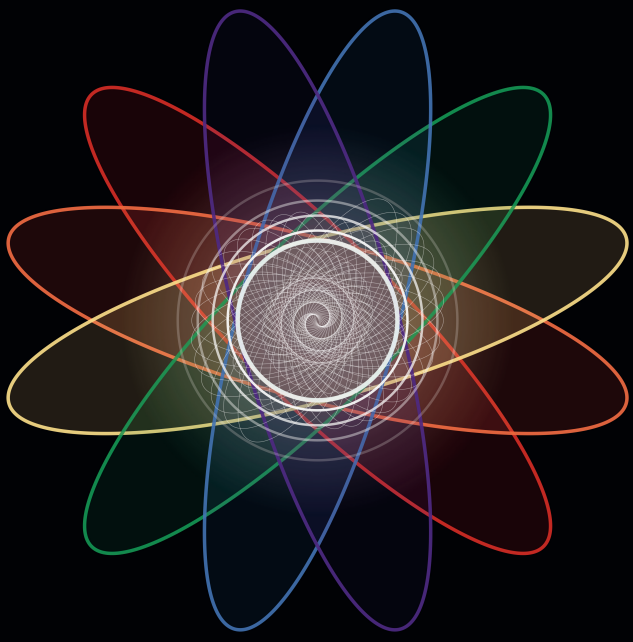
Linköping Studies in Science and Technology Dissertations No. 2359

The Dark Side of Decentralized Target Tracking

Unknown Correlations and Communication Constraints

Robin Forsling



LINKÖPING UNIVERSITY

Linköping studies in science and technology. Dissertations.
No. 2359

The Dark Side of Decentralized Target Tracking

Unknown Correlations and Communication Constraints

Robin Forsling

LINKÖPING UNIVERSITY

Cover illustration: Fusion of estimates with unknown correlations. The colored ellipses represent covariances of estimates to be fused. The white circles represent possible covariances of the fused result given different assumptions about the unknown correlations. The white ellipses are mainly there for show.

Linköping studies in science and technology. Dissertations.
No. 2359

The Dark Side of Decentralized Target Tracking: Unknown Correlations and Communication Constraints

Robin Forsling

robin.forsling@liu.se
www.control.isy.liu.se
Division of Automatic Control
Department of Electrical Engineering
Linköping University
SE–581 83 Linköping
Sweden

ISBN 978-91-8075-409-5 (print)
ISBN 978-91-8075-410-1 (PDF)
ISSN 0345-7524

Printed by LiU-Tryck, Linköping, Sweden 2023

Till Elin och Nils

Abstract

Using sensors to observe real-world systems is important in many applications. A typical use case is target tracking, where sensor measurements are used to compute estimates of targets. Two of the main purposes of the estimates are to enhance situational awareness and facilitate decision-making. Hence, the estimation quality is crucial. By utilizing multiple sensors, the estimation quality can be further improved. Here, the focus is on target tracking in decentralized sensor networks, where multiple agents estimate a common set of targets. In a decentralized context, measurements undergo local preprocessing at the agent level, resulting in local estimates. These estimates are subsequently shared among the agents for estimate fusion. Sharing information leads to correlations between estimates, which in decentralized sensor networks are often unknown. In addition, there are situations where the communication capacity is constrained, such that the shared information needs to be reduced. This thesis addresses two aspects of decentralized target tracking: (i) fusion of estimates with unknown correlations; and (ii) handling of constrained communication resources.

Decentralized sensor networks have unknown correlations because it is typically impossible to keep track of dependencies between estimates. A common approach in this case is to use conservative estimators, which can ensure that the true uncertainty of an estimate is not underestimated. This class of estimators is pursued here. A significant part of the thesis is dedicated to the widely-used conservative method known as covariance intersection (CI), while also describing and deriving alternative methods for CI. One major result related to aspect (i) is the conservative linear unbiased estimator (CLUE), which is proposed as a general framework for optimal conservative estimation. It is shown that several existing methods, including CI, are optimal CLUEs under different conditions.

A decentralized sensor network allows for less data to be communicated compared to its centralized counterpart. Yet, there are still situations where the communication load needs to be further reduced. The communication load is mostly driven by the covariance matrices since, in this scope, estimates and covariance matrices are shared. One way to reduce the communication load is to only exchange parts of the covariance matrix. To this end, several methods are proposed that preserve conservativeness. Significant results related to aspect (ii) include several algorithms for transforming exchanged estimates into a lower-dimensional subspace. Each algorithm corresponds to a certain estimation method, and for some of the algorithms, optimality is guaranteed. Moreover, a framework is developed to enable the use of the proposed dimension-reduction techniques when only local information is available at an agent. Finally, an optimization strategy is proposed to compute dimension-reduced estimates while maintaining data association quality.

Populärvetenskaplig sammanfattning

Att använda sensorer för att övervaka exempelvis personer eller fordon är viktigt inom en rad olika områden. Ett typfall är målföljning, där sensorer och algoritmer används för att skatta positioner och hastigheter hos ett antal mål över tid. Målföljning har studerats under en lång tid inom framförallt militära tillämpningar, men på senare år används det även inom många civila områden som t.ex. inom bilindustrin. Ett klassiskt exempel är problemet där radarmätningar kombineras med rörelsemodeller i ett s.k. kalmanfilter för att rekursivt skatta ett mål med så hög noggrannhet som möjligt.

Genom att använda flera sensorer är det möjligt att göra målföljningen ännu bättre. Ett sensornätverk består av sensorer och beräkningsenheter där de senare ansvarar för att beräkna skattningar av mål. Beräkningsenheterna kan potentiellt även kommunicera skattningar till andra beräkningsenheter. I en centraliserad nätverksdesign skickas mätningarna från alla sensorer till en central beräkningsenhet. Detta leder i många fall till optimala resultat med avseende på målföljningsprestanda. Nackdelen är att sensornätverkets totala förmåga blir beroende av att den centrala beräkningsenheten fungerar. Fallerar den centrala beräkningsenheten blir hela nätverket oanvändbart. För säkerhetskritiska tillämpningar är just den egenskapen ofta oacceptabel. Dessutom blir det i praktiken nästintill omöjligt att realisera en centraliserad lösning för stora sensornätverk, detta med avseende på beräkningskraft, kommunikationsresurser samt komplexitet. Därför är alternativa nätverksdesigner av intresse.

I denna avhandling studeras målföljning i decentraliserade sensornätverk. I decentraliserade sensornätverk skickas mätningar från sensorerna till lokala beräkningsenheter vilka beräknar lokala skattningar av målen. Sedan kommuniceras skattningarna till andra beräkningsenheter där de fusioneras för ökad precision. En fördel med en sådan design är att den saknar kritiska delar vilket ger ett robust sensornätverk där vissa delar kan tillåtas fallera eller vara oåtkomliga utan att nätverkets förmåga bryter ihop totalt. En nackdel med decentraliserade nätverk är att det skapas beroenden mellan skattningar som i många fall inte går att känna till explicit. Ignoreras beroendena riskeras det att redan använd information återanvänds och att den fusionerade skattningens osäkerhet blir mindre än vad den borde vara. Denna avhandling fokuserar på följande två delproblem av decentraliserad målföljning: (i) fusion av skattningar med okända beroenden; och (ii) hantering av en begränsad kommunikationsresurs.

Ett populärt angreppssätt till (i) är att använda konservativa skattare. Orsaken är att en konservativ skattare har egenskapen att inte återanvända information samt att inte underskatta osäkerheten av en beräknad skattning, något som även gäller i fallet med okända beroenden mellan skattningar. På så vis erhålls en robust fusionerad skattning, dock med baksidan att en del av precisionen går förlorad. Ett signifikant resultat kopplat till delproblem (i) är ett ramverk kallat CLUE (eng. conservative linear unbiased estimator) som löser generella konservativa skattningsproblem med hjälp av matematisk optimering. Det visas även hur flertalet etablerade konservativa skattningsmetoder faller ut som specialfall inom CLUE-ramverket.

I delproblem (ii) studeras kommunikationsbegränsningar och hur dessa slår på skattningsegenskaper som konservativitet och prestanda. Först föreslås ett antal metoder som bevarar konservativitet trots att endast en delmängd av osäkerhetsinformationen kommuniceras. Konceptet vidareutvecklas senare till att transformera skattningarna som delas till en lägre dimension. På så vis kan kommunikationskravet reduceras avsevärt. Huvudbidraget för (ii) består i flertalet algoritmer som transformerar kommunicerade skattningar på ett effektivt sätt. Slutligen adresseras två praktiska aspekter av den transformations-baserade strategin för vilka även lösningar föreslås.

Acknowledgments

I would like to express my gratitude to my supervisor, Fredrik Gustafsson, as well as my co-supervisors, Gustaf Hendeby and Zoran Sjanic, for your invaluable guidance and support. I have been fortunate to benefit from your mentorship. In particular, I have appreciated the balance between letting me work freely and providing me with feasible research questions to address. Thanks for all of your help, for your great ideas, and for always having time for discussions.

Thank you, Head of Division, Martin Enqvist, for granting me the opportunity to commence my work at the Division of Automatic Control. Your friendly treatment of people is one of the cornerstones of the pleasant mindset prevailing at Automatic Control, and thanks to all of you at Automatic Control for keeping up the good working environment. Ninna Stensgård, thanks for helping out with all the administrative tasks.

The proofreading of this thesis was conducted by Anton Kullberg, Elin Lager, Magnus Malmström, and Per Boström-Rost. I greatly appreciate the value of your comments and suggestions, which certainly improved the thesis. Special thanks go to you, Gustaf Hendeby, for your 24/7 availability and universal support with everything. You are always there! Many thanks to Anders Hansson for all your effort during our joint work, and to Johan Löfberg for always being able to reply instantly to any optimization related question. An additional thank you to my neighbor, Sofia Melin, for helping out with the artwork in Figure 1.4.

The funding for this work was provided by VINNOVA and Saab AB through the LINK-SIC Competence Center. I would like to thank Svante Gunnarsson, the Center Director, for allowing me to join LINK-SIC, and Sara Strömberg for arranging all the LINK-SIC workshops. I am grateful to Saab AB for providing me with the opportunity to begin my work towards a PhD degree.

Finally, I would like to thank my family for always supporting me, and Nils, for being you. Thank you, Kerstin and Lennart, for helping out and taking care of Nils and our dog companion, Ebba. At last, I would like to express my gratitude to Elin. You are such a good person, and without you, this thesis would not have been possible. I love you.

Linköping, November 2023
Robin Forsling

Contents

Notation

Abbreviations

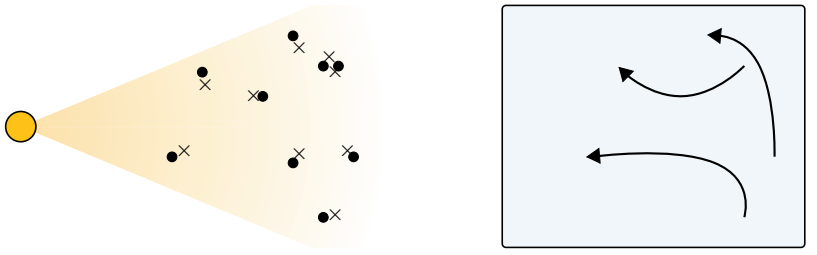
Abbreviation	Meaning
i.f.f.	if and only if
i.i.d.	independent and identically distributed
l.h.s.	left hand side
r.h.s.	right hand side
w.l.o.g.	without loss of generality
w.r.t.	with respect to
ANEES	average normalized estimation error squared
BLUE	best linear unbiased estimator
BSC	Bar-Shalom Campo
CAM	constant acceleration model
CI	covariance intersection
CIE	common information estimate
CLUE	conservative linear unbiased estimator
COIN	conservativeness index
CPM	constant position model
CRLB	Cramér-Rao lower bound
CVM	constant velocity model
DCA	diagonal covariance approximation
DR	dimension-reduced or dimension-reduction
DSN	decentralized sensor network
DTT	decentralized target tracking
EKF	extended Kalman filter
EVP	eigenvalue problem
GEVO	generalized eigenvalue optimization
GEVP	generalized eigenvalue problem
GIMF	generalized information matrix filter
GNN	global nearest neighbor
ICI	inverse covariance intersection
KF	Kalman filter or Kalman fuser
LE	largest ellipsoid (method)
LKF	local (extended) Kalman filter
LMI	linear matrix inequality
MC	Monte Carlo
MSE	mean squared error
NKF	naïve (extended) Kalman filter
PCA	principal component analysis
PCO	principal component optimization
PD	positive definite
PSD	positive semidefinite
RMSE	root mean squared error
RMT	root mean trace
RO	robust optimization
SDP	semidefinite program
SSM	state-space model

1

Introduction

In target tracking, dynamic targets are estimated over time. A target tracking scene is visualized in Figure 1.1. A sensor measures the positions of three targets at multiple time instants. The measurements are corrupted by noise and yield a rough estimate of the targets' instant positions. By processing multiple measurements over time, improved estimates can be obtained. Besides, it is possible to extract additional features, such as the velocities of the targets. This task is accomplished by a target tracking system.

A basic target tracking system is illustrated in Figure 1.2. The processing unit computes target estimates, here denoted tracks, using a sequence of measurements. The tracks are then used in human-machine interface, situational awareness, decision-making, and other high-level functions. Track quality is therefore essential for the target tracking system to be useful.



(a) Targets and measurements **(b)** Estimated target trajectories

Figure 1.1. A target tracking scene with three targets. (a) The black crosses represent the target location measurements at different time instants, while the black dots mark the target locations. (b) The result of processing multiple measurements over time for each target. The black curves and black triangles represent the estimated target trajectories and the latest target estimates, respectively.

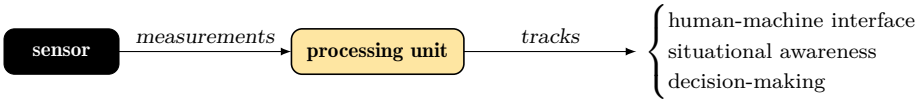
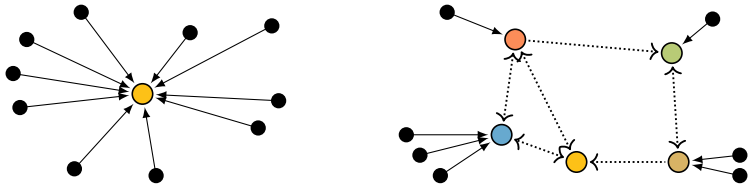


Figure 1.2. A basic target tracking system.



(a) Centralized sensor network **(b)** Decentralized sensor network

Figure 1.3. Sensor networks. Black and colored circles resemble sensor nodes and processing units, respectively. The solid and dotted arrows illustrate sensor-to-processing unit and inter-processing unit communication, respectively.

1.1 Motivation

In network-centric target tracking [112], multiple sensors and processing units constitute the nodes of a network. The nodes are connected in a communication network, which allows for information to be shared among the nodes. The communicated information is subject to data fusion in the processing units. Adding more sensors and processing units facilitates information extraction and sharing, where data fusion techniques can be used to enhance the track quality. The price paid is increased complexity in terms of computations, communication, and system design. How and to what extent the complexity is increased depends on the network design. In a centralized sensor network, see Figure 1.3a, all sensors communicate their measurements to a central processing unit. This arrangement allows for optimal tracking performance but suffers from high communication demands and single points of failure.

An important application of network-centric target tracking is the battle scene depicted in Figure 1.4. Manned and unmanned agents, e.g., aircraft, ships, and land vehicles, cooperate with support from ground stations and cloud services for a common goal. The agents have their own tracking system, with sensors and processing units that are used to extract information and track targets. Information, e.g., track data, is communicated between the agents for improved situational awareness and decision-making. For large networks, a huge amount of critical information must be exchanged. To be practically useful, this type of network needs to be robust and flexible enough to allow for heterogeneous agents to work efficiently in dynamic environments, including switching network topologies. Hence, a centralized sensor network is not suitable for this type of applications.

This thesis focuses on *decentralized target tracking* (DTT). A *decentralized sensor network* (DSN) is illustrated in Figure 1.3b. This type of network is, e.g.,

Figure 1.4. Network-centric battle scene. The network contains heterogenous agents, e.g., aircraft, ships, land vehicles, ground stations, and cloud services. The agents have their own target tracking systems and share information, e.g., track data, with each other.

realized by wireless ad hoc networks such as wireless sensor networks [170]. According to [72], a DSN is characterized by the following properties:

C1 There is no single point of failure. Failure of one node of the network does not cause a complete network breakdown.

C2 There is no central communication management system. Nodes can, in general, only communicate on a node-to-node basis.

C3 There is no global knowledge about the network topology locally available at the nodes.

In a DSN, measurements are passed to a local processing unit, where local tracks are computed. The local tracks are then exchanged between the processing units to enhance the tracking quality by fusion. A decentralized design yields the following advantages over a centralized counterpart [93, 171]:

- *Robustness.* Since there is no single point of failure, a decentralized design is inherently fault-tolerant.
- *Modularity.* The network is decomposed into smaller, self-contained subsystems, which reduces the overall complexity and makes the network scalable.

This also makes the system design process easier since these components can be developed and maintained separately.

- *Flexibility.* It is possible to connect, update, and disconnect the self-contained subsystems on-the-fly.

These properties are vital for the scene illustrated in Figure 1.4.

Remark 1.1. A DSN is an example of distributed processing. However, the terms decentralized and distributed should not be used interchangeably. While distributed sensor networks have multiple interconnected processing units, there is typically a central coordinator or some global processing result is obtained, e.g., globally available tracks [34]. This work is limited to methods that are able to cope with the constraints imposed by C1–C3, thereby excluding many distributed processing techniques.

1.2 Decentralized Target Tracking

Data association and state estimation are fundamental components of the processing unit within a target tracking system. The data association involves the assignment of measurements to tracks, which often requires solving an optimization problem in order to identify the optimal pairings between measurements and tracks based on a specified cost function. In state estimation, the tracks are updated by incorporating the assigned measurements. The existing body of literature pertaining to target tracking, particularly in a centralized setting, is extensive, with numerous publications dedicated to this topic. The famous multihypothesis tracker (MHT, [147]) was developed already in the seventies. More recent multitarget tracking solutions are often based on the random finite set framework [118, 151, 173, 177]. Comprehensive books useful for the design and evaluation of target tracking systems in general are found, e.g., in [17, 18, 27]. However, in the context of DSNs, there remains a considerable amount of work to be undertaken to properly address the specific issues that arise in DTT.

Figure 1.5 provides a schematic view of the processing unit in a DTT system which contains the following functional components:

1. *Measurement-to-track association.* Assigns measurements to the local tracks.
2. *State estimation.* Updates local tracks with the assigned measurements.
3. *Track-to-track association.* Assigns local tracks to the received tracks.
4. *Track fusion.* Fuses local tracks with the assigned tracks.
5. *Communication management.* Decides which track information to exchange.

These functions are run recursively. The dashed line in Figure 1.5 illustrates that local tracks are predicted and used in the next time step. Roughly speaking, a centralized target tracking system involves steps 1–2 but not steps 3–5.

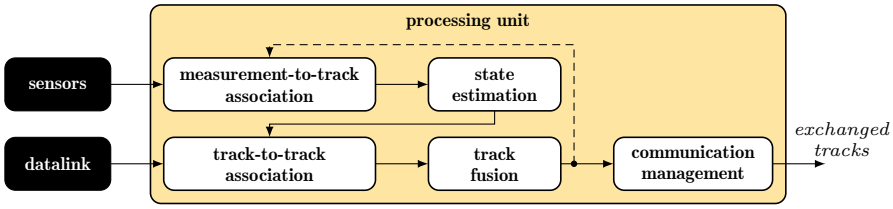


Figure 1.5. A decentralized target tracking system.

Remark 1.2. There are other ways to model a DTT system than the functional breakdown of Figure 1.5. For instance, basically all target tracking systems contain some track management logic, which is excluded here. It should, however, be noted that there is no distinct generic model used to illustrate every target tracking system. In the current scope, the model in Figure 1.5 is sufficiently rich.

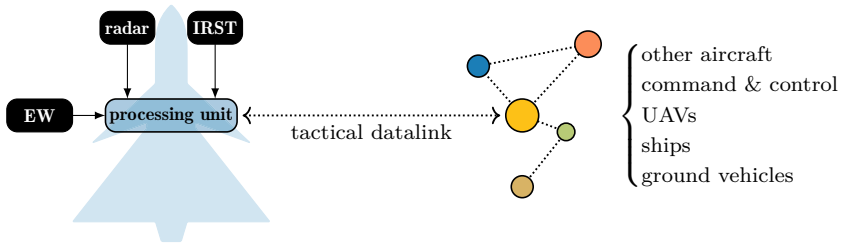
The DSNs considered in this thesis are modeled as a network of agents. Each agent typically has one or several sensors, but in some cases none. Moreover, each agent comprises a self-contained target tracking system and a communication system. The agent can operate on its own under constraints C1–C3 and independently of the other agents. Two important examples of agents are illustrated in Figure 1.6:

- *Fighter aircraft.* A modern fighter aircraft is equipped with many sensors, e.g., radar, infrared search and track (IRST), and electronic warfare (EW) sensors. It typically also has a tactical datalink through which it can communicate tracks with other aircraft, command & control, unmanned aerial vehicles (UAVs), ships, and ground vehicles.

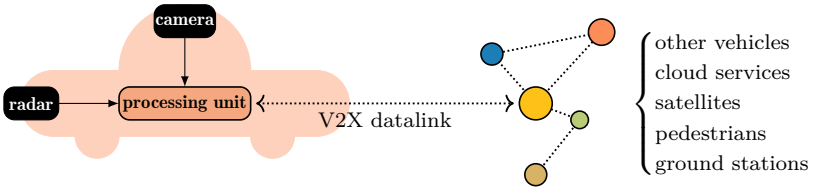
- *Autonomous vehicles.* Modern vehicles, and in particular autonomous vehicles, use many sensors, e.g., radar and cameras, for safe operations in dense environments. In vehicle-to-everything (V2X, [141]) networks, autonomous vehicles use a datalink to communicate with other agents such as other vehicles, pedestrians, satellites, cloud services, and ground stations.

The essence of both agent examples, including the DSNs they operate within, is the huge amount of data that is transferred. Tracks and other estimates are exchanged between agents and further processed. This results in dependencies, i.e., the tracks become correlated with each other. For small, simple networks, the correlations might be tractable. However, in general, it is impossible to maintain explicit knowledge about these correlations among all agents. The issue of having only partial knowledge about correlations is a fundamental aspect of DTT.

In the scope of this thesis, a track is specified by a state estimate, i.e., a vector, and a covariance matrix. These are the quantities that are being exchanged between agents for track fusion. Simply neglecting the correlations generally leads to undesirable results where the actual covariance matrix of a fused track is underestimated. In the worst case, tracks start to diverge due to incorrect uncertainty assessment [93]. To avoid this, the notion of conservativeness has been introduced. A conservative estimator is able to guarantee that the computed covariance matrix is no smaller than the covariance matrix of the actual error. Several estimation



(a) Fighter aircraft (left) connected a tactical sensor network (right)



(b) Autonomous vehicle (left) connected to a V2X network (right)

Figure 1.6. Examples of agents in decentralized sensor networks. Each agent has sensors and a processing unit, and communicates with other agents using a datalink. Examples of possible types of agents in each network are provided to the right of the corresponding subfigure.

methods that are conservative under different conditions exist today. The most well-known include covariance intersection (CI, [90]), inverse covariance intersection (ICI, [130]), and the largest ellipsoid (LE, [24]) method. However, there is no general framework for conservative estimation problems. In the literature, the notion *conservativeness* is frequently referred to as *consistency* or *covariance consistency* [90, 125, 130]. However, to prevent any potential confusion with alternative interpretations of consistency, the term conservativeness is employed throughout this work.

Another crucial aspect of network-centric estimation problems is communication constraints [145]. State estimates and covariance matrices need to be communicated for a legion of targets and to and from many agents. At some point, the communication link will become a bottleneck, and hence the finite size of the communication resource must be taken into account. There are also situations where the communication needs to be reduced for other reasons, e.g., to be able to operate with a low electromagnetic signature. Hence, there is a need to study DTT under communication constraints [100].

1.3 Problem Statement

The main goal of this thesis is to increase the performance and usability of DTT systems. The research focus is related to the *track fusion* and *communication management* components of Figure 1.5. In particular, the following two subproblems are addressed:

P1 Robust track fusion under unknown correlations. To be useful, the track fusion needs to consider both conservativeness and tracking performance.

P2 Efficient usage of the communication resource. The main design criteria for the communication management are the amount of communicated data, conservativeness, and tracking performance.

1.4 Publications

Detailed comments are provided below for each of the publications *included* in the thesis. The authors are abbreviated according to: Robin Forsling (RF), Gustaf Hendeby (GH), Fredrik Gustafsson (FG), Zoran Sjanic (ZS), Benjamin Noack (BN), Anders Hansson (AH), and Johan Löfberg (JL).

Publication I: Consistent Distributed Track Fusion Under Communication Constraints [54]

> R. Forsling, Z. Sjanic, F. Gustafsson, and G. Hendeby. Consistent distributed track fusion under communication constraints. In *Proceedings of the 22nd IEEE International Conference on Information Fusion*, Ottawa, Canada, July 2019.

Summary: This conference paper considers fusion of tracks in a network-centric target tracking problem. Explicit communication constraints are given by only allowing the diagonal elements of track covariance matrices to be communicated. Under such conditions, the conservativeness of a communicated track is typically lost. Several methods are proposed for preserving conservativeness under the assumed communication constraints.

Author's contributions: The idea behind the communication constraints came from ZS. The methods for preserving conservativeness were developed mainly by RF in collaboration with GH. The experimental evaluation was designed by RF with support from all authors. RF authored the paper with input from the coauthors. The implementations, experimental design, and simulations were carried out by RF.

Publication II: Conservative Linear Unbiased Estimation Under Partially Known Covariances [56]

> R. Forsling, A. Hansson, F. Gustafsson, Z. Sjanic, J. Löfberg, and G. Hendeby. Conservative linear unbiased estimation under partially known covariances. *IEEE Transactions on Signal Processing*, 70:3123–3135, June 2022.

Summary: In this journal paper an estimation framework called conservative linear unbiased estimation (CLUE) is formulated for estimation under unknown and partially known correlations. Fundamental properties of the CLUE are derived. A method is proposed based on robust optimization for computing a CLUE in general configurations. It is shown that several established methods are special cases of an optimal CLUE under different conditions.

Author's contributions: FG was the one who first came up with the CLUE idea. RF developed the CLUE theory and related properties, primarily in collaboration with GH and AH. RF proposed, in close collaboration with JL, using robust optimization for computing a CLUE in general cases. RF showed that several established methods are special cases of an optimal CLUE under different conditions. RF authored the paper with input from the co-authors. The examples were designed by RF, who also implemented the algorithms in the evaluation.

Publication III: A Quarter-Century of Covariance Intersection: Correlations Still Unknown? [53]

R. Forsling, B. Noack, and G. Hendeby. A quarter-century of covariance intersection: Correlations still unknown? *IEEE Control Systems Magazine*. Accepted for publication in Sep. 2023. Scheduled for the Apr. 2024 issue. © 2024 IEEE

Summary: This tutorial-like journal paper centers around a widely-used estimation method named covariance intersection (CI). Crucial aspects related to CI are addressed, for instance, sources of correlations, optimality, computational considerations, and alternatives to CI. The paper also acts like a survey for decentralized data fusion in general and for CI and its alternatives in particular. At the end, an outlook is given that predicts future applications and theoretical advances related to CI.

Author's contributions: The idea behind the paper originated from BN who had collected references for a survey paper dedicated to CI. RF contributed with the writing of the main content and side content, sorting out references, designing, and implementing the examples, and designing the outlook. All in close collaboration with BN and GH. RF and BN contributed equally to the paper.

Publication IV: Communication Efficient Decentralized Track Fusion Using Selective Information Extraction [55]

R. Forsling, Z. Sjanic, F. Gustafsson, and G. Hendeby. Communication efficient decentralized track fusion using selective information extraction. In *Proceedings of the 23rd IEEE International Conference on Information Fusion*, Rustenburg, South Africa, July 2020.

Summary: This conference paper proposes an alternative problem formulation compared to Publication I for track fusion under communication constraints. The communication is reduced by the exchange of selectively chosen information projections. This leads to an optimization problem where the information projections are derived by estimating the track fusion result as a function of the information projections.

Author's contributions: In discussions with each co-author, RF proposed the problem formulation as a continuation of Publication I. RF developed the optimization formulation in collaboration with the other authors. Two methods for selecting information were proposed by RF in collaboration with GH. RF authored the paper with input from the co-authors. RF was responsible for carrying out the implementations, experimental design, and simulations.

Publication V: Optimal Linear Fusion of Dimension-Reduced Estimates Using Eigenvalue Optimization [57]

> R. Forsling, Z. Sjanic, F. Gustafsson, and G. Hendeby. Optimal linear fusion of dimension-reduced estimates using eigenvalue optimization. In *Proceedings of the 25th IEEE International Conference on Information Fusion*, Linköping, Sweden, July 2022.

Summary: This conference paper proposes a further development of the concept introduced in Publication IV. An extension of those ideas is formulated as a dimension-reduction problem, which involves transforming exchanged estimates into a lower-dimensional subspace. It was shown that this boils down to an eigenvalue problem for which optimal dimension-reduction can be performed under certain conditions.

Author's contributions: The eigenvalue formulation for dimension-reduction was originally derived by RF, and further developed with inputs from GH and FG. Two dimension-reduction algorithms, corresponding to two different estimation methods, were proposed by RF, who also authored the paper, with input from the co-authors. RF was responsible for carrying out the implementations, experimental design, and simulations.

Publication VI: Decentralized State Estimation In A Dimension-Reduced Linear Regression [59]

> R. Forsling, F. Gustafsson, Z. Sjanic, and G. Hendeby. Decentralized state estimation in a dimension-reduced linear regression. 2023. URL `https://arxiv.org/abs/2210.06947`. Preprint, arXiv.

Summary: This journal paper generalizes the framework proposed in Publication V for additional estimation methods. Optimal guarantees are provided for all of

the dimension-reduction algorithms except the one corresponding to CI. In the case of CI, a convergence analysis is provided. The paper provides an analysis of the communication benefits and an experimental evaluation of the proposed methods. Message coding functionality is also provided. The manuscript has been submitted for possible publication in the *IEEE Transactions on Signal and Information Processing over Networks*.

Author's contributions: RF extended the methodology proposed in Publication V for additional estimation methods in discussions with the co-authors. RF developed the theoretical results, including the convergence analysis, and authored the paper with input from the co-authors. The algorithms for dimension-reduction and message coding were designed and implemented by RF. The experimental evaluation was designed and carried out by RF with input from the co-authors.

Publication VII: Decentralized Data Fusion of Dimension-Reduced Estimates Using Local Information Only [58]

> R. Forsling, F. Gustafsson, Z. Sjanic, and G. Hendeby. Decentralized data fusion of dimension-reduced estimates using local information only. In *Proceedings of the IEEE Aerospace Conference*, Big Sky, MT, USA, Mar. 2023. © 2023 IEEE

Summary: This conference paper addresses a practical aspect of the dimension-reduction techniques developed in Publications V and VI. That is, in reality, an agent who is about to exchange dimension-reduction estimates with another agent needs to have access to information local to the second agent. In general, this is not a realistic assumption. To this end, the common information estimate is proposed as a representation of information shared by the network. Several properties of the common information estimate are derived.

Author's contributions: RF and GH expanded on the methodology suggested in Publication IV to develop the concepts behind the common information estimate. The theoretical properties were derived by RF, who also authored the paper, with input from the co-authors. The implementations, experimental design, and simulations were carried out by RF.

Publication VIII: Track-To-Track Association for Fusion of Dimension-Reduced Estimates [60]

> R. Forsling, Z. Sjanic, F. Gustafsson, and G. Hendeby. Track-to-track association for fusion of dimension-reduced estimates. In *Proceedings of the 26th IEEE International Conference on Information Fusion*, Charleston, SC, USA, June 2023.

Summary: In this conference paper, dimension-reduction is based on data association performance. This is in contrast to Publications IV–VI, where track fusion performance was considered. The proposed problem formulation is analyzed theoretically, and problem properties are highlighted. An optimization algorithm is developed based on the problem analysis.

Author's contributions: RF originally proposed the problem formulation, and the co-authors helped to further develop it. The problem analysis was executed by RF, who also developed the optimization algorithm. RF authored the paper with input from the co-authors. The implementations, experimental design, and simulations were carried out by RF.

Licentiate Thesis: Decentralized Estimation Using Conservative Information Extraction [52]

> R. Forsling. *Decentralized Estimation Using Conservative Information Extraction*. Licentiate Thesis No. 1897, Linköping University, Linköping, Sweden, Dec. 2020.

Summary: The Licentiate thesis has a full overlap with Publications I and IV, and a partial overlap with Publication II. Content exclusive to the Licentiate thesis includes extensions of two estimation methods, namely, inverse covariance intersection and the largest ellipsoid method.

Author's contributions: The contributions exclusive to the Licentiate thesis were developed by RF with inputs from GH, ZS, and FG.

Not Included Publications

Detailed comments are provided below for related publications that have been *excluded*[1] from the thesis. The authors are abbreviated according to: Robin Forsling (RF), Daniel Bossér (DB), Isaac Skog (IS), Gustaf Hendeby (GH), Magnus Lundberg Nordenvaad (MLN), Jakub Matoušek (JM), and Jindřich Duník (JD).

Distributed Point-Mass Filter with Reduced Data Transfer Using Copula Theory [119]

> J. Matoušek, J. Duník, and R. Forsling. Distributed point-mass filter with reduced data transfer using copula theory. In *Proceedings of the 2023 American Control Conference*, pages 1649–1654, San Diego, CA, USA, June 2023.

[1] The publications have been excluded to narrow down the scope of the thesis.

Summary: In this conference paper, a distributed point-mass filter is proposed. The developed filter is applicable for bandwidth-constrained distributed state estimation of stochastic dynamic systems. The proposed solution is benchmarked with well-known moment-based data fusion methods.

Author's contributions: RF authored the introduction and the parts involving the moment-based methods. JM did the main work with input from JD and RF. JM carried out the implementations and simulations with support from RF.

Underwater Environment Modeling for Passive Sonar Track-Before-Detect [28]

> D. Bossér, R. Forsling, I. Skog, G. Hendeby, and M. L. Nordenvaad. Underwater environment modeling for passive sonar track-before-detect. In *OCEANS 2023*, Limerick, Ireland, June 2023.

Summary: This conference paper deals with track-before-detect using passive sonars in a complex underwater environment. Two models relating to the underwater environment are developed: an autoregressive (AR) model is proposed to improve the tracking performance in the case of fluctuating signal energy, and a multi-source model is proposed to estimate spatially distributed background noise.

Author's contributions: RF developed the AR model with input from IS and GH. RF was involved in the authoring of the first two sections, including the parts about the AR model. DB did the main work with input from IS, GH, RF, and MLN. The implementations and simulations were carried out by DB with support from RF.

1.5 Source Code Accessibility

To promote reproducibility, MATLAB® source code related to this thesis is publicly accessible at `https://github.com/robinforsling/dtt`. This includes the source code for all numerical evaluations and all examples where it is applicable. The repository also contains a DTT simulation environment that can be adapted for testing and evaluating new theory and algorithms.

1.6 Thesis Outline and Contributions

The main theme of this thesis is DTT, with a heavy focus on track fusion and communication management aspects. Publications I–VIII are distributed over Chapters 2–5. The chapters are organized as follows:

Chapter 2 provides the mathematical preliminaries, states the linear estimation problem, and defines the notion of conservativeness. Prior work is also outlined. Parts of Publication III appear in Section 2.2.

Chapter 3 introduces the decentralized single-target tracking problem, which is the primary DTT setting studied in this thesis. Methods and models for state estimation and track fusion are provided. Two main sources of correlations are described. Section 3.3 considers a DTT problem under particular communication constraints and is an edited version of Publication I. Section 3.1.4 is based on Publication III.

Chapter 4 introduces a conservative estimation framework applicable to decentralized track fusion problems. Several properties of the framework are derived. A key result is the utilization of a standard optimization methodology for general conservative estimation. It is shown how several established track fusion methods are special cases of the proposed framework. The chapter is essentially an edited version of Publication II. Parts of Publication III appear in Section 4.4.4.

Chapter 5 proposes a framework for communication efficient DTT based on dimension-reduction techniques. The chapter provides two main paradigms, including several submethods, for dimension-reduction. Optimality guarantees are derived for some of the submethods. A solution is proposed to the problem of only having access to local information when reducing dimensionality. Different practical aspects are discussed and an optimization strategy is developed for dimension-reduction in the multitarget configuration. Section 5.1.1 is based on Publications IV–VI. Sections 5.2–5.3 are edited versions of Publication VI. Section 5.4.2 is an edited version of Publication VII. Section 5.5 is an edited version of Publication VIII.

Chapter 6 concludes the thesis and provides a brief future outlook.

2

Preliminaries and Prior Work

The first part introduces necessary mathematical concepts, basic statistics, and linear estimation theory. An optimal linear estimator is defined. The concept of conservativeness is provided with a formal definition. The second part reviews prior work related to decentralized estimation in general, and conservative estimation and communication management in particular.

The content of Section 2.2 is partly based on [53] © 2024 IEEE.

2.1 Preliminaries

Linear algebra and basic statistics are the cornerstones of the theory and methods developed in this work.

2.1.1 Scalars, Vectors, Matrices, and Operators

Denote by $\mathbb{N}$, $\mathbb{R}$, $\mathbb{R}^n$, and $\mathbb{R}^{m\times n}$ the sets of all natural numbers, all real numbers, all n-dimensional real-valued vectors, and all $m \times n$ real-valued matrices, respectively. Let $\mathbb{S}^n_+$ and $\mathbb{S}^n_{++}$ denote the sets of all $n \times n$ symmetric *positive semidefinite* (PSD) matrices and all $n \times n$ symmetric *positive definite* (PD) matrices, respectively. For $A, B \in \mathbb{S}^n_+$, the matrix inequalities $\succeq$ and $\succ$ are defined as

$$A \succeq B \iff (A-B) \in \mathbb{S}^n_+, \qquad A \succ B \iff (A-B) \in \mathbb{S}^n_{++}. \tag{2.1}$$

The identity matrix is denoted I and 0 is a matrix of zeros. The transpose, inverse, and pseudoinverse of a matrix A are denoted A^T, A^{-1}, and A^+, respectively. The rank, trace, and determinant of a matrix A are denoted $\mathsf{rank}(A)$, $\mathsf{tr}(A)$, and $\mathsf{det}(A)$, respectively. Let $A \in \mathbb{S}^n_{++}$. The ellipsoid of A is given by the set of points $\mathcal{E}(A) = \{a \in \mathbb{R}^n \,|\, a^\mathsf{T} A^{-1} a \leq 1\}$. The boundary of this ellipsoid is given by the set

of points $\{a \in \mathbb{R}^n \,|\, a^\mathsf{T} A^{-1} a = 1\}$. For $A, B \in \mathbb{S}^n_{++}$, it holds that

$$A \succeq B \iff \mathcal{E}(A) \supseteq \mathcal{E}(B), \qquad A \succ B \iff \mathcal{E}(A) \supset \mathcal{E}(B). \tag{2.2}$$

Let $A, B \in \mathbb{S}^n_+$. A function $J\colon \mathbb{R}^{n\times n} \to \mathbb{R}$ is *matrix nondecreasing* if

$$A \preceq B \implies J(A) \leq J(B), \tag{2.3}$$

and *matrix increasing* if

$$A \preceq B \wedge A \neq B \implies J(A) < J(B). \tag{2.4}$$

For instance, the function $\mathbf{tr}(WA)$ is matrix nondecreasing if $W \in \mathbb{S}^n_+$, and matrix increasing if $W \in \mathbb{S}^n_{++}$ [29].

Let $\mathcal{V} \subseteq \mathbb{S}^n_+$. An element $A \in \mathcal{V}$ is the *minimum element* of $\mathcal{V}$ if

$$B \succeq A, \forall B \in \mathcal{V}, \tag{2.5}$$

and a *minimal element* of $\mathcal{V}$ if $B \in \mathcal{V}$ and

$$B \succeq A \implies B = A. \tag{2.6}$$

For $A_i \in \mathbb{R}^{m_i \times n}$, where $i = 1, 2, \ldots, N$, $\mathbf{col}(\,\cdot\,)$ is defined as

$$\mathbf{col}(A_1, A_2, \ldots, A_N) = \begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_N \end{bmatrix}. \tag{2.7}$$

For $A_i \in \mathbb{R}^{m_i \times n_i}$, where $i = 1, 2, \ldots, N$, $\mathbf{diag}(\,\cdot\,)$ is defined as

$$\mathbf{diag}(A_1, A_2, \ldots, A_N) = \begin{bmatrix} A_1 & 0 & \ldots & 0 \\ 0 & A_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \ldots & 0 & A_N \end{bmatrix}, \tag{2.8}$$

which is a block diagonal matrix. In case all A_i are scalars, this reduces to a diagonal matrix.

An eigenvalue $\lambda(A)$ of a matrix A is given by [67]

$$Au = \lambda u, \tag{2.9}$$

where u is the associated eigenvector. If $A \in \mathbb{S}^n_{++}$, then the eigenvalue problem (EVP) in (2.9) has n solutions, not necessarily unique. A generalized eigenvalue $\lambda(A, B)$ of the matrix pair (A, B) is given by [136]

$$Au = \lambda B u, \tag{2.10}$$

where u is the associated generalized eigenvector. If $A, B \in \mathbb{S}^n_{++}$ then the generalized eigenvalue problem (GEVP) in (2.10) has n solutions, not necessarily unique. The ith (generalized) eigenvalue λ_i and associated (generalized) eigenvector u_i form the pair (λ_i, u_i). The eigendecomposition of a matrix $A \in \mathbb{S}^n_+$ is defined as

$$A = U\Sigma U^{\mathsf{T}} = \sum_{i=1}^{n} \lambda_i u_i u_i^{\mathsf{T}}, \tag{2.11}$$

where $\Sigma = \mathbf{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n)$, and $U = \begin{bmatrix} u_1 & u_2 & \ldots & u_n \end{bmatrix}$ is an orthogonal matrix such that $u_i^{\mathsf{T}} u_j = \delta_{ij}$ with δ_{ij} denoting the Kronecker delta [67]. The (generalized) eigenvalues are assumed to be in descending order, i.e., $\lambda_{\max} = \lambda_1 \geq \cdots \geq \lambda_n = \lambda_{\min}$.

2.1.2 Statistics

For mathematical clarity, random variables and realizations thereof are explicitly differentiated with respect to (w.r.t.) notation. Boldface is used to express random variables, e.g., $\boldsymbol{a}$, and normal face is used to express a realization of a random variable. For instance, a is a realization of $\boldsymbol{a}$. Let $\mathrm{E}(\boldsymbol{a})$ and $\mathrm{cov}(\boldsymbol{a}) = \mathrm{E}\left((\boldsymbol{a} - \mathrm{E}(\boldsymbol{a}))(\boldsymbol{a} - \mathrm{E}(\boldsymbol{a}))^{\mathsf{T}}\right)$ denote the expected value and covariance matrix of $\boldsymbol{a}$, respectively. The inverse of a covariance matrix is called an information matrix, or simply information. A random variable $\boldsymbol{a}$ is said to be Gaussian distributed with mean $\mu = \mathrm{E}(\boldsymbol{a})$ and covariance matrix $\Sigma = \mathrm{cov}(\boldsymbol{a})$ if $\boldsymbol{a} \sim \mathcal{N}(\mu, \Sigma)$. The cross-covariance matrix of $\boldsymbol{a}$ and $\boldsymbol{b}$ is defined as $\mathrm{cov}(\boldsymbol{a}, \boldsymbol{b}) = \mathrm{E}\left((\boldsymbol{a} - \mathrm{E}(\boldsymbol{a}))(\boldsymbol{b} - \mathrm{E}(\boldsymbol{b}))^{\mathsf{T}}\right)$. For brevity, covariance matrices and cross-covariances matrices are in the following referred to as covariances and cross-covariances, respectively.

2.1.3 Linear Estimation

Let $x \in \mathbb{R}^{n_x}$ be the state to be estimated. It is assumed that x is given in a Cartesian coordinate frame. For $i = 1, 2, \ldots, N$, assume $y_i = H_i x + v_i$, where $y_i \in \mathbb{R}^{n_i}$, $H_i \in \mathbb{R}^{n_i \times n_x}$, and v_i is noise with $\mathrm{cov}(\boldsymbol{v}_i) = R_i$ and $\mathrm{E}(\boldsymbol{v}_i) = 0$. The cross-covariance between $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$ is denoted by $R_{ij} = R_{ji}^{\mathsf{T}}$. Let $n_y = \sum_{i=1}^{N} n_i$,

$$y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}, \quad H = \begin{bmatrix} H_1 \\ H_2 \\ \vdots \\ H_N \end{bmatrix}, \quad R = \begin{bmatrix} R_1 & R_{12} & \ldots & R_{1N} \\ R_{21} & R_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & R_{(N-1)N} \\ R_{N1} & \ldots & R_{N(N-1)} & R_N \end{bmatrix}. \tag{2.12}$$

and $v = \mathrm{col}(v_1, v_2, \ldots, v_N)$, where $y \in \mathbb{R}^{n_y}$ and $H \in \mathbb{R}^{n_y \times n_x}$. Hence, the model

$$\boldsymbol{y} = Hx + \boldsymbol{v}, \qquad R = \mathrm{cov}(\boldsymbol{v}), \tag{2.13}$$

is obtained. The data y and y_i are here interpreted as estimates of Hx and $H_i x$, respectively, but might as well be measurements related to x. By definition

$$\mathrm{cov}(\boldsymbol{y}) = \mathrm{cov}(\boldsymbol{y} - \mathrm{E}(\boldsymbol{y})) = \mathrm{cov}(\boldsymbol{y} - Hx) = \mathrm{cov}(\boldsymbol{v}) = R, \tag{2.14}$$

and similarly, $\mathrm{cov}(\boldsymbol{y}_i) = \mathrm{cov}(\boldsymbol{v}_i) = R_i$.

A property of a general $n_y \times n_y$ covariance R is that $R \in \mathbb{S}_+^{n_y}$. However, here it is frequently assumed that $R \in \mathbb{S}_{++}^{n_y}$. The difference is, in many cases, only technical. If $R \in \mathbb{S}_+^{n_y}$ and R is singular, then the pseudoinverse R^+ is used instead of R^{-1}. Unless otherwise stated, it is assumed that $R \in \mathbb{S}_{++}^{n_y}$. It is always assumed that $R_i \in \mathbb{S}_{++}^{n_i}$.

Consider the model in (2.13). In *linear estimation*, an estimator $\hat{\boldsymbol{x}}$ of x with covariance P is computed as

$$\hat{\boldsymbol{x}} = K\boldsymbol{y}, \qquad P = KRK^{\mathsf{T}}, \tag{2.15}$$

where $K = \begin{bmatrix} K_1 & K_2 & \dots & K_N \end{bmatrix} \in \mathbb{R}^{n_x \times n_y}$ is the estimation gain. A particular value $\hat{x} = Ky$ of $\hat{\boldsymbol{x}}$ is called an *estimate* of x, and P is also referred to as the estimate covariance. For brevity, the pair $(\hat{x}, P)$ is often just referred to as an estimate. The *estimation error* is defined as $\tilde{\boldsymbol{x}} = \hat{\boldsymbol{x}} - x$, and $\tilde{x} = \hat{x} - x$ is the true error of the estimate. The quantity $\mathrm{E}(\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}^{\mathsf{T}})$ is the covariance of the estimation error, here also referred to as the true covariance of the estimator or estimate. Moreover, the quantity $\mathrm{E}(\tilde{\boldsymbol{x}}^{\mathsf{T}}\tilde{\boldsymbol{x}})$ is the *mean squared error* (MSE) of $\hat{\boldsymbol{x}}$ [96].

If $\mathrm{E}(\hat{\boldsymbol{x}}) = x$, then $\hat{\boldsymbol{x}}$ is an *unbiased estimator* of x. By assumption, $\boldsymbol{y} = Hx + \boldsymbol{v}$ and $\mathrm{E}(\boldsymbol{v}) = 0$. Hence

$$\mathrm{E}(\hat{\boldsymbol{x}}) = \mathrm{E}(K\boldsymbol{y}) = \mathrm{E}\left(K(Hx + \boldsymbol{v})\right) = KHx + \mathrm{E}(\boldsymbol{v}) = KHx, \tag{2.16}$$

from which it follows that $\mathrm{E}(\hat{\boldsymbol{x}}) = x \iff KH = I$.

Remark 2.1. In many cases, there is no need to differentiate between an estimator $\hat{\boldsymbol{x}}$ (or $\boldsymbol{y}$) and the estimate $\hat{x}$ (or y_i); the actual meaning is given by the context. This explicit notation is kept for consistency since it is useful in some parts of this thesis.

In practice, it is often desirable to use an estimator that is optimal w.r.t. some loss function $J(P)$ of P. Here, $J(P)$ is assumed to be a matrix increasing function. The *best linear unbiased estimator* (BLUE) is defined in Definition 2.2. This alternative definition of the BLUE is adapted from [56].

Definition 2.2 (Best Linear Unbiased Estimator). Assume $\boldsymbol{y} = Hx + \boldsymbol{v}$, where $R = \mathrm{cov}(\boldsymbol{v})$. An estimator $\hat{\boldsymbol{x}}^\star = K^\star\boldsymbol{y}$ with $P^\star = K^\star R(K^\star)^{\mathsf{T}}$ is the *best linear unbiased estimator* if $K^\star$ is the solution to

$$\begin{aligned} \underset{K}{\text{minimize}} \quad & J(P) \\ \text{subject to} \quad & KH = I \\ & P = KRK^{\mathsf{T}}, \end{aligned} \tag{2.17}$$

for a given matrix increasing function J.

The solution to (2.17) is given by $K^\star = (H^{\mathsf{T}}R^{-1}H)^{-1}H^{\mathsf{T}}R^{-1}$. Hence, the BLUE estimate is computed as [94]

$$\hat{x}^\star = (H^{\mathsf{T}}R^{-1}H)^{-1}H^{\mathsf{T}}R^{-1}y, \qquad P^\star = (H^{\mathsf{T}}R^{-1}H)^{-1}. \tag{2.18}$$

In fact, by the *Gauss-Markov theorem*, it follows that $KRK^{\mathsf{T}} \succeq P^\star$ for all K such that $KH = I$. As a consequence, an estimate computed according to (2.18) is optimal for every matrix increasing $J(P)$ among all linear unbiased estimators [94]. In addition, since $\mathsf{tr}(P) = \mathsf{tr}(\mathrm{E}(\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}^{\mathsf{T}})) = \mathrm{E}(\tilde{\boldsymbol{x}}^{\mathsf{T}}\tilde{\boldsymbol{x}})$ is a matrix increasing function, the BLUE also is MSE optimal. Example 2.3 illustrates a BLUE.

Example 2.3: Best Linear Unbiased Estimator

Let $n_x = N = 2$. Assume that $H_1 = H_2 = I$, and that

$$R_1 = \begin{bmatrix} 9 & -2 \\ -2 & 2 \end{bmatrix}, \qquad R_2 = \begin{bmatrix} 2 & 2 \\ 2 & 9 \end{bmatrix}, \qquad R_{12} = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix},$$

are given. Construct H and R according to (2.12). The BLUE covariance $P^\star$ is computed using (2.18) which results in

$$P^\star = \begin{bmatrix} 0.89 & -0.77 \\ -0.77 & 1.78 \end{bmatrix}.$$

In Figure 2.1, $P^\star$ is compared to $P = KRK^{\mathsf{T}}$, where K is randomly generated subject to $KH = I$. The figure demonstrates that under these circumstances $\mathcal{E}(P^\star) \subseteq \mathcal{E}(KRK^{\mathsf{T}})$. This is an implication of the Gauss-Markov theorem, i.e., if an arbitrary K is such that $KH = I$, then $KRK^{\mathsf{T}} \succeq P^\star$.

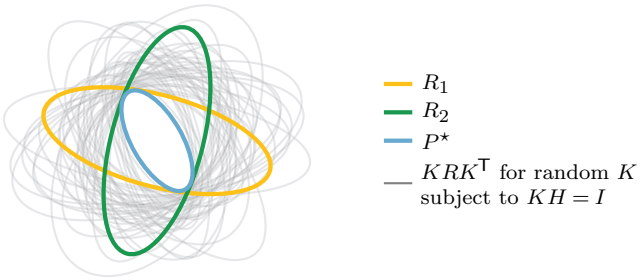


Figure 2.1. Results of Example 2.3. A geometrical interpretation of the BLUE is that the ellipse of every KRK^{T} subject to $KH = I$ is larger than, or equal to, the ellipse of the BLUE covariance $P^\star$.

2.1.4 Conservative Estimation

A central concept in this thesis is *conservativeness*. A conservative estimator is defined as:

Definition 2.4 (Conservative Estimator). An estimator $\hat{\boldsymbol{x}}$, with associated covariance P, is called a *conservative estimator* if

$$P - \mathrm{E}(\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}^{\mathsf{T}}) \succeq 0. \tag{2.19}$$

If this holds, then $(\hat{x}, P)$ is called a *conservative estimate*.

This property becomes relevant if, for instance, R is only partially known. If R is fully known, then for any linear estimator $P = KRK^\mathsf{T} = \mathrm{E}(\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}^\mathsf{T})$ such that $P - \mathrm{E}(\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}^\mathsf{T}) \equiv 0$, and hence $(\hat{x}, P)$ is a conservative estimate. Example 2.5 illustrates an estimator that is not conservative.

Example 2.5: A Non-Conservative Estimator

Assume the same setting as in Example 2.3, except that R_{12} is now unknown. A naïve way of handling that R_{12} is unknown is by assuming $R_{12} = 0$, and then use the BLUE given this assumption. This results in the gain and covariance

$$K = (H^\mathsf{T} B^{-1} H)^{-1} H^\mathsf{T} B^{-1}, \qquad P = (H^\mathsf{T} B^{-1} H)^{-1} = \begin{bmatrix} 1.27 & 0 \\ 0 & 1.27 \end{bmatrix},$$

where $B = \mathsf{diag}(R_1, R_2)$. However, in this case the covariance $P \neq \mathrm{E}(\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}^\mathsf{T})$, where $\tilde{\boldsymbol{x}} = K\boldsymbol{y} - x$ is the estimation error. The covariance $\mathrm{E}(\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}^\mathsf{T})$ is as usual given by

$$\mathrm{E}(\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}^\mathsf{T}) = KRK^\mathsf{T} = \begin{bmatrix} 1.14 & -0.64 \\ -0.64 & 1.87 \end{bmatrix}.$$

Hence, $P - \mathrm{E}(\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}^\mathsf{T}) \not\geq 0$, which means that this estimator is not conservative. The example is illustrated in Figure 2.2, where also $P^\star$ from Example 2.3 is provided.

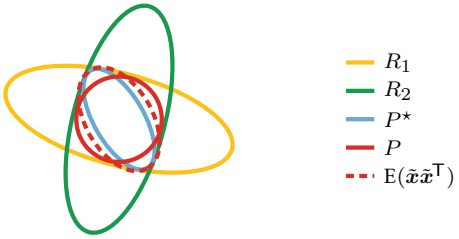


Figure 2.2. Results of Example 2.5. Since $\mathcal{E}(\mathrm{E}(\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}^\mathsf{T})) \not\subseteq \mathcal{E}(P)$, the estimate $(\hat{x}, P)$ is not conservative.

2.2 Review of Prior Work

This section starts with a brief historical overview of almost five decades of decentralized and distributed estimation. Then follows a review of work specifically related to conservative track fusion and communication management.

2.2.1 A Historical Perspective on Decentralized and Distributed Estimation

The topic of estimation in distributed and decentralized sensor networks has undergone significant development, resulting in the invention of several effective methodologies. A main issue arises from the management of correlations among estimates, resulting in different underlying assumptions for each approach.

Pioneering work focused on distributed implementations of the *Kalman filter* (KF). The technical report in [178] outlines several formulations of the KF algorithm for managing multisensor data in a distributed setting. The method in [78] exploits the structure of systems split into interconnected linear dynamical subsystems in order to decompose the KF. The authors of [162] have derived an optimal distributed version of the KF, which is extended in [179] and [26], and is similar to the later derived distributed KF in [68, 99]. Such methods represent algebraic reformulations of the KF and thereby produce optimal estimates under certain conditions, but restrict the estimates to be merged to follow specific filtering schemes. Although some relaxations exist, e.g., [148], they require specific prerequisites that can limit their use in complex network architectures. In particular, these methods prohibit a fully decentralized solution, cf. properties C1–C3 in Section 1.1.

