Joint Sensor Scheduling and Target Tracking with Efficient Bayesian Optimisation

Xingchi Liu*[1], Chenyi Lyu*[1], Seyed Ahmad Soleymani[2], Wenwu Wang[2], Lyudmila Mihaylova[1]

[1]Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield, UK
[2]Centre for Vision Speech and Signal Processing (CVSSP), University of Surrey, Guildford, UK.
Email: xingchi.liu@sheffield.ac.uk, clyu5@sheffield.ac.uk, s.soleymani@surrey.ac.uk,
w.wang@surrey.ac.uk, l.s.mihaylova@sheffield.ac.uk

***Abstract*—The received signal strength measurement has been widely used in search and tracking applications and its benefit is linked with the distance between the transmitter and receiver. This paper proposes an online Bayesian optimisation-based approach that relies on signal strength measurements to schedule multiple sensors for searching and tracking a moving target, without any prior knowledge of the target's state or motion model. A unique contribution lies in incorporating the Gaussian processes factorisation method into the Bayesian optimisation framework, which enhances the effectiveness of the proposed approach. Numerical results obtained from different sizes of measurements demonstrate that the proposed approach can efficiently schedule two unmanned aerial vehicles. Particularly, it achieves at most 21% lower computational time for deciding measurement locations and 79% lower time for updating the surrogate model as compared to the benchmark approach.**

***Index Terms*—Active sensing, Bayesian optimisation, factorised Gaussian process, target tracking, sensor management, unmanned aerial vehicles, hierarchical off-diagonal low-rank (HODLR) factorisation**

I. Introduction

Target tracking is crucial for applications including sea surveillance, autonomous vehicles, and traffic monitoring. Model-based and data-driven approaches have been proposed for this challenge, dealing with data association, group/extended object, and sensor management. However, many approaches rely on informative prior state beliefs which may be unavailable in scenarios like search and rescue or wildlife monitoring. In such cases, the active position estimation [1] becomes a significant challenge as the active sensing platform must locate and track the target simultaneously.

One way to detect and track targets is by analysing the received signal strength (RSS). By measuring the RSS, the distance between the sensor and the target can be estimated. This distance can then be used to track the target over time. Moreover, changes in the RSS signal can provide additional information about the target, such as its velocity and direction of movement [2]. Therefore, active sensing using RSS signals has become an important research area in target tracking and has demonstrated promising results in various applications.

There are three primary types of active sensing techniques for searching and tracking targets, based on the RSS-distance relationship. Geometric approaches [3] rely on model inversion and trilateration, which requires at least three receivers to estimate the target's position accurately. Statistical strategies [4], [5] are designed to account for both model inaccuracies and measurement noise by treating RSS measurements as random variables and applying statistical filtering techniques, such as the Kalman filter and particle filter, to refine raw data points. Data-driven methods leverage machine learning models, such as neural networks [6] or Gaussian processes (GP) [7], to model the RSS-distance relationship, and optimise their parameters during the training process. In this paper, we follow the data-driven idea to design a Bayesian optimisation (BO)-based probabilistic search and tracking approach.

BO is a machine learning-based optimisation method that involves building a surrogate model for the objective function, along with prediction uncertainty quantification using GP. BO iteratively locates the global optimum by using an acquisition function (AF) defined over the surrogate. BO has been applied to solve active sensing and path planning problems [8]–[10]. In these works, GP represents the RSS-distance relationship and AF is used to decide where to place the sensors to collect new measurements and plan the path of the unmanned aerial vehicles (UAVs). However, most existing works focused on modelling stationary processes (e.g., searching for static targets or planning based on static environments). BO can also be used to solve dynamic and non-stationary optimisation problems [11]–[13] by designing kernel functions to account for non-stationary and time-varying processes.

