Anoop C V\textsuperscript{1} and Anup Aprem\textsuperscript{2,3}

\textsuperscript{1}Affiliation not available
\textsuperscript{2}National Institute of Technology Calicut
\textsuperscript{3}National Institute of Technology

May 20, 2024

Anoop C V, Student Member, IEEE
National Institute of Technology Calicut, India
Anup Aprem, Member, IEEE
National Institute of Technology Calicut, India

Abstract—In this paper, we introduce a framework for detecting changes in strategy of an adversarial cognitive radar in the inverse learning context. We model the cognitive radar as a constrained utility maximizer and formulate the problem of detection of change in strategy as a changepoint detection problem in the revealed preference setting. We address the revealed preference strategy changepoint detection problem in a Bayesian setting, wherein we model the utility of cognitive radar as a random direction vector, that follows the von Mises-Fisher distribution with unknown parameters. The main contribution of the paper is the development of Bayesian changepoint detection algorithm that detects the changes in strategy of an adversarial cognitive radar, in an inverse learning context under stochastic revealed preference framework. The main advantages of the proposed method are (i) The proposed method uses a Hamiltonian Monte Carlo sampling algorithm that exploits the Afriat’s theorem as well as a subsampling structure that arises in the posterior, and hence is devoid of the computational burden of solving optimization problems in existing techniques. (ii) Numerical results demonstrate the ability to determine the changes in the beam allocation strategy of a cognitive radar in noise free and noisy adversarial settings. The proposed approach inherently resists observation noise compared to the existing naive approaches. (iii) The proposed changepoint detection performs, on average, five times faster compared to a naive, Generalized Likelihood Ratio approach for changepoint detection. Further, we also demonstrate that the proposed method generalizes existing changepoint detection under revealed preference setting and can be used for applications such as social media and finance. The changepoint detection framework, in the inverse learning context, is useful in the design of both electronic counter measures and electronic counter-counter measures.

Index Terms—Cognitive Radar, Adversarial Inference, Electronic Counter Measures, Bayesian Machine Learning, Utility maximizing behaviour, Changepoints in strategy, Changepoint detection, Constrained sampling from posterior distribution, Hamiltonian Monte Carlo on manifold.

I. INTRODUCTION

Traditionally, radar systems have followed predefined search patterns and scheduled regular track updates, regardless of the dynamic nature of the environment they are monitoring. The cognitive radar concept [1], rooted in the principles of cognition and adaptation, wherein the radar can learn from previous measurements and adjust subsequent radar illuminations, introduced a new era of radar sensing. This innovation not only improved the performance of radar systems but also expanded the horizons of what radar technology can achieve in various domains, ultimately enhancing its utility and impact [2]. The emergence of cognitive radar technology has led to a surge in research within the fields of Electronic Countermeasures (ECM) [3] and Electronic Counter-Countermeasures (ECCM) [4], wherein adversaries employ ECM/ECCM strategies to counteract the capabilities of cognitive radar. In this paper, we study an ECM problem of detecting changes in strategy (changepoint detection problem) of an adversarial cognitive radar. As in [5], we model the adversarial cognitive radar as a constrained utility maximizer, and address the problem in a revealed preference setting. In the following, we first examine the revealed preference framework for cognitive radar in Sec I.A and changepoint detection for revealed preference in Sec I.B

A. Constrained utility maximizing behaviour of cognitive radar and the inverse learning problem in revealed preference setting

A Cognitive Radar (CR) is a “system that in some sense displays intelligence, adapting its operation and its processing in response to a changing environment and target scene” [IEEE Std 686-2017]. A crucial aspect in adapting the operation of a cognitive radar is an optimization step which chooses the next set of radar parameters based on existing measurements of the target and environment. CR, in general, can be considered as a constrained utility maximizer, and articulating the system goals in a mathematical form suitable for optimization is thus critical to its operation [6]. Revealed preference, a concept originally developed in microeconomics, is a framework for non-parametric learning of utility maximizing behaviour [7], and has been widely applied in the context of inverse learning in cognitive radar [5], [8].

1. The dynamics of the cognitive radar - target interaction and the inverse learning problem

A detailed account of the general mathematical frameworks for cognitive radars are available in [9], [10], [11], [6], [12], [13], [14]. Figure 1, a generalized model adopted from [14], [5], depicts the interaction between cognitive radar and target which include: (i) the scene, which comprises the radar target and the environment, typically modelled using the parameterized transition probability, \( p_{\alpha t}(\cdot) \), of the state of the target, \( x_k \), (ii) the
sensor; that consists of a transceiver, which illuminates the environment and senses the reflections - modelled using a time varying sensor parameter, $\beta_t$, and an observation likelihood, $p_{\beta_t}(\cdot)$, (iii) the processor, which transforms the observed data to the perception of the scene - typically modelled using a Bayesian tracker, that estimates the posterior belief $\pi_k$, of target stats given past observation of the sensor, (iv) the controller, which decides the actions to be taken by the sensor and processor modules, taking into account the perception of the scene, produced by the processor, and the controller function is modelled as a constrained optimization problem.

![Figure 1. A generalized mathematical model of cognitive radar and the dynamics of its interaction with a target having Markovian state.](image)

The target can be a drone, an unmanned aerial vehicle or an electromagnetic signal that probes an enemy cognitive radar. The dynamics of the system in Figure 1 involves two different time scales, namely, (i) the fast time scale, $k = 1, 2, \ldots$, over which the Markovian state of the target, $x_k$, evolves according to the transition probability distribution $p_{\alpha_k}(\cdot)$, parameterized by $\alpha_t$, and (ii) slow time scale, $t = 1, 2, \ldots$, over which the cognitive radar adaptively updates the sensor parameters, based on the past observations of the scene. $x_k$, $y_k$ and $\pi_k$ in Figure 1 are modelled as:

$$
\begin{align*}
    x_k &\sim p_{\alpha_t}(x_k|x_{k-1}), x_0 \sim \pi_0, \\
    y_k &\sim p_{\beta_t}(y|x_k), \\
    \pi_k &\sim p(x_k|y_{1:k}).
\end{align*}
$$

The adversarial CR uses a Bayesian tracker to estimate the state (eg: position and velocity) of the target and adapts its mode (eg: choice of waveform, beam allocation strategy). The vector parameter, $\alpha_t$, in (1), at any given time $t$, in revealed preference terminology, can be considered as the probe to the CR [5]. $y_k$ in (1) is the CR’s noisy observation of $x_k$ with observation likelihood $p_{\beta_t}(y|x_k)$, parameterized by $\beta_t$. The vector $\beta_t$ is the optimal response of the radar corresponding to the probe $\alpha_t$, and, as in [5], we assume that the target is able to measure the CR’s optimal response using a radar detector. At any given time $t$, in line with the mission, the optimal decision $\beta_t$, is taken by the radar controller by maximizing some utility function, $U(\cdot)$, subject to certain constraint $\alpha^T_t \beta \leq b_t$, as:

$$
\beta_t = \arg \max_{\alpha^T_t \beta \leq b_t} U(\beta).
$$

The adaptive strategies of CR (such as selection of waveform, changing the aperture or beam orientation, etc) taken by the CR, is decided by the optimal decision $\beta_t$, and hence the choice of utility function and the constraints depends on the desired cognitive capabilities of the radars [15], [6].