A widely used class of decentralized data fusion methods relates to hierarchical systems that implement *tracklet* or *channel filter* fusion [39, 72, 77]. Other examples include the *generalized information matrix filter* (GIMF) and derivates thereof [168, 169]. These filters keep track of and explicitly subtract common information from the fusion result. By storing the state over multiple time steps, correlations induced by process noise can also be addressed to a certain degree [120, 121]. Specific methods tailored for common process noise reconstruct the joint covariances [149]. For such approaches, sample-based techniques prove to be effective [142, 163].

Another important class of distributed estimation algorithms is consensus filtering [133, 134, 183]. Initial consensus schemes did not reliably compute the covariances. By combining consensus filtering with CI, [19] have derived robust and conservative methods, which have been further developed, for instance, in [166], to increase robustness to network failures. Similarly, [84] have integrated CI into diffusion-based distributed KFs to provide conservative estimates.

For further reading about early contributions in the field of distributed and decentralized filtering, the survey provided in [97] is suggested. More recent overviews, with a particular focus on target tracking, are found in [34, 74, 110, 161]. Multisensor fusion for robotic systems is discussed in [116]. A broad review of multisensor data fusion techniques is provided by [98], which, e.g., includes the handling of imprecise information, outliers, and conflicting data. Network-centric operations are considered in the review [165]. In particular, network-induced effects such as packet delays and losses or quantization are studied. Distributed estimation is a key tool for multisensor data fusion, and [31] presents a broad overview of the underlying theory and the required Bayesian inference techniques. A concise discussion and review of the past forty years of distributed estimation presented in [40]. Modern reviews of estimation under unknown correlations can be found in [14, 106] and [53].

2.2.2 Conservative Track Fusion

Due to properties C1–C3 in Section 1.1, correlations between tracks are in general unknown [37, 90] when it comes to track fusion in DTT. Nevertheless, the correlations need to be handled carefully. Otherwise, information will be reused. In these settings, conservative methods provide a reliable option for track fusion [93].

The most famous method for conservative track fusion under unknown correlations is CI. A typical application for CI is target tracking [158]. Simultaneous localization and mapping (SLAM) applications using CI are studied in [91, 92, 127, 191]. CI has further been used for problems ranging from localization of multi-robot systems [174] and cooperative localization of robot swarms [101] over time-of-arrival localization [102] to automotive tracking, human tracking [89], data validation [76], and medical application [186]. Theoretical aspects of CI are studied in [36, 37, 150, 171].

In target tracking applications, CI needs to handle the fusion of estimates over multiple time steps in a dynamic environment. However, recursive application of CI over time may lead to unnecessarily large covariances, as shown in [11]. Therefore, MSE optimality is not preserved when applying CI recursively in dynamic systems [7]. Several fusion schemes involving CI have been developed to better preserve information over time. The algorithms in [3] and [4] are based upon fusion without and with memory, respectively, and feedback mechanisms to more efficiently use information in dynamic systems. Different robust fusion mechanisms for merging multiple estimates in time-varying systems are studied in [45, 140], where CI is compared to different fusion techniques using the joint covariance.

Since CI in many cases is overly conservative, it is relevant to consider other methods for track fusion. Two important alternatives to CI are ICI [130] and the LE method [24], which exploits partial knowledge about the correlations. Hence, these methods are able to utilize information more efficiently, but at the cost of only providing conservative results, given that certain assumptions are fulfilled. ICI and LE are evaluated in practical scenarios in [131, 132]. Theoretical properties are studied in [128, 129, 159]. Exploitation of partial knowledge is also studied in, e.g., [5, 6, 9, 10, 91, 130, 160, 182], and is in particular a major part of Chapter 4 of this thesis.

In [44, 65, 153, 157, 176], problems related to optimal conservative track fusion are solved using minimax optimization. These papers relax the conservativeness constraint into a scalar constraint. By doing this, it is possible to derive estimators with smaller covariances. However, as shown in the counterexample in [56], the solution to this minimax problem does not necessarily produce conservative estimates. Similarly, in [189], a minimax formulation is suggested where the MSE is minimized for the maximum possible cross-correlations. A game-theoretic approach is considered in [103] in terms of a two-player game, where one player tries to minimize the MSE while the other tunes the correlations to maximize the MSE. A full maximum cross-covariance is computed in [185]. These approaches have in common that they try to represent the maximum possible correlations. However, as the maximum possible correlations are typically not unique, these approaches cannot, in general, guarantee conservativeness.

2.2.3 Communication Management

In [187], it is shown that the problem of finding an optimal linear mapping that compresses measurements to be exchanged to a fusion node boils down to an eigenvalue problem. In [35], the problem is extended to the dynamic case. Estima-

tion based on dimension-reduced data is handled in a distributed context in [190] and [156], where in the latter also a non-ideal communication channel is considered. The authors of [50] address the problem of jointly assigning the dimensionality and deriving optimal data compression matrices. In [46], the authors propose communication efficient algorithms for distributed estimation based on transforming the data using linear mappings. Performance bounds of data compression techniques are analyzed in [117]. A framework for dimension-reduction including data denoising for distributed algorithms is proposed in [156]. There is numerous additional work on closely related dimension-reduction problems, see, e.g., [48, 49, 51, 155]. It is also the topic of Chapter 5, where dimension-reduced estimates are computed based on eigenvalue optimization similar to the work [187]. The authors of [69, 70] use a gradient descent algorithm to optimize the dimension-reduction.

An alternative strategy for reducing the communication load is to quantize the data to be exchanged, see, e.g., [124, 152, 172]. A key aspect of quantization in DSN is how to preserve conservativeness when the communicated data is quantized [63, 64]. In this thesis, however, it is assumed that quantization effects can be neglected.

3

Decentralized Single-Target Tracking

The considered decentralized target tracking (DTT) problem is now defined in a single-target tracking context. Figure 3.1 illustrates a scenario where multiple agents track a common target in a decentralized sensor network (DSN). The scenario encompasses state estimation, track fusion, and communication management, but no data association. The state estimation is solved using an *extended Kalman filter* (EKF). Several track fusion methods are defined and evaluated. Two main sources of correlations between estimates can be attributed to common process noise and shared information. In addition, it is demonstrated how ignoring correlations leads to a loss of conservativeness. At the end, the DSN-based single-target tracking example is evaluated under communication constraints. In particular, the agents are allowed to exchange only the diagonal entries of the local covariance matrices. The contents of this chapter are based on [54] and [53]

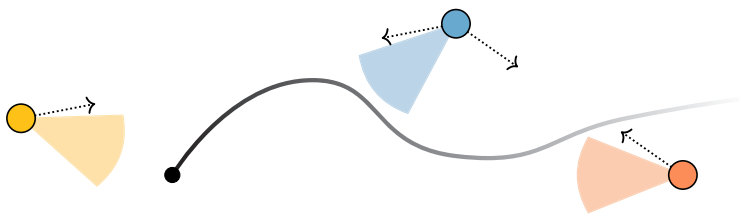


Figure 3.1. A single-target tracking scenario. Multiple agents (colored circles) track a dynamic target (black circle) using sensors (colored cones) and internal processing units. The agents exchange local track estimates over a datalink (dotted arrows) for track fusion.

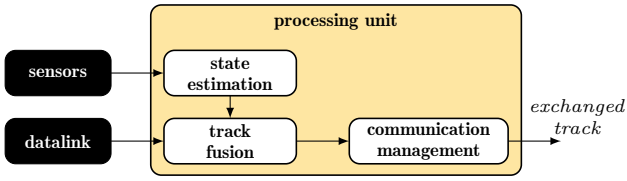


Figure 3.2. A decentralized single-target tracking system.

3.1 A Decentralized Single-Target Tracking System

In the single-target tracking case, the measurement-to-track association and track-to-track association are often trivial and can be disregarded. This case is assumed here. Hence, the DTT system comprises state estimation, track fusion, and communication management. A decentralized single-target tracking system is illustrated in Figure 3.2.

3.1.1 State Estimation

A *state-space model* (SSM) is used to describe the state of a dynamic target. The SSM comprises a *process model* for the target dynamics and a *measurement model* that relates measurements to the target state. Let x_k be the target state at time k, then a family of discrete time SSMs is given by

$$x_{k+1} = F_k x_k + w_k, \qquad \boldsymbol{w}_k \sim \mathcal{N}(0, Q_k), \tag{3.1a}$$

$$z_k = h(x_k) + e_k, \qquad \boldsymbol{e}_k \sim \mathcal{N}(0, C_k), \tag{3.1b}$$

where F_k is the state transition model, Q_k is the covariance of the process noise $\boldsymbol{w}_k$, z_k is a measurement vector, h is a nonlinear measurement function, and C_k is the covariance of the measurement noise $\boldsymbol{e}_k$. It is assumed that $\operatorname{cov}(\tilde{\boldsymbol{x}}_{k|l}, \boldsymbol{w}_k) = 0$ for all $k \geq l$, where $\tilde{\boldsymbol{x}}_{k|l} = \hat{\boldsymbol{x}}_{k|l} - x_k$. Moreover, it is assumed that $\operatorname{cov}(\boldsymbol{w}_k, \boldsymbol{e}_l) = 0$ for all k, l. Subscript $k|l$ denotes filtered quantities evaluated at time k using measurements up to and including time l.

The SSM in (3.1) is assumed throughout this work. The process model is linear, but the measurement model in general involves a nonlinear function. The EKF [86] in Algorithm 3.1 is used to recursively compute a state estimate of x_k using a time update and a measurement update in each filter recursion. If $h(x_k) = H_k x_k$, then the EKF in Algorithm 3.1 reduced to the linear Kalman filter (KF, [95]). In this scope $(\hat{x}_{k|k}, P_{k|k})$ is referred to as a *track*, a track estimate, or simply an estimate.

Process Models

A linear process model is fully specified by F_k and Q_k. Here, a *constant position model* (CPM), a *constant velocity model* (CVM), and a *constant acceleration model* (CAM) are used [107]. Let T_s be the sampling time from k and $k+1$, and let $\otimes$ denote the Kronecker product. Let I_d be the $d \times d$ identity matrix, where

Algorithm 3.1: Extended Kalman Filter

Input: State-space model (3.1), initial values $\hat{x}_{0|0} = \hat{x}_0$ and $P_{0|0} = P_0$

Time update:

$$\hat{x}_{k+1|k} = F_k \hat{x}_{k|k}, \qquad P_{k+1|k} = F_k P_{k|k} F_k^{\mathsf{T}} + Q_k. \tag{3.2}$$

Measurement update:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \left(z_k - h(\hat{x}_{k|k-1}) \right), \qquad P_{k|k} = (I - K_k H_k) P_{k|k-1}, \tag{3.3}$$

where $K_k = P_{k|k-1} H_k^{\mathsf{T}} (H_k P_{k|k-1} H_k^{\mathsf{T}} + C_k)^{-1}$ and $H_k = \frac{\partial h(x')}{\partial x'} \Big|_{x'=\hat{x}_{k|k-1}}$.

Output: $(\hat{x}_{k|k}, P_{k|k})$

Table 3.1
STATE VECTOR x FOR DIFFERENT PROCESS MODELS

d	CPM	CVM	CAM
2	$x = [\mathsf{x}\ \mathsf{y}]^{\mathsf{T}}$	$x = [\mathsf{x}\ \mathsf{y}\ \mathsf{v_x}\ \mathsf{v_y}]^{\mathsf{T}}$	$x = [\mathsf{x}\ \mathsf{y}\ \mathsf{v_x}\ \mathsf{v_y}\ \mathsf{a_x}\ \mathsf{a_y}]^{\mathsf{T}}$
3	$x = [\mathsf{x}\ \mathsf{y}\ \mathsf{z}]^{\mathsf{T}}$	$x = [\mathsf{x}\ \mathsf{y}\ \mathsf{z}\ \mathsf{v_x}\ \mathsf{v_y}\ \mathsf{v_z}]^{\mathsf{T}}$	$x = [\mathsf{x}\ \mathsf{y}\ \mathsf{z}\ \mathsf{v_x}\ \mathsf{v_y}\ \mathsf{v_z}\ \mathsf{a_x}\ \mathsf{a_y}\ \mathsf{a_z}]^{\mathsf{T}}$

d is the number of spatial dimensions. Table 3.1 summarizes what x contains for different process models and d, with omitted time indices. The x-, y-, and z-coordinates are denoted x, y, and z, respectively. The corresponding velocities are denoted $\mathsf{v_x}$, $\mathsf{v_y}$, and $\mathsf{v_z}$, respectively, and the corresponding accelerations are denoted $\mathsf{a_x}$, $\mathsf{a_y}$, and $\mathsf{a_z}$, respectively.

The CPM is defined as

$$F_k = I_d, \qquad Q_k = \sigma_{w,k}^2 T_s I_d, \tag{3.4}$$

where $\sigma_{w,k}^2$ is the variance of the process. This parameter is related to the maneuverability of the estimated target. A more maneuverable target has a larger $\sigma_{w,k}^2$ compared to a less maneuverable target. The initial estimate $(\hat{x}_{0|0}, P_{0|0})$ is derived from the initial measurement and covariance (z_0, C_0).

The CVM is defined as

$$F_k = I_d \otimes \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix}, \qquad Q_k = I_d \otimes \sigma_{w,k}^2 \begin{bmatrix} \frac{T_s^3}{3} & \frac{T_s^2}{2} \\ \frac{T_s^2}{2} & T_s \end{bmatrix}. \tag{3.5}$$

The position components of the estimate are initialized as in the CPM case. The velocity components are initialized to zero with the covariance initialized to $\mathsf{v}_{\max}^2 I_d$, where $\mathsf{v}_{\max}$ is the assumed maximum velocity along one coordinate.

The CAM is defined as

$$F_k = I_d \otimes \begin{bmatrix} 1 & T_s & \frac{T_s^2}{2} \\ 0 & 1 & T_s \\ 0 & 0 & 1 \end{bmatrix}, \qquad Q_k = I_d \otimes \sigma_{w,k}^2 \begin{bmatrix} \frac{T_s^5}{20} & \frac{T_s^4}{8} & \frac{T_s^3}{6} \\ \frac{T_s^4}{8} & \frac{T_s^3}{3} & \frac{T_s^2}{2} \\ \frac{T_s^3}{6} & \frac{T_s^2}{2} & T_s \end{bmatrix}. \tag{3.6}$$

The position and velocity components of the estimate are initialized as in the CVM case. The acceleration components are initialized to zero with the covariance initialized to $\mathsf{a}_{\max}^2 I_d$, where $\mathsf{a}_{\max}$ is the assumed maximum acceleration along one coordinate.

Measurement Models

Two types of sensor models are considered: a *linear* sensor and a *radar-like* sensor. Under the given SSM assumption of (3.1), the measurement model is fully specified by $h(x_k)$ and C_k. In both cases described below, it is assumed that the measurement noise variances are constant over time.

In the linear case, it is assumed that

$$h(x_k) = H_k x_k = \begin{bmatrix} I_d & 0 \end{bmatrix} x_k, \qquad C_k = \sigma_e^2 I_d, \tag{3.7}$$

where 0 is a zero-matrix of appropriate size, and σ_e^2 is the measurement noise variance in each measured coordinate.

Radar-like sensors measure in *polar coordinates* if $d = 2$, or *spherical coordinates* if $d = 3$. A measurement z_k is given w.r.t. the sensor location x_k^s. If $d = 2$ and $x_k^s = 0$, then the measurement model is defined as

$$h(x_k) = \begin{bmatrix} r_k \\ \theta_k \end{bmatrix} = \begin{bmatrix} \sqrt{\mathsf{x}_k^2 + \mathsf{y}_k^2} \\ \operatorname{atan2}(\mathsf{y}_k, \mathsf{x}_k) \end{bmatrix}, \qquad C_k = \operatorname{diag}(\sigma_r^2, \sigma_\theta^2), \tag{3.8}$$

where $\operatorname{atan2}(\,\cdot\,)$ is the 2-argument arctangent, and r_k and θ_k are the radial distance and azimuth angle to the target, respectively, at time k. Moreover, σ_r^2 and σ_θ^2 are the measurement noise variances of r_k and θ_k, respectively. If $d = 3$ and $x_k^s = 0$, then the measurement model is defined as

$$h(x_k) = \begin{bmatrix} r_k \\ \theta_k \\ \phi_k \end{bmatrix} = \begin{bmatrix} \sqrt{\mathsf{x}_k^2 + \mathsf{y}_k^2 + \mathsf{z}_k^2} \\ \operatorname{atan2}(\mathsf{y}_k, \mathsf{x}_k) \\ \operatorname{atan2}\left(\mathsf{z}_k, \sqrt{\mathsf{x}_k^2 + \mathsf{y}_k^2}\right) \end{bmatrix}, \qquad C_k = \operatorname{diag}(\sigma_r^2, \sigma_\theta^2, \sigma_\phi^2), \tag{3.9}$$

where ϕ_k is the elevation angle to the target at time k, and σ_ϕ^2 is the measurement noise variance of ϕ_k. If $x_k^s \neq 0$, then x_k, y_k, and z_k in (3.8) and (3.9) must be shifted accordingly.

Filter Tuning

All local filters need to be tuned for their particular applications. For well-designed sensors, this essentially boils down to designing the process noise covariance Q_k, i.e., $\sigma_{w,k}$. A systematic approach to filter tuning is described in [16]. The authors of [38] propose a methodology for auto-tuning of filters. In this scope, it is sufficient to use a simplified method based on ANEES[1] defined in Section 3.2.4: 1. Define a characteristic target trajectory. 2. Run *M Monte Carlo* (MC) simulations of the local filter using simulated measurements. 3. Set $\sigma_{w,k}$ such that an ANEES of approximately 1 is obtained along the target trajectory.

[1] ANEES = average normalized estimation error squared.

3.1.2 Track Fusion

In DTT, tracks and covariances are exchanged between agents. The performance of the *track fusion* depends on the fusion rule used to combine tracks. In essence, this fusion rule is an estimator. The following notation is used when describing track fusion:

- The local tracks to be fused are represented by $(y_1, R_1), \ldots, (y_N, R_N)$, where (y_i, R_i) is the local track in Agent i. For instance, assume Agent 1 and Agent 2 compute local tracks, e.g., using EKFs, and that these local tracks are fused in Agent 2. Then it is said that Agent 1 transmits (y_1, R_1) to Agent 2 who fuses (y_1, R_1) with its local track (y_2, R_2).
- The estimate computed in the track fusion is denoted $(\hat{x}, P)$.

Track fusion is a type of data fusion, which is an instantaneous operation, where two or more tracks evaluated at the same time are merged. This means, in contrast to state estimation, that time indices in general can be disregarded when describing track fusion. Nevertheless, for clarity, it is sometimes needed to use time indices for (y_i, R_i). In such a case, the local track is denoted $(y_{i,k|l}, R_{i,k|l})$, or simply $(y_{i,k}, R_{i,k})$, and the fused result is denoted $(\hat{x}_{i,k}, P_{i,k})$.

An important special case in this work is the fusion of $N = 2$ local estimates given as

$$y_1 = x + v_1, \qquad R_1 = \mathrm{cov}(\boldsymbol{v}_1), \tag{3.10a}$$

$$y_2 = H_2 x + v_2, \qquad R_2 = \mathrm{cov}(\boldsymbol{v}_2), \tag{3.10b}$$

with $R_{12} = \mathrm{cov}(\boldsymbol{v}_1, \boldsymbol{v}_2)$. This case is handled separately for all track fusion methods defined in this chapter.

Remark 3.2. In a fusion context, the model in (3.10) is quite general. Firstly, for a particular agent, say Agent 1, it is always possible to define x such that $H_1 = I$. Secondly, any fusion method able to fuse two estimates defined as in (3.10) with $H_1 = I$, can sequentially fuse $N > 2$ estimates (y_i, R_i), where $y_i = H_i x + v_i$ and $R_i = \mathrm{cov}(\boldsymbol{v}_i)$.

Bar-Shalom Campo Fusion

The *Bar-Shalom Campo* (BSC) formula for fusion of two correlated estimates is derived in [15]. It is essentially the BLUE in the special case of $N = 2$ and $H_1 = H_2 = I$. This method is generalized for arbitrary H_2 in Section 3.A. The BSC fuser for the model in (3.10) is given in Algorithm 3.3.

Kalman Fusion

If the model in (3.10) holds and $R_{12} = 0$, then Algorithm 3.3 reduces to

$$\hat{x} = P(R_1^{-1} y_1 + H_2^{\mathsf{T}} R_2^{-1} y_2), \qquad P = (R_1^{-1} + H_2^{\mathsf{T}} R_2^{-1} H_2)^{-1}. \tag{3.12}$$

This is equivalent to what a Kalman filter would produce when fusing (y_1, R_1) and (y_2, R_2). Hence, it is here called the *Kalman fuser* (KF). The method yields

Algorithm 3.3: Bar-Shalom Campo Fuser

Input: Estimates (y_1, R_1) and (y_2, R_2), H_2, and cross-covariance R_{12}

The estimates are fused according to:

$$\hat{x} = K_1 y_1 + K_2 y_2, \qquad P = R_1 - K_2 S K_2^{\mathsf{T}}, \tag{3.11}$$

where $K_1 = I - K_2 H_2$, $K_2 = (R_1 H_2^{\mathsf{T}} - R_{12}) S^{-1}$, and $S = H_2 R_1 H_2^{\mathsf{T}} + R_2 - H_2 R_{12} - R_{21} H_2^{\mathsf{T}}$.

Output: $(\hat{x}, P)$

Algorithm 3.4: Kalman Fuser

Input: Estimates (y_i, R_i) and H_i for $i = 1, 2, \ldots, N$

The estimates are fused according to:

$$\hat{x} = P \sum_{i=1}^{N} H_i^{\mathsf{T}} R_i^{-1} y_i, \qquad P = \left(\sum_{i=1}^{N} H_i^{\mathsf{T}} R_i^{-1} H_i \right)^{-1}. \tag{3.13}$$

Output: $(\hat{x}, P)$

optimal results if $R_{12} = 0$. However, if $R_{12} \neq 0$, then the KF in general leads to non-conservative results, as illustrated in Example 2.5. Nevertheless, KF still remains relevant in many DTT problems, e.g., as a baseline method. If, in addition, the KF yields acceptable results for a certain DTT application, then it might be the preferable option because of its simplicity. An N estimate generalization of the KF is provided in Algorithm 3.4. Essentially, Algorithm 3.4 is the BLUE given in (2.18) in the special case when $R_{ij} = 0$ for all $i \neq j$.

Covariance Intersection

CI is one of the most popular methods for fusing estimates under unknown correlations. CI fuses the two estimates defined in (3.10) according to [90]

$$\hat{x} = P(\omega R_1^{-1} y_1 + (1-\omega) H_2^{\mathsf{T}} R_2^{-1} y_2), \quad P = (\omega R_1^{-1} + (1-\omega) H_2^{\mathsf{T}} R_2^{-1} H_2)^{-1}, \tag{3.14}$$

with $\omega \in [0, 1]$ typically given by solving

$$\underset{\omega}{\text{minimize}} \quad J(P), \tag{3.15}$$

where $J(P)$ is a matrix increasing function and P is defined in (3.14). If (y_1, R_1) and (y_2, R_2) are conservative estimates, cf. Definition 2.4, then $(\hat{x}, P)$ computed according to (3.14) is conservative for all $\omega \in [0, 1]$ irrespective of the actual value of R_{12}. A generalization of CI applicable for fusing N estimates (y_i, R_i) is provided in Algorithm (3.5). An estimate $(\hat{x}, P)$ computed according to Algorithm 3.5 is conservative for all $\omega_1, \ldots, \omega_N \in [0, 1]$ satisfying $\sum_{i=1}^{N} \omega_i = 1$ given that each estimate (y_i, R_i) is conservative.

Algorithm 3.5: Covariance Intersection

Input: Estimates (y_i, R_i) and H_i for $i = 1, 2, \ldots, N$

The estimates are fused according to:

$$\hat{x} = P \sum_{i=1}^{N} \omega_i H_i^{\mathsf{T}} R_i^{-1} y_i, \qquad P = \left(\sum_{i=1}^{N} \omega H_i^{\mathsf{T}} R_i^{-1} H_i \right)^{-1}, \tag{3.16}$$

where $\omega_1, \ldots, \omega_N$ are given by

$$\begin{aligned} &\underset{\omega_1,\ldots,\omega_N}{\text{minimize}} \quad J(P) \\ &\text{subject to} \quad \omega_i \in [0,1] \quad \sum_{i=1}^{N} \omega_i = 1, \end{aligned} \tag{3.17}$$

for a matrix increasing function $J(P)$.

Output: $(\hat{x}, P)$

3.1.3 Communication Management

The *communication management* considered here involves functionality and scheduling related to *when*, *what*, and *with whom* to exchange data.

Communication Schedule

Unless otherwise stated, the communication is assumed to be scheduled according to the *basic communication schedule* provided in Rule 3.6. This schedule can be viewed as a type of time-division multiple access technology[2], where each user connected to the network has a dedicated time slot for communication [80].

Rule 3.6 (Basic Communication Schedule). Assume that there are N agents and that the scenario is sampled at time steps $k = 1, 2, 3, \ldots$. Then Agent i exchange its local tracks if and only if (i.f.f.) $k \in \{i, i+N, i+2N, \ldots\}$.

Communicated Data

In this thesis, the communicated data is reduced due to communication constraints. The following two types of data reduction strategies are considered:

1. The communication is constrained such that Agent i is restricted to transmit (y_i, D_i), where D_i is a diagonal representation of R_i. The technique developed to cope with this communication constraint is referred to as the *diagonal covariance approximation*, and is studied in Section 3.3.

2. A local track is transformed into a lower-dimensional subspace before transmission. For instance, assume $y_2 = x + v_2$ and $R_2 = \mathrm{cov}(\boldsymbol{v}_2)$, where $y_2 \in \mathbb{R}^{n_x}$.

[2] The military Link 16 is one example of a communication network that uses time-division multiple access communication.

Instead of exchanging the full estimate (y_2, R_2), Agent 2 exchanges (y_Ψ, R_Ψ), where $y_\Psi = \Psi y_2$, $R_\Psi = \Psi R_2 \Psi^\mathsf{T}$, and $\Psi \in \mathbb{R}^{m \times n_x}$ with $m < n_x$. This technique is referred to as *dimension-reduction* and is the topic of Chapter 5.

Communication Recipient

By the properties C1–C3 in Chapter 1, all communication must be assumed to be on an agent-to-agent basis and without knowledge of the global topology. Hence, it is assumed that each agent communicates only with its instantaneous nearest neighbors, a set of agents that may vary over time, and without any knowledge about what the nearest neighbors of other agents are.

3.1.4 Sources of Correlations

The article in [53] describes three sources of correlations between estimates: (i) *common process noise*, (ii) *common information*, and (iii) *correlated sensor noise*. Cases (i) and (ii) are the main sources which are relevant to this work and are further described in the next. Case (iii) is studied in, e.g., [87, 184].

Common Process Noise

Correlations between two local tracks of the same target state x_k might be present even if the two tracks have been obtained fully isolated from each other and without any interactions between the agents. What matters is that the agents estimate the same process, i.e., x_k. For instance, assume agents 1 and 2 at time k have obtained

$$y_{1,k|k} = x_k + v_{1,k|k}, \qquad R_{1,k|k} = \operatorname{cov}(\boldsymbol{v}_{1,k|k}), \tag{3.18a}$$

$$y_{2,k|k} = x_k + v_{2,k|k}, \qquad R_{2,k|k} = \operatorname{cov}(\boldsymbol{v}_{2,k|k}), \tag{3.18b}$$

where $R_{12,k|k} = \operatorname{cov}(\boldsymbol{v}_{1,k|k}, \boldsymbol{v}_{2,k|k}) = 0$. Assuming the SSM in (3.1), x_k evolves according to the process

$$x_{k+1} = F_k x_k + w_k, \qquad \boldsymbol{w}_k \sim \mathcal{N}(0, Q_k).$$

Assume $\operatorname{cov}(\boldsymbol{v}_{i,k|k}, \boldsymbol{w}_k) = 0$ as for the SSM in (3.1). Let $y_{i,k+1|k}$ denote the local estimate in Agent i computed using the time update of Algorithm 3.1. Then

$$\begin{aligned} v_{i,k+1|k} &= y_{i,k+1|k} - x_{k+1} = F_k y_{i,k|k} - (F_k x_k + w_k) = F_k \left(y_{i,k|k} - x_k\right) - w_k \\ &= F_k v_{i,k|k} - w_k, \end{aligned}$$

for $i = 1, 2$. Since by assumption $\mathrm{E}(\boldsymbol{v}_{i,k|k}) = \mathrm{E}(\boldsymbol{w}_k) = 0$, also $\mathrm{E}(\boldsymbol{v}_{i,k+1|k}) = 0$, and hence

$$\begin{aligned} R_{12,k+1|k} &= \operatorname{cov}(\boldsymbol{v}_{1,k+1|k}, \boldsymbol{v}_{2,k+1|k}) = \mathrm{E}\left((F_k \boldsymbol{v}_{1,k|k} - w_k)(F_k \boldsymbol{v}_{2,k|k} - w_k)^\mathsf{T}\right) \\ &= F_k \underbrace{\mathrm{E}(\boldsymbol{v}_{1,k|k} \boldsymbol{v}_{2,k|k}^\mathsf{T})}_{=0} F_k^\mathsf{T} - F_k \underbrace{\mathrm{E}(\boldsymbol{v}_{1,k|k} \boldsymbol{w}_k^\mathsf{T})}_{=0} - \underbrace{\mathrm{E}(\boldsymbol{w}_k \boldsymbol{v}_{2,k|k}^\mathsf{T})}_{=0} F_k^\mathsf{T} + \underbrace{\mathrm{E}(\boldsymbol{w}_k \boldsymbol{w}_k^\mathsf{T})}_{=Q_k} \\ &= Q_k. \end{aligned}$$

Despite that $R_{12,k|k} = 0$ and that there was no interactions between Agent 1 and Agent 2, $(y_{1,k+1|k}, P_{1,k+1|k})$ and $(y_{2,k+1|k}, P_{2,k+1|k})$ are now correlated.

Consider the SSM in (3.1) but with a linear measurement model $h(x_k) = H_{i,k}x_k$ for each agent. Recursive formulas for the update of R_{12} are given by [143]

$$R_{12,k+1|k} = F_k R_{12,k|k} F_k^{\mathsf{T}} + Q_k, \tag{3.19a}$$

$$R_{12,k|k} = (I - K_{i,k}H_{i,k})R_{12,k|k-1}(I - K_{i,k}H_{i,k})^{\mathsf{T}}, \tag{3.19b}$$

where $K_{i,k}$ is the Kalman gain of Agent i. The effect of common process noise over time is illustrated using a one-dimensional state x_k in Example 3.7.

Example 3.7: Common Process Noise

The dynamics of R_{12} are studied under the influence of common process noise. Let $N = 2$ agents estimate a scalar state x_k. Assume the SSM in (3.1), where the measurement model of Agent i is $z_{i,k} = x_k + e_{i,k}$ and $\text{cov}(\boldsymbol{e}_{i,k}) = \sigma_e^2$ for $i = 1, 2$. Assume a CPM, where $F_k = 1$ and $Q_k = \sigma_w^2$. Assume that initially

$$R_{0|0} = \begin{bmatrix} R_{1,k|k} & R_{12,k|k} \\ R_{12,k|k} & R_{2,k|k} \end{bmatrix}\Bigg|_{k=0} = \begin{bmatrix} \sigma_e^2 & 0 \\ 0 & \sigma_e^2 \end{bmatrix}.$$

From Algorithm 3.1 and the formulas in (3.19), it follows that R can be computed recursively as

$$R_{k+1|k} = R_{k|k} + \sigma_w^2 \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, \tag{3.20a}$$

$$R_{k|k} = \begin{bmatrix} \frac{\sigma_e^2 R_{1,k|k-1}}{R_{1,k|k-1}+\sigma_e^2} & \frac{\sigma_e^4 R_{12,k|k-1}}{(R_{1,k|k-1}+\sigma_e^2)(R_{2,k|k-1}+\sigma_e^2)} \\ \frac{\sigma_e^4 R_{12,k|k-1}}{(R_{1,k|k-1}+\sigma_e^2)(R_{2,k|k-1}+\sigma_e^2)} & \frac{\sigma_e^2 R_{2,k|k-1}}{R_{2,k|k-1}+\sigma_e^2} \end{bmatrix}. \tag{3.20b}$$

The cross-correlation between $\boldsymbol{y}_{1,k|l}$ and $\boldsymbol{y}_{2,k|l}$ is defined as

$$\rho_{k|l} = \frac{R_{12,k|l}}{\sqrt{R_{1,k|l}}\sqrt{R_{2,k|l}}} \in [-1, 1], \tag{3.21}$$

and is the quantity of interest here since it quantifies how strong the correlations between estimates are. The cross-correlation after the time update $\rho_{k|k-1}$ and the cross-correlation after the measurement update $\rho_{k|k}$ are plotted in Figure 3.3 as functions of k for different values of $q = \sigma_w^2/\sigma_e^2$. The cross-correlation ρ is higher after the time update than after the measurement update. This is intuitive since uncorrelated information is added in the measurement update. A higher q leads to a larger $\rho_{k|k}$. The same is not always true in case of $\rho_{k|k-1}$.

Common Information

Assume that Agent 2 shares an estimate with Agent 1 who fuses the received estimate with its local estimate. After the fusion, there will be information common

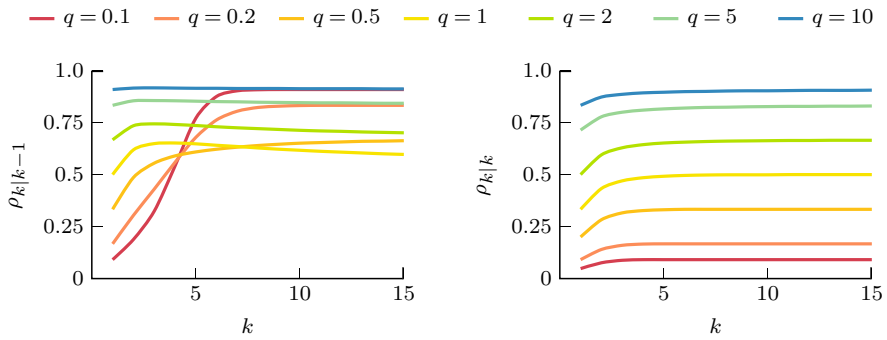


Figure 3.3. Results of Example 3.7. A higher $q = \sigma_w^2/\sigma_e^2$ leads to a higher $\rho_{k|k}$. The same is not always true in the case of $\rho_{k|k-1}$.

to both estimates. This is the problem of *common information*. As an example, consider

$$y_1 = x + v_1, \qquad R_1 = \mathrm{cov}(\boldsymbol{v}_1), \qquad y_2 = x + v_2, \qquad R_2 = \mathrm{cov}(\boldsymbol{v}_2), \tag{3.22}$$

where $R_{12} = \mathrm{cov}(\boldsymbol{v}_1, \boldsymbol{v}_2) \neq 0$. Since the estimates are correlated, Agent 1 can fuse them optimally using Algorithm 3.3. The resulting estimate $(\hat{x}, P)$ is computed as

$$\hat{x} = K_1 y_1 + K_2 y_2, \qquad P = R_1 - K_2 S K_2^{\mathsf{T}}, \tag{3.23}$$

where $K_1 = I - K_2$, $K_2 = (R_1 - R_{12})S^{-1}$, and $S = R_1 + R_2 - R_{12} - R_{21}$. The cross-covariance $R_{12,f}$ of $\hat{x}$ and y_2 is given by

$$\begin{aligned} R_{12,f} &= \mathrm{cov}(\hat{\boldsymbol{x}}, \boldsymbol{y}_2) = \mathrm{E}\left((K_1\boldsymbol{y}_1 + K_2\boldsymbol{y}_2 - x)(\boldsymbol{y}_2 - x)^{\mathsf{T}}\right) \\ &= \mathrm{E}\left(((K_1\boldsymbol{y}_1 + K_2\boldsymbol{y}_2) - (K_1 + K_2)x)\,\boldsymbol{v}_2^{\mathsf{T}}\right) = \mathrm{E}\left((K_1\boldsymbol{v}_1 + K_2\boldsymbol{v}_2)\,\boldsymbol{v}_2^{\mathsf{T}}\right) \\ &= K_1 \underbrace{\mathrm{E}(\boldsymbol{v}_1\boldsymbol{v}_2^{\mathsf{T}})}_{=R_{12}} + K_2 \underbrace{\mathrm{E}(\boldsymbol{v}_2\boldsymbol{v}_2^{\mathsf{T}})}_{=R_2} = K_1 R_{12} + K_2 R_2. \end{aligned} \tag{3.24}$$

In the special case of $R_{12} = 0$, $K_2 = PR_2^{-1}$ and hence $R_{12,f} = P$. Common information is illustrated in Example 3.8.

Example 3.8: Common Information

Assume the same scenario as in Example 3.7, but where Agent 1 and Agent 2 exchange their local estimates according to Rule 3.6. Assume that the agents have full knowledge about the cross-covariance at each point so that Algorithm 3.3 can be used. The recursions for $R_{12,k|k-1}$ and $R_{12,k|k}$ are stated in (3.20). The recursion for the cross-covariance after fusion, $R_{12,k|k,f}$, is given by

$$R_{12,k|k,f} = K_1 R_{12,k|k} + K_2 R_{2,k|k},$$

where $K_1 = 1-K_2$ and $K_2 = \frac{R_{1,k|k}-1}{R_{1,k|k}+R_{2,k|k}-2R_{12,k|k}}$. Let $R_{k|k,f} = \begin{bmatrix} R_{1,k|k,f} & R_{12,k|k,f} \\ R_{12,k|k,f} & R_{2,k|k,f} \end{bmatrix}$ be the joint covariance after fusion. Let $\rho_{k|l}$ be defined as in (3.21). The cross-correlation $\rho_{k|k,f}$ after fusion is computed as

$$\rho_{k|k,f} = \frac{R_{12,k|k,f}}{\sqrt{R_{1,k|k,f}}\sqrt{R_{2,k|k,f}}} \in [-1,1].$$

The cross-correlations $\rho_{k|k-1}$, $\rho_{k|k}$, and $\rho_{k|k,f}$ are plotted in Figure 3.4 as functions of k for different values of $q = \sigma_w^2/\sigma_e^2$. Compared to Example 3.7, $\rho_{k|k-1}$ is now significantly larger for all q, and $\rho_{k|k}$ is slightly larger for all q. As seen in the plot for $\rho_{k|k,f}$, the correlation is generally high after fusion, due to common information. For large q, $\rho_{k|k,f}$ tends to 1. For small q, $\rho_{k|k,f}$ approaches approximately 0.75.

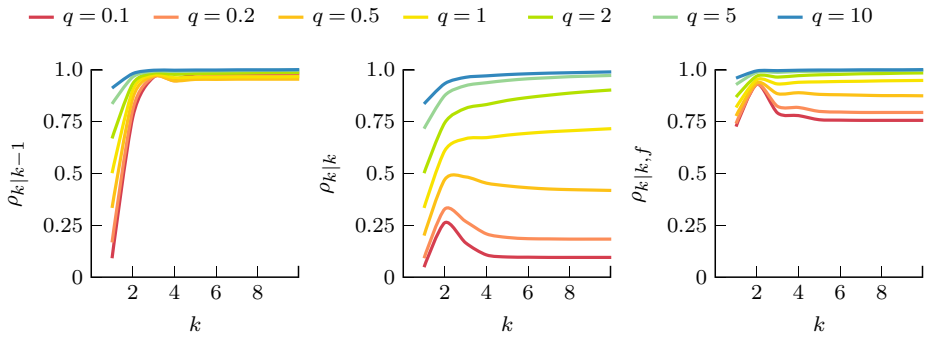


Figure 3.4. Results of Example 3.8. Compared to Example 3.7 $\rho_{k|k-1}$ is now significantly larger for all $q = \sigma_w^2/\sigma_e^2$ and $\rho_{k|k}$ is slightly larger for all q. The cross-correlation $\rho_{k|k,f}$ is large for all q.

3.1.5 Track Fusion Under Correlations

In Example 3.8, the correlations between the estimates were computed explicitly, which allowed for optimally fused estimates. The reason why this cannot be done in general DTT problems lies in properties C1–C3 of Section 1.1. For instance, consider the scenario in Figure 3.5. At time k_a, Agent 1 transmits (y_{1,k_a}, R_{1,k_a}) to Agent 2. At a later time $k_c > k_a$, Agent 1 receives (y_{3,k_c}, R_{3,k_c}) from Agent 3. Assume that Agent 1 knows that $R_{13,k_a} = 0$. However, to fuse (y_{1,k_c}, R_{1,k_c}) and (y_{3,k_c}, R_{3,k_c}) using Algorithm 3.3 requires that R_{13,k_c} is known. To compute R_{13,k_c}, the formulas in (3.19) and (3.24) can be used, but this requires that Agent 1 knows how (y_{3,k_c}, R_{3,k_c}) has been obtained. Since Agent 1 only has access to its local topology, Agent 1 does not know if (y_{3,k_c}, R_{3,k_c}) is the result of fusing, e.g., (y_{2,k_b}, R_{2,k_b}) and (y_{3,k_b}, R_{3,k_b}) at an intermediate time k_b, where $k_a < k_b < k_c$. If such fusion has occurred, then this results in one R_{13,k_c}. If such fusion has not occurred, then a different R'_{13,k_c} will be the true cross-covariance. In theory, it is possible to resolve this by exchanging information related to how (y_{3,k_c}, R_{3,k_c}) has

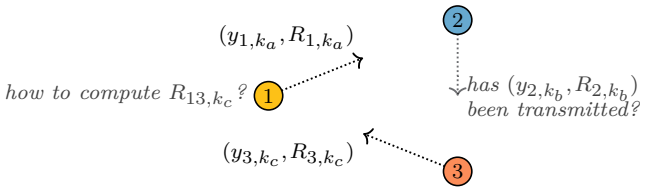


Figure 3.5. Tracking correlations in a DSN. At time k_a, Agent 1 transmits (y_{1,k_a}, R_{1,k_a}) to Agent 2. At time $k_c > k_a$, Agent 1 receives (y_{3,k_c}, R_{3,k_c}) from Agent 2. Since Agent 1 only knows the local network topology, Agent 1 does not know if (y_{2,k_b}, R_{2,k_b}) was transmitted to Agent 3 at an intermediate time k_b.

been obtained, e.g., by exchanging all relevant gains. However, such a solution is not scalable since it would require a huge amount of additional data to be communicated, if even possible given the communication protocol. For this reason, it is important to consider techniques that can be used under unknown correlations.

The simplest approach to track fusion under unknown correlations is to assume that the correlations are zero. Example 2.5 illustrates a scenario where ignoring nonzero correlations leads to non-conservative results. In Example 3.9, the effect of neglecting correlations is illustrated in a target tracking context.

Example 3.9: Neglecting Correlations

Consider a dynamic target moving along the trajectory defined in Figure 3.6. The target follows a straight path, then turns right, followed by an immediate left-hand turn, and finally another straight segment. Two agents track the target and communicate their local tracks according to Rule 3.6. The SSM model in (3.1) is assumed with CAM for the dynamics and a linear measurement equation. KF is compared to using CI.

The estimated trajectories are shown in Figure 3.6 for 50 MC runs. In each MC run, the same noise realizations and track initializations are used for KF and CI. Using KF for track fusion typically results in a clear overshooting of the curves. The overshooting occurs since KF neglects the correlations due to common process noise and common information and hence underestimates the true covariance. As a consequence, the local filters give too little attention to the measurements and too much attention to the predictions. By instead using CI for track fusion, a relatively high estimation accuracy can be maintained during the turning phase.

3.2 Evaluation Measures

In this section, several evaluation measures are stated. Most of them are well-known; one is in particular well-known in DTT; and one is new.

The numerical evaluations in this thesis are based on MC simulations. Denote by $\hat{x}_k^i$ the estimate of x_k in the ith simulation, where P_k^i is the associated covariance.

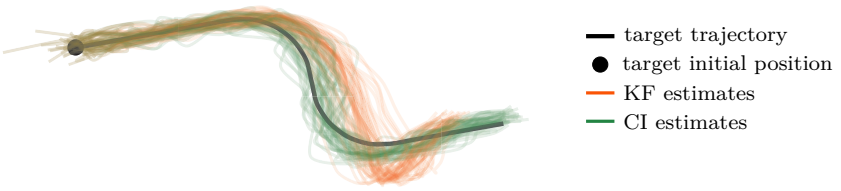


Figure 3.6. The single-target tracking scenario of Example 3.9. Since KF neglects correlations between the exchanged estimates, the resulting estimates overshoot the true target trajectory in the curves. Meanwhile, CI maintains high estimation accuracy even during the turning phase.

3.2.1 Cramér-Rao Lower Bound

Assume that the true dynamics and measurements of the state x_k are according to the SSM in (3.1), i.e.,

$$\begin{aligned} x_{k+1} &= F_k x_k + w_k, & \boldsymbol{w}_k &\sim \mathcal{N}(0, Q_k), \\ z_k &= h(x_k) + e_k, & \boldsymbol{e}_k &\sim \mathcal{N}(0, C_k). \end{aligned}$$

The parametric *Cramér-Rao lower bound* (CRLB, [167]) P^0, of an unbiased estimator of x_k, is computed recursively as [61]

$$P^0_{k+1|k} = F_k P_{k|k} F_k^{\mathsf{T}} + Q_k, \tag{3.25a}$$

$$P^0_{k|k} = \left((P^0_{k|k-1})^{-1} + (H^0_k)^{\mathsf{T}} C_k^{-1} H^0_k \right)^{-1}, \tag{3.25b}$$

where $H^0_k = \left. \frac{\partial h(x')}{\partial x'} \right|_{x'=x_k}$. Often only the position components of P^0 are of interest. This quantity is denoted P^0_{pos} and is given by the upper left block of P^0.

3.2.2 Root Mean Squared Error

The *root mean squared error* (RMSE) evaluated at time k is given by

$$\text{RMSE}_k = \sqrt{\frac{1}{M} \sum_{i=1}^{M} (\tilde{x}^i_k)^{\mathsf{T}} \tilde{x}^i_k} = \sqrt{\text{tr}\left(\hat{P}_k\right)}, \tag{3.26}$$

where $\tilde{x}^i_k = \hat{x}^i_k - x_k$ and

$$\hat{P}_k = \frac{1}{M} \sum_{i=1}^{M} \tilde{x}^i_k (\tilde{x}^i_k)^{\mathsf{T}}. \tag{3.27}$$

Given that the mean $\text{E}(\tilde{\boldsymbol{x}}_k) = 0$ is known, $\hat{P}_k$ equals the sampled covariance of the estimation error $\tilde{\boldsymbol{x}}_k$. If $\tilde{\boldsymbol{x}}_k$ in addition is Gaussian distributed, then $\hat{P}_k$ in (3.27) is the maximum likelihood estimate of $\text{E}(\tilde{\boldsymbol{x}}_k \tilde{\boldsymbol{x}}_k^{\mathsf{T}})$ [88].

3.2.3 Root Mean Trace

The *root mean trace* (RMT) of the computed covariance is given by

$$\mathrm{RMT}_k = \sqrt{\mathsf{tr}\left(\frac{1}{M}\sum_{i=1}^{M} P_k^i\right)} = \sqrt{\frac{1}{M}\sum_{i=1}^{M}\mathsf{tr}\left(P_k^i\right)}. \tag{3.28}$$

3.2.4 Average Normalized Estimation Error Squared

The *average normalized estimation error squared* (ANEES, [108]) evaluated at time k is given by

$$\mathrm{ANEES}_k = \frac{1}{n_x M}\sum_{i=1}^{M}(\tilde{x}_k^i)^{\mathsf{T}}(P_k^i)^{-1}\tilde{x}_k^i. \tag{3.29}$$

If $P_k^i = P_k$ for all i, then

$$\mathrm{ANEES}_k = \frac{1}{n_x M}\sum_{i=1}^{M}\mathsf{tr}\left(P_k^{-1}\tilde{x}_k^i(\tilde{x}_k^i)^{\mathsf{T}}\right) = \frac{1}{n_x}\mathsf{tr}\left(P_k^{-1}\hat{P}_k\right), \tag{3.30}$$

where $\hat{P}_k$ is defined in (3.27). Approximate confidence intervals for ANEES are given by [109]

$$\left[\left(1-\frac{2}{9n_xM}-p\sqrt{\frac{2}{9n_xM}}\right)^3, \left(1-\frac{2}{9n_xM}+p\sqrt{\frac{2}{9n_xM}}\right)^3\right], \tag{3.31}$$

where p is related to the confidence level. For instance, $p = 1.960$, $p = 2.576$, and $p = 3.291$ for 95%, 99%, and 99.9% confidence intervals, respectively. The confidence interval can be used as a tool in filter tuning [16]. Here, it is used to evaluate if an estimator is conservative. If the computed ANEES is larger than the upper bound of the confidence interval, then this estimator is not conservative.

3.2.5 Conservativeness Index

Let $\tilde{P}_k = \mathsf{cov}(\tilde{\boldsymbol{x}}_k)$ be the covariance of $\tilde{\boldsymbol{x}}_k$. Let $P_k = L_k L_k^{\mathsf{T}} \in \mathbb{S}_{++}^{n_x}$ be the computed covariance for the estimate $\hat{x}_k$. Then the condition $P_k \succeq \tilde{P}_k$ is equivalent to $I \succeq L_k^{-1}\tilde{P}_k L_k^{-\mathsf{T}}$. The *conservativeness index*[3] (COIN) is defined as [56]

$$\mathrm{COIN}_k = \lambda_{\max}\left(L_k^{-1}\tilde{P}_k L_k^{-\mathsf{T}}\right). \tag{3.32}$$

Definition 2.4 implies that $(\hat{x}_k, P_k)$ is conservative i.f.f. $\lambda_{\max}(L_k^{-1}\tilde{P}_k L_k^{-\mathsf{T}}) \leq 1$. If $\tilde{P}_k$ in (3.32) is replaced by $\hat{P}_k$ given in (3.27), then this last statement is only approximately true.

[3] This measure was originally proposed in [56].

3.3 Decentralized Target Tracking Under the Diagonal Covariance Approximation

Let $y_i = x + v_i$, such that $y_i \in \mathbb{R}^{n_x}$ and $R_i \in \mathbb{S}^{n_x}_{++}$. Assume that Agent i wants to exchange (y_i, R_i). Then n_x parameters for y_i and $n_x(n_x+1)/2$ parameters for R_i must be transmitted, yielding a total of $n_x(n_x+3)/2$ parameters to be transmitted. One way to reduce the communication load is to only communicate the diagonal elements of R_i, hence reducing the total number of transmitted parameters to $2n_x$.

In this section, a DTT problem is studied, where the communication is constrained such that the full state estimate y_i but only a diagonal representation of R_i are allowed to be exchanged. The problem is referred to as the *diagonal covariance approximation* (DCA) and was first proposed in [54].

3.3.1 The Diagonal Covariance Approximation

Let $d_{i,j}$ be the jth diagonal entry of R_i. Let $s_j \geq 1$ be a real-valued scalar. Then D_i and D_i^s are defined as

$$D_i = \mathsf{diag}(d_{i,1}, \ldots, d_{i,n_x}), \qquad D_i^s = \mathsf{diag}(s_1 d_{i,1}, \ldots, s_{n_x} d_{i,n_x}). \tag{3.33}$$

Provided next is an example of the implication of the DCA. Thereafter, the specific problem studied in this section is defined.

Motivating Example

Let $R_i = \left[\begin{smallmatrix} 4 & 1 \\ 1 & 1 \end{smallmatrix}\right]$, such that $D_i = \mathsf{diag}(4,1)$. Since $R_i = \mathrm{cov}(\boldsymbol{v}_i) = \mathrm{E}(\boldsymbol{v}_i \boldsymbol{v}_i^\mathsf{T})$, (y_i, R_i) is a conservative estimate. However, since

$$D_i - \mathrm{E}(\boldsymbol{v}_i \boldsymbol{v}_i^\mathsf{T}) = D_i - R_i = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \not\succeq 0,$$

the estimate (y_i, D_i) is *not* conservative. Consider now $s_1 = s_2 = 2$ such that $D_i^s = 2D_i$. In this case,

$$D_i^s - \mathrm{E}(\boldsymbol{v}_i \boldsymbol{v}_i^\mathsf{T}) = \begin{bmatrix} 4 & -1 \\ -1 & 1 \end{bmatrix} \succeq 0,$$

and hence (y_i, D_i^s) is a conservative estimate. This example is illustrated in Figure 3.7.

Problem Formulation

The previous example demonstrates that it is possible to maintain conservativeness under the DCA provided that the $D_i - R_i \neq 0$ is handled in some way, e.g., by using an inflated diagonal approximation D_i^s of R_i. Assume without loss of generality (w.l.o.g.) that Agent 2 transmits a local estimate to Agent 1 under the DCA. The goal is for Agent 1 to fuse the received estimate with its local estimate, where the fused estimate is conservative. How conservativeness is preserved under the

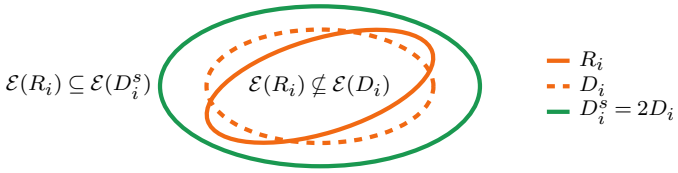


Figure 3.7. Motivating example for the DCA problem. Let (y_i, R_i) be the original local estimate. If R_i is replaced by the diagonal matrix D_i, then the resulting estimate (y_i, D_i) is not conservative. A conservative estimate (y_i, D_i^s) is obtained by replacing R_i by $D_i^s = 2D_i$.

DCA depends on what data is being exchanged. The following two options are considered:

D1 Agent 2 transmits (y_2, D_2^s) to Agent 1, where $D_2^s \succeq R_2$. In this case, Agent 2 has already preserved conservativeness, and hence, Agent 1 can use the received estimate directly without any additional action.

D2 Agent 2 transmits (y_2, D_2) to Agent 1. In this case, Agent 1 must explicitly handle that $D_2 \not\succeq R_2$ to ensure conservativeness after track fusion.

If $\lambda_{\max}(R_2) < \infty$, then it is always possible to scale D_2 using finite valued scaling factors $s_1, \ldots, s_{n_x}$ such that $D_2^s \succeq R_2$. However, since information is the inverse of covariance, inflating D_2 implies information is lost. Hence, it is desirable to scale D_2 sufficiently to ensure that $D_2^s \succeq R_2$, but no more than that.

3.3.2 Methods for Preserving Conservativeness

Four methods are proposed for preserving conservativeness[4]. The methods that use option D1 are:

- *Eigenvalue based scaling* (DCA-EIG). Agent 2 exchanges (y_2, D_2^s), where $D_2^s = s^\star D_2$ and $s^\star$ is given in Theorem 3.10.
- *Optimization based scaling* (DCA-OPT). Agent 2 exchanges (y_2, D_2^s), where D_2^s is according to (3.33) and $s_1, \ldots, s_{n_x}$ are the solutions to (3.36).
- *Diagonal-dominance based scaling* (DCA-DOM). Agent 2 exchanges (y_2, D_2^s), where D_2^s is computed according to (3.38).

The method that uses option D1 is:

- *Hyperrectangle enclosing* (DCA-HYP). Agent 1 receives (y_2, D_2) and computes a conservative estimate using (3.42).

[4]In the original paper [54] there are five methods proposed. However, here two of these are unified into a common method called DCA-HYP.

Eigenvalue Based Scaling

Assume that the scaling factors are restricted as $s_1 = \cdots = s_{n_x} = s$ such that $D_2^s = sD_2$ is obtained by uniform scaling of D_2. Under this restriction, the optimal scaling factor $s^\star$ is given by

$$\begin{aligned} \underset{s}{\text{minimize}} \quad & s \\ \text{subject to} \quad & D_2^s = sD_2 \succeq R_2. \end{aligned} \tag{3.34}$$

The solution to the problem in (3.34) is provided by Theorem 3.10. Essentially, the theorem states that $s^\star$ is given by the largest eigenvalue of the correlation matrix $C_2 = D_2^{-\frac{1}{2}} R_2 D_2^{-\frac{1}{2}}$.