While BO shows promise for solving active sensing problems, its computational complexity cannot be ignored, especially when a large number of measurements are collected by the sensing platform. This is due to the $n \times n$ covariance matrix of GP, which incurs a significant cubic computational cost ($\mathcal{O}(n^3)$) with respect to the number of measurements n. This poses challenges for real-time active sensing since both GP model updates and AF optimisation can be time-consuming, which is related to inversion and determinant evaluation of the covariance matrix. To reduce the computational complexity, this paper explores the idea of factorising the dense covariance matrix into data-sparse hierarchical off-diagonal matrices [14], [15]. This structure can provide a very close approximation to the Cholesky factorisation method with only $\mathcal{O}(n\log n^2)$ cost.

*Equal contribution.

Moreover, the computational complexity was further reduced to $\mathcal{O}(n\log n)$ with a designed low-rank approximation method.

This paper proposes a BO-assisted approach for active sensing management to search and track a moving target. In contrast to our previous work [16] that focused on distributed tracking, this study emphasises active tracking without any prior position information. The main contribution is two-fold. First, a spatial-temporal composite kernel function is designed to account for the non-stationary and time-varying nature of the RSS map. Moreover, several techniques are introduced to reduce the computational cost of the GP used in BO. The proposed approach can schedule multiple UAVs to perform efficient area search and can also be applied to activate sensors over time in sensor management problems [17].

The paper is structured as follows: In Section II, the problem formulations and the fundamentals of BO are introduced. Section III presents the proposed search and tracking approach. Section IV presents the simulation results, while Section V summarises the conclusions.

II. Problem Formulation

We first model the RSS as a black-box function of the coordinates of the measuring location and the time. Define the location of measuring the RSS in time t as $\mathbf{x}_t \in \mathcal{X} \subset \mathbb{R}^2$, where $\mathcal{X}$ is the area of interest. Denote y as the measurement, a black-box dynamic function can be represented as

$$y = f(\mathbf{x}_t, t) + \epsilon, \tag{1}$$

where ϵ is the measurement noise that is assumed to follow a zero-mean Gaussian distribution with variance σ^2. Since the expected value of an RSS measurement is related to the proximity between the target and the sensor, the location with the highest expected value of RSS measurements is identified as the target location. The task of searching and tracking a target over time is equivalent to solving a dynamic optimisation problem [18], specifically finding the maximum of this function. This optimisation problem can be formulated as

$$\max \ f(\mathbf{x}_t, t), \tag{2}$$
$$\textbf{s.t. } \mathbf{x}_t \in \mathcal{X}, \ t \in \mathcal{T}, \tag{3}$$

where $\mathcal{X}$ and $\mathcal{T}$ are the spatial and temporal search spaces, respectively. Next, we describe the GP that represents the black-box function. The UAVs positions are optimised based on the objective AF function.

A. Gaussian Process Regression

The unknown function $f(\mathbf{x}_t, t)$ is a black-box function lacking an analytical form. Therefore, a surrogate model, namely GP, is utilised to represent this function for two reasons. First, GP can quantify the uncertainty of the learned RSSs in a principled way, aiding the exploration-exploitation (EE) tradeoff for maximisation (see more details in the next section). Second, GP functions well with small volumes of data and is particularly useful in the early stages of the search process where few RSS measurements are available for building the surrogate. The GP that is placed as a prior distribution of the function $f(\mathbf{x}_t, t)$ can be written as

$$f(\mathbf{x}_t, t) \sim \mathcal{GP}\left(m(\mathbf{x}_t, t), k((\mathbf{x}_t, t), (\mathbf{x}'_t, t'))\right), \tag{4}$$

where $(\mathbf{x}_t, t)$ and $(\mathbf{x}'_t, t')$ are either the training or the testing input data. $m(\mathbf{x}_t, t)$ and $k((\mathbf{x}_t, t), (\mathbf{x}'_t, t'))$ denote the mean and the covariance functions of GP, respectively.

