UNIVERSITY OF CALIFORNIA
RIVERSIDE

Distributed Constrained Bayesian Optimization: Autonomous Camera Control

A Dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Electrical Engineering

by

Akshay Ajit Morye

December 2013

Dissertation Committee:

Dr. Jay A. Farrell , Chairperson
Dr. Amit K. Roy-Chowdhury
Dr. Elisa Franco

The Dissertation of Akshay Ajit Morye is approved:

Committee Chairperson

University of California, Riverside

Acknowledgments

I wish to thank my parents, Asmita & Ajit Morye, my grandfather Vishnu Morye, and my brother Abhi, for being my pillars of strength. I owe them everything.

This dissertation has come to fruition due to a multitude of reasons, chief among which is my advisor, Professor Jay A. Farrell. Having first joined the University of California Riverside for a Master of Science course, it was Professor Farrell who cajoled me into furthering my technical education, and afforded me the opportunity to ply my trade as a researcher. Had he not drawn parallels between the Visual Sensing application and Soccer, a sport I am a fervent follower of, I probably would not have plowed on beyond a Masters degree. Professor Farrell's infectious hunger for research in Autonomous Systems; his exceptional mentoring abilities, all coupled with an affable personality, are few of the telling factors that have changed the course of my academic career. I cannot thank him enough for his support and friendship.

Along with the immense contribution of my advisor, a number of other prominent Professors have played an invaluable role in my professional development. Professor Ertem Tuncel got me up to speed on the concepts of Stochastic Processes. Discounting that, none of the further advances I made would have been possible. Professor Anastasios Mourikis' instruction of Advanced Robotics had a similarly inspiring effect. Although the course was one of the tougher assignments, it was rewarding in the amount of knowledge gained. Professor Elisa Franco's simple, yet extremely effective approach to teaching Advanced Optimal Control gave much needed direction to my research. My fledgling understanding of Image Processing and Computer Vision was considerably strengthened under the tutelage of Professor Amit K. Roy-Chowdhury. His treatment of the concepts integral to any vision application have kept me in good standing within

a very competitive field of research. None of the research progress I have made would have been possible without his inputs.

The sole reason my family and I were able to afford my graduate education was my Ph.D. grant provided by the Office of Naval Research (ONR). I will forever be in debt of the glorious opportunity thus granted (pun intended). I would like to thank the tax-payers of India and the U.S.A., and those responsible for drafting immigration laws that provide the convenience of obtaining top-notch education. I wish to thank Amy Ricks, Kristine Phengrasamee, Adrienne Thomas, Rebecca Kyles, and Trudi Loder for tending to my student documentation and assisting me whenever need arose. Perhaps, most importantly, I would like to thank UC Riverside for providing a platform through which I could entertain the possibility of acquiring world-class education.

None of my research would have culminated in positive results had it not been for the inputs of Dr. Bi Song, Dr. Ahmed Tashrif Kamal, and Dr. Chong Ding. Bi's insight into the concepts of Game Theory, Tashrif's know-how of Consensus Algorithms, and Chong's immense acumen in Engineering have kept me in good stead. Their constructive critiques have had a profound effect on my work.

This thesis would be incomplete without the mention of my colleagues at the Bourns College of Engineering. I would like to thank Dr. Arvind Ramanandan and Dr. Anning Chen for being the best senior lab-mates one could wish for. Arvind's penchant for treating every question with a mathematical perspective, coupled with a relaxed demeanor, made him an unusually calm personality in lab. I have learnt a lot from him and regard him as a source of motivation. I wish to thank Mingyang Li and Sharat Suvarna for always being at hand to provide help in solving tricky mathematical queries. I would also like to thank Jeff Herrera for the intense and thought provoking discussions on geo-politics and organized religion. Dhananjay Ipparthi, Angelo Pozo,

and I have been party to some fantastic times at lab, and I am thankful to them for making my stay at UCR joyous.

I wish to thank Dr. Monik Khare for being a virtual omniscient, be it while playing sport together, guiding me in research, or bringing forth the welcome distraction of coffee breaks. The Khare family have been my 'other' home in southern California.

A special mention is in order for Aditi Sukhatankar for motivating me to pursue a graduate study program in the USA. I would like to thank Ngocyen Katherine Pham, Frederick Lawton, Elliot Staudt, Gina Kim, Dr. Brad Franklin, Laura Bassols, Francis Recinas, and Guillaume Dupeyron for being friends in their own inimitable ways. I wish to extend gratitude to Nikhil Nagane, Pallavi Panda, Amlan Kusum, Raghuraman Ramamurthy, and the rest of the Indian Students' Association family for providing an outlet to my Indian ways. It was a privilege to be part of, and lead the organization in the company of such wonderful people. Manish Sachdev, Kartik Mani, and Siddhant Mishra have been a special bunch of friends from back home in Mumbai. I also thank Chelsea Football Club and my fellow Footytubers for catering to my needs of following sport. My family of friends are the reason I have been able to maintain personal and professional equilibria within the fun, yet oft stressful environment of constant progress.

I wish to thank all my amazing in-laws, especially, Sangeeta, Narinder, and Naman Passi, for their immense belief and trust in me. The Passis have never flinched in their endeavor to support me to the hilt. I owe my happiness to them.

Saving the best for last, I wish to thank my wife Nimisha Passi for being my companion during all these years as a Ph.D. student. She is the one person that knows and contends with every little nook and cranny of my personality. As put so aptly by Jean Baptiste Massieu, "Gratitude is the memory of the heart", and I wish it all to her.

To Chips, T, and Au'Z.

ABSTRACT OF THE DISSERTATION

Distributed Constrained Bayesian Optimization: Autonomous Camera Control

by

Akshay Ajit Morye

Doctor of Philosophy, Graduate Program in Electrical Engineering
University of California, Riverside, December 2013
Dr. Jay A. Farrell , Chairperson

This dissertation describes methods to autonomously control an *intelligent* camera network with changeable pan, tilt, and zoom (PTZ) parameters for the purpose of obtaining high resolution facial imagery of randomly maneuvering targets. Every camera is treated as a self interested decision making agent that works in cooperation with the other agents in the network to attain a predefined system goal. The per camera per target image quality is designed and defined mathematically to formulate a distributed constrained optimization problem. Each camera is restricted to alter its own PTZ settings. All cameras use information broadcasted by neighboring cameras such that the PTZ parameters of every camera are optimized relative to the global objective. At certain times of opportunity, due to the configuration of the targets relative to the cameras, and the fact that each camera may track many targets, the camera network may be able to reconfigure itself to achieve a required target tracking specification for each target with remaining degrees-of-freedom. The remaining degrees-of-freedom can be used to obtain high resolution facial images from desirable viewing angles for certain targets. The challenge is to design algorithms that autonomously find these time instants, the appropriate imaging camera, and the appropriate parameter settings for all

cameras to capitalize on these opportunities. The methodologies and solutions proposed herein involve a *Bayesian* formulation. The Bayesian formulation automatically trades off objective maximization versus the risk of losing target tracking performance. The dissertation describes a mathematical formulation of the visual sensing problem, design of functions that provide a measure of system performance, development of distributed methodologies that allows cameras to exchange information and asymptotically converge on optimal solutions, and incorporation of planning into the PTZ optimization methodology. The work herein presents theoretical solutions and analyses of results obtained on a simulated network of smart PTZ cameras.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Static camera networks have a lower per camera cost of installation than pan-tilt-zoom (PTZ) camera sensor networks; however, PTZ camera networks can have lower total installation cost with greater performance. Static camera networks must be designed to achieve the coverage and tracking specifications given worst case target distributions. Installation cost constraints can lead to imagery sequences from static camera network applications that are quite challenging to analyze. PTZ camera networks, with appropriate software and communications, can dynamically reconfigure in response to application events and actual target distributions to optimize the acquired imagery sequences accounting for viewpoints and resolution, to facilitate image analysis and scene understanding. It is also desirable in many applications for the tracking and control mechanisms to be distributed due to bandwidth and security constraints. In fact, some applications may necessitate the use of a distributed architecture, e.g. disaster response, military applications, etc.

A prototypical application is a security screening checkpoint at the entrance lobby of a building. Over the course of each day a high volume of people travel through

the room. The room is equipped with a fixed number of cameras while the number and location of people in the room varies with time. *Tracking* a target is defined as estimating the position of the target with tracking performance better than a specified threshold $\bar{\mathbf{T}}$. *Imaging* a target is defined as obtaining an image of a target with resolution exceeding $\bar{r}$. The objective of the camera network is to track (i.e., estimate the state of) all persons in the room at a specified accuracy level at all times, and to capture high-resolution images for certain persons in the room at opportunistically selected time instants.

The challenges of such an application are development of algorithms to ensure accurate propagation of target-related information throughout the distributed network, analysis of the effect of changing network topology on solution convergence, design of distributed PTZ optimization algorithms, and the design of objective and constraint functions suitable to solving the specified problem that also have the properties necessary to ensure convergence. All these factors influence the selection of an optimization strategy.

In this dissertation, this visual sensing problem is designed using a *Bayesian* framework within a game-theoretic paradigm. The Bayesian formulation facilitates automatic trading-off of objective maximization versus the risk of losing track of any target. The game-theoretic design allows the global problem to be decoupled into local problems, one at each camera. The approach is designed with the aim of achieving the global camera network objective through optimization of local camera objectives.

To perform satisfactory surveillance, the installed smart camera network must be able to maintain state estimation performance for all targets above a user-defined threshold, extract information from the changing scene, and obtain high resolution imagery for certain targets, at selected time instants. The solution methodologies utilize distributed constrained optimization (DCOP) methods for computation of the optimal

camera parameter settings, where the feasible set for PTZ parameters is defined by constraints on target tracking performance.

1.1 Problem Statement

Facial detection and recognition are processes that are greatly facilitated by capture of high resolution images of the face from a desired aspect angle [1, 2]. This dissertation does not discuss facial recognition methods; instead, it focuses on the means to configure a network of PTZ cameras to opportunistically acquire high resolution facial imagery. We use the term 'opportunistic' as each camera must select its parameters to satisfy a tracking constraint at all times and to obtain high-resolution facial images at times-of-opportunity. Such an opportunity may arise due to the high probability of image capture at a high zoom setting from a superior aspect angle, and when tracking constraints on all the targets can be simultaneously satisfied. The tracking constraints, while useful in their own right, are necessary to enable high-resolution imaging. Note also that, due to the uncertainty of the target motion, the high zoom setting required by certain cameras to attempt to obtain high-resolution images, will impose tracking risk that must be accounted for appropriately by the camera network.

1.2 Contribution

A method for distributed, cooperative and parallel optimization on a connected camera network is defined, analyzed, and implemented in this dissertation. The camera parameter optimization process maximizes a Bayesian value function that accounts for risk arising from the uncertainty in the estimated target positions, while adhering to Bayesian constraints on the tracking performance. The value function is designed

as an ordinal potential function [3], such that it can be decoupled into local objectives known to each camera. The tracking constraint is common to all cameras. With reasonable assumptions on the connectivity of the network, utilizing a *Bayesian* constrained optimization approach, the proposed solution methodology provides feasible optimal solutions to perform opportunistic visual sensing of targets maneuvering with random trajectories. The dissertation includes an enhanced discussion and analysis of the proposed approach throughout, a detailed simulation example with discussion, development of mechanisms for convergence towards the global optima, and discussion of possible alternative methodologies.

While obtaining high resolution imagery, the camera network may select settings such that they are temporally discontinuous resulting in mechanical wear, and recording a video feed that is unsuitable for image analysis due to motion blur. To address this, the dissertation builds on the developed DCOP methods by incorporating a moving planning horizon optimization approach based on, but distinct from the well established Model Predictive Control framework [4]. Along with constraints on the tracking performance, the approach solves a problem that is expected to adhere to smoothness constraints on the changes in PTZ values. This enables the immediate actions of the camera control module to consider the future effects of these actions. Such an approach results in smoother transitions between subsequent PTZ settings, thus providing a video feed that is smooth and better suited for video analysis and scene understanding.

1.3 Organization of the Dissertation

This dissertation provides a wide-ranging review of research performed in the context of active vision methods on camera networks.

Chapter 2 investigates the visual sensing problem. The chapter begins with a description of aspects implicit to the visual sensing problem. This is followed by a brief literature review of relevant research in the fields of active vision, which is then followed by a review of recent research pertaining to the visual sensing problem herein. Subsequently, a detailed problem description along with notation is provided.

Chapter 3 provides an overview of well-known constrained optimization and game theory concepts. The chapter starts with a mathematical overview of a general centralized constrained optimization method. Subsequently, the chapter briefly describes the concept of *potentiality* within the game-theory literature. The chapter then provides a general outline of how a centralized constrained optimization problem can be decoupled into smaller local constrained optimization problems using the concepts of potentiality.

Chapter 4 focuses on the mathematical formulation of the visual sensing problem herein. The chapter begins with a description of mathematical models associated with target motion and camera measurements. The chapter them provides a detailed description of the desired objective function properties along with a general template of the components that constitute the imaging objective function.

Chapter 5 constitutes the main body of the dissertation. It provides a detailed description of the PTZ optimization methodology. The chapter first defines the global imaging value function to be maximized, along with a description of problem constraints which provide a measure of target tracking performance obtained by the camera network. After formulation of the global optimization problem, the chapter describes a method

to decouple the global problem into smaller local problems using the concepts described in Chapter 3.

Chapter 6 describes the processes involved in distributed optimization. The chapter states assumptions on network connectivity and communication. Subsequently, the chapter describes rules for camera parameter broadcasting and replacement. This is followed with a detailed description of a dynamic average consensus algorithm utilized for asymptotic convergence of local Lagrange multiplier vectors. The chapter also provides a certificate of optimality on the solutions computed using the methodology herein.

Chapter 7 focuses on incorporating a planning scheme within the PTZ optimization methodology to provide temporal PTZ smoothness. The chapter first motivates the importance of planning, and then describes how the proposed approach is distinct from traditional Model Predictive Control (MPC) methods. Thereafter, the chapter defines the objective functions and constraints over a planning horizon, and then formulates the problem within a distributed optimization scheme. This chapter then provides a certificate of optimality on the computed solutions.

Chapter 8 evaluates the proposed method within a realistic simulation environment. A target scenario is setup within an area to be surveyed by a network of cameras. This chapter discusses the results hence obtained, and provides a detailed analysis of the PTZ optimization approach. After evaluation single and multi-trial results of the single-step optimization approach, the chapter analyzes results obtained on incorporation of a planning scheme within the optimization approach. The chapter also provides a detailed discussion on possible solutions and challenges involved in a real-life implementation of the approach.

The dissertation concludes by providing a direction for future research for high resolution imaging on a network of smart vision sensors.

Chapter 2

The Visual Sensing Problem

The research proposed herein falls within the scope of active computer vision [5–7]. Active vision involves research on cooperation and coordination between many cameras in a network, for applications such as autonomous surveillance, simultaneous localization and mapping (SLAM), trajectory planning, etc. There is a large amount of recent work dealing with networks of vision sensors. The following section provides a broad outlay of relevant research.

2.1 Literature Review of Existing Approaches

Research in the area of integrated sensing and scene analysis in camera networks is evolving and standard metrics for comparison or experimental frameworks are not yet available. This remains an area of attention as research in active sensing on camera networks matures. The literature referenced in this section is separated into areas of research that are of importance to visual sensing on camera networks.

2.1.1 Tracking in Camera Networks

Some recent work has dealt with computing the statistical dependence between cameras, computing the camera network topology, tracking over unobserved areas of the network, and camera handoff [8–14]. However, there is little work that deals with distributed tracking and camera control in active PTZ camera networks. The most relevant papers on the topics of tracking and camera parameter selection are discussed below.

In [15], a distributed cluster-based Kalman filter was proposed as a target tracking approach. This method required a camera to aggregate all the measurements of a target to estimate its position before transmitting the result to a central base station. The approach in [8], used herein, considers a different network topology where each camera can only communicate with its neighboring cameras. Each camera has a consensus-based estimate of each target's state removing the need to aggregate measurements at a single cluster head.

A method for tracking targets in a network of PTZ cameras was proposed in [16]. The authors used a mixture of passive and active PTZ cameras to persistently track pedestrians in a virtual environment. This was achieved using a partially distributed, partially centralized hybrid approach. The method herein considers a completely distributed solution using consensus algorithms for target state estimation and a distributed optimization framework for camera parameter selection.

2.1.2 Game-theory in Multi-agent Systems

Game-theory as a tool for designing solutions to multi-agent problems is described in [17]. A vehicle-target assignment problem within the game-theoretic frame-

work was proposed in [18,19]. The standard target assignment problem is different from a camera-target assignment problem in that the problem described therein is a one-to-one mapping problem with the stationary targets, whereas the camera-target assignment problem allows multiple cameras to each track multiple moving targets. Nonetheless, articles [17–19] provide valuable insight into the challenges faced while designing the camera parameter optimization as a cooperative game played by multiple cameras.

A game-theoretic camera control approach for collaborative sensing is proposed in [20–23]. In [20], the agents collaborate to optimize a cost function that is the weighted combination of area coverage over regions of interest while trying to achieve high-resolution images of specific (highly weighted) targets. The authors proposed a distributed tracking and control approach that requires the camera control and tracking to run independently and in parallel. The camera control used game theory to assign camera settings that provided coverage over regions of interest while maintaining a high resolution shot of a target. Concurrently, a Kalman-Consensus filter provided tracks of each target on the ground plane.

The method proposed in this dissertation differs from the one in [20] in that the camera control is aware of the state of the Kalman-Consensus filter and actively seeks to provide it with the best measurements. Furthermore our approach considers the estimate error covariance in addition to the estimated state of each target. This allows us to gauge the risk of failing to capture a feature when attempting high resolution shots.

In [21–23], the agents account for risk and include image quality in a weighted cost function. Collaboration was ensured through a game-theoretic formulation. The quality of the solution was dependent on the specification of user-defined weights.

2.1.3 Camera Network Topology

Assumptions on the camera network communication topology play a major role in the problem solution. Preliminary work on camera networks using independent cameras that lack coordination is provided in [24]. A method involving machine learning is utilized in [12, 25] to learn the network topology. In [12], the authors employ unsupervised machine learning to establish links between target activity and the associated camera views, to decipher communication topology to determine target tracks. In addition to generating the topological structure of the network based on target motion, the method in [25] also provides a measure of inter-camera target transition times, which can be used to support predictive tracking across the camera network. In [13], authors measure the statistical dependence between observations in multiple camera views to quantify a potential interconnecting pathway for target tracks between difference camera views, to enable camera handoff. Multi-agent systems with switching topologies are studied in [26, 27]. Though the studies therein are not based on visual sensing applications, [26, 27] provide significant insight on methods potentially applicable to mobile camera networks.

2.1.4 Distributed Computer Vision

Distributed implementation of computer vision algorithms is studied in [28, 29]. Distributed implementation of state estimation algorithms over vision networks is studied in [30–33]. Though the above articles provide knowledge about the methods and challenges of implementing a computer vision system in a distributed manner, they do not focus on decentralized optimization of the PTZ parameters of the cameras to improve the quality of the acquired imagery.

A recent research survey [28] identifies various computer vision problems that can be solved in a distributed fashion within a network topology. The article mentions the effects of nodes in the network with no measurement information and how such nodes may affect convergence properties of a distributed algorithm. An analysis for stability of multi-agent systems is provided in [34] where the communication network is assumed to be time-variant. The stability analysis is based on graph-theoretic tools where the authors assume the problem to be convex.

An automated annealing approach for updating Lagrange multipliers within a constrained optimization framework to reduce dependence on agent inter-communication is provided in [35]. The method uses the probability collectives framework to generate a relation between game theory and statistical physics. The authors use a game-theoretic motivation to develop a parallel algorithm, but consider a non-cooperative game between agents, where the action of one agent is completely independent of the other agents in the network. Such an assumption is not appropriate for the application herein.

A systematic methodology for designing agent objective functions is outlined in [3], using a hierarchical decoupling of the global optimization function into local objective functions that are aligned with the global objective function. Each agent is modeled as a self-interested decision maker within a game-theoretic environment, then convergence proofs from the game theory literature are utilized. Herein, we utilize the methodology of [3] to decompose a Bayesian value function designed for the opportunistic visual sensing application into local value functions suitable for distributed implementation.

Table 2.1: Comparison between camera network reconfiguration strategies

Approach	Objective	Architecture	Results
[20]	Area coverage	Distributed	Local maxima of total area covered
[22]	Hi-res imaging	Distributed, Sequential	Obtained high-res imagery through weighted Bayesian utility functions
[36]	Static camera placement	Centralized	Global maxima of area covered while considering occlusion
[37]	Weighted area coverage based on prior activity map	Distributed	Local maxima of weighted area covered
[38]	Camera to target assignment	Centralized	Track based one to one mapping (between cameras and targets) and handoff
Methodology in Chapters 5 - 7	Hi-res imaging	Distributed, Parallel	• Opportunistic hi-res imaging using constrained optimization with certificate of optimality; • Bayesian framework to address risk; • Incorporation of planning using a moving horizon

2.1.5 Camera Network Reconfiguration

Table 2.1 provides a comparison of the different methods in the literature that have studied the optimal visual sensing problem for camera network reconfiguration. It should be noted that these approaches have different objective functions, different optimization architectures and different are designed to adhere to different performance criteria.

In [36], authors propose a method that uses a simulated annealing approach to compute a globally optimal configuration for *static* camera networks. The approaches [20, 37, 38] have independent target tracker modules running in the background. The method in [37] uses the target tracker to generate a prior (activity map) and then optimizes the PTZ settings on the weighed area. A path-planning based approach is

proposed in [38], where static cameras track targets, and PTZ cameras obtain high-resolution images. The authors therein use target tracks to predict the paths of targets to do assignments requiring minimal camera assignment switching, but break when targets mingle due to unreliable predictions. The work in [20], while using a distributed tracker, did not combine the camera control and target tracking modules.

Assuming predesigned static camera placement given certain tasks, the articles [39, 40] define deployment strategies for networks of vision sensors.

Given a map of target activity within the surveyed area, the Expectation-Maximization algorithm [41] is used to perform area coverage in [37]. Implementations for tracking a group of people in a multi-camera setup are addressed in [42, 43]. The methods proposed therein dealt with a centralized processing scheme and did not delve into the decentralized organization of cameras.

The integrated visual sensing and scene analysis approach described in this dissertation is a generalization of all of these methods and provides a framework for optimizing the image acquisition capabilities based on satisfaction of system objectives. Such an approach provides an optimal allocation of resources and provides overall efficiency to the system.

2.1.6 Optimization over a Planning Horizon

Model Predictive Control (MPC) [4], also called Receding Horizon Control, has become a popular methodology with applications in control systems where it is desired for the control sequence to cause the system to evolve optimally over a planning horizon, subject to constraints on both the control and state variables. At every sampling instant, the optimizer formulates a new optimal control problem and solves it, based on state information currently available. Intuitively, MPC is a model-based method to solve an

optimization problem with constraints that ensures that the resultant immediate action considers the future effects.

MPC is usually applied to applications with slow dynamics. To address issues related to the challenges of using MPC for applications with fast dynamics, customized MPC implementation methods [44, 45] can be designed for systems with a small set of variables and constraints, and relatively short planning horizons. Such fast MPC methods utilize the existing problem structure to speed-up computation. Performance guarantees for MPC can be shown under certain conditions [46].

A typical MPC implementation is often modeled such that a central node computes all the control inputs. For networks made up of several nodes, it becomes advantageous, and sometimes even necessary, to perform control using decentralized methods. A few examples of decentralized implementations of MPC can be found in [47–53].

A methodology to decouple a typically centralized MPC approach is described in [54]. Authors focus on the properties of the objective and constraint functions that make up the problem, and then propose methods and prerequisites for enabling decomposition into local, distributed MPC problems. Herein, we use mechanisms from [54] to decompose a global Bayesian constrained optimization problem, designed over a user-defined time horizon for the opportunistic visual sensing application, into local Bayesian constrained optimization problems, that are suitable for distributed model predictive estimation.

2.2 Problem Description and Notation

The operating environment includes N_C cameras placed at known, fixed locations and a time-varying number of targets $N_T(t)$ with independent and unknown trajectories. It is possible that $N_T(t) > N_C$. All cameras have changeable pan ($\rho \in [-180°, 180°]$), tilt ($\tau \in [-90°, 90°]$), and zoom ($\zeta \in [\underline{F}, \overline{F}]$) parameters. We assume the cameras to have calibrated parfocal zoom lenses [55] that maintain focus with changing focal length and have a negligible focus error. Although interesting, we do not consider the effects of calibration of these lenses while designing the system.

In a distributed solution framework, using information up to and including the last imaging time t_k, the i-th camera in the network will be required to optimize its parameters $(\rho_i(t_{k+1}), \tau_i(t_{k+1}), \zeta_i(t_{k+1}))$ for the next imaging instant t_{k+1}, in cooperation with other cameras to maximize an objective function. The parameters of camera C_i are organized into a three vector $\mathbf{a}_i = [\rho_i, \tau_i, \zeta_i]$. Any choice of $\mathbf{a}_i$ yields a field-of-view (FoV_i) for the resulting image. That image may contain multiple targets and each target may be imaged by multiple cameras. The parameters of all cameras are organized into a vector $\mathbf{a} = [\mathbf{a}_1, \ldots, \mathbf{a}_i, \ldots, \mathbf{a}_{N_c}]$. The vector containing all parameter vectors except those of C_i will be denoted by $\mathbf{a}_{-i}$. Additional notation is summarized in Table 2.2.

