

UNIVERSIDADE FEDERAL DO PARANÁ

EMERSON BUTYN

PROGRAMAÇÃO LINEAR DETERMINÍSTICA E ESTOCÁSTICA
APLICADA AO PROBLEMA DE DESPACHO HIDROTÉRMICO.

CURITIBA

2017

UNIVERSIDADE FEDERAL DO PARANÁ

EMERSON BUTYN

PROGRAMAÇÃO LINEAR DETERMINÍSTICA E ESTOCÁSTICA
APLICADA AO PROBLEMA DE DESPACHO HIDROTÉRMICO.

Dissertação apresentada ao Programa de Pós Graduação em Matemática da Universidade Federal do Paraná, como requisito parcial à obtenção do grau de Mestre em Matemática. Orientadora: Dra. Elizabeth Wegner Karas.

CURITIBA

2017

B989p

Butyn, Emerson

Programação linear determinística e estocástica aplicada ao problema de despacho hidrotérmico / Emerson Butyn. – Curitiba, 2017.

115 f. : il. color. ; 30 cm.

Dissertação - Universidade Federal do Paraná, Setor de Ciências Exatas, Programa de Pós-Graduação em Matemática, 2017.

Orientador: Elizabeth Wegner Karas.

Bibliografia: p. 113-115.

1. Programação estocástica. 2. Programação dinâmica. 3. Energia elétrica. I. Universidade Federal do Paraná. II. Karas, Elizabeth Wegner III. Título.

CDD: 621.31



TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em MATEMÁTICA da Universidade Federal do Paraná foram convocados para realizar a arguição da Dissertação de Mestrado de **EMERSON BUTYN** intitulada: **PROGRAMAÇÃO LINEAR DETERMINÍSTICA E ESTOCÁSTICA APLICADA AO PROBLEMA DE DESPACHO HIDROTÉRMICO**, após terem inquirido o aluno e realizado a avaliação do trabalho, são de parecer pela sua APROVAÇÃO.

Curitiba, 17 de Fevereiro de 2017.

ELIZABETH WEGNER KARAS

Presidente da Banca Examinadora (UFPR)

CLOVIS CAESAR GONZAGA

Avaliador Externo (UFSC)

GISLAÍNE APARECIDA PERICARO

Avaliador Externo (UNESPAR)

DEBORA CINTIA MARCILIO

Avaliador Externo (LACTEC)

CLAUDIA ALEJANDRA SAGASTIZÁBAL

Avaliador Externo (IMPA)



ATA DE SESSÃO PÚBLICA DE DEFESA DE MESTRADO PARA A OBTENÇÃO DO GRAU DE MESTRE EM MATEMÁTICA

No dia dezessete de Fevereiro de dois mil e dezessete às 14:00 horas, na sala 300, bloco das PAs, Rua Cel. Francisco H. dos Santos, 100 - Jardim das Américas, foram instalados os trabalhos de arguição do mestrando **EMERSON BUTYN** para a Defesa Pública de sua Dissertação intitulada **PROGRAMAÇÃO LINEAR DETERMINÍSTICA E ESTOCÁSTICA APLICADA AO PROBLEMA DE DESPACHO HIDROTÉRMICO**. A Banca Examinadora, designada pelo Colegiado do Programa de Pós-Graduação em MATEMÁTICA da Universidade Federal do Paraná, foi constituída pelos seguintes Membros: ELIZABETH WEGNER KARAS (UFPR), CLOVIS CAESAR GONZAGA (UFSC), GISLAINE APARECIDA PERIÇARO (UNESPAR), DEBORA CINTIA MARCILIO (LACTEC), CLAUDIA ALEJANDRA SAGASTIZÁBAL (IMPA). Dando início à sessão, a presidência passou a palavra ao discente, para que o mesmo expusesse seu trabalho aos presentes. Em seguida, a presidência passou a palavra a cada um dos Examinadores, para suas respectivas arguições. O aluno respondeu a cada um dos arguidores. A presidência retomou a palavra para suas considerações finais e, depois, solicitou que os presentes e o mestrando deixassem a sala. A Banca Examinadora, então, reuniu-se sigilosamente e, após a discussão de suas avaliações, decidiu-se pela APROVAÇÃO do aluno. O mestrando foi convidado a ingressar novamente na sala, bem como os demais assistentes, após o que a presidência fez a leitura do Parecer da Banca Examinadora. Nada mais havendo a tratar a presidência deu por encerrada a sessão, da qual eu, ELIZABETH WEGNER KARAS, lavrei a presente ata, que vai assinada por mim e pelos membros da Comissão Examinadora.

Curitiba, 17 de Fevereiro de 2017.

ELIZABETH WEGNER KARAS

Presidente da Banca Examinadora (UFPR)

CLOVIS CAESAR GONZAGA

Avaliador Externo (UFSC)

GISLAINE APARECIDA PERIÇARO

Avaliador Externo (UNESPAR)

DEBORA CINTIA MARCILIO

Avaliador Externo (LACTEC)

CLAUDIA ALEJANDRA SAGASTIZÁBAL

Avaliador Externo (IMPA)

Agradecimentos

Agradeço, primeiramente, à minha família e aos meus amigos pela compreensão desta difícil fase.

À minha orientadora Profa Elizabeth que foi muito paciente e sempre procurou extrair o melhor de mim durante todo o desenvolvimento da dissertação. Além de críticas, sugestões e correções que contribuíram grandiosamente para o meu crescimento.

À Profa Gislaine que esteve grande parte do tempo me acompanhando e auxiliando com a escrita e com o desenvolvimento computacional dos algoritmos, além de ter aceito o convite para participar da banca da minha defesa.

Aos professores Clóvis e Débora que estiveram presentes em alguns momentos do desenvolvimento do trabalho em que tive a oportunidade de discutir sobre algoritmos e o problema de despacho hidrotérmico, agradeço suas sugestões e correções da dissertação como membros da banca.

À Profa Claudia pelo curso de Programação Estocástica ofertado pelo IMPA, que foi uma das minhas principais motivações para escolha do tema da dissertação. Agradeço também sua participação na banca com importantes críticas e contribuições.

Aos Institutos LACTEC em conjunto com a UFPR e COPEL pelo projeto de pesquisa envolvendo o planejamento hidrotérmico brasileiro, que também foi uma das razões pela escolha do tema da dissertação, onde estarei envolvido nele possibilitando a dar continuidade com os estudos no tema.

Ao Programa de Pós Graduação em Matemática da UFPR pela oportunidade e a formação de qualidade e à CAPES, pelo apoio financeiro.

Resumo

Neste trabalho apresentamos o problema de despacho hidrotérmico, cujo objetivo é encontrar a melhor decisão a ser tomada, em um dado horizonte de planejamento, de modo que a demanda de energia seja atendida ao menor custo esperado e que os reservatórios permaneçam acima de níveis críticos. Quatro exemplos distintos deste problema foram considerados. Eles foram modelados por funções e restrições lineares e resolvidos de forma direta e pelo algoritmo Programação Dinâmica Dual Determinística (PDDD), no caso determinístico, e pela Decomposição Aninhada e Programação Dinâmica Dual Estocástica (PDDE), no caso estocástico. Para descrever estas técnicas, apresentamos alguns conceitos e formulações de problemas lineares determinísticos e estocásticos.

Palavras-chave: Programação estocástica, programação determinística, programação dinâmica, despacho hidrotérmico, PDDE ou programação dinâmica dual estocástica, técnicas de decomposição, corte de Benders, planos cortantes.

Abstract

In this work we present the hydrothermal dispatch problem, whose objective is to find the best decision to be taken, in a given planning horizon, in such a way that the energy demand is supplied at the lowest expected cost, keeping the reservoirs above critical levels. Four distinct examples were considered. They were modeled by linear functions and constraints and solved in a direct way and by the Deterministic Dynamic Dual Programming (DDDP), in the deterministic case, and by Nested Decomposition and Stochastic Dynamic Dual Programming (SDDP), in the stochastic case. To describe these techniques, we present some concepts and formulations of linear deterministic and stochastic problems.

Keywords: Stochastic Programming, Deterministic Programming, Dynamic Programming, hydrothermal dispatch, Stochastic Dynamic Dual Programming, decomposition techniques, Benders cuts, cutting plane.

Sumário

1	Introdução	10
2	Programação linear determinística	16
2.1	Programação linear em dois estágios	16
2.1.1	Formulação geral	16
2.1.2	Formulação dinâmica - Algoritmo PDDD	17
2.1.3	Construção das aproximações da função recurso	23
2.2	Exemplo de problema de dois estágios	24
2.2.1	Modelagem do problema	24
2.2.2	Resolução pela formulação geral	27
2.2.3	Resolução pelo algoritmo PDDD	28
2.3	Programação linear em multiestágios	35
2.3.1	Formulação geral	35
2.3.2	Formulação dinâmica	36
3	Programação linear estocástica	41
3.1	Exemplo de motivação	41
3.2	Considerações gerais	42
3.3	Programação linear estocástica em dois estágios	43
3.3.1	Formulação geral	43
3.3.2	Formulação dinâmica	44
3.3.3	Aproximações da função recurso	47
3.3.4	Formulação equivalente determinística	52
3.3.5	Exemplo	53
3.4	Programação linear estocástica em multiestágios	56
3.4.1	Formulação aninhada	57
3.4.2	Árvore de cenários	58
3.4.3	Formulação dinâmica - Algoritmo PDDE	60

3.4.4	Formulação equivalente determinística	65
4	Resolução de problemas de despacho hidrotérmico pelo algoritmo PDDE	68
4.1	Sistema com 3 hidrelétricas	68
4.1.1	Descrição do sistema	68
4.1.2	Dados do sistema e do problema	69
4.1.3	Modelagem do problema	72
4.1.4	Resolução pelo algoritmo PDDE	74
4.2	Sistema com 7 hidrelétricas	79
4.2.1	Descrição do sistema	79
4.2.2	Dados do sistema	81
4.2.3	Modelagem do problema	82
4.2.4	Resolução pelo algoritmo PDDE	84
5	Conclusão	92
A	Conceitos básicos	93
A.1	Probabilidade	93
A.2	Análise	94
	Bibliografia	95

Capítulo 1

Introdução

O Brasil é um país que possui recursos hídricos em abundância. Diante disto, a principal fonte de geração de energia elétrica é proveniente das usinas hidrelétricas, que, por sua vez, tem sido responsável por 61% da geração de energia elétrica no país [1]. Em conjunto com as hidrelétricas, a segunda principal fonte de geração de energia são as usinas termelétricas, as quais utilizam como combustível carvão mineral, óleo, gás natural, entre outros recursos. O sistema composto por esses dois tipos de usinas denomina-se *sistema hidrotérmico*.

Pelo fato das usinas hidrelétricas possuírem grande capacidade de armazenamento de água, torna-se imprescindível administrar seu uso na operação do sistema. Isto se deve ao fato do reabastecimento dos reservatórios, pois depende de condições climáticas futuras. Deste modo é importante determinar um bom planejamento do uso da água dos reservatórios para a geração de energia, pois isto reflete diretamente nos custos para suprir a demanda de energia no futuro.

No Brasil, o problema da operação do Setor Elétrico Brasileiro (SEB) é subdividido em etapas, as quais estão sob o comando do Operador Nacional do Sistema Elétrico (ONS). Atualmente, o planejamento da operação energética do Sistema Interligado Nacional (SIN) é realizado por uma sequência de programas que foram desenvolvidos pelo Centro de Pesquisas de Energia Elétrica (CEPEL). Esses programas são responsáveis pelo planejamento do sistema hidrotérmico brasileiro e se dividem numa cadeia em que três modelos se destacam: NEWAVE, DECOMP e DESSEM. Em [2], [3], estes modelos correspondem ao planejamento da operação de médio prazo (horizonte de 5 anos com períodos mensais), de curto prazo (horizonte de 2 a 6 meses com períodos semanais) e diário (horizonte de 1 semana com períodos de meia hora), respectivamente.

Uma das dificuldades nestes planejamentos resulta das incertezas das afluições futuras que podem ser apenas estimadas. Isso traz consequências nas decisões a serem tomadas no presente momento, pois ao utilizar a água dos reservatórios, em excesso, seguido de períodos de estiagem, significa que haverá um futuro com altos custos para o suprimento de energia do mercado. Por outro lado, se for decidido armazenar água nos reservatórios para o futuro e ocorrer períodos de altas afluições, grande quantidade de água deverá ser vertida por causa dos limites máximos de armazenamento dos reservatórios. Neste último cenário deixa-se de aproveitar toda a eficiência das usinas hidrelétricas em termos de geração de energia elétrica. A Figura

1.1 sintetiza as consequências operativas das decisões tomadas diante de diferentes cenários de afluições nos reservatórios.

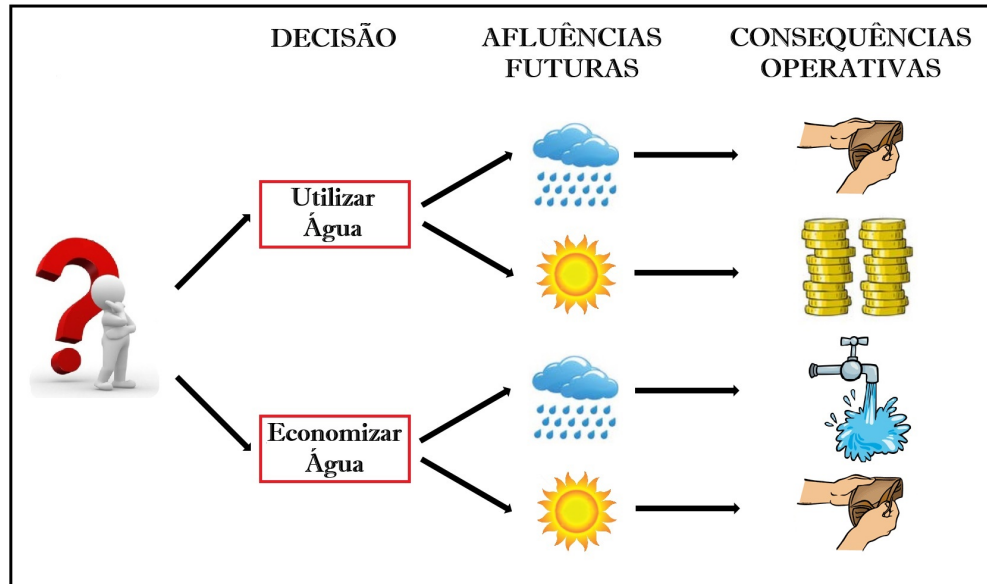


Figura 1.1: Esquema do processo de decisão sob incertezas

O problema de despacho hidrotérmico consiste em determinar uma estratégia operacional das usinas hidrelétricas (quantidade de água a ser turbinada e/ou vertida) e termelétricas (quantidade de energia a ser gerada) que minimize o custo total esperado da produção de energia elétrica em um determinado horizonte de tempo, de modo que algumas restrições sejam atendidas, tais como o atendimento à demanda, as restrições elétricas e hidrelétricas e os limites das variáveis. Refere-se aos custos os gastos com os combustíveis utilizados nas usinas termelétricas e um custo atribuído ao déficit, que corresponde a quantidade de energia que as usinas hidrelétricas e termelétricas não produziram para suprir a demanda.

Como este problema é modelado com planejamentos de diferentes horizontes de tempo, no caso brasileiro, não temos conhecimento perfeito dos acontecimentos que estão por vir, isto é, as futuras afluições e demandas. Problemas desta natureza são denominados *problemas estocásticos* e são estudados pela *programação estocástica*, a qual está envolvida em aspectos teóricos e práticos pelas áreas de otimização, estatística, probabilidade e computação.

A modelagem de problemas com esta característica apresenta incertezas em alguns de seus dados, os quais são representados por meio de funções que dependem de uma variável aleatória. Em geral, os problemas de otimização estocástica são baseados na minimização do valor esperado de uma função objetivo num espaço de probabilidades determinado.

Quando a variável aleatória é contínua, é muito raro obter uma representação analítica para o valor esperado da função objetivo. Além disto, o seu cálculo envolve, em muitas vezes, uma integral multidimensional, o que pode tornar o problema intratável computacionalmente. Uma forma de lidar com esta dificuldade é

discretizar a variável aleatória num número finito de possibilidades (realizações), com o qual o cálculo do valor esperado passa a ser um somatório da função objetivo ponderada pelas respectivas probabilidades das realizações da variável aleatória, referentes aos dados incertos do problema.

Estes problemas são modelados ao longo de um intervalo de tempo, ou seja, é determinado um *horizonte de planejamento*. Este, por sua vez, é dividido em vários períodos de tempo denominados *estágios*, onde as incertezas se revelam parcialmente. Estes dois elementos influenciam diretamente no tamanho do problema a ser tratado, além de ser uma característica que classifica os problemas, tanto determinísticos (aqueles em que todos os dados são conhecidos) quanto estocásticos, em dois estágios ou multiestágios. No caso de problemas estocásticos em multiestágios, é comum representar o conjunto de todas as possibilidades que a variável aleatória pode assumir por uma árvore de cenários, onde denomina-se *cenário* uma realização da variável aleatória que envolve unicamente todos os estágios do problema.

O objetivo da otimização estocástica em resolver um problema estocástico é determinar a melhor decisão a ser tomada diante de vários cenários. Estes cenários representam desde boas até más condições do sistema. Uma possibilidade é utilizar a formulação direta, que leva em consideração todos os cenários através de um problema determinístico, o qual pode ser resolvido por técnicas de otimização determinística. No entanto, quando há muitos cenários, resolver o problema por sua formulação direta pode ser inviável por causa de sua grande dimensão. Uma forma de contornar este obstáculo é utilizar técnicas de decomposição.

As técnicas de decomposição têm se mostrado computacionalmente atraentes. Elas permitem resolver o problema a partir de uma série de subproblemas menores associados ao problema original. Um exemplo disto são as técnicas de Programação Dinâmica Estocástica, as quais utilizam uma discretização das variáveis de estado do sistema e soluciona-se os problemas correspondentes de um determinado estágio para cada combinação dessas variáveis. O objetivo desta técnica é criar aproximações das funções que correspondem aos custos futuros do problema. Estes métodos permitem representar modelos não lineares complexos do sistema, no entanto, sofrem da conhecida *maldição da dimensionalidade* devido ao crescimento exponencial das discretizações das variáveis de estado. Segundo [4], as primeiras pesquisas em métodos de PDE se encontram em [5]–[8].

Esta técnica aplicada ao problema de despacho hidrotérmico brasileiro se limita por causa das enormes dimensões do problema quando os cinco subsistemas de geração de energia elétrica do Brasil são interligados. Deste modo, houve a necessidade de propor alternativas, ou melhorias para solucionar esta limitação. Deu-se então origem a várias pesquisas baseadas em métodos de decomposição de Benders [9].

Uma delas foi proposta em [10] e [11] e denomina-se Programação Dinâmica Dual Estocástica (PDDE), a qual finalmente superou o problema da maldição da dimensionalidade. Este é um método iterativo que combina técnicas de amostragem, onde normalmente se utiliza a simulação Monte Carlo, e decomposição, em que as soluções obtidas nos estágios são utilizadas recursivamente. O objetivo desta técnica é criar aproximações cada vez melhores das funções que representam os custos futuros por funções lineares por partes. Atualmente a PDDE é uma técnica considerada estado-da-arte na solução de problemas de planejamento de

despacho hidrotérmico, a qual se aplica nos programas NEWAVE e DECOMP, já mencionados [12]. Desde então várias pesquisas introduziram uma série de extensões, aprimoramentos e melhorias da PDDE.

A PDDE é um método baseado na adição dos cortes de Benders, os quais aparecem como restrições lineares de desigualdade no problema que aproxima o original. Deste modo, ela se limita ao uso de modelagens convexas dos problemas a serem resolvidos para garantir o critério de convergência. Diante desta limitação, em [4] são apresentadas técnicas de convexificação das funções custo futuro que permitem a extensão do algoritmo PDDE para a representação de não convexidades nos problemas. Além disto, utiliza-se o problema de planejamento da operação de sistemas hidrotérmicos como forma de testes, cuja formulação é dada a partir de um modelo multiestágios de programação linear-inteira mista.

Em [13] aborda-se o problema de planejamento de médio prazo, onde o sistema foi representado por reservatórios equivalentes de energia. Ao invés de utilizar a simulação Monte Carlo na PDDE, foram utilizadas técnicas de amostragem de árvore de cenários com redução da variância, que são *Latin Hypercube Sampling* e Quasi Monte Carlo Aleatório. Além disto, são tratadas no trabalho estratégias de solução e seleção de cortes para melhorar o desempenho da PDDE, metodologia para avaliação da qualidade da solução e também metodologias para obter uma solução com aversão a risco.

A inclusão de medidas de risco na PDDE é uma variação que tem como efeito garantir que a solução obtida pelo algoritmo esteja segura de possíveis eventos extremos como, por exemplo, um longo período de seca. Isto permite manter os níveis dos reservatórios acima de valores críticos, a ponto de evitar futuros *blackouts*. Como ponto negativo, as medidas de risco tornam o algoritmo mais carregado, visto que novas variáveis e/ou restrições devem ser acrescentadas na modelagem do problema, [14]–[16]. Outros trabalhos recentes que envolvem medidas de risco aplicados a PDDE são [17]–[21]. Neste último utiliza-se a configuração de horizonte-rolante, a qual lida com horizontes de planejamento cada vez menores ao longo do processo iterativo do algoritmo PDDE.

Segundo [22], os métodos baseados em cortes de Benders podem sofrer instabilidades no processo de otimização em iterações sucessivas do algoritmo, caracterizadas por pequenos decréscimos na função objetivo, mas com grandes diferenças nas variáveis de decisão. Além disso, não há um desenvolvimento teórico que permita a remoção de cortes mantendo a garantia de convergência do método. Com o objetivo de contornar estas desvantagens, em [22] é proposto um método de feixes inexato que permite considerar uma quantidade razoável de cenários, resolvendo-se os subproblemas resultantes de forma inexata.

Em [23], é apresentado um tutorial da técnica de *decomposição aninhada* aplicada ao problema de despacho hidrotérmico modelado em dois e três estágios. Esta técnica utiliza todo o conjunto de possibilidades da variável aleatória, o que pode limitar sua aplicabilidade para um problema que possui muitos cenários.

Em [3], é apresentada uma revisão de diferentes modelagens do problema de despacho hidrotérmico e metodologias que se diferenciam de acordo com a estrutura da modelagem. Além disso, é apresentada uma análise comparativa de diferentes estratégias de decomposição baseadas na relaxação Lagrangeana. Estas estratégias são aplicadas ao problema de programação diária da operação eletroenergética, modelado como

um problema de otimização estocástica em dois estágios. As estratégias de decomposição são avaliadas utilizando duas abordagens distintas, decomposição por cenários e decomposição espacial. Outra contribuição deste trabalho é a inclusão de incertezas no modelo matemático da PDE.

Considerando as usinas de forma individualizadas, o modelo de despacho hidrotérmico apresentado em [12] envolve funções não lineares. Para resolvê-lo, foi utilizado o método Lagrangeano Aumentado [24], que, iterativamente, aproxima o problema original de otimização restrita por subproblemas mais simples, com restrições de caixa. Em [12], esses subproblemas foram resolvidos pelo método do gradiente espectral projetado [25].

Em [26], a modelagem do problema de despacho hidrotérmico também considera as usinas hidrelétricas de forma individualizada. O método de pontos interiores primal-dual com barreira logarítmica foi utilizado na resolução do problema, desprezando-se a matriz Hessiana por motivo de sua grande esparsidade, sem afetar os resultados finais.

Objetivos

Este trabalho tem como objetivo apresentar uma revisão da programação linear determinística, com o intuito de estabelecer as relações básicas dos problemas em dois e multiestágios e apresentar um algoritmo de decomposição para facilitar o entendimento quando os dados incertos estão presentes nos problemas; e programação linear estocástica, a fim de dar suporte teórico e prático para aplicação de algoritmos na resolução de diferentes modelagens do problema de despacho hidrotérmico, nos casos determinístico e estocástico.

Diante dos problemas de dimensionalidade que facilmente aparecem em algumas abordagens, este trabalho tem como foco na aplicação do algoritmo PDDE no problema de despacho hidrotérmico modelado em curto prazo estendido, sendo o horizonte de planejamento de 1 ano com períodos mensais.

Organização do trabalho

Este trabalho se divide da seguinte maneira: no Capítulo 2, onde as principais referências foram [11] e [27], apresentamos algumas formulações dos problemas lineares determinísticos em dois e multiestágios, assim como o algoritmo Programação Dinâmica Dual Determinística (PDDD), que é uma técnica de decomposição. Além disto, duas abordagens diferentes foram utilizadas para resolver um mesmo problema de despacho hidrotérmico determinístico modelado em dois estágios, cujos dados foram obtidos de [23], sendo elas uma forma direta e o algoritmo PDDD. A principal motivação deste capítulo é facilitar o entendimento dos problemas quando estendidos para o caso estocástico.

No Capítulo 3 é incluída a estocasticidade nos problemas em dois e multiestágios trabalhados no Capítulo 2, assim como o algoritmo PDDD é estendido para o caso estocástico, obtendo o algoritmo PDDE, que é a metodologia aplicada no capítulo seguinte.

No Capítulo 4 são modelados dois problemas de despacho hidrotérmico em configurações diferentes: a

primeira configuração possui um sistema formado por 3 usinas hidrelétricas e 3 termelétricas. Já a segunda contém um sistema composto por 7 usinas hidrelétricas e uma termelétrica, sendo este sistema dividido em dois subsistemas. Ambos problemas foram resolvidos pelo algoritmo PDDE, cujas soluções obtidas foram expressas e discutidas através de gráficos.

Capítulo 2

Programação linear determinística

Programas determinísticos são problemas de otimização nos quais todos os dados de entrada da função objetivo e das restrições são conhecidos. Problemas desta natureza aparecem quando se deseja minimizar ou maximizar certa variável sujeita, ou não, a restrições de igualdade e/ou desigualdade. A partir de agora os problemas de programação linear determinísticos serão denominados apenas por problemas de programação linear, o que é comum na literatura.

O capítulo está dividido em três seções: Na Seção 2.1 será apresentado o problema de programação linear de dois estágios em duas formulações e o algoritmo Programação Dinâmica Dual Determinística (PDDD). Um exemplo de despacho hidrotérmico formulado em dois estágios será abordado na Seção 2.2 para ilustrar o uso da formulação geral e do algoritmo PDDD para resolvê-lo. Já uma extensão do problema e do algoritmo para um número maior de estágios será realizada na Seção 2.3. As referências utilizadas para o desenvolvimento das próximas seções são [11] e [27].

2.1 Programação linear em dois estágios

Nesta seção será apresentada a formulação do problema de programação linear em dois estágios de duas maneiras, uma delas denominada por *formulação geral* e a outra é a *formulação dinâmica*. A partir de algumas análises feitas na formulação dinâmica, será formulado o algoritmo PDDD, já mencionado.

2.1.1 Formulação geral

A formulação considerada *geral* do problema de programação linear em dois estágios que estamos interessados em resolver é dado por

$$\left\{ \begin{array}{ll} \text{minimizar} & c^T x + q^T y \\ \text{sujeito a} & Ax = b \\ & Tx + Wy = h \\ & x, y \geq 0, \end{array} \right. \quad (2.1)$$

onde $c \in \mathbb{R}^{n_1}$, $b \in \mathbb{R}^{m_1}$, $q \in \mathbb{R}^{n_2}$ e $h \in \mathbb{R}^{m_2}$ e as matrizes A, T e W possuem dimensões apropriadas de acordo com as dimensões dos vetores.

Veja que, nesta formulação, a primeira restrição do conjunto viável envolve somente a variável x e a segunda restrição envolve as duas variáveis, x e y . Isto sugere que (2.1) pode ser separado em duas partes, o que chamamos de estágios. O problema do primeiro estágio é definido como

$$\begin{cases} \text{minimizar} & c^T x \\ \text{sujeito a} & Ax = b \\ & x \geq 0. \end{cases} \quad (2.2)$$

Dadas as condições iniciais do problema pelos vetores b e c e a matriz A , podemos calcular uma solução \hat{x} para (2.2), que representa a decisão que deve ser tomada num momento inicial do planejamento. Em seguida, o novo problema consiste em lidar com as consequências desta decisão, donde se tem o problema de segundo estágio dado por

$$\begin{cases} \text{minimizar} & q^T y \\ \text{sujeito a} & Wy = h - T\hat{x} \\ & y \geq 0. \end{cases} \quad (2.3)$$

O problema de segundo estágio depende diretamente da decisão \hat{x} , a qual já é conhecida e, portanto, pode ser escrita no lado direito da restrição de igualdade de (2.3). Uma solução \hat{y} , associada ao problema (2.3), representa a decisão que deve ser tomada no futuro, lidando com as consequências da decisão \hat{x} do primeiro estágio. Como a decisão do primeiro estágio não leva em consideração o que irá ocorrer no futuro (segundo estágio), pode ser que as soluções \hat{x} e \hat{y} não sejam ótimas para o problema geral (2.1). Por fim, o objetivo é encontrar x e y viáveis tais que a soma $c^T x + q^T y$ seja mínima.

2.1.2 Formulação dinâmica - Algoritmo PDDD

Como foi visto na seção anterior, o problema de programação linear (2.1) pode ser separado em dois estágios, podemos utilizar a sua formulação dinâmica que é dada na seguinte forma:

$$\begin{cases} \text{minimizar} & c^T x + Q(x) \\ \text{sujeito a} & Ax = b \\ & x \geq 0, \end{cases} \quad (2.4)$$

onde a *função recurso* (em inglês, *recourse function*), $Q : \mathbb{R}^{n_1} \rightarrow \mathbb{R}$, é definida por

$$\begin{aligned} Q(x) = \min & \quad q^T y \\ \text{sujeito a} & \quad Wy = h - Tx \\ & \quad y \geq 0. \end{aligned} \quad (2.5)$$

Note que a função recurso (2.5) representa o valor ótimo do problema de segundo estágio (2.3) e, como observado anteriormente, depende diretamente da decisão x tomada no primeiro estágio. Se esta função for

conhecida, podemos resolver o problema (2.4) como o problema geral (2.1). Então, cabe aqui perguntar: qual é a vantagem de trabalhar com a formulação dinâmica sendo que a função recurso, a princípio, não é conhecida porque depende da decisão x ? A resposta desta pergunta está no fato de trabalharmos com apenas dois subproblemas com estrutura simples e que podemos empregar um método iterativo que visa obter, a cada iteração, aproximações cada vez melhores da função recurso.

Para obter aproximações para a função recurso, podemos utilizar algoritmos de programação dinâmica com a seguinte abordagem, como descrito em [11]: discretizar a variável x num conjunto $\{\hat{x}_i, i = 1, \dots, k\}$ e resolver o problema (2.5) para cada valor do conjunto, obtendo $\hat{y}_i, i = 1, \dots, k$. Os valores intermediários da função Q podem ser aproximados por uma interpolação da vizinhança dos valores $\hat{y}_i, i = 1, \dots, k$. Esta abordagem traz alguns benefícios, como a possibilidade de ser estendido facilmente para os casos multiestágios e estocástico e a possibilidade de lidar com casos não lineares.

Percebe-se que quanto mais fina for a discretização da variável do primeiro estágio, melhor será a aproximação da função Q . No entanto, essa discretização pode fazer com que o problema fique intratável. Se considerarmos, por exemplo, que $x \in \mathbb{R}^{10}$ e que cada componente é discretizada em 4 valores, teremos um total de 4^{10} discretizações, o que ultrapassa 1 milhão de possibilidades e limita a aplicabilidade desta abordagem, pois, além de precisar resolver o problema (2.5) mais de 4^{10} vezes, seria necessário fazer muitas interpolações dos valores obtidos. A este problema é dado o nome de *maldição da dimensionalidade*. Veja que este problema ocorreu para um exemplo simples, o que seria mais fácil de ocorrer em problemas maiores, como no caso multiestágios e estocástico, que serão vistos nas próximas seções e capítulo. Além disto, a região selecionada da variável de primeiro estágio para ser discretizada influencia diretamente na aproximação da função recurso, o que pode ter um efeito negativo quanto à qualidade da aproximação da função, pois é possível discretizar regiões onde a função assume valores muito altos.

Uma forma alternativa é construir a aproximação da função Q por funções analíticas, e não por um conjunto de valores discretizados. Será mostrado a seguir que a função recurso é uma função *linear por partes*. Para isto, utilizamos a formulação dual do problema (2.5), que é dada por

$$Q(x) = \max \quad \pi^T(h - Tx) \quad (2.6)$$

sujeito a $W^T \pi \leq q$.

Sabe-se pela teoria de programação linear que os valores ótimos dos problemas primal (2.5) e dual (2.6) são iguais, quando ambos forem viáveis. Neste caso, cada problema terá um conjunto não vazio de soluções ótimas.

Considere o conjunto $I(q) = \{\pi : W^T \pi \leq q\}$ dos vetores que satisfazem a condição de viabilidade do problema (2.6). Se este conjunto é não vazio, então ele é convexo, fechado, poliedral e, conseqüentemente, possui um número finito de vértices. Além disto, se $I(q)$ for limitado, então ele coincide com a combinação convexa de seus vértices. A partir disto, temos conhecimento que o conjunto de soluções ótimas é um vértice, uma aresta ou uma face [28, Cap. 13].

