

Métodos Eficientes para a Estimação de Desempenho em Sistemas de Comunicação Digital

Hallyson Luiz de Moraes Melo

Dissertação de Mestrado submetida à Coordenação dos Cursos de Pós-Graduação em Engenharia Elétrica da Universidade Federal da Paraíba - Campus II como parte dos requisitos necessários para obtenção do grau de Mestre no domínio da Engenharia Elétrica.

Área de Concentração: Processamento da Informação -
Comunicações

Prof. Francisco Marcos de Assis, Dr.
Orientador

Campina Grande, Paraíba, Brasil

©Hallyson Luiz de Moraes Melo, Novembro de 2002



M528m **Melo, Hallyson Luiz de Marais**
2002 **Métodos Eficientes para a Estimação de Desempenho em Sistemas de Comunicação Digital Hallyson Luiz de Morais Melo. - Campina Grande: UFPG, 2002.**
 94 p.: il.
 Inclui bibliografia
 Dissertação (mestrado) - UFPG/DEE
 Inclui bibliografia

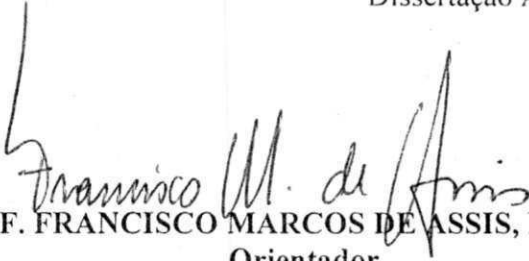
1. Sistemas de Comunicação Digital 2. Sistemas de Comunicação

CDU: 612.391

MÉTODOS EFICIENTES PARA A ESTIMAÇÃO DE DESEMPENHO EM
SISTEMAS DE COMUNICAÇÃO DIGITAL

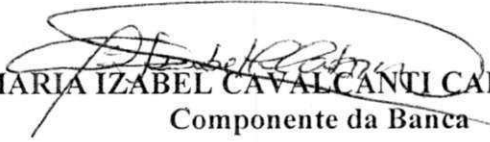
HALLYSON LUIZ DE MORAIS MELO

Dissertação Aprovada em 17.12.2002


PROF. FRANCISCO MARCOS DE ASSIS, Dr., UFCG
Orientador


PROF. BENEDITO GUIMARÃES AGUIAR NETO, Dr.-Ing., UFCG
Componente da Banca


PROF. JOSÉ EWERTON POMBO DE FARIAS, Dr., UFCG
Componente da Banca


PROF. MARIA IZABEL CAVALCANTI CABRAL, D.Sc., UFCG
Componente da Banca

CAMPINA GRANDE - PB
Dezembro - 2002

Dedicatória

Esta dissertação é dedicada a minha mãe Ruth Gonçalves de Moraes e a minha esposa Lídia Vasti Costa de Moraes Melo.

Agradecimentos

- A Deus, por tudo;
- A minha mãe Ruth Gonçalves, pelo infinito amor, incentivo, apoio e seu grande esforço para meu desenvolvimento pessoal e profissional;
- A minha esposa Lídia Vasti pelo seu amor, dedicação e compreensão em todos os momentos;
- Ao professor Francisco Marcos de Assis, pela orientação e ensinamentos decisivos para a direção desse trabalho;
- Ao professor Bruno Albert, pelo auxílio, idéias, amizade, incentivo e revisão do texto;
- Aos amigos Madhavan, Sérgio, Netto, Flávio, Christian, Josemar, Towar e Denis, pela amizade, companheirismo e momentos de descontração, e em especial ao amigo Luiz Felipe, pelo apoio e incentivo em todos os momentos desse trabalho;
- Ao amigo de infância David Wesley, pelo apoio, incentivo e amizade;
- Aos professores João Marques e José Ewerton Farias, pelos ensinamentos e amizade, e em especial ao professor e amigo Marcelo Sampaio, por ter sido um incentivador em minha carreira profissional;
- Aos demais professores do DEE-UFCG;
- A todos os funcionários do DEE-UFCG, em especial a Ângela pelo apoio;
- A todos os colegas do Genius Instituto de Tecnologia, pelos incentivos diários, idéias e momentos de descontração;
- Ao CNPq, pelo importante apoio financeiro.

Resumo

Desde a década de 70 o método de Monte Carlo tem sido a técnica predominante para a simulação de sistemas de comunicações. Entretanto, nos últimos anos uma técnica de redução de variância conhecida como valorização das amostras (*Importance Sampling*) vem demonstrando ter melhor desempenho possibilitando simulações mais curtas com a mesma precisão.

A modelagem para implementar a valorização das amostras exige um amplo conhecimento dos processos estocásticos envolvidos e a escolha de parâmetros da simulação que têm forte impacto no desempenho do método.

Nesse trabalho o método de valorização das amostras é aplicado a sistemas de comunicações codificados, utilizando o método gradiente descendente estocástico para a escolha ótima dos parâmetros da simulação. O ganho de desempenho obtido na simulação é demonstrado para um sistema com código de Hamming em canal AWGN. A redução do custo de processamento encontrada para esse sistema foi de ordem 10 quando comparada à simulação original com IS.

Abstract

Monte Carlo Trials is the main method used since the 70's to perform communication systems simulation. Nowadays, a variance-reducing method has been used to improve the simulation performance. This method is called Importance Sampling (IS) and it allows to perform faster simulations with same accuracy.

In order to use IS, it is necessary to know in detail the underlying stochastic process and to choose specific simulation parameters, which have a strong performance impact. In this work, IS is applied to encoded communication systems, using the Stochastic Gradient Descent (SGD) method to optimize the parameters search.

The performance gain sped up the simulation process ten times when compared to the original simulation with IS. Experiments were performed with Hamming encoded system over additive white Gaussian noise (AWGN) channels.

Lista de Símbolos e Abreviaturas

- AWGN* - Additive White Gaussian Noise (*Ruído Aditivo Gaussiano Branco*)
BER - Bit Error Rate (*Taxa de erros de bit*)
BPSK - Binary Phase Shift Keying (*Modulação em Fase Binária*)
CIS - Convencional Importance Sampling (*Estimador com valorização convencional das amostras*)
 d_{free} - Distância livre do código
 N_0 - Densidade espectral de potência do ruído gaussiano
 E_b - Energia de bit
 E_s - Energia do sinal
 $E_X[X]$ - Esperança (*Valor esperado da variável aleatória X*)
 f_{dp} - Função densidade de probabilidade
 $Q(\cdot)$ - Função de erro
IS - Importance Sampling (*Estimador com valorização das amostras*)
IIS - Improved Importance Sampling (*Estimador com polarização da média*)
MC - Monte Carlo (*Estimador de Monte Carlo*)
 p_b - Probabilidade de erro de bit
 p_c - Probabilidade de erro de decodificação
 p_i - Probabilidade de erro de decodificação específica
 p_s - Probabilidade de erro de símbolo
 E_b/N_0 - Relação sinal-ruído de bit
 R_b - Taxa de bit
GDE - Gradiente Descendente Estocástico (*Stochastic Gradient Descent*)
 T_s - Tempo de duração de um símbolo
 σ^2 - Variância do ruído gaussiano
 $w(\cdot)$ - Função de polarização

Lista de Figuras

1.1	Modelo para um sistema de comunicação digital.	3
1.2	Modelo para análise de desempenho de um sistema de comunicações.	10
1.3	Composição da fdp de v_k	11
2.1	Comparação entre estimadores com variâncias diferentes.	21
2.2	Intervalo de confiança para o estimador.	24
2.3	Cálculo de z_α para o intervalo de confiança.	25
3.1	Intervalo de confiança quando o valor observado é 10^{-k}	34
3.2	Representação geométrica dos sinais transmitidos no sistema BPSK.	36
3.3	Distribuição de probabilidade do sinal na entrada do receptor BPSK.	38
3.4	Simulação de Monte Carlo para o sistema BPSK sobre um canal AWGN.	39
4.1	Diagrama de blocos do Sistema simplificado.	44
4.2	Ilustração das amostras de $X(t)$	44
4.3	Polarização no método de valorização convencional das amostras (CIS).	50
4.4	Polarização no método de valorização melhorada das amostras (IIS).	51
4.5	Simulação com valorização das amostras para o sistema BPSK sobre um canal AWGN.	54
6.1	Modelo discreto para o sistema de comunicações com codificação.	66
7.1	Curva do gradiente na simulação adaptativa.	85
7.2	Convergência da média na simulação adaptativa.	86
7.3	Oscilação do gradiente em torno da origem.	87
7.4	Convergência da média polarizada a partir do método IIS.	88

Lista de Tabelas

2.1	Alguns valores de $z_{1-\alpha/2}$ para calcular o intervalo de confiança.	26
3.1	Resultados da simulação de Monte Carlo para o sistema BPSK.	40
4.1	Resultados da simulação com valorização das amostras para o sistema BPSK.	53
7.1	Alfabeto do codificador de Hamming (7,4).	81
7.2	Porcentagem de palavras códigos ignoradas na simulação.	86
7.3	Resultados da simulação do sistema com código de Hamming (7,4). . .	86
7.4	Parâmetros utilizados nas simulações.	87

Conteúdo

1	Introdução	1
1.1	Modelo do Sistema de Comunicação Digital	3
1.2	Medidas de Desempenho em Sistemas de Comunicações	5
1.3	Cálculo de Desempenho em Sistemas de Comunicações	7
1.4	Definição Geral do Problema	9
1.5	Organização do Trabalho	13
2	Estimação	15
2.1	Definição do Problema de Estimação	15
2.2	Método dos Momentos	16
2.3	Despolarização	18
2.4	Eficiência	21
2.5	Erro Normalizado	22
2.6	Consistência	22
2.7	Intervalo de Confiança	23
3	Método de Monte Carlo	27
3.1	Definição do Método de Monte Carlo	28
3.2	Despolarização e Consistência	29
3.3	Intervalo de Confiança	32
3.4	Simulação de um Sistema de Comunicações com Modulação Binária	35
3.5	Conclusões	40
4	Monte Carlo com Valorização das Amostras	42
4.1	Definição do Estimador IS	44
4.2	Despolarização e Consistência	46

4.3	Escolha da distribuição Polarizada	49
4.3.1	Polarização da Distribuição Gaussiana	49
4.4	Simulação de um Sistema de Comunicações com Modulação Binária . .	52
4.5	Conclusões	53
5	Otimização de Parâmetros em IS	55
5.1	Formulação do Problema	56
5.2	Métodos de Otimização	57
5.3	Valorização Adaptativa das amostras	58
5.3.1	Métodos de Otimização sem Derivadas	58
5.3.2	Gradiente Descendente Estocástico (GDE)	60
5.3.3	Método Estocástico de Newton	63
5.4	Conclusões	64
6	IS Aplicada a Sistemas Codificados	66
6.1	Estimação de Desempenho em sistemas com codificação	67
6.2	Valorização das Amostras para Simulação com Códigos de Bloco	68
7	Otimização de Parâmetros em IS para Sistemas com Codificação	74
7.1	Formulação do Problema	75
7.2	Desempenho do Código de Hamming (7, 4)	79
7.2.1	Estrutura do Código	79
7.2.2	Canal AWGN	82
7.3	Conclusões	88
8	Conclusões	89
8.1	Perspectivas para trabalhos futuros	90

Capítulo 1

Introdução

O surgimento de tecnologias que a cada dia possibilitam novos serviços tornou as telecomunicações um dos campos que mais evoluiu nos últimos anos e que vem atraindo um grande número de pesquisadores em seus diversos ramos de atuação.

Com a existência de um mercado tão promissor, empresas por todo mundo investem em tecnologias inovadoras que possam modificar o dia-a-dia de seus usuários. São sistemas móveis de terceira geração (CDMA2000, DECT, UTRA) [37], HDTV (*High Definition Television*), PLC (*Power Line Communication*), WLAN (*Wireless Local Area Network*), redes de alta velocidade, entre outros, proporcionando novos serviços e links cada vez mais rápidos e seguros.

Em uma área tão dinâmica novas oportunidades estão constantemente surgindo, e incorporar novas tecnologias rapidamente e eficientemente é um requisito fundamental para novos projetos.

A validação de uma nova tecnologia requer o uso de medidas de desempenho que possam garantir as melhorias estabelecidas durante o projeto. Especificamente em sistemas digitais a medida amplamente utilizada é a probabilidade de erro de bit (BER - *Bit Error Rate*), que é uma medida estatística da confiabilidade do sistema.

Para obter o valor da BER pode-se usar prototipagem, cálculos analíticos ou simulação. A prototipagem demanda custo e tempo de projeto, já a utilização de cálculos analíticos é limitada pela complexidade dos sistemas e do canal de comunicação envolvido. Por fim, simulação é uma forma mais genérica para estimar o valor da BER e tem sido amplamente utilizada na prática.

O método de simulação tradicional em sistemas de comunicações é conhecido como Método de Monte Carlo. Esse método visa reproduzir o sistema completamente uti-

lizando modelos discretos no tempo, tendo como característica principal a generalidade. O método de Monte Carlo torna-se impraticável em sistemas com probabilidade de erro muito baixa, pois o tempo de processamento necessário para obter valores estatisticamente significativos torna-se intolerável mesmo utilizando computadores de última geração. Por exemplo, levantar a curva da probabilidade de erro em sistemas com codificação turbo pode demandar até uma semana de processamento. Uma solução para esse problema é o uso de técnicas de redução de variância que são métodos mais eficientes (rápidos) para estimar a BER. Entre essas técnicas o método de valorização das amostras (IS - *Importance Sampling*) é o mais promissor, com o qual foi obtida uma redução de processamento da ordem de 10^8 [5].

Embora o ganho de desempenho com a valorização das amostras tenha sido utilizado em vários sistemas simulados, a sua aplicação não é direta, é necessário analisar o modelo do sistema e conhecer os processos aleatórios envolvidos [9]. O método baseia-se em modificar as densidades de probabilidade desses processos de uma forma conveniente para a simulação. Essa modificação é conhecida como polarização.

Mesmo após a escolha da estratégia de polarização, é necessário encontrar parâmetros que otimizem o ganho de processamento obtido pela valorização das amostras. Em muitos casos, esses parâmetros são encontrados utilizando o método da “força bruta”, ou seja, realizam-se várias simulações com parâmetros distintos e o conjunto de valores com melhor desempenho é selecionado.

Alguns métodos de otimização de parâmetros para a valorização das amostras têm sido propostos e a convergência desses métodos tem sido demonstrada [12] [13]. Entretanto, até então esses métodos foram utilizados para sistemas simples e sem memória. Nesse trabalho, um método de otimização de parâmetros em *Importance Sampling* é utilizado para auxiliar a aplicação de valorização das amostras em sistemas com codificação.

A Seção 1.1 contém alguns conceitos importantes na modelagem de sistemas digitais. As Seções 1.2 e 1.3 discutem as medidas de desempenho em sistemas de comunicações e os principais métodos para calculá-las. Na Seção 1.4 é introduzido o problema do cálculo da probabilidade de erro de bit (BER). Por fim, a Seção 1.5 descreve a organização da dissertação.

1.1 Modelo do Sistema de Comunicação Digital

Nesta seção será estabelecido um modelo para um sistema de comunicação digital. A descrição a seguir tem como objetivo introduzir conceitos utilizados no decorrer do presente trabalho.

Na Figura 1.1 é mostrado o diagrama de blocos do modelo de um sistema de comunicação digital. A informação a ser transmitida é gerada pela fonte, que emite um bit a cada T_b segundos. O modulador em seguida segmenta a seqüência de bits em blocos de k bits. Cada bloco é mapeado em um único símbolo, de modo que um símbolo é emitido a cada $T_s = k \cdot T_b$ segundos.

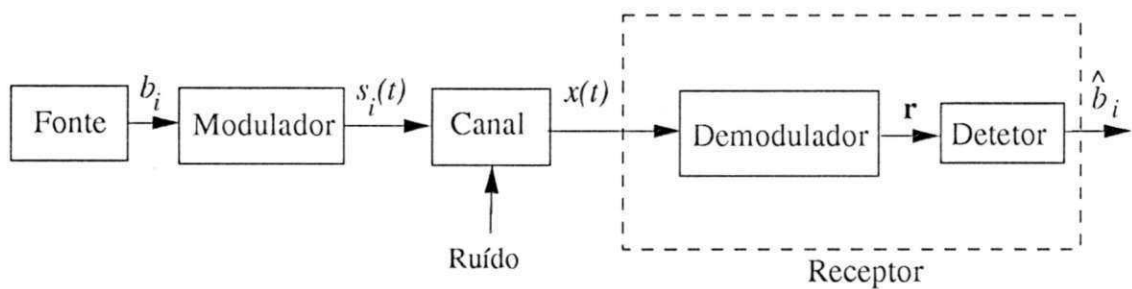


Figura 1.1: Modelo para um sistema de comunicação digital.

Os símbolos emitidos são escolhidos a partir de um alfabeto que contém M símbolos distintos, denominados por s_1, s_2, \dots, s_M . Todos os símbolos são considerados equiprováveis, ou seja, a cada intervalo T_s todos os símbolos do alfabeto têm a mesma probabilidade de serem transmitidos. Escrevendo formalmente, a probabilidade a priori p_i do símbolo s_i ser transmitido é dada por:

$$p_i = \frac{1}{M}, \quad i = 1, 2, \dots, M.$$

De modo geral, para um símbolo conter a informação de k bits, é necessário que o tamanho do alfabeto seja $M = 2^k$ símbolos.

Para cada símbolo s_i do alfabeto, o modulador associa uma forma de onda $s_i(t)$ a ser transmitida através do canal. As formas de onda possíveis na saída estão relacionadas ao método de modulação que pode ser por meio de pulsos ou uma forma de onda. Em ambos os casos as formas de onda transmitidas são modificações na amplitude, frequência, ou fase de uma portadora, ou alguma combinação dessas modificações.

Os formatos básicos de modulação digital são *amplitude shift keying* (ASK), *frequency shift keying* (FSK), e *phase shift keying* (PSK) [28]. Em alguns sistemas a combinação da modulação em fase e em amplitude apresenta algumas vantagens. Essa modulação é conhecida como *quadrature amplitude modulation* (QAM) [29].

O sinal $s_i(t)$ é necessariamente limitado em energia, que é definida como

$$E_{s_i} = \int_0^{T_s} s_i^2(t) dt, \quad i = 1, 2, \dots, M. \quad (1.1)$$

O canal está diretamente relacionado com o meio físico através do qual as formas de onda $s_i(t)$ são transmitidas. Diversos modelos para canais têm sido propostos englobando diferentes meios de comunicação. A existência desses modelos é fundamental para analisar o sistema analiticamente ou através de simulação.

Do mesmo modo que os bits de informação, e consequentemente os símbolos, têm uma estatística associada, o mesmo acontece com o canal. Após a forma de onda $s_i(t)$ ser transmitida através do canal, ela pode sofrer modificações aleatórias causadas por ruído aditivo, ruído multiplicativo, interferências, atenuação, seletividade em frequência, atrasos, desvanecimento, entre outros. Essas modificações têm uma distribuição de probabilidade associada e desde que o sinal $x(t)$ na entrada do receptor pode ser confundido com um símbolo do alfabeto diferente do símbolo transmitido.

O receptor tem a tarefa de observar o sinal recebido $x(t)$, a cada intervalo de T_s segundos, e fazer a melhor estimativa do sinal transmitido, ou equivalentemente, do símbolo transmitido. A estimativa do símbolo transmitido é feita em duas etapas. Primeiro o demodulador converte a forma de onda recebida $x(t)$ em um vetor N dimensional $\mathbf{r} = [r_1 \ r_2 \ \dots \ r_N]$, em que N é a dimensão da forma de onda transmitida. Por exemplo, na modulação ASK o sinal transmitido tem apenas uma dimensão, já na modulação 4-PSK, a dimensionalidade é dois [28]. A segunda etapa é realizada pelo detector. O vetor \mathbf{r} deve conter toda informação importante da forma de onda recebida e que é suficiente para que no processo de detecção o receptor possa decidir qual das M possíveis formas de onda é mais provável de ter sido transmitida.

Há duas formas principais de construir demoduladores, pode-se utilizar filtros casados ou correladores [29]. A construção dos detectores está ligada ao conhecimento da estatística do canal. Em geral, os detectores estabelecem uma regra baseada no vetor \mathbf{r} que maximize a probabilidade de uma decisão correta. Alguns desses critérios são [29]: máxima probabilidade a posteriori (MAP), máxima probabilidade a priori, e máxima

verossimilhança.

As probabilidades de erro de símbolo (p_s) e erro de bit (p_b) tornaram-se fundamentais no desenvolvimento e análise de novos sistemas de comunicações como medidas de desempenho. Na próxima seção algumas medidas de desempenho serão citadas e as probabilidades p_s e p_b serão definidas.

1.2 Medidas de Desempenho em Sistemas de Comunicações

O desempenho de um sistema de comunicação é caracterizado por diversos tipos de medida. O conjunto de medidas é diferente dependendo do sistema particular em estudo.

Em um sistema de múltiplo acesso como CDMA, TDMA ou FDMA, medidas como número de canais, taxa de transmissão e eficiência de uso do espectro são de natural importância. Adicionalmente, em um sistema móvel celular, medidas de desempenho importantes são, consumo de energia nos terminais móveis, tempo para *handoff*, tráfego, interferência, relação sinal-ruído, reuso de frequências, entre outros. Para sistemas de transmissão de dados, medidas importantes seriam taxa de transmissão, capacidade do servidor, número de colisões, entre outros.

Além das medidas listadas acima, há duas medidas de desempenho relacionadas à produção de erros no sistema, que são de particular importância em qualquer sistema de comunicação digital. Como discutido na seção anterior, a informação transmitida pelo sistema pode ser reconhecida em termos de bits ou de símbolos, de modo que é importante conhecer a probabilidade do sistema transmitir uma certa informação e recebê-la com erro. Essas medidas são conhecidas como probabilidade de erro de símbolo, e probabilidade de erro de bit (BER - *Bit Error Rate*). Esses valores são importantes pois demonstram o nível de confiabilidade do sistema, e têm sido utilizados desde os primeiros sistemas digitais, seja na especificação do sistema, ou como um critério de comparação para validar uma nova técnica.

Em um sistema com alfabeto de M símbolos, dado que s_i é transmitido, a probabilidade de erro é dada por

$$P(\text{erro}|s_i) = \sum_{j \neq i} P(s_j|s_i), \quad i = 1, 2, \dots, M, \quad (1.2)$$

onde, $P(s_j|s_i)$ é a probabilidade do receptor decidir pelo símbolo s_j dado que o símbolo s_i foi transmitido. A probabilidade de erro de símbolo é calculada ponderando as probabilidades de erro específicas, como segue

$$\begin{aligned} p_s &= \sum_{i=1}^M p_i \cdot P(\text{erro}|s_i) \\ &= \sum_{i=1}^M \sum_{j \neq i} p_i \cdot P(s_j|s_i), \end{aligned} \quad (1.3)$$

em que p_i é a probabilidade de transmitir o símbolo s_i . Caso os símbolos sejam equiprováveis, $p_i = \frac{1}{M}$, para $i = 1, 2, \dots, M$, e p_s é escrito como

$$p_s = \frac{1}{M} \sum_{i=1}^M \sum_{j \neq i} P(s_j|s_i) \quad (1.4)$$

A BER é definida como a probabilidade que uma unidade de informação (bit) seja interpretada erroneamente pelo receptor. Sendo $p^{(0)}$ e $p^{(1)}$ as probabilidades da fonte emitir um bit “0” e “1”, respectivamente, a probabilidade de erro de bit p_b é dada por

$$p_b = p^{(0)} \cdot p(1|0) + p^{(1)} \cdot p(0|1) \quad (1.5)$$

em que $p(y|x)$ é a probabilidade de receber o bit “ y ” dado que o bit “ x ” foi emitido pela fonte. Se os bits forem equiprováveis, $p^{(0)} = p^{(1)} = 1/2$, e p_b torna-se

$$p_b = \frac{1}{2} (p(1|0) + p(0|1)). \quad (1.6)$$

De modo geral, é possível obter a BER a partir da probabilidade de erro de símbolo p_s . Se $n(s_j, s_i)$ é o número de bits que diferem entre os símbolos s_j e s_i , então

$$p_b = \sum_{i=1}^M \sum_{j \neq i} n(s_j, s_i) \cdot p_i \cdot P(s_j|s_i), \quad (1.7)$$

em que foi feito uso das Equações 1.2 e 1.3. Em alguns sistemas o alfabeto de símbolos pode ser binário, $M = 2$. Nesse caso $p_b = p_s$ e a Equação 1.7 é simplificada à Equação 1.5.

Na próxima seção serão relacionados os principais métodos para calcular p_s e p_b . No Capítulo 6, sistemas com códigos de controle de erro são considerados. Para esses sistemas a probabilidade de erro de uma palavra código é de interesse, e como será visto, uma palavra-código pode ser interpretada como um símbolo transmitido.

1.3 Cálculo de Desempenho em Sistemas de Comunicações

Um passo necessário no desenvolvimento de um novo sistema de comunicações é a experimentação. Entretanto, construir protótipos do sistema é um processo caro e consome tempo. Além disso, depende da disponibilidade da equipe que projeta o hardware e dos componentes necessários. O uso de prototipagem é relativamente inflexível, pois a escolha de certos componentes frequentemente restringe os valores de alguns parâmetros do sistema. Felizmente é possível modelar o sistema e prever algum comportamento usando métodos para cálculo de desempenho. Como detalhado na seção anterior, as medidas de desempenho de interesse são a probabilidade de erro de símbolo e a probabilidade de erro de bit (BER). Já que a formulação para calcular essas medidas é similar, a partir de agora, as medidas de desempenho de interesse serão referenciadas apenas como cálculo da probabilidade de erro.

Há diferentes abordagens para prever o valor da probabilidade de erro, cada uma com vantagens e desvantagens próprias. A aplicabilidade de um dado método depende das particularidades do esquema de modulação, codificação, do canal e da estrutura do receptor.

A aplicação de técnicas analíticas tem a vantagem que expressões em fórmulas fechadas são obtidas baseando-se estritamente na manipulação de equações. Entretanto, na medida que a complexidade do sistema aumenta, pode-se encontrar características de não-linearidade, memória, codificação, diversidade, espalhamento espectral, embaralhamento, entre outros, que dificultam a obtenção de uma fórmula fechada para o cálculo da probabilidade de erro. Além disso, os modelos de canais que surgiram nos últimos anos são complexos, resultando em uma função densidade de probabilidade

(fdp) não gaussiana na saída do receptor [29]. O que torna os cálculos ainda mais difíceis.

Abordagens alternativas [20], baseadas em limites ou aproximações, têm sido amplamente utilizadas na literatura, entretanto, as abordagens com maior generalidade não possibilitam estimativas precisas de p_b , enquanto que estratégias mais promissoras são difíceis de aplicar e perdem alguma generalidade. As abordagens mais utilizadas são: desigualdade de Chebyshev, limite de Chernoff, aproximação gaussiana, aproximação da fdp por série, extrapolação da calda e aproximação ponto de sela (*Saddle-point*) [20]. Esses métodos têm sido aplicados em alguns sistemas, mas devido a sua imprecisão, complexidade e falta de generalidade, não têm sido muito utilizados na prática, e não serão abordados no presente trabalho.

No final da década de setenta tornou-se comum o uso de técnicas que utilizam o computador para estimar o desempenho de sistemas de comunicações. Essa tendência foi motivada pelos sistemas de satélite, principalmente devido a dificuldade de analisar em forma fechada um sistema não-linear com restrição na largura de faixa. A não-linearidade gera uma componente de ruído não-gaussiana no receptor e a restrição na faixa introduz interferência intersimbólica, que em geral caracteriza um enlace digital de satélite.

Simulação em computador de sistemas de comunicações é uma atividade multidisciplinar que envolve elementos de diversos campos, tais como modelagem de sistemas, projeto de sistemas, questões de implementação, modelagem e geração de processos aleatórios, estatística e teoria de estimação. A simulação reproduz o sistema utilizando modelos discretos no tempo, e usa geradores de números aleatórios para representar os processos estocásticos envolvidos.

Tradicionalmente as simulações são chamadas de simulações de **Monte Carlo**, e são caracterizadas por usarem geradores de números aleatórios e serem implementações diretas do estimador da média amostral, Capítulo 2. Comparado com outros métodos de cálculo de desempenho, Monte Carlo é a abordagem mais genérica. Outra vantagem da simulação de Monte Carlo é que, em contraste com os métodos de aproximação, a precisão do estimador pode ser facilmente determinada durante a simulação, como será mostrado no Capítulo 3.

