tanto ao que pode acontecer no loop interno como no loop externo.
- Para fins práticos, será preciso portanto prover um critério de parada para 3.1 e através da discretização do esquema de redução de temperaturas, garantir um número finito de passos em 3.
- No passo 3.1.4 do algoritmo, é usado um número aleatório x para implementar a parte probabilística do algoritmo de Metropolis, ou seja, para decidir a aceitação ou rejeição da nova solução.

A aceitação da nova solução (no caso $\Delta > 0$) deve ser feita com probabilidade $P[\Delta]$, ou seja, que a probabilidade de aceitar a nova solução será $P[\Delta]$.

Se X é uma variável aleatória uniformemente distribuída em $(0,1)$ temos

$$P[X < a] = \begin{cases} 0 & \text{se } a \leq 0 \\ a & \text{se } 0 \leq a \leq 1 \\ 1 & \text{se } a \geq 1 \end{cases}$$

e dado que $0 < P[\Delta] \leq 1$, então $P[X < P[\Delta]] = P[\Delta]$.

Assim, " $X < P[\Delta]$ " e "aceitar a nova solução", devem ser eventos equiprováveis.

Logo, gerando um número aleatório x uniformemente distribuído em $(0,1)$, a probabilidade de que $X < P[\Delta]$ será $P[\Delta]$, sendo equivalente à probabilidade de aceitar a nova solução viável do processo S.A.

- No passo 3.1.3 é garantida a aceitação da solução vizinha j se ela melhora (reduz) o valor na função objetivo.

Para a efetiva aplicação do algoritmo S.A. a um problema de otimização combinatória, será necessário considerar os seguintes aspectos:

- 1.- A especificação do que é uma solução viável para o problema e a determinação do custo associado com ela.

- 2.- A determinação de uma solução viável inicial.
- 3.- A determinação da temperatura inicial T_0 . Esta deve ser suficientemente alta de modo que inicialmente quase toda configuração (solução viável) seja aceita, o que fica garantido pelo fato de $P[\Delta E]$ ser aproximadamente igual a 1.
- 4.- A determinação do que significa uma solução vizinha, ou seja do esquema de perturbação (ou transformação ou geração) de novas soluções viáveis.
- 5.- A definição de estado de equilíbrio para cada valor de temperatura T .
- 6.- A determinação do esquema de esfriamento, ou seja, a sequência $\{T_m\}$ de temperaturas, estritamente decrescente e convergente a zero.
- 7.- O critério de parada do algoritmo, ou seja, o critério que especifica uma temperatura mínima T_f tal que o algoritmo pára quando $T_m < T_f$ para algum inteiro m tal que $0 < m < \infty$ [41].

Todos estes aspectos devem ser considerados na implementação do algoritmo S.A. Evidentemente passamos a ter um algoritmo heurístico ao restringi-lo a um número finito de iterações.

Por enquanto vamos considerar o algoritmo S. A. geral apresentado anteriormente, efetuando um número infinito de passos no loop externo e gerando um número infinito de soluções viáveis no loop interno. O objetivo é provar sua convergência assintótica baseada em resultados existentes para cadeias de Markov, que servem de modelo matemático para o algoritmo.

4.3 FORMULAÇÃO MATEMÁTICA DO ALGORITMO S.A.

Consideremos um problema de otimização combinatória como de minimização, especificado por um conjunto finito $S = \{1, 2, \dots, |S|\}$ de estados (ou soluções viáveis) e por uma função de custo c que seja não constante e definida para cada $j \in S$. Assim, é garantida a existência de ótimo global para o problema e a existência de soluções viáveis que não são ótimos globais.

Vamos supor que o algoritmo S.A. especifica:

- (a) uma regra de geração de soluções viáveis, a partir de outra

solução viável qualquer. Com isso fica definida a estrutura de vizinhança para o problema considerado.

Seja então $S(i)$ a vizinhança do estado i , ou seja,

$$S(i) = \{ j / j \text{ pode ser gerado a partir de } i \}$$

Por exemplo, se $S = \{0,1\}^4 = \{(x_1, x_2, x_3, x_4) / x_i \in \{0,1\}\}$

$i=(1,0,0,0)$ e a regra de geração consiste na troca de uma das componentes de i de 1 para 0 ou de 0 para 1, então

$$S(i) = \{ (0,0,0,0), (1,1,0,0), (1,0,1,0), (1,0,0,1) \}$$

Vamos supor também que $i \in S(i)$ e que a estrutura de vizinhança é simétrica no sentido de que $j \in S(i)$ sse $i \in S(j)$.

Este suposto além de natural, não afeta a convergência assintótica do algoritmo.

Definimos $g_{ij}(T)$ como a probabilidade de que o estado j seja gerado a partir do estado i , à temperatura T .

Logo, $g_{ij}(T) > 0$ sse $j \in S(i)$.

$$\therefore S(i) = \{ j / g_{ij}(T) > 0 \}$$

Com isto fica claro que se $j \notin S(i)$ então $g_{ij}(T) = 0$

Se além disso supoemos que o algoritmo sempre gera uma nova solução a partir da solução atual i , então temos

$$\sum_{j \in S(i)} g_{ij}(T) = 1$$

Como exemplo destas funções de geração temos a uniforme, ou seja,

$$g_{ij}(T) = \frac{1}{|S(i)|} ; \forall j \in S(i)$$

onde todos os vizinhos de i são gerados com igual probabilidade.

Em geral, a função $g_{ij}(T)$ pode depender do parâmetro de controle T , embora o mais comum seja que g não dependa de T como acontece no exemplo anterior. Para maior generalidade vamos continuar supondo essa dependência.

(b) uma regra ou função probabilística de aceitação.

Se i é a solução atual e $j \in S(i)$ então a função de aceitação é definida por

$$a_{ij}(T) = \begin{cases} \exp(-\Delta c_{ij}/T) & \text{se } \Delta c_{ij} \geq 0 \\ 1 & \text{se } \Delta c_{ij} \leq 0 \end{cases} \quad (1)$$

ou equivalentemente $a_{ij}(T) = \min \{ 1 ; \exp(-\Delta c_{ij}/T) \}$
 onde $\Delta c_{ij} = c(j) - c(i)$

Assim, teremos $0 \leq a_{ij}(T) \leq 1$; $\forall i, j \in S$; $\forall T > 0$ e $a_{ij}(T)$ será interpretada como a probabilidade de aceitar a solução viável j se a solução atual é i .

Notemos que : se $T \rightarrow 0$ então $a_{ij}(T) \rightarrow 0$ para $\Delta c_{ij} > 0$,
 se $T \rightarrow 0$ então $a_{ij}(T) \rightarrow 1$ para $\Delta c_{ij} < 0$,
 e $\lim_{T \rightarrow 0} a_{ij}(T)$ existe para $\Delta c_{ij} = 0$.

Estas condições são satisfeitas também por outras funções de aceitação como aparece em Anily [3].

(c) Uma regra de atualização de temperaturas, ou seja, uma sequência $\{T_m\}_{m=0}^{\infty}$ estritamente decrescente com limite zero e tal que $T_m > 0 \quad \forall m = 0, 1, 2, \dots$

Embora os resultados da convergência assintótica do algoritmo S.A. só usem estas propriedades de $\{T_m\}$, é comum usar na implementação, alguma sequência tal que $T_{m+1} = \alpha_m T_m$ sendo $0 < \alpha_m < 1$; $\forall m = 0, 1, 2, \dots$ onde inclusive α_m pode ser constante durante o processo todo, ou durante um certo número de iterações.

O algoritmo S.A. começa com uma solução viável inicial $i_0 \in S$ e produz, para cada valor de T , uma sequência $\{i_n\}_{n=1}^{\infty}$ de soluções viáveis, sendo $i_n \in S$ a solução produzida na n -ésima iteração do loop interno associado a esse valor de T .

Evidentemente poderá haver, caso a solução gerada não seja aceita, alguns estados repetidos na sequência $\{i_n\}_{n=1}^{\infty}$, por exemplo se na iteração $n+1$ se produz rejeição da solução viável gerada pelo algoritmo, então teremos $i_n = i_{n+1}$, ou seja, é

considerada como nova solução a mesma da iteração anterior (passo 3.1.4.3 no algoritmo). Para cada valor de T numa dada iteração n , cada estado $j \in S$ terá associado uma certa probabilidade (positiva ou zero) de ser atingido. Ela é zero por exemplo se $j \notin S(i_{n-1})$ e será positiva se $j \in S(i_{n-1})$.

Assim, para cada T fixo podemos considerar $X_n(T)$ ($n=1,2,\dots$) como a variável aleatória que indica o estado obtido na n -ésima iteração.

O espaço de estados para cada $X_n(T)$ é $S = \{ 1,2,\dots, |S| \}$.

Então, quando for escrito $X_n(T) = k$, devemos interpretar como que na n -ésima iteração associada ao valor de temperatura T , o algoritmo se encontra na solução viável k .

Logo, o algoritmo S.A. para cada valor de T (uma iteração do loop externo 3 no algoritmo) pode ser considerado um processo estocástico $\{X_n(T)\}_{n=1}^{\infty}$ de tempo (n) discreto e espaço de estados finito S , ou seja, uma cadeia de tempo discreto.

Sendo assim, o algoritmo S.A. completo, ou seja com T variando de acordo com o esquema de esfriamento $\{T_m\}_{m=1}^{\infty}$, pode ser modelado como uma sequência de cadeias de Markov homogêneas:

$$\left\{X_n(T_0)\right\}_{n=1}^{\infty}, \left\{X_n(T_1)\right\}_{n=1}^{\infty}, \dots, \left\{X_n(T_m)\right\}_{n=1}^{\infty}, \dots$$

onde cada uma delas representa uma iteração do loop externo 3.

Associado a cada cadeia com T fixo, temos as probabilidades de transição num passo:

$$P_{ij}(T) = P[X_{k+1} = j / X_k = i] \quad (2)$$

ou seja a probabilidade de que, à temperatura T , o estado j seja selecionado ou produzido pelo algoritmo na $k+1$ -ésima iteração, dado que na k -ésima iteração o algoritmo se encontra no estado i . Essa probabilidade, para $j \in S(i)$ é dada por :

$$P_{ij}(T) = g_{ij}(T) \cdot a_{ij}(T) \quad (3)$$

onde $g_{ij}(T)$ é a probabilidade de que o algoritmo gere o estado j a partir do estado i , à temperatura T e $a_{ij}(T)$ dada por (1) é a probabilidade de aceitar o estado j a partir do estado i , à

temperatura T .

Se $j \notin S(i)$ e $j \neq i$ então $g_{ij}(T) = 0$. Logo $P_{ij}(T) = 0$.

A definição de $P_{ii}(T)$ será feita a modo de garantir:

$$P_{ii}(T) + \sum_{j \in S(i)} P_{ij}(T) = 1$$

É o que faremos a seguir.

Evidentemente pode existir algum $j_0 \in S(i)$ tal que $a_{ij_0}(T) < 1$

Nesse caso, $P_{ij_0}(T) = g_{ij_0}(T) \cdot a_{ij_0}(T) < g_{ij_0}(T)$

o que implica em

$$\sum_{j \in S(i)} P_{ij}(T) < \sum_{j \in S(i)} g_{ij}(T) = 1$$

$\sum_{j \in S(i)} P_{ij}(T)$ representa a probabilidade de que o algoritmo passe do estado i para algum estado vizinho, e se consideramos que sempre o algoritmo produz um novo estado (não necessariamente diferente do anterior), então

$$P_{ii}(T) = 1 - \sum_{j \in S(i)} P_{ij}(T) \quad (4)$$

representa a probabilidade de que o algoritmo permaneça no estado i .

De acordo com a definição dada em (4), se existe $j \in S(i)$ tal que $a_{ij}(T) < 1$ então $P_{ii}(T) > 0$. Caso contrário ($a_{ij}(T) = 1, \forall j \in S(i)$)

$$P_{ij}(T) = g_{ij}(T), \forall j \in S(i) \Rightarrow \sum_{j \in S(i)} P_{ij}(T) = \sum_{j \in S(i)} g_{ij}(T) = 1$$

$$\Rightarrow P_{ii}(T) = 0$$

Conclui-se que $\forall i, j \in S, \forall T > 0, P_{ij}(T) \geq 0$ (5)

$$\text{e } \forall i \in S, \forall T > 0, P_{ii}(T) + \sum_{j \in S(i)} P_{ij}(T) = 1 \quad (6)$$

A matriz $P(T) = \begin{bmatrix} P_{ij}(T) \end{bmatrix}$ de ordem $|S| \times |S|$ é a matriz de transições num passo e, devido a (5) e (6) ela é estocástica.

Logo, cada cadeia $\{X_n(T)\}_{n=1}^{\infty}$ tem associada uma matriz de transição num passo que é estocástica.

Para provar que a cadeia $\{X_n(T)\}_{n=1}^{\infty}$ é homogênea, ou seja que as probabilidades $P_{ij}(T)$ não dependem da iteração em que a transição é feita, consideremos que em uma iteração do loop

externo \mathfrak{Z} (que produz uma seqüência de estados representados por $\{X_n(T)\}$) as funções $g_{ij}(T)$ e $a_{ij}(T)$ dependem unicamente dos estados i e j (T é constante) e não dependem da iteração em que essa transição é feita. Assim, segundo (3) $P_{ij}(T)$ só depende de i e j , ou seja,

$$P_{ij}(T) = P[X_{k+1}=j / X_k=i] = P[X_{m+1}=j / X_m=i]$$

onde k e m representam diferentes iterações no loop interno $\mathfrak{Z}.1$. Isso justifica a notação $P_{ij}(T)$ independente de k em (2).

Em consequência, a cadeia $\{X_n(T)\}$ para T fixo é homogênea.

OBSERVAÇÃO

A evolução completa do algoritmo S.A. pode ser também pensada como um único processo estocástico, ou seja, pode ser considerada como um processo contínuo no sentido que ele passa seguidamente de uma cadeia para outra sem isolar o que acontece em cada uma delas. Pensando deste modo imaginamos que temos uma única seqüência de estados produzidos pelo algoritmo (como realmente acontece) e onde agora as iterações transcorrem infinitamente através dos diferentes valores de T . Neste caso o modelo é simplesmente $\{X_n\}$ e a dependência em T vem implicitamente expressa através de n .

Sendo assim, esta única cadeia agora é não-homogênea pois para o conjunto total de iterações do loop interno associadas a uma iteração do loop externo que considera um valor específico T_m de T , a função de aceitação dependerá de esse valor T_m , enquanto que para o conjunto de iterações do loop interno associadas à seguinte iteração do loop externo que considera um valor T_{m+1} de T , a função de aceitação dependerá agora de T_{m+1} . Isto faz com que a dependência que $P_{ij}(T)$ tem do parâmetro T , seja traduzida numa dependência na iteração em que uma transição é feita.

A seguir provaremos que a cadeia homogênea $\{X_n(T)\}$ para T fixo e associada a uma iteração do loop externo \mathfrak{Z} , é uma cadeia

de Markov. Lembremos que $P[X_{n+1}=j_{n+1} / X_n=j_n, X_{n-1}=j_{n-1}, \dots, X_0=j_0]$ representa a probabilidade de que na iteração $n+1$, o algoritmo (para T fixo) esteja no estado j_{n+1} , dado que são conhecidos o estado atual j_n e todos os outros estados anteriores a ele (nesse patamar de temperatura). Aqui, X_0 é a variável aleatória que representa o estado com que o algoritmo inicia um loop interno.

Nessa iteração, a fase de geração só considera a vizinhança do estado atual j_n , independentemente das vizinhanças dos outros estados. A função de aceitação $a_{ij}(T)$ considera nessa iteração, os estados j_n , j_{n-1} e o valor de T , sem que seja necessário levar em conta os outros estados anteriores produzido pelo algoritmo. Em consequência,

$$P[X_{n+1}=j_{n+1} / X_n=j_n, \dots, X_0=j_0] = P[X_{n+1}=j_{n+1} / X_n=j_n]$$

satisfazendo assim a propriedade Markoviana.

Resumindo, o algoritmo S. A. pode ser modelado matematicamente como uma sequência de cadeias de Markov homogêneas, ou como uma única cadeia de Markov não-homogênea de acordo com a observação anterior.

As probabilidades de transição em n passos, do estado i ao estado j , estão dadas por:

$$P_{ij}^n(T) = P[X_{k+n}=j / X_k=i]$$

Para T fixo, ou seja em cada cadeia de Markov homogênea, denotamos por $P^n(T)$ à matriz de transições em n passos, que corresponde à n -ésima potência de $P(T)$, a matriz de transições num passo, sendo válida a equação de Chapman-Kolmogorov

$$P_{ij}^n(T) = \sum_{k \in S} P_{ik}^r(T) \cdot P_{kj}^s(T) \quad (r+s = n)$$

assim como também desigualdades do tipo

$$P_{ij}^n(T) \geq P_{ik}^r(T) \cdot P_{kj}^s(T); \quad \forall k \in S$$

ou

$$P_{ij}^n(T) \geq P_{i,j_1}(T) \cdot P_{j_1,j_2}(T) \cdot \dots \cdot P_{j_{n-1},j}(T)$$

4.4 CONVERGÊNCIA DO ALGORITMO S.A.

Nosso propósito agora é provar que o algoritmo S.A. converge com probabilidade 1 para um ótimo global. Este será um resultado

teórico que implica infinitas iterações no algoritmo, ou seja se trata de uma convergência assintótica.

