LIVRO DE MINICURSOS

Simpósio Brasileiro
de Telecomunicações e
Processamento de Sinais 2019

REALIZAÇÃO

SBrT

editora IFPB

Livro de Minicursos
SBRT 2019

Editora

Instituto Federal de Ensino, Ciência e Tecnologia da Paraíba – IFPB

Organização

Diego Barreto Haddad (CEFET/RJ)
Edmar Candeia Gurjão (UFCG)
Lisandro Lovisolo (UERJ)

Realização

Sociedade Brasileira de Telecomunicações – SBrT

Petrópolis – RJ
29 de Setembro a 02 de Outubro de 2019

Organização do SBrT 2019

Coordenação Geral
Lisandro Lovisolo (UERJ)

Coordenação Técnica
Eduardo Antônio de Barros da Silva (UFRJ)

Coordenadores Comitês Técnicos
Redes de Computadores
Marcelo Gonçalves Rubinstein (UERJ)
Processamento de Sinais
Paulo A. A. Esquefe (LNCC)
Comunicações sem Fio
Wallace A. Martins (UFRJ)
Antenas e Propagação
Maurício H. Dias (CEFET/RJ)

Coordenação da Sessão de Demonstrações
Gabriel Matos Araújo (CEFET/RJ)
Amaro Azevedo de Lima (CEFET/RJ)

Coordenação de Minicursos
Diego Barreto Haddad (CEFET/RJ)

Coordenação Local do Evento
Felipe da Rocha Henriques (CEFET/RJ)

Coordenação de TI e Inscrições
Alexandre Sztajnberg (UERJ)

Coordenação de Captação de Recursos
Lisandro Lovisolo (UERJ)
Felipe da Rocha Henriques (CEFET/RJ)
Michel P. Tcheou (UERJ)

Coordenação Financeira
Michel P. Tcheou (UERJ)

Coordenação da Competição Técnico-Científica
David Fernandes Cruz Moura (CETEX)

Liasons Internacionais
Países de língua espanhola:
Cecilia Galarza (CONYCET e UBA, Argentina)
Países de língua portuguesa:
Fernando Pereira (IST, Portugal)

Diretoria da SBrT

Presidente
Charles Casimiro Cavalcante

Vice-Presidente de Atividades Técnicas
Cecilio José Lins Pimentel

Vice-Presidente de Finanças
Marcello Luiz Rodrigues de Campos

Vice-Presidente de Desenvolvimento e Difusão
Cristiano Magalhães Panazio

Vice-Presidente de Relações Externas
Ugo Silva Dias

Membros do Conselho
Eduardo Antônio Barros da Silva
João César Moura Mota
João Marcos Travassos Romano
Lisandro Lovisolo
Paulo Cardieri
Edmar Candeia Gurjão - suplente
Felipe Rudge Barbosa - suplente

Instituto Federal de Educação Ciência e Tecnologia da Paraíba

Reitor
Cícero Nicácio do Nascimento Lopes

Pró-Reitora de Pesquisa, Inovação e Pós-Graduação
Silvana Luciene do Nascimento Cunha Costa

Editora IFPB

Diretor Executivo
Carlos Danilo Miranda Regis

Capa
Adino Bandeira

Dados Internacionais de Catalogação na Publicação - CIP

L788 Livro de minicursos SBRT 2019

Livro de minicursos SBRT 2019 / Diego Barreto Haddad; Edmar Candeia Gurjão; Lisandro Lovisolo (Orgs.). – João Pessoa : IFPB, 2019.
168 p. : il.
E-book (PDF)
ISBN: 978-85-5449-032-4

Textos originalmente apresentados no Simpósio Brasileiro de Telecomunicações e Processamento de Sinais, 2019.
1. Telecomunicações. 2. Processamento de sinais. I. Título.

CDU 621.391

Ficha Catalográfica elaborada pelo Departamento de Bibliotecas do IFPB - DBIBLIO

Prefácio

As últimas décadas presenciaram mudanças significativas tanto na indústria quanto na pesquisa associada às telecomunicações e ao processamento de sinais. Tal dinâmica concorre para a conhecida transição entre a Era Industrial e a que alguns autores prenunciam, apressadamente ou não, como a "Era da Informação". Realizado entre os dias 29 de setembro e 2 de outubro de 2019 na cidade de Petrópolis, Rio de Janeiro, o XXXVII Simpósio Brasileiro de Telecomunicações e Processamento de Sinais (SBrT 2019), com a apresentação de um pouco mais de duas centenas de artigos rigorosamente selecionados (dentre mais de quatrocentos submetidos), é um testemunho robusto deste incessante desenvolvimento tecnológico.

Neste simpósio, experientes pesquisadores ministraram sete minicursos, devidamente aprovados por uma judiciosa seleção, a qual priorizou tanto a relevância dos conteúdos quanto a sua qualidade técnica. Este livro, ao longo de seus seis capítulos, pretende disseminar parte do conhecimento tão generosamente compartilhado nestes minicursos. Cada um dos capítulos aprofunda e sistematiza o conteúdo da maioria dos minicursos respectivos. Por óbvio, as informações neles contidas são de completa responsabilidade de seus autores.

O primeiro capítulo (intitulado "Fundamentos da Geoestatística e Kriging aplicados à Mapas de Ambiente de Rádio") se reveste de um caráter multidisciplinar, já quase ubíquo em desafios nas telecomunicações. Conceitos de processos aleatórios espaciais originalmente empregados em geofísica, dentre os quais o preditor espacial "Kriging", são mobilizados para modelar o ambiente de rádio-propagação. Tal ferramenta provê uma formulação analítica para a geração e a modelagem do que se tem chamado de "mapas de ambiente de rádio", os quais têm atraído atenção crescente na área de sistemas de comunicação sem fio.

O segundo capítulo também aborda um tema de grande aplicabilidade em sistemas de comunicação sem fio: as antenas "phased array". Neste capítulo, os conceitos básicos de arranjos de antenas e da construção de seus respectivos diagramas de irradiação são didaticamente descritos. As principais janelas empregadas na redução de efeitos adversos oriundos do vazamento espectral são devidamente formuladas. Ademais, ferramentas comumente adotadas quando tais janelamentos não constituem uma alternativa viável são também descritas. Métricas utilizadas para a formação de feixes adaptativos são motivadas e detalhadas, sendo ademais

comparadas às principais arquiteturas e geometrias de arranjos de antenas. Generalizações dos modelos obtidos para sinais de banda larga são efetuadas, e importantes tópicos na prática - tais como erros, tolerância, calibração e alinhamento - são abordados de modo preciso e não menos fundamentado. Por fim, algumas importantes aplicações biomédicas e aeroespaciais são elencadas.

Outro desafio importante na área de comunicações sem fio reside na pesquisa e desenvolvimento de tecnologias para localização e posicionamento de dispositivos. Tais tecnologias têm sido consideradas chave e capazes de apresentar repercussões importantes seja no desenvolvimento, seja no aprimoramento das chamadas "cidades inteligentes" e da "internet das coisas". Assim, o terceiro capítulo deste livro, intitulado "Aprendizado de Máquina Aplicado a Localização de Usuários em Redes sem Fio: Oportunidades e Desafios", descreve os principais parâmetros de sinal adotados em redes sem fio para propósitos de localização, bem como as principais categorias dos sistemas de localização. As técnicas básicas empregadas para localização - a saber, lateração e fingerprinting (correlação de assinaturas) - são elencadas e ferramentas de inteligência artificial - tais como vizinhos mais próximos e máquinas de vetor suporte - são adotadas para modelar as dinâmicas não lineares inerentes aos parâmetros dos sinais de rádio-frequência, bem como para contornar o fato de que a quantidade disponível de medições dos parâmetros costuma ser deveras limitada, na prática.

O quarto capítulo - intitulado "Fundamentals and Techniques for the Localization of a Sensor and the Mapping of an Environment Using Videos" - trata de técnicas que efetuam simultaneamente a localização e o mapeamento (SLAM, do inglês simultaneous localization and mapping) de um ambiente por meio de sensores visuais. O formalismo da álgebra de Lie e da geometria projetiva são empregados para resolver desafios inerentes à localização e o mapeamento por meio de sensores visuais monoculares. A resolução de tais desafios é essencial para as mais diversas aplicações, dentre as quais destacam-se a direção autônoma, a realidade virtual e diversas aplicações da robótica.

Tanto a análise de variáveis latentes quanto o problema da separação de sinais são contemplados no quinto capítulo ("Separação de Sinais e Análise de Variáveis Latentes: Fundamentos e Tendências"), o qual delineia a evolução do estado da arte nestes campos. As vantagens da adoção de estatísticas de ordem superior, a aplicabilidade da análise de componentes principais, a importância da análise de componentes esparsos e a flexibilidade das técnicas de fatoração de matrizes não negativas são

tópicos devidamente enfatizados e discutidos. Como resultado, tem-se um proveitoso panorama desta área da qual, a despeito de já poder ser considerada madura, não seria de todo imprudente esperar numerosas inovações nos próximos anos.

O sexto e último capítulo, intitulado “Manual de Construção e Montagem do Cansat”, contempla a construção de pequenos satélites, mais especificamente do tipo Cansat. Importa notar que relevantes aspectos práticos da construção desses satélites são discutidos. A relevância deste tópico para a academia é significativa, dado que as dimensões reduzidas destes satélites promovem sua facilidade de uso e de lançamento. Assim, permite-se que universidades e instituições científicas, tradicionalmente alijadas dos dispendiosos recursos necessários ao desenvolvimento de satélites convencionais, possam efetuar contribuições tecnológicas no tema.

Por fim, agradecemos aos proponentes dos minicursos a submissão de material de grande qualidade técnica e o grande denodo demonstrado no cumprimento de prazos restritos. Gostaríamos ademais de manifestar nosso desejo de que as repercussões positivas deste empreendimento na pesquisa, na inovação e no desenvolvimento do país sejam ao menos proporcionais ao investimento demandado pela escrita e pela edição deste livro. Satisfeita tal aspiração, descabe exigir mais.

Diego Barreto Haddad
Coordenação de Minicursos

Sumário

CAPÍTULO

1 Fundamentos da Geoestatística e Kriging aplicados à Mapas de Ambiente de Rádio

Ricardo Augusto (Instituto Nacional de Telecomunicações, Inatel)

1.1 Introdução

O uso dos mapas de ambiente de rádio (REM - *Radio Environment Map*) tem despertado o interesse científico na área de sistemas de comunicações sem fio [1-3],[6-10]. Parte disto ocorre em razão da forma como o REM utiliza as informações de geolocalização e os potenciais benefícios desse mapa para os sistemas de comunicações sem fio, especialmente para os processos de planejamento e otimização de cobertura, importantes na área de comunicações móveis [7-8]. De fato, o impacto do conhecimento sobre informações geolocalizadas associadas aos sinais de rádio experimenta um crescimento significativo, atingindo diferentes campos das áreas das comunicações e navegação (e.g., futuras gerações de redes móveis, transportes inteligentes e redes de sensores) [4-5].

Neste contexto, as pesquisas científicas sobre a geração e a utilização dos mapas são essenciais para que o REM possa ser efetivamente inserido nas aplicações de comunicações sem fio. Este trabalho tem enfoque no aspecto de geração do REM, caracterizado como um problema de predição espacial sobre o ambiente de rádio. Especificamente, o objetivo deste trabalho consiste em apresentar um método de geração baseado em ferramentas da geoestatística, que permitem explorar a covariância espacial entre as medidas do ambiente de rádio, para que o REM seja obtido com maior acurácia.

Inicialmente, os conceitos da geoestatística são introduzidos por meio dos processos aleatórios espaciais. Em seguida, os detalhes do modelo de predição espacial para a geração do REM são apresentados, bem como as técnicas de estimação que caracterizam a etapa de treinamento do modelo. O preditor espacial Kriging, conhecido da área geoestatística, é colocado e sua formulação analítica é desenvolvida. Finalmente, simulações computacionais permitem verificar o resultados da geração do REM.

1.2 Processos Aleatórios Espaciais

O estudo sobre os processos aleatórios espaciais é baseado na escolha de modelos que buscam a representação adequada dos fenômenos espaciais em análise. Especificamente, estas representações envolvem descrições matemáticas, estatísticas e visuais dos processos aleatórios espaciais. Com isso, o principal propósito da geoestatística[1] consiste em prover uma descrição estatística sobre a variabilidade espacial dos fenômenos, para que seja possível investigá-los por meio de modelos geoestatísticos.

Um modelo geoestatístico amplamente utilizado para descrever os processos aleatórios espaciais é formado por duas componentes definidas em todas as coordenadas espaciais $\mathbf{s} = (s_x, s_y)$, i.e.,

$$P(\mathbf{s}) = \mu(\mathbf{s}) + \xi(\mathbf{s}), \text{ com } \mathbf{s} \in D, \tag{1.1}$$

em que $\mu(\mathbf{s})$ consiste na média do processo aleatório espacial e $\xi(\mathbf{s})$, que representa as variações aleatórias do processo $P(\mathbf{s})$ sobre a média $\mu(\mathbf{s})$ ao longo do domínio espacial D, definido neste trabalho como um espaço bidimensional para caracterizar o ambiente de rádio das comunicações sem fio (i.e., $D \in \mathbb{R}^2$). Sob o ponto de vista estatístico, o processo $P(\mathbf{s})$ é composto por um conjunto infinito de variáveis aleatórias $P(\mathbf{s}_i)$, cujas realizações são governadas por mecanismos aleatórios, caracterizados de foma probabilística pela distribuição i-dimensional F_P,

$$F_P = \text{Prob}[P(\mathbf{s}_1) < p(\mathbf{s}_1), ..., P(\mathbf{s}_i) < p(\mathbf{s}_i)], \tag{1.2}$$

em que $P(\mathbf{s}_i)$ e $p(\mathbf{s}_i)$ consistem na variável aleatória (VA) espacial e sua realização na coordenada $\mathbf{s}_i$, respectivamente. A Figura 1.1 ilustra a realização de um processo aleatório espacial Gaussiano, ou seja, as variáveis aleatórias espaciais que compõem este processo aleatório seguem a distribuição de probabilidade Gaussiana, resultando nas flutuações aleatórias mostradas ao longo do espaço. Assim, $P(\mathbf{s})$ é dito Gaussiano se a distribuição conjunta de $\{P(\mathbf{s}_1), ..., P(\mathbf{s}_i)\}$ para qualquer conjunto de coordenadas espaciais $\{\mathbf{s}_1, ..., \mathbf{s}_i\}$ é do tipo Gaussiana multivariada [11-16].

Os processos Gaussianos multivariados são especificados com as funções média $\mu(\mathbf{s})$ e covariância $C(\mathbf{s}_i, \mathbf{s}_j)$. A especificação destas funções e a estimação de seus parâmetros fazem parte da modelagem matemática dos fenômenos analisados pela geoestatística. A média de um processo aleatório espacial $\mu(\mathbf{s}) = \mathbb{E}\{P(\mathbf{s})\}$ consiste no valor esperado de $P(\mathbf{s})$ em uma posição espacial $\mathbf{s}_i$, i.e., $\mu(\mathbf{s}_i) = \mathbb{E}\{P(\mathbf{s}_i)\}$. Com isso, dependendo das características de $P(\mathbf{s})$, é possível que $\mu(\mathbf{s})$ assuma diferentes valores ao longo das coordenadas espaciais $\mathbf{s}$ no domínio D. Nesta circunstância, o processo aleatório espacial $P(\mathbf{s})$ exibe uma característica sistemática, denotada como tendência espacial, retratada pela variação do valor esperado do processo aleatório ao longo do espaço [11-16].

[1] A geoestatística teve seu início na década de 1950 com D. Krige, G. Matheron, B. Matérn, A. N. Kolmogorov, além de outros pesquisadores que atuavam em diferentes áreas destacando as ciências da Terra e do clima, além da indústria de mineração, óleo e gás [11],[12],[14-16].

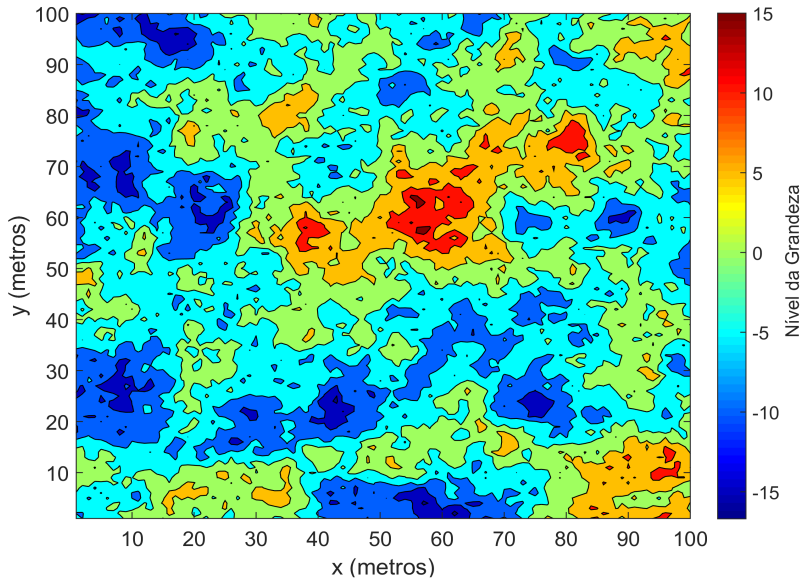


Figura 1.1 – Realização de um processo aleatório espacial Gaussiano.

Na análise dos processos aleatórios espaciais visando a realização de predições, a identificação da tendência $\mu(\mathbf{s})$ é um passo importante e, na maioria dos casos, requer algum tipo de tratamento estatístico. Em geral, este tratamento pode ser feito de duas formas: i) incorporando a modelagem selecionada para a tendência ao funcionamento do preditor espacial e ii) fazendo a estimação seguida da remoção da tendência dos dados para o posterior processamento e análise. No exemplo da Figura 1.1, o processo $P(\mathbf{s})$ não exibe tendência, pois foi gerado com média nula, i.e., $\mu(\mathbf{s}) = \mathbb{E}\{P(\mathbf{s})\} = 0$. Ainda assim, é possível visualizar as variações espaciais ocorrendo na forma de aglomerados[2].

As propriedades estatísticas das VAs espaciais que compõem $P(\mathbf{s})$ determinam a relação de dependência entre tais variáveis aleatórias. Esta relação consiste na caracterização estatística espacial de $P(\mathbf{s})$, que é descrita por sua covariância espacial,

$$C(\mathbf{s}_i, \mathbf{s}_j) = \text{cov}(P(\mathbf{s}_i), P(\mathbf{s}_j)) = \mathbb{E}\{[P(\mathbf{s}_i) - \mu(\mathbf{s}_i)][P(\mathbf{s}_j - \mu(\mathbf{s}_j)]\}. \tag{1.3}$$

O modelo selecionado para (1.3) caracteriza as flutuações espacias aleatórias de $\xi(\mathbf{s})$, expressando a similaridade entre as VAs. Verifica-se que (1.3) requer o conhecimento da média do processo $P(\mathbf{s})$ nas coordenadas espaciais $\mathbf{s}_i$ e $\mathbf{s}_j$. De outro modo, a ideia de dissimilaridade espacial entre as VAs é descrita pela função semivariograma, amplamente difundida na área geoestatística e definida como a metade da variância das diferenças entre as VAs que compõem $P(\mathbf{s})$,

$$\begin{aligned} \gamma(\mathbf{s}_i, \mathbf{s}_j) &= \frac{1}{2}\text{Var}\{P(\mathbf{s}_i) - P(\mathbf{s}_j)\} \\ &= \frac{1}{2}\mathbb{E}\{[P(\mathbf{s}_i) - P(\mathbf{s}_j) - \mathbb{E}\{P(\mathbf{s}_i) - P(\mathbf{s}_j)\}]^2\}. \end{aligned} \tag{1.4}$$

[2]Os aglomerados espaciais consistem em regiões nas quais os valores do processo aleatório são similares e sua identificação sugere (mas não permite inferir) a presença de correlação espacial no processo aleatório. Assim, de acordo os valores de realização assumidos pelas VAs, os aglomerados podem ser de alta intensidade (chamados de *hotspots*) ou de baixa intensidade (conhecidos como *coldspots*).

Na literatura geoestatística, denota-se $2\gamma(\mathbf{s}_i, \mathbf{s}_j)$ como variograma e $\gamma(\mathbf{s}_i, \mathbf{s}_j)$ como semivariograma, sendo que ambas as funções transmitem a ideia de dissimilaridade. A covariância e o semivariograma de um processo aleatório espacial $P(\mathbf{s})$ são ditos estruturados se os valores não nulos de $C(\mathbf{s}_i, \mathbf{s}_j)$ e $\gamma(\mathbf{s}_i, \mathbf{s}_j)$ apresentarem um comportamento interpretável e que possa ser descrito por modelos analíticos de covariância e semivariograma. Assim, a caracterização estatística espacial de $P(\mathbf{s})$ é descrita por modelos de covariância e semivariograma, envolvendo a relação entre $\mu(\mathbf{s})$ e $\xi(\mathbf{s})$, enquanto a utilização destas funções permite que predições espaciais e inferências possam ser realizadas sobre o processo $P(\mathbf{s})$. Deste modo, a seleção de modelos apropriados para (1.3) e (1.4) é um tópico importante, pois permitirá alcançar melhores resultados de predições espaciais não somente no sentido tradicional, i.e., predições enviesadas ou não enviesadas, mas em termos de acurácia, i.e., com erros de predição espacial menores. Isto é possível porque a concepção dos preditores geoestatísticos é baseada em métodos que exploram a covariância e a semivariância espacial do processo aleatório a favor das predições espaciais.

Neste contexto, o formato de $C(\mathbf{s}_i, \mathbf{s}_j)$ e $\gamma(\mathbf{s}_i, \mathbf{s}_j)$ é um aspecto relevante e depende da distância entre as observações de $P(\mathbf{s})$ por meio de $\mathbf{h} = (h_x, h_y)$, definido como um vetor bidimensional de separação espacial entre as coordenadas $\mathbf{s}$, i.e., $\mathbf{h} \equiv \mathbf{s}_i - \mathbf{s}_j$. Em outras palavras, a forma como estas funções se comportam em função de $\mathbf{h}$ indica que observações do processo $P(\mathbf{s})$ que estão relativamente próximas apresentam alta similaridade, enquanto observações distantes são ditas espacialmente descorrelacionadas (apresentam baixa similaridade ou alta dissimilaridade espacial). Outro ponto importante considerado na geoestatística consiste nas pressuposições de estacionariedade sobre o processo aleatório $P(\mathbf{s})$, sobretudo a estacionaridade no sentido amplo [11-16]. Neste caso, as funções (1.3) e (1.4) são invariantes à translação espacial (isotropia), isto é, a covariância e as semivariâncias entre quaisquer coordenadas espaciais de um processo $P(\mathbf{s})$ estacionário (no sentido amplo) não dependerão das posições espaciais específicas, mas do vetor de separação entre as coordenadas $\mathbf{h}$,

$$\begin{aligned} C(\mathbf{s}_i, \mathbf{s}_i + \mathbf{h}) &= \mathbb{E}\{[P(\mathbf{s}_i) - \mu][P(\mathbf{s}_i + \mathbf{h}) - \mu]\} = C(\mathbf{h}) \\ \gamma(\mathbf{s}_i, \mathbf{s}_i + \mathbf{h}) &= \frac{1}{2}\mathbb{E}\{[P(\mathbf{s}_i + \mathbf{h}) - P(\mathbf{s}_i) - \mathbb{E}\{P(\mathbf{s}_i + \mathbf{h}) - P(\mathbf{s}_i)\}]^2\} \\ &= \frac{1}{2}\mathbb{E}\{[P(\mathbf{s}_i + \mathbf{h}) - P(\mathbf{s}_i)]^2\} = \gamma(\mathbf{h}) \end{aligned} \tag{1.5}$$

Este conceito sobre a covariação dos atributos das VAs espaciais na modelagem de $P(\mathbf{s})$ constitui a ideia central da geoestatística para que o uso das funções covariância e semivariograma possibilite a geração de predições espaciais com melhor acurácia. Em sistemas de comunicações sem fio, as funções em (1.5) são estimadas a partir de um conjunto de medidas capturado, para que as estatísticas calculadas possam quantificar a covariância e o semivariograma do ambiente de rádio.

1.3 Modelagem do Ambiente de Rádio

O modelo de sistema utilizado neste trabalho é caracterizado pelo ambiente de rádio de um sistema de comunicação sem fio composto por vários dispositivos e uma estação rádio base (ERB). Algumas pressuposições iniciais são colocadas: i) primeiramente, assume-se que os dispositivos estão distribuídos de acordo com a densidade de probabilidade uniforme na área de cobertura da ERB e que suas posições, descritas pelas coordenadas espaciais $\mathbf{s}$, são conhecidas *a priori*. Além disso, assume-se que tais dispositivos são capazes de realizar as medidas de intensidade do sinal recebido, i.e., potência de recepção, e enviá-las para a ERB, que coordena a operação do sistema de comunicação sem fio. Isso significa que a ERB é responsável pelo processamento das medidas coletadas e pela geração do REM.

É importante mencionar que, na prática, as coordenadas $\mathbf{s}$ são estimadas com métodos de localização utilizados no sistema de comunicação sem fio. Os níveis de potência de recepção nos dispositivos são influenciados diretamente pelos mecanismos de propagação do ambiente de rádio como as difrações, reflexões, refrações, bem como a perda por percurso média e, consequentemente, podem afetar a geração das predições espaciais do REM.

Neste trabalho, o ambiente de rádio consiste em um processo aleatório espacial $P(\mathbf{s})$ definido em todas as coordenadas espaciais $\mathbf{s} = (s_x, s_y)$, formado a partir da tendência $\mu(\mathbf{s})$, representada pela perda por percurso média do ambiente de rádio, e das flutuações aleatórias espaciais $\xi(\mathbf{s})$, representadas pelo o sombreamento do canal sem fio ao longo da região de cobertura D atendida pela ERB. Especificamente, o modelo log-distância [17-18] é utilizado para descrever a perda por percurso média da tendência espacial $\mu(\mathbf{s})$ no espaço bidimensional e pode ser expresso de acordo com,

$$\begin{aligned}\mu(\mathbf{s}) &= P_{\text{tx}} - 10\alpha \log\left(d(\mathbf{s}_{\text{tx}}, \mathbf{s})\right) \\ &= P_{\text{tx}} - 10\alpha \log \sqrt{(s_{x\text{tx}} - s_x)^2 + (s_{y\text{tx}} - s_y)^2},\end{aligned} \tag{1.6}$$

em que P_{tx} é a potência de transmissão utilizada no transmissor da ERB e α consiste no coeficiente de propagação do modelo de perda por percurso. Nota-se que o modelo em (1.6) depende da distância entre a ERB, com localização assumida fixa em $\mathbf{s}_{\text{tx}} = (s_{x\text{tx}}, s_{y\text{tx}})$, e os dispositivos da rede que estão associados às coordenadas espaciais $\mathbf{s} = (s_x, s_y)$, previamente conhecidas pela ERB por meio dos métodos de localização.

A escolha do modelo log-distância é feita em função de duas razões: i) primeiramente devido ao seu uso consolidado e difundido na literatura científica e ii) pela possibilidade de formulação linear do problema de estimação da tendência espacial, discutida adiante. Sobre este último aspecto, é importante observar que o modelo log-distância é linear no parâmetro α e embora linear em α, o modelo não é linear nas coordenadas espaciais $\mathbf{s}$, pois envolve as funções logaritmo e raiz quadrada em razão das distâncias a serem obtidas em duas dimensões. Estas características irão influenciar as etapas de aprendizagem de parâmetros, cujos valores são desconhecidos e constantes, e das predições espaciais, cujos valores são desconhecidos e randômicos em função do processo aleatório $P(\mathbf{s})$.

O modelo (1.1) indica que as flutuações espaciais aleatórias do sombreamento $\xi(\mathbf{s})$ ocorrem em torno da potência média de recepção $\mu(\mathbf{s})$. Em um sistema de comunicação sem fio, isso ocorre devido às características do ambiente de propagação, como construções e obstáculos que estão relativamente próximos aos dispositivos. Especificamente, os sinais transmitidos por um canal sem fio experimentam variações aleatórias em suas intensidades e, uma vez que diversos aspectos dos obstáculos são desconhecidos (e.g., localização, tamanho, propriedades dielétricas), o uso de modelos estatísticos é essencial para descrever as variações aleatórias da potência de recepção nos dispositivos.

Neste contexto, dois fatores de importância são considerados: i) a distribuição de probabilidade das variações aleatórias da potência de recepção provocadas pelos efeitos do sombreamento e ii) a covariância existente entre os valores de potência de recepção ao longo do espaço. Sobre o fator i), a distribuição log-normal é uma das mais utilizadas na literatura para a modelagem dos efeitos do sombreamento do canal sem fio [17-18]. Considerando a escala logarítmica para os níveis de potência de recepção (dBm), este modelo consiste em um processo aleatório Gaussiano, capaz de capturar as variações aleatórias (dB) do sombreamento do canal sem fio de forma satisfatória [17-18]. Com isso, o modelo de sistema em (1.6) relacionado com o ambiente de rádio pode ser expandido de acordo com,

$$\begin{aligned} P(\mathbf{s}) &= \mu(\mathbf{s}) + \xi(\mathbf{s}) \\ &= P_{\text{tx}} - \underbrace{10\alpha \log d(\mathbf{s}_{\text{tx,s}})}_{\text{Perda por Percurso Média}} + \underbrace{\xi(\mathbf{s})}_{\text{Sombreamento}} , \end{aligned} \tag{1.7}$$

em que $\xi(\mathbf{s})$ é um processo aleatório Gaussiano com média nula e covariância espacial $C(\mathbf{s}_i, \mathbf{s}_j)$. Sobre o fator ii) o modelo de Gudmundson[3] é amplamente utilizado na literatura para descrever a correlação espacial do sombreamento, a partir das distâncias de separação $\mathbf{h}$ entre as medidas no ambiente de rádio [19]. Com isso, o decaimento exponencial que caracteriza a correlação do modelo de Gudmunson é adotado para as funções covariância e semivariograma, que podem ser expressas de acordo com

$$C(\mathbf{h}) = m \exp\left(-\frac{\mathbf{h}}{r}\right), \quad \gamma(\mathbf{h}) = m\left\{1 - \exp\left(-\frac{\mathbf{h}}{r}\right)\right\}, \tag{1.8}$$

em que o parâmetro m representa a variância σ^2 do sombreamento (Gaussiano) e quantifica a dispersão das variações aleatórias $\xi(\mathbf{s})$, enquanto o parâmetro r, indica o *range* dos modelos (1.8) e representa a distância de descorrelação espacial d_{corr} do sombreamento log-normal. Logo, verifica-se que o processo $P(\mathbf{s})$ é Gaussiano e espacialmente correlacionado, com média que depende das características da tendência $\mu(\mathbf{s})$ e covariância espacial $C(\mathbf{s}_i, \mathbf{s}_j)$, que depende das características do sombreamento log-normal $\xi(\mathbf{s})$.

[3] O modelo de Gudmundson indica que a correlação espacial do sombreamento decresce exponencialmente em função do aumento da distância $\mathbf{h}$ entre as medidas [17-19]. De fato, vários estudos baseados em campanhas conduzidas em ambientes de propagação constatam a validade do modelo de Gudmundson [17-19], enquanto diversos artigos científicos propõem métodos de geração do sombreamento log-normal, considerando este modelo [20].

1.4 Modelo de Predição Espacial para a Geração do REM

A modelagem do ambiente de rádio permite que o processo aleatório espacial $P(\mathbf{s})$ seja gerado com simulações computacionais por meio da combinação entre a potência de recepção média $\mu(\mathbf{s})$, obtida a partir da perda por percurso média, e o sombreamento log-normal $\xi(\mathbf{s})$ modelado através de processos Gaussianos. Diferentes técnicas podem ser utilizadas para a geração do sombreamento log-normal do canal sem fio: soma de senóides; decomposição de Cholesky, filtragem autoregressiva de primeira ordem e transformações de Fourier dos modelos de correlação do sombreamento [20]. Neste trabalho, o método de Cholesky é escolhido, uma vez que esse é baseado na decomposição da função covariância do processo $P(\mathbf{s})$, que possui relação direta com o semivariograma, principal ferramenta da geoestatística utilizada para a geração das predições espaciais.

A Figura 1.2 mostra a realização de um processo aleatório espacial Gaussiano, gerado com média nula e covariância espacial do tipo exponencial. Um conjunto de coordenadas espaciais $\mathbf{s}$, gerado aleatoriamente a partir da densidade de probabilidade uniforme, é usado para a amostragem espacial do processo $P(\mathbf{s})$. O resultado consiste em um conjunto finito de medidas de potência de recepção $\{P(\mathbf{s}_i),\ i = 1,...N\}$, em que N consiste no número total de medidas coletadas. Na geoestatística, o conjunto dos valores realizados das VAs do processo $P(\mathbf{s})$ é denominado variável regionalizada (VR). Na prática, isso significa que o conjunto de medidas capturado é somente uma parte observável da VR do processo $P(\mathbf{s})$. Por simplicidade, tal conjunto também é denotado como VR neste trabalho, ou seja, os valores regionalizados formam as N medidas que os métodos geoestatísticos possuem para que a estimações das funções covariância e semivariograma possam ser realizadas, visando as posteriores predições espaciais.

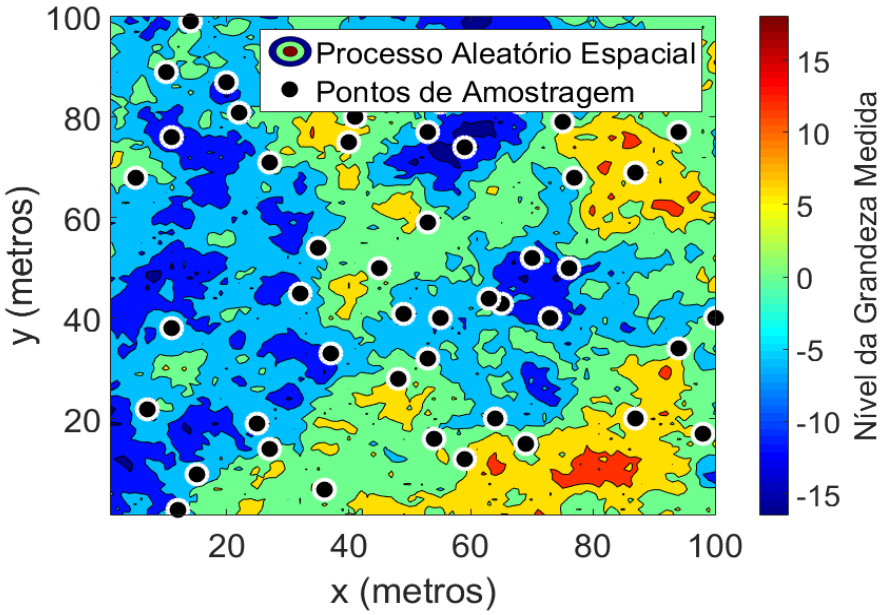


Figura 1.2 – Amostragem espacial sobre a realização de um processo aleatório Gaussiano.

O modelo de predição espacial mostrado na Figura 1.3 é utilizado para a geração do

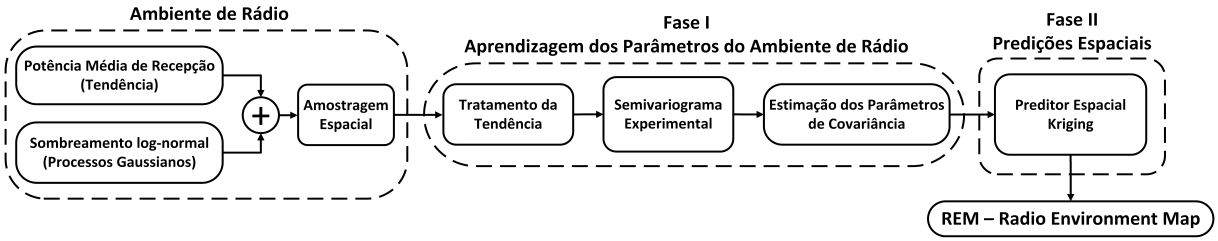


Figura 1.3 – Modelo de predição espacial para a geração do REM.

REM a partir das medidas coletadas com a amostragem espacial. O modelo se fundamenta na abordagem geoestatística e contempla a Fase I, caracterizada pela aprendizagem dos parâmetros do ambiente de rádio com três etapas de estimação: i) o tratamento da tendência; ii) a estimação do semivariograma experimental e iii) a estimação dos parâmetros de covariância do sombreamento log-normal. O êxito na aprendizagem dos parâmetros é essencial para que as predições possam ser realizadas na Fase II com a técnica Kriging e o REM seja gerado. Logo, o objetivo do modelo de predição consiste em usar métodos geoestatísticos que permitem obter o REM com maior acurácia.

Sobre o funcionamento do modelo, simulações computacionais baseadas na modelagem do ambiente de rádio (tendência e processos Gaussianos) são utilizadas para geração o processo aleatório $P(\mathbf{s})$. Em seguida, ambas as fases (I e II) contam com o conjunto finito de N medições realizadas pelos dispositivos da rede com a amostragem espacial, permitindo a formulação matricial do modelo de sistema, de acordo com

$$\mathbf{p} = \mathbf{x}\,\alpha + \xi, \tag{1.9}$$

em que $\mathbf{p} = [P(\mathbf{s}_1), \ldots, P(\mathbf{s}_N)]^{\mathrm{T}}$ é o vetor com as medidas de potência de recepção coletadas, α e $\mathbf{x} = [-10\log d(\mathbf{s}_{\mathrm{tx}}, \mathbf{s}_1), \ldots, -10\log d(\mathbf{s}_{\mathrm{tx}}, \mathbf{s}_N)]^{\mathrm{T}}$ consistem no parâmetro e no vetor de funções do modelo log-distância com $P_{\mathrm{tx}} = 0\,\mathrm{dBm}$, e $\xi = [\xi(\mathbf{s}_1), \ldots, \xi(\mathbf{s}_N)]$ é o vetor aleatório Gaussiano de média nula, desvio padrão σ e matriz de covariância $\mathbf{C}$ (com dimensões $N \times N$) do sombreamento log-normal do canal sem fio.

A primeira etapa da fase de aprendizagem de parâmetros consiste no tratamento da tendência e sua realização é importante por duas razões: i) a presença da tendência implica em flutuações da potência de recepção média, afetando a estacionaridade do processo $P(\mathbf{s})$; ii) a aplicação de técnicas de estimação (e.g., semivariograma experimental), se realizada diretamente sobre as medidas coletadas (com a presença da tendência), pode levar a resultados de predição enviesados [12]. Deste modo, os processos de estimação e remoção da tendência se tornam necessários no caso específico do ambiente de rádio. Na situação em que o modelo da tendência é linear nos parâmetros, como no caso do modelo log-distância, é possível encontrar o estimador não enviesado de mínima variância (MVUE - *Minimum Variance Unbiased Estimator* [21, pp. 85-86]) para o coeficiente α, i.e.,

$$\hat{\alpha} = (\mathbf{x}^{\mathrm{T}}\mathbf{x})^{-1}\mathbf{x}^{\mathrm{T}}\mathbf{p}. \tag{1.10}$$

É importante ressaltar que os valores de potência de recepção do vetor $\mathbf{p}$ são formados por uma parcela média, $\boldsymbol{\mu} = \mathbf{x}\alpha$, e contaminados pelas variações aleatórias ξ e, nesse sentido, o tratamento da tendência é realizado para que os valores regionalizados de interesse (sem a tendência) sejam obtidos, ou seja, a VR desejada representa as variações aleatórias relacionadas com o sombreamento do canal sem fio ξ. Especificamente, a estimativa do coeficiente de propagação $\hat{\alpha}$ com (1.10) é utilizada no modelo (1.6) a fim de obter as estimativas da potência de recepção média em diferentes coordenadas espaciais $\mathbf{s}$, denotadas como $\hat{\boldsymbol{\mu}}$. Com isso, a remoção da tendência do vetor de medidas $\mathbf{p}$ permite que predições das variações do sombreamento ξ possam ser realizadas com métodos geoestatísticos e incorporadas na geração do REM, visando maior acurácia nos resultados de predição espacial. Analiticamente, este processo de remoção da tendência é dado por

$$\mathbf{z} = \mathbf{p} - \hat{\boldsymbol{\mu}}, \tag{1.11}$$

em que $\mathbf{p} = [P(\mathbf{s}_1), \ldots, P(\mathbf{s}_N)]^{\mathrm{T}}$ é o vetor com os valores de potência de recepção coletados do ambiente de rádio, $\hat{\boldsymbol{\mu}} = [\hat{\mu}(\mathbf{s}_1), \ldots, \hat{\mu}(\mathbf{s}_N)]^{\mathrm{T}}$ é o vetor com as estimativas de potência de recepção média obtido a partir de $\hat{\alpha}$ e $\mathbf{z} = [Z(\mathbf{s}_1), \ldots, Z(\mathbf{s}_N)]^{\mathrm{T}}$, que consiste no vetor com os valores regionalizados, após a remoção da tendência. Em síntese, este processo é conhecido como *detrending* e possui dois objetivos: i) remover o viés tendencioso das medições de potência de recepção para que as variações do sombreamento log-normal possam ser estimadas pelas próximas etapas do modelo de predição espacial e ii) permitir que a componente média da potência de recepção possa ser obtida em diversas localidades do domínio espacial D para a geração do REM.

Na sequência do modelo de predição espacial, tem-se as etapas de estimação do semivariograma e covariograma experimentais, além dos parâmetros de covariância, por meio do ajuste de seus respectivos modelos analíticos. O semivariograma e o covariograma experimentais consistem nas estimativas empíricas das funções teóricas semivariograma e covariância do processo aleatório espacial, fundamentais para a geração das predições espaciais. Matematicamente, a estimação do semivariograma experimental $\hat{\gamma}(\mathbf{h})$ é obtida com o valor médio das diferenças quadráticas entre os valores regionalizados $\mathbf{z}$, considerando diferentes separações espaciais $\mathbf{h}$. O método dos momentos de Matheron (MoM) é o principal estimador utilizado para obter o semivariograma experimental, expresso de acordo com [11-16],

$$\hat{\gamma}(\mathbf{h}) = \frac{1}{2N_h} \sum_{\mathbf{s}_i - \mathbf{s}_j = \mathbf{h}} [Z(\mathbf{s}_j) - Z(\mathbf{s}_i)]^2, \ \forall \mathbf{s}_i, \mathbf{s}_j \in D, \ i, j = 1, 2, \ldots N, \tag{1.12}$$

em que N_h representa o número de pares de medidas que se distanciam de $\mathbf{h}$ tomadas no conjunto de valores regionalizados. O número total de medidas N e, consequentemente, o número de pares de medidas N_h são fatores que têm influencia nos resultados do estimador (1.12). De fato, para que as estimativas $\hat{\gamma}(\mathbf{h})$ tenham confiabilidade estatística, é necessário que N_h seja suficientemente grande, exigindo que N também seja elevado.

Sobre esse aspecto de estimação do semivariograma experimental, o fato de o vetor **z** ser composto por um conjunto finito de N medições impõe desafios ao estimador (1.12). Frente à essa situação prática, algum tipo de agrupamento é realizado sobre as distâncias e semivariâncias obtidas, para que os resultados de estimação tenham maior confiança estatística. Ainda assim, variações nos valores das estimativas $\hat{\gamma}(\mathbf{h})$ são comuns e podem ocorrer, uma vez que a média estimada via MoM é baseada na quantidade N_h, que depende da forma como agrupamento é aplicado ao conjunto total de N medidas.

A Figura 1.4 mostra as duas principais formas de estimação do semivariograma experimental: (a) a nuvem do semivariograma, que é constituída diretamente a partir dos valores das diferenças quadráticas entre todas as N medidas do vetor **z** e (b) o semivariograma experimental obtido com o agrupamento das distâncias com o estimador do MoM. É importante mencionar que cada ponto da nuvem de semivariâncias está relacionado com um par de observações espaciais, ou seja, nenhum tipo de agrupamento de distâncias é aplicado. Este formato em (a) é uma ferramenta útil na exploração da variabilidade do processo aleatório, pois permite verificar a presença de *outliers* no sentido de inspecionar semivariâncias muito elevadas em medições que estão espacialmente próximas.

De outro modo, nota-se que o agrupamento das distâncias em (b) suavizou o semivariograma experimental. De forma específica, o agrupamento é obtido com um processo de quantização aplicado à todas as distâncias **h** calculadas entre as medidas do vetor **z**, resultando em um novo conjunto de distâncias denominadas lags [9], [15-16]. Assim, todas as medidas cujas distâncias de separação foram quantizadas para uma determinada distância lag são incluídas no cômputo da respectiva semivariância. É possível verificar a quantidade de medidas usadas em cada agrupamento, mostradas nas caixas numéricas em (b). Nota-se que algumas semivariâncias foram computadas com mais medidas do que outras. Isso significa que cada estimativa $\hat{\gamma}(\mathbf{h})$ é obtida com uma precisão diferente. Portanto, a forma como a quantização é aplicada juntamente com o MoM é um ponto de relevância na implementação prática do estimador (1.12).

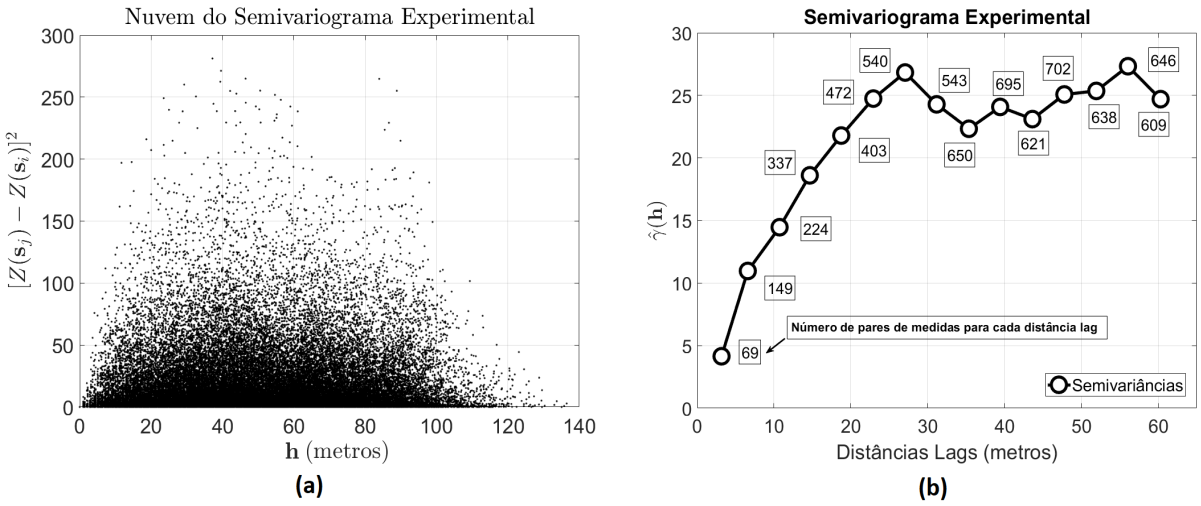


Figura 1.4 – Semivariograma experimental: (a) nuvem de semivariâncias (b) com agrupamento de medidas.

O estimador baseado no MoM também é utilizado na estimação da covariância do processo aleatório espacial, caracterizando o covariograma experimental $\hat{C}(\mathbf{h})$, dado por

$$\hat{C}(\mathbf{h}) = \frac{1}{2N_h} \sum_{\mathbf{s}_i - \mathbf{s}_j = \mathbf{h}} [Z(\mathbf{s}_i) - \hat{\mu}(\mathbf{s}_i)][Z(\mathbf{s}_j) - \hat{\mu}(\mathbf{s}_j)], \ \forall \mathbf{s}_i, \mathbf{s}_j \in D, \ i, j = 1, 2, ...N. \tag{1.13}$$

em que $\hat{\mu}(\mathbf{s}_i)$ e $\hat{\mu}(\mathbf{s}_j)$ consistem na estimativas dos valores esperados nas coordenadas espaciais $\mathbf{s}_i$ e $\mathbf{s}_j$. As covariâncias e semivariâncias experimentais obtidas com os estimadores (1.12) e (1.13) são o ponto de partida para que os modelos de covariância $C(\mathbf{h}, \boldsymbol{\theta})$ e do semivariograma $\gamma(\mathbf{h}, \boldsymbol{\theta})$ possam ser ajustados por meio da estimação de seus parâmetros $\boldsymbol{\theta}$.

A Figura 1.5 mostra as diferenças entre as partes experimentais e seus respectivos modelos para a covariância (a) e para o semivariograma (b). O modelo selecionado deve ser representativo no sentido de capturar o comportamento das estimativas $\hat{\gamma}(\mathbf{h})$ e $\hat{C}(\mathbf{h})$ em função das distâncias lags. A principal característica destes modelos é a presença do vetor de parâmetros $\boldsymbol{\theta} = [m \ r]$ e o processo de aprendizagem ou treinamento do modelo de predição consiste do ajuste dos modelos analíticos em relação às estimativas experimentais, permitindo a obtenção do vetor $\boldsymbol{\theta}$. Algoritmos de otimização são aplicados para que o ajuste dos modelos em relação aos dados experimentais seja alcançado de forma eficiente. Nota-se que, enquanto as estimativas experimentais são baseadas no conjunto finito de distâncias lags, a estimação de $\boldsymbol{\theta}$ permite que os modelos analíticos forneçam valores de semivariâncias e covariâncias para qualquer distância de separação espacial $\mathbf{h}$. Essa característica é fundamental para o funcionamento do preditor Kriging (discutido adiante). Portanto, o ajuste dos modelos exponenciais de covariância e semivariograma em (1.8) às estimativas experimentais em conjunto com as distâncias lags permitirá a estimação do vetor de parâmetros $\boldsymbol{\theta}$ do sombreamento log-normal do canal sem fio. Neste trabalho, esse processo é realizado com a técnica dos mínimos quadrados (LS - *Least Squares*) e caracteriza a finalização da etapa de treinamento do modelo de predição espacial.

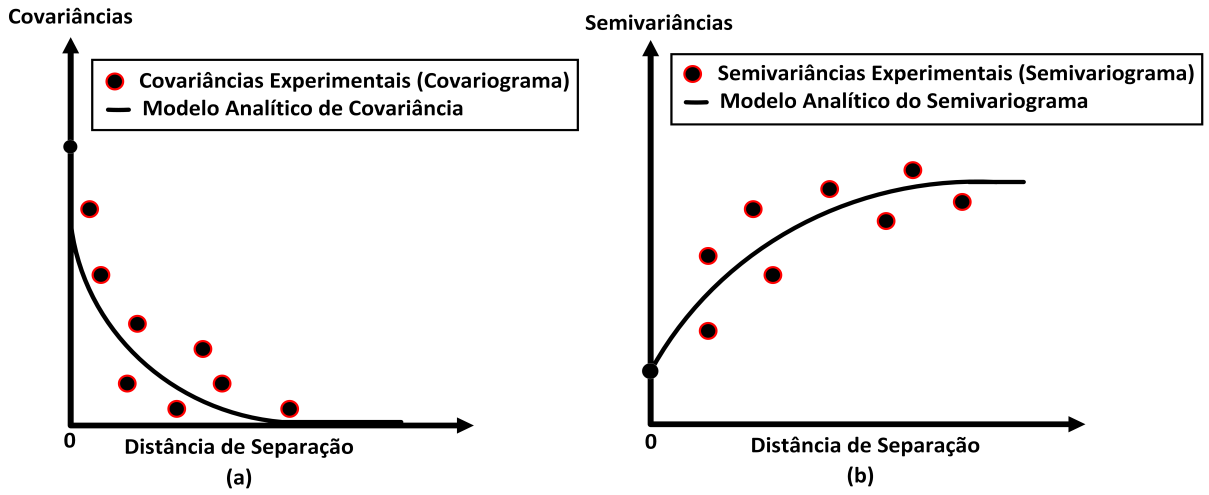


Figura 1.5 – Modelos analíticos e partes experimentais: (a) covariância e (b) semivariograma.

1.4.1 Simulação da Fase I de Aprendizagem do Ambiente de Rádio

Esta seção apresenta os resultados de simulação da Fase I de aprendizagem de parâmetros do ambiente de rádio, destacando a estimação da tendência, do semivariograma e do covariograma experimentais, bem como do vetor de parâmetros $\boldsymbol{\theta}$ por meio da técnica LS. O ambiente de rádio é caracterizado pelas transmissões de uma ERB que possui localização fixa em $\mathbf{s}_{\text{tx}} = (100, 100)$, potência de transmissão igual a 0 dBm e que, com uma antena ominidirecional, atende uma área de cobertura com dimensão espacial de 300 m × 300 m.

A Tabela 1.1 mostra os parâmetros do ambiente de rádio e as configurações utilizadas na simulação. Os valores escolhidos para as configurações buscam representar os cenários de comunicações sem fio, especialmente os ambientes de comunicações sem fio externos (principal alvo deste trabalho). As configurações mostradas retratam um cenário de propagação urbano tipicamente caracterizado pelas faixas $3 \leq \alpha \leq 5$, para o coeficiente de propagação, e $5\,\text{dB} \leq \sigma \leq 13\,\text{dB}$, para o desvio padrão do sombreamento log-normal [18]. Na literatura, vários trabalhos conduzem campanhas de medidas com o objetivo de caracterizar a distância de descorrelação espacial (d_{corr}) do sombreamento, de acordo com o tipo de ambiente de propagação, e.g., 10 m para microcélulas urbanas, 50 m a 120 m para ambientes urbanos e 50 m a 400 m para ambientes suburbanos [22]. Nesta simulação, foram coletadas $N = 200$ medidas do ambiente de rádio, considerando a ausência da incerteza de localização, i.e., as coordenadas espaciais $\mathbf{s}$ são estimadas de forma perfeita pelos métodos de localização. Além disso, é assumido que não ocorrem variações do processo aleatório em pequena escala, uma vez que existe o interesse somente nos efeitos em larga escala do canal de comunicação sem fio. A razão para isto é que os efeitos de pequena escala (e.g., múltiplos percursos) apresentam variações significativas em distâncias muito pequenas (poucos metros ou até centímetros dependendo da frequência de operação), dificultando significativamente a predição espacial baseada nas informações de localização.

Os resultados de simulação podem ser observados na Figura 1.6, onde o ambiente de rádio é composto por (a) mapa da potência de recepção média devido à perda por percurso e por (b) mapa do sombreamento log-normal do canal sem fio com covariância exponencial e isotrópica. A combinação dos mapas (a) e (b) resulta no processo aleatório espacial $P(\mathbf{s})$, que caracteriza o ambiente de rádio, mostrado em (c). Observa-se que o sombreamento log-normal influencia diretamente os valores de potência de recepção.

Tabela 1.1 – Parâmetros e Configurações da Simulação

Parâmetros do Ambiente de Rádio	Configurações
Modelo de Perda por Percurso	Log-Distância ($\alpha = 4$)
Sombreamento	Log-normal (Gaussiano)
Desvio Padrão do Sombreamento	$\sigma = 5\,\text{dB}$
Distância de descorrelação do Sombreamento	$d_{\text{corr}} = 50$ m
Amostragem Espacial	Uniforme
Número de Medidas Coletadas	$N = 200$
Parâmetros da Covariância Espacial	$\boldsymbol{\theta} = [m\ r] = [25\ 50]$

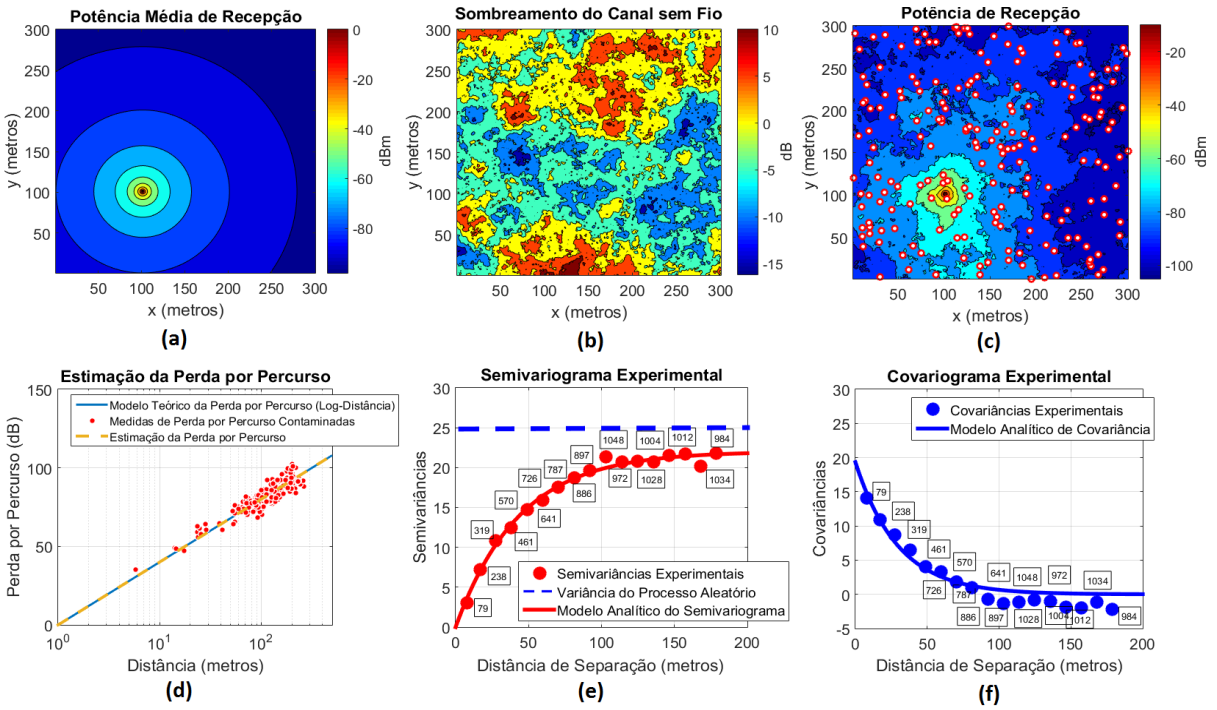


Figura 1.6 – Resultados de simulação: (a) potência média de recepção; (b) sombreamento do canal sem fio; (c) ambiente de rádio com amostragem espacial realizada a partir da densidade de probabilidade uniforme; (d) estimação da perda por percurso média; (e) semivariograma e (f) covariograma.

Os pontos da amostragem espacial são gerados na área de cobertura da ERB, a partir da densidade de probabilidade uniforme, resultando no conjunto de potências de recepção, $\{P(\mathbf{s}_1), \ldots, P(\mathbf{s}_{200})\}$. Dentro deste contexto, é importante mencionar que a densidade de probabilidade utilizada na modelagem da amostragem espacial depende de características específicas das aplicações dos sistemas de comunicações sem fio (e.g., campanhas de *drive-test* com rotas previamente definidas, aglomeração espacial de dispositivos em função das ruas e prédios das cidades). Neste sentido, a realização da amostragem espacial a partir da densidade de probabilidade uniforme é plausível nos casos onde não há um controle *a priori* sobre a localização dos dispositivos ou a falta de conhecimento sobre os padrões de movimentação espacial relacionados com os dispositivos da rede.

O conjunto $\{P(\mathbf{s}_1), \ldots, P(\mathbf{s}_{200})\}$ será utilizado na aprendizagem de parâmetros para que a variabilidade espacial do processo aleatório, provocada pelo sombreamento log-normal, possa ser capturada pelos métodos geoestatísticos, visando as predições do REM. O êxito nesta captura ocorre quando a amostragem espacial consegue acompanhar as variações espaciais do ambiente de rádio. Isto permite que o preditor Kriging explore a covariância estimada para alcançar melhores predições espaciais.

O primeiro resultado de aprendizagem de parâmetros é mostrado em (d) e consiste na regressão do modelo da tendência por meio da aplicação do estimador (1.10) sobre o conjunto de medidas $\{P(\mathbf{s}_1), \ldots, P(\mathbf{s}_{200})\}$, para obtenção de $\hat{\alpha}$. As medidas de perda por percurso afetadas pelo sombreamento log-normal são mostradas a fim de formar um comparativo entre o modelo da tendência utilizado (log-distância) e os resultados da regressão do modelo a partir da estimação de $\hat{\alpha}$.

No caso da simulação, o resultado alcançado $\hat{\alpha} = 4,005$ indica um desempenho relativamente satisfatório, pois $\alpha = 4$. A diferença entre $\hat{\alpha}$ e α ocorre em função da variabilidade das medidas em virtude do sombreamento do canal sem fio e do conjunto finito de medidas coletado. A partir dos resultados de estimação de α, é possível obter $\hat{\boldsymbol{\mu}}$ e remover a tendência das medidas $\{P(\mathbf{s}_1), \ldots, P(\mathbf{s}_{200})\}$ para que o vetor $\mathbf{z}$, que representa as variações espaciais do sombreamento, seja obtido e utilizado nas etapas posteriores da aprendizagem de parâmetros.

Os resultados de estimação do semivariograma e do covariograma experimentais são mostrados na forma de pontos em (e) e (f). A quantidade de pares de observações usados para o cálculo de cada semivariância e covariância também é mostrada nas caixas numéricas. Sobre as estimativas, é possível observar a variação no número de observações agrupadas para a obtenção dos resultados experimentais. Os agrupamentos foram realizados até a distância de 200 m, pois apenas um pequeno número de observações coletadas se distanciava acima de 200 m. Agrupamentos de distâncias lags entre 100 e 150 metros resultaram em mais de 1000 combinações (pares de observações), enquanto os agrupamentos com distâncias lags pequenas não se beneficiaram de muitas combinações.