Theorem 3.10. *The solution to the problem in* (3.34) *is given by*

$$s^\star = \lambda_{\max}\left(D_2^{-\frac{1}{2}} R_2 D_2^{-\frac{1}{2}} \right). \tag{3.35}$$

Proof: By assumption, $D_2, R_2 \in \mathbb{S}_{++}^{n_x}$. Let $C_2 = D_2^{-\frac{1}{2}} R_2 D_2^{-\frac{1}{2}} = U\Sigma U^\mathsf{T}$, where $U\Sigma U^\mathsf{T}$ is the eigendecomposition of C_2, cf. (2.11). Since $U^\mathsf{T} I U = U^\mathsf{T} U = I$, the constraint in (3.34) can hence equivalently be expressed as $sI \succeq \Sigma$. Moreover, since Σ is diagonal, with eigenvalues $\lambda_i(C_2)$ on its diagonal,

$$sI \succeq \Sigma \iff s \geq \lambda_i(C_2), \forall i,$$

which in particular holds for $s = \lambda_{\max}(C_2)$. Meanwhile, the constraint is violated for all $s < \lambda_{\max}(C_2)$. Hence, the optimal solution to (3.34) is $s^\star = \lambda_{\max}(C_2)$. □

Optimization Based Scaling

A less restrictive solution compared to DCA-EIG is obtained by allowing $s_1, \ldots, s_{n_x}$ to be different. In this case, an optimal D_2^s is given by

$$\begin{aligned} \underset{s_1,\ldots,s_{n_x} \geq 1}{\text{minimize}} \quad & J(D_2^s) \\ \text{subject to} \quad & D_2^s = \mathsf{diag}(s_1 d_{2,1}, \ldots, s_{n_x} d_{2,n_x}) \succeq R_2, \end{aligned} \tag{3.36}$$

where J is matrix increasing and $d_{2,i} = [D_2]_{ii}$. If $J(\,\cdot\,) = \mathsf{tr}(\,\cdot\,)$, then (3.36) is a standard *semidefinite program* (SDP, [29]).

Diagonal-Dominance Based Scaling

A matrix $A \in \mathbb{R}^{n \times n}$ is diagonally dominant if

$$|[A]_{ii}| \geq \sum_{j \neq i} |[A]_{ij}|, \forall i. \tag{3.37}$$

If a diagonally dominant matrix A in addition is symmetric with non-negative diagonal entries, then $A \in \mathbb{S}_+^n$ [83]. Hence, $D_2^s \succeq R_2$ is automatically satisfied for

$$D_2^s = \mathsf{diag}(s_1 d_{2,1}, \ldots, s_{n_x} d_{2,n_x}), \qquad s_i d_{2,i} = \sum_{j=1}^{n_x} |[R_2]_{ij}|. \tag{3.38}$$

Hyperrectangle Enclosing

If Agent 1 receives (y_2, D_2) from Agent 2, then Agent 1 needs to inflate D_2 in order to be able to reach a conservatively fused result. However, from the point-of-view of Agent 1, R_2 can be any element in the set

$$\mathcal{A} = \left\{ B \in \mathbb{S}_+^{n_x} \,\middle|\, [B]_{ii} = d_{2,i} \right\}, \tag{3.39}$$

where $d_{2,i} = [D_2]_{ii}$. The set $\mathcal{A}$ is interpreted as the union $\mathcal{R} = \bigcup_{B\in\mathcal{A}} \mathcal{E}(B)$. In essence, $\mathcal{R}$ is an axis-aligned *hyperrectangle*, where the length of the ith side is $2\sqrt{[R_2]_{ii}}$. The hyperrectangle is illustrated in Figure 3.8 for $n_x = 2$ and $R_2 = \left[\begin{smallmatrix} 4 & 1 \\ 1 & 1 \end{smallmatrix}\right]$. Before proceeding, a notion of tightness is provided for the current context.

Definition 3.11 (Tight Enclosing of a Hyperrectangle). Let $\mathcal{R}$ be a hyperrectangle centered at the origin. An ellipsoid $\mathcal{E}(A)$ *tightly encloses* $\mathcal{R}$, if all the corners of $\mathcal{R}$ lie on the boundary of $\mathcal{E}(A)$.

Agent 1 derives (y_2, D_2^ω) from (y_2, D_2), where D_2^ω needs to satisfy $D_2^\omega \succeq R_2$ in order to preserve conservativeness. However, since Agent 1 does not know which element $B \in \mathcal{A}$ which is equal to R_2, Agent 1 must ensure $D_2^\omega \succeq B, \forall B \in \mathcal{A}$ to be sure that $D_2^\omega \succeq R_2$. In particular, Agent 1 wants to solve

$$\begin{aligned} &\underset{s_1,\ldots,s_{n_x}\geq 1}{\text{minimize}} && J(D_2^s) \\ &\text{subject to} && D_2^s = \operatorname{diag}(s_1 d_{2,1}, \ldots, s_{n_x} d_{2,n_x}) \succeq B, \forall B \in \mathcal{A}. \end{aligned} \tag{3.40}$$

The solution to (3.40) is given by the following theorem.

Theorem 3.12. *Let $\mathcal{A}$ be defined as in* (3.39)*. Then, for D_2^ω defined as in* (3.41)*, it holds that:*

1. *$D_2^\omega \succeq B, \forall B \in \mathcal{A}$, and*
2. *the ellipsoid $\mathcal{E}(D_2^\omega)$ tightly encloses the hyperrectangle $\mathcal{R} = \bigcup_{B\in\mathcal{A}} \mathcal{E}(B)$.*

Proof: Since $D_2 \in \mathbb{S}_{++}^{n_x}$, it is equivalent to consider $I^\omega = D_2^{-\frac{1}{2}} D_2^\omega D_2^{-\frac{1}{2}}$ and $\mathcal{B} = \{D_2^{-\frac{1}{2}} B D_2^{-\frac{1}{2}} \,|\, B \in \mathcal{A}\}$. By definition, $C \in \mathcal{B}$ is a correlation matrix, which means that $\mathcal{E}(C)$ is confined to a hypercube $\mathcal{S}$ centered at the origin, where each side of $\mathcal{S}$ is of length 2. The corners of $\mathcal{S}$ are given by all combinations of $s = \begin{bmatrix} b_1 & \ldots & b_{n_x} \end{bmatrix}^\mathsf{T}$, where $b_i \equiv \pm 1$. A certain point s lies on the boundary of $\mathcal{E}(I^\omega)$ if $s^\mathsf{T}(I^\omega)^{-1}s = 1$ [29]. It is hence sufficient to show that $s^\mathsf{T}(I^\omega)^{-1}s = 1$ simultaneously holds for all s. If $\sum_{i=1}^{n_x} \omega_i = 1$, then since $b_i^2 \equiv 1$

$$s^\mathsf{T}(I^\omega)^{-1}s = \begin{bmatrix} b_1\omega_1 & \ldots & b_{n_x}\omega_{n_x} \end{bmatrix} \begin{bmatrix} b_1 \\ \vdots \\ b_{n_x} \end{bmatrix} = \sum_{i=1} b_i^2 \omega_i = \sum_{i=1} \omega_i = 1.$$

Hence, $\mathcal{E}(I^\omega)$ tightly encloses $\mathcal{S}$, which implies that $\mathcal{E}(D_2^\omega)$ tightly encloses $\mathcal{R}$. From (2.2) it follows that $D_2^\omega \succeq B, \forall B \in \mathcal{A}$. □

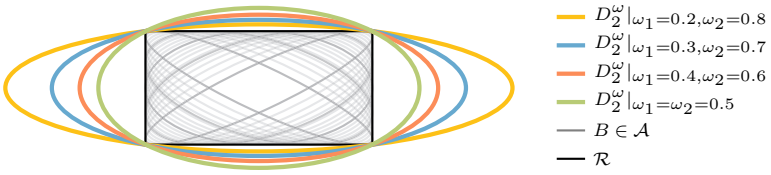


Figure 3.8. Illustration of $\mathcal{A}$. The set $\mathcal{A}$ can be geometrically interpreted as a rectangle $\mathcal{R} = \bigcup_{B\in\mathcal{A}} \mathcal{E}(B)$ if $n_x = 2$. The ellipse $\mathcal{E}(D_2^\omega)$, with D_2^ω given according to (3.41), tightly encloses $\mathcal{R}$ as the boundary of $\mathcal{E}(D_2^\omega)$ intersects all corners of $\mathcal{R}$.

Theorem 3.12 provides a specific parametrization D_2^ω of D_2^s, i.e.,

$$D_2^\omega = \operatorname{diag}\left(\frac{d_{2,1}}{\omega_1}, \ldots, \frac{d_{2,n_x}}{\omega_{n_x}}\right), \qquad \omega_i \in (0,1), \qquad \sum_{i=1} \omega_i = 1, \tag{3.41}$$

where $d_{2,i} = [D_2]_{ii}$ and $\omega_i = 1/s_i$. Crucial here are the following two properties: (i) $D_2^\omega \succeq B, \forall B \in \mathcal{A}$; and (ii) $\mathcal{E}(D_2^\omega)$ bounds $\mathcal{R}$ tightly, in the sense of Definition 3.11. Several D_2^ω, for different values of $\omega_1 = 1-\omega_2$, are illustrated in Figure 3.8, where also the tightness concept is demonstrated.

Remark 3.13. A special case of D_2^ω defined in (3.41) is when $\omega_i = 1/n_x, \forall i$. In this case, $D_2^\omega = n_x D_2$. This choice of D_2^ω leads to one of the five proposed methods for preserving conservativeness in [54]. However, this special case is not considered further in this work.

As noted in [54], D_2^ω can be integrated directly into the CI algorithm. The estimates (y_1, R_1) and (y_2, D_2) are fused conservatively using CI as

$$\hat{x} = P\left(\omega_1 R_1^{-1} y_1 + \sum_{i=1}^{n_x} \omega_{2,i} H_{2,i}^\mathsf{T} [D_2^{-1}]_{ii} [y_2]_i\right), \tag{3.42a}$$

$$P = \left(\omega_1 R_1^{-1} + \sum_{i=1}^{n_x} \omega_{2,i} H_{2,i}^\mathsf{T} [D_2^{-1}]_{ii} H_{2,i}\right)^{-1}, \tag{3.42b}$$

where $\omega_1, \omega_{2,i} \in [0,1]$, $\omega_1 + \sum_{i=1}^{n_x} \omega_{2,i} = 1$, and $H_{2,i} = \begin{bmatrix}\delta_{1i} & \ldots & \delta_{n_x i}\end{bmatrix}$.

Remark 3.14. The same technique can be applied to KF, which, for instance, is relevant if y_1 and y_2 are uncorrelated and Agent 1 receives (y_2, D_2) from Agent 2. In that case

$$\hat{x} = P\left(R_1^{-1} y_1 + \sum_{i=1}^{n_x} \omega_{2,i} H_{2,i}^\mathsf{T} [D_2^{-1}]_{ii} [y_2]_i\right), \quad P = \left(R_1^{-1} + \sum_{i=1}^{n_x} \omega_{2,i} H_{2,i}^\mathsf{T} [D_2^{-1}]_{ii} H_{2,i}\right)^{-1},$$

where $\omega_{2,i} \in [0,1]$ and $\sum_{i=1}^{n_x} \omega_{2,i} = 1$.

3.3.3 Communication Reduction

As pointed out in the beginning of Section 3.3.1, the DCA reduces the number of transmitted parameters from $n_{\text{full}} = n_x(n_x+1)/3$ to $n_{\text{dca}} = 2n_x$. Figure 3.9 illustrates the communication reduction when using the DCA methodology. If,

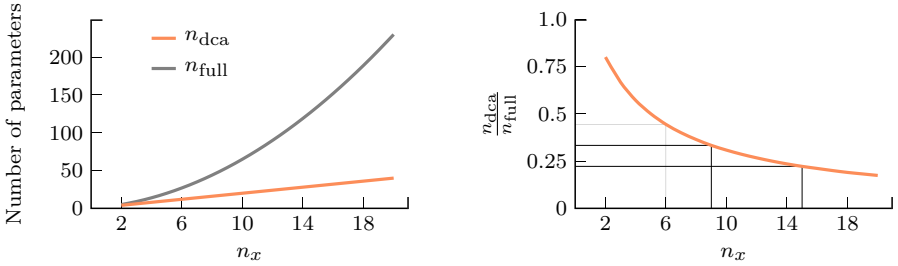


Figure 3.9. Communication reduction when using the DCA.

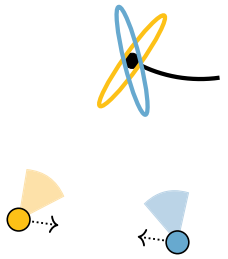


Figure 3.10. Scenario used in the numerical evaluation. The agents are placed at fixed locations $(-2000, 1000)$ m and $(5000, 0)$ m. The target is initially located at $(3000, 8000)$ m, represented by the black circle, and moves along the black trajectory. The ellipses represent the measurement error covariance of each sensor.

e.g., n_x is 6, 9, or 15, then the communication is reduced by 56%, 67%, or 78%, respectively.

3.3.4 Numerical Evaluation

The proposed methods are now evaluated in a DTT scenario.

Simulation Specifications

The scenario is illustrated in Figure 3.10. Two agents track a common target in $d = 2$ spatial dimensions. The CAM in (3.6) is assumed for the dynamics, and the sensors are defined according to the nonlinear model in (3.8). The agents communicate their local tracks according to Rule 3.6. Relevant simulation parameters are summarized in Table 3.2.

Each agent uses EKF for state estimation and CI for track fusion. This means that DCA-EIG, DCA-OPT, and DCA-DOM are combined with (3.14), and that DCA-HYP is combined with (3.42). The methods are compared to:

- LKF: A local EKF for which no tracks are shared.
- NKF: A naïve DTT system that uses KF for track fusion.

Table 3.2
PARAMETERS USED IN THE SIMULATIONS

Parameter	Comment
$d = 2$	spatial dimensionality
$n_x = 6$	state dimensionality
$T_s = 1$	sampling time [s]
$\sigma_w = 2$	standard deviation of process noise [$\mathrm{ms}^{-\frac{5}{2}}$]
$\sigma_r = 1\,000$	standard deviation of radial measurement noise [m]
$\sigma_\theta = 1$	standard deviation of azimuth measurement noise [°]
$(-2000, 1000)$	Agent 1 location [m]
$(5000, 0)$	Agent 2 location [m]
$(3000, 8000)$	target initial position [m]
$n_k = 18$	number of time steps
$M = 10\,000$	number of MC runs

- CRLB: In the RMSE plots, CRLB is represented by $\sqrt{\mathrm{tr}(P^0_{\mathrm{pos}})}$, where P^0_{pos} is computed as in Section 3.2.1.
- CI-FULL: Communication of full estimates with CI used for track fusion.

The optimization problems related to DCA-OPT and DCA-HYP are solved using `YALMIP` [114] with the `MOSEK` solver [123]. In all cases $J(\,\cdot\,) = \mathrm{tr}(\,\cdot\,)$.

Evaluation Measures

The RMSE, RMT, and CRLB are used to evaluate estimator performance. Conservativeness is evaluated using the ANEES. RMSE and RMT are computed for the position components of the error and covariance. ANEES is computed for the full state.

Results

The results for both agents are plotted in Figure 3.11. The RMT and RMSE are normalized by σ_r. ANEES, RMT, and RMSE are computed at time instants where track fusion is performed.

NKF initially performs well w.r.t. RMT and RMSE, but soon the RMSE start to diverge due to ignoring the cross-correlations and in particular double counting of information. DCA-EIG, DCA-OPT, DCA-DOM, and DCA-HYP perform similarly. However, for Agent 2 during the first time steps, DCA-HYP performs significantly better than DCA-EIG, DCA-OPT, and DCA-DOM. An explanation of this is that while DCA-EIG, DCA-OPT, and DCA-DOM scale D_2 *before* transmission, DCA-HYP essentially postpones the scaling of D_2 until *after* reception. Hence, e.g., Agent 1 is able to tune the scaling of D_2 in such a way that fusion of (y_1, R_1) with (y_2, D_2) is optimized. This is not possible for option D1 since (y_1, R_1) is unavailable for Agent 2.

All methods except NKF are conservative w.r.t. ANEES.

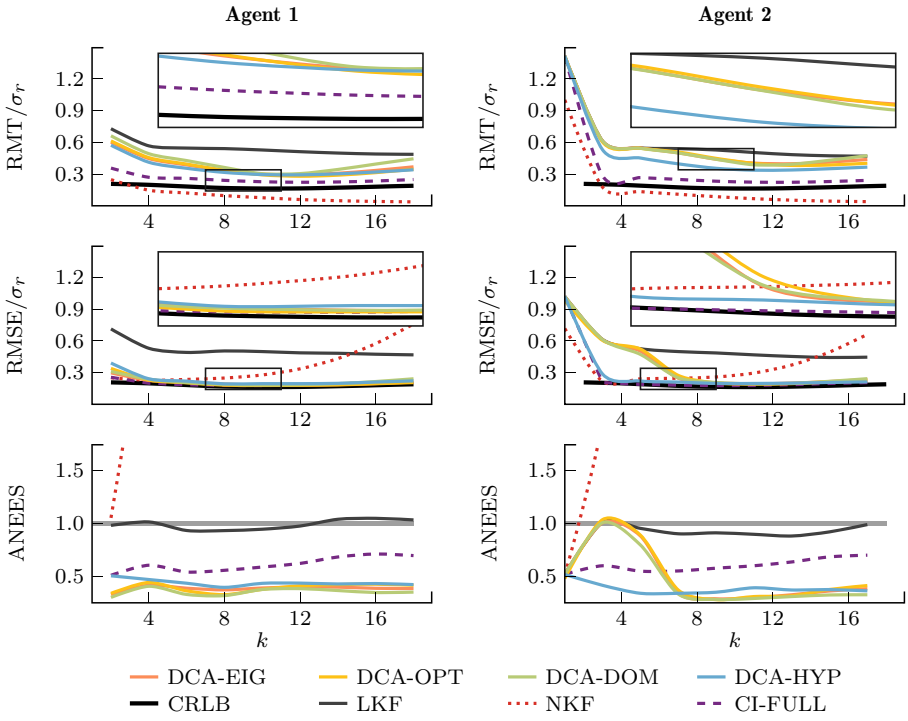


Figure 3.11. Results from the DCA evaluation. RMSE and RMT are computed for the position components of the error and covariance. ANEES is computed for the full state. The gray areas represent ANEES 99.9% confidence intervals.

3.4 Summary

A decentralized single target tracking problem was studied in this chapter as an introduction to DTT under communication constraints. The components, i.e., state estimation, track fusion, and communication management, of the considered decentralized single-target tracking systems have been defined. At the end, the DCA methodology was proposed as a framework for DTT with reduced data communication.

Two important aspects related to DTT are addressed in this chapter, and deserve to be further highlighted:

- *Optimal conservative track fusion.* Example 3.9 demonstrates the importance of ensuring conservativeness in a DTT problem to avoid diverging estimates. At the same time, it is meaningful to develop estimators that are not too conservative since this leads to unnecessarily large covariances, which, in the worst case, make the estimates useless. For instance, by examining the ANEES curves in Figure 3.11, it seems like CI is too conservative for that application. It is therefore crucial to further develop and formalize the conservative estimation problem to promote optimality under conservative-

ness. This is the topic of Chapter 4, but some of the main ideas are already introduced in this chapter. That is, the tight enclosing of a rectangle, cf. Definition 3.11, for which the DCA-HYP method provided a solution.

- *Efficient communication management.* Using the DCA framework, this chapter illustrates that it is possible to reduce communication while maintaining high estimation quality. In particular, the DCA-HYP method demonstrates the advantage of taking into account (y_1, R_1) when communicating a data reduced version of (y_2, R_2), e.g., (y_2, D_2) or (y_2, D_2^s). In the case of DCA-HYP, D_2 is not scaled before fusion with (y_1, R_1), which allows for slightly improved performance. These ideas are further pursued in Chapter 5.

Appendix

3.A A Mean Squared Error Optimal Fusion Method

Assume that

$$y_1 = x + v_1, \qquad R_1 = \mathrm{cov}(\boldsymbol{v}_1), \qquad y_2 = H_2 x + v_2, \qquad R_2 = \mathrm{cov}(\boldsymbol{v}_2),$$

with $R_{12} = \mathrm{cov}(\boldsymbol{v}_1, \boldsymbol{v}_2)$. Let $y_\Psi = \Psi y_2$ and $R_\Psi = \Psi R_2 \Psi^\mathsf{T}$, where $\Psi \in \mathbb{R}^{m \times n_2}$, $m \leq n_2$ and $\mathsf{rank}(\Psi) = m$, such that $R_{1\Psi} = \mathrm{cov}(\boldsymbol{v}_1, \Psi \boldsymbol{v}_2) = R_{12}\Psi^\mathsf{T}$. The goal is to compute an estimate $\hat{x} = K\,\mathrm{col}(y_1, y_\Psi)$, with $KH = I$ and $H = \mathrm{col}(I, \Psi H_2)$, that minimizes trace of

$$P = K \begin{bmatrix} R_1 & R_{1\Psi} \\ R_{\Psi 1} & R_\Psi \end{bmatrix} K^\mathsf{T}.$$

Let $K = \begin{bmatrix} K_1 & K_\Psi \end{bmatrix}$ such that $KH = I \implies K_1 = I - K_\Psi \Psi H_2$. Let $R \succeq 0$ and $S = H_2 R_1 H_2^\mathsf{T} + R_2 - H_2 R_{12} - R_{21} H_2^\mathsf{T} \succeq 0$, but assume Ψ is such that $\Psi S \Psi^\mathsf{T} \succ 0$. Since $K = \begin{bmatrix} I - K_\Psi \Psi H_2 & K_\Psi \end{bmatrix}$

$$\begin{aligned}
P &= R_1 - R_1 H_2^\mathsf{T} \Psi^\mathsf{T} K_\Psi^\mathsf{T} - K_\Psi \Psi H_2 R_1 + K_\Psi \Psi H_2 R_1 H_2^\mathsf{T} \Psi^\mathsf{T} K_\Psi^\mathsf{T} + K_\Psi R_{\Psi 1} \\
&\quad - K_\Psi R_{\Psi 1} H_2^\mathsf{T} \Psi^\mathsf{T} K_\Psi^\mathsf{T} + R_{1\Psi} K_\Psi^\mathsf{T} - K_\Psi \Psi H_2 R_{1\Psi} K_\Psi^\mathsf{T} + K_\Psi R_\Psi K_\Psi^\mathsf{T} \\
&= R_1 - K_\Psi (\Psi H_2 R_1 - R_{\Psi 1}) - (R_1 H_2^\mathsf{T} \Psi^\mathsf{T} - R_{1\Psi}) K_\Psi^\mathsf{T} \\
&\quad + K_\Psi (\Psi H_2 R_1 H_2^\mathsf{T} \Psi^\mathsf{T} + R_\Psi - \Psi H_2 R_{1\Psi}) K_\Psi^\mathsf{T} - R_{\Psi 1} H_2^\mathsf{T} \Psi^\mathsf{T} \\
&= R_1 - K_\Psi A - A^\mathsf{T} K_\Psi^\mathsf{T} + K_\Psi B K_\Psi^\mathsf{T},
\end{aligned}$$

where

$$\begin{aligned}
A &= \Psi H_2 R_1 - R_{\Psi 1} = \Psi \left(H_2 R_1 - R_{21} \right), \\
B &= \Psi H_2 R_1 H_2^\mathsf{T} \Psi^\mathsf{T} + R_\Psi - \Psi H_2 R_{1\Psi} - R_{\Psi 1} H_2 \Psi^\mathsf{T} \\
&= \Psi \left(H_2 R_1 H_2^\mathsf{T} + R_2 - H_2 R_{12} - R_{21} H_2^\mathsf{T} \right) \Psi^\mathsf{T} = \Psi S \Psi^\mathsf{T}.
\end{aligned}$$

Completing the square yields

$$P = R_1 - A^\mathsf{T} B^{-1} A + (K_\Psi - A^\mathsf{T} B^{-1}) B (K_\Psi^\mathsf{T} - B^{-1} A),$$

which is minimized when $K_\Psi = A^\mathsf{T} B^{-1}$. An MSE optimal estimate is given by

$$\hat{x} = K_1 y_1 + K_\Psi y_\Psi, \qquad P = R_1 - K_\Psi \Psi S \Psi^\mathsf{T} K_\Psi^\mathsf{T},$$
$$K_1 = I - K_\Psi \Psi H_2, \qquad K_\Psi = (R_1 H_2^\mathsf{T} - R_{12}) \Psi^\mathsf{T} (\Psi S \Psi^\mathsf{T})^{-1},$$

where $S = H_2 R_1 H_2^\mathsf{T} + R_2 - H_2 R_{12} - R_{21} H_2^\mathsf{T}$.

In the special case of $\Psi = I$, the estimate is given by

$$\hat{x} = K_1 y_1 + K_2 y_2, \qquad P = R_1 - K_2 S K_2^\mathsf{T},$$

where $K_1 = I - K_2 H_2$ and $K_2 = (R_1 H_2^\mathsf{T} - R_{12}) S^{-1}$.

4

Conservative Track Fusion Under Partially Known Correlations

The previous chapter demonstrates that in DTT correlations are often intractable, and therefore some blocks of $R = \mathsf{cov}(\boldsymbol{y})$ are unknown. In practice, it is typically the cross-covariances that are unknown. The *conservative linear unbiased estimator* (CLUE) is introduced as a framework applicable for estimation under *partially known* R. Hence, the CLUE framework is highly relevant for track fusion problems. In addition, an optimal CLUE is defined. Using *robust optimization* (RO), the CLUE framework can be applied to general conservative linear estimation problems. Moreover, it is shown how several existing fusion methods are special cases of the optimal CLUE under different assumptions. A numerical evaluation is provided at the end to validate the theory and methods developed in this chapter.

This chapter is an edited version of [56] © 2022 IEEE.

4.1 Conservative Linear Unbiased Estimation

Let (y, R) be defined as in (2.12). Another, more general, way of saying that parts of R are unknown, or equivalently, that R is only partially known, is that $R \in \mathcal{A}$, where $\mathcal{A} \subseteq \mathbb{S}_+^{n_y}$ is a set of PSD matrices. For a linear estimator $\hat{\boldsymbol{x}} = K\boldsymbol{y}$ with covariance P, this representation implies that

$$P \succeq \mathrm{E}(\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}^{\mathsf{T}}) \iff P \succeq KSK^{\mathsf{T}}, \forall S \in \mathcal{A}. \tag{4.1}$$

If $\hat{\boldsymbol{x}}$ is also unbiased, then the estimator is a CLUE as defined in Definition 4.1.

Definition 4.1 (Conservative Linear Unbiased Estimator). Assume that $\boldsymbol{y} = Hx + \boldsymbol{v}$, where $R = \mathsf{cov}(\boldsymbol{v})$. An estimator $(\hat{\boldsymbol{x}}, P)$ is a *conservative linear unbiased estimator* (CLUE) if

$$\hat{\boldsymbol{x}} = K\boldsymbol{y}, \qquad KH = I, \qquad P \succeq KSK^{\mathsf{T}}, \forall S \in \mathcal{A}. \tag{4.2}$$

Remark 4.2. The set $\mathcal{A}$ contains those S that are permitted given the problem formulation. Such matrices are sometimes referred to as *admissible matrices*, see, e.g., [130, 171]. Hence, $\mathcal{A}$ contains all admissible matrices.

It is assumed that the elements of $\mathcal{A}$ have only finite eigenvalues, which means that R has only finite eigenvalues. A CLUE is characterized by (K, P). The concept of modeling a partially known R as $R \in \mathcal{A}$ is demonstrated in Example 4.3.

Example 4.3: Conservative and Non-Conservative Estimators

Consider $x \in \mathbb{R}^2$ and $\mathcal{A} = \{R^a, R^b, R^c, R^d, R^e\} \subset \mathbb{S}^4_{++}$. Let (K, P) and (K^n, P^n) be two linear unbiased estimators, where P and P^n are defined by the ellipses in Figure 4.1. Since $\mathcal{E}(P) \supseteq \mathcal{E}(KSK^\mathsf{T}), \forall S \in \mathcal{A}$, and hence $P \succeq KSK^\mathsf{T}, \forall S \in \mathcal{A}$, it follows that (K, P) is a CLUE. Meanwhile, since $P^n \succeq K^n S (K^n)^\mathsf{T}$ does not hold for all $S \in \mathcal{A}$, it is concluded that (K^n, P^n) is not a CLUE.

Figure 4.1. Illustration of a conservative estimator (K, P) and a non-conservative estimator (K^n, P^n) given a fixed set $\mathcal{A}$.

4.1.1 Best Conservative Linear Unbiased Estimation

If $\mathcal{A} = \{R\}$, then the BLUE is an optimal CLUE for any matrix increasing loss function J. If $\mathcal{A}$ is not a singleton, that is no longer true. The *best CLUE* is proposed in Definition 4.4 as an optimal CLUE[1], and is a generalization of the BLUE given in Definition 2.2 to arbitrary $\mathcal{A}$.

Definition 4.4 (Best Conservative Linear Unbiased Estimator). Assume that $\boldsymbol{y} = Hx + \boldsymbol{v}$, where $\mathrm{cov}(\boldsymbol{v}) = R \in \mathcal{A} \subset \mathbb{S}^{n_y}_{++}$. An estimator $\hat{\boldsymbol{x}}^\star = K^\star \boldsymbol{y}$ with covariance $P^\star$ is called a *best conservative linear unbiased estimator* (best CLUE), if $(K^\star, P^\star)$ is the solution to

$$\begin{aligned} \underset{K,P}{\text{minimize}} \quad & J(P) \\ \text{subject to} \quad & KH = I \\ & P \succeq KSK^\mathsf{T}, \forall S \in \mathcal{A}, \end{aligned} \tag{4.3}$$

for a given matrix increasing function J.

A *feasible point* to the problem in (4.3) is a pair (K, P) for which the constraints hold [29], i.e., $KH = I$ and $P \succeq KSK^\mathsf{T}, \forall S \in \mathcal{A}$. In this regard, a CLUE is a

[1] Related formulations of optimal conservative estimators have been proposed in [36, 37, 150].

feasible point. The set of all feasible points is called the *feasible set* which equals the set of all CLUEs.

The BLUE covariance is a minimum element, cf. (2.5), of the feasible set to the problem in Definition 2.2. This strong property is unfortunately not inherited to the best CLUE case. However, Proposition 4.5 states the slightly weaker property that, if J is matrix increasing, then $P^\star$ of the solution to the problem in Definition 4.4 is a minimal element of the set of all CLUEs.

Proposition 4.5. *Let $\mathcal{V} \subseteq \mathbb{S}_+^n$ and J be matrix increasing. Then the solution to*

$$\underset{B \in \mathcal{V}}{\text{minimize}} \quad J(B), \tag{4.4}$$

is a minimal element of $\mathcal{V}$.

Proof: The proposition is shown by contradiction. Assume that A solves (4.4). If A is not a minimal element of $\mathcal{V}$, then by definition there exists another element $B \in \mathcal{V}$ such that $B \preceq A \neq B$. From (2.4) and (2.6) it follows that $J(B) < J(A)$, which is a contradiction since then A cannot be a solution to (4.4). Hence, A is a minimal element of $\mathcal{V}$. □

It is known that minimal elements $A \in \mathcal{V}$ are given by minimizing $\mathsf{tr}(WA)$ with $W \in \mathbb{S}_{++}^n$ [29]. Therefore, a natural loss function for the best CLUE problem would be $\mathsf{tr}(WA)$. The reason that a matrix increasing function J is used is that it also includes, e.g., the determinant, which is a commonly used loss function in the literature on conservative estimation.

Remark 4.6. In the literature, J is most often given by the trace or the determinant. Minimizing the trace is related to minimizing the MSE. Minimizing the determinant is related to minimizing the entropy [135].

4.1.2 Proposed Framework

The CLUE is proposed as a framework for conservative estimation. The backbone of this framework is provided in Definition 4.4. Properties of the proposed framework are derived in Section 4.2. As shown in Section 4.3, standard optimization software can be applied to compute a CLUE in general, and in some cases even guarantee a best CLUE. In Section 4.4, it is shown that several established conservative estimators are CLUEs and best CLUEs given specific assumptions on $\mathcal{A}$. An overview of the best CLUE and its relationship to the BLUE is visualized in Figure 4.2.

4.2 Problem Properties

The BLUE and best CLUE formulations are similar. However, while a closed-form solution is available for the BLUE, the additional uncertainty inherent to the best CLUE makes this problem much more complicated, and no general solution procedure is available. This section highlights differences between the two optimization problems and derives a simplified optimization problem providing a lower bound

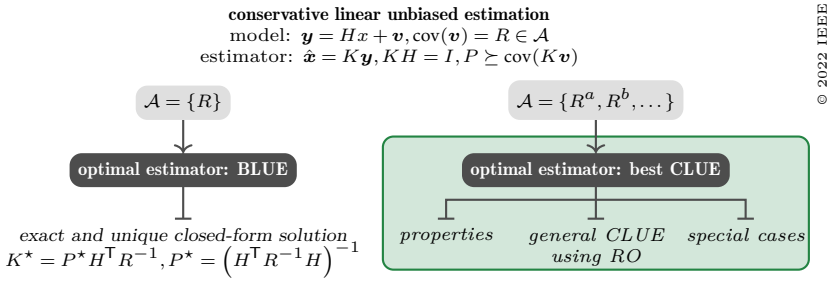


Figure 4.2. Overview of the conservative linear unbiased estimation problem. The special case with $\mathcal{A} = \{R\}$ is illustrated to the left, and the general case is illustrated to the right. The scope of this chapter is contained in the green box.

P_l of the obtainable covariance of the CLUE. In addition, an upper bound P_u is provided. It should be noted that P_l and P_u depend on $\mathcal{A}$.

4.2.1 Lower Bound on Best CLUE

The CLUE covariance $P^\star$ cannot be smaller than the BLUE covariance given any $S \in \mathcal{A}$. Consequently, it is possible to establish a lower bound for $P^\star$ by finding the smallest covariance that is greater than all BLUE covariances given $\mathcal{A}$. This approach simplifies the optimization issue compared to finding the optimal CLUE solution. In analogy to the CRLB, this lower bound can be used as a guideline for system design, e.g., to trade off between communication bandwidth and performance. It should be noted that this formulation relaxes the constraints, and hence there is no guarantee that a gain K achieving this covariance exists in the general case. A lower bound P_l for $P^\star$ is derived below, where subscript l refers to quantities related to the lower bound. It is shown that $J(P^\star) \geq J(P_l)$. If a CLUE (K, P) satisfies $J(P) = J(P_l)$, then this CLUE is also a best CLUE.

Consider the problem

$$\begin{aligned} \underset{P}{\text{minimize}} \quad & J(P) \\ \text{subject to} \quad & P \succeq (H^\mathsf{T} S^{-1} H)^{-1}, \forall S \in \mathcal{A}. \end{aligned} \tag{4.5}$$

For a given J a solution P_l to (4.5) is a lower bound[2] on $P^\star$.

Theorem 4.7. *Let $(K^\star, P^\star)$ and P_l be given by* (4.3) *and* (4.5)*, respectively. Then $J(P^\star) \geq J(P_l)$.*

Proof: By assumption, the same matrix increasing J and the same $\mathcal{A}$ are used in (4.3) and (4.5). Since $(K^\star, P^\star)$ solves (4.3) and as a consequence of the Gauss-Markov theorem [94], it holds, for each $S \in \mathcal{A}$, that

$$P^\star \succeq K^\star S (K^\star)^\mathsf{T} \succeq (H^\mathsf{T} S^{-1} H)^{-1},$$

Hence, $P^\star$ satisfies the constraints in (4.5) and therefore $J(P^\star) \geq J(P_l)$. □

[2] Section 4.5.1 provides an example in which the lower bound is strict.

From Theorem 4.7 it follows that if P_l is obtained by a CLUE, then this estimator is a best CLUE. Next, an estimator with the gain K_l computed from P_l is derived. In addition, sufficient conditions are provided for when (K_l, P_l) is a best CLUE. Start by solving for R_l in

$$P_l = (H^\mathsf{T} R_l^{-1} H)^{-1}, \tag{4.6}$$

which has a solution since $P_l \in \mathbb{S}_{++}^{n_x}$ and since H^T is full rank. Let

$$K_l = (H^\mathsf{T} R_l^{-1} H)^{-1} H^\mathsf{T} R_l^{-1}, \tag{4.7}$$

which yields an unbiased estimator since $K_l H = I$. For (K_l, P_l) to be a CLUE, it must hold that $P_l \succeq K_l S K_l^\mathsf{T}, \forall S \in \mathcal{A}$. As it follows from (4.7) that $K_l R_l K_l^\mathsf{T} = P_l$ with R_l according to (4.6), a sufficient condition for (K_l, P_l) to be a CLUE is that $R_l \succeq S, \forall S \in \mathcal{A}$ since then

$$P_l = K_l R_l K_l^\mathsf{T} \succeq K_l S K_l^\mathsf{T}, \forall S \in \mathcal{A}.$$

The results above are summarized in the following theorem.

Theorem 4.8. *Assume that P_l solves (4.5) and that K_l is according to (4.7) with R_l implicitly given in (4.6). If $R_l \succeq S, \forall S \in \mathcal{A}$, then (K_l, P_l) is a best CLUE.*

As will be seen in Section 4.5.1, it is possible to *not* satisfy $R_l \succeq S, \forall S \in \mathcal{A}$ while still satisfying $P_l \succeq K_l S K_l^\mathsf{T}, \forall S \in \mathcal{A}$.

4.2.2 Upper Bound on Best CLUE

Assume B such that $B \succeq S, \forall S \in \mathcal{A}$ is given. Then, it is possible to construct a CLUE as (K, P) with $P = KBK^\mathsf{T}$ for any K subject to $KH = I$. In particular, a CLUE can be derived by first finding a $B \succeq S, \forall S \in \mathcal{A}$, and then compute the BLUE w.r.t. this B. Finding a smallest covariance larger than all possible $R \in \mathcal{A}$ is a simpler problem than the best CLUE problem. However, approaching the problem in this way restricts the feasible set, and therefore a CLUE, but not necessarily a best CLUE, is obtained. Nevertheless, it is sometimes useful to bound $\mathcal{A}$ tightly using B, see, e.g., [10, 54, 65, 75, 146] and Section 4.4.4. In such cases the closed-form expression of (4.9) below can be used to compute a CLUE. Next, an upper bound P_u on $P^\star$ of (4.3) is derived. Subscript u is used for quantities related to the upper bound.

Introduce the set

$$\mathcal{B} = \{B \in \mathbb{S}_+^{n_y} \mid B \succeq S, \forall S \in \mathcal{A}\}, \tag{4.8}$$

which contains all matrices B that are larger than all elements $S \in \mathcal{A}$. A CLUE (K_u, P_u) is then given by

$$K_u = (H^\mathsf{T} B^{-1} H)^{-1} H^\mathsf{T} B^{-1}, \qquad P_u = (H^\mathsf{T} B^{-1} H)^{-1}, \tag{4.9}$$

where $B \in \mathcal{B}$ and $K_u H = I$. By a similar reasoning that leads up to Theorem 4.8, (K_u, P_u) according to (4.9) is a CLUE for any $B \in \mathcal{B}$.

A *minimal bound* is defined in Definition 4.9. It is also referred to as a tight bound and can be interpreted as a generalization of Definition 3.11.

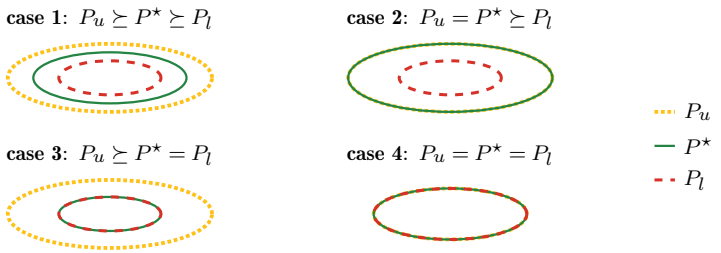


Figure 4.3. Four cases related to the mutual ordering of P_u, $P^\star$, and P_l.

Definition 4.9 (Minimal Bound). Assume that $B \succeq S, \forall S \in \mathcal{A}$. Then, B is a *minimal bound* on $\mathcal{A}$ if there exists no $A \neq B$ such that $B \succeq A$ and $A \succeq S, \forall S \in \mathcal{A}$.

In general (4.9) is too conservative to be a best CLUE, even if B is a minimal bound[3] on $\mathcal{A}$. These results are summarized in the next theorem.

Theorem 4.10. *Let $(K^\star, P^\star)$ be given by (4.3) and let (K_u, P_u) be given by (4.9), where $B \in \mathcal{B}$ with $\mathcal{B}$ as in (4.8). Then $J(P_u) \geq J(P^\star)$.*

4.2.3 Comments

If the approach in Section 4.2.1 yields a CLUE, then (K_l, P_l) is also a best CLUE. If not, then (K_l, P_l) is a *too optimistic* estimator. On the other hand, the estimator (K_u, P_u) given by (4.9) is generally *too pessimistic* to be a best CLUE. If the lower and upper bounds coincide, then, as a consequence of Theorem 4.8, a best CLUE is obtained. Four possible cases related to the mutual ordering of P_u, $P^\star$, and P_l are illustrated in Figure 4.3.

4.3 General Conservative Linear Unbiased Estimation

In this section, it is shown how RO [23] can be used to solve general CLUE problems[4]. First, it is shown that the CLUE problem fits into the *robust semidefinite programming* optimization framework [66]. Tractability[5] and optimality are then discussed. Finally, an implementation of conservative estimation using RO in `YALMIP` [114] is provided.

4.3.1 Robust Semidefinite Optimization

This chapter deals with optimization problems with semidefinite constraints. Hence, the focus is on a class of problems called *semidefinite programs* (SDP). Let $\Delta \in \mathcal{D}$ be an uncertain optimization parameter only known to reside in an *uncertainty*

[3] An example of this is provided in Section 4.5.1.
[4] Other estimation problems where RO is used are studied in, e.g., [23, 41].
[5] Tractability in the sense that a solution can be found within a reasonable amount of time.

set $\mathcal{D} \subset \mathbb{R}^p$. In RO, none of the constraints are allowed to be violated for any value $\Delta \in \mathcal{D}$ [22]. A generic SDP with an inequality constraint uncertainty can be stated as [66]

$$\begin{aligned} \underset{z}{\text{minimize}} \quad & f(z) \\ \text{subject to} \quad & \mathcal{F}(z,\Delta) \succeq 0, \Delta \in \mathcal{D}, \end{aligned} \tag{4.10}$$

for a loss function f. In (4.10), z is an optimization variable, and $\mathcal{F}(z,\Delta)$ is a matrix-valued function that depends on z and Δ. The constraint in (4.10) is a *linear matrix inequality* (LMI) for a certain fixed Δ.

The best CLUE problem stated in Definition 4.4 is aligned with the formulation in (4.10). To see this, replace z by (K,P), $\mathcal{D}$ by $\mathcal{A}$, Δ by S, and $f(z)$ by $J(P)$. Then rewrite the matrix inequality of (4.3) using *Schur complement* [67] as

$$\mathcal{F}(K,P,S) = \begin{bmatrix} P & KS \\ SK^\mathsf{T} & S \end{bmatrix} \succeq 0. \tag{4.11}$$

Since (4.11) is equivalent to $P \succeq 0 \wedge P - KSK^\mathsf{T} \succeq 0$, the problem in Definition 4.4 has been retrieved.

4.3.2 Tractability And Optimality

There are a few cases where the computed solution to the problem in (4.10), with $\mathcal{D} = \mathcal{A}$, is tractable and optimal in the sense that a minimal element is found when the RO problem is solved [21]. If $\mathcal{A}$ is a finite set, then tractability follows trivially since the uncertainty is replaced by a finite set of LMIs. The problem is also tractable if $\mathcal{A}$ is the *convex hull* [29] of a finite set. See Example 4.11 for an example of this.

Example 4.11: Convex Hull of a Finite Set

The convex hull of a set $\mathcal{V}$ is another set which contains all convex combinations of the elements of $\mathcal{V}$ [29]. For example, consider a set of covariances $\mathcal{V} = \{A, B\} \subset \mathbb{S}_+^2$ where $A = \left[\begin{smallmatrix} 4 & -2 \\ -2 & 1 \end{smallmatrix}\right]$ and $B = \left[\begin{smallmatrix} 4 & 2 \\ 2 & 1 \end{smallmatrix}\right]$. The convex hull of $\mathcal{V}$ is

$$\begin{aligned} \{\theta A + (1-\theta)B \,|\, \theta \in [0,1]\} &= \left\{ \begin{bmatrix} 4(\theta+1-\theta) & -2\theta+2(1-\theta) \\ -2\theta+2(1-\theta) & \theta+1-\theta \end{bmatrix} \,\middle|\, \theta \in [0,1] \right\} \\ &= \left\{ \begin{bmatrix} 4 & 2-4\theta \\ 2-4\theta & 1 \end{bmatrix} \,\middle|\, \theta \in [0,1] \right\} = \left\{ \begin{bmatrix} 4 & c \\ c & 1 \end{bmatrix} \,\middle|\, c \in [-2,2] \right\}. \end{aligned}$$

In cases where the unknown cross-covariance is not a scalar, it is generally not possible to express $\mathcal{A}$ as the convex hull of a finite set. Hence, for general uncertainty sets $\mathcal{A}$, there are only a few constructive results on robust counterparts for the problem in (4.10). The case treated here, where both the RO problem and the uncertainty set are defined by semidefinite constraints, is largely untreated in the literature. Not only are exact solutions absent in contrast to the simple example above, but also general, tractable, conservative approximations are missing.

4.3.3 Robust Estimation Using YALMIP

YALMIP is a MATLAB® toolbox developed to model and solve optimization problems [114], and it has the ability to derive RO problems [115]. The strategies in [115] focus on cases where exact robust counterparts can be derived which rules out problems according to the model in (4.10). However, theory has recently been developed and added to the YALMIP toolbox to support problems according to (4.10), i.e., uncertainty structures involving arbitrary intersections of conic-representable sets. These additions[6] are described in the forthcoming [113].

The key feature for us is realized by a function called `uncertain()`, which enables the uncertainty imposed by $R \in \mathcal{A}$ to be handled. A CLUE problem solved using RO in YALMIP is illustrated in Example 4.12.

Example 4.12: Robust Estimation Using YALMIP

Consider the task of computing a conservative estimate $\hat{x} = Ky$ and P where $\mathrm{tr}(P)$ is minimized. Let $y = [\begin{smallmatrix} y_1 \\ y_2 \end{smallmatrix}] = \left[\begin{smallmatrix} H_1 \\ H_2 \end{smallmatrix}\right] x + v$ where $x \in \mathbb{R}^2$, $H_1 = H_2 = I$, $R_1 = \mathrm{diag}(1,4)$ and $R_2 = \mathrm{diag}(4,1)$. Assume R_{12} is completely unknown and that $R \succeq 0$. The problem is translated into MATLAB® syntax using YALMIP below.

```
H = [eye(2);eye(2)];
R1 = diag([1 4]); R2 = diag([4 1]);
K = sdpvar(2,4); P = sdpvar(2); R12 = sdpvar(2,2,'full'); % declare SDP
    variables
R = [R1 R12;R12' R2];
F = [uncertain(R12),K*H==eye(2),[P K*R;R*K' R]>=0,R>=0]; % constraints
J = trace(P); % loss function
optimize(F,J) % solve problem
```

YALMIP functions are highlighted in **orange**. The result is $K = \left[\begin{smallmatrix} 0.8 & 0 & 0.2 & 0 \\ 0 & 0.2 & 0 & 0.8 \end{smallmatrix}\right]$ and $P = 1.6I$. In this case, YALMIP in fact finds a best CLUE which turns out to be equivalent to a solution computed using CI. However, in general the solution is approximative and the only guarantee is that the solution is a CLUE.

4.4 Special Conservative Linear Unbiased Estimation

In this section, it is shown that CI, ICI, and LE are best CLUEs under different assumptions on $\mathcal{A}$. In addition, several other methods are stated. Common to many methods is that the diagonal blocks of R are known while the off-diagonal blocks, e.g., R_{12}, are unknown. What differs between the methods is the assumption on the off-diagonal blocks. For instance, it could be that R_{12} is diagonal or that $\lambda_{\max}(R_{12}R_{12}^{\mathsf{T}}) < a$, where $a > 0$. The benefit of exploiting any correlation structure is illustrated in Example 4.13.

[6] This functionality is integrated into YALMIP, but the associated documentation is unpublished.

Example 4.13: Exploiting Correlation Structure

Assume $R = \left[\begin{smallmatrix} 4 & c \\ c & 1 \end{smallmatrix}\right]$, where c is unknown. If it is only known that $R \succeq 0$, then $c \in [-2,2]$, and hence R could be represented by any ellipse enclosed in the rectangle of Figure 4.4. A minimal bound on $\mathcal{A}$ is in this case given by B. On the other hand, if it is known that $c \in [-\frac{1}{2}, \frac{1}{2}]$, then it is possible to find an even smaller minimal bound B'. The figure also visualizes $\mathcal{A} = \{\left[\begin{smallmatrix} 4 & c \\ c & 1 \end{smallmatrix}\right] \,|\, c \in [-2,2]\}$ and $\mathcal{A}' = \{\left[\begin{smallmatrix} 4 & c \\ c & 1 \end{smallmatrix}\right] \,|\, c \in [-\frac{1}{2}, \frac{1}{2}]\} \subset \mathcal{A}$.

Figure 4.4. Example of utilizing correlation structure. *Left:* B is a minimal bound on $\mathcal{A}$. *Right:* A smaller minimal bound, $B' \prec B$, is possible for $\mathcal{A}' \subset \mathcal{A}$.

It is assumed in the following that (y_i, R_i) are available, where $i = 1, 2, \ldots, N$, and that R_{ij}, with $i \neq j$, are unknown. It should be emphasized that CI, ICI, and LE give different solutions to a problem since they are related to different assumptions on $\mathcal{A}$. Among CI, ICI, and LE, LE makes the most restrictive assumptions on $\mathcal{A}$ while CI makes the least restrictive assumptions on $\mathcal{A}$. Thereby, LE in general is less conservative than ICI, while ICI in general is less conservative than CI.

4.4.1 Covariance Intersection

CI is presented in Section 3.1.2. The method is originally based on *completely unknown cross-correlations* for which the only condition on the cross-correlation is that $R \succeq 0$, and therefore

$$\mathcal{A} = \left\{ \begin{bmatrix} S_1 & S_{12} \\ S_{21} & S_2 \end{bmatrix} \in \mathbb{S}_+^{n_y} \,\middle|\, S_1 = R_1, S_2 = R_2 \right\}. \tag{4.12}$$

Let $y = \mathrm{col}(y_1, \ldots, y_N)$, $H = \mathrm{col}(H_1, \ldots, H_N)$, and $B = \mathrm{diag}\left(\frac{R_1}{\omega_1}, \ldots, \frac{R_N}{\omega_N}\right)$. CI in information form is given by

$$P^{-1} = H^{\mathsf{T}} B^{-1} H = \sum_{i=1}^{N} \omega_i H_i^{\mathsf{T}} R_i^{-1} H_i, \quad P^{-1}\hat{x} = H^{\mathsf{T}} B^{-1} y = \sum_{i=1}^{N} \omega_i H_i^{\mathsf{T}} R_i^{-1} y_i, \tag{4.13}$$

where $\omega_i \in [0,1]$ and $\sum_{i=1}^{N} \omega_i = 1$. The parameters $\omega_1, \ldots, \omega_N$ are found by minimizing $J(P)$. The CI gain K is given by

$$K = \begin{bmatrix} K_1 & \ldots & K_N \end{bmatrix} = P \begin{bmatrix} \omega_1 H_1^{\mathsf{T}} R_1^{-1} & \ldots & \omega_N H_N^{\mathsf{T}} R_N^{-1} \end{bmatrix}. \tag{4.14}$$

Properties

CI is a linear unbiased estimation method [52]. Since CI also is conservative given completely unknown correlations [90], CI is a CLUE as long as all (y_i, R_i) to be fused are conservative.

Regarding optimality, let $N = 2$ and $H_1 = H_2 = I$. In this case, only one parameter ω is required since it is possible to define $\omega_1 = \omega$ and $\omega_2 = 1-\omega$. Let the optimal value of ω given $J(P)$ be denoted by $\omega^\star$. Moreover, let an arbitrary CLUE be given by (K', P'). In [150] it is shown that, if $\omega^\star$ is obtained by minimizing $J(P)$ w.r.t. ω with P given in (4.13), then for $(K^\star, P^\star)$ given by

$$K^\star = \begin{bmatrix} \omega^\star P^\star R_1^{-1} & (1-\omega^\star) P^\star R_2^{-1} \end{bmatrix}, \quad P^\star = \left(\omega^\star R_1^{-1} + (1-\omega^\star) R_2^{-1}\right)^{-1}, \tag{4.15}$$

it holds that

$$P' \preceq P^\star \implies P' = P^\star.$$

This means $P^\star$ is a minimal element of the feasible set. Hence, an estimator $(K^\star, P^\star)$ given according to (4.15) constitutes a best CLUE, provided that $N = 2$ and R_{12} is completely unknown.

4.4.2 Inverse Covariance Intersection

ICI is derived in [130] for the case where $N = 2$ and $H_1 = H_2 = I$. ICI is less conservative than CI, and is based on the assumption that R_{12} is due to common information[7]. Let $\Gamma^{-1} \in \mathbb{S}^{n_x}_{++}$ denote common information included in both R_1^{-1} and R_2^{-1}, and $\hat{\gamma}$ to denote the corresponding estimate such that $\Gamma = \mathrm{cov}(\hat{\gamma})$. The common information structure for two estimates is defined as

$$R_1^{-1} = (R_1^e)^{-1} + \Gamma^{-1}, \qquad R_1^{-1} y_1 = (R_1^e)^{-1} y_1^e + \Gamma^{-1}\hat{\gamma}, \tag{4.16a}$$

$$R_2^{-1} = (R_2^e)^{-1} + \Gamma^{-1}, \qquad R_2^{-1} y_2 = (R_2^e)^{-1} y_2^e + \Gamma^{-1}\hat{\gamma}, \tag{4.16b}$$

where $(R_i^e)^{-1}$ and y_i^e are the exclusive information and the exclusive estimate contained in the ith estimate, respectively. The notion exclusive here means that $\mathrm{cov}(\boldsymbol{y}_1^e, \boldsymbol{y}_2^e) = \mathrm{cov}(\boldsymbol{y}_1^e, \hat{\boldsymbol{\gamma}}) = \mathrm{cov}(\boldsymbol{y}_2^e, \hat{\boldsymbol{\gamma}}) = 0$. With y_1 and y_2 according to (4.16), the cross-covariance is given by [130]

$$R_{12} = R_1 \Gamma^{-1} R_2. \tag{4.17}$$

From (4.16) it follows that $R_1^{-1}, R_2^{-1} \succeq \Gamma^{-1}$. The set $\mathcal{A}$ is now given by

$$\mathcal{A} = \left\{ \begin{bmatrix} S_1 & S_{12} \\ S_{21} & S_2 \end{bmatrix} \in \mathbb{S}^{n_y}_{+} \;\middle|\; S_1 = R_1, S_2 = R_2, S_{12} = S_1 \Gamma^{-1} S_2, S_1^{-1}, S_2^{-1} \succeq \Gamma^{-1} \right\}. \tag{4.18}$$

An estimate is computed using ICI according to

$$P^{-1} = R_1^{-1} + R_2^{-1} - (\omega R_1 + (1-\omega) R_2)^{-1}, \tag{4.19a}$$

$$\begin{aligned} P^{-1}\hat{x} = &\left(R_1^{-1} - \omega\left(\omega R_1 + (1-\omega) R_2\right)^{-1}\right) y_1 \\ &+ \left(R_2^{-1} - (1-\omega)\left(\omega R_1 + (1-\omega) R_2\right)^{-1}\right) y_2, \end{aligned} \tag{4.19b}$$

[7] Correlations due to common information are described in Section 3.1.4.

where $\omega \in [0,1]$ is found by minimization of $J(P)$. The gain is given by $K = \begin{bmatrix} K_1 & K_2 \end{bmatrix}$, where

$$K_1 = P\left(R_1^{-1} - \omega\left(\omega R_1 + (1-\omega)R_2\right)^{-1}\right), \tag{4.20a}$$

$$K_2 = P\left(R_2^{-1} - (1-\omega)\left(\omega R_1 + (1-\omega)R_2\right)^{-1}\right), \tag{4.20b}$$

with P according to (4.19).