Suppose that by the time t, n_t RSS measurements have been received with time stamps $t_1, t_2, \cdots, t_{n_t}$. Define $\mathbf{x}_{t_i}$ as the location associated with the measurement at t_i, where $t_i \leq t$. In addition, define y_{t_i} as the measurement at t_i. Therefore, at any t, we can have a set of 3-tuple that can be denoted as $\mathcal{D}_t = \{\mathbf{x}_{t_i}, t_i, y_{t_i}\}_{i=1}^{n_t}$. Given $\mathcal{D}_t$, define $\mathbf{K}_t$ as a covariance matrix with the $(i,j)^{\text{th}}$ entry as $k((\mathbf{x}_{t_i}, t_i), (\mathbf{x}_{t_j}, t_j))$. In addition, define $\mathbf{k}_*$ as a vector with the j^{th} entry as $k((\mathbf{x}_{t_j}, t_j), (\mathbf{x}_{t_*}, t_*))$. Denote the set of measurements received until t by $\mathbf{y}_t = [y_{t_1}, y_{t_2}, \cdots, y_{t_{n_t}}]^\intercal$. The GP predictive distribution at a new input $(\mathbf{x}_*, t_*)$ can be written as

$$\mu_* = m(\mathbf{x}_*, t_*) + \mathbf{k}_*^\intercal (\mathbf{K}_t + \sigma^2\mathbf{I})^{-1}(\mathbf{y}_t - m(\mathbf{x}_*, t_*)), \tag{5}$$
$$\sigma_*^2 = k((\mathbf{x}_*, t_*), (\mathbf{x}_*, t_*)) - \mathbf{k}_*^\intercal (\mathbf{K}_t + \sigma^2\mathbf{I})^{-1}\mathbf{k}_*, \tag{6}$$

where μ_* and σ_*^2 denote the posterior predictive mean and variance of the unknown function at $(\mathbf{x}_*, t_*)$, respectively.

The hyperparameters of GP are learned from the data by maximising the log marginal likelihood that can be written as

$$\log p(\mathbf{y}_t|\mathcal{D}_t, \boldsymbol{\theta}) = -1/2\mathbf{y}_t^\intercal (\mathbf{K}_t + \sigma^2\mathbf{I})^{-1}\mathbf{y}_t - 1/2\log|\mathbf{K}_t + \sigma^2\mathbf{I}| - n_t/2\log 2\pi, \tag{7}$$

where $\boldsymbol{\theta}$ represents the set of hyperparameters.

B. Acquisition Function

Selecting measuring points to evaluate the unknown function sequentially and locate the moving target efficiently with minimum measurements is challenging. There exists an EE dilemma in this decision-making process: Exploring the unknown function provides knowledge but may lead to low search efficiency. However, exploiting the learned knowledge may miss the opportunity to measure higher RSS from under-explored areas. To address this, an AF [19] is optimised to determine measuring points while balancing the EE.

The selection strategy of the next measuring point depends on the type of AF. Here we apply the expected improvement (EI) function [20]. The objective is to find the next measuring point with the highest EI as compared to the incumbent measurement that can be defined as $\tau_{n_t} = \max_{i\in\{1,2,\cdots,n_t\}} \mathbf{y}_{t_i}$. The EI function can be written as

$$\begin{aligned}\alpha_{\text{EI}}(\mathbf{x}_t, t) &\coloneqq \mathbb{E}[[f(\mathbf{x}_t, t) - \tau_{n_t}]^+], \\ &= \sigma(\mathbf{x}_t, t)\phi\left(\frac{\Delta(\mathbf{x}_t, t)}{\sigma(\mathbf{x}_t, t)}\right) + \Delta(\mathbf{x}_t, t)\Phi\left(\frac{\Delta(\mathbf{x}_t, t)}{\sigma(\mathbf{x}_t, t)}\right),\end{aligned} \tag{8}$$

where $\mathbb{E}(\cdot)$ represents the mathematical expectation operation. $\Delta(\mathbf{x}_t, t) = \mu(\mathbf{x}_t, t) - \tau_{n_t}$ is the expected difference between the predicted RSS at a point and the incumbent target. Here $\phi(\cdot)$ and $\Phi(\cdot)$ denote the probability density and cumulative density functions, respectively. In (8), the predictive standard

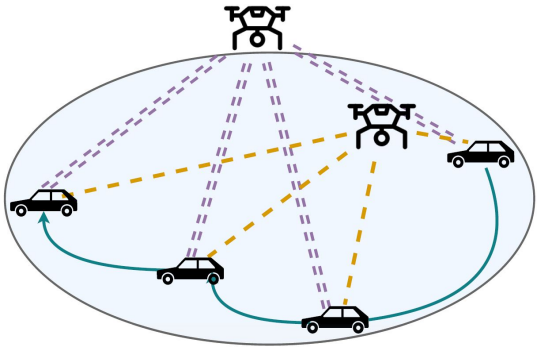

Fig. 1: UAV-based searching and tracking

deviation affects the first term and the predictive mean affects the second term. By maximising the EI function, the EE trade-off can be well-balanced. In this work, the AF is maximised using the grid search method.