Os comportamentos crescente para o semivariograma e decrescente para o covariograma refletem o aumento da dissimilaridade e a redução da similaridade à medida em que a distância entre os pares de observações é aumentada. Nota-se a formação de um patamar no semivariograma experimental que se aproxima da variância do sombreamento do canal sem fio ($\sigma^2 = 25$), mostrada com a linha tracejada em (e). Tal patamar é formado a partir de distâncias lags maiores que 50 m. Um resultado similar pode ser visualizado no comportamento do covariograma, onde uma queda mais significativa da covariância ocorre a partir de distâncias acerca de 50 m.

Finalmente, as linhas sólidas em (e) e (f) consistem nos resultados de ajuste dos modelos analíticos exponenciais contemplando a estimação de $\hat{\boldsymbol{\theta}} = [22,02 \;\; 43,46]$ com a técnica LS. As diferenças entre $\hat{\boldsymbol{\theta}} \neq \boldsymbol{\theta}$ e $\hat{\alpha} \neq \alpha$ ocorrem em razão das características do ambiente de rádio, da distribuição espacial e da quantidade de medidas coletadas e, sobretudo do desempenho atingido pelas técnicas de estimação com a abordagem geoestatística. O resultado obtido com o vetor $\hat{\boldsymbol{\theta}}$ finaliza a fase de aprendizagem de parâmetros e se mostra essencial por duas razões: i) permite que informações possam ser extraídas sobre o ambiente de rádio e ii) permite que o preditor Kriging possa explorar a covariância espacial capturada a favor das predições na geração do REM. Na prática, é importante mencionar que não há o conhecimento *a priori* do semivariograma do ambiente de rádio e, portanto, a estimação do covariograma e do semivariograma experimentais se torna necessária. Sobre este aspecto, o uso de um conjunto de medidas para teste do modelo é essencial, pois permite verificar o desempenho das predições obtidas após a fase de treinamento. A próxima seção se dedica a mostrar os detalhes sobre o funcionamento do preditor Kriging, iniciando a fase de predições do REM.

1.4.2 Preditor Kriging - Processos Gaussianos

A aprendizagem de parâmetros possibilita a predição espacial, mas tão importante quanto a realização das predições é a obtenção de um valor com elevada acurácia (confiável) para a potência de recepção em coordenadas espaciais onde não foram realizadas as medidas, especialmente no caso do REM. A teoria de processos Gaussianos fornece o preditor Kriging (da terminologia geoestatística), que consiste no melhor preditor linear para o problema de predição no sentido de minimização do erro quadrático médio de predição espacial [11-16]. Antes de apresentar as particularidades do Kriging, é fundamental analisar o problema da predição de forma geral, envolvendo a minimização do erro quadrático médio de predição (MSPE - *Mean Square Prediction Error*).

Em um primeiro momento, para encontrar o preditor geral que minimiza o MSPE, considera-se o vetor de variáveis aleatórias observadas, que representam a VR, denotado como $\mathbf{z}$, e a variável T, que consiste na VA, cujos valores aleatórios devem ser preditos, a partir do vetor de medidas $\mathbf{z}$. O preditor espacial $\hat{T}$ é uma função das medidas observadas, i.e., $\hat{T} = f(\mathbf{z})$ e o seu MSPE é dado por

$$\text{MSPE}(\hat{T}) = \mathbb{E}\{(T - \hat{T})^2\}, \tag{1.14}$$

em que $\mathbb{E}\{\cdot\}$ é o operador esperança, relacionado com a distribuição conjunta de T e $\mathbf{z}$ por meio de $\hat{T}$. A minimização do MSPE($\hat{T}$) é obtida quando o preditor consiste no valor esperado condicionado às observações, i.e., esperança condicional, $\hat{T} = \mathbb{E}\{T|\mathbf{z}\}$, conforme o desenvolvimento [16, pp.134-135]:

Demonstração.

$$\mathbb{E}\{(T - \hat{T})^2\} = \mathbb{E}_{\mathbf{z}}\{\mathbb{E}_T\{(T - \hat{T})^2|\mathbf{z}\}\}, \tag{1.15}$$

em que $\mathbb{E}_{\mathbf{z}}\{\cdot\}$ e $\mathbb{E}_T\{\cdot\}$ são aplicados em relação à $\mathbf{z}$ e à T, respectivamente. Da definição da variância, o termo $\mathbb{E}_T\{(T - \hat{T})^2|\mathbf{z}\}$ pode ser reescrito de acordo com

$$\mathbb{E}_T\{(T - \hat{T})^2|\mathbf{z}\} = \text{Var}_T\{(T - \hat{T})|\mathbf{z}\} + [\mathbb{E}_T\{(T - \hat{T})|\mathbf{z}\}]^2. \tag{1.16}$$

Condicionada à $\mathbf{z}$, a função do preditor $\hat{T} = f(\mathbf{z})$, resulta em um valor constante. Com isso, tem-se que $\text{Var}_T\{(T - \hat{T})|\mathbf{z}\} = \text{Var}_T\{T|\mathbf{z}\}$ e $\mathbb{E}_T\{(T - \hat{T})|\mathbf{z}\} = \mathbb{E}_T\{T|\mathbf{z}\} - \hat{T}$, resultando em

$$\mathbb{E}_T\{(T - \hat{T})^2|\mathbf{z}\} = \text{Var}_T\{T|\mathbf{z}\} + [\mathbb{E}_T\{T|\mathbf{z}\} - \hat{T}]^2. \tag{1.17}$$

Aplicando o operador $\mathbb{E}_{\mathbf{z}}\{\cdot\}$ sobre a equação (1.17), temos a expressão do MSPE,

$$\mathbb{E}_{\mathbf{z}}\{\mathbb{E}_T\{(T - \hat{T})^2|\mathbf{z}\} = \mathbb{E}\{(T - \hat{T})\}^2 = \mathbb{E}_{\mathbf{z}}\{\text{Var}_T\{T|\mathbf{z}\}\} + \mathbb{E}_{\mathbf{z}}\{[\mathbb{E}_T\{T|\mathbf{z}\} - \hat{T}]^2\}. \tag{1.18}$$

Verifica-se que somente o segundo termo de (1.18) depende da forma do preditor $\hat{T}$ e vai à zero se, e somente se, $\hat{T} = \mathbb{E}\{T|\mathbf{z}\}$ levando à minimização do MSPE. □

É fundamental mencionar que a obtenção do preditor $\hat{T} = \mathbb{E}\{T|\mathbf{z}\}$ segue diretamente da escolha do MSPE como critério a ser otimizado, ou seja, o preditor dado pela esperança condicional é ótimo no sentido de minimização do MSPE, que não necessariamente é a melhor métrica para a avaliação de desempenho em aplicações específicas que se baseiam em predições espaciais. Ainda assim, em função do interesse específico na acurácia e precisão dos resultados, a métrica MSPE e a sua raiz quadrada (RMSPE - *Root Mean Square Prediction Error*) são amplamente utilizadas na análise de desempenho das predições em sistemas de comunicações sem fio [6-9] e, por isso são adotadas neste trabalho. De fato, são os requisitos de desempenho das aplicações de telecomunicações que permitem qualificar os resultados de MSPE alcançados com as predições espaciais. Neste sentido, busca-se por preditores espaciais com elevada acurácia, robustos frente aos efeitos do canal sem fio e de complexidade não proibitiva, além de se adequarem ao funcionamento e a disponibilidade de recursos das aplicações.

O preditor Kriging visa atender as características citadas e, de fato, a linearidade da regressão de distribuições Gaussianas multivariadas permite demonstrar que a técnica Kriging converge para o preditor da esperança condicional quando o processo aleatório espacial é Gaussiano [12, pp. 638]. No ambiente de rádio, a característica Gaussiana deriva diretamente do sombreamento log-normal do canal sem fio. Com isso, diferentes variações do método Kriging podem ser utilizadas em função das características do problema de predição, especialmente da média do processo aleatório $P(\mathbf{s})$.

A Tabela 1.2 mostra as variantes dos métodos Kriging, de acordo com o modelo da tendência. Nas circunstâncias em que o valor da média μ é conhecido *a priori*, o método Kriging simples (KS) é aplicado. Neste caso, não há um modelo de tendência para os dados, uma vez que a média μ é conhecida. Entretanto, não é comum assumir o prévio conhecimento da média μ, sobretudo na prática, pois isto exige o acesso a várias realização do processo $P(\mathbf{s})$. Particularmente no caso dos sistemas de comunicações sem fio, o envio de medidas ao longo do tempo para a ERB (acesso a mais realizações do processo aleatório espacial), depende da capacidade da rede em prover os recursos de banda e energia necessários, além do controle para tais transmissões. Esta situação pode ser tornar crítica em aplicações que possuem restrições de recursos ou exigem consumo mínimo de energia nos dispositivos (e.g., redes de sensores). De outro modo, o método Kriging ordinário (KO) assume a estacionaridade da média do processo $P(\mathbf{s})$, ou seja, μ é constante ao longo do domínio espacial D, mas precisa ser estimada [11-16]. Com isso, o preditor KO se concentra nas variações espaciais em torno da média do processo aleatório $P(\mathbf{s})$.

Tabela 1.2 – Predições Espaciais - Métodos Kriging

Método Kriging	Média	Modelo da Tendência
Kriging Simples (KS)	Conhecida	Nenhum
Kriging Ordinário (KO)	Desconhecida	Constante
Kriging Universal (KU)	Desconhecida	Função das Coordenadas Espaciais

O comprometimento da estacionaridade em função da presença da tendência leva ao uso do preditor Kriging universal (KU), que busca realizar a modelagem não estacionária da tendência, incorporando-a ao funcionamento do preditor espacial. A principal dificuldade no uso do KU está na estimação do semivariograma, que será afetado diretamente pela característica não estacionária da tendência e que possui parâmetros a serem estimados [23]. Além de comprometer o desempenho de predição, isto implica na necessidade do conhecimento *a priori* do semivariograma do processo aleatório, incluindo restrições sobre as funções base da tendência, que devem ser linearmente independentes [11-16]. Geralmente, o KU considera monômios de baixo grau e superfícies suaves (cujas potências não excedam o grau polinomial dois) para a modelagem não estacionária da tendência [11-16]. No caso específico do ambiente de rádio, a perda por percurso é modelada com funções não lineares nas coordenadas espaciais (raiz quadrada e logaritmos), incluindo expoentes que podem ser maiores que dois, o que eleva a complexidade da incorporação da tendência ao preditor KU. A situação se torna crítica nos casos onde mais funções base são utilizadas (e.g., uso de modelos de perda por percurso mais complexos).

Frente a esta situação, este trabalho adota a estratégia *detrending*, que prioriza a estimação seguida da remoção da tendência das medidas capturadas para que os valores regionalizados possam ser obtidos e o preditor KO possa ser aplicado ao problema de predição. De forma analítica, a estimativa produzida pelo preditor KO consiste na combinação linear das medidas $\mathbf{z} = [Z(\mathbf{s}_1) \dots Z(\mathbf{s}_N)]^{\mathrm{T}}$, de acordo com

$$Z_{\mathrm{KO}}(\mathbf{s}_0) = \boldsymbol{\lambda}^{\mathrm{T}} \mathbf{z} = \sum_{i=1}^{N} \lambda_i Z(\mathbf{s}_i), \tag{1.19}$$

em que $Z_{\mathrm{KO}}(\mathbf{s}_0)$ é o resultado escalar da predição em uma coordenada espacial desconhecida $\mathbf{s}_0$, obtido por meio do vetor coluna de pesos Kriging $\boldsymbol{\lambda}$, com dimensões $N \times 1$, e do vetor coluna de medidas $\mathbf{z}$, com dimensões $N \times 1$. Nota-se que a coordenada espacial desconhecida $\mathbf{s}_0$ em (1.19) é levada em conta através dos pesos Kriging λ_i, cujos valores dependem das covariâncias relacionadas com a coordenada espacial $\mathbf{s}_0$, que é alvo da predição. É por esta razão que o modelo analítico do semivariograma e seus parâmetros $\boldsymbol{\theta}$ são essenciais para os preditores Kriging, conforme será mostrado nas explicações adiante.

As características do preditor espacial KO estão associadas a dois fatores importantes: i) a busca pelos pesos ótimos Kriging λ_i que minimizam o MSPE, que consiste na variância do erro de predição espacial, $\mathrm{Var}\{Z_{\mathrm{KO}}(\mathbf{s}_0) - Z(\mathbf{s}_0)\}$ e ii) pela condição imposta aos pesos Kriging, assumindo a estacionariedade, de modo que (1.19) seja não enviesado, i.e.,

$$\begin{aligned} \mathbb{E}\left\{Z_{\mathrm{KO}}(\mathbf{s}_0) - Z(\mathbf{s}_0)\right\} &= \mathbb{E}\left\{\sum_{i=1}^{N} \lambda_i Z(\mathbf{s}_i) - Z(\mathbf{s}_0)\right\} = \sum_{i=1}^{N} \lambda_i \, \mathbb{E}\{Z(\mathbf{s}_i)\} - \mathbb{E}\{Z(\mathbf{s}_0)\} = 0 \\ &= \sum_{i=1}^{N} \lambda_i \mu - \mu = 0 \Longrightarrow \sum_{i=1}^{N} \lambda_i = 1. \end{aligned} \tag{1.20}$$

A variância do erro de predição espacial é expressa por

$$\mathrm{Var}\Big\{Z_{\mathrm{KO}}(\mathbf{s}_0)-Z(\mathbf{s}_0)\Big\}=\mathbb{E}\left\{\Big(Z_{\mathrm{KO}}(\mathbf{s}_0)-Z(\mathbf{s}_0)\Big)^2\right\}-\mathbb{E}\left\{\Big(Z_{\mathrm{KO}}(\mathbf{s}_0)-Z(\mathbf{s}_0)\Big)\right\}^2. \tag{1.21}$$

Considerando a condição de não viés para o segundo termo em (1.21) e usando a definição do preditor KO em (1.19), é possível expandir o primeiro termo de (1.21), resultando na seguinte expressão para a variância do erro de predição espacial,

$$\begin{aligned}\mathrm{Var}\Big\{Z_{\mathrm{KO}}(\mathbf{s}_0)-Z(\mathbf{s}_0)\Big\}&=\mathbb{E}\left\{Z^2_{\mathrm{KO}}(\mathbf{s}_0)-2Z_{\mathrm{KO}}(\mathbf{s}_0)Z(\mathbf{s}_0)+Z^2(\mathbf{s}_0)\right\}\\&=\mathbb{E}\left\{Z^2_{\mathrm{KO}}(\mathbf{s}_0)\right\}+\mathbb{E}\left\{Z^2(\mathbf{s}_0)\right\}-2\,\mathbb{E}\left\{Z_{\mathrm{KO}}(\mathbf{s}_0)Z(\mathbf{s}_0)\right\}\\&=\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_i\lambda_j\,\mathbb{E}\{Z(\mathbf{s}_i)Z(\mathbf{s}_j)\}-2\sum_{i=1}^{N}\lambda_i\,\mathbb{E}\{Z(\mathbf{s}_i)Z(\mathbf{s}_0)\}+\mathbb{E}\{Z^2(\mathbf{s}_0)\}.\end{aligned} \tag{1.22}$$

Ressaltando as seguintes definições da função covariância espacial

$$\begin{aligned}\mathbb{E}\{Z(\mathbf{s}_i)Z(\mathbf{s}_j)\}&=C(\mathbf{s}_i,\mathbf{s}_j)+\mu^2;\\\mathbb{E}\{Z(\mathbf{s}_i)Z(\mathbf{s}_0)\}&=C(\mathbf{s}_i,\mathbf{s}_0)+\mu^2;\\\mathbb{E}\{Z^2(\mathbf{s}_0)\}&=C(0)+\mu^2,\end{aligned} \tag{1.23}$$

e substituindo-as em (1.22), torna-se possível encontrar a relação entre a variância do erro de predição e a função covariância, dada por

$$\mathrm{Var}\{Z_{\mathrm{KO}}(\mathbf{s}_0)-Z(\mathbf{s}_0)\}=\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_i\lambda_jC(\mathbf{s}_i,\mathbf{s}_j)\;-2\sum_{i=1}^{N}\lambda_iC(\mathbf{s}_i,\mathbf{s}_0)+C(0). \tag{1.24}$$

Conforme mencionado, o problema central do preditor KO consiste em encontrar os valores de λ_i que minimizam (1.24) sob a condição de não viés em (1.20). A solução é encontrada com a aplicação do método dos multiplicadores de Lagrange através da seguinte função lagrangiana [12, pp. 163],

$$\phi(\lambda_i,\nu)=\mathrm{Var}\{Z_{\mathrm{KO}}(\mathbf{s}_0)-Z(\mathbf{s}_0)\}+2\nu\Big\{\sum_{i=1}^{N}\lambda_i-1\Big\}, \tag{1.25}$$

em que ν é o multiplicador de Lagrange. Derivando a função (1.25) em relação aos pesos Kriging e ao operador de Lagrange, é possível obter

$$\begin{aligned}\frac{\partial\phi(\lambda_i,\nu)}{\partial\lambda_i}&=+2\sum_{j=1}^{N}\lambda_jC(\mathbf{s}_i,\mathbf{s}_j)-2\sum_{i=1}^{N}C(\mathbf{s}_i,\mathbf{s}_0)+2\nu=0,\\\frac{\partial\phi(\lambda_i,\nu)}{\partial\nu}&=+2\Big\{\sum_{i=1}^{N}\lambda_i-1\Big\}=0,\;\;\mathrm{com}\,i,j=1,...,N.\end{aligned} \tag{1.26}$$

Este resultado permite formular o sistema composto por N+1 equações de covariâncias

do preditor espacial KO, expresso de acordo com

$$\begin{cases} \sum_j \lambda_j C(\mathbf{s}_i, \mathbf{s}_j) + \nu = C(\mathbf{s}_i, \mathbf{s}_0), \;\; i, j = 1, ..., N, \\ \sum_i \lambda_i = 1. \end{cases} \tag{1.27}$$

Reescrevendo o sistema de equações (1.27) de forma matricial, tem-se $\mathbf{C}\boldsymbol{\lambda}_\nu = \mathbf{c}_0$, i.e.,

$$\begin{bmatrix} C(\mathbf{s}_1, \mathbf{s}_1) & \dots & C(\mathbf{s}_1, \mathbf{s}_N) & 1 \\ \vdots & \vdots & \vdots & \vdots \\ C(\mathbf{s}_N, \mathbf{s}_1) & \dots & C(\mathbf{s}_N, \mathbf{s}_N) & 1 \\ 1 & \dots & 1 & 0 \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_N \\ \nu \end{bmatrix} = \begin{bmatrix} C(\mathbf{s}_1, \mathbf{s}_0) \\ \vdots \\ C(\mathbf{s}_N, \mathbf{s}_0) \\ 1 \end{bmatrix},$$

em que $\mathbf{C}$ consiste na matriz de covariâncias obtida entre todas as coordenadas espaciais $\mathbf{s}$ relacionadas com as medidas observadas $\mathbf{z}$, $\boldsymbol{\lambda}_\nu$ é o vetor de pesos Kriging (incluindo o multiplicador de Lagrange ν) e $\mathbf{c}_0$ é o vetor de covariâncias entre as coordenadas espaciais $\mathbf{s}$ e a coordenada espacial alvo da predição $\mathbf{s}_0$. É importante notar que a composição das matrizes $\mathbf{C}$ e $\mathbf{c}_0$ é realizada a partir da aplicação do modelo analítico de covariância e do vetor $\hat{\boldsymbol{\theta}}$ obtido na fase de aprendizagem de parâmetros. Assim, o cálculo dos pesos ótimos Kriging para a solução do sistema de equações envolve a inversão matricial

$$\boldsymbol{\lambda}_\nu = \mathbf{C}^{-1}\mathbf{c}_0. \tag{1.28}$$

A relação entre as funções covariância e semivariograma, $C(\mathbf{s}_i, \mathbf{s}_j) = C(0) - \gamma(\mathbf{s}_i, \mathbf{s}_j)$, pode ser aplicada em (1.27) para reescrever o sistema de equações em função do semivariograma, $\boldsymbol{\Gamma}\boldsymbol{\lambda}_\nu = \boldsymbol{\beta}$,

$$\begin{bmatrix} \gamma(\mathbf{s}_1, \mathbf{s}_1) & \dots & \gamma(\mathbf{s}_1, \mathbf{s}_N) & 1 \\ \vdots & \vdots & \vdots & \vdots \\ \gamma(\mathbf{s}_N, \mathbf{s}_1) & \dots & \gamma(\mathbf{s}_N, \mathbf{s}_N) & 1 \\ 1 & \dots & 1 & 0 \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_N \\ -\nu \end{bmatrix} = \begin{bmatrix} \gamma(\mathbf{s}_1, \mathbf{s}_0) \\ \vdots \\ \gamma(\mathbf{s}_N, \mathbf{s}_0) \\ 1 \end{bmatrix}.$$

em que $\boldsymbol{\Gamma}$ é a matriz de semivariâncias entre as coordenadas espaciais $\mathbf{s}$ e $\boldsymbol{\beta}$ é o vetor de semivariâncias que relaciona as coordenadas $\mathbf{s}$ e a coordenada alvo da predição $\mathbf{s}_0$.

A predição espacial $Z_{\text{KO}}(\mathbf{s}_0)$ é dada pela combinação linear entre as medidas do vetor $\mathbf{z}$ e os pesos Kriging λ_i obtidos em (1.28). Finalmente, a predição da potência de recepção na coordenada alvo $\mathbf{s}_0$ é obtida com a estimativa da potência de recepção média $\hat{\mu}(\mathbf{s}_0)$ e a predição $Z_{\text{KO}}(\mathbf{s}_0)$, de acordo com

$$\hat{P}(\mathbf{s}_0) = \hat{\mu}(\mathbf{s}_0) + Z_{\text{KO}}(\mathbf{s}_0) = \hat{\mu}(\mathbf{s}_0) + \sum_{i=1}^{N} \lambda_i Z(\mathbf{s}_i). \tag{1.29}$$

1.4.3 Simulação da Fase II - Geração do REM

A Figura 1.7 mostra os resultados da Fase II para a geração do REM com a aplicação do preditor Kriging, a partir das medidas coletadas do processo aleatório espacial mostrado anteriormente (Figura 1.6). O mapa verdadeiro da potência de recepção, submetido aos efeitos da perda por percurso média e do sombreamento log-normal, é apresentado em (a), enquanto a geração do REM, obtida através da combinação da estimação da tendência com as predições KO, é mostrada em (b). Por meio do comparativo entre os mapas (d) e (e), é possível observar que o preditor KO foi capaz de capturar a variabilidade espacial desta realização do ambiente de rádio, que é necessária para as predições espaciais. Ainda assim, os mapas em (c) e (f) mostram que as diferenças (magnitude) entre as predições da potência de recepção e do sombreamento comparadas aos valores verdadeiros do ambiente de rádio flutuam espacialmente nos mapas. Embora as diferenças sejam relativamente pequenas em uma parte significativa da área de cobertura da ERB, é possível observar algumas regiões nas quais as predições não tiveram acurácia. Este resultado foi observado para diversas realizações do ambiente de rádio e refletem a necessidade de avaliar o desempenho esperado dos preditores com o propósito de verificar se os métodos geoestatísticos consistem em uma solução efetiva na geração confiável do REM. Sobre este aspecto, é importante ressaltar que o Kriging provê uma estimativa esperada da potência de recepção e que apresenta melhor acurácia, pois é obtida no sentido de minimização do MSPE.

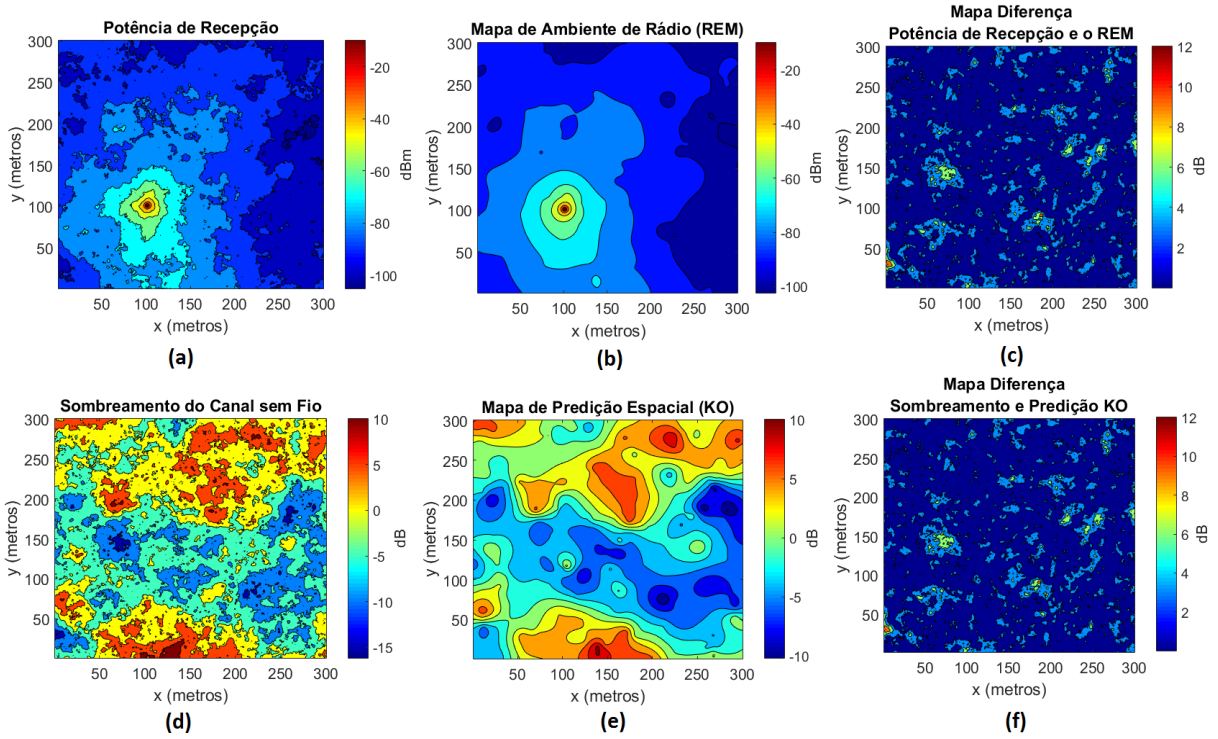


Figura 1.7 – Resultados de simulação: (a) potência de recepção no ambiente de rádio; (b) REM; (c) mapa diferença: potência de recepção e o REM; (d) sombreamento log-normal do canal sem fio; (e) predições espaciais via KO e (f) mapa diferença: sombreamento e predições KO.

1.5 Conclusões e Perspectivas

Este capítulo apresentou um modelo de predição espacial baseado em uma abordagem geoestatística para a geração de mapas de ambiente de rádio em sistemas de comunicações sem fio. Fundamentos sobre os processos aleatórios espaciais foram introduzidos e aplicados na modelagem de um ambiente de rádio em telecomunicações. O modelo de predição espacial formulado é caracterizado por uma fase de treinamento que, a partir de um conjunto limitado de medidas, possibilita a construção do REM por meio da fase de predições espaciais. Discussões sobre as características e limitações do método de geração e as técnicas de estimação foram colocadas. Simulações computacionais permitiram verificar que o preditor espacial Kriging da geoestatística conseguiu capturar a covariância espacial do sombreamento log-normal do ambiente de rádio e, com isso, consiste em uma alternativa ao desafio de geração do REM com maior acurácia.

Sobre a perspectiva de trabalhos futuros, é possível mencionar a investigação dos impactos provocados por alterações na distribuição estatística dos dados em relação à densidade de probabilidade Gaussiana, tanto na fase de aprendizagem de parâmetros como nos preditores, além do estudo de diferentes abordagens para a geração das predições do REM. Em termos de utilização, estudos das diversas aplicações de comunicações sem fio que se baseiam nos resultados de predição obtidos também se apresentam como perspectivas.

Referências Bibliográficas

[1] H. B. Yilmaz, T. Tugcu, F. Alagöz e S. Bayhan, "Radio environment map as enabler for practical cognitive radio networks," *IEEE Communications Magazine*, vol. 51, no. 12, pp. 162-169, Dez. 2013.

[2] A. Kliks, P. Kryszkiewicz, A. Umbert, J. Pérez-Romero, F. Casadevall e L. Kulacz, "Application of radio environment maps for dynamic broadband access in TV bands in urban areas," *IEEE Access*, vol. 5, pp. 19842-19863, Out. 2017.

[3] J. Riihijarvi e P. Mahonen, "Machine learning for performance prediction in mobile cellular networks," *IEEE Computational Intelligence Magazine*, vol. 13, no. 1, pp. 51-60, Fev. 2018.

[4] R. Di Taranto, S. Muppirisetty, R. Raulefs, D. Slock, T. Svensson e H. Wymeersch, "Location-aware communications for 5G networks: How location information can improve scalability, latency, and robustness of 5G," *IEEE Signal Processing Magazine*, vol. 31, no. 6, pp. 102-112, Nov. 2014.

[5] A. H. Sayed, A. Tarighat e N. Khajehnouri, "Network-based wireless location: Challenges faced in developing techniques for accurate wireless location information," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 24-40, Jul. 2005.

[6] J. Li, G. Ding, X. Zhang e Q. Wu, "Recent advances in radio environment map: A survey," in *Proc. Second International Conference, Part II, MLICOM*, Weihai, China, Ago. 2017, pp. 247-257.

[7] H. Braham, S. B. Jemaa, G. Fort, E. Moulines e B. Sayrac, "Spatial prediction under location uncertainty in cellular networks," *IEEE Transactions on Wireless Communications*, vol. 15, no. 11, pp. 7633-7643, Nov. 2016.

[8] L. S. Muppirisetty, T. Svensson e H. Wymeersch, "Spatial wireless channel prediction under location uncertainty," *IEEE Transactions on Wireless Communications*, vol. 15, no. 2, pp. 1031-1044, Fev. 2016.

[9] R. Augusto e C. M. Panazio, "On Geostatistical Methods for Radio Environment Maps Generation under Location Uncertainty," *Journal of Communication and Information Systems*, vol. 33, no. 01, Abr. 2018.

[10] C. T. Phillips, "Geostatistical techinques for practical wireless network coverage papping," Ph.D Dissertation, University of Colorado, Colorado, 2012.

[11] N. Cressie, *Statistics for spatial data*. Hoboken, NJ, USA: John Wiley & Sons, revised edition, 1993.

[12] J. P. Chilès e P. Delfine, *Geostatistics modeling spatial uncertainty*. Hoboken, New Jersey: John Wiley & Sons, Second Edition, 2012.

[13] C. Rasmussen e C. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA, USA: MIT Press, 2006.

[14] M. A. Oliver e R. Webster, *Basic steps in geostatistics: The variogram and Kriging*. Springer Cham Heidelberg New York Dordrecht London: SpringerBriefs in Agriculture, 2015.

[15] J. M. Montero, G. G. Avilés e J. Mateu, *Spatial and spatio-temporal geostatistical modeling and Kriging*. United Kingdom UK: John Wiley & Sons, 2015.

[16] P. J. Diggle e P. J. Ribeiro, *Model-based geostatistics*. New York, USA: Springer Science +Business Media, LLC, 2007.

[17] S. Haykin e M. Moher, *Modern wireless communication*, NJ, USA: Prentice-Hall, 2004.

[18] A. Goldsmith, *Wireless communications*. New York, USA: Cambridge University Press, 2005.

[19] M. Gudmundson, "Correlation model for shadow fading in mobile radio systems," *Electronics Letters*, vol. 27, no. 23, pp. 2145-2146, Nov. 1991.

[20] S. S. Szyszkowicz, H. Yanikomeroglu e J. S. Thompson, "On the feasibility of wireless shadowing correlation models," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 9, pp. 4222-4236, Nov. 2010.

[21] S. M. Kay, *Fundamentals of statistical signal processing: Estimation Theory*. Prentice Hall, 1993.

[22] J. Salo, L. Vuokko, H. M. El-Sallabi e P. Vainikainen, "An additive model as a physical basis for shadow fading," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 1, pp. 13-26, Jan. 2007.

[23] M. Armstrong, "Problems with universal Kriging," *Mathematical Geology*, vol. 16, no. 1, pp. 101-108, Dez. 1984.

CAPÍTULO

2 Antenas Phased Array

Bruno Suarez Pompeo (Centro Tecnológico do Exército), Leandro Guimarães Figueroa Pralon (Centro Tecnológico do Exército)

2.1 Introdução

Atualmente, antenas do tipo *phased arrays* são largamente utilizadas e exploradas, tanto em aplicações civis quanto em aplicações militares, devido às suas particularidades. Dentre suas principais aplicações, pode-se citar o uso em comunicações móveis, satélites, atividades óticas e acústicas, equipamentos médicos e modernos sistemas de radares.

A primeira antena desse tipo, operacional, que se tem registro foi desenvolvida durante a Segunda Guerra Mundial, obtendo baixa precisão. Durante as décadas de 50 e 60, a teoria foi estudada intensivamente tanto nos Estados Unidos, no laboratório Lincoln no *Massachusetts Institute of Technology*, quanto na antiga União Soviética no *Leningrad Electrical Engineering Institute* [1],[2]. Todavia, apesar desse estudo ter iniciado a mais de 50 anos, ainda nos dias de hoje, buscam-se métodos que exploram ao máximo as características de tais sistemas, assim como equipamentos que diminuam o seu custo [3].

A teoria de antenas *phased arrays* é muito ampla e com tópicos tão complexos que por si só são temas de diversos livros e linhas de pesquisa na literatura especializada. Nesse contexto, é importante ressaltar que o objetivo deste capítulo é prover ao leitor, em linhas gerais, os princípios básicos de funcionamento de sistemas que empregam tal tecnologia, abordando os desafios, vantagens e desvantagens existentes, bem como exemplos de aplicações, sem o aprofundamento em determinados temas específicos ou derivações matemáticas complexas, que podem ser encontrados, caso haja o interesse, nas referências indicadas ao longo do texto.

2.2 Conceitos básicos

Antenas do tipo *phased arrays* são definidas como arranjo de antenas onde é possível modificar o diagrama de radiação resultante através da mudança de fases e amplitudes de cada antena do arranjo. O princípio básico de arranjos de antenas baseia-se nas interações construtivas e destrutivas de ondas, que foram demonstradas em 1801 pelo cientista inglês Thomas Young. Em seu experimento confirmou-se que ondas que se combinam em fase em determinado ponto reforçam-se mutuamente, enquanto que ondas que se combinam em fases opostas cancelam-se uma a outra naquele ponto [4]. O mapeamento da energia associada a cada ponto do espaço, devido í s interferências das ondas eletromagnéticas irradiadas por diferentes antenas em um arranjo, é chamado de diagrama resultante do arranjo de antenas.

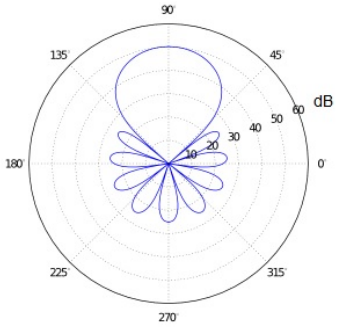


Figura 2.1 – Exemplo de diagrama de antena em coordenadas polar e potência em dB

Antenas *phased arrays* consistem em múltiplos transmissores/receptores (elementos ativos) coerentemente alimentados cada qual com respectiva fase e amplitude. A multiplicidade de elementos permite um controle mais preciso do diagrama da antena, diminuindo lóbulos secundários e modelando o seu padrão de formação [5]. Pode-se dizer que a razão principal de se utilizar esse tipo de antena é a possibilidade de direcionar o feixe principal de forma eletrí´nica, fazendo com que a mudança de direção desse feixe seja realizada quase que instantaneamente (intervalo de tempo dependente da mudança de fases e amplitudes em cada elemento ativo, geralmente na ordem de unidades de microssegundos), diferente de um equipamento de antena fixa e giro mecânico. Dessa forma, não se faz necessário o uso de um motor, componente esse, sendo um dos principais responsáveis pelo tempo médio entre falhas de um equipamento.

O diagrama gerado pelo arranjo dependerá basicamente da geometria imposta entre os elementos ativos, das ponderações (amplitude e fase) fornecidas a cada um deles e obviamente das características de cada elemento ativo. Como cada aplicação exige um tipo de antena adequada, assim como um diagrama resultante, não existe um padrão específico e otimizado que seja aplicável a qualquer projeto. As figuras 2.2a, 2.2b e 2.2c são exemplos de equipamentos que utilizam antenas *phased arrays*. O primeiro é o radar terrestre americano *Pave Paws*, o segundo é um radar aerotransportado no caça russo MiG-29 e o terceiro é o

satélite de comunicação Iridium-Next desenvolvido pelas empresas Motorola e Lockheed Martin. Note que os formatos são distintos.

(a) Radar terrestre Pave-Paws

(b) Radar aerotransportável

(c) Satélite de comunicação

Figura 2.2 – Exemplos de aplicações que utilizam arranjos de antenas.

A geometria do arranjo é compreendida pelo número de elementos ativos que compõem o sistema, pelas distâncias entre eles e pela forma como estão dispostos, podendo essa última ser unidimensional (linear), bidimensional, adotando formato circular, retangular, hexagonal, entre outros ou em casos específicos até tridimensional. Essas características impõem certos limites ao sistema, vantagens e desvantagens. Como dito anteriormente, cada aplicação exige um tipo diferente de antena, levando em conta custo e limitações físicas impostas pelo projeto. Como exemplo, arranjos dispostos em linha só conseguem varrer um único plano, mas são mais simples e menos custosos, enquanto que os planares varrem em três dimensões, mas geralmente necessitam de uma grande quantidade de elementos, e consequentemente, um maior custo e maior gasto de energia. As vantagens e desvantagens de cada formato serão discutidas na próxima seção.

As ponderações atribuídas a cada elemento são compostas por módulo e fase, e por isso descritas como grandezas complexas, possibilitando uma modelagem matemática do sistema. Considere um arranjo planar, bidimensional, contido no plano xz, com N elementos em x e M elementos em z, estando distantes entre si de d_x em uma mesma linha e d_z, na vertical, em uma mesma coluna, inclinada de um ângulo α (Figura 2.3). Aplica-se uma amplitude A_{nm} e uma defasagem ϕ_{nm} em cada elemento nm.

Para um ponto P no espaço, cujas coordenadas esféricas são dadas por (R_P,θ_P,ϕ_P) relativas ao centro do plano de antenas, com R_P muito maior do que $\frac{2L^2}{\lambda}$ (campo distante), sendo L o comprimento da maior dimensão do arranjo e λ o comprimento de onda central do sinal transmitido, pode-se considerar que a onda que chega em P é planar, assim como a onda que chega no arranjo proveniente da reflexão de um objeto contido no ponto P também é planar. A partir dessa suposição, pode-se dizer que a contribuição em fase e amplitude de cada elemento, em função do azimute e da elevação do respectivo ponto, no sinal que chega no arranjo é dada por

$$s_{nm}(\theta_P,\phi_P) = D_{e_{nm}}(\theta_P,\phi_P)A_{nm}e^{j\phi_{nm}}e^{\frac{j2\pi R_{nm}}{\lambda}} \tag{2.1}$$

Onde $[\theta_P,\phi_P]$ são, respectivamente, o azimute e a elevação do ponto P, $D_{e_{nm}}$ é o valor

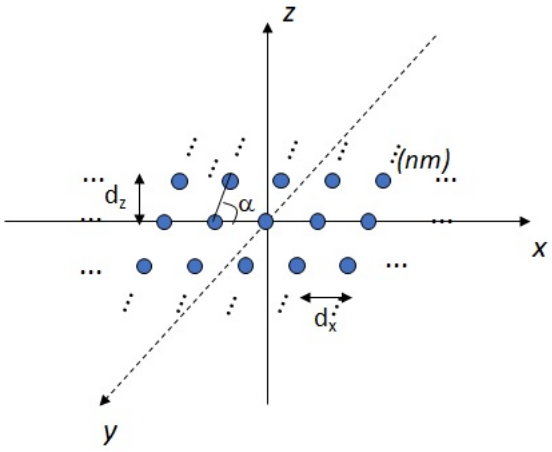


Figura 2.3 – Geometria de um arranjo planar qualquer visto de frente

do diagrama do elemento ativo nm na direção θ_P e ϕ_P, R_{nm} é a distância do ponto P ao elemento nm e λ é o comprimento de onda do sinal eletromagnético radiado.

Como $\overrightarrow{R_P} = R_P \cos\phi_P \sin\theta_P \hat{i} + R_P \cos\phi_P \cos\theta_P \hat{j} + R_P \sin\phi_P \hat{k}$, a Eq.2.1 pode ser escrita como:

$$s_{nm}(\theta_P, \phi_P) = D_{e_{nm}}(\theta_P, \phi_P) e^{\frac{j2\pi R_P}{\lambda}} A_{nm} e^{j\phi_{nm}} e^{\frac{j2\pi(x_{nm}\cos\phi\sin\theta + z_{nm}\sin\phi)}{\lambda}} \tag{2.2}$$

Sendo:

$$\begin{aligned} x_{nm} &= nd_x + \frac{md_z}{\tan\alpha} \\ z_{nm} &= md_z \end{aligned} \tag{2.3}$$

Considerando a contribuição de todo o arranjo e que todos os elementos possuem o mesmo padrão de radiação (aproximação comumente utilizada), o diagrama gerado pela antena é dado por:

$$D(\theta_P, \phi_P) = D_e(\theta_P, \phi_P) e^{\frac{j2\pi R_P}{\lambda}} \sum_{m=1}^{M} \sum_{n=1}^{N} A_{nm} e^{j\phi_{nm}} e^{\frac{j2\pi(x_{nm}\cos\phi\sin\theta + z_{nm}\sin\phi)}{\lambda}} \tag{2.4}$$

onde o primeiro termo é chamado de *Element Factor* pois depende única e exclusivamente do elemento de antena e o segundo termo é chamado de *Array Factor* (AF) pois indica a contribuição de todo arranjo devido í s defasagens e amplitudes impostas em cada elemento. Assim, omitindo a exponencial complexa multiplicativa dependente de R_P (fase constante), pode-se dizer que o diagrama da antena é dado por:

$$D(\theta_P, \phi_P) = D_e(\theta_P, \phi_P) AF(\theta_P, \phi_P) \tag{2.5}$$

Por exemplo, para um arranjo linear, ou seja, M = 1, o módulo do diagrama gerado, em dB, para $\theta = 60^o$ é ilustrado na Figura 2.4, indicando a contribuição do *Array Factor* e a contribuição do *Element Factor*. A pergunta é: como é possível direcionar o feixe para um

ângulo desejado?

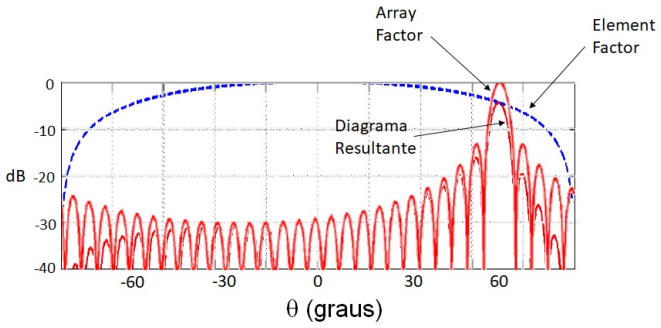


Figura 2.4 – Diagrama de um arranjo de antenas linear com apontamento $\theta = 60^o$

Analisando a Eq.2.4, note que, dada uma direção (θ, ϕ), a função *Array Factor* fornecerá o máximo valor quando $\phi_{nm} = -\frac{2\pi(x_{nm}\cos\phi\sin\theta + z_{nm}\sin\phi)}{\lambda}$, ou seja, quando o produto das duas exponenciais complexas atinge o valor máximo para todos os elementos - o valor unitário. A soma nessa direção será igual a NM (quantidade de elementos no arranjo).

Como dito anteriormente, com base na geometria, nas ponderações utilizadas em cada elemento ativo e nas características eletromagnéticas dos elementos, um diagrama resultante é formado, podendo ser um diagrama de recepção ou de transmissão. Esse diagrama é definido por características importantes, conforme mostrado na Figura 2.5, onde:

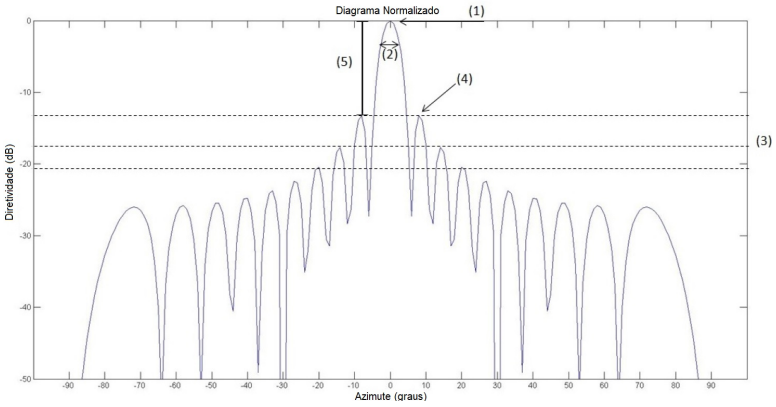


Figura 2.5 – Diagrama gerado por um arranjo de antenas linear

- (1) - Lóbulo principal (*Mainlobe*)
- (2) - Largura de 3dB (*Beamwidth*)
- (3) - Lóbulos secundários (*Sidelobes*)
- (4) - Pico dos lóbulos secundários
- (5) - Relação Primário - Secundário (SLR)

2.2.1 *Grating Lobes*

Seja um arranjo de antenas linear recebendo um sinal, cujo comprimento de onda é λ, oriundo de um ponto P, na direção θ_0 em relação ao centro do arranjo, conforme mostrado na Figura 2.6. Note que a diferença de fase entre elementos adjacentes é dada por:

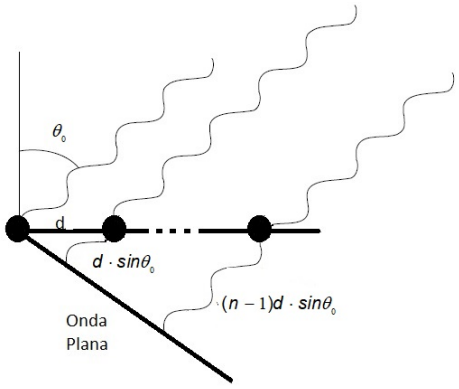


Figura 2.6 – Defasagem entre elementos devido a uma onda planar de direção θ_0

$$\Delta\phi = 2\pi d \sin\theta_0/\lambda \tag{2.6}$$

Dessa forma, a diretividade da antena passa a ser dada por:

$$D(\theta_0) = \sum_{n=0}^{N-1} A_n e^{j\frac{2\pi n d \sin\theta_0}{\lambda}} e^{j\phi_n} \tag{2.7}$$

onde ϕ_n e A_n são a fase e a amplitude adicionadas ao elemento n do arranjo.

Considerando a fase nula no centro do arranjo, para que haja interferências construtivas dos sinais recebidos por cada elemento, deve-se subtrair uma fase igual em cada elemento. Sendo assim, a fase resultante em cada elemento n, para qualquer ângulo θ, pode ser dada por:

$$e^{j2\pi n d \sin\theta_0/\lambda - j2\pi n d \sin\theta/\lambda} \tag{2.8}$$

Como a fase é uma grandeza que se repete em um período de 2π radianos, dependendo das escolhas de d e θ_0, poderá haver pontos de máximo não somente no lóbulo principal escolhido, mas também em outra direção. Esses lóbulos ambíguos são chamados de *grating lobes*. Analisando a Eq.2.8, haverá *grating lobes* quando a fase, em cada elemento, para um ângulo θ_p escolhido de ponderação, for múltiplo de 2π. Assim, a imposição da distância d para que não haja *grating lobes* dado um ângulo θ_0 de apontamento é dado por:

$$\begin{gathered}\frac{2\pi}{\lambda} d(\sin\theta_0 - \sin\theta_p) = 2k\pi \\ d = \frac{k\lambda}{\|\sin\theta_0 - \sin\theta_p\|} \rightarrow d < \frac{\lambda}{\|\sin\theta_0 - 1|}\end{gathered} \tag{2.9}$$

Logo, note que $d = \frac{\lambda}{2}$ garante a não existência de *grating lobes*, independente

do ângulo de apontamento. Para geometrias mais complexas, uma prática comum é o levantamento de áreas permitidas de varredura do feixe no espaço de tal sorte que não haja *grating lobes*. Exemplos desses gráficos serão apresentados na seção 2.3.

2.2.2 Formato do diagrama de antena

Como mencionado anteriormente, o diagrama de antena possui determinadas características que precisam ser especificadas em cada aplicação. Em um arranjo de antenas, as ponderações impostas em cada elemento causam impactos nessas características - largura de 3dB, direção do lóbulo principal, lóbulos secundários e ganho do diagrama - mudando dessa forma o formato completo do diagrama. Os diagramas gerados a partir de um arranjo de antena podem ser divididos em basicamente dois tipos: *Pencil beam* ou *Shaped Beam*. O primeiro é definido como um diagrama de alta diretividade, em elevação e azimute, fino como um lápis ("*pencil*"). Em uma antena qualquer, gerando um diagrama do tipo *Pencil Beam*, a largura de 3dB é dada pela equação 2.10.

$$\theta_{3dB} = \frac{K\lambda}{L} \tag{2.10}$$

Onde, K é uma constante conhecida como fator de largura de feixe dependente do tipo de antena e das amplitudes que a alimentam (quando se trata de um arranjo de antenas), L é o comprimento da antena em determinada dimensão (ou do arranjo de antenas) e λ é o comprimento de onda no espaço livre.

Já o diagrama do tipo *Shaped Beam* pode assumir um formato tão próximo quanto se queira, seguindo determinada função. Um exemplo clássico desse tipo de diagrama é o que segue uma cossecante ao quadrado[2].

Apresentaremos a seguir, de forma sucinta, formas de geração de diagramas a partir das ponderações impostas em cada elemento do arranjo.

Janelamento

Quando somente as amplitudes dos elementos são modificadas, deixando a fase condizente com a direção onde se quer apontar o feixe, chamamos a distribuição de amplitude de função janela ou simplesmente janelamento. Existem diversos janelamentos conhecidos na literatura e citaremos alguns nessa subseção. Essas janelas são utilizadas na análise harmí´nica para reduzir indesejáveis efeitos relacionados ao vazamento espectral, porém cada qual com suas particularidades relacionadas a detecção, resolução, confiança e facilidade de implementação [6]. A tabela 2.1 fornece os valores de K (vide Eq.2.10), da relação Primário-Secundário(SLR) e do fator multiplicativo no ganho do lóbulo principal(G).

[2]Esse tipo de diagrama é bem comum quando se quer garantir a mesma potência, independente da distância ao alvo, dada uma altitude.

Exemplos de funções janelas			
Janela	**K**	**SLR(dB)**	**G**
Uniforme	0.89	-13	1.00
Hamming(0.54)	1.30	-43	0.54
Gaussian(a=3.0)	1.55	-55	0.43
Blackman	1.68	-58	0.42
Dolph-Chebyshev(a=4.0)	1.65	-80	0.42

Tabela 2.1 – Características de Janelamentos

Percebe-se que quanto menor a largura de 3dB (quanto menor, mais estreito é o lóbulo principal), maior é o ganho, porém menor é a relação SLR. Sendo assim, a escolha de janela tem um cunho prático, sendo utilizada a que maior se adequa ao projeto em questão. Como exemplo, a distribuição em amplitude de determinados janelamentos, assim como seus espectros em frequência, são mostrados na Figura 2.7.

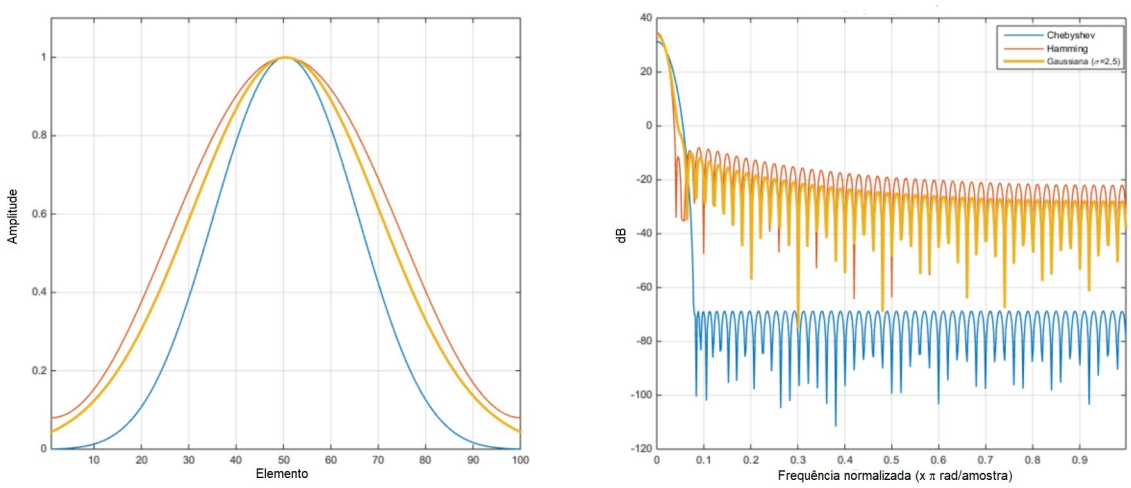


Figura 2.7 – Exemplos de janelas

Para aplicações onde se usam diferenças de fases lineares nos elementos ativos da antena, o uso de janelamentos é bem eficaz, pois o projetista conhecendo as janelas existentes é possível encontrar uma que atenda aos requisitos de projeto. As janelas de Hamming e de Taylor são bastante utilizadas na prática pois alcançam um nível baixo para o maior lóbulo secundário pois usa suas descontinuidades espectrais para cancelar esses lóbulos próximos do lóbulo principal. Essa última é interessante pois o projetista limitando o valor da SLR, existirá um janelamento de Taylor próprio para alcançar esse valor.

É importante citar que independente da janela utilizada, com o deslocamento do feixe principal de um ângulo θ há uma perda de ganho e um alargamento da largura de 3dB. Isso se deve ao fato do comprimento da antena ser diminuído virtualmente, como se ela tivesse inclinada em relação ao seu eixo principal e seu comprimento passasse a ser igual a $L' = Lcos\theta$, e consequentemente:

$$G' = G_0 cos\theta \qquad \theta'_{BW} = \frac{K\lambda}{L'} = \frac{K\lambda}{Lcos\theta} \tag{2.11}$$

Existem casos onde janelamentos não podem ser usados, como por exemplo quando se quer um diagrama com descontinuidades, ou quando se quer um diagrama com determinada forma diferente de uma função sinc(·). Para isso existem outras maneiras de se encontrar as fases e amplitudes que precisam ser utilizadas em cada um dos elementos ativos. Dentre essas maneiras existentes, serão citadas algumas.

Método da Transformada de Fourier

Esse método é usado quando se deseja um determinado formato de diagrama, indicando a função que o define. Como o *AF* é dado pela soma de produtos entre ponderações e defasagens temporais (exponenciais complexas), essas ponderações podem ser encontradas através da transformada inversa de Fourier, sendo a função o próprio *AF*, conforme mostrado na Eq. 2.12.

$$w_n = \frac{d}{\lambda} \int_{-\lambda/2d}^{\lambda/2d} AF(u) e^{-j2\pi und/\lambda} du \tag{2.12}$$

Onde $u = \sin\theta$. Esse método fornece o menor erro quadrático médio em relação ao diagrama para $d \geq \frac{\lambda}{2}$. Para distâncias menores, o domínio da integração excede a região visível e a definição do diagrama não será única [5]. Vale citar que quanto maior for o número de elementos no arranjo, menor será o erro entre o diagrama desejado e o obtido.

Método Woodward-Lawson

Esse método, assim como o da Transformada de Fourier, é usado quando se quer atingir um determinado padrão de diagrama de antena. Baseia-se na soma de diagramas do tipo *Pencil Beam* deslocados, consistindo em um algoritmo de superposição. As ponderações impostas em cada elemento do arranjo são dadas pela soma das ponderações utilizadas em cada diagrama gerado, ou seja, se o elemento i necessita de uma ponderação $w_{ik} = a_{ik}e^{j\phi_{ik}}$ para gerar a k-ésima sinc(·), para gerar o diagrama final sua ponderação será dada por:

$$w_i = \Sigma_{k=1}^{N} w_{ik} = a_i e^{j\phi_i} \tag{2.13}$$

Método polinomial de Schelkunoff

Esse método é usado quando se deseja criar nulos em determinadas direções. Baseia-se em escrever o *AF* como um poliní´mio de grau N, com N sendo o número de elementos usado no arranjo. Dessa forma, fazendo $z = e^{j\phi} = e^{j2\pi d cos\theta/\lambda + \beta}$ pode-se escrevê-lo como:

$$AF = \Sigma_{n=1}^{N} a_n z^{n-1} = a_1 + a_2 z + a_3 z^2 + ... + a_N z^{N-1}$$
$$\|AF\| = \|a_n\|\|z - z_1\|\|z - z_2\|...\|z - z_n\| \quad (2.14)$$

onde z_i, com $1 \leq i \leq N$ são as raízes do poliní´mio.

Assim, dada a estrutura da antena e o comprimento de onda do sinal transmitido/recebido, uma região de possíveis nulos é criada. A partir dessa região, define-se os nulos desejados, ou seja, os valores de z_i. Finalmente, tendo o AF criado, encontra-se os valores das ponderações necessárias.

2.2.3 Formação de feixes Adaptativa (Adaptive Beamforming)

Algoritmos adaptativos permitem que o sistema, de forma automática, manipule o diagrama gerado com base nos sinais recebidos, ou seja, com base no cenário em que se encontra. As ponderações impostas em cada elemento do arranjo são selecionadas a fim de otimizar algum critério de performance (Relação Sinal-Interferência, Erro médio quadrático, Ganho...). Serão citados em seguida alguns critérios de otimização utilizados com essa finalidade.

Mínimo Erro Médio Quadrático (MMSE)

O Mínimo Erro Médio Quadrático é um critério de otimização que visa minimizar o erro entre uma função desejada e a função gerada. O diagrama em blocos básico deste método é dado abaixo.

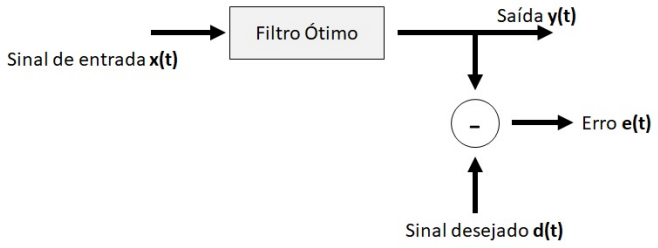


Figura 2.8 – Diagrama em bloco de filtro ótimo

O vetor **x(t)** pode ser dado pela soma de sinal **s(t)**, interferência (*jammer*) **a(t)** e ruído **n(t)**. Assim, aplicando as ponderações **w(t)** nos elementos do arranjo, a saída **y(t)** do formador de feixe é dada por:

$$\boldsymbol{y(t) = w^H x(t)} \quad (2.15)$$

Assim, o erro pode ser dado por:

$$\boldsymbol{e(t) = d(t) - w^H x(t)} \quad (2.16)$$

E assim, o erro médio quadrático será:

$$MSE = E[e^2(t)] = d^2(t) - w^H d^x(t)x(t) - [d^x(t)x(t)]^H w + w^H x(t)x^H(t)w \quad (2.17)$$

Minimizando o fator MSE, encontra-se o vetor ponderação w_{opt} dado pela Eq.2.18

$$w_{opt} = R_x^{-1} r_{dx} \quad (2.18)$$

Sendo $R_x = E[x(t)x^H(t)]$ e $r_{xd} = E[d^*(t)x(t)]$. Essa solução é chamada de solução de Wiener-Hopf.

Linearly Constrained Minimum Variance (LCMV)

O LCMV é um algoritmo que minimiza a potência total de saída do arranjo sujeita a restrições impostas em uma matriz de restrições (C^H) dada uma direção pré-determinada de apontamento. As restrições em C^H são dadas em termos de nível máximo de potência que o diagrama deve ter em determinadas direções. O número de restrições necessariamente deverá ser menor do que o número de elementos no arranjo. As ponderações impostas em cada elemento são dadas pelo vetor w_{opt}, onde:

$$w_{opt} = R_x^{-1} C (C^H R_x^{-1} C)^{-1} C^H w \quad (2.19)$$

Um exemplo de um diagrama gerado com restrições de nulos em -30^o e 10^o, sendo a direção de apontamento em 0^o é mostrado na Figura 2.9.