Examples of the cognitive radar capabilities include:

(a) Co-existence with other Communication Systems (Eg: Waveform design, Matched illumination, Adaptive nulling, Frequency adaptation, Dynamic adaptation of the search scheme and emission power level, Adaptive beam forming (ABF) of the receiver, and, Time blanking); (b) Automatic Target Recognition and non-co-operative target identification; (c) Adaptation to Changing Environment (Eg: adapt the scan strategy and search volume to the terrain, adapt detection thresholds to the clutter background, optimize the detection and track performance of the desired targets); and (d) Adaptive Resource Management (Eg: waveform parameters, beam pattern, pulse compression characteristics, illumination strategy, revisit interval, and, operating frequency). [15]. The mapping of target state and radar actions, respectively, to probes, $\alpha_t$, and optimal responses $\beta_t$, are detailed with a specific example of adaptive beam allocation capability of CR, in Sec. III.A.

$$
\pi_k = p(x_{1:k}|y_{1:k})
$$

in (1) is the posterior belief of the CR about $x_k$, state of the target, where $y_{1:k} = y_1, \ldots, y_k$. $\pi_k$ can be realized as a Bayesian tracker [14] implemented in the processor unit of CR. Kalman filter is optimal for linear state space model of the target [16]. However, in practice, the state space models are non-linear, and extensions of Kalman filters are used [16], [14], [17]. In this paper, we assume that a perfect estimate, of $\alpha_t$, which is used by the radar controller to find the optimal radar parameter, $\beta_t$, by solving the optimization problem in (2), is available to the target. However, extension to a noisy estimate scenario is immediate.

**Inverse learning**, is the process by which an adversarial counter autonomous system estimates (learns) the underlying belief ($\pi_k$), and, the strategies ($U(\cdot)$) of a cognitive radar, based on the observed, optimal, actions ($\beta_t$) taken by the radar, in response to the control probe signal, $\alpha_t$. Based on the observation that the various cognitive capabilities of a CR are the consequence of utility maximizing behaviour in (2), [5] used the concepts of revealed preference and Afriat’s theorem for the inverse learning of the constrained utility maximizing behaviour of cognitive radar.

In this paper, we extend the framework in [5], to an adversarial changepoint detection setting. The cognitive radar is probed by sequentially switching the states of a drone or a UAV by means of purposeful manoeuvres, and, the sequence of optimal actions taken by the radar, such as the choice of an optimal waveform or optimal beam scheduling, among others, [6], [15], are observed. However, the strategy of the adversaries (CR), and, hence, the underlying utility, which decides the optimal actions, may change from time to time, according to the change in their mission, and, hence it is of interest to learn about the various strategies of the cognitive radar. Furthermore,
if a CR realizes that it has been probed, our drone or UAV needs to take evasive action. In this context, given the sequence of probes and the corresponding optimal actions taken by the enemy’s CR, we address the problem of detecting the changepoints in strategy of the enemy’s CR.

B. Brief overview of changepoint detection frameworks

Accurate detection, of changepoints, the sudden transitions in the generative parameters of a time series, in addition to military applications, finds extensive applications in areas such as social media [18], [19], [20], health monitoring [21], climatology [22] and speech processing [23]. A comprehensive survey of time series changepoint detection techniques is presented in [24]. Bayesian changepoint detection approaches provide better performance compared to classical parametric machine learning techniques [25]. Further, Bayesian changepoint detection techniques are more flexible as well as robust to noise [26] compared to their classical counterparts. In addition, changepoint detection for a utility maximizing agent like CR, as we describe in Section 2, is different compared to classical changepoint detection algorithms in that the utility function is not directly observable but indirectly through the probes and responses. Apart from the adversarial inference of CR, the changepoint detection for utility maximizing behaviour also finds application in detection of outbreak of diseases from online search [27], detecting changes in political beliefs from Twitter [28], and taste changes in consumer behaviour [29]; an example of it’s application in social media is presented in Section III.D.

Extensions to changepoint detection along the line of Afriat’s theorem in revealed preference include [29], [18], [9] and [18] models change in utility as linear perturbations and developed optimization based procedures for detecting the existence of changepoint and for recovering the linear perturbations. In addition, classical revealed preference framework expects noise-free observations. Extension to noisy observations involve determination of confidence boundaries, a laborious and computationally intensive procedure [30], [18], [31].

Unlike [29], [18], [32], which used a deterministic setting of revealed preference, [33] uses a stochastic setting for the revealed preference learning problem. The stochastic setting in [33] models the utility with some unknown probability distribution and facilitates the use of Bayesian approaches of learning. In this paper, we use stochastic revealed preference setting for changepoint detection due to the following reasons: (i) A probabilistic approach allows much more flexible changepoint detection frameworks (compared to linear perturbations in [29], [18]) and modern Bayesian techniques allow inference at a reasonable computational cost [34]. (ii) In classical revealed preference, the utility function (which captures the preference of cognitive agents) is assumed to be deterministic and time invariant. However, there can be slight variations in utility, due to the stochastic behaviour of cognitive agents [35], while having the same order of preferences. Noisy classical revealed preference frameworks [30], [18], [31], models these slight variations as noisy observations, and determine the confidence bound for reliable inferences. Stochastic setting, on the other hand, can inherently capture slight variations by means of dispersion parameters of the underlying utility distributions [33]. However, a naive extension of inverse learning approaches from the stochastic revealed preference framework [33] to changepoint detection problems will be computationally demanding since it involves repeatedly solving constrained optimization problems as in (2) for evaluating the changepoint probability.

In this context, the contributions of this paper are:

1) Developed a Bayesian changepoint detection algorithm under the stochastic revealed preference setting (HBOCPD), to detect the changepoints in strategy of a cognitive radar.

2) Compared to a naive implementation which requires solving multiple linear optimization problems, we propose an algorithm, which exploits Afriat’s theorem, a subsetting structure in the posterior, and the Hamiltonian Monte Carlo (HMC) sampling for efficient implantation of changepoint detection in revealed preference.

3) We illustrate that the proposed framework detects changes in beam allocation strategy of a CR, generalizes existing changepoint detection under revealed preference [18], [33] and is inherently immune to observation noise. In addition, we illustrate that the proposed algorithm is efficient compared to state of art algorithm such as the generalized likelihood ratio [36].

4) We demonstrate that the proposed framework can be extended beyond the military setting and show its application to detect change points in social media and economic theory.

The remaining part of this section is organized as follows: Sec. 1 and 2 introduces the stochastic revealed preference and Bayesian changepoint detection framework, respectively.

