

Sensor Management with Applications in Localization and Tracking

by

Farhad Ghassemi

M.A.Sc., The University of British Columbia, 2001

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

Doctor of Philosophy

in

The Faculty of Graduate Studies

(Electrical and Computer Engineering)

The University of British Columbia

(Vancouver)

June 2009

© Farhad Ghassemi 2009

Abstract

In this dissertation, we explore several themes in sensor management with an emphasis on their applications for target localization and tracking. We consider the sensor subset selection problem where a pre-specified number of sensors must be selected in a sensor network to estimate an unknown value of a time-invariant parameter, e.g. the position of a target. We study the Lagrangian and continuous relaxations of this problem with the determinant of the Fisher information matrix as the objective function. We prove that the continuous bound is tighter than the Lagrangian bound and outline an algorithm based on the so-called natural selection process to compute the continuous bound when sensors are allowed to make more than one measurement.

We also study how a target can identify the informative sensors when it is facing a network that attempts to estimate its position or its other critical parameters. We show that by borrowing the notion of symmetric probabilistic values from cooperative game theory, the target can assign a power index to each sensor to determine how informative it is relative to the other ones. We further show that by choosing the determinant of the Fisher information matrix as the metric of estimation accuracy, the computational complexity associated with a power index gracefully increases with

the number of sensors.

Finally, we study the trajectory design problem for bearings-only tracking where the motion of a mobile sensor, called the observer, must be planned in order to estimate the position and the velocity of a moving target via bearing measurements. Our analysis of this problem demonstrates that the optimal solutions can be uniquely specified by only two ratios: (i) The distance that the observer can travel along a straight line during the observation period to the relative distance between the observer and the target. (ii) The speed of the observer relative to the speed of the target.

Table of Contents

Abstract	ii
Table of Contents	iv
List of Tables	vii
List of Figures	viii
Acknowledgements	x
1 Introduction	1
1.1 Overview of Thesis	1
1.2 Main Contributions	8
1.3 Related Works	15
1.4 Notation	19
I Management of Fixed Sensors	21
2 Upper Bounds for the Sensor Subset Selection Problem	22

Table of Contents

2.1	Overview	22
2.2	Formulation of the Sensor Subset Selection Problem	25
2.2.1	Structural Properties of the FIM and its Determinant	28
2.3	Characteristics of the Lagrangian Relaxation	34
2.3.1	Outer Approximation Algorithm	36
2.4	Characteristics of the Continuous Relaxation	40
2.4.1	Natural Selection Process	42
2.4.2	Analysis of the Performance of a Greedy Algorithm	52
2.5	Concluding Remarks	55
3	Power Index of a Sensor	56
3.1	Overview	56
3.2	Cooperative Parameter Estimation Game	58
3.3	Computation of Power Indices	61
3.4	Power Indices for Localization and Tracking	68
3.4.1	Localization	69
3.4.2	Tracking	74
3.5	Concluding Remarks	77
II	Management of Mobile Sensors	78
4	Parameterization of Optimal Observer Trajectories	79
4.1	Overview	79

Table of Contents

4.2	Formulation of the Trajectory Design Problem	81
4.3	Invariant Transformation of the Trajectory Design Problem	87
4.4	Concluding Remarks	95
5	Conclusion	97
5.1	Summary of Work Accomplished	97
5.2	Directions for Future Work	99
	Bibliography	102

List of Tables

2.1	Characteristics of the sensor subset selection problem	55
3.1	Power index of sensors in Figure 3.2 for bearings-only localization . .	74
3.2	Power index of sensors in Figure 3.3 for bearings-only localization . .	74
3.3	Power index of sensors in Figure 3.2 for range-only localization	75
3.4	Power index of sensors in Figure 3.2 for bearings-only tracking	77

List of Figures

1.1	Sensor management for localization	5
1.2	Main themes of the thesis	8
1.3	Examples of observable and unobservable scenarios	17
2.1	Geometry of bearings-only localization in three-dimensional space . .	50
2.2	A histogram for the convergence of the natural selection process . . .	52
3.1	Analyzing relative power of sensors for bearings-only localization . . .	58
3.2	A network configuration to study power indices for localization	72
3.3	A network configuration to demonstrate that the closest sensor is not the most informative one	73
3.4	A network configuration to study power indices for tracking	76
4.1	Geometry of bearings-only tracking by an observer	86
4.2	Bounded region for the trajectory design problem	89
4.3	Observer trajectories for tracking a stationary target is ($C_V = \infty$). . .	93
4.4	Observer trajectories for localization.	94

List of Figures

4.5 Observer trajectories for tracking when the target and the observer are
traveling at the same speed. 95

Acknowledgements

I am thankful to my supervisor, Dr. Vikram Krishnamurthy, for his guidance and support in this research. His enthusiasm for tackling new problems and exploring new ideas was always contagious and a source of motivation.

I appreciate the benefits of useful discussions with colleagues in the statistical signal processing lab. Many illusive concepts were clarified for me after these discussions. Finally, I would like to express my gratitude to my dear parents, my sisters and many good friends who encouraged and supported me through this time. I am grateful for a fellowship from the R. Howard Webster Foundation as well as a scholarship from the National Science and Engineering Research Council of Canada that allowed me to complete this program.

Chapter 1

Introduction

1.1 Overview of Thesis

Sensors are widely used in our lives. They provide quantifiable information in the form of measurements about events and real-world conditions which may be otherwise indiscernible to us. Unfortunately, however, the measurements from sensors can be inaccurate and indirect representatives of the quantity that is of interest to us. A primary task is then to infer the value of the quantity of interest *after* making measurements. This is known as *estimation* [1–4], a well-developed subject which we do not delve into here. Instead, we concern ourselves with a process *before* making measurements where we question how sensors can be most efficiently utilized to obtain the most accurate estimates.

We refer to this decision-making process as *sensor management* and develop a number of optimization problems for this purpose. These optimization problems have an objective function that reflects the accuracy of estimation results and some constraints that define feasible actions of sensors. Our knowledge about the quantity that must be estimated is captured by a probability density. This probability density

is also known as the *belief state*. As new measurements become available, this probability density is updated by an estimator. The estimation accuracy is related to the dispersion of this probability density. If the probability density is highly concentrated around a certain value, we can say that the quantity has been estimated with high accuracy. Formulating an optimization problem for sensor management is, therefore, established by *predicting* how this probability density evolves if a certain sequence of actions is given to the sensors. This problem is solved by finding an optimal feasible sequence of actions.

There are several statistical and information-theoretic functions that assign a real number to a probability density reflecting its dispersion (see [5] where some of these functions are described). We call such functions the *metrics of estimation accuracy*. The statistical metrics of estimation accuracy are mainly based on the covariance matrix whereas the information-theoretic metrics are based on the entropy. We work with the statistical metrics of estimation accuracy as one then only needs to study the evolution of the second moment of the probability density into the future.

Ideally, the metrics of estimation accuracy must be defined based on the covariance matrix of the specific estimator used for estimation. However, if we assume that the estimator is *efficient* [3], the metrics of estimation accuracy can be defined for a class of estimators, rather than a specific estimator. Efficiency is a desirable property for an estimator and one can expect that many estimators, at least, asymptotically satisfy this property [2,3]. Essentially, an efficient estimator is characterized by its covariance

matrix when it attains the so-called Cramer-Rao lower bound (CRLB) [1]. The inverse of the CRLB is known as the Fisher Information Matrix (FIM). One other advantage of assuming an efficient estimator is that there are formulas for updating the FIM recursively from one time instant to another [6]. We use these formulas to predict how the FIM evolves for a given sequence of actions and their corresponding measurements. The FIM can also be computed directly for a batch of measurements [1] (see (2.4) and (2.6) where the batch and recursive formulas for computing the FIM are given in the context of the problem discussed there). In the recursive form, the CRLB and FIM are often referred to as the posterior Cramer-Rao lower bound (P-CRLB) and the Bayesian Fisher information matrix (B-FIM).

As a result of the above discussion, we define our metrics of estimation accuracy based on the FIM. The FIM can be scalar when the quantity of interest only consists of a single element. However, in most cases, this quantity consists of several elements and, therefore, the FIM is a matrix. It is possible to formulate an optimization problem for sensor management with a matrix as the objective function (see [7] where optimization problems with matrix-form objective functions and constraints are discussed). However, it is more convenient to work with scalar objective functions. Therefore, we seek functions that assign a real number to the FIM. Such functions must have certain desirable properties. For instance, the real numbers assigned to the matrices must be consistent with a partial ordering on the set of matrices, i.e. if one matrix is larger than another one based on the ordering relation, its correspond-

ing real number must also be larger. There are other desirable properties for these function. One is that these functions must be concave. We discuss the implication of concavity in Chapter 2 and use this property to prove some results there. In [8], an excellent exposition on these properties is provided where it is also shown that *matrix means* are functions that satisfy these properties. Examples of matrix means are the trace of a matrix divided by L , the L -th root of the determinant and the smallest eigenvalue of the matrix where L is the number of elements in the quantity of interest.

Figure 1.1 illustrates the aforementioned concepts in the context of estimating the position of an object by a *sensor network*. Following a common practice, we call this object a *target*. Throughout this dissertation, we often discuss sensor management for estimating the position (and the velocity) of a target. Estimating the position of a stationary target is also known as *localization* while estimating the position and the velocity of a moving target is known as *tracking*.

The optimization problem for sensor management can be formulated in various forms. In its most general form, a sensor management problem is a dynamic stochastic optimization problem. The problem is formulated based a dynamic stochastic model for the the quantity of interest, e.g. the position of the target, and a stochastic measurement model for sensors. These models determine the evolution of the probability density of the quantity of interest which is the actual underlying state in this dynamic stochastic optimization problem. In order to solve this optimization problem, we must find a policy that assigns an optimal action to each sensor for every

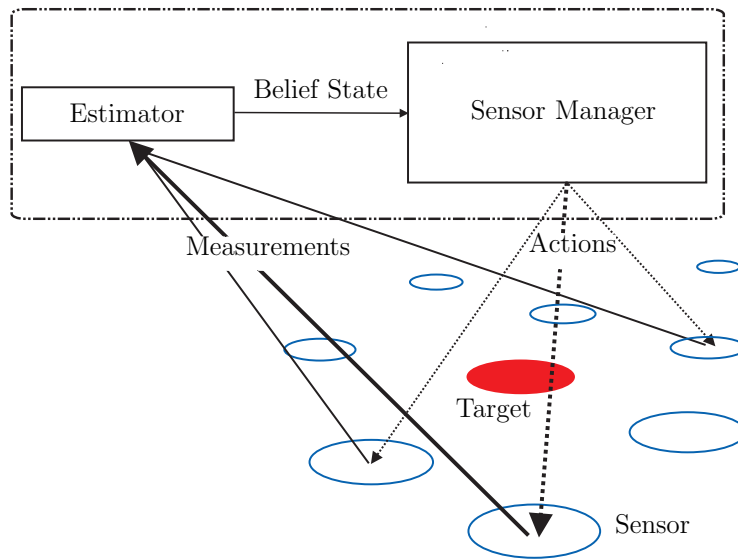


Figure 1.1: Sensor management for estimating the position of a target. The sensor manager predicts how the belief state evolves if a sequence of actions is given to the sensors. The sensor manager then determines the optimal actions and communicate these actions to the sensors. New measurements are then made by the sensors and processed by the estimator to update the belief state.

possible probability density at any given time instant. Finding such policies conveys a *closed-loop* approach to sensor management. Unfortunately, it is rarely possible to obtain closed-form expressions for such policies. Nor is it possible to compute them numerically as the domain set of these policies is an uncountable set of probability densities. In this regard, the general sensor management problem resembles partially-observable Markov decision processes (POMDP) [9] and is plagued with the same difficulties in order to be solved.

There is, however, another layer of complexity in sensor management problems that does not exist in POMDPs. In sensor management problems, the objective function, i.e. the metric of estimation accuracy, is a non-linear function of the underlying

probability density whereas in conventional POMDPs, the objective function is a linear function of the probability density. This non-linearity causes extra difficulties in solving sensor management problems which, for instance, has been discussed in [10,11] in the context of sensor management for hidden Markov models.

For certain combinations of dynamic and measurement models, however, the evolution of the probability density or at least its covariance matrix is not affected by actual measurements. In such cases, the sensor management problem becomes a dynamic deterministic optimization problem and hence an *open-loop* solution is optimal. As an example, consider a case where both the quantity of interest and sensors have a linear Gaussian model. It is well known that for this case the Kalman filter is the optimal and efficient estimator and that the covariance matrix of the Kalman filter can be calculated a priori [12,13]. Therefore, there is no advantage in delaying actions and observing measurements. As another example, consider a deterministic dynamic model for the evolution of the quantity of interest and a measurement model for sensors that consists of a non-linear function of the value of this quantity and an additive Gaussian measurement error. It can be shown that for this case, the evolution of the FIM is deterministic and independent of actual measurements [4].

We will focus on such deterministic optimization problems throughout this dissertation. We discuss two problems in this regard. The first problem is in fact a static optimization problem which involves selecting a pre-specified number of sensors in a network of fixed sensors in order to estimate the value of a time-invariant *parameter*.

In the other problem, we consider designing a trajectory for a mobile sensor in order to track a moving target. As we will see, even finding an optimal solution for these deterministic optimization problems is computationally very demanding.

In connection with the first problem, we also study a problem that is posed from the perspective of the target whose position or other critical parameters is to be estimated by the sensor network. In this problem, we assume: (i) The target has some capabilities to attack and destroy a limited number of sensors or manipulate the information sent to them. (ii) It intends to use these capabilities to adversely affect the estimate of its parameter. It then must be decided which sensors to be attacked. Ideally, the answer is the sensors that are to be selected for making measurements but it is unlikely that the target exactly knows these sensors. In the absence of such knowledge, we show how a notion borrowed from cooperative game theory can be employed to identify those sensors that can be potentially more informative.

In the rest of this chapter, we summarize the main contributions of this dissertation, provide a literature review, list the supporting publications and finally introduce the common notation.

This dissertation is divided into two main parts based on whether sensor management is concerned with fixed or mobile sensors. In the first part, which comprises Chapters 2 and 3, we focus on fixed sensors. In the second part, which comprises Chapter 4, we concentrate on mobile sensors.

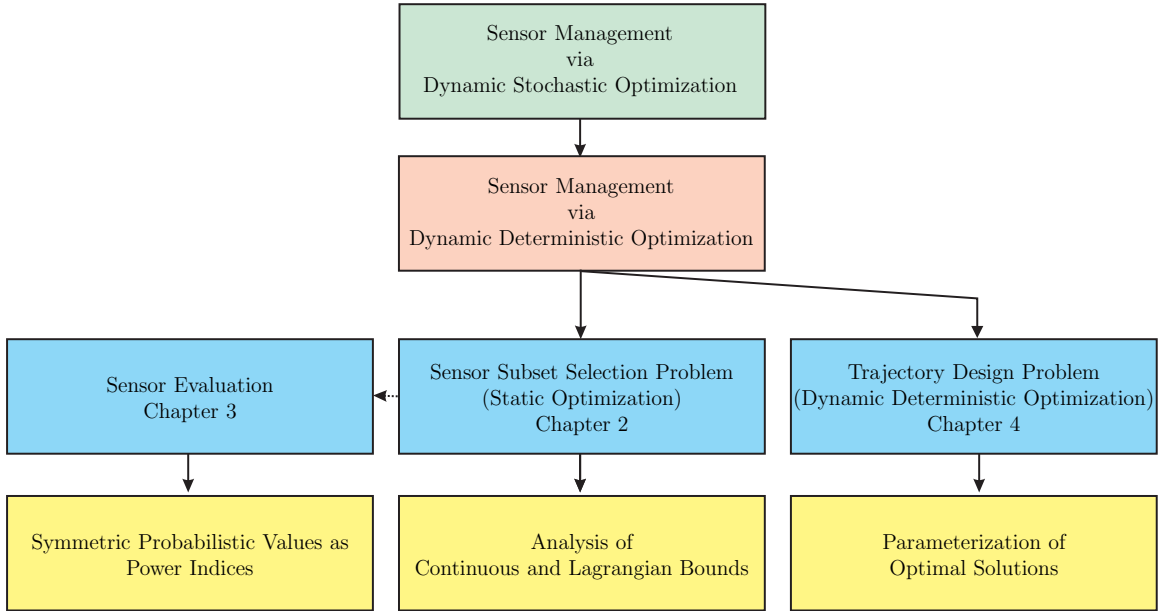


Figure 1.2: Illustration of the main themes explored in this dissertation.

1.2 Main Contributions

Figure 1.2 illustrates the main themes of this dissertation. It also shows the interaction between these themes and indicates how they have been addressed in this dissertation. In the following, we provide more details on these themes and the contributions made.

1. Analysis of Lagrangian and Continuous Relaxations of the Sensor Subset Selection Problem (Chapter 2): For a network of fixed sensors engaged in estimating the unknown value of a time-invariant parameter, a sensor management problem is how to select M out of the total of N sensors such that a metric of estimation accuracy is maximized. This problem is called the *sensor subset selection problem*. Since the parameter is time-invariant and sensors are also fixed, the sensor subset selection problem is a static combinatorial optimization problem i.e. a 0-1 in-

teger program. This problem can be considered as an instance of resource allocation problems [14] or knapsack problems [15] with special characteristics of its own. In Chapter 2, we provide a through analysis of the Lagrangian and continuous relaxation of the sensor subset selection problem and outline algorithms for computing these bounds.

Although the sensor subset selection problem is a static optimization problem, as a combinatorial optimization problem, it is still difficult to solve. Carrying out an exhaustive search is prohibitively expensive because $\binom{N}{M}$ subsets must be evaluated for this purpose. Branch-and-bound algorithms [15,16] are the alternatives which can potentially eliminate the need for an exhaustive search. However, for a branch-and-bound algorithm, we need to have good upper bounds for the problem. An upper bound is often obtained by relaxing some of the constraints of the problem and solving the relaxation problem. A solution of the relaxation problem can also be exploited as a benchmark for evaluating heuristic algorithms and as an initial point for local search algorithms. Motivated by these advantages, we study the Lagrangian and continuous relaxations of the sensor subset selection problem with the determinant of FIM as the metric of estimation accuracy.

We first show that the determinant of the FIM can be represented by a homogeneous polynomial as a function of decision variables (see Proposition 2.2.2). We further show that the determinant of the FIM is a supermodular function (see Proposition 2.2.3) and hence the sensor subset selection problem with this metric is a

supermodular knapsack problem. Gallo and Simeone [17] have shown that the Lagrangian relaxation of a supermodular knapsack problem can be solved in polynomial time. Mainly, we show that the dual function is a piecewise linear convex function with at most N breakpoints in the Lagrangian relaxation of the sensor subset selection problem. Therefore, the minimum of the dual function can be found by an outer approximation algorithm with at most N iterations. In each iteration of the outer approximation algorithm, the dual function must be evaluated at one point. The evaluation of the dual function requires finding the maximum of a supermodular function which is known to be computable in polynomial time [18].

However, we also show that the continuous bound is tighter than the Lagrangian bound while it can be still found in polynomial time. This latter statement is established by noting that the determinant shares a unique maximizer with its L -th root (where L is the number of elements in the parameter that must be estimated) and the log-determinant and the maximization of these two latter functions can be done in polynomial time because of their concavity.

While standard algorithms for constrained convex optimization [7] can be exploited in order to solve the continuous relaxation of the sensor subset selection problem, we devise a very efficient algorithm for solving the continuous relaxation of a variation of the sensor subset selection problem. In this variation, called the *measurement allocation problem*, sensors are allowed to make more than one measurement and the cost of making each measurement is equal for all sensors. Knowing that the de-

terminant can be represented by a homogenous polynomial, we then show that what is known as the *natural selection process* in population genetics [19] shares a unique stable equilibrium with the optimal allocation vector in the measurement allocation problem. Hence, by updating an initial allocation vector via the natural selection process, a sequence of allocation vectors are generated that eventually converges to the optimal allocation vector.

2. Symmetric Probabilistic Values for Identifying Informative Sensors

(Chapter 3): We establish a novel approach for identifying the informative sensors under the assumptions of Chapter 2 from the perspective of a target whose critical parameters is to be estimated by the sensor network. We assume that the target does not exactly know how many and which sensors are to be selected but it assigns a probability to each subset of sensors reflecting its belief about the selection of that subset.

We show that a *power index*, based on *symmetric probabilistic values* [20] in cooperative game theory, can be assigned to each sensor to determine how informative the sensor is relative to others. To establish this result, we first point out that parameter estimation by a sensor network can be modeled as a cooperative game where each subset is a cooperation opportunity that generates a certain estimation accuracy. It is known that in a cooperative game, symmetric probabilistic values or *semivalues*, computed as the weighted average of the marginal contributions of a player to all subsets, are indicators of the relative power of players. The weights for computing a

symmetric probabilistic value are determined by how likely it is for a subset to be actually realized as the outcome of cooperation. In the context of our discussion, these weights are determined by the probabilities that the target assigns to the subsets.