A evolução dos sistemas de comunicações, hoje admitindo transmissão de vídeo, áudio e dados a altas taxas, juntamente com o surgimento de tecnologias com transmissão de alto grau de confiabilidade (fibras-ópticas), implica que a probabilidade de

erro torna-se cada vez menor. Como será visto no Capítulo 3, quanto menor o valor da probabilidade de erro, maior o tempo necessário para a simulação obter estimativas precisas. Por isso, o método de Monte Carlo torna-se impraticável para sistemas com baixa probabilidade de erro. A origem desse problema é que se a probabilidade de erro é muito pequena, os erros, que são os eventos significantes na simulação, são raros de ocorrer e a simulação não consegue obter a informação necessária para estimar a probabilidade de erro. Quando os eventos importantes tornam-se raros, impossibilita o método de Monte Carlo extrair conclusões estatisticamente significativas.

Dadas as dificuldades da simulação de Monte Carlo, é desejável usar técnicas que reduzem o tempo necessário para obter uma estimativa precisa da probabilidade de erro, tornando a simulação mais rápida. As técnicas que aceleram o processo de estimação são conhecidas como **técnicas de redução de variância**.

Dentre essas técnicas, a mais promissora é a técnica de **valorização das amostras** (IS - *Importance Sampling*) que é apresentada no Capítulo 4. Esse método também é conhecido como método de Monte Carlo modificado, pois realiza uma simulação de Monte Carlo modificando os processos estocásticos envolvidos. É uma técnica que usa conceitos da estatística, e se baseia em aumentar a proporção dos eventos raros (erros) que são de interesse na simulação. Para tanto, a distribuição de probabilidade do ruído é polarizada na direção das regiões em que os erros ocorrem.

A idéia por trás da polarização é que a probabilidade que um erro de comunicação ocorra durante a simulação torna-se maior que em uma simulação de Monte Carlo convencional, de modo que uma simulação mais curta será suficiente para adquirir medidas estatisticamente significativas.

Na próxima seção será definido formalmente o problema do cálculo da probabilidade de erro em sistemas de comunicações digitais. Em seguida, um exemplo simples é apresentado em que a probabilidade de erro pode ser obtida analiticamente.

1.4 Definição Geral do Problema

As equações para a probabilidade de erro de bit e de símbolo obtidas na Seção 1.2 fizeram uso de estatísticas envolvidas nos processos de transmissão e recepção, que são quantidades desconhecidas no processo de simulação, e não foram usadas funções que representassem o sistema de comunicação em questão. Nessa seção será exposto um modelo de sistema de comunicação mais conveniente para o propósito de estimação, e

a probabilidade de erro será definida de uma forma mais detalhada que anteriormente.

Considere o modelo simplificado de um sistema de comunicações digitais mostrado na Figura 1.2. A função do sistema $g(\cdot)$ abrange o transmissor, o canal, e a parte do receptor que é anterior ao decisor, ou comparador. O valor de cada símbolo \hat{s}_k na saída do decisor depende do valor de cada amostra do sinal v_k em sua entrada. Idealmente, a seqüência de símbolos $\{\hat{s}_k\}$ reconstruída pelo decisor deveria ser igual à seqüência transmitida $\{s_k\}$. Entretanto, haverá um número de variáveis aleatórias X_1, X_2, \dots, X_L envolvidas na detecção do k -ésimo símbolo s_k , e portanto, haverá uma probabilidade não nula que a seqüência de saída não seja exatamente igual a seqüência de entrada. Isto é ilustrado na Figura 1.2 em que os elementos da seqüência $\{v_k\}$ dependem da seqüência de informação $\{s_k\}$ e do vetor de variáveis aleatórias $\mathbf{X} = (X_1, X_2, \dots, X_L)$.

O decisor compara o valor da amostra do sinal v_k com um conjunto de valores pré-definidos (os limiares), e gera diferentes símbolos na saída de acordo com diferentes faixas de valores de v_k . Portanto, o processo para obter a probabilidade de erro envolve encontrar a função densidade de probabilidade (fdp) $f_V(\cdot)$ da variável aleatória $V \doteq v_k$ sobre certas regiões.

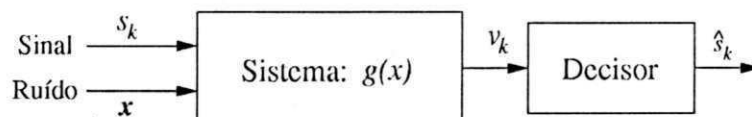


Figura 1.2: Modelo para análise de desempenho de um sistema de comunicações.

Um exemplo simples pode ser usado para ilustrar o problema geral descrito acima. Considere um sistema de comunicação em que cada unidade de informação (bit) é transmitida por um pulso retangular de duração T_s . Dependendo da informação a ser enviada a amplitude do pulso pode ser tanto A ou $-A$.

O canal modifica o sinal transmitido adicionando realizações decorrelacionadas de uma variável aleatória gaussiana com média zero e variância unitária. O receptor é considerado idealmente sincronizado com o transmissor, e realiza uma operação de integração para cada pulso transmitido. Com essa descrição, as amostras do sinal na entrada do comparador para cada bit transmitido podem ser escritas como

$$v_k = \pm A \cdot T_s + n_k$$

em que o sinal depende da unidade de informação transmitida, e n_k denota a compo-

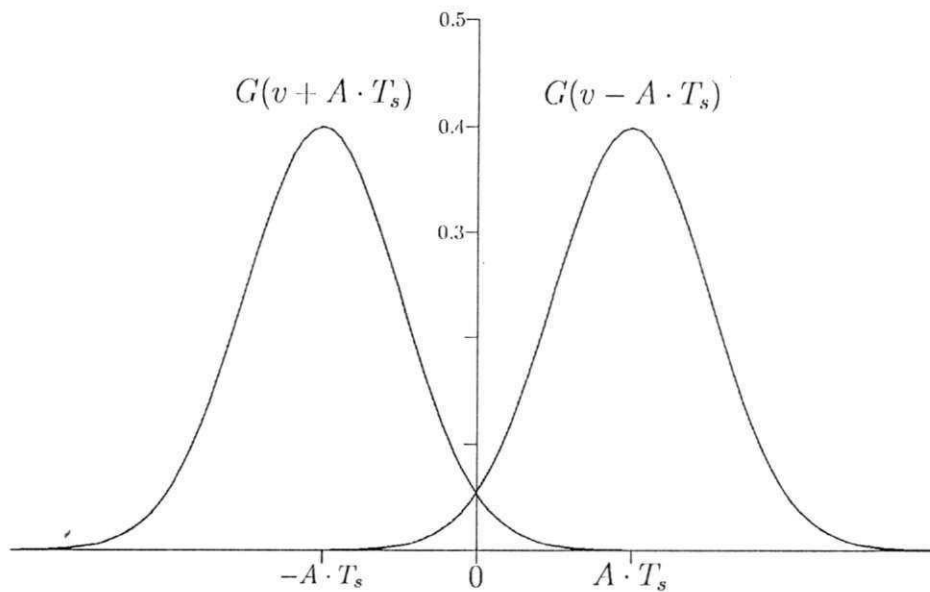


Figura 1.3: Composição da fdp de v_k .

nente do ruído gaussiano. O decisor compara v_k com um limiar τ . Quando $v_k > \tau$ o decisor determina que o símbolo transmitido foi aquele de amplitude A ; caso contrário ele conclui que um pulso com amplitude $-A$ foi transmitido. A fdp de v_k , Figura 1.3, pode então ser escrita como

$$f_V(v) = \begin{cases} G(v - A \cdot T_s), & \text{se } A \text{ foi transmitido} \\ G(v + A \cdot T_s), & \text{se } -A \text{ foi transmitido} \end{cases} \quad (1.8)$$

em que $G(\cdot)$ é a fdp gaussiana normalizada, com média nula e variância unitária, definida como

$$G(y) = \frac{1}{\sqrt{2\pi}} e^{-\frac{y^2}{2}}.$$

Se a informação transmitida é equiprovável, a probabilidade que o bit detectado seja diferente do bit transmitido é

$$\begin{aligned} p_b &= \frac{1}{2} \int_0^{\infty} G(x + A \cdot T_s) dx + \frac{1}{2} \int_{-\infty}^0 G(x - A \cdot T_s) dx \\ &= \frac{1}{2} \int_0^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{(x+A \cdot T_s)^2}{2}} dx + \frac{1}{2} \int_{-\infty}^0 \frac{1}{\sqrt{2\pi}} e^{-\frac{(x-A \cdot T_s)^2}{2}} dx. \end{aligned} \quad (1.9)$$

Devido a simetria da distribuição gaussiana as duas integrais acima são idênticas, portanto a Equação 1.9 pode ser simplificada resultando em

$$p_b = \int_0^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{(x+A \cdot T_s)^2}{2}} dx. \quad (1.10)$$

A Equação 1.10 não pode ser resolvida analiticamente e desde que tabelas para a distribuição gaussiana tem-se tornado bastante populares. É comum escrever integrais definidas da distribuição gaussiana normalizada em termos da função erro $Q(\cdot)$,

$$p_b = Q(A \cdot T_s)$$

em que

$$Q(\alpha) = \frac{1}{\sqrt{2\pi}} \int_{\alpha}^{\infty} e^{-\frac{y^2}{2}} dy. \quad (1.11)$$

Formalmente, o problema de calcular o desempenho pode ser descrito como segue. Dado o vetor de variáveis aleatórias $\mathbf{X} = (X_1, X_2, \dots, X_L)$ com fdp conjunta $f_{\mathbf{X}}(\mathbf{x})$ conhecida, que representa as diferentes variáveis aleatórias envolvidas em cada detecção. A probabilidade de erro é então obtida integrando a variável aleatória $V = g(\mathbf{X})$ sobre uma região χ , isto é,

$$\begin{aligned} p_e &= P(V \in \chi) = \int_{\chi} f_V(v) dv \\ &= \int_{-\infty}^{+\infty} H(v) f_V(v) dv \end{aligned} \quad (1.12)$$

em que χ é a região de valores de V que ocasionam erros na detecção do sinal, $g(\cdot)$ é uma função escalar que representa o modelo do sistema, como indicado na Figura 1.2, e $H(\cdot)$ é a função indicadora de erros dada por

$$H(v) \doteq \begin{cases} 1, & \text{se } v \in \chi \\ 0, & \text{se } v \notin \chi. \end{cases} \quad (1.13)$$

A definição de $H(v)$ é importante para a elaboração dos estimadores da probabilidade de erro, pois durante a simulação ela será encontrada implicitamente no processo

de decisão. Em muitos problemas a principal dificuldade é que a fdp $f_V(\cdot)$ não é conhecida. Para solucionar esse problema, a Equação 1.12 pode ser escrita em função das variáveis aleatórias da entrada [30], ou seja

$$p_e = \int_{-\infty}^{+\infty} H(g(\mathbf{x})) f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} \quad (1.14)$$

em que o vetor \mathbf{x} é uma realização do vetor aleatório $\mathbf{X} = (X_1, X_2, \dots, X_L)$. As Equações 1.12 e 1.14 são bastante semelhantes ao cálculo da esperança de uma variável aleatória. O operador valor esperado de uma variável aleatória Y com fdp $f_Y(y)$ é definido como

$$E_Y[y] = \int_{-\infty}^{+\infty} y f_Y(y) dy$$

e desde que as Equações 1.12 e 1.14 podem ser escritas como

$$\begin{aligned} p_e &= E_V[H(V)] \\ &= E_{\mathbf{X}}[H(g(\mathbf{X}))] \end{aligned} \quad (1.15)$$

em que os operadores $E_V[\cdot]$ e $E_{\mathbf{X}}[\cdot]$ representam o valor esperado calculado em relação as fdp's $f_V(\cdot)$ e $f_{\mathbf{X}}(\cdot)$, respectivamente. Como mostra a Equação 1.15, o problema de estimar a probabilidade de erro é equivalente a encontrar o valor esperado da variável aleatória $H(V)$ que indica a ocorrência de erros.

1.5 Organização do Trabalho

No Capítulo 2 é introduzido o problema de estimação. Os estimadores da média amostral e variância amostral são definidos, e suas principais propriedades são apresentadas. Além disso, medidas para avaliar o desempenho de estimadores são estabelecidas e posteriormente utilizadas com o objetivo de compará-los.

O método de Monte Carlo para a estimação da probabilidade de erro é definido no Capítulo 3. Algumas propriedades importantes são demonstradas, e medidas para avaliar a precisão das estimativas são discutidas extensivamente. Um problema analiticamente tratável em sistemas de comunicações é utilizado como exemplo em que são comparados os resultados obtidos analiticamente e com o método de Monte Carlo.

A simulação com valorização das amostras é explicada no Capítulo 4. Uma discussão sobre despolarização e precisão do estimador é apresentada. Também é estudada a questão importante de selecionar a distribuição utilizada na simulação. Para finalizar o capítulo é demonstrado o benefício da valorização das amostras em relação ao método de Monte Carlo simulando um sistema de comunicações binário.

O Capítulo 5 trata da otimização de parâmetros associados com a implementação da valorização das amostras. Após mencionar vários métodos existentes, a aplicação de métodos numéricos para otimização é discutida.

A utilização de valorização das amostras é estendida no Capítulo 6 para sistemas com codificação. Essa abordagem é introduzida para sistemas com código de bloco, e discutida superficialmente para códigos convolucionais.

O Capítulo 7 propõe um método de simulação de sistemas com codificação que usa os benefícios de valorização das amostras e métodos de otimização numérica. As vantagens do método são demonstradas para um sistema com código de Hamming em um canal aditivo gaussiano. As conclusões gerais dessa dissertação são apresentadas no Capítulo 8.

Capítulo 2

Estimação

Os métodos para estimação da probabilidade de erro discutidos nesse trabalho estão relacionados ao uso de inferência estatística, que consiste em fazer declarações probabilísticas sobre quantidades desconhecidas. Por exemplo, pode-se calcular médias, variâncias, probabilidades, ou algum outro parâmetro desconhecido a partir de um conjunto de amostras aleatórias pertencentes à distribuição de probabilidades de interesse. Inferência estatística é geralmente dividida em estimação e teste de hipóteses.

Na teoria de estimação o objetivo é obter uma estimativa de um parâmetro envolvendo uma variável aleatória que é resultado de um experimento. Tal parâmetro pode ser o valor esperado ou a variância da fdp. Em um teste de hipóteses o objetivo é utilizar os resultados do experimento para tomar uma decisão que seja estatisticamente mais adequada em um determinado contexto [31].

Nesse capítulo será definido o problema de estimação no contexto de inferência estatística. Alguns estimadores de interesse serão apresentados e suas propriedades serão estudadas. Por fim algumas medidas serão estabelecidas para avaliar o desempenho de estimadores e compará-los.

2.1 Definição do Problema de Estimação

Fazendo x_1, x_2, \dots, x_N uma seqüência de amostras da variável aleatória X , que assume-se ter função densidade de probabilidade (fdp) $f_X(x; \theta)$, que depende de um parâmetro desconhecido θ . Baseando-se na seqüência de amostras deseja-se encontrar uma função $h(x_1, x_2, \dots, x_N)$ que pode ser usada para estimar θ .

Para simplificar a notação, a função $h(x_1, x_2, \dots, x_N)$ usada para estimar o parâmetro

θ será denotada pôr

$$\hat{\theta} = h(x_1, x_2, \dots, x_N). \quad (2.1)$$

e referenciada como um estimador de θ . Para um conjunto particular de amostras x_1, x_2, \dots, x_N o valor do estimador $\hat{\theta}$ constitui uma estimativa do parâmetro θ .

Como a estimativa $\hat{\theta}$ depende dos valores x_1, x_2, \dots, x_N , que são aleatórios, um estimador também é uma variável aleatória, e a estimativa $\hat{\theta}$ é uma realização particular dela. Um estimador tem fdp própria e algumas de suas propriedades serão derivadas dessa distribuição. Observando a equação 2.1 percebe-se que para utilizar um estimador não é necessário o conhecimento da fdp $f_X(x; \theta)$, mas apenas das amostras x_1, x_2, \dots, x_N .

Os métodos mais populares para obter estimadores são o método dos momentos, Bayes e o método da máxima verossimilhança. Este último visa maximizar a fdp conjunta n -dimensional $f(X_1, X_2, \dots, X_N; \theta)$ no ponto x_1, x_2, \dots, x_N em relação a θ . Com esse método, para obter o estimador é necessário o conhecimento da fdp conjunta, além disso, o problema de maximizar a fdp limita sensivelmente a aplicação desse método.

A estimação bayesiana assume que o parâmetro desconhecido θ é uma variável aleatória. A distribuição de θ é escolhida de acordo com algum conhecimento que antecede a estimação. Por exemplo, sabendo que θ é próximo de algum valor θ_0 , pode-se escolher uma fdp para θ que tem uma pequena variância centrada em θ_0 . Para aplicar essa teoria é preciso especificar uma função de custo, tal como erro quadrático ou erro médio, e também uma distribuição a priori para o parâmetro θ . Especificações significativas podem existir, a princípio, mas podem ser difíceis e demandar muito tempo para serem obtidas [31].

Na próxima seção será descrito o método dos momentos, por ser útil na definição dos estimadores discutidos nesse trabalho.

2.2 Método dos Momentos

O método dos momentos baseia-se em equações intuitivas para estimar os momentos de uma variável aleatória X . O r -ésimo momento de X com fdp contínua¹ $f(x)$ é definido

¹Fórmulas para o caso em que X é uma variável aleatória discreta são análogas.

como [30]:

$$E(X^r; \theta) = \int_{-\infty}^{\infty} x^r f(x; \theta) dx. \quad (2.2)$$

Para um melhor entendimento, a dependência de $f(x)$ e $E(X^r)$ em relação a θ é explícita na notação. O estimador para o resultado da equação 2.2 é então obtido intuitivamente como

$$\hat{X}_r = \frac{1}{N} \sum_{i=1}^N X_i^r = h(X_1, X_2, \dots, X_N) \quad (2.3)$$

Igualando as equações 2.2 e 2.3 pode-se obter uma estimativa de θ [32]. No contexto das simulações do presente trabalho, os estimadores da média amostral e variância amostral são de maior interesse e serão discutidos a seguir.

O estimador para a média μ é obtido substituindo $r = 1$ na equação 2.3, e é dado por

$$\hat{\mu} = \frac{1}{N} \sum_{i=1}^N X_i. \quad (2.4)$$

O estimador da variância também é obtido intuitivamente utilizando o método dos momentos. A variância de X é definida como

$$\sigma_X^2 = E[(X - \mu)^2] = \int_{-\infty}^{\infty} (x - \mu)^2 f(x) dx. \quad (2.5)$$

O estimador da variância obtido intuitivamente utilizando o método dos momentos é

$$\hat{S}_X^2 = \frac{1}{N} \sum_{i=1}^N (X_i - \hat{X})^2. \quad (2.6)$$

O estimador da média é utilizado no método de Monte Carlo para a estimação da BER, como será visto no Capítulo 3. O objetivo do presente trabalho é utilizar diferentes estimadores e compará-los. Para isso é necessário conhecer um número de

propriedades dos estimadores e obter medidas de desempenho para a escolha do melhor estimador.

O termo melhor não pode ser atribuído a um estimador para todos os casos em que for aplicado, pois o desempenho depende do contexto. Por exemplo, um determinado estimador para a probabilidade de erro pode ter um bom desempenho para uma baixa relação sinal ruído (SNR), mas pode não ter boas estimativas para SNR com valores maiores.

Nas próximas seções serão descritas propriedades que facilitarão a escolha de um estimador sobre outro.

2.3 Despolarização

Como dito anteriormente, um estimador $\hat{\theta}$ é uma variável aleatória, pois depende do conjunto de realizações X_1, X_2, \dots, X_N . Assim, cada uso do estimador pode resultar em um valor diferente dentro do domínio da fdp $f_{\hat{\theta}}(\cdot)$ do estimador. Entretanto, espera-se que as estimativas obtidas sejam próximas do valor exato θ , ou seja, uma propriedade desejada é que:

$$E[\hat{\theta}] = \theta. \quad (2.7)$$

Caso essa condição seja verificada, o estimador é dito despolarizado. Por exemplo, o estimador da média é sempre despolarizado, independente da distribuição de X , como é mostrado a seguir:

$$\begin{aligned} E[\hat{\mu}] &= E \left[\frac{1}{N} \sum_{i=1}^N X_i \right] \\ &= \frac{1}{N} \sum_{i=1}^N E[X_i] \\ &= \frac{1}{N} \cdot N \cdot E[X] \\ &= E[X] = \mu. \end{aligned} \quad (2.8)$$

Alguns estimadores polarizados podem ter seus valores médios corrigidos apenas multiplicando-os por uma constante, é o caso do estimador da variância. Substituindo a Equação 2.6 em 2.7, tem-se

$$\begin{aligned}
E[\hat{S}_X^2] &= E\left[\frac{1}{N}\sum_{i=1}^N(X_i - \hat{X})^2\right] \\
&= \frac{1}{N}\sum_{i=1}^N E[X_i^2 - 2X_i\hat{X} + \hat{X}^2] \\
&= \frac{1}{N}\sum_{i=1}^N (E[X_i^2] - 2E[X_i\hat{X}] + E[\hat{X}^2]). \tag{2.9}
\end{aligned}$$

Cada valor esperado no lado direito da equação acima será obtido separadamente. A partir da definição da variância,

$$\begin{aligned}
\sigma_X^2 &= E[(X - E[X])^2] = E[X^2 - 2XE[X] + E^2[X]] \\
&= E[X^2] - 2E[X]E[X] + E^2[X] \\
&= E[X^2] - E^2[X]. \tag{2.10}
\end{aligned}$$

Isolando $E[X^2]$ resulta,

$$E[X^2] = \sigma_X^2 + E^2[X] = \sigma_X^2 + \mu^2. \tag{2.11}$$

A quantidade \hat{X} é uma variável aleatória e não pode ser retirada do valor esperado, seu cálculo é obtido como segue:

$$\begin{aligned}
E[X_i\hat{X}] &= E\left[X_i \cdot \frac{1}{N}\sum_{j=1}^N X_j\right] \\
&= \frac{1}{N}E\left[X_i^2 + \sum_{j \neq i} X_i \cdot X_j\right] \\
&= \frac{1}{N}\left(E[X_i^2] + \sum_{j \neq i} E[X_i \cdot X_j]\right) \\
&= \frac{1}{N}\left(E[X_i^2] + \sum_{j \neq i} E[X_i] \cdot E[X_j]\right) \\
&= \frac{1}{N}(\sigma_X^2 + \mu^2 + (N-1) \cdot \mu^2) \tag{2.12}
\end{aligned}$$

em que foi utilizada a independência entre X_i e X_j , para $i \neq j$. Por fim,

$$\begin{aligned}
E[\hat{X}^2] &= E\left[\left(\frac{1}{N}\sum_{j=1}^N X_j\right) \cdot \left(\frac{1}{N}\sum_{k=1}^N X_k\right)\right] \\
&= E\left[\frac{1}{N^2}\sum_{j=1}^N\sum_{k=1}^N X_j X_k\right] \\
&= \frac{1}{N^2}E\left[\sum_{j=1}^N X_j^2 + \sum_{j=1}^N\sum_{k \neq j}^N X_j X_k\right] \\
&= \frac{1}{N^2}\left(\sum_{j=1}^N E[X_j^2] + \sum_{j=1}^N\sum_{k \neq j}^N E[X_j X_k]\right) \\
&= \frac{1}{N^2}(N(\sigma_X^2 + \mu^2) + N(N-1)\mu^2). \tag{2.13}
\end{aligned}$$

Substituindo os resultados 2.11, 2.12 e 2.13 na Equação 2.9, tem-se

$$\begin{aligned}
E[\hat{S}_X^2] &= \frac{1}{N}\sum_{i=1}^N\left(\sigma_X^2 + \mu^2 - \frac{2}{N}(\sigma_X^2 + \mu^2 + (N-1)\cdot\mu^2)\right) \\
&\quad + \frac{1}{N}\sum_{i=1}^N\left(\frac{1}{N^2}(N(\sigma_X^2 + \mu^2) + N(N-1))\right) \\
&= \left[\sigma_X^2 - \frac{2\sigma_X^2}{N} + \frac{\sigma_X^2}{N}\right] + \left[\mu^2 - \frac{2N}{N}\mu^2 + \frac{N^2}{N^2}\mu^2\right] \\
&= \sigma_X^2 - \frac{\sigma_X^2}{N} \\
&= \frac{(N-1)}{N} \cdot \sigma_X^2. \tag{2.14}
\end{aligned}$$

Esse resultado demonstra que o estimador da variância obtido intuitivamente pelo método dos momentos é polarizado, mas concluí-se que,

$$\hat{\sigma}_X^2 = \frac{1}{N-1}\sum_{i=1}^N (X_i - \hat{X})^2 \tag{2.15}$$

é um estimador despolarizado de $\hat{\sigma}_X^2$. É comum corrigir a polarização em estimadores apenas multiplicando por uma constante. No contexto de estimação, o fato do estimador ser despolarizado é mais que uma propriedade, é um requisito. Na próxima seção será introduzida uma métrica que é importante para comparar o desempenho de estimadores.

2.4 Eficiência

Considerando dois estimadores $\hat{\theta}_1$ e $\hat{\theta}_2$ do parâmetro θ , deseja-se escolher o que se aproxima do valor verdadeiro o maior número de vezes. Sabe-se que a variância de uma variável aleatória dá uma indicação do espalhamento da distribuição. Como exemplo, suponha que as fdp's dos estimadores despolarizados $\hat{\theta}_1$ e $\hat{\theta}_2$ sejam como mostrado na Figura 2.1. Sendo $V[\hat{\theta}_1]$ e $V[\hat{\theta}_2]$ as variâncias dos estimadores $\hat{\theta}_1$ e $\hat{\theta}_2$, respectivamente, se $V[\hat{\theta}_2] > V[\hat{\theta}_1]$, pode-se perceber que o estimador $\hat{\theta}_1$ é mais provável de assumir valores próximos a θ que o estimador $\hat{\theta}_2$, grosseiramente falando.

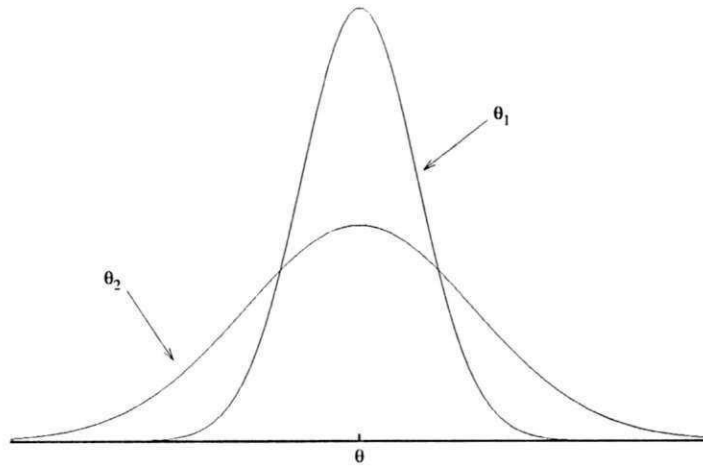


Figura 2.1: Comparação entre estimadores com variâncias diferentes.

Se $\hat{\theta}_1$ e $\hat{\theta}_2$ são estimadores despolarizados de θ , e $V[\hat{\theta}_2] > V[\hat{\theta}_1]$, diz-se que $\hat{\theta}_1$ é mais eficiente que $\hat{\theta}_2$, por outro lado, se um determinado estimador é polarizado, ter uma variância menor não implica necessariamente que o estimador é mais eficiente.

Nesse momento é importante diferenciar a estatística da amostra X da estatística dos estimadores. O estimador da variância $\hat{\sigma}_X^2$ tem a finalidade de obter um valor que se aproxime da variância da fdp de X , por outro lado, como $\hat{\sigma}_X^2$ é uma variável aleatória e conseqüentemente tem uma fdp associada, esse estimador também tem valor médio e variância associados. Ao avaliar a eficiência de um estimador, é a variância deste que é analisada, e não a variância da fdp de X .