2.3 System Processes

For solution of the overall problem, in the time interval $t \in (t_k, t_{k+1})$ several processes occur. Based on the previous images up to and including the image at t_k, the target state estimation process provides a prediction of the mean $\hat{\mathbf{x}}^j(k+1)^-$ and covariance matrix $\mathbf{P}^j(k+1)^-$ for all targets (i.e., $j = 1, \ldots, N_T$) at t_{k+1}. Every camera in

Table 2.2: Notation Summary

Parameter	**Variable**
Pan, Tilt, Zoom	(ρ, τ, ζ)
Min. focal length, Max. focal length	$\underline{F}$, $\overline{F}$
No. of Cameras, No. of Targets in region	N_C, $N_T(t)$
i-th camera, j-th target	C_i, T^j
(ρ, τ, ζ) settings for C_i, all cameras except C_i	$\mathbf{a}_i$, $\mathbf{a}_{-i}$
(ρ, τ, ζ) settings for all cameras	$\mathbf{a}$
Dimension of (ρ, τ, ζ) settings for C_i	$\mathbf{a}_i \in \Re^{n_i}$, $n_i = 3$
Dimension of (ρ, τ, ζ) settings for all cameras	$\mathbf{a} \in \Re^n$, $n = 3N_C$
(ρ, τ, ζ) settings sequence for C_i, all cameras except C_i	A_i, A_{-i}
(ρ, τ, ζ) settings sequence for all cameras	A
No. of problem constraints	m
Length of planning horizon	H
Global Bayesian imaging value over all targets	$V_I(\mathbf{a})$
Global Bayesian imaging value over all targets over H	$V_H(A)$
Local Bayesian imaging value over all targets	$V_{I_i}(\mathbf{a}_i)$
Local Bayesian imaging value over all targets over H	$V_{H_i}(A_i)$
Tracking performance vector for all targets	$\mathbf{U}_T(\mathbf{a})$
Tracking threshold vector for all targets	$\bar{\mathbf{T}}$
Bayesian tracking value vector for all targets	$\mathbf{V}_T(\mathbf{a})$
Bayesian tracking constraint vector for all targets over H	$\mathbf{g}_H(A)$
PTZ smoothness constraint vector for all cameras over H	$\mathbf{q}_H(A)$
PTZ smoothness constraint vector for C_i over H	$\mathbf{q}_{H_i}(A_i)$
Achieved global imaging value over all targets	$\bar{V}_I(\mathbf{a})$
Achieved local imaging value over all targets	$\bar{V}_{I_i}(\mathbf{a}_i)$
Achieved tracking value vector for all targets	$\bar{\mathbf{V}}_T(\mathbf{a})$
Lagrange multiplier vector for all targets	$\boldsymbol{\lambda}$
Camera C_i's version of $\boldsymbol{\lambda}$	$\boldsymbol{\lambda}_i$
Lagrange multiplier vector obtained via consensus	$\bar{\boldsymbol{\lambda}}$
Lagrangian constructed for optimization	$L(\boldsymbol{\lambda}, \mathbf{a})$
Weight for importance of imagery of T^j by C_i	w_i^j
Image resolution obtained for T^j by C_i	$r_i^j(\mathbf{a}_i)$
Relative pose quality factor between C_i and T^j	$\alpha_i^j(\mathbf{a}_i)$
State vector for T^j	$\mathbf{x}^j = [\mathbf{p}^j, \mathbf{v}^j]^\top$
State est., state est. covariance for T^j	$\hat{\mathbf{x}}^j$, $\mathbf{P}^j$
Fisher Information Matrix	$\mathbf{J}$
Measurement Vector, Measurement Covariance	$\mathbf{u}$, $\mathbf{C}$
Rotation Matrix from frame a to frame b	${}^b_a\mathbf{R}$
Entity e before, after new measurement	e^-, e^+
Entity e in global frame, frame defined by C_i	${}^g e$, ${}^i e$
Entity e at time-step t_k	$e(k)$
Entity e for target T^j	e^j
Entity e at local or global optimum	e^*

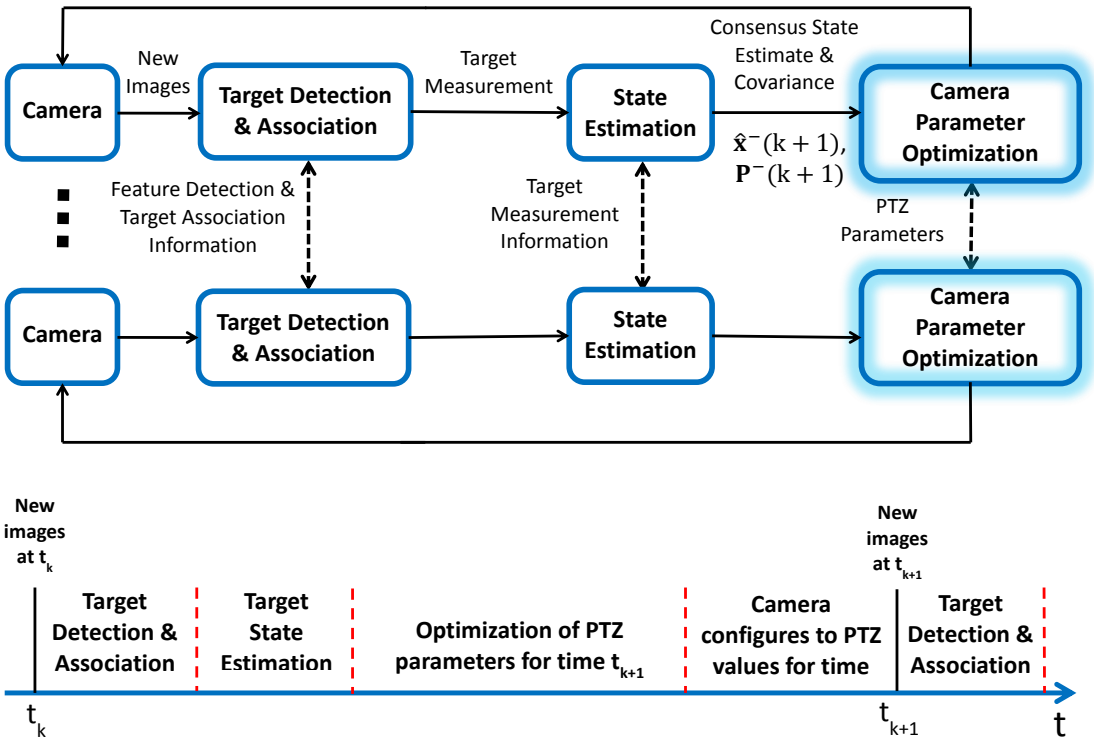


Figure 2.1: System block diagram and event time-line: Note that information exchange shown is only between neighboring cameras. The time-line of procedural events is shown between image sample times.

the network has its own embedded target detection module [8,20,21,56], an Information Weighted *Kalman*-Consensus tracker [32] that provides an estimate of the state of each target, and a distributed camera parameter optimizer.

Remark 1 *By the nature of the visual sensing application, depending on the choice of* $\mathbf{a}_i$*, Camera* C_i *may image multiple targets each at a different resolution. The attained resolution, one quantity affecting image quality, is determined by the choice of* $\mathbf{a}_i$*. In addition, depending on the choice of* $\mathbf{a}$ *any given target may be imaged by multiple cameras, yielding improved tracking accuracy.* $\diamond$

In Fig. 2.1, the top portion illustrates the information flow and the bottom portion illustrates the time sequence of processing. Images acquired at t_k are first processed for feature detection and target association. The target detection module in

each camera takes its raw image and returns the image plane positions of each target recognized in the image. Communication between cameras is allowed to enhance the processes of feature detection and association for target recognition [8]. At the end of this process, each camera has computed the pixel coordinate measurement of each recognized target within its FoV. Assuming that target T^j is within the FoV of camera C_i, this image frame measurement of the pixel location of target T^j by camera C_i valid at time t_k is denoted by ${}^i\mathbf{u}^j(k)$. This measurement is broadcast to neighboring cameras.

The measurements hence acquired are subsequently utilized for distributed state estimation [32, 57, 58]. This process ensures that the state of each target is estimated consistently by each camera in the network. The algorithm at each camera only uses its own imagery and data communicated from its neighbors' on the (possibly) time varying communication graph. Consistency and accuracy of state estimation are prerequisites that enable distributed optimization of the network parameter vector $\mathbf{a}$ for high-resolution image acquisition at t_{k+1}. On completion of the target state estimation process, a prior position estimate ${}^g\hat{\mathbf{p}}^j(k+1)^-$ is available for each target T^j at the future image sampling time t_{k+1}, along with a prior covariance matrix $\mathbf{P}^j(k+1)^-$. To simplify the notation, the time argument t_{k+1} is dropped.

Subsequently, each camera will optimize its parameter settings $\mathbf{a}_i$ using ${}^g\hat{\mathbf{x}}^j(k+1)^-$ and $\mathbf{P}^j(k+1)^-$ as inputs. In the time interval allotted for optimization, each camera will have various opportunities to adjust its parameter settings and communicate its revised settings to the network, such that the entire vector $\mathbf{a}$ converges towards the optimal settings $\mathbf{a}^*$ for the upcoming image at t_{k+1}. This dissertation focuses on distributed camera parameter optimization.

Remark 2 *If cameras take images with a period T_s, the designer could choose to re-optimize the camera parameter settings* $\mathbf{a}$ *after every M-th image, resulting in $t_k = MT_s$. In this design, all M images could still be used for target detection and tracking. In this dissertation, $M = 1$ is chosen. When* $\mathbf{a}$ *is not optimized for each frame, it could be interesting to choose M during operations in response to events (e.g., some target nearing the edge of the FoV, or targets nearing occlusion, etc.).* $\diamond$

Remark 3 *It is useful to compare different possible optimization approaches. In a* centralized optimization *approach, one entity would receive all the required information and adjust the entire vector* $\mathbf{a}$ *to maximize the expected value function. Convergence of centralized optimization methods are very well understood. In distributed approaches, each agent C_i will only adjust the proposed values of its parameters $\mathbf{a}_i$ for the next imaging time. In a* distributed sequential optimization *approach, while one camera C_i is adjusting $\mathbf{a}_i$, all other cameras C_j for $j \neq i$ are idle. Cameras sitting idle potentially save energy at the expense of time to reach convergence. The convergence of such schemes is straightforward to analyze as each camera is solving a much lower dimension optimization problem. The analysis would be similar to that for the centralized case. Distributed sequential optimization is analogous to line search methods. In a* distributed parallel optimization *approach, all cameras adjust their parameters simultaneously. Convergence of this case is more complex, requiring results from optimization and game theory to be applied to cost functions designed to meet certain technical requirements.*

Chapter 3

Background

This chapter briefly reviews concepts of optimization [59] and game theory [3, 60] necessary for the solution methodology proposed herein.

3.1 Centralized Constrained Optimization

Consider a standard convex vector optimization (e.g., maximization) problem with a differentiable primal objective function[1] f_o and differentiable inequality constraints g^j

$$\begin{aligned} &\text{maximize} \quad f_o(\mathbf{a}) \\ &\text{subject to} \quad g^j(\mathbf{a}) \geq 0, \quad j = 1, \cdots, m, \end{aligned} \tag{3.1}$$

where m is the total number of constraints and $\mathbf{a} \in \Re^n$. The *Lagrangian* $L(\boldsymbol{\lambda}, \mathbf{a})$ augments the primal objective function with the constraints

$$L(\boldsymbol{\lambda}, \mathbf{a}) = f_o(\mathbf{a}) + \sum_{j=1}^{m} \lambda^j \; g^j(\mathbf{a}) = f_o(\mathbf{a}) + \boldsymbol{\lambda}^\top \; \mathbf{g}, \tag{3.2}$$

[1] Throughout this article, we use the notation such as $f_o(\mathbf{a})$ for the objective function. The notation $f_o(\mathbf{a} : \mathbf{x}^j, j = 1, \ldots, N_T)$ is more precise in making explicit the fact that the value depends on the target state; however, it is too cumbersome to be effective.

where $\mathbf{g}$ is the vector of constraint functions and $\boldsymbol{\lambda} \in \Re^m$ is the Lagrange multiplier vector with $\lambda^j \geq 0$.

Since the objective and constraint functions $f_o,\ g^1,\ \cdots,\ g^m$ are differentiable, if an optimum $\mathbf{a}^*$ exists, then the Lagrangian $L(\boldsymbol{\lambda}^*, \mathbf{a}^*)$ attains its maximum at the primal-dual pair $(\mathbf{a}^*, \boldsymbol{\lambda}^*)$ that must satisfy the KKT conditions

$$\nabla f_o(\mathbf{a}^*) + \sum_{j=1}^{m} \lambda^{j*}\ \nabla g^j(\mathbf{a}^*) \quad = \quad 0, \tag{3.3}$$

$$g^j(\mathbf{a}^*) \quad \geq \quad 0, \tag{3.4}$$

$$\lambda^{j*} \quad \geq \quad 0, \tag{3.5}$$

$$\lambda^{j*}\ g^j(\mathbf{a}^*) \quad = \quad 0. \tag{3.6}$$

The KKT conditions provide a certificate for optimality.

In a centralized solution approach, the Lagrangian is maximized by search over the parameters $\mathbf{a}$ and $\boldsymbol{\lambda}$. This requires that all data and all parameters are available at a central controller. Although, proofs of optimality are simpler and well known for this centralized approach, for reasons stated in the introduction, we are interested in decentralized solutions.

3.2 Game Theory and Ordinal Potential Functions

For a distributed optimization approach, C_i will only adjust $\mathbf{a}_i$ and $\boldsymbol{\lambda}$. A challenge in formulating a distributed optimization problem is the decoupling of the system objective into local objectives, one for each agent. The game theory literature and the concept of *potentiality* provides guidance for addressing this challenge.

Consider $\mathbf{a} \in S$ and $\mathbf{a}_i,\ \mathbf{b}_i \in S_i$, where $S = S_1 \times \ldots \times S_{N_C}$ is the collection of all possible camera parameter settings in the game G, and S_i is the collection of all possible

camera parameter settings for camera C_i. The sets S and S_i for $i = 1$ to N_C, referred to as action sets within the game theory literature, are compact. Let[2] $\phi_i(\mathbf{a}_i : \mathbf{a}_{-i})$ denote the local objective function of C_i. Game G_p is a *potential game* if there exists a potential function $\phi_p : S \mapsto \Re$ such that $\forall \mathbf{a} \in S$ and $\forall \mathbf{a}_i, \mathbf{b}_i \in S_i$,

$$\phi_p(\mathbf{b}_i, \mathbf{a}_{-i}) - \phi_p(\mathbf{a}_i, \mathbf{a}_{-i}) = \phi_i(\mathbf{b}_i : \mathbf{a}_{-i}) - \phi_i(\mathbf{a}_i : \mathbf{a}_{-i}). \tag{3.7}$$

Game G_o is an *ordinal potential game* if there exists an ordinal potential function $\phi_o : S \mapsto \Re$ such that $\forall \mathbf{a} \in S$ and $\forall \mathbf{a}_i, \mathbf{b}_i \in S_i$,

$$\begin{aligned} \phi_o(\mathbf{b}_i, \mathbf{a}_{-i}) - \phi_o(\mathbf{a}_i, \mathbf{a}_{-i}) &> 0 \\ \Leftrightarrow \phi_i(\mathbf{b}_i : \mathbf{a}_{-i}) - \phi_i(\mathbf{a}_i : \mathbf{a}_{-i}) &> 0. \end{aligned} \tag{3.8}$$

Potential games and ordinal potential games allow the global utility maximum to be achieved by maximization of the local utilities of each camera. When Eqn. (3.8) is satisfied, the local objective functions are said to be *aligned* with the global objective. Given $\phi_o(\mathbf{a})$, if the local utilities are defined as

$$\phi_i(\mathbf{a}_i : \mathbf{a}_{-i}) = \phi_o(\mathbf{a}_i, \mathbf{a}_{-i}), \tag{3.9}$$

then it is straightforward to show that the resulting game is a potential game.

Thus, by defining the global objective function as an ordinal potential function with the individual local camera objectives aligned to it, the game becomes an ordinal potential game. When the set S is compact, and a game has a continuous potential function, then the game has at least one *Nash* Equilibrium. Therefore, given any feasible initial condition, at each step for which one camera increases its own utility, the global objective function increases correspondingly, due to G being a potential game. If ϕ_o is

[2]This notation $\phi_i(\mathbf{a}_i : \mathbf{a}_{-i})$ means that the value of the function ϕ_i may depend on both $\mathbf{a}_i$ and $\mathbf{a}_{-i}$, but that $\mathbf{a}_i$ as treated as an independent variable while $\mathbf{a}_{-i}$ is treated as a constant by C_i.

continuous and S is compact then $\phi_o(\mathbf{a})$ is bounded above; therefore, the optimization converges toward a maxima. At the maxima, no camera can achieve further improvement and thus a Nash equilibrium is reached.

3.3 Distributed Constrained Optimization

For the distributed approach define the local constrained optimization problem for the i-th camera

$$\begin{aligned} &\text{maximize} \quad f_i(\mathbf{a}_i) \\ &\text{subject to} \quad g^j(\mathbf{a}_i : \mathbf{a}_{-i}) \geq 0, \quad j = 1, \cdots, m, \end{aligned} \tag{3.10}$$

where $\mathbf{a}_i \in \Re^{n_i}$, and $\mathbf{a}_{-i} \in \Re^{n-n_i}$. Thus the local Lagrangian can be formulated as,

$$\begin{aligned} L_i(\boldsymbol{\lambda}, \mathbf{a}_i : \mathbf{a}_{-i}) &= f_i(\mathbf{a}_i) + \sum_{j=1}^{m} \lambda^j \; g^j(\mathbf{a}_i : \mathbf{a}_{-i}) \\ &= f_i(\mathbf{a}_i) + \boldsymbol{\lambda}^\top \; \mathbf{g}, \end{aligned} \tag{3.11}$$

where $\mathbf{g}$ is the vector of constraint functions and $\boldsymbol{\lambda} \in \Re^m$ with $\lambda^j \geq 0$.

If we define the global objective function as the sum of local objective functions:

$$f_o(\mathbf{a}) = \sum_{i=1}^{N_C} f_i(\mathbf{a}_i), \tag{3.12}$$

then from Eqns. (3.2), (3.3–3.6), (3.11), and (3.12), $\forall \mathbf{a} \in S$, and $\forall \mathbf{a}_i, \mathbf{b}_i \in S_i$,

$$\begin{aligned} & L(\boldsymbol{\lambda}, \mathbf{b}_i, \mathbf{a}_{-i}) - L(\boldsymbol{\lambda}, \mathbf{a}_i, \mathbf{a}_{-i}) > 0 \\ \Leftrightarrow \quad & L_i(\boldsymbol{\lambda}, \mathbf{b}_i : \mathbf{a}_{-i}) - L_i(\boldsymbol{\lambda}, \mathbf{a}_i : \mathbf{a}_{-i}) > 0, \end{aligned} \tag{3.13}$$

where the objective of each agent is to maximize its local Lagrangian. Therefore, the local and global Lagrangians are aligned, and the global Lagrangian forms an ordinal potential function.

Remark 4 *Note that in a potential game, agent C_i can only choose its own action $\mathbf{a}_i(k+1)$. In doing so, C_i will take into account the proposed actions $\mathbf{a}_{-i}(k+1)$ of all other agents. The actions of all other agents $\mathbf{a}_{-i}(k+1)$ are determined by the other agents. Because C_i is the only agent able to select $\mathbf{a}_i$, consensus between agents is inappropriate for computation of $\mathbf{a}_i$. Instead, a modified flooding approach will be used, see Section 6.2. At the same time, all cameras must collaboratively choose actions and $\boldsymbol{\lambda}$ to ensure that all constraints are satisfied. During distributed optimization each C_i has a local version of the Lagrange multiplier vector, denoted as $\boldsymbol{\lambda}_i$. A consensus algorithm is used to ensure the convergence of $\boldsymbol{\lambda}_i$ to a single value, see Section 6.3.*$\diamond$

After having had a brief overview of the distributed constrained optimization process in this chapter, and having looked at the high-level processes that govern the visual sensing system in Chapter 2, the next chapter will focus on formulating this physical problem into a mathematical problem which the camera network is tasked to solve cooperatively.

Chapter 4

Mathematical Formulation

This chapter focusses on shaping the overall visual sensing problem depicted in Fig. 2.1 into a mathematical problem. The chapter begins with the definition of a motion model for targets, and a measurement model that provides measurements of these targets when imaged by a camera.

The position of the i-th camera in the global frame is indicated by ${}^{g}\mathbf{p}_i$. In addition to the global frame, each camera defines a frame of reference. The position of T^j in the global frame would be indicated as ${}^{g}\mathbf{p}^j$ and in the frame of the i-th camera as ${}^{C_i}\mathbf{p}^j$. The time propagation models [61] for state estimation of T^j are stated below.

4.1 System Model

The continuous-time state space model of target T^j is

$$\dot{\mathbf{x}}^j(t) = \mathbf{F}\ \mathbf{x}^j(t) + \mathbf{G}\ \boldsymbol{\omega}^j(t), \tag{4.1}$$

where $j = 1, \ldots, N_T$ is the target number and $\mathbf{x}^j = [{}^{g}\mathbf{p}^j; {}^{g}\mathbf{v}^j]$ with ${}^{g}\mathbf{p}^j$ and ${}^{g}\mathbf{v}^j$ representing the position and velocity vectors in the global (earth) frame. The process noise vector $\boldsymbol{\omega}^j \in \Re^3$ is assumed to be zero mean Gaussian with power spectral density $\mathbf{Q}$.

The discrete-time equivalent model is

$$\mathbf{x}^j(k+1) = \mathbf{\Phi}\ \mathbf{x}^j(k) + \boldsymbol{\gamma}(k), \tag{4.2}$$

where $\mathbf{\Phi} = e^{\mathbf{F}T}$ is the state transition matrix, $\boldsymbol{\gamma} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q_d})$ is the process noise, and $T = t_{k+1} - t_k$ is the sampling period.

4.1.1 State Estimate Time Propagation

The state estimate and its error covariance matrix are propagated between sampling instants using [61]

$$\hat{\mathbf{x}}^j(k+1)^- = \mathbf{\Phi}\ \hat{\mathbf{x}}^j(k)^+ \tag{4.3}$$

$$\mathbf{P}^j(k+1)^- = \mathbf{\Phi}\ \mathbf{P}^j(k)^+\ \mathbf{\Phi}^\top + \mathbf{Q}_d. \tag{4.4}$$

The covariance matrix $\mathbf{P}^{j-}$ computed using Eqn. (4.4) can be written in block form as

$$\mathbf{P}^{j-} = \begin{bmatrix} \mathbf{P}^{j-}_{\mathbf{pp}} & \mathbf{P}^{j-}_{\mathbf{pv}} \\ \mathbf{P}^{j-}_{\mathbf{vp}} & \mathbf{P}^{j-}_{\mathbf{vv}} \end{bmatrix}, \tag{4.5}$$

where $\mathbf{P}^{j-}_{\mathbf{pp}}$ is the prior, position error-covariance matrix.

4.1.2 Camera Rotation Matrix

In relation with the global frame of reference g, define m_i to be the known mounting angle for camera C_i. The camera rotation matrix ${}^{C_i}_{g}\mathbf{R}(\mathbf{a}_i)$ to compute target coordinates in the i-th camera's frame is hence derived as

$${}^{C_i}_{g}\mathbf{R}(\mathbf{a}_i) = \mathbf{R}(\rho_i)\ \mathbf{R}(\tau_i)\ {}^{C_i}_{g}\mathbf{R}(m_i). \tag{4.6}$$

The matrix ${}^{C_i}_{g}\mathbf{R}(\mathbf{a}_i)$ is a function of the camera mounting angle, the pan angle ρ_i, and the tilt angle τ_i since

$$\mathbf{R}(\rho_i) = \begin{bmatrix} \cos\rho_i & 0 & -\sin\rho_i \\ 0 & 1 & 0 \\ \sin\rho_i & 0 & \cos\rho_i \end{bmatrix}, \tag{4.7}$$

and

$$\mathbf{R}(\tau_i) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\tau_i & \sin\tau_i \\ 0 & -\sin\tau_i & \cos\tau_i \end{bmatrix}. \tag{4.8}$$

4.1.3 Camera Coordinate Transformations

Target T^j's position in the i-th camera's frame, ${}^{c_i}\mathbf{p}^j$, is related to its position in the global frame ${}^{g}\mathbf{p}^j$ by

$${}^{g}\mathbf{p}^j = {}^{g}_{c_i}\mathbf{R}\ {}^{c_i}\mathbf{p}^j + {}^{g}\mathbf{p}_{c_i} \tag{4.9}$$

$${}^{c_i}\mathbf{p}^j = {}^{c_i}_{g}\mathbf{R}\big[{}^{g}\mathbf{p}^j - {}^{g}\mathbf{p}_{c_i}\big]. \tag{4.10}$$

where ${}^{g}\mathbf{p}_{C_i}$ is camera C_i's position in the global frame.

4.2 Measurement Model

Camera measurement models are derived in various references [21,23,62]. This section presents the final results of the standard pin-hole perspective projection camera measurement model. Let the coordinates of target T^j in the i-th camera frame be ${}^{c_i}\mathbf{p}^j = \big[{}^{c_i}x^j, {}^{c_i}y^j, {}^{c_i}z^j\big]^\top$. Assuming that T^j is in the FoV of C_i,

$${}^{i_i}\mathbf{u}^j = \begin{bmatrix} \dfrac{F_i}{s_x}\dfrac{{}^{c_i}x^j}{{}^{c_i}z^j} + o_x \\ \dfrac{F_i}{s_y}\dfrac{{}^{c_i}y^j}{{}^{c_i}z^j} + o_y \end{bmatrix} + {}^{i_i}\boldsymbol{\eta}^j, \tag{4.11}$$

is the standard pin-hole perspective projection camera model for target T^j when measured by camera C_i. In Eqn. (4.11), s_x and s_y give the effective size of a pixel in (m/pixel) measured in the horizontal and vertical directions, respectively; F_i is the focal length setting defined by $\mathbf{a}_i$; the point (o_x, o_y) gives the coordinates of the image plane center in pixels; and the measurement noise ${}^{i_i}\boldsymbol{\eta}^j \sim \mathcal{N}(\mathbf{0}, \mathbf{C}_i^j)$ with $\mathbf{C}_i^j(\mathbf{a}_i) \in \Re^{2\times 2}$.

Given the estimated state $\hat{\mathbf{x}}^{j^-}$ via Eqn. (4.4), and the camera model in Eqn. (4.11), the predicted measurement is

$$ {}^{i_i}\hat{\mathbf{u}}^j = \begin{bmatrix} \frac{F_i}{s_x}\frac{{}^{c_i}\hat{x}^j}{{}^{c_i}\hat{z}^j} + o_x \\ \frac{F_i}{s_y}\frac{{}^{c_i}\hat{y}^j}{{}^{c_i}\hat{z}^j} + o_y \end{bmatrix}. \tag{4.12} $$

The measurement residual ${}^{i_i}\tilde{\mathbf{u}}^j$ is

$$ {}^{i_i}\tilde{\mathbf{u}}^j = {}^{i_i}\mathbf{u}^j - {}^{i_i}\hat{\mathbf{u}}^j. \tag{4.13} $$

Remark 5 *The measurement noise covariance $\mathbf{C}_i^j$ in Eqn. (4.11) is dependent on the parameter settings $\mathbf{a}_i$. This is important because, for example, as the focal length increases, the size of a target in the image increases and the pixel uncertainty of its location changes.*

4.2.1 Observation Matrix

The observation matrix $\mathbf{H}_i^j(\mathbf{a}_i)$ is derived using the linearized relationship given by the first order Taylor series expansion of Eqn. (4.11) around the measurement estimate.

$$ {}^{i_i}\mathbf{u}^j \approx {}^{i_i}\hat{\mathbf{u}}^j + \mathbf{H}_i^j({}^g\mathbf{x}^j - {}^g\hat{\mathbf{x}}^j) \tag{4.14} $$

where $\mathbf{H}_i^j = \left.\frac{\partial {}^{i_i}\mathbf{u}^j}{\partial {}^g\mathbf{x}^j}\right|_{{}^g\hat{\mathbf{x}}^j} \in \Re^{2\times dim(\mathbf{x}^j)}$.