Computing the power index of a sensor involves a summation over all subsets of sensors. As such, the difficulty of computation increases exponentially with the number of sensors. However, we show that by choosing the determinant of the FIM as the metric of estimation accuracy and for the probabilities that only depends on the size of subsets, the computation of power indices can be done in polynomial time. This result is mainly achieved because the determinant of the FIM can be compactly represented by a (hyper-)matrix. Asserting that the determinant of the FIM has a (hyper-)matrix representation is a consequence of the fact that the determinant of the FIM can be represented by a homogenous polynomial, a result that we establish in Chapter 2. The (hyper-)matrix representation for cooperative games can be regarded as a generalization of the (hyper-)graph representation for cooperative games. Deng and Papadimitriou [21] have studied cooperative games with a (hyper-)graph representation. By extending their results, we provide explicit formulas for computing two well-known power indices: the *Banzhaf value* [22] and the *Shapley value* [23]. We also provide a formula for computing the power index when only subsets with a certain size are to be selected. From this formula, the power index for any arbitrary probability distribution over the size of subsets can be computed.

3. Parameterization of optimal observer trajectories for bearings-only tracking (Chapter 4): For a mobile sensor, a sensor management problem is how to plan the motion of the sensor such that a metric of estimation accuracy is maximized. This problem is called the *trajectory design problem*. As opposed to the sensor subset selection problem, a trajectory design problem is not a static optimization problem. A dynamic component always exists due to the presence of a mobile sensor. We consider the trajectory design problems for bearings-only tracking [3]. In *bearings-only tracking*, the position and the velocity of a moving target is estimated by measuring the direction-of-arrival (DOA) of an acoustic, electromagnetic or seismic signal emitted from the target to the sensor(s). We show that the optimal solutions of the trajectory design problem for bearings-only tracking can be parameterized by only two ratios.

In our formulation of the problem, we assume that the mobile sensor, which is also called the *observer*, travels at a constant speed but the direction of its motion, i.e. its *course*, can be controlled as desired. We also assume a constant-velocity target, i.e. a target that has a deterministic model with fixed speed and course during the observation period. As a result, it suffices to only estimate the initial position of the target and its velocity, two time-invariant parameters. This ensures that an open-loop solution is optimal.

There is, however, no closed-form expression known for the open-loop solution. The problem can be solved numerically by dynamic programming if the FIM is dis-

cretized or alternatively by approximating the course control function by a partial sum of some basis functions such as the Chebyshev polynomials [24] and finding the coefficients of these basis functions.

In general, one numerical solution must be obtained for the following set of problem-specific parameters: the initial positions of the target and the observer, the speeds of the target and the observer and the observation period. However, in the case of bearings-only localization, Hammel et al. [25] have noted that these parameters can be encompassed in what is known as the *range-to-baseline ratio*. In other words, the optimal course control function for bearings-only localization can be uniquely specified by the range-to-baseline ratio. This is the ratio of the distance that the observer can travel directly towards the target during the observation period relative to the initial distance between the target and the observer.

We provide a more generalized result for bearings-only tracking where we show that the optimal course control function in this case can be parameterized by the range-to-baseline ratio and the *velocity ratio* which is the ratio of the speed of the observer relative to the speed of the target. In our proof, we show that the FIM is scaled only by a constant if the speed of the observer, the speed of the target, the observation period and the distance between the target and the observer are all simultaneously scaled such that the range-to-baseline ratio and the speed ratio both remain constant. This proves that the optimal course control function is invariant for any metric of estimation accuracy defined based on the FIM.

1.3 Related Works

The sensor subset selection problem has gained considerable attention in recent years [5, 26–31]. A closely related problem is the sensor deployment (or placement) problem [32–35] where the position of sensors must be determined in order to obtain the maximum coverage of a field. For a general survey of sensor selection problems in sensor networks, the reader can consult with [36].

Many authors have studied the *single* sensor selection problem [5, 26–28]. Kaplan in [29] considers the sensor subset selection problem. He proposes a greedy algorithm [15] for solving this problem which consists of a fill-up phase and an exchange phase (see Algorithm 2.3). In Chapter 2, we verify the effectiveness of this algorithm for solving the sensor subset selection problem in the context of bearings-only localization by comparing the performance of the algorithm with the continuous bound of the problem. In [30], Kaplan also investigates a localized algorithm for the sensor subset selection problem where only sensors in a critical range are considered for making measurements. In [31], Tharmarasa et al. study the sensor subset selection problem for multi-target tracking where selected sensors remain active over a pre-specified number of time instants. The authors solve the continuous relaxation of this problem and use a local search algorithm to obtain a discrete solution.

The sensor subset selection problem has also been formulated in other forms. Most notably, in [37], Chhetri et al. exchange the role of the objective function and the constraint and aim at finding the minimum number of sensors that satisfy a desired

level of estimation accuracy. They use the trace of the FIM as the metric of estimation accuracy and obtain a linear objective function and a quadratic constraint (when the number of elements in the parameter is 2). Their solution approach is based upon converting this problem to a mixed-integer program and then solving the program by a branch-and-bound algorithm.

Although cooperative game theory has been considered in other works for signal processing and telecommunication applications (see [38] for packet forwarding or [39] for multiple source coding), the context for which we use cooperative game theory in Chapter 3 differs from other works in the literature. Non-cooperative game theoretic concepts have been extensively discussed in sensor networks for decentralized decision-making and learning but those are clearly different from the cooperative game theoretic concepts that we discuss here.

The trajectory design problems for bearings-only localization and tracking are well-studied subjects with their origin in submarine tracking and aircraft surveillance [40,41]. These problems have been studied within two frameworks: deterministic and stochastic, with earlier works [42–47] emphasizing on a deterministic framework. In a deterministic framework, it is assumed that measurements are error-free and the motion of the target is deterministic. It is then investigated under what conditions the position of the target can be uniquely determined from bearing measurements. The general conclusion is that the observer needs to *out-maneuver* the target in order that a one-to-one mapping can be established between the measurements and

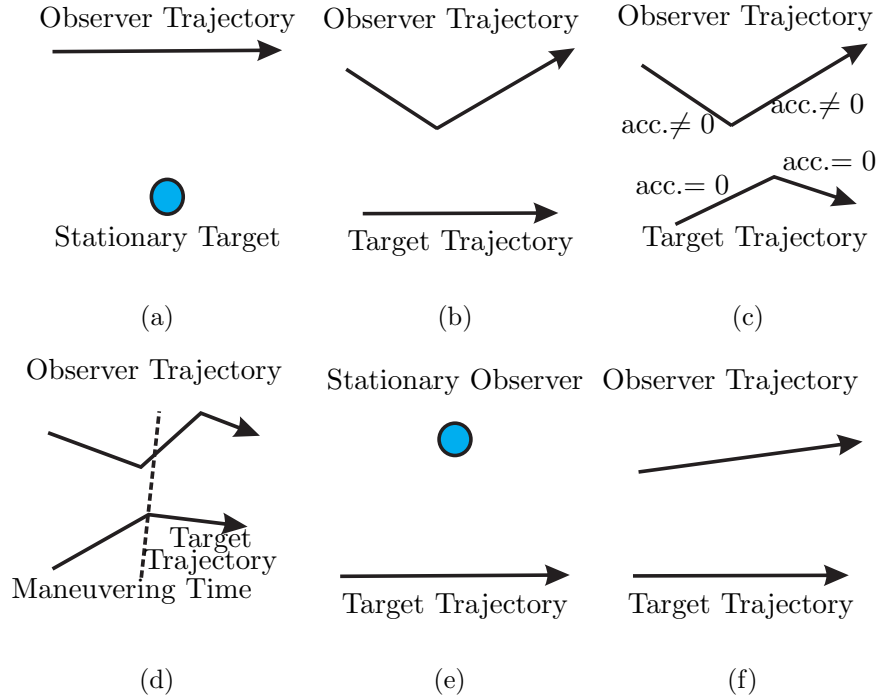


Figure 1.3: Examples of observable and unobservable scenarios for bearings-only localization and tracking. In the scenarios shown in (a), (b), (c) and (d), the position of the target can be determined from bearing measurements. However, in the scenarios shown in (e) and (f), this is not possible. With the exception of the scenario shown in (c) where the acceleration is non-zero for the observer, the speed of the target and the speed of the observer are assumed to be constant during the observation period in all scenarios.

the position of the target. More formally stated, a necessary condition is that the observer's trajectory must have higher non-zero derivatives with respect to time than the target's trajectory [43]. Illustrative examples for various scenarios are shown in Figure 1.3.

In a stochastic framework [25, 48–55], which we also pursue in Chapter 4, an additive Gaussian measurement error is present in the measurement model and the target is assumed to have a stochastic model. We earlier mentioned the contribution

of [25]. As a pioneer work [48], Fawcett considers the effect of a single maneuver of the observer on the FIM when the target has a constant-velocity model. Authors in [49] introduce the trace of the FIM as a metric of estimation accuracy in the trajectory design problem and employ a gradient-based method to solve the small instances of the problem for constant-velocity targets. In [52], authors explore the use of function approximation and the differential inclusion method when there are additional constraints imposed on the observer trajectory.

As opposed to previous works, Logothetis et al. [50, 51] consider a target with a stochastic model, namely a nearly-constant velocity target with random instantaneous accelerations. Along with other methods, they use a very crude discretization of the covariance matrix of the extended Kalman filter to solve the trajectory design problem with dynamic programming. In [53], by discretizing measurements and considering a Markov chain model for the target, the authors formulate the trajectory design problem as a POMDP. We previously mentioned this approach in the context of a general sensor management problem. In [54], the trajectory design problem for multiple observers is considered and myopic solutions are numerically computed. Finally, the authors in [55] consider a vision-based target localization application where the observer has a limited field of view and, therefore, additional constraints must be taken into account for trajectory design.

1.4 Notation

Throughout this dissertation, we use lower-case bold serif letters, e.g. \mathbf{x} , for vectors and upper-case bold serif letters, e.g. \mathbf{X} , for (hyper-)matrices. A hyper-matrix in our terminology is an array of numbers that has more than two dimensions. If we specifically work with a two-dimensional array, we use the term matrix. We also use lower-case Italic Roman letters for scalar variables, e.g. m . Parameters fixed at a given value are usually denoted by an upper-case Roman letter, e.g. N . The elements of a vector \mathbf{x} and a (hyper-)matrix \mathbf{X} are denoted by x_i and $x_{ijk\dots}$ unless otherwise stated. An alternative method for representing a vector or a (hyper-)matrix is to place brackets around the general element of that vector or (hyper-)matrix, e.g. $[x_i]$ and $[x_{ijk\dots}]$. If a vector or matrix has an index itself, parentheses are used to separate the index of the vector or the matrix from its elements. For instance, the elements of \mathbf{x}_i are denoted by $(x_i)_j$. The value of a variable at a given time instant is denoted by its symbol followed by the time stamp embraced with parentheses, e.g. $x(k)$.

The derivative of a vector-valued function \mathbf{f} with respect to \mathbf{x} is denoted by $\frac{\partial \mathbf{f}}{\partial \mathbf{x}}$ where the element in row i and column j is given by $\frac{\partial f_i}{\partial x_j}$. Although occasionally dropped, the transpose operator is denoted by T . The transpose of the derivative of a vector-valued function is denoted by $\frac{\partial^T \mathbf{f}}{\partial \mathbf{x}}$.

We use calligraphic letters for representing sequences or sets, e.g. $\mathcal{X} = \{x(k)\}_{k=0}^N$ for a sequence or \mathcal{N} for a set, with the exception of universally defined sets such as the set of integers \mathbb{Z} , or the set of real numbers \mathbb{R} . The non-negative elements of a set

is denoted by the set symbol and a plus sign attached to it, e.g. \mathbb{R}^+ denotes the set of non-negative real numbers. Similarly, the set of strictly positive elements of a set is denoted by the set symbol and two plus signs attached to it, e.g. \mathbb{Z}^{++} denotes the set of positive integers. Symbols such as \mathcal{N}^N denote the N-time Cartesian product of set \mathcal{N} . The cardinality or the size of set \mathcal{N} is denoted by $|\mathcal{N}|$.

Expectation, when taken over probability density $p(\mathbf{x})$, is denoted by $\mathbb{E}_{p(\mathbf{x})}$. The Gaussian distribution is sometimes denoted by $N(\mu, \mathbf{R})$ where μ and \mathbf{R} are the mean and the covariance matrix. The determinant, the trace and the set of eigenvalues of matrix \mathbf{X} are denoted by $\det(\mathbf{X})$, $\text{trace}(\mathbf{X})$, $\lambda(\mathbf{X})$, respectively. Other symbols are defined as needed.

Part I

Management of Fixed Sensors

Chapter 2

Upper Bounds for the Sensor

Subset Selection Problem

2.1 Overview

In recent years, networks of fixed sensors have been widely employed for pervasive surveillance in hostile and remote environments. These networks are often set up to monitor fire, wildlife habitats, boarder activities, etc. With very limited onboard energy and long periods of monitoring, it is crucial that the sensors are efficiently used in these networks.

In this chapter, we consider the sensor subset selection problem which, to some extent, addresses this issue. In this problem, M out of N sensors must be selected to estimate the value of a time-invariant parameter such that a metric of estimation accuracy is maximized. The constraint on the number of sensors that can be selected reflects a concern on the amount of the energy that the network can consume.

While various heuristic algorithms are proposed for solving the sensor subset selection problem and its variation [5, 26–31], we fully analyze the characteristics of the

Lagrangian and continuous relaxations of this problem when the determinant of the (B-)FIM is used as the metric of estimation accuracy. The results of this analysis can be used in branch-and-bound algorithms and local search algorithms as well as for evaluating the performance of heuristic algorithms. In the continuous relaxation, the (binary) decision variables are allowed to take on any real number between 0 and 1. In the Lagrangian relaxation, the constraints are moved into the objective function with the decision variables still restricted to be binary.

There are two determinant-based functions of the (B-)FIM that are commonly used in the literature as the metrics of estimation accuracy. The first one is the L -th root of the determinant, where L is the number of elements in the parameter. This function is a matrix mean [8] and, as such, satisfies a number of desirable properties. One of these properties is its concavity in the continuous domain. The concavity property guarantees that estimation accuracy is not improved by interpolation of measurements. The logarithm of the determinant (log-determinant) of the (B-)FIM is another determinant-based metric of estimation accuracy. The motivation for using the log-determinant stems from information theory where the log-determinant of the covariance matrix of a Gaussian distribution quantifies its entropy [56]. The log-determinant is also a concave function that induces a partial ordering on the set of matrices identical to the one induced by the L -th root of the determinant. Hence, the sensor subset selection problem acquires the same characteristics with both functions.

It is well known that if the objective function of a combinatorial optimization

is concave (in the continuous domain) and its feasible region is convex (again in the continuous domain), then its Lagrangian bound is tighter than its continuous bound [15]. Therefore, for the sensor subset selection problem with the L -th root of the determinant or the log-determinant as its objective function, the Lagrangian bound is tighter than the continuous bound. Unfortunately, however, while the continuous bound can be computed in polynomial time via a convex optimization algorithm [7], computing the Lagrangian bound remains as difficult as finding the optimal solution of the original problem.

The determinant of the (B-)FIM also induces an ordering on the set of matrices which is identical to the ordering induced by L -th root of the determinant and the log-determinant but the determinant is not a concave function. We prove several properties for the determinant of the (B-)FIM and study how these properties can be exploited for solving the Lagrangian and continuous relaxations of the sensor subset selection problem and its variations. We show that both bounds can be computed in polynomial time in this case and that the continuous bound is tighter than the Lagrangian bound (as opposed to what holds for the other two functions).

We also take advantage of a representation of the determinant as a homogenous polynomial to outline an algorithm for solving the continuous relaxation of a variation of the sensor subset selection problem where sensors are allowed to make more than one measurement.

This chapter is organized as follows. Section 2.2 formulates the sensor subset

selection problem and proves several important properties of its objective function. Sections 2.3 and 2.4 discuss the characteristics of the Lagrangian bound and the characteristics of the continuous bound, respectively. Section 2.5 gives some concluding remarks.

2.2 Formulation of the Sensor Subset Selection Problem

In this section, we formally define the sensor subset selection problem and derive several important properties of its objective function that will be used subsequently in this chapter and the following one.

We start by denoting the set of all sensors by $\mathcal{N} = \{1, 2, \dots, N\}$ and the parameter vector whose value is unknown to sensors and must be estimated by $\mathbf{p} \in \mathbb{R}^L$. The measurement model for sensors is characterized by

$$\mathbf{z}_i = \mathbf{h}_i(\mathbf{p}) + \mathbf{v}_i \tag{2.1}$$

for $i = 1, 2, \dots, N$ where $\mathbf{z}_i \in \mathbb{R}^D$ is the measurement vector of the i -th sensor, \mathbf{h}_i is a vector-valued differentiable function of \mathbf{p} , and $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N$ are mutually-independent zero-mean Gaussian random error vectors. The covariance matrix of \mathbf{v}_i is denoted by \mathbf{R}_i . Later in this section, we comment on the implications of using a more general measurement model. As we will see, the most critical assumption for

proving our results is that the measurements should be mutually-independent.

For optimization problems, it is often more convenient to have a vector representation of sets. In this chapter, we, therefore, mostly follow this convention. We identify set $\mathcal{S} \subset \mathcal{N}$ by vector $\mathbf{x} \in \{0, 1\}^N$ where

$$x_i = \begin{cases} 1 & \text{if } i \in \mathcal{S}, \\ 0 & \text{if } i \notin \mathcal{S} \end{cases} \quad (2.2)$$

for $i = 1, 2, \dots, N$. The vectors corresponding to sets \mathcal{N} , $\{i\}$, and \emptyset are denoted by \mathbf{e} , \mathbf{e}_i and $\mathbf{0}$, respectively. The measurements that are collected from the selected sensors are denoted by $\mathcal{Z}(\mathbf{x})$ where \mathbf{x} is given by (2.2).

Definition 2.2.1 *The sensor subset selection problem is defined as solving one of the following:*

1. If \mathbf{p} is regarded as a non-random parameter [1],

$$\begin{aligned} & \max_{\mathbf{x} \in \{0,1\}^N} \det(\mathbf{J}(\mathbf{x})) \\ & \text{such that } \mathbf{c}^T \mathbf{x} \leq M, \end{aligned} \quad (2.3)$$

where $\mathbf{c} \in \mathbb{R}^{++N}$, $M \in \mathbb{R}^{++}$ and

$$\mathbf{J}(\mathbf{x}) = \mathbb{E}_{p(\mathcal{Z}(\mathbf{x})|\mathbf{p})} \left\{ \frac{\partial^T}{\partial \mathbf{p}} \log(p(\mathcal{Z}(\mathbf{x})|\mathbf{p})) \frac{\partial}{\partial \mathbf{p}} \log(p(\mathcal{Z}(\mathbf{x})|\mathbf{p})) \right\} \quad (2.4)$$

for $\mathbf{x} \neq \mathbf{0}$ and $\mathbf{J}(\mathbf{0}) = 0$. Furthermore, it is assumed that $\sum_{i=1}^N c_i > M > 0$.

2. If \mathbf{p} is regarded as a random parameter, i.e. a prior density $p(\mathbf{p})$ exists for \mathbf{p} ,

$$\begin{aligned} & \max_{\mathbf{x} \in \{0,1\}^N} \det(\mathbf{J}_B(\mathbf{x})) \\ & \text{such that } \mathbf{c}^T \mathbf{x} \leq M, \end{aligned} \quad (2.5)$$

where $\mathbf{c} \in \mathbb{R}^{++N}$, $M \in \mathbb{R}^{++}$ such that $\sum_{i=1}^N c_i > M > 0$. Furthermore,

$$\mathbf{J}_B(\mathbf{x}) = \mathbf{Q} + \mathbb{E}_{p(\mathbf{p})}\{\mathbf{J}(\mathbf{x})\} \quad (2.6)$$

where

$$\mathbf{Q} = \mathbb{E}_{p(\mathbf{p})} \left\{ \frac{\partial}{\partial \mathbf{p}}^T \log(p(\mathbf{p})) \frac{\partial}{\partial \mathbf{p}} \log(p(\mathbf{p})) \right\}. \quad (2.7)$$

Remark Matrices $\mathbf{J}(\mathbf{x})$ and $\mathbf{J}_B(\mathbf{x})$ are the FIM and the B-FIM, respectively. If we denote by $\hat{\mathbf{p}}(\mathbf{x})$ an arbitrarily-chosen estimator of parameter \mathbf{p} when measurements are collected from the non-zero elements of \mathbf{x} , then the inverse of $\mathbf{J}(\mathbf{x})$ provides a lower bound for the covariance matrix of $\hat{\mathbf{p}}(\mathbf{x})$ in the sense that if $\mathbf{C}(\mathbf{x})$ denotes the covariance matrix, $\mathbf{C}(\mathbf{x}) - \mathbf{J}^{-1}(\mathbf{x})$ is positive semi-definite. Similarly, the inverse of $\mathbf{J}_B(\mathbf{x})$ provides a lower bound for the covariance matrix of an arbitrarily-chosen estimator of parameter \mathbf{p} when, in addition to $\mathcal{Z}(\mathbf{x})$, prior density $p(\mathbf{p})$ is also used for estimation. As a result, both $\det(\mathbf{J}(\mathbf{x}))$ and $\det(\mathbf{J}_B(\mathbf{x}))$ can be regarded as the metrics of estimation accuracy.

For the rest of this chapter, we assume that the sensor subset selection problem

is given by (2.3). However, most results either directly hold or can be worked out similarly for (2.5). We make a remark in this regard later.

2.2.1 Structural Properties of the FIM and its Determinant

We now present three results on the nature of the FIM and its determinant. These results will be exploited in Sections 2.3 and 2.4 for computing the upper bounds. In Proposition 2.2.1, we show that the FIM can be expanded as a linear combination of N positive semi-definite matrices. In Proposition 2.2.2, we show that the determinant of the FIM can be expanded as a homogeneous polynomial. Finally, from Proposition 2.2.3, it can be deduced that the determinant of the FIM is supermodular.