Como já foi mencionado, na prática o algoritmo será implementado adicionando critério de parada, tanto a nível de loop interno como externo. Com isso, passamos a ter um algoritmo finito e logicamente perdemos a convergência, mas conseguimos uma heurística que tem-se comportado muito bem para alguns problemas de otimização, como por exemplo o problema do caixeiro viajante (Kirkpatrick [36], Cerny [14], Randelman [48], Bonomi [11], Aarts [1], Aragon [4]), o problema da árvore de Steiner (Lundy [41]), o problema de alocação quadrático (Burkard [12], Connoly [17]), o problema de partição de grafos (Johnson [32], Kirkpatrick [36], Aragon [4]), o problema de coloração de grafos (Chams [15], Johnson [33], Aragon [4]), o problema da mochila com varias restrições (Drex1 [22]), o problema de localização multicapacitado (Pedrosa [46]), dentre outros.

O estudo da convergência será feito em duas etapas. Primeiro estudaremos a convergência para temperatura fixa. Aqui se prova que se a cadeia de Markov para T fixo é irredutível então ela tem uma distribuição de probabilidades estacionária dada por

$$\Pi(T) = (\Pi_1(T), \Pi_2(T), \dots, \Pi_S(T))$$

(ver observação 1 ao teorema 3.12) que pode ser chamada de "quase-estacionária" respeito da cadeia de Markov não-estacionária (Mitra [44]). Seguidamente se prova que existe $\lim_{T \rightarrow 0} \Pi(T) = \Pi(0) = (\Pi_1(0), \Pi_2(0), \dots, \Pi_S(0))$ que é um vetor (estocástico) ótimo no sentido de que $\Pi_i(0) > 0$ sse i é solução ótima global.

Se interpretamos $\Pi_i(0)$ como a probabilidade de que o algoritmo S.A. atinga a solução i (depois de infinitas iterações), então a probabilidade de que o algoritmo alcance uma solução ótima global, será dada por

$$\sum_{i \in S^*} \Pi_i(0) = \sum_{i \in S} \Pi_i(0) = 1$$

onde S^* representa o conjunto das soluções ótimas globais.

De acordo com teorema 3.11 aplicado a cadeias de Markov homogêneas para a existência da distribuição de probabilidades estacionária da cadeia, é suficiente que ela seja irredutível, aperiódica e recorrente. Lembremos que o espaço de estados é finito, garantindo assim a utilização de resultados baseados nessa propriedade.

4.4.1 IRREDUTIBILIDADE

A propriedade de irredutibilidade estabelece que dados $i, j \in S$ devem existir $m, n \geq 0$ tais que $P_{ij}^n(T) > 0$ e $P_{ji}^m(T) > 0$. Esta propriedade está estritamente ligada com as regras de geração e aceitação do algoritmo.

Como a função de aceitação $a_{ij}(T) > 0$ para cada $i, j \in S$ e para cada $T > 0$ então a responsabilidade da irredutibilidade corresponde à regra de geração.

Logo vamos considerar a seguinte

SUPosição:

começando num estado i qualquer, sempre é possível gerar qualquer outro estado j como uma sequência finita de passos, ou seja, existem estados j_1, j_2, \dots, j_k tais que $i = j_1$, $j = j_k$ e $g_{j_m j_{m+1}}(T) > 0$ para $m = 1, 2, \dots, k-1$

Com isto teremos:

$$\begin{aligned}
 P_{ij}^n(T) &\geq P_{j_1 j_2}(T) \cdot P_{j_2 j_3}(T) \cdot \dots \cdot P_{j_{k-1} j}(T) = \prod_{m=1}^{k-1} P_{j_m j_{m+1}} = \\
 &= \prod_{m=1}^{k-1} \left[g_{j_m j_{m+1}}(T) \cdot a_{j_m j_{m+1}}(T) \right] > 0
 \end{aligned}$$

Como o suposto anterior é válido para i e j quaisquer, então também teremos $P_{ji}^m(T) > 0$, para algum m inteiro positivo.

Assim, a irredutibilidade deve ser considerada só como premissa e terá então que ser verificada na aplicação do algoritmo, especificamente na definição da estrutura de vizinhança a ser usada.

4.4.2 APERIODICIDADE E RECORRÊNCIA

TEOREMA 4.1

Se a cadeia de Markov associada ao algoritmo S.A. para cada $T > 0$ é irreduzível, então a cadeia é também aperiódica e recorrente positiva, $\forall T > 0$.

PROVA

Seja i um ótimo global tal que existe $j \in S(i)$ satisfazendo $c(j) - c(i) = \Delta c_{ij} > 0$

Se não existesse tal i , todo ótimo global teria mesmo custo que todos seus vizinhos. Devido à hipótese de irreduzibilidade da cadeia, existe um caminho $i \rightarrow j_1 \rightarrow j_2 \rightarrow \dots \rightarrow j$ entre um ótimo global i e qualquer solução viável j , com probabilidades de transição estritamente positivas.

Como $j_1 \in S(i)$ então j_1 seria ótimo global; como $j_2 \in S(j_1)$ então j_2 seria ótimo global e assim sucessivamente, provando que j seria ótimo global. Logo toda solução viável seria ótimo global o que contradiz o suposto de que a função de custo c não é constante.

$$\therefore 0 < a_{ij}(T) = \exp(-\Delta c_{ij}/T) < 1 \quad ; \quad \forall T > 0$$

$$\Rightarrow P_{ij}(T) = g_{ij}(T) \cdot a_{ij}(T) < g_{ij}(T) \quad ; \quad \forall T > 0$$

$$\Rightarrow \sum_{j \in S(i)} P_{ij}(T) < \sum_{j \in S(i)} g_{ij}(T) = 1 \quad ; \quad \forall T > 0$$

$$\Rightarrow P_{ii}(T) = 1 - \sum_{j \in S(i)} P_{ij}(T) > 0 \quad ; \quad \forall T > 0$$

Logo o estado i tem período 1.

Então, por corolário 3.2, todos os estados tem período 1, sendo assim a cadeia aperiódica, $\forall T > 0$.

Por outro lado, como o espaço de estados S associado à cadeia é finito e a cadeia é irreduzível $\forall T > 0$, então de acordo com o corolário 3.4, a cadeia é recorrente, $\forall T > 0$.

Finalmente, usando corolário 3.6, temos que todos os estados são recorrente positivos, $\forall T > 0$.

■

4.4.3 PROPRIEDADES E CALCULO DOS VALORES DE $\Pi_i(T)$ COROLARIO 4.1

Se a cadeia de Markov associada ao algoritmo S.A. para cada $T > 0$ fixo é irredutível, então

$$(a) \lim_{n \rightarrow \infty} P_{ii}^{(n)}(T) = \Pi_i(T) \quad \text{existe,} \quad \forall T > 0 \quad e$$

$$\forall j \in S, \lim_{n \rightarrow \infty} P_{ji}^{(n)}(T) = \lim_{n \rightarrow \infty} P_{ii}^{(n)}(T) = \Pi_i(T); \quad \forall T > 0$$

ou seja que se o algoritmo começa no estado j , então $\Pi_i(T)$ é independente do estado inicial j escolhido pelo algoritmo

(b) Existem únicos $\Pi_i(T)$ satisfazendo o sistema :

$$\begin{array}{l} \Pi_i(T) > 0 \quad ; \quad \forall i \in S ; \quad \forall T > 0 \\ \sum_{i \in S} \Pi_i(T) = 1 \quad ; \quad \forall T > 0 \\ \Pi_j(T) = \sum_{i \in S} \Pi_i(T) \cdot P_{ij}(T) \quad ; \quad \forall j \in S ; \quad \forall T > 0 \end{array}$$

O vetor $\Pi(T) = (\Pi_1(T), \Pi_2(T), \dots, \Pi_S(T))$ é a distribuição de probabilidades estacionária (única) da cadeia, para $T > 0$ fixo.

PROVA

O resultado (a) é consequência do Teorema 3.11.

O resultado (b) é consequência do teorema 3.12 ■

OBSERVAÇÃO

O corolario 4.1 fornece varias formas de cálculo dos $\Pi_i(T)$. Uma (a) requer o cálculo das potências da matriz de transição $P(T)$ e a outra (b) equivale à solução de um sistema de equações de grande porte, além de precisar também da matriz $P(T)$. Considerando estas formas muito difíceis ou impraticáveis de implementar, buscaremos outros resultados para o cálculo dos $\Pi_i(T)$.

A existência de $\Pi_i(T) = \lim_{n \rightarrow \infty} P_{ii}^n(T)$ é considerada teóricamente como condição de equilíbrio no algoritmo S.A. ■

TEOREMA 4.2

Se a cadeia de Markov associada ao algoritmo S.A. para cada $T > 0$ é irredutível e $g_{ij}(T) = g_{ji}(T)$, $\forall i, j \in S$, $\forall T > 0$ então

$$\Pi_i(T) = \frac{\exp(-c(i)/T)}{\sum_{k \in S} \exp(-c(k)/T)}; \quad \forall i \in S, \quad \forall T > 0$$

PROVA

A existência e unicidade dos $\Pi_i(T)$ está garantida pelo corolário 4.1. Só será necessário verificar se os $\Pi_i(T)$ dados neste teorema satisfazem as propriedades dadas em (b) no corolário 4.1

(a) $\Pi_i(T) > 0$, $\forall T > 0$ pois $c(i) < \infty$, $\forall i \in S$ e $T > 0$.

$$(b) \sum_{i \in S} \Pi_i(T) = \frac{\sum_{i \in S} \exp(-c(i)/T)}{\sum_{k \in S} \exp(-c(k)/T)} = 1$$

(c) para provar que $\Pi_j(T) = \sum_{i \in S} \Pi_i(T) \cdot P_{ij}(T)$ será necessário antes provar que:

$$\exp(-c(i)/T) \cdot a_{ij}(T) = \exp(-c(j)/T) \cdot a_{ji}(T); \quad \forall i, j \in S$$

Efetivamente

- se $\Delta c_{ij} = c(j) - c(i) > 0$ então

$$\begin{aligned} \exp(-c(i)/T) \cdot a_{ij}(T) &= \exp(-c(i)/T) \cdot \exp(-[c(j) - c(i)]/T) \\ &= \exp(-c(j)/T) \\ &= \exp(-c(j)/T) \cdot a_{ji}(T) \end{aligned}$$

(pois $\Delta c_{ji} = c(i) - c(j) < 0 \Rightarrow a_{ji}(T) = 1$)

- se $\Delta c_{ij} < 0$ (que implica em $a_{ij}(T) = 1$) então

$$\begin{aligned} \exp(-c(i)/T) \cdot a_{ij}(T) &= \exp(-c(i)/T) \\ &= \exp(-c(i)/T) \cdot \exp(-c(j)/T) \cdot \exp(c(j)/T) \\ &= \exp(-c(j)/T) \cdot a_{ji}(T) \end{aligned}$$

(pois $\Delta c_{ji} > 0 \Rightarrow a_{ji}(T) = \exp(-[c(i) - c(j)]/T)$)

- se $\Delta c_{ij} = 0$ (que implica em $a_{ij}(T) = 1$ e $c(i) = c(j)$) então

$$\exp(-c(i)/T) \cdot a_{ij}(T) = \exp(-c(i)/T)$$

$$\begin{aligned}
 &= \exp(-c(j)/T) \\
 &= \exp(-c(j)/T) \cdot a_{ji}(T)
 \end{aligned}$$

Fica assim provado que $\forall i, j \in S$ e $\forall T > 0$
 $\exp(-c(i)/T) \cdot a_{ij}(T) = \exp(-c(j)/T) \cdot a_{ji}(T)$

Temos que $P_{ij}(T) = g_{ij}(T) \cdot a_{ij}(T)$; $\forall T > 0$, $\forall i \in S$, $\forall j \in S(i)$

$$\begin{aligned}
 \rightarrow \Pi_i(T) \cdot P_{ij}(T) &= \frac{\exp(-c(i)/T)}{\sum_{k \in S} \exp(-c(k)/T)} \cdot g_{ij}(T) \cdot a_{ij}(T) ; \forall T > 0 \\
 &= \frac{g_{ji}(T)}{\sum_{k \in S} \exp(-c(k)/T)} \cdot \exp(-c(i)/T) \cdot a_{ij}(T) \\
 &\quad \text{(devido à hipótese de simetria de } g) \\
 &= \frac{g_{ji}(T)}{\sum_{k \in S} \exp(-c(k)/T)} \cdot \exp(-c(j)/T) \cdot a_{ji}(T) \\
 &= \frac{\exp(-c(j)/T)}{\sum_{k \in S} \exp(-c(k)/T)} \cdot g_{ji}(T) \cdot a_{ji}(T) \\
 &= \Pi_j(T) \cdot P_{ji}(T)
 \end{aligned}$$

$$\therefore \Pi_i(T) \cdot P_{ij}(T) = \Pi_j(T) \cdot P_{ji}(T) ; \forall T > 0 , \forall i \in S , \forall j \in S(i)$$

A igualdade anterior também é válida para $j \in S(i)$ pois nesse caso $i \in S(j)$ e $P_{ij}(T) = P_{ji}(T) = 0$, $\forall T > 0$.

$$\begin{aligned}
 \therefore \sum_{i \in S} \Pi_i(T) \cdot P_{ij}(T) &= \sum_{i \in S} \Pi_j(T) \cdot P_{ji}(T) \\
 &= \Pi_j(T) \cdot \sum_{i \in S} P_{ji}(T) \\
 &= \Pi_j(T) \quad \text{pois a matriz } P \text{ é estocástica}
 \end{aligned}$$

OBSERVAÇÃO

A condição de simetria $g_{ij}(T) = g_{ji}(T)$; $\forall i, j \in S, \forall T > 0$ é uma restrição fraca que pode ser facilmente satisfeita em implementações do algoritmo S.A.

Por exemplo temos simetria se

(i) $\forall i \in S, |S(i)| = K$ (K inteiro positivo) e

(ii) $\forall i \in S, \forall j \in S(i), g_{ij}(T) = \frac{1}{|S(i)|} = \frac{1}{K}; \forall T > 0$

Neste caso g é uniforme, ou seja, a geração dentro da vizinhança é aleatória e uniformemente distribuída.

Devemos lembrar que supusemos no início que a estrutura de vizinhança é também simétrica, no sentido em que $i \in S(j)$ sse $j \in S(i)$. Sem isto, pode não verificar-se a condição de simetria para a função g ($j \in S(i) \Rightarrow g_{ij} = 1/K$ e $i \notin S(j) \Rightarrow g_{ji} = 0$, logo $g_{ij} \neq g_{ji}$). Temos então:

$$g_{ij}(T) = \frac{1}{|S(i)|} = \frac{1}{K} = \frac{1}{|S(j)|} = g_{ji}(T); \quad \forall T > 0, \forall i, j \in S$$

Resultado semelhante ao teorema 4.2 obtemos no teorema 4.3 quando trocamos a hipótese de simetria de g por outra condição como a de uniformidade, ou seja, quando supomos que

$$g_{ij}(T) = \frac{1}{|S(i)|} \quad \text{se } j \in S(i) \quad \text{e} \quad g_{ij}(T) = 0 \quad \text{se } j \notin S(i)$$

Aquí não necessariamente teremos $|S(i)| = K, \forall i \in S$, ou seja que não necessariamente todas as vizinhanças terão igual cardinalidade.

TEOREMA 4.3

Se a cadeia de Markov associada ao algoritmo S.A. para cada

$T > 0$ fixo é irredutível e $g_{ij}(T) = \frac{1}{|S(i)|}$ para $j \in S(i)$, então

$$\Pi_i(T) = \frac{|S(i)| \cdot \exp(-c(i)/T)}{\sum_{k \in S} |S(k)| \cdot \exp(-c(k)/T)}$$

PROVA

Seguindo os mesmos passos que na prova do teorema 4.2, temos:

a) $\Pi_i(T) > 0; \forall T > 0$

b) $\sum_{i \in S} \Pi_i(T) = 1$

c) $\exp(-c(i)/T) \cdot a_{ij}(T) = \exp(-c(j)/T) \cdot a_{ji}(T); \forall i, j \in S, \forall T > 0$

Logo $F_{ij}(T) = g_{ij}(T) \cdot a_{ij}(T)$

$$\begin{aligned}
&= \frac{1}{|S(i)|} \cdot a_{ij}(T) \quad \text{para cada } T > 0, i \in S \text{ e } j \in S(i) \\
\Rightarrow \Pi_i(T) \cdot P_{ij}(T) &= \frac{|S(i)| \cdot \exp(-c(i)/T)}{\sum_{k \in S} |S(k)| \cdot \exp(-c(k)/T)} \cdot \frac{1}{|S(i)|} \cdot a_{ij}(T) \\
&= \frac{1}{\sum_{k \in S} |S(k)| \cdot \exp(-c(k)/T)} \cdot \exp(-c(i)/T) \cdot a_{ij}(T) \\
&= \frac{1}{\sum_{k \in S} |S(k)| \cdot \exp(-c(k)/T)} \cdot \exp(-c(j)/T) \cdot a_{ji}(T) \\
&= \frac{\exp(-c(j)/T) \cdot |S(j)|}{\sum_{k \in S} |S(k)| \cdot \exp(-c(k)/T)} \cdot \frac{1}{|S(j)|} \cdot a_{ji}(T) \\
&= \Pi_j(T) \cdot P_{ji}(T); \quad \forall T > 0
\end{aligned}$$

De novo foi usada a simetria na estrutura de vizinhança, ou seja, $i \in S(j)$ sse $j \in S(i)$. Caso contrario ter-se-ia $j \in S(i)$ e $i \notin S(j)$ produzindo $P_{ij}(T) > 0$ e $P_{ji}(T) = 0$.