1. Stochastic Revealed Preference and Changepoint Detection Framework

An agent is a utility maximizer if at each time $t$, for the probe vector $\alpha_t = [\alpha_t(1), \alpha_t(2), \ldots, \alpha_t(n)]^T \in \mathbb{R}^n_+$, the response of the agent, $\beta_t = [\beta_t(1), \beta_t(2), \ldots, \beta_t(n)]^T \in \mathbb{R}^n$, subject to the budget constraint, $b_t \in \mathbb{R}_+$ satisfies

\[
\beta_t^* = \arg \max_{\beta_t} u_t(\beta_t) = u^T_t \beta_t \\
\text{s.t. } \alpha^T_t \beta_t \leq b_t \\
0 \leq \beta_{t,i} \leq 1, \quad i = 1, \ldots, n.
\]  

(3)

Here, $u_t(\beta)$ is the utility function that quantifies the objective of the cognitive radar in line with its mission, i.e., if the action $\beta$ is preferred over the action $\gamma$ at
any given time \( t \), then \( u_i(\beta) > u_t(\gamma) \). In this paper, we restrict the utility function to be a linear function, i.e., \( u_i(\beta) = u^T_i \beta \), where \( u_i = (u_{i,1}, u_{i,2}, \ldots, u_{i,n}) \in \mathbb{R}^n \), is the utility vector, whose elements \( u_{i,j} \) quantifies the relative preferences of the radar. For example, in optimal beam allocation problem while tracking \( n \) different targets, \( u_{i,j} \) can be modelled as the incentive gained by the CR for allocating a unit time duration beam to the target \( i \) at time \( t \).

Let \( D_t = (\beta^*_t, \alpha_t, b_t) \), the optimal action vector, the probe, and the budget constraint, be the revealed preference observation at time \( t \). A revealed preference dataset \( D_{1:T} = \{D_t = (\beta^*_t, \alpha_t, b_t), t = 1, 2, \ldots, T\} \), consists of revealed preference observations for \( T \) time instants. Stochastic revealed preference assumes that \( u_i \) follows some unknown distribution \( \mathcal{P}_u \) and \( (\alpha_t, b_t) \) follows some random distribution \( \mathcal{P}_{\alpha, b} \) independent of \( u_t \). Further, we assume that there exists \( \alpha > 0 \) such that \( \alpha_t(i) \in [\alpha, 1] \) almost surely for \( i = 1, \ldots, n \). If \( \alpha_t(i) = 0 \), then the corresponding \( \beta^*_t, \alpha_t, b_t \rightarrow \infty \) while satisfying the budget constraint in (3), leading to unbounded values in the objective of the optimization problem (3).

a: Assumption on \( \mathcal{P}_u \):

The distribution \( \mathcal{P}_u \) can be restricted to the unit hyper-sphere \( S^{n-1} = \{u : ||u||_2 = 1\} \) due to the ordinality of utility function. One suitable candidate for \( \mathcal{P}_u \) from Gaussian family is the von Mises Fisher (VMF) distribution. The probability density function of the VMF distribution for a unit norm utility vector \( u \) parameterized by \( \theta = (\mu, \kappa) \) is given by:

\[
f(u; \theta) = \frac{\exp(\kappa \mu^T u)}{\int_{u \in S^{n-1}} \exp(\kappa \mu^T u) du} \propto \exp(\kappa \mu^T u),
\]

where the unit vector \( \mu \in \mathbb{R}^n \) is the mean direction and \( \kappa \in \mathbb{R}_{>0} \) is the concentration parameter. When \( \kappa = \infty \), the distribution shrinks to a point mass on the unit hyper-sphere and the problem transforms to the deterministic setting of the revealed preference framework. Each revealed preference observation \( D_t = (\beta^*_t, \alpha_t, b_t) \) represents a region \( U_t \subset S^{n-1} \), such that the \( u_t \in U_t \) is a solution of (3).

\[
U_t := \{u_t \in S^{n-1} : \beta^*_t \text{ is an optimal solution of (3)}\}.
\]

Given the parameter \( \theta \), the likelihood of the dataset \( D_{1:T} \) is given by:

\[
\mathbb{P}(D_{1:T} | \theta) := \prod_{t=1}^{T} \mathbb{P}(D_t | \theta) = \prod_{t=1}^{T} \int_{u \in U_t} f(u; \theta) du.
\]

The posterior distribution can be obtained as:

\[
\mathbb{P}(\theta | D_{1:T}) = \frac{\mathbb{P}_0(\theta) \mathbb{P}(D_{1:T} | \theta)}{\mathbb{P}(D_{1:T})} \propto \mathbb{P}_0(\theta) \prod_{t=1}^{T} \int_{u \in U_t} f(u; \theta) du,
\]

where, \( \mathbb{P}_0(\theta) \) is the prior distribution of \( \theta \) and \( U_t \) is as defined in (5). The posterior distribution in (7) cannot be computed analytically and hence, as is usual in Bayesian inference, estimated by sampling. Algorithm 2A, given in Appendix, shows a naive Metropolis Hastings algorithm that can be used to generate the posterior distribution, in (7). Evaluation of acceptance ratio \( R \) in Step 6, of Algorithm 2A, involves estimating the ratio of likelihoods (7). A naive approach for estimation of likelihood of observations, \( \mathbb{P}(D_{1:T} | \theta) \), as given in Algorithm 2A of Appendix, requires drawing multiple samples of \( u \sim VMF(\theta) \) and solving the linear programming problem in (3) - a computationally intractable step for large \( T \) (solving a linear programme requires \( O(N^3) \) computations, where \( N \) is the number of constraints).

2. Bayesian Changepoint Detection

We adopt the Bayesian changepoint detection framework in [37] where the sequence of revealed preference observations in \( D_{1:T} \) is assumed to be consisting of non-overlapping partitions \( \{\rho_i, i = 1, 2, \ldots\} \) called runs. All revealed preference observations within a run correspond to utility maximizing behaviour with ‘similar’ utilities, and the utilities changes across different runs. The utilities, within the run \( \rho_i \), are modelled as instances of the VMF distribution in (4), characterized by its parameters \( \theta_i \). The transition between the partitions \( \rho_i \)’s are considered as changepoints. Further, we assume that the parameters \( \theta_i \) changes independently and identically across the different runs. The ‘run-length’ \( r_i \) will represent the number of revealed preferences observations in a run at time \( t \). If the subsequent revealed preference observation correspond to utilities with same \( \theta_i \), the length of the run gets incremented by one, else, there is a changepoint and the run-length falls to zero.

