MULTIPATH MITIGATION IN TIME-DELAY ESTIMATION VIA TENSOR-BASED TECHNIQUES FOR ANTENNA ARRAY-BASED GNSS RECEIVERS

DANIEL VALLE DE LIMA

DISSERTAÇÃO DE MESTRADO EM ENGENHARIA ELÉTRICA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASÍLIA

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

MULTIPATH MITIGATION IN TIME-DELAY ESTIMATION VIA TENSOR-BASED TECHNIQUES FOR ANTENNA ARRAY-BASED GNSS RECEIVERS

DANIEL VALLE DE LIMA

Orientador: PROF. DR.-ING. JOÃO PAULO C. L. DA COSTA, ENE/UNB, TU ILMENAU, FRAUNHOFER IIS

DISSERTAÇÃO DE MESTRADO EM ENGENHARIA ELÉTRICA

PUBLICAÇÃO PPGENE.DM - XXX/AAAA
BRASÍLIA-DF, 9 DE MARÇO DE 2017.

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

MULTIPATH MITIGATION IN TIME-DELAY ESTIMATION VIA TENSOR-BASED TECHNIQUES FOR ANTENNA ARRAY-BASED GNSS RECEIVERS

DANIEL VALLE DE LIMA

DISSERTAÇÃO DE MESTRADO ACADÊMICO SUBMETIDA AO DEPARTAMENTO DE ENGENHARIA ELÉTRICA DA FACULDADE DE TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA, COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM ENGENHARIA ELÉTRICA.

APROVADA POR:

Prof. Dr.-Ing. João Paulo C. L. da Costa, ENE/UnB, TU Ilmenau, Fraunhofer IIS
Orientador

Prof. Dr. Ugo Silva Dias, ENE/UnB
Examinador interno

Prof. Dr.-Ing. Felix Antreich, DETI/UFC
Examinador externo

Prof. Dr. Ricardo Zelenovsky, ENE/UnB
Suplente

BRASÍLIA, 9 DE MARÇO DE 2017.

FICHA CATALOGRÁFICA

DANIEL VALLE DE LIMA
Multipath Mitigation in Time-Delay Estimation via Tensor-based Techniques for Antenna Array-based GNSS receivers
2017xv, 63p., 201x297 mm
(ENE/FT/UnB, Mestre, Engenharia Elétrica, 2017)
Dissertação de Mestrado - Universidade de Brasília
Faculdade de Tecnologia - Departamento de Engenharia Elétrica

REFERÊNCIA BIBLIOGRÁFICA

DANIEL VALLE DE LIMA (2017) Multipath Mitigation in Time-Delay Estimation via Tensor-based Techniques for Antenna Array-based GNSS receivers. Dissertação de Mestrado em Engenharia Elétrica, Publicação xxx/AAAA, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 63p.

CESSÃO DE DIREITOS

AUTOR: Daniel Valle de Lima
TÍTULO: Multipath Mitigation in Time-Delay Estimation via Tensor-based Techniques for Antenna Array-based GNSS receivers.
GRAU: Mestre ANO: 2017

Daniel Valle de Lima
SHIN QL 7 Conj. 3 Casa 2

Agradecimentos

Aos meus pais, pelo apoio e amor incondicional que me deram.

Ao meu irmão, por seu amor fraterno e por nossas discussões filosóficas.

Ao meu orientador de Graduação, Dr. Antônio José Martins, pelas aulas de Electromagnetismo e por possibilitar minha formação inicial como Engenheiro Eletricista.

Ao meu orientador de Mestrado, Dr.-Ing. João Paulo Lustosa, por ser um exemplo a ser seguido de realização acadêmica, proatividade e curiosidade científica. Sou grato também por sua liderança e orientação, além do investimento de tempo, confiança, e paciência necessária para guiar minha formação como engenheiro e pesquisador.

Àqueles meus colegas que também escolheram o árduo caminho da Engenharia. Sempre podemos aprender com os companheiros de batalha.

Aos professores que me fizeram merecer o título de Engenheiro.

Ao Dr.-Ing. Antreich e o Dr. Dias por fazerem parte da banca e pela crítica que me ajudou a aperfeiçoar esta dissertação.

À Universidade de Brasília e à todos aqueles que possibilitam o funcionamento e a continuação desta instituição.

Dedicatória

Para meus pais.

Resumo

Motivação

Tradicionalmente sistemas satélites de navegação global, em inglês *Global Navigation Satellite System* (GNSS), como o sistema de posicionamento global, do inglês *Global Positioning System* (GPS), Galileo, o GNSS da União Européia, GLONASS, o GNSS da Federação Russa, ou o BeiDou, o GNSS da República Popular da China, foram concebidos para aplicações militares como sistemas de misseis guiados e para aplicações civis como decolagem e pouso de aviões civis. Na aviação civil, sistemas de apoio assistidos por bases terrestres que providenciam informações complementares ao do GNSS aumentam a precisão para sistemas de segurança crítica. Nos últimos anos a quantidade de aplicações de GNSS têm aumentado drasticamente. Por exemplo, autoridades de pesca utilizam GNSS para fazer a localização e rastreio automático em tempo real de barcos pesqueiros para garantir assim o gerenciamento sustentável de fontes de pesca [1]. Outra aplicação de GNSS é o rastreio de caminhões para poder saber o estado da carga em tempo real. Em aplicações de trânsito, o GNSS pode ser utilizado para efetuar um pedágio automatico [2] e para veículos autônomos que exigem altos padrões de precisão e segurança. No contexto de veículos autônomos, o veículo deve ser capaz de sensorear o ambiente e os dados processados para atingir o padrão de segurança necessário. Apesar de veículos autônomos contarem com uma quantidade grande de sensores para controle automático de velocidade de cruzeiro, do inglês *Automatic Cruise Control* (ACC), receptores GNSS exercem uma função essencial [3] devido à cobertura quase ubíqua de todas as regiões do planeta. Finalmente, em agricultura de precisão, também conhecido por agricultura por satélite, GNSS é usado para melhorar, por exemplo, a precisão com que é feita a adubagem e permite o uso de veículos agrícolas automatizados em qualquer hora do dia [4]. Assim, na agricultura de precisão, tanto máquinas quanto recursos químicos podem ser usados de forma mais segura e eficiente.

GNSS depende da estimação do atraso para estimar a posição do usuário. Isto é feito fazendo a correlação do sinal recebido com réplicas para separá-lo o sinal de cada satélite e estimar o atraso. Como componentes de multipercurso são cópias atrasadas do sinal original, estes alteram a função de correlação cruzada, assim gerando erros na estimação de atraso.

Neste dissertação estudamos um algoritmo estado-da-arte em mitigação de multipercursos para estimação de atraso baseado no autofiltro da decomposição em valores singular de

alta ordem, do inglês *Higher-Order Singular Value Decomposition* (HOSVD), de posto unitário [5], e propomos dois esquemas tensoriais para mitigação de multipercurso e estimação de atraso, para qual o esquema baseado em HOSVD é usado para comparação.

Modelo de Dados

O modelo de dados do receptor tensorial supõe um arranjo receptor de M elementos, observando D satélites visíveis, com L_d sinais vindo de cada d-ésimo satélite. Com $L_d = 1$ para o componente de linha de visada, do inglês *Line-of-Sight* (LOS), e $L_d > 1$ para os componentes de multipercurso. Coletando N amostras a cada k-ésimo período de código,

$$\mathbf{X}[k] = \sum_{d=1}^{D} \mathbf{A}_d[k]\mathbf{\Gamma}_d[k]\mathbf{C}_d[k] + \mathbf{N}[k] = \mathbf{A}[k]\mathbf{\Gamma}[k]\mathbf{C}[k] + \mathbf{N}[k] \in \mathbf{C}^{M\times N}, \tag{1}$$

em que $\mathbf{A}_d[k] = [\mathbf{a}(\phi_{d,1}), \dots, \mathbf{a}(\phi_{d,L_d})] \in \mathbb{C}^{M\times L_d}$ é a matriz que concatena em suas colunas os vetores de direção dos L_d sinais. $\mathbf{\Gamma}_d[k] = \mathrm{diag}\{\gamma_d\} = \mathrm{diag}\{[\gamma_{d,1}, \dots, \gamma_{d,L_d}]\} \in \mathbb{C}^{L_d\times L_d}$ é uma matriz diagonal que contém as amplitudes complexas dos sinais. $\mathbf{C}_d[k] = [\mathbf{c}_d[\tau_{d,1}], \dots, \mathbf{c}_d[\tau_{L_d}]]^{\mathrm{T}} \in \mathbb{R}^{L_d\times N}$ concatena em suas linhas o código pseudo-aleatório amostrado com atraso $\tau_{d,l_d}, l_d = 1, \dots, L_d$. $\mathbf{N}[k]$ é ruído gaussiano branco aditivo. A ordem do modelo é $L = \sum_{d=1}^{D} L_d$.

Aplicando o operador $\mathrm{vec}\{\cdot\}$ a (1) para transformá-lo num vetor e aplicando a propriedade descrita na subseção 2.2.4,

$$\begin{aligned}\mathrm{vec}\{\mathbf{X}[k]\} &= \mathrm{vec}\left\{\mathbf{A}[k]\mathbf{\Gamma}[k]\mathbf{C}[k] + \mathbf{N}[k]\right\} = \mathrm{vec}\left\{\mathbf{A}[k]\,\mathrm{diag}\{\boldsymbol{\gamma}[k]\}\mathbf{C}[k] + \mathbf{N}[k]\right\},\\ &= (\mathbf{C}[k]^{\mathrm{T}} \diamond \mathbf{A}[k])\boldsymbol{\gamma}[k] + \mathrm{vec}\{\mathbf{N}[k]\}.\end{aligned} \tag{2}$$

Concatenando todos os K períodos de código amostrados na direção das colunas é possível omitir o índice k:

$$\tilde{\mathbf{X}} = (\mathbf{C}^{\mathrm{T}} \diamond \mathbf{A})\tilde{\mathbf{\Gamma}} + \tilde{\mathbf{N}} = \tilde{\mathbf{X}}_0 + \tilde{\mathbf{N}} \in \mathbb{C}^{MN\times K}, \tag{3}$$

em que $\tilde{\mathbf{X}}_0$ é o sinal recebido sem ruído e $\tilde{\mathbf{\Gamma}} = [\boldsymbol{\gamma}[1] \dots, \boldsymbol{\gamma}[K]] \in \mathbb{C}^{L\times K}$ concatena as amplitudes complexas.

A transposta de $\tilde{\mathbf{X}}_0$ possui a mesma estrutura que o desdobramento do primeiro modo de um tensor de recepção sem ruído $\boldsymbol{\mathcal{X}}_0$:

$$[\boldsymbol{\mathcal{X}}_0]_{(1)} = \tilde{\mathbf{X}}_0^{\mathrm{T}} = \tilde{\mathbf{\Gamma}}^{\mathrm{T}}(\mathbf{C}^{\mathrm{T}} \diamond \mathbf{A}) \in \mathbb{C}^{K\times MN}. \tag{4}$$

Dobrando (4) num tensor e considerando o caso com ruído:

$$\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{X}}_0 + \boldsymbol{\mathcal{N}} = \boldsymbol{\mathcal{I}}_{3,L} \times_1 \tilde{\mathbf{\Gamma}}^{\mathrm{T}} \times_2 \mathbf{C}^{\mathrm{T}} \times_3 \mathbf{A} + \boldsymbol{\mathcal{N}} \in \mathbb{C}^{K \times N \times M}. \tag{5}$$

Para separar o sinal do d-ésimo satélite dos outros, um banco correlator é aplicado ao código pseudo-aleatório usando o produto de modo-2. Um banco correlator é uma matriz $\mathbf{Q}_d$ que concatena Q réplicas deslocadas da sequência pseudo-aleatória $\mathbf{c}_d \in \mathbb{R}^N$ com atraso $\tau_q, q = 1, \ldots, Q$:

$$\mathbf{Q}_d = \begin{bmatrix} \mathbf{c}_d[\tau_1] & \cdots & \mathbf{c}_d[\tau_Q] \end{bmatrix} \in \mathbb{R}^{N \times Q}. \tag{6}$$

Como a aplicação direta do banco correlator torna o ruído colorido, um banco comprimido é calculado utilizando a decomposição em valores singulares econômica à $\mathbf{Q}_d$ [22]:

$$\mathbf{Q}_d = \mathbf{Q}_{\omega,d} \mathbf{\Sigma} \mathbf{V}^{\mathrm{H}}, \tag{7}$$

em que o banco comprimido é $\mathbf{Q}_{\omega,d} \in \mathbb{C}^{N \times Q}$.

Aplicando o banco correlator comprimido à (5) para extrair o sinal do d-ésimo satélite:

$$\boldsymbol{\mathcal{Y}}_d = \boldsymbol{\mathcal{X}} \times_2 \mathbf{Q}_{\omega,d}^{\mathrm{T}} = \boldsymbol{\mathcal{I}}_{3,L} \times_1 \tilde{\mathbf{\Gamma}}_d^{\mathrm{T}} \times_2 (\mathbf{C}_d \mathbf{Q}_{\omega,d})^{\mathrm{T}} \times_3 \mathbf{A}_d + \boldsymbol{\mathcal{N}}_\omega \in \mathbb{C}^{K \times Q \times M}. \tag{8}$$

O desdobramento do terceiro modo de (8) é

$$[\boldsymbol{\mathcal{Y}}_d]_{(3)} = \mathbf{A}_d (\tilde{\mathbf{\Gamma}}_d^{\mathrm{T}} \diamond (\mathbf{C}_d \mathbf{Q}_{\omega,d})^{\mathrm{T}})^{\mathrm{T}} + [\boldsymbol{\mathcal{N}}_\omega]_{(3)} \in \mathbb{C}^{M \times KQ}. \tag{9}$$

Estado-da-Arte para Estimação de Atraso

A técnica estado-da-arte estudado neste dissertação é um autofiltro de alta ordem com pré-processamento usando média frente-costas [16], do inglês *Forward-Backward Averaging* (FBA), e suavização espacial expandida [18, 19], do inglês *Expanded Spatial Smoothing* (ESPS), como pode ser visto na Figura 3.1. A primeira etapa desta técnica é de pré-processamento em que FBA é aplicado a (9):

$$\mathbf{Z} = \begin{bmatrix} [\boldsymbol{\mathcal{Y}}_d]_{(3)} & \mathbf{\Pi}_M [\boldsymbol{\mathcal{Y}}_d]_{(3)}^{*} \mathbf{\Pi}_{KQ} \end{bmatrix} \in \mathbb{C}^{M \times 2KQ}, \tag{10}$$

em que $\mathbf{\Pi}_M \in \mathbb{R}^{M \times M}$ é uma matriz identidade invertida sobre o eixo vertical e $\mathbf{\Pi}_{KQ} \in \mathbb{R}^{KQ \times KQ}$ idem.

Seguido de suavização espacial, em inglês *Spatial Smoothing* (SPS),em que o arranjo é dividido em L_S sub-arranjos com M_S = $M - L_S + 1$ elementos. Isto é feito usando uma matriz de seleção

$$\mathbf{J}_{\mathbf{l_S}} = \begin{bmatrix} \mathbf{0}_{M_S \times l_S - 1} & \mathbf{I}_{M_S} & \mathbf{0}_{M_S \times L_S - 1} \end{bmatrix} \in \mathbb{R}^{M_S \times M}, \tag{11}$$

para $l_S = 1, \ldots, L_S$.

Usando as matrizes de seleção, suavização espacial é aplicado a (10):

$$\mathbf{W} = \begin{bmatrix} \mathbf{J}_1 \mathbf{Z} & \cdots & \mathbf{J}_{L_S} \mathbf{Z} \end{bmatrix} \in \mathbb{C}^{M_S \times 2 L_S K Q}, \tag{12}$$

e (12) é dobrado de volta para um tensor de quarta ordem $\boldsymbol{\mathcal{Z}}_{\text{ESPS}} \in \mathbb{C}^{2K \times Q \times M_S \times L_S}$. Esta é a suavização espacial expandida.

Em seguida é aplicada uma decomposição em valores singulares de alta ordem, em inglês *Higher-Order Singular Value Decomposition* (HOSVD) à $\boldsymbol{\mathcal{Z}}_{\text{ESPS}}$:

$$\boldsymbol{\mathcal{Z}}_{\text{ESPS}} = \boldsymbol{\mathcal{R}} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)} \times_4 \mathbf{U}^{(4)}, \tag{13}$$

em que $\boldsymbol{\mathcal{R}} \in \mathbb{C}^{2K \times Q \times M_S \times L_S}$ é o tensor núcleo e $\mathbf{U}^{(1)} \in \mathbb{C}^{2K \times 2K}$, $\mathbf{U}^{(2)} \in \mathbb{C}^{Q \times Q}$, $\mathbf{U}^{(3)} \in \mathbb{C}^{M_S \times M_S}$, e $\mathbf{U}^{(4)} \in \mathbb{C}^{L_S \times L_S}$ são as matrizes singulares contendo os vetores singulares dos desdobramento de (8) em cada modo, respectivamente.

Supondo que os componentes de sinais LOS são dominantes, os vetores singulares dominantes do primeiro, terceiro, e quarto modo de $\boldsymbol{\mathcal{Z}}_{\text{ESPS}}$ são mais correlatados ao componente LOS. Portanto [5] propôs o seguinte autofiltro:

$$\mathbf{q}_{\text{ESPS}} = \left(\boldsymbol{\mathcal{Z}}_{\text{ESPS}} \times_1 (\mathbf{u}_1^{(1)})^{\text{H}} \times_3 (\mathbf{u}_1^{(3)})^{\text{H}} \times_4 (\mathbf{u}_1^{(4)})^{\text{H}} \right) \boldsymbol{\Sigma} \mathbf{V}^{\text{H}}, \tag{14}$$

em que $\boldsymbol{\Sigma} \mathbf{V}^{\text{H}}$ foi calculado em (7).

Para estimação do atraso, uma interpolação de spline cúbico é aplicado ao valor absolute de (14) para criar uma função de custo $F(\tau)$. A variável τ que maximiza $F(\tau)$ é

$$\hat{\tau}_{\text{LOS}} = \arg \max_{\tau} \{F(\tau)\}. \tag{15}$$

Técnicas Tensoriais Propostas

Esquema tensorial por estimação da matriz de direção e fatorização Khatri-Rao (DoA/KRF)

O primeiro esquema tensorial é um método em três etapas que aplica estimação da direção de chegada, do inglês *Direction of Arrival* (DoA), e fatorização Khatri-Rao, do inglês *Khatri-Rao factorization* (KRF), para separar o código de cada componente incidente de forma fechada.

Após o pré-processamento de (9) usando FBA e SPS, estimação de DoA é aplicado a (12) para estimar a matriz-fator de direção de chegada $\hat{\mathbf{A}}_d$. Como $[\boldsymbol{\mathcal{Y}}_d]_{(3)} \approx \mathbf{A}_d(\tilde{\mathbf{\Gamma}}^{\mathrm{T}} \diamond (\mathbf{C}_d\mathbf{Q}_{\omega,d})^{\mathrm{T}})^{\mathrm{T}}$, o produto da pseudo-inversa de $\hat{\mathbf{A}}_d$ por $[\boldsymbol{\mathcal{Y}}_d]_{(3)}$ é

$$\hat{\mathbf{A}}_d^{+}[\boldsymbol{\mathcal{Y}}_d]_{(3)} \approx \hat{\mathbf{A}}_d^{+}\mathbf{A}_d(\tilde{\mathbf{\Gamma}}_d^{\mathrm{T}} \diamond (\mathbf{C}_d\mathbf{Q}_{\omega,d})^{\mathrm{T}})^{\mathrm{T}} \approx (\tilde{\mathbf{\Gamma}}_d^{\mathrm{T}} \diamond (\mathbf{C}_d\mathbf{Q}_{\omega,d})^{\mathrm{T}})^{\mathrm{T}}. \tag{16}$$

Em seguida, uma KRF é aplicado a (16) para encontrar as matrizes-fator $\hat{\tilde{\Gamma}}_d$ e $\hat{\mathrm{C}}_d\mathrm{Q}_{\omega,d}$. Com o código separado é possível determinar a correlação usando $\mathbf{\Sigma}\mathbf{V}^{\mathrm{H}}$:

$$\hat{\mathbf{C}}_d\mathbf{Q}_d = \hat{\mathbf{C}}_d\mathbf{Q}_{\omega,d}\mathbf{\Sigma}\mathbf{V}^{\mathrm{H}}. \tag{17}$$

Por causa da possibilidade de ambiguidade de permutação na estimação do DoA ou na KRF, um esquema de seleção é aplicado para descobrir qual linha de (17) contém a correlação com o componente de linha de visada. Dois esquemas de seleção foram propostos.

O primeiro esquema de seleção é baseado em potência e supõe que o componente LOS do sinal é o de maior potência, isto é

$$l_{\mathrm{LOS}} = \max_{l_d} \|(\hat{\tilde{\mathbf{\Gamma}}}_d)_{l_d,\cdot}\|_2, \tag{18}$$

e interpolação de spline cúbica é aplicada para estimar o atraso $\hat{\tau}$ apenas da linha selecionada de (17).

No segundo esquema de seleção, a interpolação de spline cúbica é aplicada a todas as linhas de (17) para estimar os atrasos dos componentes LOS e NLOS num vetor $\hat{\boldsymbol{\tau}}$ e supõe que o componente LOS do sinal é o de menor atraso destes, portanto

$$\hat{\tau}_{\mathrm{LOS}} = \min_{\tau}\{\hat{\boldsymbol{\tau}}\}. \tag{19}$$

Esquema tensorial por filtragem ProKRaft

O segundo esquema tensorial utiliza um desdobramento Hermitiano, calculado a partir do tensor de recepção, cuja propriedade de simetria dual [6] é explorada por um algoritmo que alterna entre a solução do problema ortogonal de Procrustes, do inglês *Orthogonal Procrustes Problem* (OPP), e fatorização Khatri-Rao de mínimos quadrados, do inglês *Least Squares Khatri-Rao Factorization* (LSKRF), para estimar iterativamente as matrizes-fator do canal. Estas são então usadas para separar o código de cada componente incidente.

O desdobramento Hermitiano calculado a partir da matriz de covariância multimodo a partir do desdobramento modo dois de (8):

$$\begin{aligned}\mathbf{R}_{\mathrm{mm}} &= [\boldsymbol{\mathcal{Y}}_d]_{(2)}^{\mathrm{T}}[\boldsymbol{\mathcal{Y}}_d]_{(2)}^{*}/N = (\mathbf{A}_d \diamond \tilde{\mathbf{\Gamma}}_d^{\mathrm{T}})(\mathbf{C}_d\mathbf{Q}_{\omega,d})(\mathbf{C}_d\mathbf{Q}_{\omega,d})^{\mathrm{H}}/N(\mathbf{A}_d \diamond \tilde{\mathbf{\Gamma}}_d^{\mathrm{T}})^{\mathrm{H}} \\ &= (\mathbf{A}_d \diamond \tilde{\mathbf{\Gamma}}_d^{\mathrm{T}})\mathbf{R}_{\mathbf{C}}(\mathbf{A}_d \diamond \tilde{\mathbf{\Gamma}}_d^{\mathrm{T}})^{\mathrm{H}} \in \mathbb{C}^{MK \times MK},\end{aligned} \tag{20}$$

em que $\mathbf{R_C} \in \mathbb{R}^{L_d \times L_d}$ é a matriz de covariância dos componentes LOS e NLOS de sinal do satélite d.