Logo provamos que $\Pi_i(T) \cdot P_{ij}(T) = \Pi_j(T) \cdot P_{ji}(T); \forall T > 0, \forall i, j \in S$, pois no caso $j \notin S(i)$ (que implica em $i \notin S(j)$) cada lado da igualdade é zero.

$$\text{Assim, } \sum_{i \in S} \Pi_i(T) \cdot P_{ij}(T) = \sum_{i \in S} \Pi_j(T) \cdot P_{ji}(T) = \Pi_j(T) \cdot \sum_{i \in S} P_{ji}(T) = \Pi_j(T)$$

■

OBSERVAÇÃO

No Teorema 4.3, se $|S(i)| = k$ (k inteiro positivo) para cada $i \in S$, obtemos o resultado do Teorema 4.2. ■

O resultado a seguir mostra as características de monotonicidade de $\Pi_j(T)$ com respeito à variável T .

De acordo com os resultados nos teoremas 4.2 e 4.3, vamos supor a seguinte estrutura para $\Pi_i(T)$:

$$\Pi_i(T) = \frac{h(i) \cdot \exp(-c(i)/T)}{\sum_{k \in S} h(k) \cdot \exp(-c(k)/T)} \quad (7)$$

Com isso, quando $h(i)=1 \forall i \in S$, temos $\Pi_i(T)$ como no teorema 4.2 e para $h(i)=|S(i)|$ temos $\Pi_i(T)$ como no teorema 4.3.

Podemos supor também que $h(i) > 0 \forall i \in S$ devido a que $\Pi_i(T) > 0, \forall i \in S$.

TEOREMA 4.4

Se a cadeia de Markov associada ao algoritmo S.A. para cada $T > 0$ é irredutível e $\Pi_i(T)$ é dado por (7) e S^* é o conjunto das soluções ótimas globais do problema de otimização combinatória, então:

(a) Para cada $i \in S^*$, $\Pi_i(T_{m+1}) - \Pi_i(T_m) > 0; \forall m \geq 0$

(b) Para cada $i \in S^*$, existe um único inteiro \hat{m}_i ($0 \leq \hat{m}_i < \infty$) tal

que $\Pi_i(T_{m+1}) - \Pi_i(T_m) > 0$ para $0 \leq m \leq \hat{m}_i - 1$

e $\Pi_i(T_{m+1}) - \Pi_i(T_m) < 0$ para $m \geq \hat{m}_i$

PROVA

Considerando $\Pi_i(T)$ como função contínua de T , temos

$$\frac{d}{dT} \Pi_i(T) = - \left[\frac{\pi_i(T)}{T} \right]^2 \cdot \frac{1}{h(i) \cdot \exp(-c(i)/T)}$$

$$\cdot \left[\sum_{j \in S} h(j) \cdot \{c(j) - c(i)\} \cdot \exp(-c(j)/T) \right]$$

ou equivalentemente

$$\begin{aligned} \left[\frac{h(i) \cdot T^2}{\pi_i^2(T)} \right] \cdot \frac{d}{dT} \Pi_i(T) &= - \sum_{j \in S} h(j) \cdot \{c(j) - c(i)\} \cdot \exp(-[c(j) - c(i)]/T) \\ &= \sum_{j/c(j) < c(i)} h(j) \cdot \{c(i) - c(j)\} \cdot \exp(-[c(j) - c(i)]/T) - \\ &\quad \sum_{j/c(j) > c(i)} h(j) \cdot \{c(j) - c(i)\} \cdot \exp(-[c(j) - c(i)]/T) \end{aligned}$$

Estamos interessados no estudo dos sinais de $\frac{d}{dT} \Pi_i(T)$ e por isso podemos desprezar o fator $\frac{h(i) \cdot T^2}{\pi_i^2(T)}$ que é estritamente positivo.

Denominando por $\Sigma_1(T)$ e $\Sigma_2(T)$ as duas somas anteriores, temos

que $\operatorname{sgn}\left(\frac{d}{dT}\Pi_i(T)\right) = \operatorname{sgn}\left(\Sigma_1(T) - \Sigma_2(T)\right)$ onde $\operatorname{sgn}(x) = \begin{cases} + & \text{se } x > 0 \\ - & \text{se } x < 0 \end{cases}$

a) Se $i \in S^*$ então não existe j tal que $c(j) < c(i)$. Logo $\Sigma_1(T) = 0$ e como $\Sigma_2(T)$ é positiva para cada $T > 0$ obtemos $\frac{d}{dT}\Pi_i(T) < 0$ $\forall T > 0$

Isto quer dizer que $\Pi_i(T)$ é monotonamente crescente com o decrescimento de T . Como $\{T_m\}$ é decrescente na medida que m aumenta obtemos $\Pi_i(T_{m+1}) - \Pi_i(T_m) > 0$; $\forall m \geq 0$

b) Se $i \notin S^*$ temos as seguintes situações:

Quando $T=0$, obtemos $\Sigma_2(0) = 0$ e $\Sigma_1(0) > 0$.

Logo $\Sigma_1(0) - \Sigma_2(0)$ é positiva.

Quando $T \rightarrow \infty$, $\Sigma_1(T)$ é decrescente e $\Sigma_2(T)$ é crescente.

Logo $\Sigma_1(T) - \Sigma_2(T)$ é decrescente com o incremento de T .

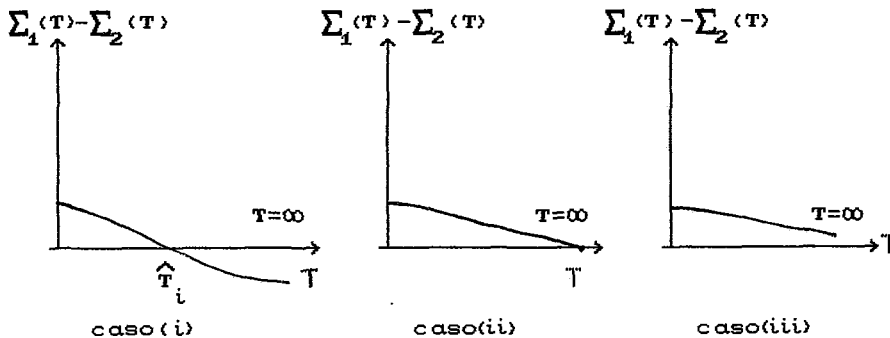
Quando $T = \infty$, $\Sigma_1(\infty) - \Sigma_2(\infty) = \sum_{j \in S} h(j) \cdot \{c(i) - c(j)\}$.

Neste caso podemos ter :

$$(i) \quad \Sigma_1(\infty) - \Sigma_2(\infty) < 0$$

$$(ii) \quad \Sigma_1(\infty) - \Sigma_2(\infty) = 0$$

$$(iii) \quad \Sigma_1(\infty) - \Sigma_2(\infty) > 0$$



(i) suponhamos que $i \notin S^*$ e que $\Sigma_1(\infty) - \Sigma_2(\infty) = \sum_{j \in S} h(j) \{c(i) - c(j)\} < 0$

ou equivalentemente $\sum_{j \in S} h(j) \cdot c(i) < \sum_{j \in S} h(j) \cdot c(j)$

$$\text{ou} \quad c(i) < \frac{\sum_{j \in S} h(j) \cdot c(j)}{\sum_{j \in S} h(j)} = C$$

onde C pode ser interpretado como custo medio ponderado.

Então a função $\frac{d}{dT}\Pi_i(T)$ deve ter um zero \hat{T}_i ($0 < \hat{T}_i < \infty$), ou seja $\frac{d}{dT}\Pi_i(\hat{T}_i) = 0$

$\frac{d}{dT}\Pi_i(T) > 0$ para $0 < T < \hat{T}_i$ e $\frac{d}{dT}\Pi_i(T) < 0$ para $\hat{T}_i < T < \infty$

Observamos que $\Sigma_1(T) - \Sigma_2(T)$ é continua com respeito a T . Logo para $i \in S^*$ e $c(i) < C$, devido a que $\{T_m\}$ é decrescente, podemos encontrar o primeiro (ou menor) \hat{m}_i tal que $T_{\hat{m}_i} = \hat{T}_i$.

Com isso,

para $0 \leq m \leq \hat{m}_i - 1$, temos $\frac{d}{dT}\Pi_i(T_m) < 0 \Rightarrow \Pi_i(T_{m+1}) - \Pi_i(T_m) > 0$

e para $m \geq \hat{m}_i$, temos $\frac{d}{dT}\Pi_i(T_m) > 0 \Rightarrow \Pi_i(T_{m+1}) - \Pi_i(T_m) < 0$

(ii) Seja agora $i \in S^*$ tal que $c(i) = C$, i.e., $\Sigma_1(\infty) - \Sigma_2(\infty) = 0$.

Neste caso, o único zero de $\Sigma_1(T) - \Sigma_2(T)$ é $T = \infty$, ou seja que $\frac{d}{dT}\Pi_i(\infty) = 0$.

Logo, $\forall 0 < T < \infty$ $\frac{d}{dT}\Pi_i(T) > 0 \Rightarrow \Pi_i(T_{m+1}) - \Pi_i(T_m) < 0, \forall m \geq 0$

Neste caso, $\hat{m}_i = 0$.

(iii) Seja finalmente $i \in S^*$ tal que $c(i) > C$, i.e., $\Sigma_1(\infty) - \Sigma_2(\infty) > 0$.

Aquí $\frac{d}{dT}\Pi_i(T) > 0, \forall 0 < T < \infty$ e então $\Pi_i(T_{m+1}) - \Pi_i(T_m) < 0, \forall m \geq 0$

Também resulta $\hat{m}_i = 0$. ■

COROLARIO 4.2

Sob as mesmas hipóteses do Teorema 4.4, existe $\bar{m} < \infty$ tal que para cada $i \in S^*$, $\Pi_i(T_{m+1}) - \Pi_i(T_m) < 0$ se $m \geq \bar{m}$.

PROVA

É só escolher $\bar{m} = \max\{\hat{m}_i / i \in S^*\}$

Assim, $\forall i \in S^*$, $\exists \hat{m}_i$ tal que $m \geq \hat{m}_i \Rightarrow \Pi_i(T_{m+1}) - \Pi_i(T_m) < 0$

\rightarrow para $m \geq \bar{m}$, $\Pi_i(T_{m+1}) - \Pi_i(T_m) < 0, \forall i \in S^*$. ■

OBSERVAÇÃO

O resultado do Teorema 4.4 estabelece que para $i \in S^*$, a função $\Pi_i(T)$ é crescente quando $T \rightarrow 0$. Para $i \in S^*$, a partir de um certo instante \bar{m} , $\Pi_i(T)$ é decrescente quando $T \rightarrow 0$. ■

As seguintes perguntas resultam naturais:

Será que $\lim_{T \rightarrow 0} \Pi_i(T) = 0$ para $i \in S^*$?

Será que $\lim_{T \rightarrow 0} \Pi_i(T)$ existe para $i \in S^*$?

Os seguintes resultados apontam justamente nessa direção.

TEOREMA 4.5

Sob as mesmas hipóteses do Teorema 4.4,

$$\lim_{T \rightarrow 0} \Pi_i(T) = \begin{cases} \frac{h(i)}{\sum_{j \in S^*} h(j)} & \text{se } i \in S^* \\ 0 & \text{se } i \notin S^* \end{cases}$$

onde $h(i) = 1$ ou $|S(i)|$

PROVA

Seja $i \in S^*$.

$$\Pi_i(T) = \frac{h(i) \cdot \exp(-c(i)/T)}{\sum_{j \in S} h(j) \cdot \exp(-c(j)/T)}$$

$$= \frac{h(i) \cdot \exp(-c(i)/T)}{\sum_{j \in S^*} h(j) \cdot \exp(-c(j)/T) + \sum_{j \notin S^*} h(j) \cdot \exp(-c(j)/T)}$$

$$= \frac{h(i)}{\sum_{j \in S^*} h(j) \cdot \exp(-[c(j)-c(i)]/T) + \sum_{j \notin S^*} h(j) \cdot \exp(-[c(j)-c(i)]/T)} \quad (8)$$

$$= \frac{h(i)}{\sum_{j \in S^*} h(j) + \sum_{j \notin S^*} h(j) \cdot \exp(-[c(j)-c(i)]/T)}$$

pois $i, j \in S^* \Rightarrow c(i) = c(j)$

Agora, $\lim_{T \rightarrow 0} \exp(-[c(j)-c(i)]/T) = 0$ para $j \notin S^*$ pois $c(j) > c(i)$

$$\text{Assim, } \lim_{T \rightarrow 0} \Pi_i(T) = \frac{h(i)}{\sum_{j \in S^*} h(j)}$$

Seja agora $i \notin S^*$.

Na expressão (8), para $j \in S^*$ temos $c(j) - c(i) < 0$, o que implica em $\exp(-[c(j)-c(i)]/T) \rightarrow \infty$ se $T \rightarrow 0$; e para $j \notin S^*$, $c(j) - c(i)$ pode

ser positivo, negativo ou zero e então $\exp(-[c(j)-c(i)]/T)$ tem limite 0, ∞ ou 1, respectivamente quando $T \rightarrow 0$.

Logo $\lim_{T \rightarrow 0} \Pi_i(T) = 0$. ■

COROLARIO 4.3

Se a cadeia de Markov associada ao algoritmo S.A. para cada $T > 0$ fixo é irreduzível e

(a) $g_{ij}(T) = g_{ji}(T), \forall T > 0, \forall i, j \in S$ então

$$\lim_{T \rightarrow 0} \Pi_i(T) = \begin{cases} \frac{1}{|S^*|} & \text{se } i \in S^* \\ 0 & \text{se } i \notin S^* \end{cases}$$

(b) $g_{ij}(T) = \frac{1}{|S(i)|}, \forall j \in S(i)$ então

$$\lim_{T \rightarrow 0} \Pi_i(T) = \begin{cases} \frac{|S(i)|}{\sum_{j \in S^*} |S(j)|} & \text{se } i \in S^* \\ 0 & \text{se } i \notin S^* \end{cases}$$

PROVA

No caso (a), de acordo ao Teorema 4.2, temos $h(i)=1; \forall i \in S$ e no caso (b), de acordo ao Teorema 4.3, temos $h(i)=|S(i)|, \forall i \in S$. ■

OBSERVAÇÃO

O limite $\lim_{T \rightarrow 0} \Pi_i(T), \forall i \in S$ que denotaremos por $\Pi_i(0)$, pode ser interpretado como a probabilidade de que o algoritmo S.A. alcance o estado i depois de infinitas iterações no loop externo

Logo, a probabilidade de que o algoritmo alcance um ótimo global é dada por:

$$\sum_{i \in S^*} \Pi_i(0) = \sum_{i \in S^*} \lim_{T \rightarrow 0} \Pi_i(T) = \lim_{T \rightarrow 0} \sum_{i \in S^*} \Pi_i(T) = \lim_{T \rightarrow 0} \sum_{i \in S^*} \frac{h(i)}{\sum_{j \in S^*} h(j)} = 1$$

■

4.5 RESUMO DO CAPITULO

Neste capítulo foi apresentado o algoritmo Simulated Annealing sob uma forma suficientemente geral de modo que possa ser aplicado a uma grande variedade de problemas de otimização combinatória.

O algoritmo é modelado matematicamente como uma sequência infinita de cadeias de Markov $\{X_n(T)\}$, cada uma delas correspondendo a uma iteração do loop interno.

São apresentados os resultados que garantem a convergência do algoritmo a um ótimo global, sob as seguintes condições:

- (a) cada cadeia de Markov $\{X_n(T)\}$ para T fixo, tem o mesmo espaço de estados finito S .
- (b) simetria de vizinhança, ou seja, $i \in S(j)$ sse $j \in S(i)$
- (c) simetria ou uniformidade na função de geração g
- (d) irredutibilidade de $\{X_n(T)\}$, para cada $T > 0$ fixo.

Mas essa convergência é assintótica (e portanto teórica), pelo fato de precisar infinitas iterações no loop interno, necessarias para garantir a existência da distribuição de probabilidades estacionária $\Pi(T)$, $\forall T > 0$ e de fazer decrescer a sequência $\{T_m\}$ infinitamente a zero, sendo o algoritmo assim concebido não diretamente implementável.

No capítulo seguinte será aplicado o algoritmo S.A. ao problema de localização capacitado, na forma de heurística ou algoritmo aproximado, ou seja, com a incorporação de criterios de parada tanto para o loop interno como externo, perdendo deste modo a propriedade de convergência (assintótica) a um ótimo global com probabilidade 1. Nesta aplicação serão estudadas diferentes alternativas com respeito a:

- determinação de uma temperatura inicial T_0
- determinação do esquema de esfriamento $\{T_m\}$
- determinação de uma temperatura final T_f que será usada como criterio de parada do algoritmo
- determinação de um número finito apropriado de iterações do loop interno.

O objetivo consiste em encontrar uma heurística para o problema de localização capacitado, cuja performance medida em termos da relação entre a qualidade da solução obtida e o tempo computacional consumido por ela, seja satisfatória.

C A P I T U L O 5

APLICAÇÃO DO ALGORITMO SIMULATED ANNEALING AO PROBLEMA DE LOCALIZAÇÃO CAPACITADO (CWLP)

5.1 O MODELO MATEMÁTICO

Consideremos a seguinte formulação do CWLP :

$$(P1) \left\{ \begin{array}{l} \text{minimizar } \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} + \sum_{i \in I} f_i y_i \\ \text{s.a. } \sum_{j \in J} x_{ij} \leq a_i y_i \quad ; \quad \forall i \in I = \{1, 2, \dots, m\} \quad (1) \\ \sum_{i \in I} x_{ij} = b_j \quad ; \quad \forall j \in J = \{1, 2, \dots, n\} \quad (2) \\ x_{ij} \geq 0 \quad ; \quad \forall i \in I, \forall j \in J \quad (3) \\ y_i \in \{0, 1\} \quad ; \quad \forall i \in I \quad (4) \end{array} \right.$$

Uma restrição do tipo $\sum_{i \in I} a_i y_i \geq \sum_{j \in J} b_j$ (5)

resulta redundante com (1) e (2), porém pode ser utilizada numa outra formulação do CWLP.