The observation matrix is

$$\mathbf{H}_i^j = \frac{F_i}{(c_i\hat{z}^j)^2}\begin{bmatrix} {}^g\mathbf{N}_1^{j\top} \\ {}^g\mathbf{N}_2^{j\top} \end{bmatrix} = \frac{F_i}{(c_i\hat{z}^j)^2}{}^g\mathbf{N}^{j\top}, \tag{4.15}$$

where the matrix ${}^g\mathbf{N}^{j\top}$ is defined as:

$${}^g\mathbf{N}^{j\top} = \begin{bmatrix} {}^g\mathbf{N}_1^{j\top} \\ {}^g\mathbf{N}_2^{j\top} \end{bmatrix}. \tag{4.16}$$

The symbol $\mathbf{N}^j$ is used as each column is normal to the vector from camera C_i's origin to the j-th target's estimated position $\hat{\mathbf{p}}^j$:

$$\begin{aligned} {}^g\mathbf{N}_1^j &= {}^g_{c_i}\mathbf{R}\ {}^{c_i}\mathbf{N}_1^j & {}^{c_i}\mathbf{N}_1^j &= (s_x)^{-1}\left[{}^{c_i}\hat{z}^j, 0, -{}^{c_i}\hat{x}^j\right]^\top \\ {}^g\mathbf{N}_2^j &= {}^g_{c_i}\mathbf{R}\ {}^{c_i}\mathbf{N}_2^j & {}^{c_i}\mathbf{N}_2^j &= (s_y)^{-1}\left[0, {}^{c_i}\hat{z}^j, -{}^{c_i}\hat{y}^j\right]^\top. \end{aligned}$$

Note that ${}^g\mathbf{N}^j$ is determined by the target location relative to C_i and ${}^g_{c_i}\mathbf{R}$ is a function of $\mathbf{a}_i$. So, $\mathbf{H}_i^j$ is a function of both ${}^g\mathbf{p}^j$ and $\mathbf{a}_i$. When T^j is not in FoV_i, then $\mathbf{H}_i^j = \mathbf{0} \in \Re^{2\times dim(\mathbf{x}^j)}$.

4.2.2 Measurement Update

Using Eqns. (4.13) and (4.15), a measurement update for the state estimates and error covariances for targets in the area can be performed using several distributed Kalman-Consensus Filter approaches [33, 57, 58].

4.3 The Objective Function

This section discusses the imaging value function $V_I(\mathbf{a})$ and the properties it should possess. The objective is to allow distributed optimization over the camera network to select camera parameters $\mathbf{a}$ such that this imaging value is maximized, subject to tracking performance criteria.

4.3.1 Properties

The imaging value function should possess the following properties:

- **Continuously differentiable:** This is necessary for proofs of convergence, and greatly facilitates the numeric optimization process.
- **Increases with image quality:** Herein, image quality is defined by two parameters: the image resolution and the relative pose between the imaging camera and the imaged target.
- **Balanced Risk:** Risk is defined as the probability that the target is outside of the FoV of the cameras that are expected to image it. Risk increases monotonically with zoom ζ, because the ground-plane area within the FoV decreases as ζ increases. Herein, we will address risk by using the expected value of the tracking constraints and the imaging value.

4.3.2 Components

Image resolution $r_i^j(\mathbf{a}_i, {}^g\hat{\mathbf{p}}^j)$, which is a positive real number, will be quantified by the number of pixels occupied by target T^j on camera C_i's image plane. Given ${}^g\hat{\mathbf{p}}^j$, the resolution increases monotonically with zoom ζ of the imaging camera.

Relative pose between camera C_i and target T^j is quantified by the pose quality factor $\alpha_i^j(\mathbf{a}_i, {}^g\hat{\mathbf{x}}^j)$. Given ${}^g\hat{\mathbf{x}}^j$, the scalar α_i^j is large when it is likely that target T^j is facing camera C_i and at the center of C_i's FoV. The quantities α_i^j and r_i^j are defined in the simulation example (see Chapter 8).

Risk is defined as the probability that the target is outside of the FoV of the cameras that are expected to image it. Risk increases monotonically with zoom ζ, because the area under the FoV decreases as ζ increases. Herein, we will address risk by using the expected value of the tracking constraints and the imaging value.

To understand the drawbacks of using a non-Bayesian value function for the visual sensing problem, it is informative to briefly consider the simple case where $N_T = 1$ and $\alpha_i^1 > 0$ for $i = 1, \ldots, N_C$. For this case, if risk was neglected and an imaging value V_I was defined with the properties mentioned above, then each camera would maximize its focal length and select its pan and tilt parameters to center on the expected target location. If instead, the value accounted appropriately for risk, then one or more camera might significantly increase its zoom parameter, while at least one of the remaining cameras would use lower zoom parameters, to decrease the tracking risk due to the uncertainty in the estimated target position. The camera at the highest zoom setting would be the one at the best aspect angle.

The next chapter will focus on a Bayesian formulation of the imaging value function and the tracking performance constraints based on properties and parameters defined in this chapter.

Chapter 5

PTZ Optimization Methodology

As discussed relative to Fig. 2.1, for every image acquired, the network of cameras detect features and associate them with targets to provide measurements. These target measurements are used in the Information Consensus Filter [32] that estimates the j-th target's state $\hat{\mathbf{x}}^{j-}$ with covariance matrices $\mathbf{P}^{j-}$, at each camera, using the model in Section 4.1.1. The selection of the camera parameters $\mathbf{a}$ will affect the accuracy of the state estimator at future time instants due to the quality of the images acquired.

The objective is to obtain high-res imagery while tracking all targets at all times to a specified accuracy $\bar{\mathbf{T}}$. Building on the importance of the Bayesian framework discussed in Chapter 4, this chapter starts with specification of the global objective and constraints that are subsequently decoupled into local objectives for each camera.

The global objective function for the constrained optimization problem is designed as a Bayesian imaging value function that accounts for the risk in imaging the target. Risk will be formulated using the Fisher information matrix defined as $\mathbf{J}^{j-}(k+1) = \left(\mathbf{P}^{j-}(k+1)\right)^{-1}$.

5.1 The Global Bayesian Imaging Value Function

We define the global Bayesian image value function as

$$\begin{aligned} V_I(\mathbf{a}) &= \sum_{i=1}^{N_C} \sum_{j=1}^{N_T(t)} E\left\langle V_{I_i}^j\left(\mathbf{a}_i,\ {}^g\mathbf{p}^j\right)\right\rangle \\ &= \sum_{i=1}^{N_C} \int_{\mathrm{FoV}_i} \left(\sum_{j=1}^{N_T(t)} V_{I_i}^j(\mathbf{a}_i,\ \mathbf{z})\ p_{\mathbf{p}^j}(\mathbf{z}) \right) d\mathbf{z}, \end{aligned} \tag{5.1}$$

where an example $V_{I_i}^j$ is defined in Eqn. (8.5) of Chapter 8. Given the assumptions herein, the probability distribution $p_{\mathbf{p}^j}(\mathbf{z})$ of the position of T^j in the global frame at the next imaging instant is the Normal distribution $\mathcal{N}({}^g\hat{\mathbf{p}}^{j-}, \mathbf{P}_{\mathbf{pp}}^{j-})$. The dummy variable $\mathbf{z}$ representing target position is used for integration over the ground plane, where the region of integration is the i-th camera's FoV.

Each camera integrates over its own FoV. The integral of image quality over FoV_i as a function of probability weighted target position yields the Bayesian value function, which provides the desired tradeoff between image quality and risk.

5.2 The Bayesian Performance Constraints

The target tracking performance criteria will be defined as a function of the posterior *Fisher Information Matrix* $\mathbf{J}^{j+}\left(\mathbf{a} : {}^g\hat{\mathbf{p}}^{j-}, \mathbf{P}_{\mathbf{pp}}^{j-}\right)$.

The posterior Fisher information matrix $\mathbf{J}^{j+}$ for T^j in block form is

$$\mathbf{J}^{j+} = \begin{bmatrix} \mathbf{J}_{\mathbf{pp}}^{j+} & \mathbf{J}_{\mathbf{pv}}^{j+} \\ \mathbf{J}_{\mathbf{vp}}^{j+} & \mathbf{J}_{\mathbf{vv}}^{j+} \end{bmatrix}, \tag{5.2}$$

where $\mathbf{J}_{\mathbf{pp}}^{j+}$ represents the posterior position information matrix, which is

$$\mathbf{J}_{\mathbf{pp}}^{j+} = \mathbf{J}_{\mathbf{pp}}^{j-} + \sum_{i=1}^{N_C} \mathbf{H}_i^{j\ \top}\ \mathbf{C}_i^{j\ -1}\ \mathbf{H}_i^j. \tag{5.3}$$

As was shown in Section 4.2.1, $\mathbf{H}_i^j$ and $\mathbf{C}_i^j$ are functions of $\mathbf{a}_i$ and $\mathbf{H}_i^j$ is a function of the target position. Therefore, $\mathbf{J}_{\mathbf{pp}}^{j+}$ depends on $\mathbf{a}$ and on the target position. Computation of the expected tracking accuracy should account for this variation and for the probability that $T^j \in FoV_i$.

5.2.1 Bayesian Tracking Value

We define a vector $\mathbf{U}_T^j(\mathbf{a} : {}^g\hat{\mathbf{p}}^{j-}, \mathbf{P}_{\mathbf{pp}}^{j-})$ as a measure of tracking performance for each target in the area. For the purpose of this dissertation, we choose

$$\mathbf{U}_T^j(\mathbf{a}) = \text{diag}\left(\mathbf{J}_{\mathbf{pp}}^{j+}\right). \tag{5.4}$$

Remark 6 *An implementation in [21] utilized a scalar tracking performance utility, which was defined as the trace of the expected posterior information* $\mathbf{J}_{\mathbf{pp}}^{j+}$. *Other definitions of* $\mathbf{U}_T^j$ *are of interest as future research.* $\diamond$

Because the quantity $\mathbf{U}_T^j(\mathbf{a})$ depends on whether T^j is within the FoV of each camera that is expected to image it, we define the global Bayesian tracking value vector $\mathbf{V}_T^j(\mathbf{a})$ as the expected value of the tracking performance vector $\mathbf{U}_T^j(\mathbf{a})$ over the position of target T^j computed across all the camera's FoVs:

$$\mathbf{V}_T^j\big(\mathbf{a}\big) = E_{\mathbf{p}^j}\left\langle \mathbf{U}_T^j(\mathbf{a}) \right\rangle = \int \left(\mathbf{U}_T^j(\mathbf{a})\ p_{\mathbf{p}^j}\left(\mathbf{z}\right)\right) d\mathbf{z}, \tag{5.5}$$

where all variables are as defined in Eqn. (5.1) and the summation over all cameras is accounted for already in eqn. (5.3), which also accounts for prior information.

5.2.2 Tracking Constraint

Each target's tracking constraint is

$$\mathbf{V}_T^j(\mathbf{a}) \succeq \bar{\mathbf{T}}^j, \tag{5.6}$$

where $\bar{\mathbf{T}}^j$ is the user specified lower bound on the tracking information about target T^j. Due to the reciprocal relation between (scalar) information and covariance, the reciprocal of $\bar{\mathbf{T}}^j$ is the upper bound on the covariance of target T^j's state estimate. The tracking performance threshold $\bar{\mathbf{T}}^j$ is hence measured in m^{-2}. The notation '$\succeq$' in Eqn. (5.6) indicates a per-element vector inequality. Stacking the Bayesian tracking value vectors for each target, we obtain

$$\mathbf{V}_T(\mathbf{a}) = \left[\mathbf{V}_T^1, \cdots, \mathbf{V}_T^j, \cdots, \mathbf{V}_T^{N_T}\right]^\top, \tag{5.7}$$

and rewrite Eqn. (5.6) for all targets presently in the area as:

$$\mathbf{V}_T(\mathbf{a}) \succeq \bar{\mathbf{T}}, \tag{5.8}$$

where $\mathbf{V}_T(\mathbf{a}), \bar{\mathbf{T}}, \mathbf{0} \in \Re^m$ with $m = N_T(t)\dim({}^g\mathbf{p}^j)$. Eqn. (5.8) is the global tracking constraint.

5.2.3 Global Problem Summary

The constrained global imaging value maximization problem can be written as

$$\begin{aligned} &\text{maximize} \quad V_I\left(\mathbf{a} : \ {}^g\hat{\mathbf{p}}^{j-}, \ \mathbf{P}_{\mathbf{pp}}^{j-}\right) \\ &\text{subject to} \quad \left[\mathbf{V}_T\left(\mathbf{a} : \ {}^g\hat{\mathbf{p}}^{j-}, \ \mathbf{P}_{\mathbf{pp}}^{j-}\right) - \bar{\mathbf{T}}\right] \succeq \mathbf{0}. \end{aligned} \tag{5.9}$$

The global *Lagrangian* $L(\boldsymbol{\lambda}, \mathbf{a})$ is

$$L(\boldsymbol{\lambda}, \mathbf{a}) = V_I\left(\mathbf{a}\right) + \boldsymbol{\lambda}^\top \left[\mathbf{V}_T(\mathbf{a}) - \bar{\mathbf{T}}\right], \tag{5.10}$$

where $L : (\boldsymbol{\lambda}, \mathbf{a}) \mapsto \Re$, and $\boldsymbol{\lambda} \in \Re^m$ is the Lagrange multiplier vector. Thus, to find the optimal primal-dual pair of solutions $(\mathbf{a}^*, \boldsymbol{\lambda}^*)$ through a central controller, the global unconstrained problem given by the Lagrangian in Eqn. (5.10) would be solved.

5.3 Decoupling the Global Problem

Due to the desired distributed nature of our solution, we need to decompose the global problem into smaller local problems that are solvable by each camera.

In our problem formulation, we allow camera C_i to optimize only its own camera parameter settings $\mathbf{a}_i$. Using this system restriction, we define the local Bayesian imaging value function for C_i as

$$V_{I_i}(\mathbf{a}_i) = \int_{FoV_i} \sum_{j=1}^{N_T(t)} \left(w_i^j(t)\ V_{I_i}^j(\mathbf{a}_i,\ \mathbf{z})\ p_{\mathbf{p}^j}(\mathbf{z}) \right) d\mathbf{z}. \tag{5.11}$$

Define $\mathbf{V}_{T_i}(\mathbf{a}_i) = \mathbf{V}_T(\mathbf{a}_i : \mathbf{a}_{-i})$. This notation concisely indicates that C_i can only alter $\mathbf{a}_i$, where for the purpose of its local optimization $\mathbf{a}_{-i}$ is fixed. Note that $\mathbf{V}_T(\mathbf{a}_i : \mathbf{a}_{-i})$ is distinct from $\mathbf{V}_T(\mathbf{a}_i, \mathbf{a}_{-i}) = \mathbf{V}_T(\mathbf{a})$ and that $\max_{\mathbf{a}_i \in S_i} \mathbf{V}_{T_i}(\mathbf{a}_i) \leq \max_{\mathbf{a} \in S} \mathbf{V}_T(\mathbf{a})$. Each agent will have the constraint $\mathbf{V}_T(\mathbf{a}_i : \mathbf{a}_{-i}) \succeq \bar{\mathbf{T}}$. While C_i is changing $\mathbf{a}_i$, the other agents are simultaneously changing their sub vectors of $\mathbf{a}_{-i}$ and all agents are broadcasting their current locally optimal values through the network. Thus the tracking constraint for camera C_i is

$$\mathbf{V}_{T_i}(\mathbf{a}_i) \succeq \bar{\mathbf{T}}. \tag{5.12}$$

Note that,

$$\mathbf{V}_{T_i}(\mathbf{a}_i) \succeq \bar{\mathbf{T}} \quad \Leftrightarrow \quad \mathbf{V}_{T_i}^j(\mathbf{a}_i) \succeq \bar{\mathbf{T}}^j \text{ for } j = 1, \ldots, N_T(t).$$

The *Fisher* Information given in Eqn. (5.3) can be reorganized as:

$$\mathbf{J}_{\mathbf{pp}}^{j+} = \left[\mathbf{J}_{\mathbf{pp}}^{j-} + \mathbf{H}_{-i}^{j^\top} \left(\mathbf{C}_{-i}^j \right)^{-1} \mathbf{H}_{-i}^j \right] + \mathbf{H}_i^{j^\top} \left(\mathbf{C}_i^j \right)^{-1} \mathbf{H}_i^j.$$

For the process of C_i optimizing its parameter vector $\mathbf{a}_i$, the contribution from prior information and all other cameras (term in brackets) is independent of $\mathbf{a}_i$ and considered by C_i to be constant and known. The term $\left[\mathbf{H}_{-i}^{j^\top}\left(\mathbf{C}_{-i}^{j}\right)^{-1}\mathbf{H}_{-i}^{j}\right]$ is computed from $\mathbf{a}_{-i}$ which will be available through the distributed optimization process discussed in Chapter 6.

Thus from Eqns. (5.3-5.5), we can write

$$E_{\mathbf{p}^j}\left\langle \operatorname{diag}\left(\mathbf{J}_i^j\right)\right\rangle \succeq \bar{\mathbf{T}} - E_{\mathbf{p}^j}\left\langle \operatorname{diag}\left(\mathbf{J}_{\mathbf{pp}}^{j-} + \mathbf{J}_{-i}^{j}\right)\right\rangle, \tag{5.13}$$

where $\mathbf{J}_i^j = \mathbf{H}_i^{j^\top}\left(\mathbf{C}_i^j\right)^{-1}\mathbf{H}_i^j$ and $\mathbf{J}_{-i}^j = \mathbf{H}_{-i}^{j^\top}\left(\mathbf{C}_{-i}^j\right)^{-1}\mathbf{H}_{-i}^j$. The right hand side of this inequality represents, for the current proposed settings of the other cameras $\mathbf{a}_{-i}$, the expected improvement in tracking accuracy required from C_i for imaging T^j to have a feasible global solution. Targets for which the right-hand side of Eqn. (5.13) is negative can be removed from the set of tracking constraints for C_i.

5.3.1 Local Problem Summary

From Eqns. (5.11) and (5.12), the local imaging value maximization problem can be written as

$$\begin{aligned} \text{maximize} \quad & V_{I_i}\left(\mathbf{a}_i :\ {}^{g}\hat{\mathbf{p}}^{j-},\ \mathbf{P}_{\mathbf{pp}}^{j-}\right) \\ \text{subject to} \quad & \mathbf{V}_{T_i}\left(\mathbf{a}_i :\ {}^{g}\hat{\mathbf{p}}^{j-},\ \mathbf{P}_{\mathbf{pp}}^{j-}\right) \succeq \bar{\mathbf{T}}. \end{aligned} \tag{5.14}$$

The local *Lagrangian* $L_i(\boldsymbol{\lambda}_i, \mathbf{a}_i)$ is

$$L_i\left(\boldsymbol{\lambda}_i, \mathbf{a}_i\right) \;=\; V_{I_i}(\mathbf{a}_i) + \boldsymbol{\lambda}_i^\top\left[\mathbf{V}_{T_i}(\mathbf{a}_i) - \bar{\mathbf{T}}\right]. \tag{5.15}$$

Thus, for camera C_i to find its local optimal primal-dual pair of solutions $(\mathbf{a}_i^*, \boldsymbol{\lambda}_i^*)$, C_i will maximize the local unconstrained *Lagrangian* given in Eqn. (5.15).

In this approach, all cameras in the network optimize simultaneously. The subscript i on $\boldsymbol{\lambda}_i$ in Eqn. (5.15) indicates that the *Lagrange* multiplier vector picked by camera C_i to solve the problem is a local variable and may not be globally the same throughout the network. In order to overcome this predicament, cameras in the network employ a variant of the algorithm described in [63, 64] to perform dynamic average consensus over the local *Lagrange* multiplier vectors. This results in a consensus-step after each optimization-step[1]. The algorithm is explained in detail in Chapter 6.

5.3.2 Lagrangian as an Ordinal Potential Function

For the problem stated in Eqn. (5.9), note that the global objective of the multi-camera network defined in Eqn. (5.1) is the sum over the local objectives defined in Eqn. (5.11)

$$V_I(\mathbf{a}) \quad = \quad \sum_{i=1}^{N_C} V_{I_i}(\mathbf{a}_i). \tag{5.16}$$

At each optimization step κ, the i-th camera adjusts $\boldsymbol{\lambda}_i(\kappa)$ and $\mathbf{a}_i(\kappa)$, leaving $\mathbf{a}_{-i}(\kappa)$ fixed, to solve the problem in Eqn. (5.14) with $V_{I_i}(\mathbf{a}_i)$ defined in Eqn. (5.11). Dynamic average consensus over $\boldsymbol{\lambda}_i$ between optimization steps forces each agent's local value toward a non-negative consensus agreement vector $\bar{\boldsymbol{\lambda}}(\kappa) = \frac{1}{N_c}\sum_i \boldsymbol{\lambda}_i(\kappa)$. Convergence of the dynamic game is assured when the local *Lagrangians* $L_i(\boldsymbol{\lambda}_i, \mathbf{a}_i)$, and the global *Lagrangian* $L(\boldsymbol{\lambda}, \mathbf{a})$ form an ordinal potential game.

[1]Every optimization and consensus step may contain multiple iterations. In fact, they may proceed simultaneously.

From Eqns. (5.10), (5.15),and (5.16), $\forall \boldsymbol{\lambda}_b,\ \boldsymbol{\lambda}_a \succeq \mathbf{0}$, let $\tilde{L} = L(\boldsymbol{\lambda}_b,\ \mathbf{b}_i,\ \mathbf{a}_{-i}) - L(\boldsymbol{\lambda}_a,\ \mathbf{a}_i,\ \mathbf{a}_{-i})$. Thus,

$$\begin{aligned}
\tilde{L} &= \sum_{i=1}^{N_C} V_{I_i}(\mathbf{b}_i) + \boldsymbol{\lambda}_b^\top \left[\mathbf{V}_T\left(\mathbf{b}_i,\ \mathbf{a}_{-i}\right) - \bar{\mathbf{T}}\right] \\
&\quad - \sum_{i=1}^{N_C} V_{I_i}(\mathbf{a}_i) - \boldsymbol{\lambda}_b^\top \left[\mathbf{V}_T\left(\mathbf{a}_i,\ \mathbf{a}_{-i}\right) - \bar{\mathbf{T}}\right] \\
&= V_{I_i}(\mathbf{b}_i) + \sum_{l\neq i} V_{I_l}(\mathbf{a}_l) + \boldsymbol{\lambda}_b^\top \left[\mathbf{V}_T\left(\mathbf{b}_i,\ \mathbf{a}_{-i}\right) - \bar{\mathbf{T}}\right] \\
&\quad - V_{I_i}(\mathbf{a}_i) - \sum_{l\neq i} V_{I_l}(\mathbf{a}_l) - \boldsymbol{\lambda}_a^\top \left[\mathbf{V}_T\left(\mathbf{a}_i,\ \mathbf{a}_{-i}\right) - \bar{\mathbf{T}}\right] \\
&= L_i(\boldsymbol{\lambda}_b,\ \mathbf{b}_i :\ \mathbf{a}_{-i}) - L_i(\boldsymbol{\lambda}_a,\ \mathbf{a}_i :\ \mathbf{a}_{-i}).
\end{aligned}$$

Hence, $\forall \mathbf{a} \in S$, $\forall \mathbf{a}_i, \mathbf{b}_i \in S_i$, and $\forall \boldsymbol{\lambda}_b,\ \boldsymbol{\lambda}_a \succeq \mathbf{0}$,

$$\begin{aligned}
& L(\boldsymbol{\lambda}_b,\ \mathbf{b}_i,\ \mathbf{a}_{-i}) - L(\boldsymbol{\lambda}_a,\ \mathbf{a}_i,\ \mathbf{a}_{-i}) > 0 \\
\Leftrightarrow\ & L_i(\boldsymbol{\lambda}_b,\ \mathbf{b}_i :\ \mathbf{a}_{-i}) - L_i(\boldsymbol{\lambda}_a,\ \mathbf{a}_i :\ \mathbf{a}_{-i}) > 0.
\end{aligned}$$

Therefore, as explained in Sections 3.2 and 3.3, Eqns. (5.10) and (5.15) form an ordinal potential game.

5.4 Challenges

Fig. 5.1 depicts the appearance of an example Bayesian Imaging Value function for a scenario with $N_T = 2$ targets in an area monitored by $N_C = 1$ camera. As shown in the figure, when $N_T > 1$, the summation of the per target expected imaging value across the targets for any camera will typically yield a multimodal (i.e., nonconvex) objective function. Given the expected target positions and their respective distributions, for a constant tilt angle τ, the plot shows how the value of the objective function changes versus zoom ($\zeta \in [\underline{\zeta}, \overline{\zeta}]$), and pan angle ($\rho \in [\underline{\rho}, \overline{\rho}]$), It can be seen that the multi-modal nature of the function is exaggerated for higher values of ζ. Thus, the possibility of

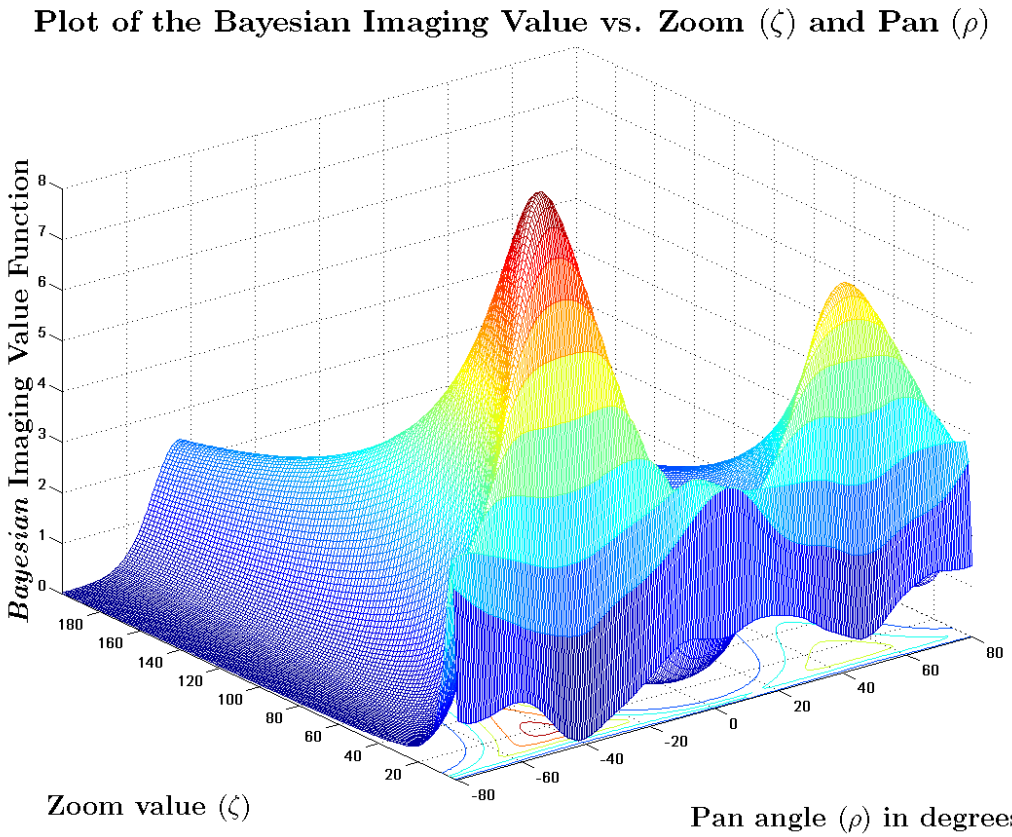


Figure 5.1: Non-convexity of the Bayesian imaging value function: The figure is a plot of an example Bayesian imaging value function that highlights the multimodal nature of the function for scenarios where $N_T > 1$.

multiple targets makes the visual sensing problem inherently non-convex and provides challenges in achieving optimal solutions. Non-convexity is further discussed in later sections.