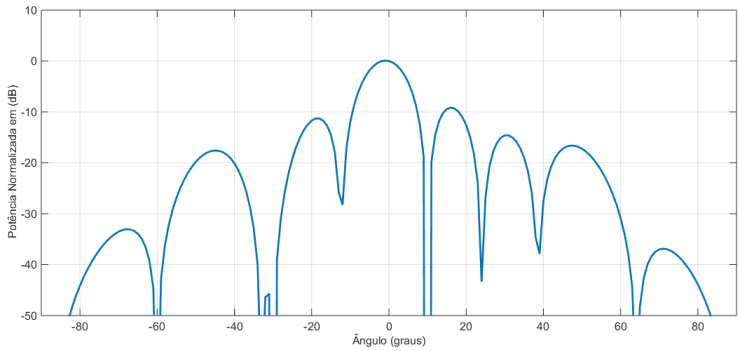


Figura 2.9 – Exemplo de diagrama gerado através do método LCMV - nulos em 10^o e -30^o

Minimum Variance Distortionless Response (MVDR)

No caso específico onde $C^H w = 1$, no algoritmo LCMV, o algoritmo se reduz í resposta MVDR. Isso quer dizer que o algoritmo preserva a potência em determinada direção de apontamento, enquanto suprime interferências e ruídos em outras direções. Assim, as

ponderações impostas em cada elemento do arranjo são dadas por:

$$\boldsymbol{w}_{opt} = \frac{\boldsymbol{R}_x^{-1}\boldsymbol{C}}{\boldsymbol{C}^H\boldsymbol{R}_x^{-1}\boldsymbol{C}} \tag{2.20}$$

Esse algoritmo equivale ao MMSE e fornece a solução de máxima relação Sinal-Ruído dentro de um fator de escala.

Existem outros algoritmos de criação de diagrama que não foram mencionados. Atualmente, pode-se citar avanços em técnicas que utilizam Algoritmo Genético, Inteligência Artificial e métodos de otimização utilizando custos e penalidades, por exemplo.

2.3 Arquiteturas e componentes de arranjos de antenas

Nesta seção serão abordadas diferentes arquiteturas e componentes de arranjos de antenas, com suas respectivas vantagens e desvantagens. Dentre os principais fatores a serem considerados durante a especificação de um arranjo de antenas, destacam-se a confiabilidade/falha dos módulos utilizados, limitações relativas ao controle de suas amplitudes e fases, tamanho, peso, fabricação, polarização, requisitos de ganhos e, principalmente, custos associados [5, 7].

Os principais componentes de um arranjo de antenas são:

- Elemento de antena

  Os elementos de antena são os responsáveis pela transição de energia entre o sistema de alimentação e o espaço livre. Em arranjos de antenas, devido í s restrições espaciais, é mais comum o uso de microstrip e stripline, substituindo os tradicionais dipolos e guias de onda.

- Transmissor

  Os transmissores desempenham um papel de muita importância em sistemas que empregam arranjos de antenas, sendo diretamente responsáveis pela eficiência do sistema, potência, aquecimento, tamanho e peso. Os primeiros sistemas a empregar arranjos de antenas utilizavam a tecnologia de tubos (*Traveling Wave Tubes*), por serem capaz de prover elevada potência (aproximadamente 40dB) a um baixo custo, apesar das elevadas perdas inseridas. Atualmente, transmissores de estado sólido são largamente empregados [2]. Apesar de apresentarem falhas, com o amadurecimento da tecnologia, o tempo médio entre falhas atual de dispositivos desta natureza passa de 500.000 horas de uso. Transmissores de estado sólido, geralmente, são dispositivos de baixa potência (até 100W). Dentre os principais materiais disponíveis no mercado, destacam-se os a base de Silício-Germânio (SiGe), Nitreto de Gálio (GaN) e Arseneto de Gálio (GaAs). A potência provida por cada dispositivo é uma função da frequência de operação. Por exemplo, componentes a base de SiGe apresentam potência acima de

100W quando operando em banda L, porém esta decai rapidamente se a frequência de operação aumenta. Os demais, apesar de apresentarem potências menores, na faixa de 20W, são mais estáveis com o aumento da frequência de operação.

- Receptor

  O principal componente de um receptor de um arranjo de antena, comumente responsável por ditar seu desempenho, é o amplificador de baixo nível de ruído (*Low Noise Amplifier* - LNA). Ademais, como arranjos de antenas usualmente são sistemas monostáticos, ou seja, o mesmo conjunto de antenas é utilizado para transmissão e recepção de sinais, circuladores que provejam eficientes isolamentos entre os referidos circuitos também se tornam essenciais.

- Defasador

  Defasadores são os responsáveis por controlar a diferença de fase entre elementos do arranjo, de forma a se conformar o feixe da forma desejada. Estes podem ser implementados através do controle via fase ou por atraso no tempo. A quantização de defasagens possíveis que estes podem prover tem papel fundamental na análise de desempenho do sistema. Dispositivos com 4 a 6 bits vêm sendo popularmentee empregados, apesar de não ser incomum encontrar dispositivos com menos (3) ou mais (até 8) bits de quantização. A maioria dos defasadores de fase são dispositivos analógicos, controlados, por uma tensão de entrada. Existem diversas tecnologias para tais componentes: ferrite, diodo, circuitos de transistor, sistemas microeletromecânico (MEMS), cada qual com vantagens e desvantagens no tocante a peso, consumo de potência, tempo de troca entre estados e perda [2]. Defasadores de ferrite, por exemplo apresentam tempo de troca de fase da ordem de dezenas de microssegundos e baixo consumo de potência, da ordem de dezenas de watts.

- Sistema de Alimentação

  O sistema de alimentação é o mecanismo responsável por distribuir coerentemente a potência entre os transmissores e receptores e os elementos do arranjo, que pode ser confinado ou espacial. Em sistemas confinados a energia é aprisionada dentro de guias de onda, cabos ou mesmo placas de circuito impresso, criando uma rede para levar o sinal a cada elemento, podendo ser em série ou em paralelo. Sistemas em série são mais simples e baratos, porém não permitem um bom controle do diagrama resultante. Ademais, cada divisor de potência insere perdas ao sistema que devem ser consideradas. Sistemas em paralelo, em contrapartida, por proporcionarem linhas de tamanho igual, são mais adequados para aplicações de elevada largura de banda, como será visto mais para frente, apesar de serem mais caros e pesados. Sistemas espaciais também são mais economicamente viáveis, apesar de requererem irradiadores em ambos os lados do arranjo (lentes) e apresentarem perdas referentes í parcela de energia que não chega ao arranjo. Todavia, a ausência de um sistema

de alimentação cabeado reduz custos e peso, tornando-se boas alternativas para comunicações móveis.

2.3.1 Radares ativos e passivos

A maneira como se alimenta em fase e amplitude os elementos ativos depende da arquitetura do sistema. Existem dois tipos principais de arquiteturas para arranjo de antenas: arranjos passivos e arranjos ativos. O primeiro tem como gerador de potência um único transmissor que divide a energia gerada por uma rede de elementos. Tem como vantagens ser uma solução simples e mais barata e como desvantagem um controle menor em amplificação e atenuação, impedindo o uso efetivo de janelamentos [8]. Adicionalmente, é conveniente destacar que tanto os defasadores, quanto o sistema de alimentação, têm de ser capazes de suportar uma elevada potência de transmissão.

Já o segundo possui um módulo Transmissor/Receptor em cada elemento de radiação (em adição aos defasadores), garantindo assim uma menor perda de potência í´hmica e capacidade de criar ponderações em amplitude, dando uma maior flexibilidade na conformação de feixes. Como desvantagens, citam-se uma maior geração de calor, ocasionando um maior aquecimento da antena e, consequentemente, a necessidade de uma quantidade maior de dissipadores e métodos de troca de calor, além de ser um sistema de custo elevado [8]. A Figura 2.10 apresenta exemplos simples dessas arquiteturas em que são utilizados circuladores, divisores de potência, amplificadores de alta potência (HPA), amplificadores de baixo nível de ruído (LNA), conversores analógico/digital e misturadores.

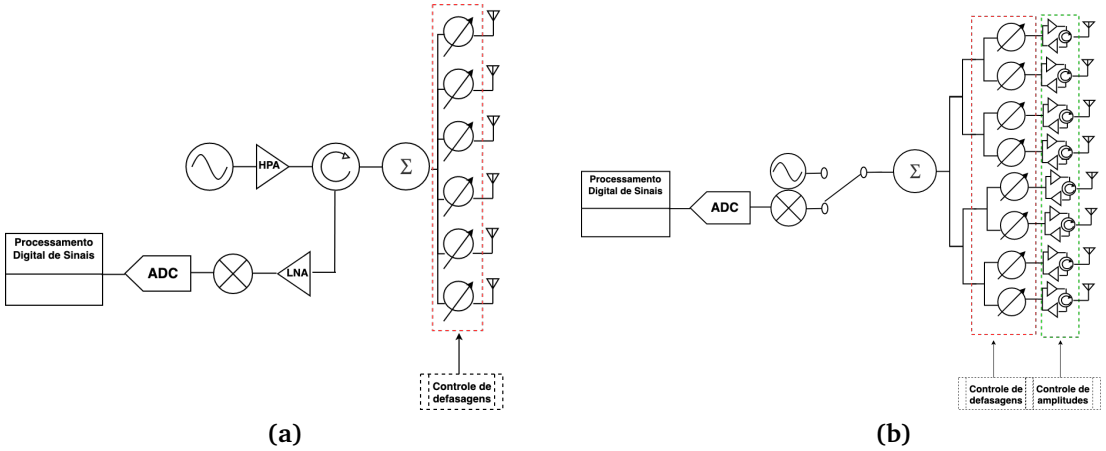


Figura 2.10 – Arquiteturas de arranjos: exemplo de (a) arranjo passivo com sistema de alimentação em séria e (b) arranjo ativo com sistema de alimentação em paralelo.

Em geral, a figura de ruído de arranjos passivos é mais elevada do que em arranjos ativos, isto porque nesses últimos os LNAs estão localizados próximos da extremidade do arranjo, minimizando as perdas que o precedem. Com relação ao custo e confiabilidade, apesar de arranjos ativos serem mais custosos e mais suscetíveis a falhas, eles não somente apresentam um custo menor por watt radiado, como também suas falhas têm impacto menor

no desempenho do sistema (por serem mais distribuídas), induzindo uma depreciação gradativa no sistema, vital para aplicações de arranjos de antenas.

2.3.2 *Digital Beamforming*

A aplicação das ponderações (amplitude e fase) para conformação do feixe de recepção pode também ser realizada de forma digital no caso de arranjos ativos. Nesse cenário, após cada LNA estaria um conversor analógico digital, para que a soma dos sinais ponderados seja feita nesse domínio. Tal arquitetura possui um elevado custo, devido í adição dos conversores, porém introduz uma flexibilidade muito grande para o sistema, permitindo a compensação de elementos com falhas, conformação de diagramas com complexos requisitos de nulos e lóbulos secundários e aplicações que exijam múltiplos feixes simultâneos, sem a necessidade de *hardwares* específicos para tal.

Existem dois grandes desafios de tal tipo de implementação. O primeiro está no consumo de potência e encapsulamento mecânico do sistema, que deve comportar os circuitos, incluindo refrigeração, em um espaçamento de aproximadamente metade do comprimeto de onda entre elementos (de modo a se evitar *grating lobes*). O segundo é no poder computacional exigido para processamento dos sinais recebidos de cada elemento de antena ou *subarray* (técnica abordada na próxima seção). A Figura 2.11a apresenta um exemplo de configuração de um arranjo ativo que emprega *digital beamforming*.

2.3.3 Subarranjos (*Subarrays*)

Configurações ativas e que empregam *digital beamforming*, pelo número de componentes necessários, se tornam custosas, e, consequentemente, muitas vezes inviáveis. Existem diversas aplicações que não necessitam de amplo grau de liberdade e por tanto, um compromisso na escolha da arquitetura pode ser estabelecido, reduzindo o número de componentes, em troca de desempenho do sistema. Nesse contexto, encontram-se as arquiteturas baseadas em subarranjos, em que elementos são agrupados entre si, reduzindo o grau de liberdade do sistema, porém, reduzindo também os custos associados [5]. A Figura 2.11b apresenta um exemplo de configuração de um arranjo que emprega 3 subarranjos e *digital beamforming*.

O *Array Factor* de um arranjo de antenas que utiliza subarranjos passa a ser escrito como

$$AF(\theta,\lambda) = \sum_{n=1}^{N} e^{\frac{-j2\pi nd}{\lambda}(\sin(\theta)-\sin(\theta_0))} \sum_{m=1}^{M} e^{\frac{-j2\pi m\Delta s}{\lambda}(\sin(\theta)-\sin(\theta_0))} \tag{2.21}$$

onde N é o número de elementos, d o espaçamento entre eles, M o número de subarranjos e Δs o espaçamento entre os centros de fase destes.

Note que o projeto de arquiteturas baseadas em subarranjos não é trivial, e dentre os compromissos inerentes a tal tecnologia, resta o fato de que a distância entre os centros

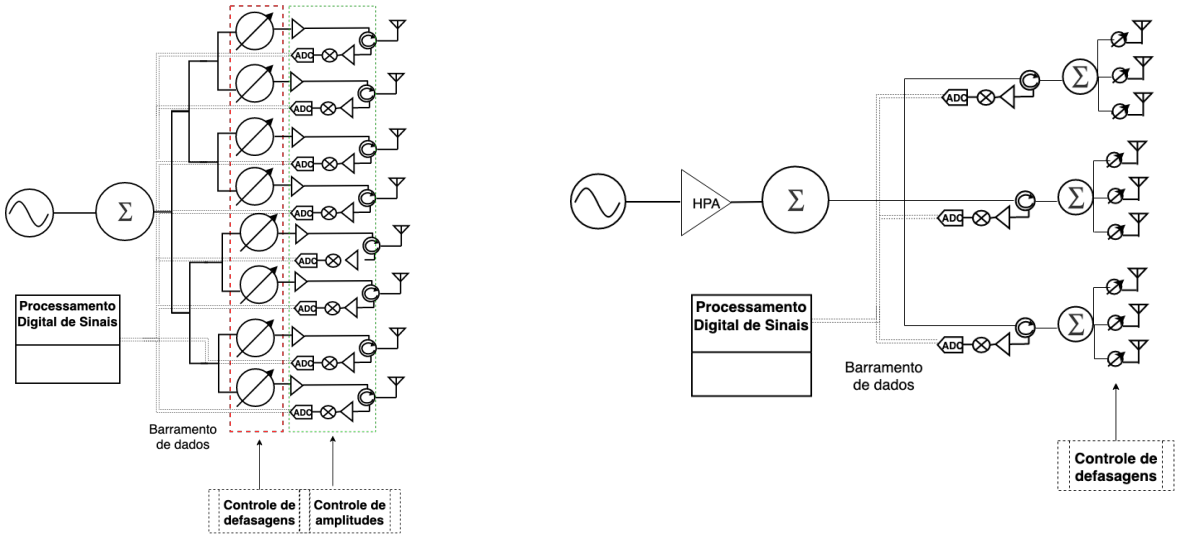


Figura 2.11 – Arquitetura de arranjos: exemplo arranjo com (a) *beamforming* digital e (b) subarranjo.

de fase de cada subarranjo normalmente é maior do que metade do comprimento de onda, indicando a necessidade de uma análise minuciosa dos *grating lobes* do sistema, visto que os subarranjos podem inserir os comumente chamados *quantization lobes* ao diagrama resultante. Nesse contexto, é importante mencionar que os elementos de cada subarranjo podem ser escolhidos de forma a otimizar o diagrama resutlante.

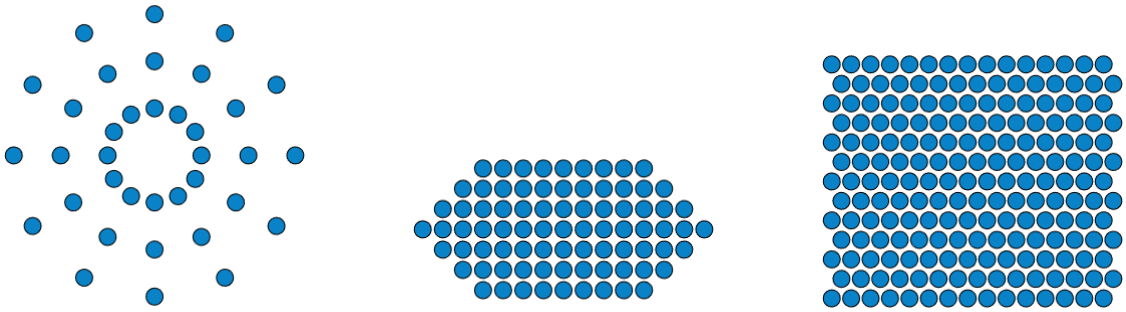
Ademais, uma desvantagem de sistemas que empregam tal arquitetura é a maior depreciação de desempenho devido í falhas de componentes. Isso porque, dependendo do elemento que apresentar mau funcionamento, todo o subarranjo pode ficar comprometido.

2.3.4 Geometrias do arranjo

A arquitetura mais simples de um arranjo de antena é a uniforme linear (ULA). Nessa configuração os elementos são espaçados de forma equidistante ao longo de uma linha. Conforme apresentado na seção anterior, seu modelamento é relativamente simples, sendo uma simplificação dos arranjos planares. Arranjos tridimensionais também podem ser utilizados, mas estão fora do escopo do presente capítulo.

Em arranjos planares, a configuração retangular pode ser considerada a mais tradicional. Seu modelamento é dado conforme a Eq.2.4, considerando $\alpha = 90°$, e portanto os diagramas resultantes em azimute e elevação são derivados diretamente de equações fechadas e conhecidas. Todavia, existem outras configurações que podem ser empregadas, quer seja para redução do número de elementos, diminuindo o custo, quer seja para adequação a uma limitação física imposta pela aplicação. A Figura 2.12 apresenta algumas geometrias comumente empregadas em arranjos de antenas planares.

Arranjos concêntricos são formados por múltiplos arranjos circulares. Considerando um único círculo, com alimentação uniforme, o diagrama resultante apresenta um formato



(a) (b) (c)

Figura 2.12 – Geometrias típicas de arranjos de antenas: (a) arranjo concêntrico, (b) arranjo hexagonal e (c) arranjo triangular.

similar ao obtido com arranjos lineares (ULAs), com diretividade dada por [5]

$$E(\theta) = \pi\sigma^2 \frac{J_1(\theta)}{\theta} \tag{2.22}$$

onde $J_1(\theta)$ é a função de Bessel de primeira espécie. Note que a complexidade no modelamento acompanha a mudança de geometria.

Arranjos hexagonais, por sua vez, apresentam um modelamento relativamente simples. Estes proporcionam um janelamento natural ao arranjo, uma vez que a diminuição de elementos nas bordas naturalmente reduz os lóbulos secundários, ao custo do alargamento do feixe principal. Por essa razão, normalmente são empregados quando há limitação no custo, inviabilizando um sistema de controle de feixes mais robusto (defasadores e amplificadores/atenuadores).

Arranjos triangulares, como o da Figura 2.12c, também são bastante comuns na literatura, provendo um espaçamento entre elementos bastante eficiente ($\alpha \neq 90°$ na Eq. 2.4). Note que com o mesmo número de elementos é possível diminuir a distância entre eles em uma das dimensões, caracterizando o diagrama resultante de outra forma.

No projeto de arranjos com geometrias específicas para determinada aplicação, é importante considerar o posicionamento dos *grating lobes* resultantes. A Figura 2.13a apresenta o exemplo de um diagrama de *grating lobes* de um arranjo de antenas retangular, enquanto a Figura 2.13b apresenta o mesmo diagrama, ambos em *sine-space*[1], considerando um arranjo triangular.

Ao realizar a varredura, altera-se as coordenadas (u,v) relativas ao lóbulo principal do arranjo (localizado dentro do círculo unitário verde) e, consequentemente, altera as demais coordenadas relativas a cada *grating lobes* (migração de *grating lobes*), que podem entrar na região de varredura do arranjo (círculo unitário verde) ou não. Os *grating lobes* representarão um problema se estiverem localizados dentro dos limites de varredura específicos da aplicação para a qual se destina o arranjo.

Note que a posição e quantidade de *grating lobes* é função da geometria do arranjo.

[1] *Sine-space* é uma representação comumente usada em sistemas que utilizam arranjo de antenas para indicar direção no espaço. Essa notação é derivada a partir de uma transformação matemática dos ângulos de azimute e elevação($[\theta, \phi] \rightarrow [u, v]$).

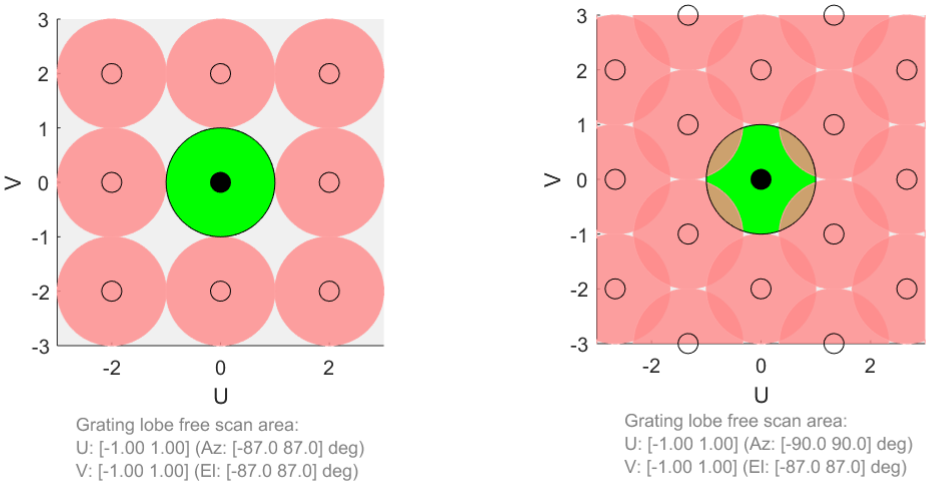


Figura 2.13 – Diagrama de *grating lobes* em arranjos planares no plano U-V onde a área verde mostra o espaço permitido sem *Grating Lobes* e as linhas vermelhas indicam os limites a partir dos quais haverá *Grating Lobes*.: (a) arranjo retangular com 225 elementos e (b) arranjo triangular com 195 elementos.

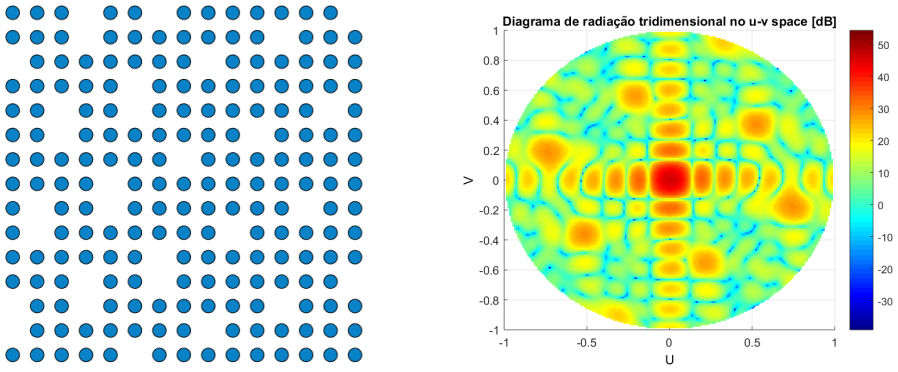
De forma geral, se os elementos estiverem espaçados a uma distância menor que metade do comprimento de onda (como já mencionado na primeira seção), os *grating lobes* estarão a distâncias angulares acima de 90°, não gerando ambiguidade em todo o espaço. Arranjos mais compactos necessitam de mais elementos, para uma mesma área efetiva, e apresentam desafios mais complexos de alimentação, refrigeração e agrupamento de componentes. Arranjos mais espaçados, em contrapartida, necessitam de menos elementos, ou seja, custos menores e não requerem atenção específica devido a proximidade dos elementos. Dessa forma, fica evidente a necessidade de se otimizar a geometria do arranjo, de forma a se minimizar o número de elementos sem a existência de *grating lobes* na região de varredura. É importante destacar que a inclinação mecânica do arranjo como um todo também influencia na posição dos *grating lobes* e deve ser considerada na análise.

A geometria de arranjos de antenas não precisa ser sempre uniforme. Para diversas aplicações é necessário que o feixe seja estreito, mas não necessariamente que o ganho seja elevado. Dessa forma, é preciso que a antena seja larga sem que o número de elementos seja alto, reduzindo consideravelmente o custo do sistema. Nesse contexto, encontram-se os arranjos não-uniformes (Figura 2.14), muito utilizados em cenários com elevada interferência eletromagnética, comunicações via satélite e interferí´metros para astronomia.

Note que o diagrama resultante de um arranjo não-uniforme apresenta distorções, principalmente no tocante a lóbulos secundários. Todavia, estes podem ser controlados e otimizados em função dos elementos removidos do arranjo.

2.3.5 Arranjos Banda larga

Até o momento foi feita análise considerando que o sinal transmitido é banda estreita, ou seja, o comprimento de onda em todo espectro pode ser considerado único. Quando a aplicação exige que o sinal transmitido/recebido pelo arranjo de antenas seja banda larga



(a) (b)

Figura 2.14 – Arranjo planar não-uniforme de dimensões 15 x 15: (a) geometria e (b) diagrama tridimensional resultante no *sine-space*.

um novo desafio é introduzido ao projetista [9]. Isso porque o *Array Factor* passa a ser dado por

$$AF(\theta,\lambda) = \sum_{n=1}^{N} e^{-j2\pi n\Delta x\left(\frac{\sin(\theta)}{\lambda}-\frac{\sin(\theta_0)}{\lambda_0}\right)} \tag{2.23}$$

Dessa forma, o máximo da função (lóbulo principal do diagrama resultante) não ocorrerá quando $\sin(\theta) = \sin(\theta_0)$, mas sim, quando $\frac{\sin(\theta)}{\lambda} = \frac{\sin(\theta_0)}{\lambda_0}$, gerando um desvio de apontamento (referido na literatura como *squint*) máximo dado por

$$\Delta\theta = -\frac{B}{f_0}\tan(\theta_s) \tag{2.24}$$

onde B é a largura de banda do sinal

Nesse contexto, é conveniente definir dois conceitos: largura de banda instantânea e largura de banda operacional. A primeira advém da definição de largura de banda como os limites de frequência em que o desvio de apontamento é metade da largura de 3dB do lóbulo principal. Dessa forma, e lembrando ainda que este alarga conforme o ângulo de apontamento, pode-se dizer que

$$B\,[\%] \approx 2\Delta\theta_{3\text{dB}}\,[^\circ] \tag{2.25}$$

onde $\Delta\theta_{3\text{dB}}$ é a largura de 3dB no apontamento 0°. Note que a largura de banda instantânea é limitada pela arquitetura do arranjo. A Eq. 2.25 pode limitar a gama de aplicações em que arranjos de antenas podem ser utilizados. Todavia, a largura de banda operacional pode ser muito maior que a descrita pela referida equação, se ao invés de defasadores de fase, forem utilizados atrasos no tempo na arquitetura, que provém a mesma variação de fase, independente da frequência do sinal. Dessa forma, pode-se afirmar que a largura de banda operacional é uma função dos componentes, considerando o aumento nos custos associados para troca.

2.4 Erros e Tolerâncias em Antenas Phased Array

Apesar da Eq. 2.4 apresentar o diagrama em campo distante de um arranjo de antenas como o produto entre o diagrama de um elemento e um *Array Factor*, esta não leva em consideração nenhum aspecto prático referente í tecnologia. Arranjos de antena são formados por componentes de *hardware* e *software* que, ao serem integrados, operam de forma conjunta para conformação de diagramas de radiação que atendam requisitos específicos de uma determinada aplicação. Cada componente do sistema apresenta limitações e erros que impactam diretamente o comportamento e o controle dos demais [10]. O correto gerenciamento dessa dinâmica indica o desempenho do arranjo. Por exemplo, o sistema pode utilizar o melhor e mais custoso método para cálculo das ponderações de amplitude e fase a serem empregadas nos transmissores/receptores, objetivando que o diagrama resultante atenda a complexos requisitos de ganho, lóbulos secundários e nulos em posições específicas, porém, se os defasadores e amplificadores não conseguirem prover tais ponderações (limitações de quantização ou erros), o diagrama resultante não terá o formato desejado [11, 12].

Antes de continuar a análise de erros em arranjos de antenas, primeiramente, é conveniente ressaltar que todo sistema apresenta uma quantidade máxima aceitável de erro, caracterizada por sua tolerância. Idealmente, este valor seria nulo, porém existem custos associados que devem ser levados em consideração [13]. Normalmente, este cresce de forma exponencial com o inverso da tolerância. Ademais, cada componente eletrí´nico ou mecânico do sistema apresenta uma determinada tolerância, que influi de maneira distinta no comportamento do arranjo. Dessa forma, torna-se uma tarefa difícil mensurar pontualmente quais requisitos cada componente deve satisfazer. Como alternativa, em caráter prático, é comum empregar como figura de mérito a acurácia de 0.2dB em amplitude e $\pm 3°$ em ângulo de apontamento do diagrama de radiação resultante.

Na teoria de arranjos de antenas, erros são normalmente caracterizados por seus valores RMS (*Root Mean Square*), que para distribuições de média nula são equivalentes ao desvio padrão. O erro de fase devido í quantização, inserido por um defasador de K bits, por exemplo, pode ser modelado por uma distribuição triangular (de parâmetro meio passo de quantização) e seu desvio padrão escrito como

$$\sigma_\phi = \frac{1}{\sqrt{3}} \frac{\pi}{2^K} \tag{2.26}$$

Os erros em arranjos de antenas podem ser constantes ou variáveis e geralmente são classificados como periódicos, aleatórios ou sistêmicos. Erros sistêmicos são aqueles previsíveis, sendo função do apontamento do feixe, de parâmetros da forma de onda empregada ou das condições de operação. Erros mecânicos devido í montagem do arranjo, que influenciem no distanciamento entre elementos podem ser classificados como erros sistêmicos, por exemplo. O impacto dos mesmos será direto nos diagramas resultantes, independente das ponderações em fase e amplitude utilizadas. Erros sistêmicos podem

ser medidos em fábrica, porém dificilmente serão totalmente compensados, devido í s incertezas introduzidas na medição e efeitos de quantização. Em geral, causam desvios de apontamentos ou picos de lóbulos secundários. A melhor forma de mitigá-los é reduzindo os intervalos de calibração e trabalhando em técnicas para compensação.

Erros aleatórios, como o próprio nome sugere, são imprevisíveis. Falhas em componentes, por exemplo, normalmente são de origem aleatória. Para o correto modelamento dos mesmos e avaliação de seus impactos no desempenho do arranjo é conveniente o uso de ferramentas estatísticas, relacionadas aos primeiros e segundos momentos. Todavia, tais ferramentas são mais descritivas em arranjos com elevado número de elementos (da ordem de centenas). Em arranjos pequenos, essas são mais complexas e o correto modelamento de seu comportamento torna-se um desafio para o projetista. Imperfeições de componentes, tais como os defasadores, atenuadores e elementos de antena, entre outros, também se enquadram nessa categoria. Seus efeitos podem ser tanto na fase quanto na amplitude do arranjo, alterando o diagrama de radiação do sistema. É importante destacar que os efeitos causados por erros aleatórios se espalham ao longo de todo arranjo, levando a uma degradação gradual, e por isso não são tão críticos.

Ao contrário dos erros aleatórios, erros periódicos tendem a se concentrar em partes específicas do diagrama. Erros oriundos da quantização dos defasadores e atenuadores podem ser enquadrados como periódicos [5], causando desvios de fase e amplitude, respectivamente.

Erros ocasionados por variação de temperatura, por outro lado podem ser periódicos ou aleatórios. Variações de temperatura podem deteriorar o comportamento de componentes eletrí´nicos, como defasadores e atenuadores, mas também podem alterar o tamanho e formato das antenas, que podem levar a erros de fase causando aumento dos lóbulos secundários ou erros de apontamento. Em sistemas que empregam arranjos de antenas, o projeto do sistema de refrigeração é vital para o seu correto funcionamento. Deseja-se, idealmente, que o sistema não apresente gradientes de temperatura, de modo que a modificação de comportamento de seus componentes seja uniforme por todo arranjo. Tradicionalmente, sistemas de refrigeração líquida têm apresentado melhor desempenho em arranjos de antenas, e por isso são os mais empregados. Sistemas de refrigeração í ar, normalmente, podem ser utilizados em conjunto com a líquida, porém dificilmente são encontrados sistemas que dependam única e exclusivamente da refrigeração í ar.

A perda de diretividade de um arranjo de antenas, que impacta diretamente na redução do seu ganho e consequente aumento de lóbulos secundários pode ser escrita como

$$\frac{D_{\text{erro}}}{D} \approx \frac{1-P}{1+\sigma_\phi^2+\sigma_A^2} \tag{2.27}$$

onde D_{erro} é a diretividade do arranjo com erros, D é a diretividade sem erros, P é a probabilidade de falha de um elemento, σ_ϕ é o desvião padrão do erro de fase e σ_A o desvio padrão do erro de amplitude. O nível médio de lóbulo secundário, por sua vez, pode ser escrito como

$$SL = \frac{\sigma_\phi^2 + \sigma_A^2}{N\epsilon_A} \tag{2.28}$$

onde ϵ_A é a eficiência da abertura.

Erros em amplitude e fase no nível elemento causam aumento de lóbulos secundários, efeito similar ao apresentado por arranjos não-uniformes (Figura 2.14), porém sem o controle proporcionado no projeto destes últimos. A Figura 2.15 ilustra o diagrama resultante de um arranjo linear uniforme (ULA) com 24 elementos e erros aleatórios em amplitude e fase, conforme distribuições apresentadas na referida figura. Note o aumento dos lóbulos secundários e alargamento do lóbulo principal. Por estar normalizado, a perda de diretividade não pode ser observada, todavia, torna-se intuitiva, considerando que o nível dos lóbulos secundários aumentam e a energia total deve ser conservada.

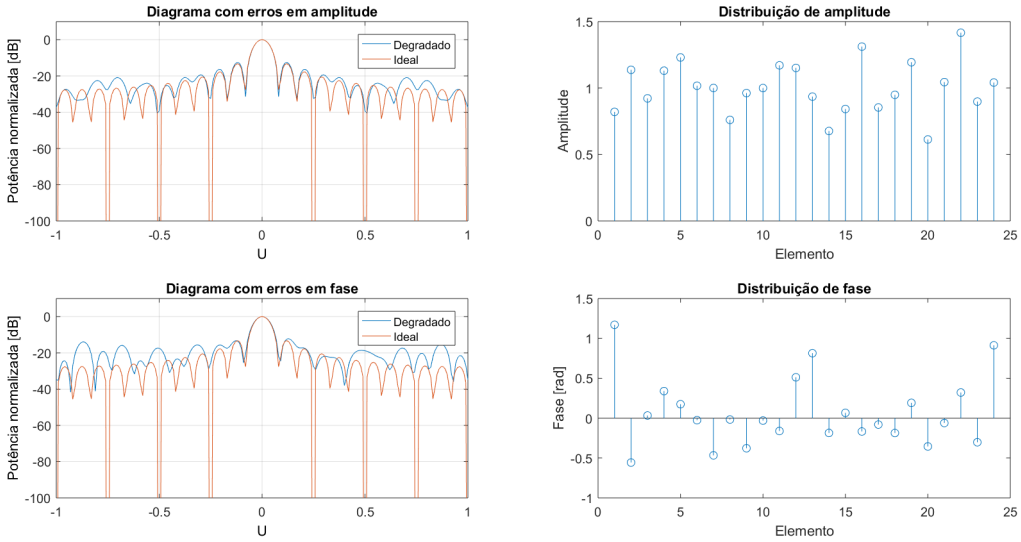


Figura 2.15 – Diagramas resultante de um arranjo de antenas com erros de amplitude e fase no *sine-space*.

Erros em subarranjos ou em um número elevado de elementos de forma correlacionada provocam erros de apontamento. O erro de apontamento pode ser escrito como

$$\sigma_\theta = \frac{\Delta_{3dB}}{1,5\sqrt{N_c}}\sigma_\phi \tag{2.29}$$

onde σ_ϕ já foi previamente definido, Δ_{3dB} é a largura de 3dB do arranjo sem erros e N_c é o número de células de correlação de erros. Note que, quanto mais aleatórios forem os erros, menor serão as N_c, fazendo com que N_c se aproxime do número de elementos do arranjo, reduzindo consideravelmente o erro de apontamento.

Erros causados pelo acoplamento mútuo entre elementos também são intrínsecos a arranjos de antenas, e devem ser corretamente mapeados. Sempre que um elemento irradia, uma parte da energia pode ser recebida por seus vizinhos, podendo modificar o diagrama de radiação do mesmo e causar variações de impedância (afetando o casamento)

em função da região iluminada. Isso significa que, na prática, devido í s interações entre elementos, principalmente na região da borda do arranjo, os diagramas de radiação de cada elemento são diferentes entre si, mesmo em arranjos que utilizem o mesmo tipo de elemento, evidenciando mais uma simplificação da Eq. 2.4.

A correta densidade de potência em campo distante, em uma determinada superfície esférica (R, θ, ϕ), de um arranjo de antenas pode ser então escrita como [5]

$$S(\theta, \phi) = \frac{1}{4\pi} \frac{P_{in}}{R^2} \epsilon_L (1 - |\Gamma|^2) D(\theta, \phi) \tag{2.30}$$

onde P_{in} é a potência de entrada na antena, ϵ_L é um fator de eficiência que leva em consideração as perdas de circuitos, Γ é o coeficiente de reflexão da antena e $D(\theta, \phi)$ é a diretividade da antena.

Note que, ao contrário de antenas unitárias, que apresentam coeficiente de reflexão bem definido e cujo ganho é dado para a situação em que a antena está casada ($\Gamma = 0$), em arranjo de antenas a impedância de entrada varia conforme o ângulo de apontamento, em função do acoplamento mútuo entre elementos. Dessa forma, deve-se levar em consideração para o cálculo do ganho tanto as perdas causadas por dissipação quanto por reflexão. Considerando um arranjo planar de área A e comprimento de onda λ, o ganho do mesmo pode ser escrito como

$$G_0 = \epsilon_L (1 - |\Gamma|^2) 4\pi \frac{A}{\lambda^2} \tag{2.31}$$

Para a recepção o ganho é o mesmo, considerando que a polarização da antena esteja casada com a polarização da onda incidente. Caso a polarização da onda incidente seja desconhecida, é comum o emprego de antenas polarizadas circularmente. Neste caso, deve-se considerar um fator de perda de polarização no cálculo da abertura efetiva da antena [5].

Outra fonte de erro, geralmente negligenciada, mas que pode impactar significativamente a análise e projeto de arranjos de antenas destinados a aplicações com severa restrição de nulos e lóbulos secundários é o erro inserido na medição. Isso porque, os diagramas de radiação devem ser medidos em campo distante, cuja distância mínima aproximada é dada por

$$R = \frac{2L^2}{\lambda} \tag{2.32}$$

onde λ é o comprimento de onda e L a maior dimensão do arranjo. Se for necessário levantar com precisão os lóbulos secundários e nulos do sistema, esta distância pode ser maior, chegando a cinco vezes o estabelecido em Eq. 2.32. Note que, para arranjos de grandes dimensões, essa medição não é trivialmente realizável, sendo necessários complexos *setups* de testes, com amplas câmaras anecoicas.

Por fim, vale ressaltar a importância da existência de métodos para medir a estabilidade do sistema que emprega arranjos de antenas e, circuitos redundantes que possam ser reprogramáveis considerando os elementos com falhas. Ademais, é

imprescindível que seja possível desligar elementos que não estejam operando conforme o esperado, uma vez que é melhor ter um elemento a menos no arranjo, do que um operando de forma imprevisível.

2.5 Calibração e Alinhamento

A Eq.2.4 modela o comportamento do diagrama resultante de um arranjo em função das ponderações (amplitude e fase) inseridas em cada elemento de antena. Tais ponderações representam as diferenças de amplitude e fase entre os elementos necessárias para que o arranjo gere um diagrama da forma desejada, considerando que os elementos não apresentam diferença de amplitude e que a diferença de fase original entre eles é função apenas dos seus posicionamentos no arranjo. Dessa forma, resta claro que, ainda que sejam calculadas as respectivas ponderações ótimas a serem aplicadas, faz-se necessário medir inicialmente e periodicamente as diferenças de fase e amplitude inseridas pelo sistema, que podem ser constantes ou variáveis com o tempo de operação. A esse procedimento dá-se o nome de **calibração**. As variações citadas podem ser oriundas de diferenças no tamanho das trilhas, variações de temperatura ao longo do arranjo (uniforme ou não), degradação dos componentes eletrí´nicos, comprimento dos cabos, entre outros, sempre tomando um elemento como referência.

Existem diversas técnicas e algoritmos de calibração disponíveis na literatura [14]. Cada qual com vantagens e desvantagens, que incluem o tempo de execução, necessidade de *hardware* adicional, repetibilidade em campo, precisão e acurácia. É importante destacar que a calibração deve ser realizada tanto para o circuito de transmissão quanto para o de recepção, os quais podem apresentar resultados bem distintos. Para calibração da cadeia de transmissão, é necessário transmitir o sinal por cada um dos elementos e amostrá-los novamente para análise. Na recepção, em contrapartida, é necessário que um sinal de referência seja recebido por cada elemento individualmente para aferição das respectivas amplitudes e fases. Nesse contexto, fica claro a dificuldade adicional em se calibrar arranjos que utilizem subarranjos, uma vez que o acesso aos elementos de antena torna-se restrito.

Um dos melhores métodos para realização da calibração é através da inclusão de um *hardware* adicional (*sampler*) ao projeto do arranjo de antena. É possível transmitir e receber um sinal específico via *sampler*, para cada um dos elementos de antena de forma isolada, sendo possível, então, a calibração, respectivamente, das cadeias de recepção e transmissão do sistema. O método de calibração via *sampler* não necessita de nenhum *hardware* externo ao sistema e pode ser executado tanto em fábrica, quanto em campo, inclusive durante a operação do sistema.

Caso o sistema não preveja uma linha de calibração (*sampler*), é possível realizar a calibração através do uso de *hardwares* adicionais. Tais técnicas, normalmente, só são realizáveis em fábrica e dificilmente são reproduzidas em campo, pois seus desempenhos normalmente são melhores quando realizadas dentro de câmaras anecoicas. Vale lembrar,

que tais métodos são realizados em campo próximo, pois a aproximação de campo distante necessita de uma distância mínima regida pela Eq. 2.32. Esta pode ser realizada através do uso de uma antena de teste (*probe*) que varre o arranjo elemento por elemento, transmitindo e recebendo sinais de referência ao longo de toda sua extensão ou com *probes* periféricos instalados em pontos fixos do arranjo [15].

A primeira configuração, apesar de ser extremamente precisa, até por reduzir os efeitos do acoplamento entre a antena e a *probe* exige um alinhamento preciso entre esta e cada elemento do arranjo, normalmente feito via laser, e tem um tempo de execução e complexidade de *setup* maior que os demais, uma vez que deve levar em consideração a movimentação da *probe*. A segunda, considera que é possível cancelar o efeito do acoplamento com várias *probes* instaladas ao longo do arranjo e assim obter uma calibração precisa. Todavia, para isso, é necessário uma análise cuidadosa dos efeitos das diferentes *probes* combinadas.

Uma vez que o uso de *samplers* exige uma modificação de projeto, e que técnicas que utilizem *hardware* externo adicional podem não ser economicamente viáveis e facilmente reproduzidas, exigindo ainda o cuidado de sincronização com o *hardware* adicional, muito estudo vem sendo realizado pela comunidade científica para realização da calibração de arranjos de antenas baseada no acoplamento mútuo entre elementos vizinhos [16]. Conforme mencionado na seção anterior, sempre que um sinal é transmitido por um elemento de antena, este é recebido pelos demais (devido ao acoplamento) e pode ser analisado para calibração do arranjo.

É importante ressaltar que para realização desse tipo de calibração, duas premissas devem ser atendidas: os elementos de antena devem ser isotrópicos (ou simétricos ao longo das linhas e colunas), garantindo o acoplamento igual entre elementos, e deve ser possível o controle individual e independente de cada elemento do arranjo. Para que este método seja viável, é necessário, também, que o acoplamento entre elementos seja alto suficiente para ser detectado, lembrando que este não pode saturar o receptor. Dessa forma, é comum que a potência de transmissão seja alterada objetivando a adequação de amplitude do sinal para realização deste tipo de calibração [14].

Os supracitados métodos são ilustrados na Figura 2.16.

Éimportante destacar que a calibração deve ser realizada para cada modo de operação do sistema, considerando as diversas frequências e temperaturas de operação, e os diversos níveis de quantização dos atenuadores e defasadores empregados (quando empregados). Isto nem sempre é possível, inserindo um compromisso entre as amostras selecionadas para realização das medidas e o desempenho da calibração.

Por fim, vale destacar que a calibração não somente é vital para o correto funcionamento de um arranjo de antenas, como também, através dela, pode-se derivar o diagrama de radiação da antena, via transformada de Fourier da distribuição de amplitude amostrada. Logo, recomenda-se que esse procedimento (independente da técnica utilizada) seja realizado em fábrica e de forma periódica durante operação.

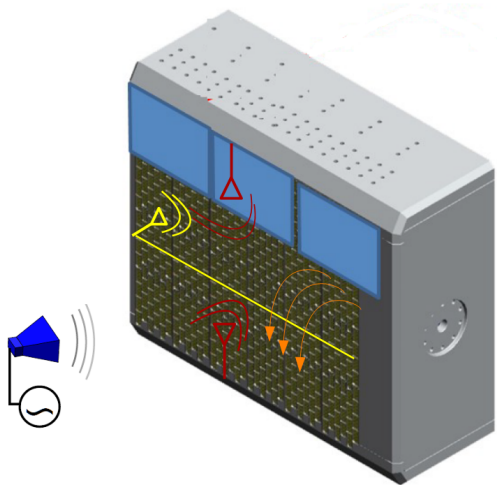


Figura 2.16 – Métodos de calibração: amarelo - *sampler*; laranja - acoplamento mútuo; azul - via *probe* externa; vermelho - via *probe* periférico.

2.6 Aplicações

Nessa seção serão apresentadas algumas aplicações para arranjo de antenas no intuito de indicar ao leitor a importância em se familiarizar com tais sistemas. Arranjos de antenas com varredura de feixes eletrí´nica foram desenvolvidos, inicialmente, para aplicação militar, em especial radares de rastreio para defesa antiaérea, durante a década de 1950 impulsionados pela invenção dos primeiros defasadores de fase. Com o passar do tempo, e a evolução tecnológica, de *hardware* e *software*, percebeu-se a imensa aplicabilidade de tal tecnologia, fazendo com que o uso extrapolasse as aplicações militares (o que continua até os dias atuais), sendo usada em diversas áreas civis, tais como comunicação, medicina e mapeamento. Algumas dessas aplicações serão apresentadas em seguida.

2.6.1 Radares militares

Nos dias de hoje, arranjos de antenas são possivelmente, o tipo de antena mais empregada em sistemas de radares militares. Isso porque esses arranjos fornecem alta confiança em sua aplicabilidade, não necessitam de partes móveis e motores, fazem varredura de forma rápida e eficiente e possibilitam um excelente controle de lóbulos secundários e criação de diagrama. Existem diversos tipos de radares militares que usam tal tecnologia. Citaremos em seguida alguns deles.

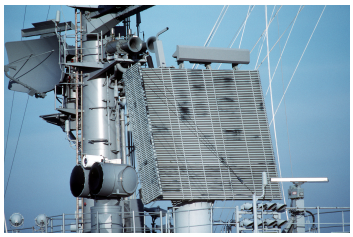
Radar de defesa antiaérea

Apesar de muitos radares utilizarem antena fixa com varredura mecânica para busca e vigilância de setores aéreos, radares mais modernos utilizam arranjo de antenas para auxiliar nessa tarefa, tendo varredura completamente eletrí´nica ou então fazendo varredura mecânica em azimute e eletrí´nica em elevação. Nesse último caso, um feixe estreito é definido em azimute, enquanto feixes multiplexados no tempo são direcionados em elevação, ou então, transmite-se com um largo feixe em elevação (geralmente seguindo

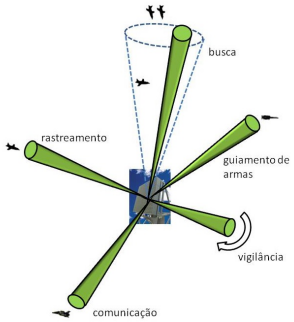
um padrão de cossecante ao quadrado) e na recepção utiliza-se a técnica de *digital beamforming*. O radar AN/SPS-48, por exemplo, utilizado em navios da Marinha americana, é um radar de busca 3-D, que utiliza arranjo de antena planar onde a varredura em azimute é mecânica e em elevação é eletrí´nica e seus elementos de antena são guias de onda.

Devido í possibilidade de paralelizar os processamentos em um arranjo de antenas, além da alta taxa de direcionamento dos feixes e otimização de diagrama resultante, um sistema como esse pode compor radares do tipo multifunção. Com esses radares é possível um único sistema desempenhar diversas atividades, geralmente multiplexadas no tempo, tais como rastreio de alvos, busca e vigilância setoriais e guiamento de mísseis e foguetes, podendo ser utilizados em solo ou acoplados em navios.

Para realizar diversas tarefas distintas, sistemas dessa natureza fazem uso de um gerenciador de recursos, o qual é responsável por listar as tarefas em ordem de prioridade e agendar a realização das mesmas, definindo o formato de diagrama a ser usado e a frequência de transmissão. Os radares mais atuais ainda permitem que o sistema possua propriedades cognitivas, as quais com base no cenário de operação atual garantem mudanças específicas na forma de onda transmitida, diagrama de antena e recursos utilizados. Por ser um sistema complexo e multidisciplinar, ultimamente radares multifunção são alvos de discussões e trabalhos ao redor do mundo, tornando-se um dos assuntos mais em evidência em simpósios e congressos sobre sistemas de radar.



(a)



(b)

(c)

Figura 2.17 – Aplicações em radares militares: (a) Radar AN/SPS-48, produzido pela ITT Gilfillan e utilizado pela Marinha americana; (b) exemplo de tarefas efetuadas em radares multifunção; (c) Radar de contrabateria AN/TPQ-36 da Raytheon

Radar aerotransportável

Radares aerotransportáveis podem ser usados para guiamento de míssil lançado pela aeronave, detecção, identificação e rastreio de ameaças aéreas, criação de *datalink* com outras aeronaves e centros de controle e combate ar-superfície. Arranjo de antenas são bem úteis nesse tipo de aplicação pois têm a capacidade de direcionar o feixe eletronicamente de um extremo ao outro ao mesmo tempo que mantém o tamanho relativamente pequeno, o que minimiza o efeito de arrasto na aeronave. Além disso, é possível utilizar *beamforming* adaptativo para cancelar alvos indesejáveis e interferências múltiplas. Esses sensores são colocados normalmente no bico da aeronave. A arquitetura empregada pode ser passiva ou ativa, apesar de prevalecer a última em radares mais modernos, já que propicia a realização das multitarefas citadas em um único sistema.

Radar de Contrabateria

Radares desse tipo consistem em sistemas cujo objetivo é detectar fogo lançado por armas inimigas, tais como morteiros, obuses e foguetes, e com base nos dados obtidos estimar a posição do ponto de lançamento desses projéteis, fornecendo informação para um sistema de arma poder atuar em contra-ataque í bateria inimiga. Um objetivo secundário desses radares é estimar o ponto de impacto dos projéteis a fim de tentar diminuir os danos que serão causados por estes. Radares de contrabateria mais modernos geralmente são do tipo tridimensional, utilizam arranjo de antenas a fim de possibilitar o escaneamento eletrí´nico em azimute, varrendo o setor definido com feixes atualizados através de uma alta taxa. Após um alvo penetrar nessa área e ser detectado, passará a ser rastreado por um único feixe direcionado somente a ele. Enquanto o rastreio permanece, o radar multiplexa no tempo essa tarefa com tarefa de varredura do setor e/ou outros. rastreamentos.

2.6.2 Comunicações Móveis - Tecnologia 5G

Devido ao alto crescimento na demanda de serviços sem fio para transmissão de voz, dados e vídeos, a tecnologia de arranjo de antenas vem ganhando força nesse setor. O uso de um arranjo de antenas ativo aumenta consideravelmente a capacidade da rede de celulares. Isso porque, devido a alta densidade populacional em centros urbanos, a relação Sinal-Interferência (SIR) torna-se um problema. Tendo um sistema onde é possível otimizar a transmissão/recepção, colocando nulos nas direções dessas interferências (como já citada a possibilidade no primeiro capítulo), é possível diminuir a influência destas nos sinais desejados. Além disso, é possível estreitar o lóbulo principal permitindo melhorar a precisão de posicionamento em áreas onde a cobertura GNSS é prejudicada.

No contexto de comunicações móveis, a tecnologia 5G prevê a utilização de elevadas frequências de transmissão, da ordem de dezenas de GHz - banda de frequência chamada de mm-Wave. Devido í s características de tal tecnologia, arranjos de antenas se tornam uma boa solução. Feixes em um arranjo planar podem ser direcionados para qualquer ponto

no espaço, possibilitando o aumento da capacidade do canal de sistemas de comunicação pessoal sem fio, alcançando altas taxas de dados, além de diminuir a relação sinal-interferência com sua otimização de síntese de diagrama [17]. Além disso, arranjos de antenas diminuem os requisitos de potência para os amplificadores, visto que a potência de transmissão total é dada pela soma da potência em cada elemento, ou seja, se cada elemento possui um transmissor de P watts, o arranjo transmitirá em seu lóbulo principal $20log_{10}P$ dB de potência [18].

Outra vantagem de se utilizar arranjos de antenas é que o direcionamento eletrí´nico do feixe (antena estática) permite que dois ou mais arranjos sejam colocados relativamente mais próximos quando comparados a antenas omnidirecionais mesmo utilizando o mesmo canal, sem comprometer o desempenho do serviço. Isso é possível pois cada arranjo pode conformar seu respectivo diagrama de radiação de modo a não interferir nos demais. Dessa forma, o tamanho da célula utilizada diminui, o que significa um maior número de usuários móveis concentrados em uma menor área sem aumentar alocação de espectro.

Finalmente, esse sistemas garantem links mais robustos e confiáveis devido a atenuação dos problemas de interferência no mesmo canal de frequência (como já mencionado) e multi-percurso, além da supressão de sinais provenientes de direções indesejáveis. Em contrapartida, vale mencionar que um desafio para o uso de arranjos em tal aplicação é o fato da necessidade de miniaturização dos elementos de antena e circuitos eletrônicos para compor a arquitetura desejada.

2.6.3 Aeroespacial-Satélites

Os sistemas de satélite *Broadcast* também podem se beneficiar de arranjo de antenas, reduzindo a potência de transmissão necessária ou aumentando a capacidade de comunicação em uma determinada saída do amplificador. Alguns sistemas atuais de TV via satélite já utilizam tal tecnologia. Comparado ao sistema parabólico tradicional, arranjo de antenas são mais robustos quanto a mudanças climáticas, tendo menor perfil e menor peso, o que lhes permite ser mais facilmente montados em paredes e telhados [19]. Além disso, a conformação de feixes adaptativa permite objetos móveis como aviões terem acesso a programas de TV via satélite, já que é possível manter o feixe principal direcionado a um determinado satélite, mesmo com o movimento da plataforma onde aquela se encontra. Essa característica também facilita o direcionamento do feixe sem afetar translação mecânica da antena, o que requer um movimento de outras partes do satélite a fim de deixá-lo estabilizado.

Além da utilização em satélites para transmissão de sinal de TV, arranjos de antena podem ser usados em satélites de mapeamento. Esses satélites geralmente operam em baixas órbitas (centenas de quilí´metros de altitude) e usam, dentre outras técnicas, *beamforming* para obter altíssima resolução de pontos da superfície terrestre, assim como nas áreas de hidrosfera, atmosfera, geosfera e biosfera.

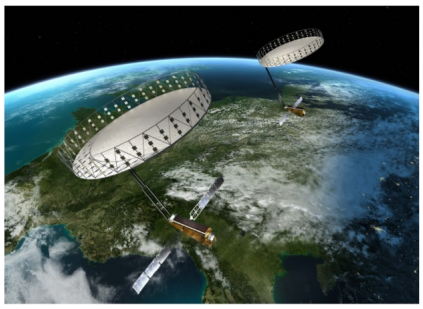


Figura 2.18 – Tandem-L - Satélite desenvolvido pelo DLR, da Alemanha, para imageamento terrestre.

2.6.4 Medicina

Um uso recente de arranjo de antenas consiste em auxiliar na radiação em tratamentos de tumor em órgãos como pulmão e fígado. Com o uso do direcionamento eletrí´nico de feixes, é possível orientar o feixe para medir de forma simultânea os sinais fisiológicos do paciente em pontos diferentes, a partir do local estimado de onde o tumor possa estar [20], conforme figura abaixo [21]. Assim, arranjos de antenas podem ser usados em conjunto com equipamentos de radioterapia.

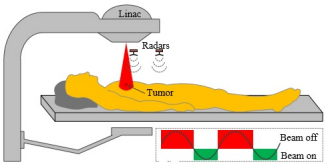


Figura 2.19 – Uso de radares com arranjo de antenas em radioterapia

Arranjos de antenas já estão sendo usados também em imageamento médico para detecção de câncer de mama em estágio inicial [22], assim como câncer de próstata [23]. A imagem criada através da emissão de micro-ondas pode fornecer uma probabilidade de detecção bem maior que a de raio-X ou ultrassom, além de ser menos prejudicial ao paciente comparado ao raio-X. Em termos de custo, o imageamento por micro-ondas é menor do que soluções alternativas e bem consolidadas no mercado tais como MRI.

Referências Bibliográficas

[1] O.G. Vendik and Y.V. Yegorov. The first phased-array antennas in russia: 1955-1960. *Antennas and Propagation Magazine, IEEE*, 42(4):46–52, 2000.

[2] A.J. Fenn, D.H. Temme, W.P. Delaney, and W.E. Courtney. The development of phased-array radar technology. *Lincoln Laboratory Journal*, 12(2):321–340, 2000.

[3] M.I. Skolnik. *Radar Handbook, Third Edition*. Electronics electrical engineering. McGraw-Hill, 2008.

[4] D. Halliday, R. Resnick, and Walker J. *Fundamentos da Física 4 - Ótica e Física Moderna, 4th ed.* LTC, 1995.

[5] R.J Mailloux. *Phased array antenna handbook*. Artech House Boston, 2017.

[6] F.J. Harris. On the use of windows for harmonic analysis with the discrete fourier transform. *Proceedings of the IEEE*, 66(1):51–83, 1978.

[7] Robert C Hansen. *Phased array antennas*, volume 2. John Wiley & Sons, 2009.

[8] S. Sabatini and M. Tarantino. *Multifunction array radar- System design and analysis*. Norwood, MA, Artech House, 1994.

[9] RL Howard, LE Corey, and SP Williams. The relationship between dispersion loss, sidelobe levels, and bandwidth in wideband radars with subarrayed antennas. In *IEEE AP-S. International Symposium, Antennas and Propagation*, pages 184–187. IEEE, 1988.

[10] Mats Gustafsson, Christian Sohl, and Gerhard Kristensson. Physical limitations on antennas of arbitrary shape. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 463(2086):2589–2607, 2007.

[11] D Cheng. Effect of arbitrary phase errors on the gain and beamwidth characteristics of radiation pattern. *IRE Transactions on Antennas and Propagation*, 3(3):145–147, 1955.

[12] K Carver, W Cooper, and W Stutzman. Beam-pointing errors of planar-phased arrays. *IEEE Transactions on Antennas and Propagation*, 21(2):199–202, 1973.

[13] John Ruze. Antenna tolerance theory: A review. *Proceedings of the IEEE,* 54(4):633–640, 1966.

[14] Ilgın Şeker. Calibration methods for phased array radars. In *Radar Sensor Technology XVII*. International Society for Optics and Photonics, 2013.

[15] M Sarcione, J Mulcahey, D Schmidt, K Chang, M Russell, R Enzmann, P Rawlinson, W Guzak, R Howard, and M Mitchell. The design, development and testing of the thaad (theater high altitude area defense) solid state phased array (formerly ground based radar). In *Proceedings of International Symposium on Phased Array Systems and Technology*, pages 260–265. IEEE, 1996.

[16] Herbert M Aumann, Alan J Fenn, and Frank G Willwerth. Phased array antenna calibration and pattern prediction using mutual coupling measurements. *IEEE Transactions on Antennas and Propagation*, 37(7):844–850, 1989.

[17] Alberto Valdes-Garcia, Sean Nicolson, Jie-Wei Lai, Arun Natarajan, Ping-Yu Chen, Scott Reynolds, Jing-Hong Conan Zhan, and Brian Floyd. A sige bicmos 16-element phased-array transmitter for 60ghz communications. In *2010 IEEE International Solid-State Circuits Conference-(ISSCC)*, pages 218–219. IEEE, 2010.

[18] Nan Guo, Robert C Qiu, Shaomin S Mo, and Kazuaki Takahashi. 60-ghz millimeter-wave radio: Principle, technology, and new results. *EURASIP journal on Wireless Communications and Networking*, 2007(1):48–48, 2007.

[19] Danial Ehyaie. Novel approaches to the design of phased array antennas. 2011.

[20] Han Ren, Jin Shao, Bayaner Arigong, Hualiang Zhang, Changzhan Gu, and Changzhi Li. Application of phased array antenna for radar respiration measurement. In *Proceedings of the 2012 IEEE International Symposium on Antennas and Propagation*, pages 1–2. IEEE, 2012.

