MMSE adaptive waveform design for active sensing with applications to MIMO radar

MMSE Adaptive Waveform Design for Active Sensing with Applications to MIMO Radar

Steven Herbert, James R. Hopgood, *Member, IEEE,* Bernard Mulgrew, *Fellow, IEEE*

***Abstract*—Minimising the expected mean squared error is one of the fundamental metrics applied to adaptive waveform design for active sensing. Previously, only cost functions corresponding to a lower bound on the expected mean squared error have been expressed for optimisation. In this paper we express an exact cost function to optimise for minimum mean squared error adaptive waveform design (MMSE-AWD). This is expressed in a general form which can be applied to non-linear systems.**

Additionally, we provide a general example for how this method of MMSE-AWD can be applied to a system that estimates the state using a particle filter (PF). We make the case that there is a compelling reason to choose to use the PF (as opposed to alternatives such as the Unscented Kalman filter and extended Kalman filter), as our MMSE-AWD implementation can re-use the particles and particle weightings from the PF, simplifying the overall computation.

Finally, we provide a numerical example, based on a simplified multiple-input-multiple-output radar system, which demonstrates that our MMSE-AWD method outperforms a simple non-adaptive radar, whose beam-pattern has a uniform angular spread, and also an existing approximate MMSE-AWD method.

***Index Terms*—Adaptive waveform design, minimum mean squared error, active sensing, MIMO, radar, Bayesian, particle filters, optimal design.**

I. Introduction

In active sensing systems, the information we acquire about the targets can inform the design of future waveforms to be transmitted in order to maximise, in some sense, the future information we expect to obtain about those targets. For example, if the angle of a point target is to be estimated by a multiple-input-multiple-output (MIMO) radar system, then as the angle estimate variance decreases the power can be steered predominantly in the direction of the target in a manner which has an appearance similar to beam-steering. This simple and intuitive principle is the basis for adaptive waveform design.

In this paper, we focus on adaptive waveform design for radar systems, however it should be appreciated that the framework used and consequent optimisation function derived herein is more generally applicable. Adaptive waveform design for radar systems is also known as *cognitive radar* and Haykin [1], Ender and Brüggenwirth [2], and Sira *et al* [3] provide general introductions to the subject. Additionally, Huleihel *et al* [4, Fig. 1] show the basic architecture of a cognitive radar system. In this paper, we primarily address adaptive waveform design in MIMO radar systems, as the resultant angular distribution of the transmitted waveform provides a nice visual demonstration of the adaptive waveform design. However the same principle could also be applied to, for example, waveform design in the frequency domain, as is the subject of existing literature tackling related problems [5], [6].

There are various approaches to adaptive waveform design for estimation of parameters associated with one or more targets, including transmitting the conjugate of the previously received signal [7] and steering the waveform towards the direction of the current estimate of the target, as considered by Huleihel *et al* [4]. A more sophisticated method it to design the waveform to maximise mutual information (MI) [6], [8], [9] and the related method of maximising signal to interference and noise ratio [10]–[13]. Another approach is designing the waveform to minimise mean squared error (MMSE), in which linear and non-linear systems are typically addressed separately. Linear systems have been extensively studied [14]–[16], however there is less literature on non-linear systems. The most promising approach to MMSE adaptive waveform for non-linear systems is that of Huleihel *et al* [4], who optimise a lower bound of the mean squared error. There exist special cases where these two approaches (i.e., maximising MI and MMSE) yield the same result [9], [17], however in general they do not, and thus they can be considered to be complementary approaches to the same problem.

In this paper we address MMSE adaptive waveform design for systems that are, in general, non-linear, in particular we express and optimise the cost function stated by Huleihel *et al* [4, equation (4)], of which the authors optimise a lower bound. As acknowledged by Huleihel *et al* when justifying their approach of optimising the lower bound, 'it is difficult to obtain an analytic expression' for this cost function; to our knowledge the analytical expression we provide in this paper is novel.

We consider the target parameter estimation to be conducted in a Bayesian framework and, in addition to contributing the aforementioned analytical cost function expression, we also demonstrate how the cost function can be optimised in the case where the posterior target probability density function (PDF) is approximated by a particle filter (PF). This is noteworthy owing to the fact that the particle weightings in the PF can be re-used in the adaptive waveform design algorithm, making it an efficient implementation. Given that we have already excluded the linear Gaussian case (i.e., for which the posterior could be analytically calculated using a Kalman filter) this is a compelling reason to use a PF for target parameter estimation,

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) Grant number EP/K014277/1 and the MOD University Defence Research Collaboration (UDRC) in Signal Processing.

S. Herbert, J. Hopgood and B. Mulgrew are with the Institute for Digital Communications, School of Engineering, University of Edinburgh, Edinburgh, UK (e-mail: S.Herbert@ed.ac.uk, James.Hopgood@ed.ac.uk, Bernie.Mulgrew@ed.ac.uk).

rather than alternatives such as an extended Kalman filter or an Unscented Kalman filter.

Our cost function is not generally convex, however it is differentiable and thus we derive an expression for the gradient of the cost function and show how the cost function can be locally optimised using gradient descent. We apply this to a numerical MIMO radar example and, whilst the results from this optimisation are promising and demonstrate the principle of our approach as well as potentially being sufficient for some applications, it is of future interest to investigate whether more sophisticated optimisation techniques can exploit the cost function surface more effectively. This is especially relevant when considering the fact that our analytical cost function has been expressed in a completely general form, and thus can apply to a broad range of fields in which adaptive waveform design can be applied, for example sonar [18] and tomography [19].

