AFRL-AFOSR-VA-TR-2022-0065

Optimizing Systems with Conflicting Objectives Competing for a Limited Resource - Resubmission

Mittelmann, Hans
ARIZONA STATE UNIVERSITY
660 S MILL AVE STE 312
TEMPE, AZ, 85281
USA

12/14/2021
Final Technical Report

Air Force Research Laboratory
Air Force Office of Scientific Research
Arlington, Virginia 22203
Air Force Materiel Command

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From - To)*
14-12-2021	Final	01 Jan 2019 - 31 Dec 2021

4. TITLE AND SUBTITLE
Optimizing Systems with Conflicting Objectives Competing for a Limited Resource - Resubmission

5a. CONTRACT NUMBER

5b. GRANT NUMBER
FA9550-19-1-0070

5c. PROGRAM ELEMENT NUMBER
61102F

6. AUTHOR(S)
Hans Mittelmann

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
ARIZONA STATE UNIVERSITY
660 S MILL AVE STE 312
TEMPE, AZ 85281
USA

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AF Office of Scientific Research
875 N. Randolph St. Room 3112
Arlington, VA 22203

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL/AFOSR RTA2

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
AFRL-AFOSR-VA-TR-2022-0065

12. DISTRIBUTION/AVAILABILITY STATEMENT
A Distribution Unlimited: PB Public Release

13. SUPPLEMENTARY NOTES

14. ABSTRACT
Decentralized and distributed autonomous sensing and control methods for networked sensor systems have many applications in surveillance, Internet of Things (IoT), autonomous cars, and UAV swarms. These decentralized autonomy methods are especially challenging when the network connecting the sensors is time varying. Moreover, when the network is large with 10s or even 100s of sensors connected, decision making for sensor resource management (e.g., decisions on sensor mobility - sensors mounted on UAVs) becomes computationally intensive, in fact, the complexity is exponential in the decision space and the number of sensors. To address these challenges, we developed an optimization framework called COLRO to optimize the limited sensing resources in a time-varying networked sensor system for a target tracking application while minimizing the computational effort. We presented the results of this research

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			WARREN ADAMS
U	U	U	UU	55	**19b. TELEPHONE NUMBER** *(Include area code)* 00000000

Standard Form 298 (Rev.8/98)
Prescribed by ANSI Std. Z39.18

Final Report on AFOSR grant FA9550-19-1-0070

PI Hans Mittelmann, Arizona State University
Co-I Shankarachary Ragi, South Dakota School of Mines and Technology

This report is divided into four sections, as explained below, where we summarize the main results obtained and refer to the papers included for further details.

I. Competing Objective Limited Resource Optimization (COLRO) framework for networked swarm systems

Decentralized and distributed autonomous sensing and control methods for networked sensor systems have many applications in surveillance, Internet of Things (IoT), autonomous cars, and UAV swarms. These decentralized autonomy methods are especially challenging when the network connecting the sensors is time varying. Moreover, when the network is large with 10s or even 100s of sensors connected, decision making for sensor resource management (e.g., decisions on sensor mobility - sensors mounted on UAVs) becomes computationally intensive, in fact, the complexity is exponential in the decision space and the number of sensors. To address these challenges, we developed an optimization framework called COLRO to optimize the limited sensing resources in a time-varying networked sensor system for a target tracking application while minimizing the computational effort. We presented the results of this research at National Aerospace and Electronics Conference 2019 (NAECON 2019) [1]. Building on these results, we further incorporated explicit optimization of the network graph connecting the UAVs to maximize the combined tracking and computing performance in the context of multi-target tracking. We compared the performance of our methods against a centralized optimization approach, where all the decision variables are optimized together providing the best achievable performance. This numerical study allowed us to quantize the performance of our decentralized approaches benchmarked against the centralized approaches. Furthermore, we proved that the optimal solution from our COLRO optimization framework is *pareto-optimal*. The performance of our decentralized UAV control methods is only marginally inferior to the centralized approaches in terms of tracking performance, while significantly outperforming the centralized approach in terms of computational efficiency. Some of the results from this work was published in the Algorithms journal in the special issue Algorithms in Stochastic Models [2] (guest-edited by Co-I Ragi).

II. Monte-Carlo tree search methods for solving long horizon optimal control problems and convergence analysis

Long-horizon optimal control problems appear naturally in robotics, advanced manufacturing, and economics, especially in applications requiring decision making in stochastic environments. Often these problems are solved via dynamic programming (DP). DP problems are notorious for their computational complexity and require approximation approaches to make them tractable referred to as approximate dynamic programs (ADPs). In this project, the goal is to develop a class of ADP methods called Monte-Carlo Tree Search (MCTS) approaches to solve long (but finite) horizon optimal control problems formulated as DPs. These MCTS methods enable smooth trade-off between the approximation error and the computational intensity. In the first phase of this research, we performed convergence analysis to show these MCTS methods converge in probability to the true cost function using variants of law of large numbers and Chebyshev's theorem. Particularly, we proved that the convergence of two kinds of MCTS methods: a) tree-branching methods, where the state possibilities are evolved as a tree; b) non-overlapping tree branching, where the state possibilities are evolved with no shared nodes between the branches except for the root/parent node. In the second phase, we demonstrated the effectiveness of these MCTS methods in two case studies: linear quadratic control problems, UAV motion control problems. Part of these results were recently presented at the 4th IEEE Conference on Control Technology and Applications (CCTA 2020) [3]. Since the possibilities grow exponentially with the planning horizon in MCTS methods, we developed pruning strategies (with polynomial-time complexity) and studied their convergence. All the results from this work were recently published in the journal IEEE Control Systems Letters [4].

III. Decentralized Data Fusion in Networked Sensor Systems

Autonomous and adaptive sensing has applications such as target tracking, surveillance, and autonomous car navigation. Particularly, target tracking via adaptive sensing is becoming increasingly important in autonomous car industry for accurate pedestrian detection and tracking. Sensors such as RADAR, LIDAR, optical sensors, thermal sensors are typically used to measure the target state including its position, velocity, and acceleration. Target tracking with multiple sensors was studied in the past, where a central fusion node is typically responsible for making sensing decisions (e.g., sensor location - assuming sensor mounted on a moving vehicle) for all the sensors combined. Clearly, sensing decisions optimized for all the sensors combined provides the best target tracking performance as these decisions are coupled via sensor data fusion. The main drawback with these centralized decision making methods is that they are computationally intensive as the computational complexity is exponential in the decision space and the number of sensors. To address this challenge, we developed a decentralized autonomous sensing method over a networked sensor system for a target tracking application. Specifically, we extended an existing approach called *average*

consensus algorithm to perform decentralized data fusion while tracking a moving target. Our preliminary studied demonstrated that our methods significantly outperform the standard decentralized Bayesian data fusion approaches. These results were recently presented at the IEEE 10th Annual Computing and Communication Workshop and Conference (IEEE CCWC 2020) [5].

IV. Waveform optimization for joint radar communications performance

In the past, we have developed optimization frameworks for waveform design in joint radar communications systems. Particularly, we developed methods to optimize the spectral shape of the radar waveform while minimizing the interference from communications systems (radar and wireless communications coexist in the same spectrum) and maximizing the radar target tracking performance. In our final report to AFOSR in 2018, we summarized the results from this research. This research helped us establish a collaboration with the Bliss Lab at Arizona State University. Building on this research, we addressed a new set of challenges associated with waveform design for joint radar communications systems for "long-term" performance. Specifically, we developed methods to design waveforms while accounting for their impact on the long-term performance. Particularly, we posed the waveform co-design for radar and communications as a *partially observable Markov decision process* (POMDP). POMDP is a stochastic control framework useful in modeling stochastic systems with Markovian dynamics and decision making. As POMDPs have PSPACE-Complete computational complexity, we solved the waveform co-design problem posed as POMDP using an existing approximate dynamic programming approach called *nominal belief-state optimization* (NBO). The NBO method allowed us to obtain the optimal (or suboptimal) waveform parameters in near real-time. Moreover, this POMDP-based waveform co-design approach proved to superior to the existing myopic waveform design methods in terms of the radar and communications data rates. These results were published in the proceedings of the 54th Asilomar Conference on Signals, Systems and Computers (Asilomar 2020) [6]. Furthermore, we extended the MCTS-based ADP schemes, discussed in Section II, to solve the co-design problem. We also studied the impact of pruning strategies (also discussed in Section II) on MCTS methods in the waveform co-design context. We considered challenges including dealing with clutter, dynamic communications data rate requirements, and also extended decentralized decision making framework COLRO (discussed in Section I) for waveform co-design in large and heterogeneous joint radar-communications systems. The complete set of results from this work is submitted to the journal IEEE Transactions on Aerospace and Electronic Systems [7], which is currently under review.

References

[1] S. Ragi, S. Dey, A. Ali, and H. D. Mittelmann, "Competing objective optimization in networked swarm systems," in Proceedings of the National Aerospace & Electronics Conference 2019 (NAECON 2019), Dayton, OH, July 15--19, 2019, pp. 88--91.

[2] A. Ali, H. D. Mittelmann, S. Ragi, "UAV Formation Shape Control via Decentralized Markov Decision Processes" *Algorithms*, special issue on *Algorithms in Stochastic Models*, vol. 14, no. 3, Mar 2021.

[3] S. Ragi and H. D. Mittelmann, "Random-Sampling Multipath Hypothesis Propagation for Cost Approximation in Long-Horizon Optimal Control," in Proceedings of the 2020 IEEE Conference on Control Technology and Applications (CCTA), Montreal, Canada, August 24--26, 2020, pp. 14--18.

[4] S. Ragi, H. D. Mittelmann, "Random-sampling Monte-Carlo tree search methods for cost approximation in long-horizon optimal control," IEEE Control Systems Letters (L-CSS), vol. 5, no. 5, pp. 1759--1764, November 2021.

[5] M. Azam, S. Dey, H. D. Mittelmann, and S. Ragi, "Average consensus-based data fusion in networked sensor systems for target tracking," in 2020 10th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, Jan 06--08, 2020, pp. 964--969.

[6] S. A. Doly, A. Chiriyath, H. D. Mittelmann, D. W. Bliss, and S. Ragi, "A decision theoretic approach for waveform design in joint radar communications applications," in Proceedings of the 54th Asilomar Conference on Signals, Systems and Computers (Asilomar 2020), Pacific Grove, CA, Nov 01--04, 2020, pp. 6--11.

[7] S. A. Doly, A. Chiriyath, H. D. Mittelmann, D. W. Bliss, and S. Ragi, "Waveform codesign for radar-communications spectral coexistence via dynamic programming," IEEE Transactions on Aerospace and Electronic Systems, submitted (under review).

Next follows the copies of all the references [1-7]

Competing Objective Optimization in Networked Swarm Systems

Shankarachary Ragi, Shawon Dey, Azam Md Ali
Department of Electrical Engineering
South Dakota School of Mines and Technology
Rapid City, SD 57701, USA.
shankarachary.ragi@sdsmt.edu,
{shawon.dey, mdali.azam}@mines.sdsmt.edu

Hans D. Mittelmann
School of Mathematical and Statistical Sciences
Arizona State University
Tempe, AZ 85287, USA.
mittelmann@asu.edu

***Abstract*—In this paper, we develop a decentralized collaborative sensing algorithm where the sensors are located on-board autonomous unmanned aerial vehicles. We develop this algorithm in the context of a target tracking application, where the objective is to maximize the tracking performance measured by the mean-squared error between the target state estimate and the ground truth. The tracking performance depends on the quality of the target measurements made at the sensors, which depends on the relative location of the sensors with respect to the target. Our goal is to control the motion of the swarm of vehicles with on-board sensors to maximize target tracking performance. Each sensor (on-board the vehicle) generates local noisy measurements of the target location, and the sensors maintain and update target state estimates via Bayesian data fusion rules using local measurements and the information received from neighboring sensors. The quality of the data fusion depends on the network graph over which the sensors exchange information, and this determines the overall target tracking performance. We also assume that each sensor is powered by a limited energy source; which we assume is drained by how frequently sensors exchange information. The goal is to optimize the collective motion of the vehicles/sensors (also determines the network graph connectivity) such that the mean-squared target tracking error and the network energy costs are jointly minimized. This problem belongs to a class of hard optimization problems called *conflicting objective limited resource optimization* (COLRO). We develop a fast heuristic algorithm, using dynamic programming principles, to solve this COLRO problem in real-time.**

***Index Terms*—Swarm systems, target tracking, competing objectives, sensor network**

I. INTRODUCTION

There is a growing interest in decentralized and distributed autonomous sensing methods [1], [2], where the network connecting the sensors may be time-varying. With increasing number of sensor and surveillance systems in public places, there is a need for decentralized methods to track moving targets (e.g. movement of an intruder, movement of enemy tanks in battle field) with a network of sensors. However, the decentralized collaborative sensing in a wireless multi-sensor network is a challenging problem, especially when there are network energy costs involved. Since the battery-powered sensor nodes have limited energy, there is a need for methods that can trade off between the target tracking performance and the energy costs of acquiring the measurements and sharing them (with peers) over a network. If a distributed set of autonomous vehicles are connected via a wireless network (vehicle is considered a wireless node), due to the movement of the vehicles, the links in the network graph may form and break as the relative distances between the nodes change over time, thus leading to a time-varying graph. There is a growing interest in controlling the motion of the vehicles with on-board sensors for various applications such as formation control [3], [4], target tracking [5]. With this motivation, we develop a stochastic decision optimization framework to control the motion of a swarm of autonomous vehicles (e.g., unmanned aerial systems) to track a moving object, where the swarm is connected via a wireless network.

As swarm-based systems tend to have a large number of vehicles, optimizing each motion control variable may lead to computationally expensive optimization problems; instead, we optimize the centroid location of the swarm. Once a desired centroid and network graph are obtained, the vehicles may choose one of infinitely many paths to achieve the desired centroid and the network graph.

As mentioned earlier, we also optimize the network graph of the swarm, which determines how well the sensors (on-board the vehicles) fuse their local sensor measurements with the measurements received from the neighboring sensors, as depicted in Figure 1. Clearly, the objectives of maximizing the tracking performance and minimizing the network energy costs are competing, i.e., emphasizing one objective deteriorates the other. We refer to these problems as *competing objective limited resource optimization* (COLRO) problems. In this paper, we focus on solving COLRO problems in real-time in the context of networked swarm systems.

II. PROBLEM SPECIFICATION AND APPROACH

Let k represent the time index. A target moves on a 2-D plane according to the *constant velocity* model [6]. Let χ_k represent the target state at time k, which includes its location, velocity, and acceleration. According to the *constant velocity*

This work was supported in part by Air Force Office of Scientific Research under grant FA9550-19-1-0070.

978-1-7281-1416-3/19/$31.00 ©2019 IEEE

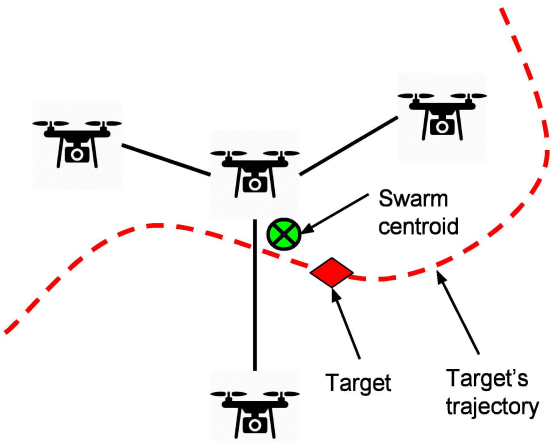


Fig. 1. Autonomous vehicle swarm tracking a target while jointly minimizing the tracking error and the energy consumption.

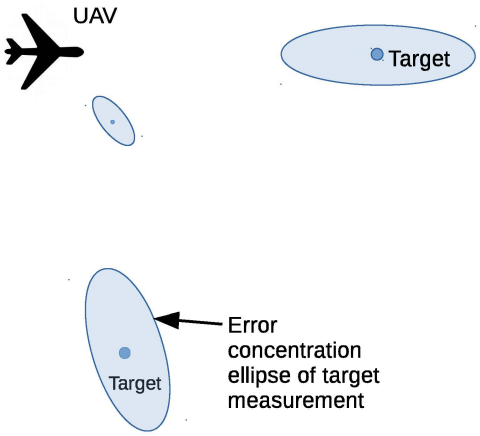


Fig. 2. Measurement error model.

model, the target state evolves according to the following equation:

$$\chi_{k+1} = F\chi_k + v_k,\ v_k \sim \mathcal{N}(0, Q)$$

where F is the state-transition matrix, v_k is the process noise, which is drawn from a zero-mean normal distribution with the co-variance matrix Q. Let n represent the number of vehicles in the swarm. We assume that the each vehicle in the swarm has an on-board sensor that generates noisy measurements of the target's location. The vehicles in the network are connected by a time-varying graph, represented by $\mathcal{G}_k$, where

$$\mathcal{G}_k = \begin{bmatrix} 0 & a_{12} & \dots & a_{1n} \\ a_{21} & 0 & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & 0 \end{bmatrix}$$

$a_{ij,i\neq j} = 1$ represents the ability of the sensors i and j to exchange measurements for data fusion at time k, and $a_{ij,i\neq j} = 0$ otherwise. Let C_k represent the centroid of the swarm at time k. We assume that the presence of a link between two sensors at time k lets the sensors exchange local measurements (generated at time k) for data fusion purpose. The sensors on-board the vehicles generate noisy measurements of the target positions in each time step. We use the standard Kalman filter to track the target state. Since the swarm is a decentralized system, each vehicle runs a local target tracking algorithm (Kalman filter), which is updated using the measurements generated locally and received from the neighboring nodes, where the measurement at ith sensor is given by:

$$z_k^i = H_{\text{pos}}\chi_k + w_k,\ w_k \sim \mathcal{N}(0, R_k(s_k^i, \chi_k)), \quad (1)$$

where H_{pos} is a matrix that captures just the position information in the target state vector χ_k, w_k is the measurement noise, and s_k^i is the position of the ith vehicle. We assume that the angular uncertainty is better than the range uncertainty; which is captured in the definition of the covariance matrix R_k, also captured in Figure 2. The state of the tracking algorithm is given by (ξ_k^i, P_k^i), where ξ_k^i and P_k^i represent the mean vector and the error covariance matrix corresponding to target state estimation at the ith sensor.

Let $f_{track}(\mathcal{G}_k, C_k)$ and $f_{energy}(\mathcal{G}_k, C_k)$ be functions representing target tracking error and the energy consumed respectively from sensor i's perspective, as defined below:

$$\begin{aligned} f_{track}(\mathcal{G}_k, C_k) &= \left\|\chi_k - \xi_k^i\right\|_2^2 \\ f_{energy}(\mathcal{G}_k, C_k) &= \sum_i \sum_j \mathcal{G}_k(i,j)\ \text{linkcost}(i,j) \end{aligned} \quad (2)$$

where $\text{linkcost}(i,j)$ represents the cost of using the link between sensors i and j for data fusion purpose. For simplicity, we assume the link cost is a constant and does not depend on i and j. As this is a decentralized system, each sensor in the system evaluates these functions using their own local target state estimates.

The goal of this study is to optimize the variables $\mathcal{G}_k$ and C_k such that the objectives f_{track} and f_{energy} are jointly minimized over a long time horizon H. In other words, the goal boils down to solving a COLRO problem as described below:

$$\min_{\mathcal{G}_k, C_k, k=0,..,H-1} \sum_{k=0}^{H-1} \mathrm{E}[p f_{track}(\mathcal{G}_k, C_k) + (1-p) f_{energy}(\mathcal{G}_k, C_k)] \quad (3)$$

where $\mathrm{E}[\cdot]$ is the expectation, and p is a weighting parameter. The above optimization problem resembles a *long-horizon optimal control* problem. These problems are notorious for high computational complexities, especially due to the presence of $\mathrm{E}[\cdot]$, which is hard to evaluate explicitly. To overcome these computational issues, a class of approximation techniques called *approximate dynamic programming* (ADP) approaches are used. With this motivation, we adopt an ADP approach

called *nominal belief-state optimization* (NBO) [6], which allows us to approximate the expectation making its evaluation tractable. According to the NBO approach, the expectation is approximated by assuming the "future" noise variables take *nominal* or mean values from the probability distributions they are drawn from. Since we model the noise variables as zero-mean Gaussian, the nominal values are zeros. After the approximation, the COLRO problem reduces to

$$\min_{\mathcal{G}_k, C_k, k=0,..,H-1} \sum_{k=0}^{H-1} [p\tilde{f}_{track}(\mathcal{G}_k, C_k) + (1-p)\tilde{f}_{energy}(\mathcal{G}_k, C_k)] \quad (4)$$

where $\tilde{f}_{track}$ and $\tilde{f}_{energy}$ are deterministic approximations to f_{track} and f_{energy} obtained from the NBO method. The reduced COLRO problem in Eq. 4 is highly nonlinear and non-convex, and also a mixed integer program since $\mathcal{G}_k$ contains discrete variables. We use a numerical optimization solver called *Knitro*, which allows solving mixed integer programs such as the above reduced COLRO problem.

With the NBO approach, $\tilde{f}_{track}(\mathcal{G}_k, C_k)$ is given by the trace of the error covariance matrix corresponding to the target state, which is obtained by running the Kalman filter by assuming: 1) the future process and measurement noise variables as zero; 2) the data fusion rules are applied according to the network graph state $\mathcal{G}_k$.

A. Evaluation of Optimal UAV Kinematic Controls

The decision variables $\mathcal{G}_k$ and $\mathcal{C}_k$ depend on the positions of the UAVs over time. Of course, once the optimal values for $\mathcal{G}_k$ and $\mathcal{C}_k$ are evaluated in Eq. 4, we still need to achieve the desired graph state and the desired swarm centroid by appropriately controlling the motion of the UAVs. Since the UAV kinematic control decisions depend on the optimal values of $\mathcal{G}_k$ and $\mathcal{C}_k$, we introduce a hierarchical model with two levels, where $\mathcal{G}_k$ and $\mathcal{C}_k$ are optimized in the upper level (by solving Eq. 4) and the UAV kinematic controls are optimized in the lower level according to the following artificial potential field approach.

Let $\mathcal{G}_k^*$ and $\mathcal{C}_k^*$ be the optimized network graph and the centroid location. At time k, on each UAV we apply an attractive potential field with the center at $\mathcal{C}_k^*$, another attractive potential field between UAVs i and j ($j \neq i$) if $\mathcal{G}_k^*(i,j) = 1$ and the repulsive field otherwise. These two potential fields allow the UAVs to approach the desired centroid while forming/breaking network links to achieve $\mathcal{G}_k^*$. In addition, we also apply short-range repulsive potential fields between each pair of UAVs to avoid collisions.

III. Results and Discussion

We implement the above-discussed methods in MATLAB for a scenario with three UAVs tracking a single target. We set the time horizon $H = 6$ and apply the *receding horizon control* [6] approach for planning and implementing the optimized decisions. For bench-marking, we also implement the centralized UAV motion planning approach discussed in [6]; we call this *centralized fusion approach*.

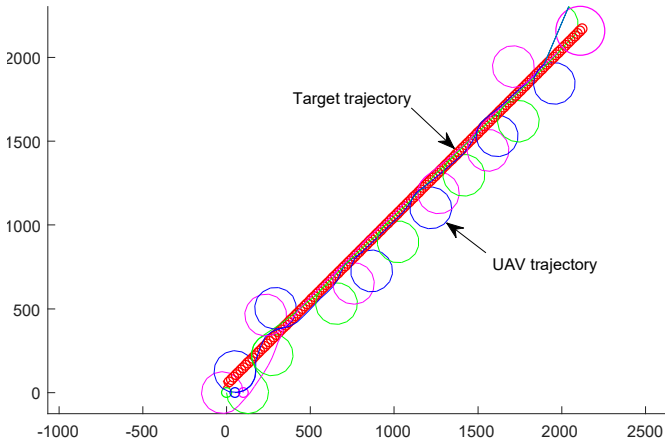


Fig. 3. Three UAVs tracking a target.

Figure 3 shows the trajectories of three UAVs tracking a target. The target and the UAVs begin their motion in the bottom-left region, and move toward the top-right region. Figures 4 and 5 show the network link status (three links for three UAVs) as a function of time for the weighting parameter in Eq. 4 set to $p = 0.2$ and $p = 0.01$ respectively. Clearly, in Figure 5, the UAVs exchange information less often compared to the scenario in Figure 4. These figures clearly demonstrate our ability to smoothly trade off between the two competing performance indices. We evaluate the normed error between the actual target location (ground truth) and the target location estimate at each sensor over time for the scenario in Figure 3. In Figure 6, we compare the performance of the above-discussed approach against the *centralized fusion approach*, which clearly shows that the tracking performance of the centralized approach is just marginally better than our COLRO-based methods discussed here. Of course, in the centralized approach, the performance with respect to the network energy costs is ignored. In other words, our approach, while slightly trading off the tracking performance, gains significantly in the performance with respect to the network energy consumption.

IV. Conclusions

In this paper, we presented a real-time heuristic approach to solve a *competing objective limited resource optimization* (COLRO) problem in the context of a networked UAV/sensors system. The objective is to optimize the motion of a swarm of UAVs (equipped with sensors) to track a moving target, while jointly minimizing the tracking error and the network energy cost. This optimization problem lead to *long horizon optimal control* problem, which is known to be computationally hard. So, we extended our previously developed approximate dynamic programming approach called *nominal belief state optimization* to solve the above COLRO problem. We tested

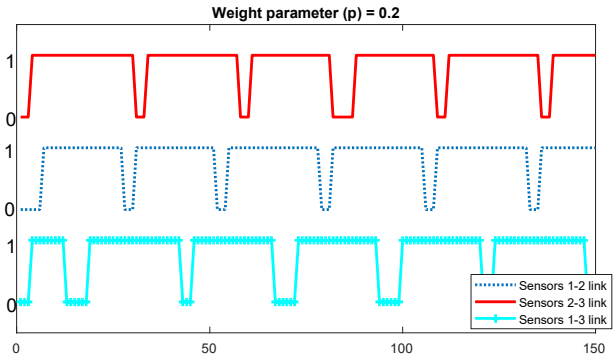


Fig. 4. Status of UAV network links over time with $p = 0.2$ (1 means active and 0 otherwise)

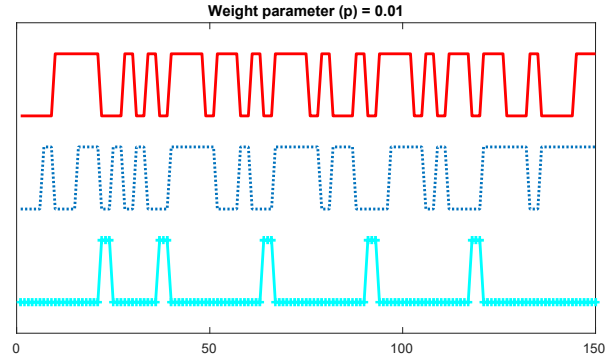


Fig. 5. Status of UAV network links over time with $p = 0.01$ (1 means active and 0 otherwise)

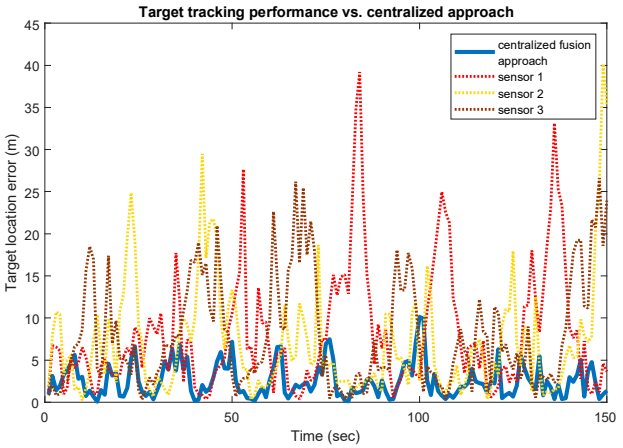


Fig. 6. Normed target location error: COLRO-based approach vs. centralized approach.

the performance of the approach in a simulated environment (implemented in MATLAB), and compared the performance of our approach with a *centralized fusion approach* (benchmark). We found our method to lose on the tracking performance only minimally compared to the centralized fusion approach, while significantly saving the network energy costs.

REFERENCES

[1] W. Xiao, C. K. Tham, and S. K. Das, "Collaborative sensing to improve information quality for target tracking in wireless sensor networks," in *2010 8th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, March 2010, pp. 99–104.

[2] M. Zhang and D. Kingston, "Time-space network based exact models for periodical monitoring routing problem," in *Proc. American Control Conf.*, Chicago, IL, July 2015, pp. 5264–5269.

[3] D. van der Walle, B. Fidan, A. Sutton, C. Yu, and B. D. O. Anderson, "Non-hierarchical UAV formation control for surveillance tasks," in *Proc. American Control Conf.*, Seattle, WA, June 2008, pp. 777–782.

[4] I. Shames, B. Fidan, and B. D. O. Anderson, "Close target reconnaissance using autonomous UAV formations," in *Proc. 47th IEEE Conf. Decision and Control*, Cancun, Mexico, Dec. 2008, pp. 1729–1734.

[5] S. A. P. Quintero, D. A. Copp, and J. P. Hespanha, "Robust UAV coordination for target tracking using output-feedback model predictive control with moving horizon estimation," in *Proc. American Control Conf.*, Chicago, IL, 2015, pp. 3758–3764.

[6] S. Ragi and E. K. P. Chong, "UAV path planning in a dynamic environment via partially observable Markov decision process," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 49, pp. 2397–2412, 2013.

algorithms

MDPI

Article

UAV Formation Shape Control via Decentralized Markov Decision Processes

Md Ali Azam [1], Hans D. Mittelmann [2] and Shankarachary Ragi [1,*]

[1] Electrical Engineering, South Dakota School of Mines and Technology, Rapid City, SD 57701, USA; azam.ete.ruet@gmail.com
[2] School of Mathematical and Statistical Sciences, Arizona State University, Tempe, AZ 85287, USA; mittelmann@asu.edu
* Correspondence: Shankarachary.Ragi@sdsmt.edu

Abstract: In this paper, we present a decentralized unmanned aerial vehicle (UAV) swarm formation control approach based on a decision theoretic approach. Specifically, we pose the UAV swarm motion control problem as a decentralized Markov decision process (Dec-MDP). Here, the goal is to drive the UAV swarm from an initial geographical region to another geographical region where the swarm must form a three-dimensional shape (e.g., surface of a sphere). As most decision-theoretic formulations suffer from the curse of dimensionality, we adapt an existing fast approximate dynamic programming method called *nominal belief-state optimization* (NBO) to approximately solve the formation control problem. We perform numerical studies in MATLAB to validate the performance of the above control algorithms.

Keywords: swarm intelligence; formation control; decentralized Markov decision process; approximate dynamic programming

Citation: Azam, M.A.; Mittelmann, H.D.; Ragi, S. UAV Formation Shape Control via Decentralized Markov Decision Processes. *Algorithms* **2021**, *14*, 91. https://doi.org/10.3390/a14030091

Academic Editor: Frank Werner

Received: 11 February 2021
Accepted: 15 March 2021
Published: 17 March 2021

1. Introduction

Unmanned Aerial Vehicle (UAV) swarm formation has applications in many areas of research, such as infrastructure inspection [1], surveillance [2,3], target tracking [4], and precision agriculture [5]. There are existing methods in the literature to control UAV swarms using centralized methods [6–11], where there is a command center (centralized system) computing optimal motion commands for the UAVs. Centralized methods are relatively easy to develop and implement, but computational complexity grows exponentially with the size of the swarm. To address this challenge, we present a decentralized UAV swarm formation control approach using decentralized Markov decision process framework. The main goal this study is to drive the swarm fly and hover in a certain geographical region while forming a certain geometrical shape. The motivation for studying such problems comes from data fusion applications with UAV swarms where the fusion performance depends on the strategic relative separation of the UAVs from each other [12,13]. We previously studied decentralized decision making frameworks for UAV swarm formation in two-dimensional (2D) scenarios [14], while in this study, we decentralized control strategies in three-dimensional (3D) scenarios.