Um detalhe importante a se observar na formulação dual (2.6) é que a variável de decisão x do problema

de primeiro estágio (2.2) se encontra na função objetivo, e não mais no conjunto viável do problema. Isto significa que o conjunto de todas as possíveis soluções, as quais correspondem aos vértices do conjunto viável $I(q)$, pode ser identificado antes de conhecer a decisão do problema de primeiro estágio.

Considere então que o conjunto

$$\Pi = \{\pi_1, \pi_2, \dots, \pi_n\}, \quad (2.7)$$

que representa todos os vértices do conjunto viável do problema (2.6), seja conhecido. Num primeiro momento, como a solução do problema pertence a este conjunto, poderíamos encontrá-la por enumeração, isto é,

$$Q(x) = \max\{\pi_i^T(h - Tx), \forall i = 1, \dots, n\}. \quad (2.8)$$

Pela expressão (2.8), a função recurso é o máximo de funções lineares, ou seja, é uma função linear por partes, como ilustra a Figura 2.1. A expressão (2.8) também pode ser reescrita como um problema de programação linear:

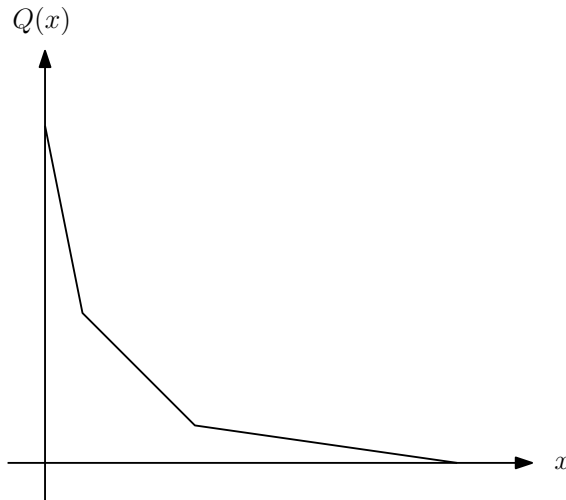


Figura 2.1: Função linear por partes

$$\begin{aligned} Q(x) = \min \quad & r \\ \text{sujeito a} \quad & r \geq \pi_1^T(h - Tx) \\ & \vdots \\ & r \geq \pi_n^T(h - Tx), \end{aligned} \quad (2.9)$$

onde r é uma variável escalar. Note que os problemas (2.8) e (2.9) são equivalentes, pois como $r \geq \pi_i^T(h - Tx)$ para todo $i = 1, \dots, n$ no problema (2.9), então $r \geq \max\{\pi_i^T(h - Tx), i = 1, \dots, n\}$. Além disto, como o problema (2.9) consiste em minimizar o valor de r , o valor ótimo será $\max\{\pi_i^T(h - Tx), i = 1, \dots, n\}$. Isto nos diz que é possível conhecer como é a função recurso sem precisar discretizar a variável x do primeiro estágio, ou seja, basta conhecer os coeficientes $\{\pi_i\}$ dos hiperplanos suportes.

Visto isso, podemos nos preocupar agora em como determinar estes vértices $\{\pi_i\}$, ou uma parte, do conjunto (2.7) para a construção da função recurso, ou uma aproximação da mesma. Uma forma de obter estes vértices é calculando as variáveis duais do problema de segundo estágio (2.5), utilizando vetores arbitrários x_i no lugar da variável x .

Seja então π_i um vetor multiplicador simplex associado às restrições do problema (2.5). Sabe-se da teoria de programação linear que este vetor é um dos vértices do conjunto de soluções (2.7) do problema dual (2.6), isto é, ele pode ser utilizado para construir um dos hiperplanos suportes da função recurso. Podemos seguir este procedimento para um conjunto $\{x_i, i = 1, \dots, m\}$ de vetores arbitrários e resolver (2.5) a fim de obter um subconjunto $\Pi_m = \{\pi_1, \dots, \pi_m\}$ de (2.7). Desta maneira podemos utilizar o subconjunto Π_m para construir uma aproximação da função recurso denotada por \hat{Q} e definida como:

$$\begin{aligned} \hat{Q}(x) = \min \quad & r \\ \text{sujeito a} \quad & r \geq \pi_i^T(h - Tx), \quad \text{para } i = 1, \dots, m. \end{aligned} \quad (2.10)$$

Veja que essa aproximação \hat{Q} é um *limite inferior* da função recurso Q , pois como as restrições de (2.10) são construídas a partir do subconjunto Π_m , que está contido em Π , somente um subconjunto das restrições farão parte do problema (2.9), concluindo-se então que $Q(x) \geq \hat{Q}(x)$, para todo x . Podemos utilizar esta aproximação da função recurso no problema de primeiro estágio (2.4), ficando

$$\left\{ \begin{array}{ll} \text{minimizar} & c^T x + \hat{Q}(x) \\ \text{sujeito a} & Ax = b \\ & x \geq 0, \end{array} \right. \quad (2.11)$$

onde, substituindo $\hat{Q}(x)$ como em (2.9), obtemos o problema de programação linear

$$\left\{ \begin{array}{ll} \text{minimizar} & c^T x + r \\ \text{sujeito a} & Ax = b \\ & r \geq \pi_1^T(h - Tx) \\ & \vdots \\ & r \geq \pi_m^T(h - Tx) \\ & x \geq 0. \end{array} \right. \quad (2.12)$$

Como está sendo utilizada uma aproximação da função recurso, não é possível afirmar que uma solução ótima de (2.12) seria uma solução ótima do problema de dois estágios (2.1). No entanto, sabemos que \hat{Q} é um limite inferior da função recurso Q . Logo o valor ótimo do problema (2.12) é um *limitante inferior* \underline{z} do verdadeiro valor ótimo de (2.1), isto é,

$$\underline{z} = c^T \hat{x} + \hat{r} = c^T \hat{x} + \hat{Q}(\hat{x}), \quad (2.13)$$

onde (\hat{x}, \hat{r}) é uma solução ótima do problema aproximado (2.12). Além disto podemos calcular um *limite*

superior \bar{z} do problema de dois estágios (2.1) resolvendo o problema de segundo estágio (2.5), para este mesmo \hat{x} , obtendo

$$\bar{z} = c^T \hat{x} + Q(\hat{x}). \quad (2.14)$$

Para verificar a precisão da aproximação da função recurso \hat{Q} em relação a Q , pode-se utilizar a diferença entre os limites superior e inferior

$$\bar{z} - \underline{z} = c^T \hat{x} + Q(\hat{x}) - (c^T \hat{x} + \hat{r}) = Q(\hat{x}) - \hat{r}, \quad (2.15)$$

pois ela mede a diferença entre a função recurso *real* (dada por $Q(\hat{x})$) e a função recurso *predita* (dada por \hat{r}) no ponto \hat{x} . Se esta diferença for menor que uma dada tolerância $\varepsilon > 0$, o problema está resolvido. Caso contrário, é preciso melhorar a aproximação da função recurso selecionando um outro conjunto de decisões tentativas para determinar novos vértices $\{\pi_i\}$. Podemos descrever este processo resumidamente, por meio do seguinte algoritmo:

Algoritmo 1

Passo (1). Selecionar um conjunto de m pontos arbitrários $\{\hat{x}_i, i = 1, \dots, m\}$, definir tolerância $\varepsilon > 0$;

Passo (2). Para cada decisão tentativa \hat{x}_i , resolver o problema de segundo estágio (2.5) e calcular um multiplicador π_i associado;

Passo (3). Utilizar o subconjunto de multiplicadores $\Pi_m = \{\pi_1, \dots, \pi_m\}$ para construir uma aproximação da função recurso \hat{Q} , como em (2.10). Resolver o problema de primeiro estágio aproximado (2.12);

Passo (4). Calcular os limites inferior \underline{z} e superior \bar{z} , como em (2.13) e (2.14). Se $\bar{z} - \underline{z}$ for menor que uma tolerância $\varepsilon > 0$ dada, pare o algoritmo. Caso contrário, volte ao Passo (1).

Um último detalhe a ser visto ainda é como determinar o conjunto inicial de decisões arbitrárias $\{\hat{x}_i\}$ para aproximar a função recurso. Uma ideia seria iniciar o algoritmo utilizando somente uma decisão inicial, sendo ela a solução ótima do problema de primeiro estágio (2.2), pois ainda não há aproximações da função recurso para utilizar (2.12). No decorrer do algoritmo podemos obter uma nova decisão para cada iteração, sendo ela a solução ótima do problema aproximado (2.12), a qual já inclui as aproximações da função recurso obtidas dos valores arbitrários anteriores. Este procedimento nos assegura de estarmos construindo aproximações numa vizinhança de interesse, ou seja, onde estão localizados pontos que são bons candidatos para uma solução ótima.

O algoritmo composto por estes detalhes descritos anteriormente é denominado por *Programação Dual Dinâmica Determinística (PDDD)* de dois estágios [29], onde seus passos, com algumas ilustrações, são apresentados a seguir.

Algoritmo 2 - PDDD em dois estágios

Passo (1). Inicialização: Definir função recurso aproximado $\hat{Q}(x) = 0$, limite superior $\bar{z} = \infty$, número de vértices $m = 0$ e tolerância $\varepsilon > 0$;

Passo (2). (Fase progressiva¹) Resolver o problema de primeiro estágio aproximado (2.11). Tome \hat{x} como a decisão tentativa. Este passo está ilustrado na Figura 2.2, em que a seta indica que a solução obtida no primeiro estágio ($t = 1$) será utilizada no segundo estágio ($t = 2$);

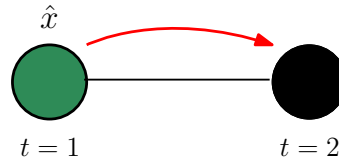


Figura 2.2: Fase progressiva

Passo (3). Calcular o limite inferior \underline{z} como em (2.13). Se $\bar{z} - \underline{z} \leq \varepsilon$, parar o algoritmo. Caso contrário, ir para o Passo (4);

Passo (4). Resolver o problema de segundo estágio (2.5), isto é, calcular $Q(\hat{x})$. Atualizar \bar{z} como em (2.14);

Passo (5). (Fase recursiva²) Atualizar o número de vértices $m \leftarrow m + 1$. Considere o multiplicador associado à solução ótima do Passo (4) como π_m . Construa a aproximação da função recurso $\hat{Q}(x)$ como em (2.10), utilizando m vértices. Este passo está ilustrado na Figura 2.3, onde a linha na diagonal que sobrepõe o primeiro círculo indica um dos hiperplanos que aproxima a função recurso;

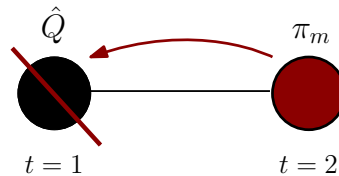


Figura 2.3: Fase recursiva

Passo (6). Ir para o Passo (2).

Esta versão de dois estágios do algoritmo PDDD é equivalente ao algoritmo decomposição de Benders [9]. Uma observação interessante é que entre duas iterações consecutivas aumenta somente uma restrição linear associada ao novo vértice calculado no Passo (5). Além disto, o algoritmo tem algumas características atraentes:

- Não é necessário discretizar a variável do primeiro estágio;
- Os limites inferior e superior são atualizados a cada iteração;

¹Encontra-se este termo na literatura como *recursão direta* e também, em inglês, como *forward step*.

²Encontra-se este termo na literatura como *recursão inversa* e também, em inglês, como *backward step*.

- Em uma dada iteração, a solução ótima do problema aproximado (2.11) do Passo (2) pode ser utilizada como ponto inicial na próxima iteração do algoritmo.

Um último detalhe a ser visto do algoritmo PDDD é a forma que as aproximações da função recurso são construídas a partir dos multiplicadores de Lagrange associados às restrições do problema (2.5). Estas aproximações são denominadas, na literatura, como *cortes de Benders* e, a partir de agora, serão chamadas simplesmente por *cortes*. Será explicado agora como podemos obter uma expressão implementável para a inclusão destes no problema.

2.1.3 Construção das aproximações da função recurso

Como foi dito anteriormente, sabemos que os valores ótimos dos problemas primal (2.5) e dual (2.6) são iguais. Consideramos então π^* um multiplicador simplex associado às restrições do problema (2.5), que é uma solução de (2.6), cujo valor ótimo é $\pi^{*T}(h - Tx)$. Assim como foi apresentado em (2.9), acrescentamos a restrição $\pi^{*T}(h - Tx) - r \leq 0$ ao problema (2.12) a cada iteração, a qual denomina-se por corte.

Para facilitar a inclusão dos cortes como restrições nos problemas nas próximas seções, denotamos seus coeficientes linear e angular, α e β respectivamente, por

$$\alpha = \pi^{*T}h \quad \text{e} \quad \beta = -\pi^{*T}T. \quad (2.16)$$

Para tratar a inclusão da restrição do corte de Benders computacionalmente, consideramos x^* e π^* as soluções ótimas dos problemas (2.5) e (2.6), cujos valores ótimos são f^* e $\pi^{*T}(h - Tx)$, respectivamente. Como os problemas primal e dual de segundo estágio possuem valores ótimos iguais, temos que:

$$\begin{aligned} \pi^{*T}(h - Tx^*) &= f^* \\ \pi^{*T}h - \pi^{*T}Tx^* &= f^* \\ \pi^{*T}h &= f^* + \pi^{*T}Tx^*. \end{aligned} \quad (2.17)$$

Substituindo (2.17) na desigualdade $\pi^{*T}(h - Tx) - r \leq 0$, obtemos:

$$\begin{aligned} \pi^{*T}(h - Tx) - r &\leq 0 \\ \pi^{*T}h - \pi^{*T}Tx - r &\leq 0 \\ (f^* + \pi^{*T}Tx^*) - \pi^{*T}Tx - r &\leq 0 \\ -\pi^{*T}Tx - r &\leq -f^* - \pi^{*T}Tx^*, \end{aligned} \quad (2.18)$$

onde a última desigualdade de (2.18) está no formato a ser implementado.

Na próxima seção será apresentado um exemplo a fim de ilustrar sua resolução pela formulação geral e também pelo algoritmo PDDD. Exibimos as matrizes e os cálculos de todas as iterações até a convergência, além de alguns detalhes computacionais.

2.2 Exemplo de problema de dois estágios

O exemplo trata de um problema de despacho hidrotérmico composto por uma usina hidrelétrica e quatro usinas termelétricas, cujos dados foram obtidos de [23]. Fazemos sua modelagem exibindo as variáveis, a função objetivo e as restrições. O problema a ser resolvido é um problema de programação linear de dois estágios, o qual será resolvido por duas abordagens diferentes: a partir de sua formulação geral (2.1), que é uma forma direta de resolver o problema, e o algoritmo PDDD descrito anteriormente.

2.2.1 Modelagem do problema

Primeiramente definimos algumas nomenclaturas na Tabela 2.1 necessárias para a modelagem e resolução do problema em sua formulação geral:

Nomenclatura	Descrição
T	Número total de estágios
t	Índice associado ao estágio $t \in \{1, \dots, T\}$
v_t	Volume do reservatório no final do estágio t
q_t	Vazão turbinada no estágio t em m^3/s
s_t	Vazão vertida no estágio t em m^3/s
d_t	Déficit de energia no estágio t em MWmês
a_t	Afluência no estágio t em m^3/s
ρ	Produtividade da hidrelétrica em MWmês/(m^3/s)
$gt_t = \rho \cdot q_t$	Geração da hidrelétrica no estágio t em MWmês
$gt_{i,t}$	Geração da usina térmica i no estágio t em MWmês
ct_i	Custo de produção associado à usina térmica i em R\$/MWmês
cd	Custo associado ao déficit de energia em R\$/mês
κ	Constante que converte m^3/s em $hm^3/mês$
D	Demanda de energia do sistema em MWmês

Tabela 2.1: Nomenclaturas

Os dados do exemplo estão nas três tabelas a seguir. A Tabela 2.2 fornece informações que indicam as condições iniciais do problema. Um dado importante a ser observado é a quantidade de estágios que serão utilizados para modelar este problema. Como o horizonte de planejamento são 2 meses, utilizamos uma

modelagem em dois estágios, isto é, o problema consiste em determinar uma decisão a ser tomada agora (mês atual) considerando, ainda, o mês seguinte, que minimize o custo total operativo de geração de energia a fim de atender a demanda do sistema. Note que a quantidade de estágios e o horizonte de planejamento coincidiram porque estamos considerando que cada estágio corresponde a um respectivo mês, o que poderia ser feito de outra forma como, por exemplo, um estágio corresponder a uma quinzena, gerando um problema em 4 estágios. Na Tabela 2.3 são apresentados os dados referentes aos custos e capacidades das usinas térmicas e na Tabela 2.4, as informações referentes às limitações físicas da usina hidrelétrica.

Horizonte de planejamento	2 meses
Número de estágios	2
Demanda do sistema	1000 MWmês
Custo do déficit	R\$ 500,00/MWmês
Afluência no primeiro estágio	150 m ³ /s
Afluência no segundo estágio	450 m ³ /s
Volume inicial do reservatório	2050 hm ³

Tabela 2.2: Dados do problema

Usina térmica	Custo (R\$/MWmês)	Geração máxima (MWmês)
UT1	10	100
UT2	20	150
UT3	40	200
UT4	80	250

Tabela 2.3: Dados das usinas térmicas

Volume máximo (hm ³)	4100
Vazão turbinada máxima (m ³ /s)	1500
Produtividade ρ (MWmês/(m ³ /s))	0,96
Geração máxima (MWmês)	1440
Constante de conversão κ (m ³ /s \rightarrow hm ³ /mês)	2,592

Tabela 2.4: Dados da usina hidrelétrica

Além destas informações, segue que o volume mínimo (v_{min}), as vazões vertida (s_{min}) e turbinada (q_{min}) mínimas serão considerados nulos e a vazão vertida máxima (s_{max}) é ilimitada, isto é, $v_{min} = s_{min} = q_{min} = 0$ e $s_{max} = \infty$. A produção de energia da usina hidrelétrica e o custo de cada usina térmica no estágio t serão dados pelos produtos $\rho \cdot q_t$ e $ct_i \cdot gt_{i,t}$, $i = 1, \dots, 4$, $t = 1, 2$, respectivamente.

A partir de agora focamos na modelagem e na resolução do problema determinístico de despacho hidro-térmico, onde o significado das componentes dos vetores e matrizes se encontra na Tabela 2.1. Segue então o vetor de decisão do problema, a função objetivo e três tipos de restrições: de balanço hídrico, de atendimento à demanda e das limitações das usinas:

- Vetor decisão:

$$x = [v_1, v_2, q_1, q_2, s_1, s_2, d_1, d_2, gt_{1,1}, gt_{2,1}, gt_{3,1}, gt_{4,1}, gt_{1,2}, gt_{2,2}, gt_{3,2}, gt_{4,2}]^T; \quad (2.19)$$

- Função objetivo: soma dos custos das usinas térmicas e do déficit de energia;

$$f(x) = c^T x = \sum_{i=1}^4 ct_i \cdot gt_{i,1} + cd \cdot d_1 + \sum_{i=1}^4 ct_i \cdot gt_{i,2} + cd \cdot d_2, \quad (2.20)$$

onde $c = [0, 0, 0, 0, 0, 0, cd, cd, ct_1, ct_2, ct_3, ct_4, ct_1, ct_2, ct_3, ct_4]$;

- Balanço hídrico: o volume atualizado do reservatório é a diferença entre o volume inicial adicionado à afluência e a soma das vazões vertida e turbinada. Note que as variáveis q_i , s_i e a_i , $i = 1, 2$, estão sendo multiplicadas por κ devido à conversão das unidades de medida;

$$\begin{aligned} v_1 &= 2050 + \kappa \cdot a_1 - \kappa(q_1 + s_1) \\ v_2 &= v_1 + \kappa \cdot a_2 - \kappa(q_2 + s_2); \end{aligned} \quad (2.21)$$

- Atendimento à demanda: a soma da geração de energia das usinas termelétricas e hidrelétricas junto com o déficit é igual a demanda;

$$\begin{aligned} \sum_{i=1}^4 gt_{i,1} + d_1 + \rho \cdot q_1 &= D \\ \sum_{i=1}^4 gt_{i,2} + d_2 + \rho \cdot q_2 &= D; \end{aligned} \quad (2.22)$$

- Limitações das variáveis:

$$\begin{aligned} 0 &\leq v_1, v_2 \leq v_{max} \\ 0 &\leq q_1, q_2 \leq q_{max} \\ s_1, s_2 &\geq 0 \\ 0 &\leq gt_{i,1}, gt_{i,2} \leq gt_{i(max)}, \quad i = 1, \dots, 4 \\ d_1, d_2 &\geq 0. \end{aligned} \quad (2.23)$$

2.2.2 Resolução pela formulação geral

O problema a ser resolvido, em termos computacionais, corresponde ao problema (2.1) adaptado, como segue:

$$\left\{ \begin{array}{l} \text{minimizar} \\ \text{sujeito a} \end{array} \right. \begin{array}{l} c^T x \\ \left[\begin{array}{cc} A \\ T + W \end{array} \right] x = \left[\begin{array}{c} b \\ h \end{array} \right] \\ \ell \leq x \leq u. \end{array} \quad (2.24)$$

Note que há somente um vetor decisão x . Isto ocorre porque resolver um problema em sua formulação geral, neste caso, é o mesmo que resolver um problema com os dois estágios agrupados. No entanto, quando utilizamos o algoritmo PDDD isto já não ocorre mais. Além disto, as componentes de primeiro e segundo estágios da variável x não estão separadas por estágio, como pode ser visto em (2.19). Se as componentes de primeiro e segundo estágios da variável x , na modelagem, estivessem separadas, a matriz em blocos das restrições de (2.24) ficaria na forma

$$\left[\begin{array}{cc} A & 0 \\ T & W \end{array} \right],$$

a qual corresponde exatamente à formulação do problema (2.1), sendo $[x \ y]^T$ a variável do problema. Note que o elemento 0 desta matriz em blocos corresponde a uma matriz nula de dimensão apropriada.

Para determinar qual é a decisão ótima do primeiro e segundo estágios do problema (2.24) devemos considerar as componentes onde $t = 1$ e $t = 2$, respectivamente, do vetor solução x , dado em (2.19). Usando as informações das Tabelas 2.2, 2.3 e 2.4, temos que as matrizes e vetores correspondentes ao problema (2.24) são dadas por:

$$c = \left[0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 500 \ 500 \ 10 \ 20 \ 40 \ 80 \ 10 \ 20 \ 40 \ 80 \right]^T$$

$$A = \left[\begin{array}{cccccccccccccccc} 1 & 0 & \kappa & 0 & \kappa & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \rho & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \end{array} \right]$$

$$T = \left[\begin{array}{cccccccccccccccc} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array} \right]$$

$$W = \left[\begin{array}{cccccccccccccccc} 0 & 1 & 0 & \kappa & 0 & \kappa & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \rho & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{array} \right]$$

$$b = \left[\begin{array}{c} 2050 + \kappa \cdot a_1 \\ D \end{array} \right]$$

$$h = \left[\begin{array}{c} \kappa \cdot a_2 \\ D \end{array} \right].$$

Veja que a matriz A possui elementos não nulos apenas nas componentes de primeiro estágio, correspondendo a primeira linha das restrições de balanço hídrico (2.21) e atendimento a demanda (2.22). A matriz T possui um único elemento não nulo, neste caso, que está associado ao volume do primeiro estágio v_1 da segunda linha das restrições de balanço hídrico (2.21). A matriz T , juntamente com a matriz W , que está associada somente às componentes de segundo estágio, correspondem às restrições de balanço hídrico e atendimento à demanda do segundo estágio. Também tem-se as limitações inferior ℓ e superior u das variáveis:

$$\ell = \left[0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \right]^T$$

$$u = \left[4100 \ 4100 \ 1500 \ 1500 \ \infty \ \infty \ \infty \ \infty \ 100 \ 150 \ 200 \ 250 \ 100 \ 150 \ 200 \ 250 \right]^T.$$

Para resolver o problema (2.24) de programação linear foi utilizada a função "linprog" (com configuração padrão) no software Matlab, cuja versão usada foi R2015a. Ao resolver o problema com estes dados, obtemos a solução aproximada

$$x = \left[630,67 \ 0 \ 697,58 \ 693,32 \ 0 \ 0 \ 0 \ 0 \ 100 \ 150 \ 80,32 \ 0 \ 100 \ 150 \ 84,42 \ 0 \right]^T,$$

onde o valor ótimo da função objetivo foi R\$ 14.589,63.

Separando a solução x obtida, a qual representa os estágios agrupados, em duas soluções, cada uma referente a um estágio, sendo x para o primeiro e y para o segundo, obtemos os seguintes vetores:

$$x = \begin{bmatrix} v_1 \\ q_1 \\ s_1 \\ d_1 \\ gt_{1,1} \\ gt_{2,1} \\ gt_{3,1} \\ gt_{4,1} \end{bmatrix} = \begin{bmatrix} 630,67 \\ 697,58 \\ 0 \\ 0 \\ 100 \\ 150 \\ 80,32 \\ 0 \end{bmatrix} \quad \text{e} \quad y = \begin{bmatrix} v_2 \\ q_2 \\ s_2 \\ d_2 \\ gt_{1,2} \\ gt_{2,2} \\ gt_{3,2} \\ gt_{4,2} \end{bmatrix} = \begin{bmatrix} 0 \\ 693,32 \\ 0 \\ 0 \\ 100 \\ 150 \\ 84,42 \\ 0 \end{bmatrix}.$$

Uma observação sobre a solução obtida de segundo estágio y é o volume que restou do reservatório, que foi $v_2 = 0$, ou seja, ele se esvaziou. Era de se esperar que isto ocorresse, visto que o custo foi minimizado sem se preocupar com o nível do reservatório ao fim da otimização. Veja que, para este problema, essa é uma das razões pela qual se evitou a ocorrência de déficit e o uso da usina termelétrica T_4 , a mais cara, nos dois estágios.

2.2.3 Resolução pelo algoritmo PDDD

Apresentamos esta resolução de forma detalhada para ilustrar o algoritmo PDDD. A sequência deste procedimento terá como referência [30]. Antes de começarmos, precisamos separar o problema em dois estágios, pois

o resolvemos utilizando os problemas (2.11), (2.12) (1º estágio) e (2.5) (2º estágio). Além disto, é necessário incluir a variável r e sua respectiva restrição que é acrescentada iterativamente, em referência aos cortes.

Introduzimos algumas nomenclaturas na Tabela 2.5, as quais incluem as características do algoritmo, como as aproximações da função recurso \hat{Q} , que serão denominadas por *cortes*. Além disto, redefinimos as matrizes e os vetores associados ao problema:

r	Variável associada a função recurso aproximada
ε	Tolerância do critério de parada
\underline{z}	Limite inferior determinado por (2.13)
\bar{z}	Limite superior determinado por (2.14)
π_k	Multiplicador simplex associado ao problema (2.5) na iteração k

Tabela 2.5: Nomenclaturas

$$\begin{aligned}
 x &= [v_1, q_1, s_1, gt_{1,1}, gt_{2,1}, gt_{3,1}, gt_{4,1}, d_1, r_1]^T \\
 y &= [v_2, q_2, s_2, gt_{1,2}, gt_{2,2}, gt_{3,2}, gt_{4,2}, d_2, r_2]^T \\
 c &= [0 \ 0 \ 0 \ 10 \ 20 \ 40 \ 80 \ 500 \ 1]^T \\
 A = W &= \begin{bmatrix} 1 & \kappa & \kappa & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \rho & 0 & 1 & 1 & 1 & 1 & 1 & 0 \end{bmatrix} \\
 T &= \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\
 b &= \begin{bmatrix} 2050 + \kappa \cdot a_1 \\ D \end{bmatrix} \quad h = \begin{bmatrix} \kappa \cdot a_2 \\ D \end{bmatrix} \\
 \ell &= [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]^T \\
 u &= [4100 \ 1500 \ \infty \ 100 \ 150 \ 200 \ 250 \ \infty \ \infty]^T.
 \end{aligned}$$

Veja que, em comparação à resolução anterior, as dimensões das matrizes A, T e W são determinadas pela dimensão da variável em seu respectivo estágio, o que não ocorreu quando as componentes de primeiro e segundo estágios não estavam separadas. Nesta modelagem, a matriz A corresponde às restrições de balanço hídrico e atendimento a demanda do primeiro estágio e as matrizes T e W correspondem a estas mesmas restrições referentes ao segundo estágio. Feitas todas estas considerações, damos início à resolução do exemplo pelo algoritmo PDDD.

1ª iteração:

Passo (1): Definimos $\hat{Q}(x) = r = 0$, $\bar{z} = \infty$, $n = 0$ e $\varepsilon = 10^{-3}$;

Passo (2): Resolvemos o problema aproximado (2.11), onde a restrição $x \geq 0$ é substituída por $\ell \leq x \leq u$. Obtemos a solução \hat{x}_1 , onde utilizamos um índice sobrescrito nas componentes da variável para associar à iteração do algoritmo:

$$\hat{x}_1 = \begin{bmatrix} v_1^{(1)} \\ q_1^{(1)} \\ s_1^{(1)} \\ gt_{1,1}^{(1)} \\ gt_{2,1}^{(1)} \\ gt_{3,1}^{(1)} \\ gt_{4,1}^{(1)} \\ d_1^{(1)} \\ \hat{r}_1^{(1)} \end{bmatrix} = \begin{bmatrix} 0,00 \\ 940,90 \\ 0,00 \\ 96,74 \\ 0,00 \\ 0,00 \\ 0,00 \\ 0,00 \\ 0,00 \end{bmatrix} .$$

Passo (3): Calculamos o limite inferior \underline{z} e testamos o critério de parada.

$$\begin{aligned} \underline{z} &= c^T \hat{x}_1 + \hat{r}_1^{(1)} \\ &= 967,41 + 0 \\ &= 967,41 \end{aligned}$$

$$\bar{z} - \underline{z} = \infty - 967,41 > \varepsilon.$$

Logo, o critério não foi satisfeito e então seguimos ao próximo passo.

Passo (4): Utilizando \hat{x}_1 , resolvemos o problema (2.5) e obtemos $Q(\hat{x}_1) = 21.440$. Assim, atualizamos $\bar{z} = c^T \hat{x}_1 + Q(\hat{x}_1) = 967,41 + 21.440 = 22.407,41$.

Passo (5): Atualizamos o número de vértice para $n = 0 + 1 = 1$. O multiplicador simplex π_1 , associado às restrições de (2.5), obtido no passo anterior, é:

$$\pi_1 = [29,63 \quad -80,00]^T.$$

Exibimos o passo a passo da construção do corte somente nesta iteração e nas próximas iterações mostramos apenas sua expressão. Primeiro veja que $x = \hat{x}_2$, $x^* = \hat{x}_1$, $f^* = Q(\hat{x}_1)$ e $\pi^* = \pi_1$ na última desigualdade de (2.18). Além disso, os produtos $Tx^* = T\hat{x}_1$ e $Tx = T\hat{x}_2$ resultam em vetores cuja única componente não nula é a primeira. Estas componentes coincidem com as primeiras componentes dos vetores \hat{x}_1 e x , que são $v_1^{(1)} = 0$ (volume atualizado do reservatório) e $v_1^{(2)}$ (volume atual do reservatório para a próxima iteração), respectivamente. Temos então que a desigualdade referente ao primeiro corte é dada por:

$$\begin{aligned}
 -\pi_1^T T \hat{x}_2 - r &\leq -Q(\hat{x}_1) - \pi_1^T T \hat{x}_1 \\
 -[29, 63 \quad -80, 00] \begin{bmatrix} v_1^{(2)} \\ 0 \end{bmatrix} - r &\leq -21.440 - [29, 63 \quad -80, 00] \begin{bmatrix} v_1^{(1)} \\ 0 \end{bmatrix} \\
 -29, 63v_1^{(2)} - r &\leq -21.440 - [29, 63 \quad -80, 00] \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\
 -29, 63v_1^{(2)} - r &\leq -21.440.
 \end{aligned}$$

Esta última desigualdade representa a restrição do corte de Benders. Ela já está no formato para ser implementada computacionalmente. Para isto, criamos outra matriz $A_{\mathcal{I}}$ e vetor $b_{\mathcal{I}}$ para armazenar as restrições de desigualdade, isto é, os cortes. Esta matriz será atualizada e, conseqüentemente, aumentará uma linha a cada iteração.

$$\begin{aligned}
 A_{\mathcal{I}} &= [-29, 63 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad -1] \\
 b_{\mathcal{I}} &= [-21.440].
 \end{aligned}$$

Para incluir esta restrição no problema (2.12) basta fazer alguns ajustes, ficando:

$$r \geq -29, 63v_1^{(2)} + 21.440.$$

Passo (6): Recomeçamos do Passo (2).