III. Efficient BO-assisted search and Tracking

Assuming UAVs can fly over the area of interest to measure RSS from the moving target, building an expressive and scalable surrogate (GP) for the dynamic function is a critical challenge. This section describes the proposed composite kernel and GP factorisation to address these challenges.

A. Kernel Design

Inspired by [13], a spatial-temporal composite kernel

$$k((\mathbf{x}_t,t),(\mathbf{x}'_t,t')) = k_{\text{S}}(\mathbf{x}_t,\mathbf{x}'_t)\cdot k_{\text{T}}(t,t'), \tag{9}$$

is introduced to capture both the spatial and temporal correlations in the unknown time-varying function, where $k_{\text{S}}(\cdot,\cdot)$ represents the spatial kernel, used for characterising the one-time RSS map, $k_{\text{T}}(\cdot,\cdot)$ represents the temporal kernel which considers the temporal correlations of the RSS.

The *designed GP spatial composite kernel* is a sum of a constant kernel $k_{\text{S,Con}}$ and a squared exponential (SE) kernel $k_{\text{S,SE}}$. The constant kernel is added considering the fact that the RSS values in a specific area are above a certain level. The SE kernel represents the smooth changes of RSS. For the *temporal kernel*, we choose the Matérn kernel $k_{\text{T,Mat}}$ since it includes a large class of kernels and is proven to be very useful for matching physical processes realistically.

The kernel function (9) can be rewritten as

$$k((\mathbf{x}_t,t),(\mathbf{x}'_t,t'))=(k_{\text{S,Con}}(\mathbf{x}_t,\mathbf{x}'_t)+k_{\text{S,SE}}(\mathbf{x}_t,\mathbf{x}'_t))\cdot k_{\text{T,Mat}}(t,t'), \tag{10}$$

$$k_{\text{S,Con}}(\mathbf{x}_t,\mathbf{x}'_t) = \Phi, \tag{11}$$

$$k_{\text{S,SE}}(\mathbf{x}_t,\mathbf{x}'_t) = \sigma_m^2 \exp\left(-\|\mathbf{x}_t-\mathbf{x}'_t\|^2/l^2\right), \tag{12}$$

$$k_{\text{T,Mat}}(t,t')=\sigma_m^2\frac{2^{1-v}}{\Gamma(v)}\left(\frac{\sqrt{2v}\|t-t'\|}{l}\right)^v K_v\left(\frac{\sqrt{2v}\|t-t'\|}{l}\right) \tag{13}$$

with σ_m^2 and l being the amplitude and length scale parameters, respectively, Φ represents a constant. $K_v(\cdot)$ is a modified Bessel function and $\Gamma(\cdot)$ is a Gamma function. Moreover, v is a smoothness parameter of Matérn kernel. Different functions belonging to the Matérn kernel can be built with varying v.

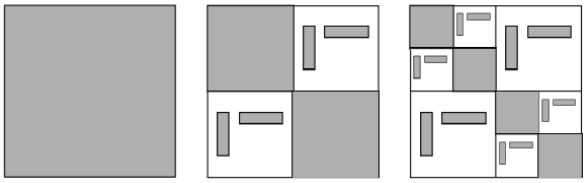

Fig. 2: HODLR matrix at different levels

B. GP Factorisation

This section introduces a novel approach that enhances the BO efficiency by integrating the hierarchical off-diagonal low-rank (HODLR) structure [21] into the GP surrogate model of the BO framework. The HODLR structure divides recursively the covariance matrix and applies rank-revealing lower-upper factorisation hierarchically to certain sub-matrices in the off-diagonal section, while retaining the diagonal parts. Consequently, a subset of columns, rows, or entries is formed, which is much quicker to calculate than the entire matrix. Generally, constructing the HODLR matrix requires a cost of $\mathcal{O}\left(n\log^2(n)\right)$. Fig. 2 gives a graphical representation of the HODLR matrices. A real symmetric matrix $\mathbf{K}_t \in \mathbb{R}^{n_t\times n_t}$ can be decomposed to a two-level HODLR matrix:

$$\mathbf{K}_t = \begin{bmatrix} \mathbf{K}_{t,1} & \mathbf{U}_1\mathbf{V}_1^T \\ \mathbf{V}_1\mathbf{U}_1^T & \mathbf{K}_{t,2} \end{bmatrix}, \tag{14}$$

with the diagonal blocks decomposed into

$$\begin{aligned} \mathbf{K}_{t,1} &= \begin{bmatrix} \mathbf{K}_{t,1}^{(2)} & \mathbf{U}_1^{(2)}\mathbf{V}_1^{(2)^T} \\ \mathbf{V}_1^{(2)}\mathbf{U}_1^{(2)^T} & \mathbf{K}_{t,2}^{(2)} \end{bmatrix}, \\ \mathbf{K}_{t,2} &= \begin{bmatrix} \mathbf{K}_{t,3}^{(2)} & \mathbf{U}_2^{(2)}\mathbf{V}_2^{(2)^T} \\ \mathbf{V}_2^{(2)}\mathbf{U}_2^{(2)^T} & \mathbf{K}_{t,4}^{(2)} \end{bmatrix}. \end{aligned} \tag{15}$$

The $\mathbf{K}_{t,1}$ and $\mathbf{K}_{t,2}$ are the $n/2^j\times n/2^j$ diagonal block matrices from the original matrix $\mathbf{K}_t$ and $\mathbf{U}^{(j)}$, $\mathbf{V}^{(j)}$ matrices are $n/2^j\times r$ matrices with $r\ll n$. j denotes the level of decomposition, which are 2 in this example, and rank r depends on the desired accuracy of the low-rank approximation. A higher rank results in less precision loss and higher computation cost.

Given an HODLR-type factorisation, rapid computation of the inverse of the entire matrix is allowed via the Sherman-Morrison-Woodbury formula [14]. This method has a computational complexity of $\mathcal{O}(n\log n)$ for both solving linear systems and computing the determinant, satisfying the requirements of real-time implementation.

C. Algorithm Overview

The proposed approach reuquires an initial set of data $\mathcal{D}_0$ and a GP prior $\mathcal{GP}_0$ as the surrogate of the unknown dynamic function. The initial input data can be randomly sampled from the search space which is an area of the location $\mathbf{x}$ and a period of time $\mathbf{t}$. The GP is used to construct the AF that leads to searching for the maximum RSS location over time. Besides, we introduce a superscript $k\in\{1,2,\cdots K\}$ to represent different UAVs, where K is the number of UAVs. In addition, at any time t_i, define the spatial search space as the whole area

of interest and the temporal search space as $\mathbf{t} = [t_s, t_s + \gamma]$, where $t_s = t_i + \psi$. A large γ value means a large time interval for one UAV to decide when to collect a measurement, thereby reducing the number of measurements when the total search time is fixed. The proposed algorithm works in an iterative process, the UAVs are scheduled to collect measurements and send them to an edge node which then updates GP and determines new points for UAVs to measure. The proposed algorithm will terminate after a certain period of time T.

The detailed process is described as Algorithm 1.