Como $\mathbf{R_C} \approx \mathbf{I}_{L_d}$ (20) pode ser aproximado como

$$\mathbf{R}_{\text{mm}} \approx (\mathbf{A}_d \diamond \tilde{\mathbf{\Gamma}}_d^{\text{T}})(\mathbf{A}_d \diamond \tilde{\mathbf{\Gamma}}_d^{\text{T}})^{\text{H}} = \mathbf{Y}_{\text{H}}, \tag{21}$$

e $\mathbf{Y}_{\text{H}}$ é o desdobramento Hermitiano de $\boldsymbol{\mathcal{Y}}_d$. Como $\mathbf{Y}_{\text{H}}$ possui a propriedade de simetria dual é possível definir uma matriz-raiz $\mathbf{Y}_{\text{H}}^{\frac{1}{2}} \in \mathbb{C}^{MK \times L_d}$ tal que:

$$\mathbf{Y}_{\text{H}} = \mathbf{Y}_{\text{H}}^{\frac{1}{2}}(\mathbf{Y}_{\text{H}}^{\frac{1}{2}})^{\text{H}}. \tag{22}$$

Usando a decomposição em valores singulares, do inglês *Singular Value Decomposition* (SVD), de (21), $\mathbf{Y}_{\text{H}} = \mathbf{U}_{\text{Y}}\mathbf{\Sigma}_{\text{Y}}\mathbf{V}_{\text{Y}}^{\text{H}}$, é possível calcular uma estimativa para $\mathbf{Y}_{\text{H}}^{\frac{1}{2}}$. Usando os vetores singulares esquerdos associados ao subespaço de sinal, $\mathbf{U}_{\text{Y}}^{[L_d]} \in \mathbb{C}^{MK \times L_d}$, os valores singulares associados ao subespaço do sinal, $\mathbf{\Sigma}_{\text{Y}}^{[L_d]} \in \mathbb{C}^{L_d \times L_d}$, e uma matriz de rotação unitária $\mathbf{W}^{\text{H}} \in \mathbb{C}^{L_d \times L_d}$:

$$\begin{aligned} \hat{\mathbf{Y}}_{\text{H}}^{\frac{1}{2}} &= \mathbf{U}_{\text{Y}}^{[L]}\mathbf{\Sigma}_{\text{Y}}^{[L]}\mathbf{W}^{\text{H}} = (\mathbf{A} \diamond \tilde{\mathbf{\Gamma}}^{\text{T}}) \\ &= \mathbf{F}\mathbf{W}^{\text{H}} = \mathbf{G}. \end{aligned} \tag{23}$$

Mapear $\mathbf{F}$ para $\mathbf{G}$ usando $\mathbf{W}^{\text{H}}$ é conhecido como o problema ortogonal de Procrustes. Uma solução conhecida para este problema é aplicar a SVD a $\mathbf{F}\mathbf{G}^{\text{H}}$ e usar os vetores singulares desta decomposição para estimar $\mathbf{W}^{\text{H}}$ [30]:

$$\begin{aligned} \mathbf{F}\mathbf{G}^{\text{H}} &= \mathbf{U}_{\text{P}}\mathbf{\Sigma}_{\text{P}}\mathbf{V}_{\text{P}}^{\text{H}} \\ \mathbf{W}^{\text{H}} &= \mathbf{U}_{\text{P}}\mathbf{V}_{\text{P}}^{\text{H}}. \end{aligned} \tag{24}$$

Um algoritmo que alterna entre (23) e (24) pode iterativamente estimar $\hat{\mathbf{A}}_d$ e $\hat{\tilde{\mathbf{\Gamma}}}_d$.

Como pode ser visto na Figura 3.3, o algoritmo inicia calculando os valores e vetores singulares de (20) e inicializando $\mathbf{W}^{(0)} = \mathbf{I}_{L_d}$.

$\mathbf{W}^{\text{H}}$ é usado para calcular $\mathbf{T}_1 = \mathbf{U}_{\text{Y}}^{[L_d]}\mathbf{\Sigma}_{\text{Y}}^{[L_d]}\mathbf{W}^{\text{H}}$. Aplicando LSKRF à $\mathbf{T}_1$ se calcula estimativas para $\hat{\mathbf{A}}_d$ e $\hat{\tilde{\mathbf{\Gamma}}}_d$, que são utilizadas para calcular $\mathbf{T}_2 = (\hat{\mathbf{A}}_d \diamond \hat{\tilde{\mathbf{\Gamma}}}_d)\mathbf{U}_{\text{Y}}^{[L_d]}\mathbf{\Sigma}_{\text{Y}}^{[L_d]}$. A SVD de $\mathbf{T}_2$ fornece as matrizes de vetores singulares $\mathbf{U}_{\text{P}}$ e $\mathbf{V}_{\text{P}}$ usadas para atualizar $\mathbf{W}^{\text{H}}$. O processo se repete até a convergência.

Com a estimativa das matrizes-fator do canal, a filtragem é feita calculando o produto modo-n com a pseudoinversa das matrizes-fator estimadas:

$$\hat{\boldsymbol{\mathcal{C}}}_d\boldsymbol{\mathcal{Q}}_{\omega,d} = \boldsymbol{\mathcal{Y}}_d \times_1 (\hat{\tilde{\mathbf{\Gamma}}}_d)^{+} \times_3 (\hat{\mathbf{A}}_d)^{+}, \tag{25}$$

onde $\hat{\boldsymbol{\mathcal{C}}}_d\boldsymbol{\mathcal{Q}}_{\omega,d}$ é o tensor que acumula os L_d códigos pseudoaleatórios correlacionados dos componentes LOS e NLOS do sinal. Cada l-ésimo código se acumula em cada vetor $(\hat{\boldsymbol{\mathcal{C}}}_d \boldsymbol{\mathcal{Q}}_{\omega,d})_{l,\cdot,l}$.

Assim como no esquema anterior proposto, pode haver ambiguidades de permutação, e um esquema de seleção igual ao descrito anteriormente é utilizado para selecionar o componente de sinal LOS.

Simulação

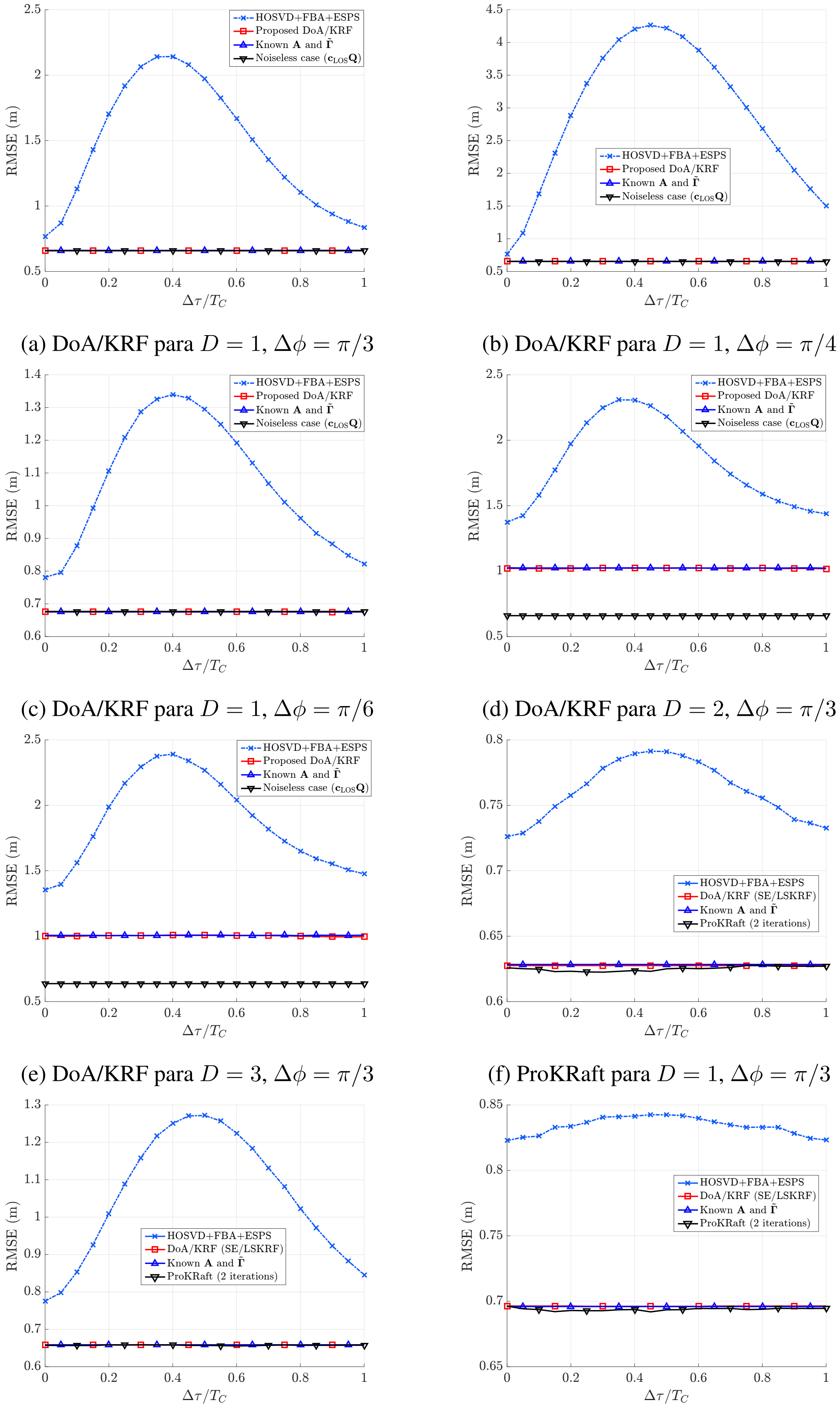
É considerado o seguinte cenário sob simulação: é utilizado um arranjo retangular uniforme, do inglês *Uniform Linear Array* (ULA), centro-hermitiano com $M = 8$ elementos e espaçamento de meio comprimento de onda. O sinal GNSS é um código pseudo aleatório C/A de GPS de D satélites, com portadora de frequência $f_c = 1575,42$ MHz, largura de banda $B = 1,023$ MHz, e duração de *chip* $T_C = 1/B = 977,52$ ns, com $N = 2046$ amostras coletadas a cada k-ésimo período de código durante $K = 30$ períodos, cada um com duração de $\Delta t = 1$ ms.

Além do sinal LOS com atraso τ_{LOS}, há um componente NLOS ($L = 2$) com atraso τ_{LOS} e diferença de azimute de $\Delta\phi$. O atraso entre τ_{LOS} e τ_{NLOS} é tal que $\tau_{\text{NLOS}} = \tau_{\text{LOS}} + \Delta t$. Para SPS/ESPS o arranjo é dividido em $L_S = 5$ subarranjos com $M_S = 4$ elementos cada.

As fases do sinal $\arg\{\gamma\} \sim \text{U}[0, 2\pi]$ independentes e identicamente distribuídos. Para as simulações do esquema DoA/KRF, as fases se mantém constante durante todo a amostragem. Para as simulações do esquema ProKRaft, as fases mudam a cada período de código. O número de correlatores no banco é $Q = 11$ igualmente espaçados entre $-T_C$ e T_C. O sinal LOS é selecionado utilizando (18). A razão portadora ruído é de 48 dB-Hz, resultando numa razão sinal-ruído pós-correlação de $\text{SNR}_{\text{pós}} \approx 15$ dB.

A simulação é de Monte Carlo com 1000 iterações. As curvas são o erro quadrático médio multiplicado por $c = 299792458$ m/s, expressas em metros, para diferentes $\Delta\tau$. Os resultados são comparados com a filtragem com conhecimento *a priori* do canal ($\mathbf{A}$ e $\tilde{\mathbf{\Gamma}}$ conhecidos) e a correlação direta sem ruído do sinal LOS com o banco correlator.

As simulações na Figura 1 incluem: o esquema DoA/KRF comparado com o autofiltro para $D = 1$ satélites e diferença de azimute $\Delta\phi$ de $\pi/3$, $\pi/4$, e $\pi/6$ nas subfiguras (a) a (c); o esquema DoA/KRF comparado com o autofiltro para $D = 2$ e $D = 3$ satélites para $\Delta\phi = \pi/3$ nas subfiguras (d) e (e); e o esquema DoA/KRF comparado com filtragem ProKRaft para $D = 1$ satélites e diferença de azimute $\Delta\phi$ de $\pi/3$, $\pi/4$, e $\pi/6$ nas subfiguras (f) a (h).



(a) DoA/KRF para $D = 1$, $\Delta\phi = \pi/3$

(b) DoA/KRF para $D = 1$, $\Delta\phi = \pi/4$

(c) DoA/KRF para $D = 1$, $\Delta\phi = \pi/6$

(d) DoA/KRF para $D = 2$, $\Delta\phi = \pi/3$

(e) DoA/KRF para $D = 3$, $\Delta\phi = \pi/3$

(f) ProKRaft para $D = 1$, $\Delta\phi = \pi/3$

(g) ProKRaft para $D = 1$, $\Delta\phi = \pi/4$

(h) ProKRaft para $D = 1$, $\Delta\phi = \pi/3$

Figura 1: Simulação dos esquemas tensoriais DoA/KRF e ProKRaft para estimação de atraso

Abstract

Traditionally Global Navigation Satellite Systems (GNSS) such as Global Positioning System (GPS), Galileo, also known as the European GNSS, GLONASS, also known as the Russian GNSS, or BeiDou, also known as the Chinese GNSS, are intended for military applications such as missile guiding-system and for takeoff and landing of civilian airplanes. In case of civilian aviation, Ground-Based Augmentation System (GBAS) supports local augmentation for safety-critical systems. However, in the last years, the amount of GNSS applications has dramatically increased. For instance, fishing authorities can use GNSS to automatically locate in real-time by satellites the fishing ships in order to guarantee the sustainable management of the fishing commonwealth [1]. Another application of GNSS is to track trucks in order to know the status of the load in real-time. In traffic related applications, GNSS can be used for an automatic toll system [2] and for autonomous vehicles that require high standards of security and precision. In the context of autonomous driving, the environment should be sensed by the vehicle and the measured data processed in order to achieve such standards. Although autonomous vehicles count on an extensive amount of sensors to perform Adaptive Cruise Control (ACC), GNSS receivers play an essential role [3] due to its ubiquity covering almost all areas of the planet. Finally, in precision farming, also known as satellite farming, GNSS is employed to improve, for instance, precision of fertilization and also to allow the usage of expensive agriculture vehicles 24 hours a day [4]. Therefore, with precision agriculture, both machinery and chemicals can be used in a safer and more efficient fashion.

GNSSs rely on time-delay estimation to estimate a user's position. This is done by correlating the incoming signal with replica sequences to separate each satellite and perform time-delay estimation. Since multipath components are delayed copies of the original signal, this affects the cross-correlation function, thus impacting time-delay estimation.

In this thesis, we study a state-of-the-art approach for multipath mitigation time-delay estimation algorithm based on the rank-one Higher-Order Singular Value Decomposition (HOSVD) eigenfilter [7], and propose two tensor-based schemes for multipath mitigation and time-delay estimation, for which the HOSVD-based scheme is a basis of comparison.

The first scheme is a three step tensor-based approach applying direction of arrival (DoA) estimation and Khatri-Rao factorization (KRF) to separate the code for each impinging component in a closed fashion. The second approach uses the dual-symmetry property of the

signal tensor [6] to calculate a Hermitian unfolding with which, by alternating between a solution to the orthogonal Procrustes problem (OPP) and least squares Khatri-Rao factorization, iteratively estimates the channel factor matrices which are then used to separate the code of each impinging component. Both our schemes outperforms the HOSVD-based eigenfilter state-of-the-art solution.

CONTENTS

LISTA OF FIGURES

Chapter 1

Introduction

Originally Global Navigation Satellite Systems (GNSS) such as Global Positioning System (GPS) were intended for military applications, such as missile guiding-systems, and for takeoff and landing of civilian airplanes. In case of civilian aviation, Ground-Based Augmentation System (GBAS) supports local augmentation for safety-critical systems. In the last years, the amount of GNSS applications has increased dramatically. For instance, GNSS can be used for automatic toll systems [2] and for autonomous vehicles that require high standards of security and precision [3]. In precision farming, GNSS is employed to improve, for instance, precision application of fertilization and also to allow the usage of expensive agriculture machinery 24 hours a day [4].

In order to compute the position on the earth, a GNSS receiver uses the time-delays of line-of-sight (LOS) components from at least four satellites. However, due to the geometry of the propagation environment, caused, for instance, by trees, poles, lamps and buildings, reflections from the LOS signal can occur creating multipath components, which are non-line-of-sight (NLOS) components. As a consequence, the multipath components interfere with the received LOS signal component. In practice, the quality of the ranging data provided by a GNSS receiver largely depends on the synchronization error, that is, on the accuracy of the propagation time-delay estimation of the LOS signal. In case the LOS signal is corrupted by several superimposed delayed replicas (reflective, diffractive, or refractive multipath), the estimation of the propagation time-delay and thus the position can be severely degraded using state-of-the-art GNSS receivers [8, 9, 10].

Several techniques have been proposed in the literature for solving the multipath problem in GNSS using one single-polarization antenna, e.g. [11, 12], but their capabilities are not sufficient for safety-critical applications (SCA) or liability critical applications (LCA). Thus, multi-antenna systems became the focus of research and technological development of multipath mitigation for SCA and LCA [13, 14]. The current state-of-the-art tensor-based multipath mitigation techniques applied to time-delay estimation [5] is based on HOSVD [15] eigenfiltering with Forward-Backward Averaging (FBA) [16, 17], and Expanded Spatial Smoothing (ESPS) [18, 19].

In this thesis we propose two new tensor-based time-delay estimation approach robust against multipath components. Our first approach starts by utilizing a direction of arrival (DoA) estimation technique on the post-correlated received tensor signals in order to reconstruct the DoA-related factor matrix. Next, the remaining factor matrices can be estimated by using Khatri-Rao factorization (KRF). Given the estimated factor matrix corresponding to the post-correlated pseudo-random (PR) sequences, the time-delay estimation can be performed for each LOS and NLOS signal component. Therefore, we also incorporate two proposed selection schemes in our framework in order to estimate the time-delay of the LOS signal component: one based on the signal power and another one selecting the smallest estimated time-delay. The second approach utilizes a Procrustes estimation and Khatri-Rao factorization (ProKRaft) technique to estimate the DoA- and complex amplitude-related factor matrices which are then used to filter and recover the post-correlated PR sequences. Again, a selection scheme is employed to estimate the time-delay of the LOS signal component.

This thesis is structured as follows: this chapter includes the introduction along with the notation used. Chapter 2 presents the notation, matrix and tensor operations, and the pre- and post-correlation data model. In Chapter 3, three approaches to time-delay estimation are presented: in Section 3.1 the state-of-the-art HOSVD eigenfilter-based approach with FBA and ESPS; in Section 3.2, we propose our time-delay estimation approach based on the direction of arrival estimation and Khatri-Rao factorization; and in Section 3.3 the ProKRaft approach. Chapter 4 presents the results of Monte Carlo (MC) simulations under different scenarios. Chapter 5 draws the conclusions.

Chapter 2

Concepts on Tensor Calculus and Data Model

While matrix operations such as eigenvalue and singular value decompositions are well known, certain specific matrix operations which are used in this thesis are more rarely seen and their usage is necessary for the tensor operations used for tensor-based filtering techniques and how the tensor-based GNSS receiver is modeled in this thesis. In this chapter, the notation used in this thesis is introduced in Section 2.1, followed by matrix operations in Section 2.2, the tensor operations and how they relate to the matrix operations in Section 2.3, and finally the data model for the tensor-based GNSS receiver before and after correlation with the correlator bank.

2.1 Notation

Scalars are represented by italic letters (a, b, A, B), vectors by lowercase bold letters $(\mathbf{a}, \mathbf{b})$, matrices by uppercase bold letters $(\mathbf{A}, \mathbf{B})$, and tensors by uppercase bold calligraphic letters $(\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}})$.

The superscripts $^{\mathrm{T}}$, *, $^{\mathrm{H}}$, $^{-1}$, and $^{+}$ denote the transpose, conjugate, conjugate transpose (Hermitian), inverse, and pseudo-inverse of a matrix, respectively.

For a matrix $\mathbf{A} \in \mathbb{C}^{M \times N}$, the element in the m-th row and n-th column is denoted by $a_{m,n}$, its m-th row is denoted by $(\mathbf{A})_{m,\cdot}$, and its n-th column is denoted by $(\mathbf{A})_{\cdot,n}$. The 2-norm of a matrix $\mathbf{A}$ is denoted by $\|\mathbf{A}\|_2$.

For a matrix $\mathbf{A} \in \mathbb{C}^{M \times N}$ with $M < N$, the $\operatorname{diag}\{\cdot\}$ operator extracts the diagonal is defined as

$$\operatorname{diag}\{\mathbf{A}\} \triangleq \begin{bmatrix} a_{1,1} \\ a_{2,2} \\ \vdots \\ a_{M,M} \end{bmatrix}. \tag{2.1}$$

The n-th mode unfolding of the tensor $\boldsymbol{\mathcal{A}}$ is denoted by $[\boldsymbol{\mathcal{A}}]_{(n)}$. The n-mode product between a tensor $\boldsymbol{\mathcal{A}}$ and a matrix B is denoted by $\boldsymbol{\mathcal{A}} \times_n \mathrm{B}$. The N-th order identity tensor of size $L \times \ldots \times L$ is denoted by $\boldsymbol{\mathcal{I}}_{N,L}$.

For two N-th order tensor $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}}$, both of size $I_1 \times I_2 \times \ldots \times I_N$, composed of individual scalar elements $a_{i_1,i_2,\ldots,i_N}$ and $b_{i_1,i_2,\ldots,i_N}$, respectively, its inner product is denoted by $\langle \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}} \rangle$, and is defined as

$$\langle \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}} \rangle \triangleq \sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \cdots \sum_{i_N=1}^{I_N} a_{i_1,i_2,\ldots,i_N} b_{i_1,i_2,\ldots,i_N}. \tag{2.2}$$

The norm of a tensor $\boldsymbol{\mathcal{A}}$, denoted by $||\boldsymbol{\mathcal{A}}||_{\mathrm{F}}$, is the Frobenius norm defined as

$$||\boldsymbol{\mathcal{A}}||_{\mathrm{F}} \triangleq \sqrt{\langle \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{A}} \rangle}. \tag{2.3}$$

2.2 Matrix Calculus

In this section four matrix operations are presented. The first two, the Kronecker and Khatri-Rao products are straightforward, the latter two have important properties that are used in the data model used in this work.