Para cada $y \in \{0, 1\}^m$, $y = (y_1, y_2, \dots, y_m)$, sejam

$$f(y) = \sum_{i \in I} f_i y_i \quad \text{o custo fixo total associado com } y, \text{ e}$$

$$T(y) = \left\{ \begin{array}{l} \text{minimizar } \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} \\ \text{s.a. } \quad (1), (2) \text{ e } (3) \end{array} \right.$$

o problema de transporte associado com y , de valor ótimo $W(y)$.

Para garantir a existência de $W(y)$, devemos ter certeza que o espaço de soluções viáveis de $T(y)$ seja não vazio. Por exemplo se $y = (0, 0, \dots, 0)$ então $W(y)$ não existe pois não temos $x = (x_{ij})$ satisfazendo (1), (2) e (3).

Porém, se $y \in \{0, 1\}^m$ é tal que satisfaz (5), então fica garantida a existência de pelo menos uma solução viável para $T(y)$ e logo a existência de $W(y)$.

Consideremos o seguinte problema:

$$(P2) \left\{ \begin{array}{l} \text{minimizar } Z(y) = f(y) + W(y) \\ \text{s.a. } y_i \in \{0,1\} \quad \forall i \in I \quad (4) \\ \sum_{i \in I} a_i y_i \geq \sum_{j \in J} b_j \quad (5) \end{array} \right.$$

Os problemas (P1) e (P2) são equivalentes no sentido que eles tem o mesmo conjunto de soluções ótimas.

É bom observar que o espaço viável de (P2) não faz referência explícita de $x=(x_{ij})$, sendo assim de cardinalidade menor do que o espaço viável de (P1). Mas, é claro que toda solução viável y de (P2) tem associado um vetor x , aquele obtido ao resolver o problema de transporte $T(y)$, com valor ótimo $W(y)$.

Logo, o cálculo da função objetivo $Z(y)$ do problema (P2) (que envolve a resolução de um problema de transporte) tem custo bem maior do que o cálculo da função objetivo de (P1).

Com tudo, vamos trabalhar com a formulação (P2) de CWLP, aproveitando a simplicidade e a menor cardinalidade do seu espaço viável, que será representado por S , ou seja,

$$S = \left\{ y \in \{0,1\}^m / \sum_{i \in I} a_i y_i \geq \sum_{j \in J} b_j \right\}$$

Após a aplicação dos testes de redução ao CWLP, obteremos o conjunto de facilidades I particionado em três conjuntos:

$$K_1 = \{i \in I / y_i = 1 \text{ em toda solução ótima}\}$$

$$K_0 = \{i \in I / y_i = 0 \text{ em toda solução ótima}\}$$

$$\text{e } K_2 = I - K_1 - K_0$$

As variáveis y_i tais que $i \in K_1 \cup K_0$ são variáveis fixadas a priori e então podemos considerar o espaço S como sendo:

$$S = \left\{ y \in \{0,1\}^m / \sum_{i \in I} a_i y_i \geq \sum_{j \in J} b_j, y_i = 1 \quad \forall i \in K_1, y_i = 0 \quad \forall i \in K_0 \right\}$$

5.2 ESTRUTURA DE VIZINHANÇA

Consideremos a seguinte regra de geração (ou transformação) de uma solução viável a partir de $y=(y_1, y_2, \dots, y_m) \in S$

Passo 1 Escolher aleatoriamente $i \in K_2$

Passo 2 Fazer $y'_k = y_k, \forall k \neq i$ e $y'_i = 1 - y_i$

Passo 3 Se $y' = (y'_1, y'_2, \dots, y'_m) \in S$ então definimos y' como solução viável vizinha de y . Caso contrário, fazemos $y' = y$ e repetimos passo 1.

O conjunto das potenciais soluções vizinhas de y é caracterizado por $S(y)$, ou seja,

$$S(y) = \{y' / y' \text{ pode ser obtida a partir de } y \text{ mediante a regra anterior}\}$$

Verificam-se as seguintes propriedades:

- $|S(y)| \leq |K_2|; \forall y \in S$
- $y' \in S(y) \text{ sse } y \in S(y')$, ou seja, temos simetria de vizinhança
- se $g_{y,y'}$ é a probabilidade de que a solução viável y' seja gerada a partir de y , então

$$g_{y,y'} = \begin{cases} |S(y)|^{-1} & \text{se } y' \in S(y) \\ 0 & \text{se } y' \notin S(y) \end{cases}$$

ou seja, temos geração uniforme e independente do parâmetro de temperatura T .

5.3 IRREDUTIBILIDADE

Esta propriedade se traduz no fato de poder atingir qualquer solução viável y^* a partir de outra solução viável y qualquer. Como foi estabelecido no capítulo anterior, esta propriedade está estritamente ligada com as regras de geração e aceitação do algoritmo S. A. Sendo que a aceitação é uma função de probabilidades estritamente positiva, a irredutibilidade se traduz em que y^* possa ser gerada a partir de y num número finito de passos, ou seja, que exista probabilidade positiva de gerar y^* a partir de y , num número finito de passos.

Dados $y, y^* \in S$, elas diferem no máximo em $|K_2|$ componentes. Logo, mediante uma sequência finita de passos, cada um deles mudando apropriadamente o valor de uma variável y_i de 1 para 0 ou de 0 para 1, podemos transformar y em y^* . Usando as mudanças inversas, transformamos y^* em y . Cada mudança corresponde à

escolha de uma solução vizinha da atual e portanto acontece com probabilidade positiva. Assim, a geração de y^* a partir de y acontece também com probabilidade positiva, ficando deste modo garantida a irredutibilidade.

Com as definições e propriedades dadas até este ponto, se satisfazem as condições de convergência global do algoritmo S.A. desde que sejam feitas um número infinito de iterações. Como já foi estabelecido, a heurística que obteremos a partir do algoritmo S.A. limita a quantidade de iterações a um número finito, como acontece com todas as aplicações deste algoritmo aos diferentes problemas de otimização.

5.4 CRITERIO DE ACEITAÇÃO

Como é sabido, na aceitação de uma solução viável $y' \in S(y)$ é considerada a função:

$$a_{y,y'}(T) = \begin{cases} \exp(-\Delta z/T) & \text{se } \Delta z > 0 \\ 1 & \text{se } \Delta z \leq 0 \end{cases}$$

onde $\Delta z = z(y') - z(y)$ é a variação do custo total ao passar de y para y' .

A seguir vamos introduzir alguma notação para representar Δz de forma conveniente. Associado a cada $y \in S$ temos:

$$K_1(y) = \{i \in K_2 / y_i = 1\} \quad \text{e} \quad K_0(y) = \{i \in K_2 / y_i = 0\}$$

Evidentemente, $K_0(y) \cup K_1(y) = K_2$; $\forall y \in S$

Com isso, podemos escrever $W(y) = W[K_1 \cup K_1(y)]$, como sendo o valor ótimo do problema de transporte com conjunto de índices oferta $K_1 \cup K_1(y)$.

Seja $y' \in S(y)$ tal que $y'_i = 1 - y_i$ e $y'_k = y_k$, $\forall k \neq i$

(y e y' só diferem na componente i -ésima)

Se $y_i = 1$ então $y'_i = 0 \Rightarrow K_1(y') = K_1(y) - \{i\}$ e $K_0(y') = K_0(y) \cup \{i\}$

Se $y_i = 0$ então $y'_i = 1 \Rightarrow K_1(y') = K_1(y) \cup \{i\}$ e $K_0(y') = K_0(y) - \{i\}$

Assim,

$$z(y) = f(y) + W(y) = \sum_{l \in K_1 \cup K_1(y)} f_l + W(y) = \sum_{l \in K_1 \cup K_1(y)} f_l + W[K_1 \cup K_1(y)] \quad \text{e}$$

$$z(y') = f(y') + W(y') = \sum_{l \in K_1 \cup K_1(y')} f_l + W(y') = \sum_{l \in K_1 \cup K_1(y')} f_l + W[K_1 + K_1(y')]]$$

$$\text{Logo, } \Delta z = z(y') - z(y) = \sum_{l \in K_1 \cup K_1(y')} f_l + W(y') - \sum_{l \in K_1 \cup K_1(y)} f_l - W(y)$$

$$= \begin{cases} W(y') - W(y) + f_i & \text{se } y_i = 0 \quad (\text{ou se } y'_i = 1) \\ W(y') - W(y) - f_i & \text{se } y_i = 1 \quad (\text{ou se } y'_i = 0) \end{cases}$$

$$= \begin{cases} W[K_1 + K_1(y')] - W[K_1 + K_1(y)] + f_i & \text{se } y_i = 0 \\ W[K_1 + K_1(y')] - W[K_1 + K_1(y)] - f_i & \text{se } y_i = 1 \end{cases}$$

$$= \begin{cases} W[K_1 + K_1(y) + i] - W[K_1 + K_1(y)] + f_i & \text{se } y_i = 0 \\ W[K_1 + K_1(y) - i] - W[K_1 + K_1(y)] - f_i & \text{se } y_i = 1 \end{cases}$$

$$= \begin{cases} f_i - \Omega(y, i) & \text{se } y_i = 0 \\ \Delta(y, i) - f_i & \text{se } y_i = 1 \end{cases}$$

onde $\Omega(y, i) = W[K_1 + K_1(y)] - W[K_1 + K_1(y) + i]$ e
 $\Delta(y, i) = W[K_1 + K_1(y) - i] - W[K_1 + K_1(y)]$

$$\text{Logo, } \Delta z = z(y') - z(y) = \begin{cases} f_i - \Omega(y, i) & \text{se } y_i = 0 \\ \Delta(y, i) - f_i & \text{se } y_i = 1 \end{cases}$$

Se y é a solução atual dada pelo algoritmo S.A. e y' é a solução gerada na iteração seguinte, o critério de aceitação estabelece que y' é aceita com probabilidade $a_{y,y'}(T)$, ou seja,

$$y' \text{ é aceita com probabilidade } \begin{cases} \exp\{-\Delta z/T\} & \text{se } \Delta z > 0 \\ 1 & \text{se } \Delta z \leq 0 \end{cases}$$

A primeira parte deste critério ($\Delta z > 0$) é implementada usando um número x uniformemente distribuído em $(0, 1)$, na seguinte forma:

Se $x < \exp\{-\Delta z/T\}$ então y' será aceita

A seguir veremos como fica este critério no Problema de Localização Capacitado. Separamos este estudo em dois casos:

(a) $y_i = 0$ (e logo $y'_i = 1$) : a i -ésima variável muda de 0 para 1

(b) $y_i=1$ (e logo $y'_i=0$) : a i -ésima variável muda de 1 para 0

(a) Neste caso,

$$a_{y,y'}(T) = \begin{cases} E = \exp\left\{-\frac{f_i - \Omega(y,i)}{T}\right\} & \text{se } \Delta z = f_i - \Omega(y,i) > 0 \\ 1 & \text{se } \Delta z = f_i - \Omega(y,i) \leq 0 \end{cases}$$

Logo, $y' \in S(y)$ será aceita com probabilidade $\begin{cases} E & \text{se } f_i > \Omega(y,i) \\ 1 & \text{se } f_i \leq \Omega(y,i) \end{cases}$

$$\begin{aligned} \text{Aqui, } x < \exp\{-\Delta z/T\} &\iff x < E = \exp\left\{-\frac{f_i - \Omega(y,i)}{T}\right\} \\ &\iff \Omega(y,i) > f_i + T \cdot \ln x \end{aligned}$$

Definindo $t^* = f_i + T \cdot \ln x$ conclui-se que

y' será aceito com probabilidade E se $t^* < \Omega(y,i) < f_i$

Assim, num caso a aceitação de $y' \in S(y)$ acontece quando $\Omega(y,i) \geq f_i$ (com probabilidade 1) e no outro caso, quando $t^* < \Omega(y,i) < f_i$ (com probabilidade E).

Em consequência, y' será aceito se $\Omega(y,i) > t^*$.

No caso em que $\Omega(y,i) \leq t^*$, ou equivalentemente $x \geq E$ teremos que rejeitar y' como nova solução viável.

Resumindo, o criterio de aceitação fica assim:

dado x uniformemente distribuído em $(0,1)$,

se $\Omega(y,i) > t^* = f_i + T \cdot \ln x$ então y' será aceita

se $\Omega(y,i) \leq t^* = f_i + T \cdot \ln x$ então y' será rejeitada

(b) No caso em que $y_i=1$ temos

$$a_{y,y'}(T) = \begin{cases} \bar{E} = \exp\left\{-\frac{\Delta(y,i) - f_i}{T}\right\} & \text{se } \Delta(y,i) - f_i > 0 \\ 1 & \text{se } \Delta(y,i) - f_i \leq 0 \end{cases}$$

Então, $y' \in S(y)$ será aceito com probabilidade $\begin{cases} \bar{E} & \text{se } \Delta(y,i) > f_i \\ 1 & \text{se } \Delta(y,i) \leq f_i \end{cases}$

$$\begin{aligned} \text{Agora } x < \exp\{-\Delta z/T\} &\iff x < \bar{E} = \exp\left\{-\frac{\Delta(y,i) - f_i}{T}\right\} \\ &\iff \Delta(y,i) < f_i - T \cdot \ln x \end{aligned}$$

Definindo $c^* = f_i - T \cdot \ln x$ conclui-se que

y' será aceito com probabilidade \bar{E} se $f_i < \Delta(y,i) < c^*$

Assim, num caso a aceitação de $y' \in S(y)$ acontece quando $\Delta(y,i) \leq f_i$ (com probabilidade 1) e no outro caso, quando $f_i < \Delta(y,i) < c^*$ (com probabilidade \bar{E}).

Em consequência, $y' \in S(y)$ será aceito se $\Delta(y,i) < c^*$.

Observamos também que $\Delta(y,i) \geq c^*$ é equivalente com $x \geq \bar{E}$ que implica a rejeição de y' .

Resumindo, o criterio de aceitação fica assim:

dado x uniformemente distribuído em $(0,1)$,

se $\Delta(y,i) < c^* = f_i - T \cdot \ln x$ então y' será aceita

se $\Delta(y,i) \geq c^* = f_i - T \cdot \ln x$ então y' será rejeitada

Tanto no caso (a) como (b), a rejeição de $y' \in S(y)$ significa que na seguinte iteração será mantida a solução viável y .

A seguir apresentamos o algoritmo Simulated Annealing sob o mesmo formato do aquele do capítulo anterior, introduzindo desta vez as modificações próprias de sua aplicação ao Problema de Localização Capacitado (CWLP), principalmente no que respeita ao loop interno 3.1.

5.5 ALGORITMO SIMULATED ANNEALING APLICADO AO CWLP.

1.- Obter uma solução viável inicial.

{ Podemos considerar $y = (y_1, y_2, \dots, y_m)$ tal que $y_i = 1, \forall i \in K_1 \cup K_2$
e $y_i = 0, \forall i \in K_0$, apos a aplicação do teste de redução }

2.- Obter uma temperatura inicial $T = T_0 > 0$.

{ Um estudo analítico para determinar T_0 será feito depois }

3.- Enquanto não for atingida uma condição de parada, fazer:

{ O criterio de parada será apresentado na fase de implementação }

3.1 Enquanto não for atingida uma condição de equilibrio,
fazer:

{ A condição de equilíbrio será apresentada na fase de implementação }

3.1.1 Escolher uma solução viável vizinha à atual

{ supondo a solução atual $y=(y_1, y_2, \dots, y_m)$ e $K_1^*=\{i \in K_2 / y_i=1\}$, bastará escolher aleatoriamente $i \in K_2$ como sendo o índice da variável que mudará seu valor de 1 para 0 ou de 0 para 1. No primeiro caso, se a solução obtida for inviável, repetimos este passo. }

3.1.2 Gerar um número aleatório x uniformemente distribuído no intervalo $(0,1)$.

3.1.3 Se $y_i=1$ (ou $i \in K_1^*$) então fazer:

3.1.3.1 Calcular $d = W[K_1 \cup K_1^* - i] - W[K_1 \cup K_1^*] = \Delta(K_1^*, i)$

3.1.3.2 Se $d < f_i - T \cdot \ln x$ então fazer $K_1^* := K_1^* - \{i\}$

{ A solução vizinha é aceita. No caso contrario é só manter $K_1^* := K_1^*$ o que se traduz em manter a solução atual como sendo a nova solução. }

3.1.4 Se $y_i=0$ (ou $i \notin K_1^*$) então fazer:

3.1.4.1 Calcular $d = W[K_1 \cup K_1^*] - W[K_1 \cup K_1^* + i] = \Omega(K_1^*, i)$

3.1.4.2 Se $d > f_i + T \cdot \ln x$ então fazer $K_1^* := K_1^* + \{i\}$

{ A solução vizinha é aceita. No caso contrario é só manter $K_1^* := K_1^*$ o que se traduz em manter a solução atual como sendo a nova solução. }

3.2 Reduzir a temperatura

{ O esquema de redução de temperatura será estudado na fase de implementação }

4.- A solução é dada pelo conjunto $K_1 \cup K_1^*$.

{ O conjunto $K_1 \cup K_1^*$ define os armazens considerados abertos em cada solução. Correspondentemente temos a solução viável y tal que $y_i=1$ sse $i \in K_1 \cup K_1^*$. O valor da função objetivo associado com y é dada por $z(y) = \sum_{i \in K_1 \cup K_1^*} f_i + W(y)$ }

Para fazer este algoritmo implementável, é necessário especificar:

- a temperatura inicial (passo 2)
- a condição de parada (passo 3)
- a condição de equilíbrio (passo 3.1)
- o esquema de redução de temperatura (passo 3.2).

E o que faremos a seguir, lembrando que após estas especificações, o algoritmo ou heurística obtida perde sua característica de convergência assintótica a um ótimo global.