Chapter 6

Distributed Optimization

The distributed optimization process can be broken down into three separate steps, where κ denotes the iteration counter:

1. **Camera Parameter Optimization:** Each camera C_i computes $(\mathbf{a}_i, \boldsymbol{\lambda}_i)$ to increase $L_i\left(\boldsymbol{\lambda}_i, \mathbf{a}_i : \mathbf{a}_{-i}\right)$ while holding $\mathbf{a}_{-i}$ constant. It then communicates the newly computed local primal-dual pair estimates $(i, \kappa, \mathbf{a}_i(\kappa), \boldsymbol{\lambda}_i(\kappa))$ and new portions of $\mathbf{a}_{-i}$ to its neighbors $\mathcal{N}_i$.

2. **Camera Parameter Replacement:** Each camera C_n that is a neighbor of C_i (i.e. $C_n \in \mathcal{N}_i$) receives $(i, \kappa, \mathbf{a}_i(\kappa), \boldsymbol{\lambda}_i(\kappa), \mathbf{a}_{-i})$. It replaces its previous value of $(\mathbf{a}_i, \mathbf{a}_{-i})$ using the rules of replacement described in Section 6.2.

3. **Consensus on Lagrange Multipliers:** C_i performs dynamic average consensus on its local *Lagrange* multiplier vector $\boldsymbol{\lambda}_i(\kappa)$ and the *Lagrange* multiplier vectors received from cameras in $\mathcal{N}_i$ to converge towards a consensus *Lagrange* multiplier vector $\bar{\boldsymbol{\lambda}}$, using the *Lagrange* multiplier update law in Eqn. (6.2), defined in Section 6.3.

This distributed optimization process is then iterated over κ until a stopping criteria is achieved.

Since the optimization problem described by Eqn. (5.14) is non-convex, any solution found may only be locally optimal. It is assumed that all agents start with identical values of $\mathbf{a}(\kappa)$ for $\kappa = 0$ and that $\mathbf{a}(0)$ is not on the separatrix dividing the domain of attraction of one local optimum from another.

6.1 Connectivity, Communication, and Consensus

The approach requires the following standard assumptions on the camera communication graph.

Assumption 1 *(Connectivity) The camera communication graph is undirected, and* connected*, i.e. there exists at least one communication path from each agent to every other agent in the network.*

Remark 7 *In [64], each agent changes $\boldsymbol{\lambda}$ and the entire vector $\mathbf{a}$ while computing a dual solution, then using consensus on both $\mathbf{a}$ and $\boldsymbol{\lambda}$. Herein, agent C_i only optimizes $\boldsymbol{\lambda}$ and $\mathbf{a}_i$, which is a subvector of $\mathbf{a}$. When the subvector $\mathbf{a}_i$ is broadcast to the neighbors of C_i, they pass it to their neighbors. Each agent receiving a newer value of $\mathbf{a}_i$ replaces their older value. Thus, for the approach herein, each camera C_i need only perform dynamic average consensus on $\boldsymbol{\lambda}_i$ and the set of* Lagrange *multiplier vectors $\{\boldsymbol{\lambda}_n\}$ for $C_n \in \mathcal{N}_i$. Connectivity ensures that the changes to $\mathbf{a}_i$ and $\boldsymbol{\lambda}_i$ by each C_i eventually affect all agents in the network. The convergence of consensus is asymptotic, but becomes trivial for strictly feasible solutions, which have $\boldsymbol{\lambda} = 0$. The effects of a change in any $\mathbf{a}_i$ are fully distributed throughout the network in a finite number of steps, which is less than the diameter of the network.*$\diamond$

Assumption 2 *(Weights Rule) There exists a scalar* $\beta > 0$ *such that for each* $i \in [1, N_c]$, $\omega_{ii}(\kappa) \geq \beta$, *and* $\omega_{in}(\kappa) \in [\beta, 1]$ *for* $C_n \in \mathcal{N}_i$. *If cameras* C_i *and* C_n *are not directly connected, then* $\omega_{in}(\kappa) = 0$.

Assumption 3 *(Double Stochasticity) Let* $B_i = C_i \bigcup \mathcal{N}_i$, *and* $\sum_{l \in B_i} \omega_{il}(\kappa) = 1$ *and* $\sum_{i \in B_i} \omega_{il}(\kappa) = 1$.

Assumption 2 ensures that all cameras are *influential* [63] while performing consensus on the local *Lagrange* multiplier vectors, and Assumption 3 ensures that all cameras asymptotically converge to a consensus *Lagrange* multiplier vector $\bar{\boldsymbol{\lambda}} \succeq \mathbf{0}$ [65].

6.2 Camera Parameter Replacement Rule

We use a variant of the flooding algorithm [66] to propagate the local variables through the network of cameras. After C_i computes $\mathbf{a}_i(\kappa)$, it delivers the information $\{i, \kappa, \mathbf{a}_i(\kappa), \boldsymbol{\lambda}_i(\kappa)\}$ to its neighbors $\mathcal{N}_i$, and will rebroadcast to its neighbors any *updated PTZ information*, $\{l, \mathbf{a}_l, \kappa_l\}$ for $l \neq i$, that it received since the last broadcast. Using rebroadcast, each agent's parameter updates travel throughout a connected network exactly one time. For the l-th subvector in $C_i's$ version of $\mathbf{a}$, C_i has a value $\mathbf{a}_l(\kappa_l)$ and a time-stamp κ_l both computed by C_l, even if C_i and C_l are not neighbors. Because the network may contain loops, C_i may receive information about other cameras via multiple paths. C_i will replace its l-th subvector with the received information only if the time-stamp in $\{l, \mathbf{a}_l, \kappa_l\}$ is more recent than the time stamp corresponding to the value it is currently using. Otherwise, the message is discarded without rebroadcast.

6.3 Distributed Lagrangian Consensus

At iteration κ camera C_i receives the set of *Lagrange* multiplier vectors $\{\boldsymbol{\lambda}_n(\kappa)\}$ for $C_n \in \mathcal{N}_i$. It also has its local copy of $\mathbf{a}$. Its local computations must jointly optimize $\mathbf{a}_i$ and $\boldsymbol{\lambda}$ (given $\mathbf{a}_{-i}$) while also converging toward agreement across the network on the value of $\boldsymbol{\lambda}$. This section describes dynamic average consensus on the local versions of *Lagrange* multiplier vectors. Following the notation in [64], we refer to this as a distributed *Lagrangian* consensus algorithm.

Camera C_i iteratively optimizes using the update law [63]:

$$\mathbf{a}_i(\kappa+1) = -s(\kappa)\ \mathbf{D}_{\mathbf{a}_i}(\kappa) \tag{6.1}$$

$$\boldsymbol{\lambda}_i(\kappa+1) = \boldsymbol{\nu}_{\boldsymbol{\lambda}_i}(\kappa) - s(\kappa)\ \mathbf{D}_{\boldsymbol{\lambda}_i}(\kappa), \tag{6.2}$$

where the scalar $s(\kappa) > 0$ is the step-size,

$$\mathbf{D}_{\mathbf{a}_i} = \nabla_{\mathbf{a}_i}\ L_i\left(\boldsymbol{\lambda}_i(\kappa),\ \mathbf{a}_i(\kappa) :\ \mathbf{a}_{-i}(\kappa)\right),$$

and

$$\begin{aligned}\mathbf{D}_{\boldsymbol{\lambda}_i} &= \nabla_{\boldsymbol{\lambda}_i}\ L_i\left(\boldsymbol{\lambda}_i(\kappa),\ \mathbf{a}_i(\kappa) :\ \mathbf{a}_{-i}(\kappa)\right) \\ &= \left[\mathbf{V}_{T_i}\left(\mathbf{a}_i^*\right) - \bar{\mathbf{T}}\right].\end{aligned}$$

The first term in Eqn. (6.2) is the consensus term, which is a convex combination of $\boldsymbol{\lambda}_i(\kappa)$ and $\{\boldsymbol{\lambda}_n(\kappa)\}$:

$$\boldsymbol{\nu}_{\boldsymbol{\lambda}_i}(\kappa) = \sum_{l \in B_i}\ \omega_{il}(\kappa)\ \boldsymbol{\lambda}_l(\kappa), \tag{6.3}$$

which always yields $\boldsymbol{\nu}_{\boldsymbol{\lambda}_i}(\kappa) \in \Re^m$ as a non-negative vector. The second term is the gradient descent term, which adjusts $\boldsymbol{\lambda}_i$ in a coordinated fashion with the change in $\mathbf{a}_i$ to converge toward an optimal and feasible solution relative to the local optimization

problem. The step-size $s(\kappa) > 0$ can be adjusted to maintain component-wise non-negativity of $\boldsymbol{\lambda}_i$.

With Assumptions 1, 2 and 3 , it is shown in [63–65] that for all $i = 1, \ldots, N_C$, there exist $\bar{\boldsymbol{\lambda}} \succeq \mathbf{0}$ such that

$$\lim_{\kappa \to \infty} \|\bar{\boldsymbol{\lambda}} - \boldsymbol{\lambda}_i(\kappa)\| = 0. \tag{6.4}$$

6.4 Certificate for Optimality

For the unconstrained maximization problem defined by Eqn. (5.15) for each agent, the optimal primal-dual pair $(\mathbf{a}_i^*, \boldsymbol{\lambda}_i^*)$ must satisfy the *KKT* conditions:

$$\nabla V_{I_i}(\mathbf{a}_i^*) + \left[\nabla \mathbf{V}_{T_i}(\mathbf{a}_i^*)\right]^\top \boldsymbol{\lambda}_i^* = 0, \tag{6.5}$$

$$\mathbf{V}_{T_i}(\mathbf{a}_i^*) - \bar{\mathbf{T}} \succeq \mathbf{0}, \tag{6.6}$$

$$\boldsymbol{\lambda}_i^* \succeq \mathbf{0}, \tag{6.7}$$

$$\boldsymbol{\lambda}_i^{*\top} \left[\mathbf{V}_{T_i}(\mathbf{a}_i^*) - \bar{\mathbf{T}}\right] = 0, \tag{6.8}$$

which provide a certificate of optimality at each agent.

All cameras optimize in parallel. Camera C_i broadcasts $\mathbf{a}_i^*$ and $\boldsymbol{\lambda}_i^*$, and new portions of $\mathbf{a}_{-i}$, to its neighbors who propagate them through the network. While C_i is locally optimizing its settings, it is accounting for an updated $\boldsymbol{\lambda}_i$, and for each target, the prior information $\mathbf{J}^{j-}$ and expected new information based on the currently best settings of all the other cameras $\mathbf{a}_{-i}$.

Optimization stops when either an optimum is achieved, a user-defined stopping condition is met, or the time interval allotted for optimization elapses (see Fig 2.1). The solution approach described in [21] optimized a weighted combination of tracking and imaging; whether or not an optimum was achieved, there was no guarantee that

the tracking specification was achieved. For the approach herein, the KKT conditions described in Eqns. (6.5 - 6.8) provide a certificate on optimality and feasibility (i.e., satisfaction of the tracking specification). Numeric algorithms to solve the constrained optimization problem defined in Eqn. (5.14), to which the KKT conditions of Eqns. (6.5–6.8) apply, first find a feasible solution, then search within the feasible set for the optimal feasible solution. Thus, when the time interval allotted for optimization elapses, even if the solution is sub-optimal, the solution obtained is guaranteed to be feasible. This results in all targets being tracked to the specified tracking accuracy at all times, while procuring high-resolution imagery when opportunity arises. After optimization, the cameras physically alter their settings to the optimal values in readiness for upcoming images at t_{k+1}.

Thus, by using the replacement step in Section 6.2 and the *Lagrange* multiplier update law from Eqn. (6.2), at each consensus iteration κ, every camera maintains an estimate of the primal-dual pairs of all cameras.

Chapter 7

Bayesian Visual Sensing with Planning

While obtaining high-resolution imagery, the camera network may select settings such that they are discontinuous across time. This could result in the optimal camera parameters at time-step t_k being vastly different from those at time-step t_{k-1}. Apart from leading to mechanical wear of cameras, such discontinuities in the PTZ settings of cameras also lead to motion blur in images, and could generally provide images that are not well suited for analysis. In this chapter, a method to enforce temporal continuity in the pan, tilt, zoom (PTZ) settings of smart vision sensors within a distributed network set up to perform optimal high resolution image capture and aid target state estimation is proposed.

When designed as an optimization problem to be solved point-wise at a regular time-step [21, 23], the solution does not consider the future effects of the computed control actions, since the optimization problem is solved without the aid of a planning procedure. To address this, the problem is designed as a constrained optimization

problem over a moving planning horizon H, subject to constraints on target tracking performance and on the smoothness of PTZ parameters. This enables the immediate actions of the camera control module to consider future effects of these actions. The objective function entails the per camera per target imaging quality over horizon H. The solution approach is designed using a *Bayesian* framework within a game-theoretic paradigm. The Bayesian formulation facilitates automatic trading-off of objective maximization versus the risk of losing track of any target, while the game-theoretic design allows the global problem to be decoupled into local problems, one at each camera. The feasible set for PTZ parameters is defined by constraints on target tracking performance and the PTZ parameter smoothness constraints. Cameras alter their own PTZ sequences by using information received from other cameras in the network, and then broadcast the updated sequences to their neighbors. The challenge is to define algorithms to autonomously perform high-res image capture while adhering to target tracking and PTZ smoothness constraints, within a distributed framework.

To motivate the benefits of incorporating a planning procedure to the PTZ optimization method, consider the scenario depicted in Fig. 7.1. Five target trajectories (shown in different colors) are being tracked within a fully connected network of four PTZ cameras. We consider five time points for the sake of clarity. Let $t = 3$ be the current measurement time instant, and that measurements for all five targets are obtained at $t = 3$. Target positions at the current time are represented by human markers that are color-coordinated with the color of the respective targets trajectory. Every target trajectory shows target positions (marked by light gray human markers) at which past measurements were obtained obtained. For a planning horizon of 2 time points forward from the current measurement instant, the predicted target positions at $t = 4$ and $t = 5$

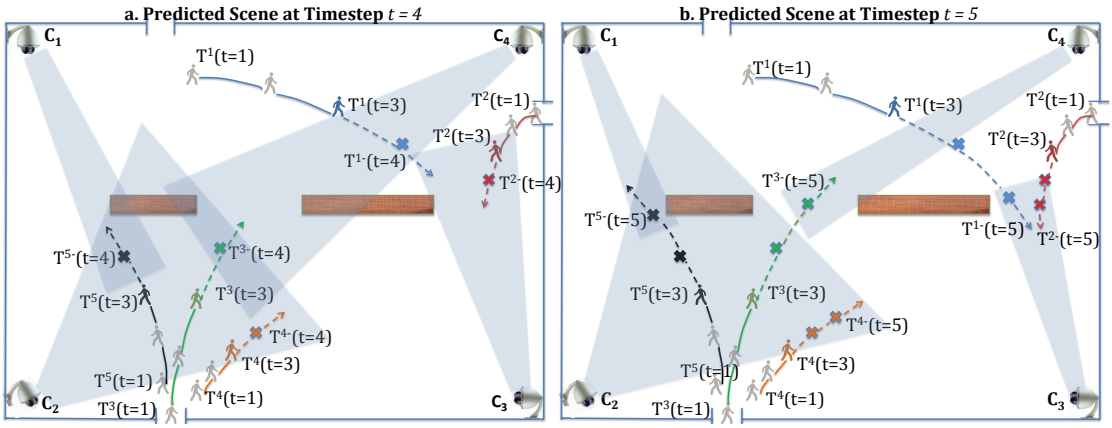


Figure 7.1: Schematic of a probable scenario for distributed camera control via planning: Fig. a (left) shows the planned optimal camera FoVs for $t = 4$, and Fig. b (right) shows the planned optimal camera FoVs for $t = 5$. The human markers indicating target positions for the present instant are color-coordinated with the color of the respective targets trajectory. Target positions for measurements obtained in the past are indicated by light gray human markers. The FoVs formed by the PTZ parameters of each camera are represented by light blue shaded polygons. Ground plane positions where cameras expect measurements to be obtained are shown by crosses.

are indicated by crosses (where cameras expect measurements to be obtained). Fig. 7.1a shows the planned optimal camera FoVs (polygons shaded in light blue) for $t = 4$, and Fig. 7.1 shows the planned optimal camera FoVs for $t = 5$.

Note that the cameras try to maximize the expected imaging value over the horizon using the target motion model by planning ahead accordingly. For example, at $t = 3$, target T^2 was invisible to camera C_3 due to occlusion. Also, at $t = 5$, both T_1 and T_2 are expected to come very close to one another (see Fig. 7.1b), and are expected to be visible to camera C_3. This should potentially provide C_3 with an opportunity to achieve a high imaging value (e.g. high-res facial capture) for both T^1 and T^2 at timestep $t = 5$. Using a planning horizon should also allow camera C_4 to plan on increasing its zoom value to image target T^3 for time $t = 5$. Similarly, camera C_1 can plan to image target T^5. Since three of the four cameras zoom into three different targets, camera C_2 plans

to zoom out gradually to enlarge its FoV and thereby to minimize the risk of losing track on targets. Such an approach should also result in the camera FoVs changing smoothly, and thus providing a video stream that is better suited for image analysis.

7.1 Moving Horizon

For a user defined planning horizon H, the j-th target's state estimate and error covariance matrix are propagated between sampling instants within the horizon $h = 1, \ldots, H$ using

$$\hat{\mathbf{x}}^j(k+h)^- = \mathbf{\Phi}^h \; \hat{\mathbf{x}}^j(k)^+, \tag{7.1}$$

$$\begin{aligned} \mathbf{P}^j(k+h)^- &= \mathbf{\Phi}^h \; \mathbf{P}^j(k)^+ \; {\mathbf{\Phi}^h}^\top \\ &+ \sum_{h=1}^{H} \mathbf{\Phi}^{h-1} \; \mathbf{Q}_d \; {\mathbf{\Phi}^{h-1}}^\top. \end{aligned} \tag{7.2}$$

At time-point t_{k+1}, for horizon H, define

$$\hat{X}^j(k+1)^- = \{\hat{\mathbf{x}}^j(k+1)^-, \ldots, \; \hat{\mathbf{x}}^j(k+H)^-\} \tag{7.3}$$

to be the predicted trajectory for target T^j. Also define

$$P^j(k+1)^- = \{\mathbf{P}^j(k+1)^-, \ldots, \; \mathbf{P}^j(k+H)^-\} \tag{7.4}$$

to be the set of covariance matrices associated with the corresponding state estimates in $\hat{X}^j(k+1)^-$. For $j = 1, \ldots, N_T$, define

$$\hat{X}^-_{(k+1)} = \{\hat{X}^j(k+1)^-\}, \tag{7.5}$$

and

$$P^-_{(k+1)} = \{P^j(k+1)^-\}, \tag{7.6}$$

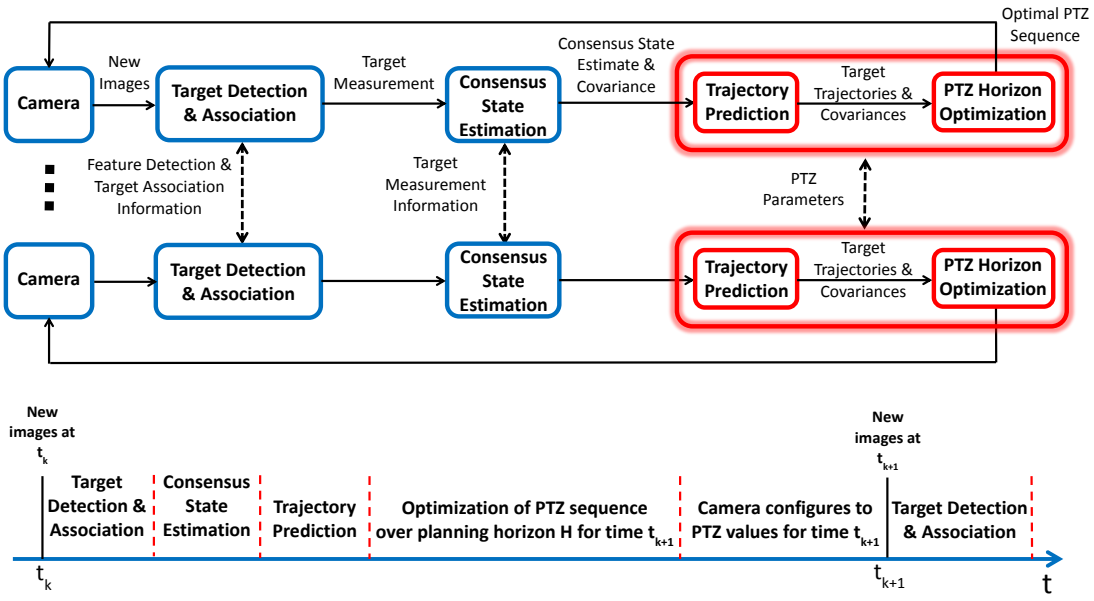


Figure 7.2: System block diagram and event time-line: Note that information exchange shown is only between neighboring cameras. The time-line of procedural events is shown between image sample times.

to be the set of target trajectories and their associated covariance matrices, for all targets, respectively. For time-point t_{k+1}, and optimization horizon H, define

$$A_{(k+1)} = \{\mathbf{a}(k+1), \ldots, \mathbf{a}(k+H)\}, \tag{7.7}$$

to be a sequence of camera network PTZ parameters. Given predictions $\hat{X}^{-}_{(k+1)}$ and $P^{-}_{(k+1)}$, a global objective maximization problem over a moving horizon H is stated as

$$\begin{aligned} \max_{A_{(k+1)}} \quad & f\left(A_{(k+1)} : \hat{X}^{-}_{(k+1)}, P^{-}_{(k+1)}\right) \\ \text{subject to} \quad & g\left(A_{(k+1)} : \hat{X}^{-}_{(k+1)}, P^{-}_{(k+1)}\right) \geq 0, \\ & \left|\mathbf{a}(k+h) - \mathbf{a}(k+h-1)\right| \leq \epsilon, \\ & \text{for } h = 1, \ldots, H, \end{aligned} \tag{7.8}$$

where the objective f is a measure of the imaging value expected to be obtained for imaging of targets in the area, and g is a measure of the expected target tracking performance. For the above problem, the first constraint governs the performance of

the state estimator for target trajectories defined by Eqns. (7.1-7.2), and the second constraint enforces smoothness over the computed PTZ sequence. The functions f and g will be defined in later sections.

To compute the optimal solution $A^*_{(k+1)}$, the optimization problem given by Eqn. (7.8) is evaluated with respect to the distribution of random target trajectories X, parameterized by mean $\hat{X}^-_{(k+1)}$ and covariance $P^-_{(k+1)}$ over horizon H. In comparison, a general Model Predictive Control problem [4] is stated as

$$\begin{aligned} \max_{\bar{A}_{(k+1)}, \bar{X}_{(k+1)}} \quad & f\left(\bar{A}_{(k+1)},\ \bar{X}_{(k+1)}\right) \\ \text{subject to} \quad & \bar{\mathbf{x}}^j(k+h) = \ell\left(\bar{\mathbf{x}}^j(k+h-1),\ \bar{\mathbf{a}}(k+h)\right), \\ & g\left(\bar{A}_{(k+1)},\ \bar{X}_{(k+1)}\right) \geq 0, \\ & \left|\bar{\mathbf{a}}(k+h) - \bar{\mathbf{a}}(k+h-1)\right| \leq \epsilon, \\ & \text{for } h = 1, \ldots, H, \end{aligned} \tag{7.9}$$

where for $j = 1, \ldots, N_T$, the set $\bar{X}_{(k+1)} = \{\bar{X}^j(k+1)\}$, with $\bar{X}^j(k+1) = \{\bar{\mathbf{x}}^j(k+1), \ldots, \bar{\mathbf{x}}^j(k+H)\}$, and the set $\bar{A}_{(k+1)} = \{\bar{\mathbf{a}}(k+1), \ldots, \bar{\mathbf{a}}(k+H)\}$. The first constraint of the problem in Eqn. (7.9) indicates that the evolution of the state $\bar{\mathbf{x}}^j(k+h)$ is dependent on the computed control input $\bar{A}_{(k+1)}$ via the function ℓ.

The problem setup in Eqn. (7.8) is distinct from that in Eqn. (7.9) since, for a visual sensing application, the target state trajectory is independent of the control action $A_{(k+1)}$. Hence, the optimization problem in Eqn. (7.8) does not require constraints on the state variables $\hat{X}_{(k+1)}$.

Remark 8 *After solving the optimization problem for time-step t_{k+1}, the controller directs the network of cameras to reconfigure to PTZ setting $\mathbf{a}(k+1)^*$ from the optimal sequence $A^*_{(k+1)}$. For the purpose of implementation, assuming a horizon of $H >= 2$, the PTZ setting $\mathbf{a}(k+2)^*$ from the sequence $A^*_{(k+1)}$ could be used as an initial value for the optimization problem to be solved for t_{k+2}.*$\diamond$

7.2 Moving Horizon Global Bayesian Imaging Value

Given the Bayesian imaging value from Eqn. (5.1), the expected global Bayesian imaging value at time[1] t_l over planning horizon H is defined as

$$V_H\left(A_{(l)}\right) = \sum_{h=l}^{l+H-1} V_I\left(\mathbf{a}(h)\right), \tag{7.10}$$

where A is as defined in Eqn. (7.7). In Eqn. (7.10), the PTZ sequence $A_{(l)}$ is the optimization variable, and is computed starting from time t_l over horizon H. From definitions in Section 7.1, and from Eqns. (5.1), (7.3) and (7.10), it should be noted that V_H is also dependent on but does not affect the predicted target trajectories $\hat{X}^-_{(l)}$, and their associated covariance matrices $P^-_{(l)}$.

7.3 Moving Horizon Tracking Performance Constraints

As explained in Section 5.2, the tracking performance constraints enforce tracking of all targets to a specified accuracy $\bar{\mathbf{T}}$. For PTZ settings $\mathbf{a}(l)$ at time t_l, and given the expected distribution of target T^j's position $\mathcal{N}({}^g\hat{\mathbf{p}}^j(l)^-, \mathbf{P}^j_{\mathbf{pp}}(l)^-)$, the tracking constraintfor target T^j is a function of the expected posterior *Fisher* information matrix $\mathbf{J}^j\left(\mathbf{a}(l) : {}^g\hat{\mathbf{p}}^j(l), \mathbf{P}^j_{\mathbf{pp}}(l)\right)^+$.

[1] For ease of notation, we define time $t_l = t_{k+1}$ wherever necessary.