[21] Changzhan Gu, Ruijiang Li, Steve B Jiang, and Changzhi Li. A multi-radar wireless system for respiratory gating and accurate tumor tracking in lung cancer radiotherapy. In *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 417–420. IEEE, 2011.

[22] Qian Liu, Qingming Luo, and B Chance. 2d phased array fluorescence wireless localizer in breast cancer detection. In *2004 2nd IEEE/EMBS International Summer School on Medical Devices and Biosensors*, pages 71–73. IEEE, 2004.

[23] Arnauld Villers, Philippe Puech, Damien Mouton, Xavier Leroy, Charles Ballereau, and Laurent Lemaitre. Dynamic contrast enhanced, pelvic phased array magnetic resonance imaging of localized prostate cancer for predicting tumor volume: correlation with radical prostatectomy findings. *The Journal of urology*, 176(6):2432–2437, 2006.

CAPÍTULO

3 Aprendizado de Máquina Aplicado a Localização de Usuários em Redes sem Fio: Oportunidades e Desafios

Larissa L. Oliveira (University of Guelph), Gustavo P. Bittencourt (Tempest Security Intelligence), Lizandro N. Silva (Instituto Federal de Pernambuco-IFPE), Robson D. A. Timoteo (Centro de Informática da UFPE), Daniel C. Cunha (Centro de Informática da UFPE) e Felipe S. Andrade (Google).

Introdução

Em vários centros urbanos, o crescimento da população ocasiona problemas de infraestrutura e de acesso limitado a recursos prejudicando a vida de milhões de pessoas. Uma maneira de melhorar a qualidade de vida dos cidadãos é prover serviços mais eficientes por meio do conceito de cidade inteligente. Não existe um consenso sobre a definição deste conceito na literatura, mas uma cidade é dita ”inteligente” quando investimentos em capital social e humano, aliados a uma infraestrutura moderna de comunicação, fomenta o crescimento econômico sustentável associado a uma alta qualidade de vida, com gerenciamento sustentável dos recursos naturais [1]. Sob o ponto de vista da Engenharia, as cidades inteligentes são estabelecidas por meio de uma infraestrutura avançada em conjunto com tecnologias de informação e comunicação modernas.

Nesse contexto, as tecnologias de localização desempenham um papel fundamental para aplicações que promovem o desenvolvimento das cidades inteligentes. As técnicas de localização se tornaram populares com o advento do sistema de posicionamento global (GPS, *global positioning system*) para aplicações *outdoor* e, nos últimos anos, com as redes Wi-Fi e *Bluetooth* para as aplicações *indoor*. Mais recentemente, a tecnologia de posicionamento tem se expandido para aplicações médicas *in-body*, ou seja, a localização de anomalias (lesões, tumores, sangramentos ou, simplesmente, dor) dentro do corpo humano [2, 3].

A diversidade de tecnologias de localização surge principalmente devido aos requisitos de acurácia associados a cada tipo de serviço de localização (*location-based service*). Diferentes aplicações possuem diferentes exigências quanto à sua acurácia. Exemplos clássicos são os serviços de localização *outdoor* e *indoor*: enquanto o primeiro exige uma acurácia da ordem de dezenas ou até mesmo centenas de metros, o segundo impõe exigências para manutenção da margem de erro em alguns poucos metros. Outro exemplo bastante atual é o desenvolvimento de tecnologias de transporte autônomo (*self-driving*) que podem chegar a exigir um nível de acurácia da ordem de alguns poucos centímetros.

Para atender as diferentes exigências de exatidão, podem ser utilizados diversos tipos de sensores. Apenas para citar alguns exemplos, existem desde sensores baseados em rádio frequência (RF), responsáveis pela captura de sinais através de GPS, Wi-Fi, *iBeacon* (*Bluetooth*) e torres de telefonia móvel, até sensores mecânicos, como magnetômetro e o acelerômetro. A análise das características comportamentais desse conjunto de sensores, juntamente com a escolha de algoritmos que satisfaçam os requisitos de acurácia para uma determinada aplicação, consiste no grande desafio imposto às tecnologias de localização, sendo responsável por impulsionar o desenvolvimento de uma nova área de pesquisa ao longo da última década.

As primeiras tecnologias de localização sem fio desenvolvidas utilizavam modelos matemáticos determinísticos para estimar a posição de um dado objeto. O GPS, por exemplo, faz uso das diferenças no tempo de chegada para derivar a posição relativa do objeto em relação a um conjunto de satélites utilizados como referencial. Outras tecnologias, como a multilateração, utilizam a atenuação na potência do sinal recebido com relação a diferentes referenciais como medida para determinar a posição do objeto através de modelos de propagação eletromagnética. Por estarem baseadas em modelos matemáticos determinísticos, tais técnicas tendem a ser impactadas por efeitos colaterais indesejados como a presença de ruído, dispersão, sombreamento, entre outros. Esses efeitos atribuem características não lineares ao sistema em questão, dificultando sua modelagem matemática e, em última instância, influenciando em sua acurácia.

Devido à complexidade natural em modelar sistemas não lineares, a aplicação de algoritmos de aprendizado de máquina em técnicas de localização parece uma escolha natural, exatamente pelos benefícios trazidos por essa tecnologia na tratativa dessa classe de problemas [4, 5, 6, 7]. A modelagem de sistemas complexos pode não ser tão óbvia e nem sempre sua implementação pode ser viável. Sendo assim, as técnicas de aprendizado de máquina tem como principal característica reduzir a dificuldade de implementação desses sistemas ao aproximar seus resultados com performance comparável, sem que seja necessário implementar explicitamente as heur'isticas e os algoritmos intrínsecos à sua modelagem.

Este capítulo está distribuído conforme a seguir. A Seção 3.1 traz uma revisão sobre localização, explorando parâmetros de sinal, classificação e as técnicas básicas de sistemas de localização de uma rede sem fio, com destaque para trilateração e *fingerprinting*. A

Seção 3.2, por sua vez, apresenta conceitos básicos de aprendizado de máquina, com êfase no algoritmo *k*-NN e na máquina de vetor de suporte. Na Seção 3.3, a aplicação da técnica *fingerprinting* baseada em máquinas de vetor de suporte é explicada em detalhes. Finalmente, a Seção 3.4 apresenta algumas oportunidades e desafios referentes à aplicação de técnicas de aprendizado de máquina em localização de dispositivos móveis em redes sem fio.

3.1 Revisitando técnicas e tecnologias de localização

Antes de recordarmos técnicas e tecnologias de localização de usuários em redes sem fio, é importante diferenciarmos três conceitos fundamentais, quais sejam, posição, localização e navegação. A *posição* corresponde às coordenadas geográficas do dispositivo no globo terrestre, ou seja, é representada pela latitude, longitude e altitude do ponto. A *localização* se refere ao contexto específico de um ponto, como, por exemplo, o endereço completo de uma residência. E por último, a *navegação* descreve a trajetória de um dispositivo para sair de um ponto e chegar a outro [8]. No minicurso em questão, assumiremos que localização e posição são sinônimos, ou seja, que localização também se refere Ãàs coordenadas geográficas do dispositivo que se deseja encontrar.

A localização de usuários móveis em uma rede sem fio é possível por meio de parâmetros dos sinais de RF. Para permitir a estimativa da posição do dispositivo móvel, certos parâmetros devem ser medidos, como, por exemplo, a potência recebida do sinal. Entretanto, o parâmetro medido deve ter uma relação física com a posição do dispositivo. Caso contrário, tal informação passa a ser irrelevante para fins de posicionamento. Considerando que a relação entre os parâmetros coletados e a posição do dispositivo é corrompida pelo ruído adicionado pelo canal de comunicação, um sistema de posicionamento possui, em geral, uma complexidade inerente. Por esse motivo, é necessário que haja mecanismos matemáticos capazes de manipular os parâmetros coletados para extrair as coordenadas geográficas da posição. Os métodos mais avançados incluem um tratamento estatístico dos erros nas medições dos parâmetros de sinal [9].

No âmbito dos sistemas de localização aplicados a redes sem fio, a presente seção apresenta os tipos mais comuns de parâmetros de sinais de RF utilizados em técnicas de localização, uma classificação baseada em características de topologia e propagação eletromagnética e, por fim, as técnicas de localização mais difundidas na literatura.

3.1.1 Tipos de parâmetros de sinal em redes sem fio

Os principais parâmetros extraídos de sinais de RF para a estimação da posição de um dispositivo móvel em redes sem fio são o tempo de chegada, a diferença entre tempos de chegada, o ângulo de chegada e a intensidade de sinal recebido. A seguir, vamos detalhar de forma básica cada um deles.

O tempo de chegada (ToA, *time of arrival*) de um sinal de RF é o tempo decorrido entre a saída do sinal do transmissor e a sua chegada no receptor. Assumindo que t_S é o instante de tempo de transmissão do sinal a partir da estação móvel (EM) e t_i é o instante de tempo da recepção do sinal na i-ésima estação radio base (ERB), podemos concluir que o ToA é o intervalo de tempo $\tau_i = t_i - t_S$. De posse do ToA e conhecida a velocidade de propagação do sinal de RF, é possível estimar a distância d_i entre a EM e a ERB. Uma vez que as coordenadas geográficas das ERBs são conhecidas e utilizando uma interpretação geométrica, é possível estabelecer um sistema de equações para a obtenção da posição da EM.

Embora seja robusta, a técnica de localização baseada em ToA apresenta como principal desvantagem a necessidade de que os transmissores e receptores envolvidos estejam perfeitamente sincronizados. Pequenos erros de sincronismo podem levar a erros consideráveis no cálculo das distâncias relacionadas. Para resolver este problema do ToA, faz-se uso da diferença entre tempos de chegada (TDoA, *time difference of arrival*) dos sinais recebidos em duas ERBs. Dessa forma, a estimativa da TDoA remove a necessidade de sincronismo entre ERBs e EM, uma vez que apenas o sincronismo entre as ERBs irá garantir a minimização do erro. Uma das razões para a remoção do sincronismo entre ERBs e EM é a redução de complexidade do sistema.

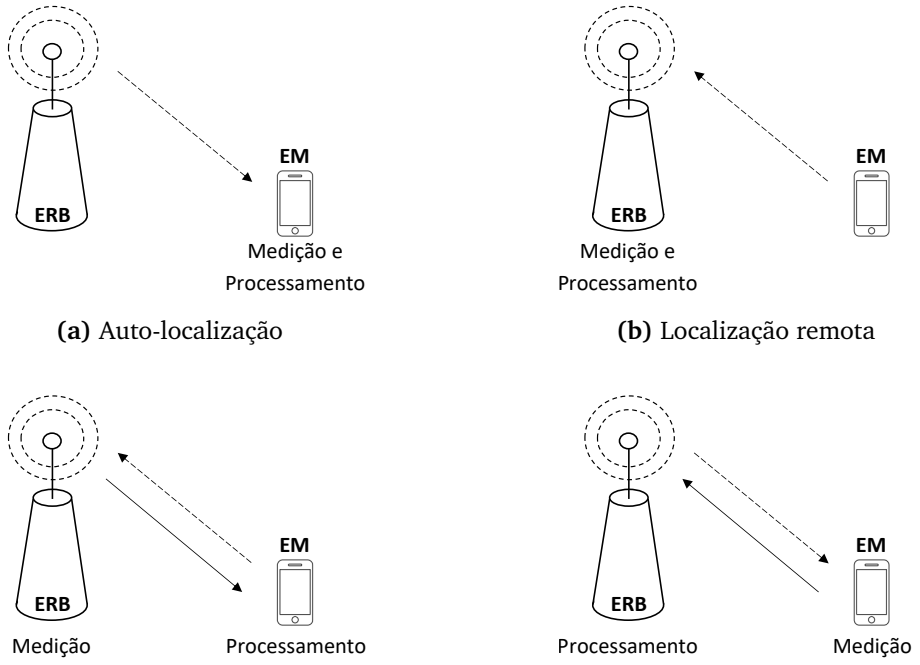
O terceiro parâmetro de um sinal de RF é o ângulo de chegada ao receptor (AoA, *angle of arrival)*. Para a utilização deste parâmetro, é necessário o uso de antenas diretivas, do conhecimento exato da topologia da rede, bem como dos pontos em que os transmissores estão instalados. O maior problema deste tipo de parâmetro é o sombreamento causado por obstáculos ao longo da trajetória do sinal entre o transmissor e o receptor. O sinal pode ser refletido e chegar no dispositivo com um ângulo diferente do esperado [10].

Finalmente, temos os sistemas de localização baseados na intensidade ou indicador de nível do sinal de RF recebido (RSSI, *received signal strength indicator*). Geralmente, o nível de um sinal de RF pode ser estabelecido por um modelo de propagação eletromagnética, no qual diversos efeitos de propagação são contemplados, tais como atenuação, sombreamento, propagação multipercurso e outros, assim como componentes de ruído e interferência. A partir de modelos de propagação, como, por exemplo, o modelo de perda por espaço livre, é possível calcular a distância máxima em que o sistema pode fornecer cobertura para os usuários. Portanto, conhecido o RSSI, é possível estimar a distância entre o dispositivo móvel e a ERB [11].

A partir deste ponto, iremos focar nos sistemas de localização baseados em RSSI. Dito isso, é importante entender como e onde o RSSI é utilizado, seja na EM ou na ERB. Isto implicará em como os sistemas de localização sem fio podem ser classificados, conforme será abordado a seguir.

3.1.2 Classificação de sistemas de localização de redes sem fio

Sistemas de localização podem ser classificados conforme diversas características. Uma delas está relacionada à topologia do sistema, que, de acordo com [12], se refere ao equipamento onde os níveis de sinal são medidos e também onde as medições obtidas são processadas para estimar as informações de posicionamento do móvel. A Figura 3.1 ilustra quatro cenários nos quais os parâmetros podem ser medidos/processados somente na EM, somente na ERB ou ainda em ambas as estações, sendo cada operação (medição/processamento) realizada em uma das estações (EM e ERB). Quando a EM realiza a medição e o processamento, o sistema é classificado como de *auto-localização* (Figura 3.1a). Para permitir que a EM se auto-localize, ela precisa conhecer a localização da ERB. Essa informação pode ser conhecida pela rede ou ser enviada por cada ERB durante a comunicação com a EM. Quando medição e processamento do sinal são executados pela rede, ou seja, por meio das ERBs, o sistema é classificado como de *localização remota* (Figura 3.1b). Quando as medições são obtidas por uma estação e processados em outra, o sistema emprega localização indireta. Em outras palavras, quando a ERB executa as medições do sinal e a EM as processa, o sistema é classificado como de *auto-localização indireta* (Figura 3.1c). Caso contrário, o sistema é classificado como de *localização remota indireta* (Figura 3.1d).



(a) Auto-localização **(b)** Localização remota

(c) Auto-localização indireta **(d)** Localização remota indireta

Figura 3.1 – Classificação do sistema de localização de acordo com a sua topologia para medição e processamento dos níveis de sinais de RF. As linhas tracejadas representam um enlace de comunicação usado para medir o nível do sinal de RF e as linhas sólidas representam uma transferência real de dados medidos. Fonte: Adaptado de [13].

Outra forma de classificar sistemas de localização é com base na cobertura da rede sem fio. Neste caso, os sistemas de localização são divididos em três categorias, quais sejam,

sistemas baseados em satélites, baseados em redes de telefonia celular e, por fim, em redes de pequena cobertura.

O principal exemplo de sistema de localização baseado em satélites é o GPS, desenvolvido pelo Departamento de Defesa dos EUA com propósito militar e liberado posteriormente para uso civil. Suas principais aplicações são em navegação, aviação, topografia, controle de frotas e agricultura de precisão, por exemplo. Muito utilizado para prover a posição de um usuário móvel em ambientes externos (*outdoor*) [14], o GPS possui uma acurácia da ordem de $3\,m$. Apesar disso, o sistema possui limitações em virtude da propagação de sinais eletromagnéticos em dias chuvosos e em áreas com alta densidade urbana, assim como em ambientes internos (*indoor*).

Em redes celulares, nas quais a cobertura está diretamente relacionada com a área de alcance das ERBs, independente da geração da rede (2G, 3G ou 4G), a acurácia é da ordem de 100 m [15]. Os sistemas de localização baseados em redes celulares são bastante utilizados em ambientes urbanos e rurais. No primeiro caso, os problemas de propagação de sinal estão relacionados a obstáculos no percurso, o que nos remete aos sistemas sem linha de visada (NLoS, *non-line of sight*), sujeitos aos efeitos do sombreamento e da propagação multipercurso [11]. No segundo caso, os problemas de obstrução com a vegetação interferem na propagação dos sinais. Por outro lado, número reduzido de ERBs instaladas para prover cobertura em grandes zonas rurais pode reduzir a acurácia dos dos sistemas de localização [16].

Finalmente, em sistemas de pequena cobertura, em que a rede se baseia em tecnologias, como, por exemplo, redes locais sem fio (WLAN, *wireless local area networks*) e *Bluetooth*, a acurácia é da ordem de 10 m [15]. Neste caso, os problemas relativos à propagação do sinal dependem do cenário em que a rede está instalada e seu uso é apropriado para ambientes internos (*indoor*) [11].

Uma vez classificados os sistemas de localização baseados em RSSI, torna-se necessário entender como um dispositivo móvel pode ser encontrado em uma rede sem fio.

3.1.3 Técnicas Básicas de Localização

Nesta Seção, serão apresentadas duas técnicas básicas de localização, quais sejam, a técnica baseada em lateração e a técnica de *fingerprinting*.

Define-se por lateração, a determinação da localização de um ponto (EM, no caso) por meio de argumentos geométricos, conhecidas as distâncias deste ponto a um certo número de pontos de referência (as ERBs, em nosso contexto) e as coordenadas geográficas dessas referências. Quando o número de pontos de referência é igual a três, a técnica recebe o nome de *trilateração*. Embora seja possível utilizar mais pontos de referência (caso da multilateração), vamos direcionar nossas explicações à trilateração.

No método da trilateração, as distâncias estimadas entre a EM e as três ERBs consideradas são obtidas a partir do RSSI extraído em cada enlace de comunicação [17, 18]. Cabe ressaltar que, na trilateração, a distância também pode ser obtida a partir do ToA. No

entanto, como o nosso foco está direcionado aos sistemas de localização baseados em RSSI, as distâncias são obtidas com base em modelos de propagação eletromagnética.

A Figura 3.2 ilustra o diagrama representativo da técnica de trilateração de potência, na qual três pontos de referência são indicados, assim como o ponto que se deseja estimar a posição. No contexto das redes celulares, as ERBs estão identificadas como Referências 1, 2 e 3, com coordenadas (x_1, y_1), (x_2, y_2) e (x_3, y_3), respectivamente. O ponto cuja posição (x, y) desejamos estimar representa a EM.

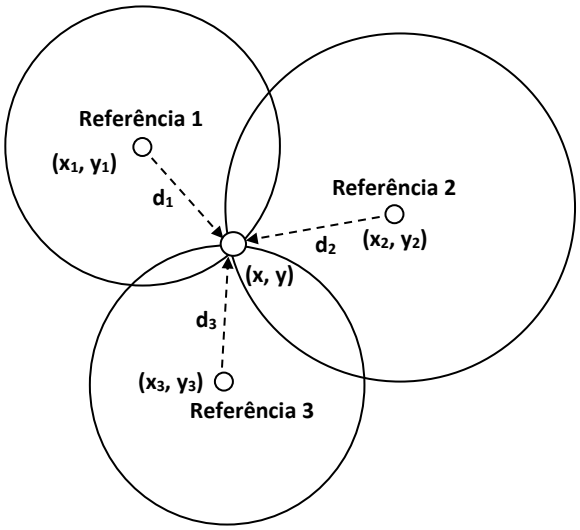


Figura 3.2 – Representação gráfica do caso ideal (não ruidoso) da trilateração de potência. Adaptado de [17].

Conforme já foi mencionado anteriormente, a partir do RSSI obtido em cada uma das ERBs, as distâncias d_1, d_2 e d_3 são estimadas com base em modelos de propagação. De posse das três equações que representam as circunferências centradas em cada uma das ERBs, podemos expressar o problema da trilateração como um sistema de equações não lineares, tal que

$$\begin{cases} (x-x_1)^2+(y-y_1)^2 &= d_1^2 \\ (x-x_2)^2+(y-y_2)^2 &= d_2^2 \\ (x-x_3)^2+(y-y_3)^2 &= d_3^2 \end{cases}$$

e cuja solução será a posição (x, y) da EM, que é a estimativa da localização desejada. Tal sistema pode ser resolvido por meio de métodos matemáticos, como, por exemplo, o método de Newton-Raphson [19] ou o de Nelder-Mead [20].

A técnica de trilateração de potência pode ser utilizada em qualquer ambiente de propagação, seja ele urbano, suburbano ou rural. Para este último, há uma peculiaridade, visto que, em várias regiões, apenas uma ou duas ERBs já são suficientes para prover cobertura em um raio de quilômetros. Neste caso, a trilateração não pode ser utilizada, uma vez que são necessários três RSSIs obtidos a partir de ERBs distintas.

A segunda técnica básica de localização abordada neste minicurso é a técnica de *fingerprinting*. Na realidade, as técnicas de *fingerprinting* correspondem a um grupo de métodos para a localização de usuários móveis, bastante adequados para ambientes sem

visada direta e que podem ser aplicados em qualquer modalidade de rede sem fio [21, 22]. Existe uma enorme variedade de técnicas de *fingerprinting*, mas como mostrado em [21], todas compartilham os mesmos elementos básicos, como, por exemplo, o *fingerprint*, o banco de dados, o servidor de localização, a técnica de redução de espaço de busca e o método de reconhecimento de padrões.

O vetor com todas os atributos (valores observados) utilizadas no método de reconhecimento de padrões é denominado *fingerprint*. Embora os parâmetros de sinal comumente usados como atributos possam ser diversos, iremos enfatizar o uso do RSSI para esta finalidade. O banco de dados, também conhecido como base de dados de correlação (CDB, *correlation database*), é implementado não apenas a partir de dados coletados em campo, mas também de predições realizadas por modelos de propagação de rádio, como, por exemplo, o modelo COST-231 [21]. Cada *fingerprint* armazenado no banco de dados é vinculado a uma posição específica. Nesse cenário, como não é viável fazer medições em todas as posições, modelos de propagação são usados para generalizá-los. O servidor de localização é o elemento de rede responsável por receber solicitações de localização, consultar o banco de dados e estimar a posição da EM.

Para estimar a posição da EM, o servidor de localização busca a similaridade do *fingerprint* medido (EM procurada) com um *fingerprint* armazenado no CDB. A ideia chave é encontrar o ponto no CDB que tenha a maior similaridade ou correlação com o valor medido em campo.

A Figura 3.3 mostra um diagrama esquemático simplificado para a técnica de localização baseada em *fingerprinting*. Na Etapa 1, o cliente envia uma solicitação ao servidor de localização. Em seguida, o servidor solicita medições da EM (destino) através da rede de acesso de rádio. Na etapa 3, a rede de acesso envia medições para o servidor de localização. Depois de receber as medições, o servidor de localização compõe o *fingerprint* e, na etapa 4, consulta o banco de correlação para obter a área de pesquisa reduzida. Na etapa 5, o servidor de localização recebe os resultados da consulta e utiliza uma função de comparação para obter a posição estimada da EM. Finalmente, na etapa 6, o servidor de localização envia uma resposta ao cliente.

Esta Seção revisou os fundamentos dos sistemas de localização numa rede sem fio. Inicialmente, foram apresentados os parâmetros de sinal ToA, TDoA, AoA e RSSI. Em seguida, os sistemas de localização baseados em RSSI foram classificados conforme a topologia e a cobertura da rede. Por último, foram definidas as técnicas de trilateração e *fingerprinting*. Portanto, esta Seção ofereceu as bases necessárias para o leitor acompanhar o restante do texto, onde o objetivo é aplicar algoritmos de aprendizado de máquina no contexto da localização de usuários em redes sem fio.

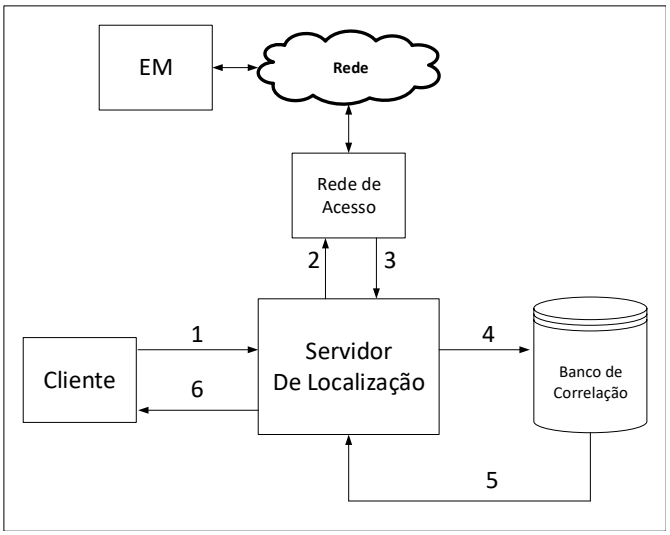


Figura 3.3 – Diagrama simplificado de um sistema de localização baseado em *fingerprinting* (Adaptado de [21]).

3.2 Conceitos básicos de aprendizado de máquina

Nos últimos anos, a utilização de dispositivos móveis trouxe mudanças significativas para a vida da população em geral, nas quais várias atividades passaram a ser realizadas por meio desses aparelhos. Nesse contexto, devido aos registros digitais dessas atividades, tivemos um aumento considerável na geração e armazenamento de dados, que combinado com o avanço da capacidade computacional dos dispositivos estão criando um ambiente propício para a aplicação de técnicas de aprendizado de máquina. Assim, várias aplicações que utilizam aprendizado de máquina passaram a fazer parte do nosso cotidiano, dentre as quais podemos citar buscadores de Internet, detectores de fraudes, algoritmos de reconhecimento facial etc. Entretanto, apesar dessas recentes aplicações, o aprendizado de máquina já está presentes em nossas vidas há décadas, em atividades especializadas, tais como jogos e reconhecimento óptico de caracteres (OCR, *optical character recognition*)[23]. Assim, temos uma definição dada por Arthur Samuel em 1959 [24], na qual aprendizado de máquina é apresentado como uma área de estudo que dá ao computador a habilidade de aprender uma atividade, mesmo sem ter sido explicitamente programado para tal.

De uma maneira mais formal, aprendizado de máquina pode ser definido como ”o estudo de algoritmos que melhoram a performance P de uma tarefa T com experiência E” [25]. Por exemplo, em um programa de computador que aprende a jogar damas, a tarefa T é jogar damas, a performance P é a porcentagem de jogos ganhos contra oponentes, e a experiência E é a experiência adquirida em jogar contra ele mesmo. Máquinas de aprendizado constroem modelos matemáticos baseados em dados de amostra para fazer previsões sem a necessidade de instruções específicas. Os componentes básicos para a aplicação de um algoritmo de aprendizado de máquina a um determinado problema são: i) o conjunto de dados, o qual possibilitará o treinamento do algoritmo; ii) a possibilidade de mapeamento aproximado por uma função hipótese, ou seja, temos que ter um padrão nos

dados; *iii*) não existir solução analítica para o problema em questão, ou seja, não ter como deduzir uma solução matemática exata [26].

Antes de entrarmos em mais detalhes sobre aprendizado de máquina, se faz necessário algumas definições básicas. O conjunto de dados de amostras utilizado para fazer a máquina aprender é chamado de conjunto de treinamento e ele tem um impacto bastante relevante no desempenho final da máquina de aprendizado. O conjunto de dados utilizado para verificar o desempenho do algoritmo é chamado de conjunto de testes. Alguns algoritmos utilizam ainda um conjunto de validação, o qual é empregado para alguns ajustes (*tuning*) antes da verificação do desempenho por meio do conjunto de teste.

Para todos os conjuntos de dados (treino, validação e teste), temos os atributos (*features*) que são utilizados pelo algoritmo para predição e treinamento. Supondo que o conjunto de dados se encontra em uma forma tabulada, os atributos seriam equivalentes às colunas da tabela. As linhas dessa tabela são chamadas de instâncias (*instances*) ou amostras (*samples*). Além dessas definições, temos ainda o rótulo (*target*), que é o valor que desejamos prever. Por exemplo, em um conjunto de dados relativos a clientes de um determinado setor, a renda mensal e a idade são considerados atributos, enquanto que cada cliente pode ser visto como uma instância. Nesse contexto, o *target* poderia ser um campo binário indicando se o cliente é um bom ou mal pagador.

No que diz respeito aos métodos de aprendizagem, as máquinas de aprendizado podem ser classificadas em três tipos principais, quais sejam, Aprendizado Supervisionado, Aprendizado Não-Supervisionado e Aprendizado por Reforço [23]. Além desses, temos ainda os algoritmos evolucionários, que são mais utilizados na resolução de problemas de otimização [27].

No Aprendizado Supervisionado, o algoritmo é treinado com dados históricos e, a partir desses, deseja-se prever dados futuros. A base de dados de treinamento contém dados de entrada (usados na predição) e a informação do valor a ser predito (rótulo ou *target*). Assim, o algoritmo busca achar uma ligação entre os dados de entrada e o valor de saída. Algoritmos de aprendizado supervisionado são usados, por exemplo, em reconhecimento de imagens [28], reconhecimento de fala [29] e predição de temperatura (clima) [30].

No Aprendizado Não Supervisionado, são usados somente dados de entrada no treinamento e não se tem a informação do valor a ser predito. Isto significa que não há um rótulo para os dados de treinamento. Nesses casos, o que se procura é um padrão existente nas instâncias. Algoritmos de aprendizagem não-supervisionada são usados, por exemplo, para clusterização, com o objetivo de encontrar as variáveis mais importantes para um problema, assim como para encontrar anomalias em uma base de dados.

No Aprendizado por Reforço, tenta-se achar a melhor forma de alcançar um objetivo e o algoritmo aprende com o passar das iterações. Um exemplo é fazer o algoritmo jogar um jogo sozinho. O algoritmo vai escolher ações que maximizem a pontuação final do jogo. Se a escolha das ações implicar em um resultado ruim em uma iteração, o algoritmo tentará outras ações na próxima iteração. Dessa forma, o algoritmo aprenderá as melhores ações

que precisa realizar para obter a melhor pontuação possível do jogo.

Por fim, nos algoritmos evolucionários, o objetivo é encontrar parâmetros em um espaço de busca de modo a minimizar ou maximizar uma função objetiva definida. Dentre os algoritmos evolucionários, destacamos os algoritmos genéticos *(GA, genetic algorithms)* [31] e otimização por enxame de partículas (*PSO, particle swarm optimization*) [32].

Quando se trata de aprendizado de máquina, temos um grande número de algoritmos que podem ser utilizados. Como exemplos de algoritmos de aprendizado supervisionado, podemos citar o *k*-NN, a máquina de vetor de suporte, o MLP (*multi-layer perceptron*) [33] e as árvores de decisão [34]. Para algoritmos de aprendizado não supervisionado temos, por exemplo, o algoritmo *k-means* [35], o SOM (*self-organized maps*) [36], o *autoencoders* [37] e o DBSCAN (*density-based spatial clustering of applications with noise*) [38]. Por fim, como exemplos de algoritmos de aprendizado por reforço, temos o *Q-learning* [39], o SARSA (*state-action-reward-state-action*) [40] e o *Deep Q-learning* [41]. A escolha de qual algoritmo utilizar dependerá do problema a ser solucionado. Embora muitos dos algoritmos citados possam ser usados para tentar solucionar o problema de localização [42, 43, 44, 45, 46], dois deles se destacam: os algoritmos *k*-NN e a máquina de vetor de suporte. Eles têm apresentado um excelente desempenho nas soluções *indoor* e *outdoor*. Assim, devido Ã forte utilização em recentes pesquisas, iremos detalhar o funcionamento de ambos.

3.2.1 *k*-NN, *k-nearest neighbors*

O algoritmo *k*-NN é um classificador que pertence a família dos algoritmos baseados em instâncias [47]. Nesse tipo de algoritmo, as instâncias de treinamento são armazenadas e a predição de uma nova instância é realizada usando as *k* instâncias mais próximas no conjunto de treinamento. Essa estratégia usa um abordagem diferente quando comparada com outros métodos, tais como as redes neurais, na qual se constrói uma função hipótese baseada nas amostras de treinamento. Assim, no algoritmo *k*-NN, a generalização só é realizada quando uma nova instância é predita.

O algoritmo *k*-NN pode ser utilizado tanto em problemas de classificação como de regressão e funciona da forma como segue [25]: Dada uma instância de teste $\mathbf{X}_i$, o primeiro passo é encontrar *k* instâncias mais próximas de $\mathbf{X}_i$, denominados de vizinhos de $\mathbf{X}_i$. Supondo que cada instância seja descrita por um vetor de atributos *m*-dimensional $\mathbf{X} = [X_{i1}, X_{i2}, \ldots, X_{im}]$, a distância entre duas instâncias $\mathbf{X}_i$ e $\mathbf{X}_j$, denotada por $d(\mathbf{X}_i, \mathbf{X}_j)$, é definida por

$$d(\mathbf{X}_i, \mathbf{X}_j) = \sqrt{\sum_{r=1}^{m} (X_{ir} - X_{jr})^2}\,. \tag{3.1}$$

Vale salientar que diferentes métricas podem ser utilizadas para calcular a distância entre as amostras. Na Equação (3.1), a distância Euclidiana é utilizada, a qual é uma das mais empregadas na aplicação do algoritmo *k*-NN [48]. Nos problemas de classificação, após o cálculo dos *k* vizinhos de $\mathbf{X}_i$ por meio de (3.1), a classe (tipo) atribuída ao rótulo será a mais comum entre os vizinhos. Já nos problemas de regressão, o valor predito para

$\mathbf{X}_i$ é dado pela média dos valores de seus k vizinhos, tal que

$$\hat{f}(\mathbf{X}_i) \leftarrow \frac{\sum_{i=1}^{k} f(\mathbf{X}_i)}{k} \tag{3.2}$$

em que $\mathbf{X}_i$ é uma instância de treino, enquanto $f(\mathbf{X}_i)$ é o rótulo para $\mathbf{X}_i$. Por último, vale ressaltar que o algoritmo k-NN não é uma boa escolha para grandes volumes de dados, uma vez que o custo computacional da busca de vizinhos pode ser alto. Por outro lado, uma grande vantagem do k-NN é que praticamente não há custo com o treinamento e o algoritmo é capaz de aprender problemas complexos por aproximação utilizando uma estratégia simples.

3.2.2 Máquina de vetor de suporte

A máquina de vetor de suporte (SVM, *support vector machine*) foi introduzido por Vapnik em 1995 [49]. A técnica SVM pode resolver tanto problemas linearmente separáveis como não-linearmente separáveis. No contexto da classificação, o objetivo é encontrar um hiperplano que separa o máximo possível os dados de classes distintas. A distância entre este hiperplano e a instância mais próxima de cada classe é chamada *margem*. A margem determina quão bem as classes podem ser separadas. Na Figura 3.4, é mostrado o hiperplano ótimo e suas margens. As instâncias que se encontram sobre as retas que delimitam as margens são chamadas de *vetores de suporte*.

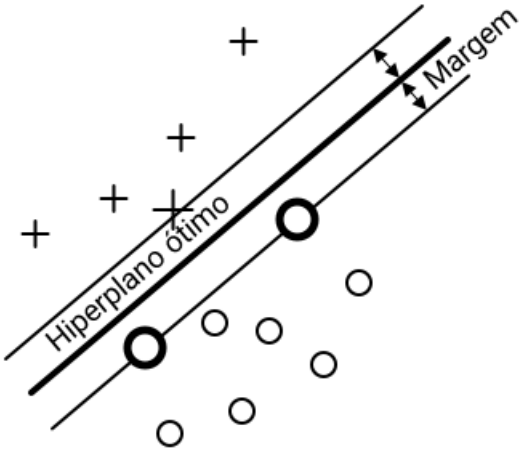


Figura 3.4 – O hiperplano ótimo é o que separa os dados de classes diferentes com a maior margem possível. Adaptado de [49].

Para lidar com problemas não linearmente separáveis, a SVM mapeia o conjunto de dados não linear em um espaço n-dimensional em que os dados podem ser linearmente separáveis. Esse mapeamento é feito por meio das funções de *kernel*. O espaço de mais alta dimensionalidade é chamado de espaço de características e nele é onde serão mapeados os dados do conjunto de entrada por meio de uma função Φ. Assim, um novo conjunto de treinamento, linearmente separável, é obtido.

Por último, vale ressaltar que a SVM foi aplicada inicialmente no contexto dos problemas de classificação. Entretanto, também pode ser utilizada em problemas de regressão. Nesse caso, recebe o nome de regressão por vetor de suporte (SVR, *support vector regression*) [49].

Assim supondo um problema de regressão, no qual o conjunto de treinamento (ou treino) seja $D = \{(\mathbf{x}_i, y_i) \in \mathbb{R}^n \times \mathbb{R}, i = 1, 2, \ldots, \ell\}$ com ℓ pares $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_\ell, y_\ell)$, em que $\mathbf{x}_i \in \mathbb{R}^n$ é um vetor n-dimensional que representa as entradas, $y_i \in \mathbb{R}$, uma variável real contínua que representa a saída e ℓ, o número de amostras no conjunto de treinamento. Em problemas de regressão, busca-se uma função que gera a saída y_i a partir das entradas $\mathbf{x}_i$.

Inicialmente, consideraremos o caso da regressão linear. Assim, temos $h(\mathbf{x}_i, \mathbf{w})$ como a função estimada entre a saída e a entrada, ou seja, é o hiperplano linear dado por

$$h(\mathbf{x}_i, \mathbf{w}) = \langle \mathbf{w}, \mathbf{x}_i \rangle + b \ , \tag{3.3}$$

em que $\mathbf{w} \in \mathbb{R}^n$ é o vetor normal ao hiperplano, $b \in \mathbb{R}$ é o *bias* (valor escalar), $\langle \cdot, \cdot \rangle$ é o operador de produto interno e $(\cdot)^T$ é o operador de transposição.

No caso da SVR, há diferentes algoritmos que podem ser utilizados, tais como ε-svr [49], ν-svr [50] e ε-bsvr [51]. Em geral, o algoritmo mais utilizado é *ε-svr*, o qual tem como objetivo encontrar uma função com um desvio mínimo de ε da saída y_i para todas as instâncias do conjunto de treino. Assim, esse será o algoritmo descrito neste trabalho. Por questões de simplicidade, o algoritmo ε-svr será denominado apenas de algoritmo svr deste ponto em diante.

Assim, faremos uso da função linear de perda de Vapnik com zona de insensitividade ε, a qual é definida como [49]

$$E(e_i) = |e_i|_\varepsilon = \begin{cases} 0 & , \, se \, |e_i| \leq \varepsilon \\ |e_i| - \varepsilon, & se \ não \end{cases} \ , \tag{3.4}$$

na qual $e_i = y_i - h(\mathbf{x}_i, \mathbf{w})$. A função linear de perda de Vapnik $E(e_i)$ pode ser vista na Figura. 3.5(a), onde a zona de insensitividade ε é destacada. Assim, a perda é nula (zero) se a diferença entre o valor real e o valor predito for menor que ε.

A solução do problema é encontrar uma função linear que estime os pares $(\mathbf{x}_i, y_i)$ com precisão ε. Ou seja, temo que encontrar um vetor $\mathbf{w}$ que minimize o erro, o que implica na resolução do problema de otimização dado por

$$\min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 \ , \tag{3.5}$$

restrito a $|e_i| \leq \varepsilon$ [52].

Para evitar grandes variações em $\mathbf{w}$, pode-se penalizar grandes resíduos. Com esse

objetivo, um termo de penalidade é incluindo em (3.5), tal que

$$\min \tag{3.6}$$

em que C é o parâmetro de custo que determina a relação entre o limite da soma dos erros maiores que ε e a variação permitida entre os coeficientes do vetor $\mathbf{w}$, também chamada de *flatness*. A função $E(e_i)$ define um tubo de espessura ε, conforme ilustrado na Figura 3.5(b), em que ε é o raio do tubo. A restrição $|e_i| \leq \varepsilon$, i.e., $y_i + \varepsilon \geq h(\mathbf{x}_i, \mathbf{w}) \geq y_i - \varepsilon$ é a condição para o valor predito estar dentro do tubo de espessura ε.

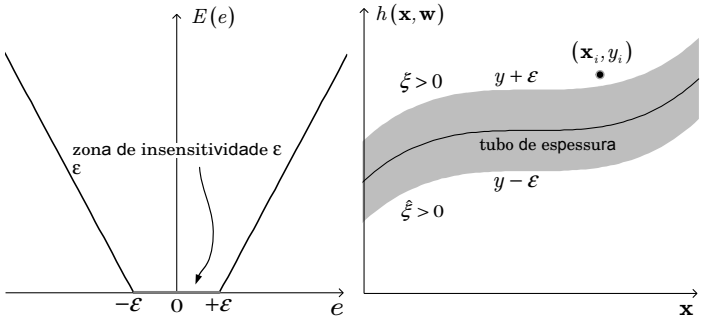


Figura 3.5 – (a) Função de perda linear de Vapnik com zona de insensitividade ε versus e. (b) Tubo de espessura ε definido a partir de $E(e)$.

O problema de otimização dado em (3.6) pode ser relaxado por meio da introdução de variáveis de folga (*slack variables*), denotadas por ξ e $\hat{\xi}$, as quais permitem lidar com pontos fora do tubo de espessura ε. Para os pontos acima do tubo, temos que $\xi > 0$ e $\hat{\xi} = 0$, enquanto para os pontos abaixo do tubo, $\xi = 0$ e $\hat{\xi} > 0$. Por último, quando os pontos estão dentro do tubo, faz-se $\xi = \hat{\xi} = 0$.

Dadas as variáveis ξ e $\hat{\xi}$, podemos reformular o problema de otimização como

$$\min_{\mathbf{w},b} \left[\frac{1}{2} \|\mathbf{w}\|^2 + C \left(\sum_{i=1}^{\ell} (\xi_i + \hat{\xi}_i) \right) \right] \tag{3.7}$$

com as restrições

$$\begin{cases} |e_i| = \varepsilon + \xi \\ |e_i| = \varepsilon + \hat{\xi} \\ \xi, \hat{\xi} \geq 0 \end{cases},$$

o qual pode ser resolvido usando multiplicadores de Lagrange, como pode ser visto em [52]. Depois de calculados os vetores dos multiplicadores de Lagrange $\boldsymbol{\alpha}$ e $\boldsymbol{\alpha}^*$, o melhor

hiperplano para regressão é dado por

$$h(\mathbf{x}_i, \mathbf{w}) = \sum_{j=1}^{\ell} (\boldsymbol{\alpha} - \boldsymbol{\alpha}^*) \left\langle \mathbf{x}_j, \mathbf{x}_i \right\rangle + b \ . \tag{3.8}$$

No caso da regressão não linear, a ideia básica é mapear os vetores de entrada $\mathbf{x}_i \in \mathbb{R}^n$ para vetores $\Phi(\mathbf{x}_i)$ de um espaço dimensional maior I, sendo Φ a representação do mapeamento. Depois da transformação, um problema não linear em $\mathbb{R}^n$ se torna um problema linear em I. Com isso, o problema de otimização pode ser reformulado como a maximização dos Lagrangianos utilizando a matriz de Hessian [53] e a solução é então dada por

$$h(\mathbf{x}_i, \mathbf{w}) = \sum_{j=1}^{\ell} (\boldsymbol{\alpha} - \boldsymbol{\alpha}^*) \left\langle \Phi(\mathbf{x}_j), \Phi(\mathbf{x}_i) \right\rangle + b \ . \tag{3.9}$$

Nota-se que o somatório não utiliza todos as amostras do conjunto de treino, mas só aquelas que têm Lagrangiano diferente de zero. Essas instâncias são chamadas de *vetores de suporte*.

O problema de otimização, representado por (3.9), envolve o cálculo de produtos internos entre vetores no novo espaço dimensional I. Portanto, se I for um espaço de alta dimensão, o cálculo de Φ pode se tornar inviável. Logo, a solução é recorrer ao truque do *kernel* para fazer a regressão sem a necessidade de calcular o mapeamento Φ de todos os vetores de entrada $\mathbf{x}_i$ para o espaço I [54]. O *kernel* é uma função que se aplica a dois vetores $\mathbf{x}_i$ e $\mathbf{x}_j$ no espaço de entrada X e retorna o produto interno desses vetores no espaço I [55], i.e.,

$$K(\mathbf{x}_i, \mathbf{x}_j) = \left\langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \right\rangle . \tag{3.10}$$

Para garantir a convexidade do problema de otimização dado por (3.9) e assegurar que é possível calcular o produto interno $\left\langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \right\rangle$, só podem ser usadas as funções de *kernel* que satisfaçam a condição de Mercer [49]. Dentre os *kernels* utilizados para regressão, destacam-se os de rbf e o polinomial [56]. Neste trabalho, foram testados os *kernels* polinomial, rbf Laplaciano e rbf Gaussiano. As expressões relacionadas a cada *kernel* estão indicadas na Tabela 3.1.

Tabela 3.1 – Tipos de *kernels* considerados no algoritmo SVR.

Kernel	Expressão	Parâmetros
Polinomial	$K(\mathbf{x}_i, \mathbf{x}_j) = \left(\beta \left\langle \mathbf{x}_i, \mathbf{x}_j \right\rangle + c\right)^z$	β, c, z
Gaussiano	$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2}{2\sigma^2}\right)$	σ
Laplaciano	$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\left\|\mathbf{x}_i - \mathbf{x}_j\right\|}{\sigma}\right)$	σ

Para o *kernel* polinomial, temos os parâmetros z, que é grau do polinômio, β, que define a escala, e c que especifica o deslocamento (*offset*) [48]. Adicionalmente, para os

kernels do tipo rbf, temos o parâmetro σ, o qual pode ser usado para o controle da escala [48].

3.3 Aprendizado de máquina aplicado a técnicas de localização

Definidas as técnicas básicas de localização e alguns algoritmos de aprendizado de máquina supervisionados, iremos abordar mais detalhadamente a implementação de um método de localização baseado em *fingerprinting* usando SVR. Para efeitos de comparação, a mesma técnica *fingerprinting* foi implementada com os modelos clássicos de propagação COST-231 e ECC-33. Uma implementação prática de trilateração utilizando aprendizado de máquina pode ser encontrada em [57], no entanto não será objeto de estudo deste capítulo.

Devido à ausência de linha de visada (NLoS, *non-line-of-sight*) no enlace ERB-EM e à propagação multipercurso, a técnica *fingerprinting* costuma apresentar resultados não tão eficientes. Com o objetivo de obter uma melhor acurácia, uma abordagem interessante é modelar o problema de geolocalização de terminais na rede celular como um problema de aprendizado de máquina. Por exemplo, em [58] é proposto um algoritmo baseado em SVR que usa o ToA da onda em seis ERBs para estimar a posição do terminal. Nesta Seção, propomos um método baseado em algoritmos SVR que usam os RSSIs medidos em três ERBs para estimar a posição da EM.

No método de localização proposto, foi aplicada uma técnica de predição de sinais de RF com SVR descrita em [59]. Uma vez que o *kernel* Laplaciano teve o melhor desempenho na predição da perda de espaço livre, ele foi adotado para os algoritmos SVR utilizados no método de localização. A técnica de localização proposta tem como objetivo estimar a posição (latitude e longitude) da EM com os RSSIs medidos a partir de três ERBs. A técnica proposta pode ser descrita em seis passos, os quais são listados no Algoritmo 1.

Algorithm 1 Detalhamento do método de localização proposto

1. Coletar as medições de RSSI do *scanner* de RF.
2. Treinar os algoritmos SVR para predição da perda de espaço livre (um para cada ERB).
3. Gerar os mapas de cobertura (um para cada ERB).
4. Coletar medições de RSSI da EM procurada nas três ERBs.
5. Aplicar o filtro de redução do espaço de busca.
6. Encontrar o ponto mais próximo no mapa de cobertura.

No primeiro passo, foram consideradas medições de nível de sinal com portadora na faixa de frequência de $1,8$ GHz. As medições foram realizadas em um ambiente urbano na cidade de Recife-PE, considerando uma rede GSM (*Global System for Mobile Communications*). O nível de intensidade de sinal foi medido com o equipamento

NEMO FS R1[1] utilizado como um *scanner* para a rede GSM. No total, foram realizadas 2.547 medições e coletados os mesmos dados especificados em [59]. Dessas medições, 2.447 foram usadas para treinar e validar o algoritmo SVR e 100 serviram como conjunto de teste para verificar o desempenho do método de localização. Na Figura 3.6, temos as medições que foram usadas para testar a precisão do algoritmo SVR (em vermelho) e as medições que foram usadas para treiná-lo e validá-lo (em verde). As localizações das três ERBs, denotadas por ERB-1, ERB-2 e ERB-3, também são indicadas na Figura 3.6. Complementando as informações sobre as ERBs, a altura e a elevação do terreno para as três ERBs são apresentadas na Tabela 3.2. A EIRP e o ângulo de abertura horizontal da antena para todas as ERBs são, respectivamente, 70, 1 dBm e 63°.

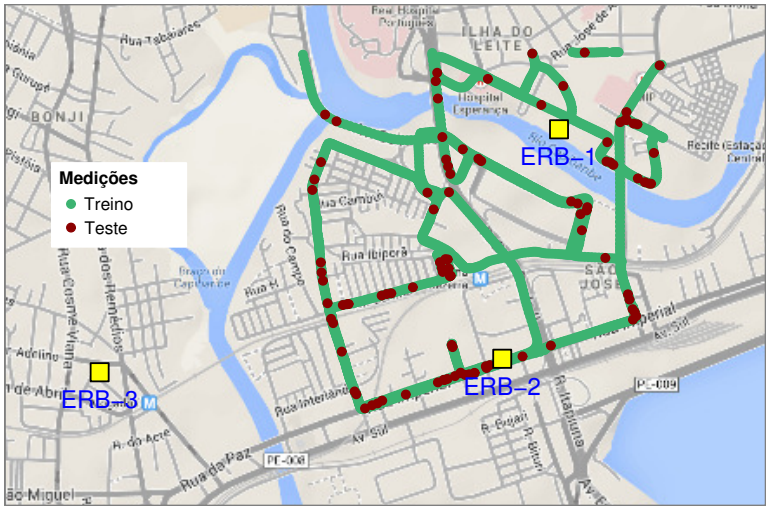


Figura 3.6 – Ambiente urbano na cidade de Recife-PE com a indicação das medições de teste, de treino e a localização das ERBs.

Tabela 3.2 – Dados relativos à configuração das ERBs no momento das medições.

Estação Rádio Base	Elevação	Altura
ERB-1	8 m	41 m
ERB-2	6 m	53 m
ERB-3	8 m	40 m

Após a coleta dos dados, o segundo passo do método consiste em treinar os algoritmos SVR-Laplaciano para a predição da PL. Para tanto, a partir dos dados coletados foram extraídos os atributos especificados em [59], e foram gerados três conjuntos de treino, um para cada ERB. Os conjuntos possuem os mesmos atributos (*features*). No processo de treinamento dos algoritmos SVR, foi utilizada a técnica 10-*fold cross-validation*, já descrita anteriormente, para a escolha da melhor configuração de parâmetros (*best fit*). Os valores dos parâmetros C e ε testados foram os mesmos definidos na predição de sinais com a SVR,

[1]NEMO FS R1 é um receptor modular de escaneamento digital de intensidade de sinais de RF.

ou seja, foram testados 18 valores (potências de 2) de C no intervalo de 2^{-2} a 2^{15}, para cada valor de ε ($\varepsilon = 0,1$ e $\varepsilon = 0,05$). Os melhores valores para C e ε estão listados na Tabela 3.3, bem como os respectivos $\bar{\mu}$ (RMSE) e μ_σ (desvio padrão). O valor do parâmetro σ de cada algoritmo SVR-Laplaciano, foi calculado como o ponto médio entre o 10° e o 90° percentis de $\left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2$ [54, 60]. Os valores de σ estimados para os algoritmos das ERB-1, ERB-2 e ERB-3 também estão listados na Tabela 3.3. Note que temos um valor de σ para cada algoritmo, uma vez que cada um deles tem seu próprio conjunto de treinamento. Vale ressaltar, que o *kernel* Laplaciano foi selecionado devido ao seu melhor desempenho nesse tipo de problema, como pode ser em [61].

Tabela 3.3 – Resultados da fase de treinamento de cada algoritmo SVR-Laplaciano usando técnica 10-*fold cross-validation*.

ERB	σ	C	ε	$\bar{\mu}(dB)$	$\mu_\sigma(dB)$
ERB-1	0,258	16	0,1	3,66	0,271
ERB-2	0,207	32	0,1	3,83	0,240
ERB-3	0,215	16	0,1	3,49	0,259

O próximo passo do algoritmo SVR é gerar o mapa de cobertura. Para isso, uma área de localização deve ser definida. Neste trabalho, será considerada uma área de localização de 1,38 km × 1,38 km com um grid de resolução de 20 m × 20 m. A área de localização e o grid podem ser vistos na Figura 3.7.

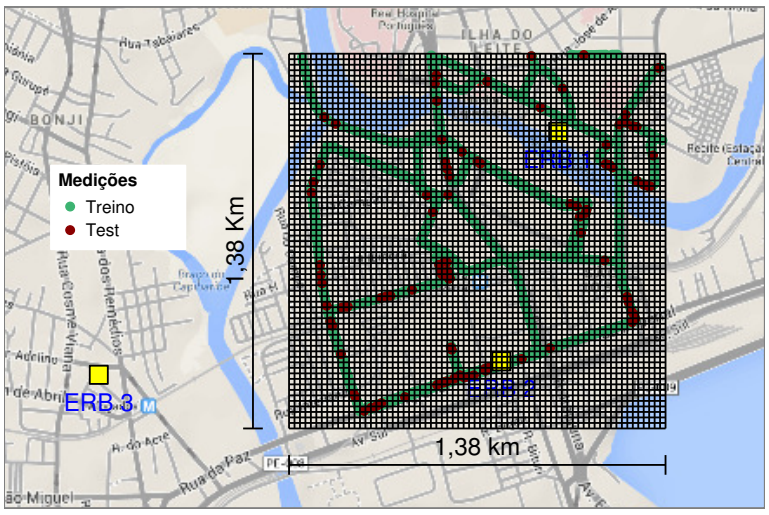


Figura 3.7 – área de localização de 1,38 km × 1,38 km com um grid de resolução de 20 m × 20 m.

Estabelecida a área de localização como um grid de q posições, o mapa de cobertura é definido como um conjunto $S = \{(\mathbf{p}_i, \mathbf{s}_i) \in \mathbb{R}^2 \times \mathbb{R}^3, i = 1, 2, \ldots, q\}$ com q pares, em que $\mathbf{p}_i = [p_i^{(1)}, p_i^{(2)}]$ é o vetor posição para o i-ésimo quadrado de área 20 m^2 no grid, sendo $p_i^{(1)}$ a longitude e $p_i^{(2)}$, a latitude do centro deste i-ésimo quadrado. Além disso, o vetor $\mathbf{s}_i = [s_i^{(1)}, s_i^{(2)}, s_i^{(3)}]$ representa as predições de RSSI feitas pelo algoritmo SVR para a ERB-1, ERB-2 e ERB-3, respectivamente. Assim, o algoritmo SVR treinado de cada ERB é utilizado

para fazer a predição de RSSI para todas as posições do vetor $\mathbf{p}_i$ do grid de localização. Na Figura 3.8, podemos ver o mapa de cobertura para cada ERB, os quais foram gerados a partir das predições dos algoritmos SVR. Na Figura 3.8, as cores indicam o nível de RSSI em cada posição do mapa.

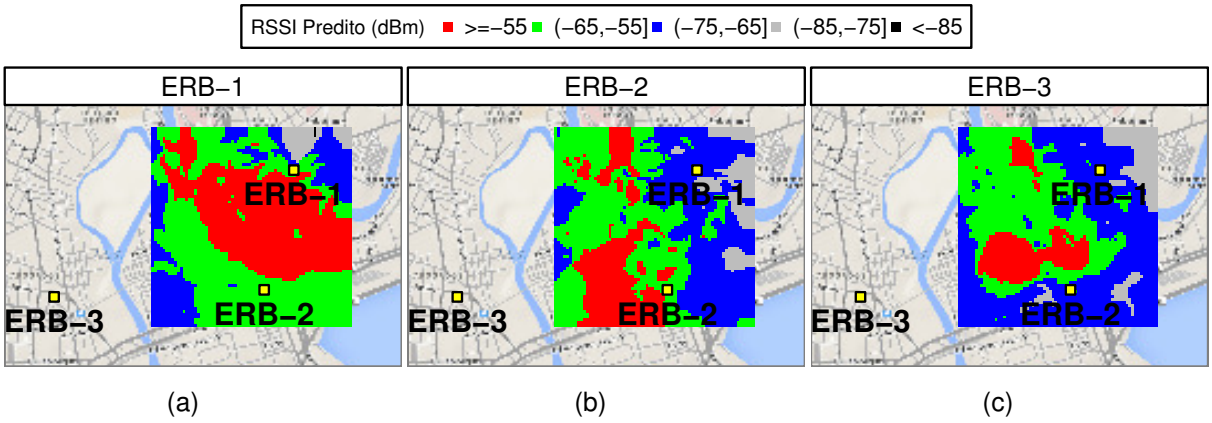


Figura 3.8 – Mapas de cobertura obtidos a partir das predições de RSSI dos algoritmos SVR para cada ERB:(a) ERB-1. (b) ERB-2. (c) ERB-3.

Dado o mapa de cobertura S, os dois últimos passos do método de localização concentram-se em estimar a posição da EM no grid de localização. Considerando $\mathbf{m} = [m_1, m_2, m_3]$, um vetor que representa o RSSI medido para terminal móvel procurado, em que m_1, m_2 e m_3 são as medições para as ERB-1, ERB-2 e ERB-3, respectivamente. Ademais, considerando $\mathbf{a} = [a_1, a_2, a_3]$ como sendo os TAs (Time Advanced) medidos em relação ao móvel procurado. Assim, depois de realizar as medições em relação ao móvel procurando, o próximo passo é a aplicação do filtro de área. O filtro tenta selecionar pontos que tenham os mesmos TAs medidos do móvel procurando (vetor $\mathbf{a}$). Primeiramente ele tenta selecionar os pontos que têm todos os TAs iguais aos TAs medidos (total correspondência), caso não haja pontos com esses valores, ele tenta selecionar pontos que têm pelo menos dois dos TAs medidos. Se ainda não há pontos com essas características, ele tenta selecionar pontos com ao menos um dos TA medidos. Por último, caso nenhum ponto tenha sido selecionado, é retornado todo o grid de localização como área de busca S_R.

Dado a área reduzida de busca (S_R), o último passo é estimar o a posição utilizado a distância Euclidiana como função de similaridade. Assim, definindo $d_i^{'}$ como sendo a distância Euclidiana entre $\mathbf{m}$ e $\mathbf{s}_i$ para i-ésima posição no grid de localização, $d_i^{'}$ pode ser expresso por

$$d_i' = \sqrt{(s_i^{(1)} - m_1)^2 + (s_i^{(2)} - m_2)^2 + (s_i^{(3)} - m_3)^2}\,, \tag{3.11}$$

$i = 1, 2, \ldots, q$. Finalmente, a melhor posição estimada $\mathbf{p}_i$ é aquela cujo vetor $\mathbf{s}_i$ tem a menor distância Euclidiana d_i' em relação ao vetor $\mathbf{m}$.

Para efeitos de comparação, foram considerados dois métodos de localização *Fingerprint* que utilizam métodos clássicos para predição de sinais. Por razões de simplicidade, adotaremos a denominação FP-COST-231 para o método de localização que utiliza a técnica de *Fingerprint* com o modelo COST-231, e FP-ECC-33 para o método de localização que utiliza a técnica de *Fingerprint* com o modelo ECC-33. Em ambos os

métodos, os modelos de propagação foram utilizados para estimar a perda em espaço livre e com isso possibilitar a construção do mapa de cobertura.

Os desempenhos dos três métodos de localização (FP-COST-231, FP-ECC-33 e o método proposto) são avaliados utilizando simulação em computadores. Todos os métodos foram implementada por meio da linguagem R [62] em conjunto com os pacotes *kernlab* [54] e *caret* [63].

Para comparamos o desempenho dos métodos, utilizaremos o erro de localização, η, que é a distância (em m) entre a posição real da EM e a posição estimada pelo método de localização. Na Tabela 3.4, temos uma análise estatística do erro de localização das predições de cada método. O erro médio de localização é representado por $\bar{\eta}$, seu desvio padrão por η_σ e os erros máximo e mínimo por η_{max} e η_{min}, respectivamente. Ainda na Tabela 3.4, podemos verificar que o método de localização proposto apresenta um erro médio de $\bar{\eta} = 54.4$ m, enquanto os métodos FP-COST-231 e FP-ECC-33 apresentam $\bar{\eta} = 211,9$ m e $\bar{\eta} = 226,3$ m, respectivamente.

Para uma melhor comparação, mapas de predição de localização podem ser construídos, como ilustrado na Figura 3.9. Todos os mapas possuem duas camadas: a primeira com os pontos de teste coletados em campo; e a segunda, com os pontos preditos pelo método de localização. Em todos os mapas, os pontos em cor cinza representam as posições relativas às medições de campo, ou seja, são as posições reais da EM.