2ª iteração:

Passo (2): O problema (2.12) pode ser reescrito como:

$$\left\{ \begin{array}{l} \text{minimizar} \quad c^T x + r \\ \text{sujeito a} \quad Ax = b \\ \quad \quad \quad A_{\mathcal{I}} x \leq b_{\mathcal{I}} \\ \quad \quad \quad \ell \leq x \leq u \end{array} \right. \Rightarrow \left\{ \begin{array}{l} \text{minimizar} \quad c^T x + r \\ \text{sujeito a} \quad Ax = b \\ \quad \quad \quad r \geq -29, 63v_1^{(2)} + 21.440 \\ \quad \quad \quad \ell \leq x \leq u. \end{array} \right.$$

Resolvendo este novo problema, obtemos o resultado

$$\hat{x}_2 = \begin{bmatrix} v_1^{(2)} \\ q_1^{(2)} \\ s_1^{(2)} \\ gt_{1,1}^{(2)} \\ gt_{2,1}^{(2)} \\ gt_{3,1}^{(2)} \\ gt_{4,1}^{(2)} \\ d_1^{(2)} \\ \hat{r}_1^{(2)} \end{bmatrix} = \begin{bmatrix} 723, 60 \\ 661, 73 \\ 0, 00 \\ 100, 00 \\ 150, 00 \\ 114, 74 \\ 0, 00 \\ 0, 00 \\ 0, 00 \end{bmatrix}.$$

Passo (3): Calculamos o limite inferior \underline{z} e testamos o critério de parada.

$$\begin{aligned}\underline{z} &= c^T \hat{x}_2 + \hat{r}_1^{(2)} \\ &= 8.589,63 + 0 \\ &= 8.589,63\end{aligned}$$

$$\bar{z} - \underline{z} = 22.407,41 - 8.589,63 > \varepsilon.$$

Logo, o critério não foi satisfeito e então seguimos ao próximo passo.

Passo (4): Utilizando \hat{x}_2 , resolvemos o problema (2.5) e obtemos que $Q(\hat{x}_2) = 6.000$. Assim, atualizamos $\bar{z} = c^T \hat{x}_2 + Q(\hat{x}_2) = 8.589,63 + 6.000 = 14.589,63$.

Passo (5): Atualizamos o número de vértice para $n = 1 + 1 = 2$. O multiplicador simplex π_2 , associado às restrições de (2.5), obtido no passo anterior, é:

$$\pi_2 = [14,81 \quad -40,00]^T.$$

As matrizes atualizadas e a restrição do novo corte ficam:

$$A_{\mathcal{I}} = \begin{bmatrix} -29,63 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\ -14,81 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}$$

$$b_{\mathcal{I}} = \begin{bmatrix} -21.440 \\ -16.720 \end{bmatrix}.$$

Esta restrição fica da seguinte forma para incluir no problema (2.12):

$$r \geq -14,81v_1^{(3)} + 16.720.$$

Passo (6): Recomeçamos do Passo (2).

3ª iteração:

Passo (2): O problema (2.12) atualizado fica na forma:

$$\left\{ \begin{array}{l} \text{minimizar} \quad c^T x + r \\ \text{sujeito a} \quad Ax = b \\ \quad \quad \quad r \geq -29,63v_1^{(3)} + 21.440 \\ \quad \quad \quad r \geq -14,81v_1^{(3)} + 16.720 \\ \quad \quad \quad \ell \leq x \leq u. \end{array} \right.$$

Resolvendo este novo problema, obtemos

$$\hat{x}_3 = \begin{bmatrix} v_1^{(3)} \\ q_1^{(3)} \\ s_1^{(3)} \\ gt_{1,1}^{(3)} \\ gt_{2,1}^{(3)} \\ gt_{3,1}^{(3)} \\ gt_{4,1}^{(3)} \\ d_1^{(3)} \\ \hat{r}_1^{(3)} \end{bmatrix} = \begin{bmatrix} 788,79 \\ 636,58 \\ 0,00 \\ 100,00 \\ 150,00 \\ 138,89 \\ 0,00 \\ 0,00 \\ 5034,18 \end{bmatrix} .$$

Passo (3): Calculamos o limite inferior \underline{z} e testamos o critério de parada.

$$\begin{aligned} \underline{z} &= c^T \hat{x}_2 + r_1^{(3)} \\ &= 9.555,45 + 5034,18 \\ &= 14.589,63, \end{aligned}$$

$$\bar{z} - \underline{z} = 14.589,63 - 14.589,63 = 0 < \varepsilon.$$

Logo, o critério de parada foi satisfeito nesta iteração. Além disso, ressaltamos que, computacionalmente, a diferença $\bar{z} - \underline{z}$ resultou num valor na ordem de 10^{-9} . A Figura 2.4 mostra os dois cortes de Benders que aproximam a função recurso, obtidos ao longo das iterações. Veja que esta função é limitada pelo custo nulo, pois não faz sentido haver custos negativos. Já a Figura 2.5 ilustra o comparativo entre os limites \bar{z} e \underline{z} , que correspondem ao critério de parada do algoritmo.

Na próxima seção estendemos o problema de dois estágios para o caso multiestágios. Do mesmo modo que foi feito em dois estágios, exibimos a formulação geral e dinâmica do problema, além de estender o algoritmo PDDD para este novo caso.

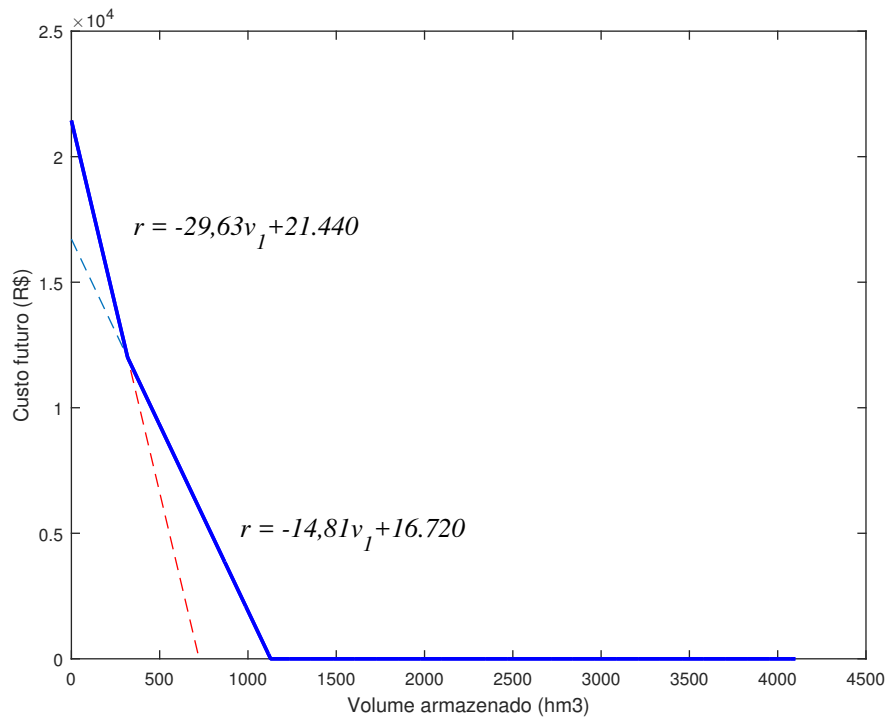


Figura 2.4: Cortes de Benders

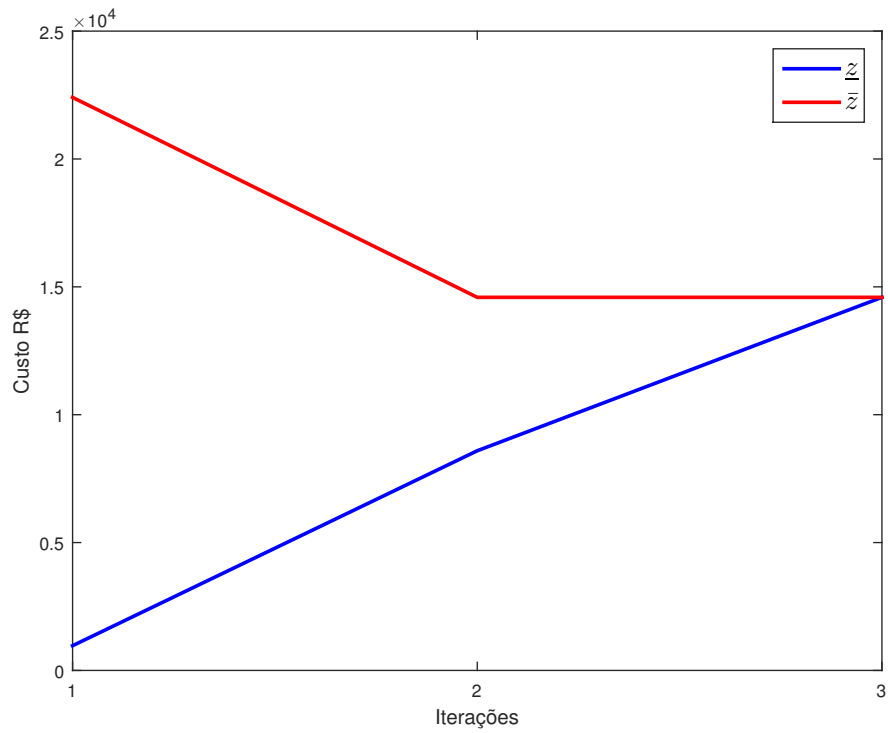


Figura 2.5: Limites inferior e superior

2.3 Programação linear em multiestágios

O problema de dois estágios pode ser facilmente generalizado para o caso multiestágios, o qual envolve um horizonte de planejamento maior. Como consequência disso, quando T é grande, os problemas envolvidos neste caso possuem um aumento considerável no número de variáveis e nas dimensões das matrizes e vetores, quando comparados com problemas de dois estágios.

Nesta seção exibimos as formulações geral e dinâmica do problema de programação linear em multiestágios, assim como a extensão do algoritmo PDDD. Ao final da seção retomamos o exemplo de despacho hidrotérmico resolvido na Seção 2.2, ainda em dois estágios, para motivar a inclusão da estocasticidade neste tipo de problema.

2.3.1 Formulação geral

Suponha que um problema tenha um horizonte de planejamento de $T > 2$ períodos (estágios). Sejam x_1, x_2, \dots, x_T os vetores decisão correspondentes aos estágios $t = 1, \dots, T$. Ainda com a notação de [27], a formulação de um problema de programação linear multiestágios, a qual pode ser entendida como uma extensão de (2.1), é dada por:

$$\left\{ \begin{array}{llllllll} \text{minimizar} & c_1^T x_1 & + & c_2^T x_2 & + & c_3^T x_3 & + & \dots & + & c_T^T x_T \\ \text{sujeito a} & A_1 x_1 & & & & & & & & = b_1 \\ & B_2 x_1 & + & A_2 x_2 & & & & & & = b_2 \\ & & & B_3 x_2 & + & A_3 x_3 & & & & = b_3 \\ & & & & & \ddots & & \ddots & & \vdots \\ & & & & & & & & B_T x_{T-1} & + & A_T x_T & = b_T \\ & x_1 \geq 0, & & x_2 \geq 0, & & x_3 \geq 0, & & \dots & & x_T \geq 0, \end{array} \right. \quad (2.25)$$

onde $c_t \in \mathbb{R}^{n_t}$, $b_t \in \mathbb{R}^{m_t}$ e as matrizes A_t e B_t , possuem dimensões apropriadas de acordo com as dimensões dos vetores. Observe que cada linha das restrições de (2.25), com exceção da primeira, relaciona dois estágios consecutivos a partir das variáveis x_{i-1} e x_i , isto é, x_i é a consequência da decisão x_{i-1} , assim como ocorre no problema de dois estágios. Fazendo alguns ajustes nas matrizes e vetores deste problema, podemos resolvê-lo de forma direta da seguinte maneira:

- Agrupar todas as variáveis de decisão x_i e vetores b_i, c_i, ℓ_i e u_i :

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_T \end{bmatrix}, \quad b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_T \end{bmatrix}, \quad c = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_T \end{bmatrix}, \quad \ell = \begin{bmatrix} \ell_1 \\ \ell_2 \\ \vdots \\ \ell_T \end{bmatrix} \quad e \quad u = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_T \end{bmatrix};$$

- No estágio $t = 1$

$$\begin{cases} \text{minimizar} & c_1^T x_1 + Q_2(x_1) \\ \text{sujeito a} & A_1 x_1 = b_1 \\ & x_1 \geq 0. \end{cases} \quad (2.29)$$

Veja que cada função recurso Q_i depende da decisão tomada no estágio anterior. Desta forma é possível ver que todos os estágios estão interligados por estas funções. O algoritmo PDDD estendido para o caso multiestágios também é separado nas etapas simulação, onde se calcula uma política viável (uma sequência de soluções $\{x_1, \dots, x_T\}$ que satisfazem as restrições dos problemas dos estágios $1, \dots, T$), e recursão, onde as aproximações das funções recurso são calculadas.

Uma das diferenças em relação ao Algoritmo 2 é a forma que o limite superior é calculado, que é dado pela soma dos custos de todos os estágios. Além disto, a partir do momento em que os cortes são calculados e utilizados para aproximar as funções recurso \hat{Q}_t nos estágios $t = 2, \dots, T - 1$, o problema dual deve incluir os multiplicadores destas novas restrições, sendo necessário refazer os cálculos dos coeficientes α e β dos cortes de Benders. Note que no estágio $t = T$ não há aproximações da função recurso, pois não consideramos a função Q_{T+1} ou, em outras palavras, $Q_{T+1}(x) = 0$ para todo x , e no primeiro estágio as aproximações são construídas a partir dos multiplicadores de Lagrange da função recurso do segundo estágio, não sendo necessário formular seu dual.

Ao longo do processo iterativo, nos estágios $t = 2, \dots, T - 1$ e na n -ésima iteração, as funções recurso Q_t são aproximadas inferiormente pelas funções \hat{Q}_t da seguinte forma:

$$\begin{aligned} \hat{Q}_t(x_{t-1}) &= \min && c_t^T x_t + r_{t+1} \\ &\text{sujeito a} && B_t x_{t-1} + A_t x_t = b_t \\ &&& r_{t+1} \geq \beta_{t+1}^i x_t + \alpha_{t+1}^i, \quad i = 1, \dots, n \\ &&& x_t \geq 0. \end{aligned} \quad (2.30)$$

Considerando π_t^n e θ_t^i , $i = 1, \dots, n$, os multiplicadores de Lagrange associados às restrições de igualdade e desigualdade do problema (2.30), respectivamente, seu dual é dado por:

$$\begin{aligned} \hat{Q}_t(x_{t-1}) &= \max_{\pi_t^n, \theta_t^i} && \pi_t^n (b_t - B_t x_{t-1}) + \sum_{i=1}^n \theta_t^i \alpha_{t+1}^i \\ &\text{sujeito a} && \pi_t^n A_t - \sum_{i=1}^n \theta_t^i \beta_t^i \leq c_t \\ &&& \sum_{i=1}^n \theta_t^i = 1 \\ &&& \theta_t^i \geq 0 \quad i = 1, \dots, n. \end{aligned} \quad (2.31)$$

Do mesmo modo que foi analisado anteriormente, os problemas (2.30) e (2.31) possuem valores ótimos iguais. Considerando então que π_t^n e θ_t^i , $i = 1, \dots, n$ é a solução de (2.31), os coeficientes dos cortes de Benders para as funções recurso Q_t , $t = 2, \dots, T - 1$ são dados por

$$\alpha_t^n = \pi_t^n b_t + \sum_{i=1}^n \theta_t^i \alpha_{t+1}^i \quad \text{e} \quad \beta_t^n = -\pi_t^n B_t, \quad (2.32)$$

enquanto que os coeficientes α_t e β_t no estágio T são dados por (2.16). Seguindo a formulação do problema (2.30), o problema de primeiro estágio aproximado é

$$\left\{ \begin{array}{l} \text{minimizar} \quad c_1^T x_1 + r_2 \\ \text{sujeito a} \quad A_1 x_1 = b_1 \\ \quad \quad \quad r_2 \geq \beta_2^i x_1 + \alpha_2^i, \quad i = 1, \dots, n \\ \quad \quad \quad x_1 \geq 0. \end{array} \right. \quad (2.33)$$

Na sequência exibimos os passos do algoritmo PDDD para o caso multiestágios:

Algoritmo 3 - PDDD em multiestágios

Passo (1). Inicialização: Seja T o número de estágios. Definir função custo futuro aproximado $\hat{Q}_t(x_t) = 0$, para $t = 2, \dots, T$, limite superior $\bar{z} = \infty$, número de vértices $n = 0$ e tolerância $\varepsilon > 0$;

Passo (2). Resolver o problema de primeiro estágio aproximado (2.33), tomar \hat{x}_1 como a decisão tentativa;

Passo (3). Calcular o limite inferior \underline{z} como em (2.13). Se $\bar{z} - \underline{z} \leq \varepsilon$, parar o algoritmo. Caso contrário, ir para o Passo (4);

Passo (4). (Fase progressiva) Para $t = 2, \dots, T - 1$, resolver (2.30) e para $t = T$, resolver (2.27), sempre utilizando a solução do estágio anterior. Guardar todas as soluções \hat{x}_i , $i = 2, \dots, T$. Os Passos (2) e (4) estão ilustrados na Figura 2.6. Cada linha da figura representa uma passagem da fase progressiva, onde na primeira calcula-se \hat{x}_1 , que está associado a $t = 1$. Na segunda linha, utiliza-se \hat{x}_1 para calcular \hat{x}_2 , que está associado ao segundo estágio e assim sucessivamente. Generalizando este procedimento, a solução \hat{x}_t obtida num estágio t é utilizada no estágio $t + 1$, $t = 1, \dots, T - 1$ para obter a solução \hat{x}_{t+1} ;

Passo (5). Atualizar limite superior \bar{z} :

$$\bar{z} = \sum_{i=1}^T c_i \hat{x}_i;$$

Passo (6). (Fase recursiva) Para $t = T$ resolver (2.27) e para $t = T - 1, \dots, 2$ resolver (2.30), utilizando as soluções obtidas no Passo (4). Seja π_t^n e θ_t^i , $i = 1, \dots, n$ os multiplicadores associados às restrições de igualdade e desigualdade do problema (2.30) na solução ótima, respectivamente. Usar os multiplicadores para construir um hiperplano suporte adicional para a aproximação da função recurso no estágio anterior, $\hat{Q}_t(x_{t-1})$, utilizando (2.32). Esse passo está ilustrado na Figura 2.7, onde cada linha representa uma passagem da fase recursiva. Na primeira linha, o último círculo indica que foram calculados os multiplicadores π_T^n e θ_T^i , $i = 1, \dots, n$ no estágio T , os quais

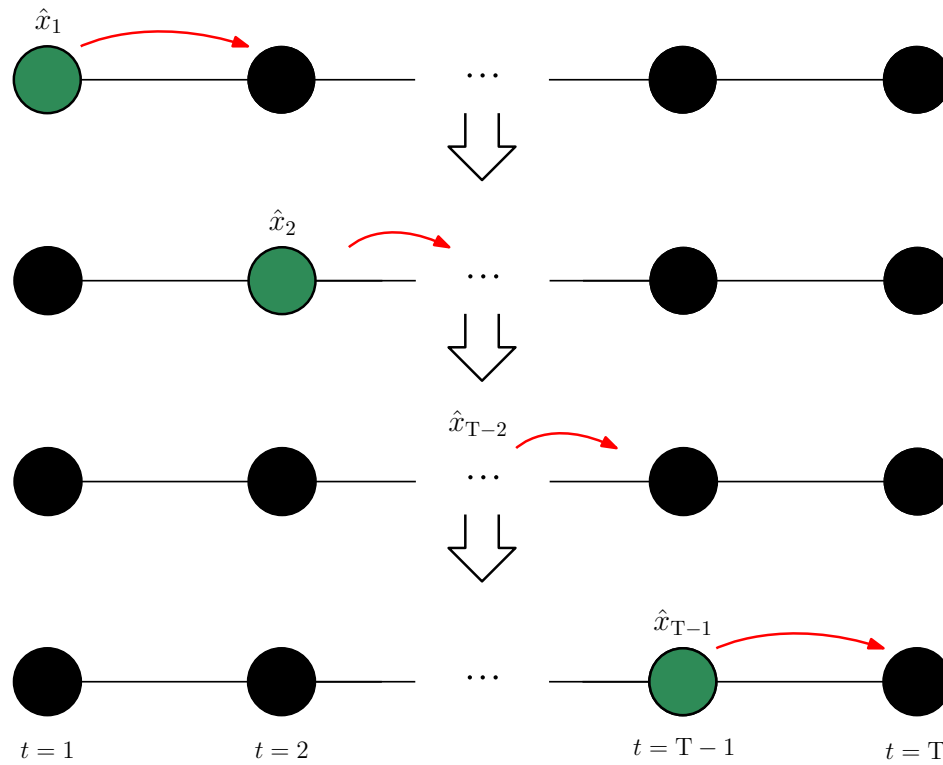


Figura 2.6: Fase progressiva

serão utilizados no estágio $T - 1$ para construir uma aproximação da função recurso \hat{Q}_T que está sendo representada pela linha inclinada no círculo imediatamente anterior. Na segunda linha, já utilizando \hat{Q}_T , calculam-se os multiplicadores π_{T-1}^n e θ_{T-1}^i , $i = 1, \dots, n$, no estágio $T - 1$, para aproximar a função recurso \hat{Q}_{T-1} . Segue esse procedimento até $t = 2$, aproximando as funções recurso \hat{Q}_i , $i = 2, \dots, T$;

Passo (7). Ir para o Passo (2).

Um detalhe importante a ser observado no algoritmo é que como \hat{Q}_i está associada a um estágio diferente, os cortes de Benders, que são calculados no decorrer das iterações, devem ser alocados em seus respectivos estágios.

Iniciamos o próximo capítulo motivando a inclusão da estocasticidade no problema de despacho hidrotérmico, assim como apresentamos algumas formulações dos problemas de dois e multiestágios estocástico, algoritmos para resolvê-los (*Decomposição Aninhada* e *Programação Dinâmica Dual Estocástica*, que é uma extensão do PDDD, incorporando incertezas nos dados do problema), e também damos suporte teórico que garanta a viabilidade e existência de solução para os problemas.

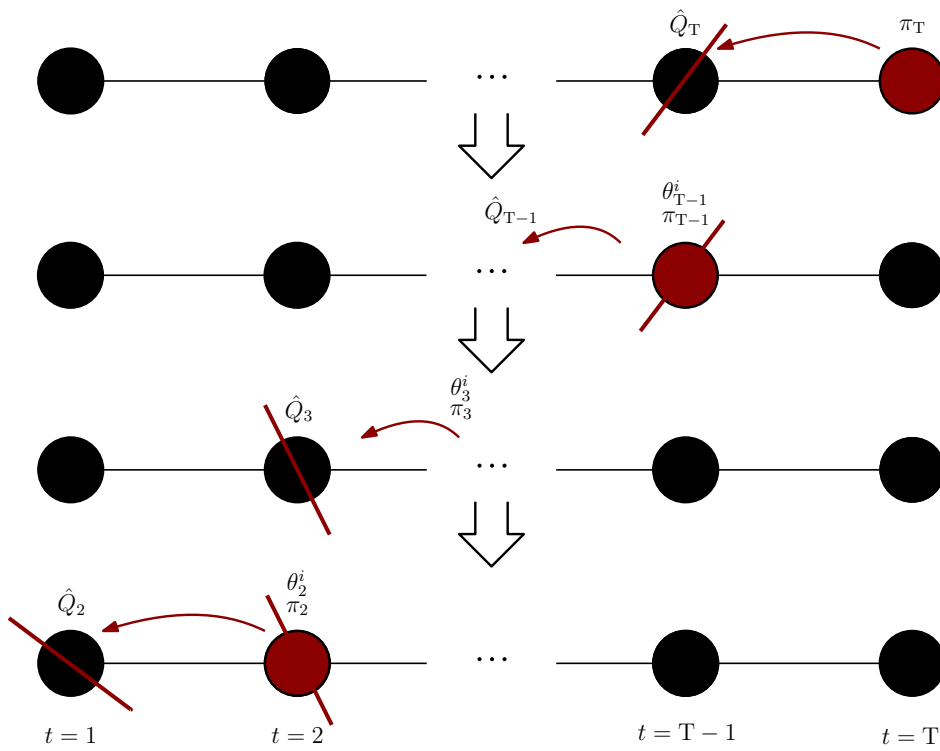


Figura 2.7: Fase recursiva

Capítulo 3

Programação linear estocástica

Em diversas áreas é comum elaborar planejamentos que envolvem períodos de tempos futuro, onde um dos objetivos se dá pela otimização de recursos, como, por exemplo, tempo, custos, lucros, entre outros. Geralmente, estes planejamentos recaem num problema de otimização. No entanto, as informações referentes ao futuro não devem ser consideradas de forma exata, visto que elas estão sujeitas a diversos fatores que podem alterá-las, o que as tornam incertas.

Este capítulo está dividido em 4 seções, onde na Seção 3.1 o problema de despacho hidrotérmico é retomado para motivar a inclusão de incertezas. A Seção 3.2 traz algumas características de um programa estocástico com recurso. Na Seção 3.3 é trabalhado o programa estocástico em dois estágios, onde são apresentadas algumas de suas formulações, o algoritmo decomposição aninhada e um exemplo. Já na Seção 3.4 o programa estocástico é estendido para o caso multiestágios. São apresentadas nesta seção algumas formulações do programa, algumas características que envolvem a árvore de cenários e o algoritmo PDDE.

3.1 Exemplo de motivação

No exemplo de despacho hidrotérmico de dois estágios trabalhado na Seção 2.2, consideramos que a afluência era única, ou seja, tínhamos conhecimento exato de sua ocorrência. No entanto, uma questão que pode surgir é: o que aconteceria se a afluência no segundo estágio fosse $300 \text{ m}^3/\text{s}$ em vez de $450 \text{ m}^3/\text{s}$ e a decisão do primeiro estágio x fosse a mesma? Em outras palavras, o que aconteceria se nossa previsão da afluência futura estivesse incorreta? Resolvendo novamente o problema, em sua formulação geral, com esta mudança, a solução de segundo estágio y e o custo da operação obtidos são:

$$y = \begin{bmatrix} v_2 \\ q_2 \\ s_2 \\ d_2 \\ gt_{1,2} \\ gt_{2,2} \\ gt_{3,2} \\ gt_{4,2} \end{bmatrix} = \begin{bmatrix} 0 \\ 543.32 \\ 0 \\ 0 \\ 100 \\ 150 \\ 200 \\ 28.44 \end{bmatrix},$$

Custo operativo:
Custo 1º estágio + Custo 2º estágio
R\$ 7.212,90 + R\$ 14.273,00
Custo total = R\$ 21.485,90.

Veja que diante desta mudança de afluência foi necessário ativar a quarta usina térmica, o que não havia sido planejado na primeira situação, aumentando o custo operativo em 47,26%. Por causa do ocorrido, temos um grande problema: se armazenarmos água nos reservatórios e ocorrer afluências altas no futuro, possivelmente será necessário verter água, pois, além de suprir a demanda com energia hidrelétrica, é provável que o nível do reservatório fique acima da capacidade máxima. Por outro lado, se muita água fosse turbinada do reservatório e ocorresse um período de baixas afluências, a maioria da energia produzida para suprir a demanda seria proveniente das usinas termelétricas, o que resultaria num custo operativo muito elevado.

A partir deste exemplo verificamos que não há uma solução ótima para todos os casos e por isto é imprescindível procurar uma solução que leve em consideração estas possibilidades. Percebemos então a necessidade de buscar uma metodologia mais apropriada para ser empregada nos casos em que incertezas são agregadas na modelagem do problema. Esse é um dos motivos que justificam a introdução da estocasticidade, a qual é tratada em Programação Estocástica.

Um problema de programação estocástica difere do determinístico no sentido que são incorporadas incertezas na modelagem do problema a partir da inclusão de variáveis aleatórias na função objetivo e/ou nas restrições com distribuição de probabilidade conhecida, o que pode ser estimado através de dados históricos.

3.2 Considerações gerais

Os programas estocásticos com recurso apresentam características que se destacam. Num programa estocástico modelado em T estágios, o primeiro estágio normalmente representa o início do tempo (presente), isto é, o momento em que as decisões devem ser tomadas. Ele também é considerado determinístico, pois todos os parâmetros correspondentes ao problema de primeiro estágio são observáveis e podem ser determinados com boa precisão. Os dados referentes aos próximos estágios são conhecidos somente num sentido probabilístico, os quais são revelados ao início de cada estágio. Para isto, utiliza-se uma variável aleatória com distribuição de probabilidade conhecida para representá-los.

Já quanto às decisões, a do primeiro estágio deve ser tomada sem conhecimento das realizações futuras da variável aleatória, sendo que as decisões correspondentes aos estágios futuros são consideradas como recursos, isto é, decisões que visam corrigir possíveis efeitos negativos das decisões tomadas nos estágios anteriores e

por isto recebem o nome de *ações corretivas* ou *ações de recurso*. Estas decisões são tomadas na medida que os dados deixam de ser incertos.

Neste trabalho utilizamos um método de programação estocástica no qual o conjunto de incertezas pode ser representado numa árvore de cenários. Uma técnica que é frequentemente utilizada é a *Sample Average Approximation* (SAA), o que pode ser encontrado em [27, Capítulo 5]. Esta técnica consiste em tomar amostras aleatórias de N realizações da variável aleatória, que pode ser vista como dados históricos da variável ou que podem ser geradas por técnicas de amostragem de Monte Carlo. Fixado o conjunto de amostras, podemos resolver o problema estocástico por técnicas de otimização determinística, obtendo uma estimativa da solução ótima do problema original. Sob algumas condições de regularidade, pela *Lei dos Grandes Números* [27, Seção 7.2.5] temos que, quando $N \rightarrow \infty$, o problema SAA converge pontualmente com probabilidade 1 (c.p.1) para a solução do problema original.

Segundo [22], "para definir matematicamente um programa estocástico, sejam \mathcal{F} a σ -álgebra gerada pelo conjunto fechado Ξ , e ξ uma possível realização da variável aleatória $\xi : (\Omega, \hat{\mathcal{F}}) \rightarrow (\Xi, \mathcal{F})$, definida em um espaço de probabilidades apropriado $(\Omega, \hat{\mathcal{F}}, \hat{P})$. A realização $\xi(\omega) = \xi$ é chamada de evento aleatório, ou simplesmente, *cenário*. Definindo a medida de probabilidades em Ξ por $P(\mathcal{A}) = \hat{P}(\{\omega \in \Omega : \xi(\omega) \in \mathcal{A}\})$, para todo subconjunto $\mathcal{A} \subset \Xi$, tem-se o *espaço de probabilidades* (Ξ, \mathcal{F}, P) induzido por ξ ."

De modo geral, um programa estocástico muito utilizado é a minimização do valor esperado \mathbb{E} das funções parametrizadas pelos cenários, $f : \mathcal{X}_1 \times \Omega \rightarrow \mathbb{R}$, o qual é escrito como

$$\min_{x \in \mathcal{X}_1} \mathbb{E}[f(x, \xi)], \quad \text{com } \mathcal{X}_1 \subseteq \mathbb{R}^{n_1}, \quad (3.1)$$

onde a esperança é tomada com respeito à distribuição de probabilidades P da variável aleatória ξ . Condições que garantem a viabilidade e a existência de uma solução para o problema (3.1) se encontram no final da seção a seguir, na qual abordamos a programação linear estocástica em dois estágios.

3.3 Programação linear estocástica em dois estágios

3.3.1 Formulação geral

Utilizando o problema (3.1), a função f de um problema estocástico em dois estágios é da forma

$$f(x, \xi) = f_1(x) + \inf_{y \in \mathcal{X}_2(x, \xi)} f_2(y, \xi), \quad (3.2)$$

onde $x \in \mathcal{X}_1 \subseteq \mathbb{R}^{n_1}$, o conjunto viável $\mathcal{X}_2(x, \xi) \subset \mathbb{R}^{n_2}$ do problema de segundo estágio é determinado pela multifunção $\mathcal{X}_2 : \mathcal{X}_1 \times \Omega \rightrightarrows \mathbb{R}^{n_2}$ e as funções objetivos que determinam os valores dos problemas de primeiro e segundo estágios são $f_1 : \mathcal{X}_1 \rightarrow \mathbb{R}$ e $f_2 : \mathbb{R}^{n_2} \times \Omega \rightarrow \mathbb{R}$, respectivamente. Como mencionado na seção anterior, não há incertezas no primeiro estágio e seu valor é determinado pelo vetor de decisões x , o qual deve ser tomado antes que o evento incerto ξ se torne conhecido.

Como nosso interesse está na programação linear estocástica em dois estágios, as funções f_1 , f_2 e os conjuntos \mathcal{X}_1 e \mathcal{X}_2 dos problemas (3.1) e (3.2) são definidos por

$$\begin{aligned} f_1(x) &= c^T x, & f_2(y, \xi) &= q^T y, \\ \mathcal{X}_1 &= \{x \in \mathbb{R}_+^{n_1} : Ax = b\} & \text{e} & \mathcal{X}_2(x, \xi) = \{y \in \mathbb{R}_+^{n_2} : Tx + Wy = h\}, \end{aligned} \quad (3.3)$$

onde $c \in \mathbb{R}^{n_1}$, $q \in \mathbb{R}^{n_2}$, $b \in \mathbb{R}^{m_1}$, $h \in \mathbb{R}^{m_2}$ e as dimensões das matrizes são apropriadas de acordo com as dimensões dos vetores e \mathcal{X}_2 é um conjunto poliedral¹. As componentes dos vetores q e h e das matrizes T e W (denominada matriz recurso), mas não necessariamente todas, são variáveis aleatórias que compõem o cenário $\xi = (q, h, T, W)$ do problema de segundo estágio, o qual é uma realização da variável aleatória $\xi(\omega) = (q(\omega), h(\omega), T(\omega), W(\omega))$.