Proposition 2.2.1 *For the measurement model (2.1), it holds that*

$$\mathbf{J}(\mathbf{x}) = \sum_{i=1}^N x_i \mathbf{S}_i \quad (2.8)$$

where $\mathbf{S}_1, \mathbf{S}_2, \dots, \mathbf{S}_N$ are positive semi-definite matrices given by

$$\mathbf{S}_i = \frac{\partial^T \mathbf{h}_i}{\partial \mathbf{p}} \mathbf{R}_i^{-1} \frac{\partial \mathbf{h}_i}{\partial \mathbf{p}}. \quad (2.9)$$

PROOF Since measurement error vectors $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N$ are mutually-independent,

we can write $p(\mathcal{Z}(\mathbf{x})|\mathbf{p})$ as

$$p(\mathcal{Z}(\mathbf{x})|\mathbf{p}) = \prod_{i \in \mathcal{S}} p(\mathbf{z}_i|\mathbf{p}) \quad (2.10)$$

where \mathcal{S} is the set corresponding to vector \mathbf{x} . It follows that

$$\begin{aligned} \mathbf{J}(\mathbf{x}) &= \mathbb{E}_{p(\mathcal{Z}(\mathbf{x})|\mathbf{p})} \left\{ \sum_{i \in \mathcal{S}} \frac{\partial}{\partial \mathbf{p}}^T \log(p(\mathbf{z}_i|\mathbf{p})) \sum_{i \in \mathcal{S}} \frac{\partial}{\partial \mathbf{p}} \log(p(\mathbf{z}_i|\mathbf{p})) \right\} \\ &= \sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{S}} \mathbb{E}_{p(\mathcal{Z}(\mathbf{x})|\mathbf{p})} \left\{ \frac{\partial}{\partial \mathbf{p}}^T \log(p(\mathbf{z}_i|\mathbf{p})) \frac{\partial}{\partial \mathbf{p}} \log(p(\mathbf{z}_j|\mathbf{p})) \right\} \\ &= \sum_{i \in \mathcal{S}} \mathbb{E}_{p(\mathbf{z}_i|\mathbf{p})} \left\{ \frac{\partial}{\partial \mathbf{p}}^T \log(p(\mathbf{z}_i|\mathbf{p})) \frac{\partial}{\partial \mathbf{p}} \log(p(\mathbf{z}_i|\mathbf{p})) \right\} \\ &\quad + \sum_{i \in \mathcal{S}} \sum_{\substack{j \in \mathcal{S} \\ j \neq i}} \mathbb{E}_{p(\mathbf{z}_i|\mathbf{p})} \left\{ \frac{\partial}{\partial \mathbf{p}}^T \log(p(\mathbf{z}_i|\mathbf{p})) \right\} \mathbb{E}_{p(\mathbf{z}_j|\mathbf{p})} \left\{ \frac{\partial}{\partial \mathbf{p}} \log(p(\mathbf{z}_j|\mathbf{p})) \right\} \end{aligned} \quad (2.11)$$

where in the last line we have used the fact that $\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_N$ are mutually-independent.

We next note that because \mathbf{v}_i has a Gaussian distribution, we can write

$$\frac{\partial}{\partial \mathbf{p}} \log(p(\mathbf{z}_i|\mathbf{p})) = (\mathbf{z}_i - \mathbf{h}_i)^T \mathbf{R}_i^{-1} \frac{\partial \mathbf{h}_i}{\partial \mathbf{p}}. \quad (2.12)$$

Hence,

$$\mathbb{E}_{p(\mathbf{z}_i|\mathbf{p})} \left\{ \frac{\partial}{\partial \mathbf{p}} \log(p(\mathbf{z}_i|\mathbf{p})) \right\} = 0 \quad (2.13)$$

and $\mathbf{J}(\mathbf{x})$ can be written as

$$\begin{aligned}
 \mathbf{J}(\mathbf{x}) &= \sum_{i \in \mathcal{S}} \mathbb{E}_{p(\mathbf{z}_i|\mathbf{p})} \left\{ \frac{\partial}{\partial \mathbf{p}}^T \log(p(\mathbf{z}_i|\mathbf{p})) \frac{\partial}{\partial \mathbf{p}} \log(p(\mathbf{z}_i|\mathbf{p})) \right\} \\
 &= \sum_{i \in \mathcal{S}} \frac{\partial^T \mathbf{h}_i}{\partial \mathbf{p}} \mathbf{R}_i^{-1} \mathbb{E}_{p(\mathbf{z}_i|\mathbf{p})} \left\{ (\mathbf{z}_i - \mathbf{h}_i)(\mathbf{z}_i - \mathbf{h}_i)^T \right\} \mathbf{R}_i^{-1} \frac{\partial \mathbf{h}_i}{\partial \mathbf{p}} \\
 &= \sum_{i \in \mathcal{S}} \frac{\partial^T \mathbf{h}_i}{\partial \mathbf{p}} \mathbf{R}_i^{-1} \frac{\partial \mathbf{h}_i}{\partial \mathbf{p}} \\
 &= \sum_{i=1}^N x_i \frac{\partial^T \mathbf{h}_i}{\partial \mathbf{p}} \mathbf{R}_i^{-1} \frac{\partial \mathbf{h}_i}{\partial \mathbf{p}}.
 \end{aligned} \tag{2.14}$$

Proposition 2.2.2 *If $\mathbf{S} = \sum_{i=1}^N x_i \mathbf{S}_i$ where $\mathbf{S}_1, \mathbf{S}_2, \dots, \mathbf{S}_N$ are $L \times L$ matrices, it*

holds that

$$\det(\mathbf{S}) = \sum_{a_1=1}^N \sum_{a_2=1}^N \cdots \sum_{a_L=1}^N w_{a_1 a_2 \dots a_L} x_{a_1} x_{a_2} \cdots x_{a_L} \tag{2.15}$$

where

$$w_{a_1 a_2 \dots a_L} = \det \left(\begin{bmatrix} (\mathbf{S}_{a_1})_{.1} & (\mathbf{S}_{a_2})_{.2} & \cdots & (\mathbf{S}_{a_L})_{.L} \end{bmatrix} \right) \tag{2.16}$$

with $(\mathbf{S}_i)_{.j}$, for $1 \leq i \leq N$ and $1 \leq j \leq L$, denoting the j -th column of matrix \mathbf{S}_i .

PROOF This is an immediate consequence of expressing the determinant of \mathbf{S} as

$$\det(\mathbf{S}) = \det \left(\begin{bmatrix} \sum_{a_1=1}^N x_{a_1} (\mathbf{S}_{a_1})_{.1} & \sum_{a_2=1}^N x_{a_2} (\mathbf{S}_{a_2})_{.2} & \cdots & \sum_{a_L=1}^N x_{a_L} (\mathbf{S}_{a_L})_{.L} \end{bmatrix} \right) \tag{2.17}$$

and noting that the determinant of a matrix is a multi-linear function of the columns of the matrix. ■

Proposition 2.2.2 states that $w_{a_1 a_2 \dots a_L}$ is the determinant of a matrix that is constructed by having the first column of \mathbf{S}_{a_1} as its first column, the second column of \mathbf{S}_{a_2} as its second column and so on.

Remark We use (hyper-)matrix $\mathbf{W} = [w_{a_1 a_2 \dots a_L}]$ to compactly represent the coefficients defined by (2.16). This (hyper-)matrix has L dimensions and N elements in each dimension.

Motivated by Proposition 2.2.1, we define the following function.

Definition 2.2.2 Let $\mathbf{S}_1, \mathbf{S}_2, \dots, \mathbf{S}_N$ be some positive semi-definite matrices. We define function v such that for any $\mathbf{x} \in \mathbb{R}^N$, $v(\mathbf{x}) = \det(\sum_{i=1}^N x_i \mathbf{S}_i)$.

As we assume for the rest of this chapter, we note that if $\mathbf{S}_1, \mathbf{S}_2, \dots, \mathbf{S}_N$ are given by (2.9), function v then extends the definition of the determinant of the FIM to the set of real vectors. We next prove that function v is supermodular over the set of non-negative real vectors.

Proposition 2.2.3 If $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^{+N}$, it holds that

$$v(\mathbf{x}_1) + v(\mathbf{x}_2) \leq v(\mathbf{x}_1 \vee \mathbf{x}_2) + v(\mathbf{x}_1 \wedge \mathbf{x}_2) \quad (2.18)$$

where $\mathbf{x}_1 \vee \mathbf{x}_2$ denotes the componentwise maximum and $\mathbf{x}_1 \wedge \mathbf{x}_2$ denotes the componentwise minimum of \mathbf{x}_1 and \mathbf{x}_2 .

PROOF We first note that function v is non-negative and non-decreasing over \mathbb{R}^{+N} . Inequality (2.18) can then be easily proved if $\sum_{i=1}^N x_i \mathbf{S}_i$ and/or $\sum_{i=1}^N x'_i \mathbf{S}_i$ are singular. We, therefore, consider the region of \mathbb{R}^{+N} where $\sum_{i=1}^N x_i \mathbf{S}_i$ is non-singular for a given \mathbf{x} . We denote this region by \mathcal{X} . To prove the supermodularity of function v over \mathcal{X} , it suffices to show that $\frac{\partial^2 v(\mathbf{x})}{\partial x_j \partial x_i} \geq 0$ for any $\mathbf{x} \in \mathcal{X}$ [57]. We note that

$$\frac{\partial v(\mathbf{x})}{\partial x_i} = \det \left(\sum_{k=1}^N x_k \mathbf{S}_k \right) \text{trace} \left(\left(\sum_{k=1}^N x_k \mathbf{S}_k \right)^{-1} \mathbf{S}_i \right) \quad (2.19)$$

and

$$\begin{aligned} \frac{\partial^2 v(\mathbf{x})}{\partial x_j \partial x_i} &= \det \left(\sum_{k=1}^N x_k \mathbf{S}_k \right) \left(\text{trace} \left(\left(\sum_{k=1}^N x_k \mathbf{S}_k \right)^{-1} \mathbf{S}_i \right) \text{trace} \left(\left(\sum_{k=1}^N x_k \mathbf{S}_k \right)^{-1} \mathbf{S}_j \right) \right. \\ &\quad \left. - \text{trace} \left(\left(\sum_{k=1}^N x_k \mathbf{S}_k \right)^{-2} \mathbf{S}_i \mathbf{S}_j \right) \right) \geq 0 \end{aligned} \quad (2.20)$$

where we have used the fact that for any two positive semi-definite matrices \mathbf{A} and \mathbf{B} , $\text{trace}(\mathbf{A})\text{trace}(\mathbf{B}) \geq \text{trace}(\mathbf{A}\mathbf{B})$ [58]. ■

Remark Although the measurement model (2.1) encompasses a large number of practical applications, a more general measurement model can also be considered for all discussions presented in this chapter if a regularity condition [2] holds for the measurement model. In particular, the general measurement model

$$\mathbf{z}_i = \mathbf{h}_i(\mathbf{p}, \mathbf{v}_i) \quad (2.21)$$

can be considered for all discussions where \mathbf{z}_i , \mathbf{p} and \mathbf{h}_i are defined as before and $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N$ are mutually-independent random error vectors but it also holds that for $i = 1, 2, \dots, N$, function \mathbf{h}_i and the probability distribution of \mathbf{v}_i are such that

$$\mathbb{E}_{p(\mathbf{z}_i|\mathbf{p})} \left\{ \frac{\partial}{\partial \mathbf{p}} \log(p(\mathbf{z}_i|\mathbf{p})) \right\} = 0 \quad (2.22)$$

The above condition, which is called the *regularity* condition, generally holds unless the support of $p(\mathbf{z}_i|\mathbf{p})$ depends on parameter \mathbf{p} . Under this condition and by using (2.11), we can write $\mathbf{J}(\mathbf{x})$ as

$$\mathbf{J}(\mathbf{x}) = \sum_{i=1}^N x_i \mathbb{E}_{p(\mathbf{z}_i|\mathbf{p})} \left\{ \frac{\partial}{\partial \mathbf{p}} \log(p(\mathbf{z}_i|\mathbf{p})) \frac{\partial}{\partial \mathbf{p}} \log(p(\mathbf{z}_i|\mathbf{p})) \right\}. \quad (2.23)$$

Therefore, by replacing the definition of \mathbf{S}_i in (2.9) with

$$\mathbf{S}_i = \mathbb{E}_{p(\mathbf{z}_i|\mathbf{p})} \left\{ \frac{\partial}{\partial \mathbf{p}} \log(p(\mathbf{z}_i|\mathbf{p})) \frac{\partial}{\partial \mathbf{p}} \log(p(\mathbf{z}_i|\mathbf{p})) \right\}, \quad (2.24)$$

we see that $\mathbf{J}(\mathbf{x})$ can be expressed as a sum of positive semi-definite matrices and all the results stated for the measurement model (2.1) can also be stated for the measurement model (2.21).

Remark Note that from (2.6) and (2.8), we can write

$$\begin{aligned} \mathbf{J}_B(\mathbf{x}) &= \mathbf{Q} + \mathbb{E}_{p(\mathbf{p})} \left\{ \sum_{i=1}^N x_i \mathbf{S}_i \right\} \\ &= \mathbf{Q} + \sum_{i=1}^N x_i \mathbb{E}_{p(\mathbf{p})} \{ \mathbf{S}_i \}. \end{aligned} \tag{2.25}$$

Hence, $\mathbf{J}_B(\mathbf{x})$ can also be written as a sum of positive semi-definite matrices where only a constant matrix \mathbf{Q} is added to the sum. Consequently, the proof given for Proposition 2.2.3 can be repeated to show that the determinant of $\mathbf{J}_B(\mathbf{x})$ is also a supermodular function. Because of the presence of matrix \mathbf{Q} , the determinant of $\mathbf{J}_B(\mathbf{x})$ cannot be immediately written as a homogenous polynomial as there will be some terms with an order lower than L . However, if the inequality constraint of the problem (2.5) is replaced with an equality constraint, i.e. $\sum_{i=1}^N \frac{c_i}{M} x_i = 1$, one can always multiply the terms of the polynomial resulted from the expansion of the determinant with $\sum_{i=1}^N \frac{c_i}{M} x_i$ as many times as needed without affecting the value of the polynomial in order to obtain a homogenous representation.

2.3 Characteristics of the Lagrangian Relaxation

In this section, we study the Lagrangian relaxation of the problem (2.3). The fact that the determinant is a supermodular function gives special characteristics to the Lagrangian relaxation of the problem (2.3). These characteristics were first noted by Gallo and Simeone in [17] for the more general case of the supermodular knapsack

problem. The following proposition summarizes those results.

Proposition 2.3.1 [17] *Consider the following combinatorial optimization problem*

$$\begin{aligned} \max_{\mathbf{x} \in \{0,1\}^N} \quad & q(\mathbf{x}) \\ \text{such that} \quad & \mathbf{c}^T \mathbf{x} \leq M \end{aligned} \tag{2.26}$$

where $\mathbf{c} \in \mathbb{R}^{++N}$, $M \in \mathbb{R}^{++}$, $\sum_{i=1}^N c_i > M > 0$ and function q is a generic function defined over $\{0, 1\}^N$ with the following properties:

- (i) $q(\mathbf{0}) = 0$,
- (ii) $\mathbf{e} = [1, 1, \dots, 1]^T$ is a maximum point of q over $\{0, 1\}^N$,
- (iii) q is a supermodular function over $\{0, 1\}^N$.

Then, the dual function of the problem (2.26) is a piecewise linear convex function with at most N breakpoints where the largest breakpoint is given by $u = \max\{\frac{q(\mathbf{e})}{c_i} \mid i = 1, 2, \dots, N\}$.

The main point of Proposition 2.3.1 is that the number of breakpoints of the dual function of the problem (2.26) does not grow exponentially with N and, in fact, it is at most N . The piecewise linearity and convexity of the dual function, otherwise, hold even if properties (i) to (iii) are not satisfied.

It is easy to see that the objective function of the problem (2.3) has all the properties listed in Proposition 2.3.1. Condition (i) follows immediately from the definition of the objective function of this problem. Condition (ii) follows from the fact that the

objective function is non-decreasing and, finally, condition (iii) follows from Proposition 2.2.3.

2.3.1 Outer Approximation Algorithm for Solving the Lagrangian Relaxation

As a result of these observations, an outer approximation algorithm [15] with at most N iterations can be applied to find the Lagrangian bound for the problem (2.3). We introduce some notation in order to describe the steps of this algorithm. The Lagrangian of the problem (2.3) is given by

$$L(\mathbf{x}, u) = v(\mathbf{x}) - u(\mathbf{c}^T \mathbf{x} - M). \quad (2.27)$$

where $u \in \mathbb{R}^+$ is the Lagrange multiplier. The dual function is an over-estimator of the maximum of function v over the feasible region. The dual function is obtained by maximizing the Lagrangian with respect to \mathbf{x} , i.e.

$$\phi(u) = \max_{\mathbf{x} \in \{0,1\}^N} L(\mathbf{x}, u). \quad (2.28)$$

The Lagrangian bound for the problem (2.3) is given by

$$\phi^* = \min_{u \in \mathbb{R}^+} \phi(u). \quad (2.29)$$

In order to find this bound, we note that, for any non-negative u , a line that passes through $(u, \phi(u))$ and has a slope equal to the subgradient of function ϕ at that point is an under-estimator of function ϕ . A subgradient of function ϕ at u is given by

$$g(u) = -(\mathbf{c}^T \mathbf{x}(u) - M) \quad (2.30)$$

where $\mathbf{x}(u)$ is the vector that maximizes $L(\mathbf{x}, u)$. Similarly, for any non-negative u' and u'' , where $u' < u''$, $g(u') < 0$ and $g(u'') > 0$, function

$$\phi_o(u; u', u'') = \max_u \{\phi(u') + g(u')(u - u'), \phi(u'') + g(u'')(u - u'')\} \quad (2.31)$$

is an under-estimator (and an outer approximation) of function ϕ . It is easy to check that the minimizer of function ϕ_o is given by

$$u^* = \frac{\phi(u'') - g(u'')u'' - (\phi(u') - g(u')u')}{g(u') - g(u'')} = \frac{v(\mathbf{x}(u'')) - v(\mathbf{x}(u'))}{\mathbf{c}^T \mathbf{x}(u'') - \mathbf{c}^T \mathbf{x}(u')}. \quad (2.32)$$

Clearly, by replacing u' with u^* if $g(u^*) \leq 0$ and by replacing u'' with u^* if $g(u^*) > 0$, another outer approximation of function ϕ can be obtained whose minimizer is closer to the minimizer of function ϕ . Since function ϕ has a maximum of N breakpoints, after N iterations, an approximation is obtained for which $\phi_o(u^*, u', u'') = \phi(u^*)$. Obviously the Lagrangian bound as the minimum of function ϕ is equal to $\phi(u^*)$.

We note that in order to compute $g(u^*)$, $\mathbf{x}(u^*)$ must be known. This, in turn,

requires solving the following problem

$$\max_{\mathbf{x} \in \{0,1\}^N} v(\mathbf{x}) - u^* \mathbf{c}^T \mathbf{x}. \quad (2.33)$$

However, since the objective function of the problem (2.33) is supermodular, a solution for this problem can be found in polynomial time. Consequently, the Lagrangian bound can be found in polynomial time.

The overall algorithm for computing the Lagrangian bound is summarized below. The computational cost of this algorithm is dominated by the computational cost of solving the problem (2.33) which subsequently depends on what type of algorithm is used for this purpose. An algorithm, which is guaranteed to solve the problem (2.33) in polynomial-time, is the ellipsoid algorithm of Gröetschel, Lovász, and Schrijver [18]. However, a more practical algorithm with an unknown run-time is the minimum-norm-point algorithm of Fujishige [59].

• Algorithm 2.1: The Lagrangian Bound for the Sensor Subset Selection

Problem

Inputs:

- A. A set of sensors: $\mathcal{N} = \{1, 2, \dots, N\}$.
- B. Matrices $\mathbf{S}_1, \mathbf{S}_2, \dots, \mathbf{S}_N$ given by (2.9) to evaluate function v .
- C. Cost vector: \mathbf{c} .

D. Total number of sensors to be selected: M .

Output:

A. Lagrangian bound for the sensor subset selection problem: ϕ^* .

Steps:

1. Let $u' = 0$ and $u'' = \max\{\frac{q(\mathbf{e})}{c_i} | i = 1, 2, \dots, N\}$.

2. Let $\mathbf{x}(u') = \mathbf{e}$ and $\mathbf{x}(u'') = \mathbf{0}$.

3. Compute

$$u^* = \frac{v(\mathbf{x}(u'')) - v(\mathbf{x}(u'))}{\mathbf{c}^T \mathbf{x}(u'') - \mathbf{c}^T \mathbf{x}(u')}$$

4. Find $\mathbf{x}(u^*) = \arg \max_{\mathbf{x} \in \{0,1\}^N} v(\mathbf{x}) - u^* \mathbf{c}^T \mathbf{x}$.

5. If

$$u^* = \frac{v(\mathbf{x}(u^*)) - v(\mathbf{x}(u'))}{\mathbf{c}^T \mathbf{x}(u^*) - \mathbf{c}^T \mathbf{x}(u')},$$

stop and return $\phi^* = v(\mathbf{x}(u^*)) - u^*(\mathbf{c}^T \mathbf{x}(u^*) - M)$.