Se um estimador $\hat{\theta}$ é despolarizado e tem variância nula, a fdp de $\hat{\theta}$ é $\delta(\hat{\theta} - \theta)$, e o estimador é dito perfeito. Na próxima seção será definida uma medida estatística para

o erro de um estimador.

2.5 Erro Normalizado

Uma medida bastante utilizada para avaliar estimadores é o erro normalizado. Esse valor é derivado do conceito de eficiência, em que quanto menor a variância, a estimativa tem maior chance de se aproximar do valor desconhecido. Seja o estimador $\hat{\theta}$ do parâmetro θ , o erro normalizado ε é dado por

$$\varepsilon = \frac{\sqrt{V[\hat{\theta}]}}{\theta}. \quad (2.16)$$

Esse valor é importante pois possibilita comparar estimadores para diferentes valores de θ . Na tentativa de escolher o estimador mais eficiente, é possível que para alguns valores de θ um estimador seja mais eficiente, e para valores maiores outro estimador tenha melhor eficiência. Como θ é desconhecido, na prática usa-se a estimativa $\hat{\theta}$ para calcular ε .

2.6 Consistência

Consistência é uma propriedade que dá uma idéia melhor do quão próximo o estimador está do valor verdadeiro. O estimador $\hat{\theta}$ de θ é dito ser consistente se, para qualquer valor pequeno $\delta > 0$,

$$\lim_{N \rightarrow \infty} P(|\theta - \hat{\theta}| < \delta) = 1 \quad (2.17)$$

em que N é o número de observações usadas pela estatística. Assim, garante-se que um estimador consistente aproxima-se de θ se o tamanho da amostra é grande o suficiente.

Pode-se mostrar que o estimador da média é consistente usando a desigualdade de Chebyshev [32]. Fazendo σ_X^2 a variância da variável aleatória X , a variância do estimador da média amostral $\hat{\mu}$ é σ_X^2/N [30], e o valor esperado é μ como mostrado na Equação 2.8. Aplicando a desigualdade tem-se

$$P(|\mu - \hat{\mu}| < \delta) = P\left(\left|\mu - \frac{1}{N} \sum_{i=1}^N X_i\right| < \delta\right) \geq 1 - \frac{\sigma_X^2}{N\delta^2}, \quad \delta > 0 \quad (2.18)$$

e como o lado direito da desigualdade tende a unidade quando N cresce, $\hat{\mu}$ é um estimador consistente de μ .

Utilizar a Equação 2.17 não é fácil, felizmente há um critério simples para determinar se um estimador é ou não consistente [32]. Usando a desigualdade de Chebyshev pode-se escrever,

$$\begin{aligned}
 P(|\hat{\theta}_N - \theta| \geq \delta) &\leq \frac{E[(\hat{\theta}_N - \theta)^2]}{2} = \frac{1}{\delta^2} E[\hat{\theta}_N - E(\hat{\theta}_N) + E(\hat{\theta}_N) - \theta]^2 \\
 &= \frac{1}{\delta^2} E\{[\hat{\theta}_N - E(\hat{\theta}_N)]^2\} + E\{[\hat{\theta}_N - \theta]^2\} \\
 &\quad + 2E[\hat{\theta}_N - E(\hat{\theta}_N)][E(\hat{\theta}_N) - \theta]^2 \\
 &= \frac{1}{\delta^2} (V(\hat{\theta}_N) + E\{[E(\hat{\theta}_N) - \theta]^2\}) \quad (2.19)
 \end{aligned}$$

em que $\hat{\theta}_N$ é o estimador do parâmetro θ baseado em uma amostra aleatória de tamanho N . Analisando a Equação 2.19 pode-se obter duas condições que se satisfeitas pode-se afirmar que o estimador é consistente, as condições são

$$\begin{cases} \lim_{N \rightarrow \infty} E(\hat{\theta}_N) = \theta \\ \lim_{N \rightarrow \infty} V(\hat{\theta}_N) \rightarrow 0 \end{cases} \quad (2.20)$$

Na prática essas condições em 2.20 são mais simples de serem aplicadas que a definição 2.17.

2.7 Intervalo de Confiança

O uso de um estimador $\hat{\theta}$ resulta em um único número que não é necessariamente igual a θ , o valor verdadeiro desconhecido, e não há nenhuma medida estatística que informe quanto o valor $\hat{\theta}$ é próximo de θ .

O cálculo de um intervalo de confiança obtém o intervalo (L, U) em que os limites são estatísticas da amostra aleatória usada pelo estimador, e associa uma probabilidade do valor desconhecido estar dentro desse intervalo.

Seja X_1, X_2, \dots, X_n uma seqüência de variáveis aleatórias utilizadas na estimação do parâmetro desconhecido θ . Os valores estatísticos L e U determinam o intervalo (L, U) , que envolve o parâmetro desconhecido θ com nível de confiança igual a $100(1 - \alpha)\%$. Com isso,

$$P(L < \theta < U) \geq 1 - \alpha, \quad 0 < \alpha < 1. \quad (2.21)$$

Ao invés de estimar um único número $\hat{\theta}$, deseja-se encontrar um intervalo $(L; U)$ que será $100(1 - \alpha)\%$ certo de conter o valor verdadeiro θ . A quantidade $(1 - \alpha)$ é chamada de coeficiente de confiança e em geral é um valor escolhido como 0.90, 0.95 ou 0.99. L e U são calculados usando a Equação 2.21 e dependem da fdp do estimador.

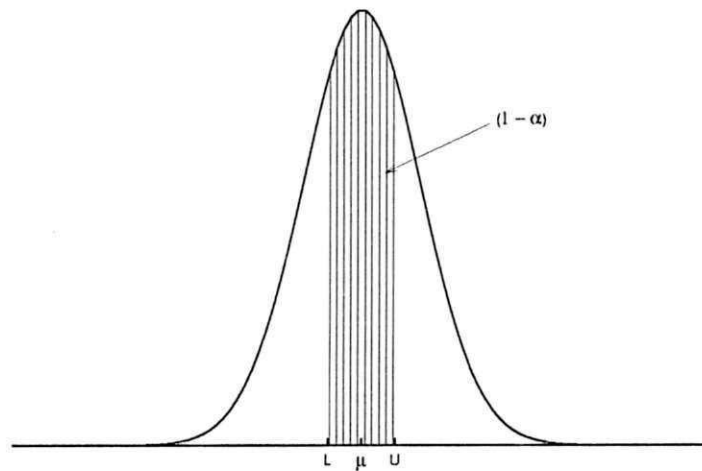


Figura 2.2: Intervalo de confiança para o estimador.

Como ilustração, a Figura 2.2 mostra a fdp de um estimador despolarizado \bar{X} para o valor médio de uma amostra aleatória gaussiana X_1, \dots, X_N . O estimador é considerado ter fdp gaussiana com média μ e variância σ_X^2/N . Deseja-se obter dois limites L e U , em função das amostras, de modo que,

$$P(L < \mu < U) \geq 1 - \alpha. \quad (2.22)$$

Substituindo a fdp gaussiana na Equação 2.22, tem-se

$$\int_L^U \frac{1}{\sqrt{2\pi\sigma_X^2/N}} \exp^{-\frac{(x-\mu)^2}{2\sigma_X^2/N}} dx \geq 1 - \alpha, \quad (2.23)$$

para algum α , $0 < \alpha < 1$. Para simplificar, é comum rescrever a Equação 2.23 em função da distribuição gaussiana normalizada². A variável aleatória

²Distribuição gaussiana com média nula e variância unitária.

$$Z = \frac{\bar{X} - \mu}{\sigma/\sqrt{N}}$$

é uma variável aleatória gaussiana normalizada e portanto um intervalo de confiança pode ser construído a partir de

$$\begin{aligned} P(-z_\alpha < Z < +z_\alpha) &= \Phi(z_\alpha) - \Phi(-z_\alpha) \\ &= 1 - \alpha \end{aligned} \tag{2.24}$$

em que $\Phi(\cdot)$ é a função gaussiana cumulativa, e o valor z_α é dado pela relação implícita, Figura 2.3,

$$\int_{-z_\alpha}^{z_\alpha} \frac{1}{\sqrt{2\pi}} \exp^{-\frac{x^2}{2}} dx = 1 - \alpha. \tag{2.25}$$

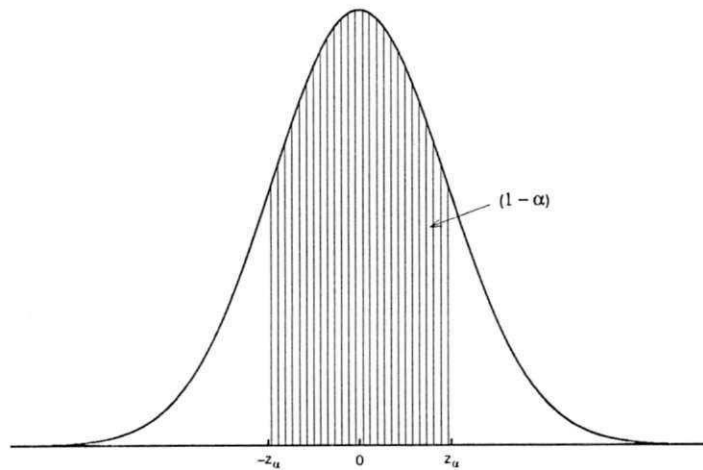


Figura 2.3: Cálculo de z_α para o intervalo de confiança.

Note que o intervalo foi escolhido simétrico porque nesse caso ele produz um intervalo mais curto. Reescrevendo os argumentos da Equação 2.24 para obter L e U , tem-se

$$\begin{aligned}
P(-z_\alpha < Z < +z_\alpha) &= P\left[-z_\alpha < \frac{\bar{X} - \mu}{\sigma/\sqrt{N}} < +z_\alpha\right] \\
&= P\left(\bar{X} - \frac{\sigma}{\sqrt{N}}z_\alpha \leq \mu \leq \bar{X} + \frac{\sigma}{\sqrt{N}}z_\alpha\right) \\
&= 1 - \alpha.
\end{aligned}$$

Portanto,

$$\begin{cases} L = \bar{X} - \frac{\sigma}{\sqrt{N}}z_\alpha \\ U = \bar{X} + \frac{\sigma}{\sqrt{N}}z_\alpha \end{cases} \quad (2.26)$$

em que z_α é uma constante que depende apenas de α . A Tabela 2.1 mostra alguns valores de z_α obtidos em uma tabela da distribuição cumulativa gaussiana normalizada [30].

Tabela 2.1: Alguns valores de $z_{1-\alpha/2}$ para calcular o intervalo de confiança.

α	z_α
0.90	1.645
0.95	1.960
0.99	2.576

Descrevendo de uma forma mais simples, para obter os valores L e U , Equação 2.26, é preciso seguir os seguintes passos:

1. Escolher o coeficiente de confiança $(1 - \alpha)$ e obter o valor z_α com o auxílio de uma tabela da distribuição gaussiana normalizada;
2. Obter a estimativa \hat{X} para a média μ aplicando o estimador da média amostral, Equação 2.4;
3. Obter a variância do estimador, ou utilizar o estimador da variância 2.9;
4. Calcular L e U usando as Equações 2.26.

No próximo capítulo a teoria de estimação é aplicada à simulação de sistemas de comunicações digitais visando estimar a probabilidade de erro do sistema. Essas simulações são conhecidas como simulações de Monte Carlo e por envolverem o uso de estimadores, as propriedades de polarização, eficiência, consistência, erro normalizado e o intervalo de confiança são de interesse e serão calculados.

Capítulo 3

Método de Monte Carlo

Como discutido no Capítulo 1, métodos analíticos exatos, com algumas exceções, não são viáveis para o cálculo de desempenho em sistemas de comunicações. A análise é geralmente muito difícil, e frequentemente requer uma simplificação excessiva do modelo do sistemas. Em alguns casos essa simplificação pode ser evitada usando métodos de aproximação, entretanto, a precisão das medidas obtidas não pode ser determinada, o que representa um obstáculo importante.

Com o aumento da capacidade de processamento dos computadores, simulação é uma alternativa que tem sido amplamente utilizada. As principais técnicas de simulação em computador são: Monte carlo, valorização das amostras¹ (*Importance Sampling* - IS), teoria do valor extremo, extrapolação da cauda e método semi-analítico [3].

A aplicabilidade de um dado método de estimação depende em geral das particularidades do sistema. Devido a maior simplicidade e generalidade, o método de Monte Carlo é sem dúvida o mais utilizado, tendo como desvantagem o crescimento do custo computacional com a diminuição da probabilidade de erro.

Neste capítulo será definido o método de Monte Carlo para a estimação da probabilidade de erro p_e em sistemas de comunicações digitais. As propriedades de despolarização e consistência serão demonstradas, e serão apresentadas três alternativas para calcular o intervalo de confiança. Por fim, será exposto um exemplo de uma simulação de Monte Carlo para um sistema simples.

¹Uma simulação de Monte Carlo modificada.

3.1 Definição do Método de Monte Carlo

O nome simulação de Monte Carlo refere-se a qualquer estratégia que envolve a geração de números aleatórios. Em geral essa técnica pode ser aplicada para calcular integrais de funções determinísticas ou estocásticas sobre regiões complicadas [33]. O método baseia-se em implementar uma seqüência de experimentos de Bernoulli [30], ou seja, conta-se o número de sucessos (erros) e divide pelo número total de símbolos transmitidos.

A aplicabilidade de Monte Carlo na estimação de probabilidades de erro de transmissão envolve reproduzir o sistema completamente. Para viabilizar o uso do computador, faz-se uma versão discreta no tempo dos diferentes sinais presentes no modelo do sistema, inclusive suas entidades aleatórias que são simuladas através de algoritmos de geração de números aleatórios [1] [34] [35]. Os blocos determinísticos presentes no modelo, como codificadores, filtros, moduladores, múltiplas antenas, etc, também precisam ser implementados no processo de simulação.

Como descrito na Seção 1.4, o objetivo do estimador da probabilidade de erro p_e é calcular uma integral da forma

$$p_e = E\{H(g(\mathbf{X}))\} = \int_{-\infty}^{\infty} H(g(\mathbf{x}))f_{\mathbf{X}}(\mathbf{x})d\mathbf{x}. \quad (3.1)$$

em que $H(\cdot)$ é a função indicadora de erro, a função do sistema $g(\cdot)$ abrange o transmissor, o canal, e a parte do receptor anterior ao decisor, e X é a vetor aleatório envolvido no processo de decisão.

No processo de simulação a função $H(\cdot)$ indica se houve erro, ou seja, ela é calculada após o decisor contido no receptor do sistema, comparando o símbolo recebido com o símbolo transmitido. Como mostra a Equação 3.1, o problema de calcular p_e pode ser visto como um cálculo da média da variável aleatória $H(g(X))$. Daí, o estimador de Monte Carlo é o estimador da média amostral

$$\hat{p}_{MC} = \frac{1}{N} \sum_{i=1}^N H(g(\mathbf{x}_i)) \quad (3.2)$$

em que \mathbf{x}_i é o i -ésimo vetor gerado de \mathbf{X} e N é o número de iterações na simulação.

A interpretação desse estimador é simples. Suponha que na i -ésima iteração do algoritmo \mathbf{x}_i seja o vetor resultante dos diferentes processos aleatórios envolvidos na

decisão do i -ésimo símbolo transmitido. O valor recebido pelo decisor do sistema será $g(\mathbf{x}_i)$. Após o processo de decisão, o valor de $H(g(\mathbf{x}_i))$ é encontrado comparando o símbolo transmitido com o símbolo recebido. Se um erro foi encontrado, o somatório na Equação 3.2 se encarrega de incrementar o contador de erros na simulação. Por fim, a estimativa de p_e é calculada como a proporção do número de erros em relação ao número total de símbolos N .

Pela definição de $H(\cdot)$, esta função retorna apenas dois valores: “1” se a amostra aleatória \mathbf{x} estiver na região de erro, e “0” caso contrário. Com isso uma simulação de Monte Carlo pode ser vista como uma seqüência de N processos de Bernoulli [30], com probabilidade de “sucesso” igual a p_e .

Processos de Bernoulli são repetições independentes do mesmo experimento em condições idênticas, como lançar uma moeda, ou atirar um dado. No contexto de estimar a probabilidade de erro, o experimento é enviar um símbolo pelo sistema de comunicações digitais e identificar se houve “sucesso” ($H(\mathbf{x}_i) = 1$), que nesse contexto é um erro de transmissão e tem probabilidade p_e .

O estimador de Monte Carlo soma o número de ocorrências de “sucessos” e divide pelo número total de repetições do experimento. A fdp do número de sucessos em N processos de Bernoulli é a distribuição Binomial. Portanto, o estimador de Monte Carlo \hat{p}_{MC} tem distribuição binomial dada por

$$P(\hat{p}_{MC} = y/N) = \binom{N}{y} p_e^y (1 - p_e)^{N-y}, \quad y = 0, \dots, N \quad (3.3)$$

em que $p_e = P[H(g(\mathbf{X})) = 1]$ é a probabilidade de “sucesso” no processo de Bernoulli. Note que o termo y/N no lado esquerdo da equação é devido à divisão por N efetuada pelo estimador.

3.2 Despolarização e Consistência

Conforme discutido no Capítulo 2, duas propriedades necessárias aos estimadores são despolarização e consistência. Como o método de Monte Carlo é um estimador da média amostral, a propriedade de despolarização é demonstrada da mesma forma que na Equação 2.8, ou seja,

$$\begin{aligned}
E[\hat{p}_{MC}] &= E\left\{\frac{1}{N}\sum_{i=1}^N H(g(\mathbf{x}_i))\right\} \\
&= \frac{1}{N} \cdot N \cdot E[H(g(\mathbf{x}))] \\
&= p_e.
\end{aligned} \tag{3.4}$$

em que foi utilizada a definição do estimador de Monte Carlo, Equação 3.2. Note que esse resultado independe da distribuição de \mathbf{X} .

Como visto na Seção 2.6, para mostrar que um estimador é consistente basta satisfazer as condições 2.20. A primeira condição já é satisfeita pela Equação 3.4, portanto basta demonstrar que o estimador também satisfaz a segunda condição. Partindo da Equação 2.10 e substituindo \hat{p}_{MC} pela definição, Equação 3.2, tem-se

$$\begin{aligned}
\sigma_{\hat{p}_{MC}}^2 &= E[\hat{p}_{MC}^2] - E^2[\hat{p}_{MC}] = E[\hat{p}_{MC}^2] - p_e^2 \\
&= E\left[\left(\frac{1}{N}\sum_{i=1}^N H(g(\mathbf{x}_i))\right) \cdot \left(\frac{1}{N}\sum_{j=1}^N H(g(\mathbf{x}_j))\right)\right] - p_e^2 \\
&= \left[\frac{1}{N^2}\sum_{i=1}^N \sum_{j=1}^N E[H(g(\mathbf{x}_i))H(g(\mathbf{x}_j))]\right] - p_e^2 \\
&= \left[\frac{1}{N^2}\sum_{i=1}^N E[H(g(\mathbf{x}_i))H(g(\mathbf{x}_i))] + \frac{2}{N^2}\sum_{i=1}^N \sum_{j=i+1}^N E[H(g(\mathbf{x}_i))H(g(\mathbf{x}_j))]\right] - p_e^2
\end{aligned}$$

em que nessa última igualdade foram separados os termos em que $i = j$ em uma única soma. Admitindo independência entre os vetores aleatórios \mathbf{x}_i e \mathbf{x}_j , para $i \neq j$, e sabendo que $H^2(g(\mathbf{x}_i)) = H(g(\mathbf{x}_i))$, ver definição na Equação 1.13, resulta

$$\begin{aligned}
\sigma_{\hat{p}_{MC}}^2 &= \left[\frac{1}{N^2}\sum_{i=1}^N E[H(g(\mathbf{x}_i))] + \frac{2}{N^2}\sum_{i=1}^N \sum_{j=i+1}^N E[H(g(\mathbf{x}_i))]E[H(g(\mathbf{x}_j))]\right] - p_e^2 \\
&= \left[\frac{1}{N^2} \cdot N \cdot p_e + \frac{2}{N^2}\sum_{i=1}^N \sum_{j=i+1}^N p_e^2\right] - p_e^2 \\
&= \left[\frac{p}{N} + \frac{2}{N^2} \cdot \frac{(N-1)N}{2} p_e^2\right] - p_e^2 \\
&= \frac{1}{N} [p_e + (N-1)p_e^2 - Np_e^2] \\
&= \frac{p_e(1-p_e)}{N}.
\end{aligned} \tag{3.5}$$

É fácil perceber que quando $N \rightarrow \infty$, $\sigma_{\hat{p}_{MC}}^2 \rightarrow 0$, satisfazendo a segunda condição em 2.20. Esse resultado demonstra que o estimador de Monte Carlo é consistente².

A Equação 3.5 é importante na análise dos resultados da simulação. Como discutido na seção 2.4, o estimador mais eficiente é aquele que tem menor variância. Em geral, a variância é uma das medidas mais utilizadas para avaliar um estimador, e será utilizada constantemente no presente trabalho.

Embora o valor de p_e não seja conhecido, e portanto o valor exato para $\sigma_{\hat{p}_{MC}}^2$ não pode ser determinado, na prática usa-se a própria estimativa \hat{p}_{MC} para obter uma aproximação para $\sigma_{\hat{p}_{MC}}^2$. A possibilidade de obter a variância do estimador após a simulação é outra vantagem do método de Monte Carlo, pois pode-se ter uma medida da precisão da estimação. Além disso, como visto no Capítulo 2, a variância é utilizada para obter o intervalo de confiança e o erro normalizado da estimativa encontrada.

Outra forma de obter a média e a variância do estimador de Monte Carlo é utilizando o fato desse estimador ser uma variável aleatória com distribuição binomial com parâmetros N e p_e . A média e a variância dessa distribuição são bem conhecidas na literatura. Para utilizar esses resultados no estimador de Monte Carlo é preciso resolver a divisão por N . Seja $Z = \sum_{i=1}^N H(g(X_i))$ uma variável aleatória com distribuição binomial com parâmetros N e p_e , a média e variância de Z são dados por [30]:

$$\begin{cases} E[Z] = N \cdot p_e \\ \sigma_Z^2 = N \cdot p_e(1 - p_e). \end{cases}$$

Sendo $W = Z/N$, a média e variância de W são dados por:

$$E[W] = E[Z/N] = \frac{1}{N}E[Z] = p_e \quad (3.6)$$

$$\begin{aligned} \sigma_W^2 &= E[(W - E[W])^2] = E\left[\left(\frac{Z}{N} - \frac{E[Z]}{N}\right)^2\right] \\ &= \frac{1}{N^2} \cdot E\{(Z - E[Z])^2\} \\ &= \sigma_Z^2/N^2 = \frac{p_e(1 - p_e)}{N} \end{aligned} \quad (3.7)$$

que são os mesmos resultados obtidos pelas equações 3.4 e 3.5. Para uso futuro, pode-se expressar a variância do método de Monte Carlo, Equação 3.5, na seguinte forma

²Esse resultado pode ser generalizado para o estimador da média 2.4.

$$\begin{aligned}\sigma_{\hat{p}_{MC}}^2 &= \left[\int_{-\infty}^{\infty} H(g(\mathbf{x})) f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} \right] \cdot \frac{(1-p_e)}{N} \\ &= \frac{1}{N} \int_{-\infty}^{\infty} H(g(\mathbf{x})) f_{\mathbf{X}}(\mathbf{x}) (1-p_e) dx\end{aligned}\quad (3.8)$$

em que foi feito uso da definição de p_e , Equação 3.1. Na próxima seção será discutido como obter o intervalo de confiança em uma simulação de Monte Carlo.

3.3 Intervalo de Confiança

O método de Monte Carlo é um dos poucos casos em que é possível obter a distribuição exata do estimador. Como visto na Seção 3.1, a distribuição do estimador de Monte Carlo é binomial, e portanto pode-se obter uma equação fechada para o intervalo de confiança [1]. Essa abordagem, embora completamente rigorosa, tem a desvantagem que as equações relacionadas devem ser resolvidas iterativamente [3]. Felizmente, para os casos de interesse os valores de p_e e N encontrados permitem aproximar a distribuição binomial pela distribuição normal.

O intervalo de confiança para o valor médio de uma fdp gaussiana com média μ e variância σ^2/N foi descrito na seção 2.7 e resultou na equação

$$P \left[-z_\alpha < \frac{\bar{X} - \mu}{\sigma/\sqrt{N}} < +z_\alpha \right] = 1 - \alpha \quad (3.9)$$

em que \bar{X} é a estimativa da média, $(1 - \alpha)$ é o nível de confiança, e z_α é um valor obtido a partir da gaussiana normalizada e é dado por

$$\int_{-z_\alpha}^{z_\alpha} \frac{1}{\sqrt{2\pi}} \exp^{-\frac{x^2}{2}} dx = 1 - \alpha, \quad (3.10)$$

com o objetivo de obter um intervalo de confiança simétrico. Quando $N \rightarrow \infty$, o estimador \hat{p}_{MC} tende para uma distribuição normal com média p_e e variância $p_e(1 - p_e)/N$. Aplicando esses resultados em 3.9, e substituindo o estimador \bar{X} por \hat{p}_{MC} , tem-se

$$P \left[-z_\alpha \leq \frac{\hat{p}_{MC} - p_e}{\sqrt{p_e(1 - p_e)/N}} \leq +z_\alpha \right] = 1 - \alpha. \quad (3.11)$$

Como o argumento na equação acima envolve o valor desconhecido p_e , é necessário isolar esse valor para obter os limites do intervalo de confiança. Elevando os termos ao quadrado, as desigualdades resumem-se a

$$\left(\frac{\hat{p}_{MC} - p_e}{\sqrt{p_e(1-p_e)/N}} \right)^2 \leq z_\alpha^2. \quad (3.12)$$

Isolando p_e na desigualdade acima, resulta

$$(N + z_\alpha^2)p_e^2 - (2N\hat{p}_{MC} + z_\alpha^2)p_e + N\hat{p}_{MC}^2 \leq 0. \quad (3.13)$$

Como 3.13 é quadrática em p_e , pode-se encontrar as raízes dessa equação. Esse cálculo resulta nos valores L e U para o intervalo de confiança,

$$\begin{aligned} P \left\{ \frac{N}{N + z_\alpha^2} \left[\hat{p}_{MC} + \frac{z_\alpha^2}{2N} - z_\alpha \left(\frac{\hat{p}_{MC}(1 - \hat{p}_{MC})}{N} + \frac{z_\alpha^2}{2N} \right)^{1/2} \right] \right. \\ \left. \leq p_e \leq \frac{N}{N + z_\alpha^2} \left[\hat{p}_{MC} + \frac{z_\alpha^2}{2N} + z_\alpha \left(\frac{\hat{p}_{MC}(1 - \hat{p}_{MC})}{N} + \frac{z_\alpha^2}{2N} \right)^{1/2} \right] \right\} \\ = 1 - \alpha \end{aligned} \quad (3.14)$$

em que p_e é o valor verdadeiro da probabilidade de erro. Além da vantagem de obter facilmente tabelas para a distribuição gaussiana normalizada, o intervalo de confiança para a gaussiana pode ser utilizado para criar uma representação gráfica generalizada. Fazendo $\hat{p}_{MC} = 10^{-\nu}$ e $N = \eta 10^\nu$, e considerando as aproximações $N/(N + z_\alpha^2) \cong 1$ e $\hat{p}_{MC}(1 - \hat{p}_{MC}) = \hat{p}_{MC}$, que são mais que amplamente satisfeitas, pode-se expressar 3.14 como³

$$P[y_- \leq p_e \leq y_+] = 1 - \alpha$$

em que o intervalo de confiança (y_-, y_+) é dado por

$$y_\pm = 10^{-\nu} \{1 + (z_\alpha^2/2\eta)[1 \pm (4\eta/z_\alpha^2 + 1)^{1/2}]\}. \quad (3.15)$$

³Note que \hat{p}_{MC} não pode ser zero, para este caso usa-se a aproximação de Poisson.

Este intervalo é plotado na Figura 3.1 para níveis de confiança de 90%, 95% e 99%. Uma regra popular para estimação é utilizar $N = 10/p_e$. Esse valor coincide com a linha vertical em $N = 10^{\nu+1}$, que produz um intervalo com 95% de confiança de aproximadamente $(0.55\hat{p}_{MC}, 1.8\hat{p}_{MC})$, isto é, um fator de aproximadamente 2 sobre a probabilidade de erro, que é uma incerteza considerada tolerável. Incrementando N por um fator de 10, ou seja, $N = 10^{\nu+2}$, a Figura 3.1 mostra um intervalo de confiança de aproximadamente $(0.8\hat{p}_{MC}, 1.25\hat{p}_{MC})$.