Generalization

A generalization of ICI is proposed in [52] for $N = 2$, but where H_1 and H_2 are not necessarily equal to I. Here, a modification of that algorithm is proposed. Before proceeding, some mathematical details need to be addressed. Let $(\hat{\gamma},\Gamma)$ correspond to a shared estimate as in (4.16), but assume $\Gamma = \mathsf{cov}(\hat{\boldsymbol{\gamma}})$ is infinite in $p < n_2$ components. Infinite variance is equivalent to zero information. Hence, the issue of infinite variance is conveniently handled by the information form. The information $\mathcal{I}_\gamma$ of $(\hat{\gamma},\Gamma)$ is given by

$$\mathcal{I}_\gamma = VDV^{\mathsf{T}}, \qquad D = \mathsf{diag}(d_1,\ldots,d_{n_x-p},0,\ldots,0) \in \mathbb{S}_+^{n_x}, \tag{4.21}$$

where $d_i > 0$ and $V = \begin{bmatrix} v_1 & \ldots & v_{n_x} \end{bmatrix}$ is an orthogonal matrix. In this sense, the components with infinite variance correspond to the directions $v_{n_x-p+1},\ldots,v_{n_x}$. Let $\Phi = \mathsf{col}(v_1^{\mathsf{T}},\ldots,v_{n_x-p}^{\mathsf{T}})$ such that $\mathcal{I}_\gamma = \Phi^{\mathsf{T}}(\Phi\Gamma\Phi^{\mathsf{T}})^{-1}\Phi$, where $\Phi\Gamma\Phi^{\mathsf{T}}$ is finite and invertible by construction. It follows that

$$\mathcal{I}_\gamma\Gamma\mathcal{I}_\gamma = \Phi^{\mathsf{T}}(\Phi\Gamma\Phi^{\mathsf{T}})^{-1}\Phi\Gamma\Phi^{\mathsf{T}}(\Phi\Gamma\Phi^{\mathsf{T}})^{-1}\Phi = \Phi^{\mathsf{T}}(\Phi\Gamma\Phi^{\mathsf{T}})^{-1}\Phi = \mathcal{I}_\gamma. \tag{4.22}$$

The proposed generalized ICI is stated in Algorithm 4.14 and is valid for $H_1 = I$ and arbitrary $H_2 \in \mathbb{R}^{n_2\times n_x}$ satisfying $\mathsf{rank}(H_2) = n_2$. Assume that $y_1 = x + v_1$, $y_2 = H_2x + v_2$, and $\hat{\gamma} = x + v_\gamma$. Let $\tilde{\boldsymbol{y}}_i^e = \boldsymbol{y}_i^e - H_ix$ and $\tilde{\boldsymbol{\gamma}} = \hat{\boldsymbol{\gamma}} - x$. Define $= R_i^e = \mathsf{cov}(\tilde{\boldsymbol{y}}_i^e)$ and $\Gamma = \mathsf{cov}(\tilde{\boldsymbol{\gamma}})$. Allow for the exclusive information $\mathcal{I}_i^e$ and the common information $\mathcal{I}_\gamma$ to be singular, but still assume $R_1 \in \mathbb{S}_{++}^{n_x}$ and $R_2 \in \mathbb{S}_{++}^{n_2}$. In this case Γ might be infinite in some directions. The common information decomposition is now defined as

$$R_1^{-1} = \mathcal{I}_1^e + \mathcal{I}_\gamma, \qquad R_1^{-1}y_1 = \mathcal{I}_1^e y_1^e + \mathcal{I}_\gamma\hat{\gamma}, \tag{4.23a}$$

$$R_2^{-1} = \mathcal{I}_2^e + H_2\mathcal{I}_\gamma H_2^{\mathsf{T}}, \qquad R_2^{-1}y_2 = \mathcal{I}_2^e y_2^e + H_2\mathcal{I}_\gamma\hat{\gamma}, \tag{4.23b}$$

where $\mathsf{cov}(\boldsymbol{y}_1^e,\boldsymbol{y}_2^e) = \mathsf{cov}(\boldsymbol{y}_1^e,\hat{\boldsymbol{\gamma}}) = 0$ and $\mathsf{cov}(\boldsymbol{y}_2^e,\hat{\boldsymbol{\gamma}}) = 0$. The cross-covariance is given by

$$\begin{aligned} R_{12} &= \mathrm{E}\left(\left(R_1\left(\mathcal{I}_1^e\boldsymbol{y}_1^e + \mathcal{I}_\gamma\hat{\boldsymbol{\gamma}}\right) - x\right)\left(R_2\left(\mathcal{I}_2^e\boldsymbol{y}_2^e + H_2\mathcal{I}_\gamma\hat{\boldsymbol{\gamma}}\right) - H_2x\right)^{\mathsf{T}}\right) \\ &= R_1\,\mathrm{E}\left(\left(\mathcal{I}_1^e\tilde{\boldsymbol{y}}_1^e + \mathcal{I}_\gamma\tilde{\boldsymbol{\gamma}}\right)\left(\mathcal{I}_2^e\tilde{\boldsymbol{y}}_2^e + H_2\mathcal{I}_\gamma\tilde{\boldsymbol{\gamma}}\right)^{\mathsf{T}}\right)R_2 = R_1\mathcal{I}_\gamma\,\mathrm{E}(\tilde{\boldsymbol{\gamma}}\tilde{\boldsymbol{\gamma}}^{\mathsf{T}})\mathcal{I}_\gamma H_2^{\mathsf{T}}R_2 \\ &= R_1\mathcal{I}_\gamma\Gamma\mathcal{I}_\gamma H_2^{\mathsf{T}}R_2 = R_1\mathcal{I}_\gamma H_2^{\mathsf{T}}R_2, \end{aligned} \tag{4.24}$$

Algorithm 4.14: Inverse Covariance Intersection

Input: Estimates (y_1, R_1) and (y_2, R_2), and H_2

The estimates are fused according to:

$$P^{-1} = R_1^{-1} + H_2^{\mathsf{T}} R_2^{-1} H_2 - \omega H_2^{\mathsf{T}} \Gamma_1^{-1} H_2 - (1-\omega) H_2^{\mathsf{T}} H_2 \Gamma_2^{-1} H_2^{\mathsf{T}} H_2, \tag{4.25a}$$

$$P^{-1}\hat{x} = \left(R_1^{-1} - \omega H_2^{\mathsf{T}} \Gamma_1^{-1} H_2\right) y_1 + \left(H_2^{\mathsf{T}} R_2^{-1} - (1-\omega) H_2^{\mathsf{T}} H_2 \Gamma_2^{-1} H_2^{\mathsf{T}}\right) y_2, \tag{4.25b}$$

where $\Gamma_1 = \omega H_2 R_1 H_2^{\mathsf{T}} + (1-\omega) R_2$ and $\Gamma_2 = \omega R_1 + (1-\omega) H_2^{\mathsf{T}} R_2 H_2$.

Output: $(\hat{x}, P)$

where $\Gamma = \mathrm{E}(\tilde{\gamma}\tilde{\gamma}^{\mathsf{T}})$ and (4.22) were used.

In [9], ICI is generalized for fusion of $N > 2$ estimates. In [129], it is shown that ICI produces conservative results even in the case of the *correlated information* structure. This structure is a generalization of the common information structure. It is also shown that ICI can handle common process noise under certain assumptions. The correlated information structure is also considered in [52] and is in particular assumed in the derivation of the generalization of ICI in [52]. However, since this structure is difficult to motivate in practice, it is not considered further in this scope.

Properties

In the following, it is assumed that $\omega \in (0,1)$. If $\omega \in \{0,1\}$, then there is no actual fusion performed, and conservativeness hence follows directly. An alternative formulation of Algorithm 4.14 is provided in Lemma 4.15. Theorem 4.16 states that ICI is a CLUE given that $H_2^{\mathsf{T}} R_2^{-1} H_2 \succeq \mathcal{I}_\gamma$ and that the decomposition in (4.23) holds. Loosely speaking, the assumption $H_2^{\mathsf{T}} R_2^{-1} H_2 \succeq \mathcal{I}_\gamma$ restricts the common information to be confined to a subspace common to both (y_1, R_1) and (y_2, R_2).

Lemma 4.15. *An estimate $(\hat{x}, P)$ computed as in Algorithm 4.14 is equivalent to $\hat{x} = PH^{\mathsf{T}} B^{-1} y$ and $P = (H^{\mathsf{T}} B^{-1} H)^{-1}$, with $y = \mathrm{col}(y_1, y_2)$, $H = \mathrm{col}(I, H_2)$, and*

$$B = \begin{bmatrix} R_1 + \frac{\omega}{1-\omega} R_1 H_2^{\mathsf{T}} R_2^{-1} H_2 R_1 & 0 \\ 0 & R_2 + \frac{1-\omega}{\omega} R_2 H_2 R_1^{-1} H_2^{\mathsf{T}} R_2 \end{bmatrix}. \tag{4.26}$$

Proof: Using *Woodbury's matrix identity* [181]

$$\begin{aligned} &\left(R_1 + \frac{\omega}{1-\omega} R_1 H_2^{\mathsf{T}} R_2^{-1} H_2 R_1\right)^{-1} \\ &\quad = R_1^{-1} - R_1^{-1} R_1 H_2^{\mathsf{T}} \left(H_2 R_1 R_1^{-1} R_1 H_2^{\mathsf{T}} + \frac{1-\omega}{\omega} R_2\right)^{-1} H_2 R_1 R_1^{-1} \\ &\quad = R_1^{-1} - \omega H_2^{\mathsf{T}} \left(\omega H_2 R_1 H_2^{\mathsf{T}} + (1-\omega) R_2\right)^{-1} H_2 = R_1^{-1} - \omega H_2^{\mathsf{T}} \Gamma_1^{-1} H_2, \end{aligned}$$

where $\Gamma_1 = \omega H_2 R_1 H_2^\mathsf{T} + (1-\omega) R_2$. Similarly

$$\left(R_1 + \frac{1-\omega}{\omega} R_2 H_2 R_1^{-1} H_2^\mathsf{T} R_2 \right)^{-1} = R_2^{-1} - (1-\omega) H_2^\mathsf{T} H_2 \Gamma_2^{-1} H_2^\mathsf{T} H_2,$$

where $\Gamma_2 = \omega R_1 + (1-\omega) H_2^\mathsf{T} R_2 H_2$. Hence

$$\begin{aligned} P^{-1} &= H^\mathsf{T} B^{-1} H = R_1^{-1} + H_2^\mathsf{T} R_2 H_2 - \omega H_2^\mathsf{T} \Gamma_1^{-1} H_2 - (1-\omega) H_2^\mathsf{T} H_2 \Gamma_2^{-1} H_2^\mathsf{T} H_2, \\ P^{-1}\hat{x} &= H^\mathsf{T} B^{-1} y \\ &= \left(R_1^{-1} y_1 - \omega H_2^\mathsf{T} \Gamma_1^{-1} H_2 \right) y_1 + \left(H_2^\mathsf{T} R_2^{-1} - (1-\omega) H_2^\mathsf{T} H_2 \Gamma_2^{-1} H_2^\mathsf{T} \right) y_2, \end{aligned}$$

which is equivalent to the output of Algorithm 4.14. □

Theorem 4.16. *Assume that* (4.23) *holds and that* $H_2^\mathsf{T} R_2^{-1} H_2 \succeq \mathcal{I}_\gamma$. *Then ICI defined in Algorithm 4.14 is a CLUE.*

Proof: Let B be given by (4.26). Lemma 4.15 states that the ICI gain is given by $K = (H^\mathsf{T} B^{-1} H)^{-1} H^\mathsf{T} B^{-1}$. Since $KH = I$, ICI is a linear unbiased estimator. To see that ICI also is conservative given (4.23), consider B defined according to (4.26). By assumption $R_1^{-1} \succeq \mathcal{I}_\gamma$ and $H_2^\mathsf{T} R_2^{-1} H_2 \succeq \mathcal{I}_\gamma$. Hence, with $R = \left[\begin{smallmatrix} R_1 & R_{12} \\ R_{21} & R_2 \end{smallmatrix} \right]$, $\mu = \frac{1-\omega}{\omega}$, and R_{12} according to (4.24), the difference $B - R$ satisfies

$$\begin{bmatrix} \frac{1}{\mu} R_1 H_2^\mathsf{T} R_2^{-1} H_2 R_1 & -R_1 \mathcal{I}_\gamma H_2^\mathsf{T} R_2 \\ -R_2 H_2 \mathcal{I}_\gamma R_1 & \mu R_2 H_2 R_1^{-1} H_2^\mathsf{T} R_2 \end{bmatrix} \succeq \begin{bmatrix} \frac{1}{\mu} R_1 \mathcal{I}_\gamma R_1 & -R_1 \mathcal{I}_\gamma H_2^\mathsf{T} R_2 \\ -R_2 H_2 \mathcal{I}_\gamma R_1 & \mu R_2 H_2 \mathcal{I}_\gamma H_2^\mathsf{T} R_2 \end{bmatrix}$$

Let $a = \mathrm{col}(a_1, a_2)$, where $a_1 \in \mathbb{R}^{n_x}$ and $a_2 \in \mathbb{R}^{n_2}$. By definition $B - R \succeq 0$ i.f.f. $a^\mathsf{T}(B-R)a \geq 0$ for all a [67]. Define $b = \mathrm{col}(b_1, b_2)$, where $b_1 = \sqrt{1/\mu} R_1 a_1$ and $b_2 = -\sqrt{\mu} H_2^\mathsf{T} R_2 a_2$. Then, since $\left[\begin{smallmatrix} \mathcal{I}_\gamma & \mathcal{I}_\gamma \\ \mathcal{I}_\gamma & \mathcal{I}_\gamma \end{smallmatrix} \right]$ is PSD, it holds, for all a, that

$$a^\mathsf{T} (B-R)\, a \geq b^\mathsf{T} \begin{bmatrix} \mathcal{I}_\gamma & \mathcal{I}_\gamma \\ \mathcal{I}_\gamma & \mathcal{I}_\gamma \end{bmatrix} b \geq 0.$$

Hence $B \succeq S, \forall S \in \mathcal{A}$, where $\mathcal{A}$ is according to (4.18), and it follows that ICI is a CLUE under the given assumptions[8]. □

Concerning optimality, assume that (4.16) holds such that $\mathcal{A}$ is according to (4.18). In [130] it is shown that if (K', P') is an arbitrary CLUE and $P^\star$ is computed according to (4.19), then

$$P' \preceq P^\star \implies P' = P^\star.$$

Hence, ICI is a best CLUE given that $N = 2$ and $\mathcal{A}$ is according to (4.18).

[8] This proof is inspired by the proof of a similar result in [129].

4.4.3 Largest Ellipsoid Method

The LE method first appeared in [24], but now goes by several names. In [73] it is called *safe fusion*, the authors of [159, 160] suggest the name *ellipsoidal intersection*, and in [188] it is referred to as *internal ellipsoid approximation*. It should be noted that there are minor differences in how the estimate is computed. The LE algorithm used here, and the generalization thereof, are based on [73].

In the derivations of LE, no explicit assumptions on $\mathcal{A}$ are made. Below, the *componentwise aligned correlations* structure is proposed. Componentwise aligned correlations are satisfied if there exists a joint transformation $T_J = \mathsf{diag}(T, T)$ such that

$$T_J R T_J^{\mathsf{T}} = \begin{bmatrix} I & D_{12}' \\ D_{12}' & D_2' \end{bmatrix}, \tag{4.27}$$

where D_2' and D_{12}' are diagonal. The condition in (4.27) is equivalent to

$$\mathcal{A} = \left\{ \begin{bmatrix} S_1 & S_{12} \\ S_{21} & S_2 \end{bmatrix} \in \mathbb{S}_+^{n_y} \,\middle|\, \begin{array}{l} S_1 = R_1, S_2 = R_2 \\ \exists T, T S_1 T^{\mathsf{T}} = I \wedge [T S_2 T^{\mathsf{T}}]_{ij} = [T S_{12} T^{\mathsf{T}}]_{ij} = 0, i \neq j \end{array} \right\}. \tag{4.28}$$

In [52], a generalized version of the LE method is proposed for the case when $H_1 = I$ while H_2 is arbitrary. This version is defined in Algorithm 4.17. In contrast to the original version, cf. [73], it is given in information form to avoid singularities. If $H_2 = I$, then the LE method computes an estimate $(\hat{x}, P)$ as

$$\hat{x} = T^{-1}\hat{x}', \qquad P = T^{-1} P' T^{-\mathsf{T}}, \tag{4.29a}$$

where $T = T_2 T_1$,

$$R_1 = U_1 \Sigma_1 U_1^{\mathsf{T}}, \qquad T_1 = \Sigma_1^{-\frac{1}{2}} U_1^{\mathsf{T}}, \tag{4.30a}$$

$$T_1 R_2 T_1^{\mathsf{T}} = U_2 \Sigma_2 U_2^{\mathsf{T}}, \qquad T_2 = U_2^{\mathsf{T}}, \tag{4.30b}$$

$$y_1' = T y_1, \qquad D_1' = T R_1 T^{\mathsf{T}} = I, \tag{4.30c}$$

$$y_2' = T y_2, \qquad D_2' = T R_2 T^{\mathsf{T}}, \tag{4.30d}$$

and

$$\left([\hat{x}']_i, [P']_{ii}\right) = \begin{cases} \left([y_1']_i, 1\right), & \text{if } 1 \leq [D_2']_{ii}, \\ \left([y_2']_i, [D_2']_{ii}\right), & \text{if } 1 > [D_2']_{ii}. \end{cases} \tag{4.31}$$

Properties

Consider the quantities in (4.29)–(4.31). The resulting gain of LE is given by

$$K = \begin{bmatrix} K_1 & K_2 \end{bmatrix} = T^{-1} \begin{bmatrix} K_1' & K_2' \end{bmatrix} T, \tag{4.32}$$

where K_1' and K_2' are the gains in the transformed domain, i.e., after transformation using T. The matrix K_1' is diagonal, where $[K_1']_{ii} = 1$ if $[D_2']_{ii} \geq 1$ and otherwise zero, and $K_2' = I - K_1'$ [52]. Since by assumption $H = \begin{bmatrix} I & I \end{bmatrix}^{\mathsf{T}}$,

$$KH = T \begin{bmatrix} K_1' & I - K_1' \end{bmatrix} T^{-1} \begin{bmatrix} I \\ I \end{bmatrix} = T K_1' T^{-1} + T I T^{-1} - T K_1' T^{-1} = I. \tag{4.33}$$

Algorithm 4.17: The Largest Ellipsoid Method

Input: Estimates (y_1, R_1), (y_2, R_2), and H_2

The estimates are fused according to:

1. Transform to the information domain

$$\iota_1 = R_1^{-1} y_1, \qquad \mathcal{I}_1 = R_1^{-1}, \qquad \iota_2 = H_2^{\mathsf{T}} R_2^{-1} y_2, \qquad \mathcal{I}_2 = H_2^{\mathsf{T}} R_2^{-1} H_2.$$

2. Factorize $\mathcal{I}_1 = U_1 \Sigma_1 U_1^{\mathsf{T}}$ and let $T_1 = \Sigma_1^{-\frac{1}{2}} U_1^{\mathsf{T}}$. Factorize $T_1 \mathcal{I}_2 T_1^{\mathsf{T}} = U_2 \Sigma_2 U_2^{\mathsf{T}}$ and let $T_2 = U_2^{\mathsf{T}}$. Transform using $T = T_2 T_1$ according to

$$\iota_1' = T\iota_1, \qquad \mathcal{I}_1' = T\mathcal{I}_1 T^{\mathsf{T}} = I, \qquad \iota_2' = T\iota_2, \qquad \mathcal{I}_2' = T\mathcal{I}_2 T^{\mathsf{T}}.$$

3. For each $i = 1, \ldots, n_x$, compute

$$\left([\iota']_i, [\mathcal{I}']_{ii}\right) = \begin{cases} \left([\iota_1']_i, 1\right), & \text{if } 1 \geq [\mathcal{I}_2']_{ii}, \\ \left([\iota_2']_i, [\mathcal{I}_2']_{ii}\right), & \text{if } 1 < [\mathcal{I}_2']_{ii}. \end{cases}$$

Output: $\hat{x} = PT^{-1}\iota'$ and $P = (T^{-1}\mathcal{I}' T^{-\mathsf{T}})^{-1}$

By construction, the LE method defined in (4.29) is a linear unbiased estimator. Theorem 4.18 states that LE is a best CLUE given that $\mathcal{A}$ is according to (4.28).

Theorem 4.18. *Assume that $y_1 = x + v_1$ and $y_2 = x + v_2$, where $R_1 = \mathrm{cov}(\boldsymbol{v}_1)$ and $R_2 = \mathrm{cov}(\boldsymbol{v}_2)$. If $\mathcal{A}$ is according to* (4.28)*, then the LE method given in* (4.29) *is a best CLUE.*

Proof: By assumption $T_J R T_J^{\mathsf{T}} = \begin{bmatrix} I & D_{12}' \\ D_{12}' & D_2' \end{bmatrix}$, where D_2' and D_{12}' are diagonal. The ith component of the first estimate is only correlated with the ith component of the other estimate. Hence, only pairwise correlated scalars need to be considered. It is then possible to use CI for the merging of scalars correlated to an unknown degree. If P' is the covariance in the transformed domain, then

$$[P']_{ii} = \omega[I]_{ii} + (1-\omega)[D_2']_{ii} = \omega + (1-\omega)[D_2']_{ii},$$

which, as a property of CI, is conservative for all $\omega \in [0,1]$. Minimizing $[P']_{ii}$ w.r.t. ω is equivalent to $[P']_{ii} = \min\left(1, [D_2']_{ii}\right)$, which in particular is the LE solution. Hence, since LE also is a linear unbiased estimator, it is a best CLUE given that $\mathcal{A}$ is according to (4.28). □

In [160] it is shown that P given by (4.29) can be computed as

$$P^{-1} = R_1^{-1} + R_2^{-1} - \Gamma^{-1}, \tag{4.34}$$

where Γ^{-1} is the interpreted as the maximum possible common information contained in R_1^{-1} and R_2^{-1}. Moreover, as pointed out in [128], the LE method in (4.29) is equivalent to the BSC fuser[9], with $R_{12} = T^{-1} D_{12}' T^{-\mathsf{T}}$, where T is the

[9] The BSC fuser is described in Section 3.1.2.

transformation matrix defined in (4.30) and the diagonal matrix D'_{12} is computed as

$$[D'_{12}]_{ii} = \min\left(1, [D'_2]_{ii}\right), \tag{4.35}$$

with D'_2 according to (4.30).

4.4.4 Other Related Methods

CI is the most well-known conservative estimation algorithm and is the parent of many other similar methods [53]. Two of the most popular derivates of CI are ICI and LE. Yet, it should be pointed out that other alternatives exist. Some of these alternatives are briefly summarized in this section based on the work in [53]. Many of the methods described below are based on the derivation of an upper bound B on S such that $B \succeq S, \forall S \in \mathcal{A}$. If such a bound B exists, then a CLUE is easily constructed using the methodology discussed in Section 4.2.2.

This thesis only considers linear track fusion methods. For a survey on nonlinear fusion methods, the paper [53] is suggested.

Federated Kalman Filter

The *federated Kalman filter* (FKF) was proposed before CI [32, 33], but can be interpreted as a special case of CI tailored to handle correlations induced by common process noise. This is accomplished by a reinitialization step, where the ith local estimate $(y_{i,k|k}, R_{i,k|k})$ is reinitialized at each k from a central estimate $(\hat{x}_{k|k}, P_{k|k})$. This is followed by process noise covariance inflation. Assume the SSM in Section 3.1.1. The FKF reinitializes the ith local estimates as

$$y_{i,k|k} = \hat{x}_{k|k}, \qquad R_{i,k|k} = \frac{1}{\beta_i} P_{k|k}, \tag{4.36}$$

and then inflates the process noise covariance according to

$$y_{i,k+1|k} = F_k y_{i,k|k}, \qquad R_{i,k+1|k} = F_k R_{i,k|k} F_k^{\mathsf{T}} + \frac{1}{\beta_i} Q_k, \tag{4.37}$$

where $\beta_i > 0$ and $\sum_{i=1}^{N} \beta_i = 1$. By doing this, the FKF creates an upper bound on $\mathrm{cov}(\boldsymbol{v}_{k+1|k})$. However, by construction, the FKF is in general not able to handle correlations due to common information.

Split Covariance Intersection

In *split CI* [93], it is assumed that each of the local estimates can be split into an independent part and a dependent part. For $N = 2$, this splitting is characterized by

$$y_1 = y_1^e + y_1^d, \qquad R_1 = R_1^e + R_1^d, \qquad y_2 = y_2^e + y_2^d, \qquad R_2 = R_2^e + R_2^d, \tag{4.38}$$

where superscript e is used for the independent part and superscript d is used for the dependent part[10]. It is assumed that $\mathrm{cov}(\boldsymbol{y}_i^e, \boldsymbol{y}_i^d) = 0$ for $i = 1, 2$, $\mathrm{cov}(\boldsymbol{y}_1^e, \boldsymbol{y}_2^e) =$

[10] This additive splitting is related to, but should not be confused with, the decomposition in (4.16) where the estimates are additive in the information domain.

0, and $\mathsf{cov}(\boldsymbol{y}_1^d, \boldsymbol{y}_2^d) = R_{12}^d$. Such a decomposition can be exploited, e.g., when it comes to the fusion of two estimates directly after local measurement updates [126].

Split CI implicitly assumes that the R_{12} is bounded as a consequence of $(R_{12}^d)^{\mathsf{T}}(R_1^d)^{-1}(R_{12}^d) \preceq R_2^d$ [53]. If (4.38) holds, then

$$R = \begin{bmatrix} R_1^e + R_1^d & R_{12}^d \\ (R_{12}^d)^{\mathsf{T}} & R_2^e + R_2^d \end{bmatrix} \preceq \begin{bmatrix} R_1^e & 0 \\ 0 & R_2^e \end{bmatrix} + \begin{bmatrix} \frac{1}{\omega} R_1^d & 0 \\ 0 & \frac{1}{1-\omega} R_2^d \end{bmatrix} = B, \tag{4.39}$$

with $\omega \in (0,1)$, where the last term is identical to the upper bound used in CI. Split CI has been applied in many practical problems, see, e.g., [89, 91, 104, 105] and [42, 111] for more recent studies.

Partitioned Covariance Intersection

Let $N = 2$ and assume that the two local estimates are partitioned as

$$y_1 = \begin{bmatrix} y_1^a \\ y_1^b \end{bmatrix}, \quad R_1 = \begin{bmatrix} R_1^{aa} & R_1^{ab} \\ R_1^{ba} & R_1^{bb} \end{bmatrix}, \quad y_2 = \begin{bmatrix} y_2^c \\ y_2^d \end{bmatrix}, \quad R_2 = \begin{bmatrix} R_2^{cc} & R_2^{cd} \\ R_2^{dc} & R_2^{dd} \end{bmatrix}, \tag{4.40}$$

where $R_\ell^{ij} = \mathsf{cov}(\boldsymbol{y}_\ell^i, \boldsymbol{y}_\ell^j)$. Say now that $R_{12}^{bd} = \mathsf{cov}(\boldsymbol{y}_1^b, \boldsymbol{y}_2^d)$ is unknown. Then an upper bound B on R is given by [138]

$$R = \begin{bmatrix} R_1^{aa} & R_1^{ab} & R_{12}^{ac} & R_{12}^{ad} \\ R_1^{ba} & R_1^{bb} & R_{12}^{bc} & R_{12}^{bd} \\ R_{12}^{ca} & R_{12}^{cb} & R_2^{cc} & R_2^{cd} \\ R_{12}^{da} & R_{12}^{db} & R_2^{dc} & R_2^{dd} \end{bmatrix} \preceq \begin{bmatrix} R_1^{aa} & R_1^{ab} & R_{12}^{ac} & R_{12}^{ad} \\ R_1^{ba} & \frac{1}{\omega} R_1^{bb} & R_{12}^{bc} & 0 \\ R_{12}^{ca} & R_{12}^{cb} & R_2^{cc} & R_2^{cd} \\ R_{12}^{da} & 0 & R_2^{dc} & \frac{1}{1-\omega} R_2^{dd} \end{bmatrix} = B, \tag{4.41}$$

where $\omega \in (0,1)$. This structure is exploited in the *partitioned CI* (PCI, [138]). In [8] it shown that the bound in (4.41) is overly conservative. The same paper proposes a tight bound B for the considered partitioning.

Factorized Covariance Intersection

In the *factorized CI* (FCI, [1, 2]) a structure closely related to (4.41) is exploited. Assume that $N = 2$ and that the two local estimates are partitioned as in (4.40), but with $c = a$ and $d = b$. In particular, FCI is based on the assumption that after some joint transformation $T_J = \mathsf{diag}(T, T)$ the covariance is given by

$$T_J R T_J^{\mathsf{T}} = R' = \begin{bmatrix} R_1^{aa'} & 0 & R_{12}^{aa'} & 0 \\ 0 & R_1^{bb'} & 0 & R_{12}^{bb'} \\ R_{21}^{aa'} & 0 & R_2^{aa'} & 0 \\ 0 & R_{21}^{bb'} & 0 & R_2^{bb'} \end{bmatrix}. \tag{4.42}$$

Assume now that all off-diagonal blocks of R' in (4.42) are unknown. An upper bound is in this case given by [2]

$$R' \preceq \mathsf{diag}\left(\frac{1}{\omega^a} R_1^{aa'}, \frac{1}{\omega^b} R_1^{bb'}, \frac{1}{1-\omega^a} R_2^{aa'}, \frac{1}{1-\omega^b} R_2^{bb'} \right) = B' = T_J B T_J^{\mathsf{T}}, \tag{4.43}$$

where $\omega^a, \omega^b \in (0,1)$ are optimized separately.

Bounded Correlation-Coefficient

Split CI, PCI, and FCI are all based on bounding the admissible set $\mathcal{A}$ with B by exploiting some additional structure in R. Once a bound is found, a CLUE can be easily constructed using the methodology discussed in Section 4.2.2. The particular bounds for split CI, PCI, and FCI are all derived using the CI framework, which involves tuning of the ω parameters.

Other work that considers bounded correlations is found in [75, 146] for $N = 2$. These papers independently derive bounds on R when R_{12} is on the form

$$(R_{12} - G)^{\mathsf{T}} R_1^{-1} (R_{12} - G) \preceq \rho_{\max}^2 R_2, \tag{4.44}$$

where $\rho_{\max} \in (0, 1]$ can be interpreted as a scalar correlation coefficient[11]. For a particular choice of G and $\rho_{\max}$, the constraint in (4.44) implies that

$$\mathcal{A} = \left\{ \begin{bmatrix} S_1 & S_{12} \\ S_{21} & S_2 \end{bmatrix} \in \mathbb{S}_+^{n_y} \;\middle|\; \begin{array}{l} S_1 = R_1, S_2 = R_2 \\ (S_{12} - G)^{\mathsf{T}} S_1^{-1} (S_{12} - G) \preceq \rho_{\max}^2 S_2 \end{array} \right\}. \tag{4.45}$$

An upper bound $B \succeq S, \forall S \in \mathcal{A}$ is in this case given by

$$B = \begin{bmatrix} (1 + \alpha\rho_{\max}) R_1 & G \\ G^{\mathsf{T}} & (1 + \frac{\rho_{\max}}{\alpha}) R_2 \end{bmatrix}, \tag{4.46}$$

where $\alpha = \frac{1}{\omega} - 1$ and $\omega \in (0, 1)$. If $G = 0$, then B in (4.46) is a minimal bound on $\mathcal{A}$ of (4.45) [75].

Bounded correlations are utilized in a Bayesian framework in [47]. In another recent paper [81], tight correlation bounds are derived using matrix decomposition techniques [10]. Specific bounds, including limiting cases, are studied in [10].

4.5 Theory and Method Evaluation

CI, ICI, and LE are now compared to each other and to the general CLUE approach based on RO. In Section 4.5.1, the methods are compared on several simple fusion examples adapted from [56]. In Section 4.5.2, a DTT scenario is considered where the aforementioned methods are used for track fusion.

4.5.1 Simple Fusion Examples

In this section, five fusion examples E1–E5 are solved. The covariances P^{CI}, P^{ICI}, and P^{LE} corresponding to CI, ICI, and LE, respectively, the lower bound P_l, and the upper bound P_u are computed wherever applicable. Each example is also solved using the RO approach, where the resulting covariance is denoted by P^{RO}. In examples E1–E4 it is assumed that $H_i = I \in \mathbb{R}^{2\times 2}$ for $i = 1, \ldots, N$ and $H = \text{col}(H_1, \ldots, H_N)$. In all cases $J(P) = \text{tr}(P)$.

[11] The setting with $\rho_{\max} = 0$ is not of interest in this context since then $R_{12} \equiv 0$.

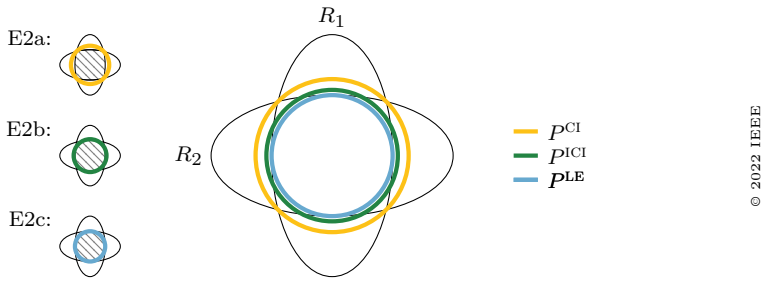


Figure 4.5. Summary of E2. If it is possible to exploit more structure in the problem, then it is possible to compute a CLUE having a smaller covariance. The hashed areas in the left part illustrate $(H^{\mathsf{T}}S^{-1}H)^{-1}, \forall S \in \mathcal{A}$.

E1: $\mathcal{A}$ Is A Finite Set

Assume that $N = 2$ and $R = \begin{bmatrix} R_1 & R_{12} \\ R_{21} & R_2 \end{bmatrix}$, where $R_1 = \left[\begin{smallmatrix} 1 & 0 \\ 0 & 4 \end{smallmatrix}\right]$ and $R_2 = \left[\begin{smallmatrix} 4 & 0 \\ 0 & 1 \end{smallmatrix}\right]$, and where $R_{12} \in \mathbb{R}^{2\times 2}$ can either be I or $-I$. Then $\mathcal{A} = \{R^a, R^b\} \in \mathbb{S}^4_{++}$, where $R^a = \begin{bmatrix} R_1 & I \\ I & R_2 \end{bmatrix}$ and $R^b = \begin{bmatrix} R_1 & -I \\ -I & R_2 \end{bmatrix}$.

The BLUE for $R = R^a$ is given by

$$K_a = \left(H^{\mathsf{T}}(R^a)^{-1}H\right)^{-1} H^{\mathsf{T}}(R^a)^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{4.47}$$

which yields $K_a R^a K_a^{\mathsf{T}} = K_a R^b K_a^{\mathsf{T}} = I$. From $KR^aK^{\mathsf{T}} \succeq K_a R^a K_a^{\mathsf{T}}, \forall K$ subject to $KH = I$ and $P^\star \succeq K_a R^b K_a^{\mathsf{T}}$ it follows that a best CLUE is given by $K^\star = K_a$ and $P^\star = I$.

Since $(H^{\mathsf{T}}(R^b)^{-1}H)^{-1} = 0.43I \prec (H^{\mathsf{T}}(R^a)^{-1}H)^{-1}$ we have $P_l = I$. Using the minimal bound $\bar{R} = \mathsf{diag}\,(R_1 + I, R_2 + I) = \mathsf{diag}(2,5,5,2)$ a strictly upper bound $P_u = 1.43I$ can be computed. RO yields $P^{\mathrm{RO}} = I$, which is guaranteed to be optimal since $\mathcal{A}$ is finite.

E2: $\mathcal{A}$ Is An Infinite Set

Assume that $N = 2$ and $R = \begin{bmatrix} R_1 & R_{12} \\ R_{21} & R_2 \end{bmatrix}$, with R_1 and R_2 defined as in E1. Assume that R_{12} is unknown and that $\mathcal{A}$ is now an infinite set. A best CLUE depends on $\mathcal{A}$. This problem is solved for three different assumptions on $\mathcal{A}$: a) completely unknown cross-correlations; b) common information; and c) componentwise aligned correlations. Since R_1 and R_2 are fixed, CI, ICI, and LE yield $P^{\mathrm{CI}} = 1.60I$, $P^{\mathrm{ICI}} = 1.18I$, and $P^{\mathrm{LE}} = I$, respectively, in all sub-cases below. E2 is summarized in Figure 4.5.

E2a) In this case, $\mathcal{A}$ is given by (4.12). Consider the two elements, $R^a = \begin{bmatrix} R_1 & G \\ G^{\mathsf{T}} & R_2 \end{bmatrix} \in \mathcal{A}$ and $R^b = \begin{bmatrix} R_1 & -G \\ -G^{\mathsf{T}} & R_2 \end{bmatrix} \in \mathcal{A}$ where $G = \left[\begin{smallmatrix} 0 & 0.999 \\ 3.999 & 0 \end{smallmatrix}\right]$. Solving (4.5), but substituting $\mathcal{A}$ with $\mathcal{A}' = \{R^a, R^b\}$, yields approximately $1.60I$. Since $\mathcal{A}' \subset \mathcal{A}$, it is concluded that $P_l \succeq 1.60I$.

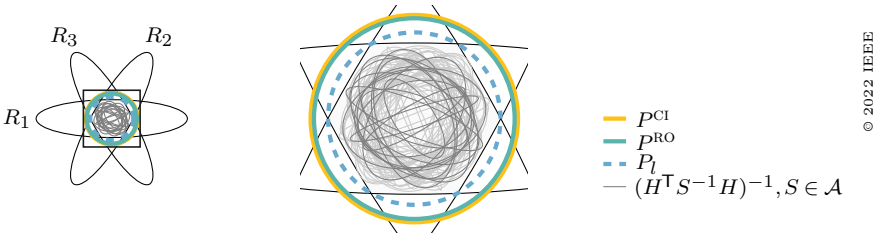


Figure 4.6. Summary of E3. Since $P^{\mathrm{CI}} \succ P^{\mathrm{RO}} \succ P_l$, CI is not a best CLUE when $N > 2$.

The matrix $B = 2\,\mathbf{diag}(R_1, R_2)$ satisfies $B \succeq S, \forall S \in \mathcal{A}$. This is true since $B - R = \left[\begin{smallmatrix} R_1 & -R_{12} \\ -R_{21} & R_2 \end{smallmatrix}\right] \succeq 0$ as a consequence of the assumption $R \succeq 0$. Using (4.9) yields $P_u = 1.60I$, which is equivalent to the solution of (4.5). Using Theorem 4.8 it is concluded that a best CLUE is given by (4.9) with $B = 2\,\mathbf{diag}(R_1, R_2)$.

E2b) In this case, $\mathcal{A}$ is given by (4.18). Solving the problem using `YALMIP` yields $P^{\mathrm{RO}} = 1.18I$ which is equivalent to the best CLUE solution $P^{\star} = P^{\mathrm{ICI}} = 1.18I$ computed using ICI.

E2c) In this case, $\mathcal{A}$ is given by (4.28). K_a according to (4.47) yields $K_a S K_a^{\mathsf{T}} = I, \forall S \in \mathcal{A}$. Hence, $K^{\star} = K_a$ and $P^{\star} = I$ constitute a best CLUE, where $P^{\star} = P^{\mathrm{LE}}$ since LE is a best CLUE in case of componentwise aligned correlations.

The matrix $R_l = \left[\begin{smallmatrix} R_1 & I \\ I & R_2 \end{smallmatrix}\right]$ does not satisfy $R_l \succeq S, \forall S \in \mathcal{A}$, e.g., for $C = \mathbf{diag}(R_1, R_2) \in \mathcal{A}$ the difference $R_l - C$ is indefinite. However, the matrix $B = 2\,\mathbf{diag}(R_1, R_2)$ satisfies $B \succeq S, \forall S \in \mathcal{A}$ and is also a minimal bound on $\mathcal{A}$. This B yields $\left(H^{\mathsf{T}} B^{-1} H\right)^{-1} = 1.60I \succ P^{\star}$.

E3: Fusion of Three Estimates

Let $N = 3$ and assume

$$R_1 = \begin{bmatrix} 16 & 0 \\ 0 & 1 \end{bmatrix}, \qquad R_2 = \begin{bmatrix} 4.75 & 6.50 \\ 6.50 & 12.25 \end{bmatrix}, \qquad R_3 = \begin{bmatrix} 4.75 & -6.50 \\ -6.50 & 12.25 \end{bmatrix}.$$

Assume that the off-diagonal blocks of R are completely unknown.

CI yields $P^{\mathrm{CI}} = 1.88I$. In this case, `YALMIP` gives us $P^{\mathrm{RO}} = 1.76I \prec P^{\mathrm{CI}}$. Hence, CI is not a best CLUE under completely unknown cross-correlations if $N > 2$. Also, $P_l = 1.31I$ is computed as the smallest ellipse that contains the intersection of the ellipses of R_1, R_2, and R_3. However, it is not possible to draw any conclusions about whether this P_l is a strictly lower bound on a best CLUE or not. The example is illustrated in Figure 4.6.

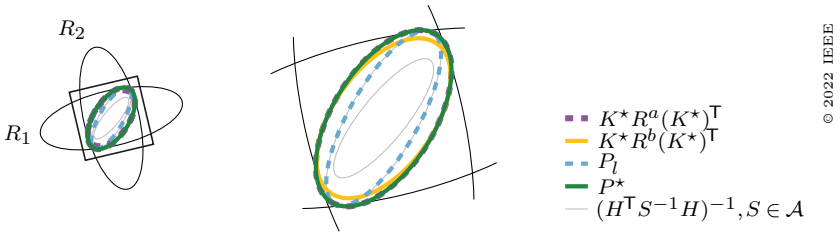


Figure 4.7. Summary of E4. In this example P_l is a strictly lower bound on $P^\star$.

E4: Lower Bound Is Strict

This example illustrates a case where $P^\star \neq P_l, P^\star \succeq P_l$. Assume that $N = 2$ and $R = \begin{bmatrix} R_1 & R_{12} \\ R_{12}^\mathsf{T} & R_2 \end{bmatrix}$, where $R_1 = \begin{bmatrix} 5 & 1 \\ 1 & 1 \end{bmatrix}$ and $R_2 = \begin{bmatrix} 1 & -1 \\ -1 & 5 \end{bmatrix}$, and $R_{12} \in \mathbb{R}^{2\times 2}$ can either be $\begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}$ or $\begin{bmatrix} -1 & 0.5 \\ 0.5 & -1 \end{bmatrix}$, which corresponds to R^a and R^b of $\mathcal{A} = \{R^a, R^b\}$, respectively.

In this case, $P_l = \begin{bmatrix} 0.40 & 0.45 \\ 0.45 & 0.93 \end{bmatrix}$ and $P^{\text{RO}} = P^\star = \begin{bmatrix} 0.56 & 0.40 \\ 0.40 & 0.95 \end{bmatrix}$. The results are visualized in Figure 4.7, where also $K^\star R^a (K^\star)^\mathsf{T}$ and $K^\star R^b (K^\star)^\mathsf{T}$ are plotted, with $K^\star$ being the best CLUE gain. The reason for having $P^\star \neq P_l, P^\star \succeq P_l$ is that there exists no K such that

$$P \succeq K R^a K^\mathsf{T} \wedge P \succeq K R^b K^\mathsf{T} \wedge P' \succeq (H^\mathsf{T} (R^a)^{-1} H)^{-1} \wedge P' \succeq (H^\mathsf{T} (R^b)^{-1} H)^{-1},$$

and $J(P) = J(P')$ hold simultaneously.

E5: Bounded Correlation-Coefficient

Assume that $N = 2$ and that the eigenvalues of R_{12} are constrained. Let $n_x = 2$ and

$$H_1 = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}, \qquad R_1 = 1, \qquad H_2 = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \qquad R_2 = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{bmatrix}.$$

In this case, $R_{12}^\mathsf{T} \in \mathbb{R}^3$ is assumed to be constrained by $R_{12}^\mathsf{T} R_1^{-1} R_{12} \leq \rho_{\max}^2 R_2$, or equivalently, $\|L_2^{-1} R_{12}^\mathsf{T} / \sqrt{R_1}\| \leq \rho_{\max}$, with $R_2 = L_2 L_2^\mathsf{T}$ and $\rho_{\max} \geq 0$. To have $R \succeq 0$, it is required that $\rho_{\max} \leq 1$. The quantity P^{RO} is computed for different values of $\rho_{\max} \in [0, 1]$. Also, P^{CI} and $P^{\text{KF}} = (H_1^\mathsf{T} R_1^{-1} H_1 + H_2^\mathsf{T} R_2^{-1} H_2)^{-1}$ are computed, where P^{KF} is equivalent to the covariance of the BLUE given $\|R_{12}\| = 0$. In Figure 4.8, the traces of P^{RO}, P^{CI}, and P^{KF} are plotted. As $\rho_{\max}$ increases from 0 to 1, $\mathsf{tr}(P^{\text{RO}})$ increases from $\mathsf{tr}(P^{\text{KF}})$ to $\mathsf{tr}(P^{\text{CI}})$.

Discussion

The results for E1–E4 are summarized in Table 4.1. It can be seen that each of CI, ICI, and LE yields the same answer for E1 and E2 since R_1 and R_2 are fixed throughout these cases. In E1 and E2, the `YALMIP` solution is equivalent to a best

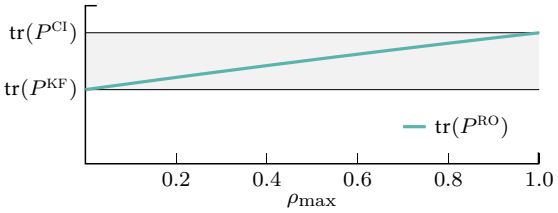


Figure 4.8. Results for E5, where $\|R_{12}\| \leq \rho_{max}$.

Table 4.1
SUMMARY OF FUSION EXAMPLES © 2022 IEEE

	P_u	P_l	P^{CI}	P^{ICI}	P^{LE}	P^{RO}	$P^{\star}$
E1	$1.43I$	I	$1.60I$	$1.18I$	I	I	I
E2a	$1.60I$	$1.60I$	$1.60I$	$1.18I$	I	$1.60I$	$1.60I$
E2b	-	-	$1.60I$	$1.18I$	I	$1.18I$	$1.18I$
E2c	$1.60I$	I	$1.60I$	$1.18I$	I	I	I
E3	$1.88I$	$1.31I$	$1.88I$	-	-	$1.76I$	-
E4	-	$\begin{bmatrix} 0.40 & 0.45 \\ 0.45 & 0.93 \end{bmatrix}$	-	-	-	$\begin{bmatrix} 0.56 & 0.40 \\ 0.40 & 0.95 \end{bmatrix}$	$\begin{bmatrix} 0.56 & 0.40 \\ 0.40 & 0.95 \end{bmatrix}$

black = CLUE, not best CLUE; **green** = best CLUE; **blue** = lower bound; **red** = not CLUE; **yellow** = CLUE, might be best CLUE. Quantities not computed are marked "-"

CLUE. The benefits of utilizing any extra structure encoded in $\mathcal{A}$ have also been demonstrated.

E3 is a counterexample of CI being a best CLUE under completely unknown cross-correlations when $N > 2$. It cannot be concluded if P^{RO} is equivalent to $P^{\star}$ since $P^{\star} \succ P_l$ is possible even if $P^{RO} = P^{\star}$, cf. Theorem 4.7. The upper bound is strict in this case.

In E5, neither H_1 nor H_2 is identity, and $R_{12}^{\mathsf{T}} R_1^{-1} R_{12} \leq \rho_{max}^2 R_2$. As ρ_{max} is varied from zero to its maximum value, $\mathsf{tr}(P^{RO})$ increases from that of the BLUE given $R_{12} = 0$ to that of CI. This result is quite specific but nevertheless verifies the generality of the RO based method.

The examples also demonstrate the generality of the CLUE framework and, in particular, the usability of $\mathcal{A}$: (i) it can be used to select an estimation method, e.g., ICI if (4.18) holds; (ii) it is the basis for deriving and solving general problems using RO; and (iii) it is used to compute lower and upper bounds on a best CLUE.

4.5.2 Decentralized Target Tracking Using CLUE

The purpose of the following DTT scenario is to evaluate some of the linear track fusion methods described in this chapter in a more realistic and dynamic context. The methods are evaluated based on MC simulations. For exactness, linear measurement models are assumed so that the SSM and the track fusion models are all linear.

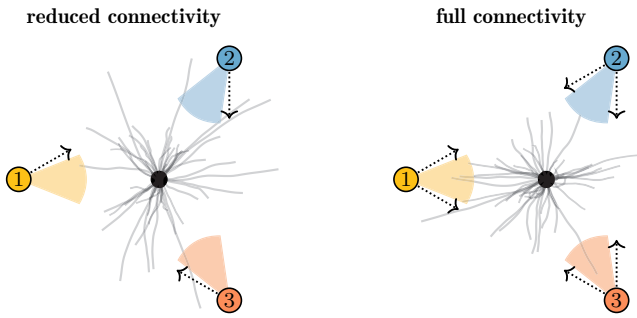


Figure 4.9. A decentralized single-target tracking scenario. The target is initially located in the black circle. Three agents (colored numbered circles) estimate the target and exchange local estimates. Two cases of communication network connectivity are considered: (i) *reduced connectivity*; and (ii) *full connectivity*. Gray lines represent target trajectories in different MC runs.

Simulation Specifications

A DTT scenario with $N = 3$ agents and one dynamic target in $d = 2$ spatial dimensions is now considered. The scenario is illustrated in Figure 4.9. The SSM in (3.1) is assumed with $x \in \mathbb{R}^4$ corresponding to a CVM. In each MC simulation, the target initial state x_0 is sampled from $\mathcal{N}(0, P_0)$, where P_0 is predetermined. The target then evolves according to the CVM in (3.5), i.e.,

$$x_{k+1} = F_k x_k + w_k, \qquad \boldsymbol{w}_k \sim \mathcal{N}(0, Q_k), \tag{4.48}$$

for a predetermined Q_k. As a consequence, the trajectory is unique in each MC simulation. The measurement model of the ith agent is given by (3.7) with

$$h_i(x_k) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} x_k, \qquad \boldsymbol{e}_{i,k} \sim \mathcal{N}(0, C_i), \tag{4.49}$$

where $\boldsymbol{e}_{i,k}$ is the measurement noise at time k and $C_i \in \mathbb{S}^2_{++}$ the measurement noise covariance of Agent i. A linear KF is used for state estimation, and linear unbiased estimation methods are used for track fusion. The Q_k used for sampling the target trajectory is the same Q_k used in all local filters. It is assumed that $H_1 = H_2 = H_3 = I$. Since the initial state, the process noise, and the measurement are all Gaussian distributed, the ith local estimate at time k is on the form

$$y_{i,k} = x_k + v_{i,k}, \qquad \boldsymbol{v}_{i,k} \sim \mathcal{N}(0, R_{i,k}). \tag{4.50}$$

Hence, in this case, the estimation error $\tilde{\boldsymbol{x}}_k$ is a zero-mean Gaussian distributed random variable. Relevant simulation parameters are summarized in Table 4.2.

The communication scheme of Rule 3.6 is assumed. The communication topology is according to the two cases illustrated in Figure 4.9:

- *Reduced connectivity*. Agent 1 only communicates to Agent 2, Agent 2 only communicates to Agent 3, and Agent 3 only communicates to Agent 1.

Table 4.2
PARAMETERS USED IN THE SIMULATIONS

Parameter	Comment
$d = 2$	spatial dimensionality
$n_x = 4$	state dimensionality
$T_s = 1$	sampling time [s]
$\sigma_w = 2$	standard deviation of process noise [$\mathrm{ms}^{-\frac{3}{2}}$]
$C_1 = \left[\begin{smallmatrix} 100 & 0 \\ 0 & 25 \end{smallmatrix}\right]$	measurement noise covariance of Agent 1 [m^2]
$C_2 = \left[\begin{smallmatrix} 44 & 32 \\ 32 & 81 \end{smallmatrix}\right]$	measurement noise covariance of Agent 2 [m^2]
$C_3 = \left[\begin{smallmatrix} 44 & -32 \\ -32 & 81 \end{smallmatrix}\right]$	measurement noise covariance of Agent 3 [m^2]
$n_k = 15$	number of time steps
$M = 10\,000$	number of MC runs

- *Full connectivity*. Each agent communicates to the other two agents.

The following methods are compared

- NKF: The naïve Kalman fuser as defined in (3.12) with $H_2 = I$.
- CI: Covariance intersection as defined in (3.14) with $H_2 = I$.
- ICI: Inverse covariance intersection as defined in (4.19).
- LE: The largest ellipsoid method as defined in (4.29).
- RO: The general CLUE approach using robust optimization as described in Section 4.3. RO assumes $\mathcal{A}$ is according to (4.45) with $\rho_{\max} = 0.5$.

The loss function $J(P) = \mathsf{tr}(P)$ is used.

Evaluation Measures

Let $(\hat{x}_k^i, P_k^i)$ be the estimate at time k of the ith MC simulation using some track fusion method. By construction, $P_k^i = P_k$ for all i. Hence, ANEES can be computed using (3.30) to evaluate estimator conservativeness. In addition, the COIN defined in (3.32) is used. If $\tilde{P}_k = \mathrm{E}(\tilde{\boldsymbol{x}}_k \tilde{\boldsymbol{x}}_k^{\mathsf{T}})$ is used, cf. (3.32), then COIN provides a *strict* measure of conservativeness. This is in contrast to ANEES, i.e., while an estimator can provide ANEES at or below 1 and still not be conservative, an estimator is conservative i.f.f. COIN is at or below 1. However, in this evaluation, $\tilde{P}_k$ is substituted for the sample covariance $\hat{P}_k$ given in (3.27). As a consequence, this strictness is only asymptotically true.

The RMSE and RMT defined in (3.26) and (3.28), respectively, are used for performance evaluation. RMT is more relevant than RMSE from a user perspective since this measure is something that is available to the end user in real-time. RMSE requires the ground truth to be known.

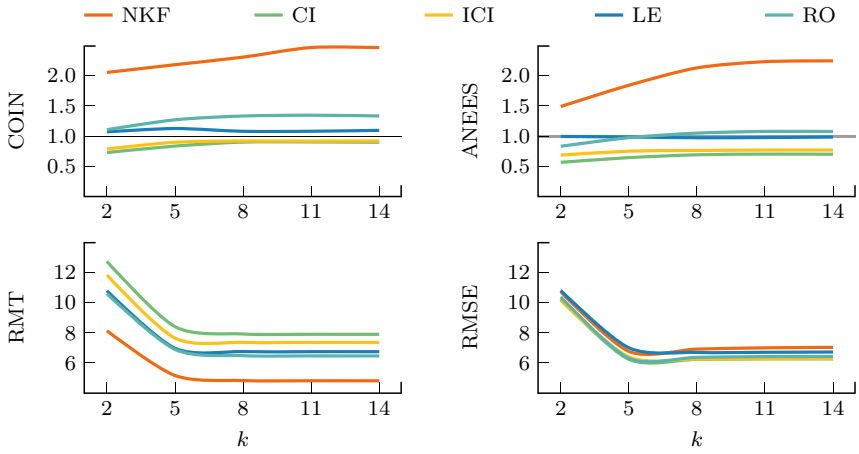


Figure 4.10. Results from the case with reduced connectivity. The gray area in the ANEES plot represents an ANEES 99.9% confidence interval.

Results

Figure 4.10 shows the results for the reduced connectivity case[12]. COIN, ANEES, RMT, and RMSE are computed at time points where track fusion is performed.

CI and ICI are conservative, as verified by the COIN. LE provides ANEES at 1, but the COIN is strictly larger than 1, hence LE is too optimistic. Also, RO provides slightly too optimistic estimates. NKF underestimates the uncertainty severely. It is notable that the spread in RMT among the methods is significantly larger than the spread in RMSE, which is quite small. All methods provide approximately the same RMSE, while the RMT differs significantly. Intuitively, RMT is the smallest for NKF and the largest for CI.

In the full connectivity case, see Figure 4.11, the exchanged information is utilized more. This is also indicated by the COIN and ANEES, as NKF diverges more quickly due to increased double counting of information. For the same reason, KF yields the highest RMSE over all k. CI and ICI are still conservative, while LE and RO still provide slightly too optimistic results.

4.6 Summary

The focus of this chapter has been the track fusion component of a DTT system. The CLUE was introduced as a framework for robust estimation under a partially known correlation structure. An optimal CLUE denoted best CLUE was proposed. Lower and upper bounds of the best CLUE have been derived. It was shown that a general CLUE can be implemented using RO which means that standard optimization software can be used in general linear estimation problems where

[12]The estimation results are only visualized for Agent 3, but the results for the other two agents are similar.