5.6 A TEMPERATURA INICIAL.

A ideia é encontrar uma temperatura T_0 suficientemente grande de modo que iniciando o algoritmo nesse nível de temperatura sejam aceitas todas ou quase todas as soluções geradas, tanto se melhoram como se deterioram o valor da função de custo.

Existem basicamente tres enfoques para lidar com esta situação:

- (a) método de ensaio-erro
- (b) utilizar uma amostra de valores Δz (diferenças de custo) no inicio do algoritmo
- (c) estabelecer analiticamente um limite superior para Δz .

Em (a) é sugerido iniciar o algoritmo com uma temperatura arbitrária T e realizar certa quantidade de iterações do algoritmo nesse nível de temperatura, para determinar a porcentagem p de soluções que são aceitas. Se $p < 0.8$ (segundo a proposta de Kirkpatrick [36]), a temperatura é duplicada e repetimos este procedimento. Se $p \geq 0.8$, inicia-se o algoritmo com esse nível de temperatura como sendo a temperatura inicial.

Em (b) é sugerido gerar m_0 soluções viáveis no inicio do algoritmo e aceitar todas elas, tanto as que melhoram como as que pioram o valor da função de custo. Calcular em cada caso, a diferença no custo (Δz) entre a solução atual e a seguinte, e utilizar os Δz na determinação de T_0 . Connolly [17], propoe usar δ_{\min} e δ_{\max} , o mínimo e o máximo dos Δz encontrados e calcular T_0 como $T_0 = \delta_{\min} + \{ \delta_{\max} - \delta_{\min} \} / 10$ que corresponde ao 10% inferior da faixa entre δ_{\min} e δ_{\max} .

Aartz [1] utiliza também o enfoque (b) e para a determinação de T_0 , procede como segue:

Sejam

m_1 = número de soluções geradas que melhoram custo ($\Delta z \leq 0$)

m_2 = número de soluções geradas que deterioram custo ($\Delta z > 0$)

$\bar{\Delta z}_+$ = media dos Δz positivos.

Numa dada temperatura T , a taxa de aceitação t_{ac} , definida pela razão entre o número de soluções aceitas e o número total de soluções geradas numa iteração completa do loop interno 3.1 do algoritmo S.A., é aproximada mediante a expressão:

$$t_{ac} \approx \frac{m_1 + m_2 \cdot \exp\{-\bar{\Delta z}_+ / T\}}{m_1 + m_2}$$

$\{ m_1 + m_2 = m_0 = \text{número total de soluções geradas} \}$

$$\text{Resulta então } T = T_0 = \frac{\bar{\Delta z}_+}{\ln \left[\frac{m_2}{m_2 \cdot t_{ac} - m_1 (1 - t_{ac})} \right]} \quad (6)$$

Assim, fixado t_{ac} num valor desejado (evidentemente próximo de 1) e gerando m_0 soluções viáveis no início do algoritmo, a temperatura inicial T_0 é calculada usando (6).

Em Johnson [32], [33], é usado um parâmetro INITPROB para a determinação de T_0 , de modo tal que nesse nível de temperatura, a proporção de soluções aceitas seja aproximadamente INITPROB. Não é explicada a forma de obter T_0 para um valor dado de INITPROB, mas podemos usar a técnica de Aartz [1] fazendo $\text{INITPROB} = t_{ac}$ e experimentalmente estudar o comportamento do algoritmo para diferentes valores de t_{ac} , como é feito pelos autores no problema de partição de grafos.

Em (c) a ideia é encontrar um limite superior U para qualquer diferença Δz no custo e usar $T_0 \gg U$ (T_0 bem maior do que U). Assim, $\exp\{-\Delta z / T_0\} \approx 1$ e portanto serão aceitas praticamente todas as soluções viáveis geradas pelo algoritmo, nesse nível de temperatura.

Lundy [41], [42] propõe usar $U \geq \max \{ \phi_y / y \in S \}$, onde $\phi_y = z(y) - z(y^*)$ e y^* é um ótimo global do problema.

Mitra [44] sugere usar $T_0 = \frac{\gamma}{\log(1+k_0)}$, onde k_0 é qualquer parâmetro que satisfaça $1 \leq k_0 < \infty$, e γ depende das características da função de custo e da cadeia de Markov associada ao processo Simulated Annealing. Segundo é definido, γ deverá satisfazer :

$$\gamma > L = \max_{y \in S} \max_{y' \in S(y)} |z(y') - z(y)| \geq \max_{y' \in S(y)} |z(y') - z(y)| ; \forall y \in S$$

ou seja que T_0 está associado à máxima variação possível na função de custo, dentro de cada vizinhança.

Na prática, o enfoque (c) é pouco usado, devido fundamentalmente à dificuldade de encontrar um limite superior para Δz de forma analítica e no caso de ser achado, a possibilidade de resultar num valor excessivamente alto de T_0 que prolongará desnecessariamente o algoritmo, sem obter melhora substancial na qualidade da solução final encontrada.

Porem, vamos tentar o enfoque (c) na determinação de T_0 , aproveitando o conhecimento adquirido na fase de testes de redução do CWLP.

Dado que o algoritmo S.A. sempre aceita melhoras, bastará analisar o caso das pioras, ou seja, T_0 será obtida analiticamente de maneira tal que a probabilidade $\exp\{-\Delta z/T_0\}$ de aceitar uma solução pior que a atual, seja aproximadamente 1.

Faremos esta análise em dois casos separadamente:

(a) $y_i = 1$ (ou $i \in K_1^*$), correspondendo ao caso 3.1.3 no algoritmo S.A.

Aquí, $d = W[K_1 U K_1^* - i] - W[K_1 U K_1^*] = \Delta[K_1^*, i]$

Se $\Delta[K_1^*, i] \leq f_i$ então a nova solução é aceita independentemente de T_0 .

{ Em 3.1.3.2, $d = \Delta[K_1^*, i] \leq f_i < f_i - T \cdot \ln x ; \forall T > 0$ }

Se $\Delta[K_1^*, i] > f_i$, a nova solução é aceita com probabilidade

$$p = \exp\{-\{\Delta[K_1^*, i] - f_i\}/T\} = \exp\{-\Delta z/T\}$$

ou equivalentemente, segundo é implementado, a nova solução é aceita se dado x uniformemente distribuído em $(0,1)$,

resulta $x < p$ ou equivalentemente $T > \frac{\Delta z}{-\ln x}$

Se queremos ter $p=1$, então $T=T_0$ deverá ser infinitamente grande.

Porém, na prática, o valor de x será escolhido uniformemente no intervalo $(0.01; 0.99)$, de modo que

$$\frac{\Delta[K_1^*, i] - f_i}{-\ln x} = \frac{\Delta z}{-\ln x} \leq \frac{\Delta z}{0.01} = 100 \cdot \Delta z.$$

Assim, escolhendo $T_0 \gg 100\Delta z = 100\{\Delta[K_1^*, i] - f_i\}$, teremos que a probabilidade de aceitar a nova solução (caso seja pior que a atual) à temperatura T_0 é :

$$\begin{aligned} \exp\{-\{\Delta[K_1^*, i] - f_i\}/T_0\} &= \exp\{-\Delta z/T_0\} \geq \exp\{-\Delta z/100\Delta z\} = \\ &= \exp\{-0.01\} \approx 0.99. \end{aligned}$$

Mas T_0 deve ser independente da solução atual (representada por K_1^*) e da nova solução gerada pelo algoritmo (que depende do índice i escolhido). Assim, não podemos usar

$$T_0 = 100\{\Delta[K_1^*, i] - f_i\}$$

Porém, devido a supermodularidade temos

$$\Delta[K_1^*, i] = W[K_1 \cup K_1^* - i] - W[K_1 \cup K_1^*] \leq W[K_1] - W[K_1 + i] = \Omega_i \leq \hat{\Omega}_i,$$

onde $\hat{\Omega}_i$ é um limite superior para Ω_i , segundo foi proposto no capítulo 2 e que teremos calculado na fase de redução do CWLP, antes da aplicação do algoritmo S.A. ao problema.

Logo, $\Delta z = \Delta[K_1^*, i] - f_i \leq \hat{\Omega}_i - f_i \leq \max_{i \in K_2} \{\hat{\Omega}_i - f_i\}$.

Então podemos definir $T_0 = 100 \cdot \max_{i \in K_2} \{\hat{\Omega}_i - f_i\}$ e teremos que a probabilidade de aceitar uma solução pior que a atual é pelo menos 0.99. Ainda mais, se queremos diminuir aquela probabilidade para 0.80 (segundo Kirkpatrick [34]), fazemos:

$$p = \exp\{-\Delta z/T\} \geq 0.80 \quad \text{que implica em } T \geq 4.48\Delta z.$$

Conclui-se que $T_0^{(a)} = 4.48 \cdot \max_{i \in K_2} \{\hat{\Omega}_i - f_i\}$ (7)

(b) $y_i = 0$ (ou $i \notin K_1^*$), correspondendo ao caso 3.1.4 no algoritmo S.A.

Aquí, $d = W[K_1 \cup K_1^*] - W[K_1 \cup K_1^* + i] = \Omega[K_1^*, i]$

Se $\Omega[K_1^*, i] \geq f_i$ então a nova solução é aceita

independentemente de T_0 .

$$\{ \text{Em 3.1.4.2, } d = \Omega[K_1^*, i] \geq f_i > f_i + T \cdot \ln x, \forall T > 0 \}$$

Se $\Omega[K_1^*, i] < f_i$, a nova solução é aceita com probabilidade

$$p = \exp\{-[f_i - \Omega[K_1^*, i]]/T\} = \exp\{-\Delta z/T\}$$

Seguindo o mesmo raciocínio que em (a), podemos concluir que escolhendo $T_0 \geq 4.48\Delta z$, teremos probabilidade no mínimo 0.80 de aceitar a nova solução (caso seja pior que a atual) à temperatura T_0 .

De novo, para fazer T_0 independente da solução atual e da nova solução gerada pelo algoritmo, usamos supermodularidade obtendo

$$\begin{aligned} \Omega[K_1^*, i] &= W[K_1 U K_1^*] - W[K_1 U K_1^* + i] \geq W[K_1 U K_2 - i] - W[K_1 U K_2] \\ &= \Delta_i \geq \Delta_i^{(G)}, \end{aligned}$$

onde $\Delta_i^{(G)}$ é um limite inferior para Δ_i segundo foi proposto no capítulo 2 e que também será conhecido após a aplicação do teste de redução do CWLP.

$$\text{Logo, } \Delta z = f_i - \Omega[K_1^*, i] \leq f_i - \Delta_i^{(G)} \leq \max_{i \in K_2} \{f_i - \Delta_i^{(G)}\}$$

$$\text{Conclui-se que } T_0^{(b)} = 4.48 \cdot \max_{i \in K_2} \{f_i - \Delta_i^{(G)}\} \quad (8)$$

Usando (7) e (8) e o fato que podemos ter $y_i = 1$ ou $y_i = 0$ numa iteração qualquer do algoritmo, definiremos

$$T_0 = \max \{T_0^{(a)}, T_0^{(b)}\} \quad (9)$$

e teremos probabilidade no mínimo 0.80 de aceitar uma nova solução, tanto se melhora como se deteriora a solução atual.

OBSERVAÇÕES

1.- O valor de T_0 dado por (9) é calculado após a aplicação do teste de redução do CWLP e portanto o custo computacional de T_0 está associado à busca dos máximos em (7) e (8) o que leva tempo $O(K_2)$.

2.- No caso em que $T[K_1]$ seja viável, Ω_i e seu limite $\hat{\Omega}_i$ estarão bem definidos, $\forall i \in K_2$. No caso em que $T[K_1]$ seja inviável, tanto Ω_i como $\hat{\Omega}_i$ não estão definidos, produzindo

impossibilidade de calcular $T_0^{(a)}$ como em (7). Devido à dificuldade de encontrar outro limite superior para $\Delta[K_1^*, i]$ que seja independente de K_1^* e de i , vamos neste caso considerar só $T_0^{(b)}$ no cálculo de T_0 , ou seja, faremos $T_0 = T_0^{(b)}$.

3.- O valor $T_0^{(b)}$ estará sempre bem definido, não produzindo qualquer problema de cálculo. Efetivamente, na primeira iteração do teste de redução (iniciada com $K_1 = K_0 = \phi$, $K_2 = I$) sempre é possível calcular $\Delta_i^{(\sigma)}$, $\forall i \in K_2 = I$, devido à hipótese óbvia de que $T[K_1 \cup K_2] = T[I]$ é viável. Se após esta iteração resulta K_1 com $T[K_1]$ inviável, o teste de redução acaba e teremos todos os $\Delta_i^{(\sigma)}$. Se $T[K_1]$ for viável, passamos ao teste de fechar facilidades acabando num conjunto K_0 . Se $K_0 = \phi$ então teremos os mesmos $\Delta_i^{(\sigma)}$ anteriores; se $K_0 \neq \phi$ ao voltar ao teste para abrir facilidades, $T[K_1 \cup K_2]$ e $T[K_1 \cup K_2 - i]$ serão viáveis pois $K_1 \subseteq K_1 \cup K_2 - i \subseteq K_1 \cup K_2$, obtendo-se novos valores $\Delta_i^{(\sigma)}$.

Assim, em qualquer caso, sempre disporemos dos $\Delta_i^{(\sigma)}$, $\forall i \in K_2$.

Obviamente os $\Delta_i^{(\sigma)}$ referidos em $T_0^{(b)}$, são aqueles associados a $i \in K_2$, para o conjunto K_2 obtido no final do teste completo de redução do problema.

4.- Embora T_0 dado por (9) tenha sido obtido pensando no caso de piora na solução corrente, teremos probabilidade no mínimo 0.80 de aceitar a nova solução tanto se melhora como se deteriora a solução corrente. Isto pois se q é a probabilidade de uma solução viável ser melhor que a atual e consequentemente $1-q$ será a probabilidade de piorar a solução atual, a probabilidade de aceitar a nova solução é:

$$P(\text{aceitar a nova solução}) =$$

$$\begin{aligned} & P(\text{aceitar nova solução se ela é melhor}) \cdot P(\text{gerar solução melhor}) + \\ & + P(\text{aceitar nova solução se ela é pior}) \cdot P(\text{gerar solução pior}) \\ & = 1 \cdot q + 0.8(1-q) \\ & = 0.8 + 0.2q \geq 0.8 \end{aligned}$$

5.- Pode acontecer na prática que o valor de T_0 dado por (9) resulte num valor muito grande, o que pode levar o algoritmo a executar um número excessivo de iterações (principalmente nos níveis iniciais de temperatura), sem produzir com isso uma substancial melhora na sua eficiência. Nesse caso, que veremos experimentalmente, podemos tentar algum tipo de aceleração no algoritmo mediante o seguinte método:

- usar $\alpha \cdot T_0$ ($0 < \alpha < 1$) como temperatura inicial, para diferentes valores de α .

5.7 CONDIÇÃO DE EQUILIBRIO

A condição de equilíbrio corresponde ao passo 3.1 do algoritmo S.A. apresentado em 5.5 e consistirá na geração de $|K_2|$ soluções em cada patamar de temperatura, ou seja que para cada T fixa vamos gerar tantas soluções como o tamanho da vizinhança o que não quer dizer que vamos percorrer a vizinhança toda de uma solução y , antes de aceitar uma nova solução.

Como uma maneira de estudar o efeito produzido por este critério, vamos compara-lo com outro que em cada patamar de temperatura gera $|K_2|/2$ soluções.

Johnson [32] e [33], utiliza este tipo de condição de equilíbrio definindo o "comprimento de temperatura" L como sendo $L = N \cdot \text{SIZEFACTOR}$, onde N é o tamanho esperado de uma vizinhança e o parâmetro SIZEFACTOR é um fator usado na experimentação.

Burkard [12] utiliza um parâmetro $\text{REP} = 2n$, onde n é a quantidade de cidades no Problema de Caixeiro Viajante e incrementa este valor com cada mudança na temperatura; Mitra [44] desenvolve seu trabalho basicamente teórico com uma iteração por cada nível de temperatura; Chams [15] seguindo a ideia de Hajek [29] propõe também a utilização de um parâmetro REP na condição de equilíbrio, baseado na estimação do salto necessário para escapar de um ótimo local (infelizmente ele não consegue obter essa estimação e na prática usa o valor 2); Drex1 [22] utiliza também um parâmetro r que varia de $n/2$ até $3n/2$ (sendo n a quantidade de artigos no problema da mochila).

5.8 ESQUEMA DE REDUÇÃO DE TEMPERATURA

Corresponde ao passo 3.2 do algoritmo S.A. e para efeitos de implementação vamos considerar os seguintes esquemas:

$$T_m := 0.95 \cdot T_{m-1} \quad \text{e} \quad T_m := 0.99 \cdot T_{m-1}$$

Johnson [32] e [33] utiliza o seguinte esquema de redução de temperatura: $T_m := \text{TEMPFACTOR} \cdot T_{m-1}$, onde TEMPFACTOR é alguma potência de 0.95. Golden [27] usa este mesmo esquema com fator 0.95 para grandes instâncias do problema de caixeiro viajante e compara com a alternativas seguinte: se τ é uma temperatura arbitrariamente alta, dividir o intervalo $(0, \tau)$ em 25 partes iguais para obter temperaturas $\tau_1, \tau_2, \dots, \tau_{25}$ tal que $\tau_1 < \tau_2 < \dots < \tau_{25} = \tau$. A experiência mostra melhores resultados no primeiro caso. Para o problema das p-medianas o mesmo autor usa fator 0.9 Connolly [17] segue a ideia de Lundy [42] na utilização do esquema de redução $T_m := T_{m-1} / (1 + \beta T_{m-1})$ onde $\beta \ll 1/T_0$. Em Cerny [14] e Kirkpatrick [35] é usado um método de ensaio e erro no problema de caixeiro viajante, no entanto Bonomi [11] utiliza fator 0.925 no mesmo problema. Burkard [12] aplica S.A. ao problema de alocação quadrático e utiliza no esquema de redução de temperaturas o fator 0.5. Chams [15] utiliza fator 0.93 na aplicação do S.A. ao problema de coloração de grafos.