6. If $\mathbf{c}^T \mathbf{x}(u^*) > M$, let $u' = u^*$. Otherwise let $u'' = u^*$ and go to Step 3.

2.4 Characteristics of the Continuous Relaxation

After studying the Lagrangian bound, we now consider the continuous relaxation of the problem (2.3) and study the characteristics of the continuous bound. Let v_c^* denote the continuous bound for the problem (2.3). This bound is computed via

$$\begin{aligned} \max_{\mathbf{x} \in [0,1]^N} \quad & v(\mathbf{x}) \\ \text{such that} \quad & \mathbf{c}^T \mathbf{x} \leq M. \end{aligned} \tag{2.34}$$

The following result establishes a relationship between the Lagrangian bound and the continuous bound.

Proposition 2.4.1 *For the problem (2.3), it holds that $v_c^* \leq \phi^*$, i.e. the continuous bound is tighter than the Lagrangian bound.*

PROOF We note that

$$\begin{aligned} v_c^* &= \max_{\mathbf{x} \in [0,1]^N} \{v(\mathbf{x}) \mid \mathbf{c}^T \mathbf{x} \leq M\} \\ &\leq \min_{u \in \mathbb{R}^+} \max_{\mathbf{x} \in [0,1]^N} L(\mathbf{x}, u) \\ &= \min_{u \in \mathbb{R}^+} \max_{\mathbf{x} \in \{0,1\}^N} L(\mathbf{x}, u) \\ &= \phi^*. \end{aligned} \tag{2.35}$$

The second line in (2.35) is due to weak duality. The third line follows from the fact that the maximum of function L for a given u always occurs at one of the extreme

points of $[0, 1]^N$. To prove this, we first note that $\frac{\partial^2 v(\mathbf{x})}{\partial x_i^2} \geq 0$ for $i = 1, 2, \dots, N$. This result was obtained as part of the proof of Proposition 2.2.3. Hence, $\frac{\partial v(\mathbf{x})}{\partial x_i}$ is a non-decreasing function. We next note that

$$\frac{\partial L(\mathbf{x}, u)}{\partial x_i} = \frac{\partial v(\mathbf{x})}{\partial x_i} - uc_i. \quad (2.36)$$

Therefore, depending on the value of u and a_i , three different patterns emerge for $\frac{\partial L(\mathbf{x}, u)}{\partial x_i}$ as x_i increases from 0 to 1. It can be increasing, decreasing, or decreasing first and then increasing from some point through this interval. However, it is never possible that $\frac{\partial L(\mathbf{x}, u)}{\partial x_i}$ increases first and then decreases. This, in turn, shows that every element of the maximizer of $L(\mathbf{x}, u)$ must be either 0 or 1. \blacksquare

In order to solve the problem (2.34), we note that it shares identical maximizer(s) with the following problems:

$$\begin{aligned} & \max_{\mathbf{x} \in [0, 1]^N} \log(v(\mathbf{x})) \\ & \text{such that } \mathbf{c}^T \mathbf{x} \leq M \end{aligned} \quad (2.34')$$

and

$$\begin{aligned} & \max_{\mathbf{x} \in [0, 1]^N} v(\mathbf{x})^{\frac{1}{L}} \\ & \text{such that } \mathbf{c}^T \mathbf{x} \leq M. \end{aligned} \quad (2.34'')$$

This follows immediately from the fact that functions $f(x) = x^{\frac{1}{L}}$ and $f(x) = \log(x)$ are monotone increasing over \mathbb{R}^+ . However, while $v(\mathbf{x})$ is not a concave function of \mathbf{x}

over $[0, 1]^N$, both $\log(v(\mathbf{x}))$ and $v(\mathbf{x})^{\frac{1}{L}}$ are. Therefore, the problems (2.34') and (2.34'') have a unique maximizer which can be found in polynomial time by using algorithms for constrained convex optimization [7]. Consequently, the problem (2.34) has a unique maximizer which can be found in polynomial time.

2.4.1 Natural Selection Process for Solving a Variation of the Continuous Relaxation

The continuous relaxation of the problem (2.3) can be even solved by a simpler algorithm than those commonly used for convex optimization if this problem is modified as follows.

Definition 2.4.1 *The measurement allocation problem is posed as solving:*

$$\begin{aligned} \max_{\mathbf{x} \in \mathbb{Z}^+} \quad & v(\mathbf{x}) \\ \text{such that} \quad & \mathbf{e}^T \mathbf{x} = M \end{aligned} \tag{2.37}$$

where $M \in \mathbb{Z}^{++}$, $M < N$ and $\mathbf{J}(\mathbf{x})$ is given by (2.4).

In this variation of the problem (2.3), sensors are allowed to make more than one measurement and the cost of making one measurement is identical for all sensors. By defining $\mathbf{m} = \frac{\mathbf{x}}{M}$ and taking into account Proposition 2.2.2, the continuous relaxation

of the problem (2.37) can be written as

$$\max_{\mathbf{m} \in \Delta^{N-1}} \sum_{1 \leq a_1, a_2, \dots, a_L \leq N} w_{a_1 a_2 \dots a_L} m_{a_1} m_{a_2} \dots m_{a_L} \quad (2.38)$$

where $\Delta^{N-1} = \{\mathbf{m} \in \mathbb{R}^{+N} \mid \sum_{i=1}^N m_i = 1\}$ denotes the standard simplex in \mathbb{R}^N .

The problem (2.38) involves the maximization of a homogenous polynomial over the standard simplex. A general homogenous polynomial may have many local maximizers over the standard simplex. In fact, finding the global maximizer of a homogenous polynomial over the standard simplex is known to be NP-Hard even for $L = 2$ as it can be associated with the maximum clique problem [60]. However, as we discussed earlier, the problem (2.38), in our particular case, has a unique maximizer. Hence, our task for finding its maximizer becomes much simpler.

We next show that the so-called natural selection process in population genetics [19] can be used to find the maximizer of the problem (2.38). The use of the natural selection process for finding the local maximizers of a quadratic homogenous polynomial, i.e. $L = 2$, over the standard simplex is noted in [61]. More generally, it can be shown that the set of local maximizers of a homogenous polynomial is equal to the set of the stable equilibria of the natural selection process (see Proposition 2.4.2 below). Thus, if a problem such as (2.38) has only a unique maximizer by updating an allocation vector \mathbf{m} via the natural selection process, this maximizer can be found.

To prove the desired result, it is more convenient that we first symmetrize (hyper-)matrix \mathbf{W} . Let us denote by $\alpha(a_1, a_2, \dots, a_L)$ all permutations of a_1, a_2, \dots, a_L .

Definition 2.4.2 For (hyper-)matrix \mathbf{W} , its symmetrization is given by (hyper-)matrix \mathbf{U} where

$$u_{a_1 a_2, \dots, a_L} = \frac{1}{L!} \sum_{\beta \in \alpha(a_1, a_2, \dots, a_L)} w_\beta \quad (2.39)$$

for $1 \leq a_1, a_2, \dots, a_L \leq N$.

For instance, if \mathbf{W} is a $N \times N$ matrix, then \mathbf{U} is given by

$$u_{ij} = \frac{w_{ij} + w_{ji}}{2} \quad (2.40)$$

and if \mathbf{W} is a $N \times N \times N$ hyper-matrix, then \mathbf{U} is given by

$$u_{ijk} = \frac{w_{ijk} + w_{ikj} + w_{jik} + w_{jki} + w_{kij} + w_{kji}}{6} \quad (2.41)$$

for $1 \leq i, j, k \leq N$.

It is easy to see that, in general, the value of a homogenous polynomial does not change if \mathbf{W} is replaced with its symmetrized version \mathbf{U} . Furthermore, it can be shown that when the elements of \mathbf{W} are obtained via (2.16), then the elements of \mathbf{U} are non-negative. The proof for this assertion in the general case is rather cumbersome. Instead, we just verify it when $L = 2$ and then take another approach to ensure the non-negativity of the elements \mathbf{U} . For $L = 2$ and $1 \leq i, j \leq N$ where

$j \neq i$, we note that $u_{ii} = w_{ii} = \det(\mathbf{S}_i) \geq 0$. Furthermore,

$$u_{ij} = \frac{1}{2}(w_{ij} + w_{ji}) = \det(\mathbf{S}_i + \mathbf{S}_j) - \det(\mathbf{S}_i) - \det(\mathbf{S}_j) \geq 0 \quad (2.42)$$

where the latter holds because the determinant is superadditive over the set of positive semi-definite matrices. However, even if there are elements of \mathbf{U} that are negative, we note that the maximizer of problem (2.38) does not change if a constant is added to all elements of \mathbf{U} . Hence, the non-negativity of the elements of \mathbf{U} can be ensured by adding an appropriate constant.

Remark Although never needed, some of the earlier results of this chapter can also be established by using the same technique discussed above. In particular, we note that by replacing $w_{a_1 a_2 \dots a_L}$ with $w_{a_1 a_2 \dots a_L} + \gamma b_{a_1} b_{a_2} \dots b_{a_L}$ in (2.15) for a given γ , the maximizer of the problem (2.3) does not change. Hence, it can be always ensured that the coefficients of the objective function of this problem are non-negative by adding an appropriate constant. The supermodularity of the objective function then follows from the non-negativity of the coefficients [15], without resorting to Proposition 2.2.3. Furthermore, in Proposition 2.4.1, it is no longer to show that the derivative of the objective function with respect to a decision variable is non-decreasing since this is readily known for a polynomial with non-negative coefficients.

The fact that (hyper-)matrix \mathbf{U} is symmetric and its elements are non-negative can be used to prove the following proposition.

Proposition 2.4.2 *If allocation vector \mathbf{m} is updated via the natural selection process, i.e.*

$$m'_i = \frac{m_i v_i(\mathbf{m})}{v(\mathbf{m})} \quad (2.43)$$

for $1 \leq i \leq N$ where

$$v_i(\mathbf{m}) = \sum_{1 \leq a_1, a_2, \dots, a_{L-1} \leq N} u_{i a_1 a_2 \dots a_{L-1}} m_{a_1} m_{a_2} \dots m_{a_{L-1}} \quad (2.44)$$

and

$$v(\mathbf{m}) = \sum_{1 \leq a_1, a_2, \dots, a_L \leq N} u_{a_1 a_2 \dots a_L} m_{a_1} m_{a_2} \dots m_{a_L}, \quad (2.45)$$

it then holds that $v(\mathbf{m}') \geq v(\mathbf{m})$.

PROOF Note that in (2.44), $v_i(\mathbf{m})$ is the partial weighted sum of \mathbf{U} when the first digit in the index is fixed to i . We first extend this notation to address the partial weighted sums in other dimensions. Let I be a fixed number between 1 and L . We then define

$$v_i^I(\mathbf{m}) = \sum_{1 \leq a_1, a_2, \dots, a_{L-1} \leq N} u_{a_1 a_2 \dots \underbrace{\phantom{a_1 a_2 \dots a_{L-1}}}_i \dots a_{L-1}} m_{a_1} m_{a_2} \dots m_{a_{L-1}}. \quad (2.46)$$

position I

Therefore, $v_i^I(m)$ is the partial weighted sum of \mathbf{U} when the index is fixed to i in dimension \mathbf{I} . For instance, when $L = 2$, $v_1^1(\mathbf{m})$ means the partial weighted sum of the first row, $v_1^2(\mathbf{m})$ means the partial weighted sum of the first column and so on. Note

that $v_i(\mathbf{m}) = v_i^1(\mathbf{m})$. Furthermore, since \mathbf{U} is symmetric, we have

$$v_i(\mathbf{m}) = v_i^I(\mathbf{m}) \quad (2.47)$$

for $1 \leq I \leq L$ and $1 \leq i \leq N$. We next rewrite the main inequality of the proposition in a form that has already been proved in the literature. We note that

$$\begin{aligned} v(\mathbf{m}') &= \sum_{1 \leq a_1, a_2, \dots, a_L \leq N} u_{a_1 a_2 \dots a_L} m'_{a_1} m'_{a_2} \dots m'_{a_L} \\ &= \sum_{1 \leq a_1, a_2, \dots, a_L \leq N} u_{a_1 a_2 \dots a_L} \frac{m_{a_1} v_{a_1}(\mathbf{m})}{v(\mathbf{m})} \frac{m_{a_2} v_{a_2}(\mathbf{m})}{v(\mathbf{m})} \dots \frac{m_{a_L} v_{a_L}(\mathbf{m})}{v(\mathbf{m})}. \end{aligned} \quad (2.48)$$

Therefore to prove that $v(\mathbf{m}') \geq v(\mathbf{m})$, we must show

$$\sum_{1 \leq a_1, a_2, \dots, a_L \leq N} u_{a_1 a_2 \dots a_L} m_{a_1} m_{a_2} \dots m_{a_L} v_{a_1}(\mathbf{m}) v_{a_2}(\mathbf{m}) \dots v_{a_L}(\mathbf{m}) \geq (v(\mathbf{m}))^{L+1}. \quad (2.49)$$

Using (2.47), we can write the above as

$$\sum_{1 \leq a_1, a_2, \dots, a_L \leq N} u_{a_1 a_2 \dots a_L} m_{a_1} m_{a_2} \dots m_{a_L} v_{a_1}^1(\mathbf{m}) v_{a_2}^2(\mathbf{m}) \dots v_{a_L}^L(\mathbf{m}) \geq (v(\mathbf{m}))^{L+1}. \quad (2.50)$$

The above inequality has been proved in [62] for (hyper-)matrices with non-negative elements. ■

According to Proposition 2.4.2, when $L = 2$, the allocation vector is updated as follows:

$$m'_1 = \frac{\text{sum of elements in the first row of } \mathbf{A}}{\text{sum of all elements in } \mathbf{A}} \quad (2.51)$$

$$m'_2 = \frac{\text{sum of elements in the second row of } \mathbf{A}}{\text{sum of all elements in } \mathbf{A}} \quad (2.52)$$

and so on for m'_3 to m'_N where $\mathbf{A} = [w_{ij}m_i m_j]$.

To initialize the natural selection process, a reasonable choice is the centroid of Δ^{N-1} , i.e. $[\frac{1}{N}, \dots, \frac{1}{N}]^T$. This choice ensures that the algorithm is not initially biased in favor of any particular solution. The overall algorithm for solving the continuous relaxation of the measurement allocation problem is given below.

• **Algorithm 2.2: The Continuous Bound for the Measurement Allocation**

Problem

Inputs:

- A. A set of sensors: $\mathcal{N} = \{1, 2, \dots, N\}$.
- B. (Hyper-)matrix of the sensor subset selection problem: \mathbf{W} .
- C. Total number of sensors to be selected: M .
- D. Termination criterion: ϵ .

Output:

A. An estimate for the continuous bound of the measurement allocation problem:

$$\tilde{v}_c.$$

Steps:

1. Let $\mathbf{m} = [\frac{1}{N}, \frac{1}{N}, \dots, \frac{1}{N}]^T$.

2. Compute

$$m'_i = \frac{m_i v_i(\mathbf{m})}{v(\mathbf{m})}$$

for $1 \leq i \leq N$ where v_i and v are given by (2.44) and (2.45), respectively.

3. If $\frac{\|\mathbf{m}' - \mathbf{m}\|}{\|\mathbf{m}\|} > \epsilon$, go to Step 2. Otherwise let $\tilde{v}_c = M^N \mathbf{v}(\mathbf{m}')$ and return.

Algorithm 2.2 typically converges in a few tens of iterations for ϵ as low as 10^{-6} (see below for a numerical example). Each iteration of this algorithm is dominated by N^L operations because the computation of v_i in (2.44) requires N^{L-1} operations and v_i must be computed for N sensors. As a result, for the parameters with two or three elements, i.e. $L = 2$ or 3 , it only takes a few milliseconds on a modest personal computer to find the optimal allocation vector even when the network has a few hundred sensors.

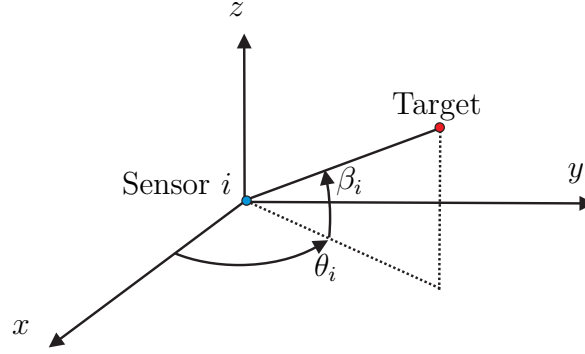


Figure 2.1: Geometry of bearings-only localization in three-dimensional space.

Numerical Analysis of the Convergence Rate of the Natural Selection Process

To study the behavior of Algorithm 2.2 numerically, we study the problem (2.37) in the context of bearings-only localization. Figure 2.1 shows the geometry of bearings-only localization in three-dimensional space [42]. The figure indicates that the true DOA to sensor i can be specified by two angles. If the position of the target is denoted by $\mathbf{p} = [x_t, y_t, z_t]$ and the position of sensor i for $i = 1, 2, \dots, N$ is denoted by $\mathbf{p}_i = [x_i, y_i, z_i]$, these angles are related to the position of the target and the position of the sensor via

$$\theta_i(\mathbf{p}) = \text{atan2}(y_t - y_i, x_t - x_i) \quad (2.53)$$

$$\beta_i(\mathbf{p}) = \text{atan2}(z_t - z_i, \sqrt{(y_t - y_i)^2 + (x_t - x_i)^2}) \quad (2.54)$$

where $\text{atan2}(y, x)$ is the four-quadrant arc tangent of y/x in the interval $[-\pi, \pi]$. As a result, we can write function \mathbf{h}_i in the measurement model (2.1) as

$$\mathbf{h}_i(\mathbf{p}) = \begin{bmatrix} \theta_i(\mathbf{p}) \\ \beta_i(\mathbf{p}) \end{bmatrix}. \quad (2.55)$$

and note that

$$\frac{\partial \mathbf{h}_i}{\partial \mathbf{p}} = \begin{bmatrix} -\frac{\sin(\theta_i)}{r_{xyi}} & \frac{\cos(\theta_i)}{r_{xyi}} & 0 \\ -\frac{\sin(\beta_i)\cos(\theta_i)}{r_i} & -\frac{\sin(\beta_i)\sin(\theta_i)}{r_i} & \frac{\cos(\beta_i)}{r_i} \end{bmatrix} \quad (2.56)$$

where $r_{xyi} = \sqrt{(y_t - y_i)^2 + (x_t - x_i)^2}$ and $r_i = \sqrt{(y_t - y_i)^2 + (x_t - x_i)^2 + (z_t - z_i)^2}$.

Consequently, \mathbf{S}_i can be computed using (2.9) and \mathbf{W} can be computed using (2.16).

For bearings-only localization in two-dimensional space, we note that just one angle is sufficient to specify the DOA. Hence, $h_i(\mathbf{p}) = \theta_i(\mathbf{p})$ and

$$\frac{\partial \theta_i}{\partial \mathbf{p}} = \begin{bmatrix} -\frac{\sin(\theta_i)}{r_{xyi}} & \frac{\cos(\theta_i)}{r_{xyi}} \end{bmatrix}. \quad (2.57)$$

We generated 500 networks in both two-dimensional space and three-dimensional space with 100 sensors in each network. The target was located at the origin of a Cartesian coordinate system and the sensors were spread around the target in a 200×200 square or a $200 \times 200 \times 200$ cube. The positions of the sensors were drawn from a uniform distribution. It was assumed that the standard deviation of the measurement error for each angle of the DOA is 10 degrees and that the measurement error for one angle is independent of the measurement error for the other angle. A histogram of the number of iterations needed for the convergence of Algorithm 2.2

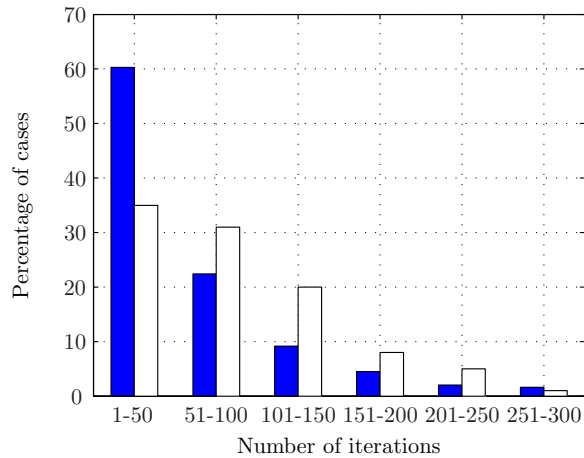


Figure 2.2: A histogram for the number of iterations needed for the convergence of Algorithm 2.2. The dark-colored bars are for localization in two-dimensional space and the white bars are for localization in three-dimensional space.

is shown in Figure 2.2 when ϵ was set to 10^{-6} . On average, it took 61.2 iterations for the algorithm to converge when localization was carried out in two-dimensional space while it took 85.1 iterations for the algorithm to converge when localization was carried out in three-dimensional space

2.4.2 Analysis of the Performance of a Greedy Algorithm for Solving the Sensor Subset Selection Problem

One of the benefits of computing upper bounds for combinatorial optimization problems is to analyze the performance of heuristic algorithms that are used for solving them. We compared the performance of a greedy algorithm [15], which has been proposed in [29, 30] for solving the problem (2.3), with the continuous bound of this problem, which can be obtained by solving the problem (2.34). In [29, 30], the performance of this algorithm is only compared with the performance of other heuristic

algorithms. We did our analysis in the context of bearings-only localization as formulated in Section 2.4.1

This greedy algorithm is generally used for solving knapsack problems. It consists of a fill-up phase and an exchange phase and can be summarized as follows.