The formation control of vehicle swarms has many applications in areas including infrastructure inspection, precision agriculture, intelligent transportation, and surveillance. In many applications in these domains, strategic placement of the vehicles (forming a certain geometrical shape, e.g., points on the surface of a sphere) can lead to significant gains in data fusion performance due to the different vantage points of the sensors on-board the vehicles observing a target of interest [10]. Suppose the vehicles carry optical cameras generating 2D images of a 3D object, and if the goal is to reconstruct the 3D shape of the object via the 2D images (i.e., tomography-based methods), the strategic placements of

the vehicles around the object can have significant impact on the performance of the 3D shape reconstruction.

Different formation control settings have been studied in the past: ground vehicles [15–17], unmanned aerial vehicles (UAVs) [18,19], surface and underwater autonomous vehicles (AUVs) [20,21]. Regardless of settings, there are many different methodologies developed by the researchers to tackle formation control problem, e.g., behavior-based, virtual structure, and leader following. The authors of [22,23] developed a behavior-based approach in which they described the desired behavior for each robot, e.g., collision avoidance, formation keeping, and target seeking. The control commands for the robot is determined by weighing the relative importance of each behavior. The virtual structure approach [24,25] takes a physical object shape as a reference and mimics the formation of that shape. The robots are required to communicate with each other in order to achieve a formation in this approach which requires significant communication costs (e.g., bandwidth). The leader following approach [15] requires a robot, assigned as a leader, which moves according to a predetermined trajectory. The other robots, the followers, are designed to follow the leader, maintaining a desired distance and orientation with respect to the leader. The main drawback of this approach is that the followers are dependent on the leader to achieve the goal (formation). The system may collapse if the leader fails when the leader possibly runs short on power or when the communications link fails. Considering the aforementioned limitations of formation control, which specifically stem from centralized approaches, we develop a decentralized Markov decision process (Dec-MDP)-based formation control approach for a UAV swarm. Our decentralized control strategies are robust to failures of individual UAVs in the swarm and also robust to communications link failures.

Centralized control strategies for UAV swarm control are well studied [7–9,11,26]. For instance, the authors of [6,7] developed UAV control strategies for target tracking in a centralized setting. In centralized systems like these, typically, there exists a notional fusion center (a computing node) that collects and fuses the sensor measurements (e.g., using Bayes' theorem) from all the UAVs and runs a tracking algorithm (e.g., Kalman filter) to maintain and update the estimate of the state of the system. More importantly, the fusion center computes the combined optimal control commands for all the UAVs to maximize the system performance. For instance, the authors of [10] used the notion of fusion center to control fixed-wing UAVs for multitarget tracking while accounting for collision avoidance and wind disturbance on UAVs. Although, these centralized control and fusion strategies are easy to implement, they are computationally expensive especially if the swarm is large. Specifically, the computational complexity increases exponentially with the number of UAVs in the swarm.

To tackle these challenges, a few studies in the literature developed decentralized control strategies [14,26–29]. The authors of [26] used the decentralized partially observable Markov decision process (Dec-POMDP) to formulate and solve a target tracking problem with a swarm of decentralized UAVs. As solving decentralized POMDP is very difficult (as is the case with solving any decision-theoretic methods), the authors introduced an approximate dynamic programming method called *nominal belief-state optimization* (NBO) to solve the control problem. The authors in [30] developed a UAV formation control approach using decentralized Model Predictive Control (MPC). In their work, the UAVs were able to avoid collisions with multiple obstacles in a decentralized manner. They used a figure of eight as a reference trajectory; their results show that the UAVs were able to avoid collision with obstacles and among themselves. Several recent papers describe the formation control of different geometric shapes, e.g., multi-agent circular shape with a leader [9]. The authors of [9] propose centralized formation control, which is not suitable for swarm control when the number of UAVs in the swarm is large. Although decentralized control methods exist in the literature, our method is novel in the sense that each UAV in the swarm optimizes its own control commands and its nearest neighbor's controls over time. Then, each UAV implements its own optimized controls, and discards the neighbor's controls. We anticipate, from this decentralized control optimization approach,

a global cooperative behavior among the UAVs emerges mimicking a centralized control approach. The authors of [31] demonstrated a successful use of a distributed UAV control framework for wildfire monitoring while avoiding in-flight collisions. The authors of [32] introduced path tracking and desired formation for networked mobile vehicles using nonlinear control theory to maintain the formation in the network. They have showed that path tracking error of each vehicle is reduced to zero and formation is achieved asymptotically. As centralized control strategies suffer from exponential computational complexity and high memory usage, the decentralized control methods are being actively pursued in the context of swarm control, especially when the size of the swarm is large. A survey of these decentralized control strategies can be found in [29].

In this paper, we develop a novel decentralized UAV swarm formation control approach using Dec-MDP formation. In this problem, the goal is to optimize the UAV control decisions (e.g., waypoints) in a decentralized manner, such that the swarm forms a certain geometrical shape while avoiding collisions. We use dynamic programming principles to solve the decentralized swarm motion control problem. As most dynamic programming problems suffer from the curse of dimensionality, we adapt a fast heuristic approach called *nominal belief-state optimization* (NBO) [10,33] to approximately solve the formation control problem. We perform simulation studies to validate our control algorithms and compare their performance with centralized approaches for bench marking the performance.

Key Contributions

- We formulate the UAV swarm formation control problem as a decentralized Markov decision process (Dec-MDP).
- We extend an approximate dynamic programming method called *nominal belief-state optimization* (NBO) to solve the formation control problem.
- We perform numerical studies in MATLAB to validate the swarm formation control algorithms developed here.
- One of the key contributions of this paper is to induce cooperative behavior among the UAVs in the swarm via the following novel decentralized control optimization strategy:
  - Each UAV i optimizes the control vector $[a_k^i, a_k^{nn}]$ at time k, where a_k^i is the control vector for UAV i, and a_k^{nn} is the control vector for its nearest neighbor.
  - Next, UAV i discards the optimized controls for its neighbor and implements just its own controls a_k^i.
  - Each UAV in the system implements the above approach.

The rest of the paper is organized as follows. Section 2 provides the problem specification and objectives. We also formulate the problem using decentralized Markov decision process in Section 2 followed by the discussion on the NBO approach in Section 3. UAV motion model and dynamics are provided in Section 4. In Section 5, we discuss simulation results to evaluate the performance of our method.

2. Problem Formulation

Unmanned aerial vehicles: We consider quadrotor motion dynamics in 3D, as modeled in [34,35]. In this study, our goal is to optimize the *waypoints* (position coordinates in 3D space) for the quadrotors to guide the UAVs to their destination formation shape while avoiding collisions.

Communications and Sensing: We assume that UAVs are equipped with sensing systems and wireless transceivers with which each UAV learns the exact location and the velocity of the nearest neighboring UAV. Our decentralized control method requires only the kinematic state (location and velocity) of the nearest neighbor to optimize the control commands of the local UAV.

Objective: The goal is to control the swarm (optimizing *waypoints*) in a decentralized manner, such that the swarm arrives at a certain pre-determined 3D geometrical surface in the shortest time possible while avoiding collisions.

We formulate the swarm formation control problem as a decentralized Markov decision process (Dec-MDP). Dec-MDP is a mathematical formulation useful for modeling control problems for decentralized decision making. This formulation has the following advantages: (1) allows us to efficiently utilize the computing resources on-board all the UAVs, (2) requires less computational time compared to centralized approaches, (3) as UAVs are decentralized, point of failure of the entire mission is minimal, (4) decentralized approach provides robustness to addition or deletion of UAVs to the swarm, (5) UAVs do not need to rely on a central command center for evaluating optimal control commands. We define the key components of Dec-MDP as follows. Here, k represents the discrete-time index.

Dec-MDP Ingredients

Agents/UAVs: We assume there are N UAVs in our system. The set of UAVs is given by an index vector $I = \{1, ..., N\}$. This index vectors may be referred to as a set of agents or set of independent decision makers. Here, a UAV can be considered an agent or a decision maker.

States: We model the system dynamics in discrete time, where k represents the time index. The state of the system s_k includes the locations and velocities of all the UAVs in the system.

Actions: The actions are the controllable aspects of the system. We define action vector $a_k = (a_k^1, \ldots, a_k^N)$, where a_k^i represents the action vector at UAV i, which includes the position coordinates in 3D for the UAV.

State Transition Law: State transition law describes how the state evolves over time. Specifically, the transition law is a conditional probability distribution of the next state given the current state and the current control actions (assuming the Markovian property holds). The transition law is given by $s_{k+1} \sim p_k(\cdot|s_k, a_k)$, where p_k is the conditional probability distribution. Since the state of the system only includes the states of the UAVs, the state transition law is completely determined by the dynamics of the UAVs (discussed in the next section). In other words, the transition law is given by $s_{k+1}^i = \psi(s_k^i, a_k^i) + \mathcal{W}_k^i, i = 1, \ldots, N$, where s_k^i represents the state of the ith UAV and a_k^i indicates the local dynamic controls (position coordinates) of ith UAV, ψ represents the motion model as discussed in Section 4, and $\mathcal{W}_k^i$ represents noise, which is modeled as a zero-mean Gaussian random variable.

Cost Function: The cost function $C(s_k, a_k)$ deals with cost of being in a given state s_k and performing actions a_k. Here, s_k represents the global state, i.e., the state of all the UAVs in the system. Since the problem is decentralized, each UAV only has access to its local state and the state of the nearest neighboring UAV. Let $b_k^i = (s_k^i, s_k^{nn})$ represent that local system state at UAV i, where s_k^{nn} is the state of the nearest neighboring UAV, and $nn \in I \backslash \{i\}$.

Let d^i be the destination location UAV i must reach, and $d_{\text{coll,thresh}}$ is the distance between the UAVs below which the UAVs are considered to be at the risk of collision. We now define the local cost function for UAV i, as follows:

$$\begin{aligned} c(b_k^i, a_k^i, a_k^{nn}) = w_1 \Big[\text{dist}(s_k^{i,\text{pos}}, d^i) + \text{dist}(\text{s}_\text{k}^{\text{nn,pos}}, \text{d}^{\text{nn}}) \Big] \\ + w_2 \Big[\text{dist}(s_k^i, s_k^{nn})^{-1} \mathbb{I}\Big(\text{dist}(s_k^i, s_k^{nn}) < d_{\text{coll,thresh}} \Big) \Big] \end{aligned} \tag{1}$$

where $s_k^{i,\text{pos}}$ represents the location of the ith UAV, w_1 and w_2 are weighting parameters, $\text{dist}(a, b)$ represents the distance between locations a and b, and $\mathbb{I}(a)$ is the indicator function, i.e., $\mathbb{I}(a) = 1$ if the argument a is true and 0 otherwise.

By minimizing the above cost function, each UAV optimizes its own control commands and that of its neighbor, but only implement its own local control commands and discards the commands optimized for its neighbor. The first part of the cost function lets the UAV reach its destination, while the second part minimizes the risk of collisions between UAVs.

The Dec-MDP starts at an initial random state s_0 and the state of the system evolves according to the state-transition law and the control commands applied at each UAV. The overall objective is to optimize the control commands at each UAV i such that the

expected cumulative local cost over a horizon H (shown below) is minimized. where b_0^i is the initial local state at UAV i, and the expectation $E[\cdot]$ is over the stochastic evolution of the local state over time (due to the random variables present in the UAV dynamic equations).

$$\min_{\{a_k^i, a_k^{nn}\}, k=0,\ldots,H-1} \mathrm{E}\left[\sum_{k=0}^{H-1} c(b_k^i, a_k^i, a_k^{nn}) \middle| b_0^i\right] \tag{2}$$

3. NBO Approach to Solve Dec-MDP

It is well know in the literature that solving Equation (2) exactly is computationally prohibitive and not practical. For this reason, we extend a heuristic approach called *nominal belief-state optimization* (NBO) [10]. As discussed in the previous section, we let a UAV optimize its own and its nearest neighbor's controls over the time horizon H. Once the UAV calculates local controls for itself and its neighbors, the UAV implements its own controls and discards its neighbors controls at each time step. Since obtaining the expectation in Equation (2) exactly is not tractable, the NBO approach approximates this expectation by assuming that all the future random variables (over which the expectation is supposed to be evaluated) assume the nominal values, i.e., the mean values. Since we model the above-mentioned random variable as zero-mean Gaussian, the nominal values are simply zeros. In summary, the NBO approach approximates the cumulative cost function in Equation (2) by replacing the expectation with the random trajectory of the states over time by a sequence of states obtained by replacing future random variables with zeros. In the NBO method, the objective function at agent i is approximated as follows:

$$J(b_0^i) \approx \sum_{k=0}^{H-1} c(\hat{b}_k^i, a_k^i, a_k^{nn}), \tag{3}$$

where $\hat{b}_1^i, \hat{b}_2^i, \ldots, \hat{b}_{H-1}^i$ is a *nominal* local state sequence.

4. UAV Motion Model

The state of the ith UAV at time k is given by $s_k^i = (x_k^i, y_k^i, z_k^i, \phi_k^i, \theta_k^i, \psi_k^i)$, where (x_k^i, y_k^i, z_k^i) are position coordinates and $[\phi_k^i, \theta_k^i, \psi_k^i] = [\textit{bank angle, pitch angle, heading angle}]$ are the Euler angles. The UAV motion dynamics are given by the following equations.

$$\begin{aligned}
u_{k+1} &= T(-g\sin(\theta_k) + r_k v_k - q_k w_k) + u_k + \mathcal{W}_k^u \\
v_{k+1} &= T(g\sin(\phi_k)\cos(\theta_k) - r_k u_k + p_k w_k) + v_k + \mathcal{W}_k^v \\
w_{k+1} &= T\left(\frac{1}{m}(-F_z) + g\cos(\phi_k)\cos(\theta_k) + q_k u_k - p_k v_k\right) + w_k + \mathcal{W}_k^w \\
p_{k+1} &= T\left(\frac{1}{I_{xx}}(L + (I_{yy} - I_{zz})q_k r_k)\right) + p_k + \mathcal{W}_k^p \\
q_{k+1} &= T\left(\frac{1}{I_{yy}}(M + (I_{zz} - I_{xx})p_k r_k)\right) + q_k + \mathcal{W}_k^q \\
r_{k+1} &= T\left(\frac{1}{I_{zz}}(N + (I_{xx} - I_{yy})p_k q_k)\right) + r_k + \mathcal{W}_k^r \\
\phi_{k+1} &= T(p_k + (q_k\sin\phi_k + r_k\cos\phi_k)\tan\theta_k) + \phi_k + \mathcal{W}_k^\phi \\
\theta_{k+1} &= T(q_k\cos\phi_k - r_k\sin\phi_k) + \theta_k + \mathcal{W}_k^\theta \\
\psi_{k+1} &= T((q_k\sin\phi_k + r_k\cos\phi_k)\sec\theta_k) + \psi_k + \mathcal{W}_k^\psi \\
x_{k+1} &= T\left(c_{\theta_k}c_{\psi_k}u^b + (-c_{\phi_k}s_{\psi_k} + s_{\phi_k}s_{\theta_k}c_{\psi_k})v^b + (s_{\phi_k}s_{\psi_k} + c_{\phi_k}s_{\theta_k}c_{\psi_k})w^b\right) + x_k + \mathcal{W}_k^x
\end{aligned}$$

$$y_{k+1} = T\Big(c_{\theta_k} s_{\psi_k} u^b + (c_{\phi_k} c_{\psi_k} + s_{\phi_k} s_{\theta_k} s_{\psi_k}) v^b + (-s_{\phi_k} c_{\psi_k} + c_{\phi_k} s_{\theta_k} s_{\psi_k}) w^b\Big) + y_k + \mathcal{W}_k^y$$
$$z_{k+1} = T\Big(-1 * (-s_{\theta_k} u^b + s_{\phi_k} c_{\theta_k} v^b + c_{\phi_k} c_{\theta_k} w^b)\Big) + z_k + \mathcal{W}_k^z$$

where, $\mathcal{W}_k$ is a zero-mean Gaussian random variables, $[u_k, v_k, w_k]$ = [*longitudinal velocity, lateral velocity, normal velocity*] are the linear velocity, and $[p_k, q_k, r_k]$ = [*roll rate, pitch rate, yaw rate*] represent the angular velocity of the vehicle at time k. $[F_x, F_y, F_z]$ are linear translation forces and $[L, M, N]$ are angular moments.

UAV Motion Control

We implement a linear controller [36] to produce the appropriate torque and thrust in order to drive the UAV to the desired state in SO(3), governed by the optimized waypoints. The Figure 1 shows how the waypoints generator works with the controller. We make the following assumptions for the linear controller.

- We linearize the trigonometric functions assuming roll angle ϕ and pitch angle θ small enough, i.e., $\cos\phi = 1$, $\sin\phi = \phi$, $\cos\theta = 1$, $\sin\theta = \theta$
- The angular velocity of the UAV is also considered small enough

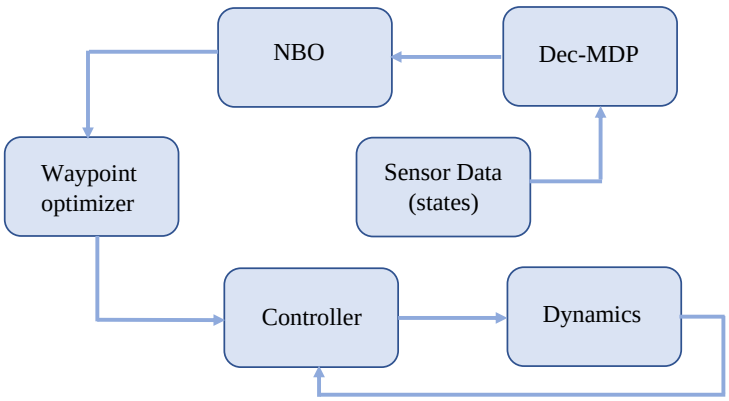


Figure 1. UAV formation shape control architecture.

The linear controller is described extensively in [37,38]. The control problem is to calculate the inputs $u_1 = \sum_{i=1}^{4} F_i$ and u_2 required to track a set of waypoints r_k^w. The input u_2 is given by the following equation.

$$u_2 = \begin{bmatrix} 0 & L & 0 & -L \\ -L & 0 & L & 0 \\ \gamma & \gamma & \gamma & \gamma \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix}$$

where, $[F_1, F_2, F_3, F_4]$ are propeller forces and γ is the drag coefficient.

Position control. The position control method use the bank and the pitch angles as inputs to drive the position of the UAV. The position controller determines the desired bank angle ϕ^{des} and desired pitch angle θ^{des}. The desired bank and pitch angles are used to calculate the desired speed of the UAV [37].

5. Simulation Results

We assume that each UAV has its own on-board computer to compute the local optimal control decisions. We implement the above-discussed NBO approach to solve the swarm control problem in MATLAB. We test our methods in two scenarios—a spherical shape with and without an obstacle. The UAVs are aware of the shape dimensions and the exact location of shape. Each UAV randomly picks a location on the formation shape, and uses

the NBO approach to arrive at this location. We use MATLAB's *fmincon* to solve the NBO optimization problem. Here, we set the horizon length to $H = 3$ time steps.

We define the following metrics to measure the performance of our formation control approach: (1) T_c*-average computation time to evaluate the optimal control commands* and (2) T_f: *time taken for the swarm to arrive on the formation shape*. As a benchmark method, we use a centralized approach to solve the above-discussed swarm formation control problem. In other words, we use a single NBO algorithm, which optimizes the motion control commands for all the UAVs together based on the global state of the system. We implement this centralized algorithm in MATLAB.

We implement the Dec-MDP approach with a spherical formation shape with and without an obstacle. The resulting swarm motion is shown in Figure 2 for the spherical formation shape in the absence of any obstacle using the cost function described in Equation (1). The scenario with an obstacle considers the following cost function.

$$\begin{aligned} c(b_k^i, a_k^i, a_k^{nn}) = w_1 \Big[\mathrm{dist}(s_k^{i,\mathrm{pos}}, d^i) + \mathrm{dist}(\mathrm{s}_\mathrm{k}^{\mathrm{nn,pos}}, \mathrm{d}^{\mathrm{nn}}) \Big] \\ + w_2 \Big[\mathrm{dist}(s_k^i, s_k^{nn})^{-1} \mathbb{I}(\mathrm{dist}\Big(s_k^i, s_k^{nn}\Big) < d_{\mathrm{coll,thresh}}\Big) \Big] \\ + w_3 \Big[\mathrm{dist}(s_k^i, s_k^{obstacle})^{-1} \mathbb{I}(\mathrm{dist}\Big(s_k^i, s_k^{obstacle}\Big) < d_{\mathrm{coll,obstacle}}\Big) \Big] \end{aligned}$$

where $s_k^{obstacle}$ is the location of an obstacle, $d_{\mathrm{coll,obstacle}}$ is a collision threshold with the obstacle, and w_3 is a weighting parameter. The indicator function $\mathbb{I}(b) = 1$, if the argument b is true and 0 otherwise. The resulting motion of the scenario with the obstacle is shown in Figure 3. For this scenario, we also plot the distance between every pair of UAVs in the swarm, as shown in Figure 4. Here, we assume that there is a collision risk between a pair of UAVs when the distance between them is less than 5 m. Clearly, the Figures 3 and 4 demonstrate that our decentralized algorithm drives the swarm to the destination while successfully avoiding collisions between the UAVs.

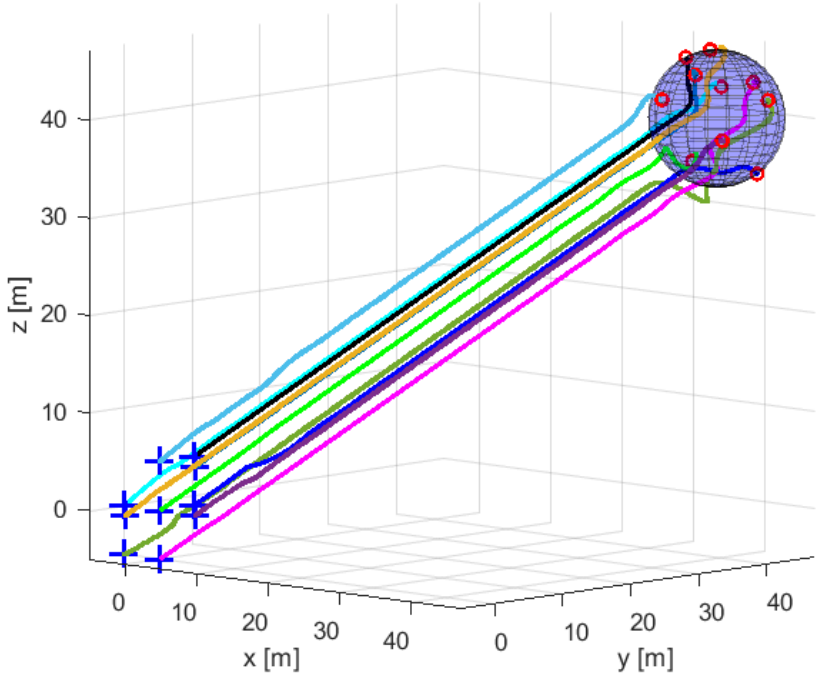


Figure 2. UAV swarm converging to the spherical formation shapes in 3D.

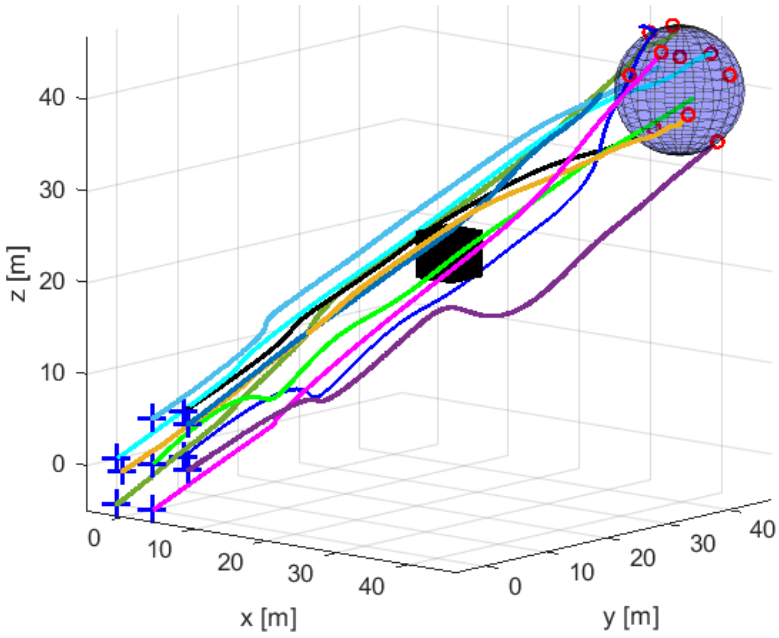


Figure 3. UAV swarm converging to the spherical formation shapes avoiding obstacle.

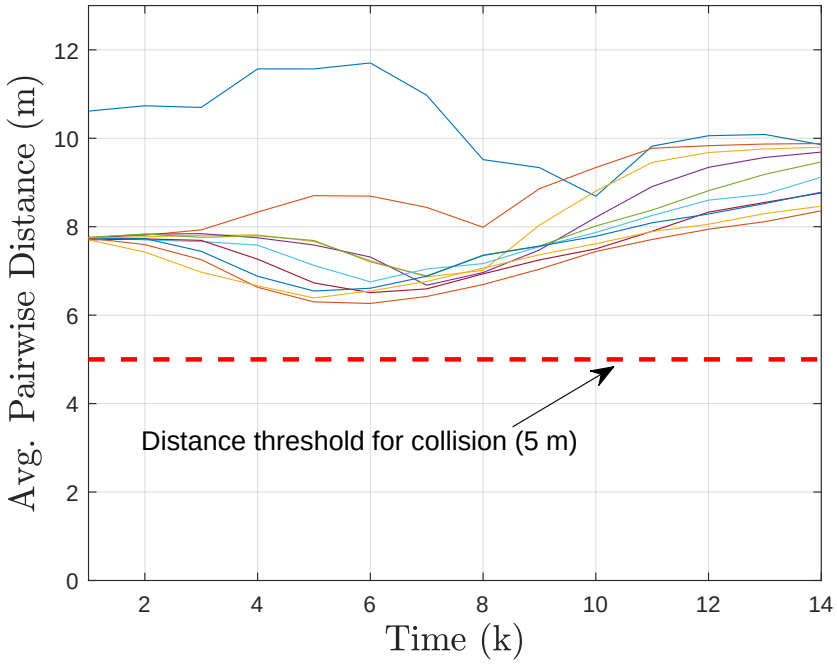


Figure 4. Distance between each pair of UAVs.

We calculate the T_c and T_f values for both the centralized and the decentralized algorithms for 10 UAVs. Figure 5 and Table 1 clearly demonstrate that our decentralized method significantly outperforms the centralized method with respect to both the metrics T_c and T_f.

Table 1. Average time taken by the swarm to arrive at the formation shape.

	Dec-MDP	**Centralized**
$T_f(sec)$	16.7	25.98

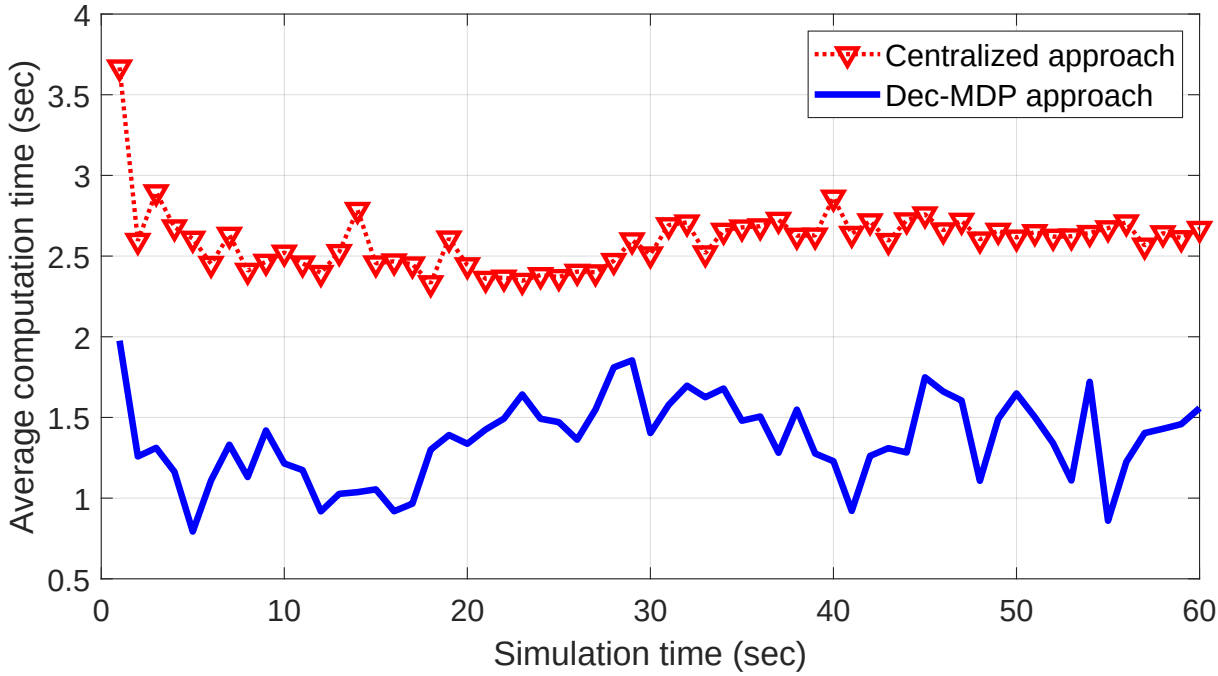


Figure 5. Computation time (T_c): centralized vs. decentralized method.

We now compute average computation time and average pairwise distance with respect to neighborhood threshold where each UAV communicates with other UAVs within the radius of neighborhood threshold. If neighborhood threshold is infinity, a UAV can communicate with all other UAVs in the swarm. UAVs optimize their decisions together with neighbors, which depend on neighborhood threshold and implement its own control. We expect that, with the increase in neighborhood threshold, average computation time rises and, after certain neighborhood threshold, average computation time saturates. Figure 6 shows average computation time rise until neighborhood threshold reach 240 m and then waves between 20 to 25 s.