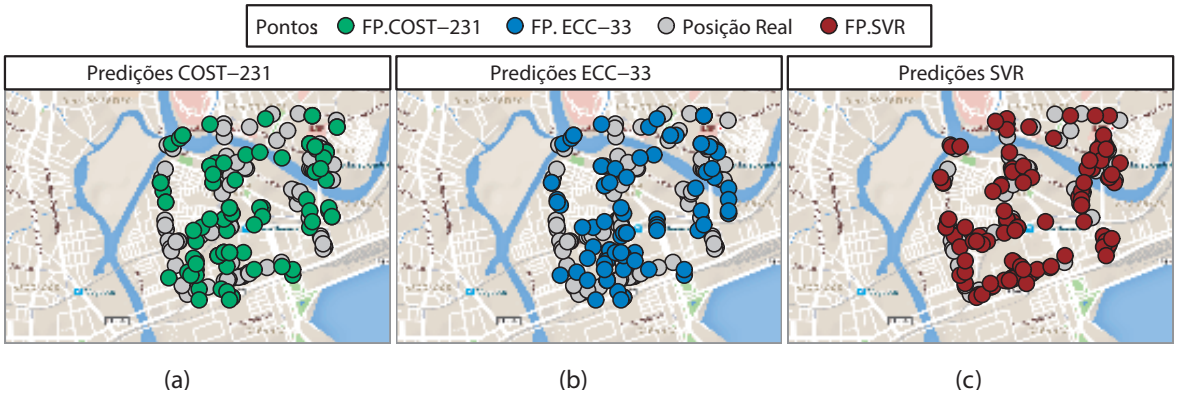


Figura 3.9 – Mapas de predição para cada método de localização:(a) *Fingerprinting* com COST-231. (b) *Fingerprinting* com ECC-33. (c) Método proposto (algoritmo SVR).

A Fig. 3.9(a) mostra a distribuição de pontos predita pelo método FP-COST-231, a qual é representada pelos pontos verdes. Na Fig. 3.9(b), temos as posições estimadas pelo método FP-ECC-33, as quais são representadas pelos pontos azuis. A Fig. 3.9(c) indica as

Tabela 3.4 – Análise estatística do erro médio da posição para os três métodos de localização.

Método Loc.	$\bar{\eta}$	η_σ	η_{max}	η_{min}
SVR	54,0 m	56,8 m	383,3 m	0,4 m
FP-COST-231	211,9 m	142,1 m	807,9 m	16,2 m
FP- ECC-33	226,3 m	150,1 m	807,9 m	16,2 m

posições estimadas obtidas pelo método de localização proposto (abordagem baseada em algoritmo SVR). Comparando as figuras, podemos verificar que, quando a SVR é utilizada, há uma maior convergência entres os pontos estimados e os pontos cinzas. Dessa forma, o método baseado em SVR se mostra mais preciso que os métodos que utilizam o COST-231 e EC-33.

Outra maneira de se comparar os métodos de localização implementados neste trabalho é por meio de histogramas. Na Figura 3.10, temos um histograma para cada método de localização, no qual o eixo das abscissas representa o erro de localização η e o eixo das ordenadas, a quantidade de amostras que tiveram o mesmo η. Analisando os três histogramas, é possível verificar que o método de localização proposto é o melhor deles, pois as amostras estão concentradas no início do histograma, isto é, os erros de localização estão contidos no intervalo de 0 a 250 m. Para os outros dois métodos, podemos verificar que os erros estão mais distribuídos no intervalo de 0 a 400 m.

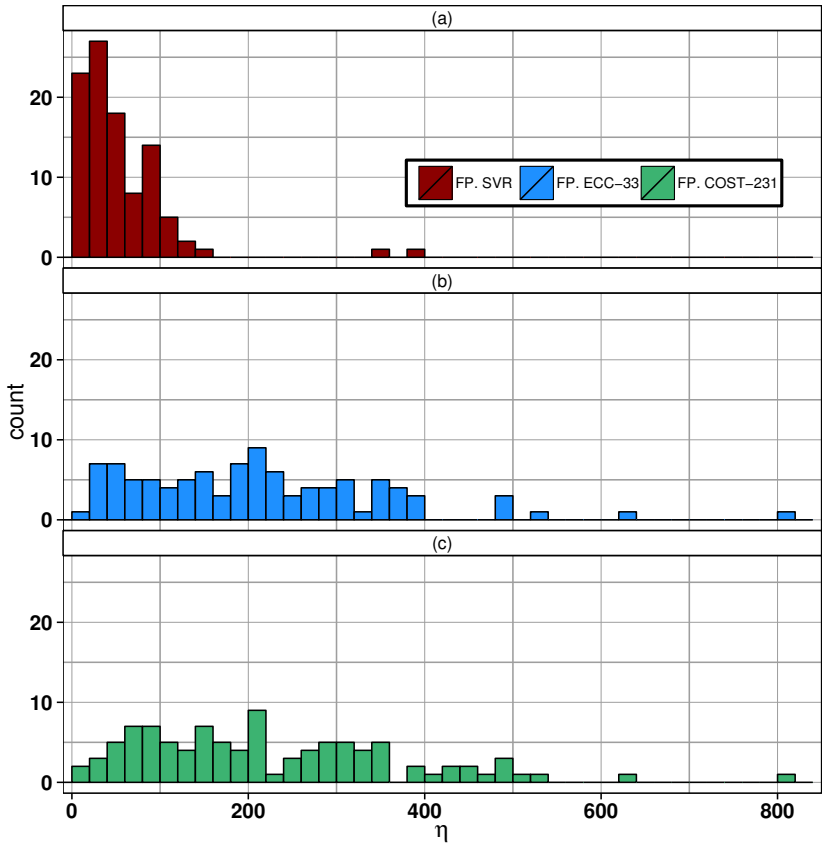


Figura 3.10 – Histograma do erro médio de localização (em m) para o conjunto de 100 amostras de teste: (a) Abordagem baseada em algoritmos SVR. (b) *Fingerprinting* com ECC-33. (c) *Fingerprinting* com COST-231.

3.4 Oportunidades e desafios

Devido a sua versatilidade, são inúmeras as possibilidades para aplicação de técnicas de aprendizado de máquina no que concerne a localização de terminais móveis em

ambientes sem fio. O espectro de desafios contempla desde avanços na engenharia de características (*feature engineering*), considerando diferentes medições que podem incluir o ToA, TDoA, RSSI, AoA, informação de estado do canal (CSI, *channel state information*), além da utilização de novas tecnologias como WLAN, UWB (*ultra-wide bandwidth*) e BLE (*Bluetooth Low Energy*).

Este abrangente leque de opções influencia diretamente na escolha e nas técnicas de refinamento que podem ser aplicadas aos diversos algoritmos existentes. Esses fatores são determinantes e decisivos para obter um bom equilíbrio entre a acurácia obtida pelo serviço de localização e seu desempenho, esse último normalmente avaliado sob a perspectiva da complexidade computacional.

Com base nas oportunidades que surgem diante dessa nova fronteira, pretendemos nessa Seção destacar algumas vertentes de pesquisa cujos resultados destacam-se por apresentar abordagens alternativas em relação àquelas discutidas até agora.

A simplicidade e a efetividade do k-NN no contexto das técnicas de localização trouxeram notoriedade para esse algoritmo, incentivando diversas pesquisas focadas em propor novas melhorias ou adaptações ao k-NN com a finalidade de aperfeiçoar sua eficácia, melhorando assim a acurácia dos sistemas de localização [64]. Em [65], os autores propuseram uma variante do k-NN batizada de FS/k-NN (*feature scaling*/k-NN). Nas aplicações tradicionais do k-NN, o valor absoluto obtido ao subtrair dois RSSIs distintos não carrega consigo informação a respeito do RSSI utilizado nas parcelas da subtração. Devido a essa característica, dois valores absolutos idênticos, obtidos da subtração de RSSIs distintos, não representam necessariamente a mesma distância. Baseando-se nessa observação, os autores criaram um mecanismo que atribui pesos aos componentes do k-NN, levando em consideração os valores de RSSI.

Outro algoritmo que se destaca devido a sua efetividade é o SVM. Assim como ocorre para o k-NN, existem pesquisas que propõem melhorias promovidas pela realização de adaptações no SVM. O trabalho em [66] mostra o emprego de uma variante do SVM conhecida como OISVM (*online independent* SVM). A grande vantagem dessa variante é a capacidade da etapa de aprendizado ser realizada de forma *online*, ou seja, o algoritmo continua refinando seu processo de aprendizado a medida que novos dados são fornecidos. Essa característica do OISVM implicou na redução da complexidade computacional associada ìs etapas de treino e predição do algoritmo, além de melhor sua acurácia.

Seguindo na linha de variantes do SVM, o RVM (*relevance vector machine*) vem sendo utilizado com sucesso para diferenciar sinais com e sem linha de visada, respectivamente. A grande vantagem do RVM em comparação ao SVM é a quantidade de vetores utilizados. O fato do RVM utilizar menos vetores permite um desempenho superior ao SVM em termos de complexidade computacional. Os autores em [67] desenvolveram um método para classificar sinais com e sem linha de visada, utilizando ToA com a tecnologia UWB.

Outras oportunidades de melhoria na acurácia e na redução de complexidade

computacional dos serviços de localização tem sido discutidas recentemente. Em [68], os autores mostram que a escolha da geometria do *grid* na criação dos mapas de rádio pode influenciar na acurácia e causar uma diminuição substancial na complexidade computacional. Esse resultado potencializa eventuais melhorias nos serviços de localização ao incluir a geometria do espaço como ponto de chave, potencializando os ganhos já obtidos através do emprego de algoritmos de aprendizado de máquina.

Para finalizar, é importante destacar que a utilização de algoritmos de aprendizado de máquina para a resolução de problemas cada vez mais complexos, onde uma grande quantidade de dados está envolvida, já é uma tendência. Várias plataformas de **Big Data** podem ser utilizadas para permitir a escalabilidade de algoritmos de aprendizado de máquina. Dentre as principais, destacam-se a Apache Spark [69] e H20 [70]. Essas plataformas permitem a utilização de redes de aprendizado profundo (*deep learning*) para a resolução de problemas que possuem uma imensa quantidade de dados [71]. No contexto de localização e por meio dessas ferramentas, é possível construir um modelo preditivo usando dados recolhidos de toda uma rede celular por exemplo, técnicas de *crowdsourcing* [72] podem ser utilizadas para a geração da base de dados. Esse tipo de estratégia permitiria a geração de um modelo preditivo com maior poder de generalização e que poderia ser utilizado em qualquer ponto da rede celular.

Como pôde ser visto, as oportunidades e desafios na área de localização de terminais móveis em ambientes sem fio são inúmeras. A aplicabilidade de técnicas de aprendizado de máquina tem evoluído constantemente. Sendo assim, a pretensão desta Seção não é consolidar uma lista definitiva, mas sim motivar e inspirar o leitor acerca da pluralidade de uma área relativamente recente, onde os problemas, desafios e consequentes soluções ainda estão para serem descobertos e explorados em sua máxima plenitude.

Referências Bibliográficas

[1] P. Kaveh, K. Prashant, and G. Yishuang. Localization challenges for the emergence of the smart world. *IEEE Access*, 3:3058–3067, 2015.

[2] K. Muzaffer and S. Memduh. Ranging for in-body localization of ultra wide band wireless endoscopy capsules using neural networks. In *24th Signal Processing and Communication Application Conference (SIU)*, Quebec, Canada, October, 2016.

[3] S. Timo and S Heinerm. On-body localization of wearable devices: An investigation of position-aware activity recognition. In *IEEE International Conference on Pervasive Computing and Communications (PerCom)*, Sidney, Australia, March, 2016.

[4] J. S. Almeida et al. Localization system for autonomous mobile robots using machine learning methods and omnidirectional sonar. *IEEE Latin America Transactions*, 16(2):368–374, 2018.

[5] Y. Jun, Z. Lin, T. Jian, C. Yuwei, C. Ruizhi, and Liang. C. Hybrid kernel based machine learning using received signal strength measurements for indoor localization. *IEEE Transactions on Vehicular Technology*, 67(3):2824–2829, 2018.

[6] P. Bile, S. Gonzalo, S. Erik, F. Markus, and W. Henk. Decentralized scheduling for cooperative localization with deep reinforcement learning. *IEEE Transactions on Vehicular Technology*, 68(5):4295–4305, 2019.

[7] A. T. Duc and N. Thinh. Localization in wireless sensor networks based on support vector machines. *IEEE Transactions on Parallel and Distributed Systems*, 19(7):981–994, 2008.

[8] X. Han and Z. He. A wireless fingerprint location method based on target tracking. In *12th International Symposium on Antennas, Propagation and EM Theory (ISAPE)*, Hangzhou, China, December. 2018.

[9] K. Baik, S. Lee, and B. Jang. Hybrid RSSI-AoA positioning system with single time-modulated array receiver for LoRa IoT. In *48th European Microwave Conference (EuMC)*, Madrid, Spain, September. 2018.

[10] S. Wielandt and L. Strycker. Indoor multipath assisted angle of arrival localization. *Sensors*, 17(11):2522–2527, 2017.

[11] T. S. Rappaport. *Wireless communications: principles and practice*. Prentice-Hall PTR, Chicago, USA., second edition, 2009.

[12] M. Vossiek, L. Wiebking, P. Gulden, J. Wieghardt, C. Hoffmann, and P. Heide. Wireless local positioning. *IEEE Microwave Magazine*, 4(4):77–86, December. 2003.

[13] J. Figueiras and S. Frattasi. *Mobile positioning and tracking: from conventional to coopetive techniques*. John Wiley & Sons Ltd, Torquay, UK, 2010.

[14] M. Antonini, M. Ruggieri, R. Prasad, U. Guida, and G. F. Corini. Vehicular remote tolling services using egnos. In *PLANS 2004. Position Location and Navigation Symposium (IEEE Cat. No.04CH37556)*, pages 375–379, April 2004.

[15] J. Conesa, A. Pérez-Navarro, J. Torres-Sospedra, and R. Montoliu. *Geographical and Fingerprinting Data to Create Systems for Indoor Positioning and Indoor/Outdoor Navigation*. Chicago, USA, September 2018.

[16] S. Agarwal and S. De. Rural broadband access via clustered collaborative communication. *IEEE/ACM Transactions on Networking*, 26(5):2160–2173, October. 2018.

[17] R. Reghelin. Um algoritmo descentralizado de localização para rede de sensores sem fio usando calibragem cooperativa e heurísticas. Master's dissertation, UFSC/IME, Florianópolis-SC, 2007.

[18] E. Cassano, F. Florio, F. De Rango, and S. Marano. A performance comparison between ROC-RSSI and trilateration localization techniques for WPAN sensor networks in a real outdoor testbed. In *Wireless Telecommunications Symposium (WTS)*, Univ. of Calabria, Dipt. di Elettron, Rende, Italia, 2009.

[19] D. Varagnolo, F. Zanella, A. Cenedese, G. Pillonetto, and L. Schenato. Netwon-Raphson consensus for distributed convex optimization. *IEEE Transactions on Automatic Control*, 9286(c):1–16, 2015.

[20] F. Gao and L. Han. Implementing the Nelder-Mead simplex algorithm with adaptive parameters. *Computational Optimization and Applications*, 51(1):259–277, 2010.

[21] R. S. Campos and L. Lovisolo. *RF Fingerprinting Location Techniques*, chapter 15, pages 487–520. John Wiley & Sons, Ltd, Chicago, EUA, 2011.

[22] Z. Zhong and T. He. Wireless sensor node localization by multisequence processing. *Transactions on Embedded Computing Systems*, 11(1), 2012.

[23] A. Géron. *Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems*. O'Reilly Media, Inc., California, EUA, 2017.

[24] A. L. Samuel. Some studies in machine learning using the game of checkers. In *Computer Games I*, pages 366–400. Springer, 1988.

[25] M. M. Tom. *Machine Learning*. McGraw-Hill, California, USA, 1997.

[26] Y. S. Abu-Mostafa, M. Magdon-Ismail, and H. Lin. *Learning from data*, volume 4. AMLBook, New York, USA, 2012.

[27] M. J. A. Berry and G. S Linoff. *Data mining techniques: for marketing, sales, and customer relationship management*. John Wiley & Sons, California, USA, 2004.

[28] F. Höppner, F. Klawonn, R. Kruse, and T. Runkler. *Fuzzy cluster analysis: methods for classification, data analysis and image recognition*. John Wiley & Sons, California, USA, 1999.

[29] D. Yu and L. Deng. *Automatic Speech Recognition*. Springer, Huang, China, 2016.

[30] F. Katopodes Chow, S. F. J. De Wekker, and B. J. Snyder. *Mountain weather research and forecasting: recent progress and current challenges*. Springer, Huang, China, 2013.

[31] D. E. Goldberg and J. H. Holland. Genetic algorithms and machine learning. *Machine learning*, 3(2):95–99, 1988.

[32] R. Eberhart and J. Kennedy. Particle swarm optimization. In *Proceedings of the IEEE international conference on neural networks*, volume 4, pages 1942–1948. Citeseer, 1995.

[33] S. S. Haykin et al. *Neural networks and learning machines*. Prentice Hall„ New York, USA, 2009.

[34] S. R. Safavian and D. Landgrebe. A survey of decision tree classifier methodology. *IEEE Transactions on Systems, Man, and Cybernetics*, 21(3):660–674, 1991.

[35] E. W. Forgy. Cluster analysis of multivariate data: efficiency versus interpretability of classifications. *Biometrics*, 21:768–769, 1965.

[36] T. Kohonen. Self-organized formation of topologically correct feature maps. *Biological cybernetics*, 43(1):59–69, 1982.

[37] V. N. Balasubramanian. Tutorial. In *22nd Annual International Conference on Advanced Computing and Communication (ADCOM)*, pages 23–28, September. 2016.

[38] M. Ester, H. Kriegel, , et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD*, volume 96, pages 226–231, 1996.

[39] C. Watkins and P. Dayan. Q-learning. *Machine learning*, 8(3-4):279–292, 1992.

[40] G. A. Rummery and M. Niranjan. *On-line Q-learning using connectionist systems*, volume 37. University of Cambridge, Dep. of Engineering Cambridge, England, 1994.

[41] V. Mnih, Kavukcuoglu, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.

[42] U. Ahmad, A. Gavrilov, S. Lee, and Y. Lee. Modular multilayer perceptron for wlan based localization. In *IEEE International Joint Conference on Neural Network Proceedings (IJCNN)*, pages 3465–3471, Paris, France, 2006. IEEE.

[43] G. Giorgetti, S. K. S. Gupta, and G. Manes. Wireless localization using self-organizing maps. In *Proceedings of the 6th International Conference on Information Processing in Sensor Networks*, pages 293–302. ACM, 2007.

[44] M. W. Kadous. Prediction of indoor location using decision trees, January 2014. US Patent: 8.639.640.

[45] A. Razavi, M. Valkama, and E. Lohan. K-means fingerprint clustering for low-complexity floor estimation in indoor mobile localization. In *IEEE Globecom Workshops (GCWkshps)*, pages 1–7, Paris, France, 2015. IEEE.

[46] K. K. Almuzaini and T. A. Gulliver. Range-based localization in wireless networks using the dbscan clustering algorithm. In *IEEE 73rd Vehicular Technology Conference (VTC Spring)*, pages 1–7, Paris, France, 2011. IEEE.

[47] D. W. Aha, D. Kibler, and M. K. Albert. Instance-based learning algorithms. *Machine Learning*, 6(1):37–66, 1991.

[48] M. Kuhn and K. Johnson. *Applied Predictive Modeling*, volume 26. Springer, California, USA, 2013.

[49] V. N. Vapnik. Constructing learning algorithms. In *The Nature of Statistical Learning Theory*, pages 119–166. Springer, California, USA, 1995.

[50] A. J. Smola. *Regression estimation with support vector learning machines*. Master's dissertation, Technische Universit at Munchen, 1996.

[51] J. B. Gao, S. R. Gunn, C. J. Harris, and M. Brown. A probabilistic Framework for SVM Regression and Error Bar Estimation. *Machine Learning*, 46(1-3):71–89, 2002.

[52] N. Lange, C. M. Bishop, and B. D. Ripley. *Neural Networks for Pattern Recognition.*, volume 92. California, USA, December 1997.

[53] V. Kecman. Support vector machines: An introduction. *Support Vector Machines: Theory and Applications*, 47:1–47, 2005.

[54] A. Karatzoglou, A. Smola, K. Hornik, and A. Zeileis. Kernlab-an S4 package for kernel methods in R. *Journal of Statistical Software*, 11(9):1–19, 2004.

[55] R. Herbrich. *Learning Kernel Classifiers: Theory and Algorithms*. MIT Press, Massachusetts, USA, 2001.

[56] A. J. Smola and B. Schölkopf. A tutorial on support vector regression. *Statistics and Computing*, 14(3):199–222, 2004.

[57] R. D. A. Timoteo, D. C. Cunha, L. N. Silva, and G. D. C. Cavalcanti. A hybrid machine learning approach for mobile user positioning in cellular networks. In *XXXV Simpósio Brasileiro de Telecomunicações e Processamento de Sinais (SBrT)*, 2017.

[58] G. Sun and W. Guo. Robust mobile geo-location algorithm based on LS-SVM. *IEEE Transactions on Vehicular Technology*, 54(3):1037–1041, 2005.

[59] R. D. A. Timoteo, D. C. Cunha, and G. D. C. Cavalcanti. A proposal for path loss prediction in urban environments using support vector regression. In *Proc. Advanced Int. Conf. Telecommun*, pages 1–5, 2014.

[60] A. Smola et al. Appearance-based Object Recognition using SVMs: Which Kernel Should I Use? In *Proc. of Workshop on Statistical Methods for Computational Experiments in visual Processing and computer Vision*, Whistler, 2002.

[61] R. D. A. Timoteo, D. C. Cunha, and G. D. C. Cavalcanti. A proposal for path loss prediction in urban environments using support vector regression. In *Proc. 10th Advanced Int. Conf. on Telecommunications (AICT)*, pages 1–5, Paris, France, 2014.

[62] R Core Team. *A language and environment for statistical computing*. Vienna, Austria, 2013.

[63] M. Kuhn. Building predictive models in R using the caret Package. *Journal of Statistical Software*, 28(5):1–26, 2008.

[64] L. L. Oliveira, G. W. A. Silva, R. D. A. Timoteo, and D. C. Cunha. An RSS-based regression model for user equipment location in cellular networks using machine learning. *Springer Wireless Networks*, pages 1–10, 2018.

[65] B. Dong Li, Z. Y. Zhang, and L. Cheng. A feature scaling based k-nearest neighbor algorithm for indoor positioning system. In *IEEE Global Communications Conference (IGCC)*, Huang, China, 2014.

[66] W. Zheng, F. Kechang, J. Esrafil, R. S. Shaeera, R. Rashid, and S. Mehrdad. A fast and resource efficient method for indoor positioning using received signal strength. *IEEE Transactions on Vehicular Technology*, 65(12):9747–9758, 2016.

[67] V. N. Thang et al. Machine learning for wideband localization. *IEEE Journal on Selected Areas in Communications*, 33(7):1357–1380, 2015.

[68] G. P. Bittencourt, A. F. Urbano, and D. C. Cunha. A proposal of an RF fingerprint-based outdoor localization technique using irregular grid maps. In *IEEE Wireless Communications and Networking Conference (WCNC)*, Bacelona, Spain, 2018.

[69] X. Meng, J. Bradley, B. Yavuz, E. Sparks, S. Venkataraman, D. Liu, J. Freeman, D. B. Tsai, M. Amde, S. Owen, et al. Mllib: Machine learning in apache spark. *The Journal of Machine Learning Research*, 17(1):1235–1241, 2016.

[70] D. Cook. *Practical machine learning with H2O: powerful, scalable techniques for deep learning and AI*. "O'Reilly Media, Inc.", 2016.

[71] I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning*. MIT press, Massachusetts, USA, 2016.

[72] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen. Zee: Zero-effort crowdsourcing for indoor localization. In *Proceedings of the 18th conference on Mobile computing and networking*, pages 293–304, Istanbul, Turkey, August, 2012. ACM.

CAPÍTULO

4 Fundamentals and Techniques for the Localization of a Sensor and the Mapping of an Environment Using Videos

Allan Freitas da Silva (UFRJ), Eduardo Antônio Barros da Silva (UFRJ) and Sergio Lima Netto (UFRJ)

Simultaneous Localization and Mapping (SLAM) is an active research area which is fundamental for several applications, such as robotics, autonomous driving and virtual reality. It allows a system to construct a map of an environment while keeping track of the sensor recording it. Among such methods, the ones that employ only visual sensors (usually referred to as visual SLAM or VSLAM) stand out due to the simplicity in the configuration. However, it comes at a cost of a higher technical difficulty, in special for the case of monocular devices.

This chapter investigates the use of monocular visual SLAM methods with focus on the Lie algebra formalism proposed by [1]. In order to introduce this algorithm to the reader, we present a description of projective geometry, which is used in most SLAM algorithms that rely only on visual content. We also present concepts of Lie algebra that are necessary for the understanding of the algorithm.

This chapter is organized as follows. In Section 4.2, we present the discussion about camera models and projective geometry, while Section 4.3 lectures about abstract algebra. Section 4.4 details the SLAM algorithm of interest and two algorithms used for comparison. In Section 4.6, some tests are performed using the three algorithms for a traditional database, and in Section 4.7 we discuss some of the current difficulties in the SLAM computation and present a challenging database. Section 4.8 summarizes the contents of this chapter.

4.1 Related Work

In general, SLAM methods use a camera and different auxiliary information, such as laser [2] or infrared signals [3]. In [4], for instance, a method developed for low-powered devices uses a camera mounted in an aerial vehicle and pointed downwards along with a height sensor and estimates visual maps using a graph-based formulation.

Among such methods, the visual SLAM is composed of approaches that use a camera as their primary sensor. Davison [5] uses a monocular camera and estimates the camera linear and angular velocities for each new frame, considering that between each measurement a random speed variation can occur. The work seen in [6] develops an object-oriented SLAM, which uses recognition algorithms to identify objects in the environment, which are used as features that are tracked along the frames.

Some of the most successful visual SLAM algorithms use stereo cameras [7], which are not widely spread. However, approaches using monocular cameras have several drawbacks since no information regarding the depth of the scene can be directly inferred from the images. To work around this problem, visual SLAM algorithms with monocular cameras usually use two main steps [1]: a visual odometry method to estimate camera poses and a loop closure step that detects if the camera returns to a known position, therefore preventing errors due to the uncertainty of the scale. The visual odometry step computes the epipolar geometry between frames, which is usually used to build submaps of the camera trajectory [8], and the results are refined with a bundle adjustment algorithm [9]. The loop closure step computes connections between submaps [8] or frames [10, 11] to detect loop closures, which are used to refine the results by minimizing a cost function.

An open source solution to solve the SLAM problem was proposed in [12]. It uses a parallel implementation to allow an efficient algorithm. In order to improve the robustness, the algorithm has an specific thread responsible for continuously optimizing and refining the results. In [13], the method was extended to the RGB-D and stereo cases.

A different family of methods deals with the image intensity values and estimates directly the camera poses without relying on intermediary structures such as image features or a fundamental matrix. One can cite [14], that performs a joint optimization of all model parameters and minimizes a photometric error. In order to achieve real-time detection, the method uses a set of sparse pixels and applies a sliding window in the frames. This method was further improved in [15], by also introducing a loop closure detection.

A visual SLAM method that has state-of-the-art results in trajectory estimation of a monocular calibrated camera has been proposed in [1]. This method develops a new formalism using notions of Lie algebra [16] with a graph-based optimization to estimate the trajectory and has shown significant advances for camera trajectory computation.

4.2 Camera Models and Projective Geometry

In this section, several concepts related to projection of the real world into a camera are depicted. This subject is of paramount importance to the understanding of the algorithms to recover the camera trajectory from a video sequence.

If several cameras record the same scene in different positions, there is a relationship between the positioning of the cameras and the images they create. As can be seen in Fig. 4.1, image points in multiple views that represent the same three-dimensional point define an intrinsic geometrical property among the cameras.

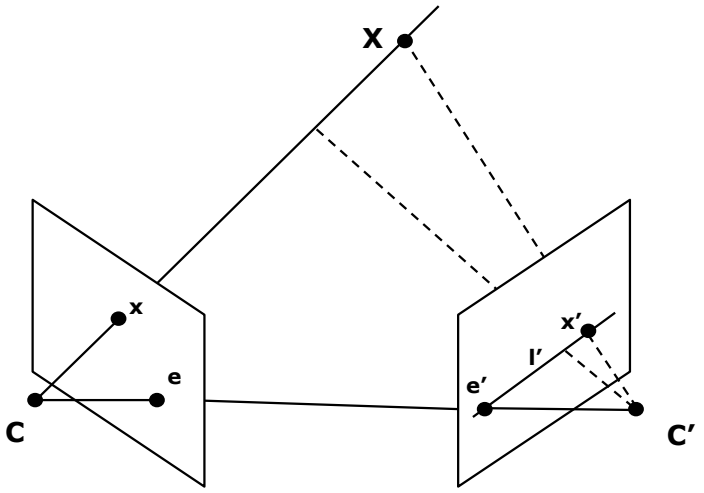


Figura 4.1 – Intrinsic geometry for two cameras representing the same scene. The three-dimensional point **X** is projected in the left image onto the point **x** and in the right image onto the point **x**′. The projection for each image is defined by the image plane and the camera center, **C** and **C**′ respectively for the left and right cameras.

In this example, a three-dimensional point **X** is projected in the left image onto the point **x**, and the projection ray passes through the camera center **C**. The point **x**′ is also a projection of the **X** on the right image, which passes through the camera center **C**′. Since the projection rays are assumed to be straight lines, the knowledge of an image point **x** in one image imposes a restriction on the position of the image point **x**′ in the other image. The next sections define a model for the camera and for the geometry between multiple views, and show how to explore these geometrical properties to infer information about the camera positioning based on the image content.

4.2.1 Homogeneous Coordinates

The mathematical definition of the camera projection and the relationship between images can be simplified if one considers the use of homogeneous coordinates. A point in the space $\mathbb{R}^2$ is usually represented by a vector $(x, y)^T$. By extending the vector to include a third component, for example, $(x, y, 1)^T$ it is said that the vector is represented in homogeneous

coordinates. This representation has the advantage of allowing a simplification of several operations.

For instance, a line in the space $\mathbb{R}^2$ is the set of points where the relation $ax+by+c=0$ is valid. Based on this definition, a line can be represented as the vector $\mathbf{l} = (a,b,c)^T$. In order to test if the point $\mathbf{x} = (x,y,1)^T$, in homogeneous coordinates, belongs to the line $\mathbf{l}$, it is necessary and sufficient to compute the internal product between $\mathbf{x}$ and $\mathbf{l}$, that is

$$\mathbf{x}^T\mathbf{l} = (x,y,1)(a,b,c)^T = ax+by+c, \tag{4.1}$$

which is zero if $\mathbf{x}$ belongs to the line $\mathbf{l}$. One should notice that if the vector $(x,y,1)$ belongs to the line $\mathbf{l}$, any vector of the form $(kx,ky,k)^T$ also belongs to this line, so that the set of vectors $(kx,ky,k)^T$ represent the same point $(x,y)^T$. Since the factor k can be arbitrary, it is often defined as $k=1$.

4.2.2 Camera Model

A simple approach to understand the operation of a camera is the model of a pinhole camera. It considers that the camera is composed of a box with a tiny aperture and a projection surface, without any lens in the exterior, and every light ray passes through the aperture and projects an inverted image onto a surface in the opposite side of the camera.

For convention, to further simplify the mathematics, it is often defined a virtual projection plane placed in front of the camera. In this paradigm, the projection occurs when the light ray crosses the projection plane towards the aperture, which is called the camera center. Considering the camera center as the origin of a coordinate system, the point $\mathbf{X}$ in the three-dimensional space and the equivalent point $\mathbf{x}$ in the projected image space, and considering the projection plane to be perpendicular to the z axis passing through the point $Z=f$, the projection of the point $\mathbf{X}$ to the point $\mathbf{x}$ can be expressed as:

$$\mathbf{x} = \mathbf{PX} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} fX \\ fY \\ Z \end{bmatrix} = \begin{bmatrix} fX/Z \\ fY/Z \\ 1 \end{bmatrix}, \tag{4.2}$$

where the matrix $\mathbf{P}$ that transforms the point $\mathbf{X}$ to the point $\mathbf{x}$ is called the camera matrix and f is the focal length.

In Fig. 4.2, the projection $\mathbf{p}$ of the camera center is considered as the origin of a coordinate system in the image. However, the origin of the coordinate system of an image is often defined as the top left or down left positions. If one wants to translate the coordinate system to a different location, the camera center must be compensated in Eq. (4.2), leading to the following equation:

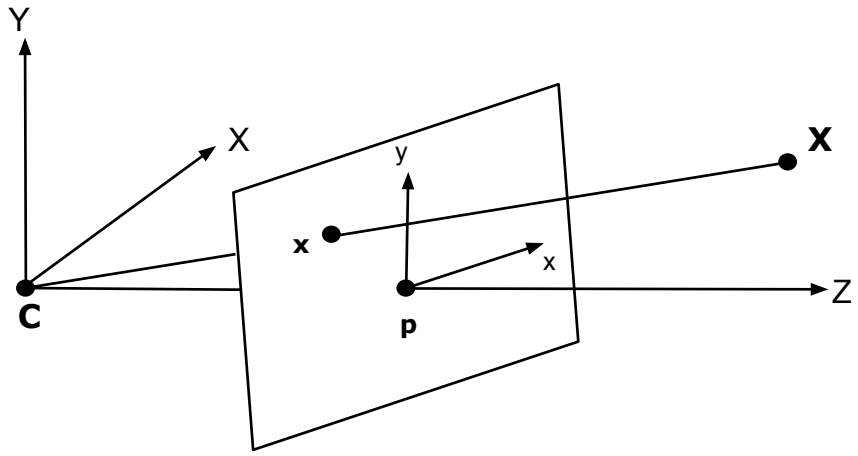


Figura 4.2 – Projection onto an image plane performed by a pinhole camera. The point **X** in the three-dimensional space is projected to the point **x** in the image plane through the light ray that crosses the camera center **C**.

$$\mathbf{x} = \begin{bmatrix} fX/Z + p_x \\ fY/Z + p_y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & p_x & 0 \\ 0 & f & p_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}. \tag{4.3}$$

If the coordinate system of the three-dimensional space should also be rotated or translated, which is the case for instance if there are multiple cameras to be modeled according to the same reference, the camera model is adapted to include a transformation that rotates and translates the coordinate system. The camera model becomes:

$$\mathbf{x} = \begin{bmatrix} f & 0 & p_x & 0 \\ 0 & f & p_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix} \mathbf{X} = \mathbf{K}\begin{bmatrix} \mathbf{R} & | & \mathbf{t} \end{bmatrix} \mathbf{X} = \mathbf{P}\mathbf{X}, \tag{4.4}$$

with

$$\mathbf{K} = \begin{bmatrix} f & 0 & p_x \\ 0 & f & p_y \\ 0 & 0 & 1 \end{bmatrix}. \tag{4.5}$$

The matrix **R** and the vector **t** represent, respectively, a rotation and a translation of the camera with respect to space coordinates, and are called the extrinsic parameters of the camera. The matrix **K** is called the calibration matrix and summarizes the intrinsic parameters of the camera, which are related to the projection of the three-dimensional points to generate the image points.

It is also possible to derive the expression of the camera center given the camera matrix **P**. Consider the points **A** and **C**, with **C** having the property that $\mathbf{PC} = \mathbf{0}$. Any point that belongs to the line connecting **A** and **C** can be generated using the following expression:

$$\mathbf{X} = \lambda\mathbf{A} + (1-\lambda)\mathbf{C}, \tag{4.6}$$

where λ is a variable used to parameterize a point in the line. The projection of the point $\mathbf{X}$ in the image is:

$$\mathbf{x} = \mathbf{P}\mathbf{X} = \lambda\mathbf{P}\mathbf{A} + (1-\lambda)\mathbf{P}\mathbf{C} = \lambda\mathbf{P}\mathbf{A}. \tag{4.7}$$

It should be noted that any point $\mathbf{X}$ defined by Eq. (4.6) possesses the same image point $\mathbf{x} = \lambda\mathbf{P}\mathbf{A}$. One can conclude that this line represents a projection ray, and since there was no assumption about the point $\mathbf{A}$, the remaining point $\mathbf{C}$ such that $\mathbf{P}\mathbf{C} = \mathbf{0}$ must be the camera center.

An even more generic model considers other effects. In CCD cameras, a pixel may not be square, which happens when the camera has different focal lengths in the horizontal (f_x) and vertical (f_y) directions. A camera may also have a shear distortion in the projected image, exemplified by the factor s, which occurs when the image axes x and y are not perpendicular. The camera model considering the aforementioned distortions has the following calibration matrix:

$$\mathbf{K} = \begin{bmatrix} f_x & s & p_x \\ 0 & f_y & p_y \\ 0 & 0 & 1 \end{bmatrix}. \tag{4.8}$$

4.2.3 Fundamental Matrix

As seen in Fig. 4.1, given two cameras recording the same scene, any point in the second camera corresponding to a point in the first camera necessarily must belong to a line that also contains the projection of the camera center of the first camera. The fundamental matrix is an object that summarizes the geometric relation between points from the two views.

The geometric relation between the image points can be explained as follows. Consider two cameras with known camera matrices $\mathbf{P}$ and $\mathbf{P}'$, and an image point $\mathbf{x}$ in the first image that is the projection of the three-dimensional point $\mathbf{X}$. The projection ray that creates the image point can be defined by two points: the camera center, where $\mathbf{P}\mathbf{C} = 0$, and any point that respects the relation $\mathbf{x} = \mathbf{P}\mathbf{X}$. According to [17], the second point can be obtained by computing the pseudoinverse of $\mathbf{P}$, as $\mathbf{X}^+ = \mathbf{P}^+\mathbf{x}$. The projection ray is a line defined as the set of points $\mathbf{X}(\lambda)$, for a parameter λ, such that:

$$\mathbf{X}(\lambda) = \mathbf{X}^+ + \lambda\mathbf{C}. \tag{4.9}$$

In the second view, the projection of any projection ray such as the one defined in Eq. (4.9) is called an epipolar line, and it passes through the projection of the two known points that were used to define it, $\mathbf{X}^+$ and $\mathbf{C}$. The projection of the camera center is

represented by $\mathbf{e}'$ and is called the epipole. The image points that define this epipolar line are:

$$\mathbf{e}' = \mathbf{P}'\mathbf{C}, \tag{4.10}$$

and

$$\mathbf{x}^+ = \mathbf{P}'\mathbf{X}^+ = \mathbf{P}'\mathbf{P}^+\mathbf{x}. \tag{4.11}$$

Representing the epipolar line as a vector, one can write [18]:

$$\mathbf{l}' = \mathbf{e}' \times \mathbf{x}^+ = [\mathbf{e}']_\times \mathbf{P}'\mathbf{P}^+\mathbf{x} = \mathbf{F}\mathbf{x}, \tag{4.12}$$

and

$$\mathbf{F} = [\mathbf{e}']_\times \mathbf{P}'\mathbf{P}^+. \tag{4.13}$$

where $[\mathbf{e}']_\times$ is an antisymmetric matrix created from the components of $\mathbf{e}' = (e'_1, e'_2, e'_3)^T$ in order to transform a vectorial product into a scalar product, using the following map:

$$[\mathbf{e}']_\times = \begin{bmatrix} 0 & -e'_3 & e'_2 \\ e'_3 & 0 & -e'_1 \\ -e'_2 & e'_1 & 0 \end{bmatrix}. \tag{4.14}$$

The matrix $\mathbf{F}$ is called a fundamental matrix and establishes a relationship between two views from the same scene: an image point $\mathbf{x}$ from the first image defines a projection ray, and the projection of this line in the second image defines the epipolar line $\mathbf{l}'$. If one knows the point $\mathbf{x}$ from one image and the fundamental matrix, the corresponding point $\mathbf{x}'$ in the second image must belong to the epipolar line $\mathbf{l}'$. Thus, one can find the following equation relating the corresponding points $\mathbf{x}$ and $\mathbf{x}'$ in the two views:

$$0 = \mathbf{x}'^T \mathbf{l}' = \mathbf{x}'^T \mathbf{F}\mathbf{x}. \tag{4.15}$$

4.2.4 Essential Matrix

The essential matrix can be interpreted as a particular case of the fundamental matrix when the calibration matrix is known. Since a camera matrix is given by the expression $\mathbf{P} = \mathbf{K}\begin{bmatrix}\mathbf{R} & | & \mathbf{t}\end{bmatrix}$, one can remove the effect of the calibration matrix, which is equivalent to using a normalized coordinate system in the image, with:

$$\hat{\mathbf{P}} = \mathbf{K}^{-1}\mathbf{P} = \begin{bmatrix}\mathbf{R} & | & \mathbf{t}\end{bmatrix}, \tag{4.16}$$

and then

$$\mathbf{x} = \mathbf{K}^{-1}\mathbf{P}\mathbf{X} = \mathbf{K}^{-1}\mathbf{x}. \tag{4.17}$$

Given a pair of normalized cameras, $\hat{\mathbf{P}} = \begin{bmatrix}\mathbf{I} & | & \mathbf{0}\end{bmatrix}$ and $\hat{\mathbf{P}}' = \begin{bmatrix}\mathbf{R} & | & \mathbf{t}\end{bmatrix}$, the essential

matrix that represents the relationship between the cameras is given by:

$$\mathbf{E} = [\mathbf{t}]_{\times}\mathbf{R} = \mathbf{R}[\mathbf{R}^T\mathbf{t}]_{\times}, \tag{4.18}$$

which shows that the essential matrix depends only on the relative rotation and translation between both cameras.

The essential matrix also defines the relation between corresponding normalized points, similarly to Eq. (4.15):

$$\hat{\mathbf{x}}'^T\mathbf{E}\hat{\mathbf{x}} = 0. \tag{4.19}$$

Replacing Eq. (4.17) in Eq. (4.15), one can find a relation between the fundamental and essential matrices, given the calibration matrices:

$$\mathbf{E} = \mathbf{K}'^T\mathbf{F}\mathbf{K}. \tag{4.20}$$

4.2.5 Computation of the Fundamental Matrix

Eq. (4.13) shows how to retrieve the fundamental matrix that relates two images if the camera matrices are known. However, a common application is the case where only the images are known, and one wants to infer properties of the positioning of the cameras and the three-dimensional space. In this situation, the fundamental matrix must be computed directly from the image information.

Feature detector and descriptor algorithms are tools that can be used to estimate corresponding points in two images. Algorithms such as the scale-invariant feature transform (SIFT) [19], the speeded-up robust features (SURF) [20], the binary robust invariant scalable keypoints (BRISK) [21], or the fast retina keypoint (FREAK) [22] detect representative points in the images and create, for each point, a feature descriptor, often based on the local information. These descriptors can be used to estimate corresponding points between the images, by pairing points with similar descriptors.

The estimation of the fundamental matrix can be made using a set of corresponding points. Each pair of corresponding points provides an equation on the elements of the fundamental matrix given by Eq. (4.15). Using a sufficient number of points, one can create a system of equations to solve for the elements of the fundamental matrix.

Some of the classical algorithms to compute the fundamental matrix are the eight-point algorithm [23] , the seven-point algorithm [24] and the five-point algorithm [25]. The eight-point algorithm [23] requires at least eight pairs of corresponding points to compute the eight unknown elements of the matrix (which is of size 3×3), assuming that in homogeneous coordinates the scale can be disregarded. The seven-point algorithm [24] also includes the restriction that $\det(\mathbf{F}) = 0$, therefore only seven points are necessary. The five-point algorithm [25] is used in the case where the camera calibration is known, and also includes restrictions on the essential matrix.

4.2.6 Reconstruction from Two Views

If there is no information about the three-dimensional space and only the images are known, it is possible to estimate information of the cameras disposition from the image contents. Combining Eqs. (4.15) and (4.2), one deduces that:

$$0 = \mathbf{x}'^T\mathbf{F}\mathbf{x} = (\mathbf{P}'\mathbf{X})^T\mathbf{F}(\mathbf{P}\mathbf{X}) = \mathbf{X}^T(\mathbf{P}'^T\mathbf{F}\mathbf{P})\mathbf{X}, \tag{4.21}$$

which, in order to be true for any $\mathbf{X}$, implies that $\mathbf{P}'^T\mathbf{F}\mathbf{P}$ is skew-symmetric.

A possible pair of camera matrices that defines the fundamental matrix $\mathbf{F}$ is the following:

$$\mathbf{P} = \begin{bmatrix}\mathbf{I} & | & \mathbf{0}\end{bmatrix} \qquad \text{and} \qquad \mathbf{P}' = \begin{bmatrix}\mathbf{S}\mathbf{F} & | & \mathbf{e}'\end{bmatrix}, \tag{4.22}$$

since

$$\begin{bmatrix}\mathbf{S}\mathbf{F} & | & \mathbf{e}'\end{bmatrix}^T \mathbf{F} \begin{bmatrix}\mathbf{I} & | & \mathbf{0}\end{bmatrix} = \begin{bmatrix}\mathbf{F}^T\mathbf{S}^T\mathbf{F} & \mathbf{0} \\ \mathbf{e}'^T\mathbf{F} & 0\end{bmatrix} = \begin{bmatrix}\mathbf{F}^T\mathbf{S}^T\mathbf{F} & \mathbf{0} \\ \mathbf{0} & 0\end{bmatrix} \tag{4.23}$$

is skew-symmetric if $\mathbf{S}$ is also skew-symmetric. In [26] it is proposed that $\mathbf{S} = [\mathbf{e}']_\times$.

In fact, there is a family of matrices similar to the ones shown in Eq. (4.22) that define the same fundamental matrix $\mathbf{F}$. They are of the form:

$$\mathbf{P} = \begin{bmatrix}\mathbf{I} & | & \mathbf{0}\end{bmatrix} \qquad \text{e} \qquad \mathbf{P}' = \begin{bmatrix}[\mathbf{e}']_\times\mathbf{F} + \mathbf{e}'\mathbf{v}^T & | & \lambda\mathbf{e}'\end{bmatrix}, \tag{4.24}$$

for any vector $\mathbf{v}$ and scalar λ. This indicates that there is an ambiguity in the reconstruction, which is further discussed in the subsection 4.2.8.

If the calibration matrices for both cameras are known beforehand, the cameras can be obtained from the essential matrix in a simpler way. As shown in Eq. (4.18), the essential matrix is defined by a rotation matrix $\mathbf{R}$ and a translation vector $\mathbf{t}$. Using an SVD decomposition of the essential matrix, it is possible to define a factorization of the form $\mathbf{E} = \mathbf{S}\mathbf{R}$. An SVD of the essential matrix is:

$$\mathbf{E} = \mathbf{U}\begin{bmatrix}1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0\end{bmatrix}\mathbf{V}^T. \tag{4.25}$$

Using

$$\mathbf{W} = \begin{bmatrix}0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0\end{bmatrix} \qquad \text{and} \qquad \mathbf{Z} = \begin{bmatrix}0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0\end{bmatrix}, \tag{4.26}$$

one finds that

$$\mathbf{S} = \mathbf{U}\mathbf{Z}\mathbf{U}^T = [\mathbf{t}]_\times \qquad \text{and} \qquad \mathbf{R} = \mathbf{U}\mathbf{W}\mathbf{V}^T \text{ or } \mathbf{U}\mathbf{W}^T\mathbf{V}^T. \tag{4.27}$$

From Eq. (4.27), one can see that there are two possible values for the matrix $\mathbf{R}$, which is due to a symmetry in the position of the cameras. In addition, with this decomposition, the matrix $\mathbf{S}$ necessarily has a Frobenius norm equal to $\sqrt{2}$ [18] and the corresponding vector

t is unitary, which shows that only a normalized vector can be estimated, unless some clue about the original scene is known beforehand.

Since the matrix **E** is also in homogeneous coordinates, it is not possible to infer the correct of sign of the components, as the scale of the matrix is normalized, therefore a scale of -1 produces the same matrix. Combining the indefinition of the sign and the two solutions for the rotation, one concludes that this decomposition defines four possible candidates for the camera position, which are symmetrical among them, as seen in Fig. 4.3. In order to distinguish among these four cases, it is often defined that all three-dimensional points must be facing the cameras. Thus, using a triangulation technique, one can obtain the solution that provides the largest number of points in front of the cameras.

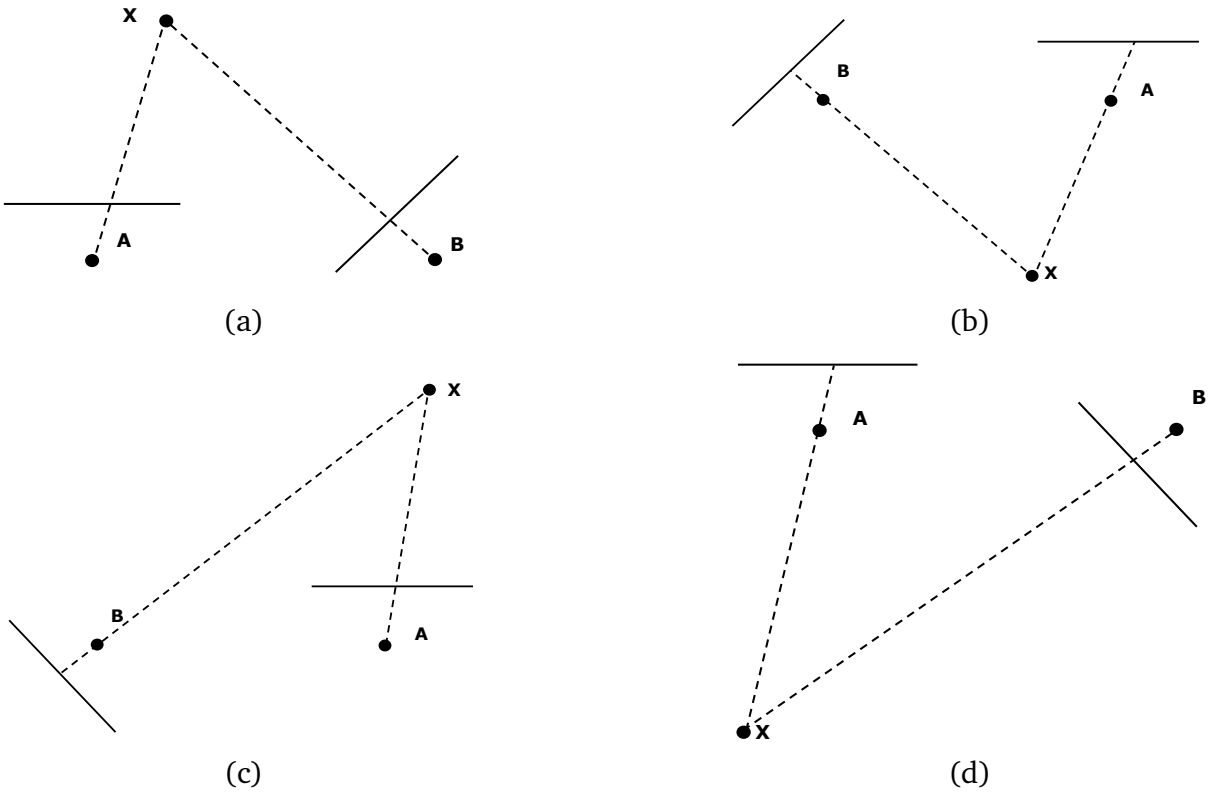


Figura 4.3 – Possible solutions for the decomposition of the essential matrix in a pair of cameras. The point **X** represents a three-dimensional point that is projected into two cameras with centers **A** and **B**. (a) Point in front of both cameras. (b) Point behind both cameras. (c) Point in front of camera **A** and behind camera **B**. (d) Point in front of camera **B** and behind camera **A**.

In analogy to Eq. (4.24), which shows the pair of cameras when the calibration is unknown, the pair of cameras obtained using the essential matrix is:

$$\mathbf{P} = \begin{bmatrix} \mathbf{I} & | & \mathbf{0} \end{bmatrix} \qquad \text{and} \qquad \mathbf{P}' = \begin{bmatrix} \mathbf{R} & | & \lambda \mathbf{t} \end{bmatrix}, \tag{4.28}$$

for a scalar λ.

4.2.7 Triangulation

With a pair of camera matrices $\mathbf{P}$ and $\mathbf{P}'$ and the corresponding image points $\mathbf{x}$ and $\mathbf{x}'$, it is possible estimate the position of the three-dimensional point that was projected into the cameras. Since a point in the image plane and the corresponding camera matrix define a projection ray, two projection rays can be found. The intersection of those lines indicates the position of the point $\mathbf{X}$ in the space, which is the point where the projections $\mathbf{x} = \mathbf{PX}$ are $\mathbf{x}' = \mathbf{P}'\mathbf{X}$ valid. Fig. 4.4 shows an example of the triangulation.

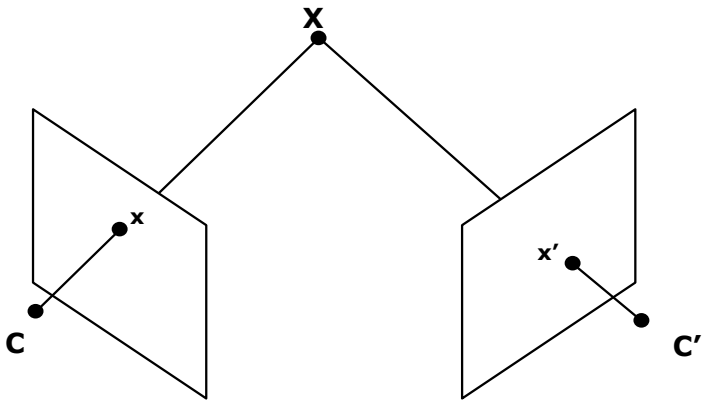


Figura 4.4 – Triangulation of a three-dimensional point without noise in the measurements. The point $\mathbf{X}$ indicates the triangulated point using the projections $\mathbf{x}$ and $\mathbf{x}'$.

Assuming that the estimation of the corresponding points has noise, the restriction defined by Eq. (4.15) may not be satisfied, which potentially leads to an error in the computation of the camera matrices. Therefore, it may not be possible to find a point $\mathbf{X}$ that is projected onto both images. In this case, one should either introduce some step to correct the measurements and define a criterion to select points that minimize a pre-defined error. Fig. 4.5 shows an example of the triangulation when the measurements have noise.

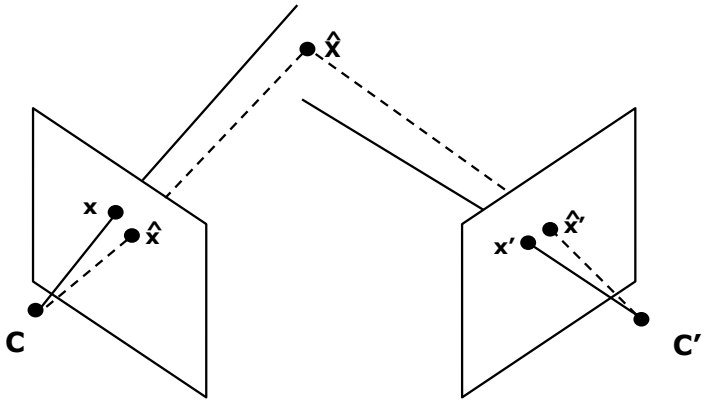


Figura 4.5 – Triangulation of a point with a noisy measurement of the corresponding points. The points $\mathbf{x}$ and $\mathbf{x}'$ represent an approximation of the corresponding points $\mathbf{x}$ and $\mathbf{x}'$ for which the projection rays intersect, and the point $\hat{\mathbf{X}}$ is the resulting triangulation.

An algorithm for the triangulation in the presence of noise is described in [18]. It minimizes a cost function that finds approximations of the corresponding points where the

epipolar geometry given by Eq. (4.15) is valid. Considering the points $\mathbf{x}$ and $\mathbf{x}'$, one aims to obtain the points $\hat{\mathbf{x}}$ and $\hat{\mathbf{x}}'$, respectively, subject to the restriction that $\hat{\mathbf{x}}^T\mathbf{F}\hat{\mathbf{x}}' = 0$, by minimizing the expression:

$$F = d(\mathbf{x}, \hat{\mathbf{x}})^2 + d(\mathbf{x}', \hat{\mathbf{x}}')^2, \tag{4.29}$$

where the operator $d(\mathbf{x}, \mathbf{y})$ represents, for instance, the L_2 norm between $\mathbf{x}$ and $\mathbf{y}$.

Since for the corrected points $\hat{\mathbf{x}}$ and $\hat{\mathbf{x}}'$ Eqs. (4.12) and (4.15) should be true, one can chose to correct the point $\mathbf{x}$ by minimizing in Eq. (4.29) the distance between the point $\mathbf{x}$ and some epipolar line $\mathbf{l}$, analogously for $\mathbf{x}'$ and $\mathbf{l}'$. In addition, since there is a relation between $\mathbf{l}$ and $\mathbf{l}'$, it is possible to parameterize both lines using the same variable t. In this scheme, the optimization searches for the value of t that minimizes:

$$F = d(\mathbf{x}, \mathbf{l}(t))^2 + d(\mathbf{x}', \mathbf{l}'(t))^2, \tag{4.30}$$

which can be found as the solution of a sixth-th order polynomium in t [18].

4.2.8 Ambiguity in the Reconstruction

Section 4.2.6 shows that even if the fundamental or essential matrix is known, it is not possible to define unequivocally the pair of cameras that generated the input images. In fact, if there is no information about the original coordinate system in the three-dimensional space, it is not possible to recover the exact location of the objects using only a projection of the space. Any valid solution and the true solution are related by a transformation. This section details this ambiguity in the reconstruction.

Using a set of corresponding points $\mathbf{x_i}$ and $\mathbf{x_i'}$, it is possible to obtain a reconstruction of the scene $\{\mathbf{P}, \mathbf{P}', \mathbf{X}_i\}$, with cameras $\mathbf{P}$ and $\mathbf{P}'$ and the triangulated points $\mathbf{X}_i$. However, for any projective transformation $\mathbf{H}$, one can find a new triangulated point $\overline{\mathbf{X}}_\mathbf{i} = \mathbf{H}\mathbf{X_i}$ and a camera matrix $\overline{\mathbf{P}} = \mathbf{P}\mathbf{H}^{-1}$ that have the same projection in the images, since:

$$\overline{\mathbf{P}\mathbf{X}}_\mathbf{i} = \mathbf{P}\mathbf{H}^{-1}\mathbf{H}\mathbf{X_i} = \mathbf{P}\mathbf{X_i} = \mathbf{x}, \tag{4.31}$$

analogously for the other camera.

One should notice that, if only the points $\mathbf{x}$ and $\mathbf{x}'$ are known, it is not possible to distinguish between the reconstructions $\{\mathbf{P}, \mathbf{P}', \mathbf{X}_i\}$ and $\{\overline{\mathbf{P}}, \overline{\mathbf{P}}', \overline{\mathbf{X}}_i\}$, since they map to the same image points and define the same epipolar geometry. In this case, it is said that any reconstruction differs from the true one by a projective transformation, as is exemplified in Fig. 4.6. This conclusion can also be drawn from Eq. (4.24), which shows a decomposition of the fundamental matrix into two cameras with several degrees of freedom. In this case, the parametrization of the cameras mirrors the degrees of freedom of the projective transformation that defines the ambiguity of the reconstruction.

If the calibration matrices are known, only the extrinsic parameters of the cameras

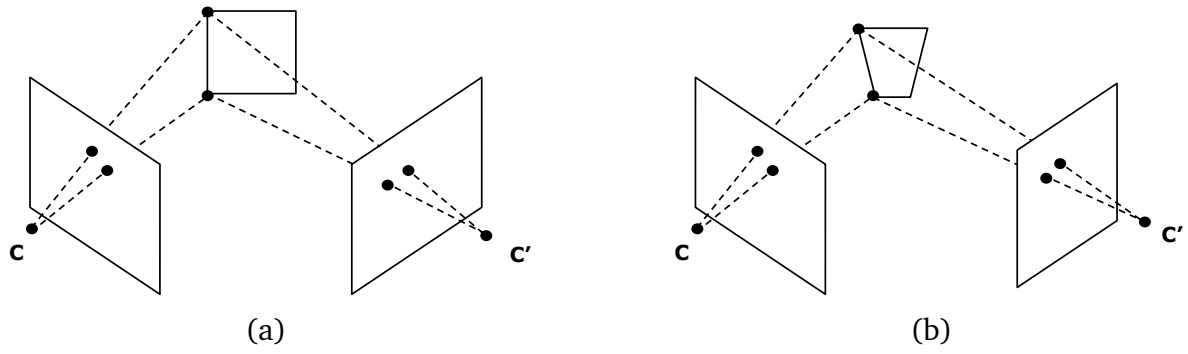


Figura 4.6 – Projective reconstruction of a scene. The reconstruction (b) differs from the real reconstruction (a) by a projective transformation.

need to be estimated. It is possible to find a reconstruction of a scene such that for the set of valid solutions, the projection rays for the image points always form the same angle with the image plane. In this case, the reconstruction is named a metric reconstruction, and any valid reconstruction is related to the true solution by a similarity transformation, as seen in Fig. 4.7. This case is analogous to Eq. (4.28), which shows a pair of reconstructed cameras where the second camera is rotated with respect to the first one and the displacement is normalized.