Algorithm 1 BO-assisted joint sensor scheduling and tracking

Require: Prior surrogate model $\mathcal{GP}_0$, initial data $\mathcal{D}_0$, UAV number K

1: **while** $t_i \leq T$ **do**
2: Receive the K RSS measurements
3: Set the time stamp $t_i = \max\{t_i^1, t_i^2, \cdots, t_i^K\}$
4: Augment data $\mathcal{D}_i \leftarrow \mathcal{D}_{i-1} \cup \left\{\mathbf{x}_{t_i}^k, t_i^k, y_{t_i}^k\right\}_{k=1}^K$
5: Generate the HODLR structure for covariance matrix
6: Update $\mathcal{GP}_i$ by maximising (7)
7: Set the start time stamp $t_\mathrm{s} \leftarrow t_i + \psi$
8: Update search bound of time interval as $\mathbf{t} = [t_\mathrm{s}, t_\mathrm{s} + \gamma]$
9: Determine $\{\mathbf{x}_{t_{i+1}}^k\}_{k=1}^K$ and $\{t_{i+1}^k\}_{k=1}^K$ by sequentially maximising AF as follows:

$$\{\mathbf{x}_{t_{i+1}}^k, t_{i+1}^k\} = \underset{\mathbf{x}_t \in \mathcal{X}, t \in \mathbf{t}}{\arg\max}\, \alpha^k(\mathbf{x}_t, t)$$

10: Send the UAVs to measure the RSSs at $\{\mathbf{x}_{t_{i+1}}^k, t_{i+1}\}_{k=1}^K$
11: $i \leftarrow i + 1$
12: **end while**

IV. Numerical Results

A. Simulation Settings

The log-distance path-loss model [22]

$$y_{t_i} = y_{0,t_i} - \eta \log_{10}(d_{t_i}) + \epsilon, \tag{16}$$

is used to generate RSS measurements, where y_{0,t_i} is a constant characterising the transmission power at t_i with the unit of dBm. An RSS y_{t_i} of a target measured by a UAV. The distance between the target and the UAV at t_i is defined as d_{t_i}. Sensor distortions and environmental interference are represented by ϵ, which is assumed to be a zero-mean Gaussian noise. The proposed algorithm is validated by setting the standard deviation of the Gaussian noise as one dB. Here η is the attenuation gain. We tested the proposed approach in a 400m x 400m area with a target trajectory based on the constant velocity model and an initial state vector of $[50m, 1m/s, 50m, 1m/s]$. Two UAVs are used for tracking, and two BO-based approaches are implemented with different factorisation methods. We also study the impact of the level of HODLR factorisation determined by the block size parameter (it controls the size of the dense diagonal grey block shown in Fig. 2). As a general rule, a smaller block size leads to a higher level of factorisation, which brings faster computation at a cost of lower accuracy than the full GP. All the results are averaged over 100 Monte Carlo simulations.

TABLE I: One-step computational time

γ	Factorisation method	GP update (sec)	AF maximisation (sec)	
			1st UAV	2nd UAV
$\gamma = 1$	HODLR	0.86	0.61	0.62
	Cholesky	0.97	0.77	0.77
$\gamma = 2$	HODLR	0.29	0.72	0.73
	Cholesky	0.49	0.84	0.85
$\gamma = 4$	HODLR	0.04	0.79	0.80
	Cholesky	0.19	0.96	0.97

B. Computational Time

The proposed HODLR factorisation-based BO is compared to Cholesky factorisation-based BO in terms of computational time. Changing γ controls measurement size and evaluates approach efficiency. A large γ value reduces the frequency of collecting measurements and also means not very frequent AF maximisation and GP updates, leading to long search intervals for one-step operation (based on lines 3, 8, and 9 in Algorithm 1). The average received numbers of measurements in the three cases are 424, 268 and 162, respectively. To ensure fairness, we calculate the time required for one-step AF maximisation and GP update. Table I shows that increasing γ decreases GP update time, while AF maximisation time increases due to less efficient grid searching with more grid points required. The AF maximisation time for each UAV is presented separately, as it can be done asynchronously. The proposed HODLR factorisation-based approach outperforms the Cholesky-based approach in terms of shorter time for both AF maximisation and GP update. Specifically, the HODLR factorisation reduces 79% of surrogate model update time when $\gamma = 4$ and up to 21% of AF maximisation time.

C. Tracking Error

The tracking errors of both approaches are presented in Fig. 3. The proposed approach with HODLR factorisation achieves slightly higher errors than the Cholesky factorisation due to a more sparse representation of the covariance matrix. However, it still performs competitively well as compared to the Cholesky factorisation. Particularly, the gap between the two approaches becomes smaller with less training data.