It is also relevant to consider whether in the future approximate MMSE waveform design methods could adequately improve the target parameter estimation performance without requiring as large a computational load as the method proposed in this paper. For example if an approximate cost function were proposed that yielded a convex optimisation surface. To this end an additional application of the exact, general cost function presented herein is that it provides a benchmark to measure the performance of any future approximate MMSE waveform design methods.

A. Contributions

In this paper, we make the following main contributions:

- We express analytically a cost function to optimise for MMSE adaptive waveform design in non-linear systems.
- We provide a general implementation where the target parameters are estimated using a PF, and the waveform is adaptively optimised using gradient descent.
- We provide a numerical example, based on a MIMO radar system, that demonstrates a decrease in root mean squared error of the target parameter estimation, compared to a non-adaptive system as well as the method proposed by Huleihel *et al*. Moreover, we reason that this specific implementation may suffice in its current form for some applications, and thus has intrinsic value, as well as acting as a proof of principle for our adaptive waveform method.

B. Notation

Throughout the paper we have strived to use standard and simple notation as much as possible. On occasion, we express functions and variables as some letter 'primed', for example x', which denotes a variation (usually simple and/ or only briefly required) of the primed function or variable and is not used in this paper to denote either differentiation, which is always explicitly expressed, or the Hermitian transpose, which is expressed $(\cdot)^H$. Other notation used in this paper includes transpose, which is expressed as $(\cdot)^T$ and complex conjugation, which is expressed as $(\cdot)^*$.

The system model is defined in terms of the kth pulse, and vectors and matrices corresponding to the kth pulse only are denoted by subscript k, i.e., $\mathbf{X}_k$. The short-hand superscript $k-1$ is used to denote the inclusion of the corresponding matrix/ vector for all steps up to $k-1$, i.e., $\mathbf{X}^{k-1}$. This use of superscript is not to be confused with the ith sample of a random variable, which is denoted by a parenthesised superscript i, i.e., $\boldsymbol{\theta}_k^{(i)}$. To aid readability, we dispense with limits on integrals when the integral is performed over the entire support of the relevant variable. Throughout the paper $p(\cdot)$ is used for probability density.

C. Paper organisation

The remainder of the paper is organised as follows: in Section II we define the system model; in Section III we derive and express a general form of the cost function to be optimised; in Section IV we provide further derivations for a particular, but still quite general, implementation using a PF; in Section V we show how this implementation can be optimised using gradient descent; and in Section VI we provide numerical results for an example of this implementation. In Section VII we state and discuss the computational complexity of this implementation; and finally in Section VIII we draw conclusions.

II. System Model

We use the system model detailed in Huleihel *et al* [4] as our basic starting point. Specifically, we consider the kth pulse (opportunity to adaptively design the waveform), which consists of L different waveforms (snapshots). The signal is transmitted by N_T elements and received by N_R elements. We transmit a waveform represented by the matrix, $\mathbf{S}_k$, the lth column of which is a column vector corresponding to the lth snapshot, and whose rows correspond to the complex signal transmitted at the given snapshot by each of the N_T transmitting elements, i.e., $\mathbf{S}_k \in \mathbb{C}^{N_T \times L}$. The received waveform is represented by the matrix, $\mathbf{X}_k \in \mathbb{C}^{N_R \times L}$, and again the columns correspond to the snapshots, with each row corresponding to the complex signal received on the respective receiving element. Thus we define the channel:

$$\mathbf{X}_k = \mathbf{H}_k(\boldsymbol{\theta}_k)\mathbf{S}_k + \mathbf{N}_k, \tag{1}$$

where $\mathbf{H}_k(\boldsymbol{\theta}_k) \in \mathbb{C}^{N_R \times N_T}$ represents the channel response as a non-linear function (in general) of $\boldsymbol{\theta}_k$, a vector of the Q parameters of the target, i.e., $\boldsymbol{\theta}_k \in \mathbb{C}^{Q \times 1}$, which do not vary within any given step. Thus the received signal is a linear function of the transmitted signal, but a non-linear function of the model parameters. In (1), $\mathbf{N}_k \in \mathbb{C}^{N_R \times L}$ represents additive white Gaussian noise (AWGN). The noise is circularly symmetric complex, i.e., each element of $\mathbf{N}_k$ is a complex number whose real part is an independent zero mean Gaussian random variable with variance σ_n^2 and whose imaginary part is also an independent zero mean Gaussian random variable with variance σ_n^2, and the various elements of $\mathbf{N}_k$ are mutually independent.

It is worth noting that this model is somewhat simplified compared to physical reality, with, for example, phenomena

such as noise that is not uniformly spread with angle (including deliberate interference sources at located at certain angles) and indeed signal dependent interference not taken into account (note that complementary existing literature does tackle the physically interesting case of signal dependent interference [20]–[23]). We have chosen to use this simplified model for consistency with that in the literature, most importantly the scenario considered by Huleihel *et al* [4] which explicitly provides the motivation for this research. Our choice to use this model is also motivated by a desire to demonstrate the principle of our method in a simple and clear manner. For completeness, it is worth noting that the Bayesian method proposed herein theoretically extends to any scenario for which $p(\mathbf{X}_k|\boldsymbol{\theta}_k, \mathbf{S}_k)$ can be expressed.

We now extend the scenario considered by Huleihel *et al* [4] slightly to include the situation in which the state can vary from step to step:

$$\boldsymbol{\theta}_k = \mathbf{f}_{k-1}(\boldsymbol{\theta}_{k-1}, \mathbf{v}_{k-1}), \tag{2}$$

where $\mathbf{f}_{k-1}(.)$ is an arbitrary function and $\mathbf{v}_{k-1}$ is noise, which is independent of $\mathbf{N}_k$.