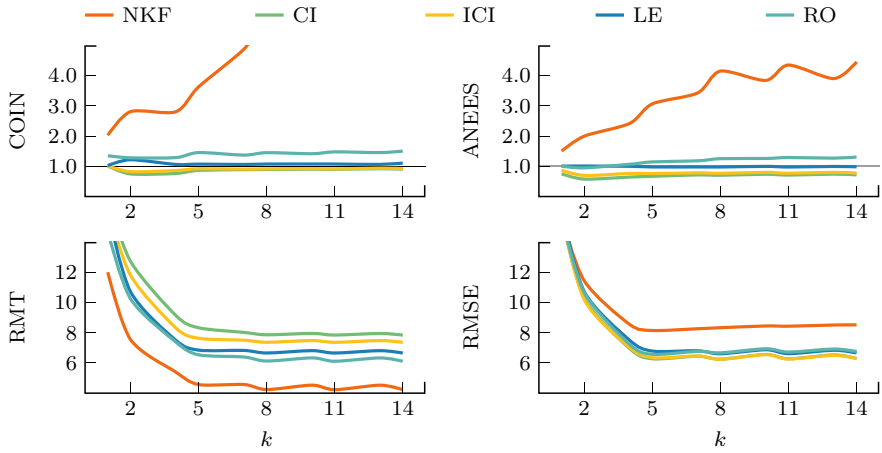


Figure 4.11. Results from the case with full connectivity. The gray area in the ANEES plot represents an ANEES 99.9% confidence interval.

there is uncertainty in the correlation structure. Moreover, it was demonstrated how many existing estimation methods fit into the CLUE framework. In particular, CI, ICI, and LE were shown to be special cases of a best CLUE.

The theory and methods were evaluated using several simple fusion examples and finally in a more realistic DTT context. In the DTT scenario it was demonstrated how different methods can be evaluated when designing the track fusion component in a DTT system. It should be stressed that a method is not automatically useless just because COIN is above 1. For instance, it might be relevant to use LE since it provides sufficiently reliable estimate w.r.t. the given DTT system criteria. Ultimately, it is up to the system engineers to compromise between different aspects, e.g., performance and conservativeness, to come up with a system design that is acceptable for the end user.

5

Dimension-Reduction for Efficient Communication Management

In Section 3.3, it was assumed that only the diagonal entries of a local covariance R_i are allowed to be communicated. This led to several DCA methods for preserving conservativeness. In this chapter, another strategy is proposed for efficient communication management. The approach is denoted *dimension-reduction* and basically transforms local estimates to a lower-dimensional subspace. Then the *dimension-reduced* (DR) estimates are communicated. The problem is formalized in Section 5.1, where also communication aspects are discussed. In Section 5.2, the DR estimates are derived using *principal component analysis* (PCA). In Section 5.3, several methods are developed for fusion optimal DR estimates. Fusion optimal DR estimates requires that the agent that is about to exchange the DR estimates have access to the estimate covariance of the receiving agent. In practice, this is an unreasonable assumption. A resolution to this problem is proposed in Section 5.4. In Section 5.5, a methodology for maintaining track-to-track association quality is proposed in a multitarget context with DR estimates.

5.1 Reducing Dimensionality of Exchanged Tracks

First, the DR problem is defined. Then, track fusion methods for fusing a local track with a received DR track are stated based on adapted versions of the *Bar-Shalom Campo* (BSC) fuser, the *Kalman fuser* (KF), *covariance intersection* (CI), *inverse covariance intersection* (ICI), and the *largest ellipsoid* (LE) method. Communication aspects are discussed at the end.

5.1.1 Dimension-Reduced Estimates

Assume (y_1, R_1) and (y_2, R_2) are according to

$$y_1 = x + v_1, \qquad R_1 = \mathsf{cov}(\boldsymbol{v}_1), \qquad y_2 = H_2 x + v_2, \qquad R_2 = \mathsf{cov}(\boldsymbol{v}_2), \tag{5.1}$$

with $R_1 \in \mathbb{S}^{n_x}_{++}$, $R_2 \in \mathbb{S}^{n_2}_{++}$, and $R_{12} = \mathsf{cov}(\boldsymbol{v}_1, \boldsymbol{v}_2)$. Assume w.l.o.g. that Agent 2 transmits its local estimate to Agent 1. If Agent 2 transmits (y_2, R_2) to Agent 1, then, as discussed in Section 3.3.1, $n_2(n_2+3)/2$ parameters must be exchanged. To reduce the communication load, Agent 2 can instead exchange (y_Ψ, R_Ψ) defined as

$$y_\Psi = \Psi y_2, \qquad R_\Psi = \Psi R_2 \Psi^\mathsf{T}, \tag{5.2}$$

where $\Psi \in \mathbb{R}^{m \times n_2}$, $m < n_2$, and $\mathsf{rank}(\Psi) = m$. The cross-covariance $R_{1\Psi} = \mathsf{cov}(\boldsymbol{v}_1, \Psi \boldsymbol{v}_2) = R_{12}\Psi^\mathsf{T}$. The operation in (5.2) is essentially a linear transformation of (y_2, R_2) from a n_2-dimensional space to a m-dimensional subspace. This transformation requires that also Ψ is exchanged. However, by coding (y_Ψ, R_Ψ, Ψ) as described Section 5.1.4, the number of exchanged parameters can be reduced to $(2mn_2 - m^2 + 3m)/2$.

5.1.2 Fusing Dimension-Reduced Estimates

Once (y_Ψ, R_Ψ, Ψ) is received, (y_1, R_2) and (y_Ψ, R_Ψ) can be fused by basically any fusion method applicable for estimates given according to (5.1). The general adaptions that need to be made are to replace y_2 by y_Ψ, R_2 by R_Ψ, and H_2 by ΨH_2. How Agent 1 computes $(\hat{x}, P)$ by fusing (y_1, R_1) and (y_Ψ, R_Ψ) is described for BSC, KF, CI, ICI, and LE. It is assumed that $H_2 \in \mathbb{R}^{n_2 \times n_x}$ and $\Psi \in \mathbb{R}^{m \times n_2}$ are known. Moreover, it is assumed that $\mathsf{rank}(H_2) = \min(n_2, n_x)$, $\mathsf{rank}(\Psi) = m$, and $R_1, R_2 \succ 0$. In all cases below, except LE, $\hat{x}$ is given as

$$\hat{x} = K_1 y_1 + K_\Psi y_\Psi = (I - K_\Psi \Psi H_2) y_1 + K_\Psi y_\Psi, \tag{5.3}$$

where $K_1 = I - K_\Psi \Psi H_2$ is a consequence of $\left[\begin{smallmatrix} K_1 & K_\Psi \end{smallmatrix}\right] \left[\begin{smallmatrix} I \\ \Psi H_2 \end{smallmatrix}\right] = I$. This means that $(\hat{x}, P)$ is fully specified by (K_Ψ, P).

Bar-Shalom Campo Fusion

An explicit derivation of the DR version of BSC is provided in Section 3.A. The estimate $(\hat{x}, P)$ is specified by

$$K_\Psi = (R_1 H_2^\mathsf{T} - R_{12})\Psi^\mathsf{T} S_\Psi^{-1}, \qquad P = R_1 - K_\Psi S_\Psi K_\Psi^\mathsf{T}, \tag{5.4}$$

where $S_\Psi = \Psi H_2 R_1 H_2^\mathsf{T} \Psi^\mathsf{T} + R_\Psi - \Psi H_2 R_{1\Psi} - R_{1\Psi}^\mathsf{T} H_2^\mathsf{T} \Psi^\mathsf{T}$. If $R = \left[\begin{smallmatrix} R_1 & R_{12} \\ R_{21} & R_2 \end{smallmatrix}\right] \succ 0$, then $S_\Psi \succ 0$. If $R = \left[\begin{smallmatrix} R_1 & R_{12} \\ R_{21} & R_2 \end{smallmatrix}\right] \succeq 0$, then $S_\Psi \succeq 0$ might be singular. In this case, the pseudoinverse S_Ψ^+ can be used instead of S_Ψ^{-1} to compute K_Ψ in (5.4). Note, to apply (5.4), $R_{1\Psi}$ must be known.

Kalman Fusion

The DR version of the KF is recovered by setting $R_{12} = 0$ in (5.4). In this case the estimate is specified by

$$K_\Psi = R_1 H_2^\mathsf{T} \Psi^\mathsf{T} S_\Psi^{-1}, \qquad P = R_1 - K_\Psi S_\Psi K_\Psi^\mathsf{T}, \tag{5.5}$$

where $S_\Psi = \Psi H_2 R_1 H_2^\mathsf{T} \Psi^\mathsf{T} + R_\Psi$.

Covariance Intersection

Consider CI defined in Algorithm 3.5 with $N = 2$ and $H_1 = I$. Assume y_2, R_2, and H_2 are replaced by y_Ψ, R_Ψ, and ΨH_2, respectively. It is then possible to define

$$K_\Psi = (1-\omega) P H_2^\mathsf{T} \Psi^\mathsf{T} R_\Psi^{-1}, \quad P = (\omega R_1^{-1} + (1-\omega) H_2^\mathsf{T} \Psi^\mathsf{T} R_\Psi^{-1} \Psi H_2)^{-1}, \tag{5.6}$$

where $\omega \in (0,1]$.

Inverse Covariance Intersection

Consider ICI defined in Algorithm 4.14. Assume y_2, R_2, and H_2 are replaced by y_Ψ, R_Ψ, and ΨH_2, respectively. It is then possible to define

$$K_\Psi = H_2^\mathsf{T} \Psi^\mathsf{T} R_\Psi^{-1} \Psi H_2 - (1-\omega) H_2^\mathsf{T} \Psi^\mathsf{T} \Psi H_2 \Gamma_\Psi^{-1} H_2^\mathsf{T} \Psi^\mathsf{T}, \tag{5.7a}$$

$$\begin{aligned} P = {} & R_1^{-1} + H_2^\mathsf{T} \Psi^\mathsf{T} R_\Psi^{-1} \Psi H_2 \\ & - \omega H_2^\mathsf{T} \Psi^\mathsf{T} \Gamma_1^{-1} \Psi H_2 - (1-\omega) H_2^\mathsf{T} \Psi^\mathsf{T} \Psi H_2 \Gamma_\Psi^{-1} H_2^\mathsf{T} \Psi^\mathsf{T}, \end{aligned} \tag{5.7b}$$

where $\Gamma_1 = \omega \Psi H_2 R_1 H_2^\mathsf{T} \Psi^\mathsf{T} + (1-\omega) R_\Psi$, $\Gamma_\Psi = \omega R_1 + (1-\omega) H_2^\mathsf{T} \Psi^\mathsf{T} R_\Psi \Psi H_2$, and $\omega \in (0,1]$.

The Largest Ellipsoid Method

The LE method can also be specified by (K_Ψ, P). However, in this case it is more convenient to directly adapt Algorithm 4.17. If y_2, R_2, and H_2 are replaced by y_Ψ, R_Ψ, and ΨH_2, respectively, then Algorithm 4.17 computes $(\hat{x}, P)$ according to

$$\hat{x} = P T^{-1} \iota, \qquad P = (T^{-1} \mathcal{I} T^{-\mathsf{T}})^{-1}, \tag{5.8}$$

where $T = T_2 T_1$,

$$\begin{aligned} \iota_1 &= R_1^{-1} y_1, & \mathcal{I}_1 &= R_1^{-1}, \\ \iota_\Psi &= H_2^\mathsf{T} \Psi^\mathsf{T} R_\Psi^{-1} y_\Psi, & \mathcal{I}_\Psi &= H_2^\mathsf{T} \Psi^\mathsf{T} R_\Psi^{-1} \Psi H_2, \\ \mathcal{I}_1 &= U_1 \Sigma_1 U_1^\mathsf{T}, & T_1 &= \Sigma_1^{-\frac{1}{2}} U_1^\mathsf{T}, \\ T_1 \mathcal{I}_\Psi T_1^\mathsf{T} &= U_\Psi \Sigma_\Psi U_\Psi^\mathsf{T}, & T_2 &= U_\Psi^\mathsf{T}, \\ \iota_1' &= T \iota_1, & \mathcal{I}_1' &= T \mathcal{I}_1 T^\mathsf{T} = I, \\ \iota_\Psi' &= T \iota_\Psi, & \mathcal{I}_\Psi' &= T \mathcal{I}_\Psi T^\mathsf{T}, \end{aligned}$$

and

$$([\iota]_i,[\mathcal{I}]_{ii}) = \begin{cases} ([\iota'_1]_i,[\mathcal{I}'_1]_{ii}), & \text{if } 1 \geq [\mathcal{I}'_\Psi]_{ii}, \\ ([\iota'_\Psi]_i,[\mathcal{I}'_\Psi]_{ii}), & \text{if } 1 < [\mathcal{I}'_\Psi]_{ii}, \end{cases} \tag{5.10}$$

for each $i = 1,\ldots,n_x$.

5.1.3 Change of Basis

To be able to utilize the communication resource as efficient as possible it is important that R_Ψ is diagonal. Otherwise, the efficient message coding described in the subsequent section is not valid. Luckily, since $R_\Psi \in \mathbb{S}^m_{++}$ for any feasible Ψ, it is always possible to transform R_Ψ by a simple rotation such that the result is diagonal. In particular, let $U\Sigma U^\mathsf{T}$ be the eigendecomposition of R_Ψ, where Σ is diagonal and U is orthogonal. Then $U^\mathsf{T} R_\Psi U = U^\mathsf{T} U \Sigma U^\mathsf{T} U = \Sigma$, where it was used that for orthogonal matrices $U^\mathsf{T} U = I$.

Transforming Ψ by T is equivalent to a *change of basis*. To see that a change of basis does not affect the fusion result, i.e., $(\hat{x},P)$ computed using any of the methods in Section 5.1.2, let $\Theta = T\Psi$ where $T^\mathsf{T} T = I$. Consider BSC, KF, CI, ICI, and LE defined in Section 5.1.2. By construction, $S_\Psi = \Psi S \Psi^\mathsf{T}$, where $S = H_2 R_1 H_2^\mathsf{T} + R_2 - H_2 R_{12} - R_{21} H_2^\mathsf{T}$ in case of BSC and $S = H_2 R_1 H_2^\mathsf{T} + R_2$ in case of KF. The terms involving Ψ are hence all on the form $\Psi^\mathsf{T}(\Psi C \Psi^\mathsf{T})^{-1} y_\Psi$, $\Psi^\mathsf{T}(\Psi C \Psi^\mathsf{T})^{-1}\Psi$, or $\Psi^\mathsf{T}\Psi$, where C is either S or R_2. By assumption

$$\begin{aligned} \Psi^\mathsf{T}(\Psi C \Psi^\mathsf{T})^{-1} y_\Psi &= \Psi^\mathsf{T} T^\mathsf{T} T (\Psi C \Psi^\mathsf{T})^{-1} T^\mathsf{T} T \Psi y_2 \\ &= \Psi^\mathsf{T} T^\mathsf{T} (T^{-\mathsf{T}} \Psi C \Psi^\mathsf{T} T^{-1})^{-1} T \Psi y_2 \\ &= \Theta^\mathsf{T}(\Theta C \Theta^\mathsf{T})^{-1}\Theta y_2, \end{aligned} \tag{5.11}$$

and similarly $\Psi^\mathsf{T}(\Psi C \Psi^\mathsf{T})^{-1}\Psi = \Theta^\mathsf{T}(\Theta C \Theta^\mathsf{T})^{-1}\Theta$. Moreover, $\Psi^\mathsf{T}\Psi = \Psi^\mathsf{T} T^\mathsf{T} T \Psi^\mathsf{T} = \Theta^\mathsf{T}\Theta$. Hence, the fusion result when using the methods defined in Section 5.1.2 is invariant to a transformation $\Theta = T\Psi$, where T is such that $T^\mathsf{T} T = I$.

Let $\mathsf{rowspan}(A)$ denote the row span of A. A more general result is stated in Proposition 5.1 from which (5.11) and $\Psi^\mathsf{T} R_\Psi^{-1}\Psi = \Theta^\mathsf{T}(\Theta R_2 \Theta^\mathsf{T})^{-1}\Theta$ follow immediately. This result is applicable for BSC, KF, CI, and LE, and will, in particular, be used in Section 5.3 when deriving fusion optimal Ψ.

Proposition 5.1. *Let $S \in \mathbb{S}^n_{++}$ and $A,B \in \mathbb{R}^{m\times n}$, where $m \leq n$. If $\mathsf{rowspan}(A) = \mathsf{rowspan}(B)$, then*

$$A^\mathsf{T}(ASA^\mathsf{T})^{-1}A = B^\mathsf{T}(BSB^\mathsf{T})^{-1}B. \tag{5.12}$$

Proof: Since $\mathsf{rowspan}(A) = \mathsf{rowspan}(B)$ there exists an invertible matrix T such that $B = TA$. Hence

$$\begin{aligned} A^\mathsf{T}(ASA^\mathsf{T})^{-1}A &= A^\mathsf{T} T^\mathsf{T} T^{-\mathsf{T}}(ASA^\mathsf{T})^{-1} T^{-1} TA = A^\mathsf{T} T^\mathsf{T}(TASA^\mathsf{T}T^\mathsf{T})^{-1} TA \\ &= B^\mathsf{T}(BSB^\mathsf{T})^{-1}B. \end{aligned}$$

□

5.1.4 Message Coding

Let $y_\Psi \in \mathbb{R}^m$ and assume that $R_\Psi \in \mathbb{S}^m_{++}$ is diagonal. Suppose now that Agent 2 is transmitting (y_Ψ, R_Ψ) to Agent 1 who fuses (y_Ψ, R_Ψ) with its local estimate. To be able to use (y_Ψ, R_Ψ) Agent 1 also requires Ψ, but to simply transmit (y_Ψ, R_Ψ, Ψ) is costly. Instead, it is more efficient to transmit (y_Ψ, Φ), where

$$\Phi = \begin{bmatrix} \phi_1 \\ \vdots \\ \phi_m \end{bmatrix} = \begin{bmatrix} r_1\psi_1 \\ \vdots \\ r_m\psi_m \end{bmatrix} = R_\Psi \Psi, \tag{5.13}$$

with $r_i = [R_\Psi]_{ii}$, and where ϕ_i and ψ_i represent the ith row of Φ and Ψ, respectively. When Agent 1 receives (y_Ψ, Φ), R_Ψ and Ψ are computed as

$$R_\Psi = \mathsf{diag}\left(\|\phi_1\|, \ldots, \|\phi_m\|\right), \qquad \Psi = R_\Psi^{-1}\Phi. \tag{5.14}$$

The number of exchanged parameters can be further reduced by exploiting the fact that $\psi_i\psi_j^\mathsf{T} = \delta_{ij}$, and hence $\phi_i\phi_j^\mathsf{T} = 0$ if $i \neq j$. For $\Phi \in \mathbb{R}^{m\times n_2}$ the agent needs to transmit m components of ϕ_1, but only $m-i+1$ components of ϕ_i. The components not transmitted are given by utilizing $\phi_i\phi_j^\mathsf{T} = 0$ for $i \neq j$ sequentially. For example, assume $m = 3$ and let $\phi_{i,j} = [\phi_i]_j$. Assume that Agent 1 does not have access to $\phi_{2,1}$, $\phi_{3,1}$ and $\phi_{3,2}$. The missing components are given by first solving for $\phi_{2,1}$ in $\phi_1\phi_2^\mathsf{T} = 0$, which is equivalent to solving $A_2\phi_{2,1} = b_2$, where

$$A_2 = \phi_{1,1}, \qquad b_2 = -\sum_{j=2}^{n_2} \phi_{1,j}\phi_{2,j}.$$

Then $\phi_{3,1}$ and $\phi_{3,2}$ are computed by solving $A_3 \begin{bmatrix} \phi_{3,1} \\ \phi_{3,2} \end{bmatrix} = b_3$, where

$$A_3 = \begin{bmatrix} \phi_{1,1} & \phi_{1,2} \\ \phi_{2,1} & \phi_{2,2} \end{bmatrix}, \qquad b_3 = -\begin{bmatrix} \sum_{j=3}^{n_2} \phi_{1,j}\phi_{3,j} \\ \sum_{j=3}^{n_2} \phi_{2,j}\phi_{3,j} \end{bmatrix}.$$

This sequential approach generalizes to arbitrary i. Let superscripts M_i and K_i, e.g., a^{M_i} and a^{K_i}, denote a vector that contains the components of a vector, e.g., a, corresponding to the missing and available components of ϕ_i, respectively. In particular, the union and intersection of the components of $\phi_i^{K_i}$ and $\phi_i^{M_i}$ correspond to ϕ_i and the empty set, respectively. For arbitrary $i \in \{2, \ldots, n_2\}$, the missing components $\phi_i^{M_i}$ are computed by solving

$$A_i(\phi_i^{M_i})^\mathsf{T} = b_i, \qquad A_i = \begin{bmatrix} \phi_1^{M_i} \\ \vdots \\ \phi_{i-1}^{M_i} \end{bmatrix}, \qquad b_i = -\begin{bmatrix} \phi_1^{K_i}(\phi_i^{K_i})^\mathsf{T} \\ \vdots \\ \phi_{i-1}^{K_i}(\phi_i^{K_i})^\mathsf{T} \end{bmatrix}. \tag{5.15}$$

Let Φ^K denote the parts of Φ that are actually transmitted using the proposed message encoding, i.e.,

$$\Phi^K = \begin{bmatrix} \phi_1 & \phi_2^{K_2} & \ldots & \phi_m^{K_m} \end{bmatrix}. \tag{5.16}$$

Care must be taken to ensure that $\det(A_i) \neq 0$, since otherwise the system of equations in (5.15) is not solvable. The condition $\det(A_i) \neq 0$ has to be checked by the transmitting agent. This means that it is not possible by default to always skip transmitting the first components of ϕ_i since this could potentially imply that $\det(A_i) = 0$. Hence, alongside the transmission of the information contained in ϕ_i, i.e., $\phi_i^{K_i}$, the agent must also transmit information about which components are not transmitted, i.e., which elements of ϕ_i are included in $\phi_i^{M_i}$. This requires the transmission of a few extra parameters. For example, if $m = 3$, then $1 + 2 = 3$ extra parameters must be transmitted that indicate which components of ϕ_1 and ϕ_2 are not transmitted. For general m, a number of

$$n_{\text{ex}} = \sum_{i=1}^{m} (i-1) = \frac{m(m-1)}{2}, \tag{5.17}$$

extra parameters must be included in the message. However, as described below, to enumerate the excluded components of the ϕ_i, where $i = 1, \ldots, m$, only requires a few extra bits which is negligible compared to the remaining parameters to be exchanged.

In summary, the transmitted message is given by $(y_\Psi, \Phi^K, \mathcal{J})$, where $\mathcal{J}$ is an n_{ex}-dimensional vector containing the indices of the missing components of ϕ_i, where $i = 1, \ldots, m$.[1]

5.1.5 Communication Reduction

The total number of components required to be transmitted when exchanging (y_Ψ, Φ^K) is given by

$$n_{\text{dr}} = m + \frac{n_2(n_2+1)}{2} - \frac{(n_2-m)(n_2-m+1)}{2} = \frac{2mn_2 - m^2 + 3m}{2}, \tag{5.18}$$

where m is due to y_Ψ and the remains are due to Φ. The ratio between n_{dr} and the total number of components $n_{\text{full}} = n_2(n_2+3)/2$ of a full estimate (y_2, R_2) is

$$\frac{n_{\text{dr}}}{n_{\text{full}}} = \frac{2mn_2 - m^2 + 3m}{n_2(n_2+3)} = \frac{m}{n_2}\left(2 - \frac{m+3}{n_2+3}\right). \tag{5.19}$$

The communication reduction as a function of n_2 when using DR is illustrated in Figure 5.1 for different values of m. For example, if $n_2 = 9$, then the communication savings are approximately 81% for $m = 1$ and 50% for $m = 3$. If $n_2 = 15$, then the communication savings are approximately 88% for $m = 1$ and 67% for $m = 3$. As a comparison, $n_{\text{dca}} = 2n_2$ is included. If $m = 2$, then $n_{\text{dr}} = 2n_2 + 1 = n_{\text{dca}} + 1$[2].

To continue the discussion about the n_{ex} extra parameters to be exchanged, assume that the size of each parameter in (y_Ψ, Φ^K) is 32 bits. Moreover, assume that the size of each parameter in $\mathcal{J}$ is 4 bits, which is valid for $n_2 \leq 16$. Then

[1] MATLAB® code for message coding is available at `https://github.com/robinforsling/dtt`.

[2] Hence, when comparing the performance of the two communication reduction techniques, it is reasonable to compare DCA with DR in the case of $m = 2$.

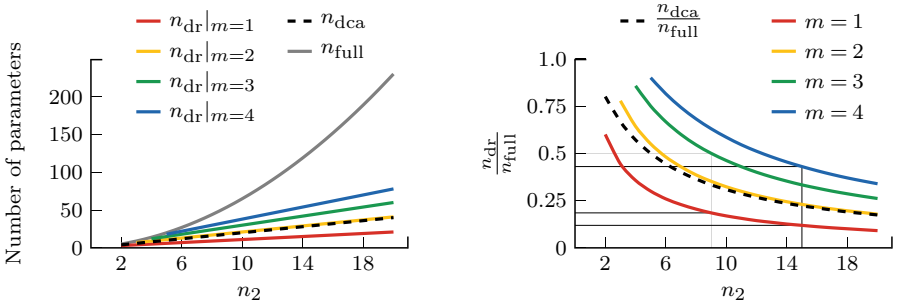


Figure 5.1. Illustration of the communication reduction as a function of n_2 when using DR for different values of m. DCA is used as a comparison.

(y_Ψ, Φ^K) and $\mathcal{J}$ consist of $32n_{\mathrm{dr}}$ bits and $4n_{\mathrm{ex}}$ bits, respectively. In Table 5.1, the ratio

$$\frac{4n_{\mathrm{ex}}}{32n_{\mathrm{dr}}} = \frac{m-1}{8(2n_2 - m + 3)}, \tag{5.20}$$

is illustrated for a number of values of m and n_2. The amount of extra bits required to transmit $\mathcal{J}$ is marginal in this configuration. The conclusion is that the bit size of (y_Ψ, Φ^K) and the bit size of $(y_\Psi, \Phi^K, \mathcal{J})$ are approximately equal.

Table 5.1
PERCENTAGE OF EXTRA BITS REQUIRED FOR $\mathcal{J}$

m	n_2	$\frac{n_{\mathrm{ex}}}{8n_{\mathrm{dr}}}$
2	4	1.39%
2	6	0.96%
3	6	2.08%

m	n_2	$\frac{n_{\mathrm{ex}}}{8n_{\mathrm{dr}}}$
3	9	1.39%
5	9	3.12%
7	9	5.36%

m	n_2	$\frac{n_{\mathrm{ex}}}{8n_{\mathrm{dr}}}$
3	12	1.04%
6	12	2.98%
9	15	4.17%

5.2 Dimension-Reduction Using Principal Component Analysis

The methods described in Section 5.1.2 are valid for any $\Psi \in \mathbb{R}^{m \times n_2}$ with $m \leq n_2$. However, it is not specified how Ψ is computed. Popular methods for DR in general are PCA [137] and the closely related *Karhunen-Loève transform* [12, 175] which both utilize the eigendecomposition. It should be pointed out that there are many other closely related approaches for DR, and these concepts are sometimes used interchangeably, e.g., low-rank strategies [43, 82, 85, 154]. In this section, PCA is deployed for DR. The fusion methods in Section 5.1.2 are evaluated in a DTT scenario where DR estimates are exchanged and Ψ derived using PCA.

Algorithm 5.2: The Principal Component Optimization Method

Input: $R_2 \in \mathbb{S}_{++}^{n_2}$ and $m \leq n_2$

1. Let $R_2 = \sum_{i=1}^{n_2} \lambda_i u_i u_i^{\mathsf{T}}$, where $\lambda_1 \geq \cdots \geq \lambda_{n_2}$ and $u_i^{\mathsf{T}} u_j = \delta_{ij}$.
2. Define $\Psi = \mathsf{col}(u_{n_2-m+1}^{\mathsf{T}}, \ldots, u_{n_2}^{\mathsf{T}})$.

Output: Ψ

5.2.1 The Principal Component Optimization Method

Assume $\Psi \in \mathbb{R}^{m \times n_2}$, where $\mathsf{rank}(\Psi) = m$ and $m < n_2$ is arbitrary but fixed. An intuitive approach to select Ψ is by minimizing some loss function $J(R_\Psi) = J(\Psi R_2 \Psi^{\mathsf{T}})$. Let $J(R_\Psi) = \mathsf{tr}(R_\Psi)$ which means that the MSE of R_Ψ is to be minimized. To avoid that Ψ becomes infinitesimal small, some additional constraints on Ψ are required. Consider the quantity $\Psi^{\mathsf{T}} R_\Psi^{-1} \Psi$. In Proposition 5.1, it is shown that this quantity is invariant under an invertible transformation T of Ψ. Since Ψ is full row rank there exists an invertible matrix T such that $\Theta = T\Psi$, $\mathsf{rowspan}(\Psi) = \mathsf{rowspan}(\Theta)$, and $\Theta\Theta^{\mathsf{T}} = I$ [67]. Therefore, assume w.l.o.g. that $\Psi\Psi^{\mathsf{T}} = I$ and hence that Ψ is computed as the solution to

$$\begin{aligned} \underset{\Psi}{\text{minimize}} \quad & \mathsf{tr}(\Psi R_2 \Psi^{\mathsf{T}}) \\ \text{subject to} \quad & \Psi\Psi^{\mathsf{T}} = I. \end{aligned} \tag{5.21}$$

The problem in (5.21) is a PCA problem [79] and the solution is given by the eigenvectors corresponding to the m largest eigenvalues of R_2^{-1} [13]. These eigenvectors are principal components of R_2^{-1}.

The *principal component optimization* (PCO) method is defined in Algorithm 5.2 and is based on the optimization formulation in (5.21). With Ψ computed as in Algorithm 5.2, $\Psi\Psi^{\mathsf{T}} = I$ and $R_\Psi = \Psi R_2 \Psi^{\mathsf{T}} = \mathsf{diag}(\lambda_{n_2-m+1}, \ldots, \lambda_{n_2})$, where $\lambda_i = \lambda_i(R_2)$. Hence, no additional change of basis is required to use the message coding described in Section 5.1.4. An example of PCO is provided in Example 5.3.

Example 5.3: The Principal Component Optimization Method

Let $n_x = n_2 = 2$ and $H_2 = I$. Assume that $R_1 = \mathsf{diag}(\sigma^2, 1)$, $R_2 = \mathsf{diag}(1, \sigma^2)$, and $R_{12} = 0$. Under the assumption $\sigma \geq 1$, Algorithm 5.2 yields $\Psi = \begin{bmatrix} 1 & 0 \end{bmatrix}$. Since $R_{12} = 0$, KF can be used. Let P and P_{full} be given by

$$P = \left(R_1^{-1} + \Psi^{\mathsf{T}} (\Psi R_2 \Psi^{\mathsf{T}})^{-1} \Psi \right)^{-1}, \qquad P_{\text{full}} = \left(R_1^{-1} + R_2^{-1} \right)^{-1},$$

which correspond to the DR case and the case where the full estimate is exchanged, respectively. By construction

$$\mathsf{tr}(P) = \frac{2\sigma^2 + 1}{\sigma^2 + 1}, \qquad \mathsf{tr}(P_{\text{full}}) = \frac{2\sigma^2}{\sigma^2 + 1}, \qquad \frac{\mathsf{tr}(P)}{\mathsf{tr}(P_{\text{full}})} = 1 + \frac{1}{2\sigma^2}.$$

The ratio $\mathsf{tr}(P)/\mathsf{tr}(P_{\text{full}})$ is plotted for different values of σ in Figure 5.2. As σ increases $\mathsf{tr}(P)/\mathsf{tr}(P_{\text{full}}) \to 1$. Hence, there is essentially no performance loss in

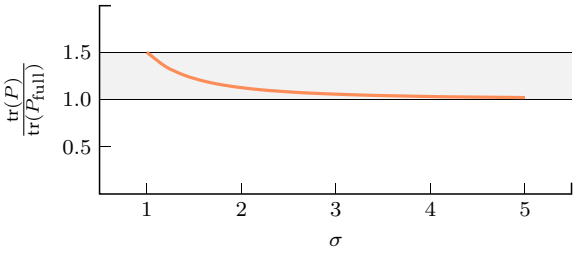


Figure 5.2. Results of Example 5.3. As σ increases, $\mathrm{tr}(P)/\mathrm{tr}(P_{\mathrm{full}})$ tends to 1.

this case when exchanging a DR estimate with Ψ derived using the PCO method compared to exchanging the full estimate.

5.2.2 Numerical Evaluation

The PCO method is now demonstrated using a DTT scenario identical to that used in Section 4.5.2 with the reduced network connectivity.

Simulation Specifications

The decentralized single-target tracking scenario is depicted in Figure 5.3. Assume exactly the same process and measurement models as in Section 4.5.2. The target is sampled in an identical way. Relevant simulation parameters are summarized in Table 5.2. The communication scheme of Rule 3.6 is assumed, and the communication topology is according to Figure 5.3. The PCO method in Algorithm 5.2 is used to derive the DR estimates.

The following methods are compared:

- NKF: The naïve Kalman fuser as defined in (5.5) with $H_2 = I$.

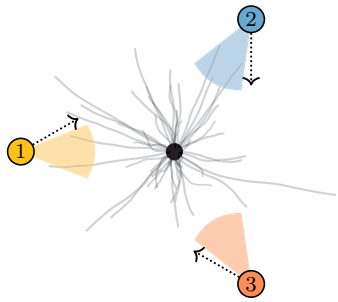


Figure 5.3. Decentralized single-target tracking scenario. The target is initially located at the black circle. Three agents (colored numbered circles) estimate the target and exchange local estimates. Gray lines represent target trajectories in different MC runs.

Table 5.2
PARAMETERS USED IN THE SIMULATIONS

Parameter	Comment
$m = 2$	dimensionality of the DR estimate y_Ψ
$d = 2$	spatial dimensionality
$n_x = 4$	state dimensionality
$T_s = 1$	sampling time [s]
$\sigma_w = 2$	standard deviation of process noise [$\mathrm{ms}^{-\frac{3}{2}}$]
$C_1 = \left[\begin{smallmatrix} 100 & 0 \\ 0 & 25 \end{smallmatrix}\right]$	measurement noise covariance of Agent 1 [m^2]
$C_2 = \left[\begin{smallmatrix} 44 & 32 \\ 32 & 81 \end{smallmatrix}\right]$	measurement noise covariance of Agent 2 [m^2]
$C_3 = \left[\begin{smallmatrix} 44 & -32 \\ -32 & 81 \end{smallmatrix}\right]$	measurement noise covariance of Agent 3 [m^2]
$n_k = 15$	number of time steps
$M = 10\,000$	number of MC runs

- CI: Covariance intersection as defined in (5.6) with $H_2 = I$.
- ICI: Inverse covariance intersection as defined in (5.7) with $H_2 = I$.
- LE: The largest ellipsoid method as defined in (5.8) with $H_2 = I$.

Evaluation Measures

COIN, ANEES, RMT, and RMSE defined in Section 3.2 are used for evaluation. These metrics are computed in the same way as in Section 4.5.2. In addition, the applicability of the PCO method is evaluated by computing the *RMT ratio* (RMTR) and the *RMSE ratio* (RMSER) defined as:

$$\mathrm{RMTR} = \frac{\text{RMT using dimension-reduced estimates}}{\text{RMT using full estimates}}, \tag{5.22a}$$

$$\mathrm{RMSER} = \frac{\text{RMSE using dimension-reduced estimates}}{\text{RMSE using full estimates}}. \tag{5.22b}$$

By construction, RMTR is larger than or equal to 1.

Results

Figure 5.4 illustrates the results for Agent 3 only[3]. COIN, ANEES, RMT, RMSE, RMTR, and RMSER are computed at time instants where track fusion is performed.

CI is the only method that is conservative w.r.t. both COIN and ANEES. Both ICI and LE are however conservative w.r.t. ANEES. NKF is never conservative. From the RMTR plot, it is seen that NKF exhibits the largest performance loss w.r.t. RMT when exchanging DR estimates compared to full estimates. Meanwhile, in terms of RMSE, CI shows the largest performance loss when exchanging DR estimates compared to full estimates. The spread in RMT is smaller than the spread in RMSE. CI yields the largest RMSE and RMT.

[3]The estimation results for the other two agents are similar.

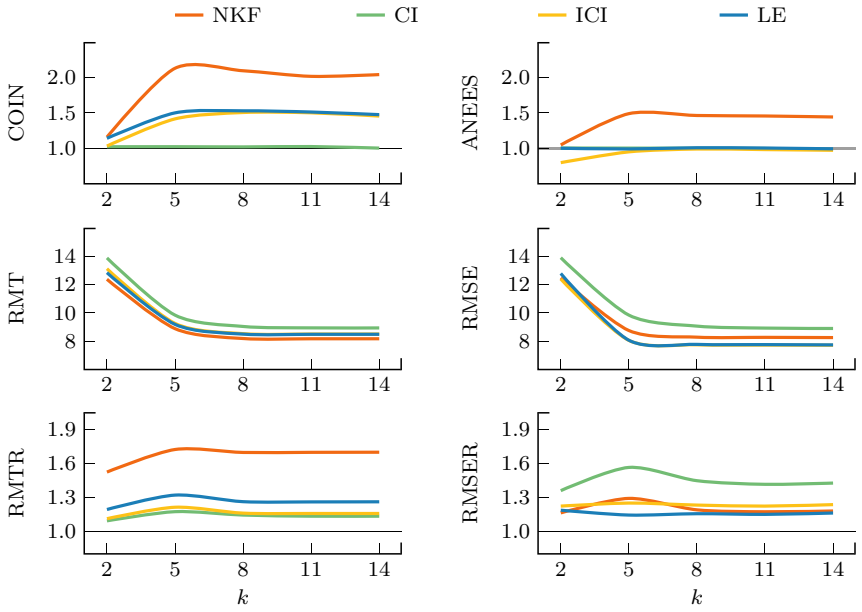


Figure 5.4. Results from the PCO method evaluation. The gray area in the ANEES plot represents an ANEES 99.9% confidence interval.

5.3 Dimension-Reduction for Optimal Track Fusion

The PCO method of the previous section provides no guarantees of optimality w.r.t. fusion performance. In fact, there are situations where PCO produces the worst possible DR estimate w.r.t. the MSE of the fused result. In this section, a framework is proposed for deriving fusion optimal DR estimates.

5.3.1 Motivating Example

The PCO method derives Ψ from the eigenvectors corresponding to the smallest eigenvalues $\lambda(R_2)$. To illustrate that this approach is in general optimal from a track fusion perspective, consider

$$R_1 = \begin{bmatrix} 3.2 & 1.2 \\ 1.2 & 1.8 \end{bmatrix}, \qquad R_2 = \begin{bmatrix} 4 & 0 \\ 0 & 1 \end{bmatrix}, \qquad \Psi = \begin{bmatrix} \cos\alpha & \sin\alpha \end{bmatrix}, \tag{5.23}$$

where $\alpha \in [0, 180]^\circ$, $H_2 = I$, and $R_{12} = 0$. For a certain Ψ, KF in (5.5) provides optimal fusion of (y_1, R_2) and (y_Ψ, R_Ψ). Let $P(\alpha) = (R_1^{-1} + \Psi^\mathsf{T}(\Psi R_2 \Psi^\mathsf{T})^{-1}\Psi)^{-1}$ with $\Psi(\alpha)$ defined in (5.23). Figure 5.5 illustrates $\mathsf{tr}(P)$ for $\alpha \in [0, 180]^\circ$. Trace of R_1 and $P_{\text{full}} = (R_1^{-1} + R_2^{-1})^{-1}$ are also included. The value $\alpha^\star$ refers to the minimizer of $\mathsf{tr}(P)$, and $\alpha = 90^\circ$ corresponds to Ψ computed by PCO. The ratio between the maximum $\max(\mathsf{tr}(P))$ and minimum value $\min(\mathsf{tr}(P))$ of $\mathsf{tr}(P)$ is approximately 1.73.

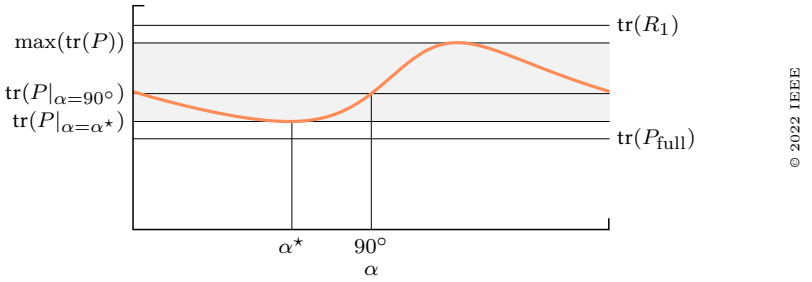


Figure 5.5. Example of when the PCO method does not yield optimal fusion performance. The value $\alpha = 90^\circ$ corresponds to Ψ computed by the PCO method. The optimal α w.r.t. $\mathrm{tr}(P)$ is given by $\alpha^\star$.

5.3.2 The Generalized Eigenvalue Optimization Method

The previous example illustrates that the PCO method in general does not compute the optimal Ψ w.r.t. the track fusion performance in Agent 1. In this section, the objective is to derive Ψ such that optimal track fusion performance is obtained. The fusion optimal Ψ depends on which method is used for fusion. The derived methodology is based upon the BSC fuser defined in (5.4). From this, methods tailored to KF, CI, and LE are derived for computing Ψ.

Remark 5.4. No method tailored to ICI is derived due to the complicated expressions of ICI defined in (5.7).

Consider P given in (5.4), i.e.,

$$P = R_1 - K_\Psi \Psi S \Psi^{\mathsf{T}} K_\Psi^{\mathsf{T}}, \tag{5.24}$$

where

$$K_\Psi = (R_1 H_2^{\mathsf{T}} - R_{12})\Psi^{\mathsf{T}}(\Psi S \Psi^{\mathsf{T}})^{-1}, \quad S = H_2 R_1 H_2^{\mathsf{T}} + R_2 - H_2 R_{12} - R_{12}^{\mathsf{T}} H_2^{\mathsf{T}}.$$

In this context an optimal $\Psi \in \mathbb{R}^{m \times n_2}$, denoted $\Psi^\star$, solves the problem

$$\underset{\Psi}{\text{minimize}} \quad \mathrm{tr}(P), \tag{5.25}$$

where P is according to (5.24). The formulation in (5.25) means that $\Psi^\star$ is MSE optimal. Similar formulations are considered in, e.g., [187, 190]. The solution to (5.25) is now derived.

Assume $S \succ 0$. The degenerate case where $S \succeq 0$ might be singular is discussed later on. Let $\Delta = R_1 H_2^{\mathsf{T}} - R_{12}$ such that P in (5.24) is given by

$$P = R_1 - \Delta \Psi^{\mathsf{T}} (\Psi S \Psi^{\mathsf{T}})^{-1} \Psi \Delta^{\mathsf{T}}. \tag{5.26}$$

Since R_1 is constant, minimization of $\mathrm{tr}(P)$ is equivalent to

$$\underset{\Psi}{\text{maximize}} \quad \mathrm{tr}\left(\Delta \Psi^{\mathsf{T}} (\Psi S \Psi^{\mathsf{T}})^{-1} \Psi \Delta^{\mathsf{T}}\right). \tag{5.27}$$

Using the cyclic property of trace, this problem is equivalently expressed as

$$\underset{\Psi}{\text{maximize}} \quad \text{tr}\left((\Psi S \Psi^{\mathsf{T}})^{-1} \Psi Q \Psi^{\mathsf{T}}\right), \tag{5.28}$$

where $Q = \Delta^{\mathsf{T}}\Delta$. Since $S \in \mathbb{S}^{n_2}_{++}$ and $\mathsf{rank}(\Psi) = m$, it follows that $\Psi S \Psi^{\mathsf{T}} \in \mathbb{S}^{m}_{++}$. This implies that there exists an invertible matrix T such that $T\Psi S \Psi^{\mathsf{T}} T^{\mathsf{T}} = I$. Hence, w.l.o.g. it can be assumed that $\Psi S \Psi^{\mathsf{T}} = I$. Moreover, $S \in \mathbb{S}^{n_2}_{++}$ implies that $S = LL^{\mathsf{T}}$, where L is invertible. Define $\Phi = \Psi L$. Then the following problem is equivalent to (5.28)

$$\begin{aligned} \underset{\Phi}{\text{maximize}} \quad & \text{tr}(\Phi A \Phi^{\mathsf{T}}) \\ \text{subject to} \quad & \Phi\Phi^{\mathsf{T}} = I, \end{aligned} \tag{5.29}$$

where $A = L^{-1}QL^{-\mathsf{T}} \in \mathbb{S}^{m}_{+}$. The solution to (5.29) is given by the eigenvectors corresponding to the largest eigenvalues of A [13]. By transforming back using $\Phi = \Psi L$, the solution to the original problem in (5.28) is obtained. The solution includes a GEVP and is summarized in the following theorem.

Theorem 5.5. *Assume $Q \in \mathbb{S}^{n_2}_{+}$ and $S \in \mathbb{S}^{n_2}_{++}$. Let $\Psi \in \mathbb{R}^{m \times n_2}$, where $m \leq n_2$ and $\mathsf{rank}(\Psi) = m$. The solution to*

$$\underset{\Psi}{\text{maximize}} \quad \text{tr}\left((\Psi S \Psi^{\mathsf{T}})^{-1} \Psi^{\mathsf{T}} Q \Psi^{\mathsf{T}}\right), \tag{5.30}$$

is given by $\Psi^{\star} = \mathsf{col}(u_1^{\mathsf{T}}, \ldots, u_m^{\mathsf{T}})$, where u_i is a generalized eigenvector associated with $\lambda_i(Q, S)$, and $\lambda_1 \geq \cdots \geq \lambda_{n_2}$.

Change of Basis

Consider $\Phi = \mathsf{col}(u_1^{\mathsf{T}}, \ldots, u_m^{\mathsf{T}})$, where $u_1, \ldots, u_m$ are derived as in Theorem 5.5. The rows of $\Phi \in \mathbb{R}^{m \times n_2}$ span an m-dimensional subspace $\mathcal{V} = \mathsf{rowspan}(\Psi^{\star}) \subseteq \mathbb{R}^{n_2}$. In this sense $\Psi^{\star}$ is a transformation

$$\Phi \colon \mathbb{R}^{n_2} \to \mathcal{V}. \tag{5.31}$$

For any two solutions u_i and u_j to $Qu = \lambda S u$, it is true that $u_i^{\mathsf{T}} S u_j = 0$ while in general $u_i^{\mathsf{T}} u_j \neq 0$, for $i \neq j$ [136]. Hence, Φ is *not* an orthogonal basis for $\mathcal{V}$. Proposition 5.1 ensures that if T is invertible, then $(\Psi^{\star})^{\mathsf{T}}(\Psi^{\star} R_2 (\Psi^{\star})^{\mathsf{T}})^{-1}\Psi^{\star}$ does not change if $\Psi^{\star}$ is substituted for $T\Psi^{\star}$. Moreover, since $\mathsf{rowspan}(\Psi^{\star}) = \mathsf{rowspan}(T\Psi^{\star})$ [67], this can be exploited to derive $\Psi^{\star}$ such that $\Psi^{\star} R_2 (\Psi^{\star})^{\mathsf{T}}$ is diagonal[4].

A *Gram-Schmidt procedure* can be expressed as $\Omega^{\mathsf{T}} = \Phi^{\mathsf{T}} T$, where Ω has orthonormal rows and T is invertible [67]. Thereby, Proposition 5.1 applies. Let $\Omega R_2 \Omega = U \Sigma U^{\mathsf{T}}$ be an eigendecomposition, where $\Sigma \in \mathbb{S}^{m}_{++}$ is diagonal. By construction $U^{\mathsf{T}} \Omega R_2 \Omega^{\mathsf{T}} U$ is diagonal. Hence, with $\Psi^{\star} = U^{\mathsf{T}}\Omega$, the covariance

[4] Ensuring that $\Psi^{\star} R_2 (\Psi^{\star})^{\mathsf{T}}$ is diagonal is important from a communication perspective as described in Section 5.1.4.

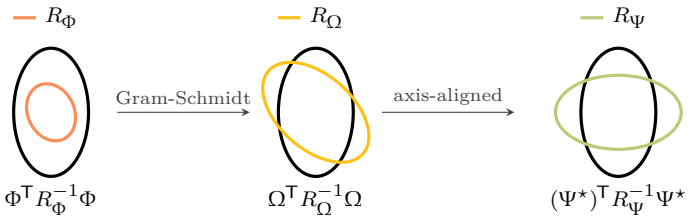


Figure 5.6. Change of basis. The information $\Psi^\mathsf{T}(\Psi R_2 \Psi^\mathsf{T})^{-1}\Psi$ is invariant to change of basis. The quantities $\Phi^\mathsf{T} R_\Phi^{-1}\Phi^\mathsf{T}$, $\Omega^\mathsf{T} R_\Omega^{-1}\Omega^\mathsf{T}$, and $(\Psi^\star)^\mathsf{T} R_\Psi^{-1}\Psi^\star$ are projected onto $\mathcal{V}$ using $\Psi^\star$.

Algorithm 5.7: The Generalized Eigenvalue Optimization Method

Input: $R_1 \in \mathbb{S}_{++}^{n_x}$, $R_2 \in \mathbb{S}_{++}^{n_2}$, $R_{12} \in \mathbb{R}^{n_x \times n_2}$, $H_2 \in \mathbb{R}^{n_2 \times n_x}$, and m

1. Let $Q = (R_1 H_2^\mathsf{T} - R_{12})^\mathsf{T}(R_1 H_2^\mathsf{T} - R_{12})$ and $S = H_2 R_1 H_2^\mathsf{T} + R_2 - H_2 R_{12} - R_{21} H_2^\mathsf{T}$.
2. Compute $\lambda_1, \ldots, \lambda_{n_2}$ and $u_1, \ldots, u_{n_2}$ by solving $Qu = \lambda S u$.
3. Define $\Phi = \mathrm{col}(u_1^\mathsf{T}, \ldots, u_m^\mathsf{T})$, and compute $\Omega = \mathrm{col}(v_1^\mathsf{T}, \ldots, v_m^\mathsf{T})$ such that $v_i^\mathsf{T} v_j = \delta_{ij}$ and $\mathsf{rowspan}(\Omega) = \mathsf{rowspan}(\Phi)$.
4. Compute $\Omega R_2 \Omega^\mathsf{T} = U \Sigma U^\mathsf{T}$ and let $\Psi^\star = U^\mathsf{T}\Omega$.

Output: $\Psi^\star$

$R_\Psi = \Psi^\star R_2 (\Psi^\star)^\mathsf{T}$ is diagonal. If $\Psi^\star = U^\mathsf{T}\Omega$, then since U^T is orthogonal and Ω has orthonormal rows

$$\Psi^\star(\Psi^\star)^\mathsf{T} = U^\mathsf{T}\Omega\Omega^\mathsf{T}U = U^\mathsf{T} I U = I,$$

i.e., $\Psi^\star$ has orthonormal rows. The change of basis procedure is illustrated in Figure 5.6, where $R_\Psi = \Psi^\star R_2(\Psi^\star)^\mathsf{T}$, $R_\Phi = \Phi R_2 \Phi^\mathsf{T}$, and $R_\Omega = \Omega R_2 \Omega^\mathsf{T}$.

Proposed Framework

The proposed *generalized eigenvalue optimization* (GEVO) methodology for deriving $\Psi^\star$ is given in Algorithm 5.7. It is a direct application of Theorem 5.5 followed by the previously discussed change of basis procedure. Step 4 ensures that $\Psi^\star R_2(\Psi^\star)^\mathsf{T}$ is diagonal. Examples 5.8 illustrates the GEVO method.

Remark 5.6. The derivation of the GEVO method demonstrate that, by transforming the original problem, an EVP can equivalently be solved for derivation of $\Psi^\star$. However, the GEVP formulation is kept for numerical considerations [136]. The proposed GEVP can, e.g., be solved by the stable *QZ algorithm* [122].

Example 5.8: The Generalized Eigenvalue Optimization Method

Assume that $n_x = n_2 = 2$, $H_2 = I$, $m = 1$, and $R_{12} = 0$. Let R_1 and R_2 be defined according to their ellipses in Figure 5.7. Solving $Qu = \lambda Su$, where $Q = R_1^2$ and $S = R_1 + R_2$, yields two solutions $\lambda_{\min}$ and $\lambda_{\max}$ with associated generalized eigenvectors $u_{\min}$ and $u_{\max}$, respectively. The quantity of interest is $\mathcal{V} = \mathsf{rowspan}(u_{\max}^{\mathsf{T}})$.

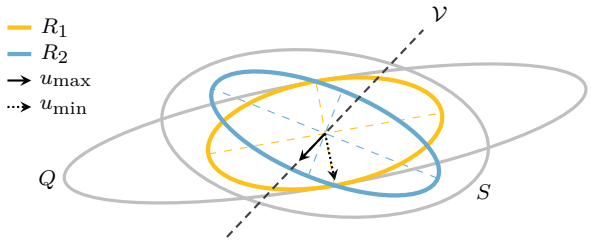


Figure 5.7. Example of the GEVO method. Since $m = 1$, $\Psi^\star = u_{\max}^{\mathsf{T}}$. The vector $u_{\max}$ spans the one-dimensional subspace $\mathcal{V}$. Principal components of R_1 and R_2 are illustrated by dashed lines.

Singular Case

If $S \succeq 0$ is singular, then $\Psi S \Psi^{\mathsf{T}}$ might be singular. If so, the inverse in (5.26) is replaced by a pseudoinverse such that

$$P = R_1 - \Delta \Psi^{\mathsf{T}} (\Psi S \Psi^{\mathsf{T}})^{+} \Psi \Delta^{\mathsf{T}}, \tag{5.32}$$

where $\Delta = R_1 H_2^{\mathsf{T}} - R_{12}$. Assume that $1 \leq m \leq \mathsf{rank}(S) = r$. Let $S = V D V^{\mathsf{T}}$, where $D = \mathsf{diag}(D_1, 0)$, $D_1 \in \mathbb{S}_{++}^r$, and 0 is an $(n_2 - r) \times (n_2 - r)$ matrix of zeros, such that

$$S^{+} = V D^{+} V^{\mathsf{T}} = V \,\mathsf{diag}(D_1^{-1}, 0) V^{\mathsf{T}}. \tag{5.33}$$

The case $m > r$ does not make sense since this does not improve P in (5.32) for the same reason that $m > n_2$ does not make sense if $S \in \mathbb{S}_{++}^{n_2}$. This is stated formally in Proposition 5.9[5]. Let $\Phi^{\mathsf{T}} = D^{\frac{1}{2}} V^{\mathsf{T}} \Psi^{\mathsf{T}}$. If $m \leq r$ and $\mathsf{rank}(\Psi) = m$, then it is still possible to impose the constraint $\Psi S \Psi^{\mathsf{T}} = I$ such that $\Phi \Phi^{\mathsf{T}} = I$. Hence, Algorithm 5.7 is valid even if S is singular given that $m \leq r$ and $\mathsf{rank}(\Psi) = m$.

Proposition 5.9. *Assume* $\mathsf{rank}(S) = \mathsf{rank}(\Psi_1) = r$, *where* $S \in \mathbb{S}_{+}^{n_2}$ *and* $\Psi_1 \in \mathbb{R}^{r \times n_2}$. *Let* $\Psi_1 S \Psi_1^{\mathsf{T}} \in \mathbb{S}_{++}^r$. *If* $\Psi = \mathsf{col}(\Psi_1, \Psi_2)$, *where* $\Psi_2 \in \mathbb{R}^{r \times n_2}$, *then*

$$\Psi^{\mathsf{T}} (\Psi S \Psi^{\mathsf{T}}) \Psi = \Psi_1^{\mathsf{T}} (\Psi_1 S \Psi_1^{\mathsf{T}}) \Psi_1. \tag{5.34}$$

Proof: Let $\Psi_2 = \mathsf{col}(\Psi_a, \Psi_b)$, where Ψ_a and Ψ_b lie in column space and in the null space of S, respectively. Hence, $\Psi_a = A \Psi_1$ and $\Psi_b S = 0$. Assume w.l.o.g. that

[5] This fact is also discussed in [187]. However, in that paper the corresponding proof is omitted.