• Algorithm 2.3: A Greedy Algorithm for Solving the Sensor Selection

Problem

Inputs:

- A. A set of sensors: $\mathcal{N} = \{1, 2, \dots, N\}$.
- B. (Hyper-)matrix of the sensor subset selection problem: \mathbf{W} .
- C. Total number of sensors to be selected: M (Assuming $M \geq 2$).

Output:

- A. A lower bound for the sensor selection problem: \underline{v} .

Phase 1 - Fill-Up

Steps:

1. Find $(k, l) \in \arg \max\{v(\mathbf{e}_i + \mathbf{e}_j) | i, j \in \mathcal{N}\}$ where

$$v(\mathbf{x}) = \sum_{1 \leq a_1, a_2, \dots, a_L \leq N} w_{a_1 a_2 \dots a_L} x_{a_1} x_{a_2} \dots x_{a_L}.$$

2. Let $\mathbf{x} = \mathbf{e}_k + \mathbf{e}_l$.
3. Let $\mathcal{K}_0 = \mathcal{N} \setminus \{k, l\}$ and $\mathcal{K}_1 = \{k, l\}$.
4. Repeat the following steps until $|\mathcal{K}_1| = M$
 - 4.1. Find $k \in \arg \max\{v(\mathbf{x} + \mathbf{e}_i) | i \in \mathcal{K}_0\}$.
 - 4.2. Let $\mathbf{x} := \mathbf{x} + \mathbf{e}_k$, $\mathcal{K}_0 := \mathcal{K}_0 \setminus \{k\}$ and $\mathcal{K}_1 := \mathcal{K}_1 \cup \{k\}$.

Phase 2 - Exchange

Steps:

1. Repeat the following steps until $\mathcal{K}_0 = \emptyset$ or $\mathcal{K}_1 = \emptyset$ in which case return $\underline{v} = v(\mathbf{x})$.
 - 1.1. Find $(k, l) \in \arg \max\{v(\mathbf{x} + \mathbf{e}_i - \mathbf{e}_j) - v(\mathbf{x}) | i \in \mathcal{K}_0, j \in \mathcal{K}_1\}$.
 - 1.2. if $v(\mathbf{x} + \mathbf{e}_k - \mathbf{e}_l) - v(\mathbf{x}) < 0$, stop and return $\underline{v} = v(\mathbf{x})$. Otherwise let $\mathbf{x} := \mathbf{x} + \mathbf{e}_k - \mathbf{e}_l$, $\mathcal{K}_0 := \mathcal{K}_0 \setminus \{k\}$ and $\mathcal{K}_1 := \mathcal{K}_1 \setminus \{l\}$.

Our empirical results indicate that Algorithm 2.3 is extremely efficient for solving the problem (2.3) in the context of bearings-only localization. The values that Algorithm 2.3 generated for the objective function of the problem (2.3) were within 1% of the continuous bound for all 500 sample networks in two-dimensional space and they were within 1% of the continuous bound for 99% sample networks in three-dimensional space.

2.5 Concluding Remarks

In this chapter, we analyzed the Lagrangian and continuous relaxations of the sensor subset selection problem with the determinant of the (B-)FIM as its objective function. Table 2.1 summarizes the characteristics of the sensor subset selection problem with various determinant-based objective functions. Based on these results, to obtain an upper bound for the sensor subset selection problem in polynomial time, one can solve the continuous relaxation of this problem with the log-determinant or the L-th root of the determinant as the objective function. On the other hand, when the constraint of making only a single measurement by each sensor is removed, an upper bound can be obtained in polynomial time by using the natural selection process with the determinant as the objective function.

Table 2.1: Characteristics of the Sensor subset selection problem with three determinant-based objective functions

Objective Function	Concave	Supermodular	Tighter Bound	Bound Computable in Polynomial Time
Logarithm and L-th Root of Determinant	Yes	No	Lagrangian	Continuous
Determinant	No	Yes	Continuous	Lagrangian and Continuous

Chapter 3

Power Index of a Sensor

3.1 Overview

In this chapter, we still work under the assumptions of the previous chapter and describe a novel approach for identifying informative sensors in a network of fixed sensors from the perspective of a target that is being observed by the network. Our premise in this chapter is that the target has some capabilities to adversely affect the measurements of a limited number of sensors by attacking and destroying them or by manipulating the measurements they make. The target, however, has no perfect knowledge as to which sensors are to be selected for making measurements. In the absence of such perfect knowledge, we show that the target can assign a value to each sensor which determines, relatively, how informative that sensor is. We call this value the power index of a sensor. The power index is computed based on the measurement model for sensors as well as the probabilities that the target assigns to each subset of sensors reflecting its perception from how likely it is that the subset will be selected. If these probabilities only depend on the size of subsets, then the notion of power indices coincides with the notation of symmetric probabilistic values in cooperative

game theory. Symmetric probabilistic values are known to be indicators of the relative power of players in cooperative games.

To provide more perspective, let us briefly discuss a simple example that we later expand in Section 3.4. Consider Figure 3.1 where three sensors can use bearing measurements to estimate the position of a target. Let us assume that the target knows that *two* sensors are to be selected for making measurements but it can attack only *one* sensor. Since the target is only concerned how accurately its position will be estimated by the sensor network, it can choose a metric of estimation accuracy and solve the sensor subset selection problem to discover that sensors 1 and 2 are the optimal choices for making measurements. Following this line of thought, we can then argue that it is inconsequential from the perspective of the target which of the two sensors is attacked and destroyed because the other sensor cannot resolve the uncertainty about the position of the target on its own. On the other hand, it is evident that if there is even a slim chance that sensor 3 is selected instead of sensor 1, it is of great consequence for the target to destroy sensor 2, because otherwise sensors 2 and 3 can resolve the uncertainty of its position. Power indices, as we will see, capture such possibilities and declare sensor 2 as the most informative sensor.

The rest of this chapter is organized as follows. In Section 3.2, we define the parameter estimation game and study its properties. In Section 3.3, we introduce the notion of power indices and discuss how they can be computed efficiently for probability distributions that only depend on the size of subsets. We also compute

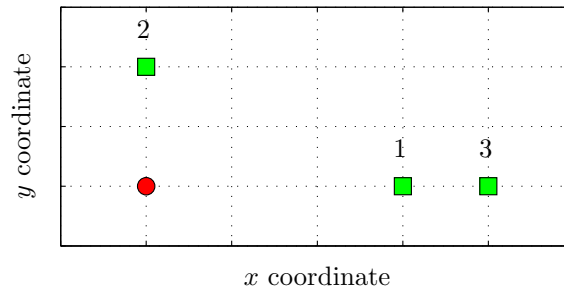


Figure 3.1: A simple network configuration to study the relative power of sensors for bearings-only localization. The circle identifies the position of the target and the squares identify the position of sensors, labeled from 1 to 3. Sensor 2 has the largest power index.

the power indices of sensors for bearings-only and range-only target localization and tracking in Section 3.4. Concluding remarks are made in Section 3.5.

3.2 Cooperative Parameter Estimation Game

In this section, we introduce the cooperative parameter estimation game. Generally, a cooperative game is identified by a pair (\mathcal{N}, v) , where $\mathcal{N} = \{1, 2, \dots, N\}$ is the set of players and $v : 2^{\mathcal{N}} \rightarrow \mathbb{R}$ is a function which assigns a real number to each subset (*coalition*) of players. As we will see, in the context of the parameter estimation game, function v is a metric of estimation accuracy. More specifically, we choose the determinant of the (B-)FIM as the metric of estimation accuracy because power indices can be then computed in polynomial time with this metric. As a result of this choice, the discussion in this section resembles the discussion in Section 2.2. We, therefore, shorten our discussion here and refer the reader to Section 2.2 for more explanations.

Let $\mathcal{N} = \{1, 2, \dots, N\}$ denote the set of all sensors and $\mathbf{p} \in \mathbb{R}^L$ denote the parameter vector of the target, which is unknown to the sensors and must be estimated. The measurement of sensor i for $i = 1, 2, \dots, N$ is related to \mathbf{p} via

$$\mathbf{z}_i = \mathbf{h}_i(\mathbf{p}) + \mathbf{v}_i \quad (3.1)$$

where $\mathbf{z}_i \in \mathbb{R}^D$ is the measurement vector, \mathbf{h}_i is a vector-valued differentiable function of \mathbf{p} , and $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N$ are mutually-independent zero-mean Gaussian random error vectors. The covariance matrix of \mathbf{v}_i is denoted by \mathbf{R}_i . Similar to Section 2.2, all results presented here also hold for the more general measurement model (2.21). The measurements that are collected from the sensors in set $\mathcal{S} \subset \mathcal{N}$ are denoted by $\mathcal{Z}_{\mathcal{S}}$ here.

Definition 3.2.1 *The cooperative parameter estimation game (\mathcal{N}, v) is a game in which \mathcal{N} is the set of sensors and $v(\mathcal{S})$ for set $\mathcal{S} \subset \mathcal{N}$ is one of the following functions:*

1. If \mathbf{p} is regarded as a non-random parameter [1],

$$v(\mathcal{S}) = \det(\mathbf{J}(\mathcal{S})) \quad (3.2)$$

where

$$\mathbf{J}(\mathcal{S}) = \mathbb{E}_{p(\mathcal{Z}_{\mathcal{S}}|\mathbf{p})} \left\{ \frac{\partial}{\partial \mathbf{p}}^T \log(p(\mathcal{Z}_{\mathcal{S}}|\mathbf{p})) \frac{\partial}{\partial \mathbf{p}} \log(p(\mathcal{Z}_{\mathcal{S}}|\mathbf{p})) \right\} \quad (3.3)$$

for $\mathcal{S} \neq \emptyset$ and $\mathbf{J}(\emptyset) = 0$.

2. If \mathbf{p} is regarded as a random parameter, i.e. a prior density $p(\mathbf{p})$ is given for \mathbf{p} ,

$$v(\mathcal{S}) = \det(\mathbf{J}_B(\mathcal{S})) \quad (3.4)$$

where

$$\mathbf{J}_B(\mathcal{S}) = \mathbf{Q} + \mathbb{E}_{p(\mathbf{p})}\{\mathbf{J}(\mathcal{S})\} \quad (3.5)$$

and

$$\mathbf{Q} = \mathbb{E}_{p(\mathbf{p})}\left\{\frac{\partial}{\partial \mathbf{p}}^T \log(p(\mathbf{p})) \frac{\partial}{\partial \mathbf{p}} \log(p(\mathbf{p}))\right\}. \quad (3.6)$$

Again, for the rest of this chapter, we assume that function v is given by (3.2). This is only for convenience and brevity in notation. All statements either directly hold or can be expanded such that they hold when function v is given by (3.4). Since a power index is computed from the target side and the exact value of \mathbf{p} is known, it is more reasonable to use (3.2). Using (3.4) for computing a power index makes sense when the target is somehow aware of the prior density that the sensors are using.

We next note that based on Proposition 2.2.2, which has been restated below, the parameter estimation game can be compactly represented by a (hyper-)matrix. By compact representation, we mean that instead of 2^N numbers, which are generally needed to specify function v , only N^L numbers are sufficient. Since N , the number of sensors, is often much larger than L , the size of the unknown parameter vector, it

generally holds that $2^N \gg N^L$.

Proposition 3.2.1 For set $\mathcal{S} \subset \mathcal{N}$, $v(\mathcal{S})$ as defined by (3.2) can be written as

$$v(\mathcal{S}) = \sum_{a_1 \in \mathcal{S}} \sum_{a_2 \in \mathcal{S}} \cdots \sum_{a_L \in \mathcal{S}} w_{a_1 a_2 \cdots a_L} \quad (3.7)$$

where

$$w_{a_1 a_2 \cdots a_L} = \det \left(\begin{bmatrix} (\mathbf{S}_{a_1})_{.1} & (\mathbf{S}_{a_2})_{.2} & \cdots & (\mathbf{S}_{a_L})_{.L} \end{bmatrix} \right) \quad (3.8)$$

in which $(\mathbf{S}_i)_{.j}$ for $i = a_i \in \mathcal{S}$ and $1 \leq j \leq L$ is the j -th column of matrix \mathbf{S}_i , where

\mathbf{S}_i is given by

$$\mathbf{S}_i = \frac{\partial^T \mathbf{h}_i}{\partial \mathbf{p}} \mathbf{R}_i^{-1} \frac{\partial \mathbf{h}_i}{\partial \mathbf{p}}. \quad (3.9)$$

Remark $\mathbf{W} = [w_{a_1 a_2 \cdots a_L}]$ is an L -dimensional (hyper-)matrix with N elements in each dimension.

3.3 Computation of Power Indices

Having established what the parameter estimation game is, we next define the power index of a sensor. A power index is determined by a probability distribution that captures the target's perception of how likely it is that a certain subset will be selected.

In particular, we compute power indices for three different probability distributions.

In the first probability distribution, which leads to the Banzhaf value, all subsets are

equally likely to be selected. In the second probability distribution, which leads to the Shapley value, all subset sizes are equally likely to be selected. Finally, we consider a probability distribution where only the probability of selecting subsets with a given size is non-zero. From the formula for this latter probability distribution, the power index for any arbitrary probability distribution that only depends on the size of subsets but not the elements of subsets can be computed.

Definition 3.3.1 *The power index of sensor i in the parameter estimation game (\mathcal{N}, v) is given by*

$$\rho_i = \sum_{\mathcal{S} \subset \mathcal{N} \setminus \{i\}} p_i(\mathcal{S})(v(\mathcal{S} \cup \{i\}) - v(\mathcal{S})), \quad (3.10)$$

where $p_i(\mathcal{S})$ is the probability that the sensors in set $\mathcal{S} \subset \mathcal{N} \setminus \{i\}$ are selected when it is known that sensor i is also selected.

In practice, it is more natural to assume that the target's probabilistic perception of subset selection is captured by a general probability distribution p_a over all subsets of \mathcal{N} from which the conditional probability distribution p_i is derived. Although in general any probability distribution can be considered as p_a , in what follows, we assume that p_a is only determined by the size of the subsets but not their elements. This assumption means that the target has some probabilistic perception about the size of the subset that is to be selected but it does not have any probabilistic perception about specific sensors to be selected. This assumption also aligns our definition of power indices with the definition of symmetric probabilistic values in cooperative

game theory and allows us to derive expressions for some well-known symmetric probabilistic values.

For instance, if it is assumed that all subsets are equally likely to be selected, the power index then becomes the Banzhaf value. In this case, $p_a(\mathcal{S})$ for all $\mathcal{S} \subset \mathcal{N}$ is given by $\frac{1}{2^N}$ and $p_a(\mathcal{S})$ for all $\mathcal{S} \subset \mathcal{N} \setminus \{i\}$ is given by $\frac{1}{2^{N-1}}$. We formalize this case by the following definition.

Definition 3.3.2 *A power index is called the Banzhaf value or the Banzhaf index when $p_i(\mathcal{S}) = \frac{1}{2^{N-1}}$ for $\mathcal{S} \subset \mathcal{N} \setminus \{i\}$.*

Proposition 3.3.1 provides a formula for computing the Banzhaf value, given the following definition.

Definition 3.3.3 *In (hyper-)matrix \mathbf{W} with L dimensions and N elements in each dimension, let $\alpha = a_1 a_2 \cdots a_L$ be the index to element w_α . For any $1 \leq i \leq N$ and $1 \leq q \leq L$, we define*

$$\kappa_q(i) = \left\{ a_1 a_2 \cdots a_L \mid \sum_{j=1}^L \mathbb{I}_j(a_1 a_2 \cdots a_L) = q, \mathbb{I}_i(a_1 a_2 \cdots a_L) = 1 \right\} \quad (3.11)$$

where

$$\mathbb{I}_i(a_1 a_2 \cdots a_L) = \begin{cases} 1 & \text{if there exists } j \text{ such that } a_j = i, \\ 0 & \text{otherwise.} \end{cases} \quad (3.12)$$

If one thinks of (hyper-)graphs, $\kappa_q(i)$ identifies the q -size edges that are shared by vertex i and $q - 1$ other vertices. Furthermore, note that since $a_1 a_2 \cdots a_L$ cannot

have more than L distinct digits, the maximum value of q is L .

Proposition 3.3.1 *The Banzhaf value of sensor i in the parameter estimation game is given by*

$$\rho_i^B = \sum_{q=1}^L \sum_{\alpha \in \kappa_q(i)} \frac{1}{2^{q-1}} w_\alpha. \quad (3.13)$$

PROOF We consider the contribution of w_α to ρ_i^B where $\alpha \in \kappa_q(i)$. For every set \mathcal{S} that does not contain i but contains the other $q-1$ sensors, with the addition of i , w_α contributes $\frac{1}{2^{N-1}} w_\alpha$ to ρ_i . There are $\binom{N-q}{S-q+1}$ sets of size S that do not contain i but contain the other $q-1$ sensors and S can be any number between $q-1$ and $N-1$. Hence, the overall contribution of w_α is

$$\frac{w_\alpha}{2^{N-1}} \sum_{s=q-1}^{N-1} \binom{N-q}{s-q+1} = \frac{w_\alpha}{2^{N-1}} \sum_{r=0}^{N-q} \binom{N-q}{r} = \frac{w_\alpha}{2^{q-1}}. \quad (3.14) \quad \blacksquare$$

As an illustration, consider a case where the unknown parameter belongs to \mathbb{R}^3 and hence \mathbf{W} becomes an $N \times N \times N$ hyper-matrix. Then, according to the above proposition, the Banzhaf value of sensor 1 is given by

$$\rho_1^B = w_{111} + \frac{1}{2} \sum_{i=2}^N (w_{1ii} + w_{i1i} + w_{ii1}) + \frac{1}{4} \sum_{i=2}^N \sum_{\substack{j=2 \\ j \neq i}}^N (w_{1ij} + w_{i1j} + w_{ij1}). \quad (3.15)$$

The first term in (3.15) corresponds to $q = 1$ in (3.13), the second term corresponds to $q = 2$, and finally the last term corresponds to $q = 3$.

Another well-known case is when all subset sizes are equally likely to be selected.

In this case, $p_a(\mathcal{S})$ for $\mathcal{S} \subset \mathcal{N}$ is given by

$$p_a(\mathcal{S}) = \frac{1}{N+1} \frac{1}{\binom{N}{S}} = \frac{S!(N-S)!}{(N+1)!} \quad (3.16)$$

where $S = |\mathcal{S}|$. Again, p_i can be determined from p_a and the following definition can be established.

Definition 3.3.4 *A power index is called the Shapley value when $p_i(\mathcal{S}) = \frac{S!(N-S-1)!}{N!}$ for $\mathcal{S} \subset \mathcal{N} \setminus \{i\}$.*

Similar to the Banzhaf value, the following proposition provides a formula for computing the Shapley value.

Proposition 3.3.2 *The Shapley value of sensor i in the parameter estimation game is given by*

$$\rho_i^{\text{Sh}} = \sum_{q=1}^L \sum_{\alpha \in \kappa_q(i)} \frac{1}{q} w_\alpha. \quad (3.17)$$

PROOF Again, we consider the contribution of w_α to ρ_i^{Sh} where $\alpha \in \kappa_q(i)$. For every set \mathcal{S} that does not contain i but contains the other $q-1$ sensors, with the addition of i , w_α contributes $\frac{S!(N-S-1)!}{N!} w_\alpha$ to ρ_i . There are $\binom{N-q}{S-q+1}$ sets of size S that do not contain i but contain the other $q-1$ sensors, and S can be any number

between $q - 1$ and $N - 1$. Hence, the overall contribution of w_α is

$$w_\alpha \sum_{s=q-1}^{N-1} \binom{N-q}{s-q+1} \frac{s!(N-s-1)!}{N!}. \quad (3.18)$$

It can be then shown by induction that the above sum is equal to $\frac{1}{q}$, and the result follows. ■

Continuing with the example given earlier, we can compute as follows the Shapley value of sensor 1 when the unknown parameter belongs to \mathbb{R}^3

$$\rho_1^{\text{Sh}} = w_{111} + \frac{1}{2} \sum_{i=2}^N (w_{1ii} + w_{i1i} + w_{ii1}) + \frac{1}{3} \sum_{i=2}^N \sum_{\substack{j=2 \\ j \neq i}}^N (w_{1ij} + w_{i1j} + w_{ij1}). \quad (3.19)$$

As can be seen, the elements of \mathbf{W} that are shared by a number of sensors are equally divided between them in the computation of the Shapley value.

Finally, we consider the probability distribution in which only subsets with size S are selected. It can be seen that in this case p_α is given by

$$p_\alpha(\mathcal{S}) = \begin{cases} \frac{1}{\binom{N}{S}} & \text{if } |\mathcal{S}| = S, \\ 0 & \text{otherwise} \end{cases} \quad (3.20)$$

and consequently

$$p_i(\mathcal{S}) = \begin{cases} \frac{1}{\binom{N-1}{S-1}} & \text{if } |\mathcal{S}| = S-1 \text{ and } i \notin \mathcal{S}, \\ 0 & \text{otherwise.} \end{cases} \quad (3.21)$$

By following the arguments given in Propositions 3.3.1 and 3.3.2, we state the following proposition without a proof.