2.2.1 Kronecker product

Given two matrices $\mathbf{A} \in \mathbb{C}^{I \times J}$ and $\mathbf{B} \in \mathbb{C}^{K \times L}$ their Kronecker product, denoted by $\otimes$, is defined as:

$$\mathbf{A} \otimes \mathbf{B} \triangleq \begin{bmatrix} a_{1,1}\mathbf{B} & \cdots & a_{1,J}\mathbf{B} \\ \vdots & \ddots & \vdots \\ a_{I,1}\mathbf{B} & \cdots & a_{I,J}\mathbf{B} \end{bmatrix} \in \mathbb{C}^{IK \times JL}. \tag{2.4}$$

2.2.2 Khatri-Rao product

Given two matrices $\mathbf{A} \in \mathbb{C}^{I\times R}$ and $\mathbf{B} \in \mathbb{C}^{K\times R}$ their Khatri-Rao product, denoted by $\diamond$, is defined as:

$$\mathbf{A} \diamond \mathbf{B} \triangleq \begin{bmatrix}(\mathbf{A})_{\cdot,1} \otimes (\mathbf{B})_{\cdot,1} & \cdots & (\mathbf{A})_{\cdot,R} \otimes (\mathbf{B})_{\cdot,R}\end{bmatrix} \in \mathbb{C}^{IJ\times R}. \tag{2.5}$$

2.2.3 Outer product

The outer product is a special case of the Kronecker product in which the outer product of two vectors $\mathbf{a} \in \mathbb{C}^{I}$ and $\mathbf{b} \in \mathbb{C}^{J}$ results in a matrix $\mathbf{C} \in \mathbb{C}^{I\times J}$:

$$\mathbf{a} \circ \mathbf{b} = \mathbf{a}\mathbf{b}^{\mathrm{T}} = \begin{bmatrix} a_1 \\ \vdots \\ a_I \end{bmatrix} \begin{bmatrix} b_1 & \cdots & b_J \end{bmatrix} \tag{2.6}$$

$$= \begin{bmatrix} a_1 b_1 & \cdots & a_1 b_J \\ \vdots & \ddots & \vdots \\ a_I b_1 & \cdots & a_I b_J \end{bmatrix} = \mathbf{C} \in \mathbb{C}^{I\times J}, \tag{2.7}$$

such that the elements of $\mathbf{C}$ satisfy $c_{ij} = a_i b_j, i \in \{1, \dots, I\}, j \in \{1, \dots, J\}$.

The outer product can also be extended into other dimensions. An outer product of three vectors results in a third-order tensor. For example, the outer product of three vectors $\mathbf{a} \in \mathbb{C}^{I}$, $\mathbf{b} \in \mathbb{C}^{J}$, and $\mathbf{c} \in \mathbb{C}^{K}$ results in a third-order tensor $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{I\times J\times K}$

$$\mathbf{a} \circ \mathbf{b} \circ \mathbf{c} = \boldsymbol{\mathcal{X}} \in \mathbb{C}^{I\times J\times K}, \tag{2.8}$$

and $x_{ijk} = a_i b_j c_k, i \in \{1, \dots, I\}, j \in \{1, \dots, J\}, k \in \{1, \dots, K\}$ holds.

2.2.4 The $\mathrm{vec}\{\cdot\}$ operator

The $\mathrm{vec}\{\cdot\}$ operator rearranges a matrix into a vector in such a way that its vectors are stacked. For a matrix $\mathbf{A} \in \mathbb{C}^{M\times N}$,

$$\mathrm{vec}\{\mathbf{A}\} = \mathrm{vec}\left\{\begin{bmatrix}\mathbf{A}_{\cdot 1} & \cdots & \mathbf{A}_{\cdot N}\end{bmatrix}\right\} \tag{2.9}$$

$$= \begin{bmatrix} \mathbf{A}_{\cdot 1} \\ \vdots \\ \mathbf{A}_{\cdot N} \end{bmatrix} \in \mathbb{C}^{MN}. \tag{2.10}$$

An important property of the $\mathrm{vec}\{\cdot\}$ operator is that for $\mathbf{X} = \mathbf{ABC}$ with $\mathbf{A} \in \mathbb{C}^{I\times J}$, a diagonal matrix $\mathbf{B} \in \mathbb{C}^{J \times J}$, and $\mathbf{C} \in \mathbb{C}^{J \times K}$

$$\begin{aligned} \mathrm{vec}\{\mathbf{X}\} &= \mathrm{vec}\{\mathbf{ABC}\} \\ &= (\mathbf{C}^{\mathrm{T}} \diamond \mathbf{A})\mathrm{diag}\{\mathbf{B}\} \in \mathbb{C}^{IK}. \end{aligned} \tag{2.11}$$

2.2.5 The $\mathrm{unvec}\{\cdot\}$ operator

The $\mathrm{unvec}\{\cdot\}$ operator rearranges a vector into a matrix of specified size. For a vector $\mathbf{a} = [\mathbf{a}_1^{\mathrm{T}}, \ldots, \mathbf{a}_N^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{C}^{MN}$,

$$\begin{aligned} \underset{M \times N}{\mathrm{unvec}}\{\mathbf{a}\} &= \underset{M \times N}{\mathrm{unvec}}\left\{\begin{bmatrix} \mathbf{a}_1 \\ \vdots \\ \mathbf{a}_N \end{bmatrix}\right\} \\ &= \begin{bmatrix} \mathbf{a}_1 & \cdots & \mathbf{a}_N \end{bmatrix} \in \mathbb{C}^{M \times N}. \end{aligned} \tag{2.12}$$

2.3 Tensor Calculus

2.3.1 Tensors

As vectors are generalizations of scalars, and matrices generalizations of vectors, tensors are generalizations of matrices but, while matrices are limited to only two dimensions, tensors can have any number of dimensions. Throughout this text the terms scalar, vector and matrix are applied to 0-, 1-, and 2-dimensional structures while the term tensor is only applied to structures with 3 or more dimensions.

In (2.13) we examplify a scalar $I \in \mathbb{C}$, a vector $\mathbf{i} \in \mathbb{C}^3$, and an identity matrix $\mathbf{I} \in \mathbb{C}^{3 \times 3}$:

$$I = 1, \qquad \mathbf{i} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \qquad \mathbf{I} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{2.13}$$

while in (2.14) a third-order identity tensor $\boldsymbol{\mathcal{I}}_{3,3} \in \mathbb{C}^{3\times3\times3}$ is shown:

$$\boldsymbol{\mathcal{I}}_{3,3} = \begin{array}{ccc} & & \begin{bmatrix}0&0&0\\0&0&0\\0&0&1\end{bmatrix} \\ & \begin{bmatrix}0&0&0\\0&1&0\\0&0&0\end{bmatrix} & \\ \begin{bmatrix}1&0&0\\0&0&0\\0&0&0\end{bmatrix} & & \end{array} . \tag{2.14}$$

While higher-order tensors can be achieved visualization becomes difficult. A N-dimensional tensor $\boldsymbol{\mathcal{A}} \in \mathbb{C}^{I_1\times I_2\times\ldots\times I_N}$ can be perceived in "slices" by keeping its first two indexes fixed while varying the remaining $N-2$ indexes in succession.

For example, by changing the third index of the previously used third-order identity tensor $\boldsymbol{\mathcal{I}}_{3,3}$ while fixing the first and second indexes:

$$(\boldsymbol{\mathcal{I}}_{3,3})_{\cdot,\cdot,1} = \begin{bmatrix}1&0&0\\0&0&0\\0&0&0\end{bmatrix} \quad (\boldsymbol{\mathcal{I}}_{3,3})_{\cdot,\cdot,2} = \begin{bmatrix}0&0&0\\0&1&0\\0&0&0\end{bmatrix} \quad (\boldsymbol{\mathcal{I}}_{3,3})_{\cdot,\cdot,3} = \begin{bmatrix}1&0&0\\0&0&0\\0&0&1\end{bmatrix} . \tag{2.15}$$

For a fourth-order identity tensor $\boldsymbol{\mathcal{I}}_{4,3} \in \mathbb{C}^{3\times3\times3\times3}$ we get

$$(\boldsymbol{\mathcal{I}}_{4,3})_{\cdot,\cdot,1,1} = \begin{bmatrix}1&0&0\\0&0&0\\0&0&0\end{bmatrix} \quad (\boldsymbol{\mathcal{I}}_{4,3})_{\cdot,\cdot,2,1} = \begin{bmatrix}0&0&0\\0&0&0\\0&0&0\end{bmatrix} \quad (\boldsymbol{\mathcal{I}}_{4,3})_{\cdot,\cdot,3,1} = \begin{bmatrix}0&0&0\\0&0&0\\0&0&0\end{bmatrix} \tag{2.16}$$

$$(\boldsymbol{\mathcal{I}}_{4,3})_{\cdot,\cdot,1,2} = \begin{bmatrix}0&0&0\\0&0&0\\0&0&0\end{bmatrix} \quad (\boldsymbol{\mathcal{I}}_{4,3})_{\cdot,\cdot,2,2} = \begin{bmatrix}0&0&0\\0&1&0\\0&0&0\end{bmatrix} \quad (\boldsymbol{\mathcal{I}}_{4,3})_{\cdot,\cdot,3,2} = \begin{bmatrix}0&0&0\\0&0&0\\0&0&0\end{bmatrix} \tag{2.17}$$

$$(\boldsymbol{\mathcal{I}}_{4,3})_{\cdot,\cdot,1,3} = \begin{bmatrix}0&0&0\\0&0&0\\0&0&0\end{bmatrix} \quad (\boldsymbol{\mathcal{I}}_{4,3})_{\cdot,\cdot,2,3} = \begin{bmatrix}0&0&0\\0&0&0\\0&0&0\end{bmatrix} \quad (\boldsymbol{\mathcal{I}}_{4,3})_{\cdot,\cdot,3,3} = \begin{bmatrix}0&0&0\\0&0&0\\0&0&1\end{bmatrix} . \tag{2.18}$$

2.3.2 n-mode unfolding

The n-mode unfolding provides a way to represent a tensor as a matrix. This is done by fixing the n-th index while varying the remaining indexes in reverse order, concatenating these vectors along the $n+1$-th dimension, then permutating the order of the dimensions (circularly) from the n-th to the $n-1$-th dimension.

For example, for a third-order tensor $\boldsymbol{\mathcal{A}} \in \mathbb{C}^{2\times2\times2}$ we can write

$$\boldsymbol{\mathcal{A}} \quad = \quad \begin{array}{c} \quad\begin{bmatrix}5 & 6\\ 7 & 8\end{bmatrix} \\ \begin{bmatrix}1 & 2\\ 3 & 4\end{bmatrix}\quad \end{array} . \tag{2.19}$$

and has the following unfoldings:

$$[\boldsymbol{\mathcal{A}}]_{(1)} = \begin{bmatrix}1 & 5 & 2 & 6\\ 3 & 7 & 4 & 8\end{bmatrix}, \tag{2.20}$$

$$[\boldsymbol{\mathcal{A}}]_{(2)} = \begin{bmatrix}1 & 3 & 5 & 7\\ 2 & 4 & 6 & 8\end{bmatrix}, \tag{2.21}$$

$$[\boldsymbol{\mathcal{A}}]_{(3)} = \begin{bmatrix}1 & 2 & 3 & 4\\ 5 & 6 & 7 & 8\end{bmatrix}. \tag{2.22}$$

For a N-dimensional tensor, $\boldsymbol{\mathcal{A}} \in \mathbb{C}^{I_1\times\ldots\times I_N}$, its n-mode unfolding, $[\boldsymbol{\mathcal{A}}]_{(n)}$, will be of size $I_n \times \Pi_{r\neq n} I_r$.

2.3.3 n-mode product

The n-mode product allows for the calculation of the product between a matrix and a tensor by using the n-mode unfolding. For an N-dimensional tensor $\boldsymbol{\mathcal{A}} \in \mathbb{C}^{I_1\times\ldots\times I_n\times\ldots\times I_N}$ and a matrix $\mathbf{B} \in \mathbb{C}^{M\times I_n}$, the n-mode product of these two matrices is denoted by $\boldsymbol{\mathcal{A}} \times_n \mathbf{B}$. The resulting tensor is the matrix product of $\mathbf{B} \cdot [\boldsymbol{\mathcal{A}}]_{(n)}$ folded back into a tensor of size $I_1 \times \ldots \times M \times \ldots \times I_N$.

Alternatively, this can be interpreted as the n-mode unfolding of the resulting tensor as the product between the matrix $\mathbf{B}$ and the unfolded tensor $\boldsymbol{\mathcal{A}}$:

$$[\boldsymbol{\mathcal{A}} \times_n \mathbf{B}]_{(n)} = \mathbf{B} \cdot [\boldsymbol{\mathcal{A}}]_{(n)}. \tag{2.23}$$

2.3.4 PARAFAC model

The PARAFAC model presupposes that a given N-dimensional tensor $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{I_1\times\ldots\times I_N}$ can be decomposed into a summation of a minimum number of rank-one tensors $\boldsymbol{\mathcal{X}}^{(i)}, i = 1, \ldots, L$:

$$\boldsymbol{\mathcal{X}} = \sum_{l=1}^{L} \boldsymbol{\mathcal{X}}^{(l)} = \sum_{l=1}^{L} \mathbf{a}_l^{(1)} \circ \ldots \circ \mathbf{a}_l^{(N)}, \tag{2.24}$$

and L is the model order of the noiseless tensor.

By defining factor matrices $\mathbf{A}^{(i)} = [\mathbf{a}_1^{(i)}, \dots, \mathbf{a}_1^{(i)}]$ the equation above can be written in terms of the n-mode products of an N-dimensional identity matrix $\boldsymbol{\mathcal{I}}_{N,L} \in \mathbb{R}^{L \times \dots \times L}$ and loading matrices $\mathbf{A}^{(i)}$:

$$\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{I}}_{N,L} \times_1 \mathbf{A}^{(1)} \times_2 \mathbf{A}^{(2)} \dots \times_N \mathbf{A}^{(N)}. \tag{2.25}$$

A related useful property is, for a third-order tensor $\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{I}}_{3,d} \times_1 \mathbf{A}^{(1)} \times_2 \mathbf{A}^{(2)} \times_3 \mathbf{A}^{(3)} \in \mathbb{C}^{I_1 \times I_2 \times I_3}$ with $\mathbf{A}^{(1)} \in \mathbb{C}^{I_1 \times d}$, $\mathbf{A}^{(2)} \in \mathbb{C}^{I_2 \times d}$, and $\mathbf{A}^{(3)} \in \mathbb{C}^{I_3 \times d}$, its unfoldings are:

$$[\boldsymbol{\mathcal{A}}]_{(1)} = \mathbf{A}^{(1)} \left(\mathbf{A}^{(2)} \diamond \mathbf{A}^{(3)}\right)^{\mathrm{T}} \in \mathbb{C}^{I_1 \times I_2 \cdot I_3}, \tag{2.26}$$

$$[\boldsymbol{\mathcal{A}}]_{(2)} = \mathbf{A}^{(2)} \left(\mathbf{A}^{(3)} \diamond \mathbf{A}^{(1)}\right)^{\mathrm{T}} \in \mathbb{C}^{I_2 \times I_3 \cdot I_1}, \tag{2.27}$$

$$[\boldsymbol{\mathcal{A}}]_{(3)} = \mathbf{A}^{(3)} \left(\mathbf{A}^{(1)} \diamond \mathbf{A}^{(2)}\right)^{\mathrm{T}} \in \mathbb{C}^{I_3 \times I_1 \cdot I_2}. \tag{2.28}$$

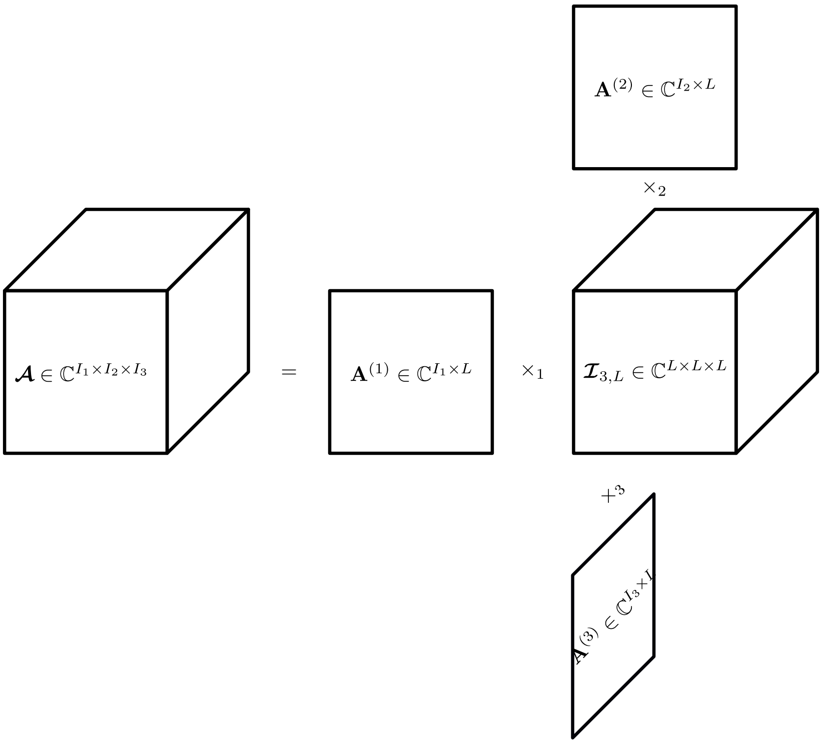

$\mathbf{A}^{(2)} \in \mathbb{C}^{I_2 \times L}$
$\times_2$
$\boldsymbol{\mathcal{A}} \in \mathbb{C}^{I_1 \times I_2 \times I_3}$
=
$\mathbf{A}^{(1)} \in \mathbb{C}^{I_1 \times L}$
$\times_1$
$\boldsymbol{\mathcal{I}}_{3,L} \in \mathbb{C}^{L \times L \times L}$
$\times_3$
$\mathbf{A}^{(3)} \in \mathbb{C}^{I_3 \times L}$

Figure 2.1: Third-order tensor PARAFAC decomposition

2.3.5 Higher-Order SVD

A matrix $\mathbf{X} \in \mathbb{C}^{I_1 \times I_2}$ can be decomposed by the Singular Value Decomposition into the following products:

$$\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{\mathrm{H}}, \tag{2.29}$$

with $\mathbf{U} \in \mathbb{C}^{I_1 \times I_1}$, $\boldsymbol{\Sigma} \in \mathbb{C}^{I_1 \times I_2}$, and $\mathbf{V} \in \mathbb{C}^{I_2 \times I_2}$.

$\mathbf{U}$ is a unitary matrix containing the left-hand singular vectors and its columns are associated with the column space of $\mathbf{X}$. $\mathbf{V}$ is a unitary matrix containing the right-hand singular vectors and its rows are associated with the row space of $\mathbf{X}$. $\mathbf{\Sigma}$ is a matrix containing the singular values $\sigma_1, \sigma_2, \ldots, \sigma_{\min(I_1, I_2)}$ in its diagonal, its columns are multiplied by $\mathbf{U}$ and its rows are multiplied by $\mathbf{V}^*$.

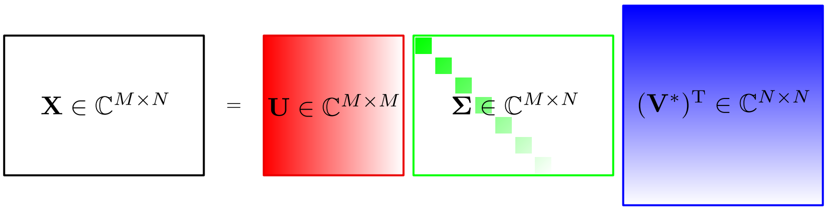

$\mathbf{X} \in \mathbb{C}^{M \times N}$
$=$
$\mathbf{U} \in \mathbb{C}^{M \times M}$
$\mathbf{\Sigma} \in \mathbb{C}^{M \times N}$
$(\mathbf{V}^*)^{\mathrm{T}} \in \mathbb{C}^{N \times N}$

Figure 2.2: Singular Value Decomposition

To generalize the SVD to an N-th order tensor the relations described above have to be retained while extending to tensors. For more than two dimensions the transpose operation does not make sense. To achieve the SVD without relying on the transpose operation while preserving the relations between $\mathbf{U}$, $\mathbf{V}$, and $\mathbf{\Sigma}$, the n-mode product described previously can be employed:

$$\begin{aligned} \mathbf{X} &= \mathbf{U \Sigma V}^{\mathrm{H}} \\ &= \mathbf{\Sigma} \times_1 \mathbf{U} \times_2 \mathbf{V}^* \\ &= \mathbf{\Sigma} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)}, \end{aligned} \tag{2.30}$$

with $\mathbf{U}^{(1)} = \mathbf{U}$ and $\mathbf{U}^{(2)} = \mathbf{V}^*$. $\mathbf{\Sigma}$ possesses the properties of *pseudodiagonality*, that is only its diagonal is non-zero, and *ordering*, in which $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_{\min(I_1, I_2)} \geq 0$.

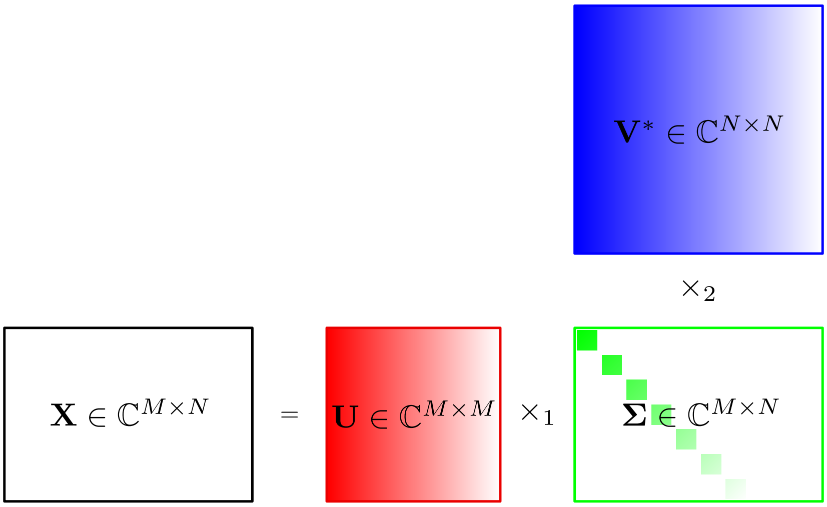

$\mathbf{V}^* \in \mathbb{C}^{N \times N}$
$\times_2$
$\mathbf{X} \in \mathbb{C}^{M \times N}$
$=$
$\mathbf{U} \in \mathbb{C}^{M \times M}$
$\times_1$
$\mathbf{\Sigma} \in \mathbb{C}^{M \times N}$

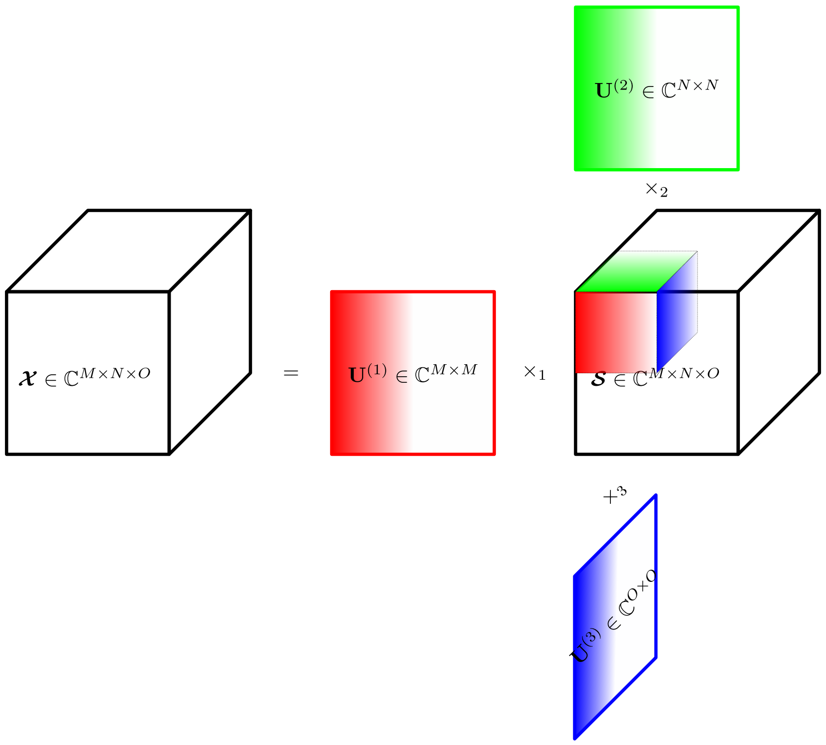
Figure 2.3: SVD generalized using the n-mode product

Generalizing the SVD to an N-th order tensor $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{I_1 \times I_2 \times \ldots \times I_N}$ in terms of an n-mode product:

$$\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{R}} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \ldots \times_N \mathbf{U}^{(N)}, \tag{2.31}$$

in which each $\mathbf{U}^{(n)} \in \mathbb{C}^{I_n \times I_n}$ is a unitary matrix that makes linear combinations of the subtensor formed by keeping the n-th index of $\boldsymbol{\mathcal{R}} \in \mathbb{C}^{I_1 \times I_2 \times \ldots \times I_N}$ fixed. This is known as the Higher-Order SVD [15].