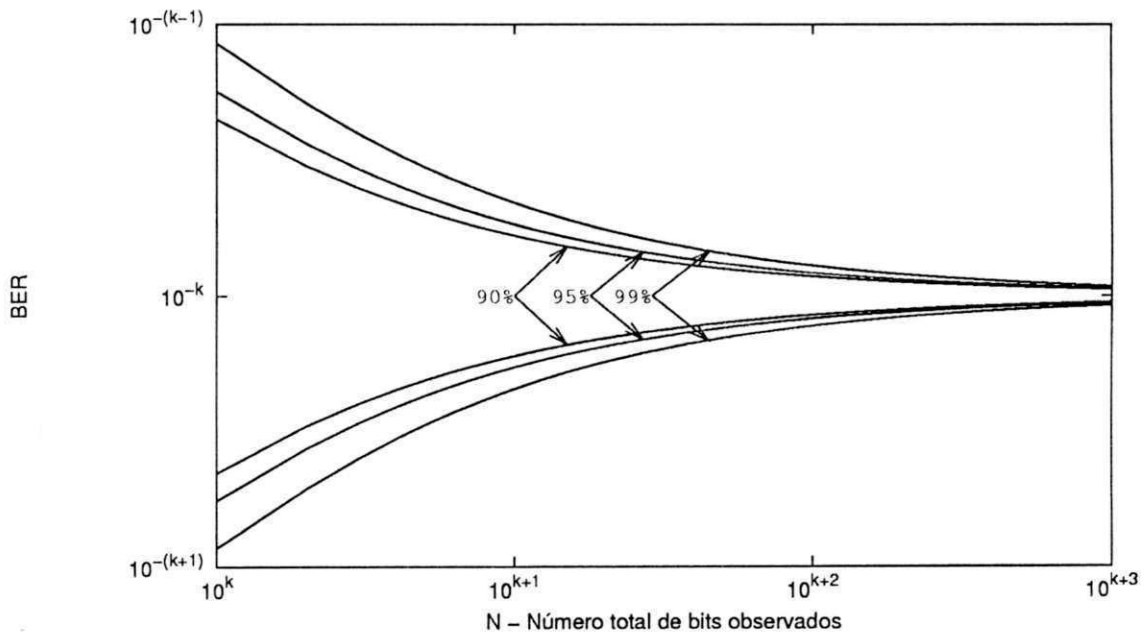


Figura 3.1: Intervalo de confiança quando o valor observado é 10^{-k} .

Outra forma bastante utilizada para encontrar o intervalo de confiança, a partir da aproximação pela normal, é simplesmente calcular as estimativas do valor médio \hat{p}_e e da variância σ_{MC}^2 , e aplicar esses valores na Equação 2.26 resultando em

$$\begin{cases} L = \hat{p}_{MC} - \frac{\sigma_{MC}}{\sqrt{N}} z_{\alpha} \\ U = \hat{p}_{MC} + \frac{\sigma_{MC}}{\sqrt{N}} z_{\alpha} \end{cases} \quad (3.16)$$

em que z_{α} é obtido utilizando a Equação 3.10 para o coeficiente de confiança especifi-

cado.

O fato do intervalo de confiança poder ser calculado no decorrer da simulação juntamente com \hat{p}_{MC} possibilita o uso de uma regra de parada. Em outras palavras, ao invés de usar um valor pré-definido para N , a simulação pode ser parada quando o intervalo de confiança atinge um tamanho satisfatório.

Uma terceira forma de obter o intervalo de confiança do estimador de Monte Carlo é utilizando a desigualdade de Chebyshev [20], entretanto, esse intervalo contém o intervalo dado por 3.16, e portanto tem um tamanho maior, o que é indesejável.

Na próxima seção será exposto um exemplo em que o método de Monte Carlo é aplicado. Além das estimativas da probabilidade de erro para diversos valores da relação sinal-ruído (SNR), também são obtidos a variância do estimador, o erro normalizado e o intervalo de confiança com aproximação pela normal.

3.4 Simulação de um Sistema de Comunicações com Modulação Binária

Nessa seção será ilustrada a aplicação do método de Monte Carlo em um sistema binário simples, em que a probabilidade de erro de bit (BER) pode ser encontrada analiticamente. São obtidas estimativas da BER para diferentes valores da relação sinal-ruído (SNR), e comparadas com os valores reais. A variância do estimador e o intervalo de confiança também foram obtidos durante a simulação e serão apresentados aqui.

Em um sistema com modulação BPSK, a cada intervalo de sinalização T_s apenas um bit b_k é gerado pela fonte binária e transmitido através do canal. O modulador BPSK (*binary phase shift keying*) modifica a fase de uma portadora senoidal de acordo com o bit b_k a ser transmitido. O sinal resultante é dado por

$$s(t) = A_c \sum_k \cos(2\pi f_c t + \phi_k) p(t - k \cdot T_s) \quad (3.17)$$

em que A_c e f_c são a amplitude e a frequência da portadora, respectivamente, $T_s \gg 1/f_c$, e

$$p(t) = \begin{cases} 1, & 0 < t < T_s \\ 0, & \text{caso contrário.} \end{cases} \quad (3.18)$$

A variável ϕ_k em 3.17 é a fase que modifica a portadora e que leva a informação. Como a modulação é binária, a fase ϕ_k pode ter apenas os valores $\{0, \pi\}$. Se o bit a ser transmitido é “0”, a portadora é modulada com fase $\phi_k = 0$, caso contrário a fase será $\phi_k = \pi$. Portanto, a cada intervalo de sinalização um dos seguintes sinais é transmitido pelo sistema,

$$s_0(t) = A_c \cos(2\pi f_c t) p(t) \quad (3.19)$$

ou

$$\begin{aligned} s_1(t) &= A_c \cos(2\pi f_c t + \pi) p(t) \\ &= -A_c \cos(2\pi f_c t) p(t) \\ &= -s_0(t), \end{aligned} \quad (3.20)$$

para a transmissão de um bit “0” ou “1”, respectivamente. Desde que $s_1(t) = -s_0(t)$, esses sinais são ditos antipodais. A energia por bit transmitido ξ_b é dada por,

$$\xi_b = \int_0^{T_s} s_0^2(t) dt = \int_0^{T_s} s_1^2(t) dt. \quad (3.21)$$

A Figura 3.2 mostra a representação geométrica dos sinais [29], em que $s_0 = \sqrt{\xi_b}$ e $s_1 = -\sqrt{\xi_b}$. O canal é modelado como um ruído aditivo gaussiano branco (AWGN) com média nula e variância $\sigma_N^2 = N_0/2$.

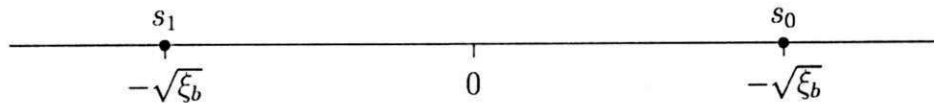


Figura 3.2: Representação geométrica dos sinais transmitidos no sistema BPSK.

O demodulador multiplica o sinal recebido $x(t) = s(t) + n(t)$ por um sinal similar a portadora, e realiza uma operação de integração para cada bit transmitido. O receptor é considerado ter perfeito sincronismo com o transmissor. Com essas operações, o sinal r_k recebido pelo decisor para cada bit b_k transmitido é

$$r_k = \begin{cases} \sqrt{\xi_b} + n_k, & \text{se } b_k = 0 \\ -\sqrt{\xi_b} + n_k, & \text{se } b_k = 1 \end{cases} \quad (3.22)$$

em que n_k representa a componente do ruído aditivo gaussiano. Nesse caso, a regra de decisão utilizada é comparar r com o limiar zero. Se $r > 0$, a decisão é feita em favor de s_0 , e se $r < 0$, a decisão é feita em favor de s_1 . Claramente, as duas fdp's condicionadas de r são⁴:

$$p(r/s_0) = \frac{1}{\sqrt{\pi N_0}} e^{-(r-\sqrt{\xi_b})^2/N_0} \quad (3.23)$$

$$p(r/s_1) = \frac{1}{\sqrt{\pi N_0}} e^{-(r+\sqrt{\xi_b})^2/N_0}. \quad (3.24)$$

Essas fdp's condicionadas são mostradas na Figura 3.3. Utilizando a regra de decisão estabelecida, as probabilidades de erro de bit são obtidas integrando as fdp's 3.23 e 3.24 nas regiões em que há erro de decisão. Dado que $s_0(t)$ foi transmitido, a probabilidade de erro é simplesmente a probabilidade que $r < 0$, isto é

$$\begin{aligned} p(e/s_0) &= \int_{-\infty}^0 p(r/s_0) dr \\ &= \frac{1}{\sqrt{\pi N_0}} \int_{-\infty}^0 e^{-(r-\sqrt{\xi_b})^2/N_0} dr \\ &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{-\sqrt{2\xi_b/N_0}} e^{-x^2/2} dx \\ &= \frac{1}{\sqrt{2\pi}} \int_{\sqrt{2\xi_b/N_0}}^{\infty} e^{-x^2/2} dx \\ &= Q(\sqrt{2\xi_b/N_0}) \end{aligned} \quad (3.25)$$

em que $Q(\cdot)$ é a função erro definida na Equação 1.11. Fazendo esse mesmo procedimento para s_1 , obtém-se o mesmo resultado, ou seja, $p(e/s_1) = Q(\sqrt{2\xi_b/N_0})$. Supondo

⁴Distribuições gaussianas com médias $\sqrt{\xi_b}$ e $-\sqrt{\xi_b}$, respectivamente.

que os sinais $s_0(t)$ e $s_1(t)$ são igualmente prováveis de serem transmitidos, a probabilidade de erro de bit p_b é dada por

$$\begin{aligned} p_b &= \frac{1}{2}p(e/s_0) + \frac{1}{2}p(e/s_1) \\ &= Q(\sqrt{2\xi_b/N_0}). \end{aligned} \quad (3.26)$$

A razão ξ_b/N_0 é conhecida como a relação sinal-ruído por bit. A Figura 3.4 mostra a curva da probabilidade de erro de bit p_e versus ξ_b/N_0 .

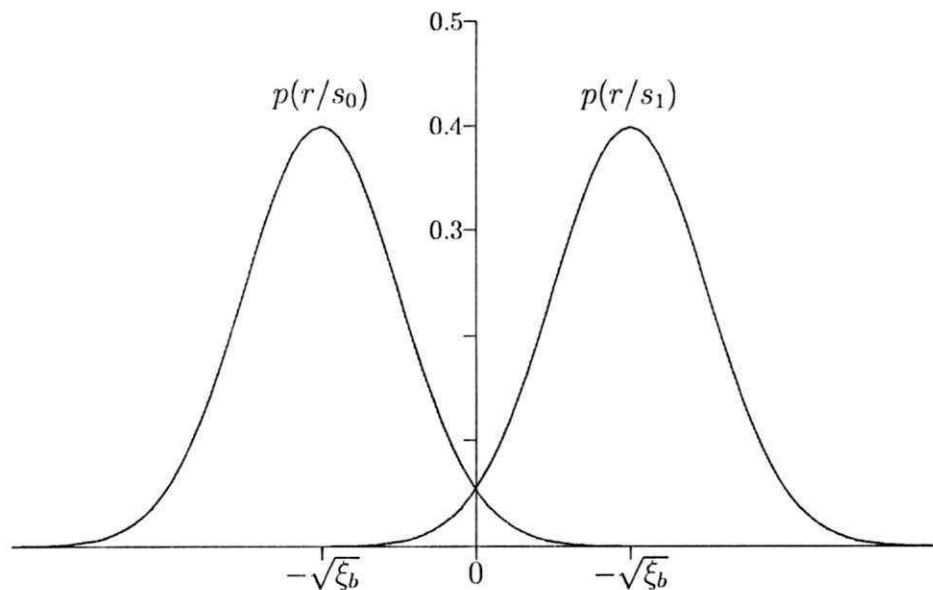


Figura 3.3: Distribuição de probabilidade do sinal na entrada do receptor BPSK.

A implementação do método de Monte Carlo para esse sistema é bastante simples, em que o estimador calcula uma estimativa \hat{p}_{MC} de p_e para cada valor $\text{SNR} = \xi_b/N_0$.

Para simplificar a implementação, a distribuição gaussiana é normalizada, ou seja, $\sigma_N^2 = N_0/2 = 1$, e conseqüentemente $N_0 = 2$. Feito isso, calcula-se ξ_b usando o valor da relação sinal ruído,

$$\xi_b = N_0 \cdot \text{SNR} = 2 \cdot \text{SNR}.$$

Como mostra a Equação 3.22, o sinal na entrada do decisor é a soma da componente do sinal transmitido (s_0 ou s_1) com a componente do ruído n_k . Como dito anteriormente, o método de Monte Carlo baseia-se em utilizar geradores de números aleatórios

para essas quantidades e contar o número de erros ocorridos, conforme a definição do método na Equação 3.2.

A implementação então utiliza um gerador de bits aleatórios e mapeia um bit b_k em um dos valores s_0 e s_1 , e também obtém a amostra n_k do ruído utilizando um gerador de amostras de uma distribuição gaussiana normalizada⁵.

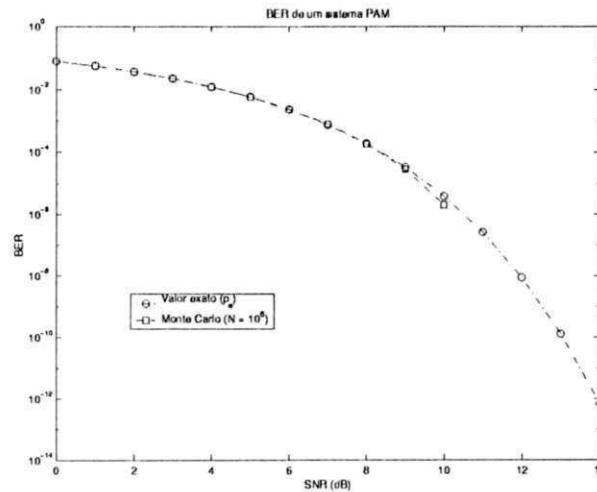


Figura 3.4: Simulação de Monte Carlo para o sistema BPSK sobre um canal AWGN.

Essas quantidades são somadas para obter a amostra r_k na entrada do decisor. Daí, faz-se o processo de decisão com limiar nulo e compara-se o bit recebido \hat{b}_k com o bit transmitido b_k . Se um erro ocorrer o contador de erros é incrementado. Esse processo ocorre N vezes. Por fim, a estimativa \hat{p}_{MC} de p_e é calculada pela proporção do total de erros em relação a N , como estabelece a Equação 3.2.

A Tabela 3.1 mostra os resultados obtidos estimando p_b utilizando o método de Monte Carlo para diversos valores da relação sinal-ruído por bit (ξ_b/N_0). Nessa mesma tabela é colocado o valor real de p_b , a variância $\hat{\sigma}_{MC}^2$ do estimador e o erro normalizado. Na Figura 3.4 tem-se o gráfico dos valores de \hat{p}_{MC} e p_b .

Os resultados listados na Tabela 3.1 foram obtidos com simulações de comprimento $N = 10^6$. Como pode-se notar, quando a probabilidade de erro atinge valores da ordem de 10^{-6} o número de erros encontrados na simulação não é suficiente para estimar p_e , e portanto o erro normalizado é muito alto. Esse mesmo resultado pode ser visto na Figura 3.4, em que a curva obtida pelo estimador de Monte Carlo para $N = 10^6$ só

⁵Média nula e variância unitária.

tem valores até a relação sinal ruído de 10dB ($p_e = 3,870 \cdot 10^{-06}$), a partir desse ponto o comprimento da simulação é insuficiente.

Tabela 3.1: Resultados da simulação de Monte Carlo para o sistema BPSK.

SNR	P_b	\hat{p}_{MC}	$\hat{\sigma}_{MC}^2$	Erro normalizado
0	$7,860 \cdot 10^{-02}$	$7,873 \cdot 10^{-02}$	$7,25 \cdot 10^{-08}$	0,34%
1	$5,630 \cdot 10^{-02}$	$5,652 \cdot 10^{-02}$	$5,33 \cdot 10^{-08}$	0,41%
2	$3,750 \cdot 10^{-02}$	$3,783 \cdot 10^{-02}$	$3,64 \cdot 10^{-08}$	0,51%
3	$2,290 \cdot 10^{-02}$	$2,284 \cdot 10^{-02}$	$2,23 \cdot 10^{-08}$	0,65%
4	$1,250 \cdot 10^{-02}$	$1,266 \cdot 10^{-02}$	$1,25 \cdot 10^{-08}$	0,89%
5	$6,000 \cdot 10^{-03}$	$5,900 \cdot 10^{-03}$	$5,87 \cdot 10^{-09}$	1,28%
6	$2,400 \cdot 10^{-03}$	$2,301 \cdot 10^{-03}$	$2,30 \cdot 10^{-09}$	2,00%
7	$7,730 \cdot 10^{-04}$	$8,300 \cdot 10^{-04}$	$8,29 \cdot 10^{-10}$	3,73%
8	$1,910 \cdot 10^{-04}$	$1,870 \cdot 10^{-04}$	$1,87 \cdot 10^{-10}$	7,16%
9	$3,360 \cdot 10^{-05}$	$2,900 \cdot 10^{-05}$	$2,90 \cdot 10^{-11}$	16,03%
10	$3,870 \cdot 10^{-06}$	$2,000 \cdot 10^{-06}$	$2,00 \cdot 10^{-12}$	36,54%

3.5 Conclusões

O método de Monte Carlo é atualmente a abordagem mais utilizada para estimação de desempenho em comunicações. Os principais motivos são a generalidade do método, o surgimento de modelos mais detalhados dos canais de comunicação e a evolução dos processadores acelerando as simulações. Ao implementar o método percebe-se outra vantagem que é a simplicidade, pois conhecendo o modelo do sistema que se deseja simular, a implementação é simples e há diversos algoritmos na literatura que podem ser reutilizados.

Analisando o método de Monte Carlo do ponto de vista de estimação, foi demonstrado que o estimador de Monte Carlo é despolarizado e consistente, e que as estimativas obtidas têm distribuição binomial. O conhecimento da estatística do estimador possibilita o cálculo do desempenho do método, que é mais uma vantagem do método de Monte Carlo. Na análise do sistema utilizando estimação é fundamental conhecer a estatística do erro cometido, essa informação pode ser obtida calculando a variância, o erro normalizado, ou o intervalo de confiança do estimador de Monte Carlo.

Embora as vantagens acima sejam importantes no contexto de simulação, o método

de Monte Carlo é impeditivo quando a probabilidade de erro é muito pequena, pois são necessárias muitas iterações para que um erro ocorra. Para sobrepor esse problema são aplicadas técnicas conhecidas como métodos de redução de variância, que se utilizados podem reduzir a variância do estimador de Monte Carlo, ou equivalentemente reduzir o necessário número de iterações para obter uma estimativa aceitável. No próximo capítulo será apresentada uma simulação de Monte Carlo modificada que utiliza uma técnica de redução de variância para melhorar o desempenho da simulação.

Capítulo 4

Monte Carlo com Valorização das Amostras

O método de Monte Carlo torna-se ineficiente na medida em que a probabilidade de erro a ser estimada diminui. Isso ocorre pois será necessário um grande número de iterações para causar um único erro, o qual é o evento importante na simulação.

Como ilustração, se o ruído é gaussiano com média nula, o valor mais provável é zero, e não induzirá nenhum erro. Possivelmente, alguma faixa de valores do ruído próximos de zero também não causará erros na transmissão. Por exemplo, para uma probabilidade de erro de 10^{-6} , na média apenas um evento de erro ocorre a cada 10^6 iterações. Assim, na maioria do tempo os eventos obtidos na simulação não são importantes. A consequência disso é que as iterações em que o ruído não pode causar erros tornam-se lixo computacional por trazerem pouca informação ao estimador.

Estimar a probabilidade de erro da ordem de p_e , com um erro normalizado menor que $1/3$, requer uma simulação de Monte Carlo da ordem de $10/p_e$. Um erro normalizado de $1/3$ corresponde a um intervalo de confiança $[\frac{1}{3}p_e, 1\frac{2}{3}p_e]$ para um nível de confiança de 95%. Com uma probabilidade de erro típica de, por exemplo, 10^{-8} , o método de Monte Carlo necessitaria de 10^9 amostras por simulação, o que é impraticável.

Em linhas gerais essa é a motivação para o método de Monte Carlo modificado, conhecido como **método de Monte Carlo com valorização das amostras** (IS - *Importance Sampling*). Esse método propõe modificar a fdp do ruído na direção de gerar erros artificialmente, que são os eventos importantes na simulação. Com essa modificação espera-se obter mais rápido a informação necessária para estimar a probabilidade de erro com a precisão desejada.

A alteração da fdp do ruído para gerar mais erros é feita de forma controlada e é chamada de polarização. Como consequência o contador de erros não pode ser o mesmo. Entretanto, conhecendo a polarização associada com cada amostra do ruído, é possível despolarizar adequadamente o contador.

A escolha de uma boa fdp polarizada para o ruído, visando melhorar a eficiência do método de Monte Carlo, é uma das tarefas mais importantes na aplicação do método de Valorização das Amostras. Usar uma polarização ruim torna o estimador menos eficiente que Monte Carlo.

Após escolher a família da fdp polarizada, é necessário encontrar os parâmetros ótimos para a simulação. Métodos para otimização de parâmetros são de particular interesse no presente trabalho e serão descritos no Capítulo 5.

A vantagem de utilizar valorização das amostras em relação ao método de Monte Carlo convencional pode ser vista de duas formas. Inicialmente como uma diminuição do custo computacional, mantendo a mesma precisão, ou equivalentemente, um aumento na precisão para o mesmo custo computacional.

Valorização das amostras pertence a um conjunto de técnicas conhecidas como métodos para redução de variância. O primeiro artigo aplicando o método de Monte Carlo Modificado em comunicações foi publicado por K.S. Shanmugan e P. Balaban [2].

Como anteriormente, a fdp deve ser polarizada visando aumentar o número de erros encontrados na simulação. Em sistemas complexos, a região da fdp do ruído em que são gerados erros no sistema não é conhecida. Caso essa região pudesse ser encontrada o problema de calcular a probabilidade de erro seria resumido ao cálculo de uma integral definida de uma função determinística¹ em uma região conhecida. Por outro lado, o problema de encontrar essa região pode ser tão complexo quanto o cálculo de p_e .

Outro fato importante relacionado à fdp polarizada é que as amostras do ruído são obtidas a partir de algoritmos geradores de números aleatórios, e portanto é preciso escolher uma fdp que seja possível implementar um algoritmo para obter as amostras aleatórias.

Além da escolha da fdp, a despolarização do estimador pode ser uma tarefa difícil. Em geral, sistemas com memória, em que várias amostras do ruído afetam um único bit, é necessário conhecer a quantidade dessas amostras para fazer a despolarização, em outras palavras, é necessário conhecer a memória do sistema.

¹Essa função refere-se a fdp do ruído.

Nesse capítulo o método de valorização das amostras (IS - *Importance Sampling*) será descrito no contexto do problema da estimação da probabilidade de erro em comunicações. Suas principais propriedades serão investigadas e comparadas com o estimador de Monte Carlo. Por fim, o ganho de desempenho do método é demonstrado em um sistema binário com canal gaussiano branco.

4.1 Definição do Estimador IS

Para desenvolver a abordagem IS, considere o modelo simplificado de um sistema digital mostrado na Figura 4.1.

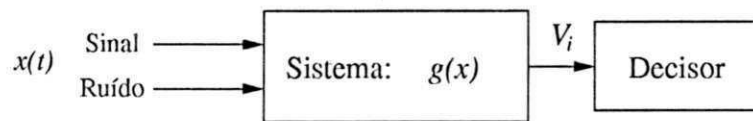


Figura 4.1: Diagrama de blocos do Sistema simplificado.

O processo de entrada $X(t)$ é composto por sinal mais ruído e $V(t)$ é o sinal na entrada do decisor. No processo de simulação $X(t)$ é discretizado no tempo e torna-se o vetor $X = \{X(t_i), X(t_i - T_s), \dots, X(t_i - (M - 1)T_s)\}$, em que T_s é o intervalo entre amostras, Figura 4.2.

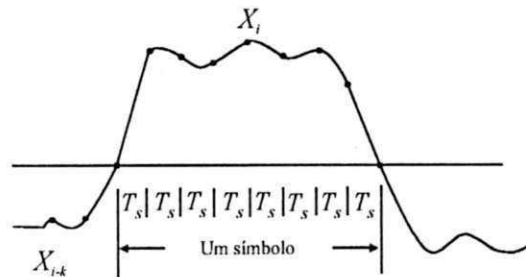


Figura 4.2: Ilustração das amostras de $X(t)$.

A dimensionalidade M do vetor X diz respeito ao número de amostras que impactam no processo de decisão do símbolo transmitido. Para cada símbolo transmitido tem-se T/T_s amostras, em que T é a duração do símbolo. Se o dispositivo decisor necessita de m símbolos para efetuar uma decisão, em que m é a memória do sistema, então são necessárias $M = m \cdot T/T_s$ amostras para realizar a decisão de um símbolo.

Admitindo que as fdp's do sinal e do ruído são conhecidas, a probabilidade de erro é dada por

$$p_e = \int_{-\infty}^{\infty} H(v) f_V(v) dv \quad (4.1)$$

$$= \int_{-\infty}^{\infty} H(g(x)) f_X(x) dx. \quad (4.2)$$

em que $g(\cdot)$ é a característica de transferência do sistema e $f_X(\cdot)$ é a fdp M dimensional de X .

A Equação 4.1 não é utilizada como base para a simulação, pois $f_V(v)$ em geral não é conhecida, e portanto não é possível implementar um gerador de números aleatórios para o vetor V . Caso seja possível, essa formulação é chamada de versão de saída do método de Valorização das Amostras [2]. Se a Equação 4.2 for usada chama-se versão de entrada. Embora a abordagem utilizada a seguir seja essa última, a versão de saída pode ser encontrada de modo similar.

O método consiste em introduzir a fdp polarizada $f_X^*(x)$ que é mais adequada para a simulação. Pode-se reescrever a Equação 4.2 utilizando $f_X^*(x)$ como segue:

$$p_e = \int_{-\infty}^{\infty} H(g(x)) \frac{f_X(x)}{f_X^*(x)} f_X^*(x) dx$$

$$= \int_{-\infty}^{\infty} H(g(x)) w(x) f_X^*(x) dx \quad (4.3)$$

em que $w(x) = f_X(x)/f_X^*(x)$ é chamada de função peso. Embora as Equações 4.2 e 4.3 sejam idênticas, a essência da polarização é que a estatística dos processos do ruído são modificados de modo que $f_X^*(x)$ não é meramente um artifício matemático, como poderia parecer na Equação 4.3, mas trata-se de uma mudança na distribuição de X para aumentar proporcionalmente o número de erros na simulação. A Equação 4.3 também pode ser escrita como

$$p_e = E_*[H(g(x))w(x)] \quad (4.4)$$

em que $E_*[\cdot]$ é a esperança com respeito a $f_X^*(x)$. Como anteriormente, o estimador para a probabilidade de erro é obtido a partir da Equação 4.4 como a média amostral

$$\hat{p}_{IS} = \frac{1}{N} \sum_{i=1}^N H(g(x_i))w(x_i). \quad (4.5)$$

No estimador de Monte Carlo, a contagem de erros é feita com incrementos de um, por outro lado, o estimador modificado pondera cada erro somando $w(x_i)$. A função peso $w(x) = f(x)/f^*(x)$ tem a função de corrigir o contador de erros devido à polarização do ruído.

Note que o valor N é usado em 3.2 e 4.5 como um símbolo genérico para o número total de símbolos transmitidos, entretanto, os valores específicos não precisam ser os mesmos. Na próxima seção será demonstrado que o estimador com valorização das amostras é despolarizado e consistente.