7.3.1 Fisher Information Over Horizon H

The Fisher information matrix $\mathbf{J}^j$ for T^j in block form is given by Eqn. (5.2), from which the posterior position information matrix at time t_l is $\mathbf{J}^{j+}_{\mathbf{pp}}$

$$\mathbf{J}^j_{\mathbf{pp}}(l)^+ = \mathbf{J}^j_{\mathbf{pp}}(l)^- + \sum_{i=1}^{N_C} \mathbf{H}^j_i(l)^\top \mathbf{C}^j_i(l)^{-1} \mathbf{H}^j_i(l). \tag{7.11}$$

As is shown in Section 4.2.1, matrices $\mathbf{H}^j_i$ and $\mathbf{C}^j_i$ are functions of $\mathbf{a}_i(l)$ and target position ${}^g\mathbf{p}^j(l)$. Therefore, $\mathbf{J}^j_{\mathbf{pp}}(l)^+$ depends on $\mathbf{a}(l)$ and on the distribution of target positions at t_l. For time t_l and horizon H, define

$$J^j(l)^+ = \left\{ \mathbf{J}^j_{\mathbf{pp}}(l)^+, \ldots, \mathbf{J}^j_{\mathbf{pp}}(l+H-1)^+ \right\} \tag{7.12}$$

to be the set of posterior position information matrices associated with target T^j. For $j = 1$ to N_T, define

$$J^+_{(l)} = \{ J^j(l)^+ \} \tag{7.13}$$

to be the set of posterior position information matrices for all targets. Note that the tracking constraints will have to account appropriately for the probability that $T^j \in FoV_i$, over horizon H, and hence will be designed within the Bayesian framework.

7.3.2 Tracking Constraints Over Horizon H

As shown in Eqn. (5.6), the tracking constraint for each target for time t_l is

$$\mathbf{V}^j_T\big(\mathbf{a}(l)\big) \succeq \bar{\mathbf{T}}^j, \tag{7.14}$$

where $\bar{\mathbf{T}}^j$ is a constant tracking accuracy parameter specified by the user and the notation '$\succeq$' indicates a per-element vector inequality.

For $j = 1, \ldots, N_T$, stacking the Bayesian tracking value vectors for each target, we obtain

$$\mathbf{V}_T\left(\mathbf{a}(l)\right) = \left[\mathbf{V}_T^1\left(\mathbf{a}(l)\right), \cdots, \mathbf{V}_T^{N_T}\left(\mathbf{a}(l)\right)\right]^\top, \tag{7.15}$$

and rewrite Eqn. (7.14) for all targets presently in the area as

$$\mathbf{g}\left(\mathbf{a}(l)\right) = \mathbf{V}_T(\mathbf{a}(l)) - \bar{\mathbf{T}} \succeq \mathbf{0}, \tag{7.16}$$

where $\mathbf{g}\left(\mathbf{a}(l)\right)$, $\mathbf{V}_T(\mathbf{a}(l))$, $\bar{\mathbf{T}}$, $\mathbf{0} \in \Re^m$ with $m = \dim({}^g\mathbf{p}^j)N_T(l)$. Eqn. (7.16) is the global tracking constraint for time t_l.

Given the Bayesian tracking constraint in Eqn. (7.16), for time t_l and horizon H, define

$$\mathbf{g}_H\left(A_{(l)}\right) = \left[\mathbf{g}\big(\mathbf{a}(l)\big)^\top, \ldots, \mathbf{g}\big(\mathbf{a}(l+H-1)\big)^\top\right]^\top \tag{7.17}$$

to be the vector of stacked global Bayesian tracking constraints computed for imaging instant t_l, over optimization horizon H, where $\mathbf{g}_H \in \Re^{mH}$. From definitions in Section 7.1, and from Eqn. (5.5), note that $\mathbf{g}_H$ is dependent on $\hat{X}_{(l)}^-$ and $P_{(l)}^-$.

7.3.3 Smoothness Constraints Over Horizon H

To enforce a measure of temporal continuity in PTZ space, the optimization problem includes the constraint

$$\mathbf{q}\big(\mathbf{a}(l)\big) = \boldsymbol{\epsilon} - |\mathbf{a}(l) - \mathbf{a}(l-1)| \succeq \mathbf{0}. \tag{7.18}$$

The vector $\boldsymbol{\epsilon} = \left[\boldsymbol{\epsilon}_1^\top, \ldots, \boldsymbol{\epsilon}_i^\top, \ldots, \boldsymbol{\epsilon}_{N_C}^\top\right]^\top \in \Re^{3N_C}$, where for $i = 1, \ldots, N_C$, the user defined upper bound on the permissible change in camera C_i's PTZ parameters between successive time-points is $\boldsymbol{\epsilon}_i = [\bar{\rho}_i, \bar{\tau}_i, \bar{\zeta}_i]^\top \in \Re^3$.

Given the smoothness constraint in Eqn. (7.18), for time t_l and horizon H, define

$$\mathbf{q}_H\left(A_{(l)}\right) = \left[\mathbf{q}\big(\mathbf{a}(l)\big)^\top, \ldots, \mathbf{q}\big(\mathbf{a}(l+H-1)\big)^\top\right]^\top, \tag{7.19}$$

to be the vector of concatenated global smoothness constraints computed for imaging instant t_l, over optimization horizon H, where $\mathbf{q}_H \in \Re^{nH}$, where $n = 3N_C$.

7.3.4 Global Moving Horizon Problem Summary

The constrained global imaging value maximization problem for imaging instant t_l over planning horizon H can be written as

$$\begin{aligned} \text{maximize} \quad & V_H\left(A_{(l)} : \hat{X}^-_{(l)}, P^-_{(l)}\right) \\ \text{subject to} \quad & \mathbf{g}_H\left(A_{(l)} : \hat{X}^-_{(l)}, P^-_{(l)}\right) \succeq \mathbf{0} \in \Re^{mH}, \\ & \mathbf{q}_H\left(A_{(l)}\right) \succeq \mathbf{0} \in \Re^{nH}. \end{aligned} \tag{7.20}$$

The global *Lagrangian* $L(\boldsymbol{\lambda}, \boldsymbol{\mu}, A)$ is

$$L(\boldsymbol{\lambda}, \boldsymbol{\mu}, A) = V_H(A) + \boldsymbol{\lambda}^\top \mathbf{g}_H(A) + \boldsymbol{\mu}^\top \mathbf{q}_H(A), \tag{7.21}$$

where $L : (\boldsymbol{\lambda}, \boldsymbol{\mu}, A) \mapsto \Re$, and $\boldsymbol{\lambda} \in \Re^{mH}$, $\boldsymbol{\mu} \in \Re^{nH}$ are the Lagrange multiplier vectors. Thus, to find the optimal primal-dual set of solutions $(A^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$ through a central controller, the global unconstrained problem given by the Lagrangian in Eqn. (7.21) would be solved.

7.4 Decoupling the Moving Horizon Global Problem

Due to the desired distributed nature of our solution, the global problem in Eqn. (7.20) is to be decoupled into smaller local problems to be solved by each camera. Given Eqn. (5.11), we define the local Bayesian imaging value for time t_l over planning horizon H as

$$V_{H_i}\left(A_{i(l)}\right) = \sum_{h=l}^{l+H-1} V_{I_i}\left(\mathbf{a}_i(h)\right), \tag{7.22}$$

where camera C_i's PTZ sequence

$$A_{i(l)} = \left\{\mathbf{a}_i(l), \ldots, \mathbf{a}_i(l+H-1)\right\}, \tag{7.23}$$

is the optimization variable in Eqn. (7.22). From definitions in Section 7.1, and from Eqns. (5.11) and (7.22), V_{H_i} is also dependent on $\hat{X}^-_{(l)}$ and $P^-_{(l)}$.

7.4.1 Bayesian Tracking Constraints

Define $\mathbf{g}_{H_i}(A_i) = \mathbf{g}_H(A_i : A_{-i})$. This notation concisely indicates that C_i can only alter A_i, where for the purpose of its local optimization A_{-i} is fixed. Thus we define the horizon tracking constraint for camera C_i as

$$\mathbf{g}_{H_i}(A_i) \succeq \mathbf{0}. \tag{7.24}$$

Note that while C_i is changing A_i, it utilizes the current locally optimal PTZ sequences A_{-i}, which it has received through its neighbors in the network. The constraint $\mathbf{g}_{H_i}(A_i) = \mathbf{g}_H(A_i : A_{-i})$ is distinct from $\mathbf{g}_H(A_i, A_{-i}) = \mathbf{g}_H(A)$, and $\max_{A_i \in S_i} \mathbf{g}_{H_i}(A_i) \leq \max_{A \in S} \mathbf{g}_H(A)$, where the set $S = S_1 \times \ldots \times S_{N_C}$ is the collection of all possible PTZ sequences, and set S_i is the collection of all possible PTZ sequences for camera C_i. Also, from Eqns. (7.14) - (7.17), for $h = 1$ to H, and $j = 1$ to $N_T(l)$,

$$\mathbf{g}_{H_i}\left(A_{i(l)}\right) \succeq \mathbf{0} \quad \Leftrightarrow \quad \mathbf{V}^j_{T_i}\left(\mathbf{a}_i(h)\right) \succeq \bar{\mathbf{T}}^j.$$

The matrix from Eqn. (5.3) can be reorganized as

$$\begin{aligned}\mathbf{J}^j_{\mathbf{pp}}(l)^+ &= \left[\mathbf{J}^j_{\mathbf{pp}}(l)^- + \mathbf{H}^j_{-i}(l)^\top\ \mathbf{C}^j_{-i}(l)^{-1}\ \mathbf{H}^j_{-i}(l)\right] \\ &+ \mathbf{H}^j_i(l)^\top\ \mathbf{C}^j_i(l)^{-1}\ \mathbf{H}^j_i(l). \end{aligned} \tag{7.25}$$

The term $\left[\mathbf{H}^{j^\top}_{-i}\ (\mathbf{C}^j_{-i})^{-1}\ \mathbf{H}^j_{-i}\right]$ is computed from $A_{-i}(l)$, which are available at C_i through frequent communication between cameras. Hence, camera C_i regards the bracketed term in Eqn. (7.25) to be constant and known.

7.4.2 Local Smoothness Constraints

Define vector

$$\mathbf{q}_i\big(\mathbf{a}_i(l)\big) = \boldsymbol{\epsilon}_i - |\mathbf{a}_i(l) - \mathbf{a}_i(l-1)| \succeq \mathbf{0}, \tag{7.26}$$

where $\boldsymbol{\epsilon}_i = [\bar{\rho}_i, \bar{\tau}_i, \bar{\zeta}_i]^\top \in \Re^3$ is as defined in Eqn. (7.18). From Eqn. (7.26), for time t_l and horizon H, define

$$\mathbf{q}_{H_i}\left(A_{i(l)}\right) = \left[\mathbf{q}_i\big(\mathbf{a}_i(l)\big)^\top, \ldots, \mathbf{q}_i\big(\mathbf{a}_i(l+H-1)\big)^\top\right]^\top, \tag{7.27}$$

to be the vector of concatenated local smoothness constraints computed for future imaging instant t_l, over optimization horizon H. The horizon smoothness constraint $\mathbf{q}_{H_i} \in \Re^{n_i H}$, where $n_i = 3$. From Eqns. (7.19) and (7.27), note that

$$\mathbf{q}_H\left(A_{(l)}\right) = \left[\mathbf{q}_{H_1}\left(A_{1(l)}\right)^\top, \ldots, \mathbf{q}_{H_{N_C}}\left(A_{N_C(l)}\right)^\top\right]^\top. \tag{7.28}$$

The constraints for camera C_i only include $\mathbf{q}_{H_i}\left(A_i\right)$.

7.4.3 Local Moving Horizon Problem Summary

From Eqns. (7.22), (7.24), and (7.27), the local constrained imaging value maximization problem, for imaging instant t_l, over planning horizon H is

$$\begin{aligned} \text{maximize} \quad & V_{H_i}\left(A_{i(k)} : \hat{X}^-_{(l)}, P^-_{(l)}\right) \\ \text{subject to} \quad & \mathbf{g}_{H_i}\left(A_{i(l)} : \hat{X}^-_{(l)}, P^-_{(l)}\right) \succeq \mathbf{0} \in \Re^{mH}, \\ & \mathbf{q}_{H_i}\left(A_i(l)\right) \succeq \mathbf{0} \in \Re^{n_i H}. \end{aligned} \tag{7.29}$$

The local *Lagrangian*[2] $L_i(\boldsymbol{\lambda}_i, \boldsymbol{\mu}_i, A_i)$ is

$$L_i(\boldsymbol{\lambda}_i, \boldsymbol{\mu}_i, A_i) = V_{H_i}(A_i) + \boldsymbol{\lambda}_i^\top \mathbf{g}_{H_i}(A_i) + \boldsymbol{\mu}_i^\top \mathbf{q}_{H_i}(A_i), \tag{7.30}$$

where $L_i : (\boldsymbol{\lambda}_i, \boldsymbol{\mu}_i, A_i) \mapsto \Re$, and $\boldsymbol{\lambda}_i \in \Re^{mH}$ and $\boldsymbol{\mu}_i \in \Re^{n_i H}$ are the Lagrange multiplier vectors. Thus, to find its local optimal primal-dual set of solutions $(A_i^*, \boldsymbol{\lambda}_i^*, \boldsymbol{\mu}_i^*)$, camera C_i solves the local unconstrained problem given by the Lagrangian in Eqn. (7.30).

7.4.4 Lagrange Multiplier Consensus

In the proposed approach, we choose that cameras in the network optimize simultaneously. As explained in Chapter 6, the Lagrange multiplier vector $\boldsymbol{\lambda}_i$ picked by camera C_i to solve the problem is a local variable and may not be globally the same throughout the network, and the subscript i on $\boldsymbol{\lambda}_i$ in Eqn. (7.30) explicitly indicates this.

In order to overcome this predicament, cameras in the network employ the Lagrange multiplier consensus algorithm explained in Section 6.3 to perform dynamic average consensus over the local *Lagrange* multiplier vectors.

[2]Since $\mathbf{g}_{H_i}\left(A_{i(l)}\right) = \mathbf{g}_H\left(A_{i(l)} : A_{-i(l)}\right)$, the i-th camera also uses the PTZ sequences A_{-i} while computing its local *Lagrangian* L_i. The dependence of L_i on A_{-i} is dropped for ease of notation, and is stated only if needed for explanation.

7.4.5 Lagrangian as an Ordinal Potential Function

For the problem stated in Eqn. (7.20), note that the global objective of the multi-camera network defined in Eqn. (7.10) is the sum over the local objectives defined in Eqn. (7.22):

$$V_H(A) \;=\; \sum_{i=1}^{N_C} V_{H_i}(A_i). \tag{7.31}$$

At each optimization step κ, camera C_i uses PTZ sequences $A_{-i(\kappa)}$ to adjust $\boldsymbol{\lambda}_{i(\kappa)}$, $\boldsymbol{\mu}_{i(\kappa)}$, and $A_{i(\kappa)}$, leaving $A_{-i(\kappa)}$ fixed, to solve the problem in Eqn. (7.29) with $V_{H_i}(A_i)$ as defined in Eqn. (7.22). Convergence of the dynamic game is assured when the local Lagrangian $L_i(\boldsymbol{\lambda}_i, \boldsymbol{\mu}_i, A_i)$, and the global Lagrangian $L(\boldsymbol{\lambda}, \boldsymbol{\mu}, A)$ form an *ordinal potential game* [60].

To prove ordinal potentiality of the global Lagrangian L, $\forall \boldsymbol{\lambda}_B,\ \boldsymbol{\lambda}_A \succeq \mathbf{0} \in \Re^{mH}$, $\forall \boldsymbol{\mu}_B,\ \boldsymbol{\mu}_A \succeq \mathbf{0} \in \Re^{nH}$, and $\forall \boldsymbol{\mu}_{B_i},\ \boldsymbol{\mu}_{A_i} \succeq \mathbf{0} \in \Re^{n_i H}$, define

$$\tilde{L} = L(\boldsymbol{\lambda}_B, \boldsymbol{\mu}_B, B_i, A_{-i}) - L(\boldsymbol{\lambda}_A, \boldsymbol{\mu}_A, A_i, A_{-i}). \tag{7.32}$$

Thus, from Eqns. (7.21), (7.30), (7.31), and (7.32),

$$\begin{aligned}
\tilde{L} \;=\;& V_H\left(B_i,\ A_{-i}\right) + \boldsymbol{\lambda}_B^\top\ \mathbf{g}_H\left(B_i,\ A_{-i}\right) \\
&+ \boldsymbol{\mu}_B^\top\ \mathbf{q}_H\left(B_i,\ A_{-i}\right) - V_H\left(A_i,\ A_{-i}\right) \\
&- \boldsymbol{\lambda}_A^\top\ \mathbf{g}_H\left(A_i,\ A_{-i}\right) - \boldsymbol{\mu}_A^\top \mathbf{q}_H\left(A_i,\ A_{-i}\right) \\
=\;& \sum_{i=1}^{N_C} V_{H_i}\left(B_i\right) + \boldsymbol{\lambda}_B^\top\ \mathbf{g}_H\left(B_i,\ A_{-i}\right) \\
&+ \sum_{i=1}^{N_C} \boldsymbol{\mu}_{B_i}^\top\ \mathbf{q}_{H_i}\left(B_i\right) - \sum_{i=1}^{N_C} V_{H_i}\big(A_i\big) \\
&- \boldsymbol{\lambda}_A^\top\ \mathbf{g}_H\left(A_i,\ A_{-i}\right) + \sum_{i=1}^{N_C} \boldsymbol{\mu}_{A_i}^\top\ \mathbf{q}_{H_i}\left(A_i\right).
\end{aligned} \tag{7.33}$$

From the definition in Eqn. (7.31), the terms $\sum_{i=1}^{N_C} V_{H_i}(B_i)$ and $\sum_{i=1}^{N_C} V_{H_i}(A_i)$ from Eqn. (7.33) can be rewritten as

$$\sum_{i=1}^{N_C} V_{H_i}\left(B_i\right) = V_{H_i}\big(B_i\big) + \sum_{j \neq i} V_{H_j}\big(A_j\big), \tag{7.34}$$

and

$$\sum_{i=1}^{N_C} V_{H_i}\left(A_i\right) = V_{H_i}(A_i) + \sum_{j \neq i} V_{H_j}(A_j), \tag{7.35}$$

respectively. Similarly, from Eqns. (7.21), (7.28), and (7.30) the terms $\sum_{i=1}^{N_C} \boldsymbol{\mu}_{B_i}^{\top}\ \mathbf{q}_{H_i}(A_i)$ and $\sum_{i=1}^{N_C} \boldsymbol{\mu}_{A_i}^{\top}\ \mathbf{q}_{H_i}(A_i)$ from Eqn. (7.33) can be rewritten as

$$\sum_{i=1}^{N_C} \boldsymbol{\mu}_{B_i}^{\top}\ \mathbf{q}_{H_i}\big(B_i\big) = \boldsymbol{\mu}_{B_i}^{\top}\ \mathbf{q}_{H_i}\left(B_i\right) + \sum_{j \neq i} \boldsymbol{\mu}_{A_j}^{\top}\ \mathbf{q}_{H_j}\left(A_j\right), \tag{7.36}$$

and

$$\sum_{i=1}^{N_C} \boldsymbol{\mu}_{A_i}^{\top}\ \mathbf{q}_{H_i}\big(A_i\big) = \boldsymbol{\mu}_{B_i}^{\top}\ \mathbf{q}_{H_i}\left(B_i\right) + \sum_{j \neq i} \boldsymbol{\mu}_{A_j}^{\top}\ \mathbf{q}_{H_j}\left(A_j\right). \tag{7.37}$$

Therefore,

$$\begin{aligned}
\tilde{L} \;=\;& V_{H_i}(B_i) + \boldsymbol{\lambda}_B^{\top}\ \mathbf{g}_H\left(B_i :\ A_{-i}\right) + \boldsymbol{\mu}_{B_i}^{\top}\ \mathbf{q}_{H_i}\left(B_i\right) \\
&+ \sum_{j \neq i} V_{H_j}\big(A_j\big) + \sum_{j \neq i} \boldsymbol{\mu}_{A_j}^{\top}\ \mathbf{q}_{H_j}\left(A_j\right) \\
&- V_{H_i}\big(A_i\big) - \boldsymbol{\lambda}_A^{\top}\ \mathbf{g}_H\left(A_i :\ A_{-i}\right) - \boldsymbol{\mu}_{A_i}^{\top}\ \mathbf{q}_{H_i}\left(A_i\right) \\
&- \sum_{j \neq i} V_{H_j}\big(A_j\big) - \sum_{j \neq i} \boldsymbol{\mu}_{A_j}^{\top}\ \mathbf{q}_{H_j}\left(A_j\right) \\
=\;& L_i\big(\boldsymbol{\lambda}_B, \boldsymbol{\mu}_{B_i}, B_i : A_{-i}\big) - L_i\big(\boldsymbol{\lambda}_A, \boldsymbol{\mu}_{A_i}, A_i : A_{-i}\big).
\end{aligned} \tag{7.38}$$

The above analysis shows that, for $i = 1$ to N_C and planning horizon H, $\forall A \in S$, $\forall A_i, B_i \in S_i$, $\forall \boldsymbol{\lambda}_A,\ \boldsymbol{\lambda}_B \succeq \mathbf{0} \in \Re^{mH}$, $\forall \boldsymbol{\mu}_A,\ \boldsymbol{\mu}_B \succeq \mathbf{0} \in \Re^{nH}$, and $\forall \boldsymbol{\mu}_{A_i},\ \boldsymbol{\mu}_{B_i} \succeq \mathbf{0} \in \Re^{n_i H}$,

$$\begin{aligned}
& L(\boldsymbol{\lambda}_B, \boldsymbol{\mu}_B, B_i, A_{-i}) - L(\boldsymbol{\lambda}_A, \boldsymbol{\mu}_A, A_i, A_{-i}) > 0 \\
\Leftrightarrow\ & L_i(\boldsymbol{\lambda}_B, \boldsymbol{\mu}_{B_i}, B_i : A_{-i}) - L_i(\boldsymbol{\lambda}_A, \boldsymbol{\mu}_{A_i}, A_i : A_{-i}) > 0.
\end{aligned}$$

Therefore, from the definition of ordinal potentiality in [60], Eqns. (7.21) and (7.30) form an ordinal potential game.

7.5 Distributed Optimization

Similar to the process in Chapter 6, the distributed optimization of PTZ parameters over a moving horizon can be broken down into three separate steps, where κ denotes the iteration counter:

1. **Camera Parameter Optimization:** Each camera C_i computes $(A_i, \boldsymbol{\lambda}_i, \boldsymbol{\mu}_i)$ to increase $L_i\left(\boldsymbol{\lambda}_i, \boldsymbol{\mu}_i, A_i\right)$ while holding A_{-i} constant. It then communicates the newly computed local primal-dual estimates $\left(A_{i(\kappa)}, \boldsymbol{\lambda}_i(\kappa), \boldsymbol{\mu}_i(\kappa)\right)$ and new portions of A_{-i} to its neighbors $\mathcal{N}_i$.

2. **Camera Parameter Replacement:** Each neighbor C_n of camera C_i (i.e. each $C_n \in \mathcal{N}_i$) receives $\left(A_{i(\kappa)}, \boldsymbol{\lambda}_i(\kappa), \boldsymbol{\mu}_i(\kappa), A_{-i(\kappa)}\right)$. It replaces its previous value of (A_i, A_{-i}) using the rules of replacement described in Section 7.5.1.

3. **Consensus on Lagrange Multipliers:** C_i performs dynamic average consensus on its local *Lagrange* multiplier vector $\boldsymbol{\lambda}_i(\kappa)$ and the *Lagrange* multiplier vectors received from cameras in $\mathcal{N}_i$ to converge towards a consensus *Lagrange* multiplier vector $\bar{\boldsymbol{\lambda}}$, using the *Lagrange* multiplier update law described in Eqn. (6.2), defined in Section 6.3.

This distributed optimization process is then iterated over κ until a stopping criteria is achieved.

Since the optimization problem described by Eqn. (7.29) is non-convex, any solution found may only be locally optimal. It is assumed that all agents start with identical values of $A(\kappa)$ for $\kappa = 0$ and that $A(0)$ is not on the separatrix dividing the domain of attraction of one local optimum from another.

7.5.1 Camera Parameter Replacement Rule

After C_i computes $A_{i(\kappa)}$, it broadcasts the information $\{i, \kappa, A_{i(\kappa)}, \boldsymbol{\lambda}_i(\kappa), \boldsymbol{\mu}_i(\kappa)\}$ to its neighbors $\mathcal{N}_i$, and will rebroadcast to its neighbors any *updated PTZ horizon information*, $\{l, A_l, \boldsymbol{\mu}_l, \kappa_l\}$, that it received since the last broadcast. Using rebroadcast, parameter updates travel throughout a connected network. For the l-th subvector in $C_i's$ version of A, C_i has values $A_{l(\kappa_l)}, \boldsymbol{\mu}_l(\kappa_l)$, and a time-stamp κ_l both computed by C_l, even if C_i and C_l are not neighbors. Because the network may contain loops, C_i may receive information about other cameras via multiple paths. C_i will replace its l-th subvector with the received information only if the time-stamp in $\{l, A_l, \boldsymbol{\mu}_l, \kappa_l\}$ is more recent than the time stamp corresponding to the value it is currently using. Otherwise, the message is discarded without rebroadcast.

7.5.2 Certificate for Optimality

For each unconstrained maximization problem given by Eqn. (7.30), the *KKT* conditions [59] are:

$$\nabla V_{H_i}(A_i^*) + \nabla \mathbf{g}_{H_i}(A_i^*)^\top \boldsymbol{\lambda}_i{}^* + \nabla \mathbf{q}_{H_i}(A_i^*)^\top \boldsymbol{\mu}_i^* = 0, \tag{7.39}$$

$$\mathbf{g}_{H_i}(A_i^*) \succeq \mathbf{0},\ \mathbf{q}_{H_i}(A_i^*) \succeq \mathbf{0}, \tag{7.40}$$

$$\boldsymbol{\lambda}_i{}^* \succeq \mathbf{0},\ \boldsymbol{\mu}_i^* \succeq \mathbf{0}, \tag{7.41}$$

$$\boldsymbol{\lambda}_i{}^{*\top} \mathbf{g}_{H_i}(A_i^*) = 0,\ \boldsymbol{\mu}_i^{*\top} \mathbf{q}_{H_i}(A_i^*) = 0. \tag{7.42}$$

The optimal primal-dual set of solutions $(A_i^*, \boldsymbol{\lambda}_i{}^*, \boldsymbol{\mu}_i^*)$ computed for the optimization problem in Eqn. (7.29) must satisfy the KKT conditions given by Eqns. (7.39 - 7.42). Optimization stops when either an optimum is achieved, a user-defined stopping condition is met, or the time interval allotted for optimization elapses (see Fig 7.2).