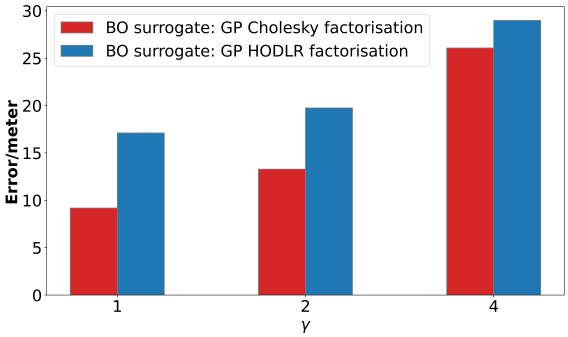

Fig. 3: Tracking error versus γ time step in the BO search (as in Algorithm 1)

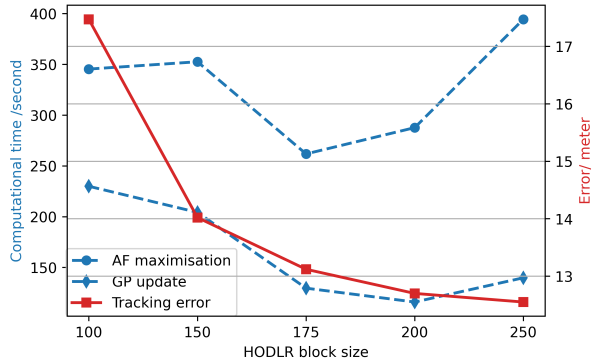

Fig. 4: Tracking error and computational time versus HODLR block size

D. HODLR Factorisation Level

The total computational time and the tracking error affected by the HODLR factorisation level are presented in Fig. 4. Here the HODLR block size is changed while γ is fixed to be one. The implementation of the algorithms was performed in Python, on a PC with i7-12700h CPU. The results reveal that although smaller block sizes do incur higher errors, the algorithm still has efficient computational time. High computational times at small block sizes are due to the computational overhead in constructing the HODLR matrix. The results highlight the importance of choosing the proper HODLR block size to achieve the best computational efficiency. Noted that better computational efficiency results could be achieved by different implementations of the proposed approach.

V. Conclusion

This paper proposes a novel joint sensor scheduling and target tracking approach to send multiple UAVs to track a moving target using RSS measurements. A spatial-temporal composite kernel comprised of a constant kernel, a squared exponential and a Matérn kernel is designed. Then a GP surrogate model for the latent process of RSS generation is constructed that varies over time. Particularly, the HODLR factorisation is integrated into the proposed algorithm to improve its efficiency. Numerical results confirm that the proposed HODLR factorisation-based BO reduces the running time as compared to the standard Cholesky factorisation-based BO while achieving competitive tracking accuracy. Future work will focus on developing non-myopic strategies to solve the sensor scheduling problem.

Acknowledgments

The research was sponsored by the US Army Research Laboratory and the UK MOD University Defence Research Collaboration (UDRC) in Signal Processing, and was accomplished under Cooperative Agreement Number W911NF-20-2-0225. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory, the MOD, the U.S. Government or the U.K. Government. The U.S. Government and U.K. Government are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein. We acknowledge the support from the EPSRC through EP/T013265/1 project NSF-EPSRC: "ShiRAS. Towards Safe and Reliable Autonomy in Sensor Driven Systems" and the support for ShiRAS by the USA National Science Foundation under Grant NSF ECCS 1903466.