In the subsequent sections, we use \( D_{1:T}^{(r)} = (D_{t-r+1}, D_{t-r+2}, \ldots, D_t) \) to represent the set of revealed preference observations constituting the run of length \( r_t = r \) at any given time \( t \). Inferring the changepoints can be accomplished by computing the posterior of the run-length \( P(r_t | D_{1:t}) \). We proceed by first finding the predictive distribution of the observation \( D_{t+1} \) given \( D_{1:t} \), as follows:

\[
\mathbb{P}(D_{t+1} | D_{1:t}) = \sum_{r_t} \mathbb{P}(D_{t+1} | r_t, D_t^{(r)}) \mathbb{P}(r_t | D_{1:t}).
\]

Since \( \theta \) exclusively defines a run, the predictive distribution conditioned over run-length, \( \mathbb{P}(D_{t+1} | r_t, D_t^{(r)}) \) can be obtained iteratively as

\[
\mathbb{P}(D_{t+1} | r_t, D_t^{(r)}) = \frac{\mathbb{P}(D_{t+1} | \theta) \mathbb{P}_0(\theta) \prod_{i=1}^{t-1} \mathbb{P}(D_{t-i} \theta)}{\mathbb{P}(D_t^{(r)})}. \tag{9}
\]

\( \mathbb{P}_0(\theta) \) in (9) is the prior over the parameter \( \theta \). The posterior distribution of run-length \( \mathbb{P}(r_t | D_{1:t}) \) in (8) can be evaluated using the recursive algorithm, as in [37], as

4
follows:
\[ P(r_t|D_{1:t}) = \frac{P(r_t, D_{1:t})}{P(D_{1:t})} = \sum_{r_{t-1}} P(r_t|r_{t-1})P(D_t|r_{t-1}, D_{t-1})P(r_{t-1}, D_{1:t-1}), \]

where \( P(r_t|r_{t-1}) \) is the conditional changepoint prior. The conditional change point prior models the behaviour of the CR, wherein it switches utility function independent of the presence of an adversary, usually modelled through a distribution with memoryless property. If the discrete a priori probability distribution over the interval between adjacent utility change points is modelled as a geometric distribution with timescale \( \lambda \), as in [37], we can model the changepoint prior as a memoryless hazard function (with time scale \( \lambda \)):
\[ P(r_t|r_{t-1}) = \begin{cases} \frac{1}{\lambda} & \text{if } r_t = 0, \\
1 - \frac{1}{\lambda} & \text{if } r_t = r_{t-1} + 1, \\
0 & \text{otherwise.} \end{cases} \]  

(11)

The steps of Bayesian changepoint detection, in inverse learning context, which implements (8) to (11), is given in Algorithm 1. Computing (9) and (10), require the

**Algorithm 1 Bayesian Changepoint Detection Algorithm**

Require: \( D_{1:t} \) and \( D_{t+1} \)

1: Initialize \( N, \lambda, P_0(\theta), r_1 \leftarrow 0, \Pi \leftarrow 0 \)
2: while \( r_t < t \) do
3: \( \Pi_1 \leftarrow \{ r_t | D_{1:t} \} \{ \text{Naive approach requires LP} \} \)
4: \( n \leftarrow 0, \Pi_2 \leftarrow 0 \)
5: \( D_{t+1} \leftarrow \{ D_{t-r+1}, D_{t-r+2}, \ldots, D_t \} \)
6: \( \Theta \leftarrow \{ \theta \sim P(\theta|D_{t+1}) \} \{ \text{Naive approach requires LP} \} \)
7: while \( n < N \) do
8: \( \Pi_3(n) \leftarrow \{ r_t | D_{1:t} \} \{ \text{Naive approach requires LP} \} \)
9: \( \Pi_4(n) \leftarrow \{ \theta | D_{t+1} \} \{ \text{Naive approach requires LP} \} \)
10: \( \Pi_2 \leftarrow \Pi_2 + \Pi_3(n) + \Pi_4(n) \)
11: \( n \leftarrow n + 1 \)
12: end while
13: \( \Pi \leftarrow \Pi_1 + \Pi_2 \)
14: \( r \leftarrow r + 1 \)
15: end while
16: \( P(D_{t+1}|D_{1:t}) \leftarrow \Pi \)

likelihoods \( \{ P(D_i|\theta) \}_{i=1}^{t+1} \), which do not have any closed form analytical expression. Hence, the likelihood must be empirically evaluated by sampling \( \theta \) from the prior. However, naive sampling approaches existing in literature (Algorithms 1A and 2A, in Appendix), for the inverse learning context, involves solving linear programming problems, which is not a feasible solution, for large \( t \) and \( N \), in Algorithm 1. Sec. II describes the proposed methodology, which avoids the linear programming in Algorithm 1.

The remaining part of this paper is structured as follows: Section II outlines our proposed methodology, while Section III presents numerical results. In Section III, we first demonstrate how we can analyze changepoints in beam allocation strategy of CR. In addition, we also demonstrate the computational efficiency and inherent immunity towards observation noise, of the proposed approach. The conclusion is covered in Section IV. The appendix addresses the naive MH algorithm and a straightforward method for assessing likelihood within an inverse learning framework.

**II. PROPOSED METHODOLOGY**

The proposed methodology combines the following three aspects: (i) characterizing, \( u_t \) in (5), in Sec. A, (ii) a subsetting structure, arising in the posterior distribution, in Sec. B, (iii) a novel sampling algorithm, exploiting the characterization of \( U_t \) and the subsetting structure, in Sec. C.

A. Characterization of \( U_t \)

Lemma II.1 characterizes \( U_t \) in (5) as a set of linear inequalities.

**Lemma II.1** For each \( U_t \), there exists a matrix \( V_t \) and a vector \( w_t \) such that \( U_t = \{ u \in S^{n-1} : V_t u \leq w_t \} \).

The proof of Lemma II.1 can be obtained as a special case of Afriat’s Theorem [38] or can be independently derived from the duality condition of the linear program in (3). Furthermore, using duality, we can explicitly characterize the matrix \( V_t \) and the vector \( w_t \) as in (12) through (14).

When \( \alpha_t^T \beta_t = b_t : \min_{i: \beta_{ti}^r > 0} \left( \frac{u_{ti}}{\alpha_{ti,r}} \right) < 0 \),

\[ \max_{i: \beta_{ti}^r > 0} \left( \frac{u_{ti}}{\alpha_{ti,r}} \right) - \min_{i: \beta_{ti}^r > 0} \left( \frac{u_{ti}}{\alpha_{ti,r}} \right) \leq 0, \]

and, when \( \alpha_t^T \beta_t < b_t : \max_{i: \beta_{ti}^r = 0, u_{ti} \neq 0} \left( \frac{u_{ti}}{\alpha_{ti,r}} \right) \leq 0. \) (14)

B. Subsetting structure of the posterior probability

Given a sequence of observations \( D_t^{(r)} \) belonging to any given run of length \( r_t = r \) at time \( t \), define \( U_t^{(r)} \):
\[ U_t^{(r)} := \{ u_t \in S^{n-1} : \beta_t^r, i = t - r + 1, \ldots, t \text{ are solutions of (3)}. \} \]

(15)