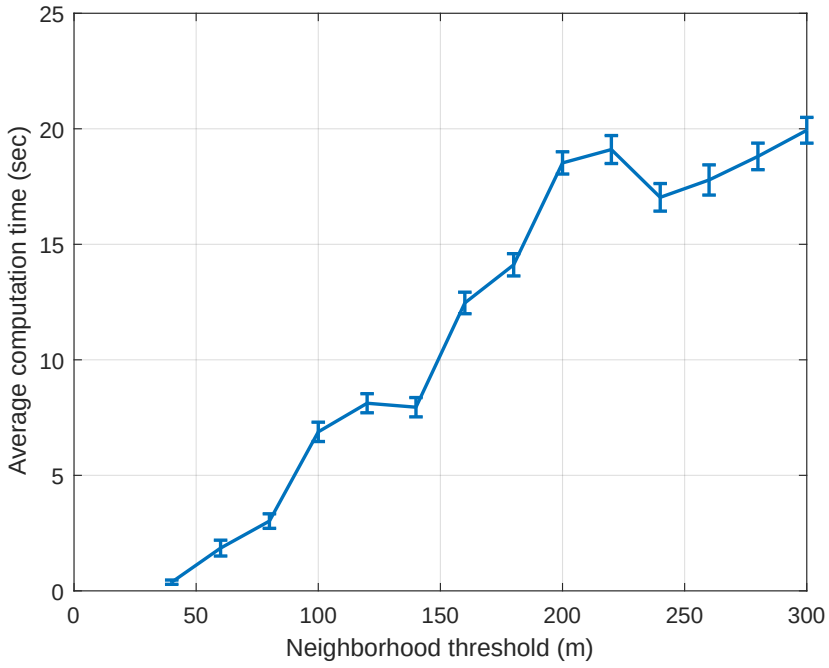


Figure 6. Average computation time with respect to neighborhood threshold.

We also expect that with the increase of neighborhood threshold, average pairwise distance drops. The reason we are interested in analyzing average pairwise distance is, we expect the swarm to be as close as possible while avoiding collision between UAVs. Small average pairwise distance allows the swarm to be more cooperative while saving battery life as communication distance depends on distance between UAVs. Figures 6 and 7 suggest that a neighborhood threshold of more than 130 m allows UAVs to stay close in the swarm with reasonable computation cost.

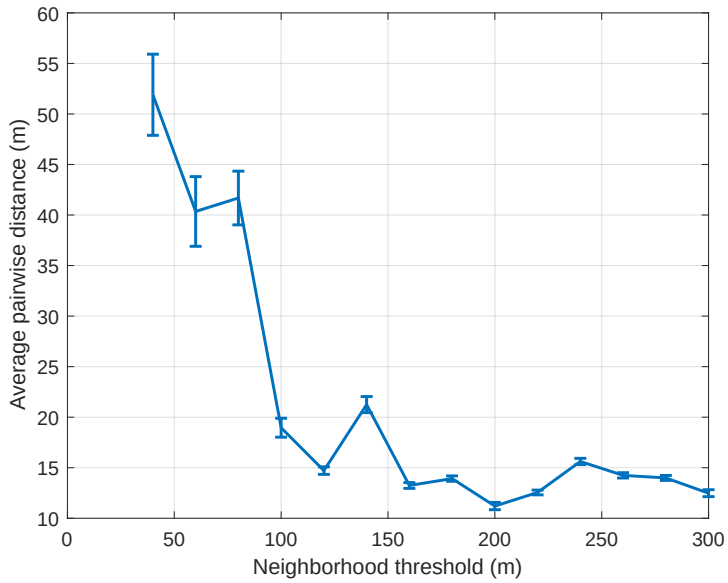


Figure 7. Average pairwise distance with respect to neighborhood threshold.

6. Conclusions

In this paper, we developed decentralized control method for UAVs in the context of formation control. Specifically, we extended a decision-theoretic formulation called *decentralized Markov decision process* (Dec-MDP) to develop near real-time decentralized control methods to drive a UAV swarm from an initial formation to a desired formation in the shortest time possible. As decision-theoretic approaches suffer from the curse of dimensionality, for computational tractability, we extended an approximate dynamic programming method called *nominal belief-state optimization* (NBO) to approximately solve the Dec-MDP. For benchmarking, we also implemented a centralized approach (Markov decision process-based) and compared the performance of our decentralized control methods against the centralized methods. In the context of the formation control problem, our results show that the average computation time for obtaining the optimal controls and the time taken for the swarm to arrive at the formation shape are significantly less with our Dec-MDP approach compared with that of the centralized methods. We also studied the impact of neighborhood threshold on multiple performance metrics in a UAV swarm.

The formation control approach discussed in this thesis can be extended to 3D formation, and these formations can be used to sense the environments for 3D reconstruction of a scene. The vantage points of the UAVs in the swarm in 3D formation can be exploited for the efficient reconstruction of the scene in 3D, while extending tomography-type approaches. The decentralized control strategies presented in this thesis can be extended to control the motion of the UAVs in the swarm to maximize the efficiency of the above 3D scene reconstruction process. These methods have several applications, including the use of drones to map unexplored and unsafe regions (e.g., caves, underground mines, toxic environments).

Author Contributions: Conceptualization, M.A.A., H.D.M. and S.R.; Methodology, M.A.A. and S.R.; Validation, M.A.A. and S.R.; Writing and Editing, M.A.A. and S.R.; and Paper Review, H.D.M. and S.R. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by Air Force Office of Scientific Research under Grant FA9550-19-1-0070.

Acknowledgments: This work was supported in part by Air Force Office of Scientific Research under Grant FA9550-19-1-0070.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Waharte, S.; Trigoni, N.; Julier, S. Coordinated Search with a Swarm of UAVs. In Proceedings of the 2009 6th IEEE Annual Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks Workshops, Rome, Italy, 22–26 June 2009; Volume 1109.
2. Walle, D.V.D.; Fidan, B.; Sutton, A.; Yu, C.; Anderson, B.D.O. Non-hierarchical UAV Formation Control for Surveillance Tasks. In Proceedings of the American Control Conference, Seattle, WA, USA, 11–13 June 2008; pp. 777–782.
3. Carthel, C.; Coraluppi, S.; Grignan, P. Multisensor tracking and fusion for maritime surveillance. In Proceedings of the 10th International Conference on Information Fusion, Quebec City, QC, Canada, 9–12 July 2007; pp. 1–6.
4. Shames, I.; Fidan, B.; Anderson, B.D.O. Close Target Reconnaissance using Autonomous UAV Formations. In Proceedings of the 47th IEEE Conference Decision and Control, Cancun, Mexico, 9–11 December 2008; pp. 1729–1734.
5. Vu, Q.; Raković, M.; Delic, V.; Ronzhin, A. Trends in development of UAV-UGV cooperation approaches in precision agriculture. In *International Conference on Interactive Collaborative Robotics*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 213–221.
6. Ragi, S.; Chong, E.K.P. Dynamic UAV Path Planning for Multitargte Tracking. In Proceedings of the American Control Conference, Montreal, QC, Canada, 27–29 June 2012; pp. 3845–3850.
7. Zhan, P.; Casbeer, D.; Swindlehurst, A. A centralized control algorithm for target tracking with UAVs. In Proceedings of the Conference Record of the Thirty-Ninth Asilomar Conference onSignals, Systems and Computers, Monterey, CA, USA, 30 October–2 November 2005; pp. 1148–1152.
8. Qiu, H.; Huang, G.; Gao, J. Centralized multi-sensor multi-target tracking with labeled random finite set. *J. Aerosp. Eng.* **2005**, *231*, 669–676. [CrossRef]
9. Zhao, L.; Ma, D. Circle Formation Control for Multi-agent Systems with a Leader. *Control Theory Technol.* **2015**, *13*, 82–88. [CrossRef]
10. Ragi, S.; Chong, E.K.P. UAV Path Planning in a Dynamic Environment via Partially Observable Markov Decision Process. *IEEE Trans. Aerosp. Electron. Syst.* **2013**, *49*, 2397–2412. [CrossRef]
11. Chong, E.K.P.; Kreucher, C.; Hero, A.O. Partially observable Markov decision process approximations for adaptive sensing. *Disc. Event Dyn. Sys.* **2009**, *19*, 377–422. [CrossRef]
12. Bar-Shalom, Y.; Willett, P.K.; Tian, X. *Tracking and Data Fusion*; YBS Publishing: Storrs, CT, USA, 2011; Volume 11.
13. Shen, D.; Chen, G.; Cruz, J.B.; Blasch, E. A game theoretic data fusion aided path planning approach for cooperative UAV ISR. In Proceedings of the 2008 IEEE Aerospace Conference, Big Sky, MT, USA, 1–8 March 2008; pp. 1–9.
14. Azam, M.A.; Ragi, S. Decentralized formation shape control of UAV swarm using dynamic programming. In Proceedings of the Signal Processing, Sensor/Information Fusion, and Target Recognition XXIX. International Society for Optics and Photonics, Bellingham, WA, USA, 27 April–8 May 2020; Volume 11423, p. 114230I.
15. Das, A.K.; Fierro, R.; Kumar, V.; Ostrowsky, J.P.; Spletzer, J.; Taylor, C. A vision-based formation control framework. *IEEE Trans. Robot. Autom.* **2002**, *18*, 813–825. [CrossRef]
16. Fax, J.A.; Murray, R.M. Information flow and cooperative control of vehicle formations. *IEEE Trans. Autom. Control* **2004**, *49*, 1465–1476. [CrossRef]
17. Ghabcheloo, R.; Pascoal, A.; Silvestre.; Kaminer, I. Coordinated path following control of multiple wheeled robots using linearization techniques. *Int. J. Syst. Sci.* **2006**, *37*, 399–414. [CrossRef]
18. Singh, S.N.; Chandler, P.; Schumacher, C.; Banda, S.; Pachter, M. Adaptive feedback linearizing nonlinear close formation control of UAVs. *Am. Control Conf.* **2000**, *2*, 854–858.
19. Koo, T.J.; Shahruz, S.M. Formation of a group of unmanned aerial vehicles (UAVs). *Am. Control Conf.* **2001**, *1*, 69–74.
20. Edwards, D.B.; Bean, T.A.; Odell, D.L.; Anderson, M.J. A leader–follower algorithm for multiple AUV formations. *IEEE/OES Auton. Underw. Veh.* **2004**, *2*, 40–46.
21. Skjetne, R.; Moi, S.; Fossen, T.I. Nonlinear formation control of marine craft. *IEEE Int. Conf. Decis. Control* **2002**, *2*.
22. Balch, T.; Arkin, R.C. Behavior-based formation control for multirobot teams. *IEEE Trans. Robot. Autom.* **1998**, *14*, 926–939. [CrossRef]
23. Lawton, J.R.; Beard, R.W.; Young, B.J. A decentralized approach to formation maneuvers. *IEEE Trans. Robot. Autom.* **2003**, *19*, 933–941. [CrossRef]
24. Do, K.D.; Pan, J. Nonlinear formation control of unicycle-type mobile robots. *Robot. Auton. Syst.* **2007**, *55*, 191–204. [CrossRef]

25. Lewis, M.A.; Tan, K.H. High precision formation control of mobile robots using virtual structures. *Auton. Robot.* **1997**, *4*, 387–403. [CrossRef]
26. Ragi, S.; Chong, E.K.P. Decentralized Guidance Control of UAVs with Explicit Optimization of Communication. *J. Intell. Robot. Syst.* **2014**, *73*, 811–822. [CrossRef]
27. Kim, Y.; Bang, H. Decentralized control of multiple unmanned aircraft for target tracking and obstacle avoidance. In Proceedings of the 2016 International Conference on Unmanned Aircraft Systems (ICUAS), Arlington, VA, USA, 7–10 June 2016; pp. 327–331.
28. Meng, W.; He, Z.; Su, R.; Shehabinia, A.R.; Lin, L.; Teo, R.; Xie, L. Decentralized control of multi-UAVs for target search, tasking and tracking. *IFAC Proc. Vol.* **2014**, *47*, 10048–10053. [CrossRef]
29. Bakule, L. Decentralized control: An overview. *Elsevier Annu. Rev. Control* **2008**, *32*, 87–98. [CrossRef]
30. Viana, I.B.; Santos, D.A.D.; Goes, L.C.S. Formation Control of Multirotor Aerial Vehicles using Decentralized MPC. *J. Braz. Soc. Mech. Sci. Eng.* **2018**, *40*, 1–12. [CrossRef]
31. Pham, H.X.; La, H.M.; Feil-Seifer, D.; Deans, M. A distributed control framework for a team of unmanned aerial vehicles for dynamic wildfire tracking. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 6648–6653.
32. Zhang, Q.; Lapierre, L.; Xiang, X. Distributed Control of Coordinated Path Tracking for Networked Nonholonomic Mobile Vehicles. *IEEE Trans. Ind. Inform.* **2013**, *9*, 472–484. [CrossRef]
33. Miller, S.A.; Harris, Z.A.; Chong, E.K.P. A POMDP framework for coordinated guidance of autonomous UAVs for multitarget tracking. *EURASIP J. Adv. Signal Process.* **2009**, *2009*, 724597. [CrossRef]
34. Schmidt, D. *Modern Flight Dynamics*; McGraw-Hill Higher Education: New York, NY, USA, 2011.
35. Stengel, R.F. *Flight Dynamics*; Princeton University Press: Princeton, NJ, USA, 2015.
36. Kumar, V.; Michael, N. Opportunities and challenges with autonomous micro aerial vehicles. *Int. J. Robot. Res.* **2012**, *31*, 1279–1291. [CrossRef]
37. Michael, N.; Mellinger, D.; Lindsey, Q.; Kumar, V. The grasp multiple micro-uav testbed. *IEEE Robot. Autom. Mag.* **2010**, *17*, 56–65. [CrossRef]
38. Lee, T.; Leok, M.; McClamroch, N.H. Geometric tracking control of a quadrotor UAV on SE (3). In Proceedings of the 49th IEEE Conference on Decision and Control (CDC), Atlanta, GA, USA, 15–17 December 2010; pp. 5420–5425.

2020 IEEE Conference on Control Technology and Applications (CCTA)
August 24-26, 2020. Montréal, Canada

Random-Sampling Multipath Hypothesis Propagation for Cost Approximation in Long-Horizon Optimal Control

Shankarachary Ragi, *IEEE Senior Member*, and Hans D. Mittelmann

***Abstract*—In this paper, we develop a Monte-Carlo based heuristic approach to approximate the objective function in long horizon optimal control problems. In this approach, we evolve the system state over multiple trajectories into the future while sampling the noise disturbances at each time-step, and find the weighted average of the costs along all the trajectories. We call these methods *random sampling - multipath hypothesis propagation* or RS-MHP. These methods (or variants) exist in the literature; however, the literature lacks convergence results for a generic class of nonlinear systems. This paper fills this knowledge gap to a certain extent. We derive convergence results for the cost approximation error from the MHP methods and discuss their convergence (in probability) as the sample size increases. As a case study, we apply RS-MHP to approximate the cost function in a linear quadratic control problem and demonstrate the benefits of our approach against an existing and closely related approximation approach called *nominal belief-state optimization*.**

***Index Terms*—Long horizon optimal control, cost approximation, approximate dynamic programming, multipath hypothesis propagation.**

I. Introduction

Long-horizon optimal control problems appear naturally in robotics, advanced manufacturing, and economics, especially in applications requiring decision making in stochastic environments. Often these problems are solved via dynamic programming (DP) formulation [1]. DP problems are notorious for their computational complexity, and require approximation approaches to make them tractable. A plethora of approximation techniques called *approximate dynamic programs* (ADPs) exist in the literature to solve these problems approximately. Some of the commonly used ADPs include *policy rollout* [2], *hindsight optimization* [3], [4], etc. A survey of the ADP approaches can found in [1]. Feature-based techniques and deep learning methods are gaining importance in the development of ADP approaches as discussed in [5]. These approximation techniques have been successfully adopted to solve real-time problems such as a UAV guidance control problem in [6]–[8]. Certain ADP approaches, especially the methods based on approximation in value space, require numerical approximation of the expectation in the objective function [6]. In this study, our objective is to develop Monte-Carlo-based approaches to approximate the expectation in the objective function in the long (but finite) horizon optimal control problems, and study their convergence.

This work was supported in part by Air Force Office of Scientific Research under grant FA9550-19-1-0070.

Shankarachary Ragi is with Department of Electrical Engineering, South Dakota School of Mines and Technology, Rapid City, SD 57701, USA shankarachary.ragi@sdsmt.edu

Hans D. Mittelmann is with the School of Mathematical and Statistical Sciences, Arizona State University, Tempe, AZ 85281, USA mittelmann@asu.edu

A. Preliminaries

A long horizon optimal control problem is described as follows. Let x_k be the state vector for a system at time k, which evolves according to a discrete stochastic process as follows:

$$x_{k+1} = f(x_k, u_k, w_k) \tag{1}$$

where $f(\cdot)$ represents the state-transition mapping, u_k is the control vector, and w_k random disturbance. Let $g(x_k, u_k)$ represent the cost (a real value) of being in state x_k and performing action u_k. The functions f and g are independent of k in our study, but can generally depend on k. The goal is to optimize the control vectors $u_k, k = 0, \ldots, H-1$ such that the expected cumulative cost is minimized, i.e., the goal leads to solving the following optimization problem

$$\min_{u_k, k=0,\ldots,H-1} \mathrm{E}\left[\sum_{k=0}^{H-1} g(x_k, u_k)\right], \tag{2}$$

where H is the length of the planning horizon. Let x_0 be the initial state and according to the dynamic programming formulation the optimal cost function is given by

$$J_0^*(x_0) = \min_{u_0} \mathrm{E}\left[g(x_0, u_0) + J_1^*(x_1)\right], \tag{3}$$

where J_1^* represents the optimal cost-to-go from time $k = 1$, and $x_1 = f(x_0, u_0, w_0)$. In this study, *long horizon* refers to the condition that H is sufficiently large that the optimal policy is approximately *stationary* (independent of k). Solving the above optimization problem is not tractable mainly due to two reasons: the expectation $E[\cdot]$ and the optimal cost-to-go J_1^* are hard to evaluate, which are usually approximated by numerical methods or ADP approaches.

An ADP approach called *nominal belief-state optimization* (NBO) [6], [9] was developed primarily to approximate the above expectation. In NBO, the expectation is replaced by a sample state trajectory generated

with an assumption that the future noise variables in the system take so called nominal or mean values, thus making the above objective function deterministic. The NBO method was developed to solve a UAV path optimization problem, which was posed as a *partially observable Markov decision process* (POMDP). POMDP generalizes the long horizon optimal control problem described in Eq. 2 in that the system state is assumed to be "partially" observable, which is inferred via using noisy observations and Bayes rules. Although the performance of the NBO approach was satisfactory, in that it allowed to obtain reasonably optimal control commands for the UAVs, it ignored the uncertainty due to noise disturbances thus leading to inaccurate evaluation of the objective function. To address this challenge, several methods exist in the literature usually referred to as Monte-Carlo Tree Search (MCTS) methods as surveyed in [10].

Inspired from the NBO method and MCTS methods, we develop a new MCTS method called *random sampling - multipath hypothesis propagation* (RS-MHP) and derive convergence results. In this study, we mainly use the NBO approach as a benchmark for performance assessment since RS-MHP builds on the NBO approach.

II. Random Sampling Multipath Hypothesis Propagation (RS-MHP)

In the NBO method, the expectation is replaced by a sample trajectory of the states (as opposed to random states) generated by

$$\tilde{x}_{k+1} = f(\tilde{x}_k, u_k, \bar{w}_k),\ k = 0, \ldots \tag{4}$$

where $\tilde{x}_0 = x_0$ (initial state or current state), and $\bar{w}_k$ is the mean of the random variable w_k. Thus, the long horizon optimal control problem, with NBO approximation, reduces to

$$\min_{u_k} \sum_{k=0}^{H-1} g(\tilde{x}_k, u_k). \tag{5}$$

The above reduced problem, without the need for evaluating the expectation, can significantly reduce the computational burden in solving the long horizon control problems. However, the downside with this approach is it completely ignores the uncertainty in the state evolution, and may generate severely sub-optimal controls. To overcome this trivialization, we develop a Monte-Carlo approach to approximate the expectation described as follows. For time step $k = 1$, we sample the probability distribution of the noise disturbance N times to generate the samples w_0^i with corresponding probability p_0^i, $i = 1, \ldots, N$. Using these, we generate N sample states at $k = 1$ generated according to

$$x_1^i = f(x_0, u_0, w_0^i),\ \forall i. \tag{6}$$

We repeat this sampling approach for time $k = 2$, i.e., we generate N noise samples w_1^i with corresponding probability p_1^i, $i = 1, \ldots, N$. Using these noise samples and the sample states from the previous time step, we generate N^2 sample states at $k = 2$ according to

$$x_2^{i,j} = f(x_1^i, u_1, w_1^j),\ \forall i, j. \tag{7}$$

We repeat the above sampling procedure until the last time step $k = H - 1$ to generate N^{H-1} possible state evolution trajectories using N noise samples generated in each time step.

One can now replace the expectation in Eq. 2 with the weighted average of the cumulative cost corresponding to each state evolution trajectory, where the weights are the probabilities or likeliness of the trajectories. Clearly, the number of possible state trajectories grow exponentially with the horizon length H. Although this approach is not novel as many such methods exist in the literature often classified as Monte-Carlo Tree Search methods, our study is focused on deriving convergence results of RS-MHP approaches.

To avoid the exponential growth in our RS-MHP approach, at each time step we retain only M sample states and prune the remaining states, and if the number of sample states at a given time instance is less than or equal to M, we do not perform pruning. For pruning, at each time k, we rank the state trajectories up to time k according to their likeliness (obtained by multiplying the probabilities of all the noise samples that generated the trajectory) and retain the top M trajectories with highest likeliness and prune the rest. With this procedure, at $k = H - 1$, there would be only M state trajectories. With pruning, the number of trajectories remains a constant irrespective of the time horizon length. An illustration of the above RS-MHP approach is shown in Figure 1 along with the NBO approach. Here, we consider pruning based on likeliness of the state trajectories as the costs from these trajectories have higher contribution in the cost function in Eq. 1 than the less likely trajectories. We will consider other pruning strategies to further improve the approximation error in our future study.

Let $i = 1, \ldots, M$ represent the indices of the M distinct state trajectories with $q_1, q_2, \ldots$ being their probabilities (likeliness). The probability q_i is evaluated by simply multiplying the probabilities of the noise samples that generate the trajectory i over time. These probabilities are normalized, i.e., $\sum_{i=1}^{M} q_i = 1$. Let J represent the actual objective function as described below

$$J = \mathrm{E}\left[\sum_{k=0}^{H-1} g(x_k, u_k)\right]. \tag{8}$$

We can now approximate the objective function J in four possible ways as described below (assuming $N >$

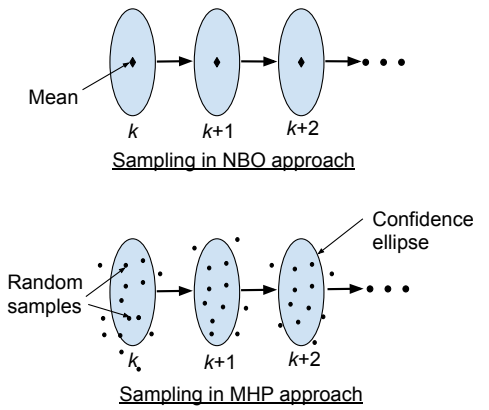


Fig. 1. Sampling probability distributions of noise variables: NBO vs. MHP.

M). Let x_k^i represent the state at time k in the ith state trajectory.

(I) *Sample Averaging*. We can simply approximate the expectation with an average over all possible trajectories as follows:

$$\text{No pruning: } J \approx \tilde{J}_{NP} = \frac{1}{N^{H-1}} \sum_{i=1}^{N^{H-1}} \left(\sum_{k=0}^{H-1} g(x_k^i, u_k) \right)$$
$$\text{With pruning: } J \approx \tilde{J}_P = \frac{1}{M} \sum_{i=1}^{M} \left(\sum_{k=0}^{H-1} g(x_k^i, u_k) \right) \tag{9}$$

(II) *Weighted Sample Averaging*. We can also approximate the expectation with a weighted average with weights being the normalized likeliness indices of the state trajectories given by $q_i, i = 1, \ldots$ (and $\bar{q}_i$ in the pruned case) as follows:

$$\text{No pruning: } J \approx \bar{J}_{NP} = \sum_{i=1}^{N^{H-1}} q_i \left(\sum_{k=0}^{H-1} g(x_k^i, u_k) \right)$$
$$\text{With pruning: } J \approx \bar{J}_P = \sum_{i=1}^{M} \bar{q}_i \left(\sum_{k=0}^{H-1} g(x_k^i, u_k) \right). \tag{10}$$

For a given sequence of control decisions $u_0, u_1, \ldots$, let g_i denote the cost of the ith trajectory given by

$$g_i = \sum_{k=0}^{H-1} g(x_k^i, u_k). \tag{11}$$

Clearly, $g_1, g_2, \ldots$ are identically distributed random variables, where $\mathrm{E}[g_i] = J, \forall i$. In dynamic programming formulations, we do not typically optimize the decision variables $u_0, u_1, \ldots$ together, except in certain ADP schemes such as the NBO, where the decision variables over a finite horizon are indeed optimized together.

The below result suggests that with sufficient number of sample state trajectories (large N), the approximation error in $\tilde{J}_{NP}$ becomes small enough to ignore.

Lemma 2.1: For any given sequence of controls $u_0, u_l, \ldots$, if the random variables $g_1, g_2, \ldots$ have finite variances, $\tilde{J}_{NP}$ converges to J almost surely.

We can verify the above result using a variation of the law of large numbers as stated below

$$\tilde{J}_{NP} = \frac{1}{N^{H-1}} \sum_{i=1}^{N^{H-1}} g_i \xrightarrow{\text{a.s.}} \mathrm{E}[g_i] = J, \tag{12}$$

where $\xrightarrow{\text{a.s.}}$ represents almost sure convergence.

In most applications, normal distributions capture the system or model uncertainties and noise characteristics well, as can be seen in our previous studies [6], [11]. Suppose, for a given sequence of actions $u_0, \ldots, u_{H-1}$, the trajectory cost variables $g_1, g_2, \ldots$ follow normal distribution with $\mathcal{N}(\mu, \sigma^2)$ where μ and σ^2 are the mean and the variance respectively. Of course, if μ is known, then we do not need an approximation strategy as $J = \mu$. However, if $g_1, g_2, \ldots$ are known to follow a normal distribution with unknown mean (μ) and variance (σ) with possibly known bounds, i.e., $\mu_{\min} \le \mu \le \mu_{\max}$ and $\sigma_{\min} \le \sigma \le \sigma_{\max}$, the following result holds significance.

Lemma 2.2: For a given sequence of actions $u_0, \ldots, u_{H-1}$

$$\bar{J}_{NP} \xrightarrow{\text{a.s.}} \frac{J}{\sqrt{2\pi\sigma^2}} \tag{13}$$

Proof: The likeliness probability of g_i is q_i. We also know that $q_i g_1, q_2 g_2, \ldots$ are identically distributed, where the expectation of this sequence is evaluated below

$$\mathrm{E}[q_i g_i] = \int_{-\infty}^{\infty} P(g_i) g_i P(g_i) dg_i. \tag{14}$$

Since $g_i \sim \mathcal{N}(\mu, \sigma^2)$, the following holds:

$$\begin{aligned}
\mathrm{E}[q_i g_i)] &= \int_{-\infty}^{\infty} g_i P(g_i)^2 dg_i \\
&= \int_{-\infty}^{\infty} g_i \left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-(g_i-\mu)^2/2\sigma^2} \right)^2 dg_i \\
&= \frac{1}{\sqrt{4\pi\sigma^2}} \int_{-\infty}^{\infty} g_i \left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-(g_i-\sqrt{2}\mu)^2/2\sigma^2} \right)^2 dg_i \\
&= \frac{\sqrt{2}\mu}{\sqrt{4\pi\sigma^2}} = \frac{\mu}{\sqrt{2\pi\sigma^2}} = \frac{J}{\sqrt{2\pi\sigma^2}}.
\end{aligned} \tag{15}$$

Therefore, due to the law of large numbers

$$\bar{J}_{NP} = \sum_{i=1}^{N^{H-1}} q_i g_i \xrightarrow{\text{a.s.}} \mathrm{E}[q_i g_i] = \frac{J}{\sqrt{2\pi\sigma^2}}. \tag{16}$$

Although the above result does not guarantee that the approximation error for $\bar{J}_{NP}$ converges to zero, we know that the ratio $\bar{J}_{NP}/J$ converges (in probability) to a limit bounded above by the constant $1/\sqrt{2\pi\sigma^2_{\min}}$.

III. Case Study

We implement the above-discussed MHP methods in the context of a linear quadratic Gaussian control (LQG) problem as discussed below. Although there are closed-form solutions for LQG problems, this example allows us to quantify the benefits of using RS-MHP methods over existing similar methods, particularly NBO.

A. Linear Quadratic Problem

Let the system state evolve according to the following linear equation:

$$x_{k+1} = (1-a)x_k + au_k + w_k, \quad w_k \sim \mathcal{N}(0,\sigma^2), \quad (17)$$

where $0 < a < 1$ is a constant, and w_k is a random disturbance modeled by a zero-mean Gaussian distribution with variance σ^2. The cost function over the time-horizon H is defined as follows:

$$J = \mathrm{E}\left[r(x_H - T)^2 + \sum_{k=0}^{H-1} u_k^2 \right], \quad (18)$$

where r and T are constants. This is a simplified oven temperature control example borrowed from [12].

If we apply the traditional NBO method, assuming $H = 2$, the cost function J is approximated (assuming nominal values or zeros for w_0 and w_1) as

$$J_{\mathrm{NBO}} = r\left((1-a)^2 x_0 + a(1-a)u_0 + au_1 - T\right)^2 + u_0^2 + u_1^2 \quad (19)$$

and the exact cost function J can be evaluated analytically as

$$J = r\left((1-a)^2 x_0 + a(1-a)u_0 + au_1 - T\right)^2 + u_0^2 + u_1^2 + r\sigma^2\left((1-a)^2 + 1\right). \quad (20)$$

We notice the approximation error due to the NBO method is

$$|J_{\mathrm{NBO}} - J| = r\sigma^2\left((1-a)^2 + 1\right). \quad (21)$$

This approximation error for a generic time-horizon H (the above error term is derived for $H = 2$) is given by

$$|J_{\mathrm{NBO}} - J| = r\sigma^2 \sum_{n=0}^{H-1} (1-a)^{2n}. \quad (22)$$

The above expression suggests that the NBO approximation error can be significantly high depending on the parameters a, σ, and r. With MHP approximation, the cost function reduces to

$$J_{\mathrm{MHP}} = \frac{1}{P}\left(\sum_{i=1}^{P} r(x_H^i - T)^2 \right) + \sum_{k=0}^{H-1} u_k^2, \quad (23)$$

where P is the number of state-trajectories generated using the MHP approach, and x_H^i is the final state in the ith trajectory. Lemma 2.1 suggests that the approximation error due to the above MHP method converges (in probability) to zero. We verify this result with a numerical simulation, where we implement the NBO and the MHP methods with the following assumptions: $x_0 = 0, r = 10, T = 1, H = 2, u_0 = 0.55, u_1 = 0.17, \sigma = 1$. We vary P from 100 to 10000 with increments of 100. Figure 2 shows the cost function approximated using MHP and NBO methods. The figure clearly demonstrates that the error due to NBO approximation can be significantly high, while MHP performs better in cost approximation.

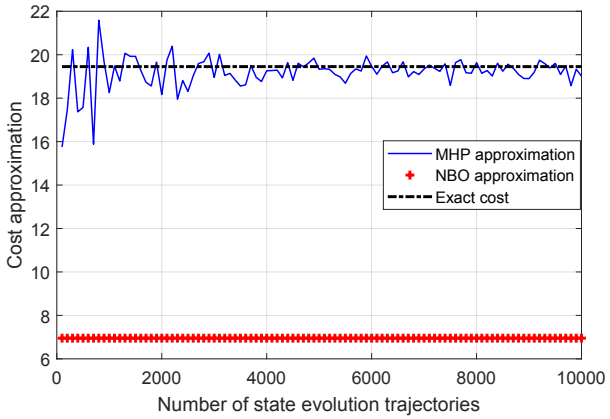


Fig. 2. MHP vs. NBO

RS-MHP has better capability in approximating the expectation operator in Eq. 1 than the NBO approach as we consider multiple hypotheses of state trajectories in RS-MHP as opposed to a single hypothesis in NBO. This is demonstrated in the above case study. In our future study, we will derive quantitative performance guarantees of RS-MHP over NBO for generic long horizon optimal control problems. The impact of parameters M and N on the approximation error will also be considered in our future study.