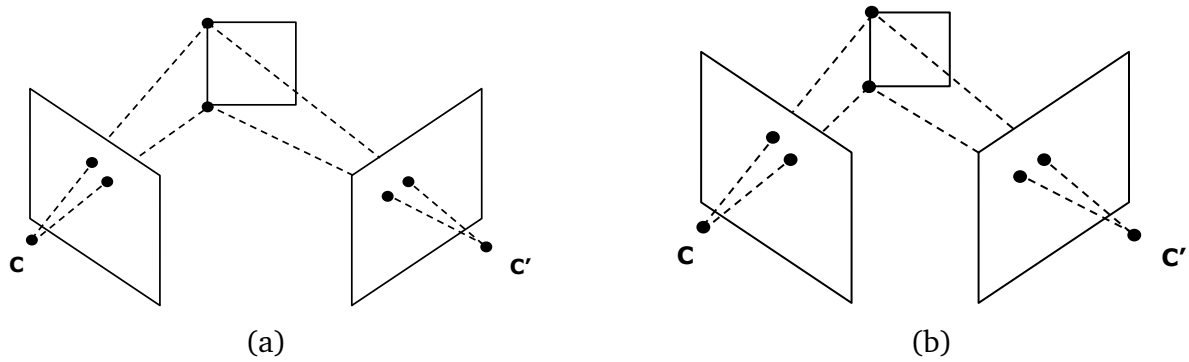


Figura 4.7 – Metric reconstruction of a scene. The reconstruction (b) differs from the real reconstruction (a) by a similarity transformation.

4.2.9 Reconstruction from Multiple Views

If three or more views are available, the mathematical development of the geometry that relates two views, described in the previous sections, can be extended. For three views, one can define a tensor [18] to relate the geometry of the views, in replacement of the fundamental matrix. Increasing the number of views, problems with an even greater dimensionality must be solved. For these cases, the solution often consists in splitting the views into pairs, reconstructing the scene for a pair of views and incrementally introducing the other views.

In order to solve this problem, one can use an initial pair of images and find corresponding points, which are used to estimate the fundamental matrix and find a

valid pair of camera matrices and a set of triangulated points. In a subsequent step, the other views are incorporated onto the initial reconstruction. For each view, one finds corresponding points between the current view and any other view already included in the reconstruction. For those points that were already used in the reconstruction, the triangulated points are already known, therefore there is a relation between three-dimensional points $\mathbf{X_i}$ that were triangulated using the previous images and image points $\mathbf{x_i}$ in the current image. With a sufficient number of points, the camera matrix for this view can be computed using Eq. (4.4). For the corresponding points that were not used in the reconstruction, new triangulated points are computed, increasing the point cloud.

A different approach involves the computation of an independent reconstruction for each pair of images, and then the normalization of every reconstruction to the same coordinate system. As discussed in Section 4.2.8, given a pair of images, there is a family of possible solutions for the reconstruction of the scene. If one uses two different pairs of images representing the same scene and finds the camera matrices independently for each pair, there is no guarantee that the reconstructions obtained for the first and second pairs are compatible, since due to the ambiguity, each one has its own coordinate system. However, among the family of possible solutions for the second reconstruction, one can assume that there should be a solution in the same coordinate system of the first one. The algorithm computes individual reconstructions and then finds the transformations that make them compatible.

Given the first two images $\mathbf{I}_1$ and $\mathbf{I}_2$, one finds the camera matrices $\mathbf{P}_1$ and $\mathbf{P}_2$ and triangulates the image points. The triangulated points $\mathbf{X_i}$ in this case are described with respect to a coordinate system based on the two cameras. Using a new view $\mathbf{I}_3$ and the last view $\mathbf{I}_2$, one creates another reconstruction with cameras $\hat{\mathbf{P}}_\mathbf{2}$ and $\hat{\mathbf{P}}_\mathbf{3}$ and a set of triangulated points $\hat{\mathbf{X}}_\mathbf{i}$. Using correspondences between the images $\mathbf{I}_1$, $\mathbf{I}_2$ and $\mathbf{I}_3$, one can find points in the first reconstruction of the scene, that is, in the set $\mathbf{X_i}$, that are equivalent to points in the set $\hat{\mathbf{X}}_\mathbf{i}$ of the second reconstruction.

Since it is desired that every reconstruction is grouped in a global one, one estimates a transformation $\mathbf{X_i} = \mathbf{H}\hat{\mathbf{X}}_\mathbf{i}$ that makes the second coordinate system coincide with the first one. In order for the projections in the image to remain the same, the new camera must be such that $\mathbf{P_3} = \hat{\mathbf{P}}_\mathbf{3}\mathbf{H}$, which represents the camera matrix $\hat{\mathbf{P}}_\mathbf{3}$ found in the second reconstruction described with respect to the coordinate system used in the first one.

4.3 Lie Groups and Lie Algebra

Lie groups arise from several structures in nature that present a continuous symmetry. In order to properly introduce the concept of Lie groups, we define some basic algebraic definitions that will be useful for the remaining of this chapter.

4.3.1 Group

Consider that G is a set and $\circ$ is a binary operation, also called group operator, that takes any two elements of G and returns an element of G. The pair $(G, \circ)$ is called a groupoid:

$$\forall g_1, g_2 \in G : g_1 \circ g_2 \in G. \tag{4.32}$$

To be considered a group, a groupoid must respect the following properties:

- Associativity: $\forall g_1, g_2, g_3 \in G : \ g_1 \circ (g_2 \circ g_3) = (g_1 \circ g_2) \circ g_3$;
- Existence of identity element: $\exists e \in G | \ \forall g \in G : \ e \circ g = g \circ e = g$;
- Existence of inverse element: $\forall g \in G : \ \exists g^{-1} \in G | \ g^{-1} \circ g = g \circ g^{-1} = e$.

4.3.2 Field

A field is a set F together with two operations $+$ and $\cdot$ from $F \times F$ to F such that the following properties are true:

- Associativity: $\forall a, b, c \in F : \ (a + b) + c = a + (b + c)$ and $(a \cdot b) \cdot c = a \cdot (b \cdot c)$;
- Commutativity: $\forall a, b \in F : \ a + b = b + a$ and $a \cdot b = b \cdot a$;
- Existence of identity element: $\exists e_a \in F | \ \forall a \in F : \ a + e_a = a$ and $\exists e_m \in F | \ \forall a \in F : \ a \cdot e_m = a$;
- Existence of inverse element: $\forall a \in F : \ \exists (-a) \in F | \ a + (-a) = e_a$ and $\forall a \in F, a \neq e_a : \ \exists (a^{-1}) \in F | \ a \cdot (a^{-1}) = e_m$;
- Distributivity of multiplication over addition: $\forall a, b, c \in F : \ a \cdot (b + c) = (a \cdot b) + (a \cdot c)$;

4.3.3 Vector Space

Given a field F and a set V, defining two operations $+$ from $V \times V$ to V and $\cdot$ from $F \times V$ to V, a vector space over F is the set V together with the operations $+$ and $\cdot$ with the following properties:

- Associativity of addition: $\forall \mathbf{u}, \mathbf{v}, \mathbf{w} \in V : \ (\mathbf{u} + \mathbf{v}) + \mathbf{w} = \mathbf{u} + (\mathbf{v} + \mathbf{w})$;
- Commutativity of addition: $\forall \mathbf{u}, \mathbf{v} \in V : \ \mathbf{u} + \mathbf{v} = \mathbf{v} + \mathbf{u}$;
- Existence of additive identity element: $\exists \mathbf{e} \in V | \ \forall \mathbf{u} \in V : \ \mathbf{u} + \mathbf{e} = \mathbf{u}$;
- Existence of additive inverse element: $\forall \mathbf{u} \in V : \ \exists (-\mathbf{u}) \in V | \ \mathbf{u} + (-\mathbf{u}) = \mathbf{e}$;
- Associativity of scalar multiplication: $\forall \mathbf{u} \in V, \ a, b \in F : \ a \cdot (b \cdot \mathbf{u}) = (ab) \cdot \mathbf{u}$;

- Distributivity of scalar multiplication with respect to vector addition: $\forall \mathbf{u}, \mathbf{v} \in V,\ a \in F:\ a \cdot (\mathbf{u} + \mathbf{v}) = a \cdot \mathbf{u} + a \cdot \mathbf{v}$;
- Distributivity of scalar multiplication with respect to field addition: $\forall \mathbf{u} \in V,\ a, b \in F:\ (a + b) \cdot \mathbf{u} = a \cdot \mathbf{u} + b \cdot \mathbf{u}$;
- Existence of scalar multiplication identity element: $\exists e_m \in F |\ \forall \mathbf{u} \in V:\ e_m \cdot \mathbf{u} = \mathbf{u}$;

4.3.4 Algebra

Assuming a field F and a vector space A over F with an additional binary operation $\cdot$ from $A \times A$ to A, we say that A is an algebra over F if it has the following properties:

- Right distributivity: $\forall \mathbf{x}, \mathbf{y}, \mathbf{z} \in A:\ (\mathbf{x} + \mathbf{y}) \cdot \mathbf{z} = \mathbf{x} \cdot \mathbf{z} + \mathbf{y} \cdot \mathbf{z}$;
- Left distributivity: $\forall \mathbf{x}, \mathbf{y}, \mathbf{z} \in A:\ \mathbf{x} \cdot (\mathbf{y} + \mathbf{z}) = \mathbf{x} \cdot \mathbf{y} + \mathbf{x} \cdot \mathbf{z}$;
- Scaling: $\forall \mathbf{x}, \mathbf{y} \in A,\ \forall a, b \in F:\ (a\mathbf{x}) \cdot (b\mathbf{y}) = (ab)(\mathbf{x} \cdot \mathbf{y})$;

4.3.5 Lie Groups and Lie Algebra

A Lie group $(G, \circ)$ is a special kind of group that has a particular geometry for which the set G is a smooth manifold, such that the mappings $a(g_1, g_2) = g_1 \circ g_2$ and $b(g) = g^{-1}$ are both analytic. Therefore, the functions $a(g)$ and $b(g)$ are continuous, infinitely differentiable and can be expressed as a Taylor series that converge around any point in its domain.

In a matrix Lie group $(G, \circ)$, the elements are $\mathbf{g} \in G \subset \mathbb{R}^{N \times N}$ and the group operator $\circ$ is the matrix multiplication. Some important examples of matrix Lie group are:

- General linear group: $GL(N, \mathbb{R}) = \left\{\mathbf{A} \in \mathbb{R}^{N \times N} | \det(\mathbf{A}) \neq 0\right\}$;
- Orthogonal group: $O(N) = \left\{\mathbf{X} \in GL(N, \mathbb{R}) |\ \mathbf{X}^T\mathbf{X} = \mathbf{I}\right\}$;
- Special orthogonal group: $SO(N) = \left\{\mathbf{X} \in GL(N, \mathbb{R}) |\ \mathbf{X}^T\mathbf{X} = \mathbf{I}, \det(\mathbf{X}) = 1\right\}$;
- Rigid body motion: $SE(3) = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix}$, with $\mathbf{R} \in SO(3)$ and $\mathbf{t} = [t_1, t_2, t_3]^T$;
- 3D similarity: $Sim(3) = \begin{bmatrix} s\mathbf{R} & \mathbf{t} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix}$, with $\mathbf{R} \in SO(3)$, $\mathbf{t} = [t_1, t_2, t_3]^T$ and $s \in \mathbb{R}^+$.

Given a matrix Lie group, elements $\mathbf{g} \in G$ close to the identity element can be written as $\mathbf{g} = \exp(X) | X \in \mathscr{G}$, where $\mathscr{G}$ is an open neighborhood of $0^{N \times N}$ in the tangent space at the identity of G, and is called the Lie algebra $\mathscr{G}$ [16]. The matrix Lie algebra $\mathscr{G}$ associated to the Lie group G is the set of all matrices X such that the exponential of each X results in an

element of the Lie group G. The opposite is also valid, and the matrix logarithm provides the inverse mapping between an element of the Lie algebra and an element of the Lie group:

$$\exp_G : \mathcal{G} \to G \tag{4.33}$$

$$\log_G : G \to \mathcal{G}. \tag{4.34}$$

The Lie algebra $\mathcal{G}$ associated to a p-dimensional Lie group G is a p-dimensional vector space, so there is also a mapping between $\mathcal{G}$ and $\mathbb{R}^p$ which is defined by the $\vee$ operator:

$$[]_G^\vee : \mathcal{G} \to \mathbb{R}^p \tag{4.35}$$

$$[]_G^\wedge : \mathbb{R}^p \to \mathcal{G}. \tag{4.36}$$

In order to reduce the notation, it is also common to denote $\exp_G([]_G^\wedge)$ as $\exp_G^\wedge$ and $\log_G([]_G^\vee)$ as $\log_G^\vee$.

The theory of Lie groups provides a tool to define symmetries from a mathematical point of view. The Lie algebra represents the space tangent to the Lie group at the identity, having a one-to-one map between them. The importance of the Lie algebra is that, in general, it is easier to work on a linear space than the "curved" space defined by the Lie group.

4.3.6 Adjoint Representation

Since the Lie groups are usually non-commutative, we define a function Ad_G, called the adjoint representation of the Lie group G, to express the non-comutativity. For $\mathbf{X} \in G$ and $a \in \mathcal{G}$, we seek an element $\mathbf{b} \in \mathcal{G}$ in order to satisfy $\mathbf{X}\exp_G(\mathbf{a}) = \exp_G(\mathbf{b})\mathbf{X}$. It can be proved that [16]:

$$\mathbf{b} = \mathbf{X}\mathbf{a}\mathbf{X}^{-1} = \mathrm{Ad}_G(\mathbf{X})\mathbf{a}. \tag{4.37}$$

Other measurement of commutativity is the function ad_G, called the adjoint representation of the Lie algebra $\mathcal{G}$. For $\mathbf{a}, \mathbf{b} \in \mathbb{R}^p$, the function ad_G is defined as:

$$\mathrm{ad}_G(\mathbf{b})\mathbf{a} = \left[[\mathbf{b}]_G^\wedge [\mathbf{a}]_G^\wedge - [\mathbf{a}]_G^\wedge [\mathbf{b}]_G^\wedge\right]_G^\vee. \tag{4.38}$$

Recall that a product on the group represents the group operator $\circ$, which maps two elements of the group in another element of the group (for instance, a combination of two successive rotations define a new rotation). Therefore, the adjoint representation embodies the multiplicative structures of the group and the algebra.

4.3.7 Baker-Campbell-Hausdorff Formula

The BCH (Baker-Campbell-Hausdorff) formula [27] expresses the group product directly in $\mathbb{R}^p$. Given $\mathbf{X} = \exp^\wedge(\mathbf{a})$ and $\mathbf{Y} = \exp^\wedge(\mathbf{b})$, with $\mathbf{X}, \mathbf{Y} \in G$, the following equation is

valid:

$$\log_G^\vee(\exp_G^\wedge(\mathbf{a})\exp_G^\wedge(\mathbf{b})) = \mathbf{b} + \mathbf{J}_G(\mathbf{b})\mathbf{a} + O(\|\mathbf{a}\|^2), \tag{4.39}$$

where

$$\mathbf{J}_G(\mathbf{b}) = \sum_{n=0}^{\infty} \frac{B_n \mathrm{ad}_G(\mathbf{b})^n}{n!} = \mathbf{I} + \frac{1}{2}\mathrm{ad}_G(\mathbf{b}) + \cdots \tag{4.40}$$

is the left Jacobian of G and B_n are Bernoulli numbers. This equation defines a first-order Taylor linearization of the group product. One should also notice that this linearization is expressed with respect to the adjoint representation. If the Lie group is commutative, then

$$\log_G^\vee(\exp_G^\wedge(\mathbf{a})\exp_G^\wedge(\mathbf{b})) = \mathbf{b} + \mathbf{a}. \tag{4.41}$$

4.3.8 Concentrated Gaussian Distribution

The distribution of $\mathbf{X} \in G$ is called a (right) concentrated Gaussian distribution on G of mean μ and covariance $\mathbf{P}$, denoted $p(\mathbf{X}) = \mathcal{N}_G^R(\mu, \mathbf{P})$, if:

$$\mathbf{X} = \exp_G^\wedge(\epsilon)\mu, \tag{4.42}$$

where $p(\epsilon) = \mathcal{N}_{\mathbb{R}^p}(0, \mathbf{P})$ and $\mathbf{P} \subset \mathbb{R}^{p\times p}$ is a symmetric positive-semidefinite matrix.

If the maximum of the eigenvalues of $\mathbf{P}$ is small, the probability mass is concentrated around μ and we may approximate $p(\mathbf{X})$ as:

$$p(\mathbf{X}) \approx \frac{1}{(2\pi)^p \det(\mathbf{P})} e^{-\frac{1}{2}\left\|\log_G^\vee(\mu^{-1}\mathbf{X})\right\|_P^2}. \tag{4.43}$$

4.3.9 Examples

Special Orthogonal Group $SO(2)$

The special orthogonal group $SO(2)$ represents the group of rotations in the two-dimensional plane, and is defined as:

$$SO(2) = \left\{\mathbf{R} \in \mathbb{R}^{2\times 2} | \mathbf{R}^T\mathbf{R} = \mathbf{I}, \det(\mathbf{R}) = 1\right\}. \tag{4.44}$$

The associated Lie algebra is:

$$so(2) = \left\{\mathbf{x} = \begin{bmatrix} 0 & -w \\ w & 0 \end{bmatrix} | w \in \mathbb{R}\right\}. \tag{4.45}$$

The adjoint representation $\mathrm{Ad}_{SO(2)}(\mathbf{R})$ is:

$$\mathrm{Ad}_{SO(2)}(\mathbf{R}) = \mathbf{I}. \tag{4.46}$$

It is important to mention that this group is commutative, since the combination of rotations with angles θ_1 and θ_2 is a rotation with angle $\theta_1 + \theta_2$. For this reason, the adjoint

is the identity matrix, which validates the commutative property.

Special Orthogonal Group $SO(3)$

The special orthogonal group $SO(3)$ represents the group of rotations in the three-dimensional space, and is defined as:

$$SO(3) = \left\{ \mathbf{R} \in \mathbb{R}^{3\times 3} | \mathbf{R}^T\mathbf{R} = \mathbf{I}, \det(\mathbf{R}) = 1 \right\}. \tag{4.47}$$

The associated Lie algebra is:

$$so(3) = \left\{ \mathbf{x} = \begin{bmatrix} 0 & -w_3 & w_2 \\ w_3 & 0 & -w_1 \\ -w_2 & w_1 & 0 \end{bmatrix} | w_1, w_2, w_3 \in \mathbb{R} \right\}. \tag{4.48}$$

The adjoint representation $\mathrm{Ad}_{SO(3)}(\mathbf{R})$ is:

$$\mathrm{Ad}_{SO(3)}(\mathbf{R}) = \mathbf{R}. \tag{4.49}$$

Contrary to the group $SO(2)$, the group $SO(3)$ is noncommutative. For instance, if one rotates an object by 90 degrees in one axis, and after that rotates it by 90 degrees in another axis, the result is different from the one obtained if the order of rotations is the inverse. For this reason, the adjoint is not the identity matrix.

Special Euclidean Group $SE(2)$

The special Euclidean group $SE(2)$ represents rigid transformations in the two-dimensional space. The group has three dimensions, corresponding to translation and rotation in the plane, and can be defined as:

$$SE(2) = \left\{ \mathbf{X} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix} | \mathbf{R} \in SO(2), \mathbf{t} = \begin{bmatrix} x \\ y \end{bmatrix} \in \mathbb{R}^2 \right\}. \tag{4.50}$$

The associated Lie algebra is:

$$se(2) = \left\{ \mathbf{x} = \begin{bmatrix} 0 & -w & v_1 \\ w & 0 & v_2 \\ 0 & 0 & 0 \end{bmatrix} | w, v_1, v_2 \in \mathbb{R} \right\}. \tag{4.51}$$

The adjoint representation $\mathrm{Ad}_{SE(2)}(\mathbf{X})$ is:

$$\mathrm{Ad}_{SE(2)}(\mathbf{X}) = \begin{bmatrix} \mathbf{R} & \mathbf{q} \\ \mathbf{0} & 1 \end{bmatrix} | \mathbf{R} \in SO(2), \mathbf{q} = \begin{bmatrix} y \\ -x \end{bmatrix} \in \mathbb{R}^2. \tag{4.52}$$

Special Euclidean Group $SE(3)$

The special Euclidean group $SE(3)$ represents rigid transformations in the three-dimensional space. The group has six dimensions, corresponding to translation and rotation in space, and can be defined as:

$$SE(3) = \left\{ \mathbf{X} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix} | \mathbf{R} \in SO(3), \mathbf{t} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \in \mathbb{R}^3 \right\}. \tag{4.53}$$

The associated Lie algebra is:

$$se(3) = \left\{ \mathbf{x} = \begin{bmatrix} 0 & -w_3 & w_2 & v_1 \\ w_3 & 0 & -w_1 & v_2 \\ -w_2 & w_1 & 0 & v_3 \\ 0 & 0 & 0 & 0 \end{bmatrix} | v_1, v_2, v_3, w_1, w_2, w_3 \in \mathbb{R} \right\}. \tag{4.54}$$

The adjoint representation $\mathrm{Ad}_{SE(3)}(\mathbf{X})$ is[1]:

$$\mathrm{Ad}_{SE(3)}(\mathbf{X}) = \begin{bmatrix} \mathbf{R} & [\mathbf{t}]_\times \mathbf{R} \\ \mathbf{0} & \mathbf{R} \end{bmatrix} | \mathbf{R} \in SO(3), [\mathbf{t}]_\times = \begin{bmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{bmatrix} \in \mathbb{R}^{3\times 3}. \tag{4.55}$$

Estimation of a Proper Three-Dimensional Rotation

In order to estimate a proper three-dimensional rotation, which has 3 degrees of freedom, one can write an expression of the matrix in function of the rotation angles. This can be made by the use of the Euler angles parametrization:

$$\mathbf{X} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix} \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{4.56}$$

However, this parametrization, despite allowing the optimization using a single 3-parameter vector $[\alpha, \beta, \gamma]$, has a drawback. If, for instance, $\beta = \pi/2$, then

$$\mathbf{X} = \begin{bmatrix} 0 & 0 & 1 \\ \sin(\alpha+\gamma) & \cos(\alpha+\gamma) & 0 \\ -\cos(\alpha+\gamma) & \sin(\alpha+\gamma) & 0 \end{bmatrix}, \tag{4.57}$$

[1]For this particular case, we express the adjoint as $\mathbf{b} = \mathrm{Ad}_G(\mathbf{X})[\mathbf{a}]_G^\wedge$ in comparison to Eq. (4.37), since it yields a simpler notation.

which means that the angles α and γ become coupled and changes in any of them produce the same result, a change in the angle $(\alpha+\gamma)$. This effect is called gimbal lock and produces a loss of a degree of freedom under certain conditions.

A second approach is to use the matrix space:

$$\mathbf{X} = \begin{bmatrix} x_1 & x_2 & x_3 \\ x_4 & x_5 & x_6 \\ x_7 & x_8 & x_9 \end{bmatrix}. \tag{4.58}$$

For this approach, the optimization is performed in the 9-parameter vector $[x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9]$ with additional constraints $\mathbf{X}^T\mathbf{X} = \mathbf{I}$ and $\det(\mathbf{X}) = 1$. Thus, an algorithm has to estimate nine parameters, in contrast to the tree parameters used in the previous representation. It must solve a constrained optimization problem, which is more complex and more susceptible to ill-conditioning.

A third approach is to model the proper three-dimensional rotation as belonging to the Lie group $SO(3)$, which has an associated Lie algebra of the form:

$$\mathbf{X} = \begin{bmatrix} 0 & -w_3 & w_2 \\ w_3 & 0 & -w_1 \\ -w_2 & w_1 & 0 \end{bmatrix} = \sum_{i=1}^{3} w_i \mathbf{E_i}, \tag{4.59}$$

for some bases

$$\mathbf{E_1} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}, \mathbf{E_2} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix} \text{ and } \mathbf{E_3} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{4.60}$$

Since the Lie algebra can be mapped to a three-dimensional Euclidean space, the optimization can be performed in a 3-parameter vector. In this case, the optimization operates in a space where every matrix has size 3×3 and intrinsically respects the constraints, which, besides providing a more elegant solution, also has better convergence properties. In addition, several operations such as composition, inversion, differentiation, and interpolation, can be addressed by the theory of Lie groups.

4.4 Robust Large Scale Monocular Video SLAM

The work developed in [1] presents an algorithm for the trajectory estimation of a monocular calibrated camera evolving in a large unknown environment. This work develops a SLAM algorithm that employs the concept of Lie groups to robustly align trajectories estimated in multiple submaps. To align a larger number of submaps, the work proposes a graph-based optimization algorithm, which also employs an efficient outlier-removal step.

This SLAM algorithm is composed of four main modules, which are depicted in Fig. 4.8. To reduce the computational complexity and also to ensure that pairs of

frames have a minimum camera displacement between them, a keyframe selection step is employed. The keyframes are split in submaps and inside each submap the algorithm estimates the camera trajectory along the frames. In order to align all submaps, three-dimensional similarities between pairs of submaps, which transform the coordinates of one submap to another, are computed. The recovered submaps and the 3D similarities between submaps are used in the relative similarity averaging step, that computes the three-dimensional similarities that take each submap to a common global coordinate.

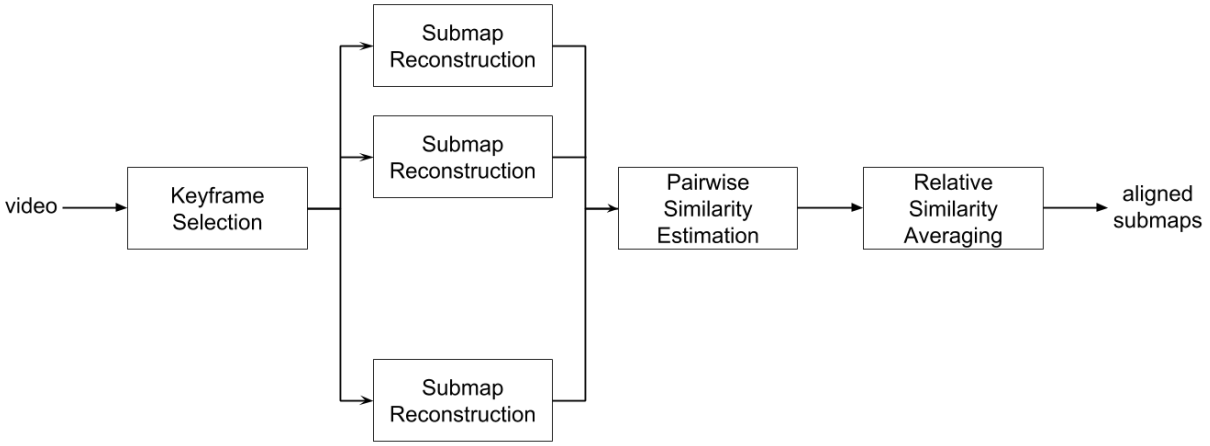


Figura 4.8 – Block diagram of the SLAM algorithm proposed in [1].

4.4.1 Keyframe Selection

To perform a keyframe selection, it is necessary to use a fast method that will be applied in the whole set of frames. Thereby, the algorithm applies a Lucas-Kanade tracker [28], which detects and tracks Harris points of interest (PoI) [29] in the video frames. A frame is selected as a keyframe when the Euclidean distance between the corresponding PoI of the current frame and the previous keyframe is bigger than a given threshold (which is typically 5% of the image width).

Fig. 4.9 exemplifies the keyframe selection step. The method starts with the first frame being considered a keyframe. The ensuing frames are tested and only the one whose content displays a substantial difference with respect to the previous keyframe, which is represented in the figure as the one where the black circle moves a minimum amount of pixels, is defined as another keyframe.

4.4.2 Submap Reconstruction

The set of keyframes selected in the previous step is split in clusters of L consecutive frames with overlap factor of 50% and, for each keyframe, SURF keypoints [20] are computed. For each cluster (or submap), SURF descriptors are matched and used in the estimation of corresponding points between pairs of keyframes. In order to increase the number of connections among frames, reducing the occurrence of incremental errors,

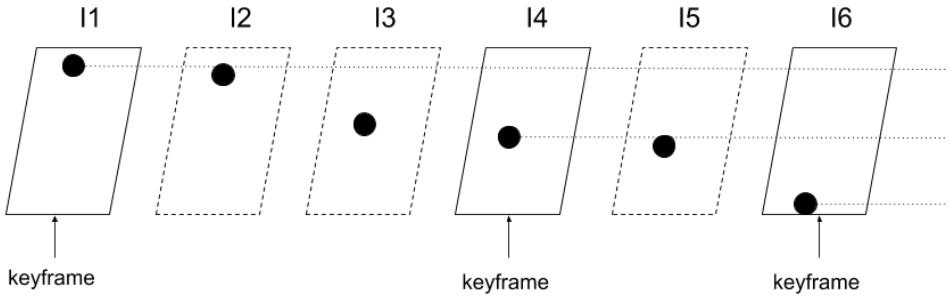


Figura 4.9 – Example of the keyframe selection step. The sequence of frames is represented as the dashed parallelograms and the ones considered as keyframes are displayed with solid lines. Significant difference from the previous keyframe is used to classify the next keyframe.

this step is performed for all pairs of consecutive frames. and also for some pairs of non-consecutive frames. The essential matrix is estimated using the five point algorithm, combined with a RANSAC algorithm [18] and a bundle adjustment optimization [9].

Using the essential matrix computed for a pair of frames, one can estimate the relative rotation between the orientation of the camera for each frame [18]. As a result of this calculation, several relative rotations between frames are estimated. These relative rotations estimated for all pairs of frames are then employed in the computation of a global orientation for each frame, in relation to a reference common to all frames. For this computation, the relative similarity averaging algorithm described in the following sections can also be employed.

After estimating a global orientation for each frame in the submap, the position of the camera for each frame still needs to be determined. In order to estimate the camera pose for each frame, keypoints are tracked among the frames and a linear programming is employed in the computation of the *known rotation problem* [30], which is described below.

Known rotation problem: For a camera matrix $\mathbf{P} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix} = \begin{bmatrix} \mathbf{R_1} & t_1 \\ \mathbf{R_2} & t_2 \\ \mathbf{R_3} & t_3 \end{bmatrix}$, $\mathbf{x} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$ is an image point with corresponding three-dimensional point $\mathbf{X}$. The reprojection error is given by:

$$E(\mathbf{X}, \mathbf{R}, \mathbf{t}) = \left\| \left(x - \frac{\mathbf{R_1 X} + t_1}{\mathbf{R_3 X} + t_3}, y - \frac{\mathbf{R_2 X} + t_2}{\mathbf{R_3 X} + t_3} \right) \right\|^2. \tag{4.61}$$

For the reprojection error to be less than a given threshold γ, this condition can be written as:

$$\|((x\mathbf{R_3} - \mathbf{R_1})\mathbf{X} + x t_3 - t_1, (y\mathbf{R_3} - \mathbf{R_2})\mathbf{X} + y t_3 - t_2)\|^2 \leq \gamma(\mathbf{R_3 X} + t_3)^2. \tag{4.62}$$

If $\mathbf{R}$ is known, this condition is a convex constraint, and linear programming can be used to solve simultaneously for $\mathbf{t}$ and $\mathbf{X}$.

In Fig. 4.10, one can see an example of the submap reconstruction step. Each submap

in this case is a set of consecutive keyframes which may contain an overlap with another submap. The camera trajectory for each submap is reconstructed by solving Eqs. (4.61) and (4.62). The dashed lines highlight the reconstructed trajectory for the frames that belong to the overlap of two submaps, therefore should represent the same trajectory. However, each reconstruction uses its own referential, so these trajectories must be rotated, scaled and translated with respect to each other. The next steps cope with the alignment of different referentials.

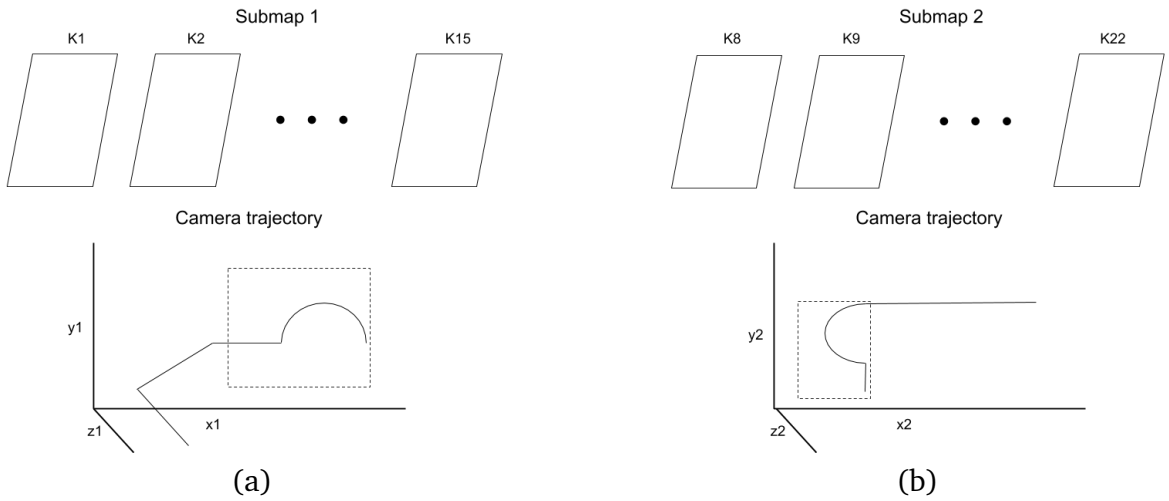


Figura 4.10 – Example of the submap reconstruction step. In this example, each submap is composed of a sequence of 15 consecutive frames with eight frames of overlap with the previous and next submaps. For each submap, a reconstruction of the camera trajectory is computed using Eqs. 4.61 and 4.62. The dashed lines highlight the reconstructed trajectory for the frames in the overlap of two consecutive submaps. (a) Submap 1. (b) Submap 2.

4.4.3 Pairwise Similarity Estimation

After the previous step, for each submap a camera trajectory and a cloud with triangulated points were estimated. However, the reconstruction for each submap was made according to a different coordinate system. In order to align all submaps, a three-dimensional similarity between pairs of submaps must be calculated, which can be seen as matrices that belong to the Lie group $Sim(3)$:

$$Sim(3) = \begin{bmatrix} s\mathbf{R} & \mathbf{t} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix}, \tag{4.63}$$

with $R \in SO(3)$, $\mathbf{t} = [t_1, t_2, t_3]^T$ and $s \in \mathbb{R}^+$.

To reduce the number of similarities to compute, a bag-of-words [31] approach is applied to three-dimensional SURF descriptors of all submaps to find a unique descriptor for the whole submap. A similarity is determined for consecutive submaps and also between each submap and its 10 nearest neighbors using the bag-of-words descriptor as a metric of distance. One should notice that if the camera returns to a known position, it is expected

that the bag-of-words descriptors should be similar, therefore this step is also responsible for performing a loop closure.

In order to estimate a similarity between two submaps, SURF descriptors for each three-dimensional point are obtained by averaging the SURF descriptors of the image points that generated this triangulated point. The descriptors of the three-dimensional points are then matched between submaps, and a three-point algorithm [32] combined with the RANSAC and a refinement step is applied to obtain a three-dimensional similarity.

Finally, considering that this similarity can be modeled as a concentrated Gaussian distribution on the group $Sim(3)$, a covariance for each similarity is also found. In the end of this step, the algorithm has computed similarities $Z_{ij} \in Sim(3)$ between the coordinate system of the submap i and the submap j along with a covariance Σ_{ij} for these estimates.

Fig. 4.11 exemplifies the pairwise similarity estimation. The frames inside each submap are used for the triangulation of three-dimensional points. By tracking triangulated points across different submaps, it is possible to compute a pairwise similarity transformation between two submaps, which is composed of rotation, translation and scaling, that aligns the axis for both reconstructions to a same common axis. The next step takes all relative similarities and maps all axes to a global reference.

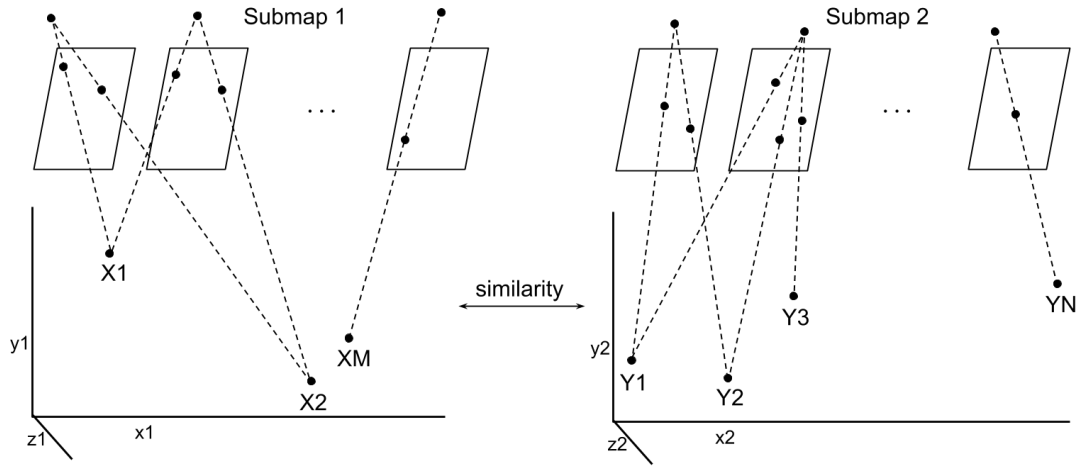


Figura 4.11 – Pairwise similarity estimation step. For each submap, keypoints inside the frames are tracked and triangulated to three-dimensional points (points X1,...,XM for submap 1 and Y1,...,YM for submap 2). Using corresponding three-dimensional points from two submaps, a similarity transformation is computed that transforms a point represented according to the axis x1,y1,z1 to a point represented according to the axis x2,y2,z2.

4.4.4 Relative Similarity Averaging

From the results obtained in the previous subsection, relative similarities Z_{ij} that align the submaps i and j were computed, along with a covariance matrix Σ_{ij}. In this step, we need to estimate the global similarities (X_{iS}, X_{jS}), that is, the three-dimensional similarities between a global reference frame S and each submap. Given the submaps i and j, one can consider that the similarity Z_{ij} that takes from the submap i to the submap j should be equivalent to going from the submap i to the reference frame S (using the global similarity

X_{iS}), and then going from the reference frame S to the submap j (using X_{jS}^{-1}). Considering the existence of noise in the measurements, represented by the covariance matrix Σ_{ij}, the following model is obtained:

$$Z_{ij} = \exp^{\wedge}(b_{ij}^{i}) X_{iS} X_{jS}^{-1}, \tag{4.64}$$

where $b_{ij}^{i} \sim \mathcal{N}_{\mathbb{R}^p}(\mathbf{0}_{p\times 1}, \Sigma_{ij})$ is a white Gaussian noise.

Considering that the measurements Z_{ij} are outlier-free, an estimate of the global similarities X_{iS} and X_{jS} can be obtained by the relative similarity averaging problem, which minimizes the following cost function:

$$\underset{\{X_{iS}\}_{i\in\mathcal{V}}}{\operatorname{argmin}} \sum_{i,j\in\mathcal{E}} \left\| \log^{\vee} Z_{ij} X_{jS} X_{iS}^{-1} \right\|_{\Sigma_{ij}}^{2}, \tag{4.65}$$

with $\|\|_{\Sigma}^{2}$ representing the Mahalanobis distance. This equation is similar to a generalized least squares problem, where one estimates the distance between a model ($X_{jS}X_{iS}^{-1}$) and the estimate (Z_{ij}), pondering by the covariance of the error (Σ_{ij}). One can also note the function $\log^{\vee}$, which maps the similarities to the Lie algebra, where the optimization is performed.

If two submaps do not possess a significant interception of regions in the scene, a relative similarity computed between them can represent an outlier, which prejudices the minimization problem represented by Eq. (4.65). Thus, an outlier removal algorithm is necessary to solve the relative similarity averaging problem.

The problem given by Eq. (4.65) can also be seen as the inference problem in a factor graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$. In this context, each vertex $\mathcal{V}_i$ corresponds to a global similarity measurement X_{iS} and each pairwise factor $\mathcal{E}_{ij}$ corresponds to a relative measurement Z_{ij} that links the vertices $\mathcal{V}_i$ and $\mathcal{V}_j$. The following subsections describe an outlier removal algorithm and a relative similarity averaging algorithm that uses notions of graph optimization.

In Fig. 4.12, one can see an example of the relative similarity averaging step. Using the relative similarities computed in the previous step, for each submap is computed a similarity transformation that maps its axis to a global referential. The trajectories found for each submap, that can now be described with respect to the same referential, are merged to define the camera trajectory for the whole input video.

4.4.5 Outlier Removal Algorithm

To remove outlier measurements in the SLAM algorithm, it is assumed that every relative similarities between consecutive submaps are inliers. For the other relative similarities, the error inside a cycle (for instance, the error of transforming from submaps a to b, from b to c and then from c to a), is tested:

$$\epsilon^{T} P^{-1} \epsilon < t_{\chi^2}, \tag{4.66}$$

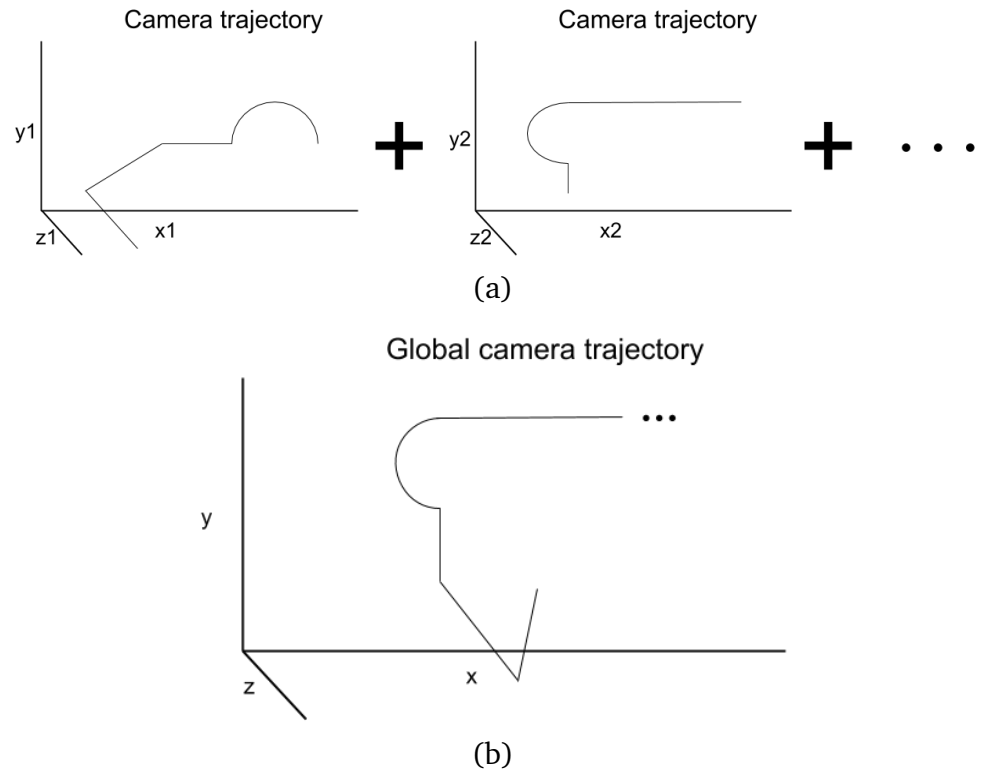


Figura 4.12 – Example of the relative similarity averaging step. The camera trajectory is estimated for each submap according to its own referential and contains only a part of the total trajectory. After computing relative similarities between pairs of submaps, all referential are mapped to a global one and the parts of the trajectory are merged to compose the total trajectory of the camera along the whole video. (a) Camera trajectories for each submap that are combined to form a single one. (b) Global camera trajectory for the whole video that is a composition of the trajectories computed for each submap.

where ϵ is the cycle error, P is the covariance associated with this cycle and t_{χ^2} is a value based on the χ^2.

A naive algorithm to test a relative similarity Z_{kl} could be to test the cycle $Z_{kl}Z_{l(l-1)}Z_{(l-1)(l-2)}...Z_{(k-1)k}$, which contains the similarity between the k-th and l-th submaps (Z_{kl}), and all consecutive similarities from the l-th to k-th submaps ($Z_{l(l-1)}...Z_{(k-1)k}$). However, this approach can fail for larger cycles, since it accumulates any small errors in each similarity. Instead of using consecutive measurements in the cycle, an algorithm proposed in [1] searches for the shortest cycles (in the sense of minimum number of connections) that contain only inliers. This algorithm is described in Alg. 2.

4.5 Other Approaches

Two other open source SLAM algorithms were also considered for this study. In this section, we provide a brief explanation for each algorithm.

Algorithm 2 Algorithm to remove outlier similarity measurements.

Input: Relative similarities Z_{ij}, covariance matrices Σ_{ij}, value of t_{χ^2}.
Output: Graph containing only inlier relative similarities.

```
Initialize an empty graph 𝒢 = {𝒱, ℰ}
Add the vertex X_1S to 𝒱
for k ∈ {1, ..., N} do
    Add the vertex X_kS to 𝒱
    Add the factor {Z_(k−1)k, Σ_(k−1)k} to ℰ
    for l ∈ {1, ..., k} do
        Find the shortest path from X_kS to X_lS in 𝒢
        Compute the cycle error ϵ and covariance P
        if ϵ^T P^−1 ϵ < t_χ² then
            Add the factor {Z_lk, Σ_lk} to ℰ
        end
    end
end
```

4.5.1 ORB-SLAM: Oriented Fast and Rotated Brief Simultaneous Localization and Mapping

ORB-SLAM [12] is a feature-based SLAM algorithm that uses the ORB [33] feature descriptor along with a parallel implementation to achieve a fast algorithm. It is composed of five main steps, which are shown in the block diagram depicted in Fig. 4.13: an initialization, that computes the first reconstruction for a pair of keyframes and estimates a map of the environment; a tracking of corresponding points between the keyframes and each new image, in order to estimate the pose of this image according to the current map; a relocalization, to be used if the tracking was lost; a local mapping, to further optimize the map and the poses; and a loop closure to prevent the error accumulation and the scale drift.

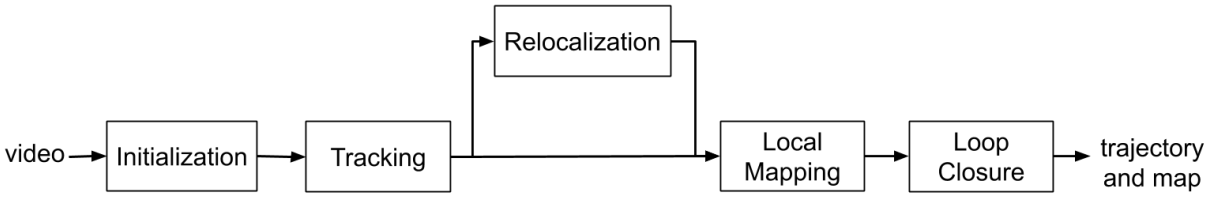


Figura 4.13 – Block diagram of the ORB-SLAM algorithm [12].

Initialization

For a first pair of keyframes, the algorithm computes the fundamental matrix and estimates the position of the cameras and a map of triangulated three-dimensional points (in this case, called map points), similar to Sections 4.2.6 and 4.2.7. Keypoints are extracted using a FAST corner detection [34] algorithm and a tracking of descriptors across the frames is made by the use of the ORB descriptor. In order to cope with planar scenes, it also

estimates a homography between the frames and decides whether to use it instead of the fundamental matrix. The initial frames are considered keyframes for the next steps.

Tracking

After the initialization of the trajectory and the map, the algorithm performs a tracking of the image keypoints in subsequent frames. For this purpose, in order to speed up the computation, it assumes initially a constant motion model. The keypoints in the previous keyframe that were triangulated (generating the map points) are searched within a certain area in the new image. If a sufficient number of matches were found, it is assumed that the constant motion model is valid and the corresponding keypoints in the new image are associated to the map points. Therefore, the pose for this frame can be computed using the correspondences between image points (keypoints) and three dimensional points (map points), using Eq. (4.4).

If it was not possible to find a sufficient number of matches, the constant motion model is disregarded. The algorithm creates a bag-of-words based on [35] for the previous keyframe and the new frame. By comparing the bag-of-words for both frames, it is able to determine the correspondences of keypoints in the images, also creating correspondences between image points and three dimensional points to estimate the pose. the algorithm is not able to find a sufficient number of corresponding points, it performs the relocalization step described bellow.

After the determination of matches for the new frame and the estimation of a pose for it, the local mapping refines the results by minimizing the reprojection error for the new frame and a set of keyframes, optimizing the camera poses and while also searching for new correspondences. Finally, the algorithm uses some criteria to determine if the frame should be considered a new keyframe.

Relocalization

If the tracking of keypoints between the new frame and the last keyframe did not yield a sufficient number of corresponding points, the algorithm searches among all keyframes for the frames that record the same scene as the current frame. For this purpose, it computes the bag-of-words described in [35] for the current frame and the set of keyframes, and searches for the keyframes that have similar features to the current frame.

If the algorithm finds similar keyframes, it uses the bag-of-words to track keypoints, in a similar manner as the previous step, creating matches between keypoints in the new frame and the triangulated map points. These correspondences are used in an attempt to estimate the pose for the new frame, followed by a refinement. Finally, the algorithm uses the number of obtained matches to decide if the pose is reliable and should be accepted.

Local Mapping

In parallel to the previous operation, the algorithm performs a local mapping step to keep up to date the poses and the map of the environment. It continuously removes and inserts new keyframes and map points, and removes and fuses point duplicates. Afterwards, it performs a refinement by the use of a bundle adjustment technique, that minimizes the reprojection error.

Loop Closure

To detect the existence of loops, the algorithm compares the bag-of-word of the current keyframe to all other keyframes that are not already connected to the current keyframe. If there is a consistent match, a similarity transform is computed [36] between the current keyframe and the loop candidate, to cope with the scale drift and other errors in the pose estimation. The algorithm uses this transformation to search for more matches between image points in the frames and between map points and image points. If enough inliers are found, the loop is accepted, and in this case, the poses are refined and corrected.

4.5.2 LDSO: Direct Sparse Odometry with Loop Closure

LDSO [15] is a method that improves the DSO [14] by introducing a loop closure. It uses image intensities instead of feature descriptors such as the ones seen in the previous methods (or any other intermediary representation) to track points across frames and estimate the poses (hence being a direct method), while using a sparse set of keypoints. The use of image intensities allows the method to be robust even in featureless regions of the frames (for instance, flat surfaces). The method estimates the poses by minimizing a photometric error (in contrast to a reprojection error), but includes the conventional computation of ORB features and bag-of-words seen in the previous methods to identify loop closures. A block diagram of this algorithm can be seen in Fig. 4.14.

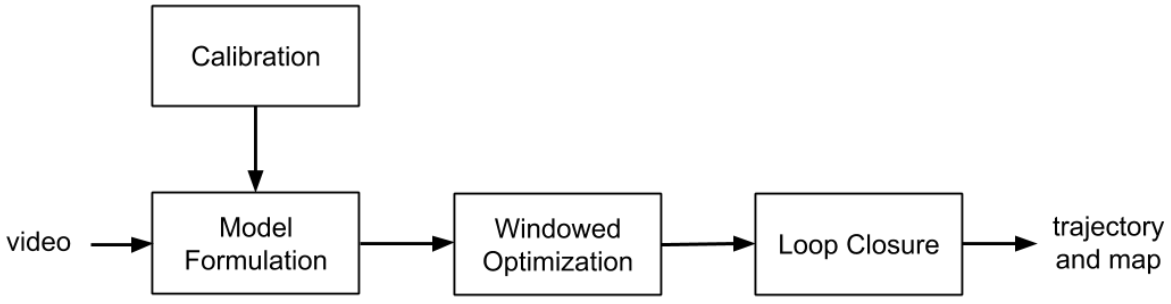


Figura 4.14 – Block diagram of the LDSO algorithm [15].

Calibration

Along with a pre-processing step that uses the pinhole camera model described in Section 4.2.2 to remove the radial distortion, this method also employs an image formation model [37] to compensate for the non-linear response function of the camera and the lens attenuation, improving the robustness to illumination changes.

Model Formulation

For each frame, the method computes the photometric error of a point that is observed both in a reference frame and a target frame as a weighted sum of squared differences over a neighborhood:

$$E_{\mathbf{p}j} := \sum_{\mathbf{p}\in\mathcal{N}_\mathbf{p}} w_\mathbf{p} \left\| \left(I_j\left[\mathbf{p}'\right] - b_j\right) - \frac{t_j e^{a_j}}{t_i e^{a_i}} \left(I_i[\mathbf{p}] - b_i\right) \right\|_\gamma \tag{4.67}$$

where in this equation $\mathcal{N}_\mathbf{p}$ indicates the neighborhood, $w_\mathbf{p}$ is a weight, $\|\|_\gamma$ is the Huber norm, t_i and t_j are the exposure times of the images I_i and I_j, and a_i, a_j, b_i and b_j account for possible differences in exposure times or illumination.

The position of the point $\mathbf{p}'$ in one image is calculated using the point $\mathbf{p}$ with the application of an inverse projection, a rigid body motion and a projection, which depend on the intrinsic parameters of the cameras and the poses involved. Therefore, one can express the photometric error with respect to the camera poses and estimate them by minimizing this error.

The full photometric error is:

$$E_{\text{photo}} := \sum_{i\in\mathcal{F}} \sum_{\mathbf{p}\in\mathcal{P}_i} \sum_{j\in\text{obs}(\mathbf{p})} E_{\mathbf{p}_j}, \tag{4.68}$$

where i accounts for all keyframes $\mathcal{F}$, $\mathbf{p}$ considers all keypoints $\mathcal{P}_i$ contained in the frame i, and j contains all frames obs($\mathbf{p}$) where the point $\mathbf{p}$ is observed.

Windowed Optimization

The algorithm minimizes the error given by Eq. (4.68) using a Gauss-Newton approach. To spare computation, it applies a sliding window in the keyframes, in order to optimize Eq. (4.68) without using the whole set of keyframes for each new frame. In addition, the algorithm continuously keeps track of which points $\mathbf{p}$ and frames $\mathcal{F}$ are used, and in which frames obs($\mathbf{p}$) a point is visible, including the use of techniques to remove outliers and detect occlusions.

Loop Closure

In order to employ a loop closure algorithm, the algorithm selects among the set of points $\mathcal{P}_i$ the ones that belong to corners and computes an ORB descriptor for each of

them. Using the features computed for each keyframe, a bag-of-words is computed and loop candidates are proposed. The algorithm searches for correspondences of ORB features in the loop candidate frames and computes a transformation between them, which is included in the optimization.

4.6 Experimental Results

The result of a SLAM algorithm is a sequence of camera matrices $\mathbf{P}_1, \cdots, \mathbf{P}_n \in SE(4)$ that are used to composed the camera trajectory (where we extend the 3×4 camera matrix defined in Section 4.2 by including a row $\begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}$). One can also assume that the ground truth can be converted to a set of poses $\mathbf{Q}_1, \cdots, \mathbf{Q}_n$ and there is a synchronism between the sequences. It is important to notice that each sequence is specified according to a different coordinate system, due to the ambiguity in the reconstruction. A common evaluation metric is the average trajectory error defined in [38] to compare the performance of a SLAM algorithm.

For the following results, the trajectory error was evaluated using an implementation in Python developed by [39]. Three algorithms were tested: the method of [1], referred to as RLS-MVSLAM, which was implemented in Matlab and is available at [40], and the C++ implementation of the algorithms ORB-SLAM2, using the monocular version of the algorithm available at [41], and LDSO, which is available at [42].

4.6.1 Average Trajectory Error

The average trajectory error (also called average pose error) measures the global consistency of the estimated trajectory. Since the trajectories may have a different coordinate system, a first step computes a rigid-body transformation $\mathbf{S}$ [36] (see Section 4.3.9) that maps the estimated trajectory onto the ground truth. For any given frame i, the absolute trajectory error can be defined as:

$$\mathbf{E}_i = \mathbf{Q}_i^{-1} \mathbf{S} \mathbf{P}_i. \tag{4.69}$$

From the error computed in Eq. (4.69), we obtain the translational component $\mathbf{t}_i$ using an operator defined as:

$$\text{trans}(\mathbf{E}_i) = \begin{bmatrix} \mathbf{1} & 0 \end{bmatrix} \mathbf{E}_i \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix}, \tag{4.70}$$

where for $\mathbf{E}_i = \begin{bmatrix} \mathbf{R_i} & \mathbf{t_i} \\ \mathbf{0} & 1 \end{bmatrix}$ we have:

$$\text{trans}(\mathbf{E}_i) = \begin{bmatrix} \mathbf{1} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R_i} & \mathbf{t_i} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{1} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{t_i} \\ 1 \end{bmatrix} = \mathbf{t_i} \tag{4.71}$$

The average trajectory error is defined as the root mean square error of the translational error:

$$\mathrm{ATE} = \left(\frac{1}{n} \sum_{i=1}^{n} \left\| \mathrm{trans}\left(\mathbf{E}_i\right) \right\|^2 \right)^{1/2} . \tag{4.72}$$

4.6.2 Tests with the KITTI Database

An experiment was performed using videos from the KITTI database [43], which is composed of videos acquired using an autonomous driving car moving along a road. This database also employs a laser scanner and a GPS to provide an accurate ground truth for the camera position. Some examples of frames from this database are shown in Fig. 4.15.

(a) (b)

(c) (d)

Figura 4.15 – Frames from sequence 2 in the KITTI odometry dataset. (a) Frame 30. (b) Frame 50. (c) Frame 1000. (d) Frame 4660.

The SLAM algorithms were tested in the sequences from the KITTI database with a ground truth in order to compare the results. Tab. 4.1 shows a comparison of the results obtained by RLS-MVSLAM, ORB-SLAM2 and LDSO. From these results, one can notice that for some sequences the methods have similar results, while for other sequences the error may differ by one order of magnitude. The method ORB-SLAM2 has the lowest error for five sequences, followed by the RLS-MVSLAM, which has the best results in four sequences.

Fig. 4.16 shows some examples of camera trajectory estimated by the three methods along with the ground truth. In this figure, the plots (a), (b), and (c) contain the trajectory obtained in sequence 0, where all methods have a similar performance. The plots (e), (f), and (g) show the results for sequence 6, where the method RLS-MVSLAM has the largest error, while the plots (h), (i), and (j) show the trajectory obtained for sequence 9, where the RLS-MVSLAM has the lowest error.

For sequence 0, even though the camera moves along a trajectory with several curves for both sides, the algorithms can estimate a trajectory for the camera that has the same shape as the ground truth. For the other sequences, at least one method shows a trajectory

that does not match the ground truth. One can associate this problem to a drift in the computation, as the trajectories present a shape similar to the ground truth.

In the algorithms, the trajectory is computed individually for different pairs of frames, whose results are transformed to match the other, and then incrementally updated. If the procedure of matching the results for different pairs of frames produces an error, it can be expected that the trajectory has a drift that increases over time.

For monocular systems, the drift is often corrected by a loop closure. Comparing, for instance, the plots (h) and (i) to (g) and the ground truth, one can see that the loop closure algorithm in (h) and (i) was not able to identify that the first and last positions of the trajectory should be close to it other. This behavior explains the difference in the order of magnitude of the error seen in Tab. 4.1.

Tabela 4.1 – Average trajectory error on the KITTI dataset. For each position i, the error $\|\text{trans}(\mathbf{E}_i)\|$ is computed. The table shows the minimum and maximum of this error, and the ATE (which is the root mean square error) defined in Eq. (4.72) for the trajectory obtained by each method.

	RLS-MVSLAM [1]			ORB-SLAM2 [12]			LDSO [15]		
	min	max	ATE	min	max	ATE	min	max	ATE
0	0.69	20.35	**7.57**	1.49	26.89	**10.19**	1.77	23.15	**10.95**
1	*	*	*	14.40	611.71	**333.95**	0.33	40.10	**9.55**
2	10.57	163.62	**62.35**	0.25	77.30	**23.20**	0.86	109.21	**25.91**
3	0.25	4.08	**1.17**	0.34	3.46	**1.97**	0.15	8.89	**2.99**
4	0.05	0.84	**0.37**	0.03	0.87	**0.39**	1.05	3.06	**1.22**
5	0.27	8.82	**5.28**	0.30	9.24	**3.99**	0.88	12.89	**4.90**
6	0.28	173.00	**93.45**	0.39	23.69	**15.35**	0.78	23.74	**13.62**
7	0.14	5.06	**2.71**	0.59	4.71	**2.33**	0.06	5.20	**2.43**
8	48.51	515.93	**188.24**	0.53	155.04	**51.17**	8.83	448.76	**128.57**
9	0.67	17.19	**9.35**	1.80	128.71	**64.09**	5.16	167.26	**75.90**
10	2.59	85.00	**32.19**	0.71	23.41	**7.63**	3.56	49.06	**17.52**

* for this case the algorithm was not able to converge.