III. Expression of the Cost Function

Given the transmitted waveform, the system estimates the system state as $\hat{\boldsymbol{\theta}}_k$, the expectation of the posterior PDF:

$$\begin{aligned} &p(\boldsymbol{\theta}_k|\mathbf{X}^k, \mathbf{S}^k) \\ &\quad = p(\boldsymbol{\theta}_k|\mathbf{X}_k, \mathbf{X}^{k-1}, \mathbf{S}_k, \mathbf{S}^{k-1}) \\ &\quad \propto p(\mathbf{X}_k|\boldsymbol{\theta}_k, \mathbf{X}^{k-1}, \mathbf{S}_k, \mathbf{S}^{k-1}) p(\boldsymbol{\theta}_k|\mathbf{X}^{k-1}, \mathbf{S}_k, \mathbf{S}^{k-1}) \\ &\quad = p(\mathbf{X}_k|\boldsymbol{\theta}_k, \mathbf{S}_k) \times p(\boldsymbol{\theta}_k|\mathbf{X}^{k-1}, \mathbf{S}^{k-1}), \end{aligned} \tag{3}$$

where the first PDF in the right-hand side (RHS) of (3) has been simplified to its final form by observing that the definition of the system model in (1) is such that $\mathbf{X}_k$ has no dependence on $\mathbf{X}^{k-1}, \mathbf{S}^{k-1}$ given $\boldsymbol{\theta}_k, \mathbf{S}_k$. For the second PDF in the RHS of (3), this has been simplified to its final form because the probability of the parameter $\boldsymbol{\theta}_k$ does not depend on the transmitted waveform $\mathbf{S}_k$ unless also conditioned on the observation $\mathbf{X}_k$.

Thus (3) is in the form of the probability given the current measurement multiplied by the prediction from previous measurements, which reduces to the *Kalman* filter for the linear Gaussian case. From the filter, we can get an estimate of the state, $\hat{\boldsymbol{\theta}}_k$ which is the expectation of $\boldsymbol{\theta}_k$. This can be expressed as a general function:

$$\begin{aligned} \hat{\boldsymbol{\theta}}_k &= \mathbb{E}(\boldsymbol{\theta}_k|\mathbf{X}_k, \mathbf{X}^{k-1}, \mathbf{S}_k, \mathbf{S}^{k-1}) \\ &= \mathbf{g}_k(\mathbf{X}_k, \mathbf{S}_k, \boldsymbol{z}_{k-1}), \end{aligned} \tag{4}$$

where $\boldsymbol{z}_{k-1}$ are the parameters associated with the PDF of the prediction of $\boldsymbol{\theta}_k$ given $(\mathbf{X}^{k-1}, \mathbf{S}^{k-1})$, i.e., $p(\boldsymbol{\theta}_k|\mathbf{X}^{k-1}, \mathbf{S}^{k-1}) = \mathbf{g}'_k(\boldsymbol{\theta}_k, \mathbf{z}_{k-1})$. Having expressed the parameter estimation in general terms, we progress to express a general cost function to optimise for adaptive waveform design. From Huleihel *et al* [4, equation (6)] we can write:

$$\text{minimise: } \Sigma_k = \mathbb{E}[(\hat{\boldsymbol{\theta}}_k - \boldsymbol{\theta}_k)(\hat{\boldsymbol{\theta}}_k - \boldsymbol{\theta}_k)^T|\mathbf{X}^{k-1}] \text{ wrt } \mathbf{S}_k, \tag{5}$$

$$\text{subj. to: } \mathrm{tr}\left(\frac{1}{L}\mathbf{S}_k\mathbf{S}_k^H\right) \leq \mathrm{P}, \tag{6}$$

where power is denoted P and $\mathrm{tr}(.)$ is the matrix trace operation, i.e., the cost in (5) is optimised subject to the maximum power constraint (6).

The minimisation term in (5) is a co-variance matrix, and thus is not strictly a meaningful objective term to optimise. Thus we choose to minimise the trace of the co-variance matrix. Including the dependence on $\mathbf{S}_k$ and explicitly expressing the dependence on $\mathbf{S}^{k-1}$, we can write:

$$\begin{aligned} \Sigma_k = \iint &(\hat{\boldsymbol{\theta}}_k - \boldsymbol{\theta}_k)^T(\hat{\boldsymbol{\theta}}_k - \boldsymbol{\theta}_k) \\ &p(\hat{\boldsymbol{\theta}}_k, \boldsymbol{\theta}_k|\mathbf{X}^{k-1}, \mathbf{S}^{k-1}, \mathbf{S}_k)\, \mathrm{d}\hat{\boldsymbol{\theta}}_k\, \mathrm{d}\boldsymbol{\theta}_k, \end{aligned} \tag{7}$$

we can express the probability term in the integrand of (7):

$$\begin{aligned} &p(\hat{\boldsymbol{\theta}}_k, \boldsymbol{\theta}_k|\mathbf{X}^{k-1}, \mathbf{S}^{k-1}, \mathbf{S}_k) \\ &\quad = p(\hat{\boldsymbol{\theta}}_k|\boldsymbol{\theta}_k, \mathbf{X}^{k-1}, \mathbf{S}^{k-1}, \mathbf{S}_k) \times p(\boldsymbol{\theta}_k|\mathbf{X}^{k-1}, \mathbf{S}^{k-1}, \mathbf{S}_k) \\ &\quad = p(\hat{\boldsymbol{\theta}}_k|\boldsymbol{\theta}_k, \mathbf{X}^{k-1}, \mathbf{S}^{k-1}, \mathbf{S}_k) \times p(\boldsymbol{\theta}_k|\mathbf{X}^{k-1}, \mathbf{S}^{k-1}). \end{aligned} \tag{8}$$