IV. Conclusions

In this paper, we developed two *approximate dynamic programming* or ADP methods to approximate the cost function in long horizon optimal control problems. Specifically, our methods called *random sampling - multipath hypothesis propagation* or RS-MHP methods are inspired from Monte-carlo Tree Search methods. The basic theme of these methods is to evolve the system state over multiple trajectories into the future while sampling the noise disturbances at each time-step. We derive convergence results that show that the cost approximation error from our RS-MHP methods converges (in probability) toward zero as the sample size increases. As a case study, we applied our methods to approximate the cost function in a linear quadratic control problem, where we demonstrated the benefits of

our approach against an existing approach called *nominal belief-state optimization* or NBO. In our future study, we will apply the above methods to more complex control problems such as the UAV motion control problem we studied in the past [6], where we applied the NBO method to approximate the cost function. Additionally, we will derive convergence results for a general classs of nonlinear systems.

V. Acknowledgment

The authors would like to thank Nicolas Lanchier, Arizona State University, for his valuable inputs and feedback on the convergence results discussed in this paper.

References

[1] E. K. P. Chong, C. M. Kreucher, and A. O. Hero, "Partially observable Markov decision process approximations for adaptive sensing," *Discrete Event Dynamic Systems*, vol. 19, no. 3, pp. 377–422, Sep 2009.

[2] D. P. Bertsekas and D. A. Castanon, "Rollout algorithms for stochastic scheduling problems," *J. Heuristics*, vol. 5, pp. 89–108, 1999.

[3] E. K. P. Chong, R. L. Givan, and H. S. Chang, "A framework for simulation-based network control via hindsight optimization," in *Proc. 39th IEEE Conf. Decision and Control*, Sydney, Australia, 2000, pp. 1433–1438.

[4] G. Wu, E. K. P. Chong, and R. Givan, "Burst-level congestion control using hindsight optimization," *IEEE Trans. Autom. Control*, vol. 47, pp. 979–991, 2002.

[5] D. Bertsekas, "Feature-based aggregation and deep reinforcement learning: A survey and some new implementations," *IEEE/CAA Journal of Automatica Sinica*, no. 1, 2019.

[6] S. Ragi and E. K. P. Chong, "UAV path planning in a dynamic environment via partially observable Markov decision process," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 49, pp. 2397–2412, 2013.

[7] ——, "Dynamic UAV path planning for multitarget tracking," in *Proc. American Control Conf.*, Montreal, Canada, 2012, pp. 3845–3850.

[8] S. Ragi and H. D. Mittelmann, "Mixed-integer nonlinear programming formulation of a UAV path optimization problem," in *Proc. American Control Conf.*, Seattle, WA, 2017, pp. 406–411.

[9] S. Miller, Z. Harris, and E. K. P. Chong, "A POMDP framework for coordinated guidance of autonomous UAVs for multitarget tracking," *EURASIP Journal on Advances in Signal Processing*, 2009.

[10] C. B. Browne, E. Powley, D. Whitehouse, S. M. Lucas, P. I. Cowling, P. Rohlfshagen, S. Tavener, D. Perez, S. Samothrakis, and S. Colton, "A survey of Monte Carlo tree search methods," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 4, no. 1, pp. 1–43, March 2012.

[11] S. Ragi and E. K. P. Chong, "Decentralized guidance control of UAVs with explicit optimization of communication," *J. Intelligent & Robotic Systems*, vol. 73, no. 1, pp. 811–822, 2014.

[12] D. P. Bertsekas. Lecture on reinforcement learning and optimal control. [Online]. Available: http://www.mit.edu/~dimitrib/Slides_Lecture2_RLOC.pdf

Random-Sampling Monte-Carlo Tree Search Methods for Cost Approximation in Long-Horizon Optimal Control

Shankarachary Ragi, *Senior Member, IEEE*, and Hans D. Mittelmann

***Abstract*—We develop Monte-Carlo based heuristic approaches to approximate the objective function in long horizon optimal control problems. In these approaches, to approximate the expectation operator in the objective function, we evolve the system state over multiple trajectories into the future while sampling the noise disturbances at each time-step, and find the average (or weighted average) of the costs along all the trajectories. We call these methods *random sampling - multipath hypothesis propagation* or RS-MHP. These methods (or variants) exist in the literature; however, the literature lacks results on how well these approximation strategies converge. This letter fills this knowledge gap to a certain extent. We derive stochastic convergence results for the cost approximation error from the RS-MHP methods and discuss their convergence (in probability) as the sample size increases. We consider two case studies to demonstrate the effectiveness of our methods - a) linear quadratic control problem; b) unmanned aerial vehicle path optimization problem.**

***Index Terms*—Optimal control, optimization, Markov processes, discrete event systems.**

I. Introduction

LONG-HORIZON optimal control problems appear naturally in robotics, advanced manufacturing, and economics, especially in applications requiring decision making in stochastic environments. Often these problems are solved via dynamic programming (DP) formulation [1]. DP problems are notorious for their computational complexity, and require approximation approaches to make them tractable. A plethora of approximation techniques called *approximate dynamic programs* (ADPs) exist in the literature to solve these problems approximately. The main advantage of the ADP methods is that these approaches aim to approximate a term called *expected-value-to-go* (EVTG) in Bellman's principle [1]–[3] while solving the DP problems, which is otherwise computationally intractable to evaluate. Commonly used ADPs include *policy rollout* [4], *hindsight optimization* [5], [6], etc. A survey of the ADP approaches can be found in [1]. Feature-based techniques and deep learning methods are gaining importance in the development of ADP approaches [7]. In general, bounding the performance of ADP approaches is hard, except when the objective function has special properties. For instance, the authors of [8], using the theory of *string submodularity* [9], bounded the performance of generic ADP schemes if the objective function satisfied certain *curvature* constraints. Bounds on the approximation error from the ADP schemes were derived for infinite horizon optimal control problems in [10] when the objective functions satisfied certain criteria. ADP methods have been successfully adopted to solve real-time problems such as unmanned aerial vehicle (UAV) guidance control [11], [12]. Certain ADP approaches, especially the methods based on approximation in value space, require numerical approximation of the expectation in the objective function [11].

In this letter, we develop Monte-Carlo-based approaches to approximate the expectation in the objective function in the long (but finite) horizon optimal control (LHC) problems, and study their convergence. We refer to these methods as *random sampling multipath hypothesis propagation* (RS-MHP) methods. RS-MHP methods are a variant of the existing broad class of approaches called Monte-Carlo tree search (MCTS) methods. Our RS-MHP methods differ from the existing MCTS methods in the following ways. Most MCTS methods (e.g., Upper Confidence Bounds for Trees [13]) apply sampling in the action space or both in the action and state space, while our methods focus on solely approximating the expectation operator in the LHC objective function while sampling the process noise distributions. This allows us to integrate RS-MHP to approximate the expectation operator in existing ADP methods including Q-learning, policy rollout, and hindsight optimization [1]. Furthermore, RS-MHP allows a smooth and parameterizable trade-off between the computational complexity of the method and its closeness to the optimal solution, a feature that is not present in the existing approaches such the

Manuscript received September 28, 2020; revised November 21, 2020; accepted December 7, 2020. Date of publication December 10, 2020; date of current version January 4, 2021. This work was supported in part by Air Force Office of Scientific Research under Grant FA9550-19-1-0070. Recommended by Senior Editor L. Menini. *(Corresponding author: Shankarachary Ragi.)*

Shankarachary Ragi is with the Department of Electrical Engineering, South Dakota School of Mines and Technology, Rapid City, SD 57701 USA (e-mail: shankarachary.ragi@sdsmt.edu).

Hans D. Mittelmann is with the School of Mathematical and Statistical Sciences, Arizona State University, Tempe, AZ 85281 USA (e-mail: mittelmann@asu.edu).

Digital Object Identifier 10.1109/LCSYS.2020.3043991

nominal belief-state optimization (NBO) [11]. A preliminary version of the parts of this letter were published as [14]. This letter differs from the conference paper [14] in the following ways: 1) we include detailed proofs omitted in the conference version; 2) we derive new convergence results and proofs in Section II-A for non-overlapping tree branching models; 3) we implement our methods for a new case study - UAV path optimization problem.

Our contributions in this letter include: a) new stochastic convergence results on Monte-Carlo based approximation methods to solve LHC problems; b) numerical studies to show the above Monte-Carlo methods outperform an existing approach to solve LHC in two case studies: linear quadratic control problem, UAV path optimization problem.

A. Preliminaries

A long horizon optimal control problem is described as follows. Let x_k be the state vector for a system at time k, which evolves according to a discrete stochastic process as $x_{k+1} = f(x_k, u_k, w_k)$, where $f(\cdot)$ represents the state-transition mapping, u_k is the control vector, and w_k random disturbance. Let $g(x_k, u_k)$ represent the cost (a real value) of being in state x_k and performing action u_k. The functions f and g are independent of k in our study, but can generally depend on k. The goal is to optimize the control vectors $u_k,\ k = 0, \ldots, H-1$ such that the expected cumulative cost is minimized, i.e., the goal leads to solving the following optimization problem

$$\min_{u_k, k=0,\ldots,H-1} \mathrm{E}\left[\sum_{k=0}^{H-1} g(x_k, u_k)\right], \tag{1}$$

where H is the length of the planning horizon. Let x_0 be the initial state and according to the dynamic programming formulation the optimal cost function is given by

$$J_0^*(x_0) = \min_{u_0} \mathrm{E}\big[g(x_0, u_0) + J_1^*(x_1)\big], \tag{2}$$

where J_1^* represents the optimal cost-to-go from time $k = 1$, and $x_1 = f(x_0, u_0, w_0)$. In this letter, *long horizon* refers to the condition that H is sufficiently large that the optimal policy is approximately *stationary* (independent of k). Solving the above optimization problem is not tractable mainly due to two reasons: the expectation $E[\cdot]$ and the optimal cost-to-go J_1^* are hard to evaluate and are usually approximated by numerical methods or ADP approaches.

An ADP approach called *nominal belief-state optimization* (NBO) [11], [15] was developed primarily to approximate the above expectation. In NBO, the expectation is replaced by a sample state trajectory generated with an assumption that the future noise variables in the system take so called nominal or mean values, thus making the above objective function deterministic. The NBO method was developed to solve a UAV path optimization problem, which was posed as a *partially observable Markov decision process* (POMDP). POMDP generalizes the long horizon optimal control problem described in (1) in that the system state is assumed to be "partially" observable, which is inferred via using noisy observations and Bayes rules. Although the performance of the NBO approach was satisfactory, in that it allowed to obtain reasonably optimal control commands for the UAVs, it ignored the uncertainty due to noise disturbances thus leading to inaccurate evaluation of the objective function. This challenge is typically overcome by Monte-Carlo Tree Search (MCTS) methods [16].

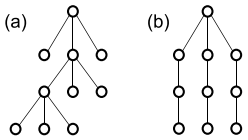


Fig. 1. State trajectory sampling models: (a) tree branching model, (b) non-overlapping branching model.

Inspired from the NBO method and MCTS methods, we develop a new MCTS method called *random sampling - multipath hypothesis propagation* (RS-MHP) and derive convergence results. In this letter, we use the NBO approach as a benchmark for performance assessment since RS-MHP builds on the NBO approach.

II. Random Sampling Multipath Hypothesis Propagation (RS-MHP)

In the NBO method, the expectation is replaced by a sample trajectory of the states (as opposed to random states) generated by

$$\tilde{x}_{k+1} = f(\tilde{x}_k, u_k, \bar{w}_k),\ k = 0, \ldots \tag{3}$$

where $\tilde{x}_0 = x_0$ (initial state or current state), and $\bar{w}_k$ is the mean of the random variable w_k. Thus, the long horizon optimal control problem, with NBO approximation, reduces to

$$\min_{u_k} \sum_{k=0}^{H-1} g(\tilde{x}_k, u_k). \tag{4}$$

The above reduced problem, without the need for evaluating the expectation, can significantly reduce the computational burden in solving the long horizon control problems. However, the downside with this approach is it completely ignores the uncertainty in the state evolution, and may generate severely sub-optimal controls. To overcome this trivialization, we develop a Monte-Carlo approach to approximate the expectation described as follows. We will follow the tree-like sampling approach as in Figure 1(a). For time step $k = 1$, we sample the probability distribution of the noise disturbance N times to generate the samples w_0^i with corresponding probability $p_0^i,\ i = 1, \ldots, N$. Using these, we generate N sample states at $k = 1$ generated according to

$$x_1^i = f(x_0, u_0, w_0^i),\ \forall i. \tag{5}$$

We repeat this sampling approach for time $k = 2$, i.e., we generate N noise samples w_1^i with corresponding probability $p_1^i,\ i = 1, \ldots, N$. Using these noise samples and the sample states from the previous time step, we generate N^2 sample states at $k = 2$ according to

$$x_2^{i,j} = f(x_1^i, u_1, w_1^j),\ \forall i, j. \tag{6}$$

We repeat the above sampling procedure until the last time step $k = H-1$ to generate N^{H-1} possible state evolution

trajectories using N noise samples generated in each time step as depicted in Figure 1(a). Sampling approach in Figure 1(b) will be discussed later.

One can now replace the expectation in (1) with the weighted average of the cumulative cost corresponding to each state evolution trajectory, where the weights are the probabilities or likeliness indices of the trajectories. Although this approach is not novel as many such methods exist in the literature classified as MCTS methods, our study is focused on the convergence of RS-MHP.

Clearly, the number of possible state trajectories in the above approach grow exponentially with the horizon length H. To avoid the exponential growth in our RS-MHP approach, at each time step we retain only M sample states and prune the remaining states, and if the number of sample states at a given time instance is less than or equal to M, we do not perform pruning. For pruning, at each time k, we rank the state trajectories up to time k according to their likeliness (obtained by multiplying the probabilities of all the noise samples that generated the trajectory) and retain the top M trajectories with highest likeliness and prune the rest. With this procedure, at $k = H-1$, there would be only M state trajectories. With pruning, the number of trajectories remains a constant irrespective of the time horizon length, i.e., the method's computational complexity grows polynomially with respect to the horizon length with pruning. An illustration of the above RS-MHP approach is shown in Figure 2 along with the NBO approach. The figure also shows an illustration of the above branch pruning strategy for a simple scenario with $N = 2$ and $M = 2$.

Let $i = 1, \ldots, M$ represent the indices of the M distinct state trajectories with $q_1, q_2, \ldots$ being their likeliness indices, where q_i is evaluated using the probabilities of the noise samples that generate the trajectory i over time. Let J represent the actual objective function as described below

$$J = \mathrm{E}\left[\sum_{k=0}^{H-1} g(x_k, u_k)\right]. \tag{7}$$

We can now approximate the objective function J in four possible ways as described below (assuming $N > M$). Let x_k^i represent the state at time k in the ith state trajectory.

(I) *Sample Averaging:* We can simply approximate the expectation with an average over all possible trajectories as follows:

$$\text{No pruning: } J \approx \tilde{J}_{NP} = \frac{1}{N^{H-1}} \sum_{i=1}^{N^{H-1}} \left(\sum_{k=0}^{H-1} g(x_k^i, u_k)\right)$$

$$\text{With pruning: } J \approx \tilde{J}_P = \frac{1}{M} \sum_{i=1}^{M} \left(\sum_{k=0}^{H-1} g(x_k^i, u_k)\right). \tag{8}$$

(II) *Weighted Sample Averaging:* We can also approximate the expectation with a weighted average with weights being the normalized likeliness indices of the state trajectories given by $q_i, i = 1, \ldots$ (and $\bar{q}_i$ in the pruned case) as follows:

$$\text{No pruning: } J \approx \bar{J}_{NP} = \frac{1}{N^{H-1}} \sum_{i=1}^{N^{H-1}} q_i \left(\sum_{k=0}^{H-1} g(x_k^i, u_k)\right)$$

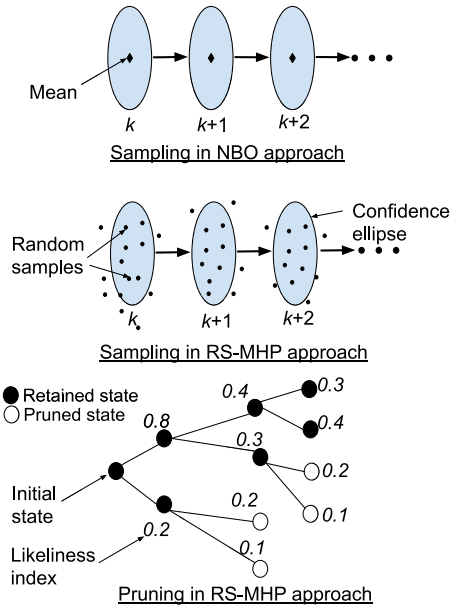


Fig. 2. Top two figures show the sampling probability distributions of noise variables: NBO vs. RS-MHP. The bottom figure shows a branch pruning strategy in RS-MHP.

$$\text{With pruning: } J \approx \bar{J}_P = \frac{1}{M} \sum_{i=1}^{M} \bar{q}_i \left(\sum_{k=0}^{H-1} g(x_k^i, u_k)\right) \tag{9}$$

where $\sum_{i=1}^{N^{H-1}} q_i = N^{H-1}$ and $\sum_{i=1}^{M} \bar{q}_i = M$.

For a given sequence of control decisions $u_0, u_1, \ldots$, let g_i denote the cost of the ith trajectory given by

$$g_i = \sum_{k=0}^{H-1} g(x_k^i, u_k). \tag{10}$$

Clearly, $g_1, g_2, \ldots$ are identically distributed random variables, but are dependent due to the overlapping state trajectories in the tree-like sampling approach in Figure 1(a), where $\mathrm{E}[g_i] = J, \forall i$.

The below result suggests that with sufficient number of sample state trajectories (large N), the approximation error in $\tilde{J}_{NP}$ becomes small enough to ignore.

Proposition 1: For any given sequence of actions $u_0, u_l, \ldots$, if the random variables $g_1, g_2, \ldots$ have finite variances, $\tilde{J}_{NP}$ converges to J in probability.

Proof: From [17, Ex. 254], we know that $\tilde{J}_{NP} \xrightarrow{\mathrm{P}} J$ if

$$\lim_{|i-j|\to\infty} \mathrm{Cov}(g_i, g_j) = 0, \tag{11}$$

where Cov() represents covariance. Suppose, the sequence $g_1, g_2, \ldots$ is arranged such that g_1 represents the cost for the left-most branch in Figure 1(a), and g_2 representing the second branch from the left, and so on. Clearly, the first $g_1, g_2, \ldots, g_N$ are dependent random variables as they share the same parent node, whereas the next N terms $g_{N+1}, g_{N+2}, \ldots, g_{2N}$, although dependent among themselves, are independent of the previous N terms (as these branches evolve from a separate parent node), and so on. Thus, $\mathrm{Cov}(g_i, g_j) = 0$ if $|i-j| > N$, which implies $\lim_{|i-j|\to\infty} \mathrm{Cov}(g_i, g_j) = 0$. ■

Furthermore, we can apply similar arguments to prove the convergence of $\bar{J}_{NP}$ in probability.

Proposition 2: For a given sequence of actions $u_0, \ldots, u_{H-1}$, if $g_1, g_2, \ldots$ have finite variances, then $\bar{J}_{NP}$ converges to J in probability.

Proof: From [18], we know that if $\tilde{J}_{NP} \xrightarrow{\text{P}} J$ (which is true as shown in Proposition 1), and if the weights $q_1, q_2, \ldots$ are monotonically decreasing, then $\bar{J}_{NP} \xrightarrow{\text{P}} J$. Without loss of generality, we can arrange the trajectory costs g_i such that their likeliness indices are monotonically decreasing, i.e., $q_1 \geq q_2 \geq q_3 \geq \ldots$, which completes the proof. ■

A. Non-Overlapping State Trajectories or Tree Branches

Suppose the state sample trajectories are generated independently of each other, where the state trajectories do not share any common state samples as depicted in Figure 1(b). In this new sampling approach, given $u_0, u_1, \ldots$ are the control decisions over the planning horizon, let p_i represent the cost associated with the ith state trajectory. We can approximate the LHC objective function as follows:

$$\bar{J}_N = \frac{1}{N}\sum_{i=1}^{N} p_i \quad \tilde{J}_N = \frac{1}{N}\sum_{i=1}^{N} q_i p_i, \tag{12}$$

where q_i represents the likeliness index of the ith trajectory and $\sum_i q_i = N$. From propositions 1 and 2, we can verify that $\bar{J}_N \xrightarrow{\text{P}} J$ and $\tilde{J}_N \xrightarrow{\text{P}} J$. Furthermore, since $p_1, p_2, \ldots$ are i.i.d., due to the strong law of large numbers, we can verify that $\bar{J}_N$ converges to J almost surely. We can further derive the rate of convergence (in probability) for a special case as discussed below. The main advantages of this non-overlapping branching approach are: a) we are able to conclude that the approximate cost function from this approach $\bar{J}_N$ converges to the the true cost function almost surely (as opposed to the weaker convergence results discussed previously with overlapping branch models); b) we are able to derive the rate of convergence (in probability) for linear LHC problems as discussed below.

Suppose the state-transition and cost functions are linear (motivated by the fact that the linear models capture the state dynamics well in most control problems) as described below:

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k + w_k,\ w_k \sim \mathcal{N}(0, \Sigma) \\ g(x_k, u_k) &= Cx_k + Du_k, \end{aligned} \tag{13}$$

where $g(x_k, u_k)$ is a scalar function. The cost from the sample trajectory i is given by

$$p_i = \sum_{k=1}^{H} g(x_k^i, u_k) = \sum_{k=1}^{H} (Cx_k^i + Du_k), \tag{14}$$

where x_k^i is the sampled state at time step k from the ith trajectory. Using the linear expressions in (13), we can verify p_i further satisfies the following equation:

$$p_i - \text{E}[p_i] = C\left[\sum_{k=0}^{H-1}\left(\sum_{q=0}^{H-k-1} A^q\right) w_k\right] = C\left[\sum_{k=0}^{H-1} \mathcal{A}_k w_k\right], \tag{15}$$

where $\mathcal{A}_k = \sum_{q=0}^{H-k-1} A^q$.

Proposition 3: For a given sequence of actions $u_0, \ldots, u_{H-1}$

$$\text{P}(|J_N - J| \geq \epsilon) \leq \frac{\text{constant}}{N\epsilon^2}. \tag{16}$$

Proof: Let p represent the cost for a sampled state trajectory. Using 15, we can verify

$$\begin{aligned} \text{Var}(p) &= \text{E}\big[(p - \text{E}[p])^{\text{T}}(p - \text{E}[p])\big] \\ &= C\left[\sum_{k=0}^{H-1} \mathcal{A}_k \Sigma \mathcal{A}_k^{\text{T}}\right] C^{\text{T}}, \end{aligned} \tag{17}$$

which is a real scalar. Thus, $\text{Var}(J_N) = \text{Var}(p)/N$.

Using Chebyshev's inequality, we can verify easily that

$$\text{P}(|J_N - J| \geq \epsilon) \leq \frac{\text{Var}(p)}{N\epsilon^2} = \frac{C\left[\sum_{k=0}^{H-1} \mathcal{A}_k \Sigma \mathcal{A}_k^{\text{T}}\right] C^{\text{T}}}{N\epsilon^2}. \tag{18}$$

Furthermore,

$$\lim_{N\to\infty} \text{P}(|J_N - J| \geq \epsilon) = 0, \tag{19}$$

which shows the convergence in probability as well. ■

From Propositions 1 and 2, it is clear that by choosing a sufficiently large N, we can make the probability of the approximation error negligible. Furthermore, from Proposition 3, by increasing N in RS-MHP (at the expense of increased computational requirements), we can tighten the upper bound on the probability of the approximation error for the non-overlapping tree branching model, i.e., RS-MHP allows a paramterizable trade-off between computational complexity and the optimality of the solution via the choice of N.

III. Case Studies

We implement the above-discussed RS-MHP methods in two case studies: (a) linear quadratic Gaussian control (LQG); (b) path planning for unmanned aerial vehicles (UAVs). These case studies are discussed below.

A. Linear Quadratic Problem

Although there are closed-form solutions for LQG problems, the below example allows us to quantify the benefits of using RS-MHP methods over existing similar methods, particularly NBO. Let the system state evolve according to the following linear equation:

$$x_{k+1} = (1 - a)x_k + au_k + w_k, \quad w_k \sim \mathcal{N}(0, \sigma^2), \tag{20}$$

where $0 < a < 1$ is a constant, and w_k is a random disturbance modeled by a zero-mean Gaussian distribution with variance σ^2. The cost function over the time-horizon H is defined as follows:

$$J = \text{E}\left[r(x_H - T)^2 + \sum_{k=0}^{H-1} u_k^2\right], \tag{21}$$

where r and T are constants. This is a simplified oven temperature control example borrowed from [19].

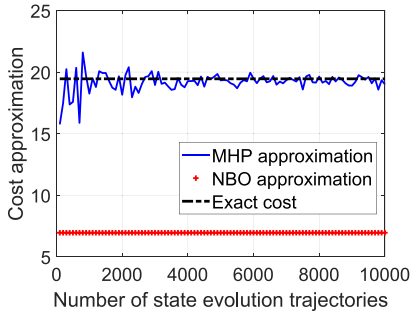


Fig. 3. LQG problem: RS-MHP vs. NBO.

If we apply the traditional NBO method, assuming $H = 2$, the cost function J is approximated (assuming nominal values or zeros for w_0 and w_1) as

$$J_{\text{NBO}} = r\Big((1-a)^2 x_0 + a(1-a)u_0 + au_1 - T\Big)^2 + u_0^2 + u_1^2 \tag{22}$$

and the exact cost function J can be evaluated analytically as

$$J = r\Big((1-a)^2 x_0 + a(1-a)u_0 + au_1 - T\Big)^2 + u_0^2 + u_1^2 + r\sigma^2\Big((1-a)^2 + 1\Big). \tag{23}$$

We notice the approximation error due to the NBO method is

$$|J_{\text{NBO}} - J| = r\sigma^2\Big((1-a)^2 + 1\Big). \tag{24}$$

This approximation error for a generic time-horizon H is given by

$$|J_{\text{NBO}} - J| = r\sigma^2 \sum_{n=0}^{H-1} (1-a)^{2n}. \tag{25}$$

The above expression suggests that the NBO approximation error can be significantly high depending on the parameters a, σ, and r. With RS-MHP approximation, the cost function reduces to

$$J_{\text{RS-MHP}} = \frac{1}{P}\left(\sum_{i=1}^{P} r(x_H^i - T)^2\right) + \sum_{k=0}^{H-1} u_k^2, \tag{26}$$

where P is the number of state-trajectories generated using the RS-MHP approach, and x_H^i is the final state in the ith trajectory. Proposition 1 shows that the approximation error due to the above RS-MHP method converges (in probability) to zero. We verify this result with a numerical simulation, where we implement the NBO and the RS-MHP methods with the following assumptions: $x_0 = 0$, $r = 10$, $T = 1$, $H = 2$, $u_0 = 0.55$, $u_1 = 0.17$, $\sigma = 1$. We vary P from 100 to 10000 with increments of 100. Figure 3 shows the cost function approximated using RS-MHP and NBO methods. The figure clearly demonstrates that the error due to NBO approximation can be significantly high, while RS-MHP performs better in cost approximation.

B. UAV Path Planning Problem

We consider a UAV path planning problem, where the goal is to optimize the kinematic controls of a UAV to maximize a target tracking performance measure. Here, the UAV is assumed to be equipped with a sensor on-board that generates the location measurements of the target (a ground-based moving vehicle) corrupted by random noise. A detailed description of the problem can be found in [11]. In [11], we posed this problem as a *partially observable Markov decision process* (POMDP), where the POMDP led to solving a long horizon optimal control problem. We applied the NBO approach to solve the above POMDP. The resulting UAV path optimization problem is summarized as follows:

$$\min_u \text{E}\left[\sum_{k=0}^{H-1} \text{tr}\,(\mathbf{P}_k(u))\right] \xrightarrow{\text{NBO approx.}} \min_u \sum_{k=0}^{H-1} \text{tr}\,(\hat{\mathbf{P}}_k(u)),$$

where $\mathbf{P}_k(u)$ (a random variable) represents the error covariance matrix corresponding to the state of the system, tr() represents the matrix trace operator, u is the sequence of UAV kinematic controls (e.g., forward acceleration and bank angle) applied over the discrete time planning horizon of length H steps. After NBO approximation, the expectation over the random evolution of $\mathbf{P}_k(u)$ is replaced with the nominal sequence of the state covariance matrices tr $(\hat{\mathbf{P}}_k(u))$.

We now approximate the above objective function using the RS-MHP approach as follows:

$$\min_u \text{E}\left[\sum_{k=0}^{H-1} \text{tr}\,(\mathbf{P}_k(u))\right] \xrightarrow{\text{RS-MHP approx.}}$$
$$\min_u \frac{1}{N_T} \sum_{i=1}^{N} \sum_{k=0}^{H-1} \text{tr}\,(\tilde{\mathbf{P}}_k^i(u)),$$

where $\tilde{\mathbf{P}}_k^i$ represents the state covariance matrix obtained from the ith state trajectory generated from the RS-MHP approach, and N_T is the number of state trajectories. We implement this approach in MATLAB and run a Monte-Carlo study to see the impact of N_T on the performance of the above UAV path planning algorithm, which is measured by the average target location estimation error. Figure 4(a) shows the cumulative distribution of average target location errors from the RS-MHP approach with $H = 6$, and for N_T set to 50, 100, and 250. The figure shows a gradual increase in the algorithm's performance with increasing N_T as expected. This result, as expected, also suggests that pruning methods (discussed in the previous section) would degrade the performance of the RS-MHP methods but can provide gains in terms of computational intensity. RS-MHP has better capability in approximating the expectation operator in 1 than the NBO approach as we consider multiple hypotheses of state trajectories in RS-MHP as opposed to a single hypothesis in NBO as demonstrated in Figure 4(b).

IV. Conclusion

In this letter, we developed a Monte-Carlo tree search method called *random sampling - multipath hypothesis propagation* or RS-MHP to approximate the expectation operator in long horizon optimal control problems. Although variants

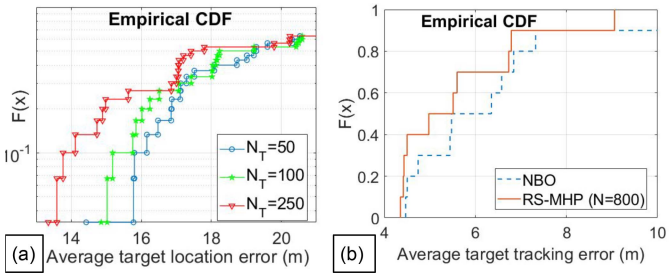


Fig. 4. (a) Cumulative distribution of average target location errors. Here N_T represents the number of state evolution trajectories. (b) Cumulative distribution of target location errors: NBO vs. RS-MHP.

of these methods exist in the literature, we focused on the convergence analysis of these approximation methods. The basic theme of these methods is to evolve the system state over multiple trajectories into the future while sampling the noise disturbances at each time-step. We derive convergence results that show that the cost approximation errors from our RS-MHP methods converge (in probability) toward zero as the sample size increases. We conducted a numerical study to assess the performance of our methods in two case studies: linear quadratic control problem and UAV path optimization problem. In both case studies, we demonstrated the benefits of our approach against an existing approach called *nominal belief-state optimization* or NBO (used as a benchmark).