\( U_t^{(r)} \) can be rewritten as: \( U_t^{(r)} = \bigcap_{i=1}^{t-r+1} U_{t,i} \), where \( U_{t,i} \) is as in (5), and characterized by the set of linear inequalities in (12) through (14). It is easy to see from this sub-setting structure of \( U_t^{(r)} \) that:
\[ P(\theta | D_t^{(r)}) = P(V_t u_t \leq w_t ; \theta \sim P(\theta | D_t^{(r-1)}) , u_t \sim VMF(\theta)). \]

(16)

We make use of the recursive expression in (16) to estimate the posterior distribution \( P(\theta | D_t^{(r)}) \) in Steps 3, 6 and 9, of Algorithm 1. This can be realized in practice by collectively considering the constraints (12) through
(14) for all the elements of \( D_t^{(r)} \) to constitute vectors \( V_{t,r+1}^{(r)} = [V_{1,r+1}, ..., V_{s,r+1}]^T \) and \( w_{t,r}^{(r)} = [w_{1,r+1}, ..., w_{s,r+1}]^T \) defining the constraints, in step 6 of Algorithm 2 in Sec. II.C.

C. Hamiltonian Monte Carlo sampling for revealed preference constraints

The posterior distribution, \( P(\theta|D_t^{(r)}) \), in (16) is intractable, and hence as done in Bayesian inference, must be approximated using some posterior sampling techniques [39]. Further, sampling from \( P(\theta|D_t^{(r)}) \) requires that we sample over the manifold, defined by the revealed preference constraints in (12) through (14), on the unit hyper sphere \( S^{n-1} \). Hence, we adopt the geodesic HMC approach in [40]. For any random variable with target distribution \( \pi(u) \), the geodesic HMC uses the Hamiltonian of the following form:

\[
H(u, p) = -\log(\pi(u)) + \frac{1}{2} \log |G(u)| + \frac{1}{2} p^T G(u)^{-1} p,
\]

where \( G \) is the metric tensor defined on the manifold and \( p \sim N(0, G(u)) \) is the additionally introduced momentum variable. The equations of Hamiltonian dynamics is given by

\[
\frac{du}{dt} = \frac{\partial H}{\partial p} = G(u)^{-1} p, \quad \text{and} \quad \frac{dp}{dt} = -\frac{\partial H}{\partial u} = \nabla_u[H(u, p)].
\]

HMC solves (18) to generate proposals. Since the Hamiltonian in (18) is not separable, in general, standard leapfrog integrator [41] cannot be used. In this paper, we use the geodesic integrator proposed in [40], to solve the Hamiltonian dynamics in (18) by splitting the Hamiltonian and alternating simulation between the exact solutions of individual terms.

For a VMF distribution: \( \pi(u) = \exp(\kappa u^T u) \), the Hamiltonian in (17) can be rewritten as

\[
H(u, p) = -\kappa \mu^T u + \frac{1}{2} p^T p,
\]

where we have assumed \( G \) to be identity. To perform geodesic integration, \( H(u, p) \) in (19) is split into two components: \( H^{[1]} = -\kappa \mu^T u \), and, \( H^{[2]} = \frac{1}{2} p^T p \). Hamiltonian dynamics for \( H^{[1]} \) and \( H^{[2]} \) respectively are

\[
\frac{du}{dt} = 0, \quad \frac{dp}{dt} = \kappa \mu, \quad \text{and} \quad \frac{du}{dt} = p, \quad \frac{dp}{dt} = 0.
\]

The proposed geodesic sampling algorithm, to sample \( \theta \sim P(\theta|D_t^{(r)}) \) in (16) is given in Algorithm 2, which forms a key step in the change detection algorithm (Algorithm 1). However, there is an error due to the assumption of \( G \) being identity in (19) - this error can be compensated by projecting the resultant of integration back onto the surface of the manifold, as in Step 27 of Algorithm 2. Further, the component of momentum, tangential to the manifold should be used for solving (20) so that the resultant \( u \) after integration lies on the manifold for sufficiently small integration time steps. Steps 9 through 13 of Algorithm 2 gives the solutions of (20) by means of geodesic integrator.

III. NUMERICAL RESULTS

Sec. III.A illustrates the framework designed for detecting changepoints in strategies of CR, taking beam allocation problem as an example. In Sec. III.B, we demonstrate the computational efficiency of the proposed approach against the state of art Generalized Likelihood
Ratio based changepoint detection framework. Sec. III.C, with the help of a numerical example, demonstrates the inherent immunity of the proposed approach to observation noise, as compared to state of the art IGLR based technique. Further, Sec. III.D demonstrates the versatility of the framework, initially developed for CR, by showcasing its applicability in detecting changepoints in the utility of any constrained utility maximizer. Sec. III.D uses Yahoo! Buzz game, a video game market dataset as an illustrative example. Analysis of changepoints in UK Family Expenditure Survey (UK FES) data, a revealed preference dataset from economics, using the proposed framework, is relegated to the supplementary material.

A. Changepoints in beam allocation strategy of a cognitive radar which tracks multiple targets

We begin by discussing the target radar interaction in an inverse learning setting. Radar is assumed to be fixed at the origin, $(0,0,0)$. Our side comprise three targets, located at $x_k(1) = (x_k(1), y_k(1), z_k(1))$, $x_k(2) = (x_k(2), y_k(2), z_k(2))$ and $x_k(3) = (x_k(3), y_k(3), z_k(3))$, respectively, at any given time $k \in \{0,1,\cdots\}$. $k$ is the fast timescale, over which states of our targets evolve, according to predefined acceleration manoeuvre. Radar adaptively switches its beam between the three targets on a slow timescale, $t \in 0.1,\cdots$, also known as epoch. We denote the fraction of time CR allocates the beam to the target $i$, during epoch $t$ as $\beta_t(i)$. It is assumed that, each target on our side can estimate the corresponding $\beta_t(i)$, with the help of a radar detector.

Similar to [5], we model the Gaussian dynamics of target and the measurements obtained by the CR as:

$$\begin{align*}
x_{k+1}(i) &= A x_k(i) + w_k(i), \quad x_0 \sim \pi_0 \\
y_{k+1}(i) &= C x_k(i) + v_k(i),
\end{align*}$$

(22)

where, $w_k(i) \sim N(0,Q_t(i))$ and $v_k(i) \sim N(0,R_t(i))$. We assume that the state covariance, $Q_t(i)$, and the observation covariance, $R_t(i)$ are known to us and the enemy [5]. $A$ and $C$ in (22) are identity matrices. Bayesian tracker (Kalman filter) of the CR estimates the states of our targets on slow timescale. Let $\alpha_t(i) = Tr \left( \Sigma_{t-1}^{-1}(i) \right)$ be the predicted accuracy of target $i \in \{1,2,3\}$, which is the trace of inverse of the predicted covariance ($\Sigma_{t-1}^{-1}(i)$) of the Kalman predictor at CR, at epoch $t$ [5]. In revealed preference terminology, the price paid by the radar at the beginning of each epoch $t$, for allocating beam to the target $i$, is $\alpha_t(i)$ [5]. In this context, where the CR is equipped with a Bayesian tracker, $\Sigma_{t-1}$ is a deterministic function of manoeuvre covariance (state covariance, $Q_t(i)$), of the target $i$, and hence can be assumed to be known to us [5]. So, we assume that, the probe, $\alpha_t(i)$, is a deterministic function of the state covariance $Q_t(i)$ that we can choose.