For the approach herein, the KKT conditions described in Eqns. (7.39 - 7.42) provide a certificate on the optimality and feasibility (i.e., satisfaction of the tracking and smoothness specifications) of the solution computed. Numeric algorithms to solve the constrained optimization problem defined in Eqn. (7.29), to which the KKT conditions of Eqns. (7.39 - 7.42) apply, first find a feasible solution, then search within the feasible set for the optimal feasible solution. Thus, when the time interval allotted for optimization elapses, even if the solution is sub-optimal, the methodology aids the network in finding a solution that is feasible. This results in all targets likely being tracked to the specified tracking accuracy, and all cameras adhering to the smoothness criterion, while procuring high-res imagery. After optimization, the cameras physically alter their settings to the optimal values in readiness for upcoming images at t_{k+1}. Thus, by using the replacement step in Section 7.5.1 and the Lagrange multiplier update law from Chapter 6, at each optimization iteration κ, every camera maintains an estimate of the primal-dual solutions of all cameras.

Chapter 8

Simulation and Analysis

This chapter describes and analyzes results obtained on a realistic *Matlab* simulation of the proposed approach. The goal of the simulation is to evaluate the performance of a distributed PTZ camera network using the methods described herein to obtain opportunistic high-res facial imagery of targets moving in a region, while tracking all targets at all times to a specified tracking accuracy. Subsequently, the chapter evaluates the moving horizon PTZ optimization approach described in 7 to observe temporal smoothness on the optimal PTZ parameter sequence.

8.1 Scenario, Setup and Experiment Details

Fig. 8.1, shows a 400 m^2 area being monitored by $N_C = 3$ calibrated cameras located at $C_1 = [10, 0, 3]^\top$, $C_2 = [0, 10, 3]^\top$, and $C_3 = [20, 10, 3]^\top m$. Camera locations are indicated by colored stars. The boundary of the FoV for each camera is drawn as a wide solid line in a color coordinated with the color of the position marker of the camera. Note that the FoV is the area in the interior of this polygon. Every target T^j is modeled as a circular disc of negligible height and a radius of 30 cm. All target discs are

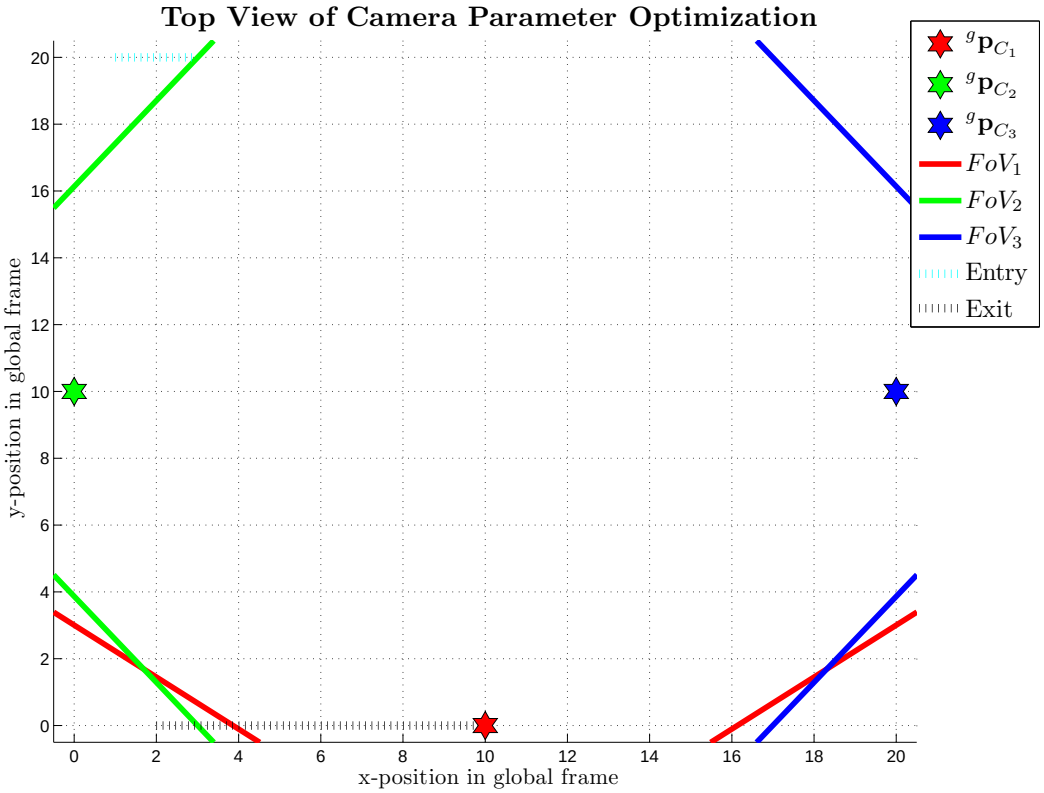


Figure 8.1: Top-view of 20×20 surveillance area at $t = 0$, before target entry: Camera locations are indicated by colored stars. The camera's FoV boundary is drawn on the ground plane using the same color as its star. The FoV of the camera is the convex area interior of this polygon.

coplanar to the ground plane. The entrance to the area is located at $y = 20, x \in [9, 11]$ and indicated by the cyan hash marks in Fig. 8.1. Targets enter through the entrance at random times; therefore, the total number of targets in the area is time variant. When a target T^j enters the area, its position coordinates are randomly initialized in $[{}^g x^j, {}^g y^j, 0]^\top$, where ${}^g x^j \in [9, 11]$ and ${}^g y^j = 20$.

When a new target is detected, the number of targets $N_T(t)$ is increased, and the target state is augmented to the state vector and included in the imaging and tracking value functions. The maximum number of targets permissible in the area was limited such that $0 \leq N_T(t) \leq \overline{N}_T$ where $\overline{N}_T = 10$. To ensure that targets entering the area are detected, the entrance must be constantly monitored. This is achieved by

inserting an artificial stationary target at (2, 20, 0) with constant position uncertainty of $2m^2$. Once a target is in the room, its motion is generated using the model in Eqn. (4.2). The exit to the room is located at $y = 0, x \in [2, 10]$ and indicated by the black hash marks in Fig. 8.1. If the target trajectory intersects the wall in this region, then the target has exited the room, in which case, the target state is removed from the state vector, excluded from the imaging and tracking value functions, and the number of targets $N_T(t)$ is decreased.

Remark 9 *Note that the target trajectory from Eqn. (4.2) may intersect a wall. If the point of intersection is the exit, then the target exits the area as described above. If the point of intersection is not the exit, then the target trajectory reflects off the wall.* $\diamond$

As discussed in Section 4.2, the measurement model depends on the camera parameters. In addition, while the image processing algorithms may compute the centroid of the feature region in the image plane to subpixel resolution, the covariance matrix used in the state estimation routine must account for the the uncertainty in the computed centroid relative to the "actual target centroid." Let $n_i^j(\mathbf{a}_i)$ represent the area occupied by T^j's image on C_i's image plane measured in sq. pixels. For this simulation, the estimation routine models the covariance of the measurement[1] of T^j by C_i as

$$\mathbf{C}_i^j(\mathbf{a}_i) = \begin{bmatrix} \frac{n_i^j(\mathbf{a}_i)}{p_i}\sigma_x^2 & 0 \\ 0 & \frac{n_i^j(\mathbf{a}_i)}{p_i}\sigma_y^2 \end{bmatrix}, \tag{8.1}$$

where p_i is the pixel resolution of C_i's image plane (in sq. pixels) and σ_x^2 and σ_y^2 (in sq. pixels) are positive constants. For this simulation, each camera C_i was set to an image resolution p_i of 800×600 sq. pixels, with $\sigma_x = \sigma_y = 5$ pixels.

[1]For computer vision applications, a model for the measurement covariance matrix is often learned from training data collected during deployment of the system.

Based on the properties described in Section 4.3, we define an imaging value associated with target T^j when imaged by camera C_i. Let vector $\mathbf{o}_{v^j}$ be the target's estimated velocity vector. Define the vector $\mathbf{o}_{C_i}$ to be the i-th camera's optical axis direction in the global frame

$$\mathbf{o}_{C_i} = {}^{g}_{c_i}\mathbf{R}\ {}^{c_i}\mathbf{e}_3, \tag{8.2}$$

where $\mathbf{e}_3 = [0, 0, 1]^\top$. Define $\mathbf{o}_{T^j}$ to be the vector from camera C_i's position to target T^j's estimated position. Using the vectors $\mathbf{o}_{v^j}$, $\mathbf{o}_{C_i}$, and $\mathbf{o}_{T^j}$ we define the scalars

$$o_c = \frac{\mathbf{o}_{C_i} \cdot \mathbf{o}_{T^j}}{\parallel \mathbf{o}_{C_i} \parallel \parallel \mathbf{o}_{T^j} \parallel}, \text{ and } o_o = \frac{\mathbf{o}_{C_i} \cdot \mathbf{o}_{v^j}}{\parallel \mathbf{o}_{C_i} \parallel \parallel \mathbf{o}_{v^j} \parallel}. \tag{8.3}$$

The scalar $o_c \in [-1, 1]$ yields the maximum possible positive value of 1 if camera C_i images target T^j such that T^j is at the center of its FoV. The scalar $o_o \in [-1, 1]$ has maximum magnitude when T^j's motion vector $\mathbf{o}_{v^j}$ is pointing directly toward or away from camera C_i.

To define the pose quality factor $\alpha_i^j(\mathbf{a}_i)$, we use the following assumption.

Assumption 4 *(Facial Direction) Target T^j faces in the direction indicated by vector $\mathbf{o}_{v^j}$.*

From Assumption 4 and Eqn. (8.3), when the scalar $o_o < 0$, T^j is likely to be facing camera C_i. This condition differentiates between targets facing C_i and those facing away from it. The relative pose quality factor is thus defined as

$$\alpha_i^j(\mathbf{a}_i) = \begin{cases} (o_c\ o_o)^2 & \text{if } o_o < 0 \\ 0 & \text{otherwise.} \end{cases} \tag{8.4}$$

Hence when $\alpha_i^j \in [0, 1]$ is large, it is likely that T^j is facing C_i and at the center of C_i's FoV.

The imaging value *expected* to be obtained by camera C_i for imaging T^j is defined as[2]

$$V_{I_i}^j\left(\mathbf{a}_i, {}^g\hat{\mathbf{x}}^j, t\right) = w_i^j(t)\; r_i^j\left(\mathbf{a}_i, {}^g\hat{\mathbf{p}}^j\right)\; \alpha_i^j\left(\mathbf{a}_i, {}^g\hat{\mathbf{x}}^j\right), \tag{8.5}$$

where, the weight $w_i^j(t)$ is defined as the continuously differentiable and bounded function

$$w_i^j(t) = \sigma_d\left(d^j(t)\right)\;\; \sigma_v\left(\bar{V}^j, V_{I_i}^j(t)\right). \tag{8.6}$$

In Eqn. (8.6), the scalars $\sigma_d = 1 + \frac{1}{1+\exp\left(l_d d^j(t)\right)}$, and $\sigma_v = \frac{1}{1+\exp\left(l_v\left(\bar{V}^j - V_{I_i}^j(t)\right)\right)}$. The symbol $d^j(t)$ is the distance between T^j's estimated position at time t and the exit. The symbol $\bar{V}^j = \max_{\tau<t,\; i\in[1,N_c]}\left(\bar{V}_{I_i}^j(\tau)\right)$ is the best image quality *achieved* for imaging target T^j from any camera and at any prior imaging time, where the imaging value *achieved* by camera C_i for imaging T^j is

$$\bar{V}_{I_i}^j\left(\mathbf{a}_i, {}^g\mathbf{x}^j, t\right) = \bar{r}_i^j\left(\mathbf{a}_i, {}^g\mathbf{p}^j\right)\; \bar{\alpha}_i^j\left(\mathbf{a}_i, {}^g\mathbf{x}^j\right). \tag{8.7}$$

In Eqn. (8.7), the scalars $\bar{r}_i^j$ and $\bar{\alpha}_i^j$ indicate the imaging resolution and camera to target pose quality actually achieved. With these definitions $\sigma_d(t) \in [1,2]$ and $\sigma_v(t) \in [0,1]$. This definition of $w_i^j(t)$ gives higher value (i.e., emphasizes) those targets nearest to the exit and those targets for which the value of the next image is expected to improve the most relative to prior imagery.

All cameras optimize simultaneously, using an interior-point method [67]. The tracking constraint in Eqn. (5.12) uses $\bar{\mathbf{T}} = 1.0\ \mathrm{m}^{-2}$. C_i receives camera parameters $\mathbf{a}_{-i}^*$ through its neighbors, and uses its current parameters $\mathbf{a}_i$ to implement the method described in Section 7.5.

[2]Note that the expectation over C_i's FoV is applied in Eqn. (5.11).

8.2 Single Step PTZ Optimization

This section analyzes simulation results obtained while applying the proposed PTZ optimization approach over a single time-step. The results in Figs. 8.2 - 8.4 correspond to a 31 sec. simulation. All cameras image at a frequency of 1 Hz, with the first images obtained at time $t = 1$ second.

8.2.1 Single Trial Results

For this simulation, targets T^1 to T^{10} entered the area at times 0.1, 10.2, 11.1, 14.2, 20.4, 22.9, 27.2, 28.8, 30.1, and 30.4 seconds, respectively. Target T^1 left the area at time 14.4 seconds. No other targets left the area. When T^j enters, a camera monitoring the entrance images it, detects the new target and augments it to its state vector. Other cameras add the new target to their state vector as they receive the new target information at the state estimation stage.

Cameras maximize their local Lagrangians $L_i(\boldsymbol{\lambda}_i^*, \mathbf{a}_i^*)$ to satisfy the tracking spec and maximize their local Bayesian imaging values. Fig. 8.2a shows that the expected Bayesian tracking value $\mathbf{V}_T^j(\mathbf{a}^*)$ is greater than the tracking spec, at all times; therefore, all primal-dual solutions $(\mathbf{a}_i^*, \boldsymbol{\lambda}_i^*)$ obtained through local optimization are *expected* to be feasible at all imaging instants. Because the solutions are strictly feasible, using Eqn. (6.8), it is trivial to prove that the dual optimal Lagrange multiplier vectors for all cameras are $\boldsymbol{\lambda}_i^*(t) = \bar{\boldsymbol{\lambda}} = \mathbf{0} \in \Re^m$, where $m = 2N_T(t)$.

The proposed approach utilizes predicted target motion based on state estimates from the last imaging time. Estimation error or unexpected maneuvers by targets, such as a simulated target reflecting of a wall, can lead to a drop in the accuracy actually

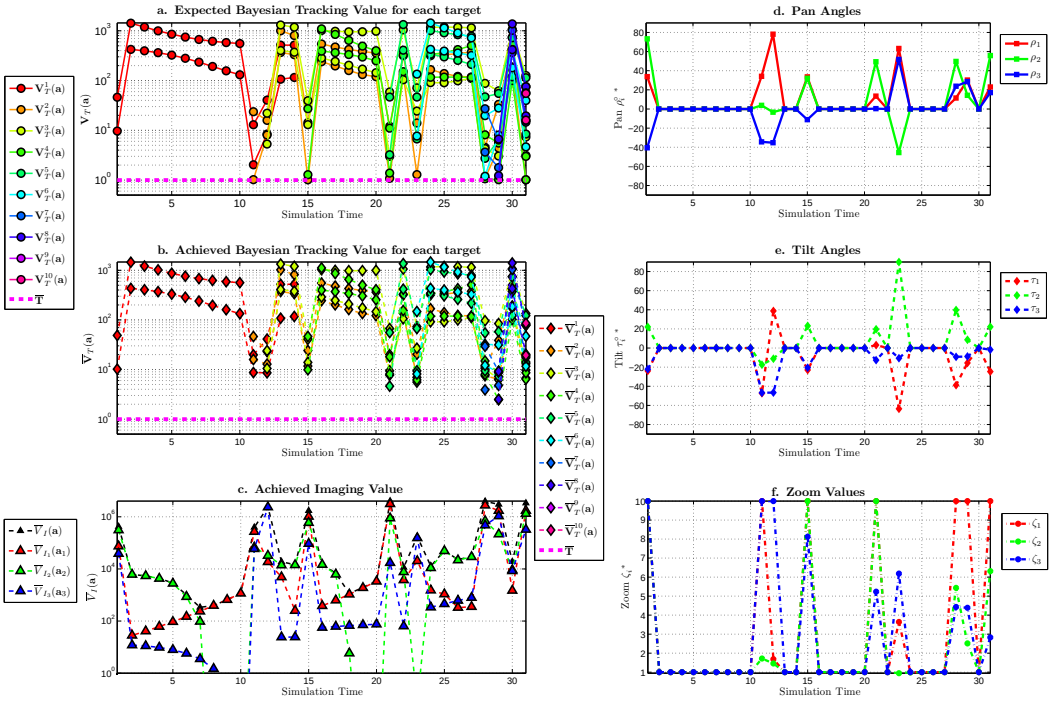
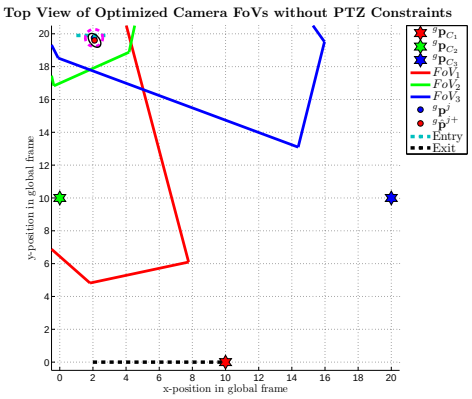


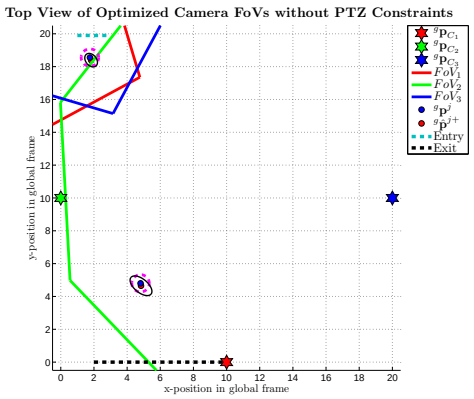
Figure 8.2: Bayesian tracking and imaging values: Fig. a (top left) shows that the camera network expects to successfully and co-operatively satisfy the tracking constraint $\bar{\mathbf{T}} = 1.0$ m^{-2} for every target, at all times. Fig. b (middle left) shows that the achieved tracking values satisfy the tracking constraint $\bar{\mathbf{T}}$. Fig. c (bottom left), plots the achieved local imaging value $\bar{V}_{I_i}(\mathbf{a}_i)$ and the achieved global imaging value $\bar{V}_I(\mathbf{a})$ (i.e., sum of the local values). Figs. d (top right), e (middle right) and f (bottom right) show the per camera optimal pan angle $\rho_i^{\circ\,*}$, tilt angle $\tau_i^{\circ\,*}$ and zoom $\zeta_i{}^*$ values, respectively.

achieved. Fig. 8.2b shows the tracking value $\bar{\mathbf{V}}_T^j(\mathbf{a}^*)$ actually achieved by the network. Various instances of differences between the expected and achieved accuracy can be observed through the simulation time. Since the target motion is a random process, there is no deterministic guarantee that the achieved accuracy meets the specification.

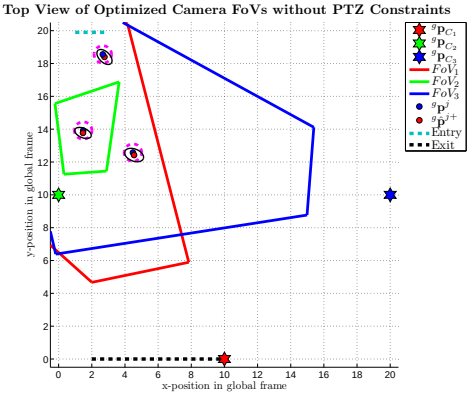
Fig. 8.2c shows the achieved imaging values of each camera and of the network of cameras. The peak values occur at those opportunistic times at which the cameras procure high-res facial images of targets, while the tracking constraints on all targets are satisfied. A high value for $\bar{V}_{I_i}(\mathbf{a}_i^*)$ indicates a high-res facial capture by camera C_i.



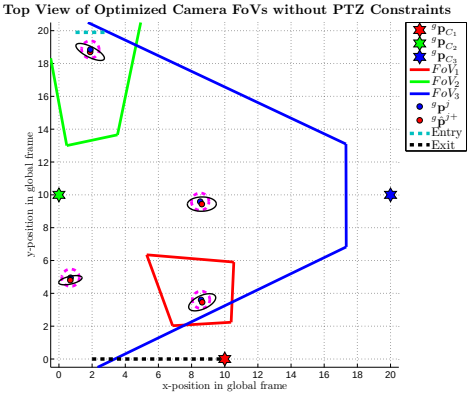
(a) Optimized FoVs at $t = 1$.



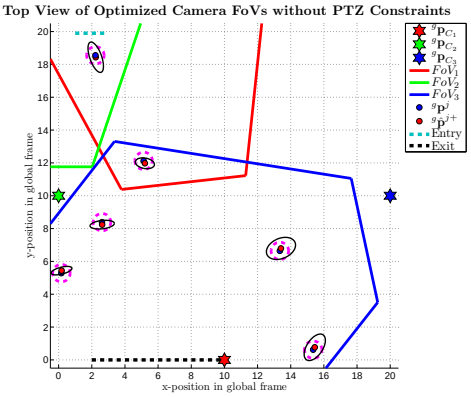
(b) Optimized FoVs at $t = 11$.



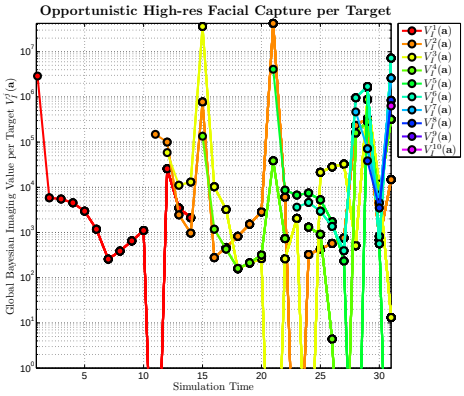
(c) Optimized FoVs at $t = 15$.



(d) Optimized FoVs at $t = 21$.



(e) Optimized FoVs at $t = 28$.



(f) Expected hi-res capture.

Figure 8.3: Top-view of optimized FoVs: Figs. a, b (both top row), c, d (both middle row), and e (bottom left) show the optimized FoVs at times of opportunity: $t = 1$, $t = 11$, $t = 15$, $t = 21$ and $t = 28$, respectively. Each figure shows the optimized FoVs of the cameras after feasible optimal solutions are achieved. Fig. f (bottom right) shows the expected per target imaging value $V_I^j(\mathbf{a}(t))$ from Eqn. (8.8).

Given the target trajectories of this simulation, all cameras availed opportunities for high-res image capture throughout simulation time.

Figs. 8.2d - 8.2f show the per camera optimized PTZ values versus time. Top-views of the camera FoVs for a selection of high-res imaging opportunities is shown in Fig. 8.3.

Figs. 8.3a, 8.3b, 8.3c, 8.3d, and 8.3e show the post-optimization FoVs of the cameras for time-steps $t = 1$, $t = 11$, $t = 15$, $t = 21$, and $t = 28$, respectively. The posterior estimate of the position of the centroid of each target is marked by a red dot. The actual position of the centroid of each target is marked by a blue dot. A wide pink dashed curve indicates the surface area occupied by a target on the ground plane, relative to the actual target centroid position. Note that the number of pixels occupied by target T^j on the image plane of camera C_i is a function of target surface area, true target location, and $\mathbf{a}_i$. The posterior $1-\sigma$ position error ellipse corresponding to the estimated position of each target is drawn as a black curve.

Target T^1 enters at time $t = 0.1$. Cameras collaboratively image T^1 at time $t = 1$, where C_2 images T^1 with the highest imaging value among all cameras (see Fig. 8.2c). Similarly, C_1 obtains an opportunistic high-res image of T^2 at time $t = 21$. Note that in all cases, the entrance and all targets are within at least one FoV. Fig. 8.3f plots the time history of the expected imaging value per target acquired by all cameras:

$$V_I^j\big(\mathbf{a}(t)\big) = \sum_i E\left\langle V_{I_i}^j\left(\mathbf{a}_i(t),\ {}^g\mathbf{p}^j(t)\right)\right\rangle, \tag{8.8}$$

where $V_{I_i}^j(\mathbf{a}_i)$ is defined in Eqn. (8.5). The number of curves is different at each time because the number of targets is time varying. A high-res image capture of T^j by any camera C_i is indicated by a spike in the global *Bayesian* imaging Value function $V_I^j\big(\mathbf{a}(t)\big)$. The figure shows that for this simulation run, the camera network obtained

at least one high-res facial image of each target in the area, at times-of-opportunity distributed throughout the time period of the simulation. Combining the information from this figure with that from Fig. 8.2c, we see for example that as T^2 moves through the room, at various times, different cameras have opportunities to image it.

Fig. 8.4 is an example, using time-step $t = 21$, of the optimization process that each camera performs before each imaging instance. All cameras simultaneously perform a few optimization iterations, then broadcast their (approximate) primal-dual solutions, update their local estimates of $\mathbf{a}^*$ using the sub-vectors received from their neighbors and resume the optimization process. The broadcast instances are indicated by the pink dashed vertical lines. This process repeats till an optimum is reached or time expires.

Remark 10 *We start each optimization iteration with a wide FoV. This choice of initial condition facilitates the search for a feasible solution. This is similar to using a metaheuristic [68, 69] to aid computation of a feasible solution.* $\diamond$

Remark 11 *To achieve high-res imagery and satisfy the tracking threshold, camera FoV's alter significantly between successive time instants. Examples of such changes in pan and tilt values can be seen in Figs. 8.2d and 8.2e throughout simulation time. Such rapid panning and tilting motion can hamper image quality due to motion blurring and may also cause mechanical wear in cameras. This is addressed by extending the PTZ optimization approach into the moving horizon optimization method described in Chapter 7. Results for the moving horizon optimization approach are discussed in Section 8.3.* $\diamond$

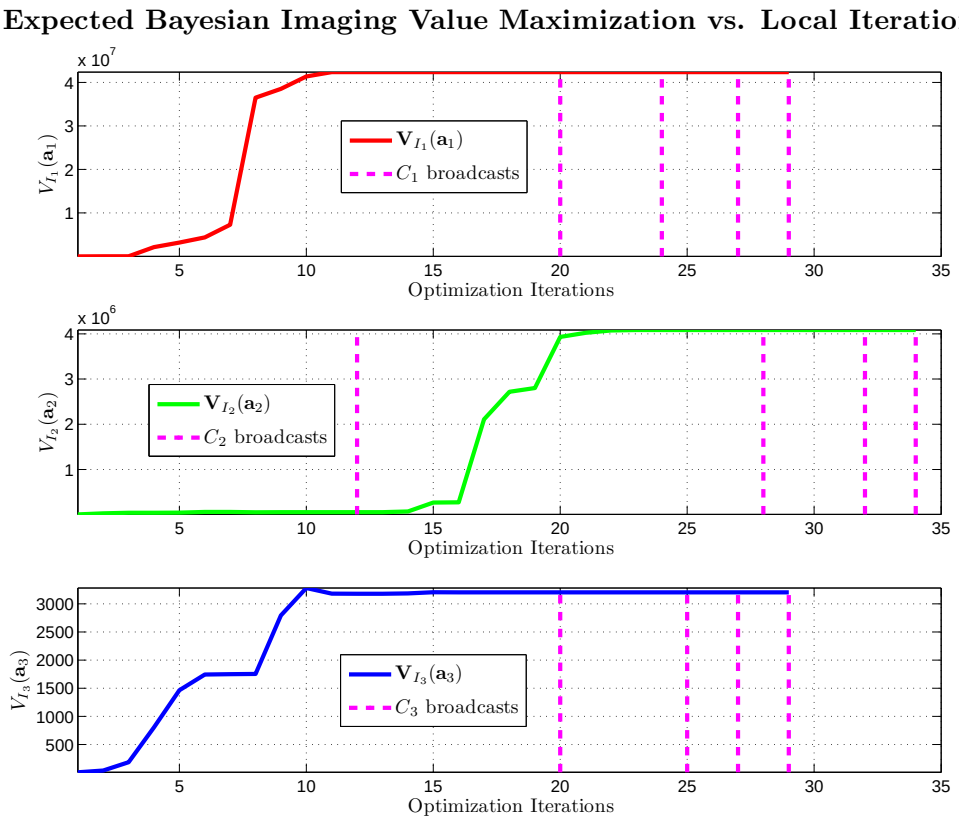


Figure 8.4: Optimization: The maximization of $V_{I_i}(\mathbf{a}_i)$ versus the number of local iterations at time-instant $t = 21$. The vertical pink dashed lines indicate local iterations κ_i at which C_i broadcast parameters as described in Section 6.2. After the cameras have collaboratively found a feasible PTZ configuration, C_1 and C_2 capitalize on the target configuration to obtain images expected to have very high values.