The tensor $\boldsymbol{\mathcal{R}} \in \mathbb{C}^{I_1 \times I_2 \times \ldots \times I_N}$ is known as "core tensor" and has the properties of *all-orthogonality*, that is, for two subtensors $\boldsymbol{\mathcal{R}}_{i_n=\alpha}$ and $\boldsymbol{\mathcal{R}}_{i_n=\beta}$ formed by keeping the index i_n fixed, their inner product $\langle \boldsymbol{\mathcal{R}}_{i_n=\alpha}, \boldsymbol{\mathcal{R}}_{i_n=\beta} \rangle = 0$ for $\alpha \neq \beta$, and *ordering*, though in this case conforming to the Frobenius norm of the subtensor formed by the fixation of each i_n-th index, that is, $\|\boldsymbol{\mathcal{R}}_{i_n=1}\|_\mathrm{F} \geq \|\boldsymbol{\mathcal{R}}_{i_n=2}\|_\mathrm{F} \geq \ldots \geq \|\boldsymbol{\mathcal{R}}_{i_n=I_n}\|_\mathrm{F} \geq 0$.



$\mathbf{U}^{(2)} \in \mathbb{C}^{N \times N}$
$\times_2$
$\boldsymbol{\mathcal{X}} \in \mathbb{C}^{M \times N \times O}$
$=$
$\mathbf{U}^{(1)} \in \mathbb{C}^{M \times M}$
$\times_1$
$\boldsymbol{\mathcal{S}} \in \mathbb{C}^{M \times N \times O}$
$\times_3$
$\mathbf{U}^{(3)} \in \mathbb{C}^{O \times O}$

Figure 2.4: Higher-Order SVD of a third-order tensor $\boldsymbol{\mathcal{X}}$

Calculation of the HOSVD can be achieved by finding each unitary left singular vector matrix $\mathbf{U}^{(n)}$ from its respective n-mode unfolding by applying the SVD to $[\boldsymbol{\mathcal{X}}]_{(n)}$. Then the core tensor $\boldsymbol{\mathcal{R}}$ can then be calculated by applying (2.31) using the n-mode products of $\mathbf{U}^{(n)}$ from the left-hand side since these are unitary:

$$\boldsymbol{\mathcal{R}} = \boldsymbol{\mathcal{X}} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \ldots \times_N \mathbf{U}^{(N)}. \tag{2.32}$$

2.3.6 Dual-symmetric tensors

A $2N$-th order tensor $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{I_1 \times \ldots \times I_N \times I_{N+1} \times \ldots I_{2N}}$ is dual-symmetric if and only if there can be a permutation of indexes P, resulting in a tensor $\boldsymbol{\mathcal{X}}_P$ which follows the particular PARAFAC decomposition [20]:

$$\boldsymbol{\mathcal{X}}_P = \boldsymbol{\mathcal{I}}_{2N,L} \times_1 \mathbf{A}^{(1)} \ldots \times_N \mathbf{A}^{(N)} \times_{N+1} (\mathbf{A}^{(1)})^* \ldots \times_{2N} (\mathbf{A}^{(N)})^*. \tag{2.33}$$

This decomposition is useful in signal processing because every correlation tensor follows this decomposition [21]. To exploit the relation in (2.33), a particular unfolding called Hermitian-symmetric unfolding [17] applied to the dual-symmetric tensor. The Hermitian-symmetric unfolding of $\boldsymbol{\mathcal{X}} \in \mathbb{C}^{I_1 \times \ldots \times I_{2N}}$, $\mathbf{X}_{\mathrm{H}}$ is defined as:

$$\mathbf{X}_{\mathrm{H}} = \underset{K \times K}{\operatorname{unvec}} \left\{ \operatorname{vec}\{\boldsymbol{\mathcal{X}}\} \right\} \in \mathbb{C}^{K \times K}, \tag{2.34}$$

with $K = I_1 \cdot \ldots \cdot I_N$.

This unfolding in terms of its factor matrices is:

$$\mathbf{X}_{\mathrm{H}} = \left(\mathbf{A}^{(N)} \diamond \ldots \diamond \mathbf{A}^{(1)} \right) \left(\mathbf{A}^{(N)} \diamond \ldots \diamond \mathbf{A}^{(1)} \right)^{\mathrm{H}}. \tag{2.35}$$

2.4 Data Model

GNSS use satellite constellations that orbit the earth in a known pattern, along with a terrestrial segment that monitors each satellite's individual position, issues corrections, and uploads the ephemeris data of each satellite which is used by the end-user to calculate his position on earth.

GNSSs such as GPS, Beidou, and Galileo use Code Division Multiple Access (CDMA) which not only gives each satellite a unique identification to separate it from other satellites, it also spreads the signal over a wider bandwidth, this is called spreading. Spreading is done by multiplying the transmitted data sequence by a higher bandwidth periodic chip (instead of bit) code called a pseudorandom (PR) sequence which increases the bandwidth of the signal while decreasing its spectral power density. Spreading allows satellites to operate over the same frequency by separating each satellite, as is done with 3G mobile users, where each satellite has its own unique PR sequence. Because the signal is now spread over a wider bandwidth it becomes more robust to interference, unfortunately it also decreases the signal to noise ratio (SNR) at which it is received, for example, in the case of GPS the signal is received below the noise floor. The pre-correlation data model in Subsection 2.4.1 models this signal being received by an antenna array.

Recovery of each satellite's individual signal is done by de-spreading. This is done by correlating the received signal by each known PR sequence. This process provides a processing gain, in the case of GPS of around 30 dB, and each satellite's signal can be recovered. The post-correlation data model in Subsection 2.4.2 models this signal.

Aside from separability and robustness, since spreading relies on a known repeating sequence, it is possible to estimate the delay between the time the signal was transmitted and the time it arrived at the user. Time-delay estimation provides the user information about his distance to each satellite which, when combined with received ephemeris data, allows for calculation of the user's position.

Since estimation of the time-delay relies on the cross-correlation function of the received PR sequence and its corresponding replica, multipath propagation can be particularly pernicious for GNSS relying on CDMA since multipath propagation consists of delayed copies of the original signal, which will distort the cross-correlation function, thus degrading time-delay estimation. Because of this, multipath mitigation is very important to GNSS. Mitigating multipath to perform time-delay estimation is seen in Chapter 3.

This section introduces the pre-correlation data model in Subsection 2.4.1, 2.4.2, which is the result of the pre-correlation signal tensor being multiplied by a compressed correlator bank [22] which separates the signal and returns the cross-correlation function. The data model used is based on [5].

2.4.1 Pre-correlation data model

Assuming an antenna array based GNSS receiver with M elements and assuming D GNSS satellites, where the d-th satellite has 1 LOS signal component and $L_d - 1$ NLOS signal components, the received signal can be modeled as follows

$$\mathbf{s}_{d,l_d}(t) = \mathbf{a}(\phi_{d,l_d})\gamma_{l_d} c_d(t - \tau_{d,l_d}), \tag{2.36}$$

with $L = \sum_{d=1}^{D} L_d$, and $\mathbf{s}_{d,l_d} \in \mathbb{C}^M$ contains the desired LOS signal for $l_d = 1$ along with the NLOS multipath signal components for $l_d = 2, \ldots, L_d$, and

$$\mathbf{s}_{d,l_d}(t) = \mathbf{a}(\phi_{d,l_d})\gamma_{l_d} c_d(t - \tau_{d,l_d}), \tag{2.37}$$

is a signal replica with its own steering vector $\mathbf{a}(\phi_{d,l_d})$, complex amplitude γ_{d,l_d}, and PR sequence $c_d(t - \tau_{d,l_d})$ with delay τ_{d,l_d}. The time index $t = 1, \ldots, N$.

Each PR sequence with N samples is spatially observed in the M receive antennas and are temporally grouped into K epochs. Therefore, the spatially observed matrix of the k-th epoch, for $k = 1, \ldots, K$, is given by

$$\mathbf{X}[k] = \Big[\mathbf{x}[(k-1)N + 1] \quad \cdots \quad \mathbf{x}[(k-1)N + N]\Big], \tag{2.38}$$

with $\mathbf{X}[k] \in \mathbb{C}^{M \times N}$.

Concatenating all the steering vectors of the k-th period into a matrix $\mathbf{A}_d[k] = [\mathbf{a}(\phi_{d,1}), \ldots, \mathbf{a}(\phi_{d,L_d})] \in \mathbb{C}^{M \times L_d}$, the complex amplitudes $\boldsymbol{\gamma}_d = [\gamma_{d,1}, \ldots, \gamma_{d,L_d}]^{\mathrm{T}}$ into a diagonal matrix $\boldsymbol{\Gamma}_d[k] = \mathrm{diag}\{\boldsymbol{\gamma}_d\} \in \mathbb{C}^{L_d \times L_d}$ and the sampled and shifted PR sequences into a matrix $\mathbf{C}_d[k] = \big[\mathbf{c}_d[\tau_{d,1}], \ldots, \mathbf{c}_d[\tau_{d,L_d}]\big]^{\mathrm{T}} \in \mathbb{R}^{L_d \times N}$ where each $\mathbf{c}_d[\tau]$ is a sampled PR sequence (of the d-th satellite) with delay τ, the signal can be written in a matrix notation:

$$\begin{aligned}\mathbf{X}[k] &= \sum_{d=1}^{D} \mathbf{A}_d[k]\boldsymbol{\Gamma}_d[k]\mathbf{C}_d[k] + \mathbf{N}[k] \\ &= \mathbf{A}[k]\boldsymbol{\Gamma}[k]\mathbf{C}[k] + \mathbf{N}[k],\end{aligned} \tag{2.39}$$

with $\mathbf{A}[k] \in \mathbb{C}^{M\times L}$, $\boldsymbol{\Gamma}[k] \in \mathbb{C}^{L\times L}$, $\mathbf{C}[k] \in \mathbb{R}^{L\times N}$, and $\mathbf{N}[k] \in \mathbb{C}^{M\times N}$.

Applying the vec$\{\cdot\}$ operator on $\mathbf{X}[k]$ to reshape it into a vector $\tilde{\mathbf{x}}[k]$, we obtain the following expression in terms of Khatri-Rao product:

$$\begin{aligned}\tilde{\mathbf{x}}[k] &= \text{vec}\{\mathbf{X}[k]\} = \text{vec}\{\mathbf{A}[k]\boldsymbol{\Gamma}[k]\mathbf{C}[k] + \mathbf{N}[k]\} \\ &= (\mathbf{C}[k]^{\mathrm{T}} \diamond \mathbf{A}[k])\boldsymbol{\gamma}[k] + \tilde{\mathbf{n}}[k].\end{aligned} \tag{2.40}$$

Concatenating all K epochs along several columns, the index k can be dropped and the following matrix representation can be obtained:

$$\begin{aligned}\tilde{\mathbf{X}} &= (\mathbf{C}^{\mathrm{T}} \diamond \mathbf{A})\tilde{\boldsymbol{\Gamma}} + \tilde{\mathbf{N}} \\ &= \tilde{\mathbf{X}}_0 + \tilde{\mathbf{N}},\end{aligned} \tag{2.41}$$

in which $\tilde{\mathbf{X}}_0 \in \mathbb{C}^{MN\times K}$ is the noiseless received signal matrix and $\tilde{\boldsymbol{\Gamma}} = \big[\boldsymbol{\gamma}[1], \ldots, \boldsymbol{\gamma}[K]\big] \in \mathbb{C}^{L\times K}$ stacks the complex amplitudes of each epoch. Note that the transpose of $\tilde{\mathbf{X}}_0$ follows the same structure as the first-mode unfolding of a noiseless received signal tensor $\boldsymbol{\mathcal{X}}_0$:

$$\tilde{\mathbf{X}}_0^{\mathrm{T}} = \tilde{\boldsymbol{\Gamma}}^{\mathrm{T}}(\mathbf{C}^{\mathrm{T}} \diamond \mathbf{A})^{\mathrm{T}} = [\boldsymbol{\mathcal{X}}_0]_{(1)}. \tag{2.42}$$

By folding the matrix $\tilde{\mathbf{X}}_0$ into the tensor form, we obtain

$$\boldsymbol{\mathcal{X}}_0 = \boldsymbol{\mathcal{I}}_{3,L} \times_1 \tilde{\boldsymbol{\Gamma}}^{\mathrm{T}} \times_2 \mathbf{C}^{\mathrm{T}} \times_3 \mathbf{A}, \tag{2.43}$$

in which $\boldsymbol{\mathcal{I}}_{3,L} \in \mathbb{R}^{L\times L\times L}$ is the third-order identity tensor and $\boldsymbol{\mathcal{X}}_0 \in \mathbb{C}^{K\times N\times M}$ is the noiseless received signal tensor.

Therefore, the matrix representation in (2.41) is equivalent to the following tensor expression:

$$\begin{aligned}\boldsymbol{\mathcal{X}} &= \boldsymbol{\mathcal{I}}_{3,L} \times_1 \tilde{\boldsymbol{\Gamma}}^{\mathrm{T}} \times_2 \mathbf{C}^{\mathrm{T}} \times_3 \mathbf{A} + \boldsymbol{\mathcal{N}} \\ &= \boldsymbol{\mathcal{X}}_0 + \boldsymbol{\mathcal{N}}.\end{aligned} \tag{2.44}$$

Note that the tensor in (2.44) has three dimensions, being two temporal dimensions (epochs and signal samples) and one spatial dimension. The first dimension of size K is associated with each epoch, the second dimension of size N corresponds to the collected samples in each epoch, and the third dimension of size M is related to the spatial diversity of the receive antenna array.

2.4.2 Post-correlation data model

As shown in [5], the received signal $\boldsymbol{\mathcal{X}}$ is multiplied by a correlator bank $\mathbf{Q}_d$ of the d-th satellite to calculate the cross-correlation vector used to estimate the time-delay of the LOS signal component, since each satellite has its own PR sequence. In practice, the correlator bank $\mathbf{Q}_d$ is a collection of Q shifted signal replicas, or taps, of the PR sequence $\mathbf{c}_d \in \mathbb{R}^N$ with delay $\tau_q, q = 1, \ldots, Q$:

$$\mathbf{Q}_d = \begin{bmatrix} \mathbf{c}_d[\tau_1] & \cdots & \mathbf{c}_d[\tau_Q] \end{bmatrix} \in \mathbb{R}^{N \times Q}. \tag{2.45}$$

Hence, the received signal of each k-th epoch according to (2.39) multiplied by the correlator bank $\mathbf{Q}_d$ is given by

$$\begin{aligned} \mathbf{Y}_d[k] &= \mathbf{X}[k]\mathbf{Q}_d \\ &= \mathbf{A}_d\boldsymbol{\Gamma}_d[k]\mathbf{C}_d\mathbf{Q}_d + \sum_{i \neq d} \mathbf{A}_i\boldsymbol{\Gamma}_i[k]\mathbf{C}_i\mathbf{Q}_d + \mathbf{N}[k]\mathbf{Q}_d \\ &\approx \mathbf{A}_d\boldsymbol{\Gamma}_d[k]\mathbf{C}_d\mathbf{Q}_d + \mathbf{N}[k]\mathbf{Q}_d \in \mathbb{C}^{M \times Q}, \end{aligned} \tag{2.46}$$

since the signal components from other satellites are nearly removed [23]. However, the noise in (2.46) becomes colored. In order to overcome this, a Fisher information-preserving compression [22] is applied to the d-th correlator bank by using the economy-size Singular Value Decomposition (SVD):

$$\mathbf{Q}_d = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{\mathrm{H}}, \tag{2.47}$$

with $\mathbf{U} \in \mathbb{C}^{N \times Q}$, $\boldsymbol{\Sigma} \in \mathbb{C}^{Q \times Q}$, and $\mathbf{V} \in \mathbb{C}^{Q \times Q}$. By defining the compressed correlator bank as $\mathbf{Q}_{\omega,d} = \mathbf{U}$, since $\mathbf{U}$ is an orthogonal and unitary matrix, preserves the statistical properties of the noise (see Appendix).

Therefore, the improved post-correlation model is given by

$$\begin{aligned} \bar{\mathbf{Y}}_d[k] &= \mathbf{A}_d\boldsymbol{\Gamma}_d[k]\mathbf{C}_d\mathbf{Q}_{\omega,d} + \mathbf{N}[k]\mathbf{Q}_{\omega,d} \\ &= \mathbf{A}_d\boldsymbol{\Gamma}_d[k]\mathbf{C}_d\mathbf{Q}_{\omega,d} + \mathbf{N}_\omega[k], \end{aligned} \tag{2.48}$$

in which $\mathbf{N}_\omega[k]$ is white Gaussian noise.

Similarly as performed from (2.41) to (2.44), we can also rewrite (2.48) into the following tensor fashion by using the n-mode product operator:

$$\boldsymbol{\mathcal{Y}}_d = \boldsymbol{\mathcal{X}} \times_2 \mathbf{Q}_{\omega,d}^{\mathrm{T}} \in \mathbb{C}^{K \times Q \times M}, \tag{2.49}$$

or, equivalently,

$$\begin{aligned}\boldsymbol{\mathcal{Y}}_d &= \boldsymbol{\mathcal{I}}_{3,L} \times_1 \tilde{\boldsymbol{\Gamma}}_d \times_2 (\mathbf{C}_d \mathbf{Q}_{\omega,d})^{\mathrm{T}} \times_3 \mathbf{A}_d + \boldsymbol{\mathcal{N}}_\omega \\ &= \boldsymbol{\mathcal{Y}}_{0,d} + \boldsymbol{\mathcal{N}}_\omega,\end{aligned} \tag{2.50}$$

and the noise tensor $\boldsymbol{\mathcal{N}}_\omega$ is white Gaussian.

The tensor formulation in (2.50) is composed of the first dimension of size K associated each epoch, the second dimension of size Q corresponds to each tap of correlator bank, and the third dimension of size M is related to the number of elements of the receive antenna array.

2.4.3 Uniform Linear Array

The algorithms presented in Chapter 3 utilize two preprocessing techniques to increase precision, FBA and SPS/ESPS. In order to use FBA and ESPS the array steering matrix must necessarily be left centro-hermitian and Vandermonde. An M-element array with steering matrix $\mathbf{A} \in \mathbb{C}^{M \times L}$ is left centro-hermitian if it satisfies the following condition:

$$\boldsymbol{\Pi}_M \mathbf{A}^* = \mathbf{A}, \tag{2.51}$$

with

$$\boldsymbol{\Pi} = \begin{bmatrix} 0 & & 1 \\ & \iddots & \\ 1 & & 0 \end{bmatrix}, \tag{2.52}$$

which means that the conjugate of $\mathbf{A}$ flipped over the horizontal axis is the same as $\mathbf{A}$.

The uniform linear array (ULA) is an example of an array whose steering matrix is left centro-hermitian. As can be seen in Figure 2.5, the ULA has M equally-spaced elements arranged in a linear fashion with spacing Δ.

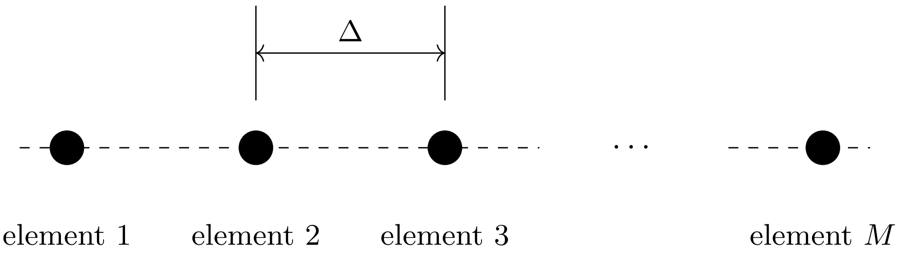


Figure 2.5: Uniform Linear Array with Δ spacing

For a sufficiently distant narrowband source, such as in GNSS, impinging wavefronts are assumed planar.

As can be seen in Figure 2.6, for adjacent elements m and $m+1$ spaced Δ apart, the wavefront coming at an angle θ will pass the first element and then the next after traveling a distance of d:

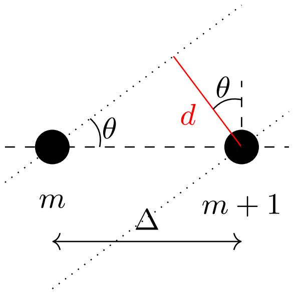


Figure 2.6: Wavefront incidence on adjacent array elements

Since the spacing is known, the distance traveled by the wavefront is $d = \Delta \sin\theta$ and the travel time between each element can be calculated as

$$\tau = \frac{d}{c} = \frac{\Delta \sin\theta}{c}. \tag{2.53}$$

A narrowband signal $s_m(t) = \mathrm{e}^{j2\pi ft}$ at the m-th element will arrive at the $m+1$-th element with a delay τ and will be

$$\begin{aligned} s_{m+1}(t) &= \mathrm{e}^{j2\pi f(t+\tau)} \\ &= \mathrm{e}^{j2\pi ft}\mathrm{e}^{j2\pi f\tau} \\ &= \mathrm{e}^{j2\pi ft}\mathrm{e}^{j\mu}. \end{aligned} \tag{2.54}$$

with μ denominated the spatial frequency and is

$$\begin{aligned} \mu &= 2\pi f\tau \\ &= 2\pi f\frac{\Delta \sin\theta}{c}. \end{aligned} \tag{2.55}$$

Since the structure of the array is uniform, the signal at each subsequent element in the array is shifted by exactly $\mathrm{e}^{j\mu}$. Arranging the signal received at all M elements into a vector $\mathbf{s}(t) \in \mathbb{C}^M$

$$\mathbf{s}(t) = \begin{bmatrix} \mathrm{e}^{j2\pi ft} \\ \mathrm{e}^{j2\pi ft}\mathrm{e}^{j\mu} \\ \vdots \\ \mathrm{e}^{j2\pi ft}\mathrm{e}^{j(M-2)\mu} \\ \mathrm{e}^{j2\pi ft}\mathrm{e}^{j(M-1)\mu} \end{bmatrix} = \mathrm{e}^{j2\pi ft} \begin{bmatrix} 1 \\ \mathrm{e}^{j\mu} \\ \vdots \\ \mathrm{e}^{j(M-2\mu} \\ \mathrm{e}^{j(M-1)\mu} \end{bmatrix} = \mathrm{e}^{j2\pi ft}\mathbf{a}(\mu). \tag{2.56}$$

in which $\mathbf{a}(\mu) \in \mathbb{C}^M$ is known as the steering vector and each m-th element is $\mathrm{e}^{j(m-1)\mu}$. This is known as the Vandermonde structure.