4.2 Despolarização e Consistência

O estimador modificado não é polarizado, e isso pode ser facilmente mostrado. O valor esperado do estimador pode ser escrito como

$$\begin{aligned} E[\hat{p}_{IS}] &= E_* \left[\frac{1}{N} \sum_{i=1}^N H(g(x_i))w(x_i) \right] \\ &= \frac{1}{N} \sum_{i=1}^N E_*[H(g(x_i))w(x_i)]. \end{aligned}$$

As diferentes amostras x_i são identicamente distribuídas, portanto

$$\begin{aligned} E[\hat{p}_{IS}] &= \frac{1}{N} \cdot N \cdot E[H(g(x))w(x)] \\ &= \int_{-\infty}^{+\infty} H(g(X))w(X)f^*(x)dx \\ &= \int_{-\infty}^{+\infty} H(g(X))\frac{f_X(x)}{f_X^*(x)}f^*(x)dx \\ &= p_e. \end{aligned} \quad (4.6)$$

Esse último resultado demonstra que a polarização da distribuição de X não afetou a propriedade de despolarização do método de Monte Carlo. Como proposto inicialmente, a valorização das amostras, se feita adequadamente, reduz a variância do estimador.

Para demonstrar que o estimador modificado é mais eficiente que o estimador de Monte Carlo, é necessário comparar suas variâncias.

A variância do estimador de Monte Carlo foi obtida na Equação 3.5. Seguindo o mesmo procedimento para o método modificado, tem-se

$$\begin{aligned}\sigma_{IS}^2 &= E_*[(\hat{p}_{IS} - E[\hat{p}_{IS}])^2] \\ &= E_*[(\hat{p}_{IS} - p_e)^2] \\ &= E_*[\hat{p}_{IS}^2] - p_e^2\end{aligned}\tag{4.7}$$

em que foram utilizados os resultados obtidos nas Equações 2.10 e 4.6. Substituindo a definição de \hat{p}_{IS} na Equação 4.7, tem-se

$$\begin{aligned}\sigma_{IS}^2 &= E_* \left[\left(\frac{1}{N} \sum_{i=1}^N H(g(x_i))w(x_i) \right) \cdot \left(\frac{1}{N} \sum_{j=1}^N H(g(x_j))w(x_j) \right) \right] - p_e^2 \\ &= \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N E_* [H(g(x_i))w(x_i)H(g(x_j))w(x_j)] - p_e^2.\end{aligned}\tag{4.8}$$

Separando os termos em que $i = j$, e admitindo independência entre os vetores aleatórios x_i e x_j , para $i \neq j$, tem-se

$$\begin{aligned}\sigma_{IS}^2 &= \frac{1}{N^2} \sum_{i=1}^N E_* [H^2(g(x_i))w^2(x_i)] \\ &\quad + \frac{2}{N^2} \sum_{i=1}^{N-1} \sum_{j=i+1}^N E_* [H(g(x_i))w(x_i)H(g(x_j))w(x_j)] - p_e^2 \\ &= \frac{1}{N^2} \sum_{i=1}^N \int_{-\infty}^{+\infty} H^2(g(x_i))w^2(x_i)f_X^*(x_i)dx_i \\ &\quad + \frac{2}{N^2} \sum_{i=1}^{N-1} \sum_{j=i+1}^N E_* [H(g(x_i))w(x_i)] E_* [H(g(x_j))w(x_j)] - p_e^2 \\ &= \frac{1}{N^2} \sum_{i=1}^N \int_{-\infty}^{+\infty} H(g(x_i))w(x_i) \frac{f_X(x_i)}{f_X^*(x_i)} f_X^*(x_i)dx_i + \frac{2}{N^2} \sum_{i=1}^{N-1} \sum_{j=i+1}^N (p_e \cdot p_e) - p_e^2\end{aligned}$$

em que foi utilizada a igualdade $H^2(g(x_i)) = H(g(x_i))$, a definição da função peso $w(x)$ e a definição de p_e escrita como o valor esperado, Equação 4.4. Retirando as expressões dos somatórios, tem-se

$$\begin{aligned}\sigma_{IS}^2 &= \frac{N}{N^2} \int_{-\infty}^{+\infty} H(g(x))w(x)f_X(x)dx + \frac{2}{N^2} \frac{N(N-1)p_e^2}{2} - p_e^2 \\ &= \frac{1}{N} \int_{-\infty}^{+\infty} H(g(x))w(x)f_X(x)dx - \frac{p_e^2}{N}.\end{aligned}\quad (4.9)$$

Similarmente ao cálculo da probabilidade de erro, a variância do estimador com valorização das amostras pode ser escrita em termos do operador valor esperado

$$\sigma_{IS}^2 = \frac{1}{N} \{E[H(g(x))w(x)] - p_e^2\}.\quad (4.10)$$

Durante a simulação é necessário estimar a precisão da estimativa obtida com a valorização das amostras, isso é feito usando o estimador da variância amostral dado por

$$\hat{\sigma}_{IS}^2 = \frac{1}{N-1} \left\{ \frac{1}{N} \sum_{k=1}^N H(g(x_k))w^2(x_k) - \hat{p}_{IS}^2 \right\}.\quad (4.11)$$

É desejável escrever 4.9 em uma forma mais adequada para análise futura. Substituindo um dos operandos de p_e pela Equação 4.2, tem-se

$$\begin{aligned}\sigma_{IS}^2 &= \frac{1}{N} \int_{-\infty}^{+\infty} H(g(x))w(x)f_X(x)dx - \frac{p_e}{N} \int_{-\infty}^{+\infty} H(g(x))f_X(x)dx \\ &= \frac{1}{N} \int_{-\infty}^{+\infty} H(g(x))f_X(x)[w(x) - p_e]dx.\end{aligned}\quad (4.12)$$

A possibilidade de redução de variância é evidente comparando as Equações 3.8 e 4.12. Para tanto, é necessário que $w(x) < 1$ nas regiões em que x resulta em erro ($H(g(x)) = 1$). Como $w(x) = f_X(x)/f_X^*(x)$, fazer $w(x) < 1$ significa escolher $f_X^*(x) > f_X(x)$ nas regiões em que há erros. Essa propriedade desejada em $f_X^*(x)$ implica no aumento de ocorrências de erros no processo de simulação, como mencionado anteriormente.

Outra informação importante a ser extraída da Equação 4.12 é a possibilidade de reduzir a variância a zero fazendo $w(x) = p_e$ para todos valores de x na região onde

ocorrem os erros. Isso implica no estimador perfeito, entretanto, como o valor de p_e é desconhecido, tal polarização não faz sentido no contexto de simulação. Embora na prática não seja possível realizar o estimador perfeito, alguma informação útil pode ser extraída.

Finalmente, como a integral em 4.12 não depende de N , isto implica que o estimador é consistente, pois se $N \rightarrow \infty$, então $\sigma_{JS}^2 \rightarrow 0$. Com isso, as condições de consistência são satisfeitas pelo estimador.

4.3 Escolha da distribuição Polarizada

Até o momento foi apresentada a definição do estimador com valorização das amostras, e foi mostrado que o estimador é despolarizado e consistente. No processo de modelagem para implementar o método em um sistema específico, a única quantidade sujeita a escolha é a distribuição polarizada $f_{\mathbf{X}}^*(\cdot)$, que até então foi tratada apenas simbolicamente.

A distribuição polarizada deve ser escolhida tal que a variância do estimador seja tão pequena quanto possível, e possa ser facilmente implementada utilizando geradores de números aleatórios. Na maioria das aplicações apenas a distribuição do ruído é polarizada, embora em algumas situações seja possível polarizar o sinal transmitido e obter algum ganho de desempenho, tais casos não são estudados nesse trabalho.

É possível obter analiticamente a distribuição polarizada ótima [2] [5], entretanto, a fdp obtida depende da probabilidade de erro de bit desconhecida p_b . Como obter essa quantidade é o objetivo da estimação, essa solução não é realizável na prática. Na próxima seção serão mostradas duas abordagens diferentes para polarizar a distribuição gaussiana.

4.3.1 Polarização da Distribuição Gaussiana

Na modelagem do canal assume-se que a fdp do ruído $f_N(\cdot)$ é uma distribuição gaussiana com média nula e variância σ_N^2 , e é dada por

$$f_N(n) = \frac{1}{\sqrt{2\pi}\sigma_N} e^{-n^2/2\sigma_N^2}. \quad (4.13)$$

Desde que a distribuição gaussiana tem dois parâmetros, média e variância, a densidade polarizada $f_N^*(\cdot)$ pode ser diferente da densidade original apenas modificando esses

parâmetros. Historicamente, o primeiro esquema sugerido para demonstrar o ganho de desempenho obtido com a valorização das amostras foi aplicado à distribuição gaussiana e é conhecido como **valorização das amostras convencional** (*Conventional Importance Sampling - CIS*). Essa abordagem baseia-se em aumentar a proporção de erros na simulação apenas incrementando a variância σ_N^2 da distribuição para o valor $\sigma_{N^*}^2$, como mostra a Figura 4.3. Com essa abordagem a distribuição polarizada é escrita como

$$\begin{aligned} f_N^*(n) &= \frac{1}{\sqrt{2\pi}\sigma_{N^*}} e^{-n^2/2\sigma_{N^*}^2} \\ &= \frac{1}{\sqrt{2\pi} \cdot [\sigma_N/\sqrt{1-\alpha}]} e^{-\frac{n^2}{2\sigma_N^2/(1-\alpha)}}, \end{aligned} \quad (4.14)$$

que é uma distribuição gaussiana com média nula e variância $\sigma_{N^*}^2 = \sigma_N^2/(1-\alpha)$. Para $0 \leq \alpha < 1$ nota-se que $\sigma_{N^*}^2 \geq \sigma_N^2$, com igualdade para $\alpha = 0$ em que a fdp polarizada reduz-se a fdp original. Para $0 < \alpha < 1$, essa modificação garante um aumento na variância da fdp. Supondo um sistema em que a calda da distribuição do ruído provoca os eventos importantes (erros) na simulação, como ilustra a Figura 4.3, o aumento da variância da fdp induzirá a ocorrência dos erros mais frequentemente. O problema de implementar CIS, portanto, é selecionar a variância $\sigma_{N^*}^2$ da distribuição polarizada, ou equivalentemente escolher a constante α .

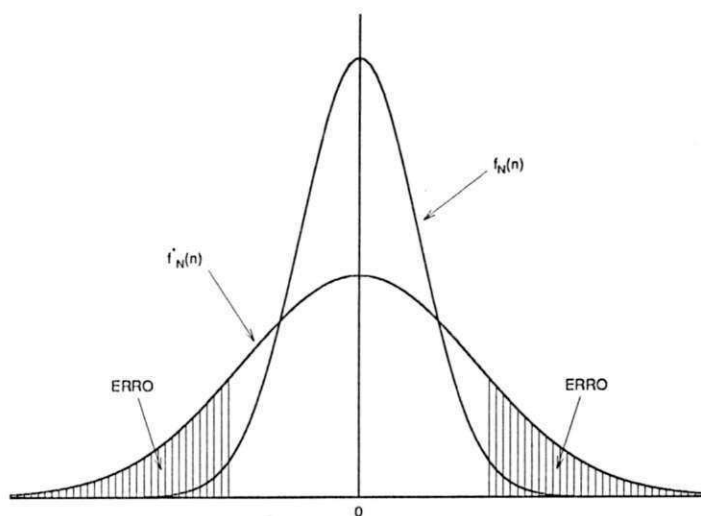


Figura 4.3: Polarização no método de valorização convencional das amostras (CIS).

Na Seção 4.4 o valor ótimo de α é obtido analiticamente para um sistema BPSK em um canal AWGN. Já no Capítulo 5, o problema de otimização de parâmetros da distribuição polarizada é tratado de uma forma mais genérica.

Uma alternativa que surgiu posteriormente para modificar a fdp gaussiana foi proposta em [5] e é conhecida como **valorização das amostras melhorada** (*Improved Importance Sampling - IIS*). A idéia é usar um processo não estacionário para modificar o valor médio da fdp gaussiana, de acordo com os valores atuais da seqüência de bits de informação. A variância da distribuição não é modificada.

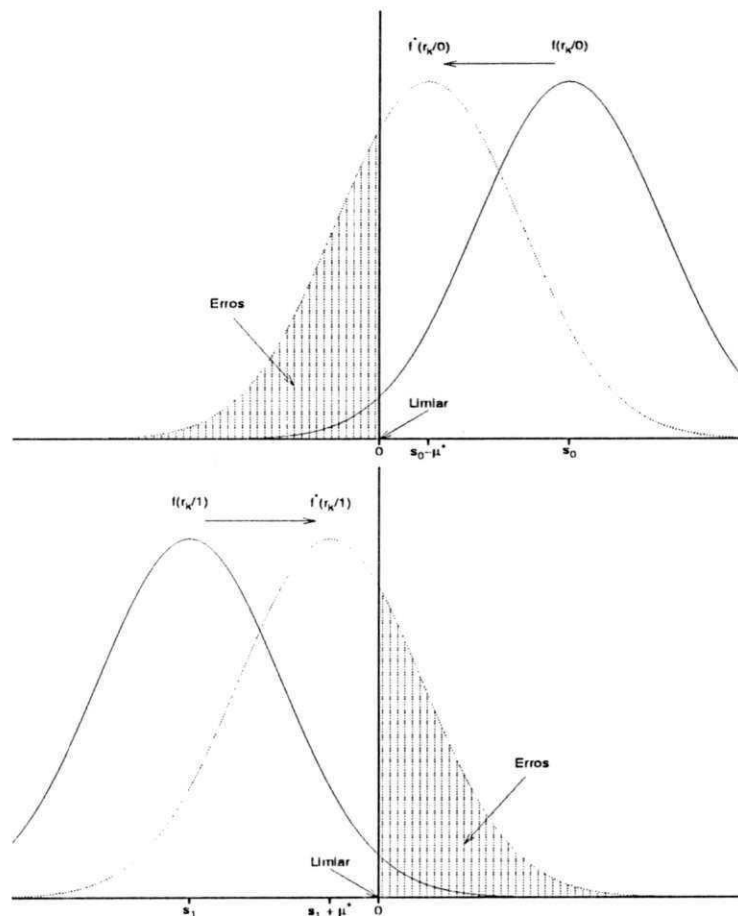


Figura 4.4: Polarização no método de valorização melhorada das amostras (IIS).

O termo não-estacionário é devido a distribuição gaussiana polarizada poder ser diferente em instantes distintos. Um simples exemplo ilustra esse procedimento. Retornando ao sistema BPSK discutido na Seção 3.4, o bit “0” é reconhecido no receptor pelo valor s_0 , e o sinal r_k na entrada do decisor tem distribuição gaussiana com média

Tabela 4.1: Resultados da simulação com valorização das amostras para o sistema BPSK.

SNR	P_b	\hat{p}_{IIS}	$\hat{\sigma}_{IIS}^2$	Erro normalizado
0	$7,860 \cdot 10^{-02}$	$7,860 \cdot 10^{-02}$	$1,109 \cdot 10^{-08}$	0,13%
1	$5,630 \cdot 10^{-02}$	$5,618 \cdot 10^{-02}$	$6,167 \cdot 10^{-09}$	0,14%
2	$3,750 \cdot 10^{-02}$	$3,756 \cdot 10^{-02}$	$2,996 \cdot 10^{-09}$	0,15%
3	$2,290 \cdot 10^{-02}$	$2,282 \cdot 10^{-02}$	$1,221 \cdot 10^{-09}$	0,15%
4	$1,250 \cdot 10^{-02}$	$1,251 \cdot 10^{-02}$	$4,040 \cdot 10^{-10}$	0,16%
5	$6,000 \cdot 10^{-03}$	$5,947 \cdot 10^{-03}$	$1,015 \cdot 10^{-10}$	0,17%
6	$2,400 \cdot 10^{-03}$	$2,387 \cdot 10^{-03}$	$1,819 \cdot 10^{-11}$	0,18%
7	$7,730 \cdot 10^{-04}$	$7,740 \cdot 10^{-04}$	$2,137 \cdot 10^{-12}$	0,19%
8	$1,910 \cdot 10^{-04}$	$1,910 \cdot 10^{-04}$	$1,460 \cdot 10^{-13}$	0,20%
9	$3,360 \cdot 10^{-05}$	$3,349 \cdot 10^{-05}$	$5,061 \cdot 10^{-15}$	0,21%
10	$3,870 \cdot 10^{-06}$	$3,884 \cdot 10^{-06}$	$7,611 \cdot 10^{-17}$	0,23%
11	$2,61 \cdot 10^{-07}$	$2,626 \cdot 10^{-07}$	$3,912 \cdot 10^{-19}$	0,24%
12	$9,01 \cdot 10^{-09}$	$9,010 \cdot 10^{-09}$	$5,230 \cdot 10^{-22}$	0,25%
13	$1,33 \cdot 10^{-10}$	$1,328 \cdot 10^{-10}$	$1,284 \cdot 10^{-25}$	0,27%
14	$6,81 \cdot 10^{-13}$	$6,806 \cdot 10^{-13}$	$3,791 \cdot 10^{-30}$	0,29%

No próximo capítulo o problema de encontrar distribuições de probabilidades polarizadas de modo a minimizar a variância do estimador será analisado.

4.5 Conclusões

O método de estimação com valorização das amostras é uma alternativa ao método de Monte Carlo que pode melhorar significativamente o desempenho das simulações. Valorização das amostras faz parte de um conjunto de técnicas conhecidas como métodos de redução de variância, que se utilizados podem reduzir a variância do estimador, ou equivalentemente reduzir o número necessário de iterações para obter uma estimativa aceitável. Essa melhoria no desempenho da simulação ocorre pois o método de valorização das amostras utiliza a estatística dos processos envolvidos para que os erros, que são os eventos importantes, ocorram com maior frequência durante a simulação.

Embora o ganho de desempenho obtido com valorização das amostras tem sido demonstrado para vários sistemas, alguns desses bastante complexos, não há uma metodologia genérica para a aplicação do método, sendo então necessário analisar os processos estocásticos presentes no modelo e então estudar a melhor forma de valorizar

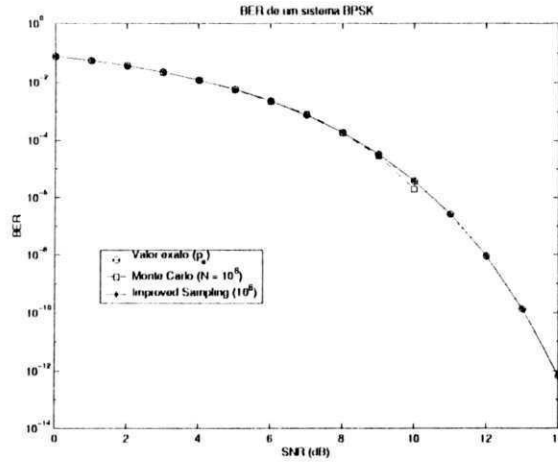


Figura 4.5: Simulação com valorização das amostras para o sistema BPSK sobre um canal AWGN.

as amostras para a simulação. Embora essa análise possa ser complicada, após encontrar um esquema de polarização adequado, as simulações serão mais rápidas, podendo recuperar o tempo tomado na análise.

A análise para encontrar a polarização mais adequada para a simulação muitas vezes resulta em um número de parâmetros a serem escolhidos. A seleção desses valores impacta fortemente no desempenho da simulação, de modo que alguns autores demonstraram a necessidade de utilizar métodos de otimização de parâmetros para selecionar valores que minimizem a variância do estimador. No próximo capítulo serão estudadas diferentes abordagens para otimização de parâmetros da simulação.

Capítulo 5

Otimização de Parâmetros em IS

A eficiência da metodologia de valorização das amostras depende fortemente da polarização das distribuições envolvidas. Isto é, a seleção da densidade de polarização determina a precisão do estimador, ou correspondentemente, o tempo de processamento necessário para obter a precisão desejada. O problema de encontrar a densidade de polarização pode ser dividido em duas partes, o primeiro passo é escolher a classe de densidades que satisfaz algumas condições gerais e que é simples de implementar. Em seguida é necessário selecionar parâmetros dessa distribuição de modo a otimizar o desempenho da valorização das amostras.

Em muitas aplicações, os elementos das classes de densidades selecionadas são caracterizados por um número de parâmetros. Por exemplo, se a classe de densidades selecionada é a família de densidades gaussianas, os elementos desta classe são caracterizados pelos valores da média e da variância.

Selecionar uma classe de densidades de probabilidade adequada não garante que a polarização será eficiente. Na realidade, um estimador eficiente resultará de uma boa escolha da classe de densidades de polarização juntamente com uma seleção ótima dos valores dos parâmetros característicos associados.

Esse capítulo estuda os métodos de otimização de parâmetros para a valorização das amostras. Na Seção 5.1 o problema de otimização de parâmetros será definido formalmente. A Seção 5.2 apresenta uma visão geral das diferentes abordagens para otimização e detalha dois métodos numéricos de interesse.

5.1 Formulação do Problema

De acordo com a Seção 4.1, o problema de estimar a probabilidade de erro de um sistema de comunicações, utilizando o método de valorização das amostras, pode ser visto como o problema de estimar o valor esperado

$$p_e = E_*[H(g(\mathbf{X}))w(\mathbf{X})].$$

em que \mathbf{X} é um vetor aleatório com densidade de probabilidade conjunta $f_{\mathbf{X}}(\cdot)$ conhecida, e $w(\cdot) = f_{\mathbf{X}}(\cdot)/f_{\mathbf{X}}^*(\cdot)$. O subscrito no valor esperado indica que a densidade que multiplica o argumento na integral do valor esperado é a densidade de polarização $f_{\mathbf{X}}^*(\cdot)$. O estimador com valorização das amostras para p_e é

$$\hat{p}_{IS} = \frac{1}{N} \sum_{i=1}^N H(g(\mathbf{x}_i))w(\mathbf{x}_i, \mathbf{r}), \quad \mathbf{x}_i \sim f_{\mathbf{X}}^*(\mathbf{x}, \mathbf{r}). \quad (5.1)$$

É assumido aqui que a densidade de polarização $f_{\mathbf{X}}^*(\cdot)$ pertence a uma classe de densidades com parâmetros característicos representados pelo vetor \mathbf{r} . Na Equação 5.1 foi escrito explicitamente a dependência que a função $f_{\mathbf{X}}^*(\cdot)$ tem desses parâmetros. Da definição da função peso $w(\cdot)$, também conclui-se que ela depende do vetor de parâmetros \mathbf{r} , como indicado em 5.1.

Retornando à Equação 4.10, a variância do estimador com valorização das amostras é

$$\sigma_{IS}^2 = \frac{1}{N} \{E[H(g(\mathbf{X}))w(\mathbf{X})] - p_e^2\} \quad (5.2)$$

$$= \frac{1}{N} \{E_*[H(g(\mathbf{X}))w^2(\mathbf{X})] - p_e^2\}. \quad (5.3)$$

Note que a Equação 5.3 é simplesmente o resultado de aplicar a definição da função peso em 5.2. Para otimizar o estimador é preciso minimizar a função custo σ_{IS}^2 em relação aos parâmetros característicos.

Para simplificar a abordagem do problema, denotando $I(\mathbf{r})$ como os valores esperados equivalentes nas Equações 5.2 e 5.3, minimizar a variância do estimador com valorização das amostras é o mesmo problema de minimizar a função

$$I(\mathbf{r}) = E[H(g(\mathbf{X}))w(\mathbf{X})] \quad (5.4)$$

$$= E_*[H(g(\mathbf{X}))w^2(\mathbf{X})]. \quad (5.5)$$

Nas próximas seções serão mostradas as principais abordagens para minimizar a função $I(\mathbf{r})$.

5.2 Métodos de Otimização

A abordagem força-bruta para otimização de parâmetros é realizar vários experimentos com diferentes valores de \mathbf{r} , e então escolher os valores de \mathbf{r} que resultaram na menor variância do estimador. A desvantagem evidente dessa abordagem é que faz-se necessário uma multiplicidade de experimentos longos.

Em alguns casos particulares é possível usar técnicas analíticas para determinar os valores ótimos dos parâmetros da simulação, como mostrado em [5], [10] e [24]. Entretanto, essa abordagem em geral é tão difícil quanto o problema de encontrar a própria probabilidade de erro.

Uma alternativa é encontrar limites para a variância do estimador e relacionar estes limites para os valores dos parâmetros \mathbf{r} . Esta abordagem foi sugerida em [11].

Desde que os métodos descritos acima são computacionalmente ineficientes (método de tentativa e erro) ou só adequados para problemas particulares, eles não estão sendo amplamente utilizados. Abordagens mais populares são baseadas em *exponential twisting* ou métodos adaptativos.

O método *exponential twisting* é relacionado aos resultados da teoria do desvio amplo (*large deviations theory*). Pode-se encontrar um tratamento extensivo desse campo em [36]. Essa técnica tem sido aplicada com sucesso na otimização de parâmetros de importance sampling em comunicações óticas [17], [18] e problemas gerais em comunicações digitais [16], [19]. Embora tenha sido mostrado que *exponential twisting* é uma ferramenta útil em alguns problemas, para ser utilizada é necessário o logaritmo da função geratriz dos momentos da densidade de probabilidade em estudo, o que em muitos problemas de comunicação isto não é disponível.

Uma importante classe de métodos com valorização das amostras é conhecida como **valorização adaptativa das amostras** (*adaptive importance sampling*), em que são

utilizados métodos numéricos para a otimização dos parâmetros da distribuição polarizada. Na próxima seção esses métodos serão analisados.

5.3 Valorização Adaptativa das amostras

A abordagem mais geral para a minimização de $I(\cdot)$ com respeito ao vetor de parâmetros r é usar métodos numéricos. Mesmo com o acréscimo de custo computacional, a generalidade desse método é uma vantagem decisiva em relação a outras abordagens de otimização.

Os métodos numéricos existentes para otimização de funções podem exigir o conhecimento de derivadas da função. Os métodos que não exigem essa informação são teoricamente mais genéricos, e não são necessariamente menos eficientes que os métodos baseados em derivadas.

As seções seguintes discutem a aplicabilidade de diferentes tipos de métodos numéricos, separados em métodos com derivadas e métodos sem derivadas, para a otimização de parâmetros em importance sampling através da minimização de $I(\cdot)$.

5.3.1 Métodos de Otimização sem Derivadas

Considere o problema de otimização unidimensional. Isso implica em procurar pelo mínimo da função $I(\cdot)$, definida na Equação 5.4, em que o argumento da função é a variável unidimensional r . O objetivo é encontrar o valor de r para o qual $I(r)$ é mínimo.

O primeiro passo para encontrar o mínimo de uma função é selecionar o intervalo $[a, c]$ em que se deseja obter o valor mínimo da função. Métodos numéricos para minimização de função envolvem um processo iterativo em que porções desse intervalo que não contém o valor mínimo são excluídos. Em linhas gerais o procedimento é como segue,

1. Dados os valores das abscissas a e c , escolhe-se um terceiro valor b , em que $a < b < c$, no qual sabe-se que $I(a) > I(b)$ e $I(b) < I(c)$.
2. Escolhe-se um valor x no intervalo (b, c) . Também poderia-se escolher no intervalo (a, b) e seguir um procedimento similar.

3. Se $I(b) < I(x)$, então o novo trio de pontos é $\{a, b, x\}$; caso contrário, o novo trio de pontos é $\{b, x, c\}$.
4. O processo é repetido iterativamente até que o valor de $I(\cdot)$ no meio do trio de pontos seja suficientemente pequeno.

O valor da abscissa do novo ponto, entre os extremos, poderia ser escolhido no centro do intervalo. Entretanto, uma forma melhor de selecionar a abscissa do novo ponto é usar uma seção de ouro (*golden section*). Isso implica que, dado o trio de pontos $\{a, b, c\}$, $a < b < c$, o novo ponto x em que $I(\cdot)$ será calculado deve ser em uma fração 0.38197 do maior intervalo entre os intervalos (a, b) e (b, c) , iniciando do ponto b [33]. Uma alternativa para a seção de ouro é utilizar as séries de Fibonacci para selecionar a abscissa do novo ponto.

Os métodos descritos sempre convergem para um mínimo local independentemente da curva da função $I(\cdot)$. quando a função é bem comportada, o método da **inter-polação parabólica inversa** é mais eficiente. Este método assume que a função é bastante similar a uma parábola na região próxima ao mínimo [33]. Obviamente, o método de interpolação parabólica inversa não é robusto porque ele conta que a função é similar a uma parábola. O **método de Brent** é um procedimento de minimização que usa o método de interpolação parabólica e, quando esse não é conveniente, muda para a procura usando a seção de ouro. O método de Brent combina um algoritmo de convergência rápida para as seções parabólica da função, com um algoritmo robusto para as outras seções.