$\Psi_1 S \Psi_1^{\mathsf{T}} = I$. In this case

$$\Psi S \Psi^{\mathsf{T}} = \begin{bmatrix} I & A^{\mathsf{T}} & 0 \\ A & AA^{\mathsf{T}} & 0 \\ 0 & 0 & 0 \end{bmatrix} = BB^{\mathsf{T}},$$

where $B = \begin{bmatrix} I & A^{\mathsf{T}} & 0 \end{bmatrix}^{\mathsf{T}}$. By construction, B is full column rank which implies that $B^{+} = (B^{\mathsf{T}}B)^{-1}B^{\mathsf{T}}$, where $B^{\mathsf{T}}B = I + A^{\mathsf{T}}A$. Moreover, $(BB^{\mathsf{T}})^{+} = (B^{\mathsf{T}})^{+}B^{+}$ [71]. Hence

$$\begin{aligned} \Psi^{\mathsf{T}}(\Psi S \Psi^{\mathsf{T}})^{+}\Psi &= \Psi^{\mathsf{T}}(BB^{\mathsf{T}})^{+}\Psi = \Psi^{\mathsf{T}}B(B^{\mathsf{T}}B)^{-1}(B^{\mathsf{T}}B)^{-1}B^{\mathsf{T}}\Psi \\ &= \begin{bmatrix} \Psi_1^{\mathsf{T}} & \Psi_1^{\mathsf{T}}A^{\mathsf{T}} & \Psi_b^{\mathsf{T}} \end{bmatrix} \begin{bmatrix} I \\ A \\ 0 \end{bmatrix} (I + A^{\mathsf{T}}A)^{-2} \begin{bmatrix} I & A^{\mathsf{T}} & 0 \end{bmatrix} \begin{bmatrix} \Psi_1 \\ A\Psi_1 \\ \Psi_b \end{bmatrix} \\ &= \Psi_1^{\mathsf{T}}(I + A^{\mathsf{T}}A)(I + A^{\mathsf{T}}A)^{-2}(I + A^{\mathsf{T}}A)\Psi_1 \\ &= \Psi_1^{\mathsf{T}}\Psi_1 = \Psi_1^{\mathsf{T}}(\Psi_1 S \Psi_1^{\mathsf{T}})^{-1}\Psi_1. \end{aligned}$$

□

Remark 5.10. If $m \leq r$, then it might still be that $\Psi Q \Psi^{\mathsf{T}} = 0$ for all feasible Ψ since Q might be singular. In this degenerated case, fusion of (y_1, R_1) with (y_Ψ, R_Ψ) yields no improvement compared to using (y_1, R_1) alone.

Choosing the Parameter m

The parameter m is essentially a design choice and might be the consequence of a communication constraint. As stated previously, there is no benefit of using $m > \mathsf{rank}(S) = r$, where $r \leq n_2$. In some cases, it is desirable to choose m adaptively. By construction

$$\mathsf{tr}\left(\Delta \Psi^{\mathsf{T}}(\Psi S \Psi^{\mathsf{T}})^{-1}\Psi \Delta^{\mathsf{T}}\right) = \mathsf{tr}\left((\Psi S \Psi^{\mathsf{T}})^{-1}\Psi Q \Psi^{\mathsf{T}}\right) = \sum_{i=1}^{m} \lambda_i, \tag{5.35}$$

where $\Psi = \Psi^{\star} = \mathsf{col}(u_1^{\mathsf{T}}, \ldots, u_m^{\mathsf{T}})$, (λ_i, u_i) is given by $Qu = \lambda S u$, and $\lambda_1 \geq \cdots \geq \lambda_{n_2}$. Consider now $\Psi = \Psi^{\star}$ as a function of m. By defining $\ell_0 = \mathsf{tr}(R_1)$ and

$$\ell_m = \mathsf{tr}(P) = \mathsf{tr}(R_1) - \mathsf{tr}\left((\Psi S \Psi^{\mathsf{T}})^{-1}\Psi Q \Psi^{\mathsf{T}}\right) = \ell_0 - \sum_{i=1}^{m} \lambda_i, \tag{5.36}$$

for $m \in \{1, 2, \ldots, r\}$, it is possible to relate m directly to the fusion gain with ℓ_m. For instance, m can be chosen adaptively to be the smallest integer m such that

$$\frac{\ell_0 - \ell_m}{\ell_0 - \ell_r} = \frac{\sum_{i=1}^{m} \lambda_i}{\sum_{i=1}^{r} \lambda_i} \geq \tau, \tag{5.37}$$

for some threshold $\tau \in [0, 1]$.

Algorithm 5.11: GEVO-KF

Input: $R_1 \in \mathbb{S}_{++}^{n_x}$, $R_2 \in \mathbb{S}_{++}^{n_2}$, $R_{12} \in \mathbb{R}^{n_x \times n_2}$, $H_2 \in \mathbb{R}^{n_2 \times n_x}$, and m

1. Let $Q = H_2 R_1^2 H_2^\mathsf{T}$ and $S = H_2 R_1 H_2^\mathsf{T} + R_2$.
2. Compute $\lambda_1, \ldots, \lambda_{n_2}$ and $u_1, \ldots, u_{n_2}$ by solving $Qu = \lambda S u$.
3. Define $\Phi = \mathrm{col}(u_1^\mathsf{T}, \ldots, u_m^\mathsf{T})$, and compute $\Omega = \mathrm{col}(v_1^\mathsf{T}, \ldots, v_m^\mathsf{T})$ such that $v_i^\mathsf{T} v_j = \delta_{ij}$ and $\mathrm{rowspan}(\Omega) = \mathrm{rowspan}(\Phi)$.
4. Compute $\Omega R_2 \Omega^\mathsf{T} = U \Sigma U^\mathsf{T}$ and let $\Psi^\star = U^\mathsf{T} \Omega$.

Output: $\Psi^\star$

5.3.3 GEVO for Kalman Fusion

The GEVO method for KF is automatically retrieved by setting $R_{12} = R_{21}^\mathsf{T} = 0$ in Algorithm 5.7, such that

$$Q = H_2 R_1^2 H_2^\mathsf{T}, \qquad S = H_2 R_1 H_2^\mathsf{T} + R_2.$$

For convenience and to be consistent with the other methods, the KF case is provided in Algorithm 5.11. KF assumes zero cross-correlations and is therefore applicable when some kind of decorrelation procedure is utilized. One example is the GIMF [169].

5.3.4 GEVO for Covariance Intersection

For fusion using CI, the GEVO method does not apply directly. The reason is the dependency on ω. In [57], an iterative algorithm based on *alternating minimization* (AM, [20]) is proposed. First, it is noted that CI only differs from KF through the ω parameter. The basic idea is to alternate between keeping ω and Φ fixed. A generalization for $m \geq 1$ of the original algorithm proposed in [57] is provided in Algorithm 5.12. The loss function value J_k is defined in (5.38).

The parameter ϵ in Algorithm 5.12 is a design parameter chosen as a compromise between computational speed and exactness of the solution given by the final iterate. Here, $\omega_0 = 1/2$ is used.

Convergence Analysis

The AM method in Algorithm 5.12 alternates between solving two different kinds of optimization problems. Each separate problem is well-posed, and a global minimum is obtained. The solution to the problem in step 1 is given by the generalized eigenvalues associated with the largest generalized eigenvalues of a GEVP. Step 2 involves solving a convex optimization problem [135] for which any local minimum is also a global minimum [29]. This does, however, not imply that the final iteration of Algorithm 5.12 is a global minimizer. Below, it is shown that the iterations of Algorithm 5.12 converge to a stationary point.

Algorithm 5.12: GEVO-CI

Input: ω_0, $R_1 \in \mathbb{S}^{n_x}_{++}$, $R_2 \in \mathbb{S}^{n_2}_{++}$, $H_2 \in \mathbb{R}^{n_2 \times n_x}$, m, $k=0$, and ϵ

1. Let $k \leftarrow k+1$. Compute $\lambda_1,\dots,\lambda_{n_2}$ and $u_1,\dots,u_p$ by solving $Qu = \lambda S u$, where $Q = H_2 R_1^2 H_2^{\mathsf{T}}/\omega_{k-1}^2$ and $S = H_2 R_1 H_2^{\mathsf{T}}/\omega_{k-1} + R_2/(1-\omega_{k-1})$. Let $\Phi_k = \mathrm{col}(u_1^{\mathsf{T}},\dots,u_m^{\mathsf{T}})$, where u_i is a generalized eigenvector associated with λ_i.
2. Let $R_\Phi = \Phi_k R_2 \Phi_k^{\mathsf{T}}$. Compute ω_k by solving
$$\underset{\omega}{\text{minimize}} \quad \mathrm{tr}\left(\left(\omega R_1^{-1} + (1-\omega) H_2^{\mathsf{T}} \Phi_k^{\mathsf{T}} R_\Phi^{-1} \Phi_k H_2\right)^{-1}\right).$$
3. Let J_k be according to (5.39). If $(J_{k-1} - J_k)/J_k > \epsilon$, then go back to step 1. Otherwise continue to step 4.
4. Define $\Phi = \mathrm{col}(u_1^{\mathsf{T}},\dots,u_m^{\mathsf{T}})$, and compute $\Omega = \mathrm{col}(v_1^{\mathsf{T}},\dots,v_m^{\mathsf{T}})$ such that $v_i^{\mathsf{T}} v_j = \delta_{ij}$ and $\mathsf{rowspan}(\Omega) = \mathsf{rowspan}(\Phi)$.
5. Compute $\Omega R_2 \Omega^{\mathsf{T}} = U \Sigma U^{\mathsf{T}}$ and let $\Psi = U^{\mathsf{T}} \Omega$.

Output: Ψ

Let $J(\omega,\Phi) = \mathsf{tr}(P)$ with P according to (5.6), i.e.,

$$J(\omega,\Phi) = \mathrm{tr}\left(\left(\omega R_1^{-1} + (1-\omega) H_2^{\mathsf{T}} \Phi^{\mathsf{T}} (\Phi R_2 \Phi^{\mathsf{T}})^{-1} \Phi H_2\right)^{-1}\right). \tag{5.38}$$

Define

$$J_{k-\frac{1}{2}} = J(\omega_{k-1},\Phi_k), \qquad J_k = J(\omega_k,\Phi_k). \tag{5.39}$$

Consider a sequence of n_k iterations and hence $2n_k$ subiterations. Each iteration and subiteration have the same feasible set. Since in each subiteration a minimum is obtained it is concluded that

$$J_{\frac{1}{2}} \geq J_1 \geq \cdots \geq J_{N-\frac{1}{2}} \geq J_N,$$

which is a nonincreasing sequence $\{J_k\}$. Denote by $\{(\omega,\Phi)_k\}$ the sequence of points generating $\{J_k\}$. Since $P \in \mathbb{S}^{n_x}_{++}$, there exists a lower bound $J_{\mathrm{low}} > 0$ such that $J(\omega,\Phi) \geq J_{\mathrm{low}}, \forall(\omega,\Phi)$. Hence, the *monotonic convergence theorem* [25] is applicable, which states that

$$\lim_{k\to\infty} J_k = J(\bar{\omega},\bar{\Phi}), \tag{5.40}$$

where $(\bar{\omega},\bar{\Phi})$ is the limit point. In the limit $J_{k-\frac{1}{2}} - J_{k+\frac{1}{2}} \to 0$ and $J_k - J_{k+1} \to 0$. Since $J(\omega,\Phi)$ is differentiable w.r.t. ω and Φ on its domain, this implies that

$$\frac{\partial}{\partial\omega} J(\omega,\Phi)\bigg|_{\omega=\bar{\omega}} = 0, \qquad \frac{\partial}{\partial\Phi} J(\omega,\Phi)\bigg|_{\Phi=\bar{\Phi}} = 0, \tag{5.41}$$

and hence $(\bar{\omega},\bar{\Phi})$ is a stationary point. The convergence results are summarized in Theorem 5.13. It should be emphasized that from Theorem 5.13 alone, it cannot be concluded if $(\bar{\omega},\bar{\Phi})$ is a local minimizer, global minimizer or a saddle point.

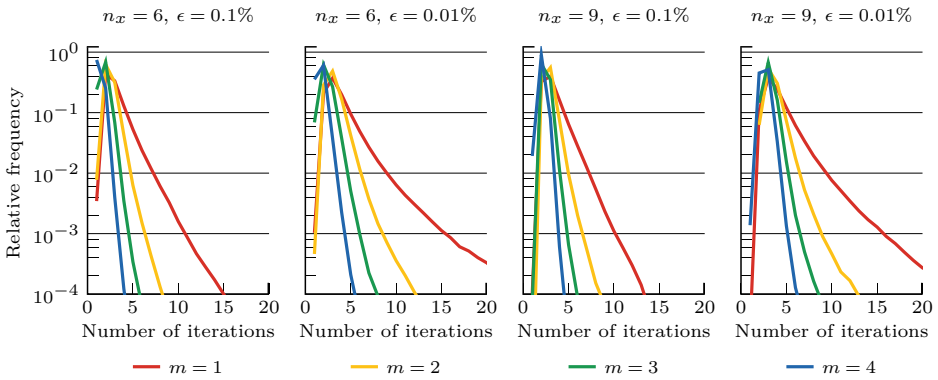


Figure 5.8. Results of the numerical convergence rate analysis, where $n_x \in \{6,9\}$ and $\epsilon \in \{0.1\%, 0.01\%\}$. The plots show the relative frequency of the number of iterations until the criterion $(J_{k-1} - J_k)/J_k \leq \epsilon$ in Algorithm 5.12 is met.

Theorem 5.13. *Let $\{(\omega,\Phi)_k\}$ be a sequence of points generated by Algorithm 5.12 and let $J(\omega,\Phi)$ be given by* (5.38)*. Then $\{(\omega,\Phi)_k\}$ converges to a stationary point $(\bar{\omega},\bar{\Phi})$ of $J(\omega,\Phi)$.*

The convergence rate is evaluated numerically. In each simulation, $R_1, R_2 \in \mathbb{S}^{n_x}_{++}$ are i.i.d. according to $R_1, R_2 \sim \mathcal{W}(I, n_x)$, where $\mathcal{W}(I, n_x)$ is the *Wishart distribution* of n_x degrees of freedoms [180]. If in a certain sample $R_2 \succeq R_1$, then R_1 is resampled until $R_2 \not\succeq R_1$. This resampling is done because $R_2 \succeq R_1$ would trivially yield $\omega = 1$ in Algorithm 5.12 for all feasible Ψ.

The evaluation is performed with $n_x \in \{6,9\}$ and $\epsilon \in \{0.1\%, 0.01\%\}$. In each case, and for each m, 1 000 000 MC simulations are conducted. The evaluation measure is the number of iterations until $(J_{k-1} - J_k)/J_k \leq \epsilon$. The results are shown in Figure 5.8. The statistics are summarized in Table 5.3, where *typ*, *mean*, and *std* refer to the typical value, the mean, and the standard deviation, respectively. The GEVO-CI algorithm converges very fast in general, e.g., for $n_x = 9$ and $\epsilon = 0.01\%$ with $m = 3$ the mean number of iterations is approximately 4.056 ± 0.634 and the typical value is 4. An interesting feature is that the convergence rate is improved by increasing m. The number of iterations increases as n_x increases and as ϵ decreases.

Table 5.3
Convergence rate analysis

	$n_x = 6,\ \epsilon = 0.1\%$			$n_x = 6,\ \epsilon = 0.01\%$			$n_x = 9,\ \epsilon = 0.1\%$			$n_x = 9,\ \epsilon = 0.01\%$		
m	**typ**	**mean**	**std**	**typ**	**mean**	**std**	**typ**	**mean**	**std**	**typ**	**mean**	**std**
1	3	3.995	1.311	4	4.795	2.099	4	4.156	1.241	4	5.108	2.059
2	3	3.421	0.645	4	4.035	0.961	4	3.744	0.663	4	4.487	0.914
3	3	2.836	0.549	3	3.298	0.691	3	3.376	0.513	4	4.056	0.634
4	2	2.260	0.449	3	2.687	0.554	3	3.062	0.313	4	3.582	0.568

Algorithm 5.14: GEVO-LE

Input: $R_1 \in \mathbb{S}^{n_x}_{++}$, $R_2 \in \mathbb{S}^{n_2}_{++}$, $H_2 \in \mathbb{R}^{n_2 \times n_x}$, and m

1. Transform into the information domain

$$\mathcal{I}_1 = R_1^{-1}, \qquad \mathcal{I}_2 = H_2^{\mathsf{T}} R_2^{-1} H_2.$$

2. Factorize $\mathcal{I}_1 = U_1 \Sigma_1 U_1^{\mathsf{T}}$ and let $T_1 = \Sigma_1^{-\frac{1}{2}} U_1^{\mathsf{T}}$. Factorize $T_1 \mathcal{I}_2 T_1^{\mathsf{T}} = U_2 \Sigma_2 U_2^{\mathsf{T}}$ and let $T_2 = U_2^{\mathsf{T}}$. Transform using $T = T_2 T_1$ according to

$$\mathcal{I}_1' = T \mathcal{I}_1 T^{\mathsf{T}} = I, \qquad \mathcal{I}_2' = T \mathcal{I}_2 T^{\mathsf{T}}.$$

3. Let D be diagonal. For each $i = 1, \ldots, n_x$ compute

$$[D]_{ii} = \min\left(1, [\mathcal{I}_2']_{ii}\right).$$

4. Let $\mathcal{I}_\gamma = T^{-1} D T^{-\mathsf{T}}$ and $R_{12} = R_1 \mathcal{I}_\gamma H_2^{\mathsf{T}} R_2$. Compute Ψ using Algorithm 5.7 with inputs R_1, R_2, R_{12}, H_2, and m.

Output: Ψ

5.3.5 GEVO for the Largest Ellipsoid Method

To be able to use the GEVO framework with the LE method, some adaptations are required. As discussed in Section 4.4.3, the LE method makes an implicit assumption about R_{12} if $H_2 = I$. Based on this and the results in (4.24), R_{12} is now derived for arbitrary H_2. Let $\tilde{y}_i = y_i - \mathrm{E}(y_i)$, $\tilde{y}_i^e = y_i^e - \mathrm{E}(y_i^e)$, and $\tilde{\gamma} = \hat{\gamma} - \mathrm{E}(\hat{\gamma})$ as before. Assume the common information decomposition in (4.23), i.e.,

$$\begin{aligned} R_1^{-1} &= \mathcal{I}_1^e + \mathcal{I}_\gamma, & R_1^{-1} y_1 &= \mathcal{I}_1^e y_1^e + \mathcal{I}_\gamma \hat{\gamma}, \\ R_2^{-1} &= \mathcal{I}_2^e + H_2 \mathcal{I}_\gamma H_2^{\mathsf{T}}, & R_2^{-1} y_2 &= \mathcal{I}_2^e y_2^e + H_2 \mathcal{I}_\gamma \hat{\gamma}. \end{aligned}$$

The key step is to compute the common information $\mathcal{I}_\gamma$ assumed implicitly in the LE method in Algoirthm 4.17. This corresponds to step 1 to 3 in Algorithm 5.14. Then (4.24) gives R_{12} according to

$$R_{12} = R_1 \mathcal{I}_\gamma H_2^{\mathsf{T}} R_2.$$

When R_{12} has been obtained, Algorithm 5.7 is applicable. The GEVO method for LE is provided in Algorithm 5.14. If $\mathsf{rank}(H_2) = n_x$, then both $\mathcal{I}_1$ and $\mathcal{I}_2$ in Algorithm 5.14 are full rank. Hence, also $\mathcal{I}_\gamma$ is full rank, i.e., $\mathcal{I}_\gamma^{-1} = \Gamma$.

In Section 4.4.3, it was noted that if $H_2 = I$, then LE corresponds to a particular R_{12}. As a consequence, Ψ computed by Algorithm 5.7 is guaranteed to be optimal. That is, Ψ computed by Algorithm 5.14 is optimal w.r.t. $\mathsf{tr}(P)$, if $H_2 = I$ and P is computed using (5.8). The LE method is a best CLUE when the following conditions hold: (i) $H_2 = I$; and (ii) R_{12} is according to the componentwise aligned structure described in Section 4.4.3. This means that Ψ computed by Algorithm 5.14 implies an optimal conservatively fused estimate if (i) and (ii) hold simultaneously.

5.3.6 Theoretical Comparison of GEVO and PCO

The GEVO method in Algorithm 5.7 computes $\Psi^\star$, which is optimal w.r.t. $\mathrm{tr}(P)$. In certain cases, $\Psi^\star$ is equal to Ψ_{pco} computed by the PCO method in Algorithm 5.2, but in general, $\Psi_{\mathrm{pco}} \neq \Psi^\star$. This has actually already been demonstrated, cf. the motivating example of Section 5.3.1. Example 5.15 illustrates a case when $\Psi_{\mathrm{pco}} = \Psi^\star$ and a case when Ψ_{pco} is as far as possible from $\Psi^\star$.

Example 5.15: Relationship Between GEVO and PCO

Assume that $n_2 = n_x = 2$, $H_2 = I$, and $R_{12} = 0$. Let $R_2 = \mathsf{diag}(4,1)$ and R_1 be defined either as: (i) $R_1 = \mathsf{diag}(1,4)$; or (ii) $R_2 = \mathsf{diag}(4,1)$. Let Ψ_{gevo} be computed using Algorithm 5.11 and Ψ_{pco} be computed using Algorithm 5.2. Let $u_{\min}$ and $u_{\max}$ be the eigenvectors associated with the minimum and maximum eigenvalues $\lambda_{\min}(R_2)$ and $\lambda_{\max}(R_2)$, respectively. In case (i)

$$\Psi_{\mathrm{gevo}} = \begin{bmatrix} 0 & 1 \end{bmatrix} = u_{\min}^{\mathsf{T}}, \qquad \Psi_{\mathrm{pco}} = \begin{bmatrix} 0 & 1 \end{bmatrix} = u_{\min}^{\mathsf{T}},$$

such that $\Psi_{\mathrm{pco}} = \Psi_{\mathrm{gevo}}$. In case (ii)

$$\Psi_{\mathrm{gevo}} = \begin{bmatrix} 1 & 0 \end{bmatrix} = u_{\max}^{\mathsf{T}}, \qquad \Psi_{\mathrm{pco}} = \begin{bmatrix} 0 & 1 \end{bmatrix} = u_{\min}^{\mathsf{T}},$$

such that $\Psi_{\mathrm{gevo}}\Psi_{\mathrm{pco}}^{\mathsf{T}} = 0$. That is, PCO yields $\Psi_{\mathrm{pco}} = u_{\min}^{\mathsf{T}}$, but to minimize $\mathrm{tr}(P)$ the $u_{\max}^{\mathsf{T}}$ should be used. As demonstrated below, in case (ii), PCO yields the worst possible choice of Ψ w.r.t. to $\mathrm{tr}(P)$.

The observation made in Example 5.15 is a special case of a more general result. Assume that $R_{12} = 0$, such that (y_1, R_1) and (y_Ψ, R_Ψ) are fused optimally using (5.5). Let $H_2 R_1 H_2^{\mathsf{T}} \in \mathbb{S}_+^{n_2}$ and $R_2 \in \mathbb{S}_{++}^{n_2}$ be given by

$$H_2 R_1 H_2^{\mathsf{T}} = V \Sigma V^{\mathsf{T}}, \qquad \Sigma = \mathsf{diag}(\mu_1, \ldots, \mu_{n_2}), \tag{5.42a}$$

$$R_2 = V \Pi V^{\mathsf{T}}, \qquad \Pi = \mathsf{diag}(\pi_1, \ldots, \pi_{n_2}), \tag{5.42b}$$

where $V = \begin{bmatrix} v_1 & \ldots & v_{n_2} \end{bmatrix}$ is an orthogonal matrix, $\mu_1 \geq \cdots \geq \mu_{n_2}$ are the eigenvalues of $H_2 R_1 H_2^{\mathsf{T}}$, and $\pi_1 \geq \cdots \geq \pi_{n_2}$ are the eigenvalues of R_2. That is, $H_2 R_1 H_2^{\mathsf{T}}$ and R_2 share eigenvectors, and their eigenvalues are ordered correspondingly. In this case

$$Q = (H_2 R_1 H_2^{\mathsf{T}})^2 = V \Sigma^2 V^{\mathsf{T}}, \qquad S = H_2 R_1 H_2^{\mathsf{T}} + R_2 = V(\Sigma + \Pi) V^{\mathsf{T}}.$$

Since $S \in \mathbb{S}_{++}^{n_2}$, the GEVP $Qu = \lambda S u$ solved in GEVO can be expressed as

$$S^{-1} Q u = V(\Sigma + \Pi)^{-1} V^{\mathsf{T}} V \Sigma^2 V = V \Lambda V^{\mathsf{T}} u = \lambda u, \tag{5.43}$$

where $\Lambda = \mathsf{diag}(\lambda_1, \ldots, \lambda_{n_2})$ and $\lambda_i = \mu_i^2/(\mu_i + \pi_i)$. The assumed ordering of μ_i and π_i implies that

$$\frac{1}{\mu_1 + \pi_1} \leq \cdots \leq \frac{1}{\mu_{n_2} + \pi_{n_2}},$$

and hence $\lambda_1 \geq \cdots \geq \lambda_{n_2}$. Since $R_{12} = 0$, Algorithm 5.11 is applicable, where the output $\Psi^\star$ minimizes $\mathsf{tr}(P)$ with P given in (5.5). In this case $\Psi^\star = \mathsf{col}(v_1^\mathsf{T}, \ldots, v_m^\mathsf{T})$. Meanwhile, using Algorithm 5.2 yields $\Psi_{\text{pco}} = \mathsf{col}(v_{n_2-m+1}^\mathsf{T}, \ldots, v_{n_2}^\mathsf{T})$. However, by inspection of Algorithm 5.11 it can be seen that this Ψ_{pco} in fact minimizes $\mathsf{tr}((\Psi S \Psi^\mathsf{T})^{-1} \Psi Q \Psi^\mathsf{T})$ and hence maximizes $\mathsf{tr}(P)$ among all feasible $\Psi \in \mathbb{R}^{m \times n_2}$. Under the given assumptions, it is concluded that PCO provides the worst possible Ψ if the goal is to minimize $\mathsf{tr}(P)$[6]. These results are summarized in Theorem 5.16.

Theorem 5.16. *Assume that $R_{12} = 0$, and that $H_2 R_1 H_2^\mathsf{T} \in \mathbb{S}_+^{n_2}$ and $R_2 \in \mathbb{S}_{++}^{n_2}$ are given according to (5.42), where $\mu_1 \geq \cdots \geq \mu_{n_2}$ and $\pi_1 \geq \cdots \geq \pi_{n_2}$. Let $\Psi \in \mathbb{R}^{m \times n_2}$ and P be given according to (5.5). Then, Ψ_{pco} computed by the PCO method in Algorithm 5.2 solves*

$$\begin{aligned} \underset{\Psi}{\text{maximize}} \quad & \mathsf{tr}(P) \\ \text{subject to} \quad & \Psi \Psi^\mathsf{T} = I. \end{aligned} \tag{5.44}$$

In practice, it is not likely to have exactly the conditions assumed in Theorem 5.16. One relevant question is what can be expected when the conditions hold approximately. Consider A and $B = A + \Delta$. From matrix perturbation theory it is known that if the eigenvalues of A are distinct, then for small Δ such that $A \approx B$, the eigenvalues and eigenvectors A and B are approximately equal [164]. This can be utilized as follows. Assume

$$H_2 R_1 H_2^\mathsf{T} = V \Sigma V^\mathsf{T} + \Delta, \qquad R_2 = V \Pi V^\mathsf{T} - \Delta,$$

where V, Σ, and Π are defined as in (5.42). Then

$$Q = V \Sigma V^\mathsf{T} + \Delta V \Sigma V^\mathsf{T} + V \Sigma V^\mathsf{T} \Delta + \Delta^2, \qquad S = V(\Sigma + \Pi) V^\mathsf{T}.$$

The matrix Λ of the EVP defined in (5.43) is now given by

$$\Lambda = \Lambda_0 + V(\Sigma + \Pi)^{-1} V^\mathsf{T} \Delta V \Sigma V^\mathsf{T} + V(\Sigma + \Pi)^{-1} \Sigma V^\mathsf{T} \Delta + \Delta^2,$$

where $\Lambda_0 = V(\Sigma + \Pi)^{-1} \Sigma^2 V^\mathsf{T}$. Assume that Δ is such that $\Lambda \approx \Lambda_0$ and $R_2 \approx V \Pi V^\mathsf{T}$. If, in addition, the eigenvalues of each of Λ_0 and $V \Pi V^\mathsf{T}$ are distinct, then the following apply: (i) PCO with R_2 yields approximately the same solution as PCO with $V \Sigma V^\mathsf{T}$; and (ii) GEVO with Λ yields approximately the same solution as GEVO with Λ_0. If so, Theorem 5.16 is expected to be approximately true.

In the previous reasoning, it is not stated what a small Δ quantitatively means. However, the main point here is in the qualitative sense. That is, if (5.42) holds approximately, then PCO likely is a bad choice for DR.

5.3.7 Parametrized Fusion Example

The GEVO method is now evaluated in a simple fusion example.

[6] PCO maximizes the MSE in this case.

Simulation Specifications

Assume that $N = 2$, $n_x = n_2 = 6$, and $H_2 = I$. The problem is to fuse (y_1, R_1) and (y_Ψ, R_Ψ). The scenario is parametrized in $\rho \in [0,1]$ according to

$$R_1^{-1} = (1-\rho)A^{-1} + \rho\Gamma^{-1}, \qquad A^{-1} = \mathbf{diag}\,(64,32,16,8,4,2),$$
$$R_2^{-1} = (1-\rho)B^{-1} + \rho\Gamma^{-1}, \qquad B^{-1} = \mathbf{diag}\,(5,8,13,21,34,55),$$

with

$$\Gamma^{-1} = \begin{bmatrix} 16 & 4 & 4 & 0 & -2 & 0 \\ 4 & 20 & 8 & -8 & -4 & -4 \\ 4 & 8 & 30 & 0 & -4 & -4 \\ 0 & -8 & 0 & 50 & 0 & 0 \\ -2 & -4 & -4 & 0 & 10 & 0 \\ 0 & -4 & -4 & 0 & 0 & 20 \end{bmatrix}.$$

The quantity $\rho\Gamma^{-1}$ is interpreted as common information. By construction, $R_{12} = \rho R_1 \Gamma^{-1} R_2$. The parameter ρ is varied in the interval $[0,1]$, and $(\hat{x}, P)$ is computed for each ρ and for each of the considered methods. As ρ increases, y_1 and y_2 become more correlated, eventually becoming fully correlated at $\rho = 1$.

The following methods are compared:

- KF: The Kalman fuser as defined in (5.5), where Ψ is derived using Algorithm 5.11.
- CI: Covariance intersection as defined in (5.6), where Ψ is derived using Algorithm 5.12.
- LE: The largest ellipsoid method as defined in (5.8), where Ψ is derived using Algorithm 5.14.

As pointed out earlier, applying KF in a DSN requires some sort of decorrelation mechanism. Otherwise, the independence assumed in KF is violated. In the next comparison, KF is based on two different assumptions: (i) y_2 is decorrelated by the removal of common information such that $R_2^{-1} = (1-\rho)B^{-1}$. This case is denoted *decorrelated KF* (dKF). (ii) y_1 and y_2 are uncorrelated. This case is denoted NKF. Table 5.4 provides a summary of how different quantities are computed in the simulations.

Table 5.4
Computation of simulated quantities

Method	R_2^{-1}	R_{12}	Ψ	P
dKF	$(1-\rho)B^{-1}$	0	Algorithm 5.11	(5.5)
CI	$(1-\rho)B^{-1} + \rho\Gamma^{-1}$	$\rho R_1\Gamma^{-1}R_2$	Algorithm 5.12	(5.6)
LE	$(1-\rho)B^{-1} + \rho\Gamma^{-1}$	$\rho R_1\Gamma^{-1}R_2$	Algorithm 5.14	(5.8)
NKF	$(1-\rho)B^{-1} + \rho\Gamma^{-1}$	$\rho R_1\Gamma^{-1}R_2$	Algorithm 5.11	(5.5)

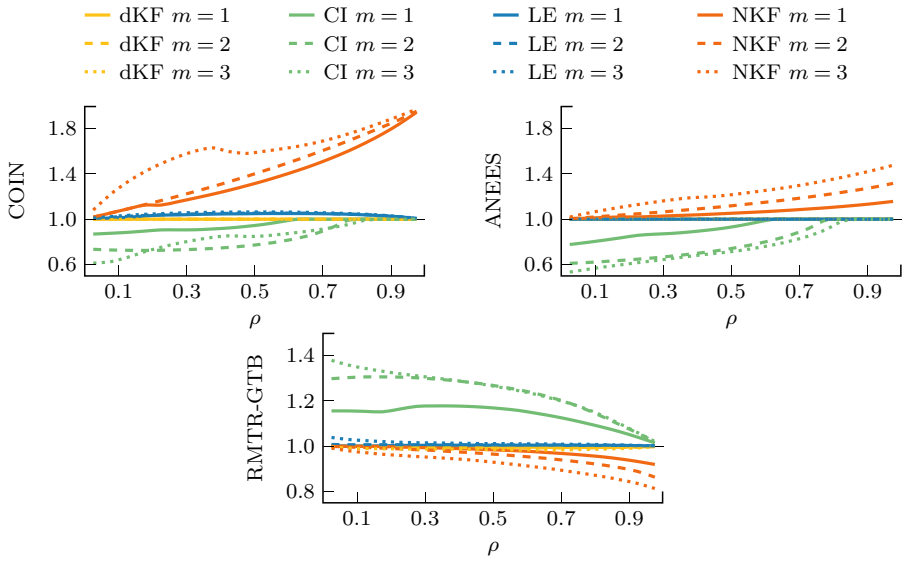


Figure 5.9. Results from the parametrized fusion example. Solid, dashed, and dotted lines refer to $m=1$, $m=2$, and $m=3$, respectively.

Evaluation Measures

COIN and ANEES are used for evaluation. In addition, *RMTR-GTB* is used, which is defined as follows. Assume a fixed m and let P be the covariance computed by a certain method defined above. Let P^{BSC} be the covariance computed using (5.4) with Ψ derived using Algorithm 5.7 where R_{12} is known. Then

$$\text{RMTR-GTB} = \frac{\sqrt{\text{tr}(P)}}{\sqrt{\text{tr}(P^{\text{BSC}})}}. \tag{5.45}$$

Results

The results are visualized in Figure 5.9. Solid, dashed, and dotted lines refer to $m=1$, $m=2$, and $m=3$, respectively.

dKF is able to achieve RMTR-GTB slightly below 1. This is possible because dKF, using the ideal decorrelation step, exploits more structure in the problem compared to what is possible using the MSE optimal estimator in (5.4). dKF is conservative w.r.t. both COIN and ANEES. On the other hand, NKF provides RMTR-GTB ≤ 1 but at the cost of not being conservative for $\rho > 0$. NKF hence utilizes more information than is actually available. This effect becomes more prominent for large m and ρ.

The performance of CI increases as ρ increases, but for small ρ, the performance is relatively poor. The reason for this is that since CI is conservative for all possible R_{12}, CI implicitly assumes strong correlations, but small ρ means weak correlations. COIN is typically larger for large m in the case of CI.

The LE method provides relatively good performance for all ρ. LE is conservative w.r.t. ANEES but not w.r.t. COIN, for which it provides values marginally larger than 1.

5.3.8 Numerical Evaluation

The GEVO method is evaluated using a DTT scenario identical to the one used in Section 5.2.2 to evaluate the PCO method. GEVO is also compared to PCO.

Simulation Specifications

The scenario is depicted in Figure 5.3 and the simulation parameters are summarized in Table 5.2. The communication scheme of Rule 3.6 is assumed. It is assumed $H_2 = I$ and $m = 2$.

The following methods are compared:

- NKF: The naïve Kalman fuser as defined in (5.5), where Ψ is derived using Algorithm 5.11.
- CI: Covariance intersection as defined in (5.6), where Ψ is derived using Algorithm 5.12.
- LE: The largest ellipsoid method as defined in (5.8), where Ψ is derived using Algorithm 5.14.

Evaluation Measures

COIN, ANEES, RMT, and RMSE defined in Section 3.2 are used for evaluation. In addition, RMTR and RMSER defined in Section 5.2.2 are used. By construction, RMTR is larger than or equal to 1.

Results

Figure 5.10 illustrates the results for Agent 3 only[7]. COIN, ANEES, RMT, RMSE, RMTR, and RMSER are computed at time instants where track fusion is performed. As a comparison, the results from the PCO method evaluation have been included for RMT and RMSE. The PCO curves are given by the dashed lines and use the same color coding as the corresponding GEVO curves, e.g., the red dashed lines refer to NKF with Ψ derived using the PCO method.

CI is the only method that is conservative w.r.t. both COIN and ANEES. LE is conservative w.r.t. ANEES. NKF is never conservative. As indicated by the RMT and RMSE plots, LE and KF achieves significantly better performance when using GEVO instead of PCO. CI achieves significantly improved RMSE, but only slightly improved RMT in the GEVO case. For LE, the performance loss of transmitting (y_Ψ, R_Ψ) instead of (y_2, R_2) is zero in this case with Ψ computed using GEVO-LE. This can be explained by the binary behavior suggested by LE, cf. Algorithm 4.17, where each component of the fused estimate is taken exclusively from either of the

[7] The estimation results for the other two agents are similar.

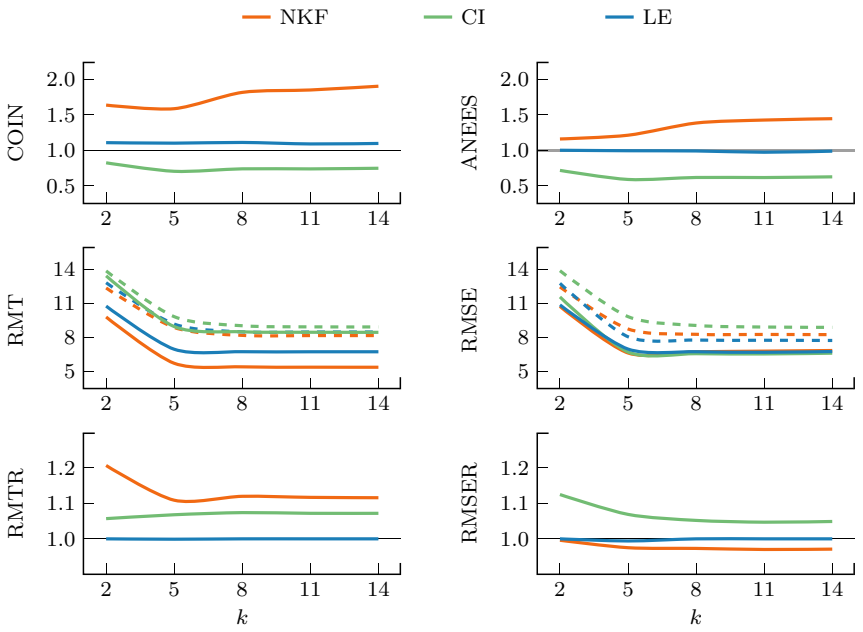


Figure 5.10. Results from the GEVO method evaluation. The dashed curves in the RMT and RMSE plots illustrate the corresponding results from the PCO method evaluation. The gray area in the ANEES plot represents an ANEES 99.9% confidence interval.

two estimates that are fused. In this case, it is useless to increase m since the additional information in (y_Ψ, R_Ψ) cannot be utilized by LE.

An interesting observation is that RMSER is smaller than 1 for NKF. The reason is that when full estimates are exchanged, the double counting of information by the naïve KF method makes the RMSE diverge. This effect is reduced when the GEVO derived DR estimates are exchanged instead of full estimates.

5.4 Dimension-Reduction Using Local Information Only

The GEVO framework for computation of Ψ requires that Agent 2 has access to R_1. This might be possible under restrictive assumptions in certain well-specified static sensor networks. However, for general DSNs, this is an unrealistic assumption. Consider the scenario in Figure 5.11. Assume $R_{12} = 0$ and that Agent 2 is about to communicate a DR estimate (y_Ψ, R_Ψ) derived using GEVO-KF to Agent 1. Since Agent 2 does not have access to R_1, Agent 2 wants to replace R_1 by some reasonable covariance matrix that is locally computable and at the same time yields a sufficiently good approximation of the optimal $\Psi^\star$ produced by GEVO-KF knowing R_1. Let $P(\Psi)$ be the covariance computed by fusing (y_1, R_1) and

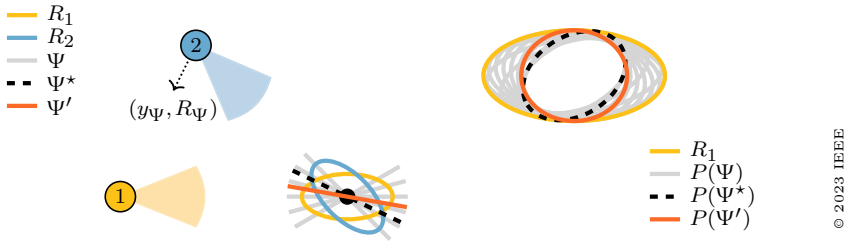


Figure 5.11. *Left:* A DTT scenario where Agent 2 transmits (y_Ψ, R_Ψ) to Agent 1. To compute $\Psi^\star$ using GEVO, Agent 2 must have access to R_1 which is unavailable. The objective is to replace R_1 and derive a reasonable approximation Ψ' of $\Psi^\star$. *Right:* Since $\mathrm{tr}(P(\Psi')) \gtrsim \mathrm{tr}(P(\Psi^\star))$, Ψ' is a good substitution for $\Psi^\star$.

(y_Ψ, R_Ψ). The objective is for Agent 2 to derive Ψ' such that

$$\mathrm{tr}\left(P(\Psi')\right) \gtrsim \mathrm{tr}\left(P(\Psi^\star)\right).$$

One such approximation is Ψ', see the r.h.s. of Figure 5.11.

The proposed solution is given by the *common information estimate* (CIE). CIE is an estimate that can be computed locally for each agent and captures the information being shared in the sensor network. As such, CIE can be interpreted as a *digital twin* [62] of the network information.

5.4.1 The Common Information Estimate

CIE is computed locally using all information that has either been received or transmitted by the agent. For instance, if Agent 2 transmits (y_Ψ, R_Ψ) to Agent 1 for fusion, then (y_Ψ, R_Ψ) is also fused with the CIE computed in Agent 2. The basic ideas originate from [55], where a similar approach is made based on fusing a similar estimate exclusively with received or transmitted information[8].

The SSM in (3.1) is assumed. State estimation is done using the EKF in Algorithm 3.1. The track fusion methods considered in this section are KF, CI, and LE, but other methods are also applicable. In this section, it is assumed for simplicity that $N = 2$. However, as a consequence of its modular construction, CIE can be used for arbitrary N.

Revisiting the Common Information Decomposition

Assume $H_2 = I$ such that $R_1, R_2 \in \mathbb{S}^{n_x}_{++}$. Recall the common information decomposition defined in (4.16), i.e.,

$$R_1^{-1} = (R_1^e)^{-1} + \Gamma^{-1}, \qquad R_1^{-1} y_1 = (R_1^e)^{-1} y_1^e + \Gamma^{-1} \hat{\gamma}, \tag{5.46a}$$

$$R_2^{-1} = (R_2^e)^{-1} + \Gamma^{-1}, \qquad R_2^{-1} y_2 = (R_2^e)^{-1} y_2^e + \Gamma^{-1} \hat{\gamma}, \tag{5.46b}$$

[8] These two special cases are not studied further in this scope.

where the estimate $(\hat{\gamma},\Gamma)$ is common to both local estimates. The basic idea used in the CIE methodology is to keep approximate track of $(\hat{\gamma},\Gamma)$. Similar to Section 4.4, $\mathcal{I}_i^e = (R_i^e)^{-1}$ is allowed to be singular.

In general, locally filtered estimates cannot be decomposed perfectly according to (5.46). However, this is not an issue for two main reasons: (i) In decentralized estimation sharing of information is the dominating source of cross-correlations between estimates and hence there is in general a large portion of common information shared between the local estimates. (ii) The purpose of the CIE is to be used as a substitute for R_1 to derive Ψ and therefore it is not essential that the underlying structure assumed in (5.46) is completely satisfied.

To illustrate the usage of CIE, assume that (5.46) holds, Γ^{-1} is known, and that $\mathcal{I}_1^e = 0$ and $(R_2^e)^{-1} \in \mathbb{S}_{++}^{n_x}$. An estimate (y_2^e, R_2^e) is recovered by subtracting $(\hat{\gamma},\Gamma)$ from (y_2, R_2). The result is two uncorrelated estimates $(\hat{\gamma},\Gamma)$ and (y_2^e, R_2^e) for which GEVO-KF can be applied by replacing R_1 and R_2 by Γ and R_2^e, respectively. After $\Psi^\star$ has been derived using Algorithm 5.11 it is possible to compute

$$y_\Psi = \Psi^\star y_2^e, \qquad\qquad R_\Psi = \Psi^\star R_2^e (\Psi^\star)^\mathsf{T},$$

which are then transmitted from Agent 2 to Agent 1.

In general, $0 \neq \mathcal{I}_1^e \in \mathbb{S}_+^{n_x}$. Assume that Agent 2 knows Γ but not $\mathcal{I}_1^e$. To investigate the effect of assuming $\mathcal{I}_1^e = 0$ when in fact $\mathcal{I}_1^e \neq 0$, assume w.l.o.g. that $\Gamma^{-1} = I$ and $\mathcal{I}_1^e = D$ where $D = \mathsf{diag}(d_1, \ldots, d_{n_x})$ is diagonal with $d_i \geq 0$. Let (y_2^e, R_2^e) be given as above. By construction, $Q = \Gamma^2 = I$ and $S = R_2^e$. The true values Q^0 and S^0 are now interpreted as perturbations of Q and S, i.e.,

$$Q^0 = \left((\Gamma^{-1} + \mathcal{I}_1^e)^{-1}\right)^2 = (I+D)^{-2}, \quad S^0 = \left(\Gamma^{-1} + \mathcal{I}_1^e\right)^{-1} + R_2^e = (I+D)^{-1} + R_2^e.$$

For small D these matrices can be approximated as

$$\begin{aligned} Q^0 &= (I + 2D + D^2)^{-1} \approx (I+2D)^{-1} = I - (I+2D)^{-1} 2D \approx I - 2D, \\ S^0 &= R_2^e + I - (I+D)^{-1} D \approx R_2^e + I - D. \end{aligned}$$

Hence, consider the following perturbations

$$Q^0 = Q + \delta Q, \qquad \delta Q = -2D, \qquad S^0 = S + \delta S, \qquad \delta S = -D.$$

Assume that (λ_i^0, u_i^0) solves $Q^0 u^0 = \lambda^0 S^0 u^0$ and that (λ_i, u_i) solves $Qu = \lambda S u$. Using the results of [164] it is implied that, if $\lambda_1^0, \ldots, \lambda_n^0$ are distinct, then for small D, (λ_i, u_i) is a reasonable approximation of (λ_i^0, u_i^0). This essentially means that for small D it can be expected that the performance degradation is small when using $\Psi = \mathsf{col}(u_1^\mathsf{T}, \ldots, u_m^\mathsf{T})$ instead of $\Psi^\star = \mathsf{col}((u_1^0)^\mathsf{T}, \ldots, (u_m^0)^\mathsf{T})$.

Defining the Common Information Estimate

CIE is denoted by $(\hat{\gamma},\Gamma)$ and is filtered in an EKF setting analogously to the local estimate. The local estimate is for now denoted by $(\hat{x}, P)$ instead of (y_2, R_2). These estimates are computed as follows:

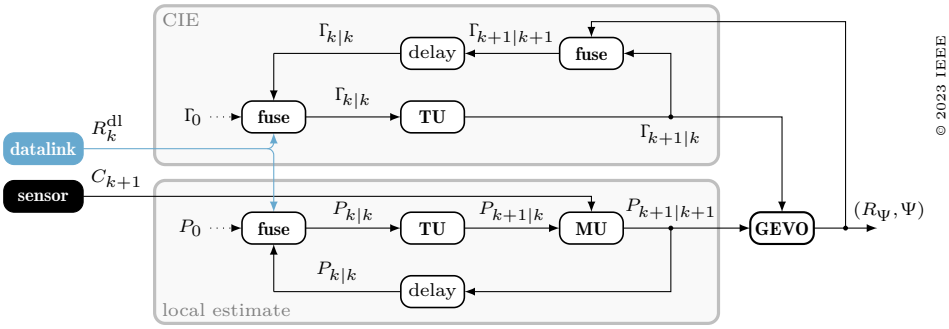


Figure 5.12. Schematics of the CIE methodology. Subscript 0 refers to initial values. Sensor information is used only in the measurement update (MU) of the local estimate. The fuse blocks are realized by one of the methods in Section 5.1.2. The GEVO block is realized by one of the methods in Section 5.3.2.

Table 5.5
GEVO INPUT MAPPING

Method	R_1	R_2	R_Ψ
GEVO-KF	Γ	$(P^{-1}-\Gamma^{-1})^{-1}$	$\Psi(P^{-1}-\Gamma^{-1})^{-1}\Psi^{\mathsf{T}}$
GEVO-CI	Γ	P	$\Psi P\Psi^{\mathsf{T}}$
GEVO-LE	Γ	P	$\Psi P\Psi^{\mathsf{T}}$

- $(\hat{x}, P)$ is predicted at each time step and filtered with local measurements (z, C), and fused with datalink estimates $(y^{\mathrm{dl}}, R^{\mathrm{dl}})$ received from other agents.
- $(\hat{\gamma}, \Gamma)$ is predicted at each time step, fused with (i) datalink estimates $(y^{\mathrm{dl}}, R^{\mathrm{dl}})$ received from others, and (ii) locally computed DR estimates (y_Ψ, R_Ψ) transmitted to other agents.

The same process model is assumed for both $(\hat{x}, P)$ and $(\hat{\gamma}, \Gamma)$. The process noise covariance Q acts as a forgetting factor that ages previously exchanged information [55]. The larger Q is, the faster previously exchanged information is forgotten.

Schematics of the CIE is provided in Figure 5.12, where only the computation of covariances is illustrated. It is suggested that $(\hat{\gamma}, \Gamma)$ is initialized at the same time as $(\hat{x}, P)$ using $\hat{\gamma}_0 = \hat{x}_0$ while $\Gamma_0 \succ P_0$ is chosen sufficiently large to be consistent with the fact that initially Γ^{-1} is negligible. Initialization of $(\hat{x}, P)$ is done by any standard procedure from target tracking [27].

Using The Common Information Estimate With GEVO

Utilizing $(\hat{\gamma}, \Gamma)$ in GEVO-CI and GEVO-LE is straighforward: R_1 is replaced by Γ and R_2 by P. Then GEVO-CI or GEVO-LE is used. To be able to apply GEVO-KF, $(\hat{x}, P)$ and $(\hat{\gamma}, \Gamma)$ must first be decorrelated. This is accomplished by subtracting $(\hat{\gamma}, \Gamma)$ from $(\hat{x}, P)$, an operation similar to the one that GIMF. In particular, R_1 is replaced by Γ and R_2 by $(P^{-1}-\Gamma^{-1})^{-1}$, and then GEVO-KF

is run. Table 5.5 summarizes the mapping between the quantities used in this section and the input variables of the different GEVO algorithms.

Decorrelation Procedure Analysis

To ensure that KF does not double count information when fusing a received estimate (y_Ψ, R_Ψ) with the local estimate, it must be shown that subtracting $(\hat{\gamma}, \Gamma)$ from $(\hat{x}, P)$ fully decorrelates the local estimate and the CIE. For the following analysis it is assumed that $P_{k|l}^{-1} \succ \Gamma_{k|l}^{-1}$. The latter is ensured if $P_0^{-1} \succ \Gamma_0^{-1}$ and only the operations described below are used. Let $\tilde{x}_{k|l} = \hat{x}_{k|l} - x_k$, $\tilde{\gamma}_{k|l} = \hat{\gamma}_{k|l} - x_k$, and $\tilde{\delta}_{k|l} = \hat{\delta}_{k|l} - x_k$. What needs to be show is that

$$\mathrm{E}(\tilde{\delta}_{k|l}\tilde{\gamma}_{k|l}^{\mathsf{T}}) = 0, \tag{5.47}$$

for all $k \geq l, l \geq 0$, where

$$\hat{\delta}_{k|l} = \Delta_{k|l}\left(P_{k|l}^{-1}\hat{x}_{k|l} - \Gamma_{k|l}^{-1}\hat{\gamma}_{k|l}\right), \qquad \Delta_{k|l} = \left(P_{k|l}^{-1} - \Gamma_{k|l}^{-1}\right)^{-1}. \tag{5.48}$$

The definition of the CIE, including the filtering scheme in Figure 5.12, essentially involves four types of operations: (i) initialization; (ii) prediction; (iii) measurement updates; and (iv) track fusion. Next, the condition in (5.47) is shown using the following argument based on the principle of induction: If $\mathrm{E}(\tilde{\delta}_{k|l}\tilde{\gamma}_{k|l}^{\mathsf{T}}) = 0$ is true *both before and after* any of the operations (i)–(iv), then $\mathrm{E}(\tilde{\delta}_{k|l}\tilde{\gamma}_{k|l}^{\mathsf{T}}) = 0$ holds for all $k \geq l, l \geq 0$ given that *only* the operations (i)–(iv) according to above are used.

Start with (i), where it can be seen that $\hat{\gamma}_0 = \hat{x}_0$ and $P_0 = \mathrm{E}(\tilde{x}_0\tilde{x}_0^{\mathsf{T}})$ yield $\mathrm{E}(\tilde{x}_0\hat{\gamma}_0^{\mathsf{T}}) = \mathrm{E}(\tilde{x}_0\tilde{x}_0^{\mathsf{T}}) = P_0$. This corresponds to

$$\hat{x}_0 = P_0\left((P_0^e)^{-1}\hat{x}_0^e + \Gamma_0^{-1}\hat{\gamma}_0\right), \qquad P_0 = \left((P_0^e)^{-1} + \Gamma_0^{-1}\right)^{-1},$$

where $\mathrm{E}(\tilde{x}_0^e\tilde{\gamma}_0^{\mathsf{T}}) = 0$, $(P_0^e)^{-1}$ is exclusive information, and $\Gamma_0 \succ P_0$ by assumption. As a consequence $\mathrm{E}(\tilde{\delta}_0\tilde{\gamma}_0^{\mathsf{T}}) = 0$.

For the prediction step (ii), assume w.l.o.g. that

$$\hat{x}_{k|k} = P_{k|k}\left((P_{k|k}^e)^{-1}\hat{x}_{k|k}^e + \Gamma_{k|k}^{-1}\hat{\gamma}_{k|k}\right), \qquad P_{k|k} = \left((P_{k|k}^e)^{-1} + \Gamma_{k|k}^{-1}\right)^{-1}, \tag{5.49}$$

where $\mathrm{E}(\tilde{x}_{k|k}^e\hat{\gamma}_{k|k}^{\mathsf{T}}) = 0$ such that $\mathrm{E}(\tilde{x}_{k|k}\tilde{\gamma}_{k|k}^{\mathsf{T}}) = P_{k|k}$. The SSM in (3.1) implies

$$\tilde{\gamma}_{k+1|k} = \hat{\gamma}_{k+1|k} - x_{k+1} = F_k\hat{\gamma}_{k|k} - (F_k x_k + w_k) = F_k\tilde{\gamma}_{k|k} - w_k,$$

and similarly $\tilde{x}_{k+1|k} = F_k\tilde{x}_{k|k} - w_k$. Since $\mathrm{E}(\tilde{x}_{k|k}w_k^{\mathsf{T}}) = \mathrm{E}(\tilde{\gamma}_{k|k}w_k^{\mathsf{T}}) = 0$ it holds that

$$\mathrm{E}(\tilde{x}_{k+1|k}\tilde{\gamma}_{k+1|k}^{\mathsf{T}}) = \mathrm{E}\left((F_k\tilde{x}_{k|k} - w_k)(F_k\tilde{\gamma}_{k|k} - w_k)^{\mathsf{T}}\right) = F_kP_{k|k}F_k^{\mathsf{T}} + Q_k = P_{k+1|k}.$$

Moreover, since $\mathrm{E}(\tilde{x}_{k|k} w_k^{\mathsf{T}}) = \mathrm{E}(\tilde{\gamma}_{k|k} w_k^{\mathsf{T}}) = 0$

$$\begin{aligned}
&\mathrm{E}(\tilde{\delta}_{k+1|k}\tilde{\gamma}_{k+1|k}) \\
&\quad = \mathrm{E}\left(\Delta_{k+1|k}\left(P_{k+1|k}^{-1}\tilde{x}_{k+1|k} - \Gamma_{k+1|k}^{-1}\tilde{\gamma}_{k+1|k}\right)\tilde{\gamma}_{k+1|k}^{\mathsf{T}}\right) \\
&\quad = \mathrm{E}\left(\Delta_{k+1|k}\left(P_{k+1|k}^{-1}(F_k\tilde{x}_{k|k} - w_k)\Gamma_{k+1|k}^{-1}(F_k\tilde{\gamma}_{k|k} - w_k)\right)(F_k\tilde{\gamma}_{k|k} - w_k)^{\mathsf{T}}\right),
\end{aligned}$$

such that

$$\begin{aligned}
\mathrm{E}(\tilde{\delta}_{k+1|k}\tilde{\gamma}_{k+1|k}) &= \Delta_{k+1|k}\left(P_{k+1|k}^{-1}(F_k P_{k|k} F_k^{\mathsf{T}} + Q_k)\Gamma_{k+1|k}^{-1}(F_k\Gamma_{k|k}F_k^{\mathsf{T}} + Q_k)\right) \\
&= \Delta_{k+1|k}\left(P_{k+1|k}^{-1}P_{k+1|k} - \Gamma_{k+1|k}^{-1}\Gamma_{k+1|k}\right) = 0.
\end{aligned}$$

This means that $\hat{\delta}_{k+1|k}$ and $\hat{\gamma}_{k+1|k}$ are fully uncorrelated.