4.7 Research Challenges

Despite being a topic with over 30 years of study,there are cases, with challenging environment or motion in which such algorithms fail [44]. In particular for monocular visual SLAM systems, the frames must have enough texture and consecutive frames must have a sufficient overlap for the algorithms to work properly [45]. In this section, we show that the videos acquired using the DORIS system do not always satisfy those requirements, presenting a new challenging scenario with several restrictions for the SLAM algorithm, that should be adopted as another benchmark to foster the development of more robust SLAM algorithms.

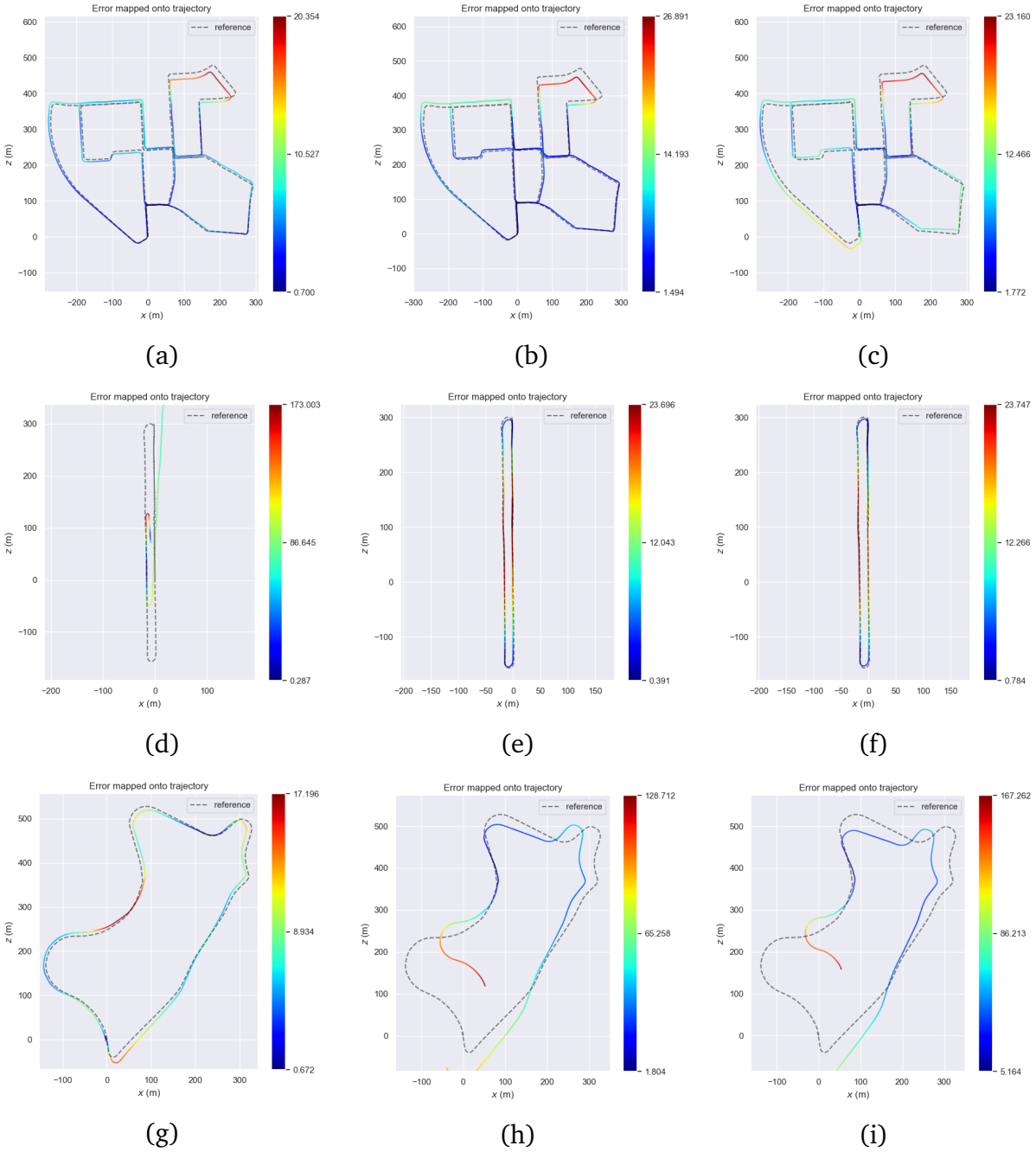
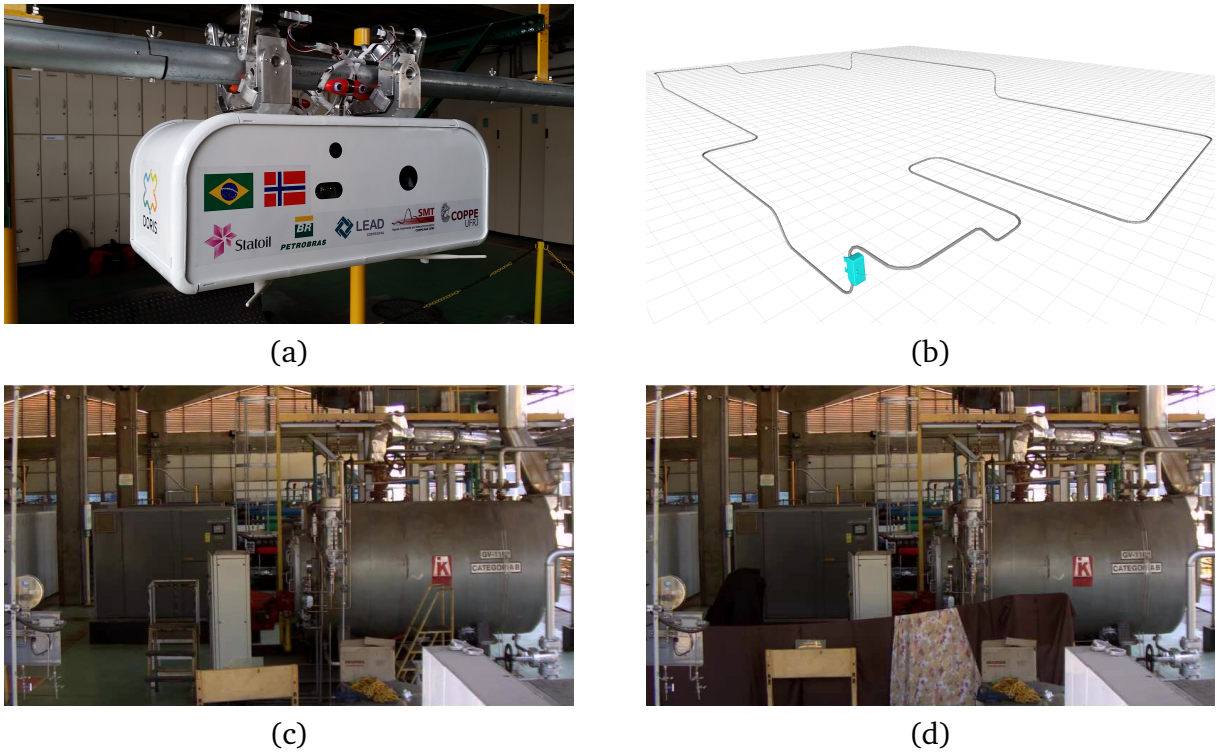


Figura 4.16 – Camera trajectory estimation for some videos of the KITTI odometry dataset. Each graph shows a view of the estimated trajectory (in solid lines) aligned with the ground truth (in dashed lines). For each position i, the error $\|\text{trans}(\mathbf{E}_i)\|$ is computed and the trajectory is colored by associating the values of the error to a heatmap. Sequence 0: a) RLS-MVSLAM. b) ORB-SLAM2. c) LDSO. Sequence 6: d) RLS-MVSLAM. e) ORB-SLAM2. f) LDSO. Sequence 9: g) RLS-MVSLAM. h) ORB-SLAM2. i) LDSO.

4.7.1 DORIS Surveillance System

DORIS - Monitoring Robots for Offshore Facilities is a project that endeavors to design and implement a surveillance system for remote supervision, diagnosis, and data acquisition on offshore facilities [46, 47, 48]. A robotic platform was installed in a industrial environment as shown in Fig. 4.17(a), and runs in a circular track whose model can be seen in Fig. 4.17(b). Several videos containing different objects were recorded, and the robotic platform also moved at different speeds. Examples of frames from this database can be seen in Figs. 4.17(c) and 4.17(d).



(a) (b)

(c) (d)

Figura 4.17 – DORIS surveillance system database. (a) Robotic platform in a industrial environment. (b) 3D model of the rail. (c) Frame from a reference video. (d) Frame from another video in the same position.

4.7.2 Tests with DORIS Videos

An attempt to estimate the trajectory for the DORIS videos revealed that the algorithms compared in Section 4.6 are unable estimate a trajectory. In these videos, an issue is that the camera sometimes passes near a pillar, which can be seen in Fig. 4.18. The frames obtained in the regions with a pillar have a flat surface that occupies most of the frame, which makes feature descriptor algorithms such as SURF and ORB not able to detect a sufficient number of keypoints. In addition, the descriptor for each keypoint is not distinctive since the image does not have a diversified content.

Hence, even if the algorithm finds a sufficient number of keypoints, the detected keypoints are not representative to describe the scene content. Consequently the procedure

of finding corresponding points, computing the epipolar geometry and estimating the camera displacements (see Section 4.4.2) becomes unreliable.

(a) (b)

Figura 4.18 – Example of frames from the DORIS videos with a flat surface occupying a significant portion of the frame. (a) Frame 7400. (b) Frame 12570.

One can try to ignore the regions with pillars and bypass the computation of the camera trajectory for these frames, for example, interpolating the displacement obtained using a frame before and a frame after the pillar. However, as can be seen from Fig. 4.19, the pillars may be so wide that there is almost no overlap in the scene before and after it, which makes the estimation of the camera trajectory unfeasible.

(a) (b)

Figura 4.19 – Example of frames from the DORIS videos showing the lack of overlap in the scenes before and after a pillar. (a) Frame obtained to the left of the pillar shown in Fig. 4.18(a). (b) Frame obtained to the right of the pillar shown in Fig. 4.18(a).

If we split each video sequence into several smaller sequences, removing the parts of the videos that are near a pillar, other problems arise. Even with the removal of the regions with pillars, several other textureless objects may occupy a large portion of the frames, due to the presence of large machinery in the industrial environment, which, as previously mentioned, deteriorates the results. Examples of the textureless objects contained in the scene can be seen in Fig. 4.20.

Another problem on the DORIS videos that can make the SLAM algorithm fail is related to the type of movement performed by the camera. In these videos, the camera moves along a direction that is perpendicular to the orientation of the camera, contrary, for example, to the videos in the KITTI dataset, which have the camera pointed to the front of a car. In this case, the viewpoints disappear much faster in the videos, which reduces the field of view in common between several views.

(a) (b)

Figura 4.20 – Example of frames from the DORIS videos containing large textureless objects. (a) Object in a sequence with curves in the rail. (b) Object in a sequence on a straight section of the rail.

4.8 Summary

This chapter described several concepts related to the estimation of the camera trajectory and mapping of the environment. It included an introduction to the field of epipolar geometry that studies the geometrical relation between multiple views recording the same scene, the three-dimensional points and their projection in the images. Since most algorithms require the computation of a matrix structure, a brief description of Lie algebra and optimization in a matrix space was presented, which allows the reader to better understand the reasoning behind the operations on the matrix space.

Three representative SLAM algorithms were depicted, with a greater focus on [1]. The methods were tested and evaluated on the KITTI dataset, a traditional dataset for the evaluation of odometry algorithms. The reader is invited to reproduce the results with the implementations referenced on this chapter.

Current challenges in visual SLAM were discussed by the end of the chapter. Characteristics that hinder the computation of the visual SLAM were exposed, along with an illustration of the occurrence of such characteristics in the DORIS videos. Such aspects reveal that the DORIS database present a new challenging scenario with several restrictions for the SLAM algorithm, that we believe can be used to stimulate the development of new algorithms.

Referências Bibliográficas

[1] G. Bourmaud and R. Mégret. Robust large scale monocular visual SLAM. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1638–1647, Boston, USA, June 2015.

[2] J. A. Castellanos, J. M. M. Montiel, J. Neira, and J. D. Tardos. The SPmap: a probabilistic framework for simultaneous localization and map building. 15:948–953, 1999.

[3] F. Endres, J. Hess, N. Engelhard, J. Sturm, D. Cremers, and W. Burgard. An

evaluation of the RGB-D SLAM system. In *IEEE International Conference on Robotics and Automation*, pages 1691–1696, Saint Paul, USA, 2012.

[4] B. Steder, G. Grisetti, C. Stachniss, and W. Burgard. Visual SLAM for flying vehicles. In *IEEE Transactions on Robotics*, volume 24, pages 1088–1093, Oct. 2008.

[5] A. J. Davison. Real-time simultaneous localisation and mapping with a single camera. In *IEEE International Conference on Computer Vision*, pages 1403–1411, Washington, USA, 2003.

[6] R. F. Salas-Moreno, R. A. Newcombe, H. Strasdat, P. H. J. Kelly, and A. J. Davison. SLAM++: simultaneous localisation and mapping at the level of objects. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1352–1359, Portland, USA, 2013.

[7] K. Konolige and M. Agrawal. FrameSLAM: from bundle adjustment to real-time visual mapping. *IEEE Transactions on Robotics*, 24(5):1066–1077, Oct. 2008.

[8] L. Clemente, A. Davison, I. Reid, J. Neira, and J. Tardós. Mapping large loops with a single hand-held camera. In *Robotics: Science and Systems*, Atlanta, USA, June 2007.

[9] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon. Bundle adjustment - a modern synthesis. In *International Workshop on Vision Algorithms: Theory and Practice*, pages 298–372, London, UK, 2000.

[10] J. Engel, T. Schöps, and D. Cremers. LSD-SLAM: large-scale direct monocular SLAM. In *European Conference on Computer Vision*, pages 834–849, Zurich, Switzerland, Sept. 2014.

[11] M. Klopschitz, C. Zach, A. Irschara, and D. Schmalstieg. Generalized detection and merging of loop closures for video sequences. In *International Symposium on 3D Data Processing, Visualization and Transmission*, Atlanta, USA, Sept. 2008.

[12] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós. ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE Transactions on Robotics*, 31(5):1147–1163, 2015.

[13] R. Mur-Artal and J. D. Tardós. ORB-SLAM2: an open-source SLAM system for monocular, stereo and RGB-D cameras. *IEEE Transactions on Robotics*, 33(5):1255–1262, 2017.

[14] J. Engel, V. Koltun, and D. Cremers. Direct sparse odometry. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(3):611–625, March 2018.

[15] X. Gao, R. Wang, N. Demmel, and D. Cremers. LDSO: Direct sparse odometry with loop closure. In *IEEE International Conference on Intelligent Robots and Systems*, pages 2198–2204, Madrid, Spain, Oct. 2018.

[16] G. S. Chirikjian. *Stochastic models, information theory, and Lie groups*. Springer-Verlag, 2nd edition, 2012.

[17] G. Xu and Z. Zhang. *Epipolar geometry in stereo, motion and object recognition*. Kluwer Academic Publishers, 1996.

[18] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge University Press, Cambridge, UK, 2nd edition, 2004.

[19] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[20] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (SURF). *Computer Vision and Image Understanding*, 110(3):346–359, June 2008.

[21] S. Leutenegger, M. Chli, and Y. Siegwart. BRISK: binary robust invariant scalable keypoints. In *IEEE International Conference on Computer Vision*, pages 2548–2555, 2011.

[22] A. Alahi, R. Ortiz, and P. Vandergheynst. FREAK: fast retina keypoint. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 510–517, Providence, USA, 2012.

[23] H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293(5828):133–135, Sept. 1981.

[24] Z. Zhang. Determining the epipolar geometry and its uncertainty: a review. *International Journal of Computer Vision*, 27(2):161–195, Apr. 1998.

[25] D. Nister. An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6):756–770, June 2004.

[26] Q.-T. Luong and T. Viéville. Canonical representations for the geometries of multiple projective views. *Computer Vision and Image Understanding*, 64(2):193–229, Sept. 1996.

[27] J. Faraut. *Analysis on Lie groups: an introduction*. Cambridge University Press, 2008.

[28] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *International Joint Conference on Artificial Intelligence*, Vancouver, Canada, 1981.

[29] C. Harris and M. Stephens. A combined corner and edge detection. In *Fourth Alvey Vision Conference*, pages 147–151, 1988.

[30] C. Olsson, A. Eriksson, and R. Hartley. Outlier removal using duality. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1450–1457, San Francisco, USA, June 2010.

[31] J. Sivic and A. Zisserman. Efficient visual search of videos cast as text retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(4):591–606, Apr. 2009.

[32] S. Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(4):376–380, Apr. 1991.

[33] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. ORB: An efficient alternative to SIFT or SURF. In *IEEE International Conference on Computer Vision*, pages 2564–2571, Barcelona, Spain, Nov. 2011.

[34] E. Rosten and T. Drummond. Machine learning for high-speed corner detection. In *European Conference on Computer Vision*, pages 430–443, Berlin, Germany, 2006.

[35] D. Galvez-López and J. D. Tardos. Bags of binary words for fast place recognition in image sequences. *IEEE Transactions on Robotics*, 28(5):1188–1197, Oct. 2012.

[36] B. K. P. Horn. Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America A*, 4(4):629–642, 1987.

[37] J. Engel, V. Usenko, and D. Cremers. A photometrically calibrated benchmark for monocular visual odometry. In *arXiv:1607.02555*, July 2016.

[38] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers. A benchmark for the evaluation of RGB-D SLAM systems. In *IEEE International Conference on Intelligent Robot Systems*, Algarve, Portugal, Oct. 2012.

[39] M. Grupp. evo: Python package for the evaluation of odometry and SLAM. `https://github.com/MichaelGrupp/evo`, 2017.

[40] G. Bourmaud. Robust large scale monocular visual SLAM. `https://www.dropbox.com/s/yglc1mlhxjosb6e/CVPR_2015_code.zip`, 2015.

[41] R. Mur-Artal. ORB-SLAM2. `https://github.com/raulmur/ORB_SLAM2`, 2015.

[42] N. Demmel. LDSO: Direct sparse odometry with loop closure. `https://github.com/tum-vision/LDSO`, 2018.

[43] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *IEEE Conference on Computer Vision and Pattern Recognition*, Providence, USA, June 2012.

[44] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on Robotics*, 32(6):1309–1332, Dec. 2016.

[45] D. Scaramuzza and F. Fraundorfer. Visual odometry [tutorial]. *IEEE Robotics Automation Magazine*, 18(4):80–92, Dec. 2011.

[46] G. P. S. de Carvalho, G. M. Freitas, R. R. Costa, G H. F. de Carvalho, J. F. L. de Oliveira, S. L. Netto, E. A. B. da Silva, M. F. S. Xaud, L. Hsu, G. Motta-Ribeiro, A. F. Neves, F. C. Lizarralde, I. Marcovistz, A. J. Peixoto, E. V. L. Nunes, P. J. From, M. Galassi, and A. Røyrøy. DORIS - monitoring robot for offshore facilities. In *Offshore Technology Conference*, Rio de Janeiro, Brazil, Oct. 2013.

[47] M. Galassi, A. Røyrøy, G. P. S. de Carvalho, G. M. Freitas, P. J. From, R. R. Costa, F. C. Lizarralde, L. Hsu, G H. F. de Carvalho, J. F. L. de Oliveira, A. A. de Lima, T. M. Prego, S. L. Netto, and E. A. B. da Silva. DORIS - a mobile robot for inspection and monitoring of offshore facilities. In *Anais do XX Congresso Brasileiro de Automática*, Belo Horizonte, Brazil, Sept. 2014.

[48] R. S. Freitas, M. F. S. Xaud, I. Marcovistz, A. F. Neves, R. O. Faria, G. P. S. de Carvalho, E. V. L. Nunes, A. J. Peixoto, F. C. Lizarralde, G. M. Freitas, R. R. Costa, M. Galassi, and P. W. J. Derks. The embedded electronics and software of DORIS offshore robot. In *IFAC Workshop on Automatic Control in Offshore Oil and Gas Production*, Florianópolis, Brazil, May 2015.

CAPÍTULO

5 Separação de Sinais e Análise de Variáveis Latentes: Fundamentos e Tendências

Leonardo Tomazeli Duarte (Faculdade de Ciências Aplicadas (FCA), Universidade Estadual de Campinas (UNICAMP)

Introdução

O problema de separação de sinais é, certamente, um dos mais desafiadores da área de processamento de sinais. As primeiras abordagens para separar sinais foram baseadas em técnicas clássicas de filtragem, impulsionadas sobretudo pelo surgimento de filtros digitais. O surgimento da filtragem adaptativa também foi um marco fundamental para a área de separação. Em particular, o cancelador adaptativo de ruído (ANC, do inglês *adaptive noise cancelling*), proposto em [1], pode ser considerado como uma das primeiras soluções que exploram a diversidade trazida por dois sinais distintos — o ANC tem como entradas uma mistura que contém o sinal de interesse e um sinal que traz informações sobre o sinal interferente observado na mistura.

A configuração padrão do filtro ANC foi generalizada na década de 1980, período considerado inicial para uma vertente da separação de sinais que se popularizou com a denominação separação cega de fontes (BSS, do inglês *blind source separation*). O trabalho pioneiro de Hérault, Jutten e Ans [2] pode ser considerado um marco para área, uma vez que trouxe duas contribuições fundamentais. A primeira delas se refere à configuração utilizada em [2], que, de certo modo, estende a configuração padrão do filtro ANC ao realizar o processo de separação de um conjunto de sinais fontes a partir de um conjunto de misturas. Desde então, essa configuração de múltiplas entradas e múltiplas saídas vem permeando praticamente todas as soluções não-supervisionadas em separação de sinais.

A segunda contribuição fundamental de [2], e talvez a mais disruptiva, diz respeito à incorporação de estatísticas de ordem superior no processo de separação. Tal abordagem, que se deu por meio do uso de correlações não-lineares como critérios de separação,

pode ser vista como uma extensão da clássica metodologia de estatística multivariada conhecida como análise de componentes principais (PCA, do inglês *principal component analysis*) [3]. De fato, enquanto que a PCA se vale de estatísticas de segunda ordem para extrair sinais descorrelacionados a partir de um conjunto de observações, a solução proposta em [2] introduziu um processo de separação no qual os sinais extraídos eram não-linearmente correlacionados, se aproximando assim do conceito de independência estatística. Tal paradigma viria a ser denominado posteriormente de análise de componentes independentes (ICA, do inglês *independent component analysis*) [4].

Outro trabalho seminal da área de BSS foi apresentado em 1994 por Pierre Comon [5], que formalizou o conceito de ICA. Nesta formalização, [5] mostrou que, num sistema linear e sem memória, a recuperação de um conjunto de fontes a partir de um conjunto de misturas pode ser feita por um processo de recuperação da independência estatística entre as estimativas das fontes, desde que as fontes originais sejam não-gaussianas e mutualmente independentes. O trabalho de Pierre Comon foi fundamental para fazer da ICA a abordagem mais utilizada na área de BSS, além de ter auxiliado na popularização dessa ferramenta como uma metodologia mais geral para análise de dados [6].

Num segundo momento dos estudos em BSS, buscou-se por soluções alternativas à ICA, sobretudo para lidar com situações nas quais a hipótese fundamental da ICA, a independência estatística entre as fontes, não é satisfeita. Isso pode ocorrer, por exemplo, na separação de sinais gerados por diferentes instrumentos musicais, uma vez que pode haver sincronismos entre tais instrumentos. A busca por essas alternativas se fez pela exploração de outros tipos de informações *a priori* sobre as fontes. Por exemplo, na análise de componentes esparsos (SCA, do inglês *sparse component analysis*) [4, 7], considera-se que as fontes podem ser representadas por um sinal esparso, ou seja, um sinal cujas amostras são, na maior parte do tempo, nulas ou próximas a zero. Um ponto relevante referente à exploração da esparsidade é que esta propriedade pode ser considerada em representações outras que a temporal, como, por exemplo, num domínio frequencial.

Outra abordagem que se consolidou na comunidade de BSS se fundamenta numa decomposição matricial conhecida como fatoração de matrizes não-negativas (NMF, do inglês *Non-negative Matrix Factorization*). A hipótese central da NMF é que os sinais fontes (e, eventualmente, os coeficientes do sistema misturador) sempre assumem valores não-negativos. Tal cenário é comum em diferentes campos de aplicação; por exemplo, na análise de sinais químicos, as fontes necessariamente assumem valores não-negativos, uma vez que elas geralmente representam concentrações (ou atividades) químicas [8]. A NMF pode ser realizada, dentre outras abordagens, a partir da formulação de um problema de inferência Bayesiana, capaz de levar em conta, além da propriedade de não-negatividade, outras características conhecidas de antemão sobre as fontes.

O breve panorama descrito nos permite concluir que há uma gama interessante de abordagens para lidar com o problema de BSS. No presente capítulo, as linhas gerais desse conjunto de abordagens serão apresentadas, de modo que o objetivo do texto é servir

como um documento introdutório sobre separação de fontes. Neste sentido, é importante salientar que o texto não é exaustivo — muitas abordagens relevantes em BSS não são discutidas. Além disso, por se tratar de um texto introdutório, consideraremos uma notação matemática simplificada. Os leitores interessados em se profundarem na área encontrarão material de grande relevância, por exemplo, nas seguintes referências: [4, 9]. Com relação à organização do capítulo, o texto se inicia, na Seção 5.1, com uma apresentação de um conjunto importante de abordagens utilizadas em BSS (ICA, SCA e NMF). Em seguida, na Seção 5.2, discutimos brevemente algumas tendências na área de BSS. O capítulo é concluído com algumas considerações finais, expostas na Seção 5.3.

5.1 Principais abordagens em separação

O problema de separação cega de fontes é ilustrado na Figura 5.1. Busca-se estimar um conjunto de N sinais fontes, representados pelo vetor $\mathbf{s}(t) = [s_1(t)\, s_2(t)\, \ldots s_N(t)]^T$, levando em conta a observação de um conjunto de M sinais, $\mathbf{x}(t) = [x_1(t)\, x_2(t)\, \ldots x_M(t)]^T$, que são obtidos a partir de um processo de mistura em $\mathbf{s}(t)$, dado por

$$\mathbf{x}(t) = \mathbf{F}(\mathbf{s}(t)), \tag{5.1}$$

onde $\mathbf{F} : \mathbb{R}^N \to \mathbb{R}^M$ representa, para cada amostra, o processo de mistura em questão.

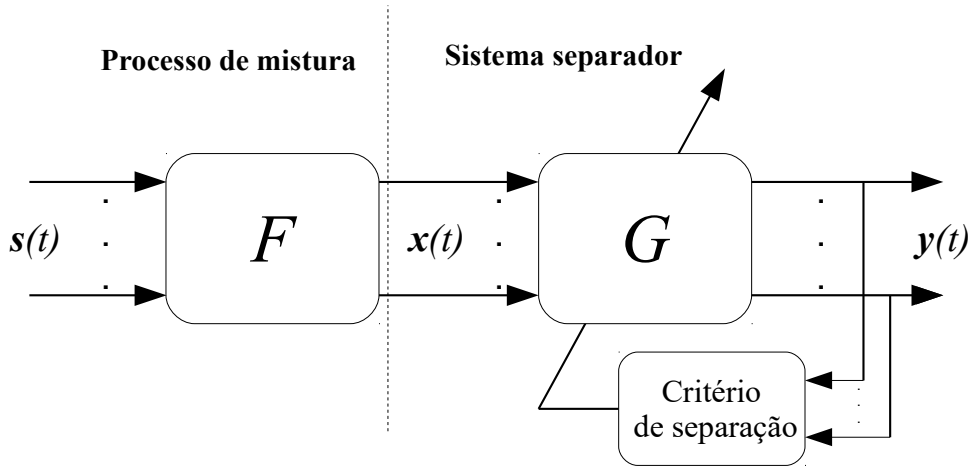


Figura 5.1 – O problema de separação de fontes.

O caráter não-supervisionado (cego) do problema de BSS diz respeito à ausência de amostras de treinamento (ou calibração), $\{\mathbf{s}(t), \mathbf{x}(t)\}$, e à ausência de informações detalhadas sobre o sistema misturador — de fato, considera-se apenas que há algumas informações sobre a natureza do modelo de mistura. Finalmente, os sinais representados pelo vetor $\mathbf{y}(t) = [y_1(t)\, y_2(t)\, \ldots y_N(t)]^T$ correspondem às estimativas das fontes fornecidas pelo método de separação, dadas por $\mathbf{y}(t) = \mathbf{G}(\mathbf{x}(t))$, onde $\mathbf{G} : \mathbb{R}^M \to \mathbb{R}^N$ representa a ação de um sistema separador.

Os estudos em BSS são categorizados de acordo com a natureza do sistema misturador. Por exemplo, as metodologias em BSS podem considerar sistemas misturadores *lineares* e *não-lineares*. Outra propriedade importante que se leva em conta é se o processo de mistura é *com memória* ou *sem memória*. Finalmente, em função dos números de fontes e misturas, o sistema misturador pode ser classificado como: i) determinado ($N = M$); ii) sobre-determinado ($N < M$); iii) sub-determinado ($N > M$). É interessante notar que, via de regra, modelos de mistura sobre-determinados são transformados em modelos determinados através da aplicação de técnicas de redução de dimensionalidade [6].

No presente trabalho, focaremos em modelos *lineares* e *sem memória*, uma vez que tal configuração é a mais usual em BSS. Neste cenário, o processo de mistura expresso em (5.1) se simplifica da seguinte maneira:

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t), \tag{5.2}$$

onde $\mathbf{A}$ é chamada de matriz de mistura. Neste caso, portanto, o problema de separação diz respeito à estimação dos sinais $\mathbf{s}(t)$ sem o conhecimento dos parâmetros da matriz $\mathbf{A}$. Via de regra, a maior parte das metodologias em BSS consideram o caso determinado. Todavia, algumas das estratégias que serão discutidas na sequência podem ser aplicadas também ao caso sub-determinado.

Análise de componentes independentes

Conforme discutido anteriormente, a gênese da ICA está diretamente relacionada ao problema de BSS. De fato, a ICA foi a primeira abordagem capaz de separar fontes de maneira não-supervisionada e ainda ocupa o posto de metodologia mais disseminada na área. Na ICA, considera-se uma modelagem probabilística do problema, de modo que as amostras de um dada fonte $s_i(t)$ são vistas com realizações de um variável aleatória. A hipótese central da ICA estabelece que as variáveis aleatórias que representam as fontes são mutualmente (estatisticamente) independentes [4]. Conforme mencionado anteriormente, há aplicações que não satisfazem tal condição de independência, e, logo, não podem ser abordadas por métodos de ICA.

Tendo em vista a hipótese de independência das fontes, a ICA se apoia no fato de que os sinais observados $\mathbf{x}(t)$ são mutualmente dependentes, pois as misturas correspondem a combinações lineares de um mesmo conjunto de variáveis aleatórias. Deste modo, a ideia central na ICA é ajustar um sistema separador tal que as estimativas fornecidas $\mathbf{y}(t)$ sejam, novamente, independentes. Ou seja, na ICA, a separação se dá por um processo de recuperação da independência estatística [4].

Matematicamente, o processo de recuperação presente na ICA pode ser expresso por um problema de otimização. No caso de um modelo de mistura linear, determinado e sem memória, o processo de otimização busca ajustar uma matriz de separação $\mathbf{W} \in \mathbb{R}^{N \times N}$, tal que $\mathbf{y}(t) = \mathbf{W}\mathbf{x}(t)$, de modo a minimizar uma função objetivo $J(\mathbf{y}(t))$ cujo valor mínimo é

atingindo para sinais estatisticamente independentes, ou seja:

$$\mathbf{W} = \underset{\tilde{\mathbf{W}}}{\operatorname{argmin}}\ J(\tilde{\mathbf{W}}\mathbf{x}(t)). \tag{5.3}$$

Uma escolha natural para a função $J(\cdot)$ se origina da divergência de Kullback-Leibler [10], que pode ser entendida com uma medida de dissimilaridade entre distribuições de probabilidade. No caso da ICA, considera-se a divergência de Kullback-Leibler entre a distribuição conjunta de $\mathbf{y}(t)$, representada por $f_{\mathbf{y}}(\mathbf{y})$, e o produto das distribuições marginais de cada estimativa $y_i(t)$, representadas por $f_{y_i}(y_i)$, dada por:

$$D\left(f_{\mathbf{y}}(\mathbf{y}), \prod_{i=1}^{N} f_{y_i}(y_i)\right) = \int f_{\mathbf{y}}(\mathbf{y}(n)) \log\left(\frac{f_{\mathbf{y}}(\mathbf{y})}{\prod_{i=1}^{N} f_{y_i}(y_i)}\right) d\mathbf{y}. \tag{5.4}$$

Dado que independência estatística é caracterizada pela seguinte condição:

$$f_{\mathbf{y}}(\mathbf{y}) = \prod_{i=1}^{N} f_{y_i}(y_i),$$

a divergência expressa em (5.4) será sempre não-negativa, de modo que se anulará somente quando as variáveis y_i forem estatisticamente independentes.

É possível mostrar que (5.4) também corresponde à informação mútua entre os elementos de $\mathbf{y}(t)$, representada por $I(\mathbf{y})$, ou seja:

$$D\left(f_{\mathbf{y}}(\mathbf{y}), \prod_{i=1}^{N} f_{y_i}(y_i)\right) = I(\mathbf{y}) = \sum_{i=1}^{N} H(y_i) - H(\mathbf{y}), \tag{5.5}$$

onde $H(\cdot)$ representa a entropia diferencial [10]. Assim, um dos principais paradigmas em ICA visa a minimização da informação mútua entre os sinais estimados, isto é:

$$\mathbf{W} = \underset{\tilde{\mathbf{W}}}{\operatorname{argmin}} \sum_{i=1}^{N} H(y_i) - H(\mathbf{y}). \tag{5.6}$$

A minimização desta função objetivo pode ser feita por um método iterativo baseado no gradiente descendente. A regra de atualização neste caso é dada por [4]:

$$\mathbf{W}^{(k+1)} = \mathbf{W}^{(k)} - \mu\left(E\{\Psi_{\mathbf{y}}(\mathbf{y})\mathbf{x}^T\} - \mathbf{W}^{-T}\right), \tag{5.7}$$

onde

$$\psi_{y_i}(y_i) = -\frac{d \log f_{y_i}(y_i)}{d y_i}$$

é conhecida como função escore de y_i.

Há duas questões centrais relacionadas ao paradigma expresso em (5.6). A primeira delas, de natureza mais teórica, refere-se às condições que devem ser observadas para que a recuperação de componentes independentes implique na separação das fontes. Ou seja, em

quais situações a ICA separa as fontes? A formalização dessa questão foi feita em [5], que, a partir do Teorema de Darmois-Skitovich, mostrou que aplicação da ICA para separação das fontes requer que: *i*) a matriz **A** seja de posto completo; *ii*) Exista, no máximo, uma fonte Gaussiana; *iii*) as fontes devem ser variáveis aleatórias mutualmente independentes. Além disso, mesmo quando tais condições são satisfeitas, o processo de separação apresenta as ambiguidades de escala e de permutação [5]. Em outras palavras, não é possível recuperar a escala e a ordem original das fontes. Via de regra, tais ambiguidades não limitam a aplicação prática da ICA, pois, na maioria das aplicações, o objetivo principal é recuperar a forma dos sinais fontes.

Uma segunda questão relevante associada à formulação (5.6) é de natureza mais prática e diz respeito à iteração expressa em (5.7). De fato, essa regra de atualização exige a estimação da função escore, que, por sua vez, requer a estimação das distribuições de probabilidade das estimativas das fontes, o que, geralmente, é custoso do ponto de vista computacional. Há uma série de outros paradigmas de ICA que podem ser entendidos como simplificações da informação mútua e que são capazes de realizar ICA. Por exemplo, na abordagem Infomax[1] [12], a regra de atualização é dada por (5.7), porém substituindo as funções escores por funções não-lineares fixas, e, logo, sem a necessidade de processos de estimação de distribuições de probabilidade. Curiosamente, a escolha das funções não-lineares no paradigma Infomax, ao menos no caso linear, não requer um procedimento refinado, dado que o algoritmo resultante é capaz de separar as fontes mesmo quando tais funções são diferentes das funções escores das fontes originais [13].

Ainda no contexto de simplicação da busca pela independência estatística na abordagem ICA, cabe mencionar os métodos baseados na minimização da não-gaussianidade [6]. Esta metodologia se fundamenta no fato que, devido ao teorema central do limite [6], os sinais misturados são mais gaussianos do que os sinais fontes. Logo, a recuperação das fontes pode ser feita por um processo de maximização da não-gaussianidade dos sinais estimados. Tal processo pode ser formulado como problemas de otimização cujas funções objetivos são dadas por medidas de gaussianidade, como a curtose e a negentropia [6]. A maximização da não-gaussianidade é o paradgima adotado por um dos algoritmos mais utilizados em ICA: o FastICA [14].

Análise de componentes esparsos

A revolução esparsa em processamento de sinais se intensificou em meados da década de 2000, sobretudo devido aos trabalhos seminais de Emmanuel Candès e David Donoho no problema de sensoriamento comprimido [15, 16]. Em breves termos, um sinal esparso pode ser representado por um vetor cujos elementos são, em sua maioria, nulos ou próximos de zero. Um tipo natural de esparsidade pode ocorrer em sinais de fala, pois, em uma conversa, uma pessoa pode passar a maior parte do tempo em períodos de silêncio. Outro aspecto

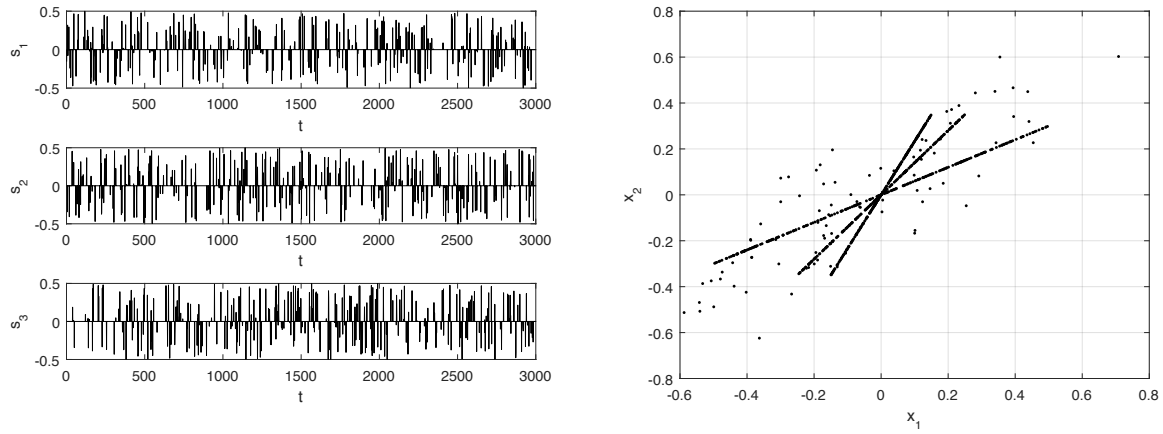
[1] O paradigma Infomax pode ser interpretado à luz do estimador de máxima verossimilhança para o modelo (5.2) [11].

interessante em processamento de sinais esparsos é que a propriedade de esparsidade pode ser explorada em representações outras que o tempo. Por exemplo, sinais de música, mesmo não sendo necessariamente esparsos no tempo, podem admitir uma representação esparsa após a aplicação de versões modificadas da transformada discreta do cosseno [17].

No contexto da BSS, as primeiras abordagens considerando a esparsidade das fontes foram propostas para tratar do caso de modelos de mistura sub-determinados [18, 19]. Tal abordagem pode ser entendida a partir da Figura 5.2, que ilustra um problema de separação de $N = 3$ fontes a partir de $M = 2$ misturas. Assumindo que os sinais são esparsos, o gráfico de dispersão bi-dimensional das misturas assume uma forma característica na qual há uma concentração de amostras em torno das direções definidas pelas coluna da matriz de mistura **A**. De fato, nas situações em que apenas uma das fontes está ativa (por exemplo, a fonte $s_1(t)$, de modo que $s_2(t) = s_3(t) = 0$), a mistura observada é proporcional à primeira coluna de **A**, ou seja

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} = \begin{bmatrix} s_1(t) \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} s_1(t)a_{11} \\ s_1(t)a_{21} \end{bmatrix}.$$

Logo, nos momentos de silêncio de $s_2(t)$ e $s_3(t)$, as misturas observadas $\mathbf{x}(t)$ são dadas por versões escalonadas (por $s_1(t)$) da primeira coluna da matriz de mistura.



(a) Fontes esparsas.

(b) Gráfico de dispersão bi-dimensional das duas misturas.

Figura 5.2 – Exemplo considerando $M = 2$ misturas de $N = 3$ de fontes esparsas. Devido à esparsidade das fontes, as misturas se agrupam em torno das direções dadas pelas colunas da matriz de mistura.

O agrupamento de fontes esparsas em torno das colunas da matriz de mistura motivou algumas abordagens para o problema de separação em modelos sub-determinados. Em [18], os autores propuseram um métodos baseado em duas etapas, a saber:

1. Estimação da matriz de mistura a partir de um procedimento de agrupamento para estimação das colunas de **A**;

2. De posse de uma estimativa de **A**, resolução do sistema linear $\mathbf{x(t)} = \mathbf{As(t)}$. Tal etapa, que é conduzida levando-se em conta uma restrição que impõe esparsidade nas fontes,

é similar ao problema inverso que surge em sensoriamento comprimido [15], e, logo, pode ser abordado por uma ampla de métodos atualmente disponíveis.

Esse procedimento de duas etapas requer, naturalmente, que as fontes sejam esparsas em um certo domínio.

A propriedade de esparsidade também pode ser utilizada no contexto de BSS em modelos determinados. Em [20], por exemplo, foi proposto um método de separação baseado na extração de fontes esparsas. Para descrever essa abordagem, consideremos uma notação matricial na qual todas as fontes são representadas pela seguinte matriz:

$$\mathbf{S} = \begin{bmatrix} \mathbf{s}_1^T \\ \mathbf{s}_2^T \\ \vdots \\ \mathbf{s}_N^T \end{bmatrix} = \begin{bmatrix} s_1(1) & s_1(2) & \dots & s_1(T) \\ s_2(1) & s_2(2) & \dots & s_2(T) \\ \vdots & \vdots & \ddots & \vdots \\ s_N(1) & s_N(2) & \dots & s_N(T) \end{bmatrix},$$

onde T é o número total de amostras das fontes. Da mesma forma, os sinais misturados podem ser representados por uma matriz $\mathbf{X}$, dada por:

$$\mathbf{X} = \mathbf{AS}, \tag{5.8}$$

onde $\mathbf{A}$ é a matriz de mistura.

No problema de extração de uma fonte, busca-se ajustar um vetor de extração $\mathbf{w}$, de modo que o sinal dado por

$$\mathbf{y}^T = \mathbf{w}^T\mathbf{X}$$

forneça uma estimativa de uma fonte. Em [20], o ajuste de $\mathbf{w}$ buscou recuperar a fonte mais esparsa no sentido da pseudo-norma L_0, que é dada pelo número de elementos não-nulos de um vetor. Matematicamente, tal abordagem pode ser expressa pelo seguinte problema de otimização:

$$\mathbf{w} = \underset{\tilde{\mathbf{w}}}{\operatorname{argmin}} \ ||\mathbf{y}||_0, \tag{5.9}$$

onde $||\mathbf{y}||_0$ representa a pseudo-norma L_0 do sinal extraído $\mathbf{y}$. Para evitar uma solução trivial, é necessário adicionar em (5.9) uma restrição associada ao vetor $\mathbf{w}$, como, por exemplo, $||\mathbf{w}||_2 = 1$.

Para o caso de $N = 2$ fontes (com$||\mathbf{s}_1||_0 < \mathbf{s}_2||_0$), [20] mostrou que uma condição suficiente para que a resolução do problema de otimização expresso em (5.9) leve à extração do componente mais esparso é dada por:

$$||\mathbf{s}_1||_0 < \frac{||\mathbf{s}_2||_0}{2}. \tag{5.10}$$

Em outras palavras, é possível extrair a fonte mais esparsa quando o seu nível de esparsidade (no sentido da pseudo-norma L_0) é no máximo metade do nível de esparsidade da outra fonte.

Ainda sobre a condição (5.10), cabe destacar que ela não se apoia na independência estatística entre as fontes. Ou seja, é possível separar fontes dependentes se houver uma diferença do nível de esparsidade entre elas. Finalmente, uma última observação é que a condição (5.10) pode ser estendida para problemas com mais de duas fontes [20]. Além disso, em [21], as condições necessárias para separação de fontes esparsas foram derivadas tanto para modelos instantâneos quanto para modelos convolutivos.

Fatoração em matrizes não-negativas

Em muitas situações práticas, as fontes e os coeficientes de mistura assumem, necessariamente, valores não-negativos. O caso prático mais emblemático dessa situação se encontra em química. De fato, as fontes nessa área geralmente representam concentrações ou atividades químicas, que são grandezas não-negativas [8]. Tal característica foi uma das motivações para o surgimento na década de 1970 da abordagem de fatoração em matrizes não-negativas (NMF) — na área de quimiometria, a NMF também é chamada de resolução multivariada de curvas (MCR, do inglês *multivariate curve resolution*) [22]. Na área de processamento de sinais e aprendizado de máquina, a NMF se popularizou após o trabalho de Lee e Seung [23]; desde então, tal ferramenta vem sendo utilizada em diferentes aplicações [24].

Matematicamente, uma possível versão da NMF pode ser formulada pelo seguinte problema de otimização

$$\min_{\mathbf{A},\mathbf{S}} \quad ||\mathbf{X}-\mathbf{AS}||_F^2, \text{ s} \tag{5.11a}$$

$$\text{sujeito a} \quad \mathbf{A},\mathbf{S} \geq 0, \tag{5.11b}$$

onde $||\cdot||_F$ representa a norma de Frobenius de uma matriz. Em outras palavras, a NMF corresponde a um problema inverso bilinear, no qual os termos do modelo de mistura são, necessariamente, não-negativos. É interessante notar que algumas variações da NMF consideram outras medidas de distâncias ou divergência alternativas à norma de Frobenius [24]. Neste sentido, um exemplo comum é o uso da divergência de Kullback-Leibler entre as observações e a representação dada pelo modelo.

Dentre as diferentes estratégias de otimização no contexto da NMF, uma abordagem popular é a formulação de um problema de quadrados mínimos alternados, matematicamente dado por:

$$\hat{\mathbf{S}}^{(k)} = \underset{\mathbf{S}}{\operatorname{argmin}} ||\mathbf{X}-\hat{\mathbf{A}}^{(k-1)}\mathbf{S}||_F^2, \text{ sujeito a } \mathbf{S} \geq 0. \tag{5.12}$$

$$\hat{\mathbf{A}}^{(k)} = \underset{\mathbf{A}}{\operatorname{argmin}} ||\mathbf{X}-\mathbf{A}\hat{\mathbf{S}}^{(k)}||_F^2, \text{ sujeito a } \mathbf{A} \geq 0. \tag{5.13}$$

Esse processo iterativo simplifica a abordagem do problema, pois, em cada um dos passos, basta resolver um problema de regressão supervisionada com uma restrição de não-

negatividade. Ao término de sua execução, esse algoritmo fornece uma estimativa das fontes e uma estimativa da matriz de mistura.

Assim como no caso de outras abordagens de BSS, uma questão relevante no contexto da NMF é se a recuperação de estimativas das fontes e dos coeficientes de misturas não-negativos implica, necessariamente, na separação das fontes. É possível mostrar [25] que, na formulação expressa em (5.11), a solução do problema não é única e que pode haver soluções que são capazes de prover uma boa representação dos dados observados porém que não correspondem às fontes originais. Diante dessa limitação, o uso de NMF em separação é feito, via de regra, com a adição de medidas de regularização. Por exemplo, uma escolha comum é considerar, como informação *a priori*, que as fontes, além de não-negativas, também são esparsas em alguma representação [26]. Outra estratégia comum é assumir que as fontes podem ser descritas por sinais suaves, que não apresentam grandes variações temporais [24].

5.2 Tendências na área

A BSS continua sendo um campo ativo de pesquisa na área de processamento de sinais, além de despertar interesse de outras comunidades (por exemplo, aprendizado de máquina). Nesta seção, apresentamos brevemente alguns tópicos mais recentes de pesquisa em BSS e que vêm ganhando destaque nos fóruns da área. Os detalhes desses assuntos podem ser consultados nas referências indicadas.

Um primeiro tópico que cabe mencionar diz respeito aos sistemas multimodais de aquisição da informação. Um exemplo de multimodalidade no contexto de BSS se encontra no problema de imageamento cerebral por meio de misturas provenientes de diferentes sistemas de aquisição, como, por exemplo, eletroencefalograma e a ressonância magnética funcional [27]. Um dos desafios neste caso é como explorar simultaneamente sinais distintos, que, via de regra, apresentam resoluções temporais e espaciais diferentes. A abordagem deste tipo de problema requer algum tipo de estratégia de fusão de dados, que pode ser realizada, por exemplo, via métodos de separação cega conjunta de fontes (JBSS, do inglês *joint blind source separation*) [28].

Outro tópico que desperta a atenção da comunidade é o uso de métodos tensorais em separação [29, 30]. Essencialmente, um tensor pode ser visto como uma generalização de uma matriz, pois pode apresentar mais de dois modos — um vetor pode ser visto como um tensor de um modo, e uma matriz pode ser vista como um tensor de dois modos. Umas das vantagens do uso de tensores em BSS é que as decomposições tensorais, como a decomposição CPD [29], apresentam aspectos interessantes de unicidade. O mais importante deles é que, via de regra, as condições de unicidade são menos restritivas se comparadas às restrições usualmente impostas em métodos convencionais de separação. Tal característica permite, a partir de um tensor de mistura, separar fontes difíceis de serem obtidas por métodos convencionais. O preço a ser pago é a necessidade de mais um tipo de

diversidade na aquisição dos dados — nos métodos convencionais, trabalha-se geralmente com apenas a diversidade espaço-temporal.

Em muitas situações práticas, há mais de uma informação *a priori* sobre os sinais fontes. Por exemplo, as fontes podem ser mutualmente independentes e, ao mesmo tempo, não-negativas. Motivados por tais casos, algumas pesquisas vêm buscando desenvolver métodos capazes de explorar simultaneamente uma coletânea de informações conhecidas sobre as fontes. A estratégia mais difundida nesse sentido é a formulação de um problema de inferência Bayesiana, no qual as informações das fontes são modeladas a partir de distribuições *a priori* de probabilidade. A abordagem Bayesiana [4] vem se mostrando eficaz em diversas aplicações práticas, geralmente difíceis de serem abordadas por métodos convencionais de BSS [31, 32].

Uma estratégia mais recente para lidar com múltiplas informações sobre as fontes considera uma formulação baseada em otimização multiobjetivo [33]. Nesta abordagem, busca-se otimizar mais de um critério de separação simultaneamente, de modo que a resposta fornecida ao usuário do método é um conjunto de soluções ditas não-dominadas. Assim, valendo-se, por exemplo, de sua experiência subjetiva sobre o problema, o usuário pode selecionar, dentre o conjunto de soluções não-dominadas, uma estimativa das fontes. Essa abordagem é interessante, portanto, em problemas nos quais o usuário do método pode contribuir diretamente no processo. Isso ocorre, por exemplo, na separação de sinais biomedicais, em geofísica, e em análise química.

Finalmente, um tópico que já vem sendo estudado desde a década de 1990, porém que ainda merece atenção dos pesquisadores, é a separação de fontes em modelos não-lineares. O desafio neste caso é que as condições de separabilidade geralmente observadas no caso linear não são válidas para o caso geral de misturas não-lineares. Uma revisão recente sobre o assunto pode ser encontrada em [34]. Além disso, cabe destacar que esse assunto foi recentemente revistado pela comunidade de aprendizado de máquina [35].

5.3 Conclusões

Este capítulo teve como objetivo prover uma primeira leitura aos leitores interessados na área de BSS. Como discutido ao longo do texto, há uma gama interessante de abordagens de separação, além de perspectivas desafiadoras de pesquisa na área. Além disso, o assunto é tratado por diferentes comunidades científicas, que incluem processamento de sinais, aprendizado de máquina, estatística aplicada e computação.

Cabe mencionar, por fim, que a generalidade da formulação do problema de BSS faz com os métodos de separação sejam aplicados em problemas reais de diferentes áreas. Uma importante área de aplicação é a separação de dados biomédicos, no contexto, por exemplo, de separação de sinais cardíacos e cerebrais. Também cabe destacar o grande interesse pelo problema de separação de sinais de áudio. Além dessas duas áreas, métodos de separação de fontes vêm sendo intensamente aplicados em controle de qualidade [36], telecomunicações,

análises químicas, sensoriamento remoto e imageamento sísmico. Um descritivo dessas aplicações pode ser encontrado em [4].

Referências Bibliográficas

[1] B. Widrow, J.R. Glover, J.M. McCool, J. Kaunitz, C.S. Williams, R.H. Hearn, J.R. Zeidler, Jr. Eugene Dong, and R.C. Goodlin. Adaptive noise cancelling: Principles and applications. *Proceedings of the IEEE*, 63(12):1692–1716, 1975.

[2] Jeanny Hérault, Christian Jutten, and Bernard Ans. Détection de grandeurs primitives dans un message composite par une architecture de calcul neuromimétique en apprentissage non supervisé. In *Xème colloque GRETSI*, pages 1017–1022, 1985.

[3] Ian Jolliffe. *Principal component analysis*. Wiley Online Library, 2002.

[4] P. Comon and C. Jutten, editors. *Handbook of blind source separation: independent component analysis and applications*. Academic Press, 2010.

[5] Pierre Comon. Independent component analysis, A new concept? *Signal Processing*, 36(3):287–314, 1994.

[6] A. Hyvarinen, J. Karhunen, and E. Oja. *Independent component analysis*. John Wiley & Sons, 2001.

[7] Yannick Deville. Sparse Component Analysis: A General Framework for Linear and Nonlinear Blind Source Separation and Mixture Identification. In *Blind Source Separation: Advances in Theory, Algorithms and Applications*, volume 1743, pages 151–196. 2014.

[8] Leonardo Tomazeli Duarte, Said Moussaoui, and Christian Jutten. Source Separation in Chemical Analysis : Recent achievements and perspectives. *IEEE Signal Processing Magazine*, 31(3):135–146, may 2014.

[9] J. M. T. Romano, R. R. F. Attux, C. C. Cavalcante, and R. Suyama. *Unsupervised signal processing: channel equalization and source separation*. CRC Press, 2011.

[10] T. M. Cover and J. A. Thomas. *Elements of information theory*. Wiley-Interscience, 1991.

[11] Jean François Cardoso. Infomax and maximum likelihood for blind source separation. *IEEE Signal Processing Letters*, 4(4):112–114, 1997.

[12] Anthony J. Bell and Terrence J. Sejnowski. An Information-Maximization Approach to Blind Separation and Blind Deconvolution. *Neural Computation*, 7(6):1129–1159, nov 1995.

[13] Te Won Lee, Mark Girolami, and Terrence J. Sejnowski. Independent component analysis using an extended infomax algorithm for mixed subgaussian and supergaussian sources. *Neural Computation*, 11(2):417–441, 1999.

[14] A. Hyvarinen. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Transactions on Neural Networks*, 10(3):626–634, may 1999.

[15] E.J. Candes and M.B. Wakin. An Introduction To Compressive Sampling. *IEEE Signal Processing Magazine*, 25(2):21–30, mar 2008.

[16] David L. Donoho. For most large underdetermined systems of linear equations the minimal l1-norm solution is also the sparsest solution. *Communications on Pure and Applied Mathematics*, 59(6):797–829, jun 2006.

[17] Mark D. Plumbley, Thomas Blumensath, Laurent Daudet, Rémi Gribonval, and Mike E. Davies. Sparse representations in audio and music: From coding to source separation. *Proceedings of the IEEE*, 98(6):995–1005, 2010.

[18] Pau Bofill and Michael Zibulevsky. Underdetermined blind source separation using sparse representations. *Signal Processing*, 81(11):2353–2362, nov 2001.

[19] Pando Georgiev, Fabian Theis, and Andrzej Cichocki. Blind source separation and sparse component analysis of overcomplete mixtures. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 5:493–496, 2004.

[20] Leonardo T Duarte, Ricardo Suyama, Romis Attux, Joao M. T. Romano, and Christian Jutten. Blind extraction of sparse components based on l0-norm minimization. In *2011 IEEE Statistical Signal Processing Workshop (SSP)*, pages 617–620, jun 2011.

[21] Yves-Marie Batany, Daniela Donno, Leonardo Tomazeli Duarte, Herve Chauris, Yannick Deville, and Joao Marcos Travassos Romano. A necessary and sufficient condition for the blind extraction of the sparsest source in convolutive mixtures. In *2016 24th European Signal Processing Conference (EUSIPCO)*, number 2, pages 1628–1632, aug 2016.

[22] William H. Lawton and Edward A. Sylvestre. Self Modeling Curve Resolution. *Technometrics*, 13(3):617–633, aug 1971.

[23] Daniel D. Lee and H. Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, oct 1999.

[24] A. Cichocki, R. Zdunek, A. H. Phan, and S. Amari. *Nonnegative matrix and tensor factorizations : applications to exploratory multiway data analysis and blind source separation*. John Wiley & Sons, 2009.

[25] S. Moussaoui, D. Brie, and J. Idier. Non-negative Source Separation: Range of Admissible Solutions and Conditions for the Uniqueness of the Solution. In *Proceedings. (ICASSP '05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, volume 5, pages 289–292, 2005.

[26] Patrik O Hoyer. Non-negative Matrix Factorization with Sparseness Constraints. *Journal of Machine Learning Research*, 5:1457–1469, 2004.

[27] Dana Lahat, Tulay Adali, and Christian Jutten. Multimodal Data Fusion: An Overview of Methods, Challenges, and Prospects. *Proceedings of the IEEE*, 103(9):1449–1477, 2015.

[28] Xun Chen, Z. Jane Wang, and Martin McKeown. Joint Blind Source Separation for Neurophysiological Data Analysis: Multiset and multimodal methods. *IEEE Signal Processing Magazine*, 33(3):86–107, may 2016.

[29] Andrzej Cichocki, Danilo Mandic, Lieven De Lathauwer, Guoxu Zhou, Qibin Zhao, Cesar Caiafa, and Huy Anh Phan. Tensor decompositions for signal processing applications: From two-way to multiway component analysis. *IEEE Signal Processing Magazine*, 32(2):145–163, 2015.

[30] Pierre Comon. Tensors : A brief introduction. *IEEE Signal Processing Magazine*, 31(3):44–53, may 2014.

[31] Leonardo Tomazeli Duarte, Christian Jutten, and Saïd Moussaoui. A Bayesian Nonlinear Source Separation Method for Smart Ion-Selective Electrode Arrays. *IEEE Sensors Journal*, 9(12):1763–1771, dec 2009.

[32] S. Moussaoui, H. Hauksdóttir, F. Schmidt, C. Jutten, J. Chanussot, D. Brie, S. Douté, and J.A. Benediktsson. On the decomposition of Mars hyperspectral data by ICA and Bayesian positive source separation. *Neurocomputing*, 71(10-12):2194–2208, jun 2008.

[33] Guilherme Dean Pelegrina, Romis Attux, and Leonardo Tomazeli Duarte. Application of multi-objective optimization to blind source separation. *Expert Systems with Applications*, 131:60–70, oct 2019.

[34] Yannick Deville and Leonardo Tomazeli Duarte. An Overview of Blind Source Separation Methods for Linear-Quadratic and Post-nonlinear Mixtures. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 9237, pages 155–167. 2015.

[35] Philemon Brakel and Yoshua Bengio. Learning Independent Features with Adversarial Nets for Non-linear ICA. pages 1–13, oct 2017.

[36] Guilherme Dean Pelegrina, Leonardo Tomazeli Duarte, and Christian Jutten. Blind source separation and feature extraction in concurrent control charts pattern recognition: Novel analyses and a comparison of different methods. *Computers & Industrial Engineering*, 92:105–114, feb 2016.

CAPÍTULO

6 Manual de Construção e Montagem do Cansat

Dionísio Fama Noque e Leonardo Dário Pinheiro
Gabinete de Gestão do Programa Espacial Nacional www.ggpen.gov.ao

6.1 Introdução

Após um longo período de exploração espacial por parte de alguns poucos países, universidades e instituições cientificas têm vindo a desenvolver a tecnologia de pequenos satélites, que possuem custos bastantes reduzidos comparando aos convencionais, permitindo que sejam desenvolvidos por qualquer instituição/país com menos recursos. Em termos gerais, pequenos satélites são qualquer satélite com peso inferior a 500kg [1]. Para classificação de alguns pequenos satélites, além do peso, deve-se também cumprir alguns critérios específicos como forma e dimensão, é o caso de Cubesats e Cansats (ver tabela 6.1).

Classificação	Massa (kg)
Minissatélite	100 - 500
Microssatélite	10 - 100
Nanossatélite	1 - 10
Picossatélite	0.1 - 10

Tabela 6.1 – Classificação de pequenos satélites.