Deste modo, a *formulação geral* do nosso problema de programação linear estocástico em dois estágios é dado na forma

$$\left\{ \begin{array}{ll} \text{minimizar} & c^T x + \mathbb{E}[q(\omega)^T y(\omega)] \\ \text{sujeito a} & Ax = b, \quad x \geq 0 \\ & T(\omega)x + W(\omega)y(\omega) = h(\omega) \quad q.c. \\ & y(\omega) \geq 0 \quad q.c., \end{array} \right. \quad (3.4)$$

onde *q.c.* significa *quase certamente*. Esta expressão, num sentido probabilístico, quer dizer que a propriedade é válida a menos de um conjunto com probabilidade zero. Nesta formulação y está num espaço de probabilidade, pois depende da realização da variável aleatória. Note que em (3.4) os vetores b e c e a matriz A não dependem da realização ω , pois, como dito anteriormente, estes dados são determinísticos. A partir daqui não será mais colocado que as matrizes e vetores dependem da realização ω da variável aleatória ξ , a menos que seja conveniente.

3.3.2 Formulação dinâmica

Da mesma forma que foi feito no caso determinístico, o problema (3.4) pode ser representado em sua *formulação dinâmica*, onde ele é separado em primeiro e segundo estágios. O primeiro estágio do programa linear estocástico é

$$\left\{ \begin{array}{ll} \text{minimizar} & c^T x + \mathcal{Q}(x) \\ \text{sujeito a} & Ax = b \\ & x \geq 0, \end{array} \right. \quad (3.5)$$

onde a função $\mathcal{Q} : \mathbb{R}^{n_1} \rightarrow \mathbb{R}$ com $\mathcal{Q}(x) = \mathbb{E}[\mathcal{Q}(x, \xi)]$ é denominada *valor esperado da função recurso* e $\mathcal{Q} : \mathbb{R}^{n_1} \times \Omega \rightarrow \mathbb{R}^n$ é a *função recurso*, a qual corresponde ao valor ótimo do problema de segundo estágio

$$\left\{ \begin{array}{ll} \text{minimizar} & q^T y \\ \text{sujeito a} & Tx + Wy = h \\ & y \geq 0, \end{array} \right. \quad (3.6)$$

¹Um conjunto \mathcal{X} é dito poliedral se \mathcal{X} é a interseção de um número finito de semi-espacos.

em que a variável de segundo estágio y não é simplesmente um vetor de \mathbb{R}^{n_2} , mas sim uma função de x e do cenário $\xi \in \Xi$, onde $\Xi \subset \mathbb{R}^d$ é o conjunto suporte da distribuição de probabilidades de ξ . A variável x do primeiro estágio é denominada variável *aqui e agora* (em inglês, *here and now variable*), pois ela representa a decisão a ser tomada antes da realização da variável aleatória. Após tomar conhecimento da realização ω , a variável de decisão do segundo estágio y corresponde a um efeito corretivo, ou ação de recurso, da decisão do primeiro estágio, já que não havia conhecimento exato do futuro, a não ser por uma função distribuição de probabilidade. Deste modo, a variável y é chamada de variável *espere e veja* (em inglês, *wait and see variable*).

Se para algum x e $\xi \in \Xi$ o problema de segundo estágio (3.6) é inviável, temos que $Q(x, \xi) = +\infty$ por definição. Além disto, é possível ocorrer que a função Q seja ilimitada inferiormente e então $Q(x, \xi) = -\infty$. Modelos que apresentam estes tipos de ocorrências são aqueles que queremos evitar.

Quando a variável aleatória ξ possui uma função distribuição de probabilidade (FDP) contínua, ou seja, o conjunto suporte Ξ possui infinitos elementos, o valor esperado da função Q em relação à distribuição de probabilidades P é definida como

$$Q(x) = \mathbb{E}[Q(x, \xi)] = \int_{\Xi} Q(x, \xi) dP(\xi). \quad (3.7)$$

Um detalhe a ser observado é que obter uma representação explícita de (3.7) é praticamente impossível. Além disto, resolver o problema de dois estágios torna-se intratável computacionalmente pelo fato de precisar calcular a integral, normalmente multidimensional, que corresponde ao valor esperado da função Q . Por este motivo, é feita uma discretização da variável aleatória ξ numa quantidade significativa de cenários $\Xi = \{\xi_1, \xi_2, \dots, \xi_N\}$, com $\xi_i = (q_i, h_i, T_i, W_i)$ e sua respectiva probabilidade $p_i > 0$, $i = 1, \dots, N$, onde $\sum_{i=1}^N p_i = 1$, com o intuito de representá-la adequadamente. É natural pensar que quanto maior for a discretização da variável aleatória, melhor será a aproximação do processo estocástico, como comentado na Seção 3.2. Em contrapartida, quanto maior N for, maior será o número de problemas a serem resolvidos.

Considerando que a variável aleatória seja discretizada, no caso contínuo, ou seja discreta com finitos elementos, o valor esperado aproximado ou real, respectivamente, passa a ser

$$Q(x) = \mathbb{E}[Q(x, \xi)] = \sum_{i=1}^N p_i Q(x, \xi_i), \quad (3.8)$$

onde N representa o número de elementos do espaço amostral Ξ , p_i representa a probabilidade de ocorrência do cenário ξ_i e o valor da função Q no cenário ξ_i é obtido resolvendo o problema de segundo estágio (3.6) utilizando os dados $\xi_i = (q_i, h_i, T_i, W_i)$, ficando

$$\begin{aligned} Q(x, \xi_i) &= \min && q_i^T y_i \\ &\text{sujeito a} && T_i x + W_i y_i = h_i \\ &&& y_i \geq 0. \end{aligned} \quad (3.9)$$

Supondo que a função Q seja bem definida e finita para todo $x \in \mathcal{X}_1$ e considerando que uma amostra

do espaço amostral Ξ seja determinada, como descrito na Seção 3.2, com cenários equiprováveis, isto é, $p_i = 1/N$, $i = 1, \dots, N$, para qualquer $x \in \mathcal{X}_1$, podemos então estimar o valor esperado da função Q tomando a média dos valores de $Q(x, \xi_i)$. Estas condições nos conduzem ao bem conhecido problema *Sample Average Approximation*, que é dado por

$$\min_{x \in \mathcal{X}_1} c^T x + \frac{1}{N} \sum_{i=1}^N Q(x, \xi_i), \quad (3.10)$$

com $Q(x, \xi_i)$ dado por (3.9).

Diante dessas considerações, resolver o problema (3.4) pela formulação dinâmica pode ser interpretado da seguinte maneira: no primeiro estágio é calculado x ; em seguida, resolve-se N subproblemas de segundo estágio (3.9) utilizando x . O objetivo é minimizar a soma do custo do primeiro estágio $c^T x$ com o valor esperado do custo de segundo estágio (3.8). A Figura 3.1, extraída de [11], ilustra este processo, onde os subproblemas são dados por (3.9).

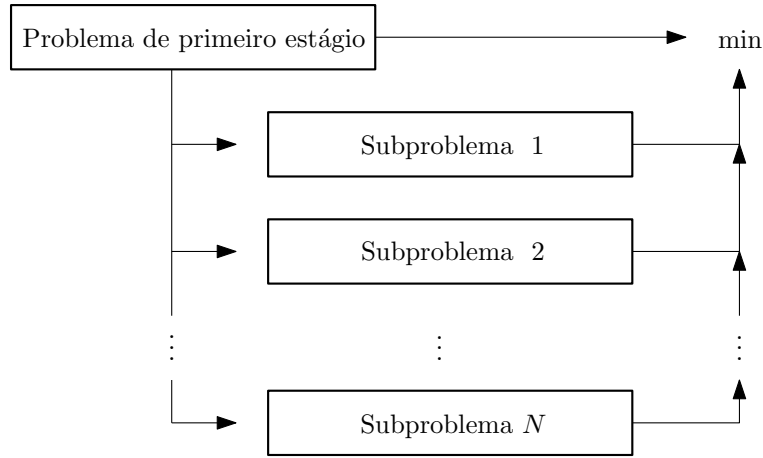


Figura 3.1: Processo de decisão dois estágios

Agora, baseado em [22], fazemos algumas considerações teóricas a respeito do problema de programação estocástica em dois estágios. A garantia de que os problemas (3.1) e (3.2) sejam viáveis, independente da discretização de Ξ , é estabelecida na Proposição 3.3.1 onde, primeiramente, é necessário definir o que é um problema de recurso fixo.

Definição 3.3.1. *O problema de dois estágios (3.5)-(3.6) é dito ter recurso fixo se a matriz W é fixa (não aleatória).*

Proposição 3.3.1. *Suponha que o problema (3.5)-(3.6) tem recurso fixo. Sejam os conjuntos*

$$K_2 = \{x : Q(x) < \infty\} \quad e \quad K_2^P = \{x : \cap_{\xi \in \Xi} \mathcal{X}_2(x, \xi) \neq \emptyset\},$$

onde $\mathcal{X}_2(x, \xi) = \{y \in \mathbb{R}_+^{n_2} : Tx + Wy = h\}$. Se a variável aleatória ξ que define o cenário $\xi = (q, h, T)$ tem variância finita, então

$$K_2 = K_2^P.$$

No caso do problema de segundo estágio ser definido como (3.6), a proposição a seguir estabelece condições suficientes para que a função $\mathcal{Q}(\cdot)$ seja bem definida, cuja prova se encontra em [22, Proposição 2.2].

Proposição 3.3.2. *Considere o programa linear definido pelo problema (3.2) com função $f_2(\cdot, \xi)$ definida em (3.3). Além disso, suponha que:*

1. o problema (3.5)-(3.6) tenha recurso fixo;
2. a variável ξ que define $\xi = (q, h, T) \in \Xi$ possui variância finita;
3. o conjunto de segundo estágio $\mathcal{X}_2(x, \xi)$ é q.c. não vazio para todo $x \in \mathcal{X}_1$.

Então a função $\mathcal{Q}(x) = \mathbb{E}[Q(x, \xi)]$ é bem definida e $\mathcal{Q}(x) > -\infty$ para todo $x \in \mathbb{R}^{n_1}$. Adicionalmente, \mathcal{Q} é convexa, semicontínua inferiormente e Lipschitz contínua no $\text{dom}\mathcal{Q}$, e seu domínio é um subconjunto fechado e convexo do \mathbb{R}^{n_1} dado por

$$\text{dom}\mathcal{Q} = \{x \in \mathbb{R}^{n_1} : h - Tx \in \text{pos } W, \text{ c.p.1}\}^2.$$

Assim como discutido em [22, p. 17], suponhamos que a função $f(\cdot, \xi)$ é q.c. semicontínua inferiormente, própria e convexa no conjunto convexo \mathcal{X}_∞ para um evento fixo $\xi \in \Xi$. Se o conjunto das variáveis de segundo estágio $\mathcal{X}_2(\cdot, \xi)$ é compacto e não vazio, estas hipóteses garantem a existência de uma solução, a menos de um conjunto de probabilidade nula em Ξ , para o problema $\min_{x \in \mathcal{X}_1} f(x, \xi)$.

Apresentamos na próxima seção um algoritmo que utiliza da formulação dinâmica de um problema estocástico em dois estágios para resolvê-lo. Este algoritmo é uma extensão do *Algoritmo 2*, onde a função \mathcal{Q} é aproximada pelos cortes de Benders, iterativamente. Para isto, é necessário verificar algumas propriedades da função recurso e seu valor esperado.

3.3.3 Aproximações da função recurso

Na Seção 2.1.3 foi visto como construir aproximações da função recurso Q , no caso determinístico. No entanto, como a função que envolve o valor do segundo estágio é a função \mathcal{Q} , que é determinada por (3.8), aproximá-la não consiste mais em resolver um único subproblema e sim N subproblemas, onde N é o número de elementos do conjunto $\Xi = (\xi_1, \dots, \xi_N)$. Nesta seção fazemos algumas considerações teóricas que são necessárias para a construção dos cortes de Benders para aproximar o valor esperado da função recurso.

Considerando que o problema linear estocástico em dois estágios (3.3) tenha recurso relativamente completo³, podemos fazer, de maneira análoga, todo o desenvolvimento realizado na Seção 2.1.2 referente à reformulação do problema de segundo estágio (3.9) para cada cenário ξ_i , utilizando o problema dual e aproximando o valor esperado da função recurso \mathcal{Q} num processo iterativo a partir da obtenção dos vértices que compõe o conjunto de soluções dos subproblemas do segundo estágio e que podem ser utilizadas para

²O conjunto $\text{pos } W$ é definido por $\{\chi : \chi = Wy, y \geq 0\}$.

³O problema linear estocástico de dois estágios (3.2)-(3.3) é dito ter recurso relativamente completo se o conjunto viável para as variáveis do segundo estágio definido pela multifunção $\mathcal{X}_2(x, \xi)$ é não vazio para todo $x \in \mathcal{X}_1$ e para q.c. todo $\xi \in \Xi$.

construir os hiperplanos suportes da função. O que veremos a seguir é como utilizar essas soluções obtidas nos N subproblemas de segundo estágio para a construção de um corte de Benders.

Considere o problema primal e dual de segundo estágio para cada cenário ξ_i :

$$\begin{aligned} Q(x, \xi_i) = \min \quad & q_i^T y_i & = \max \quad & \pi^T (h_i - T_i x) \\ \text{sujeito a} \quad & T_i x + W_i y_i = h_i & \text{sujeito a} \quad & W_i^T \pi \leq q_i. \\ & y_i \geq 0 & & \end{aligned} \quad (3.11)$$

Uma propriedade da função recurso Q é discutida na proposição a seguir, a qual pode ser aplicada para cada cenário ξ_i , $i = 1, \dots, N$.

Proposição 3.3.3. *Para qualquer $\xi \in \Xi$ dado, a função recurso $Q(\cdot, \xi)$ é convexa. Além disto, se o conjunto $\{\pi : W^T \pi \leq q\}$ é não vazio e o problema (3.11) é viável para pelo menos um x , então a função $Q(\cdot, \xi)$ é poliedral.*

Demonstração. [27, Proposição 2.1]. □

A diferenciabilidade da função recurso é dada na proposição abaixo.

Proposição 3.3.4. *Suponha que para um dado $x = x_0$ e $\xi \in \Xi$, o valor $Q(x_0, \xi)$ é finito. Então $Q(\cdot, \xi)$ é subdiferenciável no ponto x_0 e*

$$\partial Q(x_0, \xi) = -T^T \mathfrak{D}(x_0, \xi), \quad (3.12)$$

onde

$$\mathfrak{D}(x, \xi) = \arg \max_{\{\pi : W^T \pi \leq q\}} \pi^T (h - Tx)$$

é o conjunto das soluções ótimas do problema dual

Demonstração. [27, Proposição 2.2]. □

Veja que o subdiferencial da função recurso $\partial Q(x)$, quando comparado com o que foi feito no caso determinístico baseado em [11], tem relação com os vértices da função. Isto ocorre pela sua interpretação geométrica em que o subdiferencial $\partial Q(x)$ corresponde às inclinações dos hiperplanos que são suportes do epígrafo de Q em $(x, Q(x)) \in \mathcal{X} \times \mathbb{R}$, [31, p.110].

Já para a função $Q(x)$, seu subdiferencial é determinado pela proposição seguinte:

Proposição 3.3.5. *Suponha que a distribuição de probabilidade de ξ tenha suporte finito $\Xi = \{\xi_1, \dots, \xi_N\}$ e o valor esperado da função recurso $Q(\cdot)$ seja finito em pelo menos um ponto $\bar{x} \in \mathbb{R}^{n_1}$. Então a função $Q(\cdot)$ é poliedral e para qualquer $x_0 \in \text{dom } Q$,*

$$\partial Q(x_0) = \sum_{k=1}^N p_k \partial Q(x_0, \xi_k) \quad (3.13)$$

Demonstração. [27, Proposição 2.3]. □

Como os valores ótimos dos problemas primal e dual do segundo estágio (3.11) são iguais, temos que o valor da função Q pode ser escrito como

$$Q(x) = \mathbb{E}[Q(x, \xi)] = \sum_{i=1}^N p_i Q(x, \xi_i) = \sum_{i=1}^N p_i \pi_i^T (h_i - T_i x), \quad (3.14)$$

onde π_i é a solução do problema dual (3.11) do segundo estágio para a variável de decisão x do primeiro estágio no cenário ξ_i , ou seja, $\pi_i = \arg \max\{\pi^T (h_i - T_i x) : W_i^T \pi \leq q_i\}$.

Desta forma, utilizando (3.12) e (3.13) para um dado $\bar{x} \in \mathcal{X}_1$, podemos obter um subgradiente $\bar{\gamma}$ da função $Q(\bar{x})$ a partir das soluções duais do problema de segundo estágio (3.11) para cada cenário ξ_i , isto é,

$$\bar{\gamma} = - \sum_{i=1}^N p_i T_i^T \pi_i \in \partial Q(\bar{x}). \quad (3.15)$$

Utilizando a definição do subgradiente da função $Q(\cdot)$ no ponto $\bar{x} \in \mathcal{X}_1$ e as equações (3.14) e (3.15) para dado $\bar{\gamma} \in \partial Q(\bar{x})$, tem-se que para todo $x \in \mathbb{R}^{n_1}$

$$\begin{aligned} Q(x) &\geq Q(\bar{x}) + \bar{\gamma}^T (x - \bar{x}) \\ &\geq \sum_{i=1}^N p_i \pi_i^T (h_i - T_i \bar{x}) - \left(\sum_{i=1}^N p_i T_i^T \pi_i \right)^T (x - \bar{x}) \\ &\geq \sum_{i=1}^N p_i \pi_i^T (h_i - T_i \bar{x}) - \sum_{i=1}^N p_i \pi_i^T T_i x + \sum_{i=1}^N p_i \pi_i^T T_i \bar{x} \\ &\geq \sum_{i=1}^N p_i \pi_i^T (h_i - T_i x). \end{aligned} \quad (3.16)$$

Esta última desigualdade representa que a função $Q(\cdot)$ não se encontra abaixo da linearização (hiperplano) que passa pelo ponto \bar{x} da função, pois ela é convexa. A expressão do lado direito da última desigualdade de (3.16) nomeamos de corte de Benders, o que finalmente nos mostra como utilizar as soluções de todos os N subproblemas para sua construção. Seria possível obter este mesmo resultado a partir do procedimento realizado no caso determinístico. Uma ilustração de que o corte de Benders se encontra abaixo da função Q , onde consideramos $\mathcal{X}_1 \subset \mathbb{R}$, pode ser vista na Figura 3.2.

Suponhamos que a função Q tenha n vértices. Como calcular todos os vértices pode ter um custo muito caro ou até mesmo ser inviável computacionalmente então, de modo análogo ao determinístico, utiliza-se um processo iterativo para calcular um vértice de cada vez a fim de aproximar a função Q , isto é, calculamos somente um subconjunto contendo $m \leq n$ destes vértices. Desta forma, aproximamos o valor esperado da função recurso pela função $\hat{Q}(\cdot)$ definida por

$$\begin{aligned} \hat{Q}(x) &= \min \quad r \\ &\text{sujeito a} \quad r \geq \sum_{i=1}^N p_i \pi_{i,j}^T (h_i - T_i x), \quad j = 1, \dots, m, \end{aligned} \quad (3.17)$$

Para simplificar a notação, definimos α_j e β_j por

$$\alpha_j = \sum_{i=1}^N p_i \pi_{i,j}^T h_i \quad e \quad \beta_j = - \sum_{i=1}^N p_i \pi_{i,j}^T T_i. \quad (3.18)$$

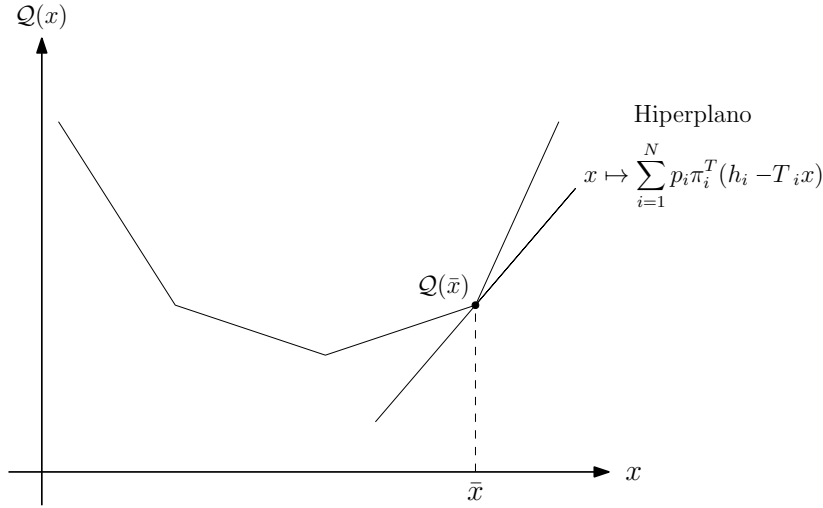


Figura 3.2: Hiperplano suporte do valor esperado da função recurso

Substituindo (3.17) e (3.18) no problema de primeiro estágio (3.5), obtemos:

$$\left\{ \begin{array}{l} \text{minimizar} \quad c^T x + \hat{Q}(x) \\ \text{sujeito a} \quad Ax = b \\ \quad \quad \quad x \geq 0, \end{array} \right. \implies \left\{ \begin{array}{l} \text{minimizar} \quad c^T x + r \\ \text{sujeito a} \quad Ax = b \\ \quad \quad \quad r \geq \beta_1 x + \alpha_1 \\ \quad \quad \quad \vdots \\ \quad \quad \quad r \geq \beta_m x + \alpha_m \\ \quad \quad \quad x \geq 0. \end{array} \right. \quad (3.19)$$

Como o problema (3.19) utiliza uma aproximação \hat{Q} da função Q , a partir de uma solução (\hat{x}, \hat{r}) de (3.19) podemos calcular um *limitante inferior* \underline{z} do verdadeiro valor ótimo de (3.5), isto é,

$$\underline{z} = c^T \hat{x} + \hat{r}. \quad (3.20)$$

Além disto, podemos calcular um *limitante superior* \bar{z} resolvendo o problema de segundo estágio (3.6) para este mesmo \hat{x} já calculado, obtendo

$$\bar{z} = c^T \hat{x} + Q(\hat{x}) = c^T \hat{x} + \sum_{i=1}^N p_i Q(\hat{x}, \xi_i). \quad (3.21)$$

Um critério de parada pode ser o mesmo utilizado no caso determinístico, isto é, quando a diferença entre os limitantes superior e inferior for menor que uma dada tolerância $\varepsilon > 0$, o algoritmo para, isto é,

$$\bar{z} - \underline{z} < \varepsilon. \quad (3.22)$$

Um algoritmo para resolver o problema de dois estágios utilizando todos os cenários se chama *Decomposição Aninhada*, que é muito semelhante ao *Algoritmo 2*, e se resume no seguinte:

Algoritmo 4 - Decomposição Aninhada em 2 estágios

Passo (1). Inicialização. Definir o valor esperado da função recurso aproximado $\hat{Q}(x) = 0$, limite superior $\bar{z} = \infty$, número de vértices $m = 0$ e tolerância $\varepsilon > 0$;

Passo (2). (Fase progressiva) Resolver o problema de primeiro estágio aproximado (3.19). Tome \hat{x} como a decisão tentativa. Este passo está ilustrado na Figura 3.3, em que a seta indica que a solução obtida no primeiro estágio ($t = 1$) será utilizada em todos os cenários ξ_i no segundo estágio ($t = 2$);

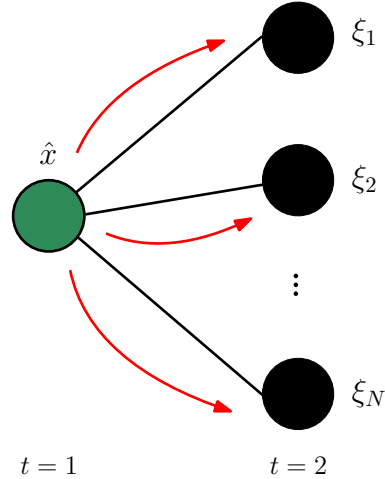


Figura 3.3: Fase progressiva

Passo (3). Calcular o limite inferior \underline{z} como em (3.20). Se $\bar{z} - \underline{z} \leq \varepsilon$, parar. Caso contrário, ir para o Passo (4);

Passo (4). Resolver o dual do problema de segundo estágio (3.11), isto é, calcular $Q(\hat{x})$. Atualizar \bar{z} como em (3.21);

Passo (5). (Fase recursiva) Atualizar o número de vértices $m \leftarrow m + 1$. A partir das soluções ótimas $\pi_{i,m}$ obtidas no Passo (4), calcular α_m e β_m , como em (3.18). Construa a aproximação da função $\hat{Q}(x)$ como em (3.17), utilizando m vértices. Este passo está ilustrado na Figura 3.4, onde as setas indicam que foram utilizados todos os cenários para a construção do hiperplano que aproxima a função Q , a qual está representada pela linha diagonal que sobrescreve o primeiro círculo;

Passo (6). Ir para o Passo (2).

Para problemas em mais estágios, o número de cenários cresce exponencialmente de acordo com o número de realizações do vetor aleatório em cada estágio. Nesse sentido, N será um número muito grande e impossibilitará a aplicação do algoritmo decomposição aninhada. Diante disto, é necessário utilizar estratégias que permitam o manuseio do problema com um número menor de cenários e que se tenha garantia de obter uma solução próxima do problema original. Um algoritmo que satisfaz estas condições é o de *Programação*

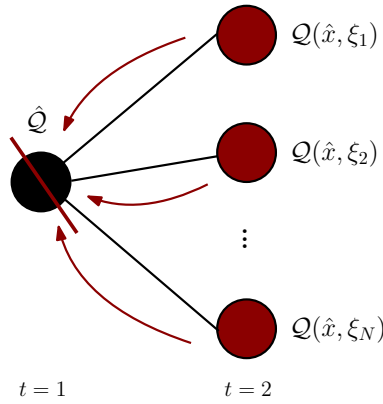


Figura 3.4: Fase recursiva

Dinâmica Dual Estocástica, que será apresentado para um programa linear estocástico em multiestágios, o que não deixa de ser válido para o problema em dois estágios.

A seguir mostramos que o programa linear estocástico em dois estágios também pode ser resolvido de forma direta quando o espaço amostral da variável aleatória ξ é finito.

3.3.4 Formulação equivalente determinística

Quando a variável aleatória ξ assume N valores, é possível escrever um único programa linear que inclui todas suas realizações, o qual pode ser resolvido de forma direta. Para isto, substituímos (3.8) e (3.9) no problema de primeiro estágio (3.5). Obtemos, então, um problema de programação linear geralmente de grande porte denominado *equivalente determinístico*:

$$\left\{ \begin{array}{ll} \text{minimizar} & c^T x + \sum_{i=1}^N p_i q_i^T y_i \\ \text{sujeito a} & Ax = b \\ & T_i x + W_i y_i = h_i \quad \forall i = 1, \dots, N \\ & x, y_i \geq 0 \quad \forall i = 1, \dots, N. \end{array} \right. \quad (3.23)$$

O problema linear, equivalente ao problema (3.23), a ser resolvido de forma direta é (2.26), o mesmo no caso determinístico. Utilizamos um índice nos dados referentes ao problema de primeiro estágio (3.5) (A_1, b_1, c_1) para diferenciá-los dos elementos do problema (2.26). A partir disto, faz-se um procedimento semelhante ao que foi feito na Seção 2.3.1 para resolvê-lo, sendo necessário redefinir a variável de decisão x , os vetores b e c e a matriz A da seguinte maneira:

- Agrupar as variáveis x_1 e y_i e os vetores b_1, c_1, q_i e h_i com $i = 1, \dots, N$, do primeiro e segundo estágios:

$$x = \begin{bmatrix} x_1 \\ y_1 \\ \vdots \\ y_N \end{bmatrix}, \quad b = \begin{bmatrix} b_1 \\ h_1 \\ \vdots \\ h_N \end{bmatrix}, \quad c = \begin{bmatrix} c_1 \\ p_1 q_1 \\ \vdots \\ p_N q_N \end{bmatrix};$$

Observação: Note que a dimensão da variável de decisão x é dada pela soma das dimensões dos vetores c e dos vetores q_i , $i = 1, \dots, N$. Além disto, já foi incluído o produto da probabilidade do cenário com o custo do problema de segundo estágio no vetor c , o que faz parte da função objetivo.

- Agrupar todas as matrizes A_1, T_i e W_i em blocos, onde os espaços em brancos são preenchidos com zeros e cada linha da matriz, a partir da segunda, representa um cenário ξ_i :

$$A = \begin{bmatrix} A_1 & & & & & \\ T_1 & W_1 & & & & \\ T_2 & & W_2 & & & \\ \vdots & & & \ddots & & \\ T_N & & & & & W_N \end{bmatrix}.$$

Resolver o problema (2.26) com estes dados pode acabar sendo inviável computacionalmente quando o número de cenários N é muito grande, pois envolve um grande número de variáveis. É por isto que se utiliza outras abordagens para resolvê-lo, como já foi descrito na seção anterior.

Na próxima seção apresentamos a aplicação da resolução direta no exemplo de despacho hidrotérmico da Seção 2.2.

3.3.5 Exemplo

Retomando o exemplo do despacho hidrotérmico da Seção 2.2, introduzimos a variável aleatória ξ , cujas componentes são as afluências no reservatório no segundo estágio. Suponhamos que esta variável assuma dois valores equiprováveis: $a_2^1 = 300 \text{ m}^3/\text{s}$ e $a_2^2 = 450 \text{ m}^3/\text{s}$, mantendo o valor da afluência no primeiro estágio $a_1^1 = 150 \text{ m}^3/\text{s}$. Uma forma de representar este problema é utilizando a árvore de cenários, o que se encontra na Figura 3.5.

Cada círculo da árvore de cenários é denominado nó e um cenário, neste caso, é considerado todo um caminho que começa no nó do primeiro estágio e termina em qualquer nó do segundo estágio. Desta forma, o número de possíveis cenários é dado pela quantidade de nós que há no último estágio $t = T$.

Este problema pode ser resolvido de diversas maneiras, mas neste momento resolvemos pela formulação equivalente determinística, que é uma forma direta de resolvê-lo. Esta formulação envolve todos os cenários e estágios em um único problema e por este motivo pode ser que o problema tenha um custo computacional muito caro, ou até mesmo ser intratável, por causa de sua dimensão.