ACKNOWLEDGMENT

The authors would like to thank Dr. Nicolas Lanchier, Arizona State University, for his valuable inputs on the convergence results discussed in this letter.

REFERENCES

[1] E. K. P. Chong, C. M. Kreucher, and A. O. Hero, "Partially observable Markov decision process approximations for adaptive sensing," *Discrete Event Dyn. Syst.*, vol. 19, no. 3, pp. 377–422, Sep. 2009.

[2] C. Wang, S. Lei, P. Ju, C. Chen, C. Peng, and Y. Hou, "MDP-based distribution network reconfiguration with renewable distributed generation: Approximate dynamic programming approach," *IEEE Trans. Smart Grid*, vol. 11, no. 4, pp. 3620–3631, Jul. 2020.

[3] H. Su, H. Zhang, H. Jiang, and Y. Wen, "Decentralized event-triggered adaptive control of discrete-time nonzero-sum games over wireless sensor-actuator networks with input constraints," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 10, pp. 4254–4266, Oct. 2020.

[4] D. P. Bertsekas and D. A. Castanon, "Rollout algorithms for stochastic scheduling problems," *J. Heuristics*, vol. 5, pp. 89–108, Apr. 1999.

[5] E. K. P. Chong, R. L. Givan, and H. S. Chang, "A framework for simulation-based network control via hindsight optimization," in *Proc. 39th IEEE Conf. Decis. Control*, Sydney, NSW, Australia, pp. 1433–1438,Apr. 2000.

[6] G. Wu, E. K. P. Chong, and R. Givan, "Burst-level congestion control using hindsight optimization," *IEEE Trans. Autom. Control*, vol. 47, no. 6, pp. 979–991, Jun. 2002.

[7] D. Bertsekas, "Feature-based aggregation and deep reinforcement learning: A survey and some new implementations," *IEEE/CAA J. Automatica Sinica*, vol. 6, no. 1, pp. 1–31, 2019.

[8] Y. Liu, E. K. P. Chong, A. Pezeshki, and Z. Zhang, "A general framework for bounding approximate dynamic programming schemes," *IEEE Control Syst. Lett.*, vol. 5, no. 2, pp. 463–468, Apr. 2021.

[9] Z. Zhang, E. K. P. Chong, A. Pezeshki, and W. Moran, "String submodular functions with curvature constraints," *IEEE Trans. Autom. Control*, vol. 61, no. 3, pp. 601–616, Mar. 2016.

[10] D. P. Bertsekas, "Dynamic programming and suboptimal control: A survey from ADP to MPC," *Eur. J. Control*, vol. 11, nos. 4–5, pp. 310–334, 2005. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0947358005710402

[11] S. Ragi and E. K. P. Chong, "UAV path planning in a dynamic environment via partially observable Markov decision process," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 49, no. 4, pp. 2397–2412, Oct. 2013.

[12] S. Ragi and H. D. Mittelmann, "Mixed-integer nonlinear programming formulation of a UAV path optimization problem," in *Proc. Amer. Control Conf.*, Seattle, WA, USA, 2017, pp. 406–411.

[13] L. Kocsis and C. Szepesvári, "Bandit based monte-carlo planning," in *Proc. 17th Eur. Conf. Mach. Learn.*, 2006, pp. 282–293.

[14] S. Ragi and H. D. Mittelmann, "Random-sampling multipath hypothesis propagation for cost approximation in long-horizon optimal control," in *Proc. IEEE Conf. Control Technol. Appl.*, Montreal, QC, Canada, 2020, pp. 14–18.

[15] S. A. Miller, Z. A. Harris, and E. K. P. Chong, "A POMDP framework for coordinated guidance of autonomous UAVs for multitarget tracking," *EURASIP J. Adv. Signal Process.*, vol. 2009, Jan. 2009.

[16] C. B. Browne *et al.*, "A survey of Monte Carlo tree search methods," *IEEE Trans. Comput. Intell. AI in Games*, vol. 4, no. 1, pp. 1–43, Mar. 2012.

[17] T. Cacoullos, *Exercises in Probability*. New York, NY, USA: Springer, 1989.

[18] N. Etemadi, "Convergence of weighted averages of random variables revisited," *Proc. Amer. Math. Soc.*, vol. 134, no. 9, pp. 2739–2744, 2006.

[19] D. P. Bertsekas. *Lecture on Reinforcement Learning and Optimal Control*. Accessed: Mar 15, 2020. [Online]. Available: http://www.mit.edu/ dimitrib/Slides_Lecture2_RLOC.pdf

Average Consensus-Based Data Fusion in Networked Sensor Systems for Target Tracking

Md Ali Azam
Electrical Engineering
South Dakota School of Mines and Technology
mdali.azam@mines.sdsmt.edu

Shawon Dey
Electrical Engineering
South Dakota School of Mines and Technology
shawon.dey@mines.sdsmt.edu

Hans D. Mittelmann
School of Mathematics and Statistical Sciences
Arizona State University
mittelmann@asu.edu

Shankarachary Ragi
Electrical Engineering
South Dakota School of Mines and Technology
Shankarachary.Ragi@sdsmt.edu

***Abstract*—Decentralized and distributed autonomous sensing over networked sensor systems has many applications in surveillance, Internet of Things (IoT), autonomous cars, and UAV swarms tactics. In this study, we develop an average consensus-based decentralized data fusion approach for a target tracking application. Specifically, we extend the standard average consensus algorithm to merge the local state estimate information with that of the neighbors. We test the performance of our consensus based data fusion approach for various network configurations. We also perform numerical studies to compare the performance of our approach against the standard Bayesian data fusion approach.**

***Index Terms*—Networked sensor systems, Decentralized average consensus, Sensor data fusion, Target tracking**

I. Introduction

Autonomous and adaptive sensing has applications such as target tracking, surveillance [1], autonomous car navigation [2], and UAV swarm tactics [3], [4]. Particularly, target tracking via adaptive sensing is becoming increasingly important in autonomous car industry for accurate pedestrian detection and tracking [5]. Sensors such as RADAR, LIDAR, optical sensors, thermal sensors are typically used to measure the target state including its position, velocity, and acceleration. Target tracking with multiple sensors was studied in the past, e.g., [3], where a central fusion node was responsible for making sensing decisions (e.g., sensor location - assuming sensor mounted on a UAV) for all the sensors combined. Clearly, sensing decisions optimized for all the sensors combined provides the best target tracking performance as these decisions are coupled via sensor data fusion. The main drawback with these centralized decision making methods is that they are computationally intensive as the computational complexity is exponential in the decision space and the number of sensors. To address this challenge, we investigated decentralized strategies in the past to some extent [4].

This work was supported in part by Air Force Office of Scientific Research under grant FA9550-19-1-0070.

In this study, we develop a decentralized autonomous sensing method over a networked sensor system for a target tracking application. Specifically, we extend an existing approach called *average consensus algorithm* to perform decentralized data fusion while tracking a moving target. The sensor network is modeled by an undirected graph, which is assumed to be non-time varying. Each sensor generates a noisy measurement of the target state. The presence of an edge between the nodes or sensors means that the sensors are allowed to exchange information/messages for data fusion. In this study, we assume that each sensor maintains a local tracker (or tracking algorithm, e.g., Kalman filter), which updates its local target state estimate using the locally generated sensor measurements and the information it receives from its neighbors. We measure the performance of the above consensus algorithm with *average target tracking error* - the mean-squared error between the target state (ground truth) and the estimate. As a benchmark, we also implement the standard Bayesian data fusion approach for performance comparison.

The authors of [6] have surveyed both classical approaches and recent advances in multi-sensor data fusion and consensus filter for sensor networks. The authors of [7] reviewed the key theories and methodologies of distributed multi-sensor data fusion and discussed their advantages like graceful degradation, scalability, and interchangeability. *Average consensus* was studied previously in distributed computing [8] and for achieving consensus among agent values (a real number possibly representing its opinion or state). In [9], a distributed consensus algorithm was developed for obtaining the averages of the node data over networks with large volume of data. N. Gupta et. al. proposed an asynchronous distributed average consensus algorithm [10] to guarantee information-theoretic privacy in multi-agent systems. In [11], the authors provide a theoretical framework for analysis of consensus algorithms for multi-agent networked systems. In [12], the authors developed a distributed consensus tracking filter to solve the target tracking problem. The authors in [13] discussed algorithms for solving decentralized consensus optimization problems.

978-1-7281-3783-4/20/$31.00 ©2020 IEEE

A. Key Contributions

- We extend the *average consensus* algorithm [9] to track a moving target via a decentralized network of sensors. We compare the performance of this method against a standard benchmark method - *decentralized Bayesian data fusion approach* [7].
- We perform a numerical study to quantify the impact of various sensor network configurations (e.g., varying degrees of the nodes) on the performance of the *average consensus* algorithm.

The rest of the paper is organized as follows. Section II presents the problem specification and the objectives. Section III provides the problem formulation and the methods followed by the simulation results in Section IV. Finally, we provide concluding remarks and future scope in Section V.

II. Problem Specification

In our study, we assume there are n sensors tracking a moving target in a decentralized setting, where the sensors are connected via an undirected graph. The target is assumed to be moving on a 2-D plane, where the motion is modeled via a stochastic process, i.e., the state-transition law is a linear model with zero-mean Gaussian noise. We assume the sensor measurement law is also linear with zero-mean Gaussian noise. Thus, each sensor maintains and updates a local target state estimate via Kalman filtering algorithm.

We assume that the sensors have limited battery power and computational capabilities, which sets limitations on the sensors in terms of how they generate measurements and communicate with other sensors. Specifically, we assume that the sensors can either sense (generate target measurements) or exchange information with neighboring sensors, but not simultaneously.

Communications: The sensors have communications capabilities, i.e, each sensor can transmit or receive data to/from the sensors they share edges in the network graph. We further assume that the communications delay is negligible.

Sensor network: The n sensors are assumed to be connected via an undirected graph. Each sensor i has a set of neighbors, denoted by $N(i)$, where sensor $j \in N(i)$ if there is an edge connecting j with i.

Performance measure: We measure the performance of the algorithms using *average tracking error*, which is the mean-squared error between the target state and the estimates averaged over all the sensors and over time.

Objective: The objective is to compare the performance the *average consensus* algorithm against the standard *decentralized Bayesian data fusion* technique for target tracking with a decentralized sensor network. We measure the performance of these algorithms for different sensor network configurations.

III. Problem Formulation

A. Tracking Approach

In our study, $\{1, ..., n\}$ represent the sensor indices, and S_i represents the 2D location of sensor i. The target's motion is described by a linear state-space model (specifically *constant velocity* model [14]):

$$x_k = Ax_{k-1} + \theta_k, \quad \theta_k \sim \mathcal{N}(0, Q) \tag{1}$$

where x_k is the state of the target at time k (which includes the target's 2D location, 2D velocity, and 2D acceleration), A is a state transition matrix, and θ_k is process noise with zero-mean normal distribution with co-variance matrix Q. Sensor i generates a position measurement z_k^i given by:

$$z_k^i = Hx_k + v_k^i \tag{2}$$

where H is the observation matrix given by

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix},$$

which means that the sensors only generate positional measurements. Here $v_k^i \sim \mathcal{N}(0, R(x_k, S_i))$ is the random measurement noise modeled as a zero-mean normal distribution, where the co-variance matrix $R(x_k, S_i)$ captures the dependence of the noise characteristics on the location of the target with respect to the sensor. Here, R_k reflects 10% range uncertainty and 0.01π radian angular uncertainty. Since the state and the observation laws are linear with zero-mean Gaussian noise disturbances, we run Kalman filter at each sensor node to maintain and update the target state posterior distribution with mean and co-variance given by $\hat{x}_{k|k}^i$ and $P_{k|k}^i$.

Clearly, if the sensors do not exchange any information, the tracking performance suffers at each node. The sensors are connected via an undirected graph, where the presence of an edge between nodes i and j means that the sensors are allowed to exchange information. So, we extend an approach called *average consensus* algorithm to allows sensors to exchange information in a manner that improves the target tracking performance across the sensor network.

B. Average Consensus

Average consensus algorithms let a network of sensors or agents reach a common consensus on certain attributes (real numbers) such as the agent opinions, sensor measurements, etc. Specifically, in these approaches, each agent or sensor updates/replaces (in an iterative manner over time) its local value by taking a weighted average between its local value and the values from all the neighbors. We extend this approach to let the sensors in our problem reach a common consensus on their state estimate parameters (mean vector and covariance matrix). Let y_k^i is a vector obtained by concatenating $\hat{x}_{k|k}^i$ and $P_{k|k}^i$ into a column vector at sensor i at time k. $N(i)$ is the set of neighbors for i^{th} sensor. Average consensus algorithm applied to our problem is captured by the following equation:

$$y_{k+1}^i = \frac{\alpha y_k^i + (1-\alpha)\sum_{j \in N(i)} y_k^j}{\alpha + |N(i)|(1-\alpha)}, \quad \forall i \tag{3}$$

where α is a weighting parameter.

This algorithm achieves its objective if all the sensors reach consensus on the state estimation parameters, i.e., $y_k^i = y_k^j$ for all i, j.

C. Decentralized Bayesian data fusion

Multi-sensor data fusion techniques can be applied in both centralized and decentralized settings. In our study, we use decentralized Bayesian data fusion techniques over the sensor network. Each sensor has a local state estimate x_k^i which is updated in each time step by fusing x_k^i with the estimates from its neighboring sensors as given by the following equations (using standard Bayes rules [15]).

$$P_{k+1}^i = \left((P_k^i)^{-1} + \sum_{j=1}^{N(i)} (P_k^j)^{-1} \right)^{-1} \tag{4}$$

$$\hat{x}_{k+1}^i = P_{k+1}^i \left((P_k^i)^{-1} \hat{x}_k^i + \sum_{j=1}^{N(i)} (P_k^i)^{-1} \hat{x}_k^j \right)^{-1} \tag{5}$$

IV. Simulation Results

We implement our methods for a scenario with 10 sensors, i.e., $n = 10$. We set $\alpha = 0.5$ in the following numerical studies except when we evaluate the performance of our algorithms with varying α. We compare the performance of the average consensus algorithm against the decentralized Bayesian data fusion approach for different sensor network configurations with *average tracking error* (defined earlier) as the performance measure. In our numerical studies, we use error bars with one standard deviation to show the spread of the performance measure for multiple network graphs generated from a given configuration as discussed below (examples of configurations in Fig. 1).

Configuration I. This corresponds to a network where each sensor has the same degree, where the degree is given by D, which is referred to as *network degree*. We generate a random graph with n sensors and D network degree.

Configuration II. In this configuration, we generate a random graph with *edge probability* P_e, where P_e represents a probability of an edge existing between two sensors. We start with n sensors with no edges at the beginning, and we create an edge between every pair of sensors with probability P_e. We repeat this process until we get a connected network.

Configuration III. This corresponds to a network with a total number of edges N_e in a connected network.

As sensors typically have limited computational capability and limited battery life, we assume they can run only tracking algorithm while generating sensor measurements or only communicate with neighbors, i.e., run the consensus or data fusion methods as described in Section II. Specifically, in our study, sensors track the target for M time steps and apply the consensus/data fusion algorithms in the next M time steps, and repeat the process. During the M time steps when the consensus/data fusion algorithms are being applied, sensors update the state estimates of the target without the measurements, i.e., perform only prediction step and ignore the measurement update step. In other words, the uncertainty in the target state estimate steadily increases during these M time-steps.

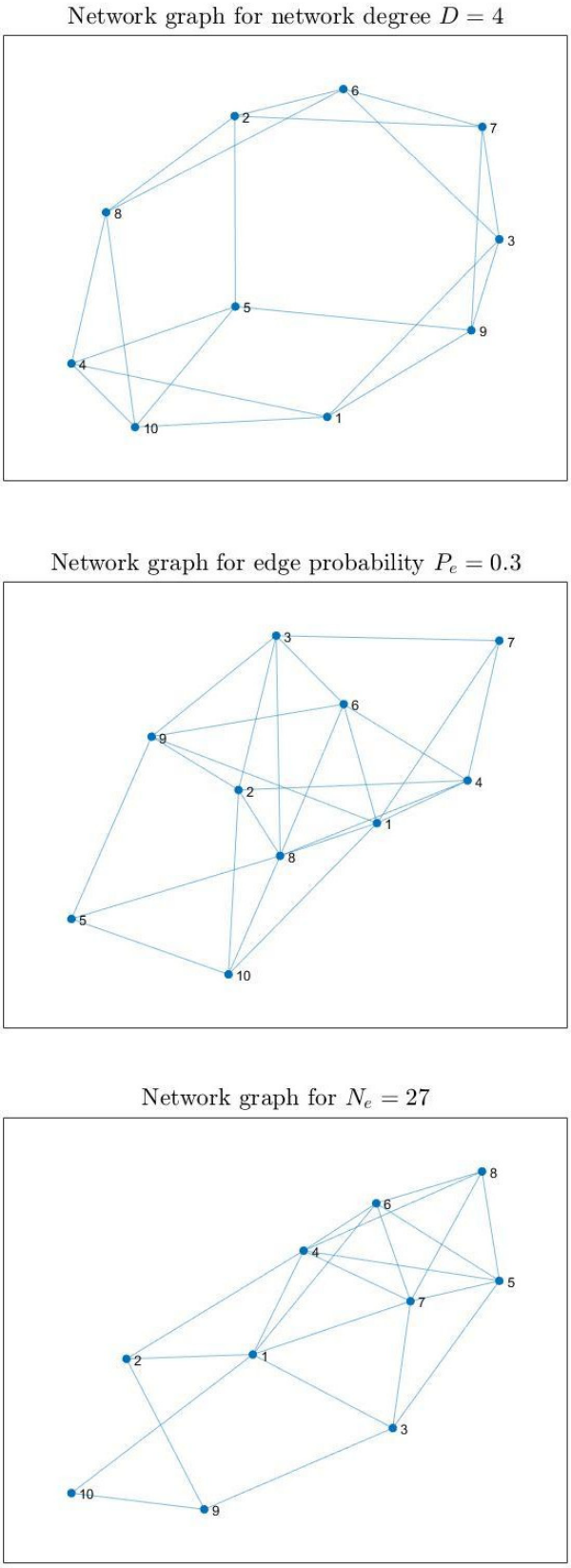


Fig. 1. Examples of configurations (configuration I, II, III from top to bottom)

Let Z represent the total number of time steps in our simulation run time. We set $Z = 300$ in this study. We define the *average tracking error* measure as follows:

$$\frac{1}{Z}\frac{1}{n}\sum_{k=1}^{Z}\sum_{i=1}^{n}\left\|\hat{x}_k^i - x_k\right\|_2^2$$

where x_k represents the ground truth at time k, and $\|\cdot\|_2$ is the Euclidean norm.

A. Average tracking error vs. M

We now compare the performance of *average consensus* and *decentralized Bayesian data fusion* algorithms for different values of M on five randomly generated graphs for $n = 10$. We evaluate the average tracking error, as defined earlier, for each value of M considered. Fig. 2 shows the average tracking error as a function of M, where $M \in \{3, 6, 9, \ldots, 24\}$. The figure suggests that the average consensus algorithm outperforms the data fusion approach for all values of M considered. The consensus algorithm seems to be more effective in merging information from multiple sensors than the standard decentralized Bayesian data fusion approach.

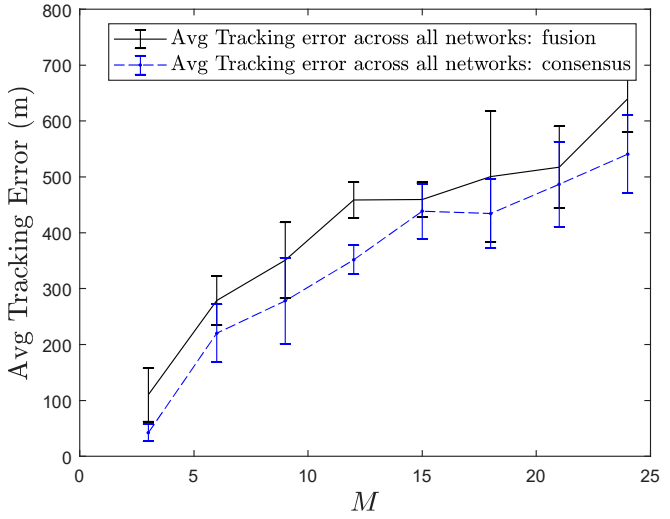


Fig. 2. Average tracking error across all sensors with respect to number of time steps, M

Fig. 3 represents average tracking error as a function of M for $M \in \{1, 2, \ldots, 9\}$. Fig. 3 shows that the average consensus and decentralized Bayesian data fusion algorithm give better performance for $M = 2$ and $M = 3$ respectively compared to all other values of M considered here.

B. Average tracking error for configuration I

We now evaluate the average tracking error as a function of the network degree as shown in Fig. 4. We compare the performance of these two algorithms on five randomly generated graphs for $M = 1$ and $n = 10$. We observe that the performance of both algorithms increase as the network degree increases. Furthermore, from Fig. 4, we observe that the average consensus algorithm performs better than the decentralized Bayesian data fusion method. This is an expected behavior since with greater network degree, the sensors have better capability in merging information from other sensors.

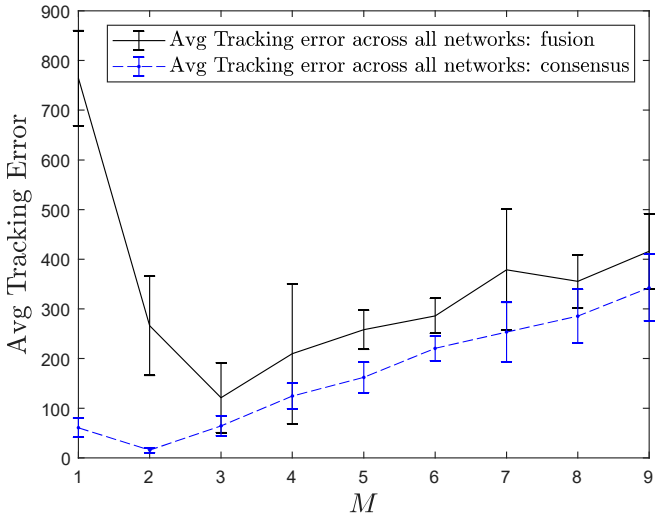


Fig. 3. Average tracking error across all sensors with respect to number of time steps, M

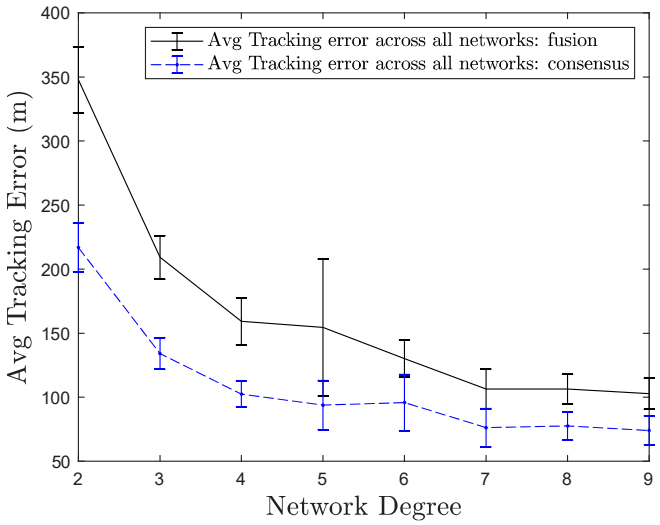


Fig. 4. Average tracking error across all sensors for configuration I

C. Average tracking error for configuration II

We now perform the same numerical study for a randomly generated graph by using Configuration II with different values of P_e drawn from the set $\{0.1, 0.2, \ldots, 1\}$. For each P_e, we generate 10 graphs. Fig. 5 shows that, for both algorithms, the average tracking error decreases with respect to P_e, which is expected since the network connectivity increases with increasing P_e. We also notice that the consensus algorithm outperforms the decentralized Bayesian data fusion approach for each P_e.

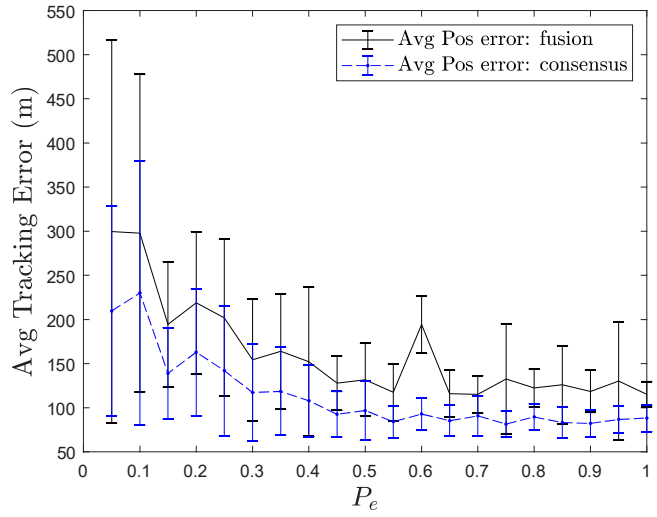


Fig. 5. Average tracking error across all sensors with respect to edge probability P_e

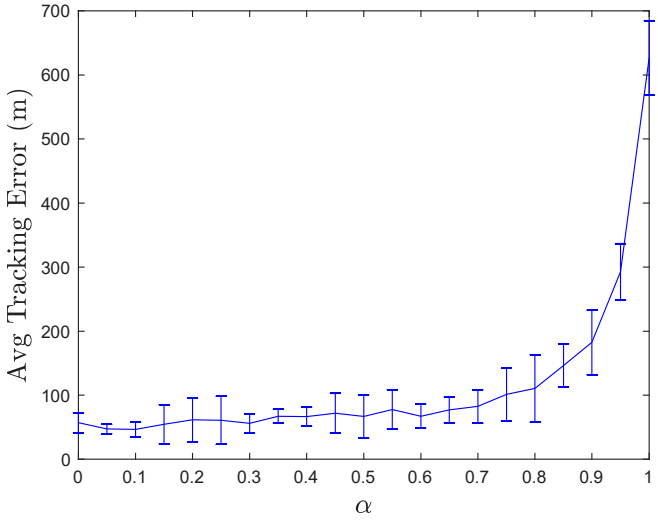


Fig. 7. Average tracking error across all sensors with respect to weighting parameter α

D. Average tracking error for configuration III

We now evaluate the average tracking error for different value of N_e as shown in Fig. 6. We generate (randomly) five graphs with Configuration III for this study. We observe that with increasing N_e, the performance of both of the algorithms increases. We fit 5^{th} degree polynomial curves for the performance plots in Fig. 6, which characterize the variation of the performance of the algorithms as a function of N_e.

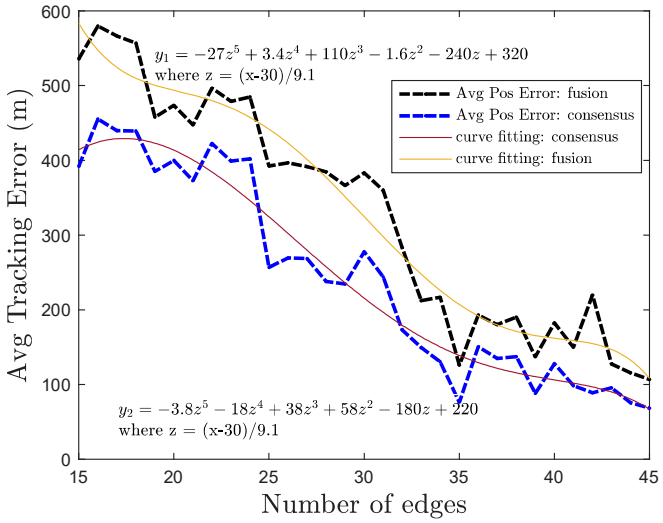


Fig. 6. Average tracking error across all sensors with respect to number of edges

E. Average tracking error for weighting parameter α

In this part, we study the performance of the average consensus algorithm with respect to the weighting parameter α. Here, $\alpha = 0$ means that the consensus algorithm replaces the local sensor's state estimate with the average of its neighbors' estimates. On the other hand, $\alpha = 1$ means that the consensus algorithm ignores the estimates from the neighbors and simply retains the local state estimate. For different values of α in the interval $[0, 1]$, we evaluate the average tracking error, as shown in Fig. 7. The figure shows that the average tracking error increases significantly when the value of α is close to 1.

V. Conclusion and Future Scope

In this study, we extended the *average consensus* algorithm for decentralized data fusion over a networked sensor system for target tracking. We studied the performance of our extended average consensus algorithm numerically for different network configurations and compared with standard Bayesian decentralized Bayesian data fusion as a benchmark. We found that the average consensus algorithm outperformed the decentralized Bayesian data fusion for all network configurations considered in this study. In the future, we will consider decentralized data fusion over time-varying sensor networks and develop graph-theoretic solutions to maximize the data fusion performance.

References

[1] C. Carthel, S. Coraluppi and P. Grignan, "Multisensor tracking and fusion for maritime surveillance," *2007 10th International Conference on Information Fusion*, Quebec, Que., 2007, pp. 1-6.

[2] A. Manzanilla, S. Reyes, M. Garcia, D. Mercado and R. Lozano, "Autonomous Navigation for Unmanned Underwater Vehicles: Real-Time Experiments Using Computer Vision," in *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1351-1356, April 2019.

[3] S. Ragi, E. K. P. Chong, "Dynamic UAV path planning for multitargte tracking," *American Control Conference,* pp. 3845-3850, 2012.

[4] S. Ragi, E. K. P. Chong, "Decentralized Guidance Control of UAVs with Explicit Optimization of Communication," *J Intell Robot Syst*, vol. 73 pp. 811-822, 2014.

[5] S. Blackman and R. Popoli, "Design and analysis of modern tracking systems," Boston, USA: Artech House, 1999.

[6] W. Li, Z. Wang, G. Wei, L. Ma, J. Hu and D. Ding, "A Survey on Multisensor Fusion and Consensus Filtering for Sensor Networks," *Discrete Dynamics in Nature and Society*,Volume 2015, Article ID 683701, 2015.

[7] M. A. Bakr, S. Lee, "Distributed Multisensor Data Fusion under Unknown Correlation and Data Inconsistency," *Sensors*, Oct. 2017.

[8] N. A. Lynch, Distributed Algorithms. San Francisco, CA: Morgan Kaufmann, 1997.

[9] T. C. Aysal, M. J. Coates and M. G. Rabbat, "Distributed Average Consensus With Dithered Quantization," *IEEE Transactions on Signal Processing*, vol. 56, no. 10, pp. 4905-4918, Oct. 2008.

[10] N. Gupta, J. Katz, and N. Chopra. "Statistical Privacy in Distributed Average Consensus on Bounded Real Inputs." *arXiv preprint* arXiv:1903.09315 (2019).

[11] R. Olfati-Saber, J. A. Fax, R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. IEEE*, vol. 95, no. 1, pp. 215-233, Jun. 2007.

[12] S. Zhu, C. Chen, W. Li, B. Yang, X. Guan, "Distributed Optimal Consensus Filter for Target Tracking in Heterogeneous Sensor Networks," *IEEE Transactions on Cybernetics*,Volume: 43 , Issue: 6 , Dec. 2013.