Remark 12 *As seen in Fig. 8.3f, in spite of the formulation of Eqn. (8.6), it is still possible that the camera will attempt to acquire images of targets with lower image value than was previously obtained. There are at least two explanations. First, this can occur inadvertently because a previously imaged target is sometimes visible in the FoVs of cameras that have been optimized for imaging other targets. Second, as long as the expected imaging value V_I^j is finite, which it always is, the optimization still receives some value for new imagery, even if it is not of higher quality than previous imagery.* $\diamond$

8.2.2 Multi-Trial Performance Analysis

This section provides an analysis of the performance of the proposed PTZ camera network approach using data from $N = 100$ *Matlab* simulation runs. Across all simulation runs, target trajectories and target times of entry were independent, with the target times of entry designed such that target T^j always entered before target T^{j+1}. To make results comparable, all other parameters (e.g. camera locations, image resolution, pixel noise, area entrances and exits, etc.) were defined to be the same for all simulation runs, as defined in Section 8.1.

For a dynamic PTZ camera network, define $\bar{V}_D^j(n)$ to be the maximum global imaging value achieved for target T^j during simulation run n. Similarly, for a constant (static) PTZ camera network, let $\bar{V}_S^j(n)$ be the maximum global imaging value achieved for target T^j during simulation run n. Define a *performance ratio* $\bar{V}_B^j(n)$ as

$$\bar{V}_B^j(n) = \frac{\bar{V}_D^j(n)}{\bar{V}_S^j(n)}, \tag{8.9}$$

where $\bar{V}_B^j(n)$ provides a measure of the relative gain in imaging value achieved by utilizing a dynamic PTZ configuration rather than a static PTZ configuration.

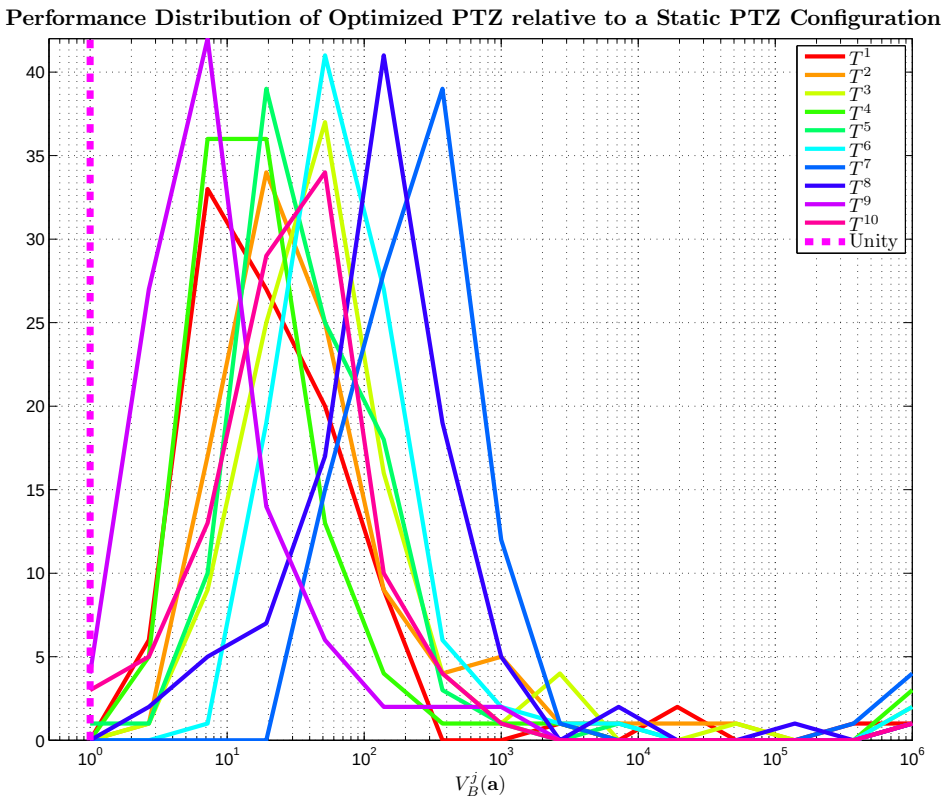


Figure 8.5: Performance improvement distribution: Imaging performance improvement ratio of dynamic PTZ camera configuration relative to a static PTZ configuration. Each colored line corresponds to a distinct target, showing the distribution of the per target performance ratio $\bar{V}_B^j$ (see Eqn. (8.9)) over 100 simulation runs. The range of $\bar{V}_B^j$ values is plotted as bins on the horizontal axis. The dynamic PTZ config. significantly outperforms the static PTZ config.

Fig. 8.5 shows the distribution (histogram) of the performance ratio $\bar{V}_B^j$ over $N = 100$ simulation runs. The network of dynamic PTZ cameras consistently outperforms the static camera configuration by procuring images of higher quality. Fig. 8.5 uses a semilog horizontal axis with a maximum of 10^6. The performance improvement ratio actually varies from 1 to 10^{12}. Such enhanced image quality is better suited for image analysis and scene understanding.

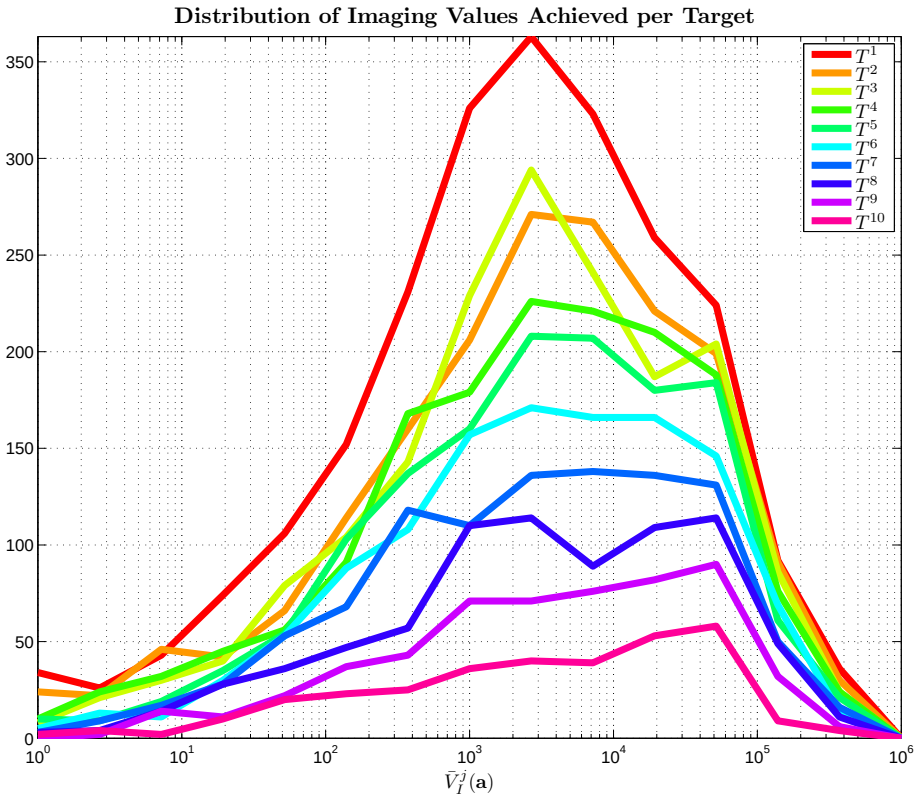


Figure 8.6: Opportunistic imaging distribution: Fig. shows the distribution of the per target achieved image value over $N = 100$ simulation runs. Each colored line corresponds to a distinct target, showing the histogram of values.

Fig. 8.6 displays the histogram of the per target achieved imaging value, which is denoted by $\bar{V}_I^j(\mathbf{a})$. The number of opportunistic high-res images obtained for target T^1 is greater than those obtained for T^2, and so on. There are at least two explanations. First, T^j always enters before target T^{j+1}; therefore, the cameras likely have more opportunities to image T^j at a higher resolution than T^{j+1}. Second, the difficulty in acquiring high-res images increases as the number of targets in the area increases, due to the increase in the number of feasibility constraints (see Eqns.(5.8), (5.12), and (5.14)).

Remark 13 *The camera positions across all simulation runs (static and dynamic) were left unchanged. The locations were selected so that, for the static configuration, all locations within the entire area were within the FoV of at least one camera. Altering the positions of entrances, exits, or the static parameter cameras may provide different performance than the static configuration used herein.* $\diamond$

This section demonstrates that the proposed method causes the cameras to cooperate to ensure that all targets are expected to be tracked to an accuracy better than $\bar{\mathbf{T}}$, and that high-res target images are obtained at times-of-opportunity implicitly defined by the feasibility constraints. The statistical analysis provides a measure of the increase in imaging performance obtained while using the proposed method.

8.3 Moving Horizon PTZ Optimization

This section analyzes simulation results obtained while applying the proposed PTZ optimization approach over a moving planning horizon H (see Chapter 7). The scenario, setup and experiment details are as defined and described in Section 8.1.

8.3.1 Single Trial Results

The results in Figs. 8.7 - 8.8 correspond to a 31 sec. simulation. All cameras image at a frequency of 1 Hz, with the first images obtained at time $t = 1$ second. All cameras optimize simultaneously, using an interior-point method [67]. The tracking constraint in Eqn. (7.24) uses $\bar{\mathbf{T}} = 1.0\ \mathrm{m}^{-2}$. C_i receives camera parameter sequences A^*_{-i} through its neighbors, and optimizes its current parameter sequence A_i to implement the method described in Chapter 7.

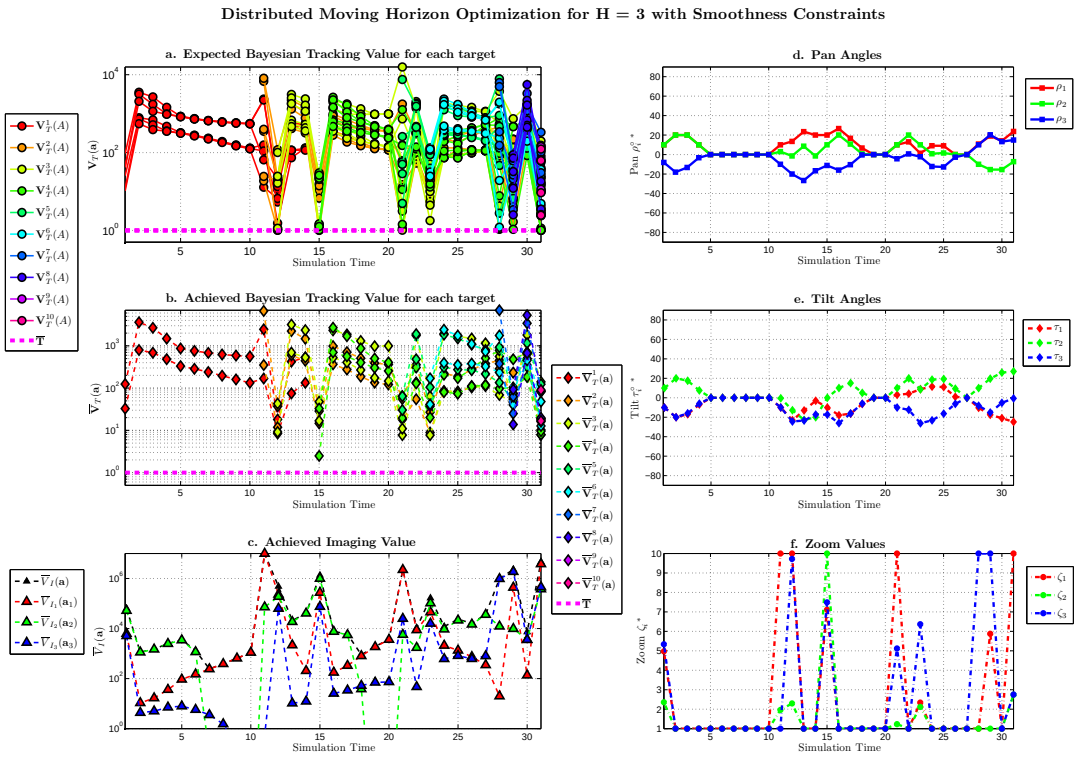


Figure 8.7: Moving horizon optimization with $H = 3$: Fig. a (top left) shows the tracking value $\mathbf{V}_T(A)$ that every camera expects to achieve for every target over planning horizon $H = 3$. Fig. b (middle left) shows the tracking value $\bar{\mathbf{V}}_T$ that every camera actually achieves. Fig. c (bottom left), plots the achieved local imaging and global imaging values (i.e., sum of the local values). Figs. d (top right), e (middle right) and f (bottom right) show the optimal pan $\rho_i^{\circ\,*}$, tilt $\tau_i^{\circ\,*}$ and zoom $\zeta_i{}^*$, respectively.

A planning horizon of $H = 3$ time-steps was selected for the purpose of this simulation. This allows each camera C_i to optimize a sequence of PTZ parameters $A_{i(k+1)} = \{\mathbf{a}_i(k+1), \mathbf{a}_i(k+2), \mathbf{a}_i(k+3)\}$ over the planning horizon $H = 3$, instead of only optimizing the PTZ parameters $\mathbf{a}_i(k+1)$. Cameras maximize their local Lagrangians $L_i(\boldsymbol{\lambda}_i^*, \boldsymbol{\mu}_i^*, A_i^*)$ to satisfy the tracking spec and the PTZ smoothness constraints, and maximize their local Bayesian imaging values. Fig. 8.7 shows the results hence obtained.

Fig. 8.7a shows that the tracking constraint $\mathbf{g}_H(A)$ from Eqn. (7.17) is *expected* to be satisfied throughout simulation time, i.e., all tracking values $\mathbf{V}_T(A)$ obtained over the planning horizon $H = 3$ are above the horizontal pink dashed line. Therefore, all

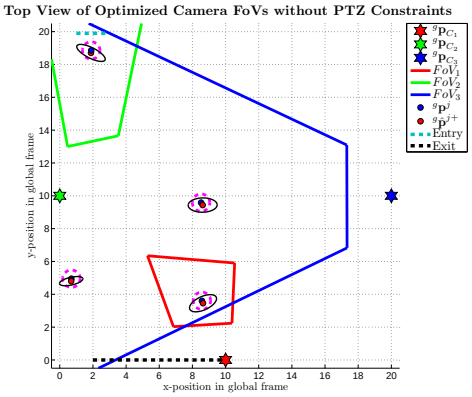
primal-dual solutions $(A_i^*, \boldsymbol{\lambda}_i^*, \boldsymbol{\mu}_i^*)$ obtained through local optimization are *expected* to be feasible at all imaging instants.

The approach utilizes predicted target trajectories based on state estimates from the last imaging time. Unexpected maneuvers by targets, such as a simulated target reflecting of a wall, can lead to a drop in the accuracy actually *achieved*. Fig. 8.7b shows the tracking value $\bar{\mathbf{V}}_T^j$ actually achieved by the network, collaboratively. Various instances of differences between the expected and achieved accuracy can be observed through simulation time. Since target motion is a random process, there is no deterministic guarantee that the achieved accuracy meets the specification. Because the expected solutions are strictly feasible, using Eqn. (7.42), it is trivial to prove that the dual optimal Lagrange multiplier vectors for all cameras are $\boldsymbol{\lambda}_i^*(t) = \bar{\boldsymbol{\lambda}} = \mathbf{0} \in \Re^{mH}$, where $m = 2N_T(t)$.

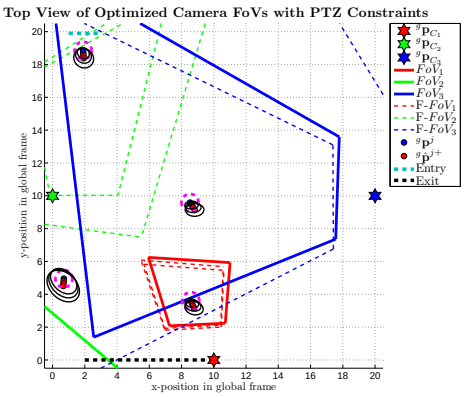
8.3.2 Comparative Analysis

Figs. 8.8a and 8.8b provide a comparison of the post-optimization FoVs computed using the single-step PTZ optimization process (i.e. $H = 1$) and the moving horizon PTZ optimization process (i.e. $H > 1$).

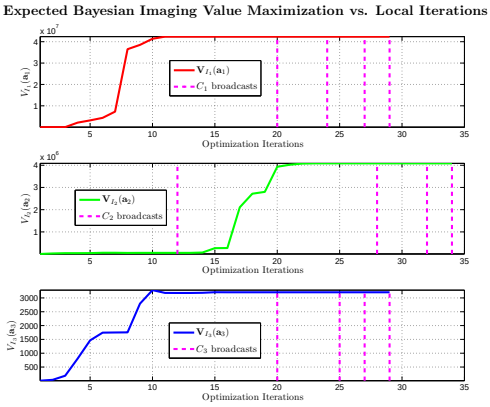
The actual position of the centroid of each target is marked by a blue dot. A wide pink dashed curve indicates the surface area occupied by a target on the ground plane, relative to the actual target centroid position. Posterior target centroid trajectory estimates are marked by red dots. The posterior $1 - \sigma$ position error ellipses corresponding to the estimated target trajectories are drawn as black curves.



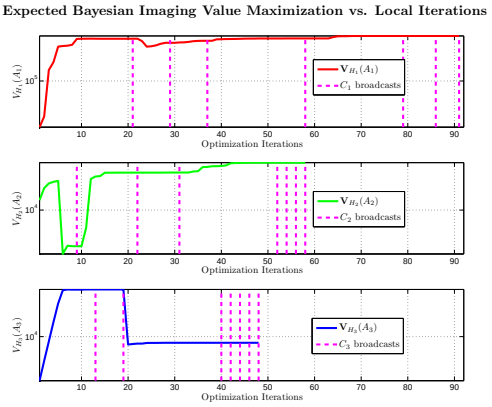
(a) FoVs at $t = 21$.



(b) FoVs at $t = 21$ over $H = 3$.



(c) Optimization for $t = 21$.



(d) Optimization for $t = 21$ over $H = 3$.

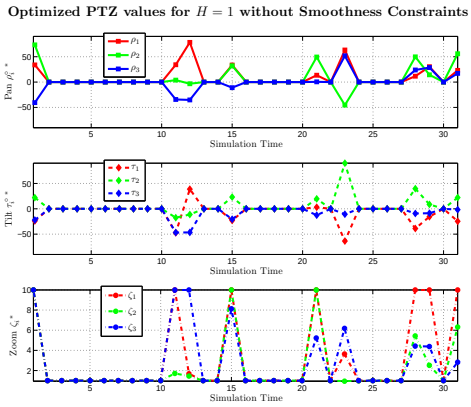
Figure 8.8: Moving horizon high-res image capture: Figs. a and b (top row) show the optimized FoVs at time $t = 21$, for $H = 1$ and $H = 3$, respectively. Fig. b shows the future FoVs computed over the planning horizon $H = 3$ as dotted polygons. Figs. c and d (bottom row) show the maximization of $V_{I_i}(\mathbf{a}_i)$ and $V_{H_i}(A_i)$ versus the number of local iterations at time-instant $t = 21$, respectively.

Figs. 8.8c and 8.8d show the maximization of $V_{I_i}(\mathbf{a}_i)$ and $V_{H_i}(A_i)$ versus the number of local iterations at time-instant $t = 21$. The vertical pink dashed lines indicate local iterations κ_i at which C_i broadcast parameters as described in Sections 6.2 and 7.5.1, respectively. As explained in Section 8.2.1, and as seen in Figs. 8.8a and 8.8c, at time $t = 21$, cameras C_1 and C_2 obtain hi-res images while all three cameras collaboratively satisfy the tracking spec. Similarly, Figs. 8.8b and 8.8d show that, using a planning scheme over a moving horizon of $H = 3$ timesteps, camera C_1 obtains a hi-res image while all three cameras collaboratively satisfy the tracking spec over the planning horizon.

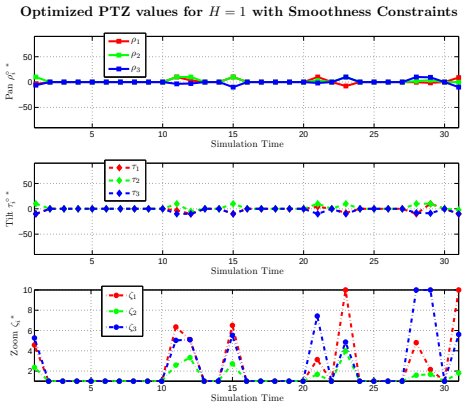
In comparison with Fig. 8.8a, the FoVs in 8.8b show that due to constraints on camera motion, C_2 is unable to pan and tilt by an angle sufficient for it to obtain a high-res image at time $t = 21$ (also see Figs. 8.2c and 8.7c). This is addressed by using the planning scheme, as at time $t = 21$, camera C_2 uses the estimated target trajectories from Eqn. (7.5), and prepares to capture a high-res image at time $t = 23$. As seen in Fig. 8.7c, camera C_2 successfully captures a high-res image at time $t = 23$.

Figs. 8.9a - 8.9d show the optimized PTZ parameters for varying planning horizon lengths H. Results in Fig. 8.9a are obtained when PTZ smoothness constraints are not applied. Figs. 8.9b - 8.9d show the effects of applying smoothness constraints over an increasing planning horizon length. Results in Fig. 8.9b are obtained when PTZ constraints are applied without planning. As can be seen, camera motion is severely constrained, which results in lower zoom values, likely resulting in low-res image capture.

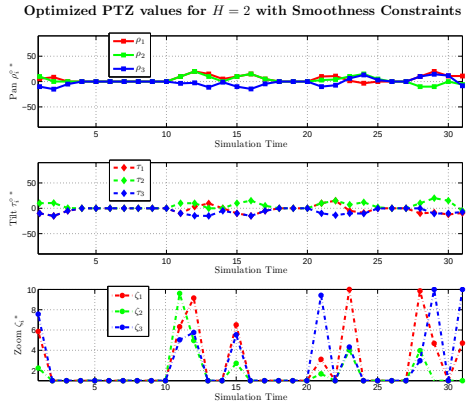
In comparison to the optimized PTZ parameters in Fig. 8.9a (where $H = 1$), the parameters in Figs. 8.9d show that an increase in horizon length to $H = 3$ results in temporally smooth panning and tilting of cameras. Also, in relation to the zoom



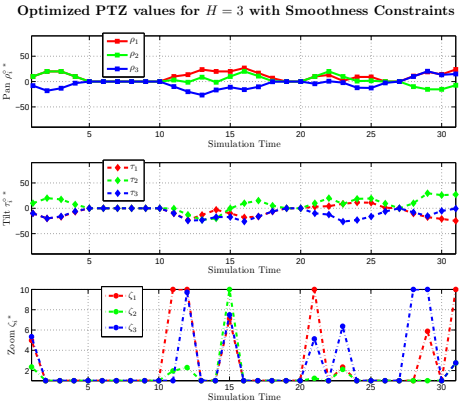
(a) PTZ values for ($H = 1$; NC).



(b) PTZ values for ($H = 1$; SC).



(c) PTZ values for ($H = 2$; SC).



(d) PTZ values for ($H = 3$; SC).

Figure 8.9: Optimal PTZ values: NC - No PTZ smoothness constraints applied; SC - PTZ smoothness constraints applied. Figs. a and b (top row) show the optimized PTZ values over $H = 1$, with and without PTZ smoothness constraints, respectively. Figs. c and d (bottom row) show the optimized PTZ values when PTZ smoothness constraints are applied over $H = 2$ and $H = 3$, respectively.

values in Figs. 8.9b and Fig. 8.9c, Fig. 8.9d shows high zoom values for more timesteps during simulation time. This results in cameras likely obtaining more images at higher resolution. Comparing Figs. 8.9d and 8.9a shows that, although camera motion obtained while applying PTZ smoothness constraints over $H > 1$ is constrained, the PTZ values are significantly smoother than those obtained when no planning is enforced.

Remark 14 *Note that the number of local iterations required for convergence to a local optimal increases with an increase in H (see Figs. 8.8c and 8.8d). This occurs before an increasing horizon length results in an increase in the number of variables to be optimized along with an increase in the number of constraints to be satisfied.* $\diamond$

Remark 15 *Greater maneuverability in the pan and tilt angle values of the cameras results in higher zoom values (see Figs. 8.9b and 8.9d). This occurs because greater maneuverability allows each camera to scan more of the area under surveillance. This increases the probability of more targets being in its FoV and thus is likely to provide more opportunities for high-res imaging.* $\diamond$

8.3.3 Multi-Trial Performance Analysis

This section provides a performance analysis of the proposed approach using data from $N = 100$ *Matlab* simulation runs. Target trajectories and target times of entry were independent across all simulation runs with the condition that target T^j always entered before target T^{j+1}. To make results comparable, all other parameters (e.g. camera locations, image resolution, pixel noise, area entrances and exits, etc.) were selected to be the same for all simulation runs, and are defined in Section 8.1.