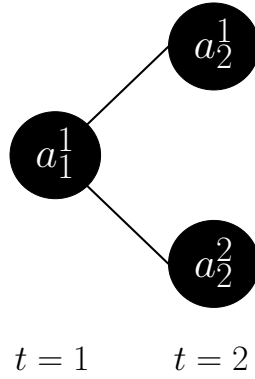


Figura 3.5: Árvore de cenários das afliências

Feitas estas observações, exibimos a modelagem do problema em sua formulação equivalente determinística detalhando a função objetivo, as variáveis e as restrições. Uma observação a ser feita é que agora há dois problemas de segundo estágio que se diferem somente pela afliência e, conseqüentemente, se diferem também nas variáveis de segundo estágio. Para diferenciá-las, incluímos um índice sobrescrito indicando o cenário associado. Além disto, as variáveis referentes ao primeiro estágio v_1, q_1, s_1 e d_1 e alguns dados fixos cd e ct_i , $i = 1, \dots, 4$ serão mantidos sem diferenciá-los por cenário. Segue então a modelagem do problema:

- Vetor decisão:

$$x = [v_1, v_2^1, v_2^2, q_1, q_2^1, q_2^2, s_1, s_2^1, s_2^2, d_1, d_2^1, d_2^2, gt_{1,1}, gt_{2,1}, gt_{3,1}, gt_{4,1}, \dots, gt_{1,2}^1, gt_{2,2}^1, gt_{3,2}^1, gt_{4,2}^1, gt_{1,2}^2, gt_{2,2}^2, gt_{3,2}^2, gt_{4,2}^2]^T;$$

- Função objetivo: soma dos custos das usinas térmicas e do déficit de energia nos dois estágios, sendo que no segundo estágio devemos calcular o valor esperado do custo considerando as probabilidades associadas aos dois cenários (lembre-se que os cenários foram considerados equiprováveis):

$$f(x) = \sum_{i=1}^4 ct_i \cdot gt_{i,1} + cd \cdot d_1 + \frac{1}{2} \left[\sum_{i=1}^4 ct_i \cdot gt_{i,2}^1 + cd \cdot d_2^1 + \sum_{i=1}^4 ct_i \cdot gt_{i,2}^2 + cd \cdot d_2^2 \right];$$

- Balanço hídrico: o volume atualizado do reservatório é a diferença entre o volume inicial adicionado à afliência e a soma das vazões vertida e turbinada. Note que as variáveis associadas à afliência, vazão turbinada e vazão vertida são multiplicadas por κ devido a conversão das unidades de medida;

$$\begin{aligned} v_1 &= 2050 + \kappa \cdot a_1 - \kappa(q_1 + s_1) \\ v_2^1 &= v_1 + \kappa \cdot a_2^1 - \kappa(q_2^1 + s_2^1) \\ v_2^2 &= v_1 + \kappa \cdot a_2^2 - \kappa(q_2^2 + s_2^2); \end{aligned}$$

- Atendimento à demanda: a soma da geração de energia das usinas termelétricas e hidrelétricas junto

com o déficit é igual a demanda;

$$\begin{aligned} \sum_{i=1}^4 gt_{i,1} + d_1 + \rho \cdot q_1 &= D \\ \sum_{i=1}^4 gt_{i,2}^1 + d_2^1 + \rho \cdot q_2^1 &= D \\ \sum_{i=1}^4 gt_{i,2}^2 + d_2^2 + \rho \cdot q_2^2 &= D; \end{aligned}$$

- Limitações das variáveis:

$$\begin{aligned} 0 &\leq v_1, v_2^1, v_2^2 \leq v_{max} \\ 0 &\leq q_1, q_2^1, q_2^2 \leq q_{max} \\ s_1, s_2^1, s_2^2 &\geq 0 \\ 0 &\leq gt_{i,1}, gt_{i,2}^1, gt_{i,2}^2 \leq gt_{i(max)}, \quad i = 1, \dots, 4 \\ d_1, d_2^1, d_2^2 &\geq 0. \end{aligned}$$

O problema deve incluir as restrições para os dois cenários, o que está sendo representado pelos subíndices nas matrizes T e W e no vetor h . Segue o problema em sua formulação equivalente determinística:

$$\left\{ \begin{array}{l} \text{minimizar} \quad c^T x \\ \text{sujeito a} \quad \begin{bmatrix} A \\ T_1 + W_1 \\ T_2 + W_2 \end{bmatrix} x = \begin{bmatrix} b \\ h_1 \\ h_2 \end{bmatrix} \\ \ell \leq x \leq u. \end{array} \right. \quad (3.24)$$

Todas as matrizes A, T_i e W_i e vetores b e h são construídos de forma semelhante ao caso determinístico já resolvido, sendo que entre os pares de matrizes T_1 e T_2 e W_1 e W_2 diferem entre si somente na posição de alguns valores correspondentes às variáveis associadas aos cenários. Além disto, nesta modelagem foi necessário realizar as somas $T_1 + W_1$ e $T_2 + W_2$ ao invés de separar as matrizes em blocos, o que é justificado pelo mesmo motivo do primeiro exemplo da seção 2.2.

Pelo fato do problema ter dois cenários e estágios, são obtidas duas soluções de segundo estágio, as quais levam em consideração a consequência da decisão tomada no primeiro estágio e o conhecimento das realizações da variável aleatória. No entanto, a parte importante na solução ótima obtida é aquela referente ao primeiro estágio, pois ela representa a decisão que deve ser tomada naquele exato momento. Resolvendo o problema obtemos a solução de primeiro estágio x e as soluções de segundo estágio y_1 e y_2 , as quais estão associadas aos cenários:

$$x = \begin{bmatrix} v_1 \\ q_1 \\ s_1 \\ d_1 \\ gt_{1,1} \\ gt_{2,1} \\ gt_{3,1} \\ gt_{4,1} \end{bmatrix} = \begin{bmatrix} 773,26 \\ 642,57 \\ 0 \\ 0 \\ 100 \\ 150 \\ 133,13 \\ 0 \end{bmatrix}, \quad y_1 = \begin{bmatrix} v_2^1 \\ q_2^1 \\ s_2^1 \\ d_2^1 \\ gt_{1,2}^1 \\ gt_{2,2}^1 \\ gt_{3,2}^1 \\ gt_{4,2}^1 \end{bmatrix} = \begin{bmatrix} 0 \\ 598,32 \\ 0 \\ 0 \\ 100 \\ 150 \\ 175,61 \\ 0 \end{bmatrix} \quad e \quad y_2 = \begin{bmatrix} v_2^2 \\ q_2^2 \\ s_2^2 \\ d_2^2 \\ gt_{1,2}^2 \\ gt_{2,2}^2 \\ gt_{3,2}^2 \\ gt_{4,2}^2 \end{bmatrix} = \begin{bmatrix} 0 \\ 748,32 \\ 0 \\ 0 \\ 100 \\ 150 \\ 31,61 \\ 0 \end{bmatrix}$$

e o valor ótimo da função objetivo obtido foi

$$\begin{aligned} & \text{Custo 1º estágio} + \text{Valor esperado do custo 2º estágio} \\ & R\$ 9.325,28 + \frac{1}{2}(R\$ 11.024,35 + R\$ 5.264,35) \\ & R\$ 17.469,63. \end{aligned}$$

Um detalhe interessante a ser notado na solução deste problema estocástico de dois estágios é que não é possível escolher uma solução que seja ótima para todas as realizações de afluências previstas. No entanto, a solução x , de primeiro estágio, é a melhor decisão que pode ser tomada levando em consideração os possíveis cenários. Além disto, retomando ao caso determinístico, podemos dizer que $R\$ 17.469,63 - R\$ 14.589,63 = R\$ 2.880,00$ é o custo de "proteção", o que é consideravelmente menor em relação ao valor de $R\$ 21.485,90 - R\$ 17.469,63 = R\$ 4.016,27$ que seriam gastos se a afluência do segundo estágio não fosse exatamente o que se previa.

3.4 Programação linear estocástica em multiestágios

Muitos problemas do nosso cotidiano envolvem um planejamento de um período longo de tempo ou mais periódico, por exemplo, um planejamento de 5 anos pode ser feito de várias formas, mensal, semestral, anual. Uma modelagem destes problemas por programas estocásticos em dois estágios não seria mais adequada, sendo necessário estendê-la de forma a incorporar, em sua estrutura, esse horizonte de tempo. Para isto, utiliza-se um programa estocástico em multiestágios, onde são considerados $T > 2$ períodos de tempo, ou estágios. Neste tipo de programa, as incertezas do processo estocástico ξ_1, \dots, ξ_T são reveladas ao longo do tempo e cada decisão x_t , $t = 2, \dots, T$ depende de todos os dados $\xi_{[t]}$ ⁴ já revelados e independe das realizações futuras ξ_{t+1}, \dots, ξ_T , de forma a se adaptar às incertezas do processo. Isto pode ser representado da seguinte forma, [27, Capítulo 3]:

$$\text{decisão } (x_1) \rightsquigarrow \text{observação } (\xi_2) \rightsquigarrow \text{decisão } (x_2) \rightsquigarrow \dots \rightsquigarrow \text{observação } (\xi_T) \rightsquigarrow \text{decisão } (x_T).$$

⁴Esta notação $\xi_{[t]} = (\xi_1, \dots, \xi_t)$ denota o histórico do cenário ξ até o estágio t .

Para definir matematicamente um programa estocástico em multiestágios, estendemos o que foi apresentado na Seção 3.2. Consideramos o espaço de probabilidades (Ξ, \mathcal{F}, P) , sendo um cenário definido por uma realização da variável aleatória ξ para cada estágio $t \in (1, \dots, T)$, isto é, $\xi = (\xi_1, \dots, \xi_T) \in \Xi$ e \mathcal{F} a σ -álgebra gerada pelo conjunto $\Xi = \Xi_1 \times \dots \times \Xi_T \subseteq \mathbb{R}^{d_1} \times \dots \times \mathbb{R}^{d_T}$.

3.4.1 Formulação aninhada

De forma genérica, considerando o processo estocástico ξ , podemos escrever um problema estocástico em T estágios em sua *formulação aninhada*:

$$\min_{x_1 \in \mathcal{X}_1} f_1(x_1) + \mathbb{E} \left[\inf_{x_2 \in \mathcal{X}_2(x_1, \xi_{[2]})} f_2(x_2, \xi_2) + \mathbb{E} \left[\dots + \mathbb{E} \left[\inf_{x_T \in \mathcal{X}_T(x_{T-1}, \xi_{[T]})} f_T(x_T, \xi_T) \right] \right] \right], \quad (3.25)$$

onde as variáveis de decisão são $x_t \in \mathcal{X}_t \subseteq \mathbb{R}^{n_t}$, $t = 1, \dots, T$, os conjuntos viáveis nos estágios $t \geq 2$ são fechados e definidos pela multifunção $\mathcal{X}_t : \mathbb{R}^{n_{t-1}} \times \Xi_{[t]} \rightrightarrows \mathbb{R}^{n_t}$ e as funções $f_t : \mathbb{R}^{n_t} \times \Xi_t \rightarrow \mathbb{R}$ são contínuas. Lembre-se que no primeiro estágio os dados determinados por ξ_1 são considerados determinísticos, ou seja, $\mathcal{X}_1 \subset \mathbb{R}^{n_1}$ e $f_1 : \mathcal{X}_1 \rightarrow \mathbb{R}$. A sequência de decisões $x(\xi_{[t]}) = (x_1(\xi), \dots, x_t(\xi)) \in \mathbb{R}^{n_t}$, com $\eta_t = \sum_{i=1}^t n_i$ é denominada por *política*. Se a política satisfizer todas as restrições de viabilidade, isto é, $x_t(\xi_{[t]}) \in \mathcal{X}_t(x_{t-1}(\xi_{[t-1]}), \xi_t)$, $t = 2, \dots, T$, c.p. 1, ela é denominada *política viável*.

Veja que as decisões $x_t(\xi)$ devem depender somente das informações disponíveis até o tempo t , ou seja, $x_{[t-1]}$ e $\xi_{[t]}$, e não das informações futuras. Este é o princípio que denomina-se *restrições de implementaridade* (em inglês, *nonanticipativity constraints*), a qual é definida em [22, Definição 2.5] da seguinte maneira:

Definição 3.4.1. *Em geral, se dois cenários têm a mesma sequência de nós entre o primeiro e o t -ésimo estágios, eles têm a mesma parcela de informações durante estes períodos. Consequentemente, decisões associadas a tais cenários devem ser idênticas até o t -ésimo período. Este requerimento é conhecido como condição de implementabilidade.*

O problema (3.25) é dito linear se as funções objetivos e as funções restrições são lineares, ou seja,

$$\begin{aligned} f_t(x_t, \xi_t) &= c_t^T x_t, \quad \mathcal{X}_1 = \{x_1 : A_1 x_1 = b_1, x_1 \geq 0\}, \\ \mathcal{X}_t(x_{t-1}, \xi_t) &= \{x_t : B_t x_{t-1} + A_t x_t = b_t, x_t \geq 0\}, \quad t = 2, \dots, T, \end{aligned} \quad (3.26)$$

onde as dimensões dos vetores são $b_t \in \mathbb{R}^{m_t}$ e $c_t \in \mathbb{R}^{n_t}$ e as dimensões das matrizes A_t e B_t são compatíveis com os vetores. Estas matrizes e vetores definem o cenário $\xi_t = (c_t, A_t, B_t, b_t) \in \Xi_t$ da variável aleatória $\xi_t(\omega) = (c_t(\omega), A_t(\omega), B_t(\omega), b_t(\omega))$, não sendo necessário que todos os elementos sejam aleatórios, onde $\xi_1 = (c_1, A_1, b_1)$ é determinístico.

Substituindo (3.26) em (3.25), obtemos a formulação aninhada de um programa linear estocástico em multiestágios:

$$\min_{\substack{A_1 x_1 = b_1 \\ x_1 \geq 0}} c_1^T x_1 + \mathbb{E} \left[\min_{\substack{B_2 x_1 + A_2 x_2 = b_2 \\ x_2 \geq 0}} c_2^T x_2 + \mathbb{E} \left[\dots + \mathbb{E} \left[\min_{\substack{B_T x_{T-1} + A_T x_T = b_T \\ x_T \geq 0}} c_T^T x_T \right] \right] \right]. \quad (3.27)$$

Veja que se a variável aleatória ξ for contínua, calcular o valor esperado do problema (3.25) pode ser muito difícil ou até mesmo impossível, ainda mais que são considerados T estágios. Novamente faz-se necessário discretizar a variável aleatória de modo que seja viável resolver o problema. Então, discretizamos cada componente da variável aleatória num número finito N_t de realizações, isto é, $\Xi_t = (\xi_t^1, \dots, \xi_t^{N_t})$ com suas respectivas probabilidades p_t^i , onde $t = 2, \dots, T$ e $i = 1, \dots, N_t$. A seguir veremos como utilizar estas discretizações para representar o conjunto de possibilidades das realizações da variável aleatória ξ .

3.4.2 Árvore de cenários

Uma forma muito utilizada para representar todo o processo estocástico envolvido no problema é por árvore de cenários, a qual possibilita visualizar todos os possíveis cenários e as relações entre as decisões de cada estágio. É ilustrado na Figura 3.6 um exemplo do processo estocástico de um problema de 4 estágios com duas discretizações do espaço amostral Ξ_t por estágio.

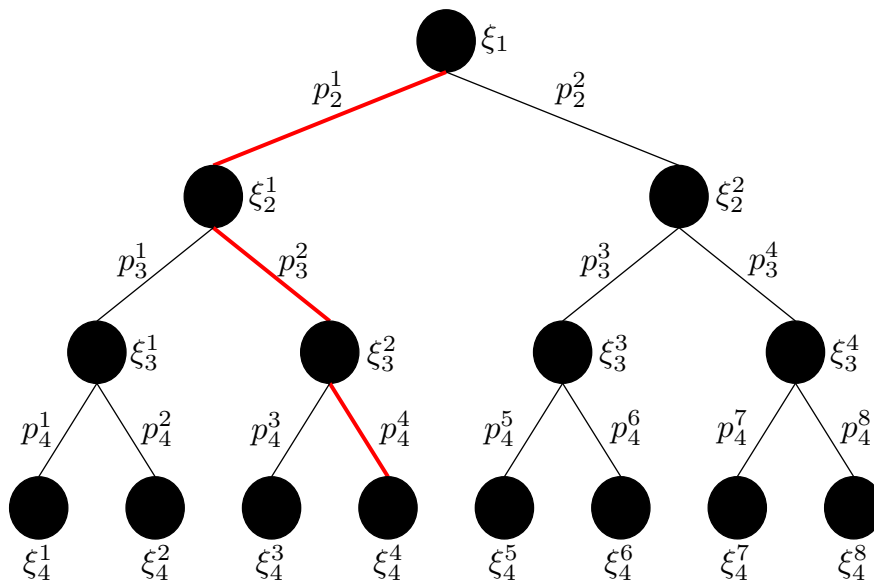


Figura 3.6: Árvore de cenários para 4 estágios

Na Figura 3.6, os círculos são denominados de *nós* e as conexões entre os nós são os *arcos*. Os valores p_i^j que estão nos arcos representam a probabilidade condicional de mover de um nó ao próximo. Qualquer trajeto que se inicia no primeiro estágio e percorre uma única vez todos os estágios até o último é denominado por *cenário*. O nó que está conectado por um arco no estágio imediatamente anterior é denominado *antecessor* e o nó que está conectado no estágio imediatamente posterior é denominado *sucessor*. A probabilidade de ocorrer um certo nó é dada pelo produto das probabilidades da sequência de nós que o antecedem, até o primeiro estágio. Por exemplo, a probabilidade de ocorrer o nó ξ_4^4 é o produto da probabilidade p_4^4 com as probabilidades dos nós antecessores, os quais estão destacados pelos arcos em vermelho, que são p_3^2 e p_2^1 , ficando $p_2^1 \cdot p_3^2 \cdot p_4^4$. Esta probabilidade coincide com a probabilidade de ocorrer o quarto cenário que é dado

pela sequência de realizações $(\xi_1, \xi_2^1, \xi_3^2, \xi_4^4)$. Veja que como há 4 estágios e 2 discretizações em cada, o número total de cenários é dado por $2^{4-1} = 8$. Se o número de discretizações de cada estágio é N_t , $t = 1, \dots, T$, e se todos os nós em cada estágio possui a mesma quantidade de nós sucessores, o número de cenários é $\prod_{i=1}^T N_i$.

Um detalhe a ser percebido é que não é necessário cada estágio possuir um mesmo número de realizações. Isto pode ser visto na Figura 3.7, a qual ilustra outra possibilidade de árvore de cenários com 4 estágios.

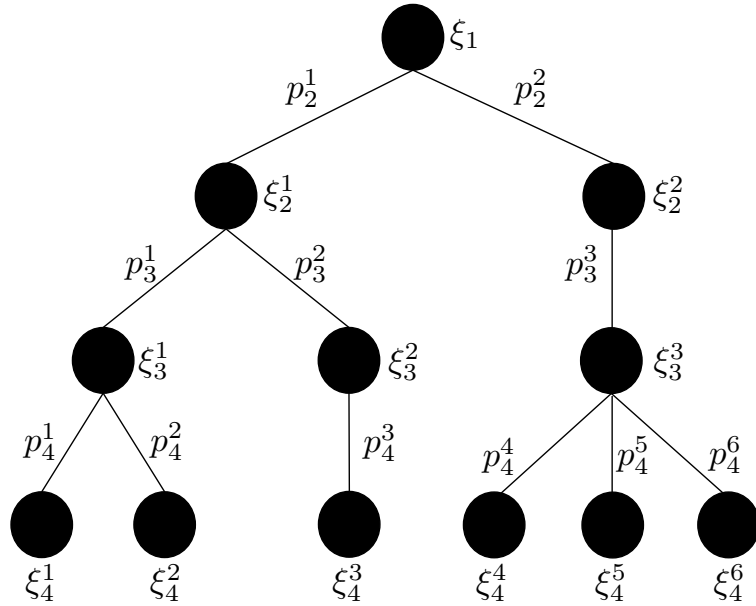


Figura 3.7: Árvore de cenários para 4 estágios

No problema que estamos interessados em resolver, o de despacho hidrotérmico, o processo $\{\xi_t\}$ é considerado *independente por estágios* (em inglês, *stagewise independent*), o que quer dizer que ξ_t é estocasticamente independente de $\xi_{[t-1]}$, $t = 2, \dots, T$. Consideramos esta hipótese no problema por simplicidade, onde as afluências, num determinado estágio, não dependem de suas ocorrências em estágios anteriores. Um processo independente por estágios é um processo *Markoviano*, isto é, se para todo $t = 1, \dots, T$, a distribuição condicional de ξ_t dado $\xi_{[t-1]}$ é a mesma que a distribuição condicional de ξ_t dado ξ_{t-1} [27, p. 63].

Segue destas considerações que as realizações em cada estágio são as mesmas para todos os nós antecedentes. Como consequência disso, a quantidade de nós sucessores para cada nó dentro de um mesmo estágio é sempre a mesma, o que ocorre na Figura 3.6, mas não na Figura 3.7, pois, já no segundo estágio, as quantidades de nós sucessores das realizações ξ_2^1 e ξ_2^2 são diferentes, assim como das realizações ξ_3^1, ξ_3^2 e ξ_3^3 . Para que a árvore de cenários da Figura 3.6 apresente as mesmas realizações em cada estágio, e assim represente um processo independente por estágios, é necessário que as seguintes igualdades ocorram: $\xi_3^1 = \xi_3^3$, $\xi_3^2 = \xi_3^4$, $\xi_4^1 = \xi_4^3 = \xi_4^5 = \xi_4^7$ e $\xi_4^2 = \xi_4^4 = \xi_4^6 = \xi_4^8$, onde a árvore pode ser reescrita em sua forma reduzida como na Figura 3.8.

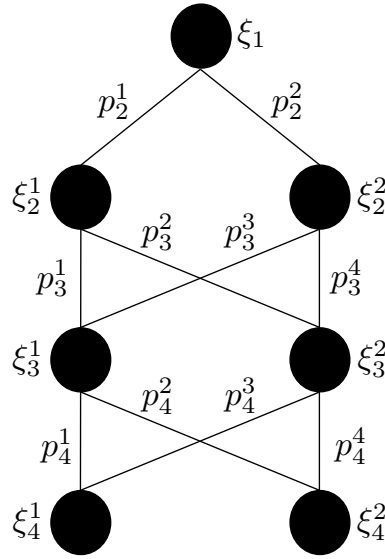


Figura 3.8: Árvore de cenários reduzida para 4 estágios

3.4.3 Formulação dinâmica - Algoritmo PDDE

O problema linear estocástico em multiestágios (3.27) também pode ser escrito em sua *formulação dinâmica*:

- A função recurso Q_t , para $t = 1, \dots, T - 1$, é dada por:

$$Q_{t+1}(x_t, \xi_{[t]}) = \mathbb{E}_{|\xi_{[t]}}[Q_{t+1}(x_t, \xi_{[t+1]})];$$

- No estágio $t = T$

$$\begin{aligned} Q_T(x_{T-1}, \xi_{[T]}) &= \min && c_T^T x_T \\ &\text{sujeito a} && B_T x_{T-1} + A_T x_T = b_T \\ &&& x_T \geq 0; \end{aligned}$$

- Nos estágios $t = 2, \dots, T - 1$

$$\begin{aligned} Q_t(x_{t-1}, \xi_{[t]}) &= \min && c_t^T x_t + Q_{t+1}(x_t, \xi_{[t]}) \\ &\text{sujeito a} && B_t x_{t-1} + A_t x_t = b_t \\ &&& x_t \geq 0; \end{aligned}$$

- No estágio $t = 1$

$$\left\{ \begin{array}{l} \text{minimizar} \quad c_1^T x_1 + Q_2(x_1, \xi_1) \\ \text{sujeito a} \quad A_1 x_1 = b_1 \\ \quad \quad \quad x_1 \geq 0. \end{array} \right.$$

Assim como foi mostrado nas Seções 2.3.1 e 3.3.3, podemos estender o processo de aproximação das funções Q_t para os estágios $t = 2, \dots, T$, utilizando os cortes de Benders, pois elas são convexas e lineares por partes. Também é possível estender o cálculo dos coeficientes α e β . Deste modo, omitimos os procedimentos

já realizados e exibimos somente a formulação dinâmica aproximada e os coeficientes dos cortes na k -ésima iteração de um algoritmo que calcula um corte por iteração.

- No estágio $t = T$

$$\begin{aligned} Q_T(x_{T-1}, \xi_{[T]}) &= \min && c_T^T x_T \\ &\text{sujeito a} && B_T x_{T-1} + A_T x_T = b_T \\ &&& x_T \geq 0; \end{aligned} \quad (3.28)$$

- Nos estágios $t = 2, \dots, T-1$

$$\begin{aligned} \underline{Q}_t(x_{t-1}, \xi_{[t]}) &= \min && c_t^T x_t + r_{t+1} \\ &\text{sujeito a} && B_t x_{t-1} + A_t x_t = b_t \\ &&& r_{t+1} \geq \alpha_{t+1}^j + \beta_{t+1}^j x_t \quad j = 1, \dots, k \\ &&& x_t \geq 0; \end{aligned} \quad (3.29)$$

- No estágio $t = 1$

$$\left\{ \begin{array}{l} \text{minimizar} \quad c_1^T x_1 + r_2 \\ \text{sujeito a} \quad A_1 x_1 = b_1 \\ \quad \quad \quad r_2 \geq \alpha_2^j + \beta_2^j x_1 \quad j = 1, \dots, k \\ \quad \quad \quad x_2 \geq 0; \end{array} \right. \quad (3.30)$$

- Os valores α_t^k e β_t^k no estágio $t = T$ são dados por (3.18). Já para os estágios $t = T-1, \dots, 2$, os valores são dados pela generalização dos coeficientes dos casos determinístico em multiestágios (2.32) e estocástico em dois estágios (3.18), o que não é difícil de ser feito, pois, em resumo, corresponde a aplicar o valor esperado nos coeficientes α_t^k e β_t^k de (2.32), resultando em

$$\alpha_t^k = \mathbb{E}_{|\xi_{t-1}} \left[\pi_t^{kT} b_t + \sum_{i=1}^k \theta_t^i \alpha_{t+1}^i \right] \quad \text{e} \quad \beta_t^k = -\mathbb{E}_{|\xi_{t-1}} \left[\pi_t^{kT} B_t \right]. \quad (3.31)$$

Como o problema em questão deste trabalho é o de despacho hidrotérmico, consideramos, por simplicidade, que o processo estocástico é independente por estágios, o que nos permite escrever $\underline{Q}_{t+1}(x_t, \xi_{[t]}) = \underline{Q}_{t+1}(x_t)$. Neste caso, uma característica a ser notada é que todos os nós de um mesmo estágio possuem realizações futuras iguais, isto é, fixado os nós ξ_t^i e ξ_t^j , as realizações futuras serão as mesmas, para todo i, j ou, em outras palavras, a função \underline{Q}_{t+1} é a mesma em cada estágio, independente do nó. Isto nos sugere que, a partir de uma única política viável $\{x_1, \dots, x_T\}$, podemos aproximar cada função \underline{Q}_t , $t = 2, \dots, T$ por um hiperplano, sem a necessidade de envolver todos os cenários para o cálculo destas aproximações. Além disto, como a função \underline{Q}_t é única para cada estágio, os hiperplanos calculados podem ser compartilhados entre todos os nós de um mesmo estágio. Esta é uma das principais vantagens de um processo estocástico independente por estágios, o qual o algoritmo PDDE utiliza.

O processo iterativo da PDDE é semelhante aos outros algoritmos já apresentados, pois ele consiste em aproximar as funções \underline{Q}_t por hiperplanos. Note que os outros algoritmos utilizavam toda a estrutura

do problema para aproximar as funções e resolver o problema. No entanto, é comum que um programa estocástico em multiestágios possua dimensões muito elevadas, o que torna inviável utilizar esta mesma abordagem. É baseado nesta dificuldade que a PDDE trabalha, pois utiliza somente um subconjunto de informações do problema durante suas iterações para construir as aproximações.

Suponhamos então que a Figura 3.6 represente a árvore de cenários do problema envolvido, onde as probabilidades p_t^i são omitidas e os nós são ajustados para que o processo estocástico seja independente por estágios. Note que poderíamos utilizar a Figura 3.8, mas não achamos conveniente para a explicação. Na fase progressiva da PDDE, consideramos somente um cenário, ou seja, calculamos somente uma política, o que pode ser visto na sequência de árvores ilustradas na Figura 3.9

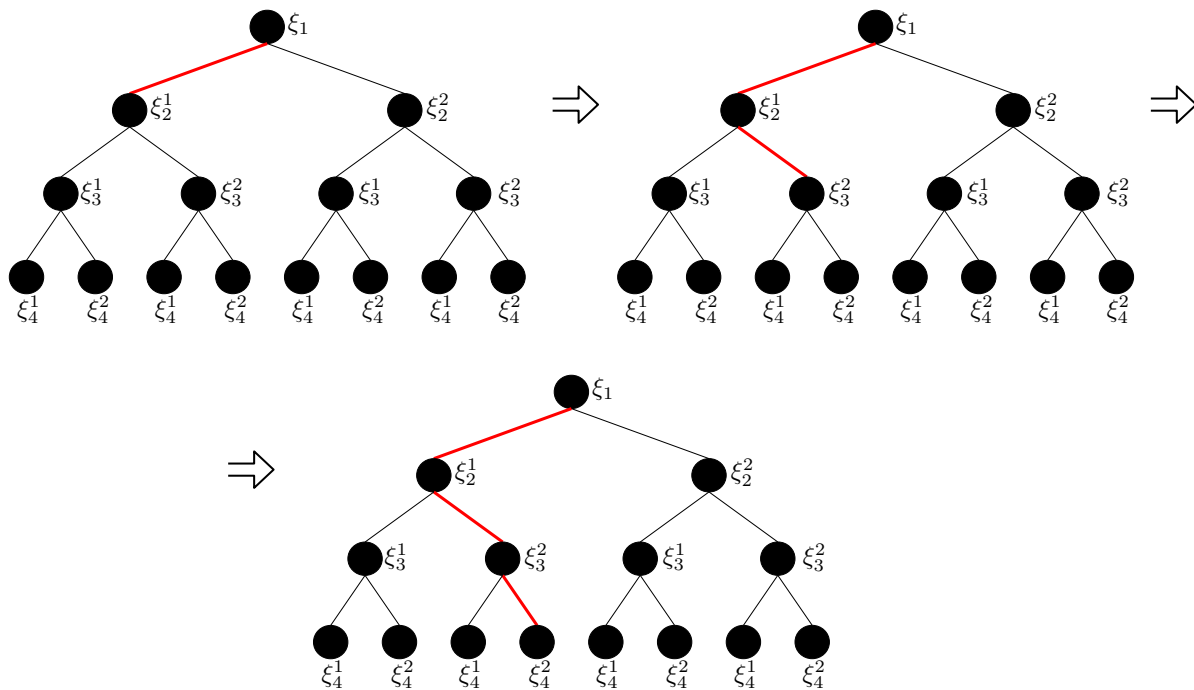


Figura 3.9: Fase progressiva do algoritmo PDDE

A fase recursiva é onde as aproximações das funções Q_t são realizadas. Nesta fase são calculados os multiplicadores de Lagrange das restrições dos problemas referentes a todas realizações da variável aleatória no estágio. Por exemplo, o último nó do cenário é ξ_4^2 , então, utilizamos os problemas referentes aos nós ξ_4^1 e ξ_4^2 para construir um hiperplano que será utilizado no estágio anterior. Retrocedendo um estágio, como o nó ξ_3^2 faz parte do cenário escolhido, utilizamos os problemas referentes aos nós ξ_3^1 e ξ_3^2 para construir outro hiperplano, e esse processo se repete até o primeiro estágio. Lembre-se que o hiperplano de cada estágio pode ser compartilhado entre todos os outros nós do mesmo estágio, pois a função Q_t é a mesma para cada estágio. Este procedimento é ilustrado na sequência de árvores de cenários na Figura 3.10.

O número de cenários na fase progressiva⁵ pode ser mais de um e também pode variar ao longo das

⁵Na literatura encontra-se este termo como *simulação* para o algoritmo PDDE

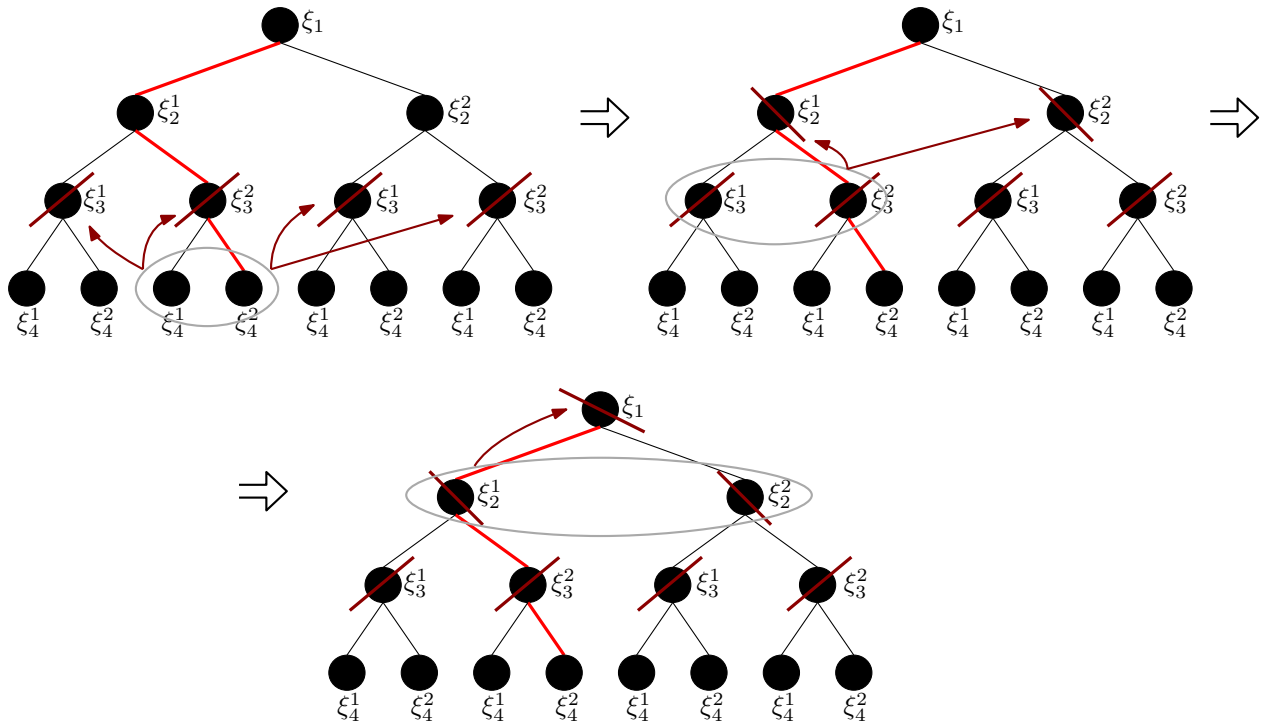


Figura 3.10: Fase recursiva do algoritmo PDDE

iterações. Para cada cenário escolhido é possível construir um corte para as funções Q_t . É por isto que quanto maior a quantidade de cenários escolhidos na fase progressiva, melhor serão as aproximações e maior será o esforço computacional para realizar uma iteração do algoritmo. Além destas considerações, um limite inferior \underline{z}^k na iteração k para o algoritmo é o valor ótimo do problema aproximado de primeiro estágio (3.30), ou seja, se (x_1^k, r_2^k) é uma solução, então

$$\underline{z}^k = c_1^T x_1^k + r_2^k. \quad (3.32)$$

Como não são utilizados todos os cenários no algoritmo para a fase progressiva, é possível ter somente uma estimativa \tilde{z}^k do limite superior \bar{z}^k na iteração k . Considere $\Xi_K = (\xi^1, \dots, \xi^K)$ o conjunto de todos os cenários do problema e $\hat{\Xi}_N^k = (\hat{\xi}^1, \dots, \hat{\xi}^N)$, com $N < K$, um subconjunto de cenários escolhidos para a fase progressiva na k -ésima iteração. Uma estimativa para o limite superior na iteração k é

$$\tilde{z}^k = \mathbb{E}_{\hat{\Xi}_N^k} \left[\sum_{i=1}^T c_t^T x_t^k \right]. \quad (3.33)$$

O critério utilizado nos algoritmos anteriores já não é mais garantido, pois temos uma estimativa do limite superior, o que pode acabar sendo menor que o limite inferior em alguma iteração. Uma possibilidade de critério de parada é sugerida em [11], a qual utiliza a simulação Monte Carlo para geração dos cenários e o desvio padrão σ dos limites superiores dos cenários em relação à média dos mesmos para construir um intervalo de confiança. O custo de um dado cenário i é

$$\bar{z}_i = c_1^T x_1 + \sum_{t=2}^T c_t^T x_t^i \quad (3.34)$$

e o custo total é

$$\bar{z}_{TOT} = c_1^T x_1 + \sum_{t=2}^T \sum_{i=1}^N \frac{c_t^T x_t^i}{N}, \quad (3.35)$$

onde N é o número de cenários. Temos então que o desvio padrão σ é

$$\sigma = \sqrt{\frac{\sum_{i=1}^N (\bar{z}_{TOT} - \bar{z}_i)^2}{N}}. \quad (3.36)$$

Então, em [11] é proposto que o algoritmo converge quando o limite inferior \underline{z} estiver dentro de um intervalo de confiança de 95%, isto é,

$$\bar{z}_{TOT} - 1,96 \frac{\sigma}{\sqrt{N}} \leq \underline{z} \leq \bar{z}_{TOT} + 1,96 \frac{\sigma}{\sqrt{N}}. \quad (3.37)$$

Em [16, p. 67] é dito que quanto maior for a variância e o nível de confiança de (3.37), o algoritmo irá parar mais cedo com este critério. Então é sugerido considerar, com 95% de confiança, que o problema foi resolvido com uma precisão $\varepsilon > 0$ se a seguinte diferença é satisfeita:

$$(\bar{z}_{TOT} + 1,96 \frac{\sigma}{\sqrt{N}}) - \underline{z} < \varepsilon. \quad (3.38)$$

Situações em que o limite inferior \underline{z} começa a se estabilizar ao longo das iterações [32] ou o algoritmo alcança um número máximo de iterações predefinido são consideradas como outros possíveis critérios de parada para a PDDE.