[13] A. Nedich, A. Olshevsky, and W. Shi, "Decentralized Consensus Optimization and Resource Allocation," Lecture Notes in Mathematics, Springer Verlag, Vol. 2227, pp. 247287, 2018 (contributions of 2017 LCCC Workshop).

[14] X. R. Li and Y. Bar-Shalom, "Design of an interacting multiple model algorithm for air traffic control tracking," in *IEEE Transactions on Control Systems Technology*, vol. 1, no. 3, pp. 186-194, Sept. 1993.

[15] J. K. Hackett and M. Shah, "Multi-sensor fusion: a perspective," Proceedings., *IEEE International Conference on Robotics and Automation*, Cincinnati, OH, USA, 1990, pp. 1324-1330 vol.2

A decision theoretic approach for waveform design in joint radar communications applications

Shammi A. Doly *, Alex Chiriyath†, Hans D. Mittelmann‡, Daniel W. Bliss†
and Shankarachary Ragi*
*Department of Electrical Engineering, South Dakota School of Mines and Technology, Rapid City, SD 57701
Email: shammi.doly@mines.sdsmt.edu & shankarachary.ragi@sdsmt.edu
†School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, AZ 85287
Email: achiriya@asu.edu & d.w.bliss@asu.edu
‡School of Mathematical and Statistical Sciences, Arizona State University, Tempe, AZ 85287
Email: mittelmann@asu.edu

2020 54th Asilomar Conference on Signals, Systems, and Computers | 978-0-7381-3126-9/20/$31.00 ©2020 IEEE | DOI: 10.1109/IEEECONF51394.2020.9443402

***Abstract*—In this paper, we develop a decision theoretic approach for radar waveform design to maximize the joint radar communications performance in spectral coexistence. Specifically, we develop an adaptive waveform design approach by posing the design problem as a *partially observable Markov decision process* (POMDP), which leads to a hard optimization problem. We extend an approximate dynamic programming approach called *nominal belief-state optimization* to solve the waveform design problem. We perform a numerical study to compare the performance of the proposed POMDP approach with the commonly used myopic approaches.**

I. Introduction

Spectral congestion is forcing legacy radar band users to investigate methods of cooperation and co-design with a growing number of communications applications [1]. The co-design of radar and wireless communications systems faces several challenges such as interference, radar and communications decoupling, and dynamic user (radar and communications) requirements. The studies in [2], [3] provide a detailed overview of the challenges and research directions in the "spectral" coexistence of radar and communications. From the study in [4], the quality of the radar return and also the communications rate is mainly determined by the spectral shape of the waveform. Moreover, one of the key challenges for any waveform design method is to meet dynamic user needs. To address these challenges, in this paper, we develop waveform shaping methods that are adaptive, and can trade-off between competing performance objectives.

A waveform design method can most effectively meet the dynamic user needs if it predicts the future user needs and allocates the resources accordingly. Previous research has considered waveform design for joint radar-communications systems, for example [5], [6]. However, existing methods often do not meet dynamic performance requirements, as they tend to be *greedy* in that they only maximize short-term performance for instantaneous benefits. For problems with dynamic performance requirements, long-term performance is critical as decisions (to choose a particular waveform) at current time epoch may lead to regret in the future.

To address these challenges, we develop an adaptive waveform design method for joint radar-communications systems based on the theory of *partially observable Markov decision process* (POMDP) [7]. Specifically, we formulate the waveform design problem as a POMDP, after which the design problem becomes a matter of solving an optimization problem. In essence, the POMDP solution provides us with the optimal decisions on the waveform design parameters [8]. However, the optimization problems resulting from POMDPs are hard to solve exactly. There is a plethora of approximation methods called *approximate dynamic programming* methods or ADP methods, as surveyed in [7]. To this end, we extend one of the computationally least intensive ADP approaches called *nominal belief-state optimization* (NBO) [8].

The POMDP framework has a natural look-ahead feature, i.e., it can trade-off short-term for long-term performance. This feature lets the POMDP naturally anticipate the dynamic user needs and optimize the resources (waveforms) to actively meet the user's needs. Typically, one studies these adaptive methods under "cognitive radio (radar)," which has a rich literature. However, this project brings formalism to these methods by posing the waveform design problem as a POMDP. This particular waveform design problem has not been studied before. Recently, POMDPs were used in [9] to develop adaptive methods for "cognitive radar," but in a different context, where the focus was on optimizing radar measurement times and not on waveform shaping. The current waveform design problem is related to a class of problems called *adaptive sensing*, where POMDP was already a proven effective framework [8], [10].

We assume that the environment consists of a maneuvering radar target, obstacle blocking radar line-of-sight, a communications user, and a joint radar-communications system node. The joint radar-communications node can sense the environment to extract target parameter information or can communicate with other communications nodes, and can also act as communications relay. The joint node can simultaneously estimate the target parameters from the radar return and decode a received communications signal. We co-design the

The work of S. Doly, S. Ragi, and H. D. Mittelmann was supported in part by the Air Force Office of Scientific Research under grant FA9550-19-1-0070.

TABLE I: Survey of Notation

Variable	Description
B	Total System Bandwidth
P_{radar}	Radar power
T_{temp}	Effective temperature
b	Communication propagation loss
P_{com}	Communications power
a	Combined antenna gain
N	Number of targets
σ^2_{CRLB}	Cramer-Rao lower bound
σ^2_{noise}	Thermal noise
σ^2_{proc}	Process noise variance
TB	Time–bandwidth product
δ	Radar duty factor
w	Measurement noise
ζ_k	Mean vector noise
τ	Time delay to m^{th} target
α	Weighting parameter
R_{comm}	Communications rate
R_{est}	Radar estimation rate
P_k	Error covariance matrix
T_{pri}	Pulse repetition interval
H	Planning horizon length

joint radar-communications system so that radar and communications systems can cooperatively share information with each other and mutually benefit from the presence of the other. In this paper, we consider target range or time-delay to be the target parameter of interest.

Table I shows the notations employed in this paper.

II. Joint Radar-Communications Preliminaries

A. Successive Interference Cancellation Receiver Model

In this section, we present the receiver model called *Successive Interference Cancellation* (successive interference cancellation (SIC)). SIC is the same optimal multiuser detection technique used for a two user multiple-access communications channel [2], [11], except it is now reformulated for a communications and radar user instead of two communications users. We assume we have some knowledge of the radar target range (or time-delay) up to some random fluctuation (also called process noise) from prior observations. We model this process noise, $n_{\text{proc}}(t)$, as a zero-mean random variable. Using this information, we can generate a predicted radar return and subtract it from the joint radar-communications received signal. After suppressing the radar return, the receiver then decodes and removes the communications signal from the radar return suppressed received waveform to obtain a radar return signal free of communications interference. This method of interference cancellation is called SIC. It is this receiver model that causes communications performance to be closely tied to the radar waveform spectral shape. It should be noted that since the predicted target location is never always accurate, the predicted radar signal suppression leaves behind a residual contribution, $n_{\text{resi}}(t)$. Consequently, the receiver will decode the communications message from the radar-suppressed joint received signal at a lower rate. The block diagram of the joint radar-communications system considered in this scenario is shown in Figure 1. When applying SIC, the interference residual plus noise signal $n_{\text{int+n}}(t)$, from the communications receiver's perspective, is given by [3], [12]

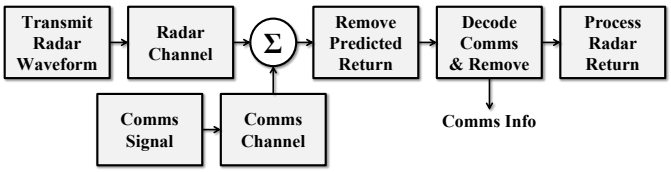


Fig. 1: Joint radar-communications system block diagram for SIC scenario. The radar and communications signals have two effective channels, but arrive converged at the joint receiver. The radar signal is predicted and removed, allowing a reduced rate communications user to operate. Assuming near perfect decoding of the communications user, the ideal signal can be reconstructed and subtracted from the original waveform, allowing for unimpeded radar access.

$$n_{\text{int+n}}(t) = n(t) + n_{\text{resi}}(t) = n(t) + \sqrt{\|a\|^2 P_{\text{rad}}}\, n_{\text{proc}}(t)\, \frac{\partial x(t-\tau)}{\partial t}, \quad (1)$$

and

$$\|n_{\text{int+n}}(t)\|^2 = \sigma^2_{\text{noise}} + a^2\, P_{\text{rad}}\, (2\pi\, B_{\text{rms}})^2\, \sigma^2_{\text{proc}}\,, \quad (2)$$

where $n_{\text{proc}}(t)$ is the process noise with variance σ^2_{proc}.

B. Radar Estimation Rate

To measure spectral efficiency for radar performance, we developed a new metric recently called *radar estimation rate*, which is formally defined as the minimum average data rate required to provide time-dependent estimates of system or target parameters, for example, target range [3], [12], [13]. The radar estimation rate is expressed as follows:

$$R_{\text{est}} = I(\mathbf{x};\mathbf{y})/T_{\text{pri}}, \quad (3)$$

where $I(\mathbf{x};\mathbf{y})$ is the mutual information between random vectors $\mathbf{x}$ and $\mathbf{y}$, and T_{pri} =T_{pulse}/δ is the pulse repetition interval of the radar system, T_{pulse} is the radar pulse duration, and δ is the radar duty factor. This rate allows construction of joint radar-communications performance bounds, and allows future system designers to score and optimize systems relative to a joint information metric. For a simple range estimation problem with a Gaussian tracking prior, this takes the form [2], [3], [14]:

$$R_{\text{est}} = (1/2T)\log_2(1 + \sigma^2_{\text{proc}}/\sigma^2_{\text{CRLB}}), \quad (4)$$

where σ^2_{proc} is the range-state process noise variance and σ^2_{CRLB} is the Cramér-Rao lower bound (CRLB) for range estimation given by [3], [12], [13] $\sigma^2_{\text{CRLB}} = \sigma^2_{\text{noise}}/8\pi^2\, B^2_{\text{rms}}\, T_p\, B\, P_{\text{rad,rx}}$, where σ^2_{noise} is the noise variance or power, T_p is the radar pulse duration, B_{rms} is the radar waveform root mean square (RMS) bandwidth, and $P_{\text{rad,rx}}$ is the radar receive power, which is inversely proportional to the distance of the target from the joint node. Immediately apparent is the similarity of above

equation to Shannon's channel capacity equation [3], [12], [13], where the ratio of the source uncertainty variance to the range estimation noise variance forms a pseudo-signal-to-noise ratio (SNR) term. In Eq. 4, the estimation rate is inversely proportional to the distance of the target from the joint node. As discussed later, we design the waveform parameters over the planning horizon while accounting for the varying estimation rate due to target's motion.

III. Technical Approach

We measure the performance of the system with two metrics: communications information rate bound and radar estimation rate bound (discussed in the previous section). The joint radar-communications performance bounds developed in [3], [12], [13] considered only local radar estimation error, therefore making simplified assumptions about the radar waveform. In [4], the results were generalized to include formulation of an optimal radar waveform for both global radar estimation rate performance and consideration of in-band communications users forced to mitigate radar returns. To demonstrate a point solution of joint radar-communications information inner bounds, we recently developed the notion of SIC [3], [15], [16].

The key to joint radar-communications is SIC, which is to predict and subtract the radar target return, where the prediction variance would therefore drive an additional residual noise term for the in-band communications user, which reduces the communications rate from the normal interference-free bound. The communications signal is then decoded and reconstructed (reapplication of forward error correction), and subtracted from the original return. The radar user can then operate unimpeded. As a result, the radar estimation rate is the same as given in (3). Radar users would like to increase the RMS bandwidth to the point where the range estimation error is minimized, but not at the expense of significant global error. The communications user, however, suffers from the additional residual noise source [3]:

$$\tilde{R}_{\text{com}} \leq B \log_2 \left[1 + \frac{b^2 P_{\text{com}}}{\sigma^2_{\text{noise}} + a^2 P_{\text{rad}} \left(2\pi B_{\text{rms}}\right)^2 \sigma^2_{\text{proc}}} \right]. \quad (5)$$

A. POMDP Formulation of Joint Waveform Design problem

We consider a particular case study, with a radar target, communications user, and the joint node, as shown in Figure 2. The line-of-sight between the radar target and the joint node may be lost as the target moves around an obstacle (e.g., urban structure). We will develop our POMDP framework for this case study, which can be easily generalized and extended to other problem scenarios. This particular case study allows us to show qualitative and quantitative benefits of POMDP in adaptive waveform design.

The key components in the waveform design algorithm based on POMDP are shown in Figure 3. The POMDP planner evaluates the belief-state (posterior distribution over the state space updated according to Bayes' rule) of the system, uses an ADP method to solve the POMDP approximately,

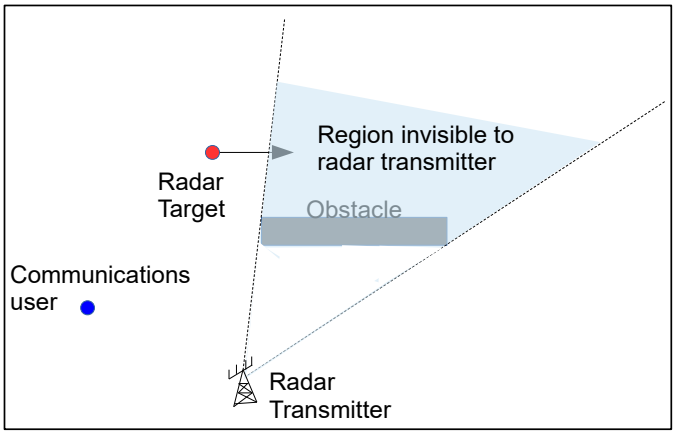


Fig. 2: Problem Scenario

and produces optimal or near-optimal decisions on waveform parameters; details are discussed later. Our objective is to design the shape of the waveforms over time to maximize the system performance. Here, we choose a weighted average of the estimation rate and the communications rate as the performance metric. First, we begin with a unimodular chirp waveform $\exp[j(\pi B/T)(t^2)]$. We control the spectral shape of this chirp signal to maximize joint performance. To achieve this, we first sample the chirp signal, and collect N samples in the frequency domain. Let $X = (X(f_1), \ldots, X(f_N))^T$ be the discretized signal in the frequency domain at frequencies $f_1, \ldots, f_N$. Let $u = (u_1, \ldots, u_N)^T$ be an array of spectral weights we will optimize as discussed below, where $u_i \in [0, 1], \forall i$. We control the spectral shape of the chirp signal by multiplying (i.e., dot product) the signal with the spectral weights in the frequency domain, i.e., the resulting signal is given by $X(f_i)u_i, \forall i$. To pose any decision making problem as a POMDP, we need to define the POMDP ingredients, which are states, actions, state-transition law, observations & observation law, and reward function, in the context of the particular problem at hand. The following is a description of the POMDP ingredients as defined specific to our waveform design problem. Hereafter, we model the system dynamics as a discrete event process, where k represents the discrete time index.

States: State at time k is defined as $x_k = (\chi_k, \xi_k, P_k)$, where χ_k represents the target state, which includes the location, velocity, and the acceleration of the target; and (ξ_k, P_k) represents the state of the tracking algorithm, e.g., Kalman filter, where ξ_k is the mean vector, and P_k is the covariance matrix.

Actions: Actions are the waveform spectral weights vector u_k as defined above.

State-Transition Law: χ_k evolves according to a motion model called *near-constant velocity model* captured by $\chi_{k+1} = F\chi_k + n_k$, where F is a transition matrix, and $n_k = n_{\text{proc}}(t = k)$ is the process noise described in Section II-A, which is modeled as a Gaussian process. ξ_k and P_k evolve according to Kalman filter equations.

Observation Law: $z_k^{\text{Targ}} = G\chi_k + w_k$ (if not occluded)

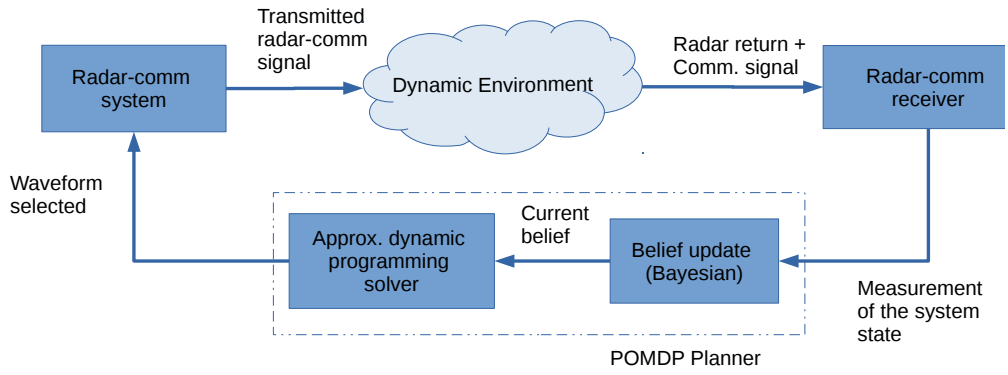


Fig. 3: Adaptive waveform optimization in a dynamic environment.

and $z_k^{\text{Targ}} = w_k$ (if occluded), where G is a transition matrix, and w_k is the measurement noise, modeled as a Gaussian process. Specifically, $w_k \sim \mathcal{N}(0, R_k)$, where R_k is the noise covariance matrix, where the entries in the matrix scale (increase) with the distance between the joint node (or sensor node) and the target. We assume the other state variables to be fully known.

Reward Function: The reward function rewards the decision u_k taken at time k given the state of the system is x_k as defined below:

$$R(x_k, u_k) = \alpha R_{\text{est}}(x_k, u_k) + (1-\alpha) R_{\text{comm}}(x_k, u_k),$$

where R_{est} is the radar estimation rate [4], R_{comm} is the communications data rate, and $\alpha \in [0, 1]$ is a weighting parameter.

Belief State: We maintain and update the posterior distribution over the state space (as the actual state is not fully observable), also known as the "belief state" given by $b_k = (b_k^{\chi}, b_k^{\xi}, b_k^{P})$, where $b_k^{\xi}(x) = \delta(x - \xi_k)$, $b_k^{P}(x) = \delta(x - P_k)$, and $b_k^{\chi} = \mathcal{N}(\xi_k, P_k)$. Here, we know the state of the tracking algorithm, so belief states corresponding to these states are just delta functions, whereas the target state is modeled as a Gaussian distribution with ξ_k and P_k as the mean vector and the error covariance matrix respectively.

B. POMDP Solution

Our goal is to optimize the actions over a long time-horizon (of length H) to maximize the expected cumulative reward. The objective function (to be maximized) is given by $J_H = \mathrm{E}\left[\sum_{k=0}^{H-1} R(x_k, u_k)\right]$. But, we can also write J_H in terms of the belief states as

$$J_H = \mathrm{E}\left[\sum_{k=0}^{H-1} r(b_k, u_k) \,\middle|\, b_0\right],$$

where, $r(b_k, u_k) = \int R(x, u_k) b_k(x)\, \mathrm{d}x$ and b_0 is the initial belief state. Let $J_H^*(b)$ represent the optimal objective function value, given the initial belief-state b. Therefore, the optimal action policy at time k is given by $\pi^*(b_k) = \arg\max_u Q(b_k, u)$, where $Q(b_k, u) = r(b_k, u) + \mathrm{E}\left[J_H^*(b_{k+1}) \,|\, b_k, u\right]$ which is also called the Q-value. [7], [8] give a detailed description of POMDP and its solution. POMDP formulations are known for their high computational complexity, particularly because it is near impossible to obtain the above-discussed Q-value in real-time [8]. There exist a plethora of approximation methods called *approximate dynamic programming* (ADP) methods that approximate the Q-value [7]. We adopt a fast ADP approach called *nominal belief-state optimization* (NBO), which we previously developed in the context of another adaptive sensing problem [8]. With NBO approximation, the POMDP formulation leads to the following optimization problem:

$$\min_{u_k, k=0,\ldots,H-1} \sum_{k=0}^{H-1} r(\tilde{b}_k, u_k), \tag{6}$$

where $\tilde{b}_k, k = 0, \ldots, H-1$ is a sequence of readily available "nominal" belief states, as opposed to b_ks which are random variables, obtained from the NBO approach.

IV. Simulation Results and Discussion

We implement the POMDP and the NBO approaches in MATLAB to solve the waveform design problem in the above described scenario. We study the methods in a scenario with two obstacles blocking the line-of-sight (LOS) between the joint node and the target as the target moves from the left to the right as shown in Figure 2. We use MATLAB's *fmincon* [17] to solve the optimization problem in Eq. 6. Additionally, we implement the receding horizon control approach while optimizing the decision variables over the moving planning horizon. The parameters used in this study are shown in Table II. The following are the main objectives of this numerical study.

- Study the impact of the planning horizon H on the joint performance with respect to the estimation and the communications rates.
- Quantitative comparison of the myopic approach ($H = 1$) and the non-myopic approach ($H > 1$).

A. Impact of Blending Parameter on the Rates

We plot the estimation rate and communications rate of the optimized waveform against $\alpha \in [0, 1]$ as shown in Figure 4. As expected, α allows us to trade-off between the two rates.

TABLE II: Parameters for Waveform Design Methods

Parameter	Value
Bandwidth (B)	5 MHz
Center frequency	3 GHz
Effective temperature (T_{temp})	1000 K
Communications range	10 km
Communications power (P_{com})	1 W
Communications antenna Gain	20 dBi
Communications receiver Side-lobe Gain	10 dBi
Radar antenna gain	30 dBi
Target cross section	10 m^2
Target process standard deviation (σ_{proc})	100 m
Time–bandwidth product (TB)	128
Radar duty factor (δ)	0.01

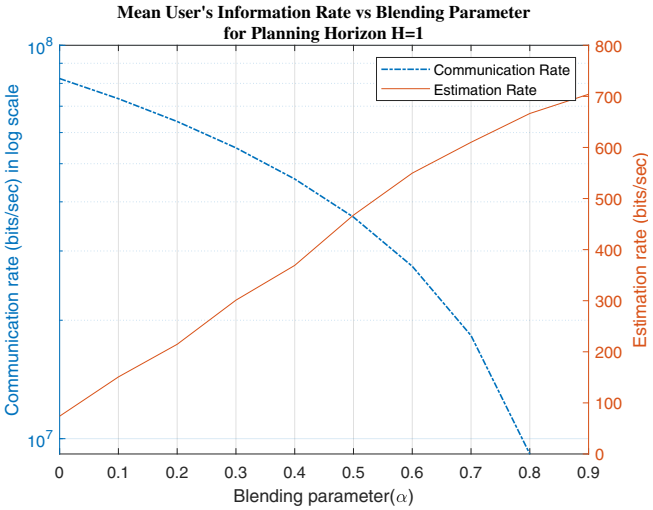


Fig. 4: Average rate vs. α

This trade-off property of the system is the reason we need to optimize the waveform parameters over a planning horizon as opposed to one-step optimization. We show a rate-rate curve showing the communications and estimation rate for different values of α where R_{comm} is the SIC communications data rate defined in Section III-A and α is a blending parameter that is varied from 0 to 1. When $\alpha = 0$, in Eq. 4 only communications rate is considered, and when $\alpha = 1$, only the radar estimation rate is considered. In between, the product is jointly maximized.

B. Impact of Planning Horizon on the Rates

In Figure 5, we plot the estimation rate for $H = 1$ and $H = 5$. At around time index 40, the line of sight is lost, which leads to reduction in the estimation rate. As the line-of-sight gets established at the time index 60, the rates go up in both cases, but the rise is significantly higher for $H = 5$, which shows that our non-myopic approach plans the waveform parameters more effectively than the myopic approach ($H = 1$). Table III summarizes the average combined rates for different planning horizon lengths as discussed above. As we increase $H = 1$ to $H = 5$ the combined rate is increased by more than five times,

TABLE III: Planing horizon length (H) Versus average combined rate for $\alpha = 0.5$

Planing horizon length (H)	Average combined rate (x10^6 bits/sec)
1	22.8
2	45.72
3	68.58
4	91.43
5	114.29

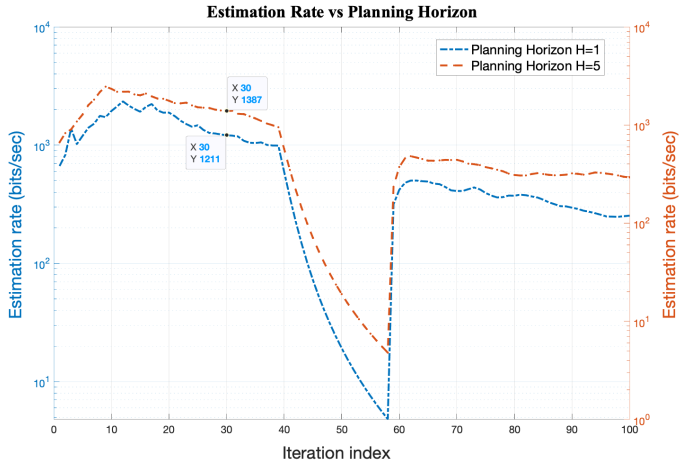


Fig. 5: Estimation rate vs. planning horizon

but at the same time the computational complexity in solving Eq. 6 with $H = 5$ is significantly higher than with $H = 1$. In fact, this computational complexity grows exponentially with H. Thus, one may need to assess if it is worth trading off computational complexity for better performance, and then determine the planning horizon length H accordingly. Figure 6 shows the quantitative comparison of radar estimation rate and average combined rate for five different planning horizon. Figure 7 shows the qualitative comparison of planning horizon $H = 1$ vs. $H = 5$. In both cases the size of the error confidence ellipse of the target increases when the target is occluded by the obstacles. But the size of the ellipse visibly reduces as we set $H = 5$. This reduction in the ellipse size is captured quantitatively in Figure 7. Table IV shows the

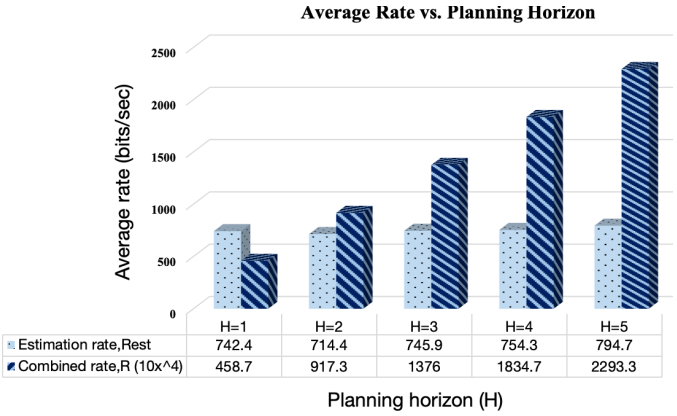
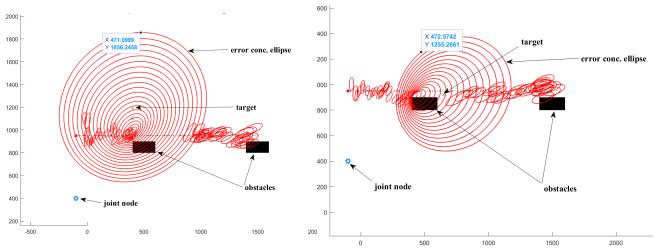


Fig. 6: Average rate vs. planning horizon



(a) Planning Horizon H=1 **(b)** Planning Horizon H=5

Fig. 7: Myopic vs. non-myopic approach

TABLE IV: Planing horizon length (H) Versus Average target location error

Planing horizon length (H)	Average target location error (m)
1	107.4344
2	102.7342
3	94.9062
4	73.7049

impact of plan horizon length on the average target location error.

V. Conclusions

We developed a decision theoretic framework for adaptive waveform design in joint radar-communications systems. Specifically, we posed the waveform design problem as a *partially observable Markov decision process* (POMDP) and extended an *approximated dynamic programming approach* to solve the problem in near real-time. Particularly, we adapted an ADP approach called *nominal belief-state optimization* or NBO. The goal is to optimize the spectral shape of the radar waveform over time to maximize the joint performance of radar and communications in spectral coexistence. We presented the quantitative benefits, in terms of communications and radar estimation rates, of our POMDP-based non-myopic approach in waveform design against myopic or greedy approaches. In our future studies, we will address challenges including time-varying communication demand and target detection probability.

References Cited

[1] J. B. Evans, "Shared Spectrum Access for Radar and Communications (SSPARC),*Online:http://www.darpa.mil/program/shared-spectrum-access-for-radar-and-communications.*

[2] D. W. Bliss and S. Govindasamy, *Adaptive Wireless Communications: MIMO Channels and Networks*. New York, New York: Cambridge University Press, 2013.

[3] D. W. Bliss, "Cooperative radar and communications signaling: The estimation and information theory odd couple," in *Proc. IEEE Radar Conference*, May 2014, pp. 50–55.

[4] B. Paul, A. R. Chiriyath, and D. W. Bliss, "Joint communications and radar performance bounds under continuous waveform optimization: The waveform awakens," in *IEEE Radar Conference*, May 2016, pp. 865–870.

[5] P. Chavali and A. Nehorai, "Cognitive radar for target tracking in multipath scenarios," in *Proc. Waveform Diversity & Design Conf.*, Niagara Falls, Canada, 2010.

[6] Ma, O., Chiriyath, A. R., Herschfelt, A., & Bliss, D. (2019). Cooperative Radar and Communications Coexistence Using Reinforcement Learning. In M. B. Matthews (Ed.), Conference Record of the 52nd Asilomar Conference on Signals, Systems and Computers, ACSSC 2018 (pp. 947-951). [8645080] (Conference Record - Asilomar Conference on Signals, Systems and Computers; Vol. 2018-October). IEEE Computer Society. https://doi.org/10.1109/ACSSC.2018.8645080

[7] E. K. P. Chong, C. Kreucher, and A. O. Hero, "Partially observable Markov decision process approximations for adaptive sensing,"*Disc. Event Dyn. Sys.*, vol. 19, pp. 377–422, 2009.

[8] S. Ragi and E. K. P. Chong, "UAV path planning in a dynamic environment via partially observable Markov decision process," *IEEE Trans. Aerosp. Electron. Syst.*,vol. 49, pp. 2397–2412, 2013.

[9] A. Charlish and F. Hoffmann, "Anticipation in cognitive radar using stochastic control," in *Proc. IEEE Radar Conf.*, Arlington, VA, 2016, pp. 1692–1697.

[10] S. Ragi, E. K. P. Chong, and H. D. Mittelmann, "Mixed-integer nonlinear programming formulation of a UAV path optimization problem," in *Proc. 2017 American Control Conf.*, Seattle, WA, 2017, pp. 406–411.

[11] T. M. Cover and J. A. Thomas, *Elements of Information Theory*,2nd ed. Hoboken, New Jersey: John Wiley & Sons, 2006.

[12] A. R. Chiriyath, B. Paul, G. M. Jacyna, and D. W. Bliss, "Inner bounds on performance of radar and communications co-existence,"*IEEE Transactions on Signal Processing*, vol. 64, no. 2, pp. 464–474, January 2016.

[13] B. Paul and D. W. Bliss, "Extending joint radar-communications bounds for FMCW radar with Doppler estimation," in *IEEE Radar Conference*, May 2015, pp. 89–94.

[14] B. Paul and D. W. Bliss, "The constant information radar," *Entropy*,vol. 18, no. 9, p. 338, 2016. [Online]. Available: http://www.mdpi.com/1099-4300/18/9/338

[15] A. Chiriyath, S. Ragi, H. D. Mittelmann, D. W. Bliss, "Novel Radar Waveform Optimization for a Cooperative Radar-Communications System,"*IEEE Transactions on Aerospace and Electronic Systems* , vol. 55, no. 3, pp. 1160–1173, April 2019.