Um cansat é uma representação de um satélite convencional, integrado no volume e na forma de um refrigerante, e se enquadra na classificação dos picossatélite. O primeiro Cansat Angolano, foi criado com base numa junção de sinergias entre o Gabinete de Gestão do programa Espacial Nacional (GGPEN) e a academia nacional. Foi construído com propósitos educativos, mediante o qual é possível adquirir experiência em "Desenho, Integração, Testes e Lançamento"munindo os formandos de conhecimentos sobre as funções, arquitectura e integração de subsistemas que compõe um satélite convencional. O cansat é então lançado a uma altitude de algumas centenas de metros por um foguete

ou largado a partir de um drone ou mediante um balão, e sua missão começa: realizar um experimento científico e conseguir um pouso seguro.

O desafio para os formandos é encaixar todos os principais subsistemas encontrados em um satélite, como alimentação eléctrica, comando e processamento de dados, comunicação e carga útil neste volume mínimo. Os cansats oferecem uma oportunidade única para os alunos terem uma primeira experiência prática de projectos espaciais reais. Eles são responsáveis por todos os aspectos: selecção da missão, projecção do cansat, integração dos componentes, programação do computador de bordo, teste, preparação do lançamento e analise os dados.

Neste manual veremos os procedimentos de montagem do Cansat Angolano.

6.2 Configuração Geral

Os satélites são normalmente considerados como um sistema subdividido em vários subsistemas. Esses subsistemas, por sua vez, podem ser um agrupamento de unidades (hardware) que realizam uma determinada função no satélite. Por exemplo, para o fornecimento de energia eléctrica no satélite, é usado o subsistema de alimentação eléctrica que agrupa todos os dispositivos electrónicos que geram, condicionam e distribuem a tensão para todo satélite.

O Cansat Angolano é composto por 5 subsistemas, como pode ser observado na Figura 6.1, onde é apresentado também a função de cada subsistema.

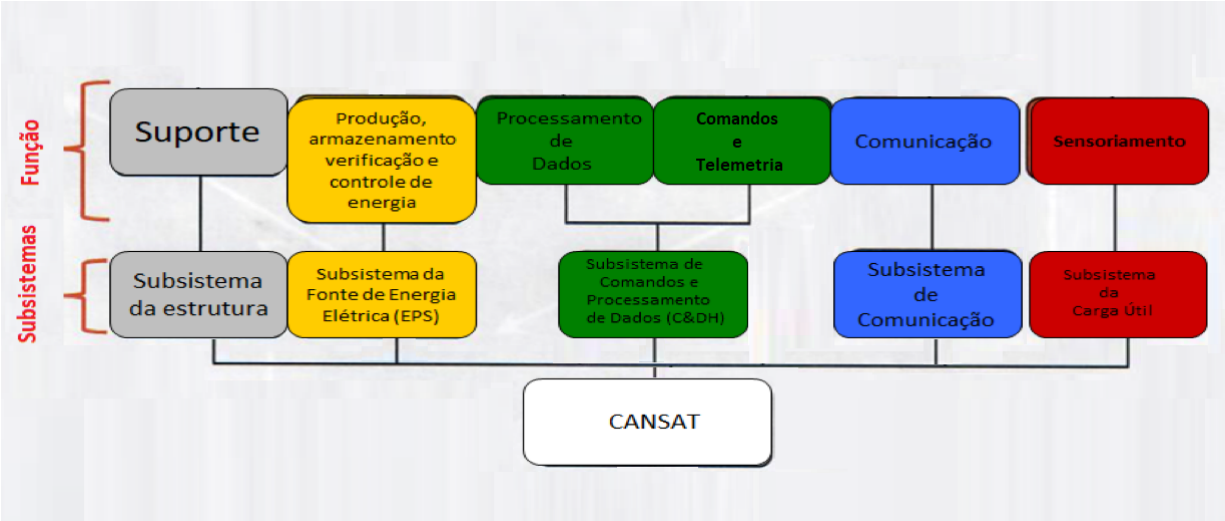


Figura 6.1 – Subsistemas do Cansat Angolano

Os 5 subsistemas estão subdivididos em 6 placas circulares, com as seguintes denominações:

- Placa GPS;
- Placa PWR (Alimentação Eléctrica);
- Placa USR (Usuário);

- Placa OBC (Computador de Bordo);
- Placa COM (Comunicação);
- Placa CAM (Câmara).

A Figura 6.2 apresenta a disposição dos elementos que compõe o cansat, dando ênfase as placas e suas unidades. As interfaces da cada uma das placas pode ser observado na Figura 6.3.

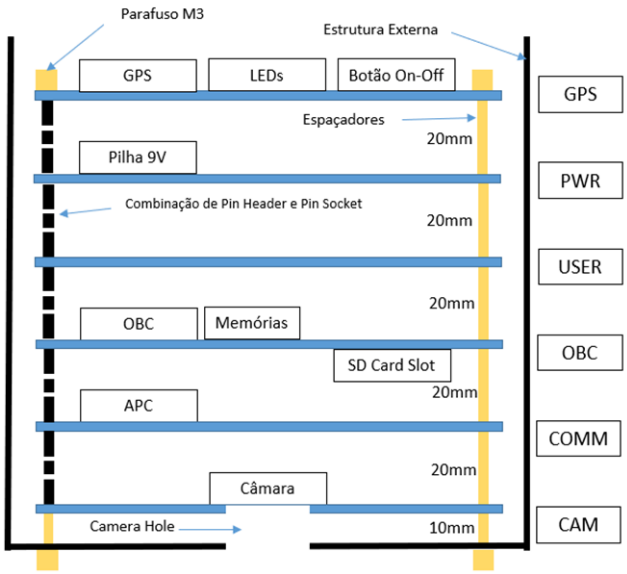


Figura 6.2 – Composição do Cansat Angolano

Cada placa tem duas faces. é chamada de superfície superior a face que contém a nomenclatura da placa (ex.: 2-PWR) e de superfície inferior a face que contém o logotipo do GGPEN (Figura 6.4). A conexão eléctrica entre as placas é feita por uma combinação de dois pin-sockets e um pin header duplo. As placas são mecanicamente fixadas usando espaçadores metálicos do tipo M3 de 10 e 20mm. A placa GPS é fixada por parafusos M3.

6.3 Descrição do Circuito Eléctrico

Nesta secção veremos em detalhes a constituição de cada placa.

6.3.1 Placa GPS

A placa GPS faz parte do subsistema da carga útil e é composta por um módulo GPS, um botão on-off, um led indicador de energia, dois leds de uso geral e um pin socket para ligação com a placa abaixo (placa PWR).

A Figura 6.5 representa o esquema eléctrico da Placa GPS e na Figura 6.6 é apresentado uma fotografia da parte superior e inferior da referida placa.

O conector *J1-GPS.PWR*, localizado na superfície inferior, faz a conexão da placa GPS com a placa PWR. O *BT-PWR* é o botão on-off que liga e desliga o circuito todo. Esse botão

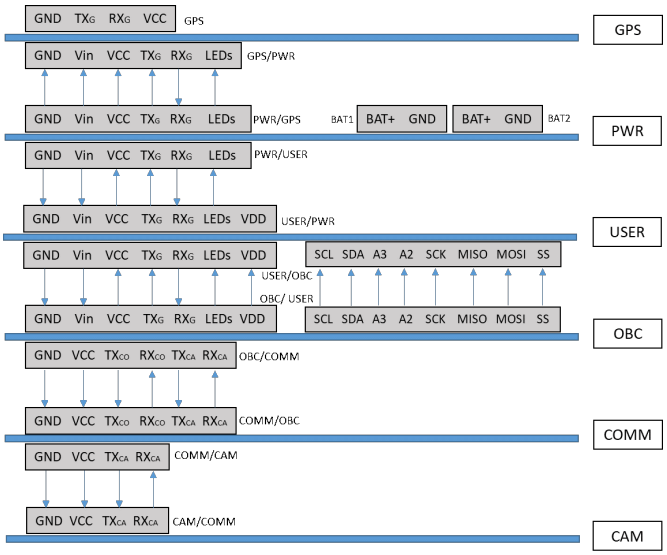


Figura 6.3 – interfaces do Cansat

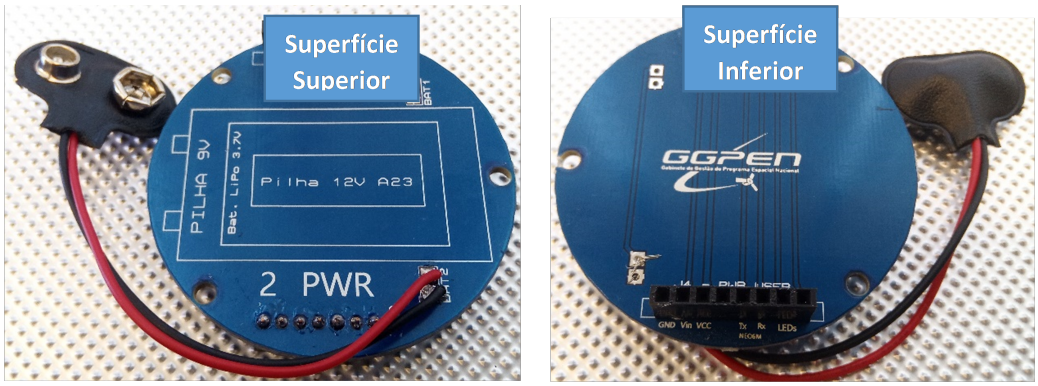


Figura 6.4 – Representação visual das placas

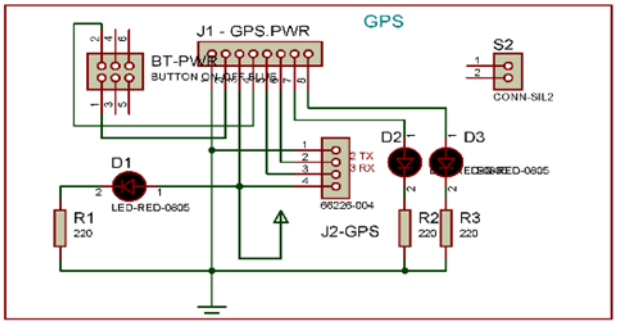


Figura 6.5 – Esquema eléctrico da Placa GPS

é o interruptor entre a pilha e o pino Vin do arduino. O conector *J2-GPS*, na superfície superior, serve para a conexão com o módulo NEO-6M. Esse módulo é também afixado por dois parafusos M3 ou pelo conector S2. O LED *D1* sinaliza que o circuito todo está alimentado (com 5V). Os LEDs *D2* e *D3* são de uso genérico, podendo ser programado pelo usuário. Comummente será usado *D2* para sinalizar a operação ou funcionamento do OBC. Os resistores *R1*, *R2* e *R3* são para a protecção dos LEDs.

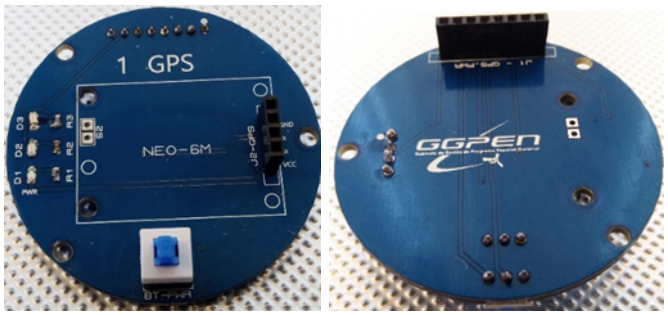


Figura 6.6 – Placa GPS, Superfície superior e inferior

6.3.2 Placa PWR

A Placa PWR faz parte do subsistema de Alimentação Eléctrica (EPS) e é composta por uma pilha de 9V que fornece tensão a todo circuito. Entretanto, o subsistema EPS é composto ainda pelos reguladores de tensão de 5 e 3.3V do arduino e pelo botão on-off.

É importante referir que serão adoptadas as denominações VCC para a tensão de 5V, VDD para a tensão de 3.3V, Vin para a tensão da pilha (9V) e GND para o terra (0V). É possível usar qualquer outra fonte acima de 7V. Por exemplo duas baterias de Lítio de 3.7V em série.

A Figura 6.7 representa o esquema eléctrico da Placa PWR e na Figura 6.8 é apresentado uma fotografia da parte superior e inferior da referida placa.

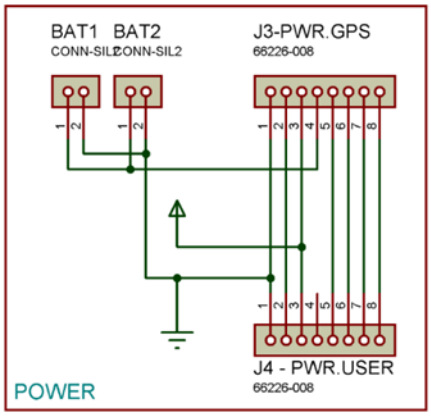


Figura 6.7 – Esquema eléctrico da Placa PWR

O conector *J3-PWR.GPS*, localizado na superfície superior, faz a conexão da placa PWR com a placa GPS. Os conectores *BAT1* e *BAT2* servem para a conexão da pilha. Para a utilização comum, é suficiente o uso de uma pilha, ligada a qualquer um dos conectores. O circuito pode funcionar com uma pilha de 9V, uma pilha A23 de 12V, ou baterias de lítio acima de 7V. O Conector *J4-PWR.USR*, na superfície inferior, faz a conexão da placa PWR com a placa USR.

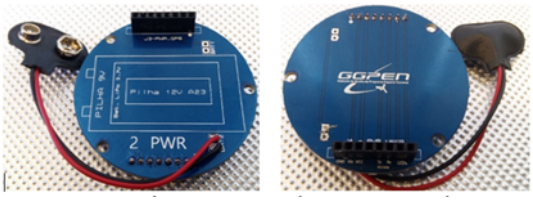


Figura 6.8 – Placa PWR, Superfície superior e inferior

6.3.3 Placa USR

A placa USR faz parte do subsistema da carga útil. Ela permite ao usuário conectar sensores de diferentes tipos compatíveis com o OBC possibilitando assim a realização de diversas missões. Na placa tem um pin socket que permite a condução dos sinais das placas GPS e PWR para a placa OBC. Tem ainda um outro pin socket que disponibiliza 4 pinos analógicos, 4 pinos digitais, comunicação SPI, comunicação I2C e comunicação UART por software. É importante lembrar que todos esses pinos são partilhados, ou seja, se usar a comunicação I2C, terá apenas mais dois pinos analógicos restantes. A Figura 6.9 ilustra o esquema eléctrico da Placa USR.

Na placa estão devidamente marcados os pinos partilhados. Na superfície superior estão marcados os pinos digitais e analógicos, na superfície inferior estão marcados os pinos de comunicação SPI e I2C, como pode ser observado na Figura 6.10.

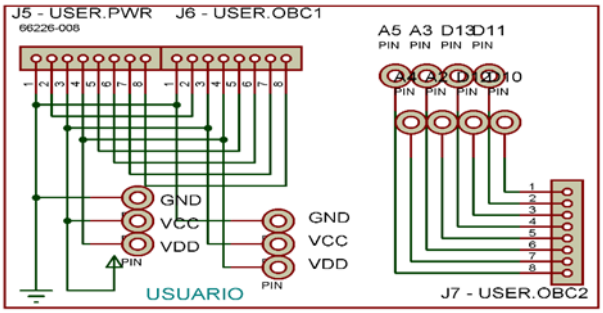


Figura 6.9 – Esquema eléctrico da Placa USR

O conector *J5-USER.PWR*, localizado na superfície superior, faz a conexão da placa USR com a placa PWR. Os Conector *J6* e *J7-USER.OBC*, na superfície inferior, fazem a conexão da placa USR com a placa OBC. Ainda na Placa USR podem ser conectados uma variedade de sensores compatíveis tais como GY-91, DHT11 ou MQ7.

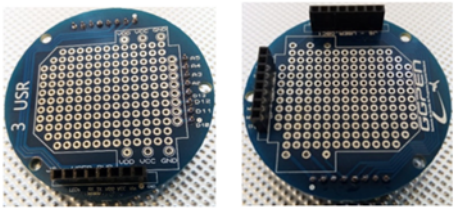


Figura 6.10 – Placa USR, Superfície superior e inferior

6.3.4 Placa OBC

A placa OBC faz parte do subsistema de Comando e Processamento de Dados. Ela é composta por um arduino nano. Na superfície inferior há também um módulo leitor de cartão micro SD que armazena as fotos da câmara.

A Figura 6.11 representa o esquema eléctrico da Placa OBC e na Figura 6.12 é apresentado uma fotografia da parte superior e inferior da referida placa.

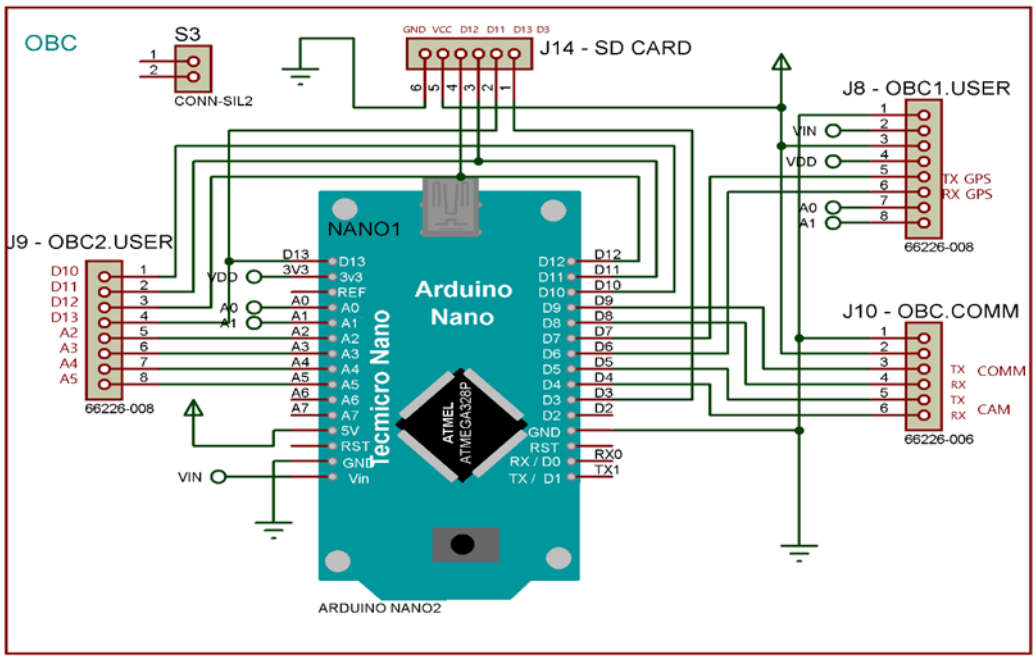


Figura 6.11 – Esquema eléctrico da Placa OBC

Os conectores *J8-OBC1.USER* e *J9-OBC2.USER*, localizados na superfície superior, fazem a conexão da placa OBC com a placa USR. O Conector *J10-OBC.COMM*, na superfície inferior, faz a conexão da placa OBC com a placa COM. O conector *J14-SD Card*, na superfície inferior, serve para a conexão do módulo leitor micro SD. O conector *S3* serve para fixação do módulo leitor micro SD. A placa possui ainda o socket (encaixe) para o arduino nano que tem a função de computador de bordo.

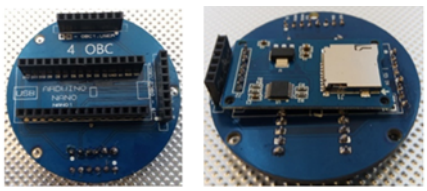


Figura 6.12 – Placa OBC, Superfície superior e inferior

6.3.5 Placa COM

A placa COM faz parte do subsistema de comunicação e possui um módulo transceptor APC220 com a capacidade de transmissão de até 1Km em campo aberto.

A Figura 6.13 representa o esquema eléctrico da Placa COM e na Figura 6.14 é apresentado uma fotografia da parte superior e inferior da referida placa.

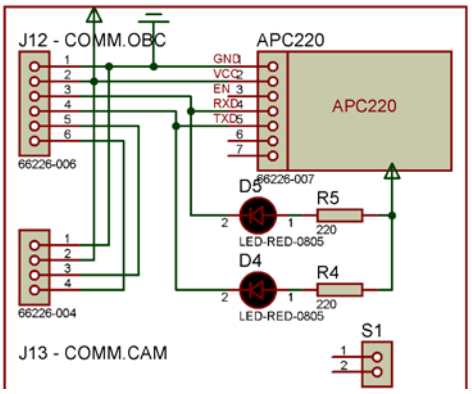


Figura 6.13 – Esquema eléctrico da Placa COM

O conector *J12-COMM.OBC*, localizado na superfície superior, faz a conexão da placa COM com a placa OBC. O Conector *J13-COMM.CAM*, na superfície inferior, faz a conexão da placa COM com a placa CAM. O módulo APC220 esta soldado directamente na placa COM. O conector *S1* serve de suporte para afixação do APC220.

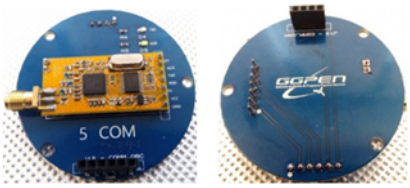


Figura 6.14 – Placa COM, Superfície superior e inferior

6.3.6 Placa CAM

A placa CAM faz parte do subsistema da Carga útil e é composta por uma câmara VC0706 (ou VC0703) cujas fotos são armazenadas no cartão micro SD. A organização das linhas de interface permite usar todos esses dispositivos sem problemas ou interferências.

A Figura 6.15 representa o esquema eléctrico da Placa CAM e na Figura 6.16 é apresentado uma fotografia da parte superior e inferior da referida placa..

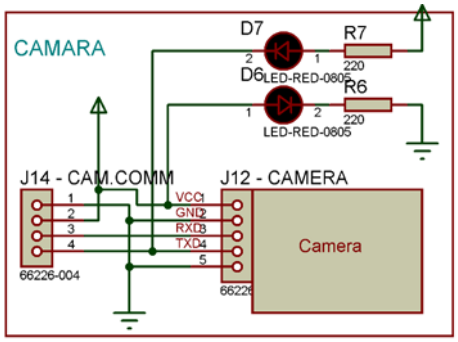


Figura 6.15 – Esquema eléctrico da Placa CAM

O conector *J14-CAM.COMM*, localizado na superfície superior, faz a conexão da placa CAM com a placa COM. O Conector *J12-CAMERA*, localizado também na superfície superior, serve para conexão da câmara VC0706. O LED *D6* serve de sinalização para os sinais de saída da câmara. O LED *D7* serve de sinalização de alimentação do circuito, dessa forma é possível ver que a placa está alimentada do topo ou da base. Os resistores *R6* e *R7* servem de protecção dos LEDs.

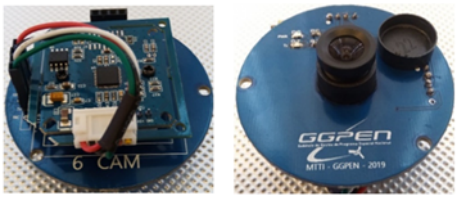


Figura 6.16 – Placa CAM, Superfície superior e inferior

6.4 Procedimentos de Montagem

Estando todas placas soldadas e preparadas, e com todos testes eléctricos feitos, o Cansat está pronto para ser montado. A montagem do cansat deve começar da placa GPS até a placa CAM. É possível também começar a montar da placa COM até a GPS, **mas iniciar o processo do meio tornará difícil a integração**. O conector da pilha de 9V não deve ser ligado antes de todos testes feitos, nem antes do envio do primeiro programa ao OBC.

O GGPEN, no seu programa de partilha de conhecimento em tecnologia e ciência espacial, desenvolveu um laboratório para experimentos práticos de montagem de cansats (kit cansat). O laboratório é constituído por um cansat, multímetro, chave, pen drive e todo componente electrónico necessário para a realização de missões predeterminadas. A Figura 6.17 ilustra a mala do kit cansat. Em a) é possível ver o cansat montado e em b) a disposição dos elementos que compõem o laboratório. Os componentes do Kit cansat podem ser verificados no anexo-1, e no anexo-2 tem-se a tabela de verificação do kit.

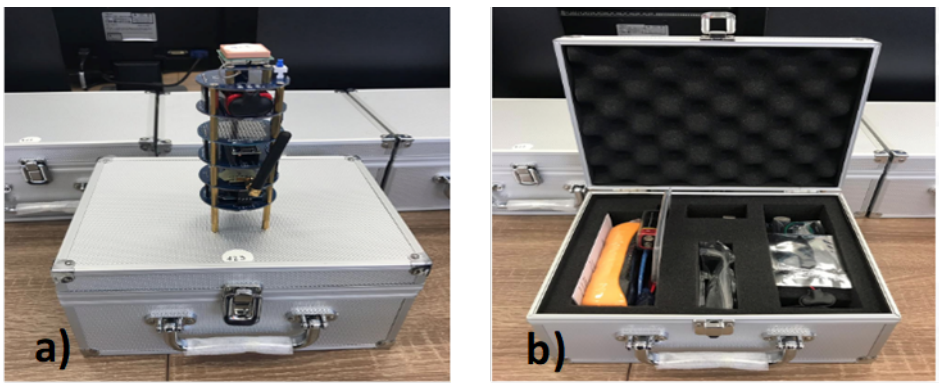


Figura 6.17 – a) Cansat montado; b) Mala do Kit Cansat

6.4.1 Placa GPS

A Figura 6.18 ilustra a sequência do procedimento de montagem da placa GPS.

1. Preparar os componentes da placa GPS;
2. Instalar o Modulo NEO-6M;
3. Instalar os espaçadores de 10mm com os parafusos, para fixação do GPS;
4. Use uma chave estrela para apertar os parafuso;
5. instalar os separadores de 20mm com um parafuso M3;
6. Resultado esperado.

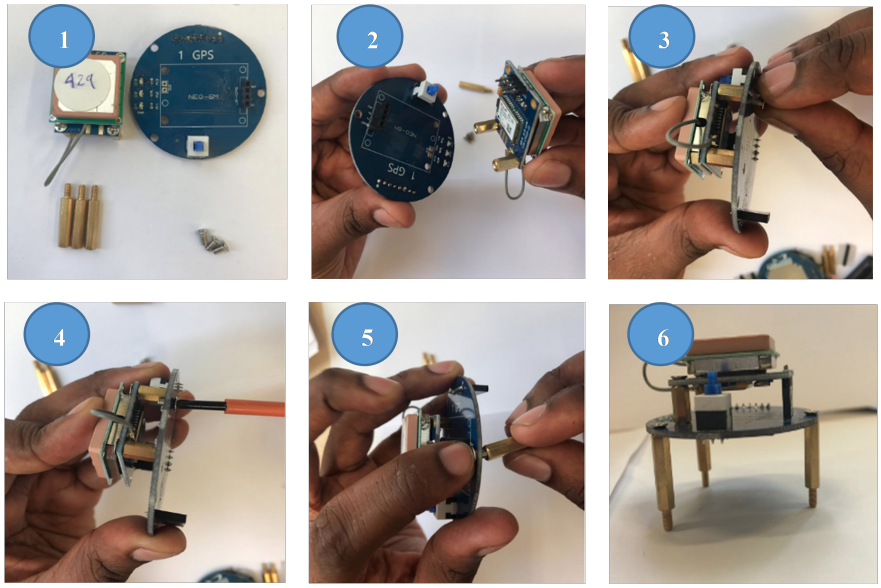


Figura 6.18 – Montagem da placa GPS

6.4.2 Placa PWR

A Figura 6.19 ilustra a sequência do procedimento de montagem da placa PWR.

1. Preparar os componente da placa PWR;
2. Instalar o pin header.
3. Aplicar a fita-cola dupla-face, para fixação da bateria;
4. Instalar a batéria;
5. Unir a placa GPS com a placa PWR;
6. Instalar os espaçadores de 20mm. Instalar um pin header de 8 pinos;
7. Resultado esperado.

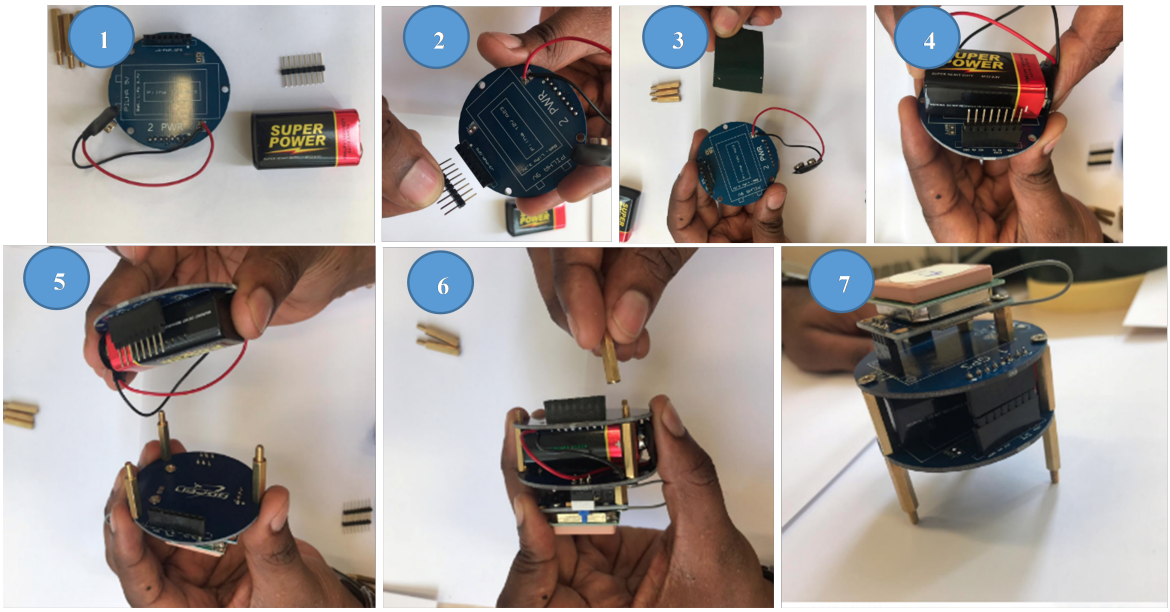


Figura 6.19 – Montagem da placa PWR

6.4.3 Placa USR

A Figura 6.20 ilustra a sequência do procedimento de montagem da placa USR.

1. Preparar os componentes da placa USR;
2. Instalar dois pin-headers de 8 pinos;
3. Unir a placa PWR com a placa USR;
4. Instalar os espaçadores de 20mm;
5. Resultado esperado.

Nota: Os componentes na placa USR podem variar dependendo da missão escolhida. Neste passo, já se deve ter a missão declarada e ter sido feito um estudo de que sensores ou actuadores serão incluidos na placa. Na placa tem pinos com interface SPI, I2C, pinos analógicos e digitais, tensão de 5 e 3.3V e ainda possibilidade de interface UART por software.

6.4.4 Placa OBC

A Figura 6.21 ilustra a sequência do procedimento de montagem da placa OBC.

1. Preparar os componentes da placa OBC;
2. Inserir o arduino nano, seguindo a orientação dos pinos. Se o arduino for inserido invertido, certamente queimará, podendo também danificar outros dispositivos;
3. Fixar devidamente o arduino;

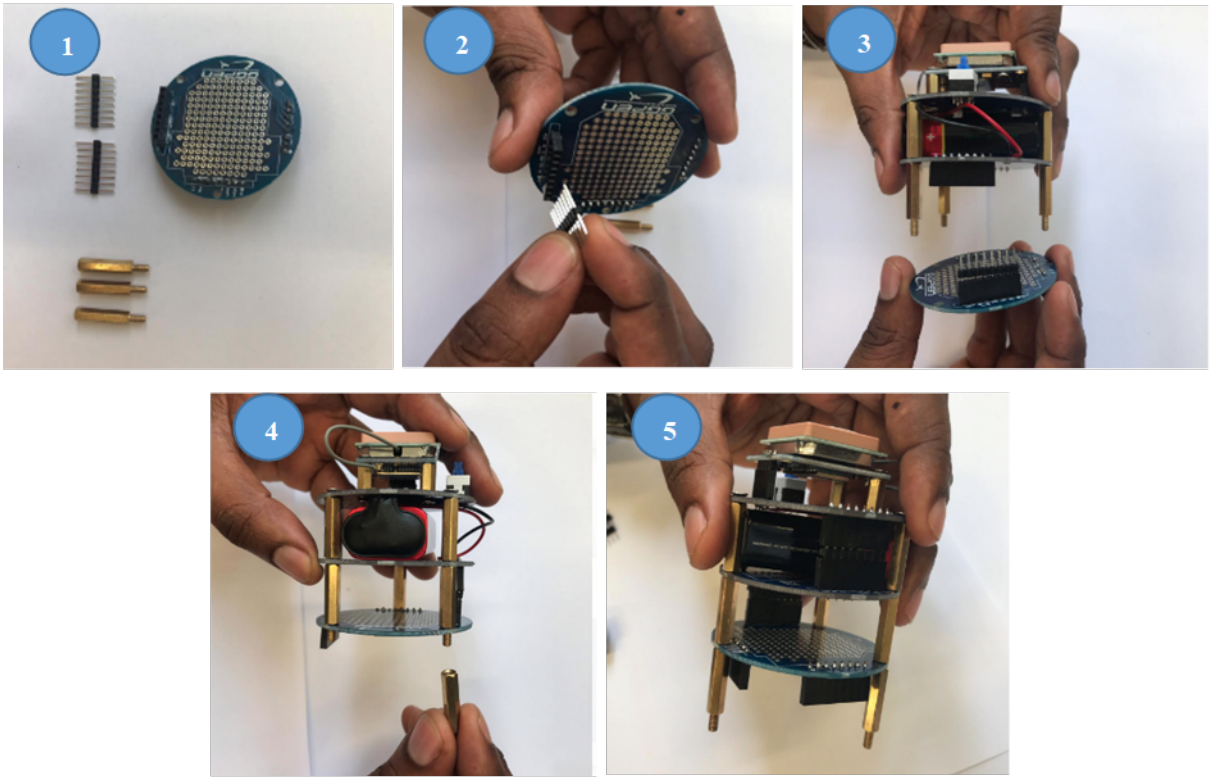


Figura 6.20 – Montagem da placa USR

4. Unir cuidadosamente a placa OBC com a placa USR. é comum nessa placa haver alguma dificuldade em alinhar os pinos. Recomenda-se verificar se todos pinos estão alinhados;
5. Instalar os espaçadores de 20mm. Instalar um pin header de 8 pinos;
6. Resultado esperado;
7. Medir a continuidade nos pinos VCC, GND, Tx e Rx entre as placas GPS e OBC. Caso não haja continuidade em algum pino significa que uma ligação foi mal feita. Verifique todos os passos anteriores.

6.4.5 Placa COM

A Figura 6.22 ilustra a sequência do procedimento de montagem da placa COM.

1. Preparar os componentes da placa COM;
2. Instalar o pin header de 6 pinos;
3. Unir a placa COM com a placa OBC;
4. Instalar os espaçadores de 20mm;
5. Instalar a antena ao módulo APC220;
6. Resultado esperado.

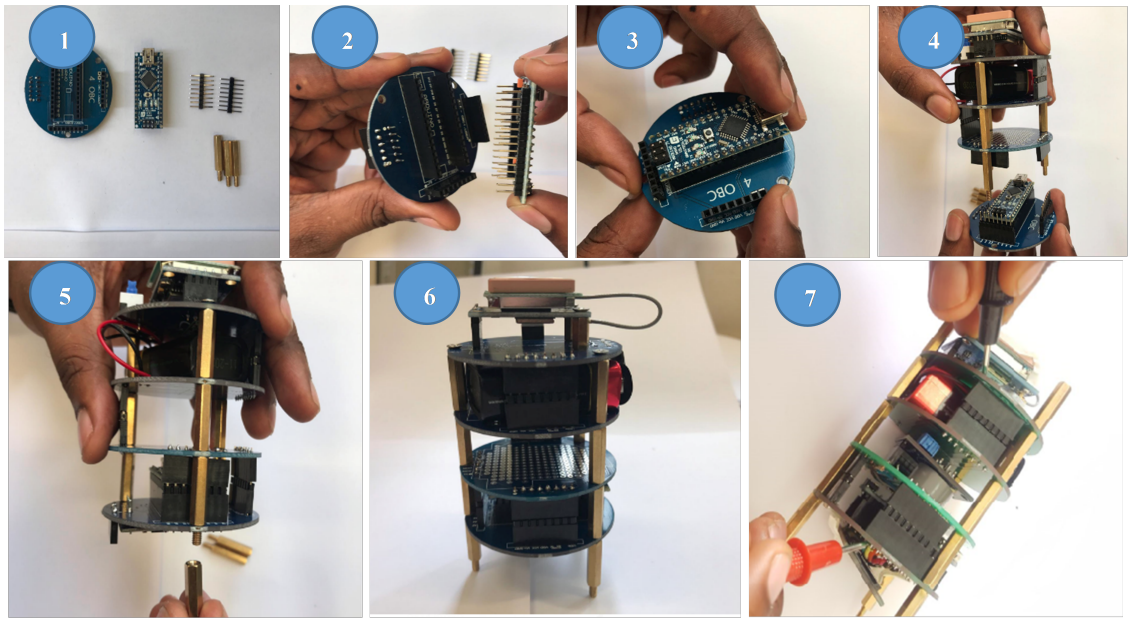


Figura 6.21 – Montagem da placa OBC

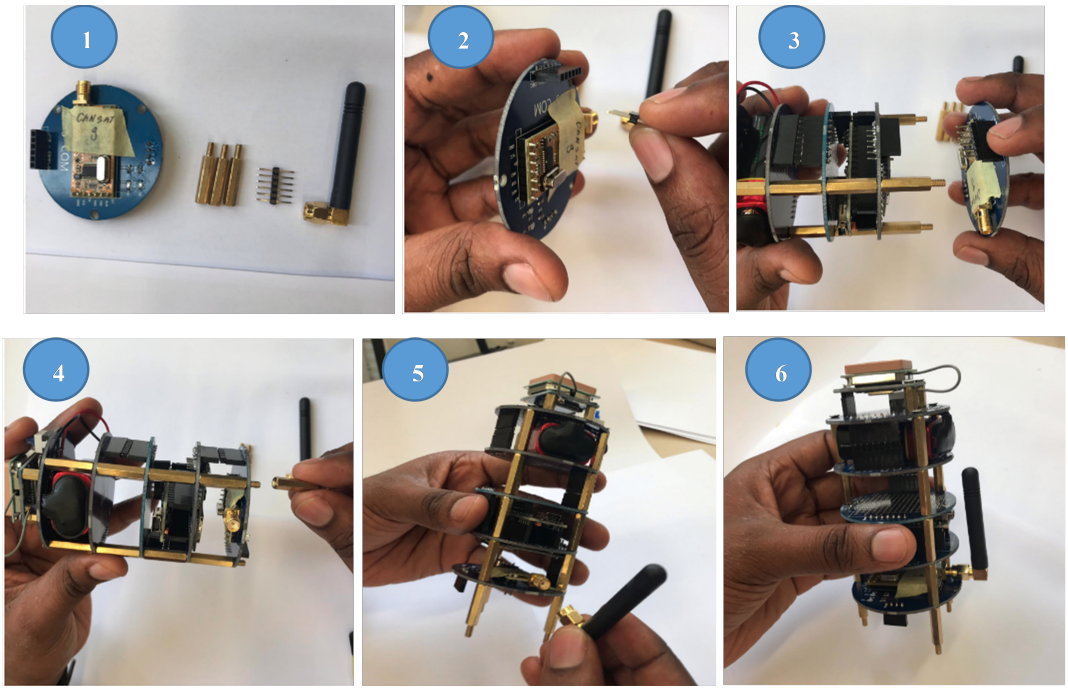


Figura 6.22 – Montagem da placa COM

6.4.6 Placa CAM

A Figura 6.23 ilustra a sequência do procedimento de montagem da placa CAM.

1. Preparar os componentes da placa CAM;
2. Instalar o pin header de 4 pinos;
3. Unir a placa CAM com a placa COM;
4. Instalar os espaçadores de 20mm;

5. Instalar os espaçadores de suporte de 10mm;

6. Testar a continuidades entre os pinos. Testar a continuidade dos pinos GND/GND e VCC/VCC entre a placa do GPS e a placa da câmara.

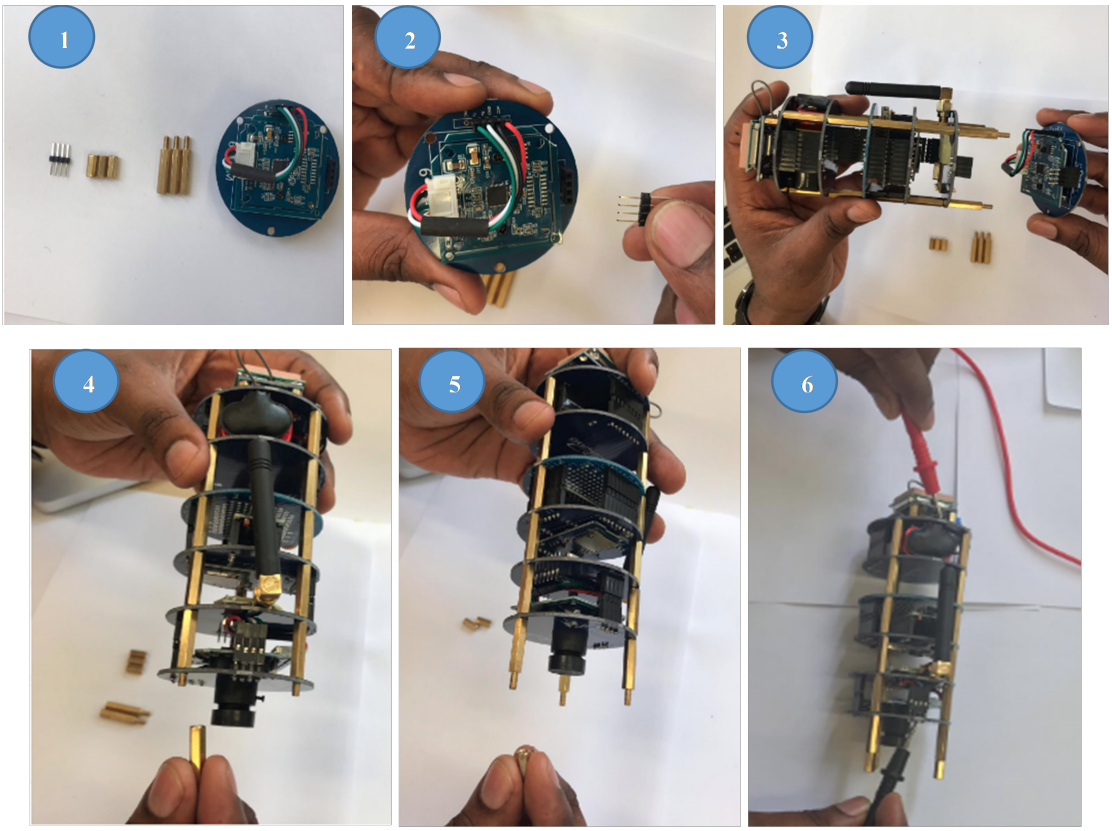


Figura 6.23 – Montagem da placa CAM

6.5 Programação e Testes

6.5.1 Programação do OBC

A programação do OBC é feita usando o arduino IDE [5]. Deve-se conectar o arduino nano ao computador usando o cabo USB,

Medição de Temperatura e Humidade Relativa

Para a realização da missão de Medição de Temperatura e Humidade Relativa, será necessário adicionar a placa USR o módulo DHT11 [6]. Na Figura 6.25 ilustra o esquema de ligação sugerido para implementação do sensor GY-91[8] e do DHT11. Para está conexão é importante referir que a alimentação do DHT11 (VCC) será feita pelo pino A2 do arduino.

Figura 6.24 – Cansat conectado ao computador

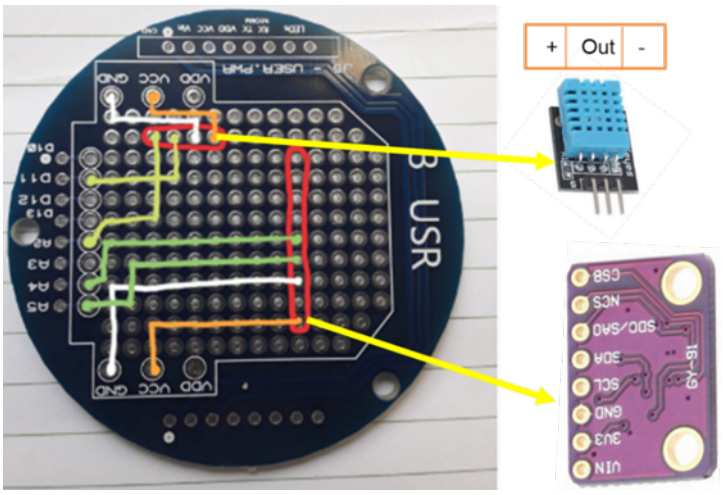


Figura 6.25 – Esquema de ligação sugerido para a missão 1

Medição de Concentração de Monóxido de Carbono

Para a realização da missão de Medição de Concentração de Monóxido de Carbono na atmosfera será necessário adicionar é placa USR o módulo MQ7 [7]. Na Figura 6.26 ilustra o esquema de ligação sugerido para implementação do sensor GY-91 e do MQ7.

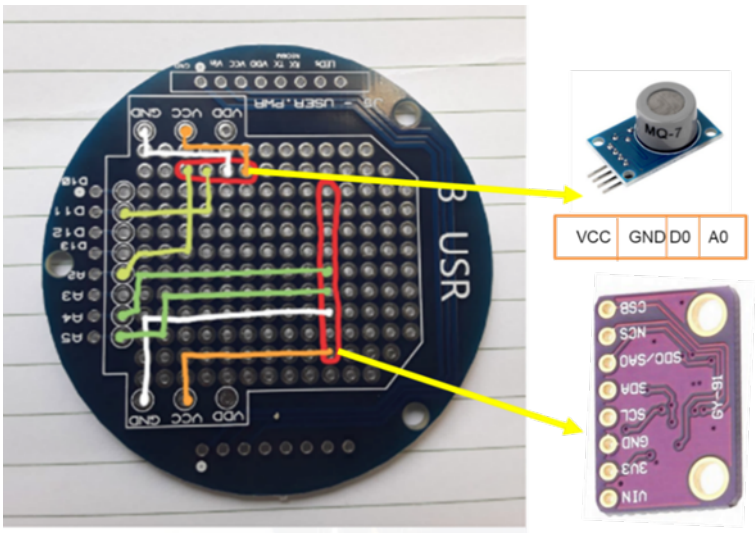


Figura 6.26 – Esquema de ligação sugerido para a missão 2

Captação de Fotos

A missão de Captação de Fotos é feita usando a câmara VC0706 e as fotos são guardadas num cartão micro SD. Essas fotos podem ser visualizadas a posterior num computador. O exemplo de ligação sugerido é apresentado na Figura 6.27.

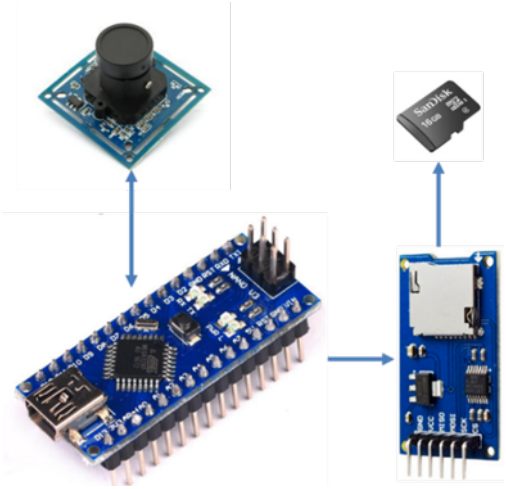


Figura 6.27 – Esquema de ligação sugerido para a missão 3

6.5.2 Estação Terrena

Para envio e recepção de dados entre o cansat e o computador, foi desenvolvido uma interface gráfica (Figura 6.28). Os dados de telemetria recebidos do APC220 vêm na forma de pacotes. Esses dados precisam ser interpretados e distribuídos nas respectivas variáveis. Essa operação é feita na interface Ground Station onde é possível ver os dados de telemetria, gráficos, mapas e também enviar comandos.

Na área de Comandos, há dois botões para controlo da câmara, a label "Estado"mostra os diferentes estados de operação da câmara. O botão "Zerar Altimetro"serve para fazer um desconto (offset) no valor fornecido pelo sensor GY-91, possibilitando definir a altura zero (altura de base) a partir de onde o cansat será lançado. Os botões 1, 2 e 3 são botões genéricos e programáveis. Ao clicar nesses botões com o botão esquerdo do rato são enviadas as informações programadas inicialmente (1, 2 ou 3) e ao clicar com o botão direito, abrirá duas caixas em que pode-se alterar o nome do botão e a informação a ser enviada.

No menu "CONEXÃO", tem as opções de conexão, escolha de velocidade (Baud Rate), escolha da porta COM e um botão de actualização de portas. No menu "SENSORES", deve-se escolher qual é a missão a ser cumprida. Deve-se escolher entre as opções "DHT11", "MQ7 CO"ou "VC0706". Existe também o botão Gráficos"que habilita ou desabilita a exibição dos gráficos.

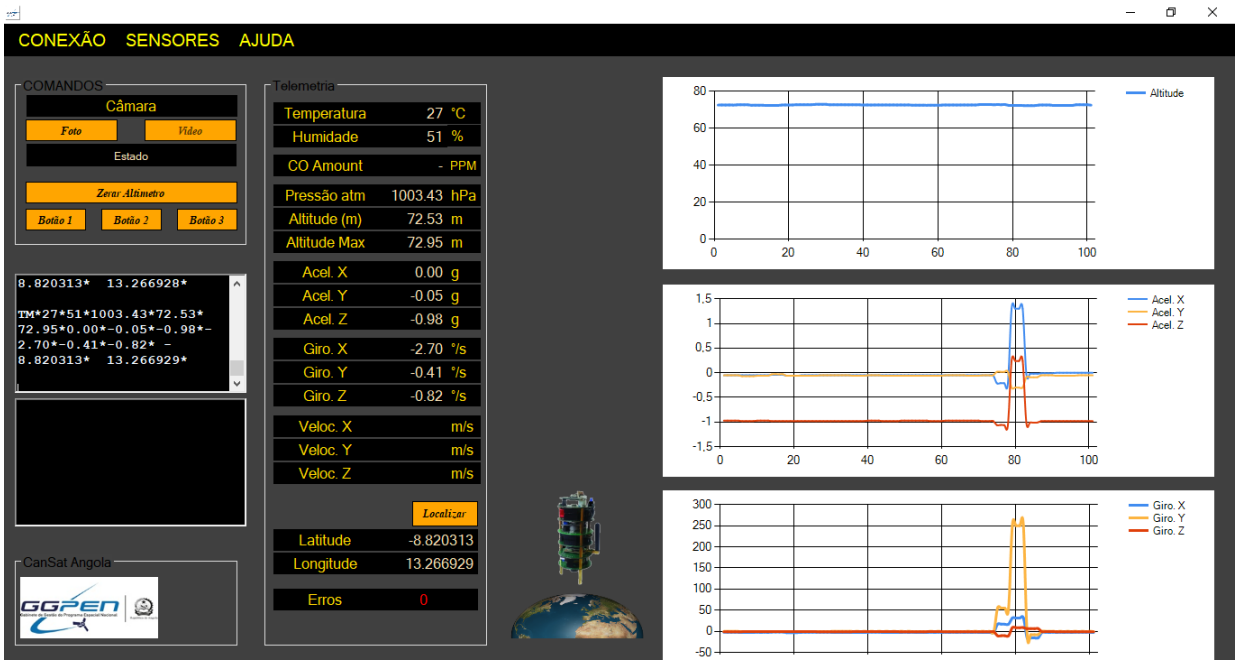


Figura 6.28 – Ground Station

Referências Bibliográficas

[1] https://alen.space/basic-guide-nanosatellites/

[2] Manual Técnico do HeptaSAT, GGPEN, 2018

[3] The Sixth CanSAt Leader Training Program (CLTP6)

[4] Manual do Cansat Angolano, GGPEN, 2018

[5] http://students.iitk.ac.in/eclub/assets/lectures/embedded14/arduino.pdf

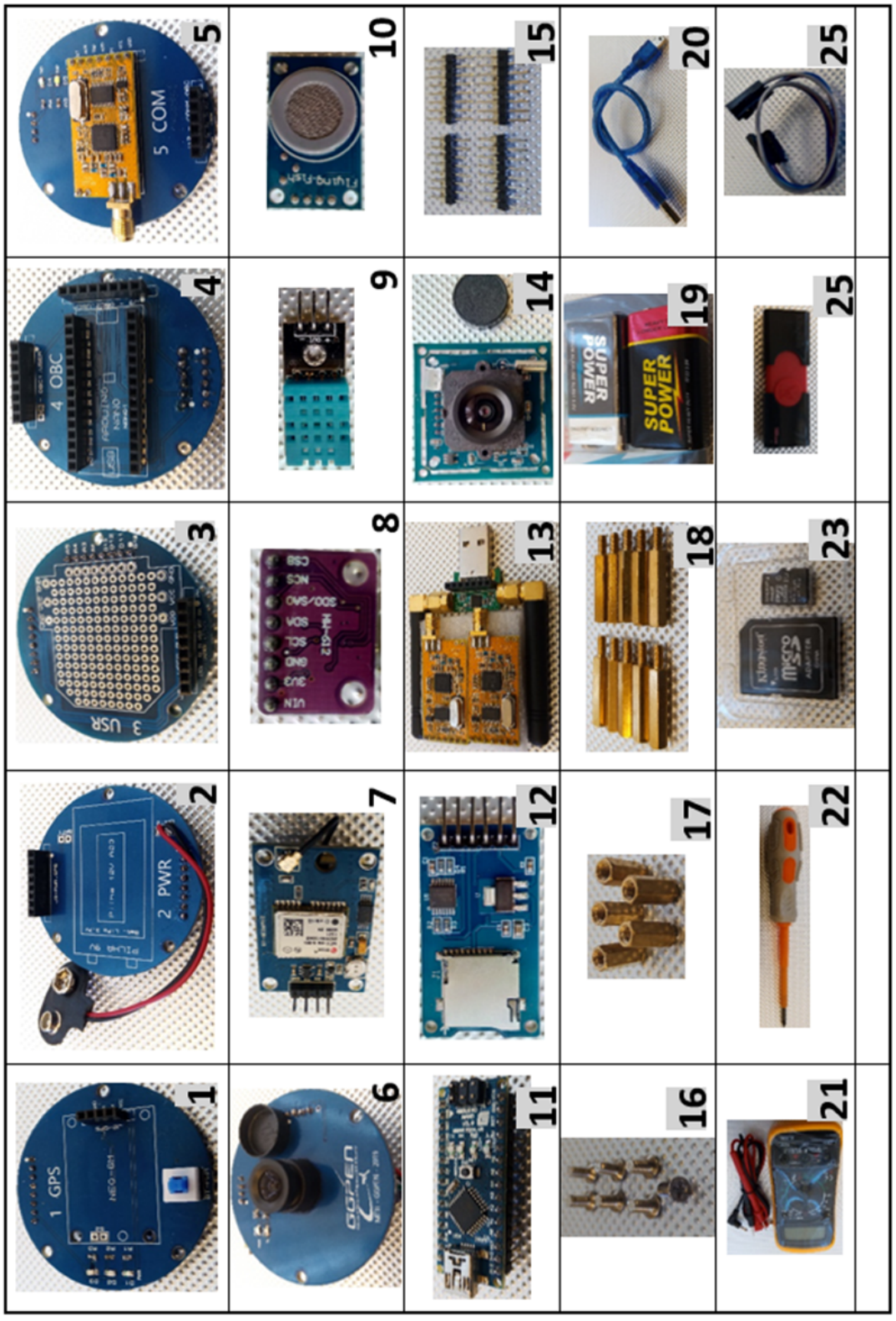
[6] https://www.mouser.com/ds/2/758/DHT11-Technical-Data-Sheet-Translated-Version-1143054.pdf

[7] https://www.sparkfun.com/datasheets/Sensors/Biometric/MQ-7.pdf

[8] https://github.com/ricardozago/GY91-MPU9250-BMP280/tree/master/Datasheet

Anexo 1

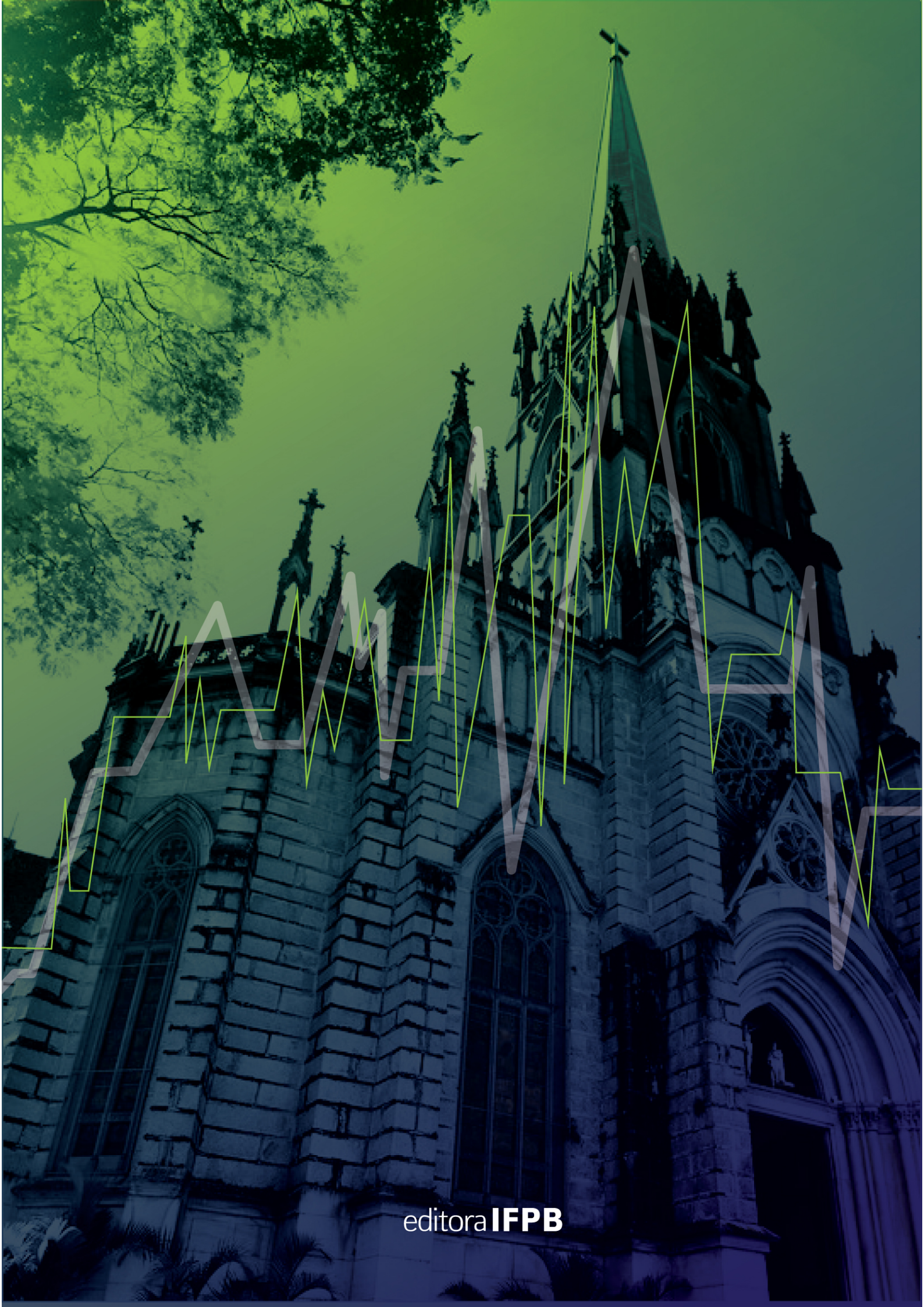
Componentes do Kit Cansat


1 GPS
2 PWR
3 USR
4 OBC
5 COM
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
25
25

Anexo 2

Tabela de Verificação (Check-List)

Nº	Item	Nº de Componentes	☑
1	Placa GPS	1	
2	Placa do Sistema de Alimentação (PWR)	1	
3	Placa USR (Payload)	1	
4	Placa do Sistema de Computador de Bordo (OBC)	1	
5	Placa do Sistema de Comunicação (COM)	1	
6	Placa Câmara (CAM)	1	
7	Módulo GPS NEO-6M (ou 8M)	1	
8	Sensor GY-91	1	
9	Sensor DHT11	1	
10	Sensor MQ7	1	
11	Arduíno Nano	1	
12	Módulo SD Card Reader	1	
13	Módulo APC220	1	
14	Câmara VC0703	1	
15	Pin Header	6	
16	Parafuso (+) M3	7	
17	Espaçador (parafuso fêmea e fêmea) (10 mm)	7	
18	Espaçador (parafuso macho e fêmea) (20 mm)	18	
19	Pilha de 9V	2	
20	Cabo USB	1	
21	Multímetro	1	
22	Chave estrela	1	
23	Cartão de memória micro SD com adaptador	1	
24	USB Pendrive	1	
25	Jumpers	2	
26	Manual do Cansat	1	



editora IFPB