5.9 CRITERIO DE PARADA

Corresponde ao passo 3 do algoritmo S.A. Usaremos o mesmo criterio proposto por Johnson [32] e [33]: um contador será incrementado em 1 toda vez que uma temperatura é alterada (ou seja quando forem feitas $|K_2|$ iterações nesse patamar de temperatura) e na qual a porcentagem de soluções aceitas seja no máximo MINPERCENT (2% em nosso caso); o contador será zerado toda vez que uma melhora aconteça. Se o contador atinge valor 5, o processo é declarado "congelado" e o algoritmo pára.

Chams [15], Burkard [12] e Drexel [22] propõem parar se em algum nível de temperatura não temos mais aceitações. Connolly [17] utiliza como criterio de parada: $T_m < T_f$, onde $T_f = \delta_{\min}$ definido em 5.6. O algoritmo proposto por Cerny [14] não apresenta criterio de parada e na fase experimental só tenta

com alguns valores decrescentes de temperatura, mesmo que faz Kirkpatrick [35]. Lundy [42] apresenta uma expressão analítica para a temperatura final T_f mas não faz experiência nenhuma.

5.10 IMPLEMENTAÇÃO E RESULTADOS COMPUTACIONAIS.

Nesta seção vamos estudar o comportamento do algoritmo S.A. dado em 5.5 adicionando para tais efeitos o seguinte conjunto de parâmetros:

- Temperatura inicial: dada por (9) e as variantes sugeridas na observação 5 da seção 5.6.
- Condição de equilíbrio: apresentada na seção 5.7
- Esquema de redução de temperatura: apresentado na seção 5.8
- Critério de parada: apresentado na seção 5.9.

As instâncias testes são as mesmas que foram usadas no capítulo 2, e apresentadas em 2.3. A implementação é feita em linguagem Pascal e os resultados que se mostram foram obtidos usando um equipo PC 386 SX.

TABELA 5.1
RESULTADOS DO ALGORITMO S.A. APLICADO A CWLP

ARQUIVO	FACTOR $\alpha = 1.0$			FACTOR $\alpha = 0.1$		
	\bar{n}	$\bar{t}(\text{seg})$	Erro	\bar{n}	$\bar{t}(\text{seg})$	Erro
CAP41	146	162	0.00	72	68	0.00
CAP42	152	172	0.00	50	48	0.00
CAP43	173	198	0.00	56	54	0.00
CAP44	184	214	0.00	51	51	0.70
CAP51	495	570	0.21	247	289	0.00
CAP61	165	238	0.00	86	112	0.00
CAP62	252	380	0.00	90	114	0.00
CAP63	988	1139	0.00	780	925	0.10
CAP64	546	810	0.00	188	234	0.00
CAP81	606	1686	0.00	301	657	0.00
CAP82	661	1838	0.20	320	709	0.00
CAP83	728	1986	0.33	370	786	0.35
CAP84	833	1781	0.31	380	844	0.90
CAP91	465	1174	0.12	238	578	0.00
CAP92	1520	3783	0.25	1102	2821	0.00
CAP93	1869	4435	0.00	1491	3790	0.10
CAP94	1081	2536	0.20	517	1294	0.00
PROMEDIOS	639	1358	0.095	372	786	0.126

Na tabela 5.1 não aparecem os resultados do algoritmo para os problemas CAP111, ..., CAP124, todos de tamanho 50x50, devido ao altíssimo custo computacional envolvido nesses 8 casos. De fato, cada um deles gasta horas de computador.

Como se trata de um algoritmo probabilístico, para uma mesma entrada pode-se ter diferentes saídas, razão pela qual foram feitas 5 rodadas de cada arquivo; o número médio de iterações (\bar{n}), o tempo médio de execução (\bar{t}) e o erro médio relativo (Erro) são apresentados na tabela, para duas alternativas de temperatura inicial: αT_0 , com $\alpha=1.0$ e $\alpha=0.1$, onde T_0 é dada por (9).

De acordo com estes resultados pode-se inferir que o algoritmo assim implementado resolve uma boa porcentagem de problemas de maneira exata (59% para $\alpha=1.0$ e 70% para $\alpha=0.1$), e que os erros médios são muito bons. Nos mesmos 17 problemas testes que se mostram na tabela 5.1 podemos comparar a qualidade da solução (Erro) obtida nos seguintes algoritmos mas não considerando neles a etapa final que corresponde ao "branch & bound".

Van Roy [55]:	Erro = 0.095;
Christofides & Beasley [16]:	Erro = 0.26;
Beasley [9]:	Erro = 0.072;

O grande problema no algoritmo proposto fica nos tempos computacionais, que são proibitivos para instâncias relativamente pequenas; a justificativa do elevado custo computacional está no fato de que o algoritmo resolve um problema de transporte por cada iteração.

Nosso objetivo agora é tentar reduzir esses tempos, usando métodos aproximados para o cálculo de d , a diferença em custo de dois problemas de transporte. O algoritmo que será obtido desta maneira, será chamado de S.A. APROXIMADO para diferenciá-lo daquele em que cada problema de transporte é resolvido de maneira exata (S.A. EXATO).

No algoritmo S.A. apresentado em 5.5 são usados dois tipos de valores d , um para $i \in K_1^*$, onde $d = W[K_1 U K_1^* - i] - W[K_1 U K_1^*]$ (ver 3.1.3.1) e outro para $i \in K_1$, onde $d = W[K_1 U K_1^*] - W[K_1 U K_1^* + i]$ (ver

3.1.4.1).

No caso $i \notin K_1^*$, foi usada a seguinte aproximação: para cada $j \in J$, calculamos $c_{\min,j} = \min\{c_{ij} / l \in K_1 \cup K_1^*\}$. Se $c_{ij} < c_{\min,j}$, então o armazém i deverá fornecer o centro de demanda j e logo ficará momentaneamente aberto. Como podem existir outros centros de demanda j para quais acontece a mesma coisa, consideramos para cada $j \in J$ a expressão $v[j] := c_{\min,j} - c_{ij}$. Levando em conta a capacidade de oferta do armazém i este fornecerá aqueles centros de demanda que tenham o maior $v[j]$, ou seja aqueles que produzem maior lucro pela incorporação do armazém i .

Corresponde então à resolução de um problema simples de mochila, ou seja, ordenar os $v[j]$ de maneira decrescente e nesse ordem satisfazer as demandas dos clientes j até que a capacidade do centro de oferta i permita.

No caso $i \in K_1^*$, se tentaram duas alternativas de aproximação; a primeira consiste em calcular a diferença $W[K_1 \cup K_1^* - i] - W[K_1 \cup K_1^*]$ considerando as capacidades dos armazéns em $K_1 \cup K_1^*$ ilimitadas; a segunda consiste em proceder como no caso $i \notin K_1^*$, fazendo $K_1^{**} := K_1^* - i$, resultando deste modo $d = W[K_1 \cup K_1^{**}] - W[K_1 \cup K_1^{**} + i]$ e calculando do mesmo jeito os $v[j]$ anteriores. Dentre estas duas alternativas, foi a segunda a que apresentou os melhores resultados e portanto foi utilizada na implementação do algoritmo S.A. aproximado.

Na implementação do algoritmo S.A. aproximado foram considerados os seguintes parâmetros:

- a temperatura inicial como no algoritmo S.A. exato, para $\alpha = 1.0; 0.5; 0.1$ (ver observação 5 em 5.6)
- o esquema de redução de temperatura, a condição de equilíbrio e o critério de parada como no algoritmo S.A. exato.

No início do algoritmo, é efetuada uma iteração tipo delta para abrir armazéns, obtendo-se um conjunto K_1 de armazéns abertos a priori. Se o problema $T[K_1]$ for viável se procede com iterações tipo omega com o objetivo de abrir (uma de cada vez) aquela que tenha a menor chance de ser fechada. Este critério é usado por Robson e Bornstein [50]. Os resultados são mostrados na tabela 5.2.

Observando os resultados na Tabela 5.1 do algoritmo S.A. exato e na Tabela 5.2 do algoritmo S.A. aproximado, pode-se obter as seguintes conclusões:

- Nos arquivos CAP111...CAP124 de tamanhos 50x50, os tempos computacionais dados pelo algoritmo aproximado (Tabela 5.2) são inteiramente razoáveis, em contraposição ao algoritmo exato, que leva horas de trabalho. Desde perspectiva a aproximação proposta tem sentido. Comparando somente os demais arquivos, a redução em tempo é de aproximadamente 99%.

- Os erros medios na Tabela 5.2 ficam bem maiores do que na Tabela 5.1. A justificativa disto é devida à aceitação de erros relativamente grandes nos arquivos CAP83, CAP84, CAP113, CAP114, produzidos pelo fato de que ficam incorporados dentro de K_1 na fase de redução, alguns armazéns que não formam parte na solução ótima, coisa que acontece na aplicação da segunda fase do teste de redução, ou seja, quando se incorporam armazéns usando teste de fechamento. Com tudo, a qualidade das soluções finais obtidas pelo algoritmo aproximado em relação aos tempos envolvidos na sua determinação, são bastante bons.

- Observando simplesmente a Tabela 5.2, onde se mostram resultados relacionados com três tipos de algoritmos, um para cada valor de α , podemos concluir que o valor de α não afeta o resultado dos arquivos mencionados no parágrafo anterior. O algoritmo com $\alpha=0.1$ reduz o tempo medio em aproximadamente 24% com respeito aquele com $\alpha=1.0$ e o erro medio é incrementado de 13% aproximadamente.

Considerando que estes tempos são já aceitáveis, vamos continuar nossas comparações com o algoritmo aproximado que usa $\alpha=1.0$.

TABELA 5.2

RESULTADOS DO ALGORITMO S.A.APROXIMADO APLICADO A CWLP.

ARQUIVO	FACTOR $\alpha=0.1$			FACTOR $\alpha=0.5$			FACTOR $\alpha=1.0$		
	\bar{n}	\bar{t} (seg)	ERRO (%)	\bar{n}	\bar{t} (seg)	ERRO (%)	\bar{n}	\bar{t} (seg)	ERRO (%)
CAP41	28	2.5	0.00	76	2.9	0.00	90	3.1	0.00
CAP42	40	2.6	0.00	69	2.9	0.00	128	3.5	0.00
CAP43	30	4.7	0.00	104	5.3	0.00	106	5.5	0.00
CAP44	40	6.1	0.00	52	6.2	0.00	86	6.5	0.00
CAP51	173	4.7	0.07	323	6.8	0.00	386	7.7	0.09
CAP61	114	2.4	0.05	268	3.8	0.00	287	4.3	0.00
CAP62	231	3.4	0.13	333	4.9	0.13	393	5.8	0.00
CAP63	216	5.2	0.00	432	7.7	0.09	510	9.0	0.00
CAP64	364	6.8	0.72	540	9.5	0.72	648	10.6	0.72
CAP81	325	8.2	0.27	658	11.6	0.34	710	12.7	0.11
CAP82	442	15.4	1.81	624	17.7	1.81	689	18.8	1.81
CAP83	173	20.2	2.88	368	23.3	2.88	537	25.3	2.01
CAP84	212	28.5	1.93	429	31.1	1.93	546	32.8	1.93
CAP91	400	6.5	0.15	710	10.0	0.24	800	11.6	0.13
CAP92	506	7.8	0.06	791	12.0	0.06	950	14.3	0.06
CAP93	588	9.4	1.80	1127	16.4	1.95	1141	17.5	1.26
CAP94	574	15.6	0.62	903	20.1	0.62	1015	22.3	0.62
CAP111	899	57.6	0.37	1532	67.3	0.49	1634	71.0	0.37
CAP112	646	64.8	0.00	1380	76.1	0.09	1469	78.6	0.20
CAP113	1406	100.7	2.43	2090	110.0	2.28	2356	114.2	2.43
CAP114	671	111.6	1.97	1266	119.9	1.88	1621	125.6	2.05
CAP121	1419	24.2	0.25	2393	37.7	0.38	2594	42.9	0.50
CAP122	1200	22.6	0.16	2055	36.2	0.26	2295	42.3	0.19
CAP123	1625	39.4	1.16	2330	53.4	1.01	2591	59.8	0.16
CAP124	1349	46.4	0.49	1932	56.9	0.66	2361	64.3	0.62
PROM.	546	24.6	0.69	911	29.9	0.71	1037	32.4	0.61

A seguir, tentaremos melhorar a performance do nosso algoritmo mediante a variação de outro dos parâmetros nele considerado. Desta vez vamos mudar o esquema de redução de temperatura, passando de $T_m = 0.9 T_{m-1}$ para $T_m = 0.99 T_{m-1}$ ou seja estaremos fazendo uma redução mais lenta da temperatura no algoritmo. Vamos manter os três algoritmos que utilizam $\alpha = 0.1, 0.5, 1.0$ e os resultados destas modificações são apresentados na Tabela 5.3.

TABELA 5.3

RESULTADOS DO ALGORITMO S.A.APROXIMADO APLICADO A CWLP.

ARQ.	FACTOR $\alpha=0.1$			FACTOR $\alpha=0.5$			FACTOR $\alpha=1.0$		
	\bar{n}	\bar{t} (seg)	ERRO (%)	\bar{n}	\bar{t} (seg)	ERRO (%)	\bar{n}	\bar{t} (seg)	ERRO (%)
CAP41	28	2.5	0.00	472	6.3	0.00	560	7.5	0.00
CAP42	38	2.6	0.00	590	7.7	0.00	976	11.6	0.00
CAP43	45	4.8	0.00	337	7.6	0.00	620	10.1	0.00
CAP44	45	6.1	0.00	234	7.9	0.00	668	12.1	0.00
CAP51	1280	14.8	0.00	2793	34.9	0.00	3353	43.4	0.00
CAP61	546	5.7	0.00	1500	16.2	0.05	2151	23.1	0.10
CAP62	942	9.7	0.00	2592	28.1	0.00	3474	38.3	0.00
CAP63	1392	16.1	0.00	3480	41.3	0.00	4260	53.4	0.09
CAP64	3232	34.4	0.72	4680	57.2	0.72	5516	67.4	0.72
CAP81	3609	32.7	0.11	4541	51.7	0.11	6768	77.1	0.34
CAP82	3445	45.6	1.81	5776	79.1	1.81	6183	87.7	1.81
CAP83	1291	31.9	2.88	3020	58.0	2.88	4272	77.9	2.88
CAP84	1022	36.1	1.93	3605	65.1	1.93	4285	76.8	1.93
CAP91	3580	32.4	0.16	5465	59.6	0.10	5580	69.5	0.00
CAP92	4104	41.5	0.06	7106	81.8	0.06	8334	106.9	0.06
CAP93	3654	43.7	0.00	6566	93.9	0.00	8323	122.6	0.00
CAP94	4795	55.8	0.49	7854	101.7	0.62	10038	131.1	0.62
CAP111	5218	93.7	0.33	11096	198.8	0.27	13882	251.0	0.40
CAP112	5535	116.0	0.02	10298	205.3	0.18	13971	264.9	0.01
CAP113	10564	188.9	2.26	18392	294.1	2.26	20950	343.2	2.44
CAP114	3964	142.3	1.80	10032	234.8	1.82	12312	281.3	1.88
CAP121	5747	68.3	0.06	17959	228.8	0.06	16669	255.7	0.00
CAP122	9885	144.5	0.06	18105	253.4	0.16	20715	311.1	0.18
CAP123	11515	158.3	0.18	19780	312.1	0.25	21267	368.6	0.85
CAP124	9982	139.6	0.66	18461	267.8	0.64	21190	345.3	0.66
PROM.	3818	57.5	0.54	7389	111.7	0.55	8652	137.5	0.59

A primeira coisa que salta à vista na tabela 5.3 é o grande incremento no número de iterações nos algoritmos, coisa que era de esperar. Junto com isso um incremento nos tempos computacionais, que no caso do algoritmo com $\alpha=1.0$ é de aproximadamente 76%, sendo que a redução no erro relativo foi apenas de 3%. Consideramos portanto que esta variação não implica em uma melhora substancial no algoritmo e manteremos o esquema de redução de temperatura como sendo $T_m = 0.9 T_{m-1}$.

Uma outra variante em nosso algoritmo se produz quando

mudamos a condição de equilíbrio, que até agora foi de gerar $|K_2|$ soluções em cada patamar de temperatura. Tentaremos a metade desse número de soluções geradas, esperando com isso uma redução no número de iterações do algoritmo e portanto uma redução também nos tempos computacionais. Os resultados obtidos neste caso foram um tempo medio de 22 seg. e o erro medio de 0.85, nos 25 arquivos para $\alpha=0.1$. Se comparados com a respectiva coluna na tabela 5.2 podemos concluir que houve um incremento de 23% no erro relativo junto com uma redução no tempo de 10% aproximadamente. Estes resultados fazem pensar que esta variação não é uma melhora significativa no algoritmo aproximado inicial cujos resultados aparecem na tabela 5.2.

Em Robson & Bornstein [50] e Robson [51] se apresentam resultados de heurísticas baseadas nos procedimentos ADD e DROP referidos no Capítulo 1. Quando aplicados sobre as 12 instâncias CAP4, CAP6 e CAP8 (as que são mencionadas explicitamente), eles obtiveram um erro medio relativo de 0.8% (48% acima da nossa proposta) e um tempo medio de 11.4 seg., similar ao nosso tempo sobre os mesmos problemas testes. Porém, resultados recentes devidos a Campelo [13] trabalhando com heurísticas baseadas também nos procedimentos ADD e DROP e geradas a partir do trabalho de Robson & Bornstein [50], mostram excelente comportamento, atingindo-se erros medios relativos inferiores a 0.05% em tempos inferiores a 1 seg., num PC AT/486.