To make the Vandermonde structure steering vector $\mathbf{a}(\mu)$ conform to the condition set in (2.51), it has to be multiplied by $\mathrm{e}^{-j\frac{M-1}{2}\mu}$ for even M or $\mathrm{e}^{-j\lfloor\frac{M}{2}\rfloor\mu}$ for odd M, and it becomes

$$\underset{\text{even } M}{\mathbf{a}(\mu)} = \begin{bmatrix} \mathrm{e}^{-j\frac{M-1}{2}\mu} \\ \mathrm{e}^{-j\frac{M-2}{2}\mu} \\ \vdots \\ \mathrm{e}^{-j\frac{1}{2}\mu} \\ \mathrm{e}^{j\frac{1}{2}\mu} \\ \vdots \\ \mathrm{e}^{j\frac{M-2}{2}\mu} \\ \mathrm{e}^{j\frac{M-1}{2}\mu} \end{bmatrix} \qquad \underset{\text{odd } M}{\mathbf{a}(\mu)} = \begin{bmatrix} \mathrm{e}^{-j\lfloor\frac{M}{2}\rfloor\mu} \\ \mathrm{e}^{-j\lfloor\frac{M-2}{2}\rfloor\mu} \\ \vdots \\ 1 \\ \vdots \\ \mathrm{e}^{j\lfloor\frac{M-2}{2}\rfloor\mu} \\ \mathrm{e}^{j\lfloor\frac{M}{2}\rfloor\mu} \end{bmatrix}. \tag{2.57}$$

For L sources impinging on the array the contribution of each $\mathbf{a}(\mu_l), l = 1, \ldots, L$ is concatenated into a steering matrix $\mathbf{A} = [\mathbf{a}(\mu_1), \ldots, \mathbf{a}(\mu_L)] \in \mathbb{C}^{M\times L}$:

$$\underset{\text{even } M}{\mathbf{A}} = \begin{bmatrix} \mathrm{e}^{-j\frac{M-1}{2}\mu_1} & \cdots & \mathrm{e}^{-j\frac{M-1}{2}\mu_L} \\ \mathrm{e}^{-j\frac{M-2}{2}\mu_1} & \cdots & \mathrm{e}^{-j\frac{M-2}{2}\mu_L} \\ \vdots & \vdots & \vdots \\ \mathrm{e}^{-j\frac{1}{2}\mu_1} & \vdots & \mathrm{e}^{-j\frac{1}{2}\mu_L} \\ \mathrm{e}^{j\frac{1}{2}\mu_1} & \vdots & \mathrm{e}^{j\frac{1}{2}\mu_L} \\ \vdots & \vdots & \vdots \\ \mathrm{e}^{j\frac{M-2}{2}\mu_1} & \cdots & \mathrm{e}^{j\frac{M-2}{2}\mu_L} \\ \mathrm{e}^{j\frac{M-1}{2}\mu_1} & \cdots & \mathrm{e}^{j\frac{M-1}{2}\mu_L} \end{bmatrix} \quad \underset{\text{odd } M}{\mathbf{A}} = \begin{bmatrix} \mathrm{e}^{-j\lfloor\frac{M-1}{2}\rfloor\mu_1} & \cdots & \mathrm{e}^{-j\lfloor\frac{M}{2}\rfloor\mu_L} \\ \mathrm{e}^{-j\lfloor\frac{M-2}{2}\rfloor\mu_1} & \cdots & \mathrm{e}^{-j\lfloor\frac{M-2}{2}\rfloor\mu_L} \\ \vdots & \vdots & \vdots \\ 1 & \vdots & 1 \\ \vdots & \vdots & \vdots \\ \mathrm{e}^{j\lfloor\frac{M-2}{2}\rfloor\mu_1} & \cdots & \mathrm{e}^{j\lfloor\frac{M-2}{2}\rfloor\mu_L} \\ \mathrm{e}^{j\lfloor\frac{M-1}{2}\rfloor\mu_1} & \cdots & \mathrm{e}^{j\lfloor\frac{M}{2}\rfloor\mu_L} \end{bmatrix}, \tag{2.58}$$

and (2.51) holds.

For a ULA with half-wavelength spacing $\Delta = \lambda/2$ the spatial frequency is

$$\begin{aligned} \mu &= 2\pi f \frac{(\lambda/2)\sin\theta}{c} \\ &= \pi \sin\theta. \end{aligned} \tag{2.59}$$

If the array steering matrix is not left centro-hermitian and Vandermonde, we refer the reader to the interpolation scheme in [24], which can also account for real word imperfections resultant from faulty array construction or calibration.

Chapter 3

Tensor-based Approaches to Time-Delay Estimation

In this chapter, three algorithms are presented. The state-of-the-art tensor-based time-delay estimation from [5], and two novel approaches for filtering and time-delay estimation. The first approach is based on closed DoA estimation of the steering factor matrix followed by a simultaneous estimation of the complex amplitude and code factor matrices using Khatri-Rao factorization. The second approach uses an iterative least-squares orthogonal Procrustes problem (OPP) and Khatri-Rao factorization to estimate the steering and complex amplitude factor matrices.

3.1 State-of-the-Art Tensor-Based Time-Delay Estimation

In this section, we summarize the state-of-the-art tensor-based time-delay estimation approach, which is an HOSVD based eigenfilter with Forward Backward Averaging (FBA) and Expanded Spatial Smoothing (ESPS).

This chapter is divided into three sections which are also the steps of the state-of-the-art approach according to Figure 3.1. First a pre-processing step is applied to incorporate FBA and ESPS as illustrated in Figure 3.1 and detailed in Section 3.1.1. Next the HOSVD based rank-one filters are computed for three dimensions of the tensor and an improved output of the correlator bank is obtained as shown in Figure 3.1 and also according to Section 2.3.5. Finally, the amount of points is increased via a cubic spline interpolation and one dimension peak search is performed in order to locate the time-delay of the multidimensionally filtered output as presented in Section 3.1.3.

3.1.1 Forward-Backward Averaging and Expanded Spatial Smoothing

Similarly to matrix-based FBA [16], flipped identity matrices are used to duplicate the number of samples. The left-hand multiplication matrix $\mathbf{\Pi}_M \in \mathbb{R}^{M\times M}$ is an identity matrix of size M flipped along its vertical axis and the right-hand multiplication matrix $\mathbf{\Pi}_{KQ} \in \mathbb{R}^{KQ\times KQ}$ likewise. These matrices are applied to the spatial (third-mode) unfolding of the signal tensor in the following manner:

$$\mathbf{Z} = \begin{bmatrix}[\boldsymbol{\mathcal{Y}}_d]_{(3)} & \mathbf{\Pi}_M [\boldsymbol{\mathcal{Y}}_d]^*_{(3)} \mathbf{\Pi}_{KQ}\end{bmatrix} \in \mathbb{C}^{M\times 2KQ}. \tag{3.1}$$

In a similar fashion to matrix-based SPS [18], selection matrices that divide the array into L_S subarrays with $M_S = M - L_S + 1$ elements are used. The selection matrices are defined as

$$\mathbf{J}_{l_S} = \begin{bmatrix}\mathbf{0}_{M_S\times l_S-1} & \mathbf{I}_{M_S} & \mathbf{0}_{M_S\times L_S-l_S}\end{bmatrix} \in \mathbb{R}^{M_S\times M}, \tag{3.2}$$

for $l_S = 1, \ldots, L_S$.

Using the selection matrices, spatial smoothing is applied to the forward-backward averaged spatial unfolding of the signal tensor

$$\mathbf{W} = \begin{bmatrix}\mathbf{J}_1\mathbf{Z} & \cdots & \mathbf{J}_{L_S}\mathbf{Z}\end{bmatrix} \in \mathbb{C}^{M_S\times 2L_S KQ}, \tag{3.3}$$

and $\mathbf{W}$ is third-mode folded back into a forward-backward averaged spatially-smoothed fourth-order tensor $\boldsymbol{\mathcal{Z}}_{\text{ESPS}} \in \mathbb{C}^{2K\times Q\times M_S\times L_S}$.

3.1.2 Higher-Order SVD eigenfiltering

For time-delay estimation the HOSVD is applied to $\boldsymbol{\mathcal{Z}}_{\text{ESPS}}$

$$\boldsymbol{\mathcal{Z}}_{\text{ESPS}} = \boldsymbol{\mathcal{R}} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)} \times_4 \mathbf{U}^{(4)}, \tag{3.4}$$

in which $\boldsymbol{\mathcal{R}} \in \mathbb{C}^{2K\times Q\times M_S\times L_S}$ is the core tensor, and $\mathbf{U}^{(1)} \in \mathbb{C}^{2K\times 2K}$, $\mathbf{U}^{(2)} \in \mathbb{C}^{Q\times Q}$, $\mathbf{U}^{(3)} \in \mathbb{C}^{M_S\times M_S}$, and $\mathbf{U}^{(4)} \in \mathbb{C}^{L_S\times L_S}$ are unitary matrices collecting singular vectors of each mode's respective unfolding [15].

By assuming the LOS signal component has the greatest power compared with NLOS signal components, the dominant singular vectors of the first, third and fourth dimensions of $\boldsymbol{\mathcal{Z}}_{\text{ESPS}}$ in (3.4) are mostly correlated to the LOS signal component. Therefore, [5] has proposed the following HOSVD based eigenfilter:

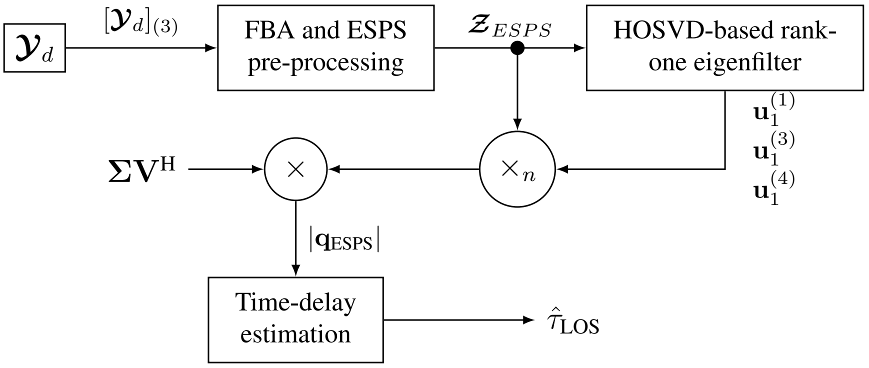
$$\mathbf{q}_{\text{ESPS}} = \Big(\boldsymbol{\mathcal{Z}}_{\text{ESPS}} \times_1 (\mathbf{u}_1^{(1)})^{\text{H}} \times_3 (\mathbf{u}_1^{(3)})^{\text{H}} \times_4 (\mathbf{u}_1^{(4)})^{\text{H}}\Big)\mathbf{\Sigma}\mathbf{V}^{\text{H}}. \tag{3.5}$$

Note that $\Sigma \mathbf{V}^{\mathrm{H}}$ is computed in (2.47).

3.1.3 Time-Delay Estimation

In order to obtain the time-delay estimation from $\mathbf{q}_{\text{ESPS}}$, a cubic spline interpolation based on $|\mathbf{q}_{\text{ESPS}}|$ is used to generate a cost function $F(\tau)$. The τ variable that maximizes the cost function is computed as follows

$$\hat{\tau}_{\text{LOS}} = \arg\max_{\tau}\{F(\tau)\}. \tag{3.6}$$



$\boldsymbol{\mathcal{Y}}_d$
$[\boldsymbol{\mathcal{Y}}_d]_{(3)}$
FBA and ESPS pre-processing
$\boldsymbol{\mathcal{Z}}_{ESPS}$
HOSVD-based rank-one eigenfilter
$\mathbf{u}_1^{(1)}$
$\mathbf{u}_1^{(3)}$
$\mathbf{u}_1^{(4)}$
$\Sigma\mathbf{V}^{\mathrm{H}}$
$\times$
$\times_n$
$|\mathbf{q}_{\text{ESPS}}|$
Time-delay estimation
$\hat{\tau}_{\text{LOS}}$

Figure 3.1: Block diagram of the state-of-the-art HOSVD eigenfilter-based time-delay estimation approach

3.2 Proposed DoA Estimation and KRF Approach

In this chapter, we propose a three step approach based on the direction of arrival (DoA) estimation, the Khatri-Rao factorization (KRF) and the selection of the estimated LOS time-delay. Note that to estimate the DoA of the LOS and NLOS signal components, the model order L_d should be known. Therefore, we refer to the tensor based model order selection schemes in [25, 26, 27].

This section is divided into three subsections. After a pre-processing step via FBA and ESPS is applied as illustrated in Figure 3.2, DoA estimation is utilized to rebuild the factor matrix $\mathbf{A}$ as described in Subsection 3.2.1. Then Khatri-Rao factorization is applied to the product of $\hat{\mathbf{A}}_d^{+}$ and $[\boldsymbol{\mathcal{Y}}_d]_{(3)}$ to estimate the factor matrices $\hat{\hat{\boldsymbol{\Gamma}}}_d$ and $\hat{\mathbf{C}}_d\mathbf{Q}_{\omega,d}$ according to Subsection 3.2.2. Finally, the LOS signal component of the estimated signal is found the time-delay is estimated as described in Subsection 3.2.3.

3.2.1 Estimation of DoA factor matrix

Since $\mathbf{Q}_{\omega,d}$ has been applied to the received signals as shown in (2.48), there are only L_d signal components and we desire to estimate the DoA of all L_d components. Note that $\hat{L}_d$ can be estimated by using the tensor based model order selection schemes discussed in [25, 26, 27].

As shown in Figure 3.2, DoA estimation is applied to the forward-backward averaged spatially smoothed signal matrix $\mathbf{W}$ calculated in (3.3).

Although there are several DoA schemes in the literature, due to the simplicity, good accuracy and low computational complexity, we use in our work the Estimation of Signal Parameter via Rotational Invariance Technique (ESPRIT) [28]. We assume the GNSS receiver is equipped with an antenna array, whose geometry is Vandermonde and left centro-hermitian.

3.2.2 PR code and complex amplitude estimation via Khatri-Rao Factorization

By rewriting the spatial unfolding of the noiseless signal from (2.50), the following expression is obtained:

$$[\boldsymbol{\mathcal{Y}}_{0,d}]_{(3)} = \mathbf{A}_d \left(\tilde{\boldsymbol{\Gamma}}_d^{\mathrm{T}} \diamond (\mathbf{C}_d\mathbf{Q}_{\omega,d})^{\mathrm{T}}\right)^{\mathrm{T}} \in \mathbb{C}^{M\times KQ}. \tag{3.7}$$

Once $\hat{\mathbf{A}}_d$ has been estimated, its pseudo-inverse can be applied to (3.7) such that

$$\begin{aligned}\hat{\mathbf{A}}_d^{+}[\boldsymbol{\mathcal{Y}}_{0,d}]_{(3)} &= \hat{\mathbf{A}}_d^{+}\mathbf{A}_d \left(\tilde{\boldsymbol{\Gamma}}_d^{\mathrm{T}} \diamond (\mathbf{C}_d\mathbf{Q}_{\omega,d})^{\mathrm{T}}\right)^{\mathrm{T}} \\ &\approx \left(\tilde{\boldsymbol{\Gamma}}_d^{\mathrm{T}} \diamond (\mathbf{C}_d\mathbf{Q}_{\omega,d})^{\mathrm{T}}\right)^{\mathrm{T}} \in \mathbb{C}^{L_d\times KQ},\end{aligned} \tag{3.8}$$

and the factor matrices $\tilde{\boldsymbol{\Gamma}}_d$ and $\mathbf{C}_d\mathbf{Q}_{\omega,d}$ can be estimated by Least Squares Khatri-Rao factorization (LSKRF) [29].

Given $(\tilde{\boldsymbol{\Gamma}}_d^{\mathrm{T}}\diamond(\mathbf{C}_d\mathbf{Q}_{\omega,d})^{\mathrm{T}})$, and considering it's l_d-th column can be calculated as the Khatri-Rao product of the l_d-th column of $\tilde{\boldsymbol{\Gamma}}_d^{\mathrm{T}}$ and $(\mathbf{C}_d\mathbf{Q}_{\omega,d})^{\mathrm{T}}$:

$$\left(\tilde{\boldsymbol{\Gamma}}_d^{\mathrm{T}} \diamond (\mathbf{C}_d\mathbf{Q}_{\omega,d})^{\mathrm{T}}\right)_{\cdot,l_d} = (\tilde{\boldsymbol{\Gamma}}_d^{\mathrm{T}})_{\cdot,l_d} \diamond (\mathbf{C}_d\mathbf{Q}_{\omega,d})^{\mathrm{T}}_{\cdot,l_d}, \tag{3.9}$$

with each column $(\tilde{\boldsymbol{\Gamma}}_d^{\mathrm{T}} \diamond (\mathbf{C}_d\mathbf{Q}_{\omega,d})^{\mathrm{T}})_{\cdot,l_d} \in \mathbb{C}^{KQ}$.

To solve for estimates of $\tilde{\boldsymbol{\Gamma}}_d$ and $\mathbf{C}_d\mathbf{Q}_{\omega,d}$, we reshape (3.9) into a matrix of size $Q\times K$:

$$\underset{Q\times K}{\mathrm{unvec}}\left\{\left(\tilde{\boldsymbol{\Gamma}}_d^{\mathrm{T}} \diamond (\mathbf{C}_d\mathbf{Q}_{\omega,d})^{\mathrm{T}}\right)_{\cdot,l_d}\right\} = (\mathbf{C}_d\mathbf{Q}_{\omega,d})^{\mathrm{T}}_{\cdot,l_d}(\tilde{\boldsymbol{\Gamma}}_d^{\mathrm{T}})^{\mathrm{T}}_{\cdot,l_d}. \tag{3.10}$$

Since (3.10) is a rank-one matrix, we can use a SVD-based rank-one approximation:

$$\underset{Q\times K}{\operatorname{unvec}}\left\{\left(\tilde{\mathbf{\Gamma}}_d^{\mathrm{T}} \diamond (\mathbf{C}_d\mathbf{Q}_{\omega,d})^{\mathrm{T}}\right)_{\cdot l_d}\right\} = \mathbf{U}_{l_d}\mathbf{\Sigma}_{l_d}\mathbf{V}_{l_d}. \tag{3.11}$$

The estimates for $(\tilde{\mathbf{\Gamma}}_d^{\mathrm{T}})_{\cdot,l_d}$ and $((\mathbf{C}_d\mathbf{Q}_{\omega,d})^{\mathrm{T}})_{\cdot,l_d}$ are $\sqrt{\sigma_{l_d,1}}\mathbf{v}^*_{l_d,1}$ and $\sqrt{\sigma_{l_d,1}}\mathbf{u}_{l_d,1}$, respectively, where $\sigma_{l_d,1}$ is the dominant singular value of $\mathbf{\Sigma}_{l_d}$, $\mathbf{u}_{l_d,1}$ is the dominant left singular vector of $\mathbf{U}_{l_d}$, and $\mathbf{v}^*_{l_d,1}$ is the conjugate of the dominant right singular vector of $\mathbf{V}_{l_d}$. This is repeated for $l_d = 1, \ldots, L_d$.

Once $\hat{\mathbf{C}}_d\mathbf{Q}_{\omega,d}$ is estimated using LSKRF, the next step is to find which row corresponds to the LOS signal component due to possible permutation ambiguities in $\hat{\mathbf{A}}_d$ estimated in 3.2.1.

3.2.3 LOS Time-Delay Estimation

In order to find the time-delay of the LOS signal component, two schemes are proposed, namely, greatest power based scheme and smallest delay based scheme.

For the greatest power based scheme, we assume that the LOS signal component is not blocked. Therefore, the LOS signal component has the greatest power in comparison with the multipath components. In this case, the following expression can be used to locate the estimated LOS signal component, $\hat{\mathbf{c}}_{\mathrm{LOS}}$:

$$l_{\mathrm{LOS}} = \max_{l_d} \|\hat{\tilde{\mathbf{\Gamma}}}_{d,l_d\cdot}\|_2, \tag{3.12}$$

in which $\hat{\tilde{\mathbf{\Gamma}}}_{d,l_d\cdot}$ is the l_d-th row of the matrix $\hat{\tilde{\mathbf{\Gamma}}}_d$. Note that in the greatest power based scheme, we just compute the delay of the selected component using the following expression

$$\mathbf{q}_{\mathrm{DoA/KRF}} = \hat{\mathbf{c}}_{\mathrm{LOS}}^{\mathrm{T}}\mathbf{Q}_{\omega,d}\mathbf{\Sigma}\mathbf{V}^{\mathrm{H}}, \tag{3.13}$$

followed by a cubic spline interpolation to obtain a cost function and estimate the time-delay as in (3.6).

For the smallest time-delay based scheme, we compute the L_d time-delays of $\hat{\mathbf{c}}_{l_d}$ for $l_d = 1, \ldots, L_d$. Using the resulting time-delay estimation vector $\hat{\boldsymbol{\tau}} = [\hat{\tau}_1, \ldots, \hat{\tau}_{L_d}]^{\mathrm{T}} \in \mathbb{R}^{L_d}$, the LOS time-delay is found by solving

$$\hat{\tau}_{\mathrm{LOS}} = \min_{\tau}\{\hat{\boldsymbol{\tau}}\}. \tag{3.14}$$

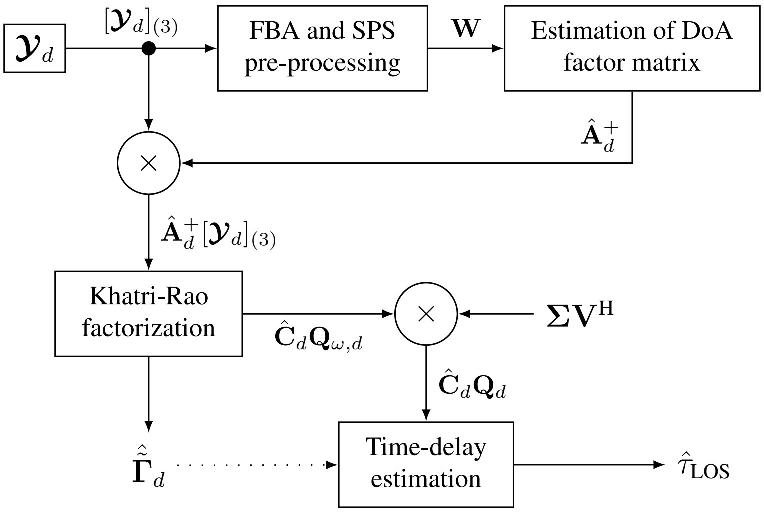

$\mathcal{Y}_d$
$[\mathcal{Y}_d]_{(3)}$
FBA and SPS pre-processing
$\mathbf{W}$
Estimation of DoA factor matrix
$\hat{\mathbf{A}}_d^+$
×
$\hat{\mathbf{A}}_d^+[\mathcal{Y}_d]_{(3)}$
Khatri-Rao factorization
$\hat{\mathbf{C}}_d\mathbf{Q}_{\omega,d}$
×
$\mathbf{\Sigma}\mathbf{V}^{\mathrm{H}}$
$\hat{\mathbf{C}}_d\mathbf{Q}_d$
$\hat{\tilde{\mathbf{\Gamma}}}_d$
Time-delay estimation
$\hat{\tau}_{\mathrm{LOS}}$

Figure 3.2: Block Diagram of the proposed DoA/KRF time-delay estimation approach

3.3 Procrustes estimation and Khatri-Rao factorization (ProKRaft) filtering

In this section we present an iterative approach based on the Orthogonal Procrustes Problem (OPP) and KRF to calculate the DoA and complex amplitude factor matrices simultaneously. Since this technique relies on separating the subspace of the SVD, the model order L_d should be known.