No contexto de natureza estocástica em que o problema de otimização de parâmetros de importance sampling está inserido, o valor da função $I(\cdot)$ é calculado pelo estimador

$$\hat{I}(r) = \frac{1}{N} \sum_{i=1}^N H(g(x_i)) w^2(x_i, r), \quad x_i \sim f_X^*(x, r). \quad (5.6)$$

Isto implica que os valores da função usada nos métodos de otimização descritos até agora são variáveis aleatórias. Com isso, é possível que em um determinado valor de $\hat{I}(\cdot)$ o intervalo escolhido pelo algoritmo seja incorreto. Por exemplo, escolhendo um trio $\{a, b, c\}$, $a < b < c$, para obter um novo ponto y para a próxima iteração é preciso comparar $I(y)$ com $I(b)$. Se $I(y) < I(b)$, um valor de $\hat{I}(y)$ muito maior que o valor verdadeiro $I(y)$ poderia fazer com que o algoritmo conclua que $I(y) > I(b)$, e

portanto escolheria um novo trio de abscissas incorreto para a próxima iteração. Isso significa que a aleatoriamente envolvida no cálculo da função $I(\cdot)$ constitui uma séria desvantagem em sua otimização.

Para concluir, métodos de minimização que não exigem derivadas da função, não são convenientes para a otimização de parâmetros no método de valorização das amostras. Nas próximas seções serão estudados dois métodos bastante conhecidos que usam derivadas para determinar o valor mínimo de uma função, e são adequados para a otimização da função estocástica $\hat{I}(\cdot)$.

5.3.2 Gradiente Descendente Estocástico (GDE)

O gradiente descendente estocástico (*stochastic gradient descent*), é um método bem conhecido para a otimização de funções multi-dimensionais. Basicamente ele envolve um processo iterativo em que o gradiente da função em um ponto é usado para determinar o ponto da próxima iteração.

O gradiente estocástico é uma versão modificada do método *gradient descent* para realizar a minimização de funções estocásticas. A aplicação dessa abordagem na otimização de parâmetros em *importance sampling* tem sido mostrada em diferentes problemas em sistemas de comunicações [13], [14] e [12].

Para iniciar o problema de minimizar $I(\mathbf{r})$, sendo o gradiente de $I(\mathbf{r})$ em relação a \mathbf{r} denotado por $\nabla_{\mathbf{r}}I(\mathbf{r})$, é bem conhecido [13] que

$$\nabla_{\mathbf{r}}I(\mathbf{r}) \Big|_{\mathbf{r}=\mathbf{r}_{\text{opt}}} = \mathbf{0} \quad (5.7)$$

constitui uma condição necessária, mas não suficiente, para que o vetor \mathbf{r}_{opt} seja um mínimo local ou global de $I(\mathbf{r})$. Assim, na procura do mínimo, o algoritmo de otimização envolve uma descida (*descent*) na função $I(\cdot)$ em uma direção dada por $\nabla_{\mathbf{r}}I(\mathbf{r})$. Esse algoritmo é conhecido como gradiente descendente determinístico (DGD - *Deterministic Gradient Descent*), e é um esquema iterativo com a k -ésima iteração dada por

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \delta_k \nabla_{\mathbf{r}}I(\mathbf{r}_k), \quad m = 1, 2, \dots \quad (5.8)$$

em que δ_k é o tamanho do passo utilizado na k -ésima iteração, e é obtido experimentalmente.

Como em muitas aplicações uma expressão em forma fechada de $\nabla_{\mathbf{r}}I(\cdot)$ não é disponível, usa-se uma estimativa despolarizada $\hat{\nabla}_{\mathbf{r}}I(\cdot)$ do gradiente. Mudando o gradiente determinístico pela estimativa estocástica despolarizada resulta no seguinte algoritmo gradiente descendente estocástico (GDE)

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \delta_k \hat{\nabla}_{\mathbf{r}}I(\mathbf{r}_k), \quad k = 1, 2, \dots \quad (5.9)$$

O gradiente da função $I(\cdot)$ é obtido como segue. Reescrevendo a definição de $I(\cdot)$ utilizando a integral do valor esperado, tem-se

$$\begin{aligned} I(\mathbf{r}) &= E[H(g(\mathbf{X}))w(\mathbf{X}, \mathbf{r})] \\ &= \int_{-\infty}^{\infty} H(g(\mathbf{x}))w(\mathbf{x}, \mathbf{r})f_{\mathbf{X}}(\mathbf{x})d\mathbf{x}. \end{aligned} \quad (5.10)$$

Como mostrado em [13], o operador gradiente pode ser transposto com o operador integral, que resulta,

$$\begin{aligned} \nabla_{\mathbf{r}}I(\mathbf{r}) &= \int_{-\infty}^{\infty} H(g(\mathbf{x}))\nabla_{\mathbf{r}}w(\mathbf{x}, \mathbf{r})f_{\mathbf{X}}(\mathbf{x})d\mathbf{x} \\ &= \int_{-\infty}^{\infty} H(g(\mathbf{x}))w(\mathbf{x}, \mathbf{r})\nabla_{\mathbf{r}}w(\mathbf{x}, \mathbf{r})f_{\mathbf{X}}^*(\mathbf{x})d\mathbf{x} \\ &= E_*[H(g(\mathbf{X}))w(\mathbf{X}, \mathbf{r})\nabla_{\mathbf{r}}w(\mathbf{X}, \mathbf{r})]. \end{aligned} \quad (5.11)$$

O estimador despolarizado para $\nabla_{\mathbf{r}}I(\mathbf{r})$ é obtido de modo similar ao estimador com valorização das amostras, ou seja,

$$\hat{\nabla}_{\mathbf{r}}I(\mathbf{r}) = \frac{1}{N} \sum_{i=1}^N H(g(\mathbf{x}_i))w(\mathbf{x}_i, \mathbf{r})\nabla_{\mathbf{r}}w(\mathbf{x}_i, \mathbf{r}), \quad \mathbf{x}_i \sim f_{\mathbf{X}}^*(\mathbf{x}, \mathbf{r}). \quad (5.12)$$

A implementação do algoritmo gradiente descendente estocástico é como segue:

1. Encontra-se a expressão analítica para o gradiente da função peso que aparece na Equação 5.12.
2. Escolhe-se um valor inicial dos parâmetros característicos $\mathbf{r} = \mathbf{r}_1$.
3. Faz-se uma simulação curta para estimar o gradiente de $I(\cdot)$ nesse ponto, de acordo com a Equação 5.12.

4. Calcula-se \mathbf{r}_2 utilizando a Equação 5.9.

5. Repete-se 3 e 4 iterativamente.

Como dito anteriormente, o passo δ_k é obtido experimentalmente. Um alto valor para o passo pode produzir desvios do caminho de convergência do algoritmo, por outro lado, um passo curto implicará em um alto custo computacional. Portanto, a escolha do valor dessa variável envolve uma relação de compromisso entre o custo computacional e a convergência do algoritmo.

De acordo com a descrição clássica do algoritmo DGD, o passo pode ser escolhido como $\delta_k = 1/\|\hat{\nabla}_{\mathbf{r}} I(\mathbf{r})\|$. No contexto da otimização dos parâmetros do método de valorização das amostras, pode-se usar $\delta_k = \Delta/\hat{\sigma}_{IS}^2$, em que o denominador é a estimativa da variância do estimador com valorização das amostras dada pela Equação 4.11, e Δ é um número fixo que é escolhido experimentalmente.

A escolha do passo é bastante flexível, pode-se iniciar a busca do valor mínimo com um passo maior, acelerando assim a simulação, e na medida que os valores do gradiente comecem a oscilar, pode-se encurtar o passo.

É importante notar que para o cálculo do k -ésimo vetor de parâmetros \mathbf{r}_k , uma simulação completa é realizada, ou seja, N_k símbolos são gerados e o modelo do sistema é aplicado. Com isso, as amostras aleatórias \mathbf{X} poderão ser utilizadas não apenas para estimar o gradiente usando a Equação 5.12, mas também pode-se utilizar essas amostras para calcular o valor de $I(\mathbf{r}_k)$ utilizando o estimador

$$\hat{I}(\mathbf{r}_k) = \frac{1}{N_k} \sum_{i=1}^{N_k} H(g(\mathbf{x}_i)) w^2(\mathbf{x}_i, \mathbf{r}_k), \quad \mathbf{x}_i \sim f_{\mathbf{X}}^*(\mathbf{x}, \mathbf{r}_k). \quad (5.13)$$

em que o subscrito k indica que esta simulação é realizada com uma densidade de polarização com parâmetros característicos $\mathbf{r} = \mathbf{r}_k$. Também pode-se usar os mesmos resultados da k -ésima simulação para estimar a probabilidade de erro p_e usando a equação

$$\hat{p}_{IS_k} = \frac{1}{N_k} \sum_{i=1}^{N_k} H(g(\mathbf{x}_i)) w(\mathbf{x}_i, \mathbf{r}_k), \quad \mathbf{x}_i \sim f_{\mathbf{X}}^*(\mathbf{x}, \mathbf{r}_k). \quad (5.14)$$

Além disso, a partir da estimativa $\hat{I}(\mathbf{r}_k)$ pode-se encontrar uma estimativa da variância do estimador \hat{p}_{IS_k} utilizando as Equações 5.2 e 5.4. Comparando essa variância

e o valor \hat{p}_{IS_k} obtido, tem-se uma medida da precisão do estimador, e portanto pode-se decidir quando parar o processo iterativo.

No algoritmo GDE, a função de polarização $f_{\mathbf{X}}^*(\mathbf{x}, \mathbf{r}_k)$, cujos parâmetros característicos são obtidos iterativamente, será ótima para estimar p_c quando $k \rightarrow \infty$. Entretanto, essa distribuição não será necessariamente ótima para estimar o vetor $\nabla_{\mathbf{r}} I(\mathbf{r})$. Felizmente, em muitos casos práticos a função $f_{\mathbf{X}}^*(\mathbf{x}, \mathbf{r}_k)$ provê estimativas com precisão suficiente e o algoritmo converge.

O algoritmo GDE exige que a função peso seja diferenciável com respeito a \mathbf{r} . Felizmente, a maioria das densidades de probabilidade contínuas são diferenciáveis com respeito a seus parâmetros característicos, e em geral não será difícil selecionar a densidade de polarização de uma classe de densidades tal que a função peso é diferenciável.

Na próxima seção será abordado o método de Newton, que é bastante similar ao método do gradiente, tendo como diferença a utilização da primeira e segunda derivada de $I(\cdot)$ ao invés do gradiente.

5.3.3 Método Estocástico de Newton

O método de Newton é uma segunda abordagem numérica amplamente utilizada na minimização de funções determinística. O princípio desse método é encontrar uma direção no sentido do mínimo para a aproximação quadrática da função, ao invés da aproximação linear como é feito no método do gradiente.

A aplicação do método de Newton para a otimização dos parâmetros característicos \mathbf{r} é bastante recente, e até o momento não tem sido amplamente utilizada. Embora o método de otimização utilizado nesse trabalho seja o método do gradiente, a seguir é mostrado um resumo da aplicação do método de Newton.

Para simplificar, considere o problema unidimensional, ou seja, a minimização da função $I(r)$ em que r é um escalar. Semelhantemente ao que foi feito na seção anterior para obter o gradiente de $I(\cdot)$, a derivada dessa função com respeito a r é dada por

$$\begin{aligned} I'(r) &= \int_{-\infty}^{\infty} H(g(x))w'(\mathbf{x}, r)f_{\mathbf{X}}(\mathbf{x})d\mathbf{x} \\ &= \int_{-\infty}^{\infty} H(g(\mathbf{x}))w(\mathbf{x}, r)w'(\mathbf{x}, r)f_{\mathbf{X}}^*(\mathbf{x})d\mathbf{x} \\ &= E_* [H(g(\mathbf{X}))w(\mathbf{X}, \mathbf{r})w'(\mathbf{x}, r)]. \end{aligned} \tag{5.15}$$

Procedendo de maneira similar, a segunda derivada de $I(\cdot)$ pode ser escrita como

$$I''(r) = E_* [H(g(\mathbf{X}))w(\mathbf{X}, \mathbf{r})w''(\mathbf{X}, r)]. \quad (5.16)$$

Semelhantemente ao gradiente, essas duas derivadas precisam ser estimadas. Dois estimadores despolarizados para a primeira e segunda derivada, respectivamente, são

$$\hat{I}'(r) = \frac{1}{N} \sum_{i=1}^N H(g(\mathbf{x}_i))w(\mathbf{x}_i, \mathbf{r})w'(\mathbf{x}_i, r) \quad \mathbf{x}_i \sim f_{\mathbf{X}}^* \quad (5.17)$$

e

$$\hat{I}''(r) = \frac{1}{N} \sum_{i=1}^N H(g(\mathbf{x}_i))w(\mathbf{x}_i, \mathbf{r})w''(\mathbf{x}_i, r) \quad \mathbf{x}_i \sim f_{\mathbf{X}}^*. \quad (5.18)$$

Essas estimativas podem ser usadas para implementar o método de Newton para minimizar $I(\cdot)$. Utilizar as estimativas nas Equações 5.17 e 5.18 ao invés dos valores reais, configura uma versão estocástica do método de Newton, desde que esse método é aqui nomeado como método de Newton estocástico, e é definido como

$$r_{k+1} = r_k - \delta_r \frac{\hat{I}'(r)}{\hat{I}''(r)}. \quad (5.19)$$

A quantidade δ_r é um passo fixo escolhido experimentalmente. Esse passo é menor que a unidade, e semelhantemente ao passo utilizado no método do gradiente, a escolha de δ_r configura uma relação de compromisso entre custo computacional e convergência. Pode-se notar comparando as Equações 5.9 e 5.19 que a implementação do algoritmo estocástico de Newton é similar ao método do gradiente.

O algoritmo de Newton exige que a função $I(r)$ seja duas vezes diferenciável com respeito a r . Felizmente, a maioria das densidades de probabilidade contínuas têm essa propriedade, e em geral não será difícil selecionar a densidade de polarização de uma classe de densidades tal que isso seja verdade.

5.4 Conclusões

Entre os métodos de otimização de parâmetros citados, o método GDE é o mais simples de ser implementado e é o mais utilizado. Embora a convergência do algoritmo tenha

sido afirmada pelos autores, ainda não foi feito um estudo específico para a aplicação em sistemas de comunicações que demonstrasse formalmente a convergência. Além disso, a utilização de estimativas do gradiente ao invés do valor exato não tem sido considerada pelos autores no estudo da convergência. Uma questão aberta é conhecer o comprimento da simulação necessário para estimar o gradiente suficientemente para a convergência do método.

Outra desvantagem é a necessidade de calcular analiticamente o gradiente da função peso, embora na maioria das distribuições em comunicações esse cálculo seja simples.

Embora o custo de processamento desse método seja alto, pois várias simulações são realizadas até que os parâmetros ótimos sejam encontrados, após encontrar esses valores é necessário realizar apenas uma simulação. Portanto, é comum realizar simulações curtas para obter os parâmetros ótimos e em seguida faz-se uma simulação longa para obter uma melhor precisão.

No próximo capítulo o método de valorização das amostras é estendido para sistemas com codificação. No Capítulo 7 o método GDE é aplicado a sistemas com codificação e os resultados das simulações são apresentados.

Capítulo 6

IS Aplicada a Sistemas Codificados

Muitos sistemas de comunicações utilizam codificação para realizar algum tipo de controle de erro durante a recepção da informação transmitida. Tais sistemas têm memória, desde que a decisão de um bit é feita de acordo com uma seqüência de símbolos transmitidos. Em outras palavras, existe uma dependência entre um número de bits consecutivos recebidos, que é explorada pelo receptor para decidir sobre um único bit.

A Figura 6.1 ilustra o modelo discreto no tempo para o sistema. O codificador localizado no transmissor é responsável por introduzir uma redundância estruturada entre os bits de informação. O decodificador, localizado no receptor do sistema, explora essa redundância para detectar, e se possível corrigir, erros causados pelo canal.

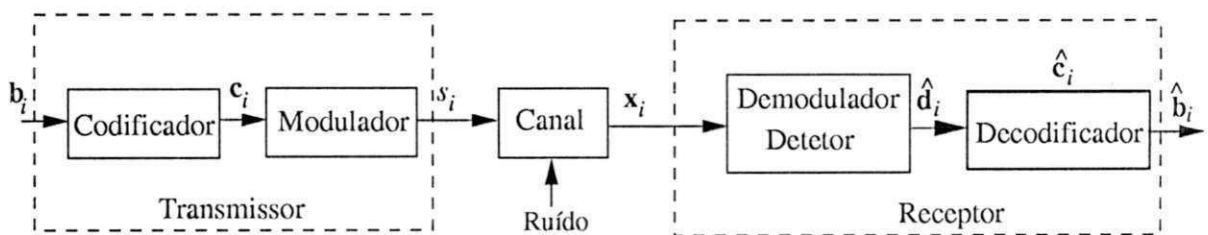


Figura 6.1: Modelo discreto para o sistema de comunicações com codificação.

Nesse capítulo será discutida a aplicação do estimador com valorização das amostras para sistemas com codificação. Os códigos de bloco serão discutidos com maior detalhe por serem o objetivo das simulações desse trabalho. Na Seção 6.1 será discutido o uso do estimador de Monte Carlo e as dificuldades para implementar o estimador com valorização das amostras para tais sistemas. Na Seção 6.2 o método proposto em [22] para adequar valorização das amostras a sistemas com codificação será detalhado para

códigos de bloco.

6.1 Estimação de Desempenho em sistemas com codificação

O desempenho de sistemas com codificação tem sido amplamente estimado via o método de Monte Carlo padrão, pelas mesmas razões discutidas no Capítulo 3. Recentemente, simulação com valorização das amostras tem sido utilizada com sucesso para esses sistemas [22] [23] [6].

Para descrever a aplicação do método de Monte Carlo para esses sistemas, será utilizado o modelo ilustrado na Figura 6.1.

O bloco de bits de informação \mathbf{b}_k é obtido usando geradores de bits aleatórios e é codificado de acordo com a estrutura do codificador. Os bits já codificados são mapeados em formas de onda (seqüência de amostras) pelo modulador e corrompidos pelo ruído obtido por geradores de números aleatórios. O processo inverso é aplicado ao sinal na entrada do receptor. Sempre que um erro ocorre, um contador de erros é incrementado, enquanto que o contador de bits é obtido de acordo com o número de bits de informação efetivamente transmitidos. A BER é obtida como a relação entre o número de bits recebidos com erro e o número de bits de informação transmitidos. Como pode-se notar, o uso do estimador de Monte Carlo é bastante similar ao descrito na Seção 3.4 para um sistema BPSK não codificado. Novamente, o método aqui torna-se impraticável para pequenos valores da probabilidade de erro, mas por outro lado persiste a vantagem da generalidade inerente da simulação de Monte Carlo.

A aplicação da técnica de valorização das amostras para sistemas com codificação não é tão direta como no caso do sistema BPSK não codificado. Duas dificuldades básicas são encontradas, a **memória** do sistema e a **dimensionalidade** do problema.

Os bits de informação são agora transmitidos pelos bits codificados. Como a polarização do ruído é feita de acordo com a seqüência de bits que trafega pelo canal, que são os bits codificados, obter o peso de cada bit de informação decodificado com erro não é tão simples como no sistema BPSK não codificado. Por exemplo, suponha que devido ao canal um número de bits, maior que a distância livre do código d_{free} , foi corrompido. O decodificador então faz uma decisão por uma palavra código diferente da transmitida e um número de bits de informação são recebidos com erro. O problema

é encontrar um valor para o peso de cada erro, pois devido a memória do sistema, vários bits contribuem para um erro, e conseqüentemente várias amostras do ruído. Assim, utilizar o peso de todas as amostras do ruído envolvidas no erro particular é uma abordagem que reduzirá o desempenho do estimador.

A segunda dificuldade, a dimensionalidade, está mais ligada ao problema de escolher a distribuição polarizada usada na simulação com valorização das amostras. Para o sistema BPSK foi possível encontrar as direções para polarizar o ruído, pois há apenas um limiar de decisão e só um erro possível de ocorrer. No caso de sistemas com codificação, a escolha da direção de polarização não é trivial, pois cada palavra-código transmitida constitui um ponto no espaço multidimensional que deve ser considerado um potencial centro de erros. Os centros determinam a direção de polarização, e durante a simulação todos os centros (erros) devem ser testados, mesmo que seja um número elevado. Esse problema pode ser estendido para sistemas com um grande número de formas de onda, espalhadas em um espaço multidimensional [22].

Para tratar com esse problema uma abordagem foi proposta [22] [23], em que a probabilidade de erro para cada centro é estimada via um experimento diferente. Para esse fim, o espaço multidimensional é dividido em duas regiões, correspondendo ao erro específico e seu complemento. Assim, a polarização em cada simulação será feita partindo de uma única palavra código transmitida, na direção de um centro (erro) específico. Por fim, a soma das probabilidades de erro das simulações de cada centro é utilizada para calcular a BER.

Essa abordagem será descrita na próxima seção para códigos de bloco. Em [23] esse método é detalhado para sistemas com código convolucional.

6.2 Valorização das Amostras para Simulação com Códigos de Bloco

Um código de bloco (n, k) é simplesmente uma coleção de $M = 2^k$ palavras-código de n bits, em que k é o número de bits de informação, n é o comprimento do código e $(n - k)$ é o número de bits de redundância que são relacionados algebricamente com os k bits de informação. A razão $r = k/n$ é conhecida como a taxa do código, em que $0 < r < 1$.

Fazendo¹ $\mathbf{b}_i = [b_i^{(1)}, b_i^{(2)}, \dots, b_i^{(k)}]$, $i = 1, \dots, M$, um bloco de k bits de informação, aplicando essa seqüência de bits a um codificador de bloco produz a palavra-código $\mathbf{c}_i = [c_i^{(1)}, c_i^{(2)}, \dots, c_i^{(n)}]$, que é transmitida pelo sistema. Como mostra a Figura 6.1, o vetor de amostras x_i na entrada do receptor é a composição do sinal transmitido e o ruído adicionado pelo canal. Após o demodulador e o detetor, a seqüência de bits $\hat{\mathbf{c}} = [\hat{c}^{(1)}, \hat{c}^{(2)}, \dots, \hat{c}^{(n)}]$ na entrada do decodificador é a seqüência mais provável de ter sido transmitida². O decodificador utiliza a redundância presente em $\hat{\mathbf{c}}$ para corrigir possíveis erros inseridos pelo canal, e obtém o bloco de bits $\hat{\mathbf{b}}_i = [\hat{b}_i^{(1)}, \hat{b}_i^{(2)}, \dots, \hat{b}_i^{(k)}]$ que é uma estimativa dos bits de informação transmitidos.

A regra de decodificação pode ser especificada por regiões de decisão [23]. Para cada palavra código \mathbf{c}_i , $i = 1, \dots, M$, é associada uma região de decisão $\mathfrak{R}(\mathbf{c}_i)$ de modo que $\mathfrak{R}(\mathbf{c}_i) \cap \mathfrak{R}(\mathbf{c}_j) = \emptyset$, para $i \neq j$. A regra de decodificação é selecionar \mathbf{c}_i como a palavra-código transmitida, sempre que $\hat{\mathbf{c}} \in \mathfrak{R}(\mathbf{c}_i)$.

Para simplificar a exposição, assumindo que a seqüência de informação \mathbf{b}_1 é transmitida, a probabilidade de erro de decodificação p_c é dada por

$$p_c = \sum_{i=2}^M p_i \quad (6.1)$$

em que $p_i = p[(\hat{\mathbf{c}} \in \mathfrak{R}(\mathbf{c}_i))/\mathbf{c}_1]$ é a **probabilidade de erro de decodificação específica** quando a palavra-código \mathbf{c}_i , $i \neq 1$, é decodificada dado que \mathbf{c}_1 foi transmitido. Em outras palavras, p_i é a probabilidade que a palavra-código decodificada pertença à região de decisão $\mathfrak{R}(\mathbf{c}_i)$, ao invés de $\mathfrak{R}(\mathbf{c}_1)$.

O propósito aqui é estimar a probabilidade de erro de bit p_b . Entretanto, como será mostrado posteriormente, p_b pode ser facilmente obtido a partir da probabilidade de erro de decodificação p_c .

Como exposto anteriormente, a abordagem proposta em [22] é estimar cada probabilidade específica p_i , Equação 6.1, separadamente, com experimentos que objetivam cada tipo de erro, ou seja, para cada simulação tem-se uma distribuição polarizada diferente.

Com o mesmo procedimento utilizado no capítulo 1, a probabilidade de erro específica pode ser escrita como uma integral, ou utilizando o valor esperado como segue

¹O subscripto i representa o índice dentro do alfabeto ao qual o bloco pertence.

²Note que \mathbf{c}_i é uma das $M = 2^k$ palavras do código, e $\hat{\mathbf{c}}$ pode ser qualquer uma das 2^n palavras possíveis de n bits.

$$p_i = \int H_i(g(\mathbf{x})) f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} \quad (6.2)$$

$$= E[H_i(g(\mathbf{X}))], \quad i = 2, \dots, M \quad (6.3)$$

em que $H_i(\cdot)$ é a função indicadora de erro para a região $\mathfrak{R}(c_i)$. Na simulação com valorização das amostras, o ruído é então gerado a partir da fdp polarizada $f_{\mathbf{X}_i}^*(\cdot)$ ao invés da fdp verdadeira $f_{\mathbf{X}}(\cdot)$. Reescrevendo as Equações 6.2 e 6.3 em termos da distribuição polarizada, tem-se

$$p_i = \int H_i(g(\mathbf{x})) w_i(\mathbf{x}) f_{\mathbf{X}_i}^*(\mathbf{x}) d\mathbf{x} \quad (6.4)$$

$$= E_{\star}[H_i(g(\mathbf{X})) w_i(\mathbf{X})], \quad i = 2, \dots, M, \quad (6.5)$$

em que $w_i(\cdot) = f_{\mathbf{X}}(\cdot)/f_{\mathbf{X}_i}^*(\cdot)$ é a função peso com respeito a distribuição polarizada $f_{\mathbf{X}_i}^*(\cdot)$. O estimador para a probabilidade de erro específica p_i é então a média amostral

$$\hat{p}_i = \frac{1}{N_i} \sum_{k=1}^{N_i} H_i(g(\mathbf{x}_k)) w_i(\mathbf{x}_k) \quad (6.6)$$

em que $N_i \neq 0$ é o número de palavras-código utilizadas na simulação, e as amostras \mathbf{x}_k são obtidas a partir da distribuição polarizada $f_{\mathbf{X}_i}^*(\cdot)$. Dado que \mathbf{b}_1 é transmitido, ou equivalentemente \mathbf{c}_1 , uma estimativa da probabilidade de erro de decodificação é

$$\hat{p}_c = \sum_{i=2}^M \hat{p}_i = \sum_{i=2}^M \sum_{k=1}^{N_i} \frac{1}{N_i} H_i(g(\mathbf{x}_k)) w_i(\mathbf{x}_k). \quad (6.7)$$

Antes de continuar a descrever a aplicação dessa abordagem é necessário obter o valor esperado e a variância dos estimadores das Equações 6.6 e 6.7. Fazendo $E_{\star}[\cdot]$ o valor esperado para a distribuição da simulação, e considerando que $\mathbf{x}_1, \dots, \mathbf{x}_{N_i}$ são independentes, o valor esperado do estimador \hat{p}_i é

$$\begin{aligned}
E_*[\hat{p}_i] &= E_* \left[\frac{1}{N_i} \sum_{k=1}^{N_i} H_i(g(\mathbf{x}_k)) w_i(\mathbf{x}_k) \right] \\
&= \frac{1}{N_i} \sum_{k=1}^{N_i} E_*[H_i(g(\mathbf{x}_k)) w_i(\mathbf{x}_k)] \\
&= \int H_i(g(\mathbf{x})) \frac{f_{\mathbf{X}}(\mathbf{x})}{f_{\mathbf{X}_i}^*(\mathbf{x})} f_{\mathbf{X}_i}^*(\mathbf{x}) d\mathbf{x} \\
&= \int H_i(g(\mathbf{x})) f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} \\
&= p_i,
\end{aligned} \tag{6.8}$$

e portanto o estimador 6.6 é despolarizado. Um cálculo similar indica que a variância do estimador é

$$\sigma_{\hat{p}_i}^2 = \frac{1}{N_i} \left\{ \int H_i(g(\mathbf{x})) w_i(\mathbf{x}) f_{\mathbf{X}_i}(\mathbf{x}) d\mathbf{x} - p_i^2 \right\} \tag{6.9}$$

$$= \frac{1}{N_i} \left\{ \int H_i(g(\mathbf{x})) w_i^2(\mathbf{x}) f_{\mathbf{X}_i}^*(\mathbf{x}) d\mathbf{x} - p_i^2 \right\} \tag{6.10}$$

que pode-se reescrever em termos do valor esperado

$$\sigma_{\hat{p}_i}^2 = \frac{1}{N_i} \left\{ E[H_i(g(\mathbf{x})) w_i(\mathbf{x})] - p_i^2 \right\} \tag{6.11}$$

$$= \frac{1}{N_i} \left\{ E_*[H_i(g(\mathbf{x})) w_i^2(\mathbf{x})] - p_i^2 \right\}. \tag{6.12}$$

Com os resultados 6.8 e 6.12 pode-se encontrar o valor esperado e a variância do estimador da probabilidade de erro de decodificação \hat{p}_c como segue

$$E[\hat{p}_c] = E \left[\sum_{i=2}^M \hat{p}_i \right] = \sum_{i=2}^M E[\hat{p}_i] \tag{6.13}$$

$$= \sum_{i=2}^M p_i = p_c, \tag{6.14}$$

e

$$\sigma_{\hat{p}}^2 = \sum_{i=2}^M \sigma_{\hat{p}_i}^2, \tag{6.15}$$

respectivamente. Durante a simulação é necessário estimar a precisão da estimativa obtida com a valorização das amostras, isso é feito usando o estimador da variância amostral dado por

$$\hat{\sigma}_{\hat{p}_i}^2 = \frac{1}{N_i - 1} \left\{ \frac{1}{N_i} \sum_{k=1}^{N_i} H_i(g(\mathbf{x}_k)) w_i^2(\mathbf{x}_k) - \hat{p}_i^2 \right\}. \quad (6.16)$$

Como dito anteriormente, o estimador da probabilidade de erro específica \hat{p}_i pode ser usado para estimar a probabilidade de erro de bit p_b . Fazendo $n_i(\mathbf{b}_1, \mathbf{b}_i)$ como o número de bits errados causados por decodificar \mathbf{b}_i ao invés de \mathbf{b}_1 , e dado que \mathbf{b}_1 é transmitido, a probabilidade de erro de bit média é

$$p_b = \frac{1}{k} \sum_{i=2}^M n_i(\mathbf{b}_1, \mathbf{b}_i) \cdot p_i, \quad (6.17)$$

em que $k = \log_2(M)$ é o número de bits de informação transmitidos por uma palavra código. Portanto, um estimador despolarizado de p_b é

$$\hat{p}_b(b_1) = \frac{1}{k} \sum_{i=2}^M n_i(\mathbf{b}_1, \mathbf{b}_i) \cdot \hat{p}_i. \quad (6.18)$$

No caso de canais binários simétricos e sem memória, com códigos de bloco lineares e decodificação com máxima verossimilhança, a probabilidade de erro de bit $p_b = p_b(\mathbf{b}_i)$, $i = 1, \dots, M$, é independente da mensagem transmitida [29].