Proposition 3.3.3 *When only the subsets with size S are selected, the power index of sensor i in the parameter estimation game is given by*

$$\rho_i^S = \sum_{q=1}^{\min(S,L)} \sum_{\alpha \in \kappa_q(i)} \frac{\binom{N-q}{S-q}}{\binom{N-1}{S-1}} w_\alpha. \quad (3.22)$$

Note that the power index of sensor i for any probability distribution that only depends on the size of subsets can be computed as the weighted average of $\rho_i^1, \rho_i^2, \dots, \rho_i^N$ where the weights represent the probability that a certain size is selected. For instance, when $\rho_i^1, \rho_i^2, \dots, \rho_i^N$ are equally weighed by $\frac{1}{N}$, their average is equal to the Shapley value.

Remark In [21], a cooperative game is defined over a weighted (hyper-)graph where

the characteristic function of the game for any subset of vertices is the sum of the weights of the edges connecting those vertices. This definition can be extended to (hyper-)matrices as follows.

Definition 3.3.5 *Let \mathbf{W} be an L -dimensional (hyper-)matrix with N elements in each dimension. Let $\mathcal{N} = \{1, 2, \dots, N\}$. A cooperative (hyper-)matrix game, denoted by $(\mathcal{N}, \mathbf{W})$, is a cooperative game defined over \mathcal{N} where the value of set $\mathcal{S} \subset \mathcal{N}$ is given by*

$$v(\mathcal{S}) = \sum_{a_1 \in \mathcal{S}} \sum_{a_2 \in \mathcal{S}} \cdots \sum_{a_L \in \mathcal{S}} w_{a_1 a_2 \cdots a_L}. \quad (3.23)$$

Note that the parameter estimation game is a (hyper-)matrix game. Therefore, (3.13), (3.17) and (3.22) show how various power indices can be computed in (hyper-)matrix games.

3.4 Power Indices for Localization and Tracking

Arguably, the most important parameters that a target must protect are its position and velocity. It is therefore critical to know which sensors can be more informative for this purpose. Typically, target localization and tracking is carried out by using bearing or range measurements. In this section, we compute the power indices of sensors for bearings-only and range-only localization and tracking. Interestingly, we discover that power indices have a very similar structure for both measurement models and that what causes the power index of a sensor to be large for bearings-only localiza-

tion also causes the power index of a sensor to be large for range-only localization (see (3.31) and (3.37)).

To keep notation simple, for localization, we assume that the position of the target is estimated in two-dimensional space, i.e. $\mathbf{p} = [x_t, y_t]$. For tracking, we assume that the target's position and velocity are estimated in one-dimensional space, i.e. $\mathbf{p} = [x_t, v_{xt}]$. When $\mathbf{p} \in \mathbb{R}^2$, it can be seen from (3.13) and (3.17) that the Banzhaf value and the Shapley value are equal and given by

$$\rho_i^{\text{B}} = \rho_i^{\text{Sh}} = w_{ii} + \frac{1}{2} \sum_{\substack{j=1 \\ j \neq i}}^N (w_{ij} + w_{ji}). \quad (3.24)$$

Similarly, from (3.22), it can be seen that the power index of sensor i when the subset size is S for $1 \leq S \leq N$ is given by

$$\rho_i^{\text{S}} = w_{ii} + \frac{S-1}{N} \sum_{\substack{j=1 \\ j \neq i}}^N (w_{ij} + w_{ji}). \quad (3.25)$$

Therefore, in two-dimensional space, if $w_{ii} = 0$ for $i = 1, 2, \dots, N$, all power indices induce the same ranking among the sensors.

3.4.1 Localization

We denote the position of the target by $\mathbf{p} = [x_t, y_t]$ and the position of the i -th sensor for $i = 1, 2, \dots, N$ by $\mathbf{p}_i = [x_i, y_i]$. First, we consider bearings-only localization. The

measurement of sensor i is given by

$$z_i = \theta_i(\mathbf{p}) + v_i \quad (3.26)$$

where

$$\theta_i(\mathbf{p}) = \text{atan2}(y_t - y_i, x_t - x_i) \quad (3.27)$$

and v_1, v_2, \dots, v_N are mutually-independent zero-mean Gaussian random errors with variance σ_{bi}^2 . We note that

$$\frac{\partial \theta_i}{\partial \mathbf{p}} = \frac{1}{r_i} \begin{bmatrix} -\sin(\theta_i) & \cos(\theta_i) \end{bmatrix} \quad (3.28)$$

where $r_i = \sqrt{(x_t - x_i)^2 + (y_t - y_i)^2}$ is the relative range of sensor i from the target.

Consequently, from (3.9), \mathbf{S}_i can be written as

$$\mathbf{S}_i = \frac{1}{r_i^2 \sigma_{bi}^2} \begin{bmatrix} \sin^2(\theta_i) & -\sin(\theta_i)\cos(\theta_i) \\ -\sin(\theta_i)\cos(\theta_i) & \cos^2(\theta_i) \end{bmatrix} \quad (3.29)$$

and, from (3.8), w_{ij} for $1 \leq i, j \leq N$ can be written as

$$\begin{aligned} w_{ij} &= \det \left(\begin{bmatrix} \frac{\sin^2(\theta_i)}{r_i^2 \sigma_{bi}^2} & -\frac{\sin(\theta_j)\cos(\theta_j)}{r_j^2 \sigma_{bj}^2} \\ -\frac{\sin(\theta_i)\cos(\theta_i)}{r_i^2 \sigma_{bi}^2} & \frac{\cos^2(\theta_j)}{r_j^2 \sigma_{bj}^2} \end{bmatrix} \right) \\ &= \frac{\sin^2(\theta_i)\cos^2(\theta_j) - \sin(\theta_i)\cos(\theta_i)\sin(\theta_j)\cos(\theta_j)}{r_i^2 \sigma_{bi}^2 r_j^2 \sigma_{bj}^2}. \end{aligned} \quad (3.30)$$

Finally, by plugging (3.30) into (3.24), the power index of a sensor for bearings-only localization can be expressed as

$$\rho_i^{\text{Bearing}} = \sum_{j=1}^N \frac{\sin^2(\theta_i - \theta_j)}{2r_i^2 \sigma_{bi}^2 r_j^2 \sigma_{bj}^2}. \quad (3.31)$$

As a numerical example, consider a network of 5 sensors shown in Figure 3.2 with a target located at $[0, 0]$. We assume that the standard deviation of the measurement error is 10 degrees for all sensors, i.e. $\sigma_{bi} = 10^\circ$. Using (3.31), we compute the power index of the sensors. These numbers are given in Table 3.1, which suggest that sensor 1 is the most informative one followed by sensors 3, 2, 4 and 5.

To verify these results, we note that sensors 4 and 5 are farther away from the target and their measurements are almost identical. Hence, these two sensors must be the least informative ones. Among the other three sensors, the distance between sensor 2 and the target is larger while the information that can be gained from the measurement of this sensor can also be gained from the measurement of sensor 1. Therefore, this sensor must have the lowest power index among the three. Finally, between sensors 1 and 3, sensor 1 is more uniquely positioned. Hence it must have a higher power index.

The above example may imply that the most informative sensor is always the one that is closest to the target. However, this is not necessarily true. As an example, consider Figure 3.3 where we have only changed the position of sensor 1 and moved it closer to sensor 3. In this configuration, sensor 2 provides more diversity in measure-

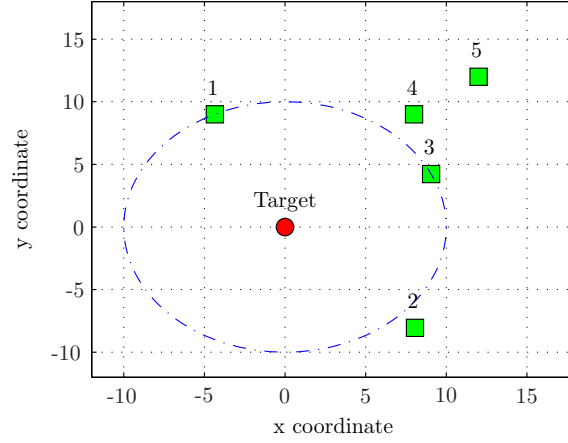


Figure 3.2: A more general network configuration to study the power index of sensors for bearings-only and range-only localization. The circle identifies the position of the target and the squares identify the position of sensors which are labeled from 1 to 5.

ments than sensor 1. Hence, as Table 3.2 suggests, its power index becomes larger than the power index of sensor 1 despite the fact that it is farther away from the target.

For range-only localization, the measurement of sensor i is given by

$$z_i = h_i(\mathbf{p}) + v_i \quad (3.32)$$

where

$$h_i(\mathbf{p}) = \frac{a^2}{(x_t - x_i)^2 + (y_t - y_i)^2} \quad (3.33)$$

and a is a signal energy factor. Measurement errors v_1, v_2, \dots, v_N are mutually-independent zero-mean Gaussian random variables with variance $\sigma_{r_i}^2$. Following the

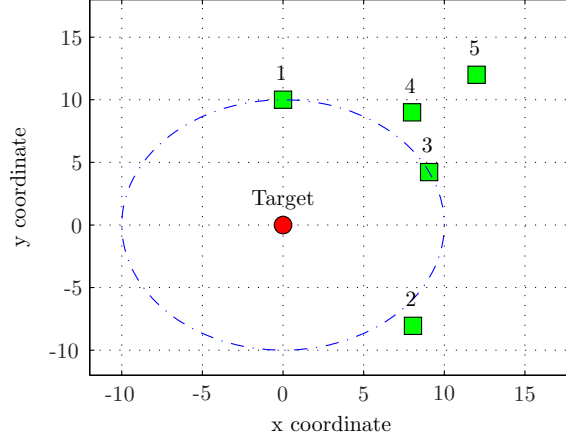


Figure 3.3: This network configuration demonstrates that the closest sensor is not necessarily the most informative sensor. The circle identifies the position of the target and the squares identify the position of sensors which are labeled from 1 to 5.

same procedure that we did for bearings-only localization, we determine that

$$\frac{\partial h_i}{\partial \mathbf{p}} = \frac{-2a^2}{r_i^3} \begin{bmatrix} \cos(\theta_i) & \sin(\theta_i) \end{bmatrix} \quad (3.34)$$

and

$$\mathbf{S}_i = \frac{4a^4}{r_i^6 \sigma_{r_i}^2} \begin{bmatrix} \cos^2(\theta_i) & \sin(\theta_i)\cos(\theta_i) \\ \sin(\theta_i)\cos(\theta_i) & \sin^2(\theta_i) \end{bmatrix}. \quad (3.35)$$

Consequently, w_{ij} for $1 \leq i, j \leq N$ is given by

$$\begin{aligned} w_{ij} &= \det \left(\begin{bmatrix} \frac{4a^4 \cos^2(\theta_i)}{r_i^6 \sigma_{r_i}^2} & \frac{4a^4 \sin(\theta_j)\cos(\theta_j)}{r_j^6 \sigma_{r_j}^2} \\ \frac{4a^4 \sin(\theta_i)\cos(\theta_i)}{r_i^6 \sigma_{r_i}^2} & \frac{4a^4 \cos^2(\theta_j)}{r_j^6 \sigma_{r_j}^2} \end{bmatrix} \right) \\ &= 16a^8 \frac{\sin^2(\theta_i)\cos^2(\theta_j) - \sin(\theta_i)\cos(\theta_i)\sin(\theta_j)\cos(\theta_j)}{r_i^6 \sigma_{r_i}^2 r_j^6 \sigma_{r_j}^2}. \end{aligned} \quad (3.36)$$

Table 3.1: Power index of sensors in Figure 3.2 for bearings-only localization

Sensor number	1	2	3	4	5
Power index ($10^{-1} \times$)	1.07	0.84	1.00	0.67	0.33

Table 3.2: Power index of sensors in Figure 3.3 for bearings-only localization

Sensor number	1	2	3	4	5
Power index ($10^{-1} \times$)	0.91	1.00	0.89	0.51	0.26

and the power index of a sensor for range-only localization can be expressed as

$$\rho_i^{\text{Range}} = 8a^8 \sum_{j=1}^N \frac{\sin^2(\theta_i - \theta_j)}{r_i^6 \sigma_{r_i}^2 r_j^6 \sigma_{r_j}^2}. \quad (3.37)$$

As a numerical example, we again consider the configuration shown in Figure 3.2. Table 3.3 contains the power index of the sensors, assuming that $\sigma_{r_i} = 2$ for all sensors and $a^2 = 700$. This table suggests that sensor 3 is the most informative sensor for range-only localization.

3.4.2 Tracking

This subsection illustrates briefly how the computation of power indices can be extended to tracking applications. We consider bearings-only tracking in one dimension where the position and velocity of a target must be estimated after collecting measurements for T time instants. The position of the target at time k for $k = 1, 2, \dots, T$

Table 3.3: Power index of sensors in Figure 3.2 for range-only localization

Sensor number	1	2	3	4	5
Power index ($10^{-1} \times$)	1.64	0.74	1.75	0.58	0.07

is denoted by $x_t(k)$ and is determined by

$$x_t(k) = x_o + kv \quad (3.38)$$

where x_o is the initial position of the target and v is its velocity. To track the target, it suffices to know $\mathbf{p} = [x_o, v]$. The positions of sensors are denoted by $\mathbf{p}_i = [x_i, y_i]$ for $i = 1, 2, \dots, N$. The measurements of sensor i are related to x_o and v via

$$\mathbf{z}_i = \mathbf{h}_i(\mathbf{p}) + \mathbf{v}_i \quad (3.39)$$

where

$$\mathbf{h}_i(\mathbf{p}) = \begin{bmatrix} \text{atan2}(-y_i, x_o + v - x_i) \\ \text{atan2}(-y_i, x_o + 2v - x_i) \\ \vdots \\ \text{atan2}(-y_i, x_o + Tv - x_i) \end{bmatrix} \quad (3.40)$$

and $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_N$ are mutually-independent zero-mean Gaussian random error vectors with covariance matrix $\mathbf{R}_i = \sigma_{b_i}^2 \mathbf{I}$. The derivative of \mathbf{h}_i with respect to \mathbf{p} is given

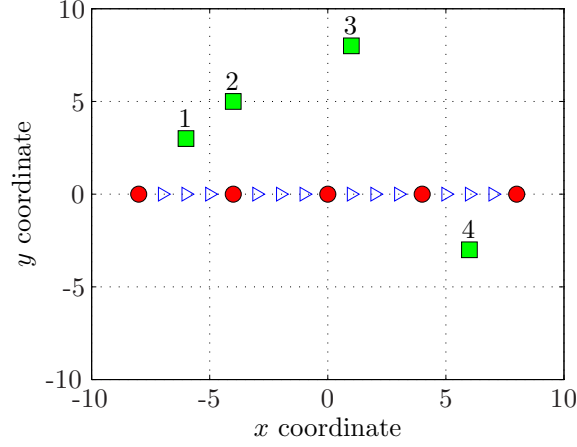


Figure 3.4: A network configuration to study the power index of sensors for bearings-only tracking. Circles identify the position of the target where the measurements are made and the squares identify the position of sensors labeled from 1 to 4.

by

$$\frac{\partial \mathbf{h}_i}{\partial \mathbf{p}} = \begin{bmatrix} -\frac{\sin(\theta_i(1))}{r_i(1)} & \frac{\cos(\theta_i(1))}{r_i(1)} \\ -\frac{\sin(\theta_i(2))}{r_i(2)} & \frac{\cos(\theta_i(2))}{r_i(2)} \\ \vdots & \vdots \\ -\frac{\sin(\theta_i(T))}{r_i(T)} & \frac{\cos(\theta_i(T))}{r_i(T)} \end{bmatrix} \quad (3.41)$$

where $\theta_i(k) = \text{atan2}(-y_i, x_o + kv - x_i)$ and $r_i(k) = \sqrt{y_i^2 + (x_o + kv - x_i)^2}$. The power indices of the sensors can then be computed from (3.24) or (3.25) by first computing $\mathbf{S}_1, \mathbf{S}_2, \dots, \mathbf{S}_N$ according to (3.9) and then computing \mathbf{W} according to (3.8).

As a numerical example, consider the configuration shown in Figure 3.4, where the position and velocity of the target must be estimated after observing the target for $T = 5$ time instants. Table 3.3 contains the power index of the sensors, assuming $\sigma_{b_i} = 10^\circ$ for all sensors. In this example, sensor 3 is found to be the most informative sensor.

Table 3.4: Power index of sensors in Figure 3.2 for bearings-only tracking

Sensor number	1	2	3	4
Power index	6.52	5.54	7.63	6.87

3.5 Concluding Remarks

In this chapter, we demonstrated how a proposed power index can be useful in identifying the informative sensors for distributed parameter estimation. Furthermore, it was shown how this power index can be efficiently computed by choosing an appropriate metric of estimation accuracy. A duality can be noted between solving the sensor subset selection problem and computing power indices. For solving the sensor subset selection problem, some knowledge of the target's parameters is needed although such knowledge may not exist. For computing power indices, some knowledge of the sensors' parameters is needed although such knowledge may not exist. As a remedy, one can assume a probability distribution for certain parameters of sensors and compute the *expected* value of power indices. This can be considered as a natural extension of this work.

The analysis of power indices can ultimately be used in the design of attack-resistant sensor networks by reinforcing the sensors that are deemed more informative for critical values of the target's parameters. Furthermore, model-specific parameters of sensors can be selected such that the power indices of sensors become closer to each other.

Part II

Management of Mobile Sensors

Chapter 4

Parameterization of Optimal Observer Trajectories for Bearings-Only Tracking

4.1 Overview

In this chapter, we focus on the trajectory design problem for bearings-only tracking in two-dimensional space. This problem has been long studied in the literature [25, 48–55] and addresses the management of a mobile sensor, also called the observer, for estimating the position and the velocity of a moving target via bearing measurements. By a mobile sensor, we simply mean a platform that is capable of moving and is equipped with some sensory devices for making measurements.

Due to difficulties with obtaining closed-form expressions for the optimal solution, this problem has been consistently only tackled numerically. A notable attempt in order to obtain closed-form expressions for the optimal solution is the work by Hammel et al. [25] where the authors have shown under the assumptions that they

make for the dynamic model of the observer, the optimal solution for bearings-only localization can be uniquely uniquely specified by a single parameter. We extend these results to the more general case of bearings-only tracking.

In our formulation of the problem, we assume a constant-velocity target, i.e. a target that travels at a constant speed along a straight line. We also assume that measurements are corrupted by mutually-independent additive errors with a Gaussian distribution. With these models for the target and measurements, an open-loop solution of the trajectory design problem is optimal. We, therefore, refer to an open-loop solution as an optimal solution. The optimality of an open-loop solution generally holds for any deterministic model observed via measurements that have mutually-independent additive errors with a Gaussian distribution [4].

Following the assumptions made in [25], we consider an observer that travels at a constant speed but its course can be controlled as desired. We then show that the optimal course control function can be uniquely specified by the so-called range-to-baseline ratio (see Definition 4.3.1) and the speed ratio (see Definition 4.3.2). Our results hold for any statistical or information-theoretic metric of estimation accuracy but we prove these results based on the assumption that a scalar function of the FIM is the metric of estimation accuracy.

We prove our results by showing that, in general, the FIM is only scaled by a constant if a scaling transformation of the problem-specific parameters preserves the value of the range-to-baseline ratio and the speed ratio. More specifically, we show

that if the speed of the observer, the speed of the target, the observation period and the initial distance between the observer and the target are scaled such that the range-to-baseline ratio and the speed ratio remain constant, the FIM is only scaled by a constant and, therefore, the optimal course control function does not change.

We finally numerically generate the optimal course control function for the given values of the range-to-baseline ratio and the speed ratio. For this purpose, we approximate the course control function by a partial sum of the Chebyshev polynomials and numerically find the coefficients of the partial sum that are optimal for the trajectory design problem.

The organization of this chapter is as follows. In Section 4.2, we state the trajectory design problem for bearings-only tracking. In Section 4.3, we prove that the optimal course control functions of the trajectory design problem are invariant under the discussed transformation. We also determine the values of the range-to-baseline ratio and the speed ratio for which the objective function of the trajectory design problem is bounded. Some concluding remarks are also made in Section 4.4.

4.2 Formulation of the Trajectory Design

Problem for Bearings-Only Tracking

In this section, we formulate the trajectory design problem for bearings-only tracking.

We start by introducing the dynamic models for the target and the observer. For the

target, we assume that the following model holds

$$\begin{bmatrix} x_t(k+1) \\ y_t(k+1) \end{bmatrix} = \begin{bmatrix} x_t(k) \\ y_t(k) \end{bmatrix} + T \begin{bmatrix} V_{xt} \\ V_{yt} \end{bmatrix}, \quad (4.1)$$

where T is the sampling time, $[x_t(k), y_t(k)]$ is the position of the target at time instant k for $k = 1, 2, \dots, N$ and $[V_{xt}, V_{yt}]$ denotes its constant velocity vector. Based on this model, the position of the target is uniquely determined at any given time if parameter $\mathbf{p}_{p,v} = [x_t(0), y_t(0), V_{xt}, V_{yt}]$ is known. Hence, the target is tracked if its initial position and its velocity are known.