3.3.1 Simultaneous DoA and amplitude factor matrix estimation

To simultaneously estimate the factor matrices $\mathbf{A}_d$ and $\mathbf{\Gamma}_d$, the ProKRaft approach relies on the fact that a Hermitian unfolding can be achieved by calculating the multimode covariance matrix, $\mathbf{R}_{\mathrm{mm}}$, which is the product of the transpose and conjugate of the second-mode unfolding of $\mathcal{Y}_d$ divided by the number of samples, N,

$$\mathbf{R}_{\mathrm{mm}} = [\mathcal{Y}_d]^{\mathrm{T}}_{(2)}[\mathcal{Y}_d]^{*}_{(2)}/N \tag{3.15}$$

$$= (\mathbf{A}_d \diamond \tilde{\mathbf{\Gamma}}^{\mathrm{T}}_d)\underbrace{(\mathbf{C}_d\mathbf{Q}_{\omega,d})(\mathbf{C}_d\mathbf{Q}_{\omega,d})^{\mathrm{H}}/N}_{\mathbf{R}_{\mathbf{C}}}(\mathbf{A}_d \diamond \tilde{\mathbf{\Gamma}}^{\mathrm{T}}_d)^{\mathrm{H}} \tag{3.16}$$

$$= (\mathbf{A}_d \diamond \tilde{\mathbf{\Gamma}}^{\mathrm{T}}_d)\mathbf{R}_{\mathbf{C}}(\mathbf{A}_d \diamond \tilde{\mathbf{\Gamma}}^{\mathrm{T}}_d)^{\mathrm{H}} \in \mathbb{C}^{MK\times MK}, \tag{3.17}$$

in which $\mathbf{R}_{\mathbf{C}} \in \mathbb{R}^{L_d\times L_d}$ is the covariance matrix of the LOS and NLOS signal components from satellite d.

Considering $\mathbf{R}_{\mathbf{C}} \approx \mathbf{I}_{L_d}$, the multimode covariance matrix can be approximated as

$$\mathbf{R}_{\mathrm{mm}} \approx (\mathbf{A}_d \diamond \tilde{\mathbf{\Gamma}}^{\mathrm{T}}_d)(\mathbf{A}_d \diamond \tilde{\mathbf{\Gamma}}^{\mathrm{T}}_d)^{\mathrm{H}}, \tag{3.18}$$

which is the Hermitian-symmetric unfolding of $\mathcal{Y}_d$, $\mathbf{Y}_{\mathrm{H}}$, whose structure is the same as that in (2.35).

Given this Hermitian-symmetric unfolding we can define a square-root factor matrix $\mathbf{Y}_{\mathrm{H}}^{\frac{1}{2}} \in \mathbb{C}^{MK \times L_d}$ such that

$$\mathbf{Y}_{\mathrm{H}} = \mathbf{Y}_{\mathrm{H}}^{\frac{1}{2}} (\mathbf{Y}_{\mathrm{H}}^{\frac{1}{2}})^{\mathrm{H}}. \tag{3.19}$$

An estimate for $\mathbf{Y}_{\mathrm{H}}^{\frac{1}{2}}$ can be achieved by using the SVD of $\mathbf{Y}_{\mathrm{H}} = \mathbf{U}_{\mathrm{Y}} \mathbf{\Sigma}_{\mathrm{Y}} \mathbf{V}_{\mathrm{Y}}^{\mathrm{H}}$ by using the left-handed singular vectors and singular values which span the signal subspace, $\mathbf{U}_{\mathrm{Y}}^{[L_d]} \in \mathbb{C}^{MK \times L_d}$ and $\mathbf{\Sigma}_{\mathrm{Y}}^{[L_d]} \in \mathbb{C}^{L_d \times L_d}$, and a unitary rotation matrix $\mathbf{W}^{\mathrm{H}} \in \mathbb{C}^{L_d \times L_d}$:

$$\begin{aligned} \mathbf{Y}_{\mathrm{H}}^{\frac{1}{2}} &= \mathbf{U}_{\mathrm{Y}}^{[L_d]} \mathbf{\Sigma}_{\mathrm{Y}}^{[L_d]} \mathbf{W}^{\mathrm{H}} = (\mathbf{A}_d \diamond \tilde{\mathbf{\Gamma}}_d^{\mathrm{T}}) \\ &= \mathbf{F} \mathbf{W}^{\mathrm{H}} = \mathbf{G}. \end{aligned} \tag{3.20}$$

Mapping $\mathbf{F}$ to $\mathbf{G}$ using a unitary matrix $\mathbf{W}^{\mathrm{H}}$ is known as the "orthogonal Procrustes problem," which has a solution by applying the SVD to $\mathbf{G}^{\mathrm{H}} \mathbf{F}$ and using its singular vectors to estimate $\mathbf{W}^{\mathrm{H}}$ [30]:

$$\begin{aligned} \mathbf{G}^{\mathrm{H}} \mathbf{F} &= \mathbf{U}_{\mathrm{P}} \mathbf{\Sigma}_{\mathrm{P}} \mathbf{V}_{\mathrm{P}}^{\mathrm{H}}, \\ \mathbf{W}^{\mathrm{H}} &= \mathbf{U}_{\mathrm{P}} \mathbf{V}_{\mathrm{P}}^{\mathrm{H}}. \end{aligned} \tag{3.21}$$

An iterative algorithm can solve for estimates of $\hat{\mathbf{A}}_d$ and $\hat{\tilde{\mathbf{\Gamma}}}_d$ by alternating between (3.20) and (3.21).

As can be seen in Figure 3.3, ProKRaft begins by calculating the left singular vectors and singular values of $\mathbf{R}_{\mathrm{mm}}$, and initializing $\mathbf{W}^{(0)} = \mathbf{I}_L$.

$\mathbf{W}^{\mathrm{H}}$ is used to calculate $\mathbf{T}_1 = \mathbf{U}^{[L_d]} \mathbf{\Sigma}^{[L_d]} \mathbf{W}^{\mathrm{H}}$. By applying KRF to $\mathbf{T}_1$, estimates for $\hat{\mathbf{A}}$ and $\hat{\tilde{\mathbf{\Gamma}}}$ are acquired, which are used to calculate $\mathbf{T}_2 = (\hat{\mathbf{A}} \diamond \hat{\tilde{\mathbf{\Gamma}}}) \mathbf{U}^{[L_d]} \mathbf{\Sigma}^{[L_d]}$. The SVD of $\mathbf{T}_2$ provides the singular vector matrices $\mathbf{U}_{\mathrm{P}}$ and $\mathbf{V}_{\mathrm{P}}^{\mathrm{H}}$ used to update $\mathbf{W}^{\mathrm{H}}$. The alternating process repeats until convergence.

Once the factor matrices have been estimated, filtering is done by calculating the n-mode products of the pseudoinverse of the estimated factor matrices:

$$\hat{\boldsymbol{\mathcal{C}}}_d \boldsymbol{\mathcal{Q}}_{\omega,d} = \boldsymbol{\mathcal{Y}}_d \times_1 (\hat{\tilde{\mathbf{\Gamma}}}_d^{\mathrm{T}})^+ \times_3 (\hat{\mathbf{A}})^+, \tag{3.22}$$

in which $\hat{\boldsymbol{\mathcal{C}}}_d \boldsymbol{\mathcal{Q}}_{\omega,d}$ is the tensor that accumulates the L_d correlated PR codes of the LOS and NLOS signal components. Each l-th code accumulates in each $(\hat{\boldsymbol{\mathcal{C}}}_d \boldsymbol{\mathcal{Q}}_{\omega,d})_{l,\cdot,l}$ vector.

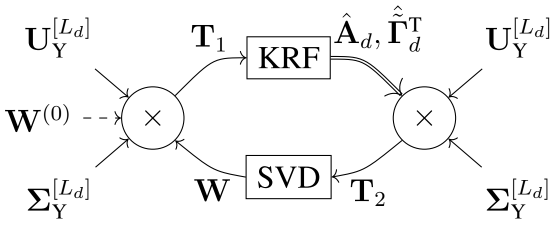

$\mathbf{U}_{\mathrm{Y}}^{[L_d]}$
$\mathbf{T}_1$
KRF
$\hat{\mathbf{A}}_d, \hat{\tilde{\mathbf{\Gamma}}}_d^{\mathrm{T}}$
$\mathbf{U}_{\mathrm{Y}}^{[L_d]}$
$\mathbf{W}^{(0)}$
×
×
$\mathbf{\Sigma}_{\mathrm{Y}}^{[L_d]}$
$\mathbf{W}$
SVD
$\mathbf{T}_2$
$\mathbf{\Sigma}_{\mathrm{Y}}^{[L_d]}$

Figure 3.3: Block Diagram of the proposed ProKRaft factor matrices estimation approach

3.3.2 LOS Time-Delay Estimation

Because permutation ambiguities are possible when estimating the factor matrices, selection schemes like the ones described in Subsection 3.2.3 can be applied followed by the cubic spline interpolation described in Subsection 3.1.3 to refine the time-delay estimation.

Chapter 4

Simulations

Similarly to [5], we consider the following scenario with a left centro-hermitian uniform linear array with $M = 8$ elements and half-wavelength ($\Delta = \lambda/2$) spacing. The GNSS signal is a GPS course acquisition PR code from $D = 1$ satellite at a carrier frequency $f_c = 1575.42$ MHz, bandwidth $B = 1.023$ MHz and chip duration $T_C = 1/B = 977.52$ ns with $N = 2046$ samples collected every k-th observation period during $K = 30$ epochs. Each epoch has duration $\Delta t = 1$ ms.

Aside from the LOS signal with time-delay τ_{LOS}, there is one NLOS multipath component ($L = 2$) with time-delay τ_{NLOS}, and an azimuth angle difference $\Delta\phi$. The delay between τ_{LOS} and τ_{NLOS} is $\Delta\tau$ such that $\tau_{\text{NLOS}} = \tau_{\text{LOS}} + \Delta\tau$. For SPS/ESPS the array is divided into $L_S = 5$ subarrays with $M_S = 4$ elements each. Signal phases $\arg\{\boldsymbol{\gamma}\}$ are independent and identically distributed $\sim$ U$[0, 2\pi[$. The number of correlators in the bank is $Q = 11$ equally spaced between $-T_C$ and T_C. The LOS signal is selected using (3.12).

The carrier-to-noise ratio is $C/N_0 = 48$ dB-Hz, resulting in a pre-correlation $\text{SNR}_{\text{pre}} = C/N_0 - 10\log_{10}(2B) \approx -15.11$ dB. Given the processing gain $G = 10\log_{10}(B\Delta t) \approx 30.1$ dB, post-correlation $\text{SNR}_{\text{post}} = \text{SNR}_{\text{pre}} + G \approx 15$ dB and signal to multipath ratio (SMR) of 5 dB. To compare results, the root mean squared error (RMSE) of the time-delay multiplied by the speed of light, $c = 299792458$ m/s, is used, expressed in meters. For each $\Delta\tau/T_C$ a Monte Carlo (MC) simulation with P iterations the RMSE is calculated by

$$\text{RMSE (m)} = c \cdot \sqrt{\frac{1}{P}\sum_{p=1}^{P}(\tau_p - \hat{\tau}_p)^2}. \tag{4.1}$$

1000 MC simulations were performed to plot the RMSE (m) curves of time-delay estimation utilizing HOSVD with FBA+ESPS explained in Section 3.1, filtering supposing known $\mathbf{A}$ and $\tilde{\mathbf{\Gamma}}$, direct noiseless correlation ($\mathbf{c}_{\text{LOS}}\mathbf{Q}$) and the proposed DoA/KRF approach using the spatially-smoothed forward-backward averaged signal data in (3.3) to estimate the DoAs and rebuild $\hat{\mathbf{A}}$.

In Figure 4.1 the MC simulation is executed with the parameters described above and azimuth angle difference of $\Delta\phi = \pi/3$ radians. The HOSVD+FBA+ESPS eigenfiltering approach achieves its best results when the NLOS signal is either very strongly correlated, $\Delta\tau/T_C \approx [0-0.1]$, or weakly correlated, $\Delta\tau/T_C \approx [0.8-1]$, to the LOS signal, with its error peaking at approximately 2.1 m at $\Delta\tau/T_C \approx 0.5$. The proposed DoA/KRF approach is consistent with the results provided by filtering a priori knowledge of $\mathbf{A}$ and $\tilde{\mathbf{\Gamma}}$.

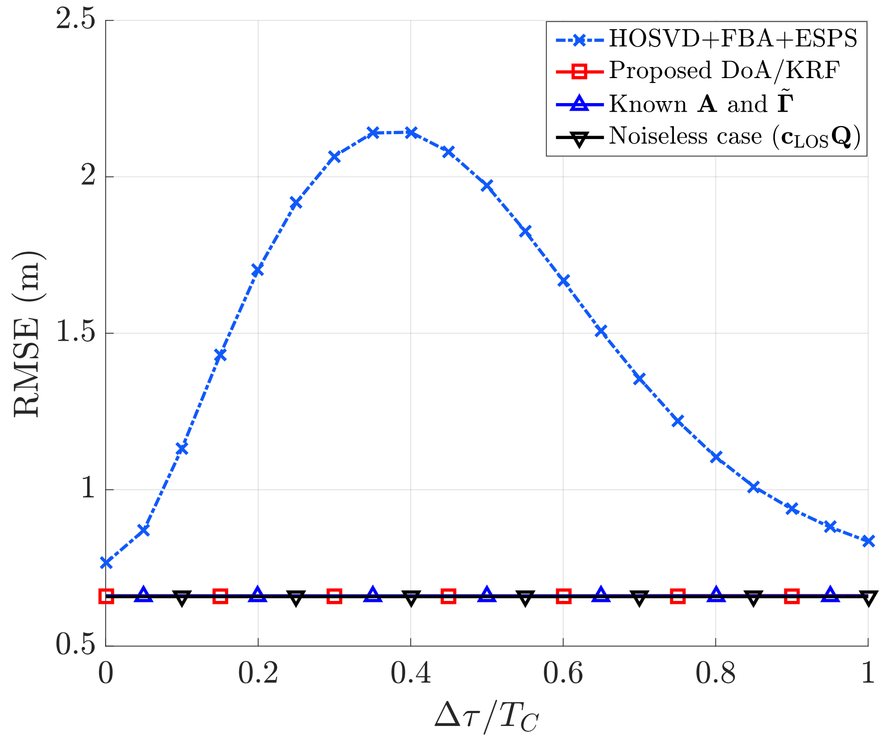


Figure 4.1: Simulation results for DoA/KRF, $D = 1$, $\Delta\phi = \pi/3$

In Figure 4.2 the MC simulation is executed with the parameters described above and azimuth angle difference of $\Delta\phi = \pi/4$ radians. The HOSVD+FBA+ESPS eigenfiltering approach achieves its best results when the NLOS signal is strongly correlated, $\Delta\tau/T_C \approx [0-0.1]$, with similar results for $\Delta\tau/T_C \approx [0.1-2]$ or $\Delta\tau/T_C \approx [0.75-1]$, and its error peaking at approximately 4.3 m at $\Delta\tau/T_C \approx 0.45$. The proposed DoA/KRF approach obtains an optimal performance, since its curve practically superposes the curve with filtering with a priori knowledge of $\mathbf{A}$ and $\tilde{\mathbf{\Gamma}}$.

In Figure 4.3 the MC simulation is executed with the parameters described above and azimuth angle difference of $\Delta\phi = \pi/6$ radians. The HOSVD+FBA+ESPS eigenfiltering approach achieves its best results when the NLOS signal is strongly correlated, $\Delta\tau/T_C \approx [0-0.1]$, or weakly correlated, $\Delta\tau/T_C \approx [0.65-1]$, with its error peaking at approximately 1.35 m at $\Delta\tau/T_C \approx 0.4$. Similarly to Figures 4.1 and 4.2, the curve of DoA/KRF almost superposes the curve of the a priori $\mathbf{A}$ and $\tilde{\mathbf{\Gamma}}$ filtering.

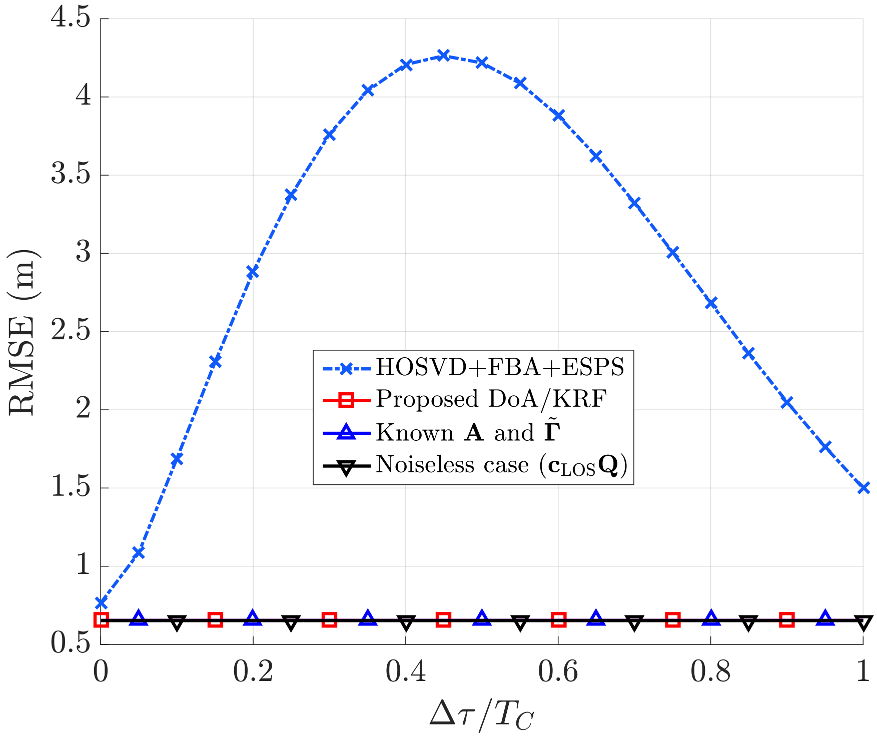


Figure 4.2: Simulation results for DoA/KRF, $D = 1$, $\Delta\phi = \pi/4$

4.1 ProKRaft filtering time-delay estimation

Simulations using ProKRaft filtering are done using the scenario nearly identical to the described above, except that the complex amplitude factor matrix $\tilde{\mathbf{\Gamma}}$ in (2.50) suffers phase shifts during the collection interval.

In Figure 4.4 the MC simulation is executed with the azimuth angle difference of $\Delta\phi = \pi/3$ radians. The HOSVD+FBA+ESPS results are different because of the changing phases of $\mathbf{\Gamma}$. The ProKRaft approach is consistent with the DoA/KRF approach and both have similar results to filtering with a priori knowledge of $\mathbf{A}$ and $\tilde{\mathbf{\Gamma}}$.

In Figure 4.5 the MC simulation is executed with the azimuth angle difference of $\Delta\phi = \pi/4$ radians. Again the results for HOSVD+FBA+ESPS are different because of the changing phases but are still consistent with those in Figure 4.2. The ProKRaft approach is consistent with the DoA/KRF approach and both have similar results to filtering with a priori knowledge of $\mathbf{A}$ and $\tilde{\mathbf{\Gamma}}$.

In Figure 4.5 the MC simulation is executed with the azimuth angle difference of $\Delta\phi = \pi/6$ radians. The ProKRaft approach is consistent with the DoA/KRF approach and both have similar results to filtering with a priori knowledge of $\mathbf{A}$ and $\tilde{\mathbf{\Gamma}}$.

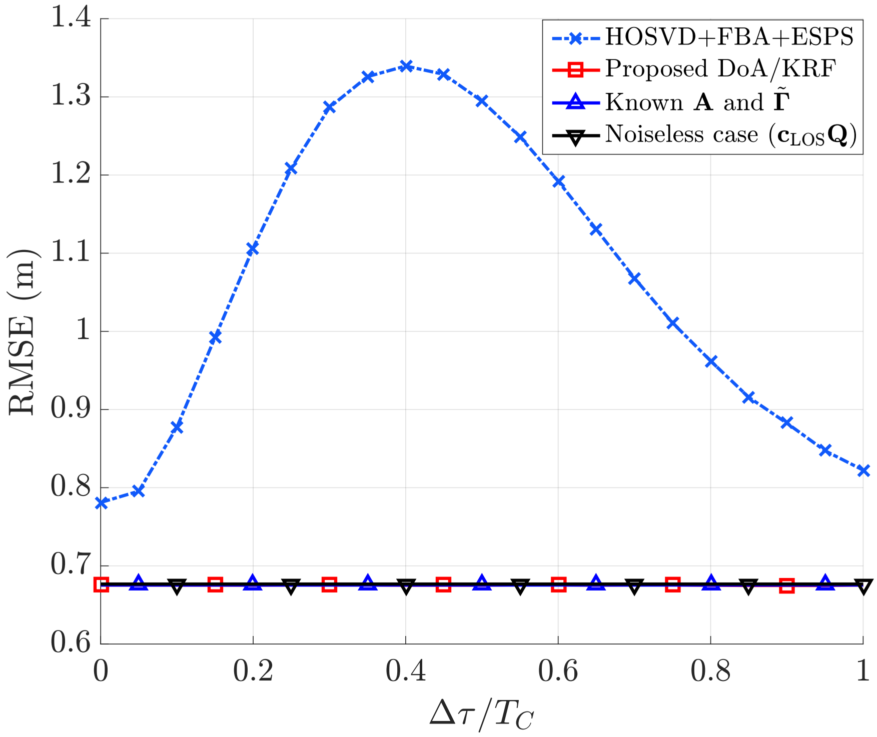


Figure 4.3: Simulation results for DoA/KRF, $D = 1$, $\Delta\phi = \pi/6$

4.2 Simulation with multiple satellites

To compare the results above where only one satellite and a single multipath component are simulated, simulations with $D = 2$ and $D = 3$ satellites were performed. The scenario is similar to the proposed previously but only $\Delta\phi = \pi/3$ is considered and aside from the desired satellite the other satellites do not have multipath components.

In Figure 4.7 the MC simulation was performed with $D = 2$, $L_1 = 2$, $L_2 = 1$ and $\Delta\phi = \pi/3$. While the results were worse than for $D = 1$ the proposed DoA/KRF scheme performs similarly to filtering with prior channel information and outperforms the state-of-the-art HOSVD+FBA+ESPS.

In Figure 4.8 the MC simulation was performed with $D = 3$, $L_1 = 2$, $L_2, L_3 = 1$ and $\Delta\phi = \pi/3$. The results were similar to those in Figure 4.7.

In Figure 4.9 the MC simulation was performed with $D = 4$, $L_1 = 2$, $L_2, L_3, L_4 = 1$ and $\Delta\phi = \pi/3$. The results were similar to those in Figures 4.7 and 4.8 but with the HOSVD+FBA+ESPS curve becoming flatter.