The revealed preference interaction between our side and the CR can be summarized as in Fig. 2. At any given slow timescale, $t$, our side probes the radar with a probe vector $\alpha_t = [\alpha_t(1), \alpha_t(2), \alpha_t(3)]^T$, where, the individual elements are the trace of precision matrix (inverse covariance matrix) of the corresponding target, estimated by the CR tracker, which in turn is a deterministic function of our acceleration manoeuvre [5]. The CR responds to our probe with the optimal response vector, $\beta^*_t = [\beta_t(1), \beta_t(2), \beta_t(3)]^T$, whose elements are the fraction of time, beam is allocated to each target by solving the constrained utility maximization problem in (23).

$$\beta^*_t = \arg \max_{\beta_t} u_t(\beta_t) = u_t^T \beta_t$$

(23)

s.t. $\alpha^*_t \beta_t \leq b_t$, $0 \leq \beta_t(i) \leq 1, i = 1,2,3,$

where, $u_t$, is a linear utility vector, which quantify the radar’s preferences or strategy of beam allocation. The rationale for budget constraint, in (23), is that, a CR who is periodically switching its beam between multiple targets, has incentive to devote more time for targets that are cheaper (targets with low prediction accuracy). Furthermore, the budget, $b_t$, can be considered to quantify the maximum achievable average precision, over all targets, of the CR with a given resource constraints, can achieve [5].

For numerical experiments, as in [5], we generated 22 revealed preference observations,

$$D_{t:22} = \{D_t = (\alpha_t, \beta^*_t, b_t)\}_{t=1}^{22},$$

(24)

based on the constrained utility maximizing behaviour in (23). The probe vector $\alpha_t = [1,0.8,1.8]^T$ and budget constraint $b_t = 1.67$ were kept the same for all 22 observations, and the aforementioned values were obtained as random samples, each drawn from $\alpha_t \sim \text{Unif}[0.5, 2]^3$ and $b_t \sim \text{Unif}[1, 3]$ as in [5], [42]. However, since the utility is ordinal, by virtue of Afriat’s theorem, $b_t$ can be chosen as 1 without loss of generality, and hence need not be known to our side in order to make revealed preference inferences [5].

The sequence of optimal responses, $\{\beta^*_t\}_{t=1}^{22}$, are generated by solving (23) (we used simplex method), with the corresponding sequence of directional utility vectors, $\{u_t\}_{t=1}^{22}$, sampled from VMF distribution, as in (25), such that there are three changepoints, at $t = 6,18$, and 20, respectively.

$$u_t \sim \begin{cases} 
\text{VMF}(\theta_1 = (\mu_1, \kappa)), & t = 1,\cdots,5 \\
\text{VMF}(\theta_2 = (\mu_2, \kappa)), & t = 6,\cdots,17 \\
\text{VMF}(\theta_1 = (\mu_1, \kappa)), & t = 18,19 \\
\text{VMF}(\theta_2 = (\mu_2, \kappa)), & t = 20,21,22 
\end{cases}$$

(25)

where $\mu_1 = [0.55,0.1,0.83]^T$, and $\mu_2 = [0.1,0.71,0.70]^T$, are the mean directions, and $\kappa = 3$ is the concentration parameter, of the utility vectors corresponding to two different beam allocation strategies of CR. In the context of CR beam allocation, the elements, of the utility vector, $u_t = [u_t(1), u_t(2), u_t(3)]$, quantify the marginal economic incentive obtained by the CR when allocating a single unit of resource (beam time)
exclusively to each respective target, while allocating no resources to the remaining targets.

The revealed preference dataset, \( D_{1:22} \), in (24), was subsequently subjected to the proposed Bayesian change-point detection algorithm, and the results are discussed in this section. Fig. 3, visualizes the posterior probability \( P(r_t|D_{1:t}) \) as a heat map having an inverted gray scale colour map. The horizontal axis represents the slow timescale \( t \) and the vertical axis corresponds to the run length \( r_t \). A darker gray scale value, in Fig. 3, indicates higher probability. It can be observed from Fig. 3, that \( P(r_t|D_{1:t}) \) falls to zero for all possible non-zero values of \( r_t \) at \( t = 6, 18 \) and 20, which are the changepoints in \( D_{1:22} \). Additionally, we used the proposed sampling technique to empirically estimate \( P(\theta|D_t) \), for all the observations \( D_t \in D_{1:22} \). The relative variations in mean direction of utility \( \mu_t = (\mu_{t,1}, \mu_{t,2}, \mu_{t,3}) \), at changepoints is visualized as jump changes in individual components of empirically reconstructed utility vector, in Fig. 4. Further, distinct values of the components of mean utility in Fig. 4, indicates distinct preferences towards different targets, with 1 being the highest and 0 being the lowest possible values.

Figure 2. The Adversarial inference system model involving an enemy radar fixed at the center and 3 uav’s from our side.

Figure 3. Posterior probability \( P(r_t|D_{1:t}) \) at each time step. The probability value is proportional to the inverted gray scale colour map. Change points can be observed at times 6, 18 and 20.

Figure 4. Components of empirically reconstructed mean value of utility distribution. Corresponding to the change-points in Fig. 1, mean directions can be seen to change at times \( t = 6, 18 \) and 20.

B. Comparison of computation time with Generalized Likelihood Ratio sequential change point detection

Generalized Likelihood Ratio (GLR) change point detection [43, 36], is an extension of the most famous change point detection algorithm - Cumulative Sum (CUSUM) [44], to the case when distributions (assuming they come from the same canonical exponential family, \( \mathcal{E} \)) before \( (p_{\theta_1}) \) and after \( (p_{\theta_2}) \) the changepoint of piecewise iid time series are not known. GLR approaches detects the change points by means of thresholding of the ratio of empirical means of contiguous segments of the time series [36]. Let \( \{u_t\}_{t=1}^n \) be the time series of \( n \) utilities, then the GLR approach estimates the changepoint, \( t = \tau_n(c; \mathcal{E}) \), for a given threshold, \( c \), as [36]:

\[
\tau_n(c; \mathcal{E}) = \min \left\{ t \in [1, n] : \max_{s \in [0,t]} G_{t_0:s:t}^G \geq c \right\}
\]

(26)

where,

\[
G_{t_0:s:t}^G = \sup_{\theta_1, \theta_2} \sum_{t' = t_0}^{s} \log p_{\theta_1}(u_{t'}) + \sum_{t' = s+1}^{t} \log p_{\theta_2}(u_{t'})
\]

\[
- \sup_{\theta} \sum_{t' = t_0}^{t} \log p_{\theta}(u_{t'})
\]

(27)

The components of likelihood ratio, \( G_{t_0:s:t}^G \), in (27) can be approximated using empirical means of the respective utility vectors [36]. However, explicit calculation of empirical means of utilities is not possible in inverse learning context, since the time series of utility \( \{u_t\}_{t=1}^n \) is not directly observed, but, the probes and optimal responses \( \{D_t = (\alpha_t, \beta_t, h_t)\}_{t=1}^n \) of the utility maximizing agent. However, along with the discretization of the latent space of utility on the hypersphere, Algorithm 2A from the Appendix, designed for assessing likelihood in an inverse learning scenario, can substitute the empirical mean computation in the GLR approach, to adapt it to an inverse learning setting, assuming the underlying utility has VMF distribution.