For the observer, we assume that the following model holds

$$\begin{bmatrix} x_o(k+1) \\ y_o(k+1) \end{bmatrix} = \begin{bmatrix} x_o(k) \\ y_o(k) \end{bmatrix} + V_o T \begin{bmatrix} \cos(u(k)) \\ \sin(u(k)) \end{bmatrix}, \quad (4.2)$$

where $[x_o(k), y_o(k)]$ is the position of the observer at time instant k , $u(k)$ is the observer course control action at time instant k , V_o is its constant speed. As the observer moves according to the course control sequence $\mathcal{U} = \{u(k)\}_{k=0}^{N-1}$, it collects N bearing measurements from time instant $k = 1$ to time instant $k = N$. A measurement at time instant k is related to the position of the target and the position of the observer at that time instant by

$$z(k) = h(k) + v(k), \quad (4.3)$$

where

$$h(k) = \text{atan2}(y_t - y_o(k), x_t - x_o(k)), \quad (4.4)$$

and $v(1), v(2), \dots, v(N)$ are mutually-independent zero-mean Gaussian random errors with variance σ_v^2 .

We next determine how the FIM in bearings-only tracking can be related to the positions of the target and the observer over the observation period.

Proposition 4.2.1 *Based on dynamic models (4.1) and (4.2) and measurement model (4.3), the FIM for estimating $\mathbf{p}_{p,v} = [x_t(0), y_t(0), V_{xt}, V_{yt}]$ when the observer moves according to the course control sequence \mathcal{U} is given by*

$$\mathbf{J}_{p,v}(\mathcal{U}) = \frac{1}{\sigma_v^2} \sum_{k=1}^N \begin{bmatrix} g_1(k) \\ g_2(k) \\ g_3(k) \\ g_4(k) \end{bmatrix} \begin{bmatrix} g_1(k) & g_2(k) & g_3(k) & g_4(k) \end{bmatrix}, \quad (4.5)$$

where

$$g_1(k) = -\frac{y_t(0) + kTV_{yt} - y_o(k)}{(x_t(0) + kTV_{xt} - x_o(k))^2 + (y_t(0) + kTV_{yt} - y_o(k))^2} \quad (4.6)$$

$$g_2(k) = \frac{x_t(0) + kTV_{xt} - x_o(k)}{(x_t(0) + kTV_{xt} - x_o(k))^2 + (y_t(0) + kTV_{yt} - y_o(k))^2} \quad (4.7)$$

$$g_3(k) = kTg_1(k) \quad (4.8)$$

$$g_4(k) = kTg_2(k). \quad (4.9)$$

PROOF According to its definition [1], the FIM is given by

$$\mathbf{J}_{p,v}(\mathcal{U}) = \mathbb{E}_{p(\mathcal{Z}(\mathcal{U})|\mathbf{p})} \left\{ \frac{\partial^T}{\partial \mathbf{p}} \log(p(\mathcal{Z}(\mathcal{U})|\mathbf{p})) \frac{\partial}{\partial \mathbf{p}_{p,v}} \log(p(\mathcal{Z}(\mathcal{U})|\mathbf{p})) \right\}. \quad (4.10)$$

where $\mathcal{Z}(\mathcal{U}) = \{z(k)\}_{k=1}^N$ is the measurement sequence and $p(\mathcal{Z}(\mathcal{U})|\mathbf{p})$ is the probability density of $\mathcal{Z}(\mathcal{U})$ conditioned on \mathbf{p} . From (4.3), we note that

$$p(\mathcal{Z}(\mathcal{U})|\mathbf{p}) = \prod_{k=1}^N \frac{1}{\sigma_v \sqrt{2\pi}} \exp\left(-\frac{(z(k) - h(k))^2}{2\sigma_v^2}\right). \quad (4.11)$$

By plugging (4.11) into (4.10) and following steps similar to those given in the proof of Lemma 2.2.1, the desired result can then be achieved. \blacksquare

Considered separately, the dynamic models for the target and the observe can be fully specified only if $N, T, x_t(0), y_t(0), x_o(0), y_o(0), V_{xt}, V_{yt}, V_o$ and sequence $\mathcal{U} = \{u(k)\}_{k=0}^{N-1}$ are known. However, when these models are considered together, by changing the coordinate system and writing the position of the target relative to the position of the observe, we only need to know N, T, R, V_t, V_o and sequence $\mathcal{U} = \{u(k)\}_{k=0}^{N-1}$ where

$$R = \sqrt{(x_t(0) - x_o(0))^2 + (y_t(0) - y_o(0))^2} \quad (4.12)$$

and

$$V_t = \sqrt{V_{xt}^2 + V_{yt}^2}. \quad (4.13)$$

To this end, we construct a coordinate system following the arrangement shown in

Figure 4.1. In this coordinate system, the origin is the initial position of the observer. The x axis is along the line that connecting the target to the observer at time instant 0 and the y axis is along the direction that the target travels.

It is more convenient to prove our results in continuous domain. We, therefore, let the sampling time \mathbb{T} approach zero in (4.5). By plugging (4.6) and (4.7) into (4.5), we can then express elements of $\mathbf{J}_{p,v}$ as

$$J_{11}(u) = \frac{1}{\sigma_v^2} \int_{t=0}^{\mathbb{T}_f} \frac{(V_t t - y_o(t))^2}{((R - x_o(t))^2 + (V_t t - y_o(t))^2)^2} dt \quad (4.14)$$

$$J_{22}(u) = \frac{1}{\sigma_v^2} \int_{t=0}^{\mathbb{T}_f} \frac{(R - x_o(t))^2}{((R - x_o(t))^2 + (V_t t - y_o(t))^2)^2} dt \quad (4.15)$$

$$J_{33}(u) = \frac{1}{\sigma_v^2} \int_{t=0}^{\mathbb{T}_f} \frac{t^2 (V_t t - y_o(t))^2}{((R - x_o(t))^2 + (V_t t - y_o(t))^2)^2} dt \quad (4.16)$$

$$J_{44}(u) = \frac{1}{\sigma_v^2} \int_{t=0}^{\mathbb{T}_f} \frac{t^2 (R - x_o(t))^2}{((R - x_o(t))^2 + (V_t t - y_o(t))^2)^2} dt \quad (4.17)$$

$$J_{12}(u) = J_{21}(u) = -\frac{1}{\sigma_v^2} \int_{t=0}^{\mathbb{T}_f} \frac{(V_t t - y_o(t))(x_t - x_o(t))}{((R - x_o(t))^2 + (V_t t - y_o(t))^2)^2} dt \quad (4.18)$$

$$J_{13}(u) = J_{31}(u) = \frac{1}{\sigma_v^2} \int_{t=0}^{\mathbb{T}_f} \frac{t(V_t t - y_o(t))^2}{((R - x_o(t))^2 + (V_t t - y_o(t))^2)^2} dt \quad (4.19)$$

$$J_{24}(u) = J_{42}(u) = \frac{1}{\sigma_v^2} \int_{t=0}^{\mathbb{T}_f} \frac{t(R - x_o(t))^2}{((R - x_o(t))^2 + (V_t t - y_o(t))^2)^2} dt \quad (4.20)$$

$$J_{34}(u) = J_{43}(u) = -\frac{1}{\sigma_v^2} \int_{t=0}^{\mathbb{T}_f} \frac{t^2 (V_t t - y_o(t))(x_t - x_o(t))}{((R - x_o(t))^2 + (V_t t - y_o(t))^2)^2} dt \quad (4.21)$$

$$J_{14}(u) = J_{41}(u) = J_{23}(u) = J_{32}(u) = -\frac{1}{\sigma_v^2} \int_{t=0}^{\mathbb{T}_f} \frac{t(V_t t - y_o(t))(R - x_o(t))}{((R - x_o(t))^2 + (V_t t - y_o(t))^2)^2} dt \quad (4.22)$$

where \mathbb{T}_f denotes the observation period and the course control sequence \mathcal{U} in discrete domain has been replaced with the course control function u in continuous domain.

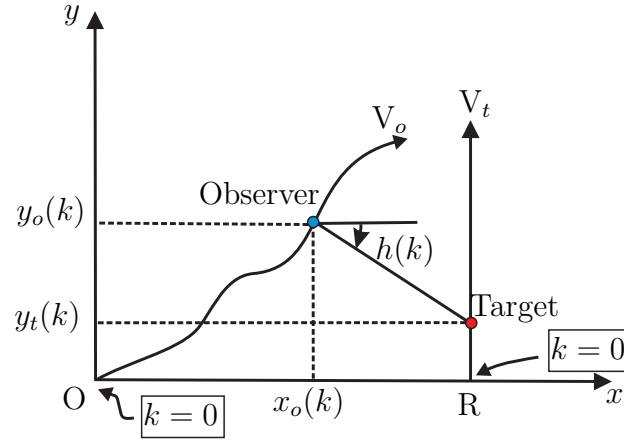


Figure 4.1: Geometry of bearings-only tracking by an observer.

We can now formally define the trajectory design problem.

Definition 4.2.1 *The observer trajectory design problem in continuous domain for bearings-only tracking can be posed as solving*

$$\begin{aligned}
 & \max_u \quad \psi(\mathbf{J}_{p,v}(u)) \\
 & \text{such that} \quad \frac{dx_o(t)}{dt} = V \cos(u(t)) \quad \frac{dy_o(t)}{dt} = V \sin(u(t)), \quad (4.23) \\
 & \quad \quad \quad x_o(0) = y_o(0) = 0
 \end{aligned}$$

over the interval $[0, T_f]$ where the elements of $\mathbf{J}_{p,v}(u)$ are given by (4.14) to (4.22) and ψ is an appropriate function, e.g. the determinant, that assigns a real number to $\mathbf{J}_{p,v}(u)$.

4.3 Invariant Transformation of the Trajectory

Design Problem

We now define the range-to-baseline ratio and the speed ratio and prove that an optimal solution of problem (4.23) is invariant under a transformation that keeps these two ratios constant. However, prior to proving this result, we determine for what values of the range-to-baseline ratio and the speed ratio, the objective function of problem (4.23) is unbounded. These are the values for which the observer can reach the target over the observation period (see (4.14) to (4.22)). When the observer can reach the target, the optimal course control function is the one that produces this outcome and there is no need to solve the trajectory design problem.

Definition 4.3.1 *The range-to-baseline ratio in problem (4.23) is defined as*

$$C_B = \frac{V_o T_f}{R}. \quad (4.24)$$

Definition 4.3.2 *The speed ratio in problem (4.23) is defined as*

$$C_V = \frac{V_o}{V_t}. \quad (4.25)$$

While we can specify for what combinations of T_f , R , V_t and V_o , the observer can reach the target during the observation period, the following proposition specifies such combinations in terms of C_B and C_V .

Proposition 4.3.1 *The objective function of problem (4.23) is only bounded in the following two cases: (i) $0 < C_B < 1$ and $0 < C_V \leq \infty$ or (ii) $C_B \geq 1$ and $0 < C_V < \frac{C_B}{\sqrt{C_B^2 - 1}}$.*

PROOF Regardless of the value of the speed ratio, if $V_o T_f < R$, the observer can never cover the initial distance between itself and the target during the observation period and hence the first case is bounded. If, on the other hand, $V_o T_f \geq R$ and the observer can reach the target during the observation period, there must be some trajectory for which $y_o(T_f) \geq V_t T_f$. The shortest path to achieve this goal is by traveling along the line that connects $[0, 0]$ to $[R, V_t T_f]$. Therefore, the objective function of the problem is bounded when

$$V_t T_f > V_o T_f \sin(\theta_o) \quad (4.26)$$

$$R = V_o T_f \cos(\theta_o) \quad (4.27)$$

where $\theta_o = \text{atan2}(V_t T_f, R)$. The above relations yield

$$\frac{1}{C_V} > \sqrt{1 - \frac{1}{C_B^2}} \quad (4.28)$$

which proves the second case. ■

Figure 4.2 illustrates the values of the range-to-baseline ratio and the speed ratio for which the objective function of problem (4.23) is bounded. We are now ready to state the main result of this chapter.

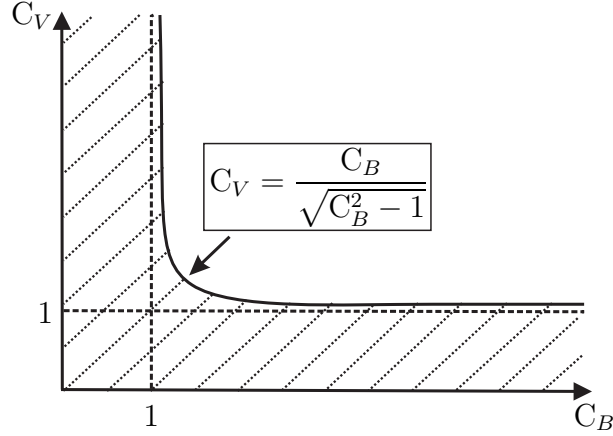


Figure 4.2: Bounded region for the trajectory design problem. The shaded area indicates the values of C_B and C_V for which objective function of the trajectory design problem is bounded.

Proposition 4.3.2 *Consider two constant-velocity targets whose dynamic models are given by (4.1) and two observers whose dynamic models are given by (4.2). The first observer must estimate the position of the first target over the interval $[0, T_1]$ while it travels at the constant speed of V_{o1} . Similarly, the second observer must estimate the position of the second target over the interval $[0, T_2]$ while it travels at the constant speed of V_{o2} . Both observers start from the origin of the coordinate system and have identical measurement models given by (4.3). The initial distance of target 1 from observer 1 is R_1 and the initial distance of target 2 from observer 2 is R_2 . Targets 1 and 2 travel at the speed of V_{t1} and V_{t2} along the y axis, respectively. For a given function u defined over the absolute interval $[0, 1]$, the course control function of observer 1 at time t is determined by $u_1(t) = u(t/T_1)$ and the course control function of observer 2 at time t is determined by $u_2(t) = u(t/T_2)$. Under these two scenarios, it then holds that for some constant K , $\mathbf{J}_{p,v}(u_1) = K\mathbf{J}_{p,v}(u_2)$ as long as the*

relation between $R_1, R_2, V_{t1}, V_{t2}, V_{o1}, V_{o2}, T_1$ and T_2 is such that $\frac{V_{o1}T_1}{R_1} = \frac{V_{o2}T_2}{R_2}$ and $\frac{V_{o1}}{V_{t1}} = \frac{V_{o2}}{V_{t2}}$.

PROOF We prove that the result holds for the first elements of $\mathbf{J}_{p,v}(u_1)$ and $\mathbf{J}_{p,v}(u_2)$.

A similar argument can be then pursued for other elements. Instead of working with two different time scales, we re-parameterize the trajectories of both observers as follows

$$\begin{aligned} \frac{dx_{o1}(\xi)}{d\xi} &= V_{o1}T_1 \cos(u(\xi)), & \frac{dy_{o1}(\xi)}{d\xi} &= V_{o1}T_1 \sin(u(\xi)), \\ x_{o1}(0) &= y_{o1}(0) = 0, \end{aligned} \tag{4.29}$$

$$\begin{aligned} \frac{dx_{o2}(\xi)}{d\xi} &= V_{o2}T_2 \cos(u(\xi)), & \frac{dy_{o2}(\xi)}{d\xi} &= V_{o2}T_2 \sin(u(\xi)), \\ x_{o2}(0) &= y_{o2}(0) = 0 \end{aligned} \tag{4.30}$$

where $[x_{o1}(\xi), y_{o1}(\xi)]$ and $[x_{o2}(\xi), y_{o2}(\xi)]$ denote the positions of observers 1 and 2 for a given $\xi \in [0, 1]$. We then note that

$$\begin{aligned} x_{o2}(\xi) &= \int_0^\xi V_{o2}T_2 \cos(u(\alpha) + u(0))d\alpha = \frac{V_{o2}T_2}{V_{o1}T_1}x_{o1}(\xi) \\ y_{o2}(\xi) &= \int_0^\xi V_{o2}T_2 \sin(u(\alpha) + u(0))d\alpha = \frac{V_{o2}T_2}{V_{o1}T_1}y_{o1}(\xi). \end{aligned} \tag{4.31}$$

Therefore, the first element of $\mathbf{J}_{p,v}(u_2)$ can be expressed as

$$\begin{aligned}
 \frac{1}{\sigma_v^2} \int_{t=0}^{T_2} \frac{(V_{t_2}t - y_{o2}(t))^2 dt}{((R_2 - x_{o2}(t))^2 + (V_{t_2}t - y_{o2}(t))^2)^2} &= \frac{T_2}{\sigma_v^2} \int_{\xi=0}^1 \frac{(V_{t_2}T_2\xi - y_{o2}(\xi))^2 d\xi}{((R_2 - x_{o2}(\xi))^2 + (V_{t_2}T_2\xi - y_{o2}(\xi))^2)^2} \\
 &= \frac{T_2}{\sigma_v^2} \int_{\xi=0}^1 \frac{(\frac{V_{o2}T_2}{V_{o1}T_1}V_{t_1}T_1\xi - \frac{V_{o2}T_2}{V_{o1}T_1}y_{o1}(\xi))^2 d\xi}{((\frac{V_{o2}T_2}{V_{o1}T_1}R_1 - \frac{V_{o2}T_2}{V_{o1}T_1}x_{o1}(\xi))^2 + (\frac{V_{o2}T_2}{V_{o1}T_1}V_{t_1}T_1\xi - \frac{V_{o2}T_2}{V_{o1}T_1}y_{o1}(\xi))^2)^2} \\
 &= \frac{V_{o1}^2 T_1^2}{V_{o2}^2 T_2 \sigma_v^2} \int_{\xi=0}^1 \frac{(V_{t_1}T_1\xi - y_{o1}(\xi))^2 d\xi}{((R_1 - x_{o1}(\xi))^2 + (V_{t_1}T_1\xi - y_{o1}(\xi))^2)^2} \\
 &= \frac{V_{o1}^2 T_1}{V_{o2}^2 T_2 \sigma_v^2} \int_{t=0}^{T_1} \frac{(V_{t_1}t - y_{o1}(t))^2 dt}{((R_1 - x_{o1}(t))^2 + (V_{t_1}t - y_{o1}(t))^2)^2} \\
 &= \frac{K}{\sigma_v^2} \int_{t=0}^{T_1} \frac{(V_{t_1}t - y_{o1}(t))^2 dt}{((x_{t_1} - x_{o1}(t))^2 + (V_{t_1}t - y_{o1}(t))^2)^2}
 \end{aligned} \tag{4.32}$$

where we have used (4.31) and the fact that the range-to-baseline ratio and the speed ratio are constant to prove the desired result. ■

As a result of the above proposition, if a course control function is an optimal solution of problem (4.23) with T_1, R_1, V_{t_1} and V_{o1} , it is also an optimal solution of problem (4.23) with T_2, R_2, V_{t_2} and V_{o2} . It must be only ensured that the function is applied over the appropriate time scale.

Remark By following similar steps, we can show that the FIM for estimating the position of a stationary target when the course control function u is applied can be expressed as

$$\mathbf{J}_p(u) = \left[\begin{array}{cc} J_{11} & J_{12} \\ J_{12} & J_{22} \end{array} \right] \Bigg|_{V_t=0} \tag{4.33}$$

where J_{11} , J_{22} and J_{12} are given by (4.14), (4.15) and (4.18), respectively. Hence,

Proposition 4.3.2 also implies that the optimal course control function for bearings-only localization is uniquely specified by the range-to-baseline ratio. Note, however, that the optimal course control function for estimating just the position of a stationary target is different than the optimal course control function for estimating both the position and the velocity of that target (see Figures 4.3 and 4.4). In both cases, the target is stationary but we are certain about this fact only in the first case. The first case involves finding a function that maximize a scalar function of $\mathbf{J}_p(u)$ whereas the second case involves finding a function that maximize a scalar function of $\mathbf{J}_{p,v}(u)$.

Although we have shown that the optimal solutions of problem (4.23) can be specified by the range-to-baseline ratio and the speed ratio, it is still not possible to obtain closed-form expressions for these solutions and one has resort to numerical methods for computing them.

To this end, we can restrict the admissible course control functions to those which can be represented by a partial sum of the Chebyshev polynomials [24], i.e. for a given integer Q , function u can be written as

$$u(t) = \sum_{i=0}^Q m_i f_i(t), \quad (4.34)$$

where

$$\begin{aligned} f_0(t) &= 1, & f_1(t) &= t, \\ f_i(t) &= 2t f_{i-1}(t) - f_{i-2}(t). \end{aligned} \quad (4.35)$$

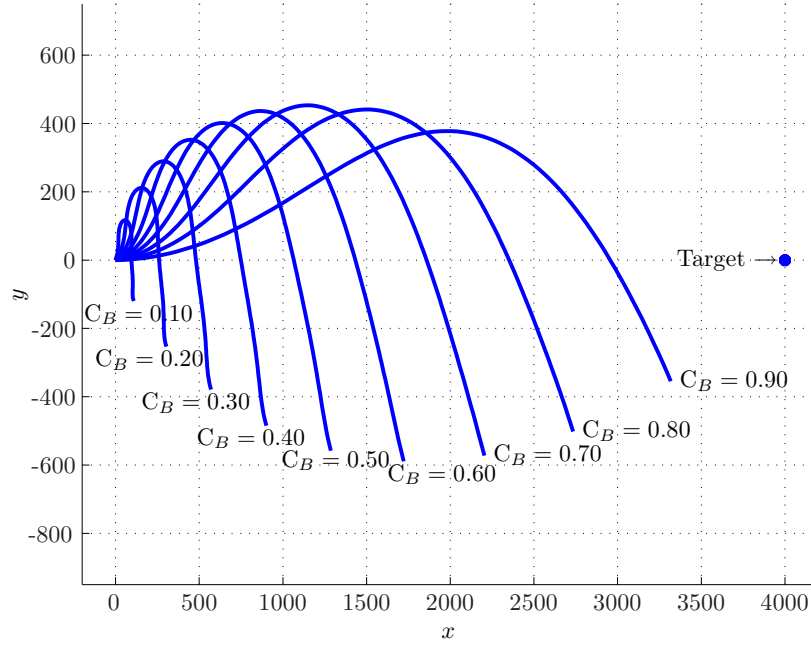


Figure 4.3: Observer trajectories for bearings-only tracking when the target is stationary ($C_V = \infty$).