4.3 Effects of improper model order estimation

Both DoA/KRF and ProKRaft rely on proper estimation of the model order d to function properly. The following simulations were done to illustrate the effect of over- and underesti-

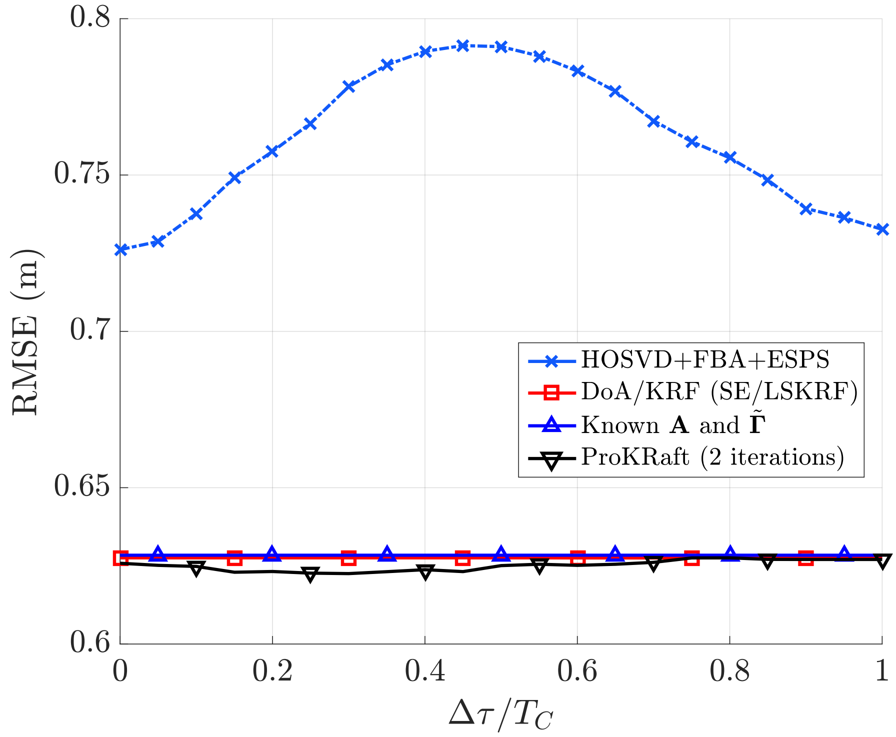


Figure 4.4: Simulation results for ProKRaft, $D = 1$, $\Delta\phi = \pi/3$

mating the model order on time-delay estimation.

For reference, simulations were executed for both DoA/KRF and ProKRaft with $L_1 = 3$ obtaining the following results shown in Figures 4.10 and 4.11, respectively. DoA/KRF performs well for $L_1 = 3$, while ProKRaft is outperformed by the HOSVD eigenfilter. Simulation parameters are those of DoA/KRF and ProKRaft used previously.

4.3.1 Underestimation

The expected effect of underestimation of the model order is that remaining components whose singular values are weaker are modeled as noise or interference, in effect increasing SNR/SINR and decreasing performance.

In Figure 4.12, $D = 1$, $L_1 = 2$ but the estimated model order is $\hat{L}_1 = 1$. Performance is slightly affected.

In Figure 4.13, $D = 1$, $L_1 = 3$ but the estimated model order is $\hat{L}_1 = 2$. Performance degradation is significant.

In Figure 4.14, $D = 1$, $L_1 = 2$ but the estimated model order is $\hat{L}_1 = 1$. Performance degradation is significant.

In Figure 4.15, $D = 1$, $L_1 = 3$ but the estimated model order is $\hat{L}_1 = 2$. Performance degradation is significant.

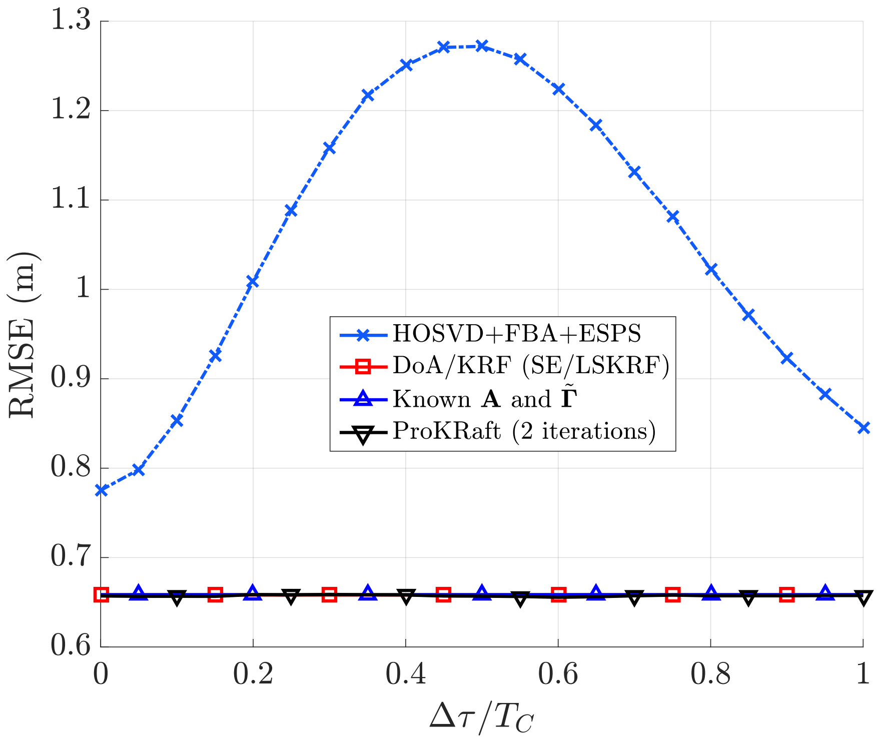


Figure 4.5: Simulation results for ProKRaft, $D = 1$, $\Delta\phi = \pi/4$

4.3.2 Overestimation

The expected effect of overestimation is more drastic since it implies calculating parameters of inexistent components as if they were present.

In Figure 4.16, $D = 1$, $L_1 = 2$ but the estimated model order is $\hat{L}_1 = 3$. DoA/KRF fails.

In Figure 4.17, $D = 1$, $L_1 = 3$ but the estimated model order is $\hat{L}_1 = 4$. DoA/KRF fails.

In Figure 4.18, $D = 1$, $L_1 = 2$ but the estimated model order is $\hat{L}_1 = 3$. DoA/KRF suffered a performance loss. ProKRaft suffers from very poor performance.

In Figure 4.19, $D = 1$, $L_1 = 3$ but the estimated model order is $\hat{L}_1 = 4$. Both DoA/KRF and ProKRaft fail.

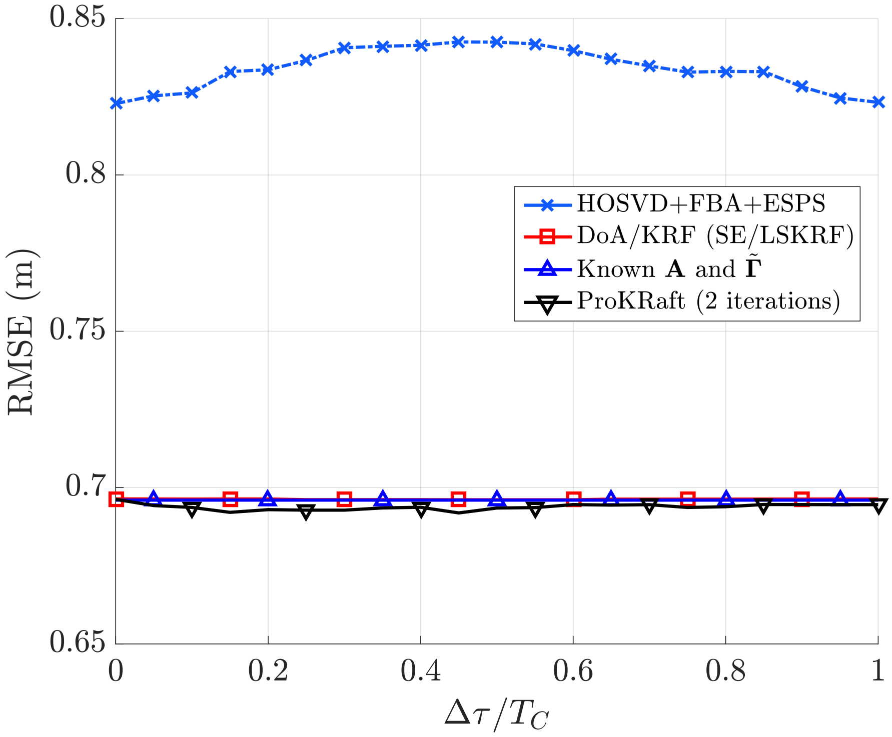


Figure 4.6: Simulation results for ProKRaft, $D = 1$, $\Delta\phi = \pi/6$

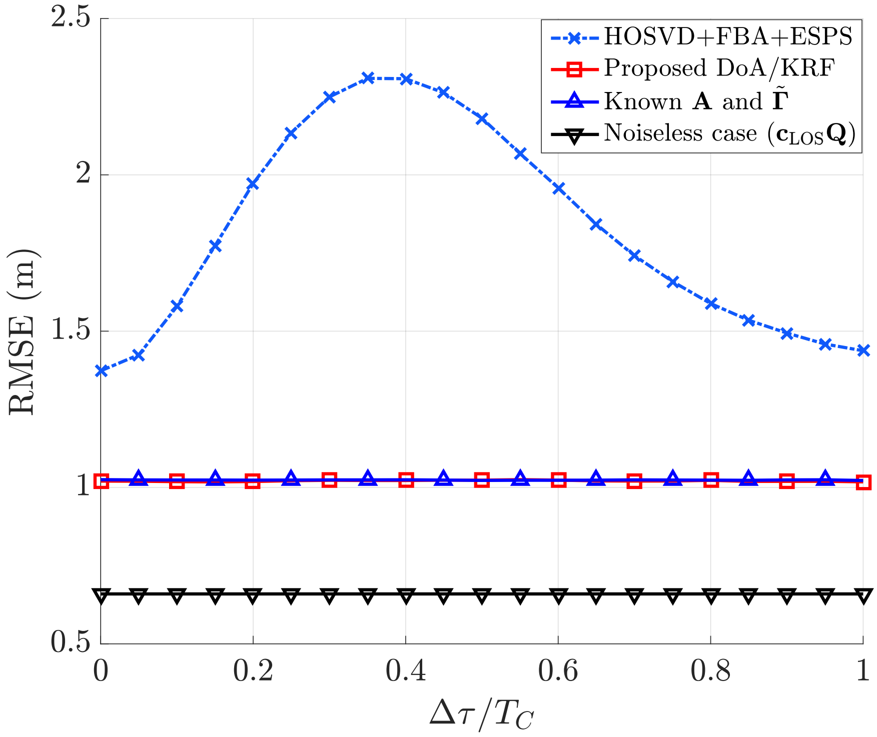


Figure 4.7: Simulation results for DoA/KRF, $D = 2$, $\Delta\phi = \pi/3$

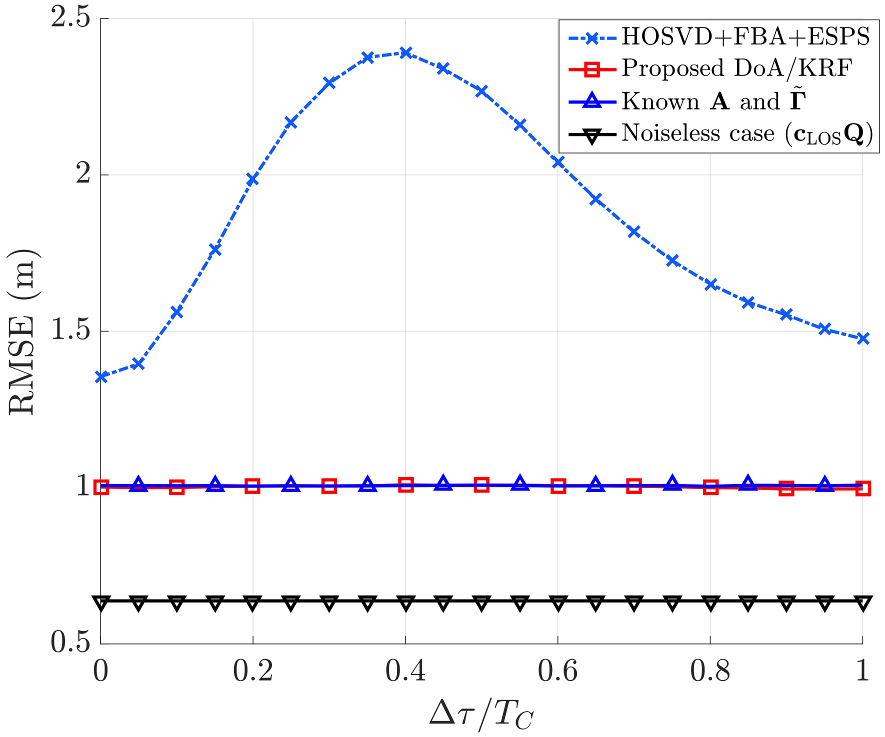


Figure 4.8: Simulation results for DoA/KRF, $D = 3$, $\Delta\phi = \pi/3$

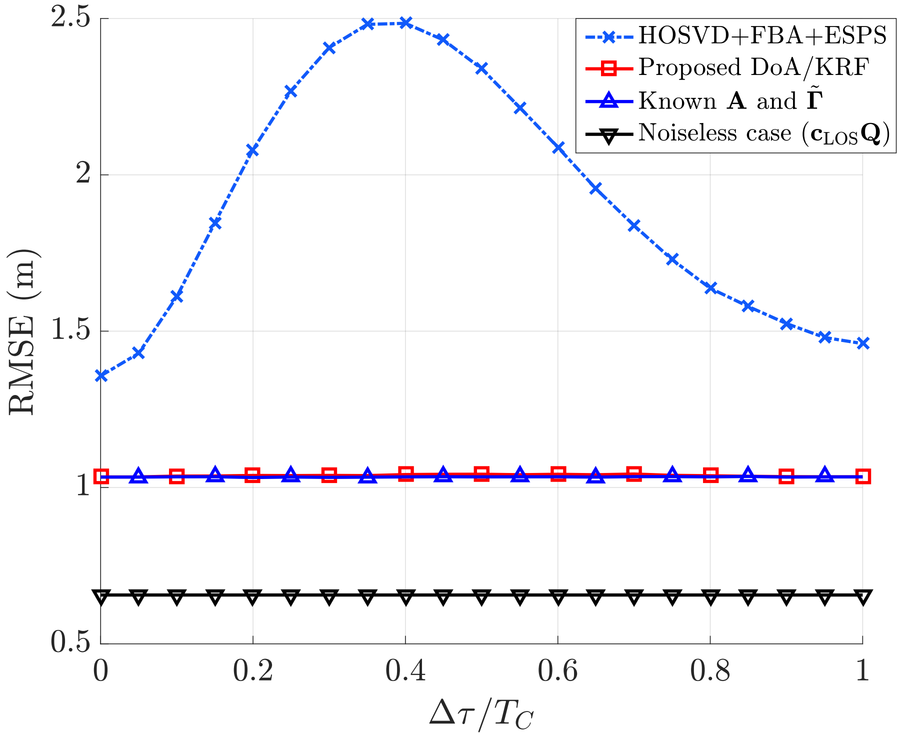


Figure 4.9: Simulation results for DoA/KRF, $D = 4$, $\Delta\phi = \pi/3$

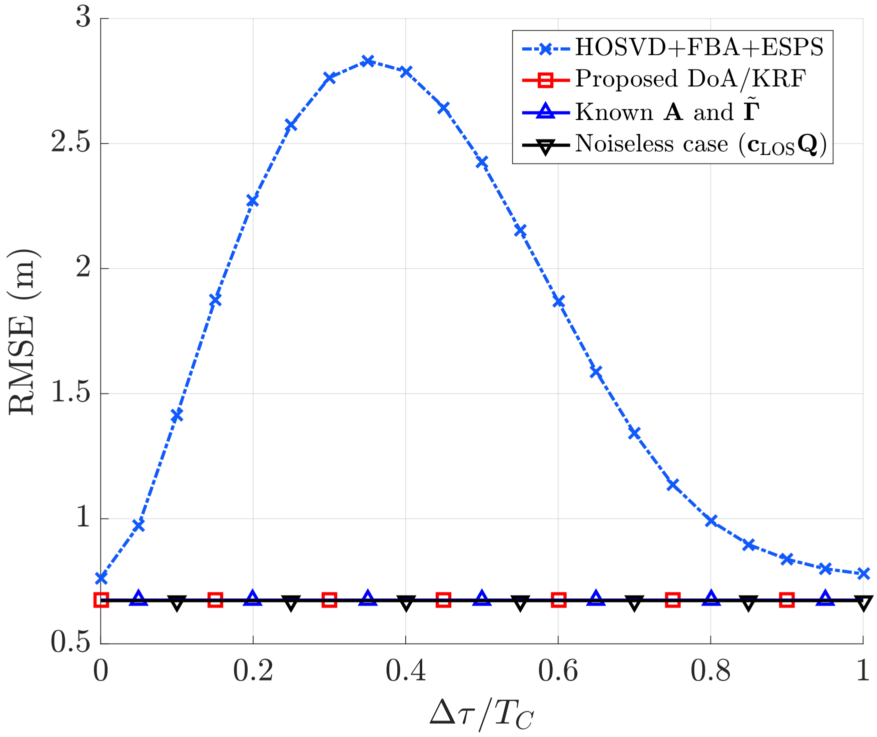


Figure 4.10: Simulation results for DoA/KRF, $D = 1$, $\Delta\phi = \pi/3$, $L_1 = 3$, $\hat{L}_1 = 3$

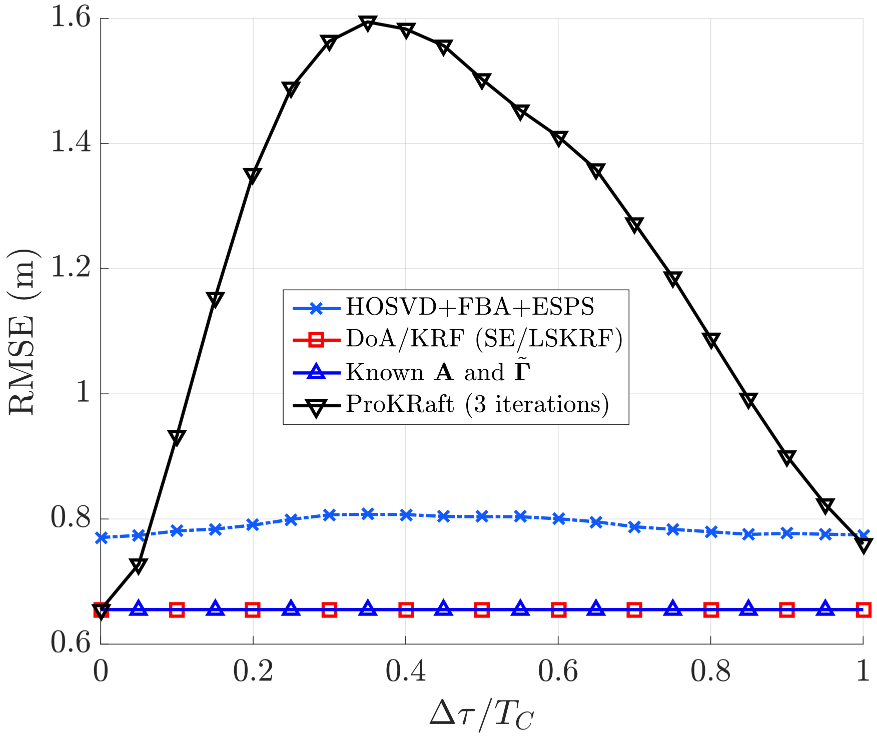


Figure 4.11: Simulation results for ProKRaft, $D = 1$, $\Delta\phi = \pi/3$, $L_1 = 3$, $\hat{L}_1 = 3$

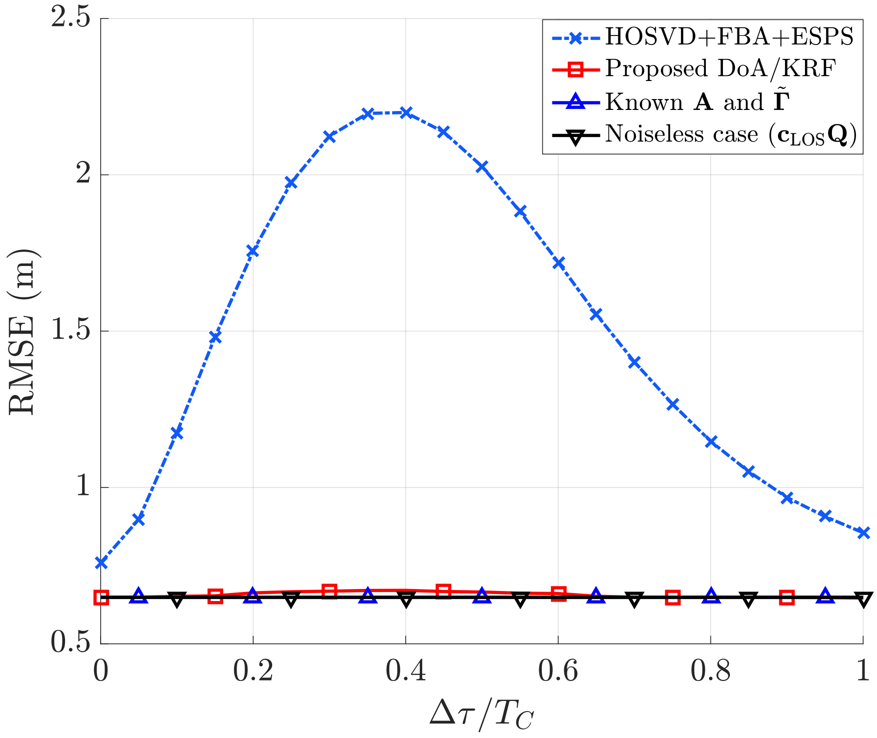


Figure 4.12: Simulation results for DoA/KRF, $D = 1$, $\Delta\phi = \pi/3$, $L_1 = 2$, $\hat{L}_1 = 1$

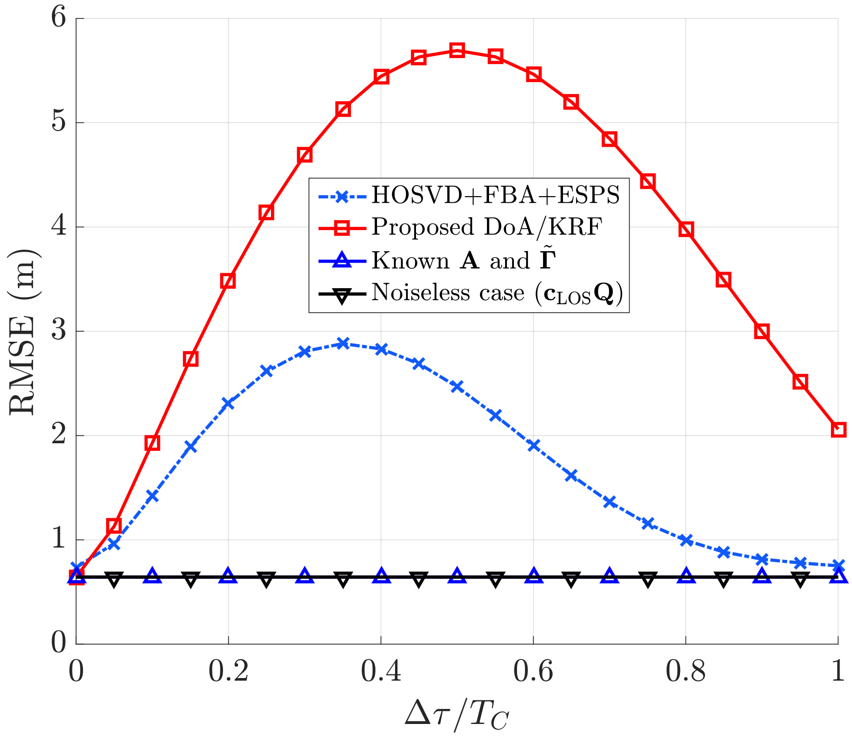


Figure 4.13: Simulation results for DoA/KRF, $D = 1$, $\Delta\phi = \pi/3$, $L_1 = 3$, $\hat{L}_1 = 2$