To explain the merits of the proposed approach, we will discuss and analyze results obtained when implementing four distinct PTZ optimization schemes on common simulation scenarios. Scheme 1 does not employ planning (i.e. $H = 1$), nor does it employ PTZ smoothness constraints. Scheme 2 enforces PTZ smoothness constraints without planning (i.e. $H = 1$). Scheme 3 utilizes a planning horizon $H > 1$ and optimizes a PTZ sequence while adhering to PTZ smoothness constraints over planning horizon H. Fig. 8.10 corresponds to results from 21 sec. simulation runs with $\overline{N}_T = 4$. Figs. 8.10a, 8.10b, 8.10c and 8.10d show the distribution (histogram) of the maximum imaging values achieved by the camera network for imaging targets T^1, T^2, T^3 and T^4, respectively. The Fig. provides a comparative analysis between imaging performance obtained while employing the distinct optimization schemes described above.

In Fig. 8.10, the imaging performance distribution achieved while imaging with Scheme 1 is shown by lines colored in red. As discussed in Section 1.2, better scene understanding and image analysis is likely possible if cameras in the network adhered to smoothness constraints on their respective PTZ parameters. The green lines show the detrimental effect on imaging performance when PTZ smoothness constraints are enforced without planning (also see Fig. 8.9b). Fig. 8.10 shows that the camera network achieves the best target imaging performance under Scheme 1, and that significant imaging performance is lost when imaging under Scheme 2.

The cyan and purple lines show the distribution of achieved imaging performance when the camera network images using the proposed approach with planning horizons of $H = 2$ and $H = 3$, respectively. By comparing the green lines with the cyan lines, it can be deduced that incorporating a planning horizon along with enforcement of PTZ smoothness constraints leads to an improvement in imaging performance. The

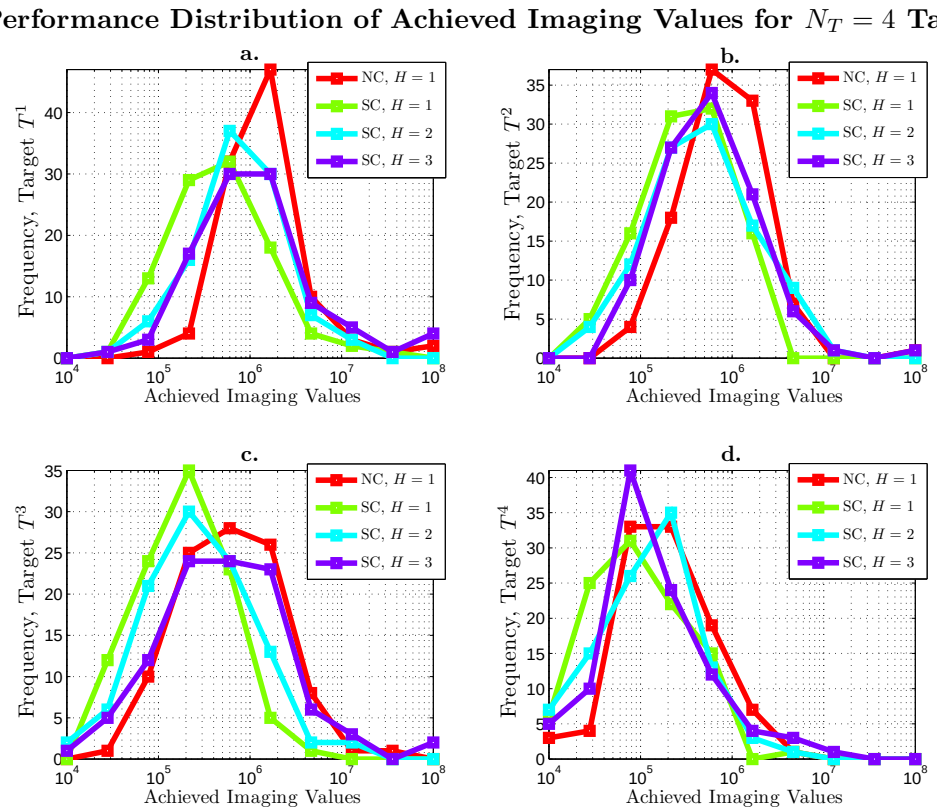


Figure 8.10: Imaging performance distribution: *NC* - No PTZ smoothness constraints applied; *SC* - PTZ smoothness constraints applied. Fig. shows the comparative imaging performance achieved while imaging target T^j when cameras optimize their PTZ parameters using varying horizon lengths.

lines in purple show that further performance improvement is observed by increasing the planning horizon to $H = 3$ (also see the zoom values obtained in Figs. 8.9c and 8.9d).

Remark 16 *Note that the achieved imaging value can differ from the expected imaging value since the imaging value achieved is a function of the number of pixels occupied by a target on the camera's image plane (see Eqn. 8.5). A minor random deviation in the target's true position from its expected position results in a pixel count on the image plane that is often different from the expected pixel count. This may result in the imaging value obtained to be significantly different than the expected imaging value.* $\diamond$

8.4 Discussion of Implementation Issues

For convex problems, the proposed distributed optimization methodology would converge to the unique global optimum for each imaging time instant. As with many practical applications, visual sensing problems such as the one considered herein are inherently non-convex (refer to Fig. 5.1), and thus the solution obtained may only be locally optimal. The large variable space makes design of an exhaustive search impractical. In addition to being non-convex, the local imaging value and the constraint functions are nonlinear. Our implementation used the Matlab function '*fmincon*', which is offered as part of the Optimization Toolbox and is designed to solve nonlinear optimization problems with nonlinear constraints.

To facilitate the search for a feasible solution, at the start of each optimization interval, we initiated each camera using the optimal pan and tilt values from the end of the prior optimization interval, but reset the zoom parameter to its minimum value (i.e., widest FoV). The wide FoV initialization was preferred as it enhances feasibility and convexity of the value function, see Fig. 5.1 and Remark 10. This initialization method ensures that all agents begin the optimization process from the same value of $\mathbf{a}$. This initialization worked well in the sense that a feasible solution was found for every imaging instant of every trial; nonetheless, alternative initialization and relaxation techniques, could be investigated.

The moving horizon optimization approach described in Chapter 7 is a process that requires frequent replanning as the visual sensing application as a process with fast dynamics. MPC applications with a large number of variables and lengthy planning horizons utilize FastMPC methods [44] to generate look-up tables for reduction in computation time. Target motion is stochastic and hence using a predetermined look-up

table of objective function values for the visual sensing application is inappropriate. Since the imaging value is a function of target trajectories, using Machine Learning algorithms [70] to generate a probabilistic metric on target trajectories for generation of FastMPC look-up tables could help reduce computation power consumed. These are interesting topics for future research.

Finally, it is important to note that if the initial parameters of the cameras were not identical and were distributed about a saddle point of the value function, such that some initial parameter vectors were in the domains of attraction (DOA) of different local optima, then different agents could conceivably converge toward different locally optimal points prior to communicating their new settings. After the communication, there would be no guarantee that the camera parameter settings of different agents are all within the DOA of the same locally optimal point. We ensured that all agents start with the same value for $\mathbf{a}$. This issue and methods to ensure convergence to the global optimum are interesting areas for future research.

Chapter 9

Conclusion and Future Research

This dissertation addressed the design of a method for a distributed network of smart imaging sensors to collaboratively track all targets to a specified accuracy while also acquiring high resolution images at times-of-opportunity. Subsequently, the problem of maintaining temporal smoothness on the optimal PTZ parameters, while obtaining high-res imagery is also addressed. The solution uses a *Bayesian* framework that trades off higher imaging value versus increased risk of the target not being in the field-of-view.

The single-step Bayesian imaging value depends on the target's expected position, direction-of-motion, image resolution and camera relative pose. Along with the above, the Bayesian imaging value computed over a planning horizon also depends on the target's predicted trajectory and the associated covariance matrices, which are derived from the distributed state estimator. The imaging performance obtained is also dependent on the length of the planning horizon. The proposed approach results in optimized PTZ parameters that are temporally smooth.

The approach includes a dynamic target weighting scheme. In the example, we demonstrate the utility of this feature in two ways. First, the importance of a target increases as the target approaches the exit, to help ensure that all targets are imaged at least once. Second, the weight at any time instant for a target is dependent on the image quality previously acquired for that target; therefore, subsequent images of each target receive little value unless a better quality image is expected to be acquired.

The method proposed herein allows all agents to optimize simultaneously. The global optimization problem is formulated as a *potential game* with the global objective decoupled into smaller local problems with *aligned* local objectives. A *Lagrangian* consensus algorithm is used to perform distributed, co-operative and simultaneous optimization across all cameras in the network. This formulation enables use of existing convergence proofs from the game theory literature. Convergence of the Lagrange multiplier vector is achieved by consensus methods and of the camera PTZ parameters is achieved by a modified flooding algorithm. Future research could explore probabilistic communication schemes [71] to decrease communication loading and latency issues.

It is possible to design alternative optimization methods that use a combination of parallel and sequential processing. Graph partitioning [72] is a branch of optimization that decouples an existing communication graph into smaller subgraphs, often dynamically, subject to problem constraints. A property implicit to a visual sensing application is that the communication graph and the vision graph [29] are often different. This occurs since every camera in the network may not obtain a measurement of every target in the area, which often results in cameras [33] 'naive' with respect to certain targets. Using vision graph discovery methods to design a set of constraints, the existing communication graph G_c could be decoupled into smaller subgraphs G^j, on the basis of measurements on target T^j. Agents that are nodes of subgraph G^j may optimize

sequentially, while subgraph G^j optimizes in parallel with respect to other subgraphs in the network. This topic is largely unexplored and could provide beneficial results, especially in scenarios where a network of cameras is assigned to survey a large area with camera visibility constraints.

The method proposed herein can be extended to locally convex discontinuous functions using the subgradient-based distributed constrained optimization approaches described in [63, 64]. Though preliminary work on modifying the approach therein for application to a visual sensing problem was done in [73], further research on the method is required.

Design of alternative imaging value functions and constraints that are dependent on advanced image processing techniques to aid target gait [74], gesture [75], and activity [76] recognition is another interesting topic. As discussed in Remark 2, various alternative designs are possible for $M > 1$. For example, a sequence of high rate images could be taken with various focal lengths allowing a complete reformulation of the trade-off between risk and image quality. Implementation on a Camera Network test-bed is also of interest and in process.

Fault-tolerant systems are designed to be operational in spite of a reduction in throughput or an increase in response time in the event of some partial failure. Evaluation of fault tolerance could be of vital importance for self configuring camera networks deployed for surveillance applications.

The visual sensing problem is inherently non-convex (refer Fig. 5.1), and thus the solution obtained is only locally optimal. We avoid cameras from reaching domains of attraction of different local optima by ensuring frequent communication between cameras by setting an arbitrary rate of communication between cameras in relation with the per camera local optimization iteration counts. Finding a relation between the

camera communication rate, the local optimization iteration count, and the planning horizon length is an interesting problem and worth investigating since it may provide a per-application analysis of computation vs. communication cost.

To aid computation of feasible solutions, we initiate optimization iterations at every camera with a wide FoV. Other such relaxation techniques to perform constraint relaxation are worth researching.

The method proposed herein can be extended to locally convex discontinuous functions using the subgradient-based distributed constrained optimization approaches described in [63, 64]. Though preliminary work on modifying the approach therein for application to a visual sensing problem was done in [73], further research on the method is required.

Design of alternative imaging value functions and constraints that are dependent on advanced image processing techniques to aid target gait [74], gesture [75], and activity [76] recognition is another topic worth pursuing.

Bibliography

[1] Hjelmas, E. and Low, B. K., "Face Detection: A Survey," in *Computer Vision and Image Understanding*, vol. 83, pp. 236–274, Sep. 2001.

[2] Zhao, W. and Chellappa, R. and Phillips, P. J. and Rosenfeld, A., "Face Recognition: A Literature Survey," in *ACM Computing Surveys*, vol. 35, pp. 399–458, Dec. 2003.

[3] Li, N. and Marden, J. R., "Designing Games for Distributed Optimization," in *Conf. on Decision Control*, pp. 2434–2440, Dec. 2011.

[4] Garcia, C. E. and Prett, D. M. and Morari, M., "Model Predictive Control: Theory and Practice - A survey," in *Automatica*, vol. 25, pp. 335 – 348, May. 1989.

[5] Aloimonos, Y. and Weiss, I. and Bandyopadhyay, A., "Active Vision," in *Intl. Jour. of Computer Vision*, no. 4, pp. 333–356, 1988.

[6] Bajcsy, R., "Active Perception," in *Proc. of the IEEE*, vol. 76, pp. 996–1005, Aug. 1988.

[7] Ballard, D. H., "Animate Vision," in *Artificial Intelligence*, vol. 48, pp. 57–86, Feb. 1991.

[8] Song, B. and Roy-Chowdhury, A. K., "Stochastic Adaptive Tracking in a Camera Network," in *IEEE Intl. Conf. on Computer Vision*, pp. 1–8, Oct. 2007.

[9] Q. Cai and J. K. Aggarwal, "Automatic Tracking of Human Motion in Indoor Scenes Across Multiple Synchronized Video Streams," in *IEEE Intl. Conf. on Computer Vision*, pp. 356–362, Jan. 1998.

[10] Khan, S. and Javed, O. and Rasheed, Z. and Shah, M., "Human Tracking in Multiple Cameras," in *IEEE Intl. Conf. on Computer Vision*, vol. 1, pp. 331–336, Jul. 2001.

[11] Li, Y. and Bhanu, B., "Utility-based Dynamic Camera Assignment and Hand-off in a Video Network," in *IEEE/ACM Intl. Conf. on Distributed Smart Cameras*, pp. 1–9, Sep. 2008.

[12] Makris, D. and Ellis, T. and Black, J., "Bridging the Gap Between Cameras," in *IEEE Conf. on Computer Vision and Pattern Recognition*, vol. 2, pp. 205–210, Jul. 2004.

[13] Tieu, K. and Dalley, G. and Grimson, W. E. L., "Inference of Non-Overlapping Camera Network Topology by Measuring Statistical Dependence," in *IEEE Intl. Conf. on Computer Vision*, vol. 2, pp. 1842–1849, Oct. 2005.

[14] Stancil, B. A. and Zhang, C. and Chen, T., "Active Multicamera Networks: From Rendering to Surveillance," in *IEEE Jour. of Selected Topics in Signal Processing*, vol. 2, pp. 597–605, Aug. 2008.

[15] Medeiros, H. and Park, J. and Kak, A., "Distributed Object Tracking Using a Cluster-Based Kalman Filter in Wireless Camera Networks," in *IEEE Jour. of Selected Topics in Signal Processing*, vol. 2, pp. 448–463, Aug. 2008.

[16] Qureshi, F. Z. and Terzopoulos, D., "Surveillance in Virtual Reality: System Design and Multi-Camera Control," in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1–8, Jun. 2007.

[17] Vincent, T. L., "Game Theory as a Design Tool," in *Jour. of Mechanisms, Transmissions and Automation in Design*, vol. 105, pp. 165–170, Jun. 1983.

[18] Arslan, G. and Shamma, J. S., "Distributed Vehicle-Target Assignment Using Learning in Games," in *Conf. on Decision Control*, pp. 2837–2842, Dec. 2006.

[19] Arslan, G. and Marden, J. and Shamma, J., "Autonomous Vehicle-Target Assignment: A Game-Theoretical Formulation," in *ASME Jour. of Dynamic Systems, Measurement and Control*, vol. 129, pp. 584–596, Sep. 2007.

[20] Soto, C. and Song, B. and Roy-Chowdhury, A. K., "Distributed Multi-target Tracking in a Self-configuring Camera Network," in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1486–1493, Jun. 2009.

[21] Morye, A. A. and Ding, C. and Song, B. and Roy-Chowdhury, A. K. and Farrell, J. A., "Optimized Imaging and Target Tracking within a Distributed Camera Network," in *American Control Conf.*, pp. 474–480, Jun. 2011.

[22] Ding, C. and Song, B. and Morye, A. A. and Farrell, J. A. and Roy-Chowdhury, A. K., "Collaborative Sensing in a Distributed PTZ Camera Network," in *IEEE Trans. on Image Processing*, pp. 3282–3295, Jul. 2012.

[23] Morye, A. A. and Ding, C. and Roy-Chowdhury, A. K. and Farrell, J. A., "Constrained Optimization for Opportunistic Distributed Visual Sensing," in *American Control Conf.*, pp. 6294–6301, Jun. 2013.

[24] Stillman, S. and Tanawongsuwan, T., "Tracking Multiple People with Multiple Cameras," in *Intl. Conf. on Audio-and Video-based Biometric Person Authentication*, pp. 1–6, Mar. 1999.

[25] Ellis, T. J. and Makris, D. and Black, J. K., "Learning a Multi-camera Topology," in *IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, pp. 165–171, Oct. 2003.

[26] Ren, W. and Beard, R. W., "Consensus Seeking in Multiagent Systems under Dynamically Changing Interaction Topologies," in *IEEE Trans. on Automatic Control*, vol. 50, pp. 655–661, May 2005.

[27] Xiao, F. and Wang, L., "Asynchronous Consensus in Continuous-Time Multi-Agent Systems With Switching Topology and Time-Varying Delays," in *IEEE Trans. on Automatic Control*, vol. 53, pp. 1804–1816, Sep. 2008.

[28] Tron, R. and Vidal, R., "Distributed Computer Vision Algorithms," in *IEEE Signal Processing Magazine*, vol. 28, pp. 32–45, May 2011.

[29] Tron, R. and Vidal, R., "Distributed Computer Vision Algorithms through Distributed Averaging," in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 57–63, Jun. 2011.

[30] Taj, M. and Cavallaro, A., "Distributed and Decentralized Multicamera Tracking," in *IEEE Signal Processing Magazine*, vol. 28, pp. 46–58, May 2011.

[31] Song, B. and Ding, C. and Kamal, A. T. and Farrell, J. A. and Roy-Chowdhury, A. K., "Distributed Camera Networks," in *IEEE Signal Processing Magazine*, pp. 20–31, May. 2011.

[32] Kamal, A. T. and Farrell, J. A. and Roy-Chowdhury, A. K., "Information Weighted Consensus," in *Conf. on Decision Control*, Dec. 2012.

[33] Kamal, A. T. and Farrell, J. A. and Roy-Chowdhury, A. K., "Information Weighted Consensus Filters and their Application in Distributed Camera Networks," in *IEEE Transactions on Automatic Control*, in press.

[34] Moreau, L., "Stability of Multiagent Systems with Time-dependent Communication Links," in *IEEE Trans. on Automatic Control*, vol. 50, pp. 169–182, Feb. 2005.

[35] Macready, W. and Wolpert, D., "Distributed Constrained Optimization," in *Intl. Conf. on Complex Systems*, vol. Bar-Yam (Ed.), pp. 1–12, Perseus books, 2004.

[36] Mittal, A. and Davis, L. S., "A General Method for Sensor Planning in Multi-Sensor Systems: Extension to Random Occlusion," in *Intl. Jour. of Computer Vision*, vol. 76, pp. 31–52, Jan. 2008.

[37] Piciarelli, C. and Micheloni, C. and Foresti, G. L., "PTZ Camera Network Reconfiguration," in *IEEE/ACM Intl. Conf. on Distributed Smart Cameras*, pp. 1–7, Sep. 2009.

[38] Qureshi, F. Z. and Terzopoulos, D., "Planning ahead for PTZ camera assignment and handoff," in *IEEE/ACM Intl. Conf. on Distributed Smart Cameras*, pp. 1–8, Sep. 2009.

[39] Zhao, J. and Cheung, S. and Nguyen, T., "Optimal Camera Network Configurations for Visual Tagging," in *IEEE Jour. of Selected Topics in Signal Processing*, vol. 2, pp. 464–479, Aug. 2009.

[40] Erdem, M. and Sclaroff, S., "Automated Camera Layout to Satisfy Task-specific and Floor Plan-specific Coverage Requirements," in *Computer Vision and Image Understanding*, vol. 103, pp. 156–169, Sep. 2006.

[41] Moon, T. K., "The Expectation-Maximization Algorithm," in *IEEE Signal Processing Magazine*, pp. 47–60, Nov. 1996.

[42] Du, W. and Piater, J., "Multi-camera People Tracking by Collaborative Particle Filters and Principal Axis-based Integration," in *Asian Conf. on Computer Vision*, pp. 365–374, Nov. 2007.

[43] Khan, S. and Shah, M., "A Multiview Approach to Tracking People in Crowded Scenes Using a Planar Homography Constraint," in *Euro. Conf. on Computer Vision*, pp. 133–146, May 2006.

[44] Wang, Y. and Boyd, S., "Fast Model Predictive Control Using Online Optimization," in *IEEE Trans. on Control Systems Technology*, vol. 18, pp. 267 – 278, Mar. 2010.

[45] Alessio, A. and Bemporad, A., "A Survey on Explicit Model Predictive Control," in *Lecture Notes in Control and Information Sciences*, vol. 384, pp. 345 – 369, 2006.

[46] Mayne, D. Q. and Rawlings, J. B. and Rao, C. V. and Scokaert, P. O. M., "Constrained Model Predictive Control: Stability and Optimality," in *Automatica*, vol. 36, pp. 789 – 814, Jun. 2000.

[47] Acar, L., "Some Examples for the Decentralized Receding Horizon Control," in *Conf. on Decision Control*, vol. 2, pp. 1356–1359, Dec. 1992.

[48] Georges, G., "Decentralized Adaptive Control for a Water Distribution System," in *IEEE Intl. Conf. on Control Applications*, vol. 2, pp. 1411–1416, Aug. 1994.

[49] Sawadogo, S. and Faye, R.M. and Malaterre, P.O. and Mora-Camino, F., "Decentralized vs. Model Predictive Control of an Industrial Glass Tube Manufacturing Process," in *IEEE Trans. on Systems, Man and Cybernetics*, vol. 4, pp. 3880–3884, Oct. 1998.

[50] Ochs, S. and Engell, S. and Draeger, A., "Decentralized vs. Model Predictive Control of an Industrial Glass Tube Manufacturing Process," in *IEEE Intl. Conf. on Control Applications*, vol. 1, pp. 16–20, Sep. 1998.

[51] El Fawal, H. and Georges, D. and Bornard,G., "Optimal Control of Complex Irrigation Systems via Decomposition-coordination and the Use of Augmented Lagrangian," in *IEEE Trans. on Systems, Man and Cybernetics*, vol. 4, pp. 3874–3879, Oct. 1998.

[52] Baglietto, M. and Parisini, T. and Zoppoli, R., "Neural Approximators and Team Theory for Dynamic Routing: A Receding-horizon Approach," in *Conf. on Decision Control*, vol. 4, pp. 3283–3288, Dec. 1999.

[53] Aicardi, M. and Casalino, G. and Minciardi, R. and Zoppoli, R., "On the Existence of Stationary Optimal Receding-Horizon Strategies for Dynamic Teams with Common Past Information Structures," in *IEEE Trans. on Automatic Control*, vol. 37, pp. 1767–1771, Nov. 1992.

[54] Camponogara, E. and Jia, D. and Krogh, B. H. and Talukdar, S., "Distributed Model Predictive Control," in *Control Systems Magazine, IEEE*, vol. 22, pp. 44 – 52, Feb. 2002.

[55] Choate, A. G., "Telecentric, Parfocal, Multiple Magnification Optical System for Video-inspection Apparatus," in *United States Patent*, no. 5668665, pp. 1–9, Sep. 1997.

[56] Song, B. and Soto, C. and Roy-Chowdhury, A. K. and Farrell, J. A., "Decentralized Camera Network Control using Game Theory," in *IEEE/ACM Intl. Conf. on Distributed Smart Cameras*, pp. 1–8, Sep. 2008.

[57] Olfati-Saber, R. and Sandell, N. F., "Distributed Tracking in Sensor Networks with Limited Sensing Range," in *American Control Conf.*, pp. 3157–3162, Jun. 2008.

[58] Olfati-Saber, R., "Kalman-Consensus Filter: Optimality, Stability, and Performance," in *Conf. on Decision Control*, pp. 7036–7042, Dec. 2009.

[59] Boyd, S. and Vandenberghe, L., "Convex Optimization," pp. 241–246, Cambridge University Press, 2004.

[60] Monderer, D. and Shapley, L. S., "Potential Games," in *Games and Economic Behavior*, vol. 14, pp. 124–143, 1996.

[61] Farrell, J. A., "Aided Navigation: GPS with High Rate Sensors," pp. 72–75, Mcgraw Hill, 2008.

[62] Trucco, E. and Verri, A., "Introductory Techniques for 3-D Computer Vision," pp. 26–27, Prentice Hall, 1998.

[63] Nedic, A. and Ozdaglar, A. and Parrilo, P. A., "Constrained Consensus and Optimization in Multi-Agent Networks," in *IEEE Trans. on Automatic Control*, vol. 55, pp. 922–938, Apr. 2010.

[64] Zhu, M. and Martinez, S., "On Distributed Optimization under Inequality Constraints via Lagrangian Primal-Dual Methods," in *American Control Conf.*, pp. 4863–4868, Jul. 2010.

[65] Olfati-Saber, R. and Fax, J. A. and Murray, R. M., "Consensus and Cooperation in Networked Multi-Agent Systems," in *Proc. of the IEEE*, vol. 95, pp. 215–233, Jan. 2007.

[66] Tanenbaum, A. S. and Wetherall, D. J., "Computer Networks (5th Edition)," pp. 368–370, Pearson Education Inc., 2010.

[67] Boyd, S. and Vandenberghe, L., "Convex Optimization," pp. 609–615, Cambridge University Press, 2004.

[68] Ibaraki, T. and Nonobe, K. and Yagiura F., "Metaheuristics: Progress as Real Problem Solvers," pp. 1–28, Springer Science + Business Media, Inc., 2005.

[69] van Laarhoven, P. J. and Aarts, E. H., "Simulated Annealing: Theory and Applications," pp. 1–6, Kluver Academic Publishers, 1987.

[70] Goldberg, D. E. and Holland, J. H., "Genetic Algorithms and Machine Learning," vol. 3, pp. 95–99, Kluwer Academic Publishers-Plenum Publishers, Oct. 1988.

[71] Sasson, Y. and Cavin, D. and Schiper, A., "Probabilistic broadcast for flooding in wireless mobile ad hoc networks," in *Wireless Communications and Networking*, vol. 2, pp. 1124–1130, Mar. 2003.

[72] Lee, L. and Grimson, W. E. L., "Continuous Graph Partitioning for Camera Network Surveillance," in *IFAC Workshop on Distributed Estimation and Control in Networked Systems*, vol. 3, pp. 228–233, Sep. 2012.

[73] Morye, A. A. and Ding, C. and Roy-Chowdhury, A. K. and Farrell, J. A., "Cooperative, Opportunistic Imaging within a Bayesian Distributed Constrained Optimization Framework," in *BCOE Research, Bourns College of Engineering, UC Riverside*, Sep. 2012.

[74] Lee, L. and Grimson, W. E. L., "Gait Analysis for Recognition and Classification," in *IEEE Intl. Conf. on n Automatic Face and Gesture Recognition*, pp. 148–155, May 2002.

[75] Wu, Y. and Huang, T. S., "Vision-Based Gesture Recognition: A Review," pp. 103–115, Springer-Verlag Berlin Heidelberg, 1999.

[76] Bodor R. and Jackson B. and Papanikolopoulos N., "Vision-based Human Tracking and Activity Recognition," in *Mediterranean Conf. on Control and Automation*, pp. 18–20, Jun. 2003.