Notice that the second PDF in the RHS of (8) is the prediction for the parameter estimation (i.e., in (3)), and therefore available. Also notice that, as in (3), we have made the final simplification in the second PDF in the RHS of (8) because the probability of the parameter $\boldsymbol{\theta}_k$ does not depend on the transmitted waveform $\mathbf{S}_k$ unless also conditioned on the observation $\mathbf{X}_k$. Focussing therefore on the first term in the RHS of (8), the PDF of the next parameter estimation, which we express by marginalising out $\mathbf{X}_k$:

$$\begin{aligned} p(\hat{\boldsymbol{\theta}}_k|\boldsymbol{\theta}_k, \mathbf{X}^{k-1}, \mathbf{S}^{k-1}, \mathbf{S}_k) &= \int p(\hat{\boldsymbol{\theta}}_k|\mathbf{X}_k, \boldsymbol{\theta}_k, \mathbf{X}^{k-1}, \mathbf{S}^{k-1}, \mathbf{S}_k) \\ &\qquad p(\mathbf{X}_k|\boldsymbol{\theta}_k, \mathbf{X}^{k-1}, \mathbf{S}^{k-1}, \mathbf{S}_k)\, \mathrm{d}\mathbf{X}_k \\ &= \int p(\hat{\boldsymbol{\theta}}_k|\mathbf{X}_k, \mathbf{X}^{k-1}, \mathbf{S}^{k-1}, \mathbf{S}_k) \\ &\qquad p(\mathbf{X}_k|\boldsymbol{\theta}_k, \mathbf{X}^{k-1}, \mathbf{S}^{k-1}, \mathbf{S}_k)\, \mathrm{d}\mathbf{X}_k \\ &= \int \delta(\hat{\boldsymbol{\theta}}_k - \mathbf{g}_k(\mathbf{X}_k, \mathbf{S}_k, \boldsymbol{z}_{k-1})) \\ &\qquad p(\mathbf{X}_k|\boldsymbol{\theta}_k, \mathbf{X}^{k-1}, \mathbf{S}^{k-1}, \mathbf{S}_k)\, \mathrm{d}\mathbf{X}_k, \end{aligned} \tag{9}$$

where $\delta(.)$ is the *Dirac* delta function. Substituting (9) into (8) and then into (7):

$$\begin{aligned} \Sigma_k = \iint &(\hat{\boldsymbol{\theta}}_k - \boldsymbol{\theta}_k)^T(\hat{\boldsymbol{\theta}}_k - \boldsymbol{\theta}_k) p(\boldsymbol{\theta}_k|\mathbf{X}^{k-1}, \mathbf{S}^{k-1}) \\ &\left(\int \delta(\hat{\boldsymbol{\theta}}_k - \mathbf{g}_k(\mathbf{X}_k, \mathbf{S}_k, \boldsymbol{z}_{k-1})) \right. \\ &\left. p(\mathbf{X}_k|\boldsymbol{\theta}_k, \mathbf{X}^{k-1}, \mathbf{S}^{k-1}, \mathbf{S}_k)\, \mathrm{d}\mathbf{X}_k \right) \mathrm{d}\hat{\boldsymbol{\theta}}_k\, \mathrm{d}\boldsymbol{\theta}_k. \\ = \iiint &(\hat{\boldsymbol{\theta}}_k - \boldsymbol{\theta}_k)^T(\hat{\boldsymbol{\theta}}_k - \boldsymbol{\theta}_k) p(\boldsymbol{\theta}_k|\mathbf{X}^{k-1}, \mathbf{S}^{k-1}) \\ &\delta(\hat{\boldsymbol{\theta}}_k - \mathbf{g}_k(\mathbf{X}_k, \mathbf{S}_k, \boldsymbol{z}_{k-1})) \\ &p(\mathbf{X}_k|\boldsymbol{\theta}_k, \mathbf{X}^{k-1}, \mathbf{S}^{k-1}, \mathbf{S}_k)\, \mathrm{d}\mathbf{X}_k\, \mathrm{d}\hat{\boldsymbol{\theta}}_k\, \mathrm{d}\boldsymbol{\theta}_k \\ = \iiint &(\hat{\boldsymbol{\theta}}_k - \boldsymbol{\theta}_k)^T(\hat{\boldsymbol{\theta}}_k - \boldsymbol{\theta}_k) p(\boldsymbol{\theta}_k|\mathbf{X}^{k-1}, \mathbf{S}^{k-1}) \\ &\delta(\hat{\boldsymbol{\theta}}_k - \mathbf{g}_k(\mathbf{X}_k, \mathbf{S}_k, \boldsymbol{z}_{k-1})) \\ &p(\mathbf{X}_k|\boldsymbol{\theta}_k, \mathbf{X}^{k-1}, \mathbf{S}^{k-1}, \mathbf{S}_k)\, \mathrm{d}\hat{\boldsymbol{\theta}}_k\, \mathrm{d}\mathbf{X}_k\, \mathrm{d}\boldsymbol{\theta}_k, \end{aligned} \tag{10}$$

note that, as specified in Section I-B, all the integrals are evaluated over the entire support of the corresponding variables, thus the integrations over $\hat{\boldsymbol{\theta}}_k$ and $\mathbf{X}_k$ are independent and so it is permissible to switch the order of integration as in (10). From (10), using the sifting property of the delta function, we get:

$$\Sigma_k = \iint (\mathbf{g}_k(\mathbf{X}_k, \mathbf{S}_k, \boldsymbol{z}_{k-1}) - \boldsymbol{\theta}_k)^T (\mathbf{g}_k(\mathbf{X}_k, \mathbf{S}_k, \boldsymbol{z}_{k-1}) - \boldsymbol{\theta}_k) \\ p(\boldsymbol{\theta}_k|\mathbf{X}^{k-1}, \mathbf{S}^{k-1}) p(\mathbf{X}_k|\boldsymbol{\theta}_k, \mathbf{S}_k) \, \mathrm{d}\mathbf{X}_k \, \mathrm{d}\boldsymbol{\theta}_k, \quad (11)$$

which has again been simplified by noticing that $\mathbf{X}_k$ has no dependence on $\mathbf{X}^{k-1}, \mathbf{S}^{k-1}$ given $\boldsymbol{\theta}_k, \mathbf{S}_k$ (i.e., as in (3)).

Thus optimising (11) with respect to the next transmitted waveform, $\mathbf{S}_k$, minimises the expectation of the squared error of the next estimate of the state. In practise, however, evaluation of $p(\boldsymbol{\theta}_k|\mathbf{X}^k, \mathbf{S}^k)$ may only be approximately possible, for example by a discrete approximation, which has implications for the nature of $\mathbf{g}_k(\mathbf{X}_k, \mathbf{S}_k, \boldsymbol{z}_{k-1})$.

Discussion of derivation of cost function

The derivation of (11), an analytical expression for (5), is novel as previously only solutions based on lower bounding exist for the general (i.e., non-linear) case. The crucial insight that enabled us to express this exact form of the MMSE cost function is that, prior to designing the kth waveform, both $\boldsymbol{\theta}_k$ and $\hat{\boldsymbol{\theta}}_k$ in (5) are random variables, the latter of which depends on the waveform to be designed, $\mathbf{S}_k$. Thus it is possible to rearrange (5) as a cost which is a function of $\mathbf{S}_k$.

Ostensibly, it may appear that a simpler expression equivalent to (11) exists, i.e., substituting (8) into (7) directly yields:

$$\Sigma_k = \iint (\hat{\boldsymbol{\theta}}_k - \boldsymbol{\theta}_k)^T (\hat{\boldsymbol{\theta}}_k - \boldsymbol{\theta}_k) \times p(\hat{\boldsymbol{\theta}}_k|\boldsymbol{\theta}_k, \mathbf{X}^{k-1}, \mathbf{S}^{k-1}, \mathbf{S}_k) \\ \times p(\boldsymbol{\theta}_k|\mathbf{X}^{k-1}, \mathbf{S}^{k-1}) \, \mathrm{d}\hat{\boldsymbol{\theta}}_k \, \mathrm{d}\boldsymbol{\theta}_k, \quad (12)$$

However, a little consideration suggests that $p(\hat{\boldsymbol{\theta}}_k|\boldsymbol{\theta}_k, \mathbf{X}^{k-1}, \mathbf{S}^{k-1}, \mathbf{S}_k)$, as required in (12), may not be easy to express, owing to the fact that, whilst each $\mathbf{X}_k$ deterministically yields a single value of $\hat{\boldsymbol{\theta}}_k$ given $\mathbf{X}^{k-1}$, $\mathbf{S}^{k-1}$ and $\mathbf{S}_k$ (i.e., from (9)) the reverse is not true. That is, there are potentially many unique $\mathbf{X}_k$'s that yield the *same* $\hat{\boldsymbol{\theta}}_k$. Thus some form of marginalisation over $\mathbf{X}_k$ would still be required, in addition to integration over $\hat{\boldsymbol{\theta}}_k$ and $\boldsymbol{\theta}_k$ as defined in (12), whereas in our preferred expression (11) integration over $\hat{\boldsymbol{\theta}}_k$ is performed using the sifting property of the delta function. This is especially relevant when considering the computational complexity of real problems, in which integration is likely to be performed numerically, and thus the analytic integration over one of the three terms, $\hat{\boldsymbol{\theta}}_k$, i.e., using the sifting property is a significant advantage of our preferred expression, (11).

IV. Implementation using Particle Filtering and Monte-Carlo Integration

As stated in Section III, calculation of the posterior PDF $p(\boldsymbol{\theta}_k|\mathbf{X}_k, \mathbf{X}^{k-1}, \mathbf{S}_k, \mathbf{S}^{k-1})$ may not be possible in algebraic form. Thus one may be forced to use an approximate filtering method, for example an extended Kalman filter, Unscented Kalman filter or PF. We focus solely on implementation using a PF, as this yields an approximation of the posterior PDF, which is later required in the waveform design, whereas the others do not, which is a compelling reason to favour the PF.

A. Estimation of state using a particle filter

In the PF, the particles correspond to realisations of the target parameter vector $\boldsymbol{\theta}_k$. Let N_P be the number of particles, the ith of which has a weight $w_k^{(i)}$. The particle filter enables the continuous PDF of $\boldsymbol{\theta}_k$ to be discretely approximated:

$$p(\boldsymbol{\theta}_k|\mathbf{X}^{k-1}, \mathbf{S}^{k-1}) \approx \sum_{i=1}^{N_P} w_k^{(i)} \delta(\boldsymbol{\theta}_k - \boldsymbol{\theta}_k^{(i)}). \quad (13)$$