[16] A. Chiriyath, S. Ragi, H. D. Mittelmann, D. W. Bliss, "Radar Waveform Optimization for Joint Radar Communications Performance," *Electronics*, special issue on *Cooperative Communications for Future Wireless Systems*, vol. 8, no. 12, December 2019.

[17] MATLAB's fmincon. 2016. [Online]. Available: https://www.mathworks.com/help/optim/ug/fmincon.

[18] B. Paul, A. R. Chiriyath, and D. W. Bliss. Survey of rf communications and sensing convergence research. *IEEE Access*, 5:252–270, 2017.

[19] S. Ragi and E. K. P. Chong, "Decentralized guidance control of UAVs with explicit optimization of communication," *J. Intell. Robot. Syst.*, vol. 73, pp. 811–822, 2014.

[20] A. R. Chiriyath and D. W. Bliss, "Joint radar-communications performance bounds: Data versus estimation information rates," in *2015 IEEE Military Communications Conference, MILCOM*, October 2015, pp. 1491–1496.

[21] A. R. Chiriyath and D. W. Bliss, "Effect of clutter on joint radar-communications system performance inner bounds," in *2015 49th Asilomar Conference on Signals, Systems and Computers*, November 2015, pp. 1379–1383.

[22] B. Paul, D. W. Bliss, and A. Papandreou-Suppappola, "Radar tracking waveform design in continuous space and optimization selection using differential evolution," in *2014 48th Asilomar Conf. Signals, Systems and Computers*, November 2014, pp. 2032–2036.

[23] J. R. Guerci, R. M. Guerci, A. Lackpour, and D. Moskowitz, "Joint design and operation of shared spectrum access for radar and communications," in*IEEE Radar Conference*, May 2015, pp. 761–766.

[24] M. Bica, K.-W. Huang, V. Koivunen, and U. Mitra, "Mutual information based radar waveform design for joint radar and cellular communication systems," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2016, pp. 3671–3675.

[25] M. R. Bell, N. Devroye, D. Erricolo, T. Koduri, S. Rao, and D. Tuninetti, "Results on spectrum sharing between a radar and a communications system," in *2014 International Conference on Electromagnetics in Advanced Applications (ICEAA)*, August 2014, pp. 826–829.

[26] R. M. Gutierrez, A. Herschfelt, H. Yu, H. Lee, and D. W. Bliss. Joint radar-communications system implementation using software defined radios: Feasibility and results. In *2017 51st Asilomar Conference on Signals, Systems, and Computers*, pages 1127–1132, Oct 2017.

[27] A. R. Chiriyath, B. Paul, and D. W. Bliss. Radar-communications convergence: Coexistence, cooperation, and co-design.*IEEE Transactions on Cognitive Communications and Networking*,3(1):1–12, March 2017.

[28] B. Paul, C. D. Chapman, A. R. Chiriyath, and D. W. Bliss. Bridging mixture model estimation and information bounds using i-mmse. *IEEE Transactions on Signal Processing*, 65(18):4821–4832,Sept. 2017.

Waveform codesign for radar-communications spectral coexistence via dynamic programming

Shammi A. Doly *, Alex Chiriyath*, Hans D. Mittelmann†, Daniel W. Bliss* and Shankarachary Ragi‡
* School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, AZ 85287
Email: sdoly@asu.edu, achiriya@asu.edu & d.w.bliss@asu.edu
† School of Mathematical and Statistical Sciences, Arizona State University, Tempe, AZ 85287
Email: mittelmann@asu.edu
‡ Department of Electrical Engineering, South Dakota School Mines & Technology, Rapid City, SD 57701
Email: shankarachary.ragi@sdsmt.edu

***Abstract*—We develop a new waveform codesign approach for radar-communications spectral coexistence using a decision-theoretic framework called *partially observable Markov decision process* (POMDP). The POMDP framework's natural look-ahead feature allows us to trade-off short-term for long-term performance, which is necessary in waveform codesign problems with competing objectives and dynamic user needs. As POMDPs are computationally intractable, we extend two approximation methods called *nominal belief-state optimization* and *random-sampling multipath hypothesis propagation* to make the codesign approaches tractable.**

I. Introduction

Spectral congestion is forcing legacy radar band users to investigate cooperation and co-design methods with a growing number of communications applications [1]. The codesign of radar and wireless communications systems faces several challenges: interference, radar, communications decoupling, and dynamic user (radar and communications) requirements. The studies in [2], [3] provide a detailed overview of the challenges and research directions in the "spectral" coexistence of radar and communications. In the study in [4], the quality of the radar return and the communications rate is mainly determined by the waveform's spectral shape. Moreover, one of the critical challenges for any waveform design method is to meet dynamic user needs. In this paper, we develop waveform shaping methods that are adaptive and can trade-off between competing performance objectives to address these challenges. A waveform design method can most effectively meet the dynamic user needs if it predicts the future user needs and allocates the resources accordingly. Previous research has considered waveform design for joint radar-communications systems, for example, [5], [6]. However, existing methods often do not meet dynamic performance requirements, as they tend to be *greedy* in that they only maximize short-term performance for immediate benefits. For problems with dynamic performance requirements, long-term performance is critical as decisions (to choose a particular waveform) at the current time epoch may lead to regret in the future. To address these challenges, we develop an adaptive waveform design method for joint radar-communications systems based on the theory of *partially observable Markov decision process* (POMDP) [7], [8]. Specifically, we formulate the waveform design problem as a POMDP [8], after which the design problem becomes a matter of solving an optimization problem. In essence, the POMDP solution provides us with the optimal decisions on the waveform design parameters [9]. The optimization problems resulting from POMDPs are hard to solve precisely; specifically, these problems are PSPACE-complete [10]. The optimization problems resulting from POMDP formulation are typically reformulated as *dynamic programming* problems, which allows us to apply *Bellman's principle of optimality*, leading to a plethora of approximation methods called *approximate dynamic programming* methods or ADP methods as surveyed in [7]. In this study, we adopt two different ADP approaches called nominal belief-state optimization (NBO) [7], and *random sampling multipath hypothesis propagation* (RS-MHP) [11], [12] to maximize the reward in the long horizon decision problems. RS-MHP methods are a variant of the existing broad class of Monte-Carlo tree search (MCTS) methods. The POMDP framework has a natural look-ahead feature, i.e., it can trade-off short-term for long-term performance. This feature lets the POMDP naturally anticipate the dynamic user needs and optimize the resources (waveforms) to actively meet the user's needs. Typically, one studies these adaptive methods under "cognitive radio (radar)," which has a rich literature. The current waveform design problem is related to a class of problems called *adaptive sensing*, where POMDP was already a proven effective framework [9], [13]. However, this paper brings formalism to these methods by posing the waveform design problem as a POMDP. Recently, POMDPs were used in [14] to develop adaptive methods for "cognitive radar," but in a different context, where the focus was on optimizing radar measurement times and not on waveform

The work of S. Doly, S. Ragi, and H. D. Mittelmann was supported in part by the Air Force Office of Scientific Research under grant FA9550-19-1-0070.

shaping.

A. Literature Review

Modern spectrum sharing techniques proposed waveform co-design and operation as a necessary construct for joint radar-communications [15], [16]. Various methods employ optimization theory to select a jointly optimal waveform [17]–[19] or jointly maximizing information criteria for radar and orthogonal frequency-division multiplexing (OFDM) communications users to minimize mutual interference for dynamic bandwidth allocation [20]. Other avenues for co-design have also been investigated [21]–[30]. Most modern co-design approaches do not take the long term needs of the system into consideration. The proposed POMDP-based waveform co-design framework is able to evaluate the needs of the system into the future and trade performance in the short-term versus the long-term.

Cognitive techniques in radar were primarily used for enhanced dynamic behavior in complex environments [31], [32], but researchers have begun to look at cognitive radar as a solution to the spectral scarcity problem via radar scheduling [33] or employing cognitive radio spectrum sensing techniques, emitter localization, and power allocation to avoid interference [34]–[39]. Others have investigated cognitive radar as a solution to the spectral congestion problem [40]–[43]. Most research efforts tend to adaptively use the spectrum to avoid interference. Such methods are akin to the traditional spectrum sharing solution of isolation in space, time and/or frequency, which can limit joint system performance as opposed to a co-design approach, where both systems cooperatively utilize the spectrum. Co-design approaches, such as our POMDP-based approach, show better joint system performance due to better cooperation between systems.

Relationships between radar estimation sidelobe ambiguity and communications channel coding were previously studied [44]. Others have suggested specific coding techniques with favorable properties such as finite Heisenberg-Weyl groups [45], Golay waveforms with Doppler resilient properties [46], and complementary sequences [47]. These approaches tend to prioritize the performance of one system over the other, and as such are sub-optimal in performance to most modern co-design approaches.

OFDM was investigated as a viable option in vehicle-to-vehicle applications [48]–[51], software-defined radio (SDR) architectures [52], etc. However, results show conflicting cyclic prefix requirements, data-dependent ambiguities, and trouble mitigating peak-to-average power ratio (PAPR) for typical radar power requirements. Researchers focused on developing joint systems that could mitigate the effects of these problems, such as suppressing side-lobes [53], maintaining a constant envelope [54], or reducing PAPR [55]. An OFDM approach is fundamentally more favorable to communications system performance and most research efforts lie in improving radar performance to an acceptable level. However, co-design approaches such as ours are more beneficial in the long-term due to them giving both systems equal importance.

TABLE I: Survey of Notation

Variable	Description
B	Total system bandwidth
B_{rms}	Root-mean-squared radar bandwidth
B_{com}	Communications-only subband
P_{rad}	Radar power
T_{temp}	Effective temperature
b	Communications propagation loss
P_{com}	Communications power
P_{rad}	Communications power
$x(t)$	Unit-variance transmitted radar signal
a	Combined antenna gain
N	Number of samples
σ^2_{CRLB}	Cramer-Rao lower bound
σ^2_{noise}	Thermal noise
σ^2_{proc}	Process noise variance
TB	Time-bandwidth product
δ	Radar duty factor
w	Measurement noise
ζ_k	Mean vector noise
τ	Time delay to m^{th} target
α	Weighting parameter
R_{comm}	Communications rate
R_{est}	Radar estimation rate
P_k	Error covariance matrix
T_{pri}	Pulse repetition interval
H	Planning horizon length

B. Key Contributions

Below are the key contributions of this study.

- We formulate the joint radar waveform codesign problem as a POMDP.
- We extend ADP methods NBO and RS-MHP to solve the waveform design problem posed as POMDP.
- We implement the POMDP-based waveform codesign algorithms in simulated environments and conduct a numerical study to quantify the impact of the planning horizon on the performance of our methods.

A preliminary version of the parts of this paper was published as [8]. This paper differs from the conference paper [8] in the following ways: 1) along with the previous numerical results in [8] we conduct an empirical study to assess the impact of the planning horizon H in POMDP on the radar and communications performance; 2) we extend a new ADP approach RS-MHP [11], [12] to solve the waveform codesign problem, and benchmark its performance against the NBO approach we previously used in [8].

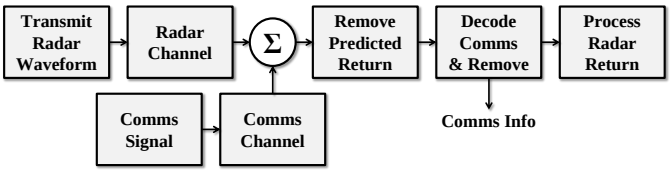


Fig. 1: Joint radar-communications system block diagram for SIC scenario. The radar and communications signals have two effective channels, but arrive converged at the joint receiver. The radar signal is predicted and removed, allowing a reduced rate communications user to operate. Assuming near perfect decoding of the communications user, the ideal signal can be reconstructed and subtracted from the original waveform, allowing for unimpeded radar access.

II. Joint-Radar Communications Premise

A. Successive Interference Cancellation Receiver Model

Table I shows the notations employed in this paper. In this study, we use an optimal multi-user receiver model called successive interference cancellation (SIC) [2], [57] to remove the communication signal from the radar return. Based on the prior observations of the radar target range (or time-delay) up to some random fluctuation (also called process noise) $n_{\text{proc}}(t)$ as a zero-mean random variable we generate the radar return. Then we subtract the predicted radar return from the joint radar-communications signal received. After suppressing the radar return, the receiver then decodes and removes the communications signal from the received signals. It is this receiver model that causes communications performance to be closely tied to the radar waveform spectral shape. The block diagram of the joint radar-communications system considered in this scenario is shown in Figure 1. When applying SIC, the interference residual plus noise signal $n_{\text{int+n}}(t)$, from the communications receiver's perspective, is given by [3], [58]

$$\begin{aligned} n_{\text{int+n}}(t) &= n(t) + n_{\text{resi}}(t) \\ &= n(t) + \sqrt{\|a\|^2 P_{\text{rad}}}\, n_{\text{proc}}(t)\, \frac{\partial x(t-\tau)}{\partial t}, \end{aligned} \quad (1)$$

and

$$\|n_{\text{int+n}}(t)\|^2 = \sigma^2_{\text{noise}} + a^2\, P_{\text{rad}}\, (2\pi\, B_{\text{rms}})^2\, \sigma^2_{\text{proc}}\,, \quad (2)$$

where $n_{\text{proc}}(t)$ is the process noise with variance σ^2_{proc}.

B. Radar Estimation Rate

To measure spectral efficiency for radar performance, we developed a new metric recently called *radar estimation rate*, which is formally defined as the minimum average data rate required to provide time-dependent estimates of system or target parameters, for example, target range [3], [58], [59]. The radar estimation rate is expressed as follows:

$$R_{\text{est}} = I(\mathbf{x};\mathbf{y})/T_{\text{pri}}, \quad (3)$$

where $I(\mathbf{x};\mathbf{y})$ is the mutual information between random vectors $\mathbf{x}$ and $\mathbf{y}$, and T_{pri} =T_{pulse}/δ is the pulse repetition interval of the radar system, T_{pulse} is the radar pulse duration, and δ is the radar duty factor. This rate allows construction of joint radar-communications performance bounds, and allows future system designers to score and optimize systems relative to a joint information metric. For a simple range estimation problem with a Gaussian tracking prior, this takes the form [2], [3], [60]:

$$R_{\text{est}} = (1/2T)\log_2(1 + \sigma^2_{\text{proc}}/\sigma^2_{\text{CRLB}}), \quad (4)$$

where σ^2_{proc} is the range-state process noise variance and σ^2_{CRLB} is the Cramér-Rao lower bound (CRLB) for range estimation given by [3], [58], [59]

$$\sigma^2_{\text{CRLB}} = \frac{\sigma^2_{\text{noise}}}{8\pi^2\, B^2_{\text{rms}}\, T_p\, B\, P_{\text{rad,rx}}} \quad (5)$$

where σ^2_{noise} is the noise variance or power, T_p is the radar pulse duration, B_{rms} is the radar waveform root mean square (RMS) bandwidth, and $P_{\text{rad,rx}}$ is the radar receive power, which is inversely proportional to the distance of the target from the joint node. Immediately apparent is the similarity of above equation to Shannon's channel capacity equation [3], [58], [59], where the ratio of the source uncertainty variance to the range estimation noise variance forms a pseudo-signal-to-noise ratio (SNR) term. In Eq. 4, the estimation rate is inversely proportional to the distance of the target from the joint node. As discussed later, we design the waveform parameters over the planning horizon while accounting for the varying estimation rate due to target's motion.

C. Inner rate bounds

We measure the performance of the system with two metrics: communications information rate bound and radar estimation rate bound (discussed in the previous section). The joint radar-communications performance bounds developed in [3], [58], [59] considered only local radar estimation error, therefore making simplified assumptions about the radar waveform. In [4], the results were generalized to include formulation of an optimal radar waveform for both global radar estimation rate performance and consideration of in-band communications users forced to mitigate radar returns. After the SIC process, some radar residual will be left in the communications signal (due to error in predicted target location and actual target location). If $R_{est} \approx 0$ is sufficiently low, then the communications operates according to the bound determined by the isolated communications system [2]. The highest possible communications rate when decoding the post-SIC received signal is given by

$$\tilde{R}_{\text{com}} \leq B \log_2 \left[1 + \frac{b^2 P_{\text{com}}}{\sigma^2_{\text{noise}}}\right]. \quad (6)$$

If $\tilde{R}_{com}$ is sufficiently low for a given transmit power then the communications signal can be decoded and subtracted

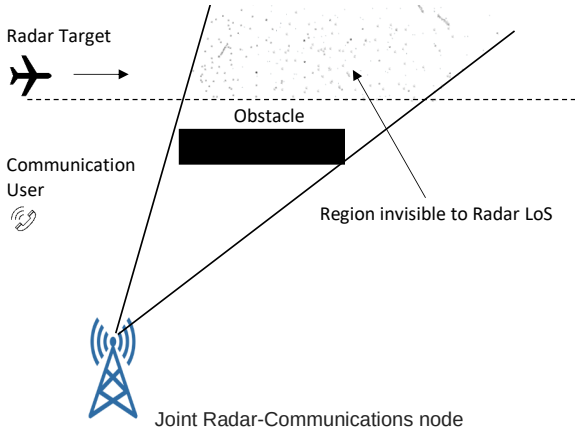


Fig. 2: Target tracking problem scenario

completely from the underlying signal, so that the radar parameters can be estimated without contamination,

$$\tilde{R}_{\text{com}} \leq B \log_2 \left[1 + \frac{b^2 P_{\text{com}}}{\sigma^2_{\text{noise}} + a^2 \, P_{\text{rad}} \, (2\pi \, B_{\text{rms}})^2 \, \sigma^2_{\text{proc}}} \right], \tag{7}$$

In this regime, the corresponding estimation rate bound R_{est} is given by Eq. 4. An achievable rate lies within the imaginary triangle constructed by the Eq. 4, Eq. 6, and Eq. 7.

III. Problem Specification

We consider a case study with a radar target, communications user, and the joint node, as shown in Figure 2. We consider a single clutter condition as shown in Figure 2 where an obstacle may occlude the line-of-sight of the target from the joint node. Total clutter residue acts as extra additive noise in the system, which causes the channel to appear more degraded. Radar estimation rates are also reduced (radar and communications overlap) once the clutter occludes the target. We do not consider any external interference or a jamming condition in this paper. We will develop our POMDP framework for this case study, which can be easily generalized and extended to other problem scenarios. This particular case study allows us to show the qualitative and quantitative benefits of POMDP in adaptive waveform design. The key components in the waveform design algorithm based on POMDP are shown in Figure 3. The POMDP planner evaluates the belief-state (posterior distribution over the state space updated according to Bayes' rule) of the system, uses an ADP method to solve the POMDP approximately, and produces optimal or near-optimal decisions on waveform parameters; details are discussed later. Our objective is to design the shape of the waveforms over time to maximize the system's performance. First, we begin with a unimodular chirp waveform $\exp[j(\pi B/T)(t^2)]$. We control the spectral shape of this chirp signal to maximize joint performance. We first sample the chirp signal and collect m samples in the frequency domain to achieve this. Let $X = (X(f_1), \ldots, X(f_m))^T$ be the discretized signal in the frequency domain at frequencies $f_1, \ldots, f_m$. Let $u = (u(1), \ldots, u(m))^T$ be an array of spectral weights we will optimize as discussed below, where $u(i) \in [0, 1], \forall i$. We control the chirp signal's spectral shape by multiplying (i.e., dot product) the signal with the spectral weights in the frequency domain, i.e., the resulting signal is given by $X(f_i)u(i), \forall i$.

IV. POMDP Formulation for Joint Waveform Codesign

To pose any decision making problem as a POMDP, we need to define the POMDP ingredients, namely states, actions, state-transition law, observations and observation law, and reward function, in the context of the particular problem at hand. Below is a description of the POMDP ingredients as defined specific to our waveform design problem. Hereafter, we model the system dynamics as a discrete event process, where k represents the discrete time index.

States: State at time k is defined as $x_k = (\chi_k, \xi_k, P_k)$, where χ_k represents the target state, which includes the location, velocity, and the acceleration of the target; and (ξ_k, P_k) represents the state of the tracking algorithm, e.g., Kalman filter, where ξ_k is the mean vector, and P_k is the covariance matrix.

Actions: Actions are the waveform spectral weights vector u_k, at time k, as defined previously.

State-Transition Law: χ_k evolves according to a target motion model *near-constant velocity model* [9] captured by $\chi_{k+1} = F\chi_k + n_k$, where F is a transition matrix, and $n_k = n_{\text{proc}}(t = k)$ is the process noise described in Section II-A, which is modeled as a Gaussian process. ξ_k and P_k evolve according to Kalman filter equations.

Observation Law: $z_k^{\text{Targ}} = G\chi_k + w_k$ (if not occluded) and $z_k^{\text{Targ}} = w_k$ (if occluded), where G is a transition matrix, and w_k is the measurement noise, modeled as a Gaussian process. Specifically, $w_k \sim \mathcal{N}(0, R_k)$, where R_k is the noise covariance matrix, where the entries in the matrix scale (increase) with the distance between the joint node (or sensor node) and the target. We assume the other state variables to be fully known.

Reward Function: The reward function rewards the decision u_k taken at time k given the state of the system is x_k as defined below:

$$R(x_k, u_k) = \alpha R_{\text{est}}(x_k, u_k) + (1 - \alpha) R_{\text{comm}}(x_k, u_k) \tag{8}$$

where R_{est} is the radar estimation rate [4], R_{comm} is the communications data rate, and $\alpha \in [0, 1]$ is a weighting parameter. The dependence of the rates on the waveform spectral weights u_k is explained as follows. Both the rates $R_{\text{est}}(x_k, u_k)$ and $R_{\text{comm}}(x_k, u_k)$ is a function of the RMS bandwidth B_{rms} of the waveform as can be seen from equations 4, 5, and 7. The RMS bandwidth clearly depends on the shape of the

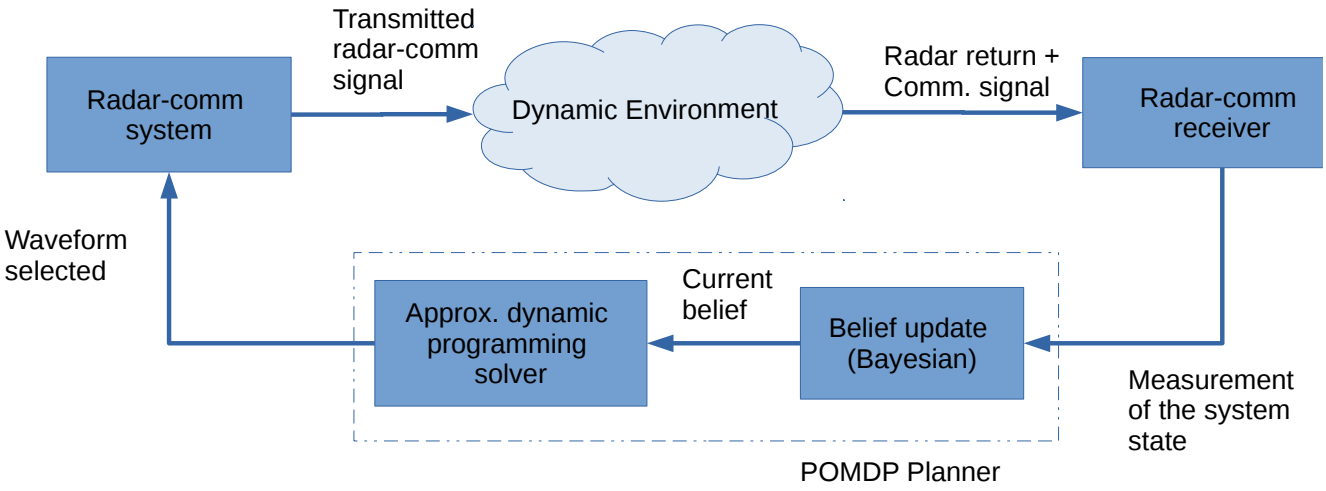


Fig. 3: Adaptive waveform optimization in a dynamic environment.

waveform spectrum, which is determined by the waveform spectral weights u_k.

Belief State: We maintain and update the posterior distribution over the state space (as the actual state is not fully observable), also known as the "belief state" given by $b_k = (b_k^{\chi}, b_k^{\xi}, b_k^{P})$, where $b_k^{\xi}(x) = \delta(x - \xi_k)$, $b_k^{P}(x) = \delta(x - P_k)$, and $b_k^{\chi} = \mathcal{N}(\xi_k, P_k)$. Here, we know the state of the tracking algorithm, so belief states corresponding to these states are just delta functions, whereas the target state is modeled as a Gaussian distribution with ξ_k and P_k as the mean vector and the error covariance matrix respectively. Our goal is to optimize the actions over a long time-horizon (of length H) to maximize the expected cumulative reward. The objective function (to be maximized) is given by $J_H = \mathrm{E}\left[\sum_{k=0}^{H-1} R(x_k, u_k)\right]$. But, we can also write J_H in terms of the belief states as

$$J_H = \mathrm{E}\left[\sum_{k=0}^{H-1} r(b_k, u_k) \,\middle|\, b_0\right], \tag{9}$$

where, $r(b_k, u_k) = \int R(x, u_k) b_k(x)\,\mathrm{d}x$ and b_0 is the initial belief state. Let $J_H^*(b)$ represent the optimal objective function value, given the initial belief-state b. Therefore, the optimal action policy at time k is given by $\pi^*(b_k) = \arg\max_u Q(b_k, u)$, where $Q(b_k, u) = r(b_k, u) + \mathrm{E}\left[J_H^*(b_{k+1}) \,|\, b_k, u\right]$ which is also called the Q-value. A detailed description of POMDP and its solution can be found in [7], [9]. POMDP formulations are notorious for their high computational complexity (PSPACE-complete [10]), particularly because it is near impossible to obtain the above-discussed Q-value in real-time [9]. Most ADP methods approximate the Q-value [7]. We adopt two ADP approaches: *nominal belief-state optimization* (NBO) [9] and *random sampling - multipath hypothesis propagation* (RS-MHP) [11], [12].

A. POMDP Solution via NBO

With NBO approximation, the POMDP formulation leads to the following optimization problem:

$$\max_{u_k, k=0,\ldots,H-1} \sum_{k=0}^{H-1} r(\tilde{b}_k, u_k), \tag{10}$$

where $\tilde{b}_k, k = 0, \ldots, H-1$ is a sequence of readily available "nominal" belief states, as opposed to b_ks which are random variables, obtained from the NBO approach. In NBO, the expectation is replaced by a sample state trajectory generated with an assumption that the future noise variables in the system collapse to the nominal or mean values (Figure 4), thus making the above objective function deterministic. The NBO method was developed to solve a UAV path optimization problem, which was posed as a partially observable Markov decision process (POMDP) [9]. POMDP generalizes the long horizon optimal control problem described in [11] in that the system state is assumed to be "partially" observable, which is inferred via the use of noisy observations and Bayes rules. Although the performance of the NBO approach was satisfactory in that it allowed to obtain reasonably optimal reward commands for the decision problem to be received,it ignored the uncertainty due to noise disturbances, thus leading to inaccurate evaluation of the objective function. This challenge can be overcome by the RS-MHP approach as discussed below.

B. POMDP solution via RS-MHP

The tree-like sampling of the states in the RS-MHP approach, as shown in figures 4 and 5, allows us to incorporate the uncertainty of the state evolution into the decision-making criteria, albeit with the increased computational burden compared to NBO. However, the sampling approach allows us to trade-off between the computational intensity and the

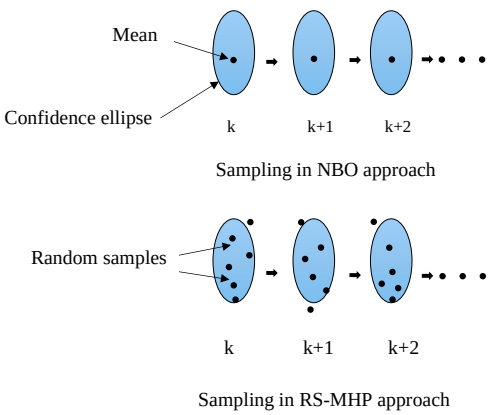


Fig. 4: Sampling in NBO vs. RS-MHP approach

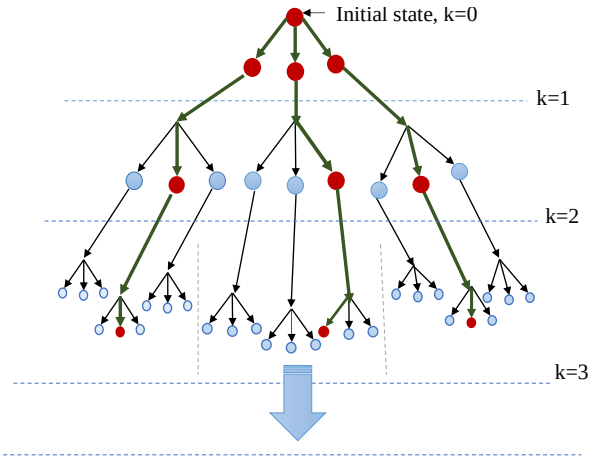


Fig. 5: Sampling in RS-MHP approach with pruning (3 nodes allowed to remain at each stage).

solution's optimality (determined by our choice of the number of samples/branches in RS-MHP). In RS-MHP approach, we sample the probability distribution of the state of the system (a random variable) N times at each time step and generate a sampling tree as shown in Figure 5 (here, $N = 3$). To avoid the exponential growth of the state sample nodes in this approach, at each time step we retain only M sample states and prune the remaining samples. If the number of the sample states at a given time instance is less than or equal to M, we do not perform pruning. Figure 5 shows an illustration of the above branch pruning strategy for a scenario with $N = 3$ and $M = 3$. We prune the tree branches based on their likeliness indices [11], [12], i.e., we retain the top M branches at each time step with the highest sample probabilities. We approximate the expectation with an average over the possible state trajectories or tree branches as follows:

$$\frac{1}{M}\sum_{i=1}^{M}\left(\sum_{k=0}^{H-1} r(x_k^i, u_k)\right) \tag{11}$$

where x_k^i represents the sample state node from the ith trajectory at time k. Clearly, as $N \to \infty$ and $M \to \infty$, the above approximation converges to the true objective function in Eq. (9).

V. Simulation and Results

We study the efficacy of the above-mentioned waveform codesign methods in a scenario with two obstacles blocking the line-of-sight (LOS) between the joint node and the radar target as the target moves from the left to the right, as shown in Figure 7. Furthermore, we implement the receding horizon control approach while optimizing the decision variables over the moving planning horizon [9]. We implement the NBO & RS-MHP approaches to solve the joint radar waveform optimization problem, in the above context, in MATLAB. We use MATLAB's *fmincon* [61] (an optimization tool in MATLAB) to solve the optimization problems discussed in the previous section. The following are the main objectives of this numerical study.