Considerando então que o processo estocástico é independente por estágios e que o processo de geração de cenários é dado pela simulação Monte Carlo, segue o algoritmo.

Algoritmo 5 - Programação Dinâmica Dual Estocástica multiestágios

Passo (1). Inicialização. Definir o valor esperado das funções recurso aproximado $r_t = 0$, $t = 2, \dots, T$, limite superior $\bar{z} = \infty$, número de vértices $m = 0$, número de iterações $k = 0$ e tolerância $\varepsilon > 0$;

Passo (2). (Fase progressiva) Selecionar N_k cenários da árvore de cenários. Resolver o problema de primeiro estágio aproximado (3.30). Tome \hat{x}_1 como a decisão tentativa e calcular o limite inferior \underline{z} como em (3.32);

Passo (3). Para $t = 2, \dots, T$ e para as realizações ξ_t^i , $i = 1, \dots, N$, da variável aleatória, calcular os problemas aproximados (3.29) e (3.28) utilizando as soluções x_{t-1}^i obtidas no estágio anterior do seu respectivo cenário, obtendo as soluções x_t^i . Este passo pode ser ilustrado semelhante à Figura 3.9, onde se destacariam N caminhos do primeiro ao último estágio. Calcular a estimativa do limite superior \bar{z}^k e o desvio padrão σ como em (3.35) e (3.36), respectivamente;

Passo (4). Verificar o critério de parada (3.38). Se não satisfizer, prosseguir para o próximo passo, caso contrário pare;

Passo (5). (Fase recursiva) Atualizar o número de vértices $m \leftarrow m + N$. Para cada estágio $t = T, \dots, 2$, para cada cenário $i = 1, \dots, N$ e para cada realização da variável aleatória ξ_t^j , $j = 1, \dots, NR_t$, onde NR_t é o número de realizações da variável aleatória no estágio t , calcular os multiplicadores de Lagrange π_t^i e θ_t^j das restrições de igualdade e desigualdade, respectivamente, dos problemas (3.29) e (3.28). Utilizar (3.18) para $t = T$ e (3.31) para $t = T - 1, \dots, 2$ para calcular os coeficientes dos cortes, substituindo-os nos problemas (3.29) e (3.30). Este passo poderia ser ilustrado pela Figura (3.10), onde, ao invés de haver somente um grupo circulado no último estágio na primeira árvore de cenários, haveriam N ;

Passo (6). Atualizar o número de iterações $k \leftarrow k + 1$. Ir para o Passo (2).

Veja que o critério de parada do algoritmo poderia não ser único e também poderia ser qualquer um daqueles que foram citados nesta seção e, dependendo de qual critério a ser utilizado, ele deveria ser colocado em um passo apropriado.

Caso o problema não fosse independente por estágios, cada nó da árvore de cenários teria uma função \mathcal{Q} própria. Deste modo, o algoritmo utilizado para este tipo de problema seria a extensão do algoritmo 4, o qual pode ser inviável computacionalmente para este tamanho de problema. Dependendo do caso, pode ser possível transformar este problema para o caso independente por estágios adicionando variáveis de estado ao modelo, o que pode ser visto em [16, p.63], assim como as propriedades de convergência do algoritmo PDDE aplicadas à formulação dinâmica do problema SAA.

3.4.4 Formulação equivalente determinística

Da mesma forma que foi feito em dois estágios, podemos agrupar todos os cenários e resolver um único problema de grande porte de forma direta, o qual é denominado por *equivalente determinístico*. Primeiramente são feitas algumas considerações para o entendimento da sua formulação:

- Consideramos que o espaço amostral de cada estágio Ξ_t , $t = 2, \dots, T$ foi discretizado em N_t elementos;
- Denotamos $\xi_t = (c_t, A_t, b_t)$ um elemento de Ξ_t ou, em outras palavras, algum nó do estágio t . Por isto, quando utilizarmos o subíndice ou superíndice ξ_t num vetor ou matriz, queremos dizer que esta matriz ou vetor é de algum nó $\xi_t \in \Xi_t$;
- Para não sobrecarregar as notações neste momento, não utilizamos qualquer notação na formulação para especificar qual o nó $\xi_t \in \Xi_t$ que está sendo utilizado, como foi feito nas figuras anteriores;
- Supomos que todas as realizações dentro de um mesmo estágio sejam equiprováveis, isto é, a probabilidade $\rho_{(\xi_t)}$ de qualquer arco entre os nós dos estágios $t - 1$ e t é $\rho_{(\xi_t)} = 1/N_t$;
- A probabilidade $p_{(\xi_t)}$ de alcançar um nó $\xi_t \in \Xi_t$ é o produto das probabilidades dos nós que formam um caminho do primeiro estágio até o nó ξ_t , o que é dado por $p_{(\xi_t)} = \rho_{(\xi_2)} \cdot \rho_{(\xi_3)} \cdot \dots \cdot \rho_{(\xi_t)} = \prod_{i=2}^t 1/N_i$;

- Um antecessor do nó ξ_t é denotado por $a(\xi_t) \in \Xi_{t-1}$. Isto quer dizer que $x_t^{a(\xi_t)}$ denota a solução obtida x_{t-1} no nó do estágio anterior que está conectado por um arco com ξ_t .

Diante destas considerações, segue a formulação equivalente determinística:

$$\left\{ \begin{array}{l} \min \quad c_1^T x_1 + \sum_{\xi_2 \in \Xi_2} p(\xi_2) c_{\xi_2}^T x_{\xi_2} + \sum_{\xi_3 \in \Xi_3} p(\xi_3) c_{\xi_3}^T x_{\xi_3} + \cdots + \sum_{\xi_T \in \Xi_T} p(\xi_T) c_{\xi_T}^T x_{\xi_T} \\ \text{s.a} \quad A_1 x_1 = b_1 \\ \quad B_{\xi_2} x_1 + A_{\xi_2} x_{\xi_2} = b_{\xi_2}, \quad \forall \xi_2 \in \Xi_2 \\ \quad \quad B_{\xi_3} x_2^{a(\xi_3)} + A_{\xi_3} x_{\xi_3} = b_{\xi_3}, \quad \forall \xi_3 \in \Xi_3 \\ \quad \quad \quad \ddots \quad \quad \quad \ddots \quad \quad \quad \vdots \\ \quad \quad \quad B_{\xi_T} x_{T-1}^{a(\xi_T)} + A_{\xi_T} x_{\xi_T} = b_{\xi_T}, \quad \forall \xi_T \in \Xi_T. \end{array} \right. \quad (3.39)$$

Representar o problema (3.39) de modo que seja implementado computacionalmente e resolvido de forma direta pelo problema (2.26) é semelhante ao que foi feito nos casos anteriores, mas deve-se tomar alguns cuidados quanto ao cálculo das probabilidades envolvidas e à associação das sequências de nós de forma correta, o que influencia diretamente nas restrições do problema. A partir de agora definimos $M_t = \prod_{i=2}^t N_i$, por praticidade, e utilizamos a distinção de cada nó em seu respectivo estágio por $\xi_t^i = (c_t^i, A_t^i, b_t^i)$, com $i = 1, \dots, N_t$ e $t = 2, \dots, T$, isto é, $A_{\xi_t^i}$ representa a matriz A localizada no nó i do estágio t .

Para simplificar o tamanho dos vetores b e c , são definidos dois vetores b_t e c_t em blocos de modo que agrupe todas as realizações dentro de um mesmo estágio, onde $t = 2, \dots, T$:

$$c_t = \begin{bmatrix} c_{\xi_t^1} \\ \vdots \\ c_{\xi_t^{N_t}} \end{bmatrix} \quad \text{e} \quad b_t = \begin{bmatrix} b_{\xi_t^1} \\ \vdots \\ b_{\xi_t^{N_t}} \end{bmatrix}$$

Precisamos redefinir a variável de decisão x , os vetores b e c e a matriz A do problema (2.26) da seguinte maneira:

- Agrupar as variáveis x_1 e $x_{\xi_t^i}$ e os vetores $b_t^{\xi_t^i}$ e $c_t^{\xi_t^i}$ com $i = 1, \dots, N_t$ e $t = 2, \dots, T$:

$$x = \begin{bmatrix} x_1 \\ x_{\xi_2^1} \\ \vdots \\ x_{\xi_2^{M_2}} \\ x_{\xi_3^1} \\ \vdots \\ x_{\xi_3^{M_3}} \\ \vdots \\ x_{\xi_T^1} \\ \vdots \\ x_{\xi_T^{M_T}} \end{bmatrix}, \quad b = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_3 \\ \vdots \\ b_T \\ \vdots \\ b_T \end{bmatrix} \quad \left. \begin{array}{l} \left. \begin{array}{l} b_2 \\ b_3 \\ \vdots \\ b_3 \end{array} \right\} \text{ Repete} \\ \left. \begin{array}{l} b_3 \\ \vdots \\ b_T \\ \vdots \\ b_T \end{array} \right\} \text{ Repete} \end{array} \right\} \begin{array}{l} M_2 \text{ vezes} \\ M_{T-1} \text{ vezes} \end{array}$$

$$c = \begin{bmatrix} c_1 \\ p(\xi_2) \cdot c_2 \\ p(\xi_3) \cdot c_3 \\ \vdots \\ p(\xi_3) \cdot c_3 \\ \vdots \\ p(\xi_T) \cdot c_T \\ \vdots \\ p(\xi_T) \cdot c_T \end{bmatrix} \quad \left. \begin{array}{l} \left. \begin{array}{l} p(\xi_2) \cdot c_2 \\ p(\xi_3) \cdot c_3 \\ \vdots \\ p(\xi_3) \cdot c_3 \end{array} \right\} \text{ Repete} \\ \left. \begin{array}{l} p(\xi_T) \cdot c_T \\ \vdots \\ p(\xi_T) \cdot c_T \end{array} \right\} \text{ Repete} \end{array} \right\} \begin{array}{l} M_2 \text{ vezes} \\ M_{T-1} \text{ vezes} \end{array}$$

Observação: As probabilidades integradas no vetor c só se repetem dentro de um mesmo estágio por causa da consideração que foi feita sobre as realizações da variável aleatória em cada estágio serem

Capítulo 4

Resolução de problemas de despacho hidrotérmico pelo algoritmo PDDE

Este capítulo é dedicado à modelagem e resolução do problema de despacho hidrotérmico em duas configurações fictícias diferentes. A primeira, apresentada na Seção 4.1, é composta por um sistema de 3 usinas hidrelétricas e 3 termelétricas e foi desenvolvida juntamente com o Dr. Clóvis Caesar Gonzaga, professor aposentado da Universidade Federal de Santa Catarina e da Universidade Federal do Rio de Janeiro. Na Seção 4.2 apresentamos a segunda configuração composta por 7 usinas hidrelétricas e uma termelétrica, cujos dados foram obtidos de [33], os quais correspondem a 7 usinas do sul do país de um sistema teste de 33 barras. Apesar das modelagens representarem o problema com horizonte de planejamento a curto prazo estendido (1 ano), utilizou-se o método aplicado para o problema de planejamento a médio prazo (5 anos), o PDDE, descrito na Seção 3.4.3. Estes problemas possuem recurso relativamente completo, pois a demanda de energia sempre será atendida, visto que o déficit é considerado ilimitado. A partir de gráficos que ilustram o comportamento dos níveis dos reservatórios, das gerações de energia e dos custos, apresentamos os resultados obtidos pelo algoritmo junto com algumas análises.

4.1 Sistema com 3 hidrelétricas

4.1.1 Descrição do sistema

Iniciamos esta seção apresentando uma ilustração do sistema composto por 3 usinas hidrelétricas com reservatórios e 3 termelétricas, como segue na Figura 4.1, na qual as usinas hidrelétricas estão representadas pelas letras H_i , as termelétricas por T_i , as afluentes que chegam aos reservatórios por a_i e a água liberada por turbinamento ou vertimento das hidrelétricas por q_i e s_i , respectivamente. As setas indicam o fluxo da água no sistema, indicando se ela entra ou sai dos reservatórios das usinas hidrelétricas. Veja que a água que chega ao reservatório de H_2 já considera o fato das hidrelétricas H_1 e H_2 estarem em cascata, isto é, elas

estão num mesmo rio e H_1 está a montante¹ de H_2 , o que implica que H_2 recebe a água das vazões de H_1 e da afluência natural a_2 . Já a hidrelétrica H_3 está em paralelo com as outras duas.

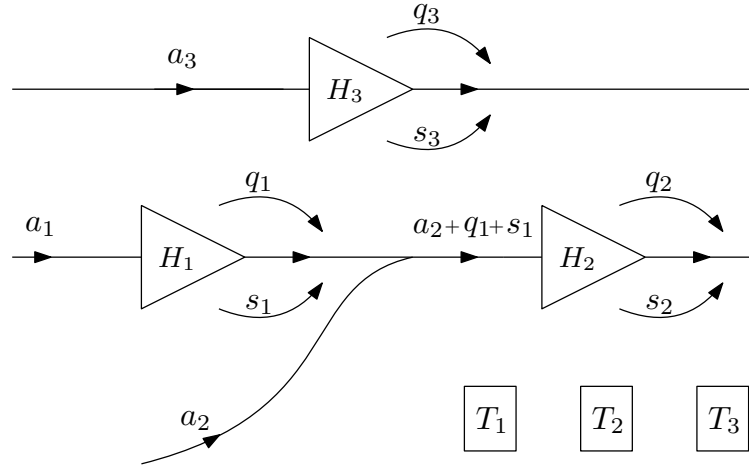


Figura 4.1: Sistema de 3 usinas hidrelétricas e 3 termelétricas.

4.1.2 Dados do sistema e do problema

Nesta seção apresentamos todos os dados referentes ao problema que estamos interessados em resolver e as características do sistema. A Tabela 4.1 exibe as nomenclaturas utilizadas na modelagem. Algumas vezes utilizamos simplesmente o termo "usina" para nos referir a uma usina hidrelétrica.

As próximas tabelas trazem informações referentes ao problema e ao sistema. A Tabela 4.2 contém os dados do problema, tais como horizonte de planejamento, número de estágios, a demanda energética do sistema, entre outros. Já as Tabelas 4.3 e 4.4 contém dados das usinas hidrelétricas e termelétricas que, por sua vez, foram utilizados valores pequenos e inteiros, os quais facilitarão na verificação de alguns resultados.

Além destas informações, como estamos tratando de um problema estocástico, as aflúências são consideradas como variável aleatória, onde fizemos o seguinte procedimento para criar um conjunto de sequências hidrológicas, isto é, os cenários para as 3 usinas ao longo dos 12 estágios:

- Determinamos uma matriz ref de referência para nos auxiliar na construção da árvore de cenários. Esta matriz possui dimensões 3×12 , onde a componente $ref_{i,j}$ da matriz representa uma afluência para a usina i no estágio j , a qual será utilizada como base para construção dos cenários. A matriz ref possui 12 colunas porque estamos modelando o problema em 12 estágios. Segue a matriz:

$$ref = \begin{bmatrix} 19,5 & 18,0 & 16,5 & 15,0 & 13,5 & 12,0 & 10,5 & 12,0 & 15,0 & 15,0 & 16,5 & 18,0 \\ 10,4 & 9,6 & 8,8 & 8,0 & 7,2 & 6,4 & 5,6 & 6,4 & 8,0 & 8,0 & 8,8 & 9,6 \\ 19,5 & 18,0 & 16,5 & 15,0 & 13,5 & 12,0 & 10,5 & 12,0 & 15,0 & 15,0 & 16,5 & 18,0 \end{bmatrix};$$

¹Termo utilizado para se referir a usinas que estão em pontos mais altos, ou na direção da nascente do rio.

Nomenclatura	Descrição
T	Número total de estágios
t	Índice associado ao estágio $t \in \{1, \dots, T\}$
v_t^i	Volume do reservatório i no final do estágio t
q_t^i	Vazão turbinada da usina i no estágio t em $\text{hm}^3/\text{mês}$
s_t^i	Vazão vertida da usina i no estágio t em $\text{hm}^3/\text{mês}$
d_t	Déficit de energia no estágio t em MWmês
a_t^i	Afluência no reservatório i estágio t em $\text{hm}^3/\text{mês}$
ρ_i	Produtividade da usina i em $\text{MWmês}/\text{hm}^3/\text{mês}$
gh_t^i	Geração da usina i no estágio t em MWmês
gt_t^i	Geração da usina térmica i no estágio t em MWmês
ge_t	Geração total das térmicas e do déficit no estágio t em MWmês
ct_i	Custo de produção associado à usina térmica i em $\text{R}\$/\text{MWmês}$
cd	Custo associado ao déficit de energia em $\text{R}\$/\text{mês}$
ce_t	Custo da geração ge_t associado ao estágio t em $\text{R}\$/\text{mês}$
df_t	Fator desconto no estágio t
D	Demanda de energia do sistema em MWmês
r_t	Variável associada ao valor esperado aproximado da função recurso
ε	Tolerância do critério de parada
\underline{z}	Limite inferior determinado por (3.32)
\tilde{z}	Estimativa do limite superior determinado por (3.33)

Tabela 4.1: Nomenclaturas

- Construímos uma matriz auxiliar denominada *aux* de dimensões 3×125 a partir de três cartesianos de um mesmo conjunto formado por valores que correspondem às probabilidades acumuladas de 5%, 27%, 50%, 72% e 95% de uma distribuição normal de média 1 e desvio padrão 0,3, os quais correspondem a valores igualmente espaçados no intervalo entre os números 0,05 e 0,95. Esta matriz será utilizada para a calcular as possíveis afluências de cada reservatório em cada estágio do problema, como mostram os próximos itens;

Horizonte de planejamento	1 ano
Número de estágios	12
Demanda do sistema	120 MWmês
Custo do déficit	R\$ 10,00/MWmês
Fator desconto	$e^{-0,01(t-1)}$

Tabela 4.2: Dados do problema

Usina térmica	Custo (R\$/MWmês)	Geração máxima (MWmês)
T1	1	10
T2	2	10
T3	4	10

Tabela 4.3: Dados das usinas térmicas

	H_1	H_2	H_3
Volume máximo (hm ³)	100	50	100
Volume inicial (hm ³)	50	25	50
Vazão turbinada máxima (hm ³ /mês)	60	80	60
Produtividade ρ_i (MWmês/hm ³ /mês)	1	1	1

Tabela 4.4: Dados da usina hidrelétrica

- A matriz aux é multiplicada por cada coluna da matriz ref , onde cada elemento da coluna multiplica a respectiva linha da matriz aux . Para cada coluna multiplicada obtemos uma matriz 3×125 , onde armazenamos os produtos numa matriz tridimensional de dimensões $3 \times 125 \times 12$;
- Por fim, sorteamos aleatoriamente 10 das 125 colunas de cada nível² da matriz tridimensional do item anterior, obtendo uma nova matriz denominada $aflu$ com dimensões $3 \times 10 \times 12$, onde, fixado um nível k , cada coluna representa uma realização da variável aleatória no estágio k , isto é, esta matriz contém 10 valores da variável aleatória para cada estágio. Um cenário deste problema corresponde a escolher uma única coluna de cada nível da matriz $aflu$, obtendo uma sequência de afluições para cada reservatório em cada estágio;
- Diante desse processo para obter valores de afluições para os reservatórios, resultamos numa árvore com 10^{11} cenários.

4.1.3 Modelagem do problema

A modelagem deste problema é muito próxima daquela elaborada no exemplo de despacho hidrotérmico determinístico resolvido na Seção 2.2. No entanto, há algumas diferenças por questão da quantidade de usinas e estágios.

Por questão do número de estágios, incluímos o fator desconto (df_t), o qual aparece multiplicando o custo das térmicas e do déficit. Este fator faz com que o custo diminua no decorrer dos estágios e, como consequência disso, a produção de energia hidrelétrica será priorizada, enquanto que a produção de energia termelétrica será deixada para os últimos estágios, quando possível. Esse fator desconto "é um indicador financeiro para trazer o custo de operação futuro a valor presente", [13]. Portanto, o custo c_t de um determinado estágio t é dado por

$$c_t = df_t \cdot \left[cd \cdot d_t + \sum_{i=1}^3 ct_i \cdot gt_t^i \right]. \quad (4.1)$$

Percebe-se que o custo das termelétricas e déficit podem ser escritos por uma função linear por partes, a qual corresponde ao máximo entre cada função linear que representa o custo das usinas térmicas e o déficit, pois ao utilizar a geração máxima de uma usina térmica, aciona-se a seguinte até atingir sua geração máxima. Essa sequência segue até o déficit completar o suprimento de energia do sistema, caso seja necessário. As funções lineares por partes se diferem minimamente para cada estágio por causa do fator desconto, que influencia nos coeficientes das funções lineares referentes aos custos das termelétricas e déficit.

Deste modo, podemos representar o custo do sistema em qualquer estágio sem utilizar uma variável para cada termelétrica e déficit. Para isto, agrupamos as informações referentes à geração de energia térmica e ao déficit pela variável ge_t e por ce_t a função linear por partes que representa o custo de cada unidade desta energia no estágio t , que é dada pela Figura 4.2.

²Denominamos por nível k da matriz tridimensional elementos da forma $a_{i,j,k}$, com k fixo.

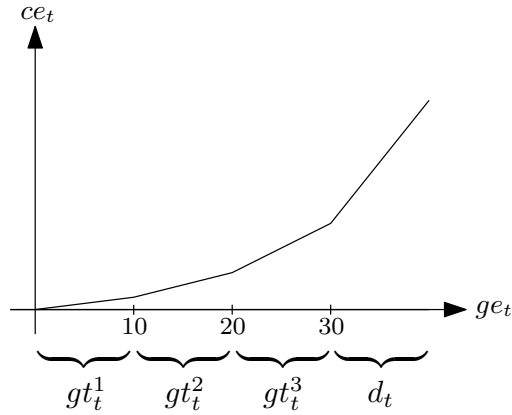


Figura 4.2: Função custo das termelétricas e déficit linear por partes.

Veja que deste modo estamos considerando que o déficit é uma usina térmica fictícia com um custo de geração elevado. Portanto, o custo c_t num estágio t é modelado do seguinte modo:

$$c_t = ce_t \cdot ge_t. \quad (4.2)$$

As últimas considerações a serem feitas são que o volume mínimo (v_{min}^i), as vazões vertida (s_{min}^i) e turbinada (q_{min}^i) mínimas de uma usina H_i , $i = 1, 2, 3$, serão consideradas nulas e a vazão vertida máxima (s_{max}^i) é ilimitada, isto é, $v_{min}^i = q_{min}^i = s_{min}^i = 0$ e $s_{max}^i = \infty$, para todos os estágios. Como foi remodelado a quantidade de energia gerada pelas termelétricas e o déficit, as limitações mínima e máxima da variável ge_t são 0 e ∞ , respectivamente. Além disto, a produção de energia da usina i num estágio t é dado pela variável gh_t^i , a qual corresponde a $gh_t^i = \rho_i \cdot q_t^i$, $i = 1, 2, 3$.

A partir de agora exibimos para cada estágio t o vetor de decisão do problema, a função objetivo e quatro tipos de restrições: de balanço hídrico, de atendimento à demanda, das limitações das usinas e da função custo:

- Vetor decisão:

$$x_t = [v_t^1, v_t^2, v_t^3, q_t^1, q_t^2, q_t^3, s_t^1, s_t^2, s_t^3, gh_t^1, gh_t^2, gh_t^3, ge_t, ce_t, r_{t+1}]^T \quad (4.3)$$

Lembramos que a componente r_t do vetor decisão é referente à aproximação do valor esperado da função recurso para cada estágio;

- Função objetivo: soma dos custos das usinas térmicas e do déficit de energia com o valor esperado do custo dos estágios futuros;

$$f_t(x_t) = ce_t \cdot ge_t + r_{t+1}, \quad (4.4)$$

onde $r_{T+1} = 0$.

- Balanço hídrico: o volume atualizado dos reservatórios das usinas H_1 e H_3 são dados pela diferença entre o volume inicial adicionado à afluência e a soma das vazões vertida e turbinada das usinas. Como

a usina H_2 está em cascata com H_1 , a expressão de seu volume é análoga ao das demais, adicionando-se as vazões turbinada e vertida da usina H_1 :

$$\begin{aligned} v_t^1 &= 50 + a_t^1 - (q_t^1 + s_t^1) \\ v_t^2 &= 25 + a_t^2 - (q_t^2 + s_t^2) + (q_t^1 + s_t^1) \\ v_t^3 &= 50 + a_t^3 - (q_t^3 + s_t^3); \end{aligned} \quad (4.5)$$

- Atendimento à demanda: a soma da geração de energia das usinas termelétricas e hidrelétricas junto com o déficit é igual à demanda:

$$\sum_{i=1}^3 gh_t^i + ge_t + d_t = D; \quad (4.6)$$

- Limitações das variáveis:

$$\begin{aligned} 0 &\leq v_t^i \leq v_{max}^i, \quad i = 1, 2, 3 \\ 0 &\leq q_t^i \leq q_{max}^i, \quad i = 1, 2, 3 \\ s_t^i &\geq 0, \quad i = 1, 2, 3 \\ 0 &\leq ge_t \leq \infty. \end{aligned} \quad (4.7)$$

- Custo: Desta forma que o problema foi modelado, é necessário ter um certo cuidado com as restrições que envolvem o custo, pois elas mudam a cada estágio por causa do fator desconto. Como queremos minimizar o custo ce_t e sabemos que ele é o máximo de funções lineares por partes determinadas pelas funções de custo de cada termelétrica e déficit, temos as seguintes restrições:

$$ce_t \geq m_t^i + n_t^i \cdot \left(D - \sum_{j=1}^3 gh_t^j - d_t \right), \quad i = 1, \dots, 4, \quad (4.8)$$

onde m_t^i e n_t^i são os coeficientes linear e angular, respectivamente, das funções custo de cada termelétrica e do déficit no estágio t . Lembre-se que os dados que mudam no decorrer dos estágios por causa do fator desconto são os coeficientes m_t^i e n_t^i .

Este problema foi resolvido pelo algoritmo PDDE. Desta vez não apresentamos as matrizes e vetores do problema, pois esse processo já foi realizado em outros exemplos. Exibimos apenas a configuração utilizada no algoritmo e gráficos que ilustram as principais características do algoritmo e de uma solução obtida a partir da simulação de um cenário, tais como os níveis do reservatório, as produções de energia, as vazões turbinadas das usinas e um comparativo entre o custo futuro esperado e o custo do cenário.

4.1.4 Resolução pelo algoritmo PDDE

Primeiramente relembremos que estamos considerando que o problema é independente por estágios por simplicidade, onde temos 10 aflúncias para cada estágio resultando em 10^{11} cenários, os quais assumimos equiprováveis. Da forma que a matriz de aflúncias foi construída, também há 10 possibilidades de aflúncias para o primeiro estágio, no entanto o consideramos determinístico, isto é, conhecemos as aflúncias que irão

chegar aos reservatórios. Deste modo, fixamos uma coluna qualquer do primeiro nível da matriz tridimensional $aflu$ para representar esta afluência.

Como foi descrito na Seção 3.4.3, a quantidade de cenários escolhidos para a fase progressiva do algoritmo PDDE é arbitrário e isto influencia diretamente no esforço computacional para executá-lo e na qualidade da solução. Em [32] são apresentados alguns testes computacionais comparando diferentes estratégias para a escolha da quantidade de cenários a serem tomados na fase progressiva do algoritmo. A partir destes testes, sugere-se que utilizar uma estratégia onde se eleva o número de cenários na fase progressiva durante o processo iterativo pode gerar resultados melhores do que fixar um número de cenários. Em outras palavras, isto se resume em utilizar um número pequeno de cenários durante as primeiras iterações, de forma a obter maiores valores para o limite inferior num tempo computacional menor, e aumentar o número de cenários nas iterações seguintes.

Considerou-se então a estratégia "FW_4" para os testes computacionais desta seção, sugerida em [32]. Nesta estratégia são considerados 25 cenários aleatórios nas 16 primeiras iterações e 200 cenários aleatórios nas 13 iterações seguintes na fase progressiva. No entanto, para que o algoritmo finalizasse num tempo computacional razoável, foi considerada uma redução de 80% no número de cenários, resultando um total de $\frac{80}{100} \times (25 \times 16 + 13 \times 200) = 600$ cortes, calculados na fase recursiva, que aproximam cada função Q_t , $t = 2, \dots, 12$, ao longo de 29 iterações. Deste modo, foram resolvidos um total de $[1 + 16 \times 5 \times (12 - 1)] + [1 + 13 \times 40 \times (12 - 1)] = 6.602$ subproblemas na fase progressiva e $[16 \times 5 \times 10 \times (12 - 1)] + [13 \times 40 \times 10 \times (12 - 1)] = 66.000$ subproblemas na fase recursiva.

O critério de parada adotado no algoritmo foi pelo número máximo de iterações $k_{max} = 29$. No entanto, não deixamos de verificar os outros critérios de parada, aqueles sugeridos em [11] e [16], que são determinados respectivamente pelas expressões (3.37) e (3.38), e o critério de estabilização do limitante inferior \underline{z} , no qual verificamos se o desvio padrão dos 5 últimos limitantes inferiores calculados pelo algoritmo é menor que uma tolerância $\varepsilon = 10^{-4}$.

O algoritmo PDDE foi implementado no Matlab versão 8.5.0.197613 (R2015a) e aplicado em um computador desktop Intel Core i7-7700k, CPU 4.2 GHz com 16GB RAM, Windows-10 Pro 64 bits.