This approximation allows us to express $\hat{\boldsymbol{\theta}}_k$, from (4):

$$\hat{\boldsymbol{\theta}}_k = \mathrm{g}_k(\mathbf{X}_k, \mathbf{S}_k, \boldsymbol{z}_{k-1}) \approx \frac{\sum_{i=1}^{N_P} w_k^{(i)} p(\mathbf{X}_k|\boldsymbol{\theta}_k^{(i)}, \mathbf{S}_k) \boldsymbol{\theta}_k^{(i)}}{\sum_{i=1}^{N_P} w_k^{(i)} p(\mathbf{X}_k|\boldsymbol{\theta}_k^{(i)}, \mathbf{S}_k)}. \quad (14)$$

Substituting (14) into (11), and converting the integral with respect to $\boldsymbol{\theta}_k$ into a summation accordingly yields:

$$\Sigma_k \approx \Sigma_k' = \int \sum_{j=1}^{N_P} \left(\frac{\sum_{i=1}^{N_P} w_k^{(i)} p(\mathbf{X}_k|\boldsymbol{\theta}_k^{(i)}, \mathbf{S}_k) \boldsymbol{\theta}_k^{(i)}}{\sum_{i=1}^{N_P} w_k^{(i)} p(\mathbf{X}_k|\boldsymbol{\theta}_k^{(i)}, \mathbf{S}_k)} - \boldsymbol{\theta}_k^{(j)} \right)^T \\ \times \left(\frac{\sum_{i=1}^{N_P} w_k^{(i)} p(\mathbf{X}_k|\boldsymbol{\theta}_k^{(i)}, \mathbf{S}_k) \boldsymbol{\theta}_k^{(i)}}{\sum_{i=1}^{N_P} w_k^{(i)} p(\mathbf{X}_k|\boldsymbol{\theta}_k^{(i)}, \mathbf{S}_k)} - \boldsymbol{\theta}_k^{(j)} \right) \\ \times w_k^{(j)} p(\mathbf{X}_k|\boldsymbol{\theta}_k^{(j)}, \mathbf{S}_k) \, \mathrm{d}\mathbf{X}_k. \quad (15)$$

For subsequent analysis, it is also necessary to explicitly define a variable, $\boldsymbol{\theta}_k'$, that has been drawn from the discrete approximation of the PDF:

$$\boldsymbol{\theta}_k' \sim \sum_{i=1}^{N_P} w_k^{(i)} \delta(\boldsymbol{\theta}_k' - \boldsymbol{\theta}_k^{(i)}) \quad (16)$$

B. Approximate cost function evaluation using Monte-Carlo integration

Observe that $p(\mathbf{X}_k|\boldsymbol{\theta}_k^{(j)}, \mathbf{S}_k)$, in (15), is Gaussian as defined in (1), although it should be noted that the mean varies for each term in the summation, owing to its reliance on $\mathbf{H}(\boldsymbol{\theta}_k^{(j)})$ (and likewise for $p(\mathbf{X}_k|\boldsymbol{\theta}_k^{(i)}, \mathbf{S}_k)$). In general, the integral over $\mathbf{X}_k$ in (15) may not be algebraically solvable, and thus Monte-Carlo (MC) integration can be used instead:

$$\Sigma_k' \approx \Sigma_k'' = \\ \sum_{m=1}^{N_S} \frac{p(\mathbf{X}_k^{(m)}|\boldsymbol{\theta}_k'^{(m)}, \mathbf{S}_k) / p(\mathbf{X}_k^{(m)}|\boldsymbol{\theta}_k'^{(m)}, \mathbf{S}_k(0))}{\sum_{m'=1}^{N_S} p(\mathbf{X}_k^{(m')}|\boldsymbol{\theta}_k'^{(m')}, \mathbf{S}_k) / p(\mathbf{X}_k^{(m')}|\boldsymbol{\theta}_k'^{(m')}, \mathbf{S}_k(0))} \\ \times \left(\frac{\sum_{i=1}^{N_P} w_k^{(i)} p(\mathbf{X}_k^{(m)}|\boldsymbol{\theta}_k^{(i)}, \mathbf{S}_k) \boldsymbol{\theta}_k^{(i)}}{\sum_{i=1}^{N_P} w_k^{(i)} p(\mathbf{X}_k^{(m)}|\boldsymbol{\theta}_k^{(i)}, \mathbf{S}_k)} - \boldsymbol{\theta}_k'^{(m)} \right)^T \\ \times \left(\frac{\sum_{i=1}^{N_P} w_k^{(i)} p(\mathbf{X}_k^{(m)}|\boldsymbol{\theta}_k^{(i)}, \mathbf{S}_k) \boldsymbol{\theta}_k^{(i)}}{\sum_{i=1}^{N_P} w_k^{(i)} p(\mathbf{X}_k^{(m)}|\boldsymbol{\theta}_k^{(i)}, \mathbf{S}_k)} - \boldsymbol{\theta}_k'^{(m)} \right), \quad (17)$$

$$\frac{\partial(\mathbf{u}_m^T\mathbf{u}_m)}{\partial s'_{k,n}} = 2\mathbf{u}_m^T \left(\frac{\sum_{i=1}^{N_P} w_k^{(i)} \frac{\partial p(\mathbf{x}_k'^{(m)}|\boldsymbol{\theta}_k^{(i)},\mathbf{s}'_k)}{\partial s'_{k,n}} \boldsymbol{\theta}_k^{(i)}}{\sum_{i=1}^{N_P} w_k^{(i)} p(\mathbf{x}_k'^{(m)}|\boldsymbol{\theta}_k^{(i)},\mathbf{s}'_k)} - \frac{\left(\sum_{i=1}^{N_P} w_k^{(i)} p(\mathbf{x}_k'^{(m)}|\boldsymbol{\theta}_k^{(i)},\mathbf{s}'_k)\boldsymbol{\theta}_k^{(i)}\right)\left(\sum_{i=1}^{N_P} w_k^{(i)} \frac{\partial p(\mathbf{x}_k'^{(m)}|\boldsymbol{\theta}_k^{(i)},\mathbf{s}'_k)}{\partial s'_{k,n}}\right)}{\left(\sum_{i=1}^{N_P} w_k^{(i)} p(\mathbf{x}_k'^{(m)}|\boldsymbol{\theta}_k^{(i)},\mathbf{s}'_k)\right)^2} \right) \tag{24}$$