CAPITULO 6

CONCLUSÕES

Nesta tese trabalhou-se em cima do problema de localização capacitado, abordando fundamentalmente dois aspectos : primeiro um estudo analítico e experimental dos testes de redução e seguidamente a geração de uma heurística baseada em Simulated Annealing.

Na primeira parte se estuda os mecanismos de redução que existem na literatura e se tenta atingir melhoras na eficiencia dos mesmos. Aqui é feito um estudo baseado em relaxação lagrangeana, através do qual se conseguem reproduzir a maioria dos limites existentes para o incremento no custo de transporte ao fechar uma facilidade, e ainda mais, torna-se possível a determinação de um conjunto de valores para as variáveis duais, que produzem um novo limite que pode ser considerado uma melhora nos testes de redução do problema. Isto é justificado mais tarde com os resultados obtidos na aplicação destas melhoras nas instâncias testes que forem usadas no desenvolvimento deste trabalho.

A segunda parte implica na geração de uma heurística tipo Simulated Annealing para o problema. Inicialmente se apresenta o problema num formato geral. Percebe-se imediatamente a dificuldade da avaliação da diferença Δz entre as soluções ótimas de dois problemas de transporte. Contudo, implementa-se este algoritmo com a ideia de apreciar o comportamento em termos da qualidade da solução final obtida. Dado que os resultados foram muito razoáveis, se procurou a maneira de reduzir os tempos computacionais.

Esta redução foi obtida com a utilização de aproximações nos Δz , coisa que não é conhecida neste tipo de implementações. Representou portanto outra contribuição deste trabalho. Depois de procurar varias alternativas, conseguiu-se achar uma boa forma de aproximar os cálculos Δz onde em vez de um problema de transporte, agora só é necessário resolver um problema simples de mochila. Os testes computacionais mostraram a enorme redução

em tempo atingida, com uma pequena redução na qualidade da solução obtida.

Nesta fase de implementação tentou-se varias combinações dos parâmetros do algoritmo simulated annealing. Aqui foi aproveitada a informação analítica obtida na fase de redução, para a obtenção da temperatura inicial o que também significou uma contribuição desta tese. Esta técnica geralmente não é utilizada na aplicação deste tipo de algoritmos em outros problemas de otimização combinatória.

A comparação do desempenho global do algoritmo S.A. com outras heurísticas (mais especificamente heurísticas ADD/DROP) para o problema de localização capacitado, não demonstra uma nítida vantagem para o primeiro. Isto pode ser devido à relativa complexidade do sub-problema gerado (problema de transporte) que se requer, seja resolvido um grande número de vezes. Por outro lado, resalta-se a grande robustez do algoritmo o que permite a sua aplicação em uma gama variada de problemas semelhantes.

REFERENCIAS BIBLIOGRAFICAS

- (1) AARTZ E.H.L., KORST J.H.M. e VAN LAARHOVEN P.J.M.- " A quantitative Analysis of the Simulated Annealing Algorithm:A Case Study for the Traveling Salesman Problem", Journal of Statistical Physics, Vol 50, N^{os}1-2, pp.187-206, (1988).
- (2) AKINC U. e KHUMAWALA B.M., " An Efficient Branch and Bound Algorithm for the Capacitated Warehouse Location Problem", Management Science, Vol 23, N^o6, pp.585-594, (1977).
- (3) ANILY S.e FEDERGRUEN A., "Simulated Annealing Methods with General Acceptance Probabilities, Journal of Applied Probabilities, N^o24, pp.657-667, (1987).
- (4) ARAGON C.R., JOHNSON D.S. , MCGEOCH L.A. e SCHEVON C., " Optimization by Simulated Annealing: An Experimental Evaluation", Workshop on Statistical Physics in Engineering and Biology, (1984).
- (5) BAKER B.M., "Linear Relaxations of the Capacitated Warehouse Location Problem" , Journal of the Operational Research Society, Vol 33, pp.475-479, (1982).
- (6) BARTEZZAGHI E.,COLORNI A.e PALERMO P.C., " A Search Tree Algorithm for Plant Location Problems ", European Journal of Operational Research, Vol 7, pp.371-379, (1981).
- (7) BARBOSA V. " Redes Neuronalis e Simulated Annealing como Ferramentas para Otimização Combinatória " , Investigación Operativa, Vol 1, N^o2, pp.125-142, (1989).
- (8) BARCELO J. e CASANOVAS J., "A Heuristic Lagrangean Algorithm for the Capacitated Plant Location Problem " , European Journal of Operational Research, Vol 15, pp.212-226, (1984).
- (9) BEASLEY J.E., "An Algorithm for Solving Large Capacitated Warehouse Location Problems", European Journal of Operational Research, Vol 33, pp.314-325, (1988).

- (10) BITRAN G., CHANDRU V., SEMPOLINSKI D. e SHAPIRO J., "Inverse Optimization: An application to the Capacitated Plant Location Problem", *Management Science*, Vol 27, N° 10, pp.1120-1141, (1981)
- (11) BONOMI E. e LUTTON J., "The N-city Traveling Salesman Problem: Statistical Mechanics and the Metropolis Algorithm" *Siam Review*, Vol 26, N°4, pp.550-568, (1984).
- (12) BURKARD R.E. e RENDL F., " A thermodynamical Motivated Simulated Procedure for Combinatorial Optimization Problems ", *European Journal of Operational Research*, Vol 17 pp.169-174, (1984).
- (13) CAMPÊLO NETO M. B., " Testes de Redução e Heurísticas ADD/DROP para o Problema de Localização Capacitado", Tese Mestrado, COPPE/UFRJ, (1993).
- (14) CERNY V. , " Thermodynamical Approach to the Traveling Salesman Problem: An Efficient Simulation Algorithm", *Journal of Optimization Theory and Applications*, Vol 45, N°1, pp.41-51, (1985).
- (15) CHAMS M., HERTZ A. e DE WERRA D., " Some experiments with Simulated Annealing for Coloring Graphs ", *European Journal of Operational Research*, Vol 32, pp.260-266, (1987).
- (16) CHRISTOFIDES N. e BEASLEY J.E., "Extensions to a Lagrangean Relaxation Approach for the Capacitated Warehouse Location Problem", *European Journal of Operational Research*, Vol 12, pp.19-28 , (1983).
- (17) CONNOLLY D., "An improved Simulated Annealing Scheme for the Q.A.P.", *European Journal of Operational Research*, Vol 46, pp.93-100, (1990).
- (18) CONNOLLY D. "General Purpose Simulated Annealing", *Journal Opl.Res.Soc.*, Vol 43, pp.495-505, (1992).
- (19) CORNUEJOLS G., SRIDHARAN R. e THIZY J., " A Comparison of

Heuristics and Relaxations for the Capacitated Plant Location Problem", Management Science Research Report, MSRR 541, (1988).

- (20) DAVIS P.S. e RAY T.L., " A Branch-Bound Algorithm for the Capacitated Facilities Location Problem", Naval Research Logistic Quarterly, Vol 16, pp.331-343, (1969).
- (21) DOMSCHKE W.e DREXL A., " Add-Heuristics Starting Procedures for Capacitated Plant Location Models", European Journal of Operational Research, Vol 21, pp.47-53, (1985).
- (22) DREXL A. , " A Simulated Annealing Approach to the Multiconstraint Zero-One Knapsack Problem", Computing, Vol 40 pp.1-8, (1988).
- (23) EFROYMSON M.A.e RAY T.L., " A Branch-Bound Algorithm for Plant Location", Operations Research, Vol 14, pp. 361-368, (1966).
- (24) FAIGLE U. e KERN W., " Note on the Convergence of Simulated Annealing Algorithms ", Siam Journal on Control and Optimization, Vol 29, N°1, pp.153-159, (1991).
- (25) FELDMAN E., LEHRER F.A. e RAY T.L., " Warehouse Location Under Continuous Economies of Scale ", Management Science, Vol 12, N°9, pp.670-684, (1966).
- (26) GEOFFRION A. e MCBRIDE R., " Lagrangean Relaxation Applied to Capacitated Facility Location Problems", AIIE Transact. Vol 10, N°1, pp.40-47, (1978).
- (27) GOLDEN B.L.e SKISCIM C.C., " Using Simulated Annealing to solve Routing and Location Problems ", Naval Research Logistic Quarterly, Vol 33, pp.261-279, (1986).
- (28) GOLDSTEIN L. e WATERMAN M., " Neighborhood size in the Simulated Annealing Algorithm ", American Journal of Mathematical and Management Sciences, Vol 8, N°s 3-4, pp.409-423, (1988).

- (29) HAJEK B., "Cooling Schedules for Optimal Annealing", Mathematics of Operation Research, Vol 13, N°2, pp.311-329, (1988).
- (30) ISAACSON e MADSEN, "Markov Chains:Theory and Applications", Wiley, New York, (1976).
- (31) JACOBSEN S., "Heuristics for the Capacitated Plant Location Model, "European Journal of Operational Research, Vol 12, pp. 253-261, (1983).
- (32) JOHNSON D., ARAGON C. R., MCGEOCH L. A. e SCHEVON C. , " Optimization by Simulated Annealing : An Experimental Evaluation; Part I, Graph Partitioning",Operations Research Vol 37, N°6, pp.865-892, (1989).
- (33) JOHNSON D., ARAGON C. R., MCGEOCH L. A. e SCHEVON C. , " Optimization by Simulated Annealing : An Experimental Evaluation; Part II,Graph Coloring and Number Partitioning" Operations Research, Vol 39, N°3, pp.378-406, (1991).
- (34) KELLY D.L.e KHUMAWALA B.M., "Capacitated Warehouse Location with Concave Costs", Journal of the Operational Research Society, Vol 33, pp.817-826, (1982).
- (35) KIRKPATRICK S., GELATT C.D. e VECCHI M.P., " Optimization by Simulated Annealing", Science, Vol 20, N° 4598, pp.671 - 680, (1983).
- (36) KIRKPATRICK S., " Optimization by Simulated Annealing: Quantitative Studies", Journal of Statistical Physics, Vol 34, N°5-6, pp.975-986, (1984).
- (37) KLINCEWICZ J.G.e LUSS H., "A Lagrangian Relaxation Heuristic for Capacitated Facility Location with Single - Source Constraint",Journal of the Operational Research Society,Vol 37, N°5, pp.495-500, (1986).
- (38) KUEHN A.A. e HAMBURGER M.J., " A Heuristic Program for Locating Warehouses", Management Science, Vol 9, pp.643-666

(1963).

- (39) LEUNG J.M.Y. e MAGNANTI T.L., "Valid Inequalities and Facets of the Capacitated Plant Location Problem", *Mathematical Programming*, Vol 44, pp.271-291, (1989).
- (40) LIN S.e KERNIGHAN B.W., "An Effective Heuristic Algorithm for the Traveling Salesman Problem", *Operations Research*, Vol 21 pp.498-516, (1973).
- (41) LUNDY M., " Applications of the Annealing Algorithm to Combinatorial Problems in Statistics", *Biometrika*, Vol 72 pp.191-198, (1985).
- (42) LUNDY M. e MEES A., " Convergence of an Annealing algorithm" *Mathematical Programming*, Vol 34, pp.111-124, (1986).
- (43) METROPOLIS N., ROSENBLUTH A., ROSENBLUTH M. e TELLER A., "Equation of State Calculations by Fast Computing Machines" *Journal of Chemical Physics*, Vol 21, N^o6, pp.1087-1092, (1953).
- (44) MITRA D., ROMEO F.e SANGIOVANNI-VINCENTELLI A., "Convergence and Finite-Time behaviour of Simulated Annealing", *Advance Applied Probabilistic*, Vol 18, pp.747-771, (1986).
- (45) MONTEROSO C.D.B. e BORNSTEIN C.T., "A Heuristic Algorithm for the Location and Dimensioning of Warehouses ", *Methods of Operations Research*, Vol 35, pp.315-328, (1979).
- (46) NAUSS R.M., " An Improved Algorithm for the Capacitated Facility Location Problem ", *Journal of the Operational Research Society*, Vol 29, N^o12, pp.1195-1201, (1978).
- (47) PEDROSA FILHO E., "Heurísticas Descendentes-Ascendentes para o Problema de Localização Multicapacitado", *Tese Mestrado, PUC-RJ*, (1989).
- (48) RANDELMAN R.E.e GREY G.S., " N - City Traveling Salesman Problem: Optimization by Simulated Annealing ", *Journal of Statistical Physics*, Vol 45, N^{os}5-6, pp.885-890, (1986).

- (49) RANGEL M.C., "Critério de Dominância para Problemas de Localização de Armazéns Capacitados de Grande Porte ", Tese Mestrado, COPPE-UFRJ, Rio de Janeiro, (1991).
- (50) ROBSON MATEUS G.e BORNSTEIN C.T., " Dominance Criteria for the Capacitated Warehouse Location Problem", Journal of the Operational Research Society, Vol 42, N^o2, pp.145-149, (1991).
- (51) ROBSON MATEUS G., " Algoritmo Exato e Heurísticas para o Problema de Localização", Tese Doutorado, COPPE-UFRJ, (1986).
- (52) ROMEO F.e SANGIOVANNI-VINCENTELLI A., " Probabilistic Hill Climbing Algorithms: Properties and Applications " , ERL Memorandum, University of California, Berkeley, (1984).
- (53) SA G., " Branch-and-Bound and Approximate Solutions to the Capacitated Plant-Location Problem", Operations Research, Vol 17, pp.1005-1016, (1969).
- (54) SOLAND R.M., "Optimal Facility Location with Concave Costs" Operations Research, Vol.22, pp.373-382, (1974).
- (55) VAN ROY T. J. , " A Cross Decomposition Algorithm for Capacitated Facility Location", Operations Research, Vol 34 N^o1, pp.145-163, (1986).

APENDICE

TEOREMA 1 (Abel)

Seja $\{a_k\}_{k=1}^{\infty}$ uma sequência de números reais.

a) Se $\sum_{k=0}^{\infty} a_k$ converge para um número real a , então

$$\lim_{s \rightarrow 1^-} \sum_{k=0}^{\infty} a_k \cdot s^k = \sum_{k=0}^{\infty} a_k = a$$

b) Se $a_k \geq 0$ e $\lim_{s \rightarrow 1^-} \sum_{k=0}^{\infty} a_k \cdot s^k = a \leq \infty$, então

$$\sum_{k=0}^{\infty} a_k \text{ converge para } a \quad (\text{escrevemos, } \sum_{k=0}^{\infty} a_k = a)$$

PROVA

(a) Provaremos que $\lim_{s \rightarrow 1^-} \left| \sum_{k=0}^{\infty} a_k (s^k - 1) \right| = 0$, ou seja que para s

suficientemente próximo de 1, $\left| \sum_{k=0}^{\infty} a_k (s^k - 1) \right| < \varepsilon$.

$$\sum_{k=0}^{\infty} a_k \Rightarrow \lim_{l \rightarrow \infty} \sum_{k=0}^l a_k = a < \infty$$

$$\Rightarrow \forall \varepsilon > 0, \exists N \text{ tal que para } N' \geq N, \left| \sum_{k=0}^{N'} a_k - a \right| < \varepsilon/2$$

$$\Rightarrow \forall \varepsilon > 0, \exists N \text{ tal que para } N' \geq N, \left| \sum_{k=0}^{N'} a_k - \sum_{k=0}^{\infty} a_k \right| < \varepsilon/2$$

$$\Rightarrow \forall \varepsilon > 0, \exists N \text{ tal que para } N' \geq N, \left| \sum_{k=N+1}^{\infty} a_k \right| < \varepsilon/2$$

$$\Rightarrow \forall \varepsilon > 0, \exists N \text{ tal que para } N' \geq N, \left| \sum_{k=N'}^{\infty} a_k \right| < \varepsilon/2$$

Denotando por $A_r = \sum_{k=r}^{\infty} a_k$, temos que

$$\forall \varepsilon > 0, \exists N \text{ tal que } |A_{N+1}| < \varepsilon/2, |A_{N+2}| < \varepsilon/2, \dots$$

$$\text{Agora, } \left| \sum_{k=0}^{\infty} a_k (s^k - 1) \right| \leq \left| \sum_{k=0}^N a_k (s^k - 1) \right| + \left| \sum_{k=N+1}^{\infty} a_k (s^k - 1) \right|$$

$$\begin{aligned} \text{Para } 0 \leq \varepsilon \leq 1, \quad \left| \sum_{k=0}^N a_k (\varepsilon^k - 1) \right| &\leq \sum_{k=0}^N |a_k| \cdot |\varepsilon^k - 1| \\ &\leq M \cdot \sum_{k=0}^N |\varepsilon^k - 1| \quad \text{onde } M = \max_{0 \leq k \leq N} \{|a_k|\} < \infty \\ &\leq M \cdot N \cdot |\varepsilon^N - 1| \end{aligned}$$

Como $\lim_{\varepsilon \rightarrow \infty} |\varepsilon^N - 1| = 0$, temos que para ε suficientemente próximo

$$\text{de } 1, \quad \left| \sum_{k=0}^N a_k (\varepsilon^k - 1) \right| < \varepsilon/2$$

Por outra parte,

$$\begin{aligned} \left| \sum_{k=N+1}^{\infty} a_k (\varepsilon^k - 1) \right| &= \left| \sum_{k=N+1}^{\infty} (A_k - A_{k+1}) \cdot (\varepsilon^k - 1) \right| \\ &= \left| A_{N+1} \cdot (\varepsilon^{N+1} - 1) + \sum_{k=N+2}^{\infty} A_k \cdot (\varepsilon^k - \varepsilon^{k-1}) \right| \\ &\leq |A_{N+1}| \cdot |\varepsilon^{N+1} - 1| + \sum_{k=N+2}^{\infty} |A_k| \cdot |\varepsilon^k - \varepsilon^{k-1}| \\ &\leq \frac{\varepsilon}{2} |\varepsilon^{N+1} - 1| + \frac{\varepsilon}{2} \sum_{k=N+2}^{\infty} |\varepsilon^k - \varepsilon^{k-1}| \\ &= \frac{\varepsilon}{2} |\varepsilon^{N+1} - 1| + \frac{\varepsilon}{2} \sum_{k=N+2}^{\infty} (\varepsilon^{k-1} - \varepsilon^k) \\ &= \frac{\varepsilon}{2} (1 - \varepsilon^{N+1}) + \frac{\varepsilon}{2} \varepsilon^{N+1} = \varepsilon/2 \end{aligned}$$

Assim, para ε suficientemente próximo de 1,

$$\left| \sum_{k=0}^{\infty} a_k (\varepsilon^k - 1) \right| \leq \left| \sum_{k=0}^N a_k (\varepsilon^k - 1) \right| + \left| \sum_{k=N+1}^{\infty} a_k (\varepsilon^k - 1) \right| < \frac{\varepsilon}{2} + \frac{\varepsilon}{2} = \varepsilon$$

$$(b) \text{ para } 0 < \varepsilon < 1, \quad 0 \leq \sum_{k=0}^{\infty} a_k \cdot \varepsilon^k \leq \sum_{k=0}^{\infty} a_k$$

Logo, se $\lim_{\varepsilon \rightarrow 1} \sum_{k=0}^{\infty} a_k \cdot \varepsilon^k = \infty$ então $\sum_{k=0}^{\infty} a_k = \infty$

No caso que $\lim_{s \rightarrow 1^-} \sum_{k=0}^{\infty} a_k \cdot s^k = a < \infty$, temos para cada $0 < s < 1$

$$\sum_{k=0}^n a_k \cdot s^k \leq \sum_{k=0}^{\infty} a_k \cdot s^k$$

que implica em $\sum_{k=0}^n a_k = \lim_{s \rightarrow 1^-} \sum_{k=0}^n a_k \cdot s^k \leq \lim_{s \rightarrow 1^-} \sum_{k=0}^{\infty} a_k \cdot s^k ; \forall n$

Logo $\sum_{k=0}^n a_k \leq a < \infty, \forall n$

Conclui-se que $\left\{ \sum_{k=0}^n a_k \right\}_{n=1}^{\infty}$ é uma sequência monotonamente crescente ($a_k \geq 0, \forall k$) e limitada superiormente por a .