References

[1] L. Varotto, A. Cenedese, and A. Cavallaro, "Active sensing for search and tracking: A review," *arXiv preprint arXiv:2112.02381*, 2021.
[2] O. Kaltiokallio, R. Hostettler, and N. Patwari, "A novel Bayesian filter for RSS-based device-free localization and tracking," *IEEE Transactions on Mobile Computing*, vol. 20, no. 3, pp. 780–795, 2021.
[3] V. C. Paterna, A. C. Auge, J. P. Aspas, and M. P. Bullones, "A bluetooth low energy indoor positioning system with channel diversity, weighted trilateration and Kalman filtering," *Sensors*, vol. 17, no. 12, p. 2927, 2017.
[4] J. Röbesaat, P. Zhang, M. Abdelaal, and O. Theel, "An improved BLE indoor localization with Kalman-based fusion: An experimental study," *Sensors*, vol. 17, no. 5, p. 951, 2017.
[5] A. Cenedese, G. Ortolan, and M. Bertinato, "Low-density wireless sensor networks for localization and tracking in critical environments," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 6, pp. 2951–2962, 2010.
[6] X. Liu, P. Li, and Z. Zhu, "Bayesian optimisation-assisted neural network training technique for radio localisation," in *Proceedings of the 2022 95th IEEE Vehicular Technology Conference*, 2022, pp. 1–5.
[7] F. Yin and F. Gunnarsson, "Distributed recursive Gaussian processes for RSS map applied to target tracking," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 3, pp. 492–503, 2017.
[8] M. Carpin, S. Rosati, M. E. Khan, and B. Rimoldi, "UAVs using Bayesian optimization to locate WiFi devices," *arXiv:1510.03592*, 2015.
[9] R. Marchant and F. Ramos, "Bayesian optimisation for intelligent environmental monitoring," in *Proceedings of the IEEE/RSJ International conf. on Intelligent Robots and Systems*. IEEE, 2012, pp. 2242–2249.
[10] A. A. Meera, M. Popović, A. Millane, and R. Siegwart, "Obstacle-aware adaptive informative path planning for UAV-based target search," in *Proceedings of the International Conference on Robotics and Automation*, 2019, pp. 718–724.
[11] J. Snoek, K. Swersky, R. Zemel, and R. Adams, "Input warping for Bayesian optimization of non-stationary functions," in *Proceedings of the 31st International Conference on Machine Learning*, 2014.
[12] R. Martinez-Cantin, "Bayesian optimization with adaptive kernels for robot control," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2017, pp. 3350–3356.
[13] F. M. Nyikosa, M. A. Osborne, and S. J. Roberts, "Bayesian optimization for dynamic problems," *arXiv preprint arXiv:1803.03432*, 2018.
[14] S. Ambikasaran, D. Foreman-Mackey, L. Greengard, D. W. Hogg, and M. O'Neil, "Fast direct methods for Gaussian processes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 2, pp. 252–265, 2016.
[15] A. Aminfar, S. Ambikasaran, and E. Darve, "A fast block low-rank dense solver with applications to finite-element matrices," *Journal of Computational Physics*, vol. 304, pp. 170–188, 2016.
[16] X. Liu, L. Mihaylova, J. George, and T. Pham, "Gaussian process upper confidence bounds in distributed point target tracking over wireless sensor networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 17, no. 1, pp. 295–310, 2023.
[17] A. O. Hero and D. Cochran, "Sensor management: Past, present, and future," *IEEE Sensors Journal*, vol. 11, no. 12, pp. 3064–3075, 2011.
[18] C. Cruz, J. R. González, and D. A. Pelta, "Optimization in dynamic environments: a survey on problems, methods and measures," *Soft Computing*, vol. 15, no. 7, pp. 1427–1448, 2011.
[19] P. I. Frazier, "A tutorial on Bayesian optimization," *arXiv preprint arXiv:1807.02811*, 2018.
[20] J. Mockus, V. Tiesis, and A. Zilinskas, "The application of Bayesian methods for seeking the extremum," *Towards Global Optimization*, vol. 2, pp. 117–129, 1978.
[21] C.-T. Pan, "On the existence and computation of rank-revealing LU factorisations," *Linear Algebra and its Applications*, vol. 316, no. 1-3, pp. 199–222, 2000.
[22] L. Mihaylova, D. Angelova, S. Honary, D. R. Bull, C. N. Canagarajah, and B. Ristic, "Mobility tracking in cellular networks using particle filtering," *IEEE Transactions on Wireless Communications*, vol. 6, no. 10, pp. 3589–3599, 2007.