where N_S samples of of $\boldsymbol{\theta}'_k$ are drawn from (16), and for each of which a corresponding sample of $\mathbf{X}_k$ is drawn, i.e., for the mth sample:

$$\begin{array}{rcl} \boldsymbol{\theta}_k'^{(m)} & \sim & \sum_{i=1}^{N_P} w_k^{(i)} \delta(\boldsymbol{\theta}_k'^{(m)} - \boldsymbol{\theta}_k^{(i)}) \\ \mathbf{X}_k^{(m)} & \sim & p(\mathbf{X}_k^{(m)}|\boldsymbol{\theta}_k'^{(m)}, \mathbf{S}_k(0)) \end{array}, \tag{18}$$

where $\mathbf{S}_k(0)$ is a fixed realisation of $\mathbf{S}_k$, for example, it could be the initial $\mathbf{S}_k$ prior to its variation through gradient descent (hence the nomenclature '$\mathbf{S}_k(0)$'). Accordingly, the first term of the summation in (17) corresponds to importance sampling, which is required to mitigate false convergence. To see why this importance sampling is required consider the converse, where a fresh sample of $\mathbf{X}_k^{(m)}$ is drawn from $p(\mathbf{X}_k^{(m)}|\boldsymbol{\theta}_k'^{(m)}, \mathbf{S}_k)$ at each new location on the cost function surface in the optimisation process. In this case, it is possible that (17) (dispensing with the first term in the summation, as it would no longer be necessary) could return a very low value of Σ'_k owing to the random sample of $\mathbf{X}_k^{(m)}$, rather than because $\mathbf{S}_k$ is locally optimal. This theoretical possibility of false convergence was confirmed as relevant in practise by the relatively poor performance of the waveform design method in the numerical simulations when implemented without this importance sampling.

Before progressing to demonstrate how gradient descent can be used to optimise this cost function, it is worth addressing the question of why the sampling defined in (18) is used for MC integration. In particular, an alternative would be to use the entirety of the set of samples of $\boldsymbol{\theta}_k^{(j)}$, in (15), and for each of which to sample $\mathbf{X}_k$ a number, N'_S, of times, however this would mean N'_S terms being summed for each $\boldsymbol{\theta}_k^{(j)}$, regardless of whether that sample represented a likely or unlikely realisation of the target parameters. By contrast, the approach expressed in (17) re-samples from the discrete approximation of the target parameter PDF (constructed by the PF) and thus heavily weighted (high probability) particles are likely to be drawn multiple times (so long as a suitable value of N_S is chosen) achieving the required diversity of samples of $\mathbf{X}_k$ at high probability values of $\boldsymbol{\theta}_k^{(j)}$, however low weight values of $\boldsymbol{\theta}_k^{(j)}$ are drawn less frequently, reducing the computational load that would otherwise arise from calculations which have little impact on the final result.

V. Optimisation of Cost Function using Gradient Descent

In general the cost function is not convex, which we were able to show through a counter-example (i.e., to the proposition that the cost function is convex, as given in Appendix A). Consequently efficient convex optimisation algorithms cannot be used, however our cost function is differentiable (as shown below) and thus we choose to use gradient descent to optimise our cost function. We also choose gradient descent as it is a generally accepted and well understood optimisation method, and is therefore suitable to prove the principle of the adaptive waveform design method proposed herein. That is not to say, however, that the cost function can be best optimised by gradient descent, as discussed later.

To express the gradient of the cost function, it is convenient to use vectorised forms of $\mathbf{S}_k$ and $\mathbf{X}_k$ split into real and imaginary components. We define $\mathbf{s}'_k \triangleq [\Re(\mathbf{s}_{k,1}); \Re(\mathbf{s}_{k,2}); \ldots; \Re(\mathbf{s}_{k,L}); \Im(\mathbf{s}_{k,1}); \Im(\mathbf{s}_{k,2}); \ldots; \Im(\mathbf{s}_{k,L})]$ and $\mathbf{x}'_k \triangleq [\Re(\mathbf{x}_{k,1}); \Re(\mathbf{x}_{k,2}); \ldots; \Re(\mathbf{x}_{k,L}); \Im(\mathbf{x}_{k,1}); \Im(\mathbf{x}_{k,2}); \ldots; \Im(\mathbf{x}_{k,L})]$. Furthermore, we define $\mathbf{H}''_k$ as L copies of $\mathbf{H}_k$ positioned on the leading diagonal of a larger matrix, whose other elements are all zero, and from which we define $\mathbf{H}'_k \triangleq [\Re(\mathbf{H}''_k), -\Im(\mathbf{H}''_k); \Im(\mathbf{H}''_k), \Re(\mathbf{H}''_k)]$. Note that here and below we drop the bracketed '$(\boldsymbol{\theta}_k)$' and simply use '$\mathbf{H}_k$', '$\mathbf{H}'_k$' and '$\mathbf{H}''_k$' for brevity.