A seguir, apresentamos a Figura 4.3 que contém os dois primeiros critérios de parada citados anteriormente, os quais foram verificados no decorrer das iterações:

A linha vermelha representa a estimativa do limitante superior \tilde{z} juntamente com o intervalo de confiança dado pela expressão (3.37). Lembre-se que os valores de \tilde{z} dependem diretamente dos cenários escolhidos na fase progressiva, pois eles são calculados como a média dos custos destes cenários. A linha azul representa o limitante inferior \underline{z} e a linha verde é a diferença entre o limite superior L_{sup} do intervalo de confiança e o limitante inferior, o qual descreve o critério sugerido por [16], onde a linha tracejada é a tolerância determinada para este critério. Veja que pelo critério de intervalo de confiança o algoritmo teria parado já na segunda iteração, pois a linha azul se encontra dentro do intervalo. O segundo critério, determinado pelas linhas verde e tracejada, foi satisfeito já na terceira iteração, pois o limite superior do intervalo de

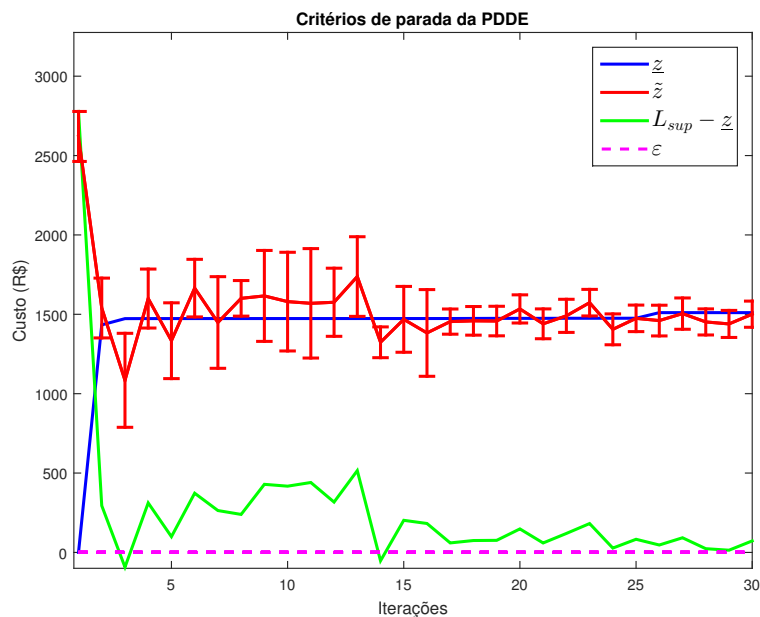


Figura 4.3: Critérios de parada sugeridos por [11] e [16].

confiança L_{sup} se encontra abaixo do limitante inferior do algoritmo z , sendo a diferença entre eles negativa. Além disto, a partir da terceira iteração o limitante inferior começa a apresentar variações muito pequenas e uma variação um pouco maior na iteração 26. Verifica-se na Figura 4.4 que, apesar do desvio padrão das 5 últimas iterações com a tolerância ε estarem coincidindo, aparentemente, o desvio padrão não ficou menor que a tolerância, sendo 0,0316 o menor valor atingido na última iteração.

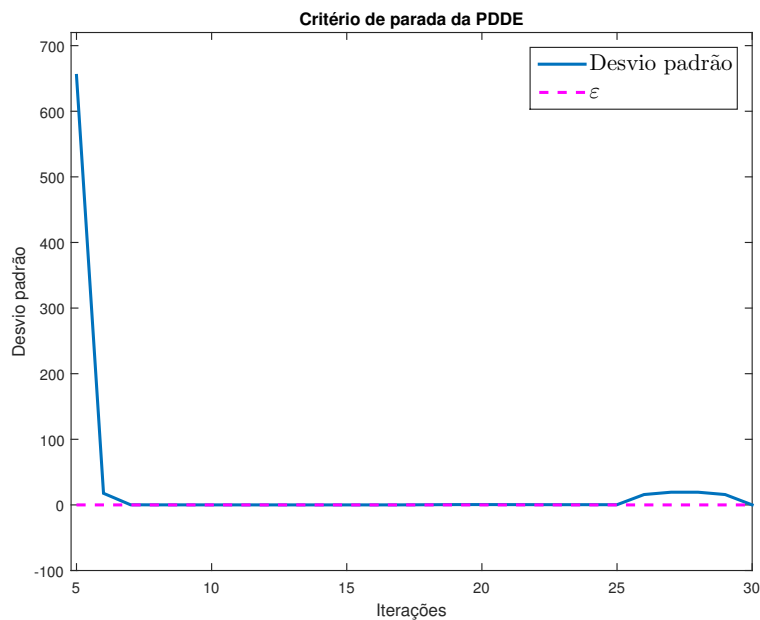


Figura 4.4: Desvio padrão das 5 últimas iterações.

Feitas as análises sobre os critérios de parada, verificaremos a média das soluções obtidas a partir da simulação de 1000 cenários escolhidos aleatoriamente, onde mantivemos a afluência do primeiro estágio. Para gerar estes cenários, foram escolhidas 10 colunas aleatoriamente de cada nível da matriz de dimensões $3 \times 125 \times 12$, já mencionada. A partir desta simulação podemos verificar se as aproximações do valor esperado das funções recurso, calculadas pelo algoritmo, estão relativamente próximas a simulação de um conjunto de cenários possivelmente distintos.

Ao fazer a simulação para este conjunto de cenários aleatórios, obtemos que, de acordo com a média das vazões turbinadas das usinas (Figura 4.5), a média dos níveis dos reservatórios, exibidos na Figura 4.6, se esvaziaram ao longo dos estágios. A geração de energia térmica, ilustrado na Figura 4.7, em média, se manteve no máximo até o sétimo estágio, de modo que o déficit fosse mínimo ao longo de todos os estágios. No entanto, mesmo gerando energia térmica ao máximo, ocorreu déficit nos estágios 8, 9 e 10. Isto se reflete nos baixos níveis de água dos três reservatórios a partir do sétimo estágio. Isso já era esperado por influência do fator desconto, pois prioriza-se a geração de energia das hidrelétricas. Nas Figuras 4.5 e 4.6 as linhas tracejadas indicam as limitações máximas das variáveis, onde a segunda figura começa no estágio $t = 0$, pois representa o nível inicial dos reservatórios.

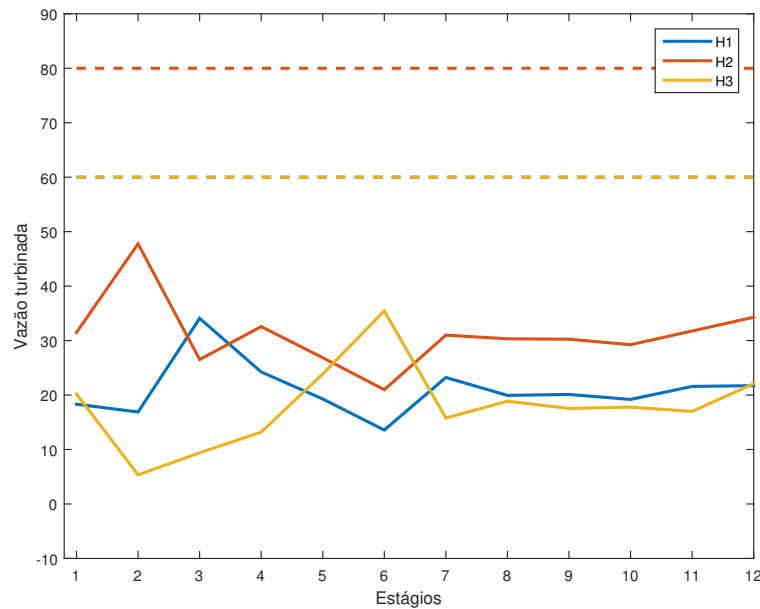


Figura 4.5: Vazões turbinadas das usinas.

Não exibimos o gráfico das vazões vertidas porque todas foram nulas em todos os estágios. Outro detalhe a ser visto para esta sequência hidrológica é o seu custo operacional, o qual foi denominado por "custo futuro". Na Figura 4.8 exibimos a comparação do custo gerado por esta simulação de 1000 cenários pelo valor esperado aproximado dos custos deste processo estocástico.

Quando comparamos os dois custos da Figura 4.8 percebemos que a aproximação do valor esperado da função recurso calculado pelo algoritmo foi próxima a média dos custos da simulação, sendo este um pouco

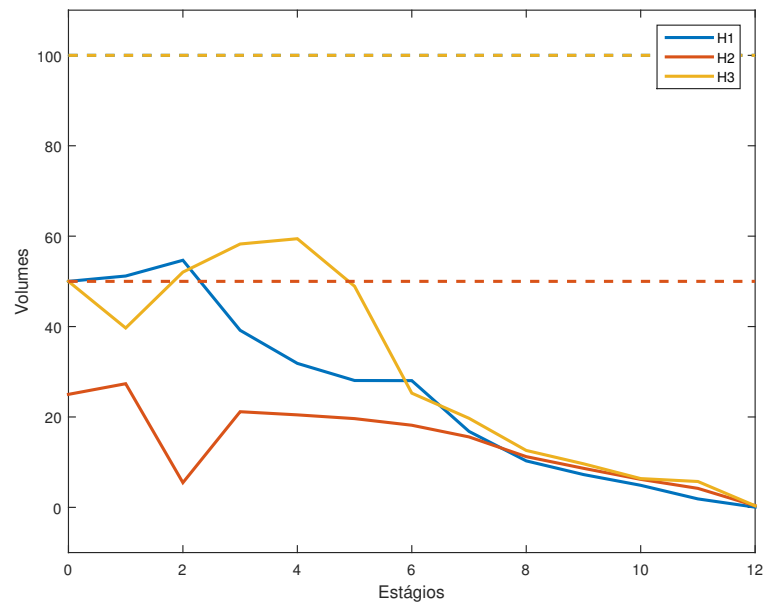


Figura 4.6: Volumes dos reservatórios.

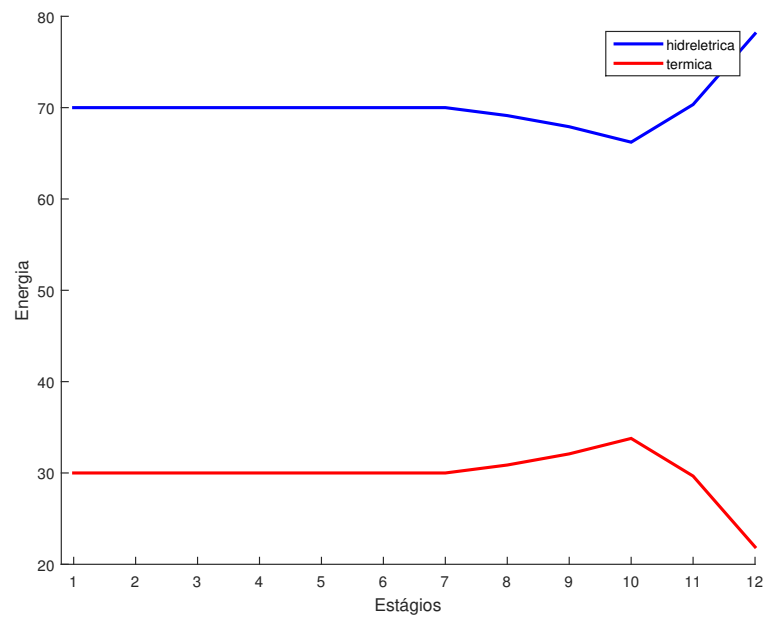


Figura 4.7: Gerações de energia hidrelétrica e termelétrica.

mais barato do que se previu. A Figura 4.9 justifica essa diferença, pois percebe-se que a média das afluências sorteadas para o algoritmo, linha vermelha, é inferior à média das afluências da simulação, linha azul.

Último detalhe a ser observado no algoritmo é o grande aumento de tempo no decorrer das iterações, o que pode ser visto na Figura 4.10, principalmente a partir da iteração 17. Isto se deve ao de fato que, neste exemplo, nas 16 primeiras iterações acrescentava-se 5 novas restrições nos subproblemas e a partir da iteração 17 acrescentava-se 40, devido a inclusão das aproximações.

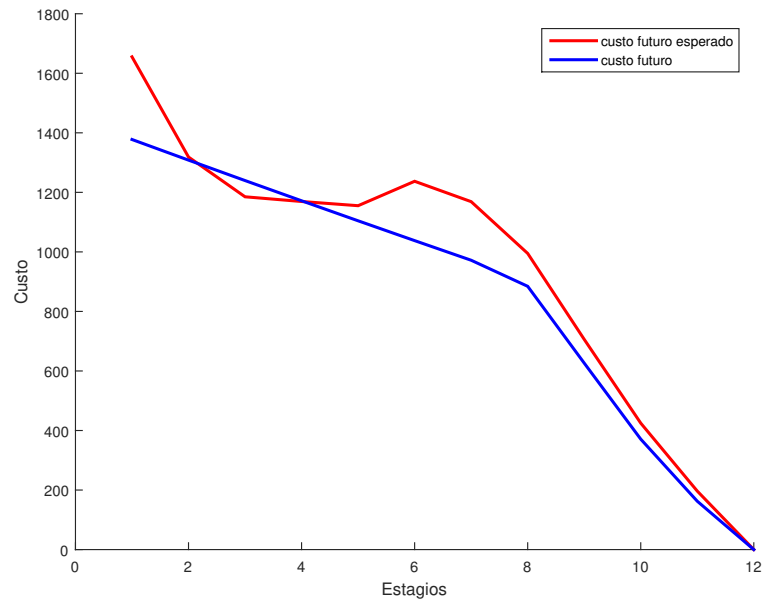


Figura 4.8: Custo operativo do cenário \times valor esperado dos custos.

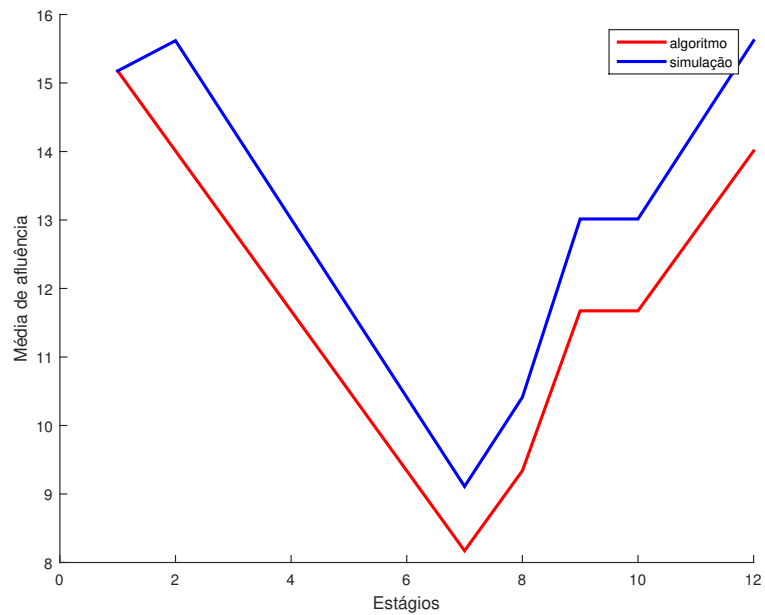


Figura 4.9: Média das afliências do algoritmo \times Média das afliências da simulação.

4.2 Sistema com 7 hidrelétricas

4.2.1 Descrição do sistema

Neste outro exemplo de despacho hidrotérmico, o sistema composto por 7 usinas hidrelétricas e uma termelétrica é dividido em dois subsistemas, o que difere em algumas características dos demais exemplos. Utilizamos esta composição em dois subsistemas com o intuito de aplicar o algoritmo num problema que

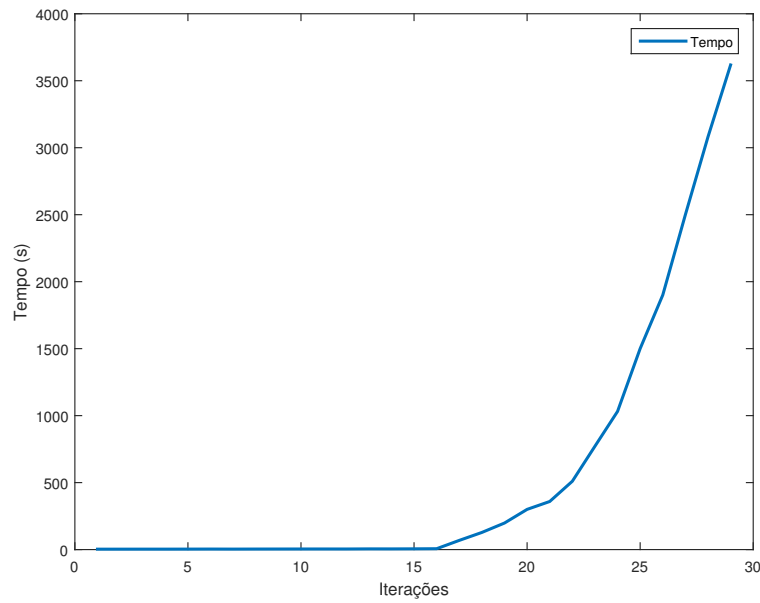


Figura 4.10: Tempo gasto nas iterações.

tenha características mais próximas do problema real de despacho hidrotérmico do Brasil, que é composto por 5 subsistemas. Primeiramente ilustramos na Figura 4.11 a topologia deste sistema.

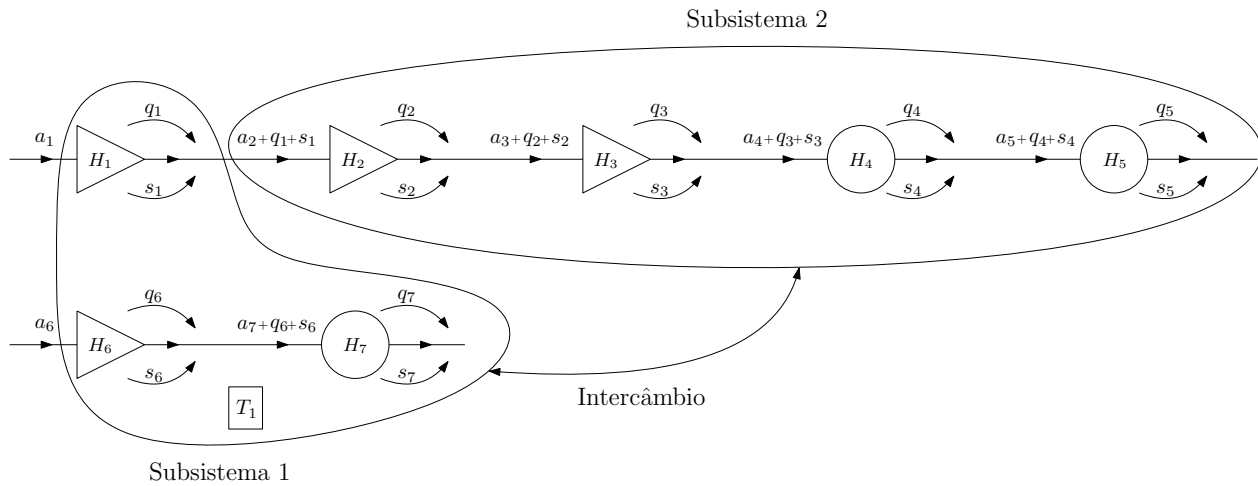


Figura 4.11: Sistema de 7 usinas hidrelétricas e uma termelétrica.

O primeiro subsistema é composto por 3 usinas, das quais duas são em cascata e outra é em paralelo, e uma termelétrica. Já o segundo é composto por 4 usinas em cascata. Note que na Figura 4.11 há duas representações para as usinas: os triângulos indicam as usinas com reservatórios e os círculos são as usinas que não possuem, as quais são denominadas usinas a fio d'água. Além disto, veja que existem afluições entre as usinas. Estas afluições representam a água vinda das chuvas. Além disto já representamos na figura o efeito cascata, onde a água que chega aos reservatórios tem origem da afluição natural e das vazões

turbinadas e vertidas da usina a montante, como foi descrito na seção anterior.

Cada subsistema é responsável por produzir energia elétrica para uma determinada região. Quando um deles não consegue suprir a demanda do mercado, utiliza-se o *intercâmbio* de energia entre os subsistemas. Isto significa que um subsistema irá transferir parte de sua energia produzida para o outro subsistema, cujo objetivo é evitar a ocorrência de déficit na região em que o subsistema é responsável.

4.2.2 Dados do sistema

As nomenclaturas utilizadas para esta modelagem são as mesmas da Tabela 4.1, a menos do déficit, que possui uma variável para os subsistemas 1 e 2 nos estágios t , sendo elas d_t^1 e d_t^2 , respectivamente, das demandas D_1 e D_2 , do intercâmbio INT_t , a qual representa o envio de energia do subsistema 1 para o subsistema 2, quando assume valores positivos, e o contrário, quando assume valores negativos, num determinado estágio t , e também o fator *produtividade de armazenamento*, determinado pelas constantes η_i para cada reservatório. Observamos que, em relação aos dados fornecidos, preferimos triplicar a demanda, pois, com os dados fornecidos, o custo esperado do problema era nulo. A seguir exibimos as características do problema e das usinas hidrelétricas e termelétricas nas Tabelas 4.5, 4.6 e 4.7.

Horizonte de planejamento	1 ano
Número de estágios	12
Demanda do Subsistema 1	8104,25 MWmês
Demanda do Subsistema 2	1684,37 MWmês
Custo do déficit no Subsistema 1	R\$ 1000,00/MWmês
Custo do déficit no Subsistema 2	R\$ 1000,00/MWmês
Fator desconto	$e^{-0,01(t-1)}$

Tabela 4.5: Dados do problema

Usina térmica	Custo (R\$/MWmês)	Geração máxima (MWmês)
T1	470,00	458,325

Tabela 4.6: Dados da usina térmica

Uma observação a ser feita referente à Tabela 4.7 é que, como as usinas H_4 , H_5 e H_7 são do tipo fio d'água, elas não possuem reservatórios e por isto não armazenam água. Isso justifica suas constantes produtividades de armazenamento η_i serem nulas.

As limitações de intercâmbio de energia, INT_{min} e INT_{max} , entre os subsistemas para todos os estágios estão na tabela a seguir:

	H_1	H_2	H_3	H_4	H_5	H_6	H_7
Volume mínimo (hm ³)	1974	2562	2662	1124	3573	2283	5100
Volume máximo (hm ³)	5779	2950	6775	1124	3573	3340	5100
Volume inicial (hm ³)	5779	2950	6775	1124	3573	3340	5100
Vazão turbinada máxima (hm ³ /mês)	1376	1268	1576	1784	2100	1311	1590
Produtividade ρ_i (MWmês/hm ³ /mês)	0,9357	0,6664	0,6456	0,939	1,21	1,227	1,084
Prod. de armazen. η_i (MWmês/hm ³ /mês)	0,1117	0,1057	0,06295	0	0	0,1521	0

Tabela 4.7: Dados das usinas hidrelétricas

Intercâmbio INT_{min} mínimo	-3693,6 MWmês
Intercâmbio INT_{max} máximo	3693,6 MWmês

Tabela 4.8: Limitações do intercâmbio.

4.2.3 Modelagem do problema

Uma das novidades desta modelagem é a inclusão o fator produtividade de armazenamento, que aparece na expressão de geração de energia das usinas hidrelétricas. Acrescentamos esta informação porque consideramos que o volume de água armazenado nos reservatórios também influencia na produção de energia, o que faz sentido, pois quanto maior forem os níveis dos reservatórios, maior será a energia potencial da água. Esta será transformada em energia cinética no canal de adução, movendo as turbinas, e, por fim, transformada em energia elétrica pelos geradores. Segue uma ilustração na Figura 4.12 que representa a estrutura de uma usina hidrelétrica com reservatório:

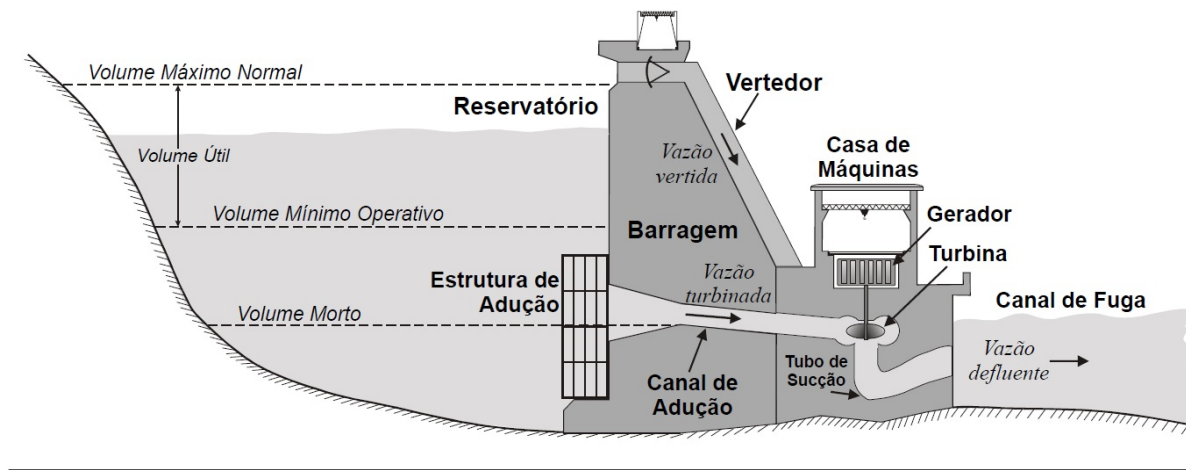


Figura 4.12: Estrutura de uma usina hidrelétrica com reservatório. Fonte: adaptado de [34]

Deste modo, a geração de energia gh_t^i da usina i num estágio t é dada da seguinte forma:

$$gh_t^i = \begin{cases} 0, & \text{se } q_t^i = 0 \\ \rho_i \cdot q_t^i + \eta_i \cdot v_t^i, & \text{se } q_t^i > 0. \end{cases} \quad (4.9)$$

Note que se a vazão turbinada q_t^i de uma usina hidrelétrica i num estágio t é nula, então não faz sentido haver geração de energia elétrica, obtendo $gh_t^i = 0$. A energia só é gerada a partir do turbinamento de água, o que ocorre quando $q_t^i > 0$, ou seja, $gh_t^i = \rho_i \cdot q_t^i + \eta_i \cdot v_t^i$.

Outra novidade foi a modelagem em dois subsistemas, o que nos permitiu a introdução do intercâmbio de energia entre eles. A variável INT_t faz parte do vetor de decisões e também da restrição de atendimento à demanda, pois podemos considerá-la como uma nova fonte de energia para suprir a demanda dos subsistemas.

Na exemplo da Seção 4.1 utilizamos o fato de que os custos das usinas termelétricas e do déficit poderiam ser representados por uma função linear por partes, a qual variava a cada estágio por causa do fator desconto. No entanto, outra forma de representá-lo é incluir o produto do fator desconto no vetor de custos, tomando o devido cuidado de não multiplicar a componente r_t do vetor decisão. A última característica a ser tratada, antes da formulação da função objetivo e das restrições do problema, é que as variáveis que limitam as vazões vertidas mínima e máxima das usinas são $s_{min}^i = 0$ e $s_{max}^i = \infty$, $i = 1, \dots, 7$.

- Vetor decisão:

$$x_t = [v_t^1, \dots, v_t^7, q_t^1, \dots, q_t^7, s_t^1, \dots, s_t^7, gh_t^1, \dots, gh_t^7, gt_t^1, d_t^1, d_t^2, INT_t, r_{t+1}]^T; \quad (4.10)$$

- Função objetivo: soma dos custos das usinas térmicas e do déficit de energia dos subsistemas.

$$f_t(x_t) = c_t^T x_t = df_t \cdot [gt_t^1 \cdot ct_1 + cd \cdot (d_t^1 + d_t^2)] + r_{t+1}, \quad (4.11)$$

onde $r_{T+1} = 0$ e

$$c_t = \underbrace{[0, 0, \dots, 0]}_{28 \text{ vezes}}, df_t \cdot ct_1, df_t \cdot cd, df_t \cdot cd, 0, 1];$$

- Balanço hídrico: o volume atualizado dos reservatórios das usinas H_1 e H_6 são dados pela diferença entre o volume inicial adicionado à afluência e a soma das vazões vertida e turbinada das usinas. Quanto ao reservatório da usina H_2, H_3, H_4, H_5 e H_7 , o volume atualizado tem a mesma expressão das outras, no entanto são adicionadas as vazões turbinada e vertida da usina H_1, H_2, H_3, H_4 e H_6 , respectivamente:

$$\begin{aligned} v_t^1 &= 5779 + a_t^1 - (q_t^1 + s_t^1) \\ v_t^2 &= 2950 + a_t^2 - (q_t^2 + s_t^2) + (q_t^1 + s_t^1) \\ v_t^3 &= 6775 + a_t^3 - (q_t^3 + s_t^3) + (q_t^2 + s_t^2) \\ v_t^4 &= 1124 + a_t^4 - (q_t^4 + s_t^4) + (q_t^3 + s_t^3) \\ v_t^5 &= 3573 + a_t^5 - (q_t^5 + s_t^5) + (q_t^4 + s_t^4) \\ v_t^6 &= 3340 + a_t^6 - (q_t^6 + s_t^6) \\ v_t^7 &= 5100 + a_t^7 - (q_t^7 + s_t^7) + (q_t^6 + s_t^6); \end{aligned} \quad (4.12)$$

- Atendimento à demanda: a soma da geração de energia das usinas termelétricas e hidrelétricas junto com o déficit e intercâmbio de energia entre os subsistemas é igual à demanda do subsistema:

$$\begin{aligned} \sum_{i \in \{1,6,7\}} gh_t^i + gt_1 + d_t^1 - INT_t &= D_1 \implies \text{subsistema 1} \\ \sum_{i=2}^5 gh_t^i + d_t^2 + INT_t &= D_2 \implies \text{subsistema 2,} \end{aligned} \quad (4.13)$$

onde gh_t^i é dado por (4.9).

- Limitações das variáveis:

$$\begin{aligned} v_{min}^i &\leq v_t^i \leq v_{max}^i, \quad i = 1, \dots, 7 \\ 0 &\leq q_t^i \leq q_{max}^i, \quad i = 1, \dots, 7 \\ s_t^i &\geq 0, \quad i = 1, \dots, 7 \\ 0 &\leq gt_1 \leq 458,325, \end{aligned} \quad (4.14)$$

onde v_{min}^i, v_{max}^i e q_{max}^i são dados pelas linhas 1, 2 e 4 da Tabela 4.7, respectivamente.

Desta vez consideramos que o volume inicial dos reservatórios é o próprio volume máximo, pois, ao considerar somente a metade do volume máximo, obtivemos problemas com o algoritmo por não encontrar solução para o problema no primeiro estágio por causa das usinas a fio d'água. Isto ocorreu porque a afluência que chegaria a pelo menos uma delas não seria suficiente para que o volume estivesse entre as limitações mínimas e máximas, as quais são iguais para este tipo de usina.

As afluências consideradas foram a série história do período de janeiro de 1931 até dezembro de 2013, resultando em 83 anos ou 996 meses, dos seguintes reservatórios: Foz do Areia (H_1), Segredo (H_2), Salto Santiago (H_3), Salto Osório (H_4), Salto Caxias (H_5), Machadinho (H_6) e Itá (H_7). Os dados referentes às afluências foram separados em dois grupos, um deles contendo os 54 primeiros anos para criar as aproximações do valor esperado das funções recurso e outro contendo os 29 anos restantes para realizar a simulação das aproximações obtidas. A partir destes dados construímos uma matriz tridimensional destas afluências com dimensões $7 \times 83 \times 12$. Desta matriz obtivemos 54 e 29 afluências para cada mês do ano para o algoritmo em si e a simulação, ou seja, o número total de cenários correspondentes a estes dados é de 54^{11} e 29^{11} , respectivamente. Realizadas todas as considerações a respeito da modelagem, prosseguimos com a resolução do problema.

4.2.4 Resolução pelo algoritmo PDDE

Para este problema utilizamos a mesma tolerância $\varepsilon = 10^{-4}$ para os critérios de parada determinados pelas expressões (3.37) e (3.38). O critério utilizado foi por número máximo de iterações $k_{max} = 29$, onde analisamos os outros critérios citados, assim como na seção anterior. Utilizamos também a mesma estratégia "FW_4" da seção anterior, no entanto reduzimos 84% do número de cenários da fase progressiva para que o algoritmo finalizasse num tempo razoável. Deste modo, foram calculados $\frac{84}{100} \times (25 \times 16 + 13 \times 200) = 480$

cortes na fase recursiva do algoritmo que aproximam cada função Q_t , $t = 2, \dots, 12$, ao longo de 29 iterações. Além disto, foram resolvidos um total de $[1 + 16 \times 4 \times (12 - 1)] + [1 + 13 \times 32 \times (12 - 1)] = 5.106$ subproblemas na fase progressiva e $[16 \times 4 \times 54 \times (12 - 1)] + [13 \times 32 \times 54 \times (12 - 1)] = 285.120$ subproblemas na fase recursiva.

Ao fim das iterações do algoritmo, verifica-se na Figura 4.13 que o algoritmo teria parado na terceira iteração pelo critério do intervalo de confiança. Quanto ao critério determinado por (3.38), o algoritmo haveria parado na iteração 9.

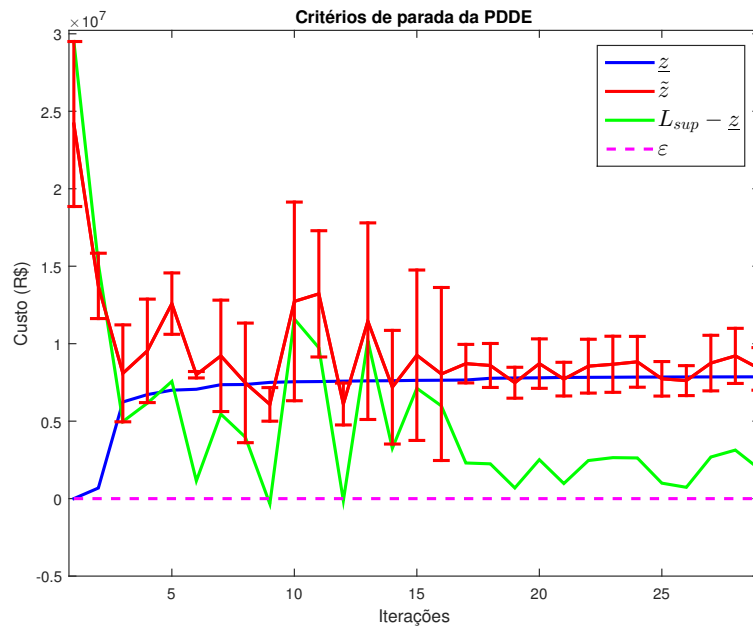


Figura 4.13: Critérios de parada sugeridos por [11] e [16].