We have used the first nine observations of the revealed preference dataset in (24) to compare the difference in execution time between the GLR in revealed
C. Effect of observation error in changepoint detection

In Sections III. A and III. B, we assumed that the estimated response of CR, in each revealed preference observation, is the optimal solution, $\hat{\beta}_t^*$, of (2). However, in practice, our estimate, $\hat{\beta}_t$, of the CR’s response may not be identical to the optimal solution, $\beta_t^*$, of (2), due to observation error. Observation error may occur due to the following reasons: (i) Error produced by the radar detector for sensing the beam time allocated to individual targets, (ii) Our assumption regarding the number of targets the radar is tracking, in the system model given by Fig. 2, might be incorrect. It’s possible that the radar is allocating a portion of its beam time to additional targets as well, (iii) If the radar gets to know its being probed, it may intentionally take sub-optimal decisions, to confuse the enemy, as commonly used in ECM’s and ECCM’s, such as metacognitive radar in [8].

This section demonstrates the robustness of the proposed HBOCPD framework against the observation error, compared to the naive GLR based sequential changepoint detection technique discussed in Sec. III.B. We considered a total of 8 revealed preference observations, $D_{1:8}^t$, the subset, $D_{2:9}^t$, of revealed preference observations in (24), such that $D_{1:4}^t$ belong to strategy 1 of the CR, and $D_{5:8}^t$ to strategy 2, with, a changepoint at time $t' = 5$. The euclidean distance between the optimal response vectors before and after changepoint is, $l_{\beta} = 0.94$. We define the observation error as, $e = ||\beta_t^* - \hat{\beta}_t||_2$. To evaluate the performance with error, the optimal response vector in $D_{2}^t$ is replaced with an erroneous response vector, with $e = 0.5$. For simplicity, we have introduced error in the observation of beam time allocated to one of the targets alone. However, an extension to the general case is immediate. A statistical formulation of the observation error, for example the model considered in [18], is beyond the scope of the current work.

**a: Changepoint detection in noise with GLR algorithm:**

GLR based changepoint detection involve three hyperparameters, namely, the scan statistics ($c$ in (26)), the required number of samples ($N$ in Algorithm 2A), and the threshold for comparison ($c$ in Algorithm 2A). Among the aforementioned hyperparameters, $c$ exhibits very high sensitivity to observation error. GLR algorithm was applied to the sequence of observations (with and without observation error), and 30 iterations were performed for each $c \in \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$. In each iteration, the changepoint was detected using (26), with the value of $c = 0$. For all $s$, if $G_c^{t_s} \leq 1:8$, in (26), is less than $c$, the GLR algorithm infers that no changepoint is detected in $D_{1:8}^t$, and is referred in Table II as a ‘Miss’. The terms False Alarm (FA), Detection Delay (DD) and Hit are the cases corresponding to $\tau_n(c, \mathcal{E}) < 5$, $\tau_n(c, \mathcal{E}) > 5$ and $\tau_n(c, \mathcal{E}) = 5$, respectively, in (26). The mean values of FA and DD are defined as $\mathbb{E}(5 - \tau_n(c, \mathcal{E})) \times \mathbb{I}_{\{\tau_n(c, \mathcal{E})<5\}}$ and $\mathbb{E}((\tau_n(c, \mathcal{E}) - 5) \times \mathbb{I}_{\{\tau_n(c, \mathcal{E})>5\}})$, respectively. The effect of the choice of $c$ on changepoint detection performance of a naive GLR based approach, with and without observation error, is demonstrated in Table II.

In practice, $l_{\beta}$ and $e$ are not known to us, and hence the smallest possible value of $c$, which maximizes the probability of hit, must be chosen for GLR based changepoint detection approach. In the case, with no observation error, as can be observed from Table II, $c = 0.3$ is the smallest value that gives 100% accuracy for the given revealed preference observations. However, when an erroneous observation is introduced in the second position, the accuracy of changepoint detection (probability of hit) gets drastically reduced to 13%. Since $l_{\beta}$ and $e$ are not known to us, determining the optimal value of $c$, in the presence of observation noise, is a challenging task in naive extension of GLR to adversarial changepoint detection.

![Figure 5. Posterior probability $P(r_t'\mid D^{1:t}_t)$ of HBOCPD algorithm when: (a) $0 \leq e \leq l_{\beta}$, and (b) $e > l_{\beta}$. In (a), the changepoint is accurately identified at $t' = 5$, since the erroneous observation does not violate the constraints (12) through (14). However, in case (b), the constraints are violated and results in false alarms at positions $t' = 2$ and $t' = 3$, apart from the actual changepoint.](image-url)
is greater than \( P(r_t > 0|D_t^{(r)}) \). As can be inferred from Fig. 5(a), the proposed HBOCPD approach is able to accurately detect the changepoint, when \( e < l_\beta \). The inherent immunity of the proposed approach to the observation error, lies in the fact that, constraint optimization problems (similar to (23)), are not involved in inferring the posterior distribution, \( P(\theta|D_t^{(r)}) \), using (16). The observation noise does not affect the proposed HBOCPD approach, as long as the erroneous observation satisfy the constraints (12) through (14). However, \( e > l_\beta \), resulted in false alarms at \( t' = 2 \) and \( t' = 5 \), as shown in Figure. 5(b), a consequence of the inequalities (12) through (14) being violated by the erroneous observation. A detailed examination of error bounds, along with an analysis of how the accuracy of changepoint detection is influenced by the run length, the number of changepoints, and the dimensionality of the latent utility space, is designated for future research.

D. Extending the changepoint detection framework to general revealed preference dataset: Yahoo! Buzz game: Video game market

In this section, we leverage the Yahoo! Buzz game dataset—an extensively utilized revealed preference dataset in the literature. This choice serves to demonstrate the effectiveness and applicability of the proposed changepoint detection framework when analyzing the utility-maximizing behavior of any constrained utility maximizer.