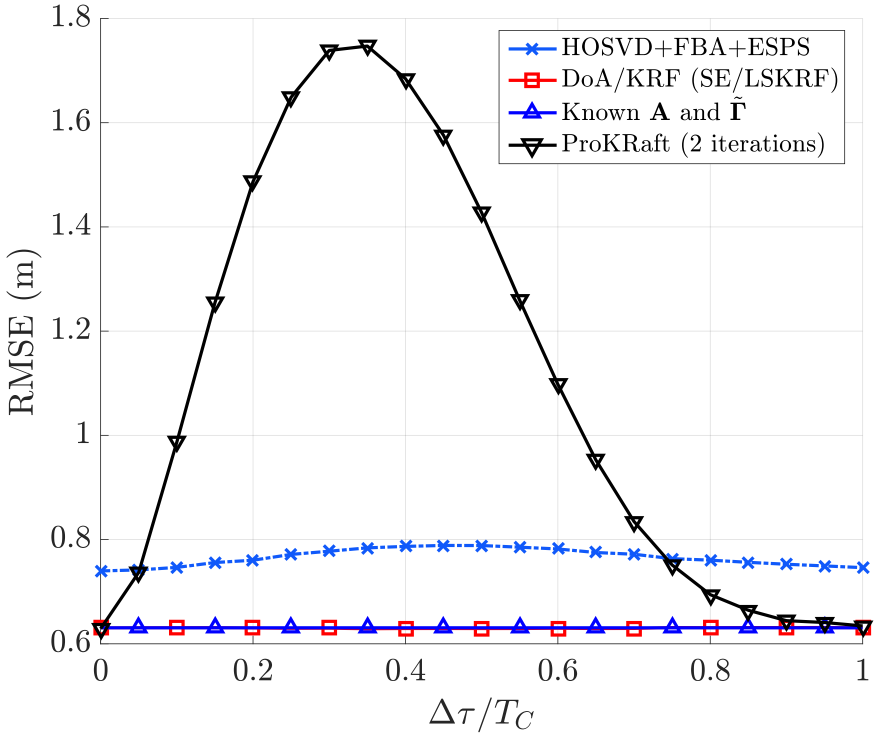


Figure 4.14: Simulation results for ProKRaft, $D = 1$, $\Delta\phi = \pi/3$, $L_1 = 2$, $\hat{L}_1 = 1$

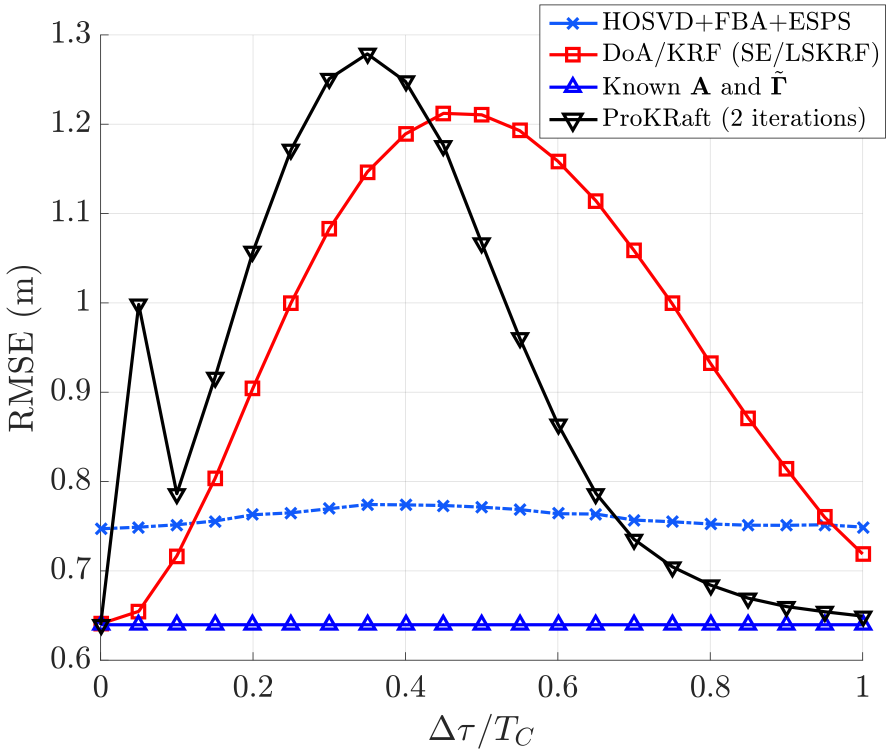


Figure 4.15: Simulation results for ProKRaft, $D = 1$, $\Delta\phi = \pi/3$, $L_1 = 3$, $\hat{L}_1 = 2$

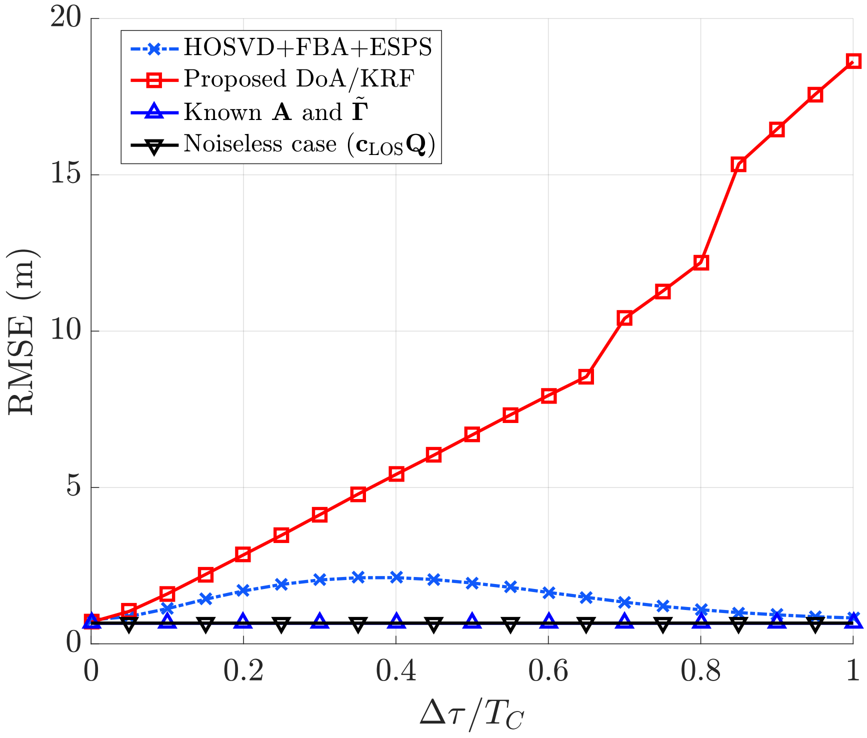


Figure 4.16: Simulation results for DoA/KRF, $D = 1$, $\Delta\phi = \pi/3$, $L_1 = 2$, $\hat{L}_1 = 3$

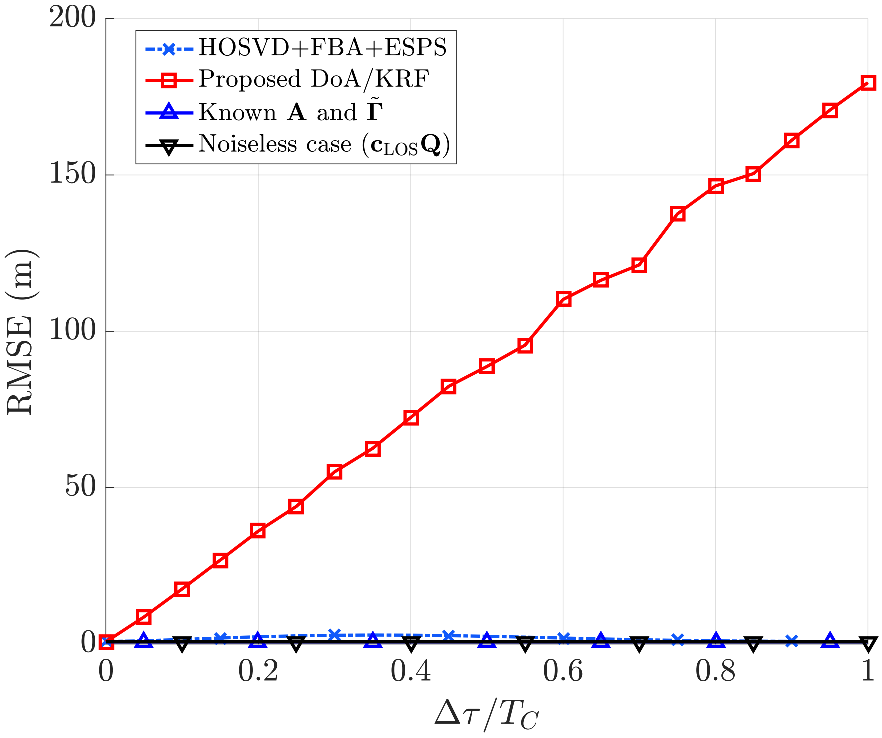


Figure 4.17: Simulation results for DoA/KRF, $D = 1$, $\Delta\phi = \pi/3$, $L_1 = 3$, $\hat{L}_1 = 4$

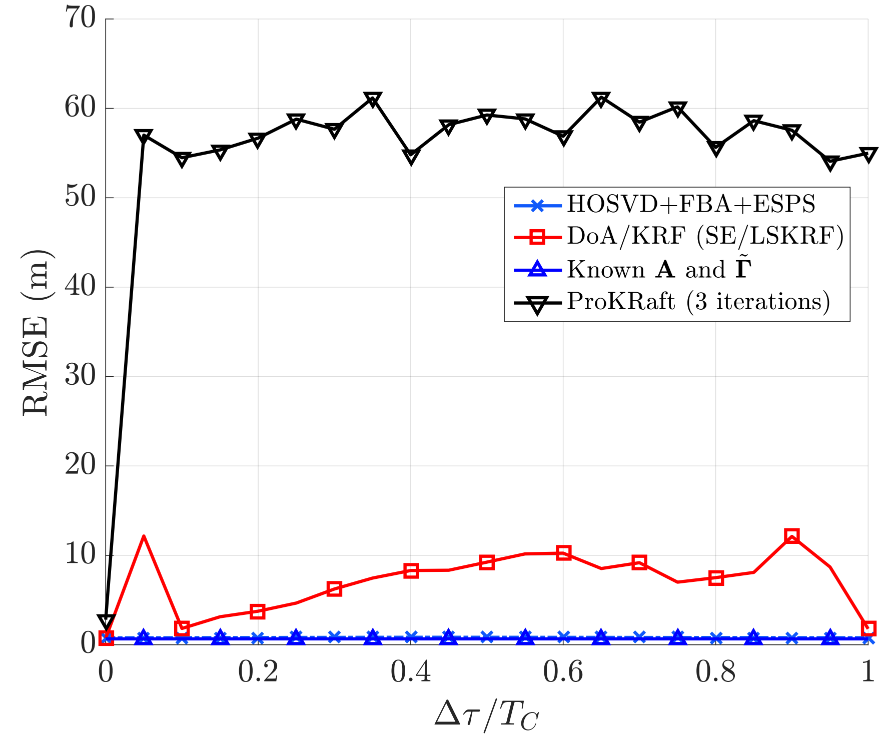


Figure 4.18: Simulation results for ProKRaft, $D = 1$, $\Delta\phi = \pi/3$, $L_1 = 2$, $\hat{L}_1 = 3$

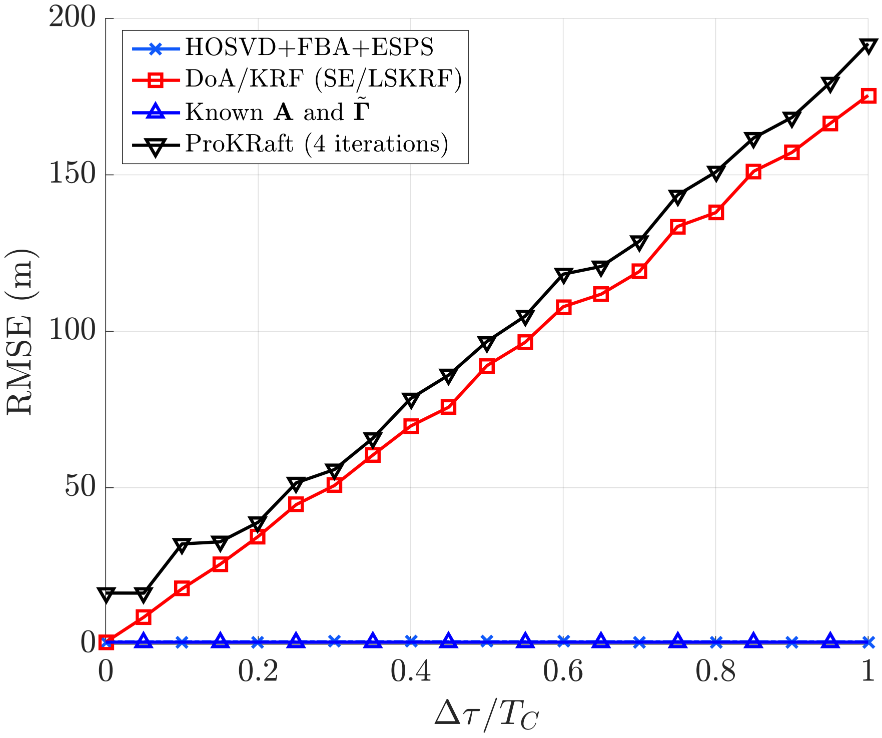


Figure 4.19: Simulation results for ProKRaft, $D = 1$, $\Delta\phi = \pi/3$, $L_1 = 3$, $\hat{L}_1 = 4$

Chapter 5

Conclusion

In this thesis we studied a state-of-the-art HOSVD eigenfiltering approach for multipath mitigation and time-delay estimation [5] and proposed two approaches that significantly outperforms it. The first proposed approach offers a closed-form solution to multipath mitigation which was able to perform well when the phase of the complex amplitude remained constant and when it changed every epoch. The second proposed approach offers an iterative solution which is more complex and only works when the phase of the complex amplitude changed every epoch.

Both proposed approaches are more complex than HOSVD eigenfiltering but both achieved an accuracy close to the optimal case, which is defined as filtering with a priori channel information when model order is correctly estimated. Our scheme completely mitigates the effect of the multipath components being, therefore, very attractive for SCA and LCA.

The advantage of the HOSVD eigenfiltering approach is that no information about the model order is necessary, since it is filtering using the dominant singular vectors to recover the correlated code. While our two proposed approaches depend on knowledge of the model order and will perform worse when the model order is underestimated and will fail when the model order is overestimated. Both spend resources calculating what are called “nuisance parameters” since it also processes information about any existent multipath components.

Future developments could include a simplified correlated path model like the one in [31] to avoid utilizing computational resources on nuisance parameters. This could also be integrated and extended into a low-rank approximation approach to increase precision in time-delay estimation.

Bibliography

[1] K. Uchida, N. Arai, K. Moriya, Y. Miyamoto, T. Kakihara, and T. Tokai, "Development of automatic system for monitoring fishing effort in conger-eel tube fishery using radio frequency identification and global positioning system," *Fisheries Science*, vol. 71, no. 5, pp. 992–1002, 2005. [Online]. Available: http://dx.doi.org/10.1111/j.1444-2906.2005.01056.x

[2] M. Antonini, M. Ruggieri, R. Prasad, U. Guida, and G. F. Corini, "Vehicular remote tolling services using EGNOS," *IEEE Aerospace and Electronic Systems Magazine*, vol. 20, no. 10, pp. 3–8, Oct 2005.

[3] C. Kang, S. H. Lee, and C. C. Chung, "On-Road Vehicle Localization with GPS under Long Term Failure of a Vision Sensor," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, Sept 2015, pp. 1638–1643.

[4] T. Esau, Q. Zaman, D. Groulx, K. Corscadden, Y. Chang, A. Schumann, and P. Havard, "Economic analysis for smart sprayer application in wild blueberry fields," *Precision Agriculture*, vol. 17, no. 6, pp. 753–765, 2016.

[5] B. Hammoud, "Tensor-Based Approach for Multipath Mitigation for GNSS," Master's thesis, Institute for Circuit Theory and Signal Processing, Munich University of Technology, 2015.

[6] M. Weis, F. Roemer, M. Haardt, and P. Husar, "Dual-symmetric parallel factor analysis using procrustes estimation and khatri-rao factorization," in *2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*, Aug 2012, pp. 270–274.

[7] B. Hammoud, F. Antreich, J. A. Nossek, J. P. C. L. da Costa, and A. L. F. de Almeida, "Tensor-Based Approach for Time-Delay Estimation," in *WSA 2016; 20th International ITG Workshop on Smart Antennas*, March 2016, pp. 1–7.

[8] M. S. Braasch and A. J. van Dierendonck, "GPS Receiver Architecture and Measurements," *Proceedings of the IEEE*, vol. 87, no. 1, pp. 48–64, January 1999.

[9] R. D. J. van Nee, "Spread-spectrum code and carrier synchronization errors caused by multipath and interference," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 29, no. 4, pp. 1359–1365, October 1993.

[10] J. J. Spilker Jr. and B. W. Parkinson, Eds., *Global Positioning System: Theory and Applications*, ser. Progress in Astronautics and Aeronautics. American Institute of Aeronautics and Astronautics (AIAA), 1996, vol. I.

[11] R. D. J. van Nee, J. Siereveld, P. C. Fenton, and B. R. Townsend, "The multipath estimating delay lock loop: approaching theoretical accuracy limits," in *Position Location and Navigation Symposium, 1994., IEEE*, April 1994, pp. 246–251.

[12] J. Soubielle, I. Fijalkow, P. Duvaut, and A. Bibaut, "GPS positioning in a multipath environment," *IEEE Transactions on Signal Processing*, vol. 50, no. 1, pp. 141–150, January 2002.

[13] G. Siegert, G. Del Galdo, F. Klier, J. Mahr, G. Rohmer, and A. Ruegamer, "Multidirectional Over The Air testbed for robustness testing of GNSS receivers against Jammers and Spoofers," in *31st AIAA International Communications Satellite Systems Conference*, International Communications Satellite Systems Conferences (ICSSC). Florence, Italy: American Institute of Aeronautics and Astronautics (AIAA), October 2013.

[14] A. Konovaltsev, M. Cuntz, C. Haettich, and M. Meurer, "Performance Analysis of Joint Multi-Antenna Spoofing Detection and Attitude Estimation," in *Proceedings of the 2013 International Technical Meeting of The Institute of Navigation*, San Diego, California, January 2013, pp. 864–872.

[15] L. De Lathauwer, B. De Moor, and J. Vandewalle, "A Multilinear Singular Value Decomposition," *SIAM Journal on Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1253–1278, 2000.

[16] S. U. Pillai and B. H. Kwon, "Forward/backward spatial smoothing techniques for coherent signal identification," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 1, pp. 8–15, January 1989.

[17] M. Haardt, F. Roemer, and G. Del Galdo, "Higher-Order SVD-Based Subspace Estimation to Improve the Parameter Estimation Accuracy in Multidimensional Harmonic Retrieval Problems," *IEEE Transactions on Signal Processing*, vol. 56, no. 7, pp. 3198–3213, July 2008.

[18] T.-J. Shan, M. Wax, and T. Kailath, "On spatial smoothing for direction-of-arrival estimation of coherent signals," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 33, no. 4, pp. 806–811, August 1985.

[19] P. R. B. Gomes, A. L. F. de Almeida, and J. P. C. L. da Costa, "Tensor-based methods for blind spatial signature estimation under arbitrary and unknown source covariance structure," *Digital Signal Processing*, 2016, scheduled to appear.

[20] F. Roemer and M. Haardt, “Tensor-Based Channel Estimation and Iterative Refinements for Two-Way Relaying With Multiple Antennas and Spatial Reuse,” *IEEE Transactions on Signal Processing*, vol. 58, no. 11, pp. 5720–5735, November 2010.

[21] M. Weis, G. Del Galdo, and M. Haardt, “A correlation tensor-based model for time variant frequency selective MIMO channels,” in *WSA Workshop on Smart Antennas*, Vienna, Austria, February 2007.

[22] J. Selva-Vera, “Efficient Multipath Mitigation in Navigation Systems,” Ph.D. dissertation, Department of Signal Signal Theory and Communications, Universitat Politècnica de Catalunya, 2004.

[23] K. Borre, D. Akos, N. Bertelsen, P. Rinder, and S. Jensen, *A Software-Defined GPS and Galileo Receiver: A Single-Frequency Approach*, ser. Applied and Numerical Harmonic Analysis. Birkhäuser Boston, 2007.

[24] M. A. M. Marinho, F. Antreich, J. P. C. L. da Costa, and J. A. Nossek, “A signal adaptive array interpolation approach with reduced transformation bias for DOA estimation of highly correlated signals,” in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2014, pp. 2272–2276.

[25] J. P. C. L. da Costa, F. Roemer, M. Haardt, and R. T. de Sousa, “Multi-dimensional model order selection,” *EURASIP Journal on Advances in Signal Processing*, vol. 2011, no. 1, p. 26, 2011.

[26] K. Liu, J. P. C. L. da Costa, H. C. So, L. Huang, and J. Ye, “Detection of number of components in CANDECOMP/PARAFAC models via minimum description length,” *Digital Signal Processing*, vol. 51, pp. 110 – 123, 2016.

[27] J. P. C. L. da Costa, *Parameter Estimation Techniques for Multi-dimensional Array Signal Processing*, ser. Berichte aus der Kommunikationstechnik. Shaker, 2010.

[28] A. Paulraj, R. Roy, and T. Kailath, “Estimation Of Signal Parameters Via Rotational Invariance Techniques — ESPRIT,” in *Circuits, Systems and Computers, 1985. Nineteeth Asilomar Conference on*, Nov 1985, pp. 83–89.

[29] F. Roemer and M. Haardt, “Tensor-based channel estimation (TENCE) for two-way relaying with multiple antennas and spatial reuse,” in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, April 2009, pp. 3641–3644.

[30] P. H. Schönemann, “A generalized solution of the orthogonal procrustes problem,” *Psychometrika*, vol. 31, no. 1, pp. 1–10, mar 1966.

[31] F. Wendler, F. Antreich, J. A. Nossek, and A. L. Swindlehurst, “Dual-polarization time delay estimation for multipath mitigation,” in *WSA 2015; 19th International ITG Workshop on Smart Antennas*, March 2015, pp. 1–6.

Appendix

Correlation and Noise

Given the compression matrix $\mathbf{Q}_{\omega,d}$ is orthogonal and unitary, the post-correlation noise at the k-th period $\mathbf{N}_{\omega}[k] = \mathbf{N}[k]\mathbf{Q}_{\omega,d} \in \mathbb{C}^{M \times Q}$ preserves the statistical properties of the input noise at the output of the bank of correlators.

$$\begin{aligned}
\mathrm{E}\{\mathbf{N}_{\omega}^{\mathrm{H}}[k]\mathbf{N}_{\omega}[k]\} &= \mathrm{E}\{(\mathbf{N}[k]\mathbf{Q}_{\omega,d})^{\mathrm{H}}\mathbf{N}[k]\mathbf{Q}_{\omega,d}\} \\
&= \mathrm{E}\{\mathbf{Q}_{\omega,d}^{\mathrm{H}} \underbrace{\mathbf{N}^{\mathrm{H}}[k]\mathbf{N}[k]}_{=\sigma_{\mathbf{N}}^{2}\mathbf{I}_{N}} \mathbf{Q}_{\omega,d}\} \\
&= \mathbf{Q}_{\omega,d}^{\mathrm{H}}\sigma_{\mathbf{N}}^{2}\mathbf{I}_{N}\mathbf{Q}_{\omega,d} \\
&= \sigma_{\mathbf{N}}^{2}\mathbf{Q}_{\omega,d}^{\mathrm{H}}\mathbf{Q}_{\omega,d} \\
&= \sigma_{\mathbf{N}}^{2}\mathbf{I}_{Q}.
\end{aligned}$$