Um exemplo simples pode ilustrar melhor o método descrito nessa seção. Suponha um código de bloco (5, 2) qualquer, em que a seqüência de bits de informação $\mathbf{b}_1 = \{0, 0\}$ é codificada na palavra-código $\mathbf{c}_1 = \{0, 0, 0, 0, 0\}$. Dado que \mathbf{c}_1 é transmitido pelo canal, deseja-se obter a probabilidade de erro específica p_2 que corresponde à probabilidade de decodificar a palavra-código $\mathbf{c}_2 = \{1, 0, 1, 0, 1\}$ ao invés de \mathbf{c}_1 . Como pode-se notar, os bits 1, 3 e 5 de \mathbf{c}_2 são os únicos bits diferentes em relação a \mathbf{c}_1 . Portanto, o método propõe polarizar a distribuição do ruído na direção da região $\mathfrak{R}(\mathbf{c}_2)$ para estimar p_2 . Isso é feito polarizando as amostras do ruído associadas aos bits 1, 3 e 5, e deixando as amostras relacionadas aos bits 2 e 4 sem polarização. Esse modelo de simulação torna o processo do ruído não estacionário, já que quando os bits correspondentes das palavras-código envolvidas \mathbf{c}_1 e \mathbf{c}_2 são diferentes, a fdp do ruído é polarizada, e quando os bits são iguais a fdp verdadeira é utilizada.

Na próximo capítulo o método descrito acima será aplicado a um código de bloco bastante conhecido, juntamente com o algoritmo de otimização GDE descrito no Capítulo 5. Os detalhes da implementação e resultados serão comentados.

Capítulo 7

Otimização de Parâmetros em IS para Sistemas com Codificação

Embora muitos resultados obtidos durante esse trabalho estejam distribuídos nas diversas seções do texto, nesse capítulo se concentram as simulações que são o real objetivo dessa dissertação.

Até o momento foram introduzidos os conceitos importantes relacionados à simulação com valorização das amostras, e a abordagem descrita no Capítulo 6 estendeu o uso desse método para sistemas com codificação.

Devido a inerente dependência da eficiência da valorização das amostras em relação à escolha da distribuição polarizada, no Capítulo 5 foram descritos métodos computacionalmente eficientes para a escolha ótima dos parâmetros dessa distribuição.

Os trabalhos principais, e que validaram o uso de valorização das amostras para sistemas com codificação [22] [23], utilizaram como parâmetros da distribuição valores obtidos pelo método de tentativa e erro, ou aplicaram translação da média utilizando diretamente os resultados da valorização melhorada das amostras (*IIS - Improved Importance Sampling*), deslocando o centro da distribuição do ruído para o limiar de decisão de bit. Além disso, os parâmetros da distribuição da simulação permaneceram os mesmos para todas palavras-código.

Nesse Capítulo serão examinados em mais detalhes os resultados obtidos com o uso da valorização das amostras em sistemas com codificação. Será proposta uma nova metodologia para esse problema que é capaz de otimizar os parâmetros da distribuição polarizada e utiliza parâmetros diferentes para a simulação separada de cada palavra-código.

7.1 Formulação do Problema

Para introduzir o método proposto nesse Capítulo, a seguir será descrito o problema em questão.

A utilização de valorização das amostras foi descrita no Capítulo 4, e baseia-se em dividir o problema de obter a probabilidade de erro de decodificação calculando separadamente a probabilidade de erro específica de cada erro possível dentro do alfabeto do código. Assumindo que a seqüência de informação \mathbf{b}_1 é transmitida, a probabilidade de erro de decodificação p_c é dada por

$$p_c = \sum_{i=2}^M p_i \quad (7.1)$$

em que p_i é a probabilidade de erro de decodificação específica quando a palavra-código \mathbf{c}_i , $i \neq 1$, é decodificada dado que \mathbf{c}_1 foi transmitido, e pode ser escrito como

$$p_i = \int H_i(g(\mathbf{x})) w_i(\mathbf{x}) f_{\mathbf{X}_i}^*(\mathbf{x}) d\mathbf{x} \quad (7.2)$$

$$= E_*[H_i(g(\mathbf{X})) w_i(\mathbf{X})] \quad (7.3)$$

em que $w_i(\cdot) = f_{\mathbf{X}_i}(\cdot)/f_{\mathbf{X}}^*(\cdot)$ é a função peso com respeito à distribuição polarizada $f_{\mathbf{X}_i}^*(\cdot)$, $H_i(\cdot)$ é a função indicadora do erro de decodificar \mathbf{c}_i , $i = 2, \dots, M$, ao invés de \mathbf{c}_1 . O estimador com valorização das amostras para p_i é definido como a média amostral

$$\hat{p}_i = \frac{1}{N_i} \sum_{k=1}^{N_i} H_i(g(\mathbf{x}_k)) w_i(\mathbf{x}_k) \quad (7.4)$$

em que $N_i \neq 0$ é o número de palavras-código utilizadas na simulação, e as amostras \mathbf{x}_k são obtidas a partir da distribuição polarizada $f_{\mathbf{X}_i}^*(\cdot)$. Como mostrado no Capítulo 6, o estimador 7.4 é despolarizado e tem variância dada por

$$\sigma_{\hat{p}_i}^2 = \frac{1}{N_i} \{ E [H_i(g(\mathbf{x})) w_i(\mathbf{x})] d\mathbf{x} - p_i^2 \} \quad (7.5)$$

$$= \frac{1}{N_i} \{ E_* [H_i(g(\mathbf{x})) w_i^2(\mathbf{x}) d\mathbf{x}] - p_i^2 \} \quad (7.6)$$

que pode ser obtida durante a simulação com o uso do estimador da variância amostral

$$\hat{\sigma}_{\hat{p}_i}^2 = \frac{1}{N_i} \left\{ \frac{1}{N_i} \sum_{k=1}^{N_i} H_i(g(\mathbf{x}_k)) w_i^2(\mathbf{x}_k) - \hat{p}_i^2 \right\}. \quad (7.7)$$

Deseja-se então encontrar a distribuição polarizada $f_{\mathbf{X}_i}^*(\cdot)$ que minimize a função custo $\sigma_{\hat{p}_i}^2$. Como introduzido no Capítulo 5, o problema de selecionar a distribuição polarizada pode ser dividido em duas partes. Inicialmente escolhe-se uma classe de densidades que satisfaz algumas condições gerais, e em seguida é selecionado um vetor de parâmetros \mathbf{r}_i que determina uma fdp específica pertencente a essa classe e que será efetivamente utilizada na simulação

A distribuição polarizada $f_{\mathbf{X}_i}^*(\mathbf{x})$ pode então ser escrita como $f_{\mathbf{X}_i}^*(\mathbf{x}; \mathbf{r}_i)$, em que é escrito explicitamente a dependência em relação aos parâmetros \mathbf{r}_i . Utilizando o mesmo procedimento de otimização de parâmetros utilizado no Capítulo 5, deseja-se encontrar \mathbf{r}_i de modo a minimizar a função custo $\sigma_{\hat{p}_i}^2$, Equação 7.6. Para simplificar a abordagem do problema, denotando $I(\mathbf{r}_i)$ como o valor esperado na Equação 7.6, minimizar a variância do estimador com valorização das amostras é o mesmo problema de minimizar a função

$$I(\mathbf{r}_i) = E[H_i(g(\mathbf{X}))w_i(\mathbf{X}; \mathbf{r}_i)] \quad (7.8)$$

$$= E_*[H_i(g(\mathbf{X}))w_i^2(\mathbf{X}; \mathbf{r}_i)]. \quad (7.9)$$

Devido às vantagens de simplicidade e generalidade, o método utilizado para minimizar $I(\cdot)$ em relação ao vetor \mathbf{r}_i será o método Gradiente Descendente Estocástico (GDE), introduzido na Seção 5.3.2.

O método baseia-se em deslocar o vetor de parâmetros \mathbf{r}_i na direção oposta ao gradiente da função $I(\cdot)$ dado por

$$\nabla_{\mathbf{r}_i} I(\mathbf{r}_i) = \int_{-\infty}^{\infty} H_i(g(\mathbf{x})) w_i(\mathbf{x}, \mathbf{r}_i) \nabla_{\mathbf{r}_i} w(\mathbf{x}, \mathbf{r}_i) f_{\mathbf{X}_i}^*(\mathbf{x}, \mathbf{r}_i) d\mathbf{x}. \quad (7.10)$$

O cálculo do gradiente acima é em geral tão difícil quanto encontrar a probabilidade de erro desejada, portanto durante a simulação usa-se uma estimativa despolarizada de $\nabla_{\mathbf{r}_i} I()$ dada por

$$\hat{\nabla}_{\mathbf{r}} I(\mathbf{r}) = \frac{1}{N} \sum_{i=1}^N H(g(\mathbf{x}_i)) w(\mathbf{x}_i, \mathbf{r}) \nabla_{\mathbf{r}} w(\mathbf{x}_i, \mathbf{r}), \quad \mathbf{x}_i \sim f_{\mathbf{X}}^*(\mathbf{x}, \mathbf{r}). \quad (7.11)$$

A implementação do algoritmo GDE envolve um número de simulações curtas do sistema, de modo que a cada simulação o gradiente de $I(\cdot)$ é estimado e um novo valor para o vetor de parâmetros \mathbf{r}_i é calculado utilizando o esquema iterativo

$$\mathbf{r}_i^{(m+1)} = \mathbf{r}_i^{(m)} - \delta^{(m)} \hat{\nabla}_{\mathbf{r}} I(\mathbf{r}_i^{(m)}), \quad m = 1, 2, \dots \quad (7.12)$$

em que $\delta^{(m)}$ é o tamanho do passo utilizado na m -ésima iteração da otimização, e é obtido experimentalmente. É importante notar que para o cálculo do m -ésimo vetor de parâmetros $\mathbf{r}_i^{(m)}$, uma simulação completa é realizada, ou seja, N_i símbolos são gerados e o modelo do sistema é aplicado. Portanto, as amostras aleatórias geradas durante a simulação poderão ser utilizadas para estimar a variância do estimador e a probabilidade de erro utilizando as Equações 7.7 e 7.4, respectivamente.

A formulação da simulação é indicada com uma cuidadosa seleção das palavras-código (centros de erro), que levam a erros de decodificação. Os centros de erro para um código específico são escolhidos de acordo com o peso de Hamming das palavras código, pois os erros mais prováveis de ocorrer (menor distância de Hamming) têm maior influência no valor da probabilidade de erro de decodificação.

A informação das palavras-código possíveis é obtida por inspeção da estrutura em treliça do código particular, ou simplesmente a partir da matriz geradora do código [28]. Então, uma simulação é realizada separadamente para cada centro de erro selecionado, Equação 7.1. Assumindo que a palavra-código transmitida \mathbf{c}_1 é a seqüência de todos os bits zero, as amostras do ruído que afetam os bits que diferem entre \mathbf{c}_1 e o centro de erro \mathbf{c}_i serão polarizados, por outro lado, as amostras do ruído que afetam os bits que são iguais não serão polarizadas, sendo então geradas a partir da distribuição original do ruído.

Sempre que um erro é detectado após a decodificação, o peso do erro particular é calculado como uma multiplicação dos pesos de todos os bits individuais. Obviamente, os bits que não foram polarizados terão peso unitário. Além disso, o gradiente do peso é calculado analiticamente para o vetor de amostras \mathbf{x}_k e o vetor de parâmetros \mathbf{r}_i . Como dito no Capítulo 5, encontrar uma expressão para o gradiente do peso é um requisito

para aplicar o método GDE, entretanto, a maioria das distribuições importantes em comunicações têm essa característica.

Quando um erro ocorre os valores obtidos para o peso $w_i(\mathbf{x}_k; \mathbf{r}_i)$ e o gradiente do peso $\nabla_{\mathbf{r}} w(\mathbf{x}_i, \mathbf{r})$ são multiplicados e adicionados ao contador, como mostra a Equação 7.11, de modo que após N_i iterações da simulação corrente é calculada a estimativa do gradiente da função custo $I(\cdot)$. Então, a Equação recursiva 7.12 é utilizada para obter o novo valor de \mathbf{r}_i .

Para exemplificar o número de simulações e iterações necessários para obter a estimativa da probabilidade de erro de decodificação p_c , admitindo que \mathbf{c}_1 é transmitido, obtém-se a estimativa da probabilidade de decodificação específica \hat{p}_i para todos os centros de erros selecionados. Cada estimativa \hat{p}_i é obtida por um processo iterativo objetivando a minimização da função custo $I(\cdot)$. Em outras palavras, um número de simulações curtas são realizadas para estimar p_i . Além disso, todas as simulações citadas acima são realizadas apenas para calcular a probabilidade de erro quando \mathbf{c}_1 é transmitido.

Em linhas gerais esse é o procedimento proposto nesse trabalho. Felizmente, vários sistemas com codificação de interesse têm características que simplificam o número de simulações necessário para estimar p_c . Sistemas com códigos lineares e decodificação com máxima verossimilhança em canais binários simétricos, têm probabilidade de erro de decodificação independente da palavra-código transmitida. Além disso, a probabilidade de erro específica é igual para erros de decodificação com a mesma distância de Hamming. Por exemplo, um código de bloco (9, 8) simples, com apenas um bit de paridade, tem um alfabeto de $M = 2^8 = 256$ palavras código, cada uma com nove bits. Calcular a probabilidade de erro específica para todas as palavras do código é impraticável. Entretanto, analisando a estrutura do código nota-se que há 32 palavras com distância 2, 126 com distância 4, 84 com distância 6, e 9 com distância 8. Portanto, para obter a probabilidade de erro p_c do exemplo acima é necessário calcular apenas quatro probabilidades de decodificação específicas p_i , uma para cada distância de Hamming existente.

Outra simplificação ainda pode ser usada, se a relação sinal-ruído empregada na simulação é alta, os erros com maior distância de Hamming são raros e portanto têm probabilidade muito pequena em relação às palavras com menor distância de Hamming. Assim, a soma para calcular p_c na Equação 7.1 pode ser truncada, diminuindo significativamente o número necessário de simulações para estimar a probabilidade de erro.

É comum observar nas simulações que ao estimar a probabilidade de erro específica p_i de uma palavra com distância de Hamming pequena, a ordem de grandeza do erro cometido na estimação pode ser muito maior em relação a probabilidade específica para uma palavra-código com distância de Hamming maior.

Há uma decisão importante na implementação que ainda não foi discutida. Dado que a palavra-código \mathbf{c}_1 é transmitida, na simulação para estimar a probabilidade de erro específica p_2 , o ruído é polarizado na direção do erro específico \mathbf{c}_2 , mas se um erro diferente ocorrer, por exemplo \mathbf{c}_3 , é necessário considerá-lo ou não? Uma alternativa seria ignorar o erro encontrado ou adicionar esse erro na estimação de p_3 para não polarizar a estimativa \hat{p}_2 . Um experimento realizado com polarização na direção de \mathbf{c}_2 está longe de ser ótimo para a estimação de p_3 , e como introduzido na Seção 4.2, a escolha de uma distribuição polarizada ineficiente pode tornar a variância do estimador igual ou até mesmo maior que a variância do estimador de Monte Carlo. Embora esse evento seja raro, se ocorrer poderá destruir a eficiência obtida com a valorização das amostras. Durante as simulações ilustradas nesse capítulo, a porcentagem de erros que diferem da direção da polarização é monitorada e será discutida posteriormente.

7.2 Desempenho do Código de Hamming (7, 4)

Nessa seção serão mostrados e comentados os resultados obtidos aplicando o método de valorização adaptativa das amostras para sistemas com codificação. O modelo do sistema empregado aqui é mostrado na Figura 6.1 e contém o código de Hamming (7, 4) e modulação BPSK. A avaliação de desempenho é feita para o canal AWGN.

7.2.1 Estrutura do Código

Códigos de Hamming é uma família de códigos de bloco lineares (n, k) em que $n = 2^m - 1$ e $k = n - m$, para $m \geq 3$. O código de Hamming (7, 4), correspondendo a $m = 3$, foi escolhido para análise devido a simplicidade e já ter sido estudado extensivamente por vários autores.

Sendo $\mathbf{b} = (b^{(1)}, b^{(2)}, \dots, b^{(k)})$ a seqüência de informação composta de k bits quaisquer, e $\mathbf{c} = (c^{(1)}, c^{(2)}, \dots, c^{(n)})$ a palavra do código correspondente a \mathbf{b} , o código é completamente descrito pela matriz geradora do código \mathbf{G} que relaciona \mathbf{b} e \mathbf{c} com a equação matricial

$$\mathbf{c} = \mathbf{b} \cdot \mathbf{G}. \quad (7.13)$$

Para o código de Hamming (7, 4), a matriz geradora do código é dada por

$$\mathbf{G} = \left[\begin{array}{ccc|ccc} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{array} \right]$$

Como a matriz \mathbf{G} acima tem as últimas quatro colunas formando uma matriz identidade, o código é dito ser sistemático, pois os bits de informação são transmitidos sem modificação em parte da palavra código ($c^{(4)} = b^{(1)}$, $c^{(5)} = b^{(2)}$, $c^{(6)} = b^{(3)}$ e $c^{(7)} = b^{(4)}$), e os bits restantes ($c^{(1)}$, $c^{(2)}$ e $c^{(3)}$) são referenciados como os bits de paridade do código e calculados pela regra de codificação estabelecida por \mathbf{G} , que determina a estrutura matemática do código.

O mapeamento realizado no processo de codificação é listado na Tabela 7.1. Para uma seqüência de bits de informação \mathbf{b} , a palavra código correspondente é obtida usando a Equação 7.13. Também é listado na Tabela 7.1 o peso de Hamming de cada palavra do código. Como pode-se notar, a menor distância de Hamming entre as palavras é $d_{\min} = 3$.

Admitindo que a palavra $\mathbf{c}_1 = (0, 0, 0, 0, 0, 0, 0)$ é transmitida, as palavras restantes (centros de erro) podem ser agrupadas de acordo com a distância de Hamming em relação a \mathbf{c}_1 , então tem-se 7 palavras com distância 3, 7 com distância 4, e 1 com distância 7. Como descrito anteriormente, a probabilidade de erro específica p_i é igual para palavras código de mesma distância de Hamming, o que reduz o número de simulações no processo de estimação. Portanto, apenas as palavras \mathbf{c}_2 , \mathbf{c}_3 e \mathbf{c}_{16} serão efetivamente utilizadas para obter as probabilidades de erro específicas p_2 , p_3 e p_{16} , respectivamente. Como o objetivo é encontrar uma estimativa para a probabilidade de erro de decodificação p_e , Equação 7.1, as probabilidades de erro específicas para as palavras-código restantes são obtidas pelas igualdades

$$\begin{aligned} p_4 &= p_5 = p_7 = p_9 = p_{11} = p_{14} = p_2 \\ p_6 &= p_8 = p_{10} = p_{12} = p_{13} = p_{15} = p_3 \end{aligned}$$

de modo que o somatório na Equação 7.1 é simplificado para

Tabela 7.1: Alfabeto do codificador de Hamming (7, 4).

Seqüência de informação	Palavra código	Peso de Hamming
0 0 0 0	0 0 0 0 0 0 0	0
0 0 0 1	1 0 1 0 0 0 1	3
0 0 1 0	1 1 1 0 0 1 0	4
0 0 1 1	0 1 0 0 0 1 1	3
0 1 0 0	0 1 1 0 1 0 0	3
0 1 0 1	1 1 0 0 1 0 1	4
0 1 1 0	1 0 0 0 1 1 0	3
0 1 1 1	0 0 1 0 1 1 1	4
1 0 0 0	1 1 0 1 0 0 0	3
1 0 0 1	0 1 1 1 0 0 1	4
1 0 1 0	0 0 1 1 0 1 0	3
1 0 1 1	1 0 0 1 0 1 1	4
1 1 0 0	1 0 1 1 1 0 0	4
1 1 0 1	0 0 0 1 1 0 1	3
1 1 1 0	0 1 0 1 1 1 0	4
1 1 1 1	1 1 1 1 1 1 1	7

$$p_c = 7 \cdot p_2 + 7 \cdot p_3 + p_{16}. \quad (7.14)$$

A probabilidade de erro de bit é calculada utilizando a Equação 6.17, que resulta

$$p_b = \frac{1}{4} \sum_{i=2}^{16} n_i(\mathbf{b}_1, \mathbf{b}_i) \cdot p_i, \quad (7.15)$$

em que $n_i(\mathbf{b}_1, \mathbf{b}_i)$ é o número de bits que diferem entre as seqüências de bits de informação \mathbf{b}_1 e \mathbf{b}_i .

O cálculo de p_c e p_b envolve três simulações separadas, cada uma com valorização das amostras específica, ou seja, as simulações para as estimativas p_2 , p_3 e p_{16} são realizadas polarizando a distribuição do ruído na direção de \mathbf{c}_2 , \mathbf{c}_3 e \mathbf{c}_{16} , respectivamente. Para polarizar o canal em direções distintas, usa-se o processo não-estacionário para o ruído descrito na Seção 6.2. Como exemplo, suponha que deseja-se estimar p_2 , então o ruído

é polarizado de modo que os bits da palavra código transmitida $\mathbf{c}_1 = (0, 0, 0, 0, 0, 0, 0)$ tenham mais chance de serem decodificados como $\mathbf{c}_2 = (1, 0, 1, 0, 0, 0, 1)$. Para implementar essa polarização é necessário modificar a distribuição do ruído apenas nos instantes em que os bits de \mathbf{c}_1 são diferentes dos de \mathbf{c}_2 (bits 1, 3 e 7). Quando os bits restantes são transmitidos, o ruído não é polarizado.

Na próxima seção serão mostrados os resultados obtidos com a simulação do código de Hamming (7, 4) para o canal AWGN.

7.2.2 Canal AWGN

O modelo para o canal AWGN é caracterizado por adicionar ruído gaussiano branco à forma de onda transmitida pelo canal de acordo com a relação sinal-ruído estabelecida. Denotando por $\mathbf{z} = (z^{(1)}, z^{(2)}, \dots, z^{(n)})$ as amostras do ruído equivalentes aos bits $c_1^{(1)}, c_1^{(2)}, \dots, c_1^{(n)}$ da palavra código transmitida \mathbf{c}_1 , respectivamente, e $z_*^{(k)}$, $k = 1, 2, \dots, n$ a amostra do ruído polarizada para o k -ésimo bit, a simulação da probabilidade de erro específica p_i , quando \mathbf{c}_i é decodificado ao invés de \mathbf{c}_1 , é implementada de modo que os bits em que \mathbf{c}_1 e \mathbf{c}_i diferem são corrompidos pelo ruído $z_*^{(k)}$, enquanto que o restante dos bits são corrompidos por $z^{(k)}$. Considerando as amostras do ruído independentes, a fdp conjunta do canal AWGN é dada por

$$f_{\mathbf{Z}}(z^{(1)}, z^{(2)}, \dots, z^{(n)}) = \prod_{k=1}^n f_N(z^{(k)}) \quad (7.16)$$

em que $f_N(\cdot)$ é a distribuição gaussiana com média nula e variância $\sigma_N^2 = N_0/2$, definida na Equação 4.13. Em uma simulação de Monte Carlo convencional a fdp conjunta $f_{\mathbf{Z}}(\cdot)$ é utilizada para obter as amostras do ruído que corrompem o sinal. Na simulação com valorização das amostras deseja-se obter uma distribuição para simulação diferente para cada palavra-código. Para simplificar a exposição, admitindo que apenas os primeiros K_i bits de \mathbf{c}_i são diferentes de \mathbf{c}_1 , a distribuição polarizada é dada por

$$f_{\mathbf{Z}}^*(z^{(1)}, z^{(2)}, \dots, z^{(n)}) = \left(\prod_{k=1}^{K_i} f_N^*(z^{(k)}) \right) \cdot \left(\prod_{k=K_i+1}^n f_N(z^{(k)}) \right) \quad (7.17)$$

em que $f_N^*(\cdot)$ é a distribuição gaussiana polarizada. Como discutido na Seção 4.3.1, essa distribuição é usualmente derivada de $f_N(\cdot)$ modificando ou a média ou a variância da

distribuição. Baseando-se nos resultados de [22], a polarização por translação da média (*Improved Importance Sampling - IIS*) é escolhida aqui por ter melhor desempenho. Portanto, a fdp gaussiana polarizada é dada por

$$f_N^*(z) = \frac{1}{\sqrt{2\pi}\sigma_N} e^{-(z-\mu^*)^2/2\sigma_N^2} \quad (7.18)$$

em que μ^* é a quantidade em que o valor médio da distribuição foi deslocado, e é referenciada como a **média polarizada**. Substituindo as definições 4.13 e 7.18 na Equação 7.17, tem-se

$$\begin{aligned} f_{Z_i}^*(z^{(1)}, z^{(2)}, \dots, z^{(n)}) &= \left(\prod_{k=1}^{K_i} \frac{1}{\sqrt{2\pi}\sigma_N} e^{-\frac{(z^{(k)}-\mu^*)^2}{2\sigma_N^2}} \right) \cdot \left(\prod_{k=K_i}^n \frac{1}{\sqrt{2\pi}\sigma_N} e^{-\frac{(z^{(k)})^2}{2\sigma_N^2}} \right) \\ &= \left(\frac{1}{\sqrt{2\pi}\sigma_N} \right)^n \cdot e^{-\sum_{k=1}^{K_i} \frac{(z^{(k)}-\mu^*)^2}{2\sigma_N^2}} \cdot e^{-\sum_{k=K_i}^n \frac{(z^{(k)})^2}{2\sigma_N^2}} \\ &= \left(\frac{1}{\sqrt{2\pi}\sigma_N} \right)^n \cdot e^{-\frac{1}{2\sigma_N^2} (\sum_{k=1}^{K_i} (z^{(k)}-\mu^*)^2 + \sum_{k=K_i}^n (z^{(k)})^2)}. \end{aligned} \quad (7.19)$$

A função peso $w_i(\cdot) = f_Z(\cdot)/f_{Z_i}^*(\cdot)$ utilizada para ponderar o contador de erros durante a simulação é obtida substituindo as Equações 7.16 e 7.17 na definição de $w_i(\cdot)$, como segue

$$\begin{aligned} w_i(z^{(1)}, z^{(2)}, \dots, z^{(n)}) &= \frac{\prod_{k=1}^n f_N(z^{(k)})}{\left(\prod_{k=1}^{K_i} f_N^*(z^{(k)}) \right) \cdot \left(\prod_{k=K_i}^n f_N(z^{(k)}) \right)} \\ &= \prod_{k=1}^{K_i} \frac{f_N(z^{(k)})}{f_N^*(z^{(k)})} \\ &= \prod_{k=1}^{K_i} \frac{\frac{1}{\sqrt{2\pi}\sigma_N} e^{-\frac{(z^{(k)})^2}{2\sigma_N^2}}}{\frac{1}{\sqrt{2\pi}\sigma_N} e^{-\frac{(z^{(k)}-\mu^*)^2}{2\sigma_N^2}}} \\ &= \prod_{k=1}^{K_i} e^{\frac{-(z^{(k)})^2 + (z^{(k)}-\mu^*)^2}{2\sigma_N^2}} \\ &= \prod_{k=1}^{K_i} e^{\frac{(\mu^*)^2 - 2\mu^* z^{(k)}}{2\sigma_N^2}} \\ &= \exp \left(\frac{K_i (\mu^*)^2 - 2\mu^* \left(\sum_{k=1}^{K_i} z^{(k)} \right)}{2\sigma_N^2} \right). \end{aligned} \quad (7.20)$$

O gradiente da função peso é necessário para estimar o gradiente da função custo $I(\cdot)$, como mostrado na Equação 7.11. O vetor de parâmetros para esse esquema de polarização é unidimensional, pois o único parâmetro da distribuição gaussiana polarizada é a média μ^* . O gradiente da função peso é então obtido derivando a Equação 7.20 em relação a μ^*

$$\begin{aligned} \nabla_{\mu^*} w_i(z^{(1)}, z^{(2)}, \dots, z^{(n)}) &= \frac{d}{d\mu^*} \exp\left(\frac{K_i(\mu^*)^2 - 2\mu^* \left(\sum_{k=1}^{K_i} z^{(k)}\right)}{2\sigma_N^2}\right) \\ &= \frac{K_i\mu^* - \left(\sum_{k=1}^{K_i} z^{(k)}\right)}{\sigma_N^2} \cdot w_i(z^{(1)}, z^{(2)}, \dots, z^{(n)}) \end{aligned} \quad (7.21)$$

As Equações 7.20 e 7.21 são substituídas na Equação 7.11 para obter o estimador do gradiente da função custo $I(\cdot)$. A simulação é então realizada seguindo o processo iterativo estabelecido pela Equação 7.12.