Logo existe $\lim_{n \rightarrow \infty} \sum_{k=0}^n a_k = \sum_{k=0}^{\infty} a_k = a'$

Usando a parte (a) deste teorema, $a = \lim_{s \rightarrow 1^-} \sum_{k=0}^{\infty} a_k \cdot s^k = \sum_{k=0}^{\infty} a_k = a'$

TEOREMA 2

Sejam $\{a_k\}, \{u_k\}, \{b_k\}$ ($k = 0, 1, 2, \dots$) três sequências

de números reais tais que $a_k, u_k, b_k \geq 0, \sum_{k=0}^{\infty} a_k = 1, \sum_{k=0}^{\infty} b_k < \infty$, a sequência $\{u_k\}$ seja limitada e $\text{MDC} \{k / a_k > 0\} = 1$.

Se a equação $u_n - \sum_{k=0}^n a_{n-k} \cdot u_k = b_n$ é satisfeita para $n = 0, 1, 2, \dots$ então

(a) $\lim_{n \rightarrow \infty} u_n$ existe

$$(b) \lim_{n \rightarrow \infty} u_n = \begin{cases} \frac{\sum_{k=0}^{\infty} b_k}{\sum_{k=0}^{\infty} k \cdot a_k} & \text{se } \sum_{k=0}^{\infty} k \cdot a_k < \infty \\ 0 & \text{se } \sum_{k=0}^{\infty} k \cdot a_k = \infty \end{cases}$$

PROVA

Primeiramente vamos supor $a_1 > 0$. Com isso $\text{MDC}\{k/a_k > 0\} = 1$
 No final da prova se comenta o caso $a_1 = 0$.

$\{u_n\}$ limitada $\Rightarrow \exists \lambda = \lim_{n \rightarrow \infty} \sup u_n < \infty$

Seja n_1, n_2, \dots uma subsequência de $\{u_n\}$ tal que $\lim_{j \rightarrow \infty} u_{n_j} = \lambda$

Usando a condição $a_1 > 0$, provaremos que $\lim_{j \rightarrow \infty} u_{n_j-1} = \lambda$

Suponhamos que $\lim_{j \rightarrow \infty} u_{n_j-1} \neq \lambda$ (ou seja $\lim_{j \rightarrow \infty} u_{n_j-1} = \lambda^* < \lambda$)

Pela definição de λ , existe $\lambda' < \lambda$ tal que $u_{n_j-1} < \lambda'$ (1)

para um número infinito de valores j (por exemplo $\lambda' = \frac{1}{2}(\lambda - \lambda^*)$)

Seja $\varepsilon = \frac{a_1(\lambda - \lambda')}{4}$ e $M = \sup\{u_n / n \geq 0\}$

Como $\sum_{k=0}^{\infty} a_k = 1$ então $\exists N$ tal que para $n \geq N$ $\sum_{k=0}^n a_k > 1 - \frac{\varepsilon}{M}$ (2)

Agora escolhamos j suficientemente grande de modo tal que:

$$\begin{aligned} n_j &\geq N; \\ u_{n_j} &> \lambda - \varepsilon \quad (\text{resulta de } \lim_{j \rightarrow \infty} u_{n_j} = \lambda) \end{aligned} \quad (3)$$

$$u_{n_j-1} < \lambda' < \lambda \quad (\text{de (1)})$$

$$0 \leq b_{n_j} < \varepsilon \quad \left(\sum_{k=0}^{\infty} b_k < \infty \Rightarrow b_k \rightarrow 0 \right) \quad \text{e}$$

$u_n < \lambda + \varepsilon$, para cada $n \geq n_j - N$ pois segundo a definição de λ , deve existir K tal que para $n > K$, $u_n < \lambda + \varepsilon$. Então é só achar j tal que $n_j - N > K$.

Então, como a equação $u_n - \sum_{k=0}^n a_{n-k} \cdot u_k = b_n$ é equivalente com

$u_n - \sum_{k=0}^n a_k \cdot u_{n-k} = b_n$ (para $n = 0, 1, 2, \dots$) que é satisfeita por u_n , também será satisfeita por u_{n_j} .

$$\begin{aligned} \text{Assim, } u_{n_j} - \sum_{k=0}^{n_j} a_k \cdot u_{n_j-k} &= b_{n_j} < \varepsilon \\ \Rightarrow u_{n_j} &< \sum_{k=0}^N a_k \cdot u_{n_j-k} + \sum_{k=N+1}^{n_j} a_k \cdot u_{n_j-k} + \varepsilon \end{aligned}$$

$$\begin{aligned}
&\leq \sum_{k=0}^N a_k \cdot u_{n_j-k} + M \sum_{k=N+1}^n a_k + \varepsilon \quad (\text{pois } u_n \leq M, \forall n) \\
&< \sum_{k=0}^N a_k \cdot u_{n_j-k} + 2\varepsilon \\
&\left[\text{pois } 1 - \frac{\varepsilon}{M} < \sum_{k=0}^n a_k \leq 1 \text{ (devido a (2))} \Rightarrow \sum_{k=0}^N a_k > 1 - \frac{\varepsilon}{M} \right] \\
&\Rightarrow \sum_{k=N+1}^n a_k = \sum_{k=0}^n a_k - \sum_{k=0}^N a_k \leq 1 - (1 - \frac{\varepsilon}{M}) = \frac{\varepsilon}{M} \\
&= a_0 \cdot u_{n_j} + a_1 \cdot u_{n_j-1} + \dots + a_N \cdot u_{n_j-N} + 2\varepsilon \\
&< a_0(\lambda + \varepsilon) + a_1(\lambda') + \dots + a_N(\lambda + \varepsilon) + 2\varepsilon \\
&= (a_0 + a_2 + a_3 + \dots + a_N)(\lambda + \varepsilon) + a_1\lambda' + 2\varepsilon \\
&\leq (1 - a_1)(\lambda + \varepsilon) + a_1\lambda' + 2\varepsilon \quad \left[\text{pois } \sum_{k=0}^N a_k \leq \sum_{k=0}^{\infty} a_k = 1 \right] \\
&= (1 - a_1)\varepsilon + \lambda - a_1(\lambda - \lambda') + 2\varepsilon \\
&= (1 - a_1)\varepsilon + \lambda - 4\varepsilon + 2\varepsilon \quad \left[\text{pois } \varepsilon = \frac{a_1(\lambda - \lambda')}{4} \right] \\
&= \lambda - \varepsilon(1 + a_1) \\
&\leq \lambda - \varepsilon
\end{aligned}$$

$\therefore u_{n_j} < \lambda - \varepsilon$ o que contradiz (3).

Assim fica provado que $\lim_{j \rightarrow \infty} u_{n_j} = \lambda$

Repetindo o argumento anterior (aplicado agora sobre u_{n_j-1} em vez de u_{n_j}), temos que $\lim_{j \rightarrow \infty} u_{n_j-2} = \lambda$. Assim indutivamente resulta $\lim_{j \rightarrow \infty} u_{n_j-d} = \lambda$, para cada d inteiro não negativo.

Seja agora $r_n = a_{n+1} + a_{n+2} + \dots$

$$\text{Temos } \sum_{k=0}^{\infty} k \cdot a_k = a_1 + 2a_2 + 3a_3 + \dots + ma_m + \dots = r_1 + r_2 + \dots = \sum_{n=0}^{\infty} r_n$$

(tanto se $\sum_{k=0}^{\infty} k \cdot a_k$ converge ou não).

Alem disso, $a_1 = r_0 - r_1$, $a_2 = r_1 - r_2$, \dots , $a_k = r_{k-1} - r_k$ ($k=1, 2, \dots$)

$$\begin{aligned}
 \text{Logo } u_n - \sum_{k=0}^n a_k \cdot u_{n-k} &= b_n \iff u_n = a_0 u_n + \sum_{k=1}^n a_k \cdot u_{n-k} + b_n \\
 \iff u_n &= \sum_{k=1}^n (r_{k-1} - r_k) u_{n-k} + b_n + a_0 u_n \\
 \iff u_n + \sum_{k=1}^n r_k \cdot u_{n-k} &= \sum_{k=1}^n r_{k-1} \cdot u_{n-k} + b_n + a_0 u_n \\
 \iff u_n + r_1 u_{n-1} + \dots + r_n u_0 &= r_0 u_{n-1} + r_1 u_{n-2} + \dots + r_{n-1} u_0 + b_n + a_0 u_n \\
 \iff (1-a_0) u_n + r_1 u_{n-1} + \dots + r_n u_0 &= r_0 u_{n-1} + \dots + r_{n-1} u_0 + b_n \\
 \iff r_0 u_n + r_1 u_{n-1} + \dots + r_n u_0 &= r_0 u_{n-1} + \dots + r_{n-1} u_0 + b_n
 \end{aligned}$$

Se $A_n = r_0 u_n + \dots + r_n u_0$, a relação anterior fica

$$A_n = A_{n-1} + b_n \quad \text{para } n = 1, 2, 3, \dots \text{ e } A_0 = r_0 u_0 = (1-a_0) u_0 = b_0$$

$$\begin{aligned}
 \text{Logo } A_1 &= A_0 + b_1 = b_0 + b_1 \\
 A_2 &= A_1 + b_2 = b_0 + b_1 + b_2 \\
 &\vdots \\
 &\vdots \\
 A_n &= \sum_{i=0}^n b_i
 \end{aligned}$$

De $r_n \geq 0$ e $u_n \geq 0$ obtemos para $N > 0$ fixo e $j > 0$ fixo ($n_j \geq N$)

$$\begin{aligned}
 r_0 u_{n_j} + r_1 u_{n_j-1} + \dots + r_N u_{n_j-N} &\leq r_0 u_{n_j} + r_1 u_{n_j-1} + \dots + r_N u_{n_j-N} + \dots + r_{n_j} u_0 \\
 &= A_{n_j} \\
 &= \sum_{i=0}^{n_j} b_i
 \end{aligned}$$

Fazendo $j \rightarrow \infty$ obtemos $(r_0 + r_1 + \dots + r_N) \lambda \leq \sum_{i=0}^{\infty} b_i$

$$\text{ou equivalentemente } \lambda = \frac{\sum_{i=0}^{\infty} b_i}{\sum_{n=0}^N r_n} \quad \text{para } N \text{ arbitrário}$$

Fazendo agora $N \rightarrow \infty$ obtemos

$$\lambda \leq \frac{\sum_{i=0}^{\infty} b_i}{\sum_{n=0}^{\infty} r_n} = \frac{\sum_{i=0}^{\infty} b_i}{\sum_{k=0}^{\infty} k a_k} \quad (4)$$

Se $\sum_{k=0}^{\infty} k a_k = \infty$ então a relação anterior implica em $\lambda \leq 0$.

Como $u_n \geq 0, \forall n$ e $\liminf_{n \rightarrow \infty} u_n \geq 0$ conclui-se que

$0 \geq \lambda = \limsup_{n \rightarrow \infty} u_n \geq \liminf_{n \rightarrow \infty} u_n \geq 0$ e consequentemente $\lim_{n \rightarrow \infty} u_n = 0$

Se $\sum_{k=0}^{\infty} k a_k < \infty$, seja $\mu = \liminf_{n \rightarrow \infty} u_n$. Raciocinando como no caso de \limsup , obtemos que se $\lim_{j \rightarrow \infty} u_{n_j} = \mu$ então $\lim_{j \rightarrow \infty} u_{n_j-d} = \mu$ para cada d inteiro não negativo.

Se $g(N) = \sum_{n=N+1}^{\infty} r_n$ então

$$\lim_{N \rightarrow \infty} g(N) = \lim_{N \rightarrow \infty} \sum_{n=N+1}^{\infty} r_n = 0 \quad \text{pois} \quad \sum_{n=N+1}^{\infty} r_n = \sum_{n=0}^{\infty} r_n - \sum_{n=0}^N r_n$$

$$\text{Assim, } \sum_{n=0}^j b_n = A_{n_j} = r_0 u_{n_j} + \dots + r_{n_j} u_0$$

$$= r_0 u_{n_j} + \dots + r_N u_{n_j-N} + r_{N+1} u_{n_j-N-1} + \dots + r_{n_j} u_0$$

$$\leq r_0 u_{n_j} + r_1 u_{n_j-1} + \dots + r_N u_{n_j-N} + M \cdot \sum_{n=N+1}^j r_n$$

$$\leq r_0 u_{n_j} + r_1 u_{n_j-1} + \dots + r_N u_{n_j-N} + M \cdot g(N)$$

Fazendo $j \rightarrow \infty$ obtemos

$$\sum_{n=0}^{\infty} b_n \leq (r_0 + r_1 + \dots + r_N) \mu + M \cdot g(N)$$

Fazendo agora $N \rightarrow \infty$ resulta

$$\sum_{n=0}^{\infty} b_n \leq \mu \cdot \sum_{n=0}^{\infty} r_n \quad \text{ou} \quad \mu \geq \frac{\sum_{n=0}^{\infty} b_n}{\sum_{n=0}^{\infty} r_n} \quad (5)$$

De (4) e (5) obtemos $\mu \geq \lambda$ e como $\mu \leq \lambda$ (por definição de μ e λ) resulta $\lim_{n \rightarrow \infty} \inf u_n = \lim_{n \rightarrow \infty} \sup u_n = \lambda = \mu$.

Assim fica provado que $\lim_{n \rightarrow \infty} u_n$ existe e que

$$\lim_{n \rightarrow \infty} u_n = \frac{\sum_{n=0}^{\infty} b_n}{\sum_{k=0}^{\infty} k a_k} \quad \text{quando} \quad \sum_{k=0}^{\infty} k a_k < \infty$$

Seguidamente mostramos como segue a prova no caso em que $a_1 = 0$ e $\text{MDC} \{ k / a_k > 0 \} = 1$.

Aquí, existem s_1, s_2, \dots, s_m inteiros positivos tais que $a_{s_1} > 0, a_{s_2} > 0, \dots, a_{s_m} > 0$ e $\text{MDC} \{ s_1, s_2, \dots, s_m \} = 1$

Por simplicidade na notação, vamos supor que existem s, t tais que $a_s > 0, a_t > 0$ e $\text{MDC} \{ s, t \} = 1$.

Seja $\lambda = \lim_{n \rightarrow \infty} \sup u_n < \infty$

Se $\{n_j\}$ é uma subsequência tal que $\lim_{j \rightarrow \infty} u_{n_j} = \lambda$ então pode-se

provar como no caso anterior que $\lim_{j \rightarrow \infty} u_{n_j-s} = \lambda$ e $\lim_{j \rightarrow \infty} u_{n_j-t} = \lambda$

Aplicando esse resultado temos $\lim_{j \rightarrow \infty} u_{n_j-s-t} = \lambda$

Repetindo sucessivamente se prova que $\lim_{j \rightarrow \infty} u_{n_j-\alpha s-\beta t} = \lambda$ para cada par de inteiros não negativos α e β .

Como $\text{MDC} \{ s, t \} = 1$ então existe N tal que para $N' \geq N, N' = \alpha s + \beta t$.

Usando o anterior, conclui-se que $\lim_{j \rightarrow \infty} u_{n_j-d} = \lambda, \forall d \geq 0$

A prova continua como foi desenvolvida aqui.