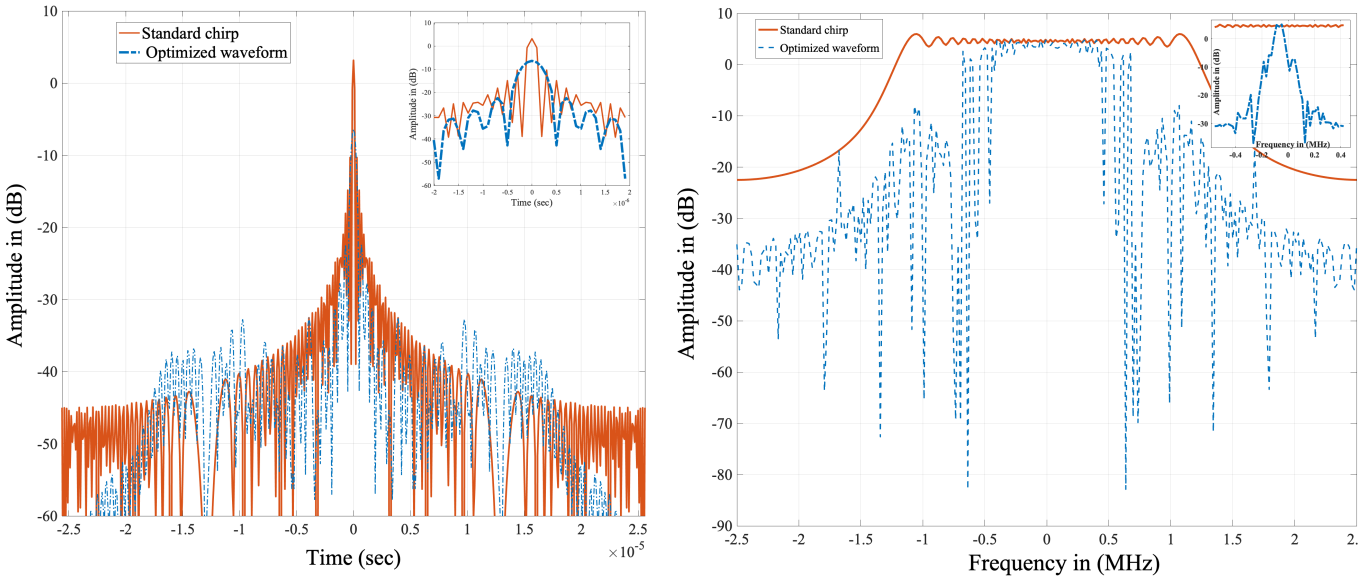
TABLE II: Parameters for Waveform Design Methods

Parameter	Value
Bandwidth (B)	5 MHz
Center frequency	3 GHz
Effective temperature (T_{temp})	1000 K
Communications range	10 km
Communications power (P_{com})	1 W
Communications receiver Side-lobe gain	20 dBi
Radar antenna gain	30 dBi
Target cross section	10 m^2
Target process standard deviation (σ_{proc})	100 m
Time-bandwidth product (TB)	128
Radar duty factor (δ)	0.01

- Study the optimal radar waveform properties.
- Study the impact of the planning horizon H on the joint performance with respect to the estimation and the communications rates.
- Performance comparison of NBO vs. RS-MHP ADP approaches in the non-myopic approach ($H > 1$).

A. Optimal radar waveform properties

We assume that the joint radar-communications receiver shares a single antenna front end and that the communications signal is received through an antenna sidelobe while the



(a) Radar waveform autocorrelation function of the optimized waveform with $\alpha = 0.5$ and $H = 1$

(b) Radar waveform spectrum with $\alpha = 0.5$ and $H = 1$. The standard chirp is depicted by the red line, and the optimal waveform spectrum is shown by the blue dotted line

Fig. 6: Optimized waveform vs. the standard chirp.

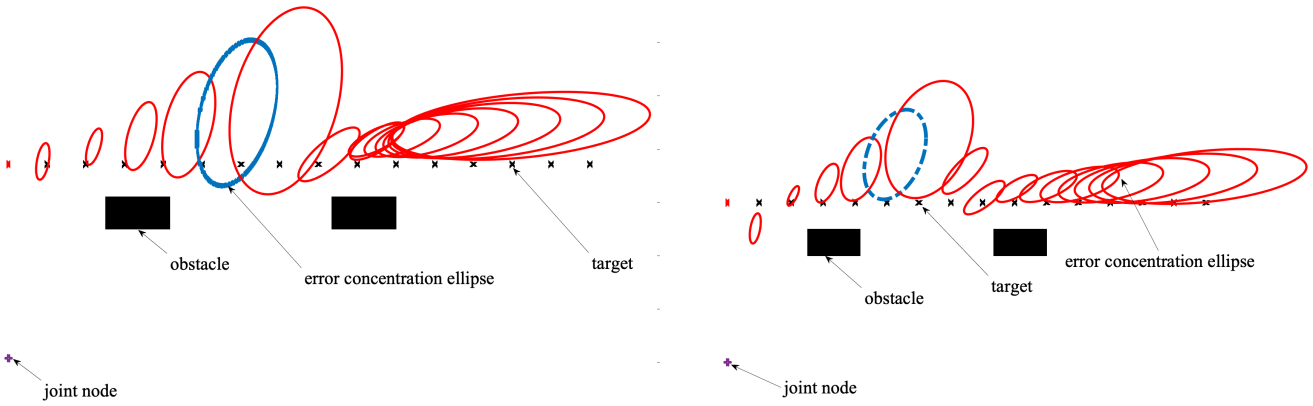
radar return is received through the same antenna mainlobe, so that the radar and communications receive gain are not identical. From the simulation results, it can be seen that the SNR varies from 19 dB to 23 dB roughly. The SNR in the NBO approach is 19.1419dB, and the RS-MHP approach is 22.4310dB. The parameters used in our simulation studies are shown in Table II. In Figure 6 (a) we show the radar waveform spectral autocorrelation function of optimized waveform with blending parameter $\alpha = 0.5$ and planning horizon $H = 1$ at a time step $k = 1$. We plot the spectrum of the optimized waveform with $\alpha = 0.5$ along with the original unmasked chirp waveform as shown in Figures 6 (b). This waveform spectrum shows the joint radar-communications optimal and has more energy at the bandwidth center than the sidebands. Radar waveform spectrum with $\alpha = 0.1$ and $\alpha = 1$ along with the original unmasked chirp waveform shown in Figure 8.

B. Effect of planning horizon length on the joint performance

We implement the NBO approach for $H = 1$ and $H = 9$ as shown in Figure 7. In both cases, the size of the error confidence ellipse of the target increases when the target is occluded by the obstacles. The growth of the ellipse size visibly reduces for $H = 9$ compared to $H = 1$. So, the non-myopic method ($H > 1$) has a better capability in keeping the growth of the target-sate uncertainty small compared to a myopic approach ($H = 1$). Figure 9 shows the estimation and the communications rates as a function of the blending parameter α. As expected, α allows us to smoothly trade-off between the two rates. Furthermore, in Figure 10, we plot the estimation rate as a function of time for the above two scenarios with $H = 1$ and $H = 9$, which shows the quantitative benefit of a non-myopic approach ($H > 1$) over a myopic approach ($H = 1$) in terms of the radar estimation rate. Figure 11 shows a gradual increase in the joint radar-communications performance with increasing H as expected in a non-myopic approach, however, the computational complexity in solving Eq. 9 grows exponentially with H.

C. Performance comparison of NBO vs. RS-MHP ADP approaches

Here we implement the RS-MHP approach for waveform codesign in the same simulation scenario described earlier. Figure 12 shows the cumulative distribution of the radar estimation rates using RS-MHP and NBO methods for $H = 3$. The figure clearly demonstrates that the RS-MHP approach outperforms the NBO approach and that the performance improves as we increase the number of samples N in the RS-MHP approach. Figure 13 shows the average radar estimation rates for N set to 10, 50, 100, 150, and 200 for $H = 3$. The figure shows a gradual increase in the algorithm's performance (in terms of the estimation rate) with increasing N as expected. This result also suggests that the pruning step in RS-MHP method would degrade the performance but can provide gains in terms of computational intensity. In summary, our numerical study confirms that the RS-MHP's performance has a clear statistical edge over that of the NBO approach in terms of the estimation rate.



(a) Planning Horizon H=1

(b) Planning Horizon H=9

Fig. 7: Error concentration ellipse (95% confidence) of the dynamic target at different location in both myopic ($H = 1$) and non-myopic ($H > 1$) approaches for α = 0.5 by red lines. The number of iteration indexes is considered $k = 15$ to demonstrate which locations match which ellipses more precisely. For example, the solid blue line shows the error concentration ellipse at the time index $k = 5$ for $H = 1$, and the error concentration ellipse for $H = 9$ at the time index $k = 5$ is shown by the blue dotted line. We see that with the non-myopic method ($H > 1$), we could minimize the size of the error concentration ellipse as the target tracking error as determined by the spectral mask we chose.

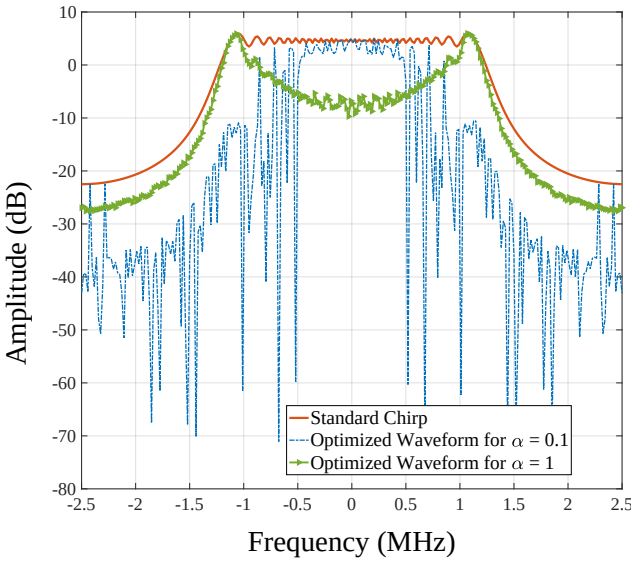


Fig. 8: The original unmasked chirp is depicted by the solid red line. The optimized waveform is depicted for $\alpha = 0.1$ by the blue dotted line. This waveform spectrum is communication-optimal and has more energy in the edges of the bandwidth. The optimized waveform is depicted by the green dashed line for $\alpha = 1$ and, this waveform spectrum is radar-optimal and has more energy in the center of the bandwidth.

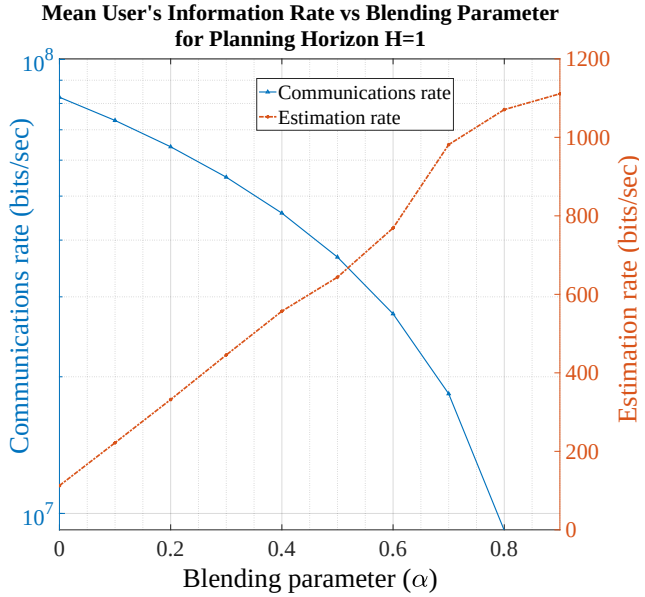


Fig. 9: Rate–rate curve depicting communications and estimation rate vs. α. Communications and estimation rate pairs are shown $\alpha \in [0, 1]$.

VI. CONCLUSIONS

We developed a waveform codesign approach for joint-radar communications systems using a decision-theoretic framework called *partially observable Markov decision processes* (POMDPs). The goal is to optimize the spectral shape of the radar waveform over time to maximize the joint performance of radar and communications in spectral coexistence measured in terms of radar estimation and communications rates. As most decision-theoretic formulations suffer from the *curse of dimensionality*, we extended two approximation strategies or *approximate dynamic programming* (ADP) methods to solve the POMDP - *nominal belief-state optimization* (NBO) and *random sampling multipath hypothesis propagation* (RS-MHP). Our numerical study confirmed that the POMDP-based non-myopic waveform codesign approach has a better

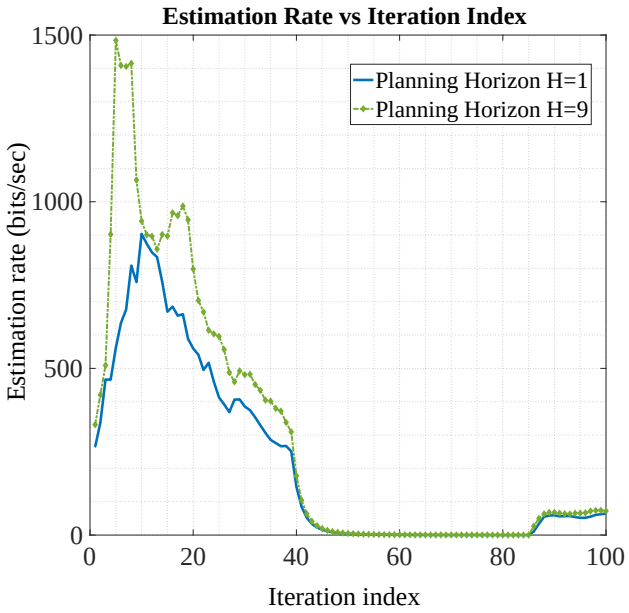
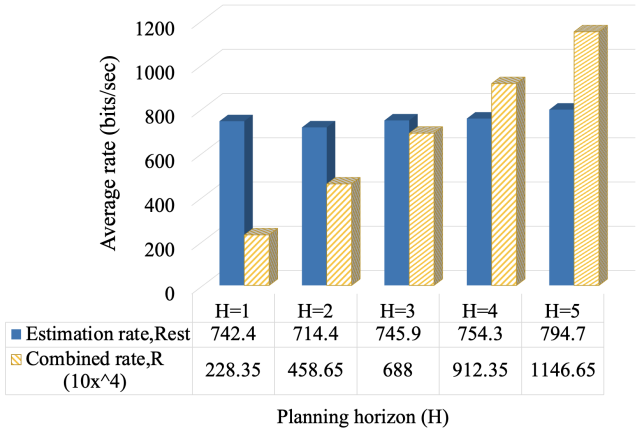


Fig. 10: Estimation rate vs. iteration index for both myopic and non-myopic approaches.



	H=1	H=2	H=3	H=4	H=5
Estimation rate,Rest	742.4	714.4	745.9	754.3	794.7
Combined rate,R (10x^4)	228.35	458.65	688	912.35	1146.65

Fig. 11: Average estimation and communication rates vs. planning horizon $H \in \{1, 2, 3, 4, 5\}$.

capability in keeping the growth of target state uncertainty small compared to a myopic approach. We also presented the quantitative benefits, in terms of the communications and the radar estimation rates, of our POMDP-based non-myopic approach against the traditional myopic approaches. Our results also confirmed a gradual increase in the joint radar-communications performance with increasing planning horizon length, which was expected in a non-myopic approach. Our numerical studies also confirmed that the ADP approach RS-MHP outperformed the NBO approach in terms of the target estimation rate.

References Cited

[1] J. B. Evans, "Shared Spectrum Access for Radar and Communications (SSPARC)," *[Online]*. http://www.darpa.mil/program/shared-spectrum-access-for-radar-and-communications.

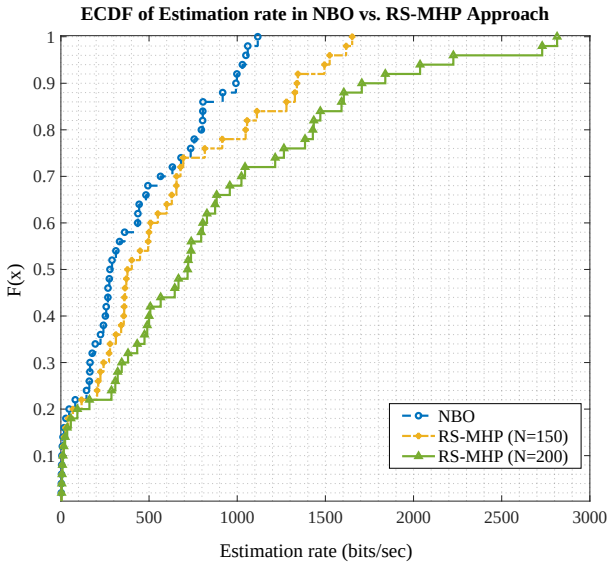


Fig. 12: Cumulative distribution of estimation rate for NBO vs. RS-MHP approaches. Here N represents the number of samples.

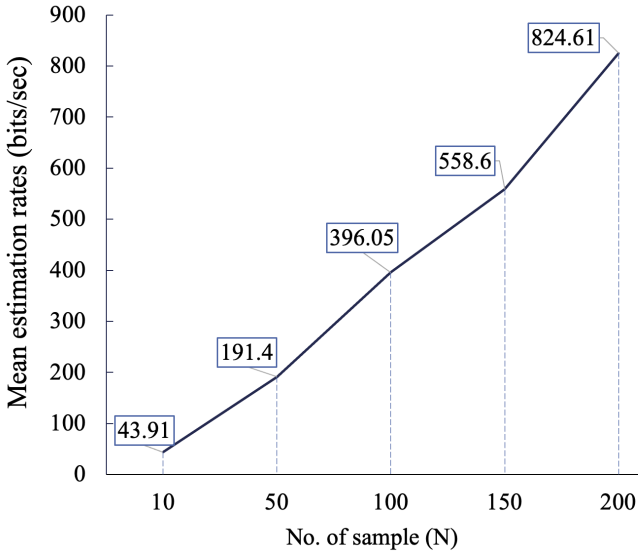


Fig. 13: Mean estimation rate vs. number of samples $N \in \{10, 50, 100, 150, 200\}$.

[2] D. W. Bliss and S. Govindasamy, *Adaptive Wireless Communications: MIMO Channels and Networks*. New York, New York: Cambridge University Press, 2013.

[3] D. W. Bliss, "Cooperative radar and communications signaling: The estimation and information theory odd couple," in *Proc. IEEE Radar Conference*, May 2014, pp. 50–55.

[4] B. Paul, A. R. Chiriyath, and D. W. Bliss, "Joint communications and radar performance bounds under continuous waveform optimization: The waveform awakens," in *IEEE Radar Conference*, May 2016, pp. 865–870.

[5] P. Chavali and A. Nehorai, "Cognitive radar for target tracking in multipath scenarios," in *Proc. Waveform Diversity & Design Conf.*, Niagara Falls, Canada, 2010.

[6] Ma, O., Chiriyath, A. R., Herschfelt, A., & Bliss, D. (2019). Cooperative Radar and Communications Coexistence Using Reinforcement Learning. In M. B. Matthews (Ed.), Conference Record of the 52nd Asilomar

Conference on Signals, Systems and Computers, ACSSC 2018 (pp. 947-951).

[7] E. K. P. Chong, C. Kreucher, and A. O. Hero, "Partially observable Markov decision process approximations for adaptive sensing,"*Disc. Event Dyn. Sys.*, vol. 19, pp. 377–422, 2009.

[8] S. A. Doly, A. Chiriyath, H. D. Mittelmann, D. W. Bliss, and S. Ragi, "A decision theoretic approach for waveform design in joint radar communications applications," in Proceedings of the 54th Asilomar Conference on Signals, Systems and Computers (Asilomar 2020), Pacific Grove, CA, Nov 01–04, 2020.

[9] S. Ragi and E. K. P. Chong, "UAV path planning in a dynamic environment via partially observable Markov decision process," *IEEE Trans. Aerosp. Electron. Syst.*,vol. 49, pp. 2397–2412, 2013.

[10] C. H. Papadimitriou and J. N. Tsitsiklis, The complexity of Markov decision processes,"Mathematics of Operations Research, vol. 12(3), pp. 441450, 1987."

[11] S. Ragi and H. D. Mittelmann, "Random-Sampling Multipath Hypothesis Propagation for Cost Approximation in Long-Horizon Optimal Control," in Proceedings of the 2020 IEEE Conference on Control Technology and Applications (CCTA), Montreal, Canada, August 24–26, 2020, pp. 14–18.

[12] S. Ragi and H. D. Mittelmann, "Random-Sampling Monte-Carlo Tree Search Methods for Cost Approximation in Long-Horizon Optimal Control," in IEEE Control Systems Letters, vol. 5, no. 5, pp. 1759-1764, Nov. 2021, doi: 10.1109/LCSYS.2020.3043991.

[13] S. Ragi, E. K. P. Chong, and H. D. Mittelmann, "Mixed-integer nonlinear programming formulation of a UAV path optimization problem," in *Proc. 2017 American Control Conf.*, Seattle, WA, 2017, pp. 406–411.

[14] A. Charlish and F. Hoffmann, "Anticipation in cognitive radar using stochastic control," in *Proc. IEEE Radar Conf.*, Arlington, VA, 2016, pp. 1692–1697.

[15] B. Paul, A. R. Chiriyath, and D. W. Bliss. Survey of rf communications and sensing convergence research. *IEEE Access*, 5:252–270, 2017.

[16] J. R. Guerci, R. M. Guerci, A. Lackpour, and D. Moskowitz, "Joint design and operation of shared spectrum access for radar and communications," in*IEEE Radar Conference*, May 2015, pp. 761–766.

[17] X. Wang, A. Hassanien, and M. G. Amin. Dual-function mimo radar communications system design via sparse array optimization. *IEEE Transactions on Aerospace and Electronic Systems*,55(3):1213–1226, June 2019.

[18] S. Zhou, X. Liang, Y. Yu, and H. Liu. Joint radar-communications couse waveform design using optimized phase perturbation.*IEEE Transactions on Aerospace and Electronic Systems*,55(3):1227–1240, June 2019.

[19] F. Wang, H. Li, and M. A. Govoni. Power allocation and co-design of multicarrier communication and radar systems for spectral coexistence. *IEEE Transactions on Signal Processing*, 67(14):3818–3831,2019.

[20] A. Turlapaty and Y. Jin, "A joint design of transmit waveforms for radar and communications systems in coexistence," in *IEEE Radar Conference*, May 2014, pp. 315–319.

[21] M. Zatman and M. Scharrenbroich, "Joint radar-communications resource management," in *2016 IEEE Radar Conference (RadarConf)*, May 2016, pp.1–6.

[22] Y. L. Sit, C. Sturm, and T. Zwick, "One-stage selective interference cancellation for the OFDM joint radar-communication system," in *The 7th German Microwave Conference*, March 2012, pp. 1–4.

[23] G. C. Tavik, C. L. Hilterbrick, J. B. Evins, J. J. Alter, J. Joseph G. Crnkovich, J. W. de Graaf, W. H. II, G. P. Hrin, S. A. Lessin, D. C. Wu, and S. M. Hagewood, "The advanced multifunction RF concept," *IEEE Transactions on Microwave Theory and Techniques*, vol. 53, no. 3, pp. 1009–1020, March 2005.

[24] L. Wang, J. McGeehan, C. Williams, and A. Doufexi, "Application of cooperative sensing in radar-communications coexistence," *IET Communications*,vol. 2, no. 6, pp. 856–868, July 2008.

[25] M. R. Bell, N. Devroye, D. Erricolo, T. Koduri, S. Rao, and D. Tuninetti, "Results on spectrum sharing between a radar and a communications system," in *2014 International Conference on Electromagnetics in Advanced Applications (ICEAA)*, August 2014, pp. 826–829.

[26] Marian Bica, Kuan-Wen Huang, Urbashi Mitra, and Visa Koivunen. Opportunistic radar waveform design in joint radar and cellular communication systems.In *IEEE Global Communications Conference (GLOBECOM)*, pages 1–7, December 2015.

[27] Alex Lackpour, Joseph R. Guerci, Alan Rosenwinkel, David Ryan, and Apurva N.Mody. Design and analysis of an information exchange-based radar/communications spectrum sharing system (RCS3). In *2016 IEEE Radar Conference (RadarConf)*, pages 1–6, May2016.

[28] C. D. Richmond, P. Basu, R. Learned, J. Vian, A. P. Worthen, and M. Lockard, "Performance bounds on cooperative radar and communication systems operation," in *2016 IEEE Radar Conference (RadarConf)*, May 2016, pp.1–6.

[29] M. P. Fitz, T. R. Halford, I. Hossain, and S. W. Enserink, "Towards simultaneous radar and spectral sensing," in *IEEE International Symposium on Dynamic Spectrum Access Networks (DYSPAN)*, April 2014, pp. 15–19.

[30] A. D. Harper, J. T. Reed, J. L. Odom, and A. D. Lanterman, "Performance of a joint radar-communication system in doubly-selective channels," in *49th Asilomar Conference on Signals, Systems and Computers*, November 2015, pp. 1369–1373.

[31] Simon Haykin. Cognitive radar: A way of the future. *IEEE Signal Processing Magazine*, 23(1):30–40, January 2006.

[32] J.R. Guerci. *Cognitive Radar: The Knowledge-aided Fully Adaptive Approach*. Artech House radar library. Artech House, 2010.

[33] Pawan Setlur and Natasha Devroye. Adaptive waveform scheduling in radar: an information theoretic approach. 8361, May 2012.

[34] Yogesh Nijsure, Yifan Chen, Chau Yuen, and Yong Huat Chew. Location-aware spectrum and power allocation in joint cognitive communication-radar networks. In *Sixth International ICST Conference on Cognitive Radio Oriented Wireless Networks and Communications (CROWNCOM)*, pages 171–175, June 2011.

[35] B. H. Kirk, R. M. Narayanan, K. A. Gallagher, A. F. Martone, and K. D. Sherbondy. Avoidance of time-varying radio frequency interference with software-defined cognitive radar. *IEEE Transactions on Aerospace and Electronic Systems*, 55(3):1090–1107, June 2019.

[36] Riccardo Palamà, Hugh Griffiths, and Francis Watson. Joint dynamic spectrum access and target-matched illumination for cognitive radar.*IET Radar, Sonar & Navigation*, 13:750–759(9), May 2019.

[37] Brandon Ravenscroft, Jonathan W. Owen, John Jakabosky, Shannon D. Blunt, Anthony F. Martone, and Kelly D. Sherbondy. Experimental demonstration and analysis of cognitive spectrum sensing and notching for radar. *IET Radar, Sonar & Navigation*, 12:1466–1475(9), December 2018.

[38] J. W. Owen, C. A. Mohr, B. H. Kirk, S. D. Blunt, A. F. Martone, and K. D. Sherbondy. Demonstration of real-time cognitive radar using spectrally-notched random fm waveforms. In *2020 IEEE International Radar Conference (RADAR)*, pages 123–128, 2020.

[39] B. Ravenscroft, J. W. Owen, B. H. Kirk, S. D. Blunt, A. F. Martone, K. D. Sherbondy, and R. M. Narayanan. Experimental assessment of joint range-doppler processing to addressclutter modulation from dynamic radar spectrum sharing. In *2020 IEEE International Radar Conference (RADAR)*, pages 448–453, 2020.

[40] A. Aubry, A. De Maio, M. Piezzo, M. M. Naghsh, M. Soltanalian, and P. Stoica. Cognitive radar waveform design for spectral coexistence in signal-dependent interference. In *IEEE Radar Conference*, pages 474–478, May 2014.

[41] Yogesh Nijsure, Yifan Chen, Said Boussakta, Chau Yuen, Yong Huat Chew, and Zhiguo Ding. Novel system architecture and waveform design for cognitive radar radio networks. *IEEE Transactions on Vehicular Technology*, 61(8):3630–3642, October 2012.

[42] Kuan-Wen Huang, Marian Bică, Urbashi Mitra, and Visa Koivunen. Radar waveform design in spectrum sharing environment: Coexistence and cognition.In *2015 IEEE Radar Conference*, pages 1698–1703, May 2015.

[43] A. F. Martone, K. A. Gallagher, and K. D. Sherbondy. Joint radar and communication system optimization for spectrum sharing.In *2019 IEEE Radar Conference*, pages 1–6, 2019.

[44] S. Howard, W. Moran, A. Calderbank, H. Schmitt, and C. Savage, "Channel parameters estimation for cognitive radar systems," in *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, March 2005, pp. 897–900.

[45] S. Howard, A. Calderbank, and W. Moran, "The finite Heisenberg-Weyl groups in radar and communications," vol. 2006, pp. 1–12, April 2006.

[46] A. Pezeshki, A. R. Calderbank, W. Moran, and S. D. Howard, "Doppler

resilient Golay complementary waveforms," *IEEE Transactions on Information Theory*, vol. 54, no. 9, pp. 4254–4266, September 2008.
[47] L. Xiaobai, Y. Ruijuan, Z. Zunquan, and C. Wei, "Research of constructing method of complete complementary sequence in integrated radar and communication," in *IEEE 11th International Conference on Signal Processing (ICSP)*, vol. 3, October 2012, pp. 1729–1732.
[48] M. Braun, C. Sturm, A. Niethammer, and F. K. Jondral, "Parametrization of joint OFDM-based radar and communication systems for vehicular applications," in *IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications*, September 2009, pp. 3020–3024.
[49] C. Sturm, T. Zwick, and W. Wiesbeck, "An OFDM system concept for joint radar and communications operations," in *IEEE 69th Vehicular Technology Conference*, April 2009, pp. 1–5.
[50] Y. L. Sit, C. Sturm, L. Reichardt, T. Zwick, and W. Wiesbeck, "The OFDM joint radar-communication system: An overview," in *The Third International Conference on Advances in Satellite and Space Communications*, April 2011, pp. 69–74.
[51] Y. L. Sit, L. Reichardt, C. Sturm, and T. Zwick, "Extension of the OFDM joint radar-communication system for a multipath, multiuser scenario," in *2011 IEEE RadarCon (RADAR)*, May 2011, pp. 718–723.
[52] C. W. Rossler, E. Ertin, and R. L. Moses, "A software defined radar system for joint communication and sensing," in *IEEE Radar Conference (RADAR)*,May 2011, pp. 1050–1055.
[53] T. Guo and R. Qiu, "OFDM waveform design compromising spectral nulling,side-lobe suppression and range resolution," in *IEEE Radar Conference*, May 2014, pp. 1424–1429.
[54] S. C. Thompson and J. P. Stralka, "Constant envelope OFDM for power-efficient radar and data communications," in *International Waveform Diversity and Design Conference*, February 2009, pp. 291–295.
[55] R. F. Tigrek, W. J. A. de Heij, and P. van Genderen, "Relation between the peak to average power ratio and Doppler sidelobes of the multi-carrier radar signal," in *International Radar Conference - Surveillance for a Safer World*, October 2009, pp. 1–6.
[56] Thornton, C. E.; Buehrer, R. M.; Martone, A. F. & Sherbondy, K. D., "Experimental Analysis of Reinforcement Learning Techniques for Spectrum Sharing Radar," 2020 IEEE International Radar Conference (RADAR), 2020, 67-72
[57] T. M. Cover and J. A. Thomas, *Elements of Information Theory*,2nd ed. Hoboken, New Jersey: John Wiley & Sons, 2006.
[58] A. R. Chiriyath, B. Paul, G. M. Jacyna, and D. W. Bliss, "Inner bounds on performance of radar and communications co-existence,"*IEEE Transactions on Signal Processing*, vol. 64, no. 2, pp. 464–474, January 2016.
[59] B. Paul and D. W. Bliss, "Extending joint radar-communications bounds for FMCW radar withDoppler estimation," in *IEEE Radar Conference*, May 2015, pp.89–94.
[60] B. Paul and D. W. Bliss, "The constant information radar," *Entropy*,vol. 18, no. 9, p. 338, 2016. [Online]. Available: http://www.mdpi.com/1099-4300/18/9/338
[61] MATLAB's fmincon. 2016. [Online]. Available: https://www. mathworks.com/help/optim/ug/fmincon.