For the measurement update (iii) and the fusion step (iv), let superscript f denote the result after a measurement update or a fusion update. By assumption, both (iii) and (iv) only involve merging of uncorrelated information, these operations are given as

$$(P_{k|k}^f)^{-1}\hat{x}_{k|k}^f = P_{k|l}^{-1}\hat{x}_{k|l} + A^{-1}\hat{a}, \qquad (P_{k|k}^f)^{-1} = P_{k|l}^{-1} + A^{-1},$$

and

$$(\Gamma_{k|k}^f)^{-1}\hat{\gamma}_{k|k}^f = \Gamma_{k|l}^{-1}\hat{\gamma}_{k|l} + B^{-1}\hat{b}, \qquad (\Gamma_{k|k}^f)^{-1} = \Gamma_{k|l}^{-1} + B^{-1},$$

where $P_{k|l}^{-1}\hat{x}_{k|l} = (P_{k|l}^e)^{-1}\hat{x}_{k|l}^e + \Gamma_{k|l}^{-1}\hat{\gamma}_{k|l}$ and $P_{k|l}^{-1} = (P_{k|l}^e)^{-1} + \Gamma_{k|l}^{-1}$. Hence the common information decomposition is preserved directly and all correlations are removed by the decorrelation step. For instance, in (iii), $A^{-1} \succeq 0$ and $B^{-1} = 0$. Since $\mathrm{E}(\tilde{a}(\tilde{\gamma}_{k|k}^f)^{\mathsf{T}}) = 0$ it follows that

$$\mathrm{E}(\tilde{\delta}_{k|k}^f(\tilde{\gamma}_{k|k}^f)^{\mathsf{T}}) = \mathrm{E}\left(\Delta_{k|l}^f\left((P_{k|l}^e)^{-1}\hat{x}_{k|l}^e + A^{-1}\tilde{a}\right)(\tilde{\gamma}_{k|k}^f)^{\mathsf{T}}\right) = 0.$$

where $\tilde{\delta}_{k|k}^f = \hat{\delta}_{k|k}^f - x_k$, and $(\hat{\delta}_{k|k}^f, \Delta_{k|k}^f)$ is defined analogous to $(\hat{\delta}_{k|k}, \Delta_{k|k})$ in (5.48). Theorem 5.17 summarizes the decorrelation analysis.

Theorem 5.17. *Assume $(\hat{x}_{k|l}, P_{k|l})$ and $(\hat{\gamma}_{k|l}, \Gamma_{k|l})$ are initialized and computed according to Figure 5.12. Let $(\hat{\delta}_{k|l}, \Delta_{k|l})$ be computed according to* (5.48)*. Let $\tilde{\delta}_{k|l} = \hat{\delta}_{k|l} - x_k$ and $\tilde{\gamma}_{k|l} = \hat{\gamma}_{k|l} - x_k$. Then*

$$\mathrm{E}(\tilde{\delta}_{k|l}\tilde{\gamma}_{k|l}^{\mathsf{T}}) = 0,$$

for all $k \geq l, l \geq 0$.

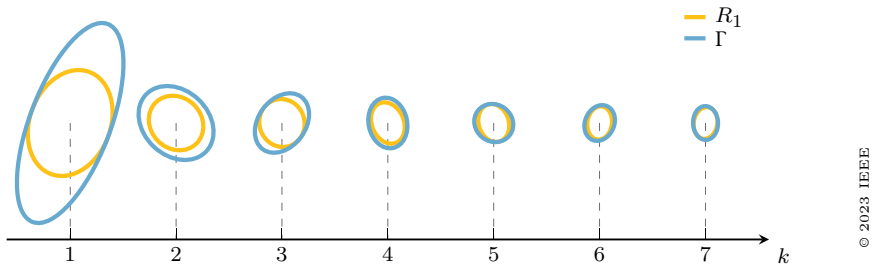


Figure 5.13. One example of convergence of the CIE. Two agents estimate the position of a target using sensor measurements and exchanged DR estimates. The ellipses for R_1, the local estimate at Agent 1, and Γ, the CIE covariance at Agent 2, are plotted for different k. For this case, Γ approaches R_1 relatively fast.

Convergence Example

The applicability of the CIE is related to how well Γ approximates R_1. Next, convergence properties of Γ are illustrated using a simple example. Let $x \in \mathbb{R}^2$ be stationary. Two agents estimate x using sensors that measure both components of x with constant measurement covariances. Since x is stationary, $F_k = I$ and $Q_k = 0$. The example is simulated for time points $k = 1, 2, \ldots, 7$. At odd k, Agent 1 transmits a DR estimate to Agent 2, and at even k, Agent 2 transmits a DR estimate to Agent 1. KF is used for fusion, and GEVO-KF is used for computing Ψ. The resulting ellipses for R_1 and Γ for different k are illustrated in Figure 5.13. In this case, Γ converges to approximately R_1 relatively fast.

Summary

To circumvent the issue of not having access to R_1, the CIE $(\hat{\gamma}, \Gamma)$ has been proposed. The benefits of $(\hat{\gamma}, \Gamma)$ are twofold: (i) it replaces (y_1, R_1) such that it becomes possible to compute Ψ in a decentralized context; and (ii) it can be used along with a decorrelation procedure for GEVO-KF and KF to be applied without double counting of information. Theorem 5.17 states that (y_Ψ, R_Ψ) computed in this way and transmitted to another Agent i, is uncorrelated with $(\hat{\gamma}, \Gamma)$. This means that (y_Ψ, R_Ψ) is uncorrelated with previously transmitted information which is crucial for using KF to fuse (y_1, R_2) with (y_Ψ, R_Ψ). However, since $(\hat{\gamma}, \Gamma)$ is not equivalent to (y_1, R_1), the computed Ψ is in general suboptimal.

5.4.2 Numerical Evaluation

The GEVO framework with CIE is now evaluated in a DTT scenario. The scenario is identical to the one used in the DCA evaluation in Section 3.3.4.

Simulation Specifications

The scenario is illustrated in Figure 5.14. Two agents track a common target in $d = 2$ spatial dimensions. The CAM in (3.6) is assumed for the dynamics, and

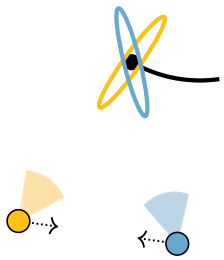


Figure 5.14. Scenario used in the numerical evaluation. The agents are placed at fixed locations $(-2\,000, 1\,000)$ m and $(5\,000, 0)$ m. The target is initially located at $(3000, 8000)$ m, represented by the black circle, and moves along the black trajectory. The ellipse represent the measurement error covariance of each sensor.

Table 5.6
PARAMETERS USED IN THE SIMULATIONS

Parameter	Comment
$d = 2$	spatial dimensionality
$n_x = 6$	state dimensionality
$T_s = 1$	sampling time [s]
$\sigma_w = 2$	standard deviation of process noise [$\text{ms}^{-\frac{5}{2}}$]
$\sigma_r = 1\,000$	standard deviation of radial measurement noise [m]
$\sigma_\theta = 1$	standard deviation of azimuth measurement noise [°]
$(-2000, 1000)$	Agent 1 location [m]
$(5000, 0)$	Agent 2 location [m]
$(3000, 8000)$	target initial position [m]
$n_k = 18$	number of time steps
$M = 10\,000$	number of MC runs

the sensors are defined according to the nonlinear model in (3.8). The agents communicate their local tracks according to the scheme in Rule 3.6. Relevant simulation parameters are summarized in Table 5.6.

CIE is evaluated in combination with the following methods:

- dKF: The Kalman fuser as defined in (5.5), where Ψ is derived using Algorithm 5.11. The decorrelation step described in Section 5.4.1 is used to decorrelate exchanged estimates.
- CI: Covariance intersection as defined in (5.6), where Ψ is derived using Algorithm 5.12.
- LE: The largest ellipsoid method as defined in (5.8), where Ψ is derived using Algorithm 5.14.

These methods are compared to:

- LKF: A local EKF for which no tracks are shared.
- CRLB: In the RMSE plots, CRLB is represented by $\sqrt{\text{tr}(P^0_{\text{pos}})}$, where P^0_{pos} is computed as in Section 3.2.1.

- DCA-EIG: The eigenvalue based scaling method proposed in Section 3.3.2, where CI is used for track fusion.

Evaluation Measures

RMSE, RMT, and ANEES are used for evaluation. In addition, a variant of RMTR is used to compare the relative performance of using GEVO with CIE to using GEVO with globally available knowledge[9]. Consider a certain agent and method. Let $P_k^i(\Gamma)$ denote the covariance computed in the ith MC run at time k using CIE. Likewise, let $P_k^i(R_j)$ denote the covariance computed in the ith MC run at time k using global knowledge about R_j. The *RMTR-LTG* is defined as

$$\text{RMTR-LTG}_k = \frac{\sqrt{\frac{1}{M}\sum_{i=1}^{M} \text{tr}(P_k^i(\Gamma))}}{\sqrt{\frac{1}{M}\sum_{i=1}^{M} \text{tr}(P_k^i(R_j))}}. \tag{5.50}$$

RMSE, RMT, and RMTR-LTG are computed for the position components of the error and covariance. ANEES is computed for the full state.

Results

The results for Agent 1 are shown in Figure 5.15[10]. RMT and RMSE are normalized by σ_r.

The RMTR-LTG plot indicates that the relative performance is close to 1 in all cases, and in particular in the case of dKF. Hence, the performance degradation when using CIE is small. In fact, for LE and $m = 1$, the performance w.r.t. RMT is slightly improved when using CIE. However, this is not a general result. LE and dKF approach the CRLB in terms of both the RMT and the RMSE. CI performs worse than dKF and LE, but better than DCA-EIG for all m.

CI and DCA-EIG are conservative w.r.t. ANEES. LE and dKF are slightly too optimistic. The latter is a consequence of the fact that the decorrelation procedure described in Section 5.4.1 cannot fully cope with common process noise.

Remark 5.18. In Section 5.3.5 it is stated that Ψ computed by GEVO-LE is optimal if R_1 it known. Hence, the RMTR-LTG plot seems like a counterexample to this statement. However, Ψ computed by GEVO-LE is optimal w.r.t. $\text{tr}(P)$ for the full state, but the RMTR-LTG plot only visualizes the position components.

5.5 Dimension-Reduction for Association Quality

Figure 5.16 illustrates a multitarget tracking scenario. Agent 2 communicates DR versions of its local tracks to Agent 1. Before Agent 1 can fuse the received DR estimates, Agent 1 needs to associate them with its local tracks. So far, the *track-to-track association* has been neglected. However, reducing dimensionality

[9] Globally available knowledge in the sense that, in each time step, Agent 2 has access to R_1 and Agent 1 has access to R_2.

[10] The results for Agent 2 are similar.

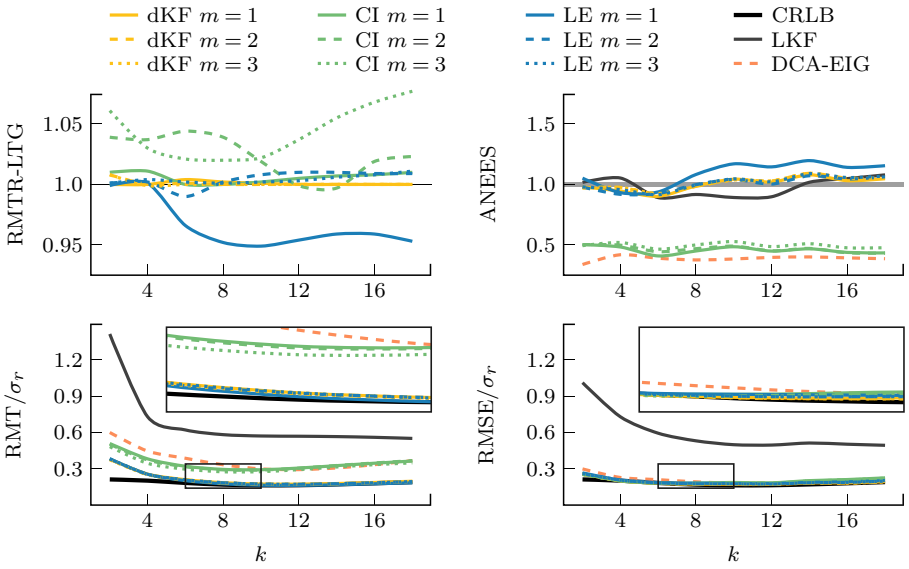


Figure 5.15. Results from the CIE evaluation. RMSE, RMT, and RMTR-LTG are computed for the position components of the error and covariance. ANEES is computed for the full state. The gray area represents an ANEES 99.9% confidence interval.

might have a severe negative effect on the association quality. In Section 5.3, the communication management is designed for enhanced track fusion performance. In this section, the problem is to design the communication management to promote track-to-track association quality.

5.5.1 Problem Model

Assume that $N = 2$ and that there are n_t targets. Targets and estimates are distinguished by subscript (i), e.g., the state of the ith target is $x_{(i)} \in \mathbb{R}^{n_x}$.

Multitarget Estimation Model

Let

$$y_{1(i)} = x_{(i)} + v_{1(i)}, \qquad \boldsymbol{v}_{1(i)} \sim \mathcal{N}(0, R_{1(i)}), \tag{5.51a}$$

$$y_{2(i)} = x_{(i)} + v_{2(i)}, \qquad \boldsymbol{v}_{2(i)} \sim \mathcal{N}(0, R_{2(i)}), \tag{5.51b}$$

be the local estimates of $x_{(i)}$ in Agent 1 and Agent 2, respectively. For instance, $y_{1(i)}$ is the state estimate and $R_{1(i)}$ the corresponding covariance of the ith target in Agent 1. All cross-covariances $R_{12(i)} = \mathrm{E}(\boldsymbol{v}_{1(i)}\boldsymbol{v}_{2(i)}^{\mathsf{T}})$ are assumed to be zero[11].

[11] It is assumed that estimates have been decorrelated before they are communicated. For instance, by using the techniques in [169] or Section 5.4.1.

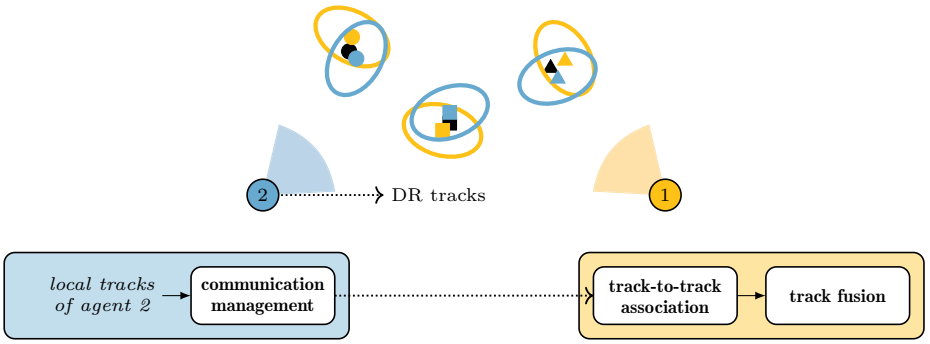


Figure 5.16. *Top:* Multitarget tracking scenario, where Agent 2 communicates DR tracks to Agent 1. *Bottom:* Before the track fusion step, the received DR tracks must be associated with the local tracks.

The ith DR estimate is given by

$$y_{\Psi(i)} = \Psi_{(i)} y_{2(i)}, \qquad R_{\Psi(i)} = \Psi_{(i)} R_{2(i)} \Psi_{(i)}^{\mathsf{T}}, \tag{5.52}$$

where $\Psi_{(i)} \in \mathbb{R}^{m \times n_x}$ with $m < n_x$ and $\mathsf{rank}(\Psi_{(i)}) = m$.

The sets of local tracks of Agent 1 and Agent 2 are, respectively, given by

$$\mathcal{Y}_1 = \left\{ (y_{1(1)}, R_{1(1)}), \ldots, (y_{1(n_t)}, R_{1(n_t)}) \right\}, \tag{5.53a}$$

$$\mathcal{Y}_2 = \left\{ (y_{2(1)}, R_{2(1)}), \ldots, (y_{2(n_t)}, R_{2(n_t)}) \right\}. \tag{5.53b}$$

Agent 1 and Agent 2 track exactly the same targets and hence have the same number of tracks. Moreover, it is assumed that the elements of $\mathcal{Y}_1$ and $\mathcal{Y}_2$ are labeled according to $x_{(1)}, \ldots, x_{(n_t)}$, e.g., $(y_{1(i)}, R_{1(i)})$ and $(y_{2(i)}, R_{2(i)})$ are estimates of the same target $x_{(i)}$. This might sound a bit counterintuitive w.r.t. the normal association problem. However, the assumption is not a restriction in this case since here the actual correct association result is known, as described later[12], and the task is to compute $\Psi_{(1)}, \ldots, \Psi_{(n_t)}$. Let

$$\mathcal{Y}_\Psi = \left\{ (y_{\Psi(1)}, R_{\Psi(1)}), \ldots, (y_{\Psi(n_t)}, R_{\Psi(n_t)}) \right\}. \tag{5.54}$$

Since $R_{12(i)} = 0$, $(y_{1(i)}, R_{1(i)})$ and $(y_{\Psi(i)}, R_{\Psi(i)})$ are optimally fused using KF, which in this case yields

$$\hat{x}_{(i)} = P_{(i)} \left(R_{1(i)}^{-1} y_{1(i)} + \Psi_{(i)}^{\mathsf{T}} R_{\Psi(i)}^{-1} y_{\Psi(i)} \right), \quad P_{(i)} = \left(R_{1(i)}^{-1} + \Psi_{(i)}^{\mathsf{T}} R_{\Psi(i)}^{-1} \Psi_{(i)} \right)^{-1}. \tag{5.55a}$$

[12] In short, the correct association result can be assumed to be known in the proposed solution since there Agent 2 approximates $\mathcal{Y}_1$ by $\mathcal{Y}_2$, where $\mathcal{Y}_2$ of course is fully known to Agent 2.

Association Model

The association problem is formulated as a linear assignment problem [30]. In case of full estimates, the assignment matrix is

$$\mathcal{M}_{\text{full}} = \begin{bmatrix} d^2_{(11)} & \cdots & d^2_{(1n_t)} \\ \vdots & \ddots & \vdots \\ d^2_{(n_t1)} & \cdots & d^2_{(n_tn_t)} \end{bmatrix}, \tag{5.56}$$

where $d^2_{(ij)}$ is a *Mahalanobis distance* (MD) given by

$$d^2_{(ij)} = \bar{y}^{\mathsf{T}}_{(ij)} S^{-1}_{(ij)} \bar{y}_{(ij)}, \qquad \bar{y}_{(ij)} = y_{1(i)} - y_{2(j)}, \qquad S_{(ij)} = R_{1(i)} + R_{2(j)}. \tag{5.57}$$

Similarly, the DR assignment matrix $\mathcal{M}_{\text{dr}}$ is defined as

$$\mathcal{M}_{\text{dr}} = \begin{bmatrix} r^2_{(11)} & \cdots & r^2_{(1n_t)} \\ \vdots & \ddots & \vdots \\ r^2_{(n_t1)} & \cdots & r^2_{(n_tn_t)} \end{bmatrix}, \tag{5.58}$$

where $r^2_{(ij)}$ is an MD given by

$$\begin{aligned} r^2_{(ij)} &= (\Psi_{(j)} y_{1(i)} - y_{\Psi(j)})^{\mathsf{T}} \left(\Psi_{(j)} R_{1(i)} \Psi^{\mathsf{T}}_{(j)} + R_{\Psi(j)} \right)^{-1} (\Psi_{(j)} y_{1(i)} - y_{\Psi(j)}) \\ &= \bar{y}^{\mathsf{T}}_{(ij)} \Psi^{\mathsf{T}}_{(j)} \left(\Psi_{(j)} S_{(ij)} \Psi^{\mathsf{T}}_{(j)} \right)^{-1} \Psi_{(j)} \bar{y}_{(ij)}. \end{aligned} \tag{5.59}$$

Agent 1 receives estimates from Agent 2 and solves the association problem using *global nearest neighbor* (GNN) association. Let $\mathbb{P}^{n_t}$ be the set of all $n_t \times n_t$ permutation matrices, i.e.,

$$\mathbb{P}^{n_t} = \left\{ \Pi \in \mathbb{R}^{n_t \times n_t} \;\middle|\; [\Pi]_{ij} \in \{0,1\}, \Pi\Pi^{\mathsf{T}} = I \right\}. \tag{5.60}$$

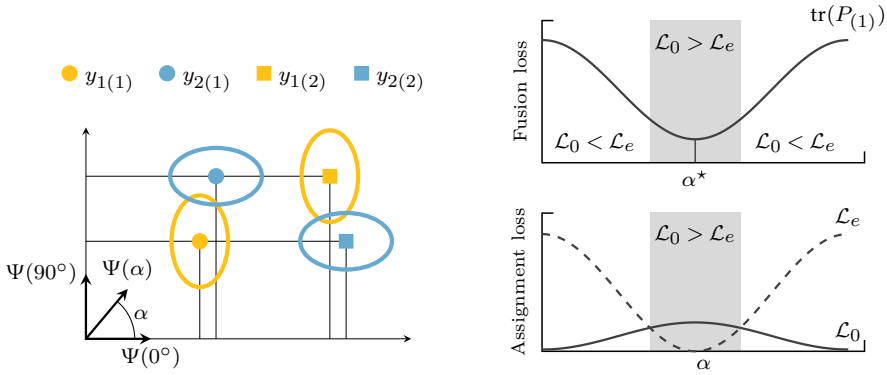
A permutation matrix $\Pi \in \mathbb{P}^{n_t}$ assigns exactly one estimate in $\mathcal{Y}_1$ to each of the estimates in $\mathcal{Y}_\Psi$. The optimal Π for a certain assignment matrix $\mathcal{M}$ is computed using [30]

$$\begin{aligned} \underset{\Pi}{\text{minimize}} \quad & \text{tr}(\Pi\mathcal{M}) \\ \text{subject to} \quad & \Pi \in \mathbb{P}^{n_t}. \end{aligned} \tag{5.61}$$

Note, in this formulation correct assignment is given by $\Pi_0 = I$.

Motivating Example

As an example of how $\Psi_{(i)}$ affects the association performance, consider the scenario in Figure 5.17a, where $n_t = 2$, $n_x = 2$, and $m = 1$. Each agent has a local estimate of each of the two targets as defined in Figure 5.17a, where $R_{1(1)} = R_{1(2)}$



(a) Target tracking scenario

(b) Fusion and assignment loss functions

Figure 5.17. Two agents estimating two targets. The scenario is illustrated in (a). Projections of the state estimates along Ψ evaluated at $\alpha = 0°$ and $\alpha = 90°$ are illustrated in (a). The fusion loss function $\mathrm{tr}(P_{(i)})$ and the assignment loss function $\mathrm{tr}(\Pi\mathcal{M}_{\mathrm{dr}})$ are shown in (b).

and $R_{2(1)} = R_{2(2)}$. Assume $\Psi_{(1)} = \Psi_{(2)} = \Psi = \begin{bmatrix}\cos\alpha & \sin\alpha\end{bmatrix}$, where α is an angle. Based on this parametrization, it is possible to define $\mathcal{M}_{\mathrm{dr}}$ as a function of α. Let

$$\mathcal{L}_0 = \mathrm{tr}(\Pi\mathcal{M}_{\mathrm{dr}})\big|_{\Pi=\Pi_0} = r_{(11)}^2 + r_{(22)}^2, \quad \mathcal{L}_e = \mathrm{tr}(\Pi\mathcal{M}_{\mathrm{dr}})\Big|_{\Pi=\left[\begin{smallmatrix}0 & 1\\ 1 & 0\end{smallmatrix}\right]} = r_{(12)}^2 + r_{(21)}^2,$$

be the cost corresponding to correct and incorrect assignments, respectively. By construction, $\mathcal{L}_0$, $\mathcal{L}_e$, and $\mathrm{tr}(P_{(1)}) = \mathrm{tr}(P_{(2)})$ are functions of α.

The fusion and association performance w.r.t. α is evaluated by computing $\mathcal{L}_0$, $\mathcal{L}_e$, and $\mathrm{tr}(P_{(i)})$ for each $\alpha \in [0°, 180°]$. The results are shown in Figure 5.17b. The fusion optimal Ψ corresponds to $\alpha^\star = 90°$. However, this Ψ lies in the interval where $\mathcal{L}_0 > \mathcal{L}_e$, which would imply incorrect assignment. To have correct assignment in the dimension-reduced case while maintaining good fusion performance, the selected Ψ should be such that it minimizes $\mathrm{tr}(P_{(i)})$ subject to $\mathcal{L}_0 < \mathcal{L}_e$.

Problem Formalization

Assume the targets $x_{(1)}, \ldots, x_{(n_t)}$ are sufficiently separated such that solving the assignment problem in (5.61) with $\mathcal{M} = \mathcal{M}_{\mathrm{full}}$ yields Π_0. Moreover, assume that Agent 2 has no knowledge about $\mathcal{Y}_1$. The problem is for Agent 2 to compute $\Psi_{(1)}^{\mathsf{T}}, \ldots, \Psi_{(n_t)}^{\mathsf{T}} \in \mathbb{R}^{n_x}$ such that when Agent 1 solves (5.61) with $\mathcal{M} = \mathcal{M}_{\mathrm{dr}}$, the solution Π is as close as possible to Π_0. In other words, since it in general is not possible to obtain the correct association in the dimension-reduced case, it is desirable to compute $\Psi_{(1)}, \ldots, \Psi_{(n_t)}$ in such a way that the association is not degraded too much. The focus is on the case $m = 1$. However, some of the results are given for arbitrary $m \geq 1$.

Remark 5.19. The considered problem is not the common association problem of DTT, where received tracks are associated with local tracks and the correct assignment Π_0 is *unknown*. Here, the correct assignment is *known* by construction, and hence, there is the freedom of defining $\mathcal{M}_{\text{full}}$ and $\mathcal{M}_{\text{dr}}$ such that $\Pi_0 = I$.

5.5.2 Problem Analysis

Properties of the considered association problem are examined. Sufficient conditions for correct assignment are given. An example is used to show that the problem is further complicated by inherent randomness. Statistical properties of the problem are derived at the end to be used in the subsequent section.

A Sufficient Condition for Correct Assignment

Consider now an oracle's perspective. The motivating example of Section 5.5.1 illustrates an important property of the problem. That is, for $\Psi_{(j)} \neq 0$ and $\bar{y}_{(ij)} \neq 0$

$$\Psi_{(j)} \perp \bar{y}_{(ij)}^{\mathsf{T}} \iff \Psi_{(j)} \bar{y}_{(ij)} = 0,$$

where $\bar{y}_{(ij)} = y_{1(i)} - y_{2(j)}$. From this it can be inferred that for the association it is desirable to have

$$\Psi_{(j)} \bar{y}_{(jj)} = 0 \quad \wedge \quad i \neq j \implies \Psi_{(j)} \bar{y}_{(ij)} \neq 0, \tag{5.62}$$

since in this case $r^2_{(jj)} = 0$ and $r^2_{(ij)} > 0$ if $i \neq j$. A sufficient condition for correct assignment is hence that (5.62) holds for all j as this would imply $\mathsf{tr}(\mathcal{M}_{\text{dr}}) = 0$. However, by assumption Agent 2 has no knowledge about $\mathcal{Y}_1$ and hence without further knowledge Agent 2 cannot compute $\Psi_{(j)}$ such that (5.62) is satisfied.

Problem Properties

In the example of the previous section the fusion optimal Ψ gave incorrect association. Luckily, it is not generally the case that the fusion optimal Ψ yields incorrect assignments. Unfortunately, it is impossible to say something general about tradeoffs between fusion and association performance. The main reasons for this are described below.

Consider $\Psi_{(j)}^{\mathsf{T}} \in \mathbb{R}^{n_x}$, and let $Q_{(j)} = R^2_{1(j)} \in \mathbb{S}^{n_x}_{++}$ and $S_{(jj)} = R_{1(j)} + R_{2(j)} \in \mathbb{S}^{n_x}_{++}$. In the fusion case the optimal $\Psi_{(j)}$ solves [57]

$$\underset{\Psi_{(j)}}{\text{maximize}} \quad \frac{\Psi_{(j)} Q_{(j)} \Psi_{(j)}^{\mathsf{T}}}{\Psi_{(j)} S_{(jj)} \Psi_{(j)}^{\mathsf{T}}}. \tag{5.63}$$

Hence the fusion optimal $\Psi_{(j)}$ for a certain target $x_{(j)}$ can be solved isolated from the other targets. This is not true in the association problem where optimal $\Psi_{(j)}$ for a certain target $x_{(j)}$ depends on all estimates in both $\mathcal{Y}_1$ and $\mathcal{Y}_2$ through $\mathcal{M}_{\text{dr}}$.

A slightly less restrictive sufficient condition for correct assignment, cf. (5.62), is that for each j

$$r^2_{(jj)} < r^2_{(ij)}, \quad \forall i \neq j. \tag{5.64}$$

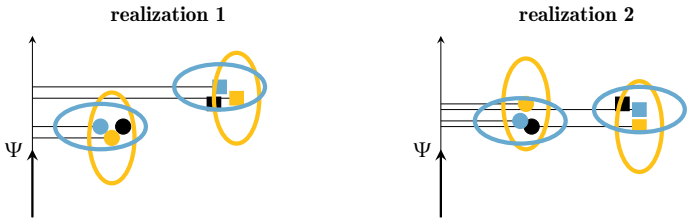


Figure 5.18. Two noise realizations of the same scenario. In realization 1 correct assignment is obtained while in realization 2 incorrect assignment is obtained.

If this condition holds, then nearest neighbor [27] association yields the same results as GNN association. The condition in (5.64) can also be expressed as

$$\frac{\Psi_{(j)}\bar{y}_{(jj)}\bar{y}_{(jj)}^{\mathsf{T}}\Psi_{(j)}^{\mathsf{T}}}{\Psi_{(j)}S_{(jj)}\Psi_{(j)}^{\mathsf{T}}} < \frac{\Psi_{(j)}\bar{y}_{(ij)}\bar{y}_{(ij)}^{\mathsf{T}}\Psi_{(j)}^{\mathsf{T}}}{\Psi_{(j)}S_{(ij)}\Psi_{(j)}^{\mathsf{T}}}, \quad \forall i \neq j, \tag{5.65}$$

where each fraction is structurally similar to the fraction in (5.63). However, a complication compared to the fusion case is that $r_{(ij)}^2$ is a realization of a random variable

$$\boldsymbol{r}_{(ij)}^2 = \bar{\boldsymbol{y}}_{(ij)}^{\mathsf{T}}\Psi_{(j)}^{\mathsf{T}}\left(\Psi_{(j)}S_{(ij)}\Psi_{(j)}^{\mathsf{T}}\right)^{-1}\Psi_{(j)}\bar{\boldsymbol{y}}_{(ij)},$$

where $\bar{\boldsymbol{y}}_{(ij)} = \boldsymbol{y}_{1(i)} - \boldsymbol{y}_{2(j)}$. Hence, assuming that Agent 2 has access to $R_{1(i)}$ and a good estimate of $x_{(i)}$, the fusion optimal $\Psi_{(i)}$ could be computed while it would still be difficult to predict $r_{(ij)}^2$ due to randomness. Figure 5.18 shows two possible realizations of each of the random variables

$$\begin{aligned} \boldsymbol{y}_{1(1)} &= x_{(1)} + \boldsymbol{v}_{1(1)}, & \boldsymbol{y}_{2(1)} &= x_{(1)} + \boldsymbol{v}_{2(1)}, \\ \boldsymbol{y}_{1(2)} &= x_{(2)} + \boldsymbol{v}_{1(2)}, & \boldsymbol{y}_{2(2)} &= x_{(2)} + \boldsymbol{v}_{2(2)}, \end{aligned}$$

where $\boldsymbol{v}_{1(1)}, \boldsymbol{v}_{1(2)} \sim \mathcal{N}(0, R_1)$ and $\boldsymbol{v}_{2(1)}, \boldsymbol{v}_{2(2)} \sim \mathcal{N}(0, R_2)$. Since the covariances are the same in each case and since by assumption $R_{1(1)} = R_{1(2)} = R_1$ and $R_{2(1)} = R_{2(2)} = R_2$ the fusion optimal $\Psi_{(j)}$ satisfy $\Psi_{(1)} = \Psi_{(2)} = \Psi$ in both cases. Computing $\mathcal{M}_{\mathrm{dr}}(\Psi)$ in realization 1 and realization 2 yields

$$\mathcal{M}_1 = \begin{bmatrix} 0.05 & 1.01 \\ 0.31 & 0.05 \end{bmatrix}, \qquad \mathcal{M}_2 = \begin{bmatrix} 0.11 & 0.01 \\ 0.01 & 0.11 \end{bmatrix},$$

respectively. In realization 1 correct assignment Π_0 is obtained while in realization 2 the incorrect combination is chosen. The example illustrates that, due to the inherent randomness, it is in general impossible to decide if a fusion optimal $\Psi_{(j)}$ will imply correct or incorrect assignment without knowing the actual realization.

Statistical Properties

Assume $m \geq 1$. By construction

$$\Psi_{(j)}\bar{\boldsymbol{y}}_{(ij)} \sim \mathcal{N}\left(\Psi_{(j)}\bar{x}_{(ij)}, \Psi_{(j)}S_{(ij)}\Psi_{(j)}^{\mathsf{T}}\right),$$

where $\bar{x}_{(ij)} = x_{(i)} - x_{(j)}$. Hence [144]

$$\boldsymbol{r}^2_{(ij)} \sim \begin{cases} \chi^2_m, & \text{if } i = j, \\ \chi^2_{m,\nu}, & \text{if } i \neq j, \end{cases} \tag{5.66}$$

where χ^2_m is the central chi-squared distribution with m degrees of freedom, and $\chi^2_{m,\nu}$ is the noncentral chi-squared distribution, where ν is the noncentrality parameter. The expectation value and the variance are given by [144]

$$\mathrm{E}(\boldsymbol{r}^2_{(ij)}) = m + \nu_{(ij)}, \qquad \mathsf{var}(\boldsymbol{r}^2_{(ij)}) = 2m + 4\nu_{(ij)}, \tag{5.67}$$

with $\nu_{(ij)} = \bar{x}^{\mathsf{T}}_{(ij)} \Psi^{\mathsf{T}}_{(j)} \left(\Psi_{(j)} S_{(ij)} \Psi^{\mathsf{T}}_{(j)} \right)^{-1} \Psi_{(j)} \bar{x}_{(ij)}$. One conclusion is that as $\nu_{(ij)}$ increases the relative effect of randomness decreases since $\mathrm{E}(\boldsymbol{r}^2_{(ij)})$ scales as $\nu_{(ij)}$ while $\sqrt{\mathsf{var}(\boldsymbol{r}^2_{(ij)})}$ only scales as $\sqrt{\nu_{(ij)}}$. This result is important and is used in the solution proposed in the next section.

5.5.3 Preserving Correct Assignment With Dimension-Reduced Estimates

In this section a method for preserving high association quality is suggested. Based on the analysis of Section 5.5.2, an optimization formulation is provided for computation of $\Psi_{(j)}$. This leads to the proposed descent based optimization strategy, where a key ingredient is the adaptive step size.

Approximated Assignment Matrix

The proposed solution is based on the analysis of the previous section. In particular, $r^2_{(ij)}$ is estimated using $\mathrm{E}(\boldsymbol{r}^2_{(ij)})$ in (5.67). To compute $r^2_{(ij)}$ Agent 2 must have access to both $(y_{1(i)}, R_{1(i)})$ and $(y_{2(j)}, R_{2(j)})$, but $(y_{1(i)}, R_{1(i)})$ is unknown to Agent 2. A substitute to $(y_{1(i)}, R_{1(i)})$ that is locally available is $(y_{2(i)}, R_{2(i)})$. Let

$$\hat{r}^2_{(ij)} = \hat{y}^{\mathsf{T}}_{(ij)} \Psi^{\mathsf{T}}_{(j)} \left(\Psi_{(j)} \hat{S}_{(ij)} \Psi^{\mathsf{T}}_{(j)} \right)^{-1} \Psi_{(j)} \hat{y}_{(ij)}, \tag{5.68}$$

where $\hat{y}_{(ij)} = y_{2(i)} - y_{2(j)}$ and $\hat{S}_{(ij)} = R_{2(i)} + R_{2(j)}$ such that $\hat{\boldsymbol{y}}_{(ij)} \sim \mathcal{N}(\bar{x}_{(ij)}, \hat{S}_{(ij)})$. This is consistent with $\boldsymbol{r}^2_{(ij)}$ in the sense that

$$\mathrm{E}(\hat{\boldsymbol{r}}^2_{(ij)}) = \begin{cases} m, & \text{if } i = j, \\ m + \bar{x}^{\mathsf{T}}_{(ij)} \Psi^{\mathsf{T}}_{(j)} \left(\Psi_{(j)} \hat{S}_{(ij)} \Psi^{\mathsf{T}}_{(j)} \right)^{-1} \Psi_{(j)} \bar{x}_{(ij)}, & \text{if } i \neq j. \end{cases}$$

which is identical to (5.67) except that $S_{(ij)}$ is replaced by $\hat{S}_{(ij)}$. Let the *approximated assignment matrix* be defined as

$$\hat{\mathcal{M}}_{\mathrm{dr}} = \begin{bmatrix} \hat{r}^2_{(11)} & \cdots & \hat{r}^2_{(1n_t)} \\ \vdots & \ddots & \vdots \\ \hat{r}^2_{(n_t)} & \cdots & \hat{r}^2_{(n_t n_t)} \end{bmatrix}. \tag{5.69}$$

Proposed Solution

Since only an approximation $\hat{\mathcal{M}}_{\mathrm{dr}}$ of $\mathcal{M}_{\mathrm{dr}}$ is accessible, $\Psi_{(j)}$ is computed based on the sufficient condition in Section 5.5.2, cf. (5.64). This condition is utilized because it is desirable to have some margin when choosing $\Psi_{(j)}$, to avoid that $r^2_{(ij)}$ is zero or very small if $i \neq j$. Moreover, if $\Psi_{(j)}$ satisfies this sufficient condition there is no need to take into account the other $\Psi_{(i)}, i \neq j$ when computing $\Psi_{(j)}$—correct assignment is obtained regardlessly.

Consider now a certain j and $\Psi_{(j)}$. Let

$$f_i(z) = \frac{z^{\mathsf{T}} \hat{y}_{(ij)} z}{z^{\mathsf{T}} \hat{S}_{(ij)} z}, \qquad \hat{Y}_{(ij)} = \hat{y}_{(ij)} \hat{y}_{(ij)}^{\mathsf{T}}, \tag{5.70}$$

be defined for all $i \neq j$. To maximize all $f_i(z)$ simultaneously is in general impossible since this is a *multiobjective optimization* problem. However, it is possible to consider a worst-case approach and maximize the smallest $f_i(z)$. This implies a *maximin* formulation where $\Psi_{(j)}$ is computed using

$$\underset{\Psi_{(j)}}{\text{maximize}} \quad \left(\min_{i \neq j} \quad f_i(\Psi_{(j)}^{\mathsf{T}}) \right). \tag{5.71}$$

The problem in (5.71) is a *nonconvex problem* involving optimization over a finite set of quadratic form ratios. The problem is difficult to solve in general and therefore the following optimization strategy is proposed.

Optimization Strategy

For each individual $f_i(z)$, the z that maximizes $f_i(z)$ is known to be given by the eigenvector u that corresponds to the maximum eigenvalue λ of

$$\hat{Y}_{(ij)} u = \lambda \hat{S}_{(ij)} u. \tag{5.72}$$

As $\hat{Y}_{(ij)} \in \mathbb{S}_+^n$ and $\mathsf{rank}(\hat{Y}_{(ij)}) = 1$ this eigenvalue problem has only one strictly positive eigenvalue λ for which the corresponding eigenvector is denoted by u_i. Since u_i in general differ for different i, it is not possible to maximize all $f_i(z)$ simultaneously. However, for a certain z the values of all $f_i(z)$ are known, and hence it is possible to compute

$$i^* = \underset{i \neq j}{\arg\min} \quad f_i(z). \tag{5.73}$$

To increase f_{i^*} it is suggested that

$$z \leftarrow z + \alpha u_{i^*}, \tag{5.74}$$

where α resembles the step size to traverse along u_{i^*}. Using a too large $|\alpha|$ there is a risk that f_i for some other $i \neq i^*$ is severely decreased. Too small $|\alpha|$ means slow convergence. Consider now Proposition 5.20.

Proposition 5.20. *Let $u, z \in \mathbb{R}^n$, $Y, S \in \mathbb{R}^{n\times n}$, and $f(z) = (z^\mathsf{T} Y z)/(z^\mathsf{T} S z)$, where $z \neq 0$ and $\mathsf{rank}(S) = n$. Then a first-order approximation of $f(z+\alpha u)$, for any scalar α, is given by*

$$f(z+\alpha u) \approx f(z) + 2\alpha \frac{u^\mathsf{T}(Y - f(z)S)z}{z^\mathsf{T} S z}. \tag{5.75}$$

Proof: From [139]

$$\frac{\partial f(z)}{\partial z} = -\frac{2Szz^\mathsf{T}Yz}{(z^\mathsf{T}Sz)^2} + \frac{2Yz}{z^\mathsf{T}Sz} = -\frac{2f(z)Sz}{z^\mathsf{T}Sz} + \frac{2Yz}{z^\mathsf{T}Sz} = 2\frac{(Y - f(z)S)}{z^\mathsf{T}Sz}z.$$

A first-order approximation of $f(z+\alpha u)$ is given by

$$f(z+\alpha u) \approx f(z) + \alpha u^\mathsf{T} \frac{\partial f(z')}{\partial z'}\bigg|_{z'=z} = f(z) + 2\alpha \frac{u^\mathsf{T}(Y - f(z)S)z}{z^\mathsf{T} S z}.$$

□

Proposition 5.20 states that a first-order approximation of f_i evaluated at z in the direction of αu_{i^*} is given by

$$f_i(z+\alpha u_{i^*}) \approx f_i(z) + 2\alpha \frac{u_{i^*}^\mathsf{T}(\hat{Y}_{(ij)} - f_i(z)\hat{S}_{(ij)})z}{z^\mathsf{T}\hat{S}_{(ij)}z}. \tag{5.76}$$

Proceed by solving

$$f_i(z) + 2\alpha \frac{u_{i^*}^\mathsf{T}(\hat{Y}_{(ij)} - f_i(z)\hat{S}_{(ij)})z}{z^\mathsf{T}\hat{S}_{(ij)}z} = f_{i^*}(z) + 2\alpha \frac{u_{i^*}^\mathsf{T}(\hat{Y}_{i^*j} - f_i(z)\hat{S}_{i^*j})z}{z^\mathsf{T}\hat{S}_{i^*j}z}, \tag{5.77}$$

for each $i \neq j, i^*$. This yields $n_t - 2$ solutions for α, where some might be negative and other positive. Since the task is to increase f_{i^*} while not decreasing the other f_i too much, α is chosen such that $|\alpha|$ is the smallest among all those that satisfy

$$\alpha \frac{u_{i^*}^\mathsf{T}(\hat{Y}_{i^*j} - f_i(z)\hat{S}_{i^*j})z}{z^\mathsf{T}\hat{S}_{i^*j}z} > 0. \tag{5.78}$$

This last condition is introduced to ensure that the correct sign is chosen for α.

The operations in (5.73)–(5.78) are performed iteratively until some termination criterion is met. The optimization algorithm is summarized in Algorithm 5.21.

Example

As an example of the proposed optimization strategy, consider a scenario with $n_t = 3$ and $n_x = 4$. Assume $j = 3$. The following two loss functions need to be examined

$$f_1(z) = \frac{z^\mathsf{T}\hat{Y}_{(13)}z}{z^\mathsf{T}\hat{S}_{(13)}z}, \qquad f_2(z) = \frac{z^\mathsf{T}\hat{Y}_{(23)}z}{z^\mathsf{T}\hat{S}_{(23)}z}.$$

Algorithm 5.21: Association Quality Based Dimension-Reduction

Input: $\mathcal{Y}_2$, j, $\alpha_{\min}$, and $\alpha_{\max}$

1. For each $i \neq j$: Let $\hat{Y}_{(ij)} = \hat{y}_{(ij)}\hat{y}_{(ij)}^{\mathsf{T}}$, $\hat{S}_{(ij)} = R_{2(i)} + R_{2(j)}$ and $f_i(z) = (z^{\mathsf{T}}\hat{Y}_{(ij)}z)/(z^{\mathsf{T}}\hat{S}_{(ij)}z)$. Let u_i be the eigenvector corresponding to the maximum eigenvalue λ_i of $\hat{Y}_{(ij)}u = \lambda\hat{S}_{(ij)}u$. Let $k = 0$ and $z_0 \leftarrow z_0/\|z_0\|$, where $z_0 = \sum_{i=1,i\neq j}^{N} \frac{1}{\lambda_i}u_i$.
2. Let $k \leftarrow k+1$. Compute
$$i^* = \underset{i\neq j}{\arg\min} \quad f_i(z_{k-1}).$$
3. For each $i \neq j$: Define
$$\hat{f}_i(z_{k-1} + \alpha u_{i^*}) = f_i(z_{k-1}) + 2\alpha \frac{u_{i^*}^{\mathsf{T}}(\hat{Y}_{(ij)} - f_i(z_{k-1})\hat{S}_{(ij)})z_{k-1}}{z_{k-1}^{\mathsf{T}}\hat{S}_{(ij)}z_{k-1}}.$$
4. For each $i \neq j, i^*$: Solve for α in $\hat{f}_i = \hat{f}_{i^*}$. Store the different α in a vector a.
5. If
$$\alpha \frac{u_{i^*}^{\mathsf{T}}(\hat{Y}_{i^*j} - f_i(z_{k-1})\hat{S}_{i^*j})z_{k-1}}{z_{k-1}^{\mathsf{T}}\hat{S}_{i^*j}z_{k-1}} > 0,$$
then let α_k be given by the minimum positive element of a. Otherwise, let α_k be given by the maximum negative element of a. If $|\alpha_k| < \alpha_{\min}$, then let $\alpha_k \leftarrow \mathsf{sign}(\alpha_k)\alpha_{\min}$. If $|\alpha_k| > \alpha_{\max}$, then let $\alpha_k \leftarrow \mathsf{sign}(\alpha_k)\alpha_{\max}$.
6. Let $z_k \leftarrow z_k/\|z_k\|$, where $z_k = z_{k-1} + \alpha_k u_{i^*}$.
7. Terminate with $\Psi_{(j)} = z_k^{\mathsf{T}}$ if a predefined stopping criterion is met. Otherwise, go back to step 2.

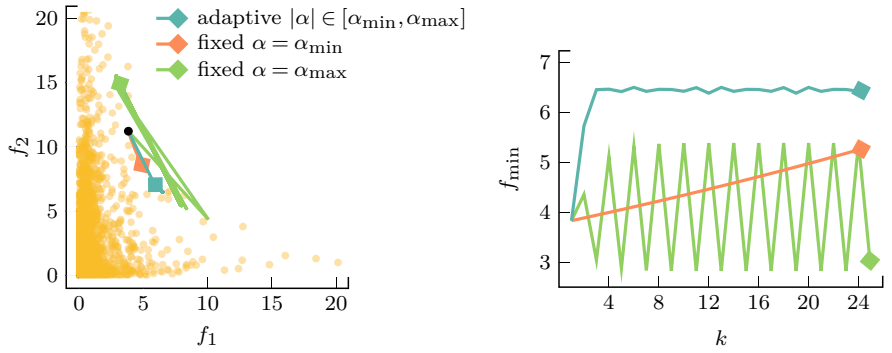
Output: $\Psi_{(j)}$

The multiobjective problem of maximizing f_1 and f_2 simultaneously is not solvable, and therefore Algorithm 5.21 is used. The original Algorithm 5.21 uses an adaptive step size $|\alpha| \in [\alpha_{\min}, \alpha_{\max}]$. This is compared to the same algorithm with: (i) a small fixed step size $\alpha = \alpha_{\min}$; and (ii) a large fixed step size $\alpha = \alpha_{\max}$.

The optimization results for the three cases, which all use the same initial vector z_0, are shown in Fig. 5.19 for $k_{\max} = 25$ iterations. In Figure 5.19a, f_1 is plotted against f_2. The yellow dots resemble f_1 and f_2 at randomly sampled z. Figure 5.19b visualizes

$$f_{\min} = \min\left(f_1, f_2\right),$$

for each iteration, $k = 1, 2, \ldots, k_{\max}$. In this case, the adaptive step size provides the best results. The small step size gives slow convergence, while the large step oscillates as it becomes inaccurate due to the large step size. It cannot be concluded if Algorithm 5.21 has reached a global maximum or a stationary point.



(a) Loss function f_1 versus f_2 **(b)** Loss function $f_{\min}$ versus k

Figure 5.19. Example of the proposed optimization strategy with $n_t = 3$ and $n_x = 4$. Algorithm 5.21 is compared to the same algorithm but with fixed step size. The black circle marks the common initial value z_0. The squares mark the final point of each case. Yellow dots resemble f_1 and f_2 evaluated at randomly generated z.

Comments

In essence, the proposed optimization strategy in Algorithm 5.21 is an iterative descent based optimization method, where the descent directions are chosen from a finite set of predefined directions. Steps 4–6 correspond to a backtracking line search where the step size α is selected. Algorithm 5.21 takes $\alpha_{\min} > 0$ as an input to avoid getting stuck at local minima, and $\alpha_{\max} > \alpha_{\min}$ such that the linear approximation given by (5.76) does not become too poor. The stopping criterion used here is $k > k_{\max}$, i.e., the algorithm terminates after $k_{\max}$ iterations.

It should be emphasized that there are no guarantees that Algorithm 5.21 converges to a global maximum w.r.t. the problem in (5.71). In fact, simulations verify that, in general, only local maxima are reached.

5.5.4 Numerical Evaluation

Algorithm 5.21 is now evaluated using a numerical example. The association performance when computing $\Psi_{(j)}$ using Algorithm 5.21 is compared to the case when $\Psi_{(j)}$ is computed using Algorithm 5.11.

Simulation Specification

A target tracking scenario with $n_t = 10$ targets is assumed. It is assumed that the dimensionality $n_x = 6$, which here is interpreted as a CAM. For each target $x_{(i)}$, a pair of covariances $R_{1(i)}$ and $R_{2(i)}$ are defined, which are held fixed throughout the simulations. An MC approach is used, where in each MC run state estimates $y_{1(i)}$ and $y_{2(i)}$ are sampled using $R_{1(i)}$ and $R_{2(i)}$, respectively, and the model in (5.51). A *scaling factor* c is used to scale the two spatial uncertainty components. Hence, for larger c, the association problem becomes more difficult to solve. The assumed target tracking scenario is depicted in Figure 5.20 with $c = 1$.

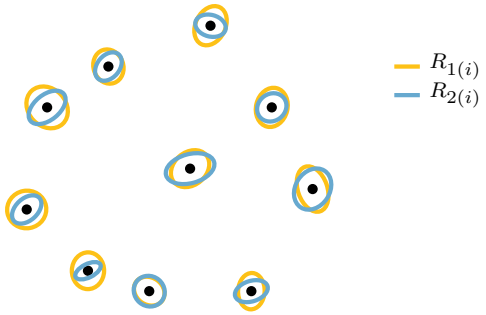


Figure 5.20. Scenario used in the evaluation. The targets are represented by black dots. The ellipse around a target illustrates the uncertainty of the corresponding estimate in the two spatial dimensions.

The association performance is evaluated using the *probability of incorrect assignment* (PIA). This quantity is computed for a certain c as the mean over all MC runs of the number of incorrect assignments divided by n_t. It is computed for the following cases:

- $(\mathcal{Y}_1, \mathcal{Y}_2)$: The full estimate configuration, where Agent 1 receives $\mathcal{Y}_2$ from Agent 2.
- $(\mathcal{Y}_1, \mathcal{Y}_\Psi) + \Psi_{(j)}$ using Algorithm 5.11: A dimension-reduced configuration, where Agent 1 receives $\mathcal{Y}_\Psi$ from Agent 2 and $\Psi_{(j)}$ is computed using Algorithm 5.11. In this case it is assumed that Agent 2 has access to $\mathcal{Y}_1$ such that fusion optimal $\Psi_{(j)}$ can be computed.
- $(\mathcal{Y}_1, \mathcal{Y}_\Psi) + \Psi_{(j)}$ using Algorithm 5.21: A dimension-reduced configuration, where Agent 1 receives $\mathcal{Y}_\Psi$ from Agent 2 and $\Psi_{(j)}$ is computed using the proposed optimization strategy in Algorithm 5.21.

The standard deviation of PIA is also computed.

Results

The simulation results are visualized in Figure 5.21, where PIA is plotted against c. For each value of c, $M = 1000$ MC runs are evaluated. PIA is computed in the same realizations of $\mathcal{Y}_1$ and $\mathcal{Y}_2$ for each of the cases described previously. The shaded areas in the plot resemble 1-σ confidence intervals.

Perfect association is maintained in the full estimate case for all values of c. The approach that utilizes Algorithm 5.21 clearly outperforms the approach that computes $\Psi_{(j)}$ for optimal fusion performance.

5.6 Summary

This chapter has examined the problem of designing efficient communication management based on reducing the dimensionality of exchanged tracks. The PCO

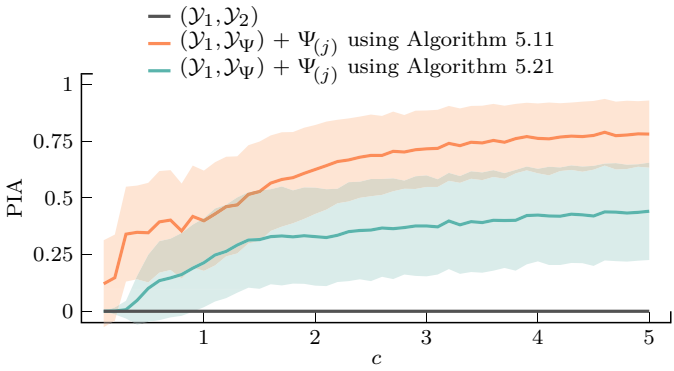


Figure 5.21. Results of the numerical evaluation. The incorrect assignment rate PIA is computed as a sample mean for each of the three cases for different values of $c \in [0.1, 5.0]$. The shaded areas illustrate the standard deviation of PIA.

method, which utilizes a PCA, was developed to be used for deriving DR estimates. From a track fusion perspective, however, the PCO method yields poor performance in many cases. The GEVO framework was developed to overcome the limitations of the PCO method. It was shown that the GEVO method computes DR estimates, which are MSE optimal w.r.t. fusion performance in cases of BSC, KF, and LE.

The GEVO framework requires globally available knowledge about local covariances to work optimally. This, in general, unrealistic assumption was relaxed by the CIE. The CIE framework was designed to keep approximate track of information shared among the agents within the DSN. It was demonstrated that the GEVO method with the CIE does not degrade the performance too much while only using local information.

Finally, the track-to-track association problem was formalized in a DR context. The goal was to design the communication management to preserve association quality when exchanging DR estimates. To this end, an optimization algorithm was developed. However, it needs to be addressed that this novel problem was just recently formalized and that the proposed solution is only a first step to handle the issue of track-to-track association in a DR context.