We can then find coefficients $m_1^*, m_2^*, \dots, m_Q^*$ that maximize the objective function of problem (4.23) for the given values of C_B and C_V .

Figure 4.3 illustrates the trajectories that are generated based on using the first 7 Chebyshev polynomials when $C_V = \infty$ and C_B varies from 0 to 1, i.e. over the range that the objective function of problem (4.23) is bounded. These trajectories are computed with considering the determinant of the FIM as the objective function of problem (4.23). Since $C_V = \infty$, these trajectories are for a stationary target. Following our earlier remark, however, note that in computing these trajectories, it is assumed that both the position and the velocity of the target are being estimated, i.e. these trajectories are for bearings-only tracking. The trajectories suggest that the observer must visit both half-planes of the plane below and above the x axis in

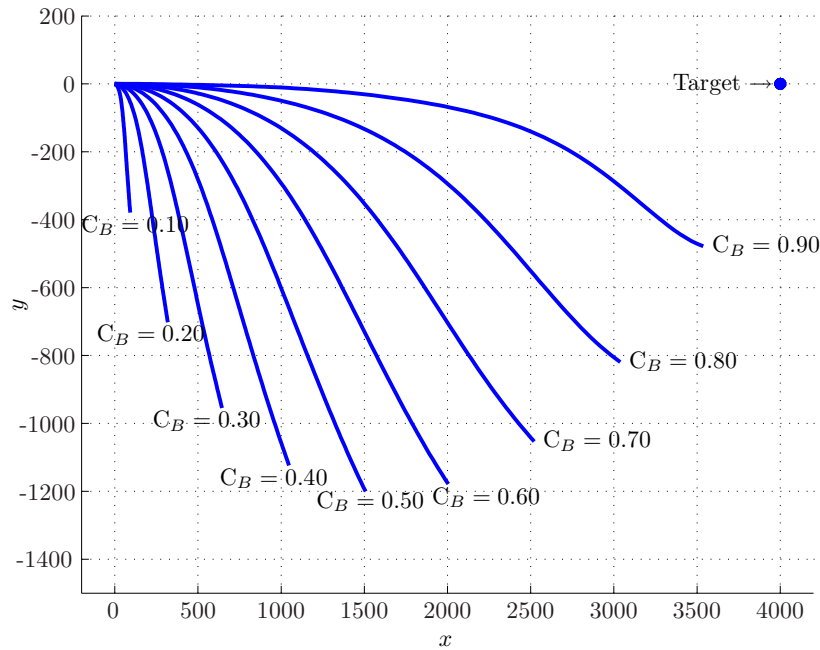


Figure 4.4: Observer trajectories for bearings-only localization ($C_V = \infty$).

order to obtain an accurate estimate of the velocity of the target. Compare these trajectories with those shown in Figure 4.4 where the target is still assumed to be stationary but only its position is being estimated, i.e. these trajectories are for bearings-only localization. They suggest that the target does not need to visit both half-planes of the plane below and above the x axis for bearings-only localization. A general trend is also seen in these trajectories. In the early stages of the observation period, the observer attempts to move directly towards the target and reduces its distance with the target. However, as we approach the final stages of the observation period, the observer changes its course and attempts to generate more diversity in bearing measurements.

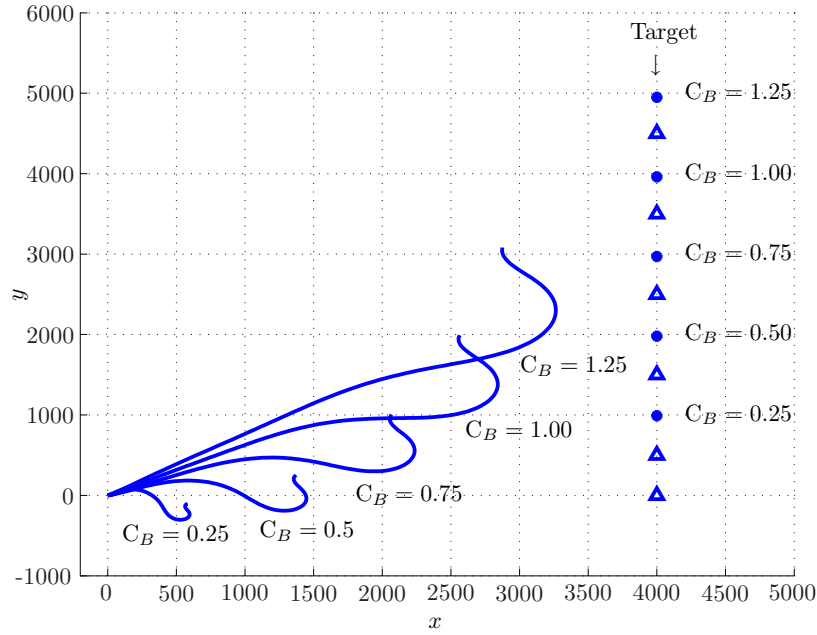


Figure 4.5: Observer trajectories for bearings-only tracking when the target and the observer are traveling at the same speed ($C_V = \infty$). The dots indicate the position of the target at the end of the observation period for a given value of the range-to-baseline ratio.

Figure 4.5 also illustrates the trajectories when $C_V = 1$ and C_B varies from 0.25 to 1.25. These trajectories have some similarities with those shown in Figure 4.3. However, since the target moves here, the observer attempts to pursue the target and maintain its distance from it low. Also, a sharp turn is more pronounced at the end of the trajectories.

4.4 Concluding Remarks

In this chapter, we showed how the trajectory design problem for bearings-only tracking can be uniquely specified by only two ratios. Reducing the number of the parameters that the optimal solutions of a problem depends on can be greatly beneficial

when only numerical solutions of that problem can be obtained. For the trajectory design problem, a family of optimal solutions can be generated based on the two discussed ratios and used for any scenario encountered in real-world applications.

Chapter 5

Conclusion

5.1 Summary of Work Accomplished

With recent advances in embedded processors, microelectromechanical systems and wireless technology, it can be only expected that the world becomes a more instrumented place. As the applications for sensors broaden, new challenges also arise on how to manage the operation of these devices. In this dissertation, we explored several important themes on this subject. In particular, we studied the sensor subset selection problem for fixed sensors and the trajectory design problem for mobile sensors. We highlighted some of the characteristics of these problems that can be exploited in the development of more efficient solution algorithms. In connection with the sensor subset selection problem, we also proposed a novel approach to quantify how informative a sensor is relative to other sensors.

For the sensor subset selection problem, we mainly focused on the determinant of the FIM as the metric of estimation accuracy. As a combinatorial optimization problem, we studied two well-known upper bounds for this problem: (i) the Lagrangian bound and (ii) the continuous bound. We showed that the determinant of the FIM

is a supermodular function from which it followed that the Lagrangian bound for the sensor subset selection problem can be computed in polynomial time. We outlined an algorithm for computing this bound. We further noted that the continuous relaxation of the sensor subset selection problem can be transformed to a convex optimization problem from which it followed that the continuous bound can be computed in polynomial time as well. Through numerical results and by comparing its performance with the continuous bound, we showed that a greedy algorithm is very efficient for solving the sensor subset selection problem. We pointed to the benefit of using the natural selection process to solve the continuous relaxation of a variation of the sensor subset selection problem where sensors are allowed to make more than one measurement.

Motivated by the concept of probabilistic symmetric values in cooperative game theory, we introduced a power index for sensors to identify which ones are more informative than others. We proved that the power index can be computed in polynomial time if the determinant of the FIM is chosen as the metric of estimation accuracy. We derived formulas to compute the power indices for various probability distributions over the size of the selected subset. We discovered that if the parameter with the unknown value has only two elements and the measurement from a single sensor cannot provide information on both elements of the parameter simultaneously, the power indices induce the same ranking among the sensors regardless of the probability distribution considered for the size of the selected subset. Finally, we provided formulas

to compute the power indices for range and bearing (angle) sensors. Interestingly enough, we discovered that these formulas have very similar structures as each type reveals information about one dimension of the target's position that the other type misses.

For the trajectory design problem, we recognized that the optimal solutions of this problem can be uniquely specified by the range-to-baseline ratio and the speed ratio. We noted the range of these values that bounds the trajectory design problem. We also numerically investigated the characteristics of the optimal solutions of the trajectory design problem by approximating these solutions with a partial sum of the Chebyshev polynomials.

5.2 Directions for Future Work

Here we present some directions and recommendations for future research.

1. **Alternative upper bounds for the sensor subset selection problem:**

In addition to the Lagrangian and continuous bound, there are other upper bounds for combinatorial optimization problems. These upper bounds are, in general, tighter than the bounds we discussed but they are also more difficult to compute. We noted that the sensor subset selection problem is a non-linear (supermodular) knapsack problem. There are at least two types of alternative bounds that have been studied in the literature for the quadratic knapsack problem (see [15] and references therein). These bounds can be further investigated in the context of the sensor subset selection

problem which is more general than the quadratic knapsack problem. One alternative bound is obtained from the Lagrangian decomposition of the quadratic knapsack problem where extra variables are injected such that the dual function can be written in two parts. From these two parts, one can be solved in polynomial time. The effort can then be concentrated on solving the other part. There are also bounds proposed based on the idea of upper planes where the quadratic objective function is replaced with a linear one that is larger everywhere in the feasible region. One interesting question related to these bounds for the sensor subset selection problem asks how the coefficients of the linear function can be determined directly from the parameters of the problem such as the positions of sensors and the target.

2. Power indices of sensors in dynamic scenarios: It is interesting to investigate how a power index can be defined and computed in dynamic scenarios. In Chapter 3, we analyzed the power indices of sensors for tracking a target (see Section 3.4.2). However, we assumed that the same selected subset remains active during the entire observation period. It is, in fact, more realistic to assume that this subset will change as time passes. One approach for defining and computing the power index in dynamic scenarios can be established by considering a sequence of cooperative parameter estimation games which are parameterized by the position of the target at a given time instant. We then define the value of a subset as the sum of its value in each individual game and compute the power index based on that. This approach resembles the work discussed in [63] on dynamic cooperative games and

their solution concepts.

3. Symmetry analysis of the trajectory design problem: The results that we obtained in Chapter 4 can be potentially used as a basis for the symmetry analysis [64] of the trajectory design problem. The parameterization of the optimal solutions of the trajectory design problem, in fact, points to the symmetries of this problem. As we saw, these symmetries reduced the complexity of the problem. The analysis of these or other symmetries can eventually lead to further insight into the characteristics of the optimal solutions.

4. Multiple Targets/Multiple Observers: A natural extension to what we discussed in Chapters 2 and 3 is to consider estimating multiple parameters, i.e. the position of multiple targets. For multiple parameters, various formulations of the sensor subset selection problem may arise depending how we assume the sensors make their measurements and how the constraints of the problem are set up. Similarly, we can analyze how to parameterize the optimal solutions of the trajectory design problem if multiple targets or observers exist.

Bibliography

- [1] H. L. V. Trees, *Detection, Estimation, and Modulation Theory*. John Wiley, 1968.
- [2] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice Hall, 1993.
- [3] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons, 2001.
- [4] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Artech House, 2004.
- [5] M. Chu, H. Haussecker, and F. Zhao, “Scalable information-driven sensor querying and routing for ad hoc heterogeneous sensor networks,” *International Journal of High Performance Computing Applications*, vol. 16, no. 3, pp. 293–314, 2002.
- [6] P. Tichavsky, C. H. Muravchik, and A. Nehorai, “Posterior Cramer-Rao bounds for discrete-time nonlinear filtering,” *IEEE Transactions on Signal Processing*, vol. 46, no. 5, pp. 1386–1396, 1998.

- [7] S. P. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.
- [8] F. Pukelsheim, *Optimal Design of Experiments*. SIAM, 2006.
- [9] D. P. Bertsekas, *Dynamic Programming and Optimal Control*. Athena Scientific, 2000.
- [10] J. Evans and V. Krishnamurthy, “Optimal sensor scheduling for hidden Markov model state estimation,” *International Journal of Control*, vol. 74, no. 18, pp. 1737–1742, 2001.
- [11] V. Krishnamurthy, “Algorithms for optimal scheduling and management of hidden Markov model sensors,” *IEEE Transactions on Signal Processing*, vol. 50, no. 6, pp. 1382–1397, 2002.
- [12] L. Meier, J. P. Peschon, and R. M. Dressler, “Optimal control of measurement subsystems,” *IEEE Transactions on Automatic Control*, vol. 12, no. 5, pp. 528–536, 1967.
- [13] A. Athans, “On the determination of optimal costly measurement strategies for linear stochastic systems,” *Automatica*, vol. 8, pp. 397–412, 1972.
- [14] T. Ibaraki and N. Katoh, *Resource Allocation Problems*. MIT Press, 1988.
- [15] D. Li and X. Sun, *Nonlinear Integer Programming*. Springer-Verlag, 2006.

Bibliography

- [16] L. A. Wolsey and G. L. Nemhauser, *Integer and Combinatorial Optimization*. Wiley-Interscience, 1999.
- [17] G. Gallo and B. Simeone, “On the supermodular knapsack problem,” *Mathematical Programming*, vol. 45, no. 1-3, pp. 295–309, 1989.
- [18] M. Groetschel, L. Lovasz, and A. Schrijver, “The ellipsoid method and its consequences in combinatorial optimization,” *Combinatorica*, vol. 1, pp. 169–197, 1981.
- [19] C. C. Li, *First Course in Population Genetics*. Boxwood Press, 1976.
- [20] D. Monderer and D. Samet, “Variations on the Shapley value,” in *Handbook of Game Theory with Economic Applications*, R. Aumann and S. Hart, Eds. Elsevier Science, 2002, vol. 3, ch. 54, pp. 2055–2076.
- [21] X. Deng and C. H. Papadimitriou, “On the complexity of cooperative solution concepts,” *Mathematics of Operations Research*, vol. 90, no. 2, pp. 257–266, 1994.
- [22] J. Banzhaf, “Weighted voting doesn’t work: a mathematical analysis,” *Rutgers Law Review*, vol. 19, pp. 317–343, 1965.
- [23] L. S. Shapley and M. Shubik, “A method for evaluating the distribution of power in a committee system,” *American Political Science Review*, vol. 48, pp. 787–792, 1954.
- [24] R. L. Burden and J. D. Faires, *Numerical Analysis*. Brooks Cole, 2004.

- [25] S. E. Hammel, P. T. Liu, E. J. Hillard, and K. F. Fong, “Optimal observer motion for localization with bearing measurements,” *Computers and Mathematics with Applications*, vol. 18, no. 1-3, pp. 171–180, 1989.
- [26] F. Zhao, J. Shin, and J. Reich, “Information-driven dynamic sensor collaboration,” *IEEE Signal Processing Magazine*, vol. 16, pp. 293–313, 2002.
- [27] J. Liu, J. Reich, and F. Zhao, “Collaborative in-network processing for target tracking,” *EURASIP Journal on Applied Signal Processing*, vol. v 2003, no. 4, pp. 378–391, 2003.
- [28] H. Wang, G. Pottie, K. Yao, and D. Estrin, “Entropy-based sensor selection heuristic for target localization,” in *Proceedings of International Symposium on Information Processing in Sensor Networks*, Berkeley, California, 2004.
- [29] L. Kaplan, “Global node selection for localization in a distributed sensor network,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 42, no. 1, pp. 113–135, 2006.
- [30] —, “Local node selection for localization in a distributed sensor network,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 42, no. 1, pp. 136–146, 2006.
- [31] R. Tharmarasa, T. Kirubarajan, and M. L. Hernandez, “Large-scale optimal sensor array management for multitarget tracking,” *IEEE Transactions on Systems, Man and Cybernetics, Part C*, vol. 37, no. 5, pp. 803–814, 2007.

- [32] A. Krause and C. Guestrin, “Near-optimal nonmyopic value of information in graphical models,” in *Proceedings of Uncertainty in Artificial Intelligence*, Edinburgh, Scotland, 2005.
- [33] M. Cardei and D. Du, “Improving wireless sensor network lifetime through power-aware organization,” vol. 11, no. 3, pp. 233–340, 2005.
- [34] M. L. Hernandez, T. Kirubarajan, and Y. Bar-Shalom, “Multisensor resource deployment using posterior cramer-rao bounds,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 40, no. 2, pp. 399–416, 2004.
- [35] A. Krause, A. Singh, and C. Guestrin, “Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies,” *Journal of Machine Learning Research*, vol. 9, pp. 235–284, 2008.
- [36] H. Rowaihy, S. Eswaran, M. Johnson, D. Verma, A. Bar-Noy, T. Brown, and T. La Porta, “A survey of sensor selection schemes in wireless sensor networks,” in *Proceedings of SPIE*, vol. 6562, no. 65621A1, 2007.
- [37] A. Chhetri, D. Morell, and A. Papandreou-Supapola, “Sensor scheduling using a 0-1 mixed integer programming framework,” in *Proceedings of IEEE Workshop on Sensor Array and Multichannel Processing*, Waltham, MA, 2006.
- [38] Z. Han and H. V. Poor, “Coalition games with cooperative transmission: a cure for the curse of boundary nodes in selfish packet-forwarding wireless networks,”

in *Proceedings of International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks*, Limassol, Cyprus.

- [39] M. Madiman, “Cores of cooperative games in information theory,” *EURASIP Journal on Wireless Communications and Networking*, no. 318704, 2008.
- [40] J. C. Hassab, *Underwater Signal and Data Processing*. CRC Press, 1989.
- [41] S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*. Artech House, 1999.
- [42] S. E. Hammel and V. J. Aidala, “Observability requirements for three-dimensional tracking via angle measurements,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 21, no. 2, pp. 200–207, 1985.
- [43] E. Fogel and M. Gavish, “Nth-order dynamics target observability from angle measurements,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 24, no. 3, pp. 305–308, 1988.
- [44] R. R. Mohler and C. S. Hwang, “Nonlinear data observability and information,” *Journal of the Franklin Institute*, vol. 325, no. 4, pp. 443–464, 1988.
- [45] K. Becker, “Simple linear theory approach to tma observability,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 29, no. 2, pp. 575–578, 1993.

- [46] T. L. Song, “Observability of target tracking with bearings-only measurements,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 32, no. 4, pp. 1468–1472, 1996.
- [47] P. Shar and X. R. Li, “A practical approach to observability of bearings-only target tracking,” in *Proceedings of SPIE Conference on Signal and Data Processing of Small Targets*, Denver, Colorado, 1999.
- [48] J. A. Fawcett, “Effect of course maneuvers on bearings-only range tracking,” *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 36, no. 8, pp. 1193–1199, 1988.
- [49] J. P. Helferty and D. R. Mudgett, “Optimal observer trajectories for bearings-only tracking by minimizing the trace of the cramer-rao lower bound,” in *Proceedings of IEEE Conference on Decision and Control*, San Antonio, Texas, 1993.
- [50] A. Logothetis, A. Isaksson, and R. Evans, “An information theoretic approach to observer path design for bearings-only tracking,” in *Proceedings of IEEE Conference on Decision and Control*, San Diego, California, 1997.
- [51] A. Logothetis, A. Isaksson, and R. J. Evans, “Comparison of suboptimal strategies for optimal own-ship maneuvers in bearings-only tracking,” in *Proceedings of American Control Conference*, Philadelphia, Pennsylvania, 1998.

- [52] Y. Oshman and P. Davidson, “Optimization of observer trajectories for bearings-only target localization,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 35, no. 3, pp. 892–902, 1999.
- [53] O. Trémois and J.-P. L. Cadre, “Optimal observer trajectory in bearings-only tracking for manoeuvring sources,” *IEE Proceedings of Radar, Sonar and Navigation*, vol. 146, no. 1, pp. 31–39, 1999.
- [54] M. Herdandez, “Optimal sensor trajectories in bearings-only tracking,” in *Proceedings of International Conference on Information Fusion*, Stockholm, Sweden, 2004.
- [55] E. W. Frew and S. M. Rock, “Exploratory motion generation for monocular vision-based target localization,” in *Proceedings of IEEE Aerospace Conference*, 2002.
- [56] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Wiley-Interscience, 1991.
- [57] D. M. Topkis, *Supermodularity and Complementarity*. Princeton University Press, 1998.
- [58] F. Zhang, *Matrix Theory*. Springer-Verlag, 1999.
- [59] S. Fujishige, *Submodular Functions and Optimization*. Elsevier, 1991.

Bibliography

- [60] T. S. Motzkin and E. G. Straus, “Maxima for graphs and a new proof of a theorem of Tuán,” *Canadian Journal of Mathematics*, vol. 17, pp. 533–540, 1965.
- [61] I. M. Bomze and V. Stix, “Genetic engineering via negative fitness: Evolutionary dynamics for global optimization,” *Annals of Operations Research*, vol. 89, pp. 297–318, 1999.
- [62] J. F. C. Kingman, “On an inequality in partial averages,” *Quarterly Journal of Mathematics*, vol. 12, pp. 78–80, 1961.
- [63] J. A. Filar and L. A. Petrosjan, “Dynamic cooperative games,” *International Game Theory Review*, vol. 2, no. 1, pp. 47–65, 2000.
- [64] G. Bluman, *Symmetry and Integration Methods for Differential Equations*. Springer-Verlag, 2002.