Essa abordagem foi então aplicada ao sistema com código de Hamming (7, 4) em um canal AWGN com relação sinal-ruído de 5dB. O valor inicial escolhido para a média polarizada μ^* foi zero, de modo que a primeira simulação é uma simulação de Monte Carlo convencional. Após a primeira simulação o gradiente é utilizado para direcionar o valor de μ^* na tentativa de minimizar a variância do estimador. Essa mesma configuração foi aplicada para estimar p_2 , p_3 e p_{16} , e os resultados obtidos são listados na Tabela 7.3.

A Figura 7.1 mostra o deslocamento da média da distribuição polarizada no decorrer das simulações para a estimação de p_2 . Na primeira simulação a média é nula, e o passo é $\delta_0 = 1000$, quando o próximo valor da média polarizada μ^* é calculado, o produto $\delta^{(n)} \cdot \hat{\nabla}_{\mu^*} I(\mu^*)$ na Equação 7.11 é alto e desloca a média no caminho do valor ótimo rapidamente, entretanto, a partir da segunda simulação o gradiente é menor e a convergência é lenta devido ao passo fixo escolhido. Observando a Figura 7.1, na simulação 76 há uma descontinuidade da curva, essa mudança de direção ocorreu devido a mudança no passo escolhido. A partir da simulação 76 o passo foi alterado para $\delta_{76} = 10000$. Observando os dados obtidos na simulação, o algoritmo converge para $\mu^* = -1.38$, enquanto que o estimador IIS usa a polarização $\mu_{IIS}^* = -2.51$.

O salto inicial encontrado na curva da média polarizada μ^* pode ser visto na curva do gradiente $\hat{\nabla}_{\mu^*} I(\mu^*)$ ilustrada na Figura 7.2. O gradiente diminui em valor absoluto

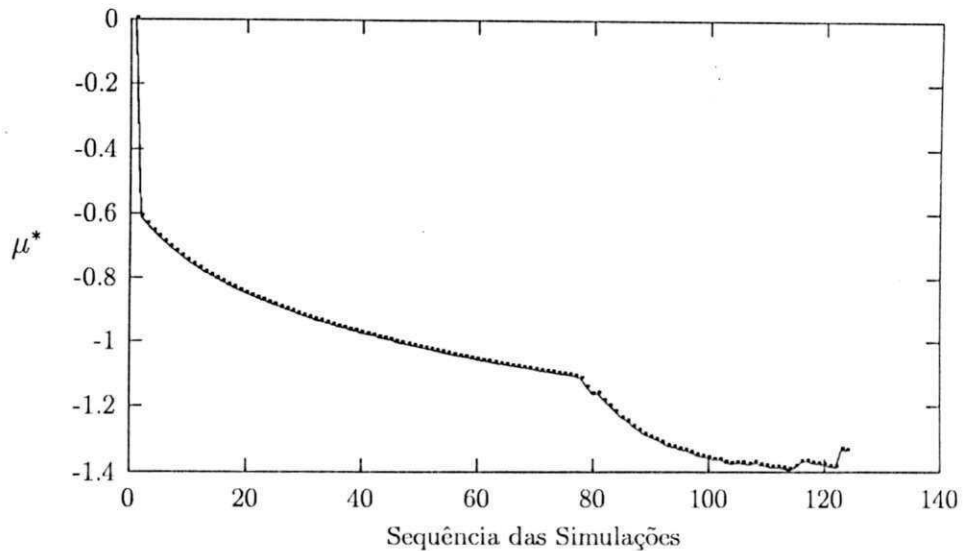


Figura 7.1: Curva do gradiente na simulação adaptativa.

no decorrer das simulações, até que começa a oscilar em torno da origem, Figura 7.3. Nesse instante, se o valor da variância ainda não é pequeno suficiente para finalizar as simulações, é preciso diminuir o passo ou aumentar o comprimento da simulação.

Visando comparar essa abordagem com o método IIS, a simulação adaptativa também foi iniciada com a média polarizada igual à média utilizada em IIS, ou seja, $\mu_0^* = \mu_{IIS}^*$. Como mostra a Figura 7.4, o gradiente deslocou a média polarizada diminuindo o nível de polarização aplicado. Esse resultado já era esperado, pois para que o erro de decodificação c_2 ocorra não é necessário que os três bits que diferem entre c_2 e c_1 sejam diferentes antes da decodificação. Se apenas dois desses bits forem errados, é suficiente para que a decodificação seja feita em favor de c_2 e não c_1 .

Outro resultado que demonstra essa propriedade é obtido observando o número de erros diferentes de c_2 que ocorrem durante a simulação. Usando IIS, 1.8% das iterações resultaram em erros de decodificação diferentes de c_2 , enquanto que para o método adaptativo esse número é apenas 0.7%. Embora a ordem de grandeza dessas porcentagens seja pequena, nas simulações de p_3 e p_{16} essa diferença torna-se mais aparente, como lista a Tabela 7.2. Esse resultado demonstra que a polarização utilizada no método IIS é menos direcionada a c_2 que no método adaptativo.

Para quantificar a melhoria de desempenho obtida com o método adaptativo, a Tabela 7.3 lista as estimativas obtidas para p_2 , p_3 e p_{16} , e a variância para os estimadores

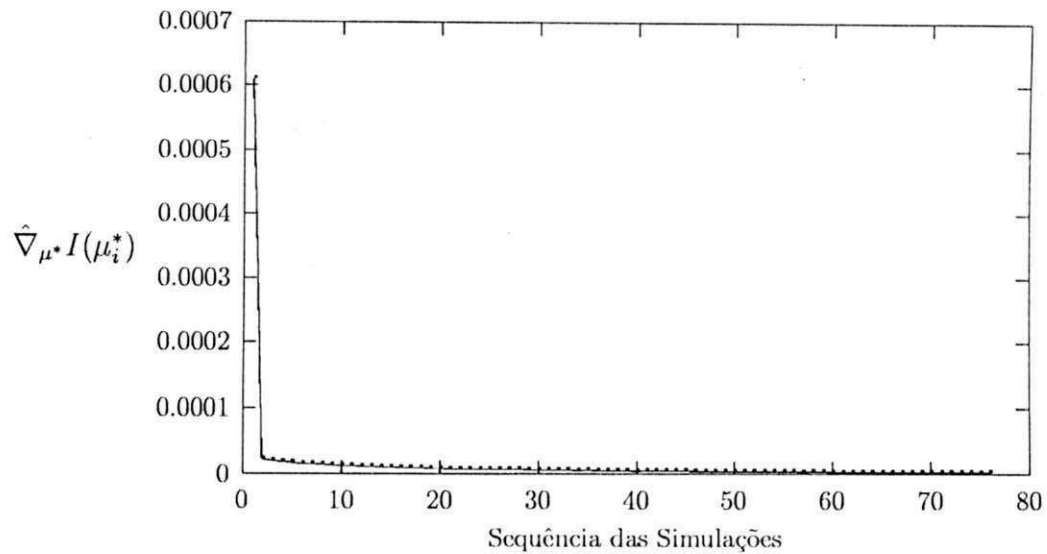


Figura 7.2: Convergência da média na simulação adaptativa.

Tabela 7.2: Porcentagem de palavras códigos ignoradas na simulação.

	Estimador IIS	Estimador Adaptativo
\hat{p}_2	1.8%	0.7%
\hat{p}_3	38.5%	23.2%
\hat{p}_{16}	87.5%	78.5%

de Monte Carlo, IIS e adaptativo. Os valores exatos e os parâmetros da simulação são listados na Tabela 7.4.

Analisando a Tabela 7.3 nota-se que as variâncias do método adaptativo são menores que as variâncias dos outros métodos. Observando os valores ótimos μ_{opt}^* nota-se que as palavras-código com maior distância de Hamming têm maior polarização. Esse resultado pode ser utilizado para escolher o ponto inicial para μ^* a partir do valor ótimo obtido para a distância de Hamming imediatamente inferior ou superior.

Tabela 7.3: Resultados da simulação do sistema com código de Hamming (7, 4).

	Monte Carlo		Estimador IIS		Método Adaptativo	
	\hat{p}_{MC}	$\hat{\sigma}_{MC}^2$	\hat{p}_{IIS}	$\hat{\sigma}_{IIS}^2$	\hat{p}_{AIS}	$\hat{\sigma}_{AIS}^2$
\hat{p}_2	$1,09 \cdot 10^{-4}$	$1,09 \cdot 10^{-10}$	$1,03 \cdot 10^{-4}$	$3,51 \cdot 10^{-11}$	$1,049 \cdot 10^{-4}$	$1,09 \cdot 10^{-12}$
\hat{p}_3	—	—	$7,95 \cdot 10^{-7}$	$2,3 \cdot 10^{-15}$	$8,43 \cdot 10^{-7}$	$6,00 \cdot 10^{-16}$
\hat{p}_{16}	—	—	$2,46 \cdot 10^{-13}$	$1,16 \cdot 10^{-27}$	$2,89 \cdot 10^{-13}$	$1,10 \cdot 10^{-27}$

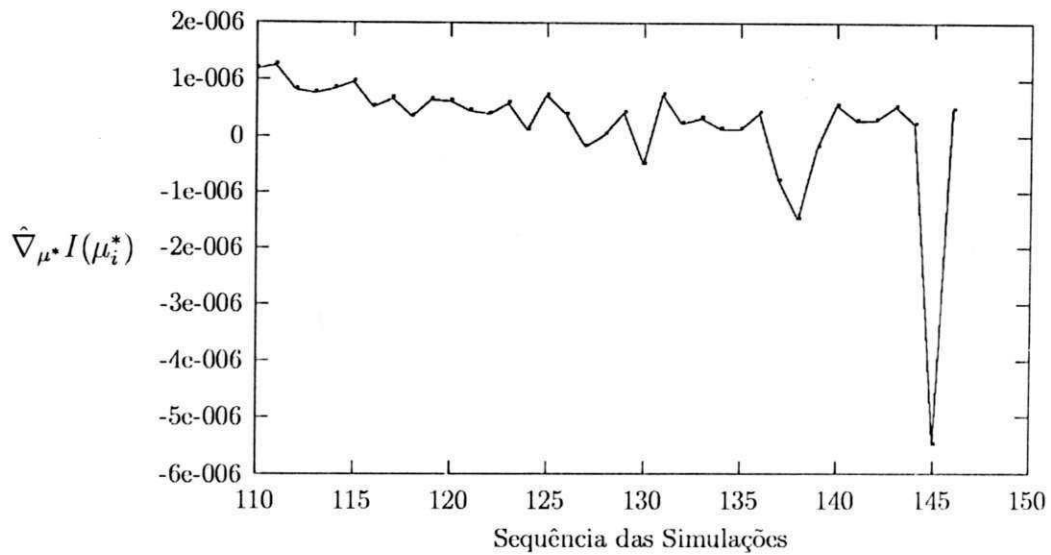


Figura 7.3: Oscilação do gradiente em torno da origem.

Tabela 7.4: Parâmetros utilizados nas simulações.

	Valores exatos p_i	Estimador IIS μ^*	Estimador Adaptativo μ_{opt}^*
\hat{p}_2	$1,05 \cdot 10^{-4}$	-2,51	-1,36
\hat{p}_3	$8,45 \cdot 10^{-7}$	-2,51	-1,88
\hat{p}_{16}	$3,25 \cdot 10^{-13}$	-2,51	-2,85
N_i	10^6	10^6	10^6

Durante as simulações foi observado que os valores ótimos μ_{opt}^* são independentes do comprimento da simulação. Esse resultado é interessante pois pode-se utilizar simulações curtas para estimar μ_{opt}^* , e em seguida realizar uma simulação mais longa e portanto mais precisa.

Observando a Tabela 7.4, nota-se que a média obtida para estimar p_{16} no método adaptativo é maior, em termos absolutos, que a média polarizada utilizada em IIS. Isso ocorre pois no estimador IIS apenas metade dos bits são invertidos, de modo que a palavra código com distância de Hamming 7 dificilmente é decodificada. Como mostra a Tabela 7.2, para o estimador IIS 87.5% das palavras-código decodificadas com erro não foram c_{16} , e com a polarização utilizada pelo método adaptativo esse valor é reduzido para 78.5%.

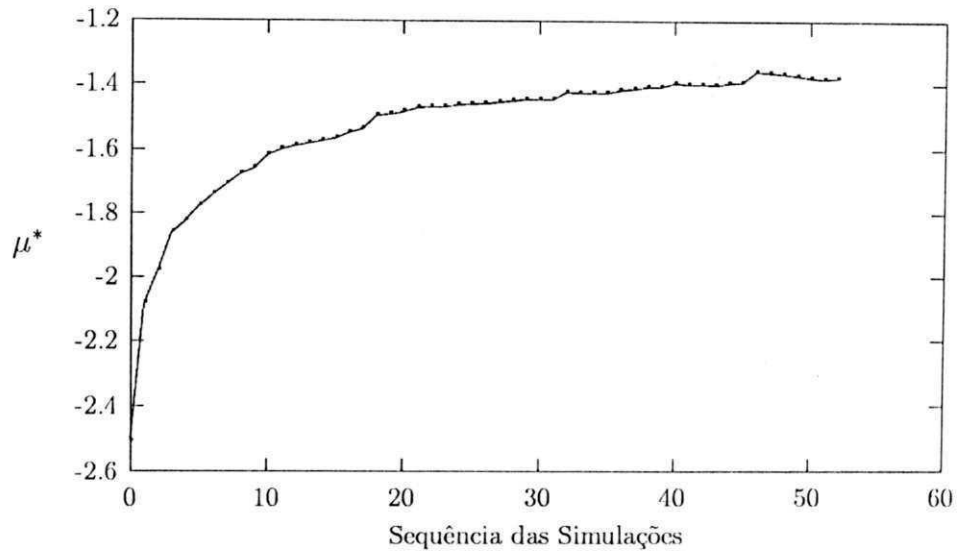


Figura 7.4: Convergência da média polarizada a partir do método IIS.

7.3 Conclusões

Os resultados apresentados nesse capítulo demonstraram que a valorização das amostras e o algoritmo de otimização GDE, podem ser utilizados em conjunto para encontrar a direção de polarização que minimiza a variância do estimador em aplicações envolvendo sistemas com codificação.

A aplicação dessa abordagem apresentou ganho de desempenho em relação ao método IIS, e demonstrou que para polarizar a distribuição não é suficiente escolher os bits da palavra-código que terão polarização, mas também é necessário utilizar diferentes parâmetros de polarização para palavras-código com distância de Hamming diferentes.

Uma vantagem desse método é a simplicidade, pois após calcular o gradiente da função peso, o esquema iterativo para o algoritmo GDE é facilmente implementado por um único laço de programação.

No próximo capítulo serão apresentadas as conclusões gerais dessa dissertação.

Capítulo 8

Conclusões

A aplicação de valorização das amostras em problemas de sistemas de comunicações tem sido amplamente estudada e diversas técnicas têm sido propostas. Entretanto, até o momento as aplicações envolvendo sistemas com codificação apenas modificavam os métodos tradicionais para adequá-los às particularidades envolvidas no processo de codificação e decodificação, que dificultam o uso de valorização das amostras devido à memória e dimensionalidade do sistema.

Os métodos propostos em [23] e [22] utilizaram distribuições de polarização a partir do método IIS, ou utilizam o método força-bruta para encontrar os parâmetros da polarização. Nenhuma dessas abordagens prevê o uso de técnicas de otimização de parâmetros para melhorar o desempenho da polarização.

O presente trabalho propôs a junção de duas técnicas para a solução desse problema: a abordagem apresentada em [22] para adequar a valorização das amostras a sistemas com codificação, e o método GDE para otimizar os parâmetros da distribuição polarizada, melhorando assim o desempenho da simulação.

Como demonstrado no Capítulo 7, o método de polarização IIS não é ótimo para sistemas com codificação, pois a direção da polarização não é eficiente. Nas aplicações em que IIS é ótimo, sua utilização força a simulação a obter erros na metade das iterações realizadas. Entretanto, em sistemas com codificação o comportamento de IIS não é o mesmo, pois dependendo da distância de Hamming da palavra código, aplicar um parâmetro de polarização fixo, significa que a polarização relativa entre as palavras-código é diferente. No método proposto esse problema é minimizado, pois como demonstrado com os resultados obtidos na Seção 7.2.2, as palavras-código com menor distância de Hamming tiveram polarização ótima menor que a utilizada em IIS,

enquanto que as palavras com maior distância de Hamming tiveram polarização maior.

O ganho de desempenho obtido no exemplo mostrado na Seção 7.2.2 foi de ordem 10. Embora esse valor não seja significativo, espera-se que o ganho seja maior para sistemas em que seja difícil encontrar os parâmetros da polarização, pois o desconhecimento das regiões de erro para os processos aleatórios envolvidos não é requisito para a aplicação desse novo método.

8.1 Perspectivas para trabalhos futuros

Como continuação das atividades de pesquisa realizadas podem ser citadas as seguintes sugestões:

- Aplicar o método adaptativo a sistemas com codificação convolucional e codificação turbo;
- Aplicar o método adaptativo a outros modelos de canais de comunicação, como por exemplo o canal com desvanecimento Rayleigh;
- Investigar a convergência do algoritmo GDE em relação ao comprimento da simulação que estima o gradiente;
- Consolidar os resultados obtidos para outros códigos de bloco e descrever os parâmetros ótimos em termos de equações algébricas.

Bibliografia

- [1] Michel C. Jeruchim, Philip Balaban, and K. Sam Shanmugan. *Simulation of Communication Systems*. New York: Plenum, 1992.
- [2] K. S. Shanmugan and P. Balaban. *A modified Monte Carlo simulation technique for the evaluation of error rate in digital communications systems*. *IEEE Trans. Commun.*, vol. COM-28, no. 11, pp. 1916-1924, Nov. 1980.
- [3] Michel C. Jeruchim. *Techniques for estimating the bit error rate in the simulation of digital communications systems*. *IEEE J. Select. Areas Commun.*, vol. SAC-2, no. 1, pp. 153-170, Jan. 1984.
- [4] Bruce R. Davis. *An improved importance sampling method for digital communication system simulations*. *IEEE Trans. Commun.*, vol. COM-34, no. 7, pp. 715-719, July 1986.
- [5] D. Lu and K. Yao. *Improved importance sampling technique for efficient simulation of digital communication systems*. *IEEE J. Select. Areas Commun.*, vol. 6, no. 1, pp. 67-75, Jan. 1988.
- [6] Adrian Bohdanowicz. *On Efficient BER Evaluation of Digital Communication Systems via Importance Sampling*. *Ubiquitous Communications Program*, Faculty of Information Technology and Systems, Delft University of Technology, The Netherlands.
- [7] Peter M. Hahn and Michel C. Jeruchim. *Developments in the Theory and Application of Importance Sampling*. *IEEE Trans. Commun.*, vol. COM-35, no. 7, pp. 706-714, July 1987.

- [8] J.-C. Chen, D. Lu, J. S. Sadowsky, and K. Yao. *On Importance Sampling in Digital Communications - Part I: Fundamentals*. *IEEE J. Select. Areas Commun.*, vol. 11, no. 3, pp. 289-299, April 1993.
- [9] Peter J. Smith, Mansoor Shafi, and Hongsheng Gao. *Quick Simulation: A Review of Importance Sampling Techniques in communications Systems*. *IEEE J. Select. Areas Commun.*, vol. 15, no. 4, pp 597-613, May 1997.
- [10] Robert J. Wolfe, Peter M. Hahn, and Michel C. Jeruchim. *On Optimum and Suboptimum Biasing Procedures for Importance Sampling in Communication Simulation*. *IEEE Trans. Commun.*, vol. 38, no. 5, pp. 639-647, May 1990.
- [11] D. Lu and K. Yao. *Estimation Variance Bounds of Importance Sampling Simulations in Digital Communication Systems*. *IEEE Trans. Commun.*, vol 39, no. 10, pp. 1413-1417, Oct. 1991.
- [12] Michael Devetsikiotis and J. Keith Townsend. *An Algorithmic Approach to the Optimization of Importance Sampling Parameters in Digital Communication System Simulation*. *IEEE Trans. Commun.*, vol. 41, no. 10, pp. 1464-1473, Oct. 1993.
- [13] W. A. Al-Qaq, Michael Devetsikiotis, and J. Keith Townsend. *Stochastic Gradient Optimization of Importance Sampling for the Efficient Simulation of Digital communication Systems*. *IEEE Trans. Commun.*, vol. 43, no. 12, pp. 2975-2985, Dec. 1995.
- [14] W. A. Al-Qaq and J. Keith Townsend. *A Stochastic Importance Sampling Methodology for the Efficient Simulation of Adaptive Systems in Frequency Nonselective Rayleigh Fading channels*. *IEEE J. Select. Areas Commun.*, vol. 15, no. 4, pp. 614-625, May 1997.
- [15] Michel C. Jeruchim, Peter M. Hahn, Kevin P. Smyntek, and Robert T. Ray. *An Experimental Investigation of Conventional and Efficient Importance Sampling*. *IEEE Trans. Commun.*, vol 37, no. 6, pp. 578-587, June 1989.
- [16] J. S. Sadowsky. *On the optimality and stability of exponential twisting in Monte Carlo estimation*. *IEEE Trans. Inform. Theory*, vol. 39, no. 1, pp 119-128, Jan. 1993.

- [17] K. Ben Letaief and J. S. Sadowsky. *Computing bit-error probabilities for avalanche photodiode receivers by large deviations theory*. *IEEE Trans. Inform. Theory*, vol. 38, pp 1162-1169, Mar. 1992.
- [18] K. Ben Letaief. *Performance analysis of digital lightwave systems using efficient computer simulation techniques*. *IEEE Trans. Commun.*, vol. 43, pp 240-251, Feb. 1995.
- [19] R. K. Bahr and J. A. Bucklew. *Quick simulation of detector error probabilities in the presence of memory and nonlinearity*. *IEEE Trans. Commun.*, vol. 41, pp 1610-1617, Nov. 1993.
- [20] David Remondo Bueno. *Performance Evaluation of Communication Systems via Importance Sampling*. Thesis University of Twente, Enschede, The Netherlands, 2000.
- [21] W. Zhuang. *Adaptive importance sampling for fading channels*. *Electronics Letters*, vol. 30, no. 14, pp. 1114-1116, July 1994.
- [22] S. Bellini and C. Molinari. *Efficient Simulation of Multilevel and Coded Transmission Systems*. *IEE Proceedings*, vol 140, no. 5, pp. 365-371, Oct. 1993.
- [23] John s. Sadowsky. *A New Method for Viterbi Decoder Simulation Using Importance Sampling*. *IEEE Trans. Commun.*, vol 38, no. 9, pp. 1341-1351, Oct. 1993.
- [24] K. Ben Letaief, K. Muhammad, and J. S. Sadowsky. *Fast simulation of DS/CDMA with and without coding in multipath fading channels*. *IEEE J. Select. Areas Commun.*, vol 15, pp. 626-639, May 1997.
- [25] Geoffrey C. Orsak. *Efficient Importance Sampling Techniques for Simulation of Multiuser Communication Systems*. *IEEE Trans. Commun.*, vol 40, no. 6, pp. 1111-1118, Jun. 1992.
- [26] Norman C. Beaulieu. *An investigation of Gaussian tail and Rayleigh tail density functions for importance sampling digital communication system simulation*. *IEEE Trans. Commun.*, vol. 38, no. 9, pp. 1288-1292, Sept. 1990.
- [27] Armin Dekorsy and Volker Kühn. *Aplicability of Importance Sampling for Rayleigh Fading Mobile Radio Channels*. Department of Communication Engineering, University of Paderborn, Germany.

- [28] S. S. Haykin. *Digital Communications*. John Wiley & Sons, 1988. ISBN 0-471-62947-2.
- [29] J. G Proakis. *Digital Communications*. McGraw-Hill, New York, 1989.
- [30] Athanasios Papoulis. *Probability, Random Variables, and Stochastic Processes*. Third Edition, McGraw-Hill, 1991.
- [31] Morris H. DeGroot and Mark J. Schervish. *Probability and Statistics*. Third Edition, Addison-Wesley, 2002.
- [32] Ian F. Blake. *An Introduction to Applied Probability*. University of Waterloo. John Wiley & Sons, New York, 1979.
- [33] William H. Press, Brian P. Flannery, Saul A. Teukolsky and William T. Vetterling. *Numerical Recipes in C, The Art of Scientific Computing*. Cambridge University Press. New York, 1988.
- [34] R. W. Hamming. *Numerical Methods for Scientists and Engineers*. Dover Publications, Inc., New York.
- [35] Germund Dahlquist. *Numerical Methods*. Prentice-Hall, 1974.
- [36] J. A. Bucklew. *Large Deviation Techniques in Decision, Simulation and Estimation*. John Wiley & Sons, Inc., New York, U.S.A., 1990.
- [37] Hallyson L. M. Melo. *Sistemas de Terceira Geração. Relatório da Disciplina Comunicações Móveis*. Laboratório de Comunicações/DEE-UFCEG, Campina Grande-Pb, Fev. 2000.