6

Final Remarks

The *decentralized target tracking* (DTT) problem was addressed. This DTT problem involved network-centric operations with multiple heterogenous agents. Two components of a DTT system were in particular studied:

- *Track fusion.* Agents exchange local track estimates with other agents. The track estimates are generally correlated, which must be handled. This thesis has focused on conservative track fusion techniques. Several methods and adaptations to established methods have been proposed.
- *Communication management.* In certain DTT problems, e.g., battery-driven and other low-power networks, there are communication constraints that restrict the amount of track data being exchanged. Communication reduction is also crucial in low-signature operations. This thesis has addressed both cases, and different data reduction methodologies have been proposed.

The contributions involved both theoretical and practical developments. All theory and method evaluations were based on numerical experiments.

Summary

The *conservative linear unbiased estimator* (CLUE) was introduced as a framework for robust track fusion under partially known correlations. A best CLUE was proposed for optimal conservative track fusion. A key contribution was the *robust optimization* (RO) based technique applicable to general CLUE problems. *Inverse covariance intersection* (ICI) and the *largest ellipsoid* (LE) method were adapted for more general models. Several properties related to the CLUE framework and the proposed methods were derived.

Two approaches to reduced communication were examined. At first, the *diagonal covariance approximation* (DCA) was introduced, and several methods were

developed for preserving conservativeness. The second approach was to exchange *dimension-reduced* (DR) estimates. Theory and methods were developed to this end, with the main results being the *generalized eigenvalue optimization* (GEVO) framework. Optimality guarantees were provided for some of the GEVO methods. In addition, the thesis proposed the *common information estimate* (CIE) as a resolution to the issue of having access to local information only. At last, an optimization algorithm was developed for computing DR estimates for association quality.

Conclusions

Track fusion and communication management correspond to subproblems P1 and P2 of Section 1.3, respectively.

Concerning P1, most of the contributions are found in the CLUE framework. The CLUE framework is provided as a toolbox for system engineers to design, analyze, and build fusion subsystems for target tracking applications. For instance, the lower and upper bounds can be used as guidelines in the system design, and the RO based methodology can be used as a reference when tailoring fusion algorithms for specific applications. The CLUE framework was developed with both conservativeness and tracking performance in mind. In the end, these are the criteria that a system engineer must compromise between when designing a track fusion component.

The DCA and DR approaches are dedicated to P2. The DCA methodology supplies, e.g., system engineers with theory to handle certain communication constraints robustly. The DCA methodology is generic in the sense that it focuses on the preservation of conservativeness and therefore can be used with many different track fusion methods. The primary strength of the DR framework, and in particular GEVO, lies in its foundation in mathematical optimization, which allows for optimal communication reduction. By adding the CIE to the DR framework, it is possible to efficiently reduce the communicated data in any DTT problems. This enables a flexible and modular system design while taking a finite bandwidth into account.

Future Outlook

A key aspect of the CLUE framework is optimality under conservativeness. For instance, *covariance intersection* (CI) is a best CLUE given that the correlations are completely unknown. However, when there is only partial knowledge available about the correlations, CI is overly conservative. In the numerical DTT examples simulated in this thesis, it can be seen that CI is often overly conservative in dynamic target tracking problems. At the same time, other methods such as ICI and LE are not strictly conservative in the same examples. A future topic to study is how to exploit structure even more in these examples so that improved tracking performance can be obtained while at the same time guaranteeing conservative estimates. It would be interesting to see how *stochastic optimization* can be used in a similar way to how RO is used for general CLUE problems.

With larger sensor networks and advances in sensor technology and data-driven methods, it follows that more data is processed and shared. Hence, there is a need to utilize and communicate data more efficiently. The DCA and DR frameworks considered in this thesis are two relevant techniques that can be further exploited and tailored to particular problems. The GEVO framework is tailored to specific linear track fusion methods but could, e.g., be adapted to machine learning algorithms and nonlinear track fusion problems. The considered data reduction techniques can be developed to work more adaptively. For instance, to integrate decision logic for when and with whom to communicate. Moreover, it would be interesting to extend the GEVO framework to incorporate the estimation performance within a future time horizon time step instead of the instantaneous track fusion performance only.

Bibliography

[1] N. R. Ahmed, W. W. Whitacre, S. Moon, and E. W. Frew. Scalable decentralized partial state estimation with sensor uncertainties using factorized data fusion. In *Proceeding of the AIAA SciTech 2016 Infotech@Aerospace Conference*, San Diego, CA, USA, Jan. 2016.

[2] N. R. Ahmed, W. W. Whitacre, S. Moon, and E. W. Frew. Factorized covariance intersection for scalable partial state decentralized data fusion. In *Proceedings of the 19th IEEE International Conference on Information Fusion*, Heidelberg, Germany, July 2016.

[3] J. Ajgl and O. Straka. Covariance intersection in track-to-track fusion without memory. In *Proceedings of the 19th IEEE International Conference on Information Fusion*, Heidelberg, Germany, July 2016.

[4] J. Ajgl and O. Straka. Covariance intersection in track-to-track fusion with memory. In *Proceedings of the 2016 IEEE International Conference on Multisensor Fusion and Integration*, Baden-Baden, Germany, Sept. 2016.

[5] J. Ajgl and O. Straka. Analysis of partial knowledge of correlations in an estimation fusion problem. In *Proceedings of the 21st IEEE International Conference on Information Fusion*, Cambridge, UK, July 2018.

[6] J. Ajgl and O. Straka. Comparison of fusions under unknown and partially known correlations. In *Proceedings of the 7th IFAC Workshop on Distributed Estimation and Control in Networked Systems*, volume 51, pages 295–300, Groningen, Netherlands, Aug. 2018.

[7] J. Ajgl and O. Straka. Fusion of multiple estimates by covariance intersection: Why and how it is suboptimal. *International Journal of Applied Mathematics and Computer Science*, 28(3):521–530, Sept. 2018.

[8] J. Ajgl and O. Straka. Rectification of partitioned covariance intersection. In *Proceedings of the 2019 American Control Conference*, Philadelphia, PA, USA, July 2019.

[9] J. Ajgl and O. Straka. Inverse covariance intersection fusion of multiple estimates. In *Proceedings of the 23rd IEEE International Conference on Information Fusion*, Rustenburg, South Africa, July 2020.

[10] J. Ajgl and O. Straka. Lower bounds in estimation fusion with partial knowledge of correlations. In *Proceedings of the 2021 IEEE International Conference on Multisensor Fusion and Integration*, Karlsruhe, Germany, Sept. 2021.

[11] J. Ajgl, M. Šimandl, M. Reinhardt, B. Noack, and U. D. Hanebeck. Covariance intersection in state estimation of dynamical systems. In *Proceedings of the 17th IEEE International Conference on Information Fusion*, Salamanca, Spain, July 2014.

[12] A. Amar, A. Leshem, and M. Gastpar. Recursive implementation of the distributed Karhunen-Loève transform. *IEEE Transactions on Signal Processing*, 58(10):5320–5330, 2010.

[13] K. Anstreicher and H. Wolkowicz. On Lagrangian relaxation of quadratic matrix con-

straints. *SIAM Journal on Matrix Analysis and Applications*, 22(1):41–55, 2000.

[14] M. A. Bakr and S. Lee. Distributed multisensor data fusion under unknown correlation and data inconsistency. *Sensors*, 17(11):2472, Oct. 2017.

[15] Y. Bar-Shalom and L. Campo. The effect of the common process noise on the two-sensor fused-track covariance. *IEEE Transactions on Aerospace and Electronic Systems*, 22(6): 803–805, Nov. 1986.

[16] Y. Bar-Shalom, X. Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons, Ltd, New York, NY, USA, 2001.

[17] Y. Bar-Shalom, X. Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software*. John Wiley & Sons, Ltd, New York, NY, USA, 2004.

[18] Y. Bar-Shalom, P. K. Willet, and X. Tian. *Tracking and Data Fusion: A Handbook of Algorithms*. YBS Publishing, Storrs, CT, USA, 2011.

[19] G. Battistelli, L. Chisci, G. Mugnai, A. Farina, and A. Graziano. Consensus-based linear and nonlinear filtering. *IEEE Transactions on Automatic Control*, 60(5):1410–1415, May 2015.

[20] A. Beck. *First-Order Methods in Optimization*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2017.

[21] A. Ben-Tal and A. Nemirovski. Robust convex optimization. *Mathematics of Operations Research*, 23(4):769–805, Nov. 1998.

[22] A. Ben-Tal and A. Nemirovski. Robust optimization — Methodology and applications. *Math. Program.*, 92:453–480, May 2002.

[23] A. Ben-Tal, L. E. Ghaoui, and A. Nemirovski. *Robust Optimization*. Princeton University Press, Princeton, NJ, USA, 2009.

[24] A. R. Benaskeur. Consistent fusion of correlated data sources. In *Proceedings of the 28th Annual Conference of the IEEE Industrial Electronics Society*, pages 2652–2656, Sevilla, Spain, Nov. 2002.

[25] J. Bibby. Axiomatisations of the average and a further generalisation of monotonic sequences. *Glasgow Mathematical Journal*, 15(1):63–65, 1974.

[26] G. J. Bierman and M. R. Belzer. A decentralized square root information filter/smoother. In *Proceedings of the 24th IEEE Conference Decision and Control*, pages 1902–1905, Fort Lauderdale, FL, USA, Dec. 1985.

[27] S. S. Blackman and R. Popoli. *Design and analysis of modern tracking systems*. Artech House, Norwood, MA, USA, 1999.

[28] D. Bossér, R. Forsling, I. Skog, G. Hendeby, and M. L. Nordenvaad. Underwater environment modeling for passive sonar track-before-detect. In *OCEANS 2023*, Limerick, Ireland, June 2023.

[29] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004.

[30] R. E. Burkard, M. Dell'Amico, and S. Martello. *Assignment Problems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2009.

[31] M. E. Campbell and N. R. Ahmed. Distributed data fusion: Neighbors, rumors, and the art of collective knowledge. *IEEE Control Systems Magazine*, 36(4):83–109, Aug. 2016.

[32] N. A. Carlson. Federated filter for fault-tolerant integrated navigation systems. In *Proceedings of the 1988 IEEE Position Location and Navigation Symposium*, pages 110–119, Orlando, FL, USA, Dec. 1988.

[33] N. A. Carlson. Federated square root filter for decentralized parallel processors. *IEEE*

Transactions on Aerospace and Electronic Systems, 26:517–525, May 1990.

[34] F. Castanedo. A review of data fusion techniques. *The Scientific World Journal*, 2013:1–19, 2013.

[35] H. Chen, K. Zhang, and X.-R. Li. Optimal data compression for multisensor target tracking with communication constraints. In *Proceedings of the 43rd IEEE Conference Decision and Control*, volume 3, pages 2650–2655, Atlantis, Paradise Island, Bahamas, Dec. 2004.

[36] L. Chen, P. Arambel, and R. Mehra. Fusion under unknown correlation: Covariance intersection as a special case. In *Proceedings of the 5th IEEE International Conference on Information Fusion*, Annapolis, MD, USA, July 2002.

[37] L. Chen, P. Arambel, and R. Mehra. Estimation under unknown correlation: Covariance intersection revisited. *IEEE Transactions on Automatic Control*, 47:1879–1882, Nov. 2002.

[38] Z. Chen, C. Heckman, S. Julier, and N. Ahmed. Weak in the NEES?: Auto-tuning Kalman filters with Bayesian optimization. In *Proceedings of the 21st IEEE International Conference on Information Fusion*, pages 1072–1079, Cambridge, UK, July 2018.

[39] C.-Y. Chong. Hierarchical estimation. In *Proceedings of 2nd MIT/ONR Workshop on Distributed Information and Decision Systems*, Monterey, CA, USA, July 1979.

[40] C.-Y. Chong, K. Chang, and S. Mori. A review of forty years of distributed estimation. In *Proceedings of the 21st IEEE International Conference on Information Fusion*, Cambridge, UK, July 2018.

[41] G. O. Corrêa and A. Talavera. Competitive robust estimation for uncertain linear dynamic models. *IEEE Transactions on Signal Processing*, 65(18):4847–4861, Sept. 2017.

[42] C. Cros, P.-O. Amblard, C. Prieur, and J.-F. Da Rocha. Fusion of distance measurements between agents with unknown correlations. *IEEE Control Systems Letters*, 7:2143–2148, 2023.

[43] R. C. de Lamare and R. Sampaio-Neto. Reduced-rank adaptive filtering based on joint iterative optimization of adaptive filters. *IEEE Signal Processing Letters*, 14(12):980–983, 2007.

[44] E. Delage and Y. Ye. Distributionally robust optimization under moment uncertainty with application to data-driven problems. *Operations Research*, 58(3):595–612, 2010.

[45] Z. Deng, P. Zhang, W. Qi, G. Yuan, J. Liu, Z. Deng, P. Zhang, W. Qi, Yuan, Gao, and J. Liu. The accuracy comparison of multisensor covariance intersection fuser and three weighting fusers. *Information Fusion*, 14(2):177–185, Apr. 2013.

[46] Z. Duan and X.-R. Li. Lossless linear transformation of sensor data for distributed estimation fusion. *IEEE Transactions on Signal Processing*, 59(1):362–372, 2011.

[47] B. Dulek, S. T. Topaloglu, and S. Gezici. Optimal linear MMSE estimation under correlation uncertainty in restricted Bayesian framework. *IEEE Transactions on Aerospace and Electronic Systems*, 59(4):4744–4752, 2023.

[48] J. Fang and H. Li. Dimensionality reduction with automatic dimension assignment for distributed estimation. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 2729–2732, Las Vegas, NV, USA, Mar. 2008.

[49] J. Fang and H. Li. Joint dimension assignment and compression for distributed multisensor estimation. *IEEE Signal Processing Letters*, 15:174–177, 2008.

[50] J. Fang and H. Li. Optimal/near-optimal dimensionality reduction for distributed estimation in homogeneous and certain inhomogeneous scenarios. *IEEE Transactions on Signal Processing*, 58(8):4339–4353, 2010.

[51] J. Fang and H. Li. Dimensionality reduction design for distributed estimation in certain inhomogeneous scenarios. In *Proceedings of the 17th International Conference on Digital Signal Processing*, pages 1–6, Corfu, Greece, July 2011.

[52] R. Forsling. *Decentralized Estimation Using Conservative Information Extraction*. Licentiate Thesis No. 1897, Linköping University, Linköping, Sweden, Dec. 2020.

[53] R. Forsling, B. Noack, and G. Hendeby. A quarter-century of covariance intersection: Correlations still unknown? *IEEE Control Systems Magazine*. Accepted for publication in Sep. 2023. Scheduled for the Apr. 2024 issue.

[54] R. Forsling, Z. Sjanic, F. Gustafsson, and G. Hendeby. Consistent distributed track fusion under communication constraints. In *Proceedings of the 22nd IEEE International Conference on Information Fusion*, Ottawa, Canada, July 2019.

[55] R. Forsling, Z. Sjanic, F. Gustafsson, and G. Hendeby. Communication efficient decentralized track fusion using selective information extraction. In *Proceedings of the 23rd IEEE International Conference on Information Fusion*, Rustenburg, South Africa, July 2020.

[56] R. Forsling, A. Hansson, F. Gustafsson, Z. Sjanic, J. Löfberg, and G. Hendeby. Conservative linear unbiased estimation under partially known covariances. *IEEE Transactions on Signal Processing*, 70:3123–3135, June 2022.

[57] R. Forsling, Z. Sjanic, F. Gustafsson, and G. Hendeby. Optimal linear fusion of dimension-reduced estimates using eigenvalue optimization. In *Proceedings of the 25th IEEE International Conference on Information Fusion*, Linköping, Sweden, July 2022.

[58] R. Forsling, F. Gustafsson, Z. Sjanic, and G. Hendeby. Decentralized data fusion of dimension-reduced estimates using local information only. In *Proceedings of the IEEE Aerospace Conference*, Big Sky, MT, USA, Mar. 2023.

[59] R. Forsling, F. Gustafsson, Z. Sjanic, and G. Hendeby. Decentralized state estimation in a dimension-reduced linear regression. 2023. URL `https://arxiv.org/abs/2210.06947`. Preprint, arXiv.

[60] R. Forsling, Z. Sjanic, F. Gustafsson, and G. Hendeby. Track-to-track association for fusion of dimension-reduced estimates. In *Proceedings of the 26th IEEE International Conference on Information Fusion*, Charleston, SC, USA, June 2023.

[61] C. Fritsche, U. Orguner, and F. Gustafsson. On parametric lower bounds for discrete-time filtering. In *Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4338–4342, Shanghai, China, Mar. 2016.

[62] A. Fuller, Z. Fan, C. Day, and C. Barlow. Digital twin: Enabling technologies, challenges and open research. *IEEE Access*, 8:108952–108971, 2020.

[63] C. Funk, B. Noack, and U. D. Hanebeck. Conservative quantization of fast covariance intersection. In *Proceedings of the 2020 IEEE International Conference on Multisensor Fusion and Integration*, pages 68–74, Virtual Conference, Sept. 2020.

[64] C. Funk, B. Noack, and U. D. Hanebeck. Conservative quantization of covariance matrices with applications to decentralized information fusion. *Sensors*, 21(9), 2021.

[65] Y. Gao, X.-R. Li, and E. Song. Robust linear estimation fusion with allowable unknown cross-covariance. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 46(9): 1314–1325, 2016.

[66] L. E. Ghaoui, F. Oustry, and H. Lebgret. Robust solutions to uncertain semidefinite programs. *SIAM Journal on Optimization*, 9(1):33–52, May 1998.

[67] G. H. Golub and C. F. van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, MD, USA, 4 edition, 2013.

[68] F. Govaers and W. Koch. Distributed Kalman filter fusion at arbitrary instants of time. In *Proceedings of the 13th IEEE International Conference on Information Fusion*, pages 1–8, Edinburgh, Scotland, July 2010.

[69] M. Greiff and K. Berntorp. Optimal measurement projections with adaptive mixture Kalman filtering for GNSS positioning. In *Proceedings of the 2020 American Control Conference*, pages 4435–4441, Denver, CO, USA, July 2020.

[70] M. Greiff, A. Robertsson, and K. Berntorp. MSE-optimal measurement dimension reduction in Gaussian filtering. In *Proceedings of the 2020 IEEE Conference on Control Technology and Applications*, pages 126–133, Aug. 2020.

[71] T. N. E. Greville. Note on the generalized inverse of a matrix product. *SIAM Review*, 8 (4):518–521, 1966.

[72] S. H. Grime and H. F. Durrant-Whyte. Data fusion in decentralized sensor networks. *Control Engineering Practice*, 2(5):849–863, Oct. 1994.

[73] F. Gustafsson. *Statistical Sensor Fusion*. Studentlitteratur, Lund, Sweden, 2018.

[74] D. Hall, C.-Y. Chong, J. Llinas, and M. Liggins, editors. CRC Press, Boca Raton, FL, USA, 2012.

[75] U. Hanebeck, K. Briechle, and J. Horn. A tight bound for the joint covariance of two random vectors with unknown but constrained cross-correlation. In *Proceedings of the 2001 IEEE International Conference on Multisensor Fusion and Integration*, pages 85–90, Baden-Baden, Germany, Aug. 2001.

[76] U. D. Hanebeck and J. Horn. Data validation in the presence of imprecisely known correlations. In *Proceedings of the 2003 European Control Conference*, Cambridge, UK, Sept. 2003.

[77] H. R. Hashemipour, S. Roy, and A. J. Laub. Decentralized structures for parallel Kalman filtering. *IEEE Transactions on Automatic Control*, 33(1):88–94, 1988.

[78] M. Hassan, G. Salut, M. Singh, and A. Titli. A decentralized computational algorithm for the global Kalman filter. *IEEE Transactions on Automatic Control*, 23(2):262–268, 1978.

[79] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning: Data mining, inference and prediction*. Springer-Verlag, New York, NY, USA, 2 edition, 2009.

[80] S. Haykin. *Digital Communication Systems*. John Wiley & Sons, Ltd, Hoboken, NJ, USA, 1 edition, 2014.

[81] C. W. Hays and T. Henderson. Matrix decomposition approaches for mutual information approximation with applications to covariance intersection techniques. *Information Fusion*, 95:446–453, 2023.

[82] M. Honig and J. Goldstein. Adaptive reduced-rank interference suppression based on the multistage Wiener filter. *IEEE Transactions on Communications*, 50(6):986–994, 2002.

[83] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, New York, NY, USA, 2012.

[84] J. Hu, L. Xie, and C. Zhang. Diffusion Kalman filtering based on covariance intersection. *IEEE Transactions on Signal Processing*, 60(2):891–902, 2012.

[85] Y. Hua, M. Nikpour, and P. Stoica. Optimal reduced-rank estimation and filtering. *IEEE Transactions on Signal Processing*, 49(3):457–469, 2001.

[86] A. Jazwinski. *Stochastic processes and filtering theory*. Academic Press, New York, NY, USA, 1970.

[87] A. Jindal and K. Psounis. Modeling spatially-correlated sensor network data. In *Proceedings of the 1st IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks*, pages 162–171, Santa Clara, CA, USA, Jan. 2004.

[88] R. Johnson and D. Wichern. *Applied Multivariate Statistical Analysis*. Pearson Prentice Hall, Upper Saddle River, NJ, USA, 6 edition, 2007.

[89] S. J. Julier and J. J. LaViola Jr. An empirical study into the robustness of split covariance addition (SCA) for human motion tracking. In *Proceedings of the 2004 American Control Conference*, Boston, MA, USA, June 2004.

[90] S. J. Julier and J. K. Uhlmann. A non-divergent estimation algorithm in the presence of

unknown correlations. In *Proceedings of the 1997 American Control Conference*, pages 2369–2373, Albuquerque, NM, USA, June 1997.

[91] S. J. Julier and J. K. Uhlmann. Simultaneous localisation and map building using split covariance intersection. In *Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Maui, HI, USA, Nov. 2001.

[92] S. J. Julier and J. K. Uhlmann. Using covariance intersection for SLAM. *Robotics and Autonomous Systems*, 55(1):3–20, Jan. 2007.

[93] S. J. Julier and J. K. Uhlmann. General decentralized data fusion with covariance intersection. In Liggins et al. [110], chapter 14.

[94] T. Kailath, A. H. Sayed, and B. Hassibi. *Linear Estimation*. Prentice Hall, Upper Saddle River, NJ, USA, 2000.

[95] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D):35–45, 1960.

[96] S. M. Kay. *Fundamentals of Statistical Signal Processing: Estimation theory*. Prentice Hall, Upper Saddle River, NJ, USA, 1993.

[97] T. H. Kerr and L. Chin. *Advances in the Techniques and Technology of the Application of Nonlinear Filters and Kalman Filters*, chapter The theory and techniques of discrete-time decentralized filters, pages 55–93. Number 256. North Atlantic Treaty Organization: The Advisory Group for Aerospace Research and Development, Neuilly sur Seine, France, 1982.

[98] B. Khaleghi, A. Khamis, F. O. Karray, and S. N. Razavi. Multisensor data fusion: A review of the state-of-the-art. *Information Fusion*, 14(1):28–44, Jan. 2013.

[99] U. A. Khan and J. M. F. Moura. Distributing the Kalman filter for large-scale systems. *IEEE Transactions on Signal Processing*, 56(10):4919–4935, Oct. 2008.

[100] N. Kimura and S. Latifi. A survey on data compression in wireless sensor networks. In *Proceedings of the IEEE International Conference on Information Technology: Coding and Computing*, volume 2, pages 8–13, Las Vegas, NV, USA, Apr. 2005.

[101] J. Klingner, N. Ahmed, and N. Correll. Fault-tolerant covariance intersection for localizing robot swarms. *Robotics and Autonomous Systems*, 122, 2019.

[102] J. Lai, Y. Zhou, J. Lin, Y. Cong, and J. Yang. Cooperative localization based on efficient covariance intersection. *IEEE Communications Letters*, 23(5):871–874, 2019.

[103] S. Leonardos and K. Daniilidis. A game-theoretic approach to robust fusion and Kalman filtering under unknown correlations. In *Proceedings of the 2017 American Control Conference*, Seattle, WA, USA, July 2017.

[104] H. Li, F. Nashashibi, B. Lefaudeux, and E. Pollard. Track-to-track fusion using split covariance intersection filter-information matrix filter (SCIF-IMF) for vehicle surrounding environment perception. In *Proceedings of the 16th IEEE International Conference on Intelligent Transportation Systems*, Hague, Netherlands, Oct. 2013.

[105] H. Li, F. Nashashibi, and M. Yang. Split covariance intersection filter: Theory and its application to vehicle localization. 14(4):1860–1871, Dec. 2013.

[106] W. Li and F. Yang. Information fusion over network dynamics with unknown correlations: An overview. *International Journal of Network Dynamics and Intelligence*, Feb. 2023.

[107] X.-R. Li and V. P. Jilkov. Survey of maneuvering target tracking. Part I. Dynamic models. *IEEE Transactions on Aerospace and Electronic Systems*, 39(4):1333–1364, 2003.

[108] X.-R. Li and Z. Zhao. Measuring estimator's credibility: Noncredibility index. In *Proceedings of the 9th IEEE International Conference on Information Fusion*, Florence, Italy, July 2006.

[109] X.-R. Li, Z. Zhao, and V. P. Jilkov. Practical measures and test for credibility of an estimator. In *Proceedings of Workshop on Estimation, Tracking and Fusion: A tribute to*

Yaakov Bar-Shalom, Monterey, CA, USA, May 2001.

[110] M. Liggins, D. Hall, and J. Llinas, editors. CRC Press, Boca Raton, FL, USA, 2009.

[111] A. Lima, P. Bonnifait, V. Cherfaoui, and J. A. Hage. Data fusion with split covariance intersection for cooperative perception. In *Proceedings of the 24th IEEE International Conference on Intelligent Transportation Systems*, Indianapolis, IN, USA, Sept. 2021.

[112] J. Llinas. Network-centric concepts: Impacts to distributed fusion system design. In Hall et al. [74], chapter 3.

[113] J. Löfberg. Support for robust conic-conic optimization in YALMIP. Unpublished.

[114] J. Löfberg. YALMIP: A toolbox for modeling and optimization in MATLAB. In *Proceedings of the CACSD Conference*, volume 3, Taipei, Taiwan, Sept. 2004.

[115] J. Löfberg. Automatic robust convex programming. *Optimization Methods and Software*, 27(1):115–129, Feb. 2012.

[116] R. C. Luo, C. C. Chang, and C. C. Lai. Multisensor fusion and integration: Theories, applications, and its perspectives. *IEEE Sensors Journal*, 11(12):3122–3138, Aug. 2011.

[117] H. Ma, Y.-H. Yang, Y. Chen, K. J. R. Liu, and Q. Wang. Distributed state estimation with dimension reduction preprocessing. *IEEE Transactions on Signal Processing*, 62(12): 3098–3110, 2014.

[118] R. P. S. Mahler. Multitarget Bayes filtering via first-order multitarget moments. *IEEE Transactions on Aerospace and Electronic Systems*, 39(4):1152–1178, 2003.

[119] J. Matoušek, J. Duník, and R. Forsling. Distributed point-mass filter with reduced data transfer using copula theory. In *Proceedings of the 2023 American Control Conference*, pages 1649–1654, San Diego, CA, USA, June 2023.

[120] S. P. McLaughlin, R. J. Evans, and V. Krishnamurthy. Data incest removal in a survivable estimation fusion architecture. In *Proceedings of the 6th IEEE International Conference on Information Fusion*, Cairns, Queensland, Australia, July 2003.

[121] S. P. McLaughlin, V. Krishnamurthy, and S. Challa. Managing data incest in a distributed sensor network. In *Proceedings of the 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing*, Hong Kong, China, Apr. 2003.

[122] C. B. Moler and G. W. Stewart. An algorithm for generalized matrix eigenvalue problems. *SIAM Journal on Numerical Analysis*, 10(2):241–256, 1973.

[123] MOSEK ApS. *The MOSEK optimization toolbox for MATLAB manual. Version 9.2.*, 2021. URL `http://docs.mosek.com/9.2/toolbox/index.html`.

[124] E. J. Msechu and G. B. Giannakis. Sensor-centric data reduction for estimation with WSNs via censoring and quantization. *IEEE Transactions on Signal Processing*, 60(1):400–414, 2012.

[125] D. Nicholson, S. Reece, A. Rogers, S. Roberts, and N. Jennings. Distributed data fusion overarching design concerns and some new approaches. In Hall et al. [74], chapter 2.

[126] B. Noack, M. Baum, and U. D. Hanebeck. Automatic exploitation of independencies for covariance bounding in fully decentralized estimation. In *Proceedings of the 18th Triennial IFAC World Congress*, Milan, Italy, Aug. 2011.

[127] B. Noack, S. J. Julier, and U. D. Hanebeck. Treatment of biased and dependent sensor data in graph-based SLAM. In *Proceedings of the 18th IEEE International Conference on Information Fusion*, Washington, DC, USA, July 2015.

[128] B. Noack, J. Sijs, and U. D. Hanebeck. Algebraic analysis of data fusion with ellipsoidal intersection. In *Proceedings of the 2016 IEEE International Conference on Multisensor Fusion and Integration*, pages 365–370, Baden-Baden, Germany, Sept. 2016.

[129] B. Noack, J. Sijs, and U. D. Hanebeck. Inverse covariance intersection: New insights

and properties. In *Proceedings of the 20th IEEE International Conference on Information Fusion*, Xi'an, China, July 2017.

[130] B. Noack, J. Sijs, M. Reinhardt, and U. D. Hanebeck. Decentralized data fusion with inverse covariance intersection. *Automatica*, 79:35–41, May 2017.

[131] J. Nygårds, V. Deleskog, and G. Hendeby. Safe fusion compared to established distributed fusion methods. In *Proceedings of the 2016 IEEE International Conference on Multisensor Fusion and Integration*, Baden-Baden, Germany, Sept. 2016.

[132] J. Nygårds, V. Deleskog, and G. Hendeby. Decentralized tracking in sensor networks with varying coverage. In *Proceedings of the 21st IEEE International Conference on Information Fusion*, Cambridge, UK, July 2018.

[133] R. Olfati-Saber and R. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9):1520–1533, Sept. 2004.

[134] R. Olfati-Saber and J. S. Shamma. Consensus filters for sensor networks and distributed sensor fusion. In *Proceedings of the 44th IEEE Conference Decision and Control*, pages 6698–6703, Sevilla, Spain, Dec. 2005.

[135] U. Orguner. Approximate analytical solutions for the weight optimization problems of CI and ICI. In *Proceedings of the 2017 IEEE Symposium on Sensor Data Fusion: Trends, Solutions, Applications*, pages 1–6, Bonn, Germany, Oct. 2017.

[136] B. N. Parlett. *The Symmetric Eigenvalue Problem*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1998.

[137] K. Pearson. LIII. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2(11):559–572, 1901.

[138] A. Petersen and M.-A. Beyer. Partitioned covariance intersection. In *Proceedings of the 2011 International Symposium Information on Ships*, Hamburg, Germany, Sept. 2011.

[139] K. B. Petersen and M. S. Pedersen. The matrix cookbook, Nov. 2012.

[140] W. Qi, P. Zhang, and Z. Deng. Robust weighted fusion Kalman filters for multisensor time-varying systems with uncertain noise variances. *Signal Processing*, 99:185–200, June 2014.

[141] H. Qiu, M. Qiu, Z. Lu, and G. Memmi. An efficient key distribution system for data fusion in V2X heterogeneous networks. *Information Fusion*, 50:212–220, 2019.

[142] S. Radtke, B. Noack, U. D. Hanebeck, and O. Straka. Reconstruction of cross-correlations with constant number of deterministic samples. In *Proceedings of the 21st IEEE International Conference on Information Fusion*, Cambridge, UK, July 2018.

[143] S. Radtke, B. Noack, and U. D. Hanebeck. Fully decentralized estimation using square-root decompositions. In *Proceedings of the 23rd IEEE International Conference on Information Fusion*, Rustenburg, South Africa, July 2020.

[144] C. R. Rao. *Linear Statistical Inference and its Applications*. John Wiley & Sons, Ltd, New York, NY, USA, 2 edition, 1973.

[145] M. A. Razzaque, C. Bleakley, and S. Dobson. Compression in wireless sensor networks: A survey and comparative evaluation. *ACM Transactions on Sensor Networks*, 10(1), Dec. 2013.

[146] S. Reece and S. Roberts. Robust, low-bandwidth, multi-vehicle mapping. In *Proceedings of the 8th IEEE International Conference on Information Fusion*, Philadelphia, PA, USA, July 2005.

[147] D. B. Reid. An algorithm for tracking multiple targets. *IEEE Transactions on Automatic Control*, 24(6):843–854, 1979.

[148] M. Reinhardt, B. Noack, and U. D. Hanebeck. The hypothesizing distributed Kalman

filter. In *Proceedings of the 2012 IEEE International Conference on Multisensor Fusion and Integration*, Hamburg, Germany, Sept. 2012.

[149] M. Reinhardt, B. Noack, and U. D. Hanebeck. Reconstruction of joint covariance matrices in networked linear systems. In *Proceedings of the 48th Annual Conference on Information Sciences and Systems*, Princeton, NJ, USA, 2014.

[150] M. Reinhardt, B. Noack, P. O. Arambel, and U. D. Hanebeck. Minimum covariance bounds for the fusion under unknown correlations. *IEEE Signal Processing Letters*, 22(9):1210–1214, Sept. 2015.

[151] S. Reuter, B.-T. Vo, B.-N. Vo, and K. Dietmayer. The labeled multi-Bernoulli filter. *IEEE Transactions on Signal Processing*, 62(12):3246–3260, 2014.

[152] A. Ribeiro and G. Giannakis. Bandwidth-constrained distributed estimation for wireless sensor networks-part I: Gaussian case. *IEEE Transactions on Signal Processing*, 54(3): 1131–1143, 2006.

[153] A. H. Sayed. A framework for state-space estimation with uncertain models. *IEEE Transactions on Automatic Control*, 46(7):998–1013, 2001.

[154] L. Scharf and D. Tufts. Rank reduction for modeling stationary signals. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 35(3):350–355, 1987.

[155] I. D. Schizas and A. Aduroja. A distributed framework for dimensionality reduction and denoising. *IEEE Transactions on Signal Processing*, 63(23):6379–6394, 2015.

[156] I. D. Schizas, G. B. Giannakis, and Z.-Q. Luo. Distributed estimation using reduced-dimensionality sensor observations. *IEEE Transactions on Signal Processing*, 55(8):4284–4299, 2007.

[157] S. Shafieezadeh-Abadeh, V. A. Nguyen, D. Kuhn, and P. M. Esfahani. Wasserstein distributionally robust Kalman filtering. In *NeurIPS*, pages 8483–8492, Sept. 2018.

[158] V. Shin, V. Hamdipoor, and Y. Kim. Reduced-order multisensory fusion estimation with application to object tracking. *IET Signal Processing*, 16(4):463–478, Apr. 2022.

[159] J. Sijs and M. Lazar. Empirical case-studies of state fusion via ellipsoidal intersection. In *Proceedings of the 14th IEEE International Conference on Information Fusion*, Chicago, IL, USA, July 2011.

[160] J. Sijs, M. Lazar, and P. P. J. v. d. Bosch. State fusion with unknown correlation: Ellipsoidal intersection. In *Proceedings of the 2010 American Control Conference*, pages 3992–3997, Baltimore, MD, USA, June 2010.

[161] D. Smith and S. Singh. Approaches to multisensor data fusion in target tracking: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 18(12):1696–1710, Oct. 2006.

[162] J. L. Speyer. Computation and transmission requirements for a decentralized linear-quadratic-Gaussian control problem. *IEEE Transactions on Automatic Control*, 24(2): 266–269, Apr. 1979.

[163] J. Steinbring, B. Noack, M. Reinhardt, and U. D. Hanebeck. Optimal sample-based fusion for distributed state estimation. In *Proceedings of the 19th IEEE International Conference on Information Fusion*, Heidelberg, Germany, July 2016.

[164] G. W. Stewart and J.-G. Sun. *Matrix Perturbation Theory*. Academic Press, Boston, MA, USA, 1990.

[165] S. Sun, H. Lin, J. Ma, and X. Li. Multi-sensor distributed fusion estimation with applications in networked systems: A review paper. *Information Fusion*, 38:122–134, Nov. 2017.

[166] A. Tamjidi, R. Oftadeh, M. N. G. Mohamed, D. Yu, S. Chakravorty, and D. Shell. Unifying consensus and covariance intersection for efficient distributed state estimation over unreliable networks. *IEEE Transactions on Robotics*, 37(5), Oct. 2021.

[167] J. H. Taylor. The Cramér-Rao estimation error lower bound computation for deterministic nonlinear systems. *IEEE Transactions on Automatic Control*, 24(2):343–344, Apr. 1979.

[168] X. Tian and Y. Bar-Shalom. Exact algorithms for four track-to-track fusion configurations: All you wanted to know but were afraid to ask. In *Proceedings of the 12th IEEE International Conference on Information Fusion*, pages 537–544, Seattle, WA, USA, July 2009.

[169] X. Tian and Y. Bar-Shalom. On algorithms for asynchronous track-to-track fusion. In *Proceedings of the 13th IEEE International Conference on Information Fusion*, Edinburgh, Scotland, July 2010.

[170] C. K. Toh. *Wireless ATM and Ad-Hoc Networks: Protocols and Architectures*. Springer-Verlag, New York, NY, USA, 1997.

[171] J. K. Uhlmann. Covariance consistency methods for fault-tolerant distributed data fusion. *Information Fusion*, 4(3):201–215, Sept. 2003.

[172] P. Venkitasubramaniam, L. Tong, and A. Swami. Quantization for maximin ARE in distributed estimation. *IEEE Transactions on Signal Processing*, 55(7):3596–3605, 2007.

[173] B.-N. Vo, B.-T. Vo, and D. Phung. Labeled random finite sets and the Bayes multi-target tracking filter. *IEEE Transactions on Signal Processing*, 62(24):6554–6567, 2014.

[174] T. R. Wanasinghe, G. K. I. Mann, and R. G. Gosine. Decentralized cooperative localization approach for autonomous multirobot systems. *Journal of Robotics*, Apr. 2016.

[175] R. Wang. *Karhunen-Loève transform and principal component analysis*, page 412–460. Cambridge University Press, Cambridge, UK, 2012.

[176] S. Wang and Z.-S. Ye. Distributionally robust state estimation for linear systems subject to uncertainty and outlier. *IEEE Transactions on Signal Processing*, 70:452–467, Dec. 2022.

[177] J. L. Williams. Marginal multi-Bernoulli filters: RFS derivation of MHT, JIPDA, and data association-based member. *IEEE Transactions on Aerospace and Electronic Systems*, 51 (3):1664–1687, 2015.

[178] D. Willner, C.-B. Chang, and K.-P. Dunn. Kalman filter configurations for multiple radar systems, Apr. 1976. Technical Note.

[179] A. Willsky, M. Bello, D. Castanon, B. Levy, and G. Verghese. Combining and updating of local estimates and regional maps along sets of one-dimensional tracks. *IEEE Transactions on Automatic Control*, 27(4):799–813, 1982.

[180] J. Wishart. The generalised product moment distribution in samples from a normal multivariate population. *Biometrika*, 20A(1/2):32–52, 1928.

[181] M. A. Woodbury. *Inverting modified matrices*. Department of Statistics, Princeton University, Princeton, NJ, USA, 1950.

[182] Z. Wu, Q. Cai, and M. Fu. Covariance intersection for partially correlated random vectors. *IEEE Transactions on Automatic Control*, 63(3):619–629, Mar. 2018.

[183] L. Xiao, S. Boyd, and S. Lall. A scheme for robust distributed sensor fusion based on average consensus. In *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks*, pages 63–70, Apr. 2005.

[184] L. Yan, X.-R. Li, Y. Xia, and M. Fu. Optimal sequential and distributed fusion for state estimation in cross-correlated noise. *Automatica*, 49(12):3607–3612, Dec. 2013.

[185] M. Zarei-Jalalabadi, S. M. B. Malaek, and S. S. Kia. A track-to-track fusion method for tracks with unknown correlations. *IEEE Signal Processing Letters*, 2(2):189–194, Apr. 2018.

[186] J. Zhang, G. Scebba, and W. Karlen. Covariance intersection to improve the robustness of the photoplethysmogram derived respiratory rate. In *Proceedings of the 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society*, Montréal,

Canada, July 2020.

[187] K. Zhang, X.-R. Li, P. Zhang, and H. Li. Optimal linear estimation fusion - part VI: Sensor data compression. In *Proceedings of the 6th IEEE International Conference on Information Fusion*, pages 221–228, Cairns, Queensland, Australia, July 2003.

[188] Y. Zhou and J. Li. Data fusion of unknown correlations using internal ellipsoidal approximation. In *Proceedings of the 17th Triennial IFAC World Congress*, pages 2856–2860, Seoul, Korea, July 2008.

[189] H. Zhu, Q. Zhai, M. Yu, and C. Han. Estimation fusion algorithms in the presence of partially known cross-correlation of local estimation errors. *Information Fusion*, 18:187–196, July 2014.

[190] Y. Zhu, E. Song, J. Zhou, and Z. You. Optimal dimensionality reduction of sensor data in multisensor estimation fusion. *IEEE Transactions on Signal Processing*, 53(5):1631–1639, 2005.

[191] Z. Zhu and C. N. Taylor. Conservative uncertainty estimation in map-based vision-aided navigation. *IEEE Transactions on Aerospace and Electronic Systems*, 53(2):941–949, Apr. 2017.

PhD Dissertations
Division of Automatic Control
Linköping University

M. Millnert: Identification and control of systems subject to abrupt changes. Thesis No. 82, 1982. ISBN 91-7372-542-0.

A. J. M. van Overbeek: On-line structure selection for the identification of multivariable systems. Thesis No. 86, 1982. ISBN 91-7372-586-2.

B. Bengtsson: On some control problems for queues. Thesis No. 87, 1982. ISBN 91-7372-593-5.

S. Ljung: Fast algorithms for integral equations and least squares identification problems. Thesis No. 93, 1983. ISBN 91-7372-641-9.

H. Jonson: A Newton method for solving non-linear optimal control problems with general constraints. Thesis No. 104, 1983. ISBN 91-7372-718-0.

E. Trulsson: Adaptive control based on explicit criterion minimization. Thesis No. 106, 1983. ISBN 91-7372-728-8.

K. Nordström: Uncertainty, robustness and sensitivity reduction in the design of single input control systems. Thesis No. 162, 1987. ISBN 91-7870-170-8.

B. Wahlberg: On the identification and approximation of linear systems. Thesis No. 163, 1987. ISBN 91-7870-175-9.

S. Gunnarsson: Frequency domain aspects of modeling and control in adaptive systems. Thesis No. 194, 1988. ISBN 91-7870-380-8.

A. Isaksson: On system identification in one and two dimensions with signal processing applications. Thesis No. 196, 1988. ISBN 91-7870-383-2.

M. Viberg: Subspace fitting concepts in sensor array processing. Thesis No. 217, 1989. ISBN 91-7870-529-0.

K. Forsman: Constructive commutative algebra in nonlinear control theory. Thesis No. 261, 1991. ISBN 91-7870-827-3.

F. Gustafsson: Estimation of discrete parameters in linear systems. Thesis No. 271, 1992. ISBN 91-7870-876-1.

P. Nagy: Tools for knowledge-based signal processing with applications to system identification. Thesis No. 280, 1992. ISBN 91-7870-962-8.

T. Svensson: Mathematical tools and software for analysis and design of nonlinear control systems. Thesis No. 285, 1992. ISBN 91-7870-989-X.

S. Andersson: On dimension reduction in sensor array signal processing. Thesis No. 290, 1992. ISBN 91-7871-015-4.

H. Hjalmarsson: Aspects on incomplete modeling in system identification. Thesis No. 298, 1993. ISBN 91-7871-070-7.

I. Klein: Automatic synthesis of sequential control schemes. Thesis No. 305, 1993. ISBN 91-7871-090-1.

J.-E. Strömberg: A mode switching modelling philosophy. Thesis No. 353, 1994. ISBN 91-7871-430-3.

K. Wang Chen: Transformation and symbolic calculations in filtering and control. Thesis No. 361, 1994. ISBN 91-7871-467-2.

T. McKelvey: Identification of state-space models from time and frequency data. Thesis No. 380, 1995. ISBN 91-7871-531-8.

J. Sjöberg: Non-linear system identification with neural networks. Thesis No. 381, 1995. ISBN 91-7871-534-2.

R. Germundsson: Symbolic systems – theory, computation and applications. Thesis No. 389, 1995. ISBN 91-7871-578-4.

P. Pucar: Modeling and segmentation using multiple models. Thesis No. 405, 1995. ISBN 91-7871-627-6.

H. Fortell: Algebraic approaches to normal forms and zero dynamics. Thesis No. 407, 1995. ISBN 91-7871-629-2.

A. Helmersson: Methods for robust gain scheduling. Thesis No. 406, 1995. ISBN 91-7871-628-4.

P. Lindskog: Methods, algorithms and tools for system identification based on prior knowledge. Thesis No. 436, 1996. ISBN 91-7871-424-8.

J. Gunnarsson: Symbolic methods and tools for discrete event dynamic systems. Thesis No. 477, 1997. ISBN 91-7871-917-8.

M. Jirstrand: Constructive methods for inequality constraints in control. Thesis No. 527, 1998. ISBN 91-7219-187-2.

U. Forssell: Closed-loop identification: Methods, theory, and applications. Thesis No. 566, 1999. ISBN 91-7219-432-4.

A. Stenman: Model on demand: Algorithms, analysis and applications. Thesis No. 571, 1999. ISBN 91-7219-450-2.

N. Bergman: Recursive Bayesian estimation: Navigation and tracking applications. Thesis No. 579, 1999. ISBN 91-7219-473-1.

K. Edström: Switched bond graphs: Simulation and analysis. Thesis No. 586, 1999. ISBN 91-7219-493-6.

M. Larsson: Behavioral and structural model based approaches to discrete diagnosis. Thesis No. 608, 1999. ISBN 91-7219-615-5.

F. Gunnarsson: Power control in cellular radio systems: Analysis, design and estimation. Thesis No. 623, 2000. ISBN 91-7219-689-0.

V. Einarsson: Model checking methods for mode switching systems. Thesis No. 652, 2000. ISBN 91-7219-836-2.

M. Norrlöf: Iterative learning control: Analysis, design, and experiments. Thesis No. 653, 2000. ISBN 91-7219-837-0.

F. Tjärnström: Variance expressions and model reduction in system identification. Thesis No. 730, 2002. ISBN 91-7373-253-2.

J. Löfberg: Minimax approaches to robust model predictive control. Thesis No. 812, 2003. ISBN 91-7373-622-8.

J. Roll: Local and piecewise affine approaches to system identification. Thesis No. 802, 2003. ISBN 91-7373-608-2.

J. Elbornsson: Analysis, estimation and compensation of mismatch effects in A/D converters. Thesis No. 811, 2003. ISBN 91-7373-621-X.

O. Härkegård: Backstepping and control allocation with applications to flight control. Thesis No. 820, 2003. ISBN 91-7373-647-3.

R. Wallin: Optimization algorithms for system analysis and identification. Thesis No. 919, 2004. ISBN 91-85297-19-4.

D. Lindgren: Projection methods for classification and identification. Thesis No. 915, 2005. ISBN 91-85297-06-2.

R. Karlsson: Particle Filtering for Positioning and Tracking Applications. Thesis No. 924, 2005. ISBN 91-85297-34-8.

J. Jansson: Collision Avoidance Theory with Applications to Automotive Collision Mitigation. Thesis No. 950, 2005. ISBN 91-85299-45-6.

E. Geijer Lundin: Uplink Load in CDMA Cellular Radio Systems. Thesis No. 977, 2005. ISBN 91-85457-49-3.

M. Enqvist: Linear Models of Nonlinear Systems. Thesis No. 985, 2005. ISBN 91-85457-64-7.

T. B. Schön: Estimation of Nonlinear Dynamic Systems — Theory and Applications. Thesis No. 998, 2006. ISBN 91-85497-03-7.

I. Lind: Regressor and Structure Selection — Uses of ANOVA in System Identification. Thesis No. 1012, 2006. ISBN 91-85523-98-4.

J. Gillberg: Frequency Domain Identification of Continuous-Time Systems Reconstruction and Robustness. Thesis No. 1031, 2006. ISBN 91-85523-34-8.
M. Gerdin: Identification and Estimation for Models Described by Differential-Algebraic Equations. Thesis No. 1046, 2006. ISBN 91-85643-87-4.
C. Grönwall: Ground Object Recognition using Laser Radar Data – Geometric Fitting, Performance Analysis, and Applications. Thesis No. 1055, 2006. ISBN 91-85643-53-X.
A. Eidehall: Tracking and threat assessment for automotive collision avoidance. Thesis No. 1066, 2007. ISBN 91-85643-10-6.
F. Eng: Non-Uniform Sampling in Statistical Signal Processing. Thesis No. 1082, 2007. ISBN 978-91-85715-49-7.
E. Wernholt: Multivariable Frequency-Domain Identification of Industrial Robots. Thesis No. 1138, 2007. ISBN 978-91-85895-72-4.
D. Axehill: Integer Quadratic Programming for Control and Communication. Thesis No. 1158, 2008. ISBN 978-91-85523-03-0.
G. Hendeby: Performance and Implementation Aspects of Nonlinear Filtering. Thesis No. 1161, 2008. ISBN 978-91-7393-979-9.
J. Sjöberg: Optimal Control and Model Reduction of Nonlinear DAE Models. Thesis No. 1166, 2008. ISBN 978-91-7393-964-5.
D. Törnqvist: Estimation and Detection with Applications to Navigation. Thesis No. 1216, 2008. ISBN 978-91-7393-785-6.
P-J. Nordlund: Efficient Estimation and Detection Methods for Airborne Applications. Thesis No. 1231, 2008. ISBN 978-91-7393-720-7.
H. Tidefelt: Differential-algebraic equations and matrix-valued singular perturbation. Thesis No. 1292, 2009. ISBN 978-91-7393-479-4.
H. Ohlsson: Regularization for Sparseness and Smoothness — Applications in System Identification and Signal Processing. Thesis No. 1351, 2010. ISBN 978-91-7393-287-5.
S. Moberg: Modeling and Control of Flexible Manipulators. Thesis No. 1349, 2010. ISBN 978-91-7393-289-9.
J. Wallén: Estimation-based iterative learning control. Thesis No. 1358, 2011. ISBN 978-91-7393-255-4.
J. D. Hol: Sensor Fusion and Calibration of Inertial Sensors, Vision, Ultra-Wideband and GPS. Thesis No. 1368, 2011. ISBN 978-91-7393-197-7.
D. Ankelhed: On the Design of Low Order H-infinity Controllers. Thesis No. 1371, 2011. ISBN 978-91-7393-157-1.
C. Lundquist: Sensor Fusion for Automotive Applications. Thesis No. 1409, 2011. ISBN 978-91-7393-023-9.
P. Skoglar: Tracking and Planning for Surveillance Applications. Thesis No. 1432, 2012. ISBN 978-91-7519-941-2.
K. Granström: Extended target tracking using PHD filters. Thesis No. 1476, 2012. ISBN 978-91-7519-796-8.
C. Lyzell: Structural Reformulations in System Identification. Thesis No. 1475, 2012. ISBN 978-91-7519-800-2.
J. Callmer: Autonomous Localization in Unknown Environments. Thesis No. 1520, 2013. ISBN 978-91-7519-620-6.
D. Petersson: A Nonlinear Optimization Approach to H2-Optimal Modeling and Control. Thesis No. 1528, 2013. ISBN 978-91-7519-567-4.
Z. Sjanic: Navigation and Mapping for Aerial Vehicles Based on Inertial and Imaging Sensors. Thesis No. 1533, 2013. ISBN 978-91-7519-553-7.
F. Lindsten: Particle Filters and Markov Chains for Learning of Dynamical Systems. Thesis No. 1530, 2013. ISBN 978-91-7519-559-9.

P. Axelsson: Sensor Fusion and Control Applied to Industrial Manipulators. Thesis No. 1585, 2014. ISBN 978-91-7519-368-7.

A. Carvalho Bittencourt: Modeling and Diagnosis of Friction and Wear in Industrial Robots. Thesis No. 1617, 2014. ISBN 978-91-7519-251-2.

M. Skoglund: Inertial Navigation and Mapping for Autonomous Vehicles. Thesis No. 1623, 2014. ISBN 978-91-7519-233-8.

S. Khoshfetrat Pakazad: Divide and Conquer: Distributed Optimization and Robustness Analysis. Thesis No. 1676, 2015. ISBN 978-91-7519-050-1.

T. Ardeshiri: Analytical Approximations for Bayesian Inference. Thesis No. 1710, 2015. ISBN 978-91-7685-930-8.

N. Wahlström: Modeling of Magnetic Fields and Extended Objects for Localization Applications. Thesis No. 1723, 2015. ISBN 978-91-7685-903-2.

J. Dahlin: Accelerating Monte Carlo methods for Bayesian inference in dynamical models. Thesis No. 1754, 2016. ISBN 978-91-7685-797-7.

M. Kok: Probabilistic modeling for sensor fusion with inertial measurements. Thesis No. 1814, 2016. ISBN 978-91-7685-621-5.

J. Linder: Indirect System Identification for Unknown Input Problems: With Applications to Ships. Thesis No. 1829, 2017. ISBN 978-91-7685-588-1.

M. Roth: Advanced Kalman Filtering Approaches to Bayesian State Estimation. Thesis No. 1832, 2017. ISBN 978-91-7685-578-2.

I. Nielsen: Structure-Exploiting Numerical Algorithms for Optimal Control. Thesis No. 1848, 2017. ISBN 978-91-7685-528-7.

D. Simon: Fighter Aircraft Maneuver Limiting Using MPC: Theory and Application. Thesis No. 1881, 2017. ISBN 978-91-7685-450-1.

C. Veibäck: Tracking the Wanders of Nature. Thesis No. 1958, 2018. ISBN 978-91-7685-200-2.

C. Andersson Naesseth: Machine learning using approximate inference: Variational and sequential Monte Carlo methods. Thesis No. 1969, 2018. ISBN 978-91-7685-161-6.

Y. Jung: Inverse system identification with applications in predistortion. Thesis No. 1966, 2018. ISBN 978-91-7685-171-5.

Y. Zhao: Gaussian Processes for Positioning Using Radio Signal Strength Measurements. Thesis No. 1968, 2019. ISBN 978-91-7685-162-3.

R. Larsson: Flight Test System Identification. Thesis No. 1990, 2019. ISBN 978-91-7685-070-1.

P. Kasebzadeh: Learning Human Gait. Thesis No. 2012, 2019. ISBN 978-91-7519-014-3.

K. Radnosrati: Time of flight estimation for radio network positioning. Thesis No. 2054, 2020. ISBN 978-91-7929-884-5.

O. Ljungqvist: Motion planning and feedback control techniques with applications to long tractor-trailer vehicles. Thesis No. 2070, 2020. ISBN 978-91-7929-858-6.

G. Lindmark: Controllability of Complex Networks at Minimum Cost. Thesis No. 2074, 2020. ISBN 978-91-7929-847-0.

K. Bergman: Exploiting Direct Optimal Control for Motion Planning in Unstructured Environments. Thesis No. 2133, 2021. ISBN 978-91-7929-677-3.

P. Boström-Rost: Sensor Management for Target Tracking Applications. Thesis No. 2137, 2021. ISBN 978-91-7929-672-8.

A. Fontan: Collective decision-making on networked systems in presence of antagonistic interactions. Thesis No. 2166, 2021. ISBN 978-91-7929-017-7.

S. Parvini Ahmadi: Distributed Optimization for Control and Estimation. Thesis No. 2207, 2022. ISBN 978-91-7929-197-6.

F. Ljungberg: Identification of Nonlinear Marine Systems. Thesis No. 2258, 2022. ISBN 978-91-7929-493-9.

A. Zenere: Integration of epigenetic, transcriptomic and proteomic data. Thesis No. 2294, 2023. ISBN 978-91-8075-068-4.
K. Nielsen: Localization of Autonomous Vehicles in Underground Mines. Thesis No. 2318, 2023. ISBN 978-91-8075-167-4.
D. Arnström: Real-Time Certified MPC: Reliable Active-Set QP Solvers. Thesis No. 2324, 2023. ISBN 978-91-8075-218-3.
M. Malmström: Approximative Uncertainty in Neural Network Predictions. Thesis No. 2358, 2023. ISBN 978-91-8075-405-7.

FACULTY OF SCIENCE AND ENGINEERING

Linköping Studies in Science and Technology, Dissertations No. 2359
Department of Electrical Engineering

Linköping University
SE-581 83 Linköping, Sweden

www.liu.se