In 2005, Yahoo and O’Reilly Media jointly launched a fantasy market in which the technologies and products which were trending at that point of time were put as stocks, competing against one another. The “Buzz” referred in this dataset corresponds to the online search index proportional to the number of Yahoo! searches made about the technology or brand, and “price” is the value of a single share in the fantasy market. The players traded these stocks and made money, as in a real stock market. An overview of the Buzz game is available in [45]. Further, [46] presents an empirical study of Yahoo! Buzz game data, which infers that the dataset follows utility maximizing behaviour. We consider a subset of this dataset containing the “Video Game” market which comprises 3 different video gaming brands namely “NINTENDO”, “PLAYSTATION” and “XBOX”.

To apply the framework developed in this paper, we transform the Buzz and price as in (28) and (29), respectively. Analogous to probe and optimal response, being the input and output to a utility maximizing consumer in (23), respectively, the Buzz score and the share price are the input and the output of the utility maximizing behaviour of the Yahoo! Buzz game traders. Furthermore, the transformation in (28) captures the phenomena that as Buzz of a brand increases, higher volume of stocks gets traded. (29) normalizes the share price of a stock, the output of the utility maximizing behaviour of traders. As the number of stocks in the Yahoo! Buzz game is fixed, an increase in volume of trades in a stock, tend to increase the share price of the stock.

\[
\alpha_i(t) = 1 - \frac{buzz_i(t) - \min(buzz_i)}{\max(buzz_i) - \min(buzz_i)} + \epsilon, \quad (28)
\]

\[
\beta_i(t) = \frac{price_i(t) - \min(price_i)}{\max(price_i) - \min(price_i)}, \quad (29)
\]

In (28), \( \epsilon \ll 1 \) is a very small positive value that ensures the \( \alpha_i(t) \) never falls to zero. “NINTENDO”, “PLAYSTATION” and “XBOX” are represented by \( i = 1, 2 \) and 3, respectively, in (28) and (29). The budget is assumed as \( b_i = \alpha_i^T \beta_i \). Buzz scores and prices of all three stocks from 1 April 2005 to 19 April 2005 is considered. The proposed framework is applied to this dataset, and the results are given in Fig. 6 and 7. Fig. 6 visualizes the posterior probability \( P(r_t|D_{1:t}) \) as a heat map. The horizontal axis represents the time \( t \) and the vertical axis corresponds to the current run-length \( r_t \). The inverted gray scale intensity represents the \( P(r_t|D_{1:t}) \) in (10), with a darker scale corresponding to higher probability. It is evident from the Fig. 6 that \( P(r_t|D_{1:t}) \) falls to 0, for all possible non-zero run-lengths, at time instances 16 and 17.
indicating two changepoints. The result we have obtained clearly captures the ground truth – the first changepoint corresponds to Playstation market overtaking the Xbox with a drastic 6% jump hike in price, despite having significantly lower Buzz score. The ascent of Xbox price to the top spot following a notable 9.5% increase from its decline justifies the second changepoint at position 16.

Further, we used the proposed sampling technique to empirically estimate $P(\theta|D_t)$, for all the observations $D_t \in D_{1:19}$. The relative variations in mean direction of utility $\mu_t = (\mu_{t,1}, \mu_{t,2}, \mu_{t,3})$, at changepoints, for a fixed value of the dispersion parameter $\kappa = 3$, is visualized as jump changes in individual components of the empirically reconstructed mean utility vector, in Fig. 7. Values of all the components of the reconstructed mean of the relative utility vector are crowded around a constant value, except for at the changepoints, indicating that the players did not have specific preference on any brand over the others. But the components of mean utility at changepoint, $t = 16$, in Fig. 7 indicates that the Playstation market was preferred over Xbox and Nintendo even though the Buzz scores remained consistent.

IV. Conclusion

This paper (i) introduced a Bayesian changepoint detection algorithm under the stochastic revealed preference setting, to detect the changepoints in strategy of a cognitive radar, (ii) formulated utility of cognitive radars as random unit vectors distributed according to Von Mises fisher distribution, with unknown parameters, on a hypersphere, (iii) developed a novel HMC sampling algorithm, that exploits the Afriat’s theorem as well as a subsetting structure that arises in the posterior, which solves computational intractability inherent in existing techniques and (iv) inferred changepoints in beam allocation strategies of a CR in an adversarial inference setting, (v) showed that the proposed changepoint detection performs, on an average, five times faster compared to a naive, GLR approach for changepoint detection, (vi) demonstrated with a numerical example that the proposed approach can accurately detect changepoints, in the presence of noisy revealed preference observations, (vii) demonstrated that the proposed algorithm generalizes existing changepoint detection under revealed preference, and detected the changepoints in preferences of traders in the Yahoo! Tech buzz dataset. The changepoint detection framework, in the inverse learning context, is useful in the design of both ECMs and ECCMs.

APPENDIX

A. Naive Metropolis Hasting algorithm to sample from the posterior distribution

```
Algorithm 1A Naive Metropolis Hasting algorithm to generate posterior distribution – Existing approach [33]
1: $P_0(\theta) \leftarrow$ Prior distribution, $n \leftarrow 0$, $\theta_{lst} = \{}$, $b \leftarrow$ Burn in and $l \leftarrow$ Lag , Initialize $\theta_0 \sim P_0(\theta)$
2: while $n \leq N$ do
3:   $n \leftarrow n + 1$
4:   $\theta \sim Q(\theta_{n-1}) : Q()$ is the proposal distribution
5:   $R \leftarrow \min\{\frac{P(D_{1:t}|\theta^*)}{P(D_{1:t}|\theta)} : 1\}$
6:   $u \sim Unif(0,1)$
7:   if $u < R$ then
8:      $\theta_n \leftarrow \theta$
9:   else
10:      $\theta_n \leftarrow \theta_{n-1}$
11: end if
12: if $(n > b)$ and $(n \mod l = 0)$ then
13:      $\theta_{lst}.append(\theta)$
14: end if
15: end while
```
Algorithm 2A Naive algorithm for evaluating likelihood in inverse learning context – Existing approach [33]

Require: $D_t = (\beta_t, \alpha_t, b_t)$ and $\theta$

1: $N \leftarrow$ Number of samples per likelihood estimation
2: $\epsilon \leftarrow$ Threshold of comparison
3: $n \leftarrow 1$, count $\leftarrow 0$
4: while $n \leq N$ do
5: $n \leftarrow n + 1$
6: $u \leftarrow V M F(\theta)$
7: $\hat{x} \leftarrow L P(u, \alpha_t, b_t)$ \{Stages in main document\}
8: $d = \|\hat{x}_t - \hat{x}\|_2$
9: if $d \leq \epsilon$ then
10: count $\leftarrow$ count $+ 1$
11: end if
12: end while
13: $P((\beta_t, \alpha_t, b_t)|\theta) \leftarrow \frac{\text{count}}{N}$

References

models with approximate bayesian computation,” *Cognitive science*, vol. 43, no. 6, p. e12738, 2019.