Veja na Figura 4.14 que o limitante inferior \underline{z} apresenta um pequeno crescimento ao longo das iterações, porém está cada vez mais próximo de ser estável. No entanto, este critério não foi alcançado dentro das 29 iterações, sendo 4986,62 o desvio padrão do limitante inferior \underline{z} da iteração 25 até a última.

Como este problema possui um número maior de variáveis e restrições, quando comparado ao problema da Seção 4.1, já era de se esperar que o tempo computacional fosse muito superior, o que, de fato, ocorreu. O tempo gasto pelo algoritmo ao longo das iterações teve um crescimento muito significativo a partir da iteração 17, o que pode ser visto na Figura 4.15, onde o tempo total decorrido de sua execução foi de aproximadamente 12 horas.

Feitas as considerações gerais sobre o algoritmo aplicado no sistema de 7 usinas, fizemos uma simulação de 1000 cenários aleatórios. Os gráficos exibidos a seguir são referentes às médias de todas as soluções obtidas da simulação.

Analisando a média dos níveis dos reservatórios resultantes da simulação, verificamos que, assim como ocorreu no problema da Seção 4.1, os reservatórios atingiram, ou ficaram muito próximos, dos níveis mínimos

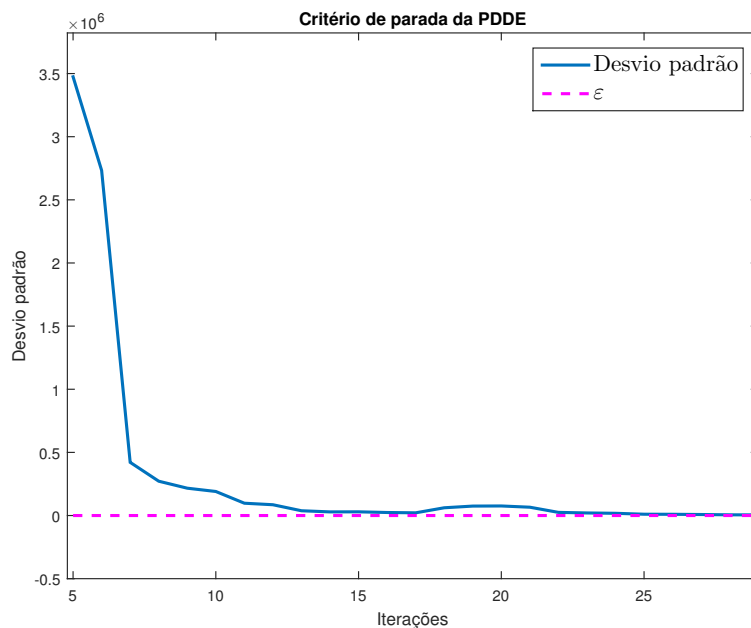


Figura 4.14: Desvio padrão das 5 últimas iterações.

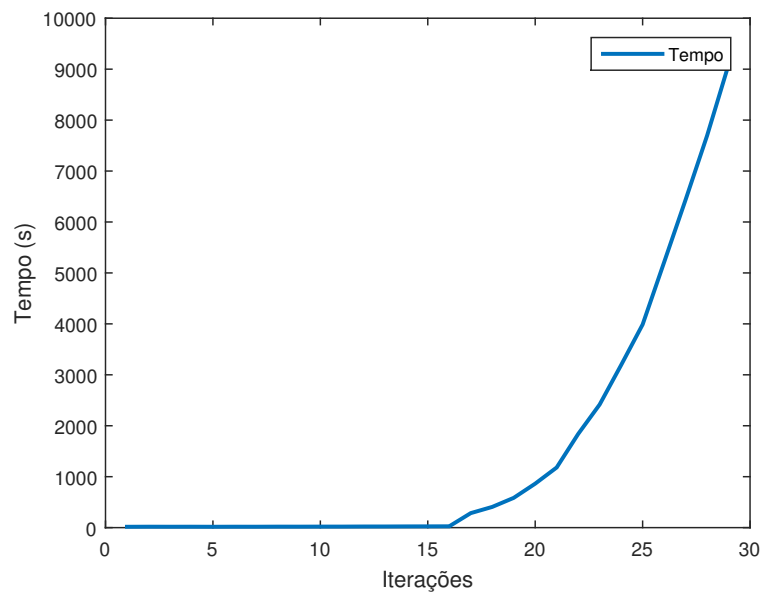


Figura 4.15: Tempo gasto nas iterações.

no último estágio, como pode ser visto na Figura 4.16.

Já nas Figuras 4.17 e 4.18 podemos visualizar a média dos volumes de água das vazões turbinadas e vertidas das usinas, respectivamente.

Para poder argumentar a respeito de todas estas informações apresentadas sobre as usinas, é necessário visualizar os gráficos referentes às médias de gerações de energia de cada subsistema, lembrando que a energia do subsistema 1 é gerada pelas usinas H_1, H_6 e H_7 e pela termelétrica T_1 e o subsistema 2 pelas

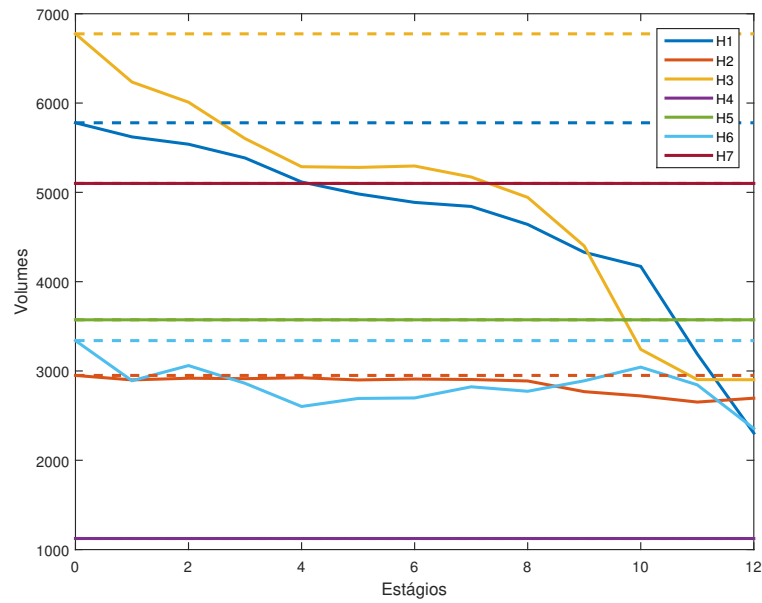


Figura 4.16: Volumes dos reservatórios.

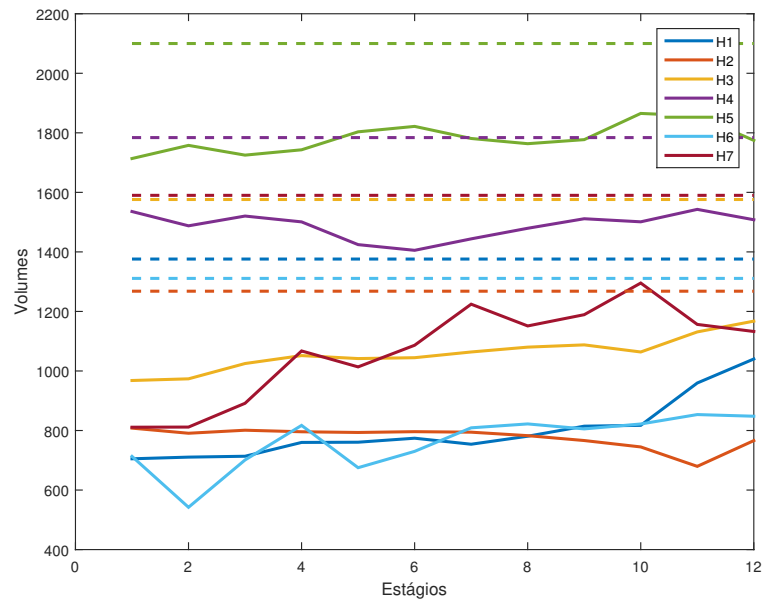


Figura 4.17: Vazões turbinada das usinas.

usinas H_2 , H_3 , H_4 e H_5 . Também faz parte do suprimento de energia dos subsistemas o intercâmbio. Estes elementos estão ilustrados nas Figuras 4.19, 4.20 e 4.21.

Primeiramente veja na Figura 4.20 que o subsistema 2, em média, sempre gerou uma quantidade de energia muito superior à demanda do subsistema, o que se reflete na constante transferência de energia para o subsistema 1, como mostram os valores negativos da Figura 4.21. Verifica-se que, em praticamente todos os estágios, o limite mínimo da variável foi quase atingido. Isso justifica as diversas ocorrências de vertimento

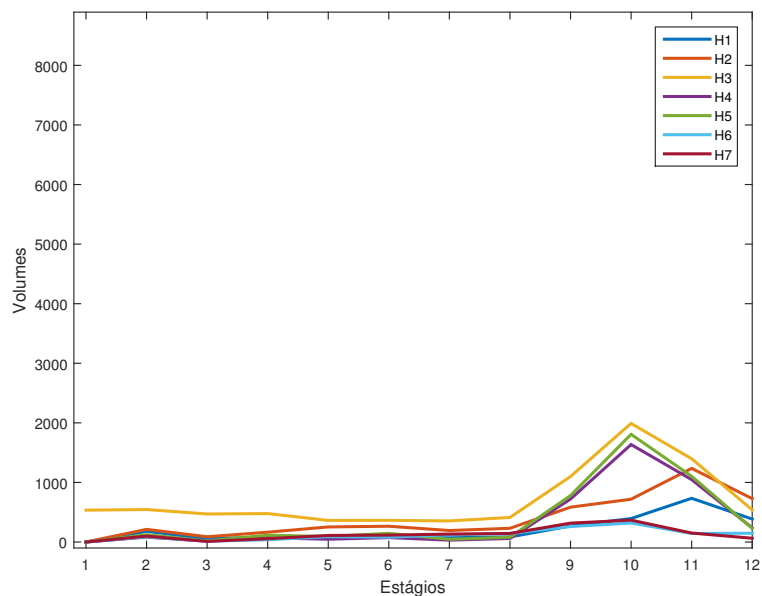


Figura 4.18: Vazões vertida das usinas.

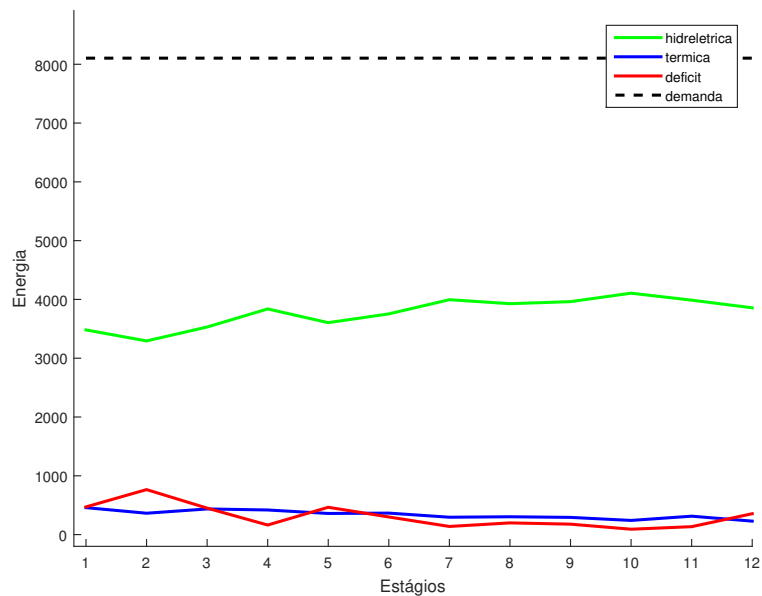


Figura 4.19: Geração de energia no subsistema 1.

por parte das usinas, principalmente do subsistema 2, como pode ser visto na Figura 4.18, pois, além de suprir a demanda do subsistema, elas geravam o máximo de energia possível para fazer intercâmbio com o subsistema 1 para minimizar o déficit ocorrido, e vertiam o restante.

Já no subsistema 1 verifica-se em muitos estágios o uso máximo da termelétrica T_1 para diminuir o déficit de energia. Cabe perguntar por que, por exemplo, as usinas H_1 e H_6 não turbinaram mais água nos primeiros estágios onde ocorreu o maior déficit de energia, sendo que seus reservatórios estavam num nível alto de água e que o turbinamento não estava em seu limite máximo? Note que não incluímos na pergunta a usina H_7 ,

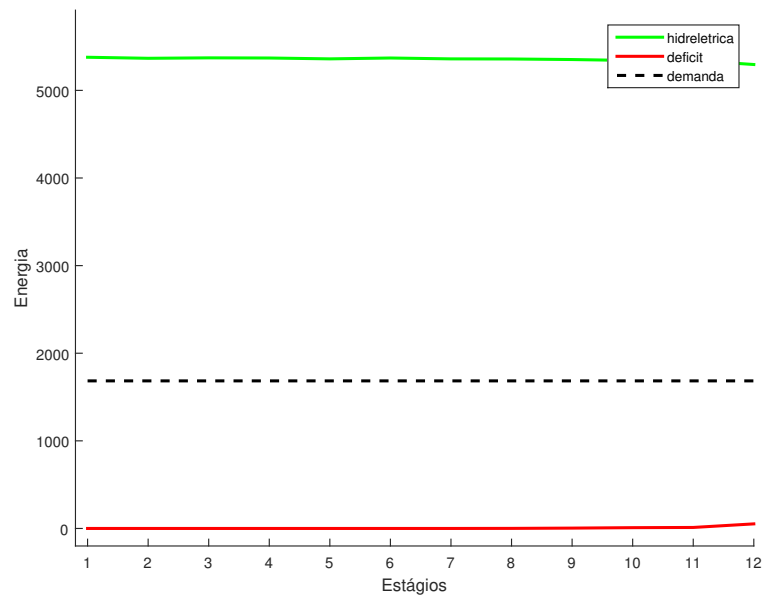


Figura 4.20: Geração de energia no subsistema 2.

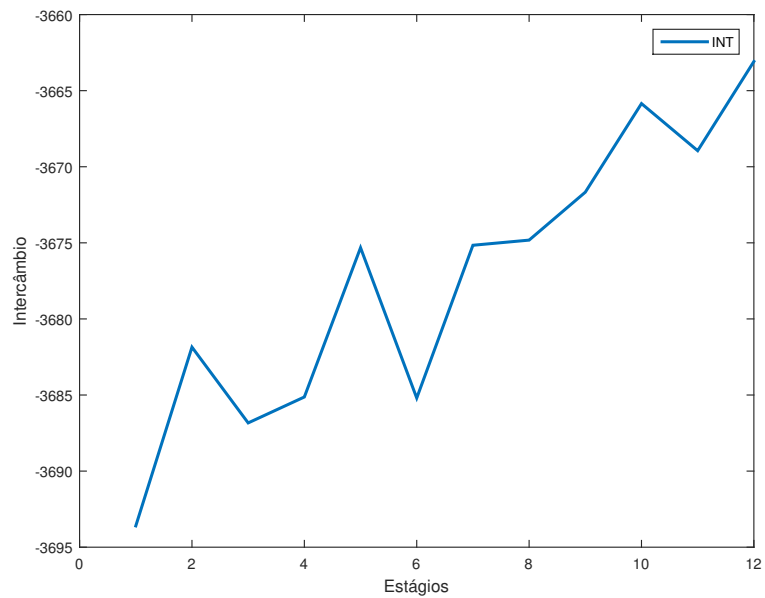


Figura 4.21: Intercâmbio de energia entre os subsistemas.

que também faz parte do subsistema 1, pois ela é a fio d'água, ou seja, ela turbinava exatamente a quantidade de água que chega nela.

A resposta desta pergunta é que se considerássemos este problema determinístico, isto é, se tivéssemos o conhecimento exato das afluições que iriam ocorrer no futuro, não precisaríamos das aproximações e, certamente, isto ocorreria. No entanto, lembre-se que estamos lidando com um problema estocástico, cuja solução obtida leva em consideração muitos cenários. Desta forma podemos dizer que as aproximações

calculadas pelo algoritmo indicavam que haveria um custo baixo nos primeiros estágios, isto é, períodos com volumes maiores de aflúências. Isto poderá ser visto com mais detalhes na sequência. Na Figura 4.22, podemos verificar a comparação das médias dos custos esperados e os da simulação, junto com o desvio padrão entre eles.

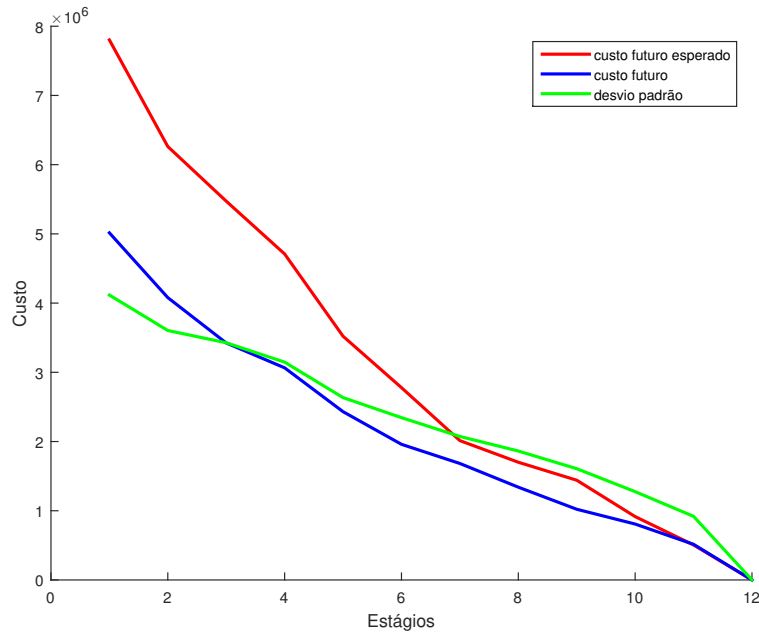


Figura 4.22: Custo operativo do cenário \times valor esperado dos custos.

Veja que, mesmo a média dos custos sendo mais elevada do que se esperava, o que pode ser visto pela linha em azul na Figura 4.22 quando comparada com a linha vermelha, que é o valor esperado dos custos futuros, a média dos custos da simulação tem semelhança com o custo esperado. A linha verde representa o desvio padrão dos custos dos cenários da simulação onde foi tomada a média dos custos futuros esperados como a média do desvio padrão para cada estágio. Verifica-se que o desvio padrão assume valores altos, o que pode ser justificado pelas diversas possibilidades de cenários com baixas e altas aflúências.

Podemos verificar pela Figura 4.22 que se esperava custos maiores em todos os estágios, o que pode ser explicado pela Figura 4.23, onde a linha em vermelho representa a média de aflúências da árvore de cenários do algoritmo e a linha em azul a média de aflúências da árvore de cenários da simulação, por estágio. Observa-se que a média de aflúências do algoritmo é inferior em praticamente todos os estágios em relação a média de aflúências da simulação e por isto as aproximações calculadas pelo algoritmo consideraram aflúências menores no futuro, prevendo assim custos futuros maiores.

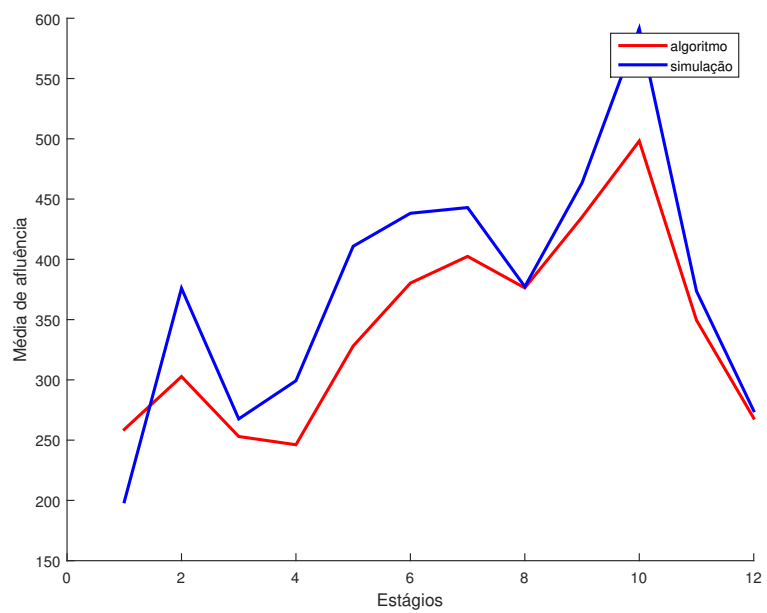


Figura 4.23: Média das afluências do algoritmo \times Média das afluências da simulação.

Capítulo 5

Conclusão

Neste trabalho foram apresentados alguns conceitos e formulações de problemas determinísticos e estocásticos que foram modelados em dois e multiestágios. A partir destas formulações, exibimos duas formas de resolver os problemas, sendo uma delas de forma direta e a outra por técnicas de decomposição, onde, nesta última forma, foram abordados a PDDD, no caso determinístico, e a decomposição aninhada e PDDE, no caso estocástico.

Sabemos que como o Brasil é um país que possui recursos hídricos em abundância e o uso destes recursos necessita um planejamento adequado, apresentamos o problema de despacho hidrotérmico e alguns exemplos, os quais foram descritos e modelados detalhadamente. Estes exemplos foram utilizados para ilustrar a aplicação das técnicas estudadas de programação determinística e estocástica. A partir destas aplicações, notamos que houve um crescimento de tempo muito elevado entre as iterações do algoritmo PDDE, que foi o principal a ser analisado, o que se dá pelo grande número de restrições que são adicionadas por causa da quantidade de cenários considerados na etapa de simulação.

Diante disto, para trabalhos futuros, vimos a necessidade de aprimorar o algoritmo PDDE, a fim melhorar sua eficiência, reduzindo seu tempo de execução; realizar modelagens mais realistas do problema de despacho hidrotérmico, o que pode deixá-lo de ser linear por causa da relação entre o volume do reservatório e a geração de energia hidrelétrica, o que motivaria a busca por outras metodologias de resolução; utilizar séries temporais ao invés das séries históricas de aflúncias; incluir na formulação do problema a estocasticidade na demanda; fazer um estudo de aversão ao risco.

Apêndice A

Conceitos básicos

Neste apêndice apresentamos algumas definições básicas de probabilidade e análise que foram úteis e importantes no desenvolvimento deste trabalho.

A.1 Probabilidade

Definição A.1.1. Uma σ -álgebra é uma coleção não-vazia \mathcal{F} de subconjuntos de Ω que satisfaz as seguintes propriedades:

- Se $A \in \mathcal{F}$, então $A^c \in \mathcal{F}$;
- Se $A_i \in \mathcal{F}$ é uma sequência enumerável de conjuntos, então $\bigcup_{i \in \mathbb{N}} A_i \in \mathcal{F}$.

Definição A.1.2. Um espaço de probabilidade é uma tripla (Ω, \mathcal{F}, P) , onde Ω representa o espaço amostral correspondente aos resultados de um experimento, \mathcal{F} é a σ -álgebra de subconjuntos (eventos) de Ω e $P : \mathcal{F} \rightarrow [0, 1]$ é uma função que atribui probabilidades aos eventos.

Definição A.1.3. Uma medida é uma função $\mu : \mathcal{F} \rightarrow \mathbb{R}$ que satisfaz as seguintes propriedades:

- $\mu(\emptyset) = 0$;
- $\mu(A) \geq 0, \forall A \in \mathcal{F}$;
- Se A_i é uma sequência enumerável de conjuntos disjuntos, então

$$\mu\left(\bigcup_i A_i\right) = \sum_i \mu(A_i).$$

Se $\mu(\Omega) = 1$ então μ é uma medida de probabilidade, que é usualmente denotada por P . Deste modo, $P : \mathcal{F} \rightarrow [0, 1]$ é uma medida de probabilidade.

A.2 Análise

Definição A.2.1. Uma função $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é dita *semicontínua inferiormente (sci)* num ponto $x_0 \in \mathbb{R}^n$ se $f(x_0) \leq \liminf_{x \rightarrow x_0} f(x)$. É dito que f é *semicontínua inferiormente* se é *semicontínua inferiormente* para todo ponto de \mathbb{R}^n .

Definição A.2.2. Uma função $f : X \subset \mathbb{R}^n \rightarrow \mathbb{R}$ *contínua* é *Lipschitz contínua* (ou *Lipschitziana*) em X se existe um número não negativo L tal que $|f(x) - f(y)| \leq L\|x - y\|$, para todo $x, y \in X$.

Definição A.2.3. Um conjunto $\mathcal{X} \subset \mathbb{R}^n$ é dito *convexo* quando dados $x, y \in \mathcal{X}$, o segmento $[x, y] = \{(1-t)x + ty \mid t \in [0, 1]\}$ estiver inteiramente contido em \mathcal{X} .

Definição A.2.4. Seja $\mathcal{X} \subset \mathbb{R}^n$ um conjunto convexo. Dizemos que a função $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é *convexa* em \mathcal{X} quando

$$f((1-t)x + ty) \leq (1-t)f(x) + tf(y),$$

para todos $x, y \in \mathcal{X}$ e $t \in [0, 1]$.

Definição A.2.5. Considere uma função real estendida $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$. É dito que o vetor $z \in \mathbb{R}^n$ é um *subgradiente* de f em x_0 se

$$f(x) \geq f(x_0) + z^T(x - x_0), \quad \forall x \in \mathbb{R}^n.$$

O conjunto de todos os subgradientes de f em x_0 é chamado de *subdiferencial* e é denotado por $\partial f(x_0)$

Definição A.2.6. Uma função real estendida $f : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{\infty\}$ é chamada *poliedral* se ela é própria, convexa e *semicontínua inferiormente*, seu domínio é um poliedro convexo, fechado e $f(\cdot)$ é linear por partes em seu domínio.

Definição A.2.7. O *epígrafo* de uma função $f : \mathbb{R}^n \rightarrow \mathbb{R}$ é dado por

$$\text{epi}f = \{(x, \alpha) \in \mathbb{R}^{n+1} : f(x) \leq \alpha\}.$$

Bibliografia

- [1] ANEEL, *Capacidade de geração do Brasil*, <http://www.aneel.com.br>, Acesso em: 01/02/2017.
- [2] ONS, *O SIN e os modelos NEWAVE e DECOMP utilizados no planejamento da operação energética e no cálculo do PLD*, <http://www.ons.org.br>, Acesso em: 03/03/2017.
- [3] M. R. Scuzziato, “Modelo de otimização estocástica de dois estágios para o problema de programação diária da operação eletroenergética.”, tese de doutorado, UFSC, 2016.
- [4] F. S. Thomé, “Representação de não-convexidade no planejamento da operação hidrotérmica utilizando PDDE”, tese de doutorado, COPPE - UFRJ, 2013.
- [5] W. A. Hall e N. Buras, “The dynamic programming approach to water-resources development”, *J Geophys Res*, vol. 66, n° 2, pp. 517–520, 1961.
- [6] N. Buras, “Dynamic programming in water resources development”, *Adv Hydrosci*, vol. 3, pp. 372–412, 1966.
- [7] A. Turgeon, “A decomposition method for the long-term scheduling of reservoirs in series”, *Water Resour Res*, vol. 17, n° 6, pp. 1565–1570, 1981.
- [8] A. Turgeon, “Optimal short-term hydro scheduling from the principle of progressive optimality”, *Water Resour Res*, vol. 17, n° 3, pp. 481–486, 1981.
- [9] J. F. Benders, “Partitioning procedures for solving mixed variables programming problems”, *Numer Math*, vol. 4, pp. 238–252, 1962.
- [10] M. V. F. Pereira, “Optimal stochastic operations scheduling of large hydroelectric systems”, *Int J Elec Power*, vol. 11, pp. 161–169, 1989.
- [11] M. V. F. Pereira e L. M. V. G. Pinto, “Multi-Stage stochastic optimization applied to energy planning”, *Math Program*, vol. 52, pp. 359–375, 1991.
- [12] D. C. Marcilio, “Otimização do despacho hidrotérmico utilizando o método de Lagrangeano aumentado com gradiente espectral projetado”, tese de doutorado, UFPR, 2014.
- [13] V. L. de Matos, “Um modelo para o planejamento anual da operação energética considerando técnicas avançadas de otimização estocástica”, tese de doutorado, UFSC, 2012.

- [14] V. Guigues e W. Römisch, “Sampling-based decomposition methods for multistage stochastic programs based on extended polyhedral risk measures”, *Siam J Optim*, vol. 12, pp. 286–312, 2010.
- [15] A. B. Philpott e V. L. de Matos, “Dynamic sampling algorithms for multi-stage stochastic programs with risk aversion”, *Eur J Oper Res*, 2011.
- [16] A. Shapiro, “Analysis of stochastic dual dynamic programming method”, *Eur J Oper Res*, vol. 209, pp. 63–72, 2011.
- [17] P. V. Larroyd, “Comparação de políticas com aversão a risco para o planejamento da operação hidrotérmica de médio prazo”, dissertação de mestrado, UFSC, 2012.
- [18] V. Guigues e C. Sagastizábal, “Exploiting structure of autoregressive processes in risk-averse multistage stochastic linear programs”, 2010.
- [19] A. Shapiro, “On a time consistency concept in risk averse multi-stage stochastic programming”, *Oper Res Lett*, vol. 37, pp. 143–147, 2009.
- [20] V. Guigues, “SDDP for some interstage dependent risk averse problems and application to hydrothermal planning”, *Comput Optim Appl*, 2011.
- [21] V. Guigues e C. Sagastizábal, “The value of rolling-horizon policies for risk-averse hydro-thermal planning”, *Eur J Oper Res*, vol. 217, pp. 129–140, 2012.
- [22] W. L. de Oliveira, “Métodos de feixes inexatos aplicados a programação estocástica”, tese de doutorado, COPPE - UFRJ, 2011.
- [23] E. C. Finardi, B. U. Decker e V. L. de Matos, “An introductory tutorial on stochastic programming using a long-term hydrothermal scheduling problem”, *J Control Autom Electr Syst*, vol. 24, pp. 361–376, 2013.
- [24] E. G. Birgin e J. M. Martínez, *Practical Augmented Lagrangian Methods for Constrained Optimization*. SIAM, 2014.
- [25] E. G. Birgin, J. M. Martínez e M. Raydan, “Algorithm SPG-software for convex constrained optimization”, *Trans. Math. Software*, vol. 27, pp. 340–349, 2001.
- [26] M. Kleina, “O método de pontos interiores aplicado ao problema do despacho hidrotérmico”, dissertação de mestrado, UFPR, 2012.
- [27] A. Shapiro, D. Dentcheva e A. Ruszczyński, *Lectures on Stochastic Programming: Modeling and Theory*. MPS Mathematical Programming Society, SIAM Society for Industrial e Applied Mathematics, 2009.
- [28] J. Nocedal e S. J. Wright, *Numerical Optimization*, 2nd. New York: Springer, 2006.
- [29] E. C. Finardi, “Planejamento da operação de sistemas hidrotérmicos utilizando computação de alto desempenho”, dissertação de mestrado, UFSC, 1999.

- [30] A. L. M. Marcato, “Representação híbrida de sistemas equivalentes e individualizados para o planejamento da operação de médio prazo de sistemas de potência de grande porte”, tese de doutorado, PUC-RJ, 2002.
- [31] J. F. Bonnans, J. C. Gilbert, C. Lemaréchal e C. A. Sagastizábal, *Numerical Optimization. Teorical and Pratical Aspects*. Springer, 2006.
- [32] A. Shapiro, W. Tekaya, J. P. da Costa e M. P. Soares, “Report for technical cooperation between Georgia Institute of Technology and ONS - Operador Nacional do Sistema Elétrico”, 2011.
- [33] W. F. Alves, “Proposição de sistemas-teste para análise computacional de sistemas de potência”, dissertação de mestrado, UFF, 2007.
- [34] D. da Silva Filho, “Dimensionamento de usinas hidroelétricas através de técnicas de otimização evolutiva”, tese de doutorado, USP, 2003.
- [35] A. A. Ribeiro e E. W. Karas, *Otimização Contínua: Aspectos Teóricos e Computacionais*. Cengage Learning, 2013.
- [36] K. G. Murty e F.-T. Yu, *Linear Complementarity, Linear and Nonlinear Programming Internet Edition*. University of Michigan - Ann Arbor, 2010.
- [37] H. J. Bortolossi e B. K. Pagnocelli, *Uma Introdução à Otimização sob Incerteza*. PUC-RJ: XI Simpósio de Pesquisa Operacional e Logística da Marinha, 2008.