Turning now to the optimisation of the cost function, we re-write (17):

$$\Sigma''_k(\mathbf{s}'_k) = \sum_{m=1}^{N_S} \frac{v_m}{\tilde{v}} \mathbf{u}_m^T \mathbf{u}_m, \tag{19}$$

where:

$$v_m = \frac{p(\mathbf{x}_k'^{(m)}|\boldsymbol{\theta}_k'^{(m)}, \mathbf{s}'_k)}{p(\mathbf{x}_k'^{(m)}|\boldsymbol{\theta}_k'^{(m)}, \mathbf{s}'_k(0))}, \tag{20}$$

$$\tilde{v} = \sum_{m'=1}^{N_S} \frac{p(\mathbf{x}_k'^{(m')}|\boldsymbol{\theta}_k'^{(m')}, \mathbf{s}'_k)}{p(\mathbf{x}_k'^{(m')}|\boldsymbol{\theta}_k'^{(m')}, \mathbf{s}'_k(0))}, \tag{21}$$

$$\mathbf{u}_m = \left(\frac{\sum_{i=1}^{N_P} w_k^{(i)} p(\mathbf{x}_k'^{(m)}|\boldsymbol{\theta}_k^{(i)}, \mathbf{s}'_k)\boldsymbol{\theta}_k^{(i)}}{\sum_{i=1}^{N_P} w_k^{(i)} p(\mathbf{x}_k'^{(m)}|\boldsymbol{\theta}_k^{(i)}, \mathbf{s}'_k)} - \boldsymbol{\theta}_k'^{(m)} \right), \tag{22}$$

which we use to express the gradient:

$$\nabla_{\mathbf{s}'_k}(\Sigma''_k) = \sum_{m=1}^{N_S} \frac{v_m}{\tilde{v}} \nabla_{\mathbf{s}'_k}(\mathbf{u}_m^T \mathbf{u}_m) + \frac{\tilde{v}\nabla_{\mathbf{s}'_k}(v_m) - v_m \nabla_{\mathbf{s}'_k}(\tilde{v})}{\tilde{v}^2} \mathbf{u}_m^T \mathbf{u}_m \tag{23}$$

where a single element of $\nabla_{\mathbf{s}'_k}(\mathbf{u}_m^T\mathbf{u}_m)$ is expressed in (24) (we express a single element to avoid ambiguity as $\mathbf{u}_m$ is already expressed in terms of the vector $\boldsymbol{\theta}_k'^{(m)}$), also:

$$\nabla_{\mathbf{s}'_k}(v_m) = \frac{\nabla_{\mathbf{s}'_k}(p(\mathbf{x}_k'^{(m)}|\boldsymbol{\theta}_k'^{(m)}, \mathbf{s}'_k))}{p(\mathbf{x}_k'^{(m)}|\boldsymbol{\theta}_k'^{(m)}, \mathbf{s}'_k(0))} \tag{25}$$

Algorithm 1 MMSE waveform design algorithm (bracketed numbers indicate the equation of the corresponding function).

$\texttt{Initialise: } k=1, \mathbf{s}'_0, \left\{\boldsymbol{\theta}_0^{(1)} \dots \boldsymbol{\theta}_0^{(N_P)}\right\}, \left\{w_0^{(1)} \dots w_0^{(N_P)}\right\}$
$\texttt{while } k < K\texttt{:}$
  $\texttt{transmit } \mathbf{s}'_{k-1}$
  $\texttt{receive } \mathbf{x}'_{k-1}$
  $\left[\left\{\boldsymbol{\theta}_k^{(1)} \dots \boldsymbol{\theta}_k^{(N_P)}\right\}, \left\{w_k^{(1)} \dots w_k^{(N_P)}\right\}\right] = \texttt{PF}\left(\mathbf{x}'_{k-1}, \mathbf{s}'_{k-1}, \left\{\boldsymbol{\theta}_{k-1}^{(1)} \dots \boldsymbol{\theta}_{k-1}^{(N_P)}\right\}, \left\{w_{k-1}^{(1)} \dots w_{k-1}^{(N_P)}\right\}\right)$
  $[\mathbf{s}'_k] = \texttt{Sdesign}\left(\left\{\boldsymbol{\theta}_k^{(1)} \dots \boldsymbol{\theta}_k^{(N_P)}\right\}, \left\{w_k^{(1)} \dots w_k^{(N_P)}\right\}\right)$
  $k = k+1$

$\texttt{function } [\mathbf{s}'_k] = \texttt{Sdesign}\left(\left\{\boldsymbol{\theta}_k^{(1)} \dots \boldsymbol{\theta}_k^{(N_P)}\right\}, \left\{w_k^{(1)} \dots w_k^{(N_P)}\right\}\right)$

$\texttt{Initialise: } \gamma, \gamma_{min}, \mathbf{s}'_k(0), \mathbf{s}'_k = \mathbf{s}'_k(0), \texttt{needdir} = \texttt{true}$
$\left[\left\{\boldsymbol{\theta}_k'^{(1)} \dots \boldsymbol{\theta}_k'^{(N_S)}\right\}, \left\{\mathbf{x}_k'^{(1)} \dots \mathbf{x}_k'^{(N_S)}\right\}\right] = \texttt{sample}\left(\left\{\boldsymbol{\theta}_k^{(1)} \dots \boldsymbol{\theta}_k^{(N_P)}\right\}, \left\{w_k^{(1)} \dots w_k^{(N_P)}\right\}, \mathbf{s}'_k\right)$ (18)
$\texttt{Obtain : px0} = \{p(\mathbf{x}_k'^{(1)}|\boldsymbol{\theta}_k'^{(1)}, \mathbf{s}'(0)) \dots p(\mathbf{x}_k'^{(N_S)}|\boldsymbol{\theta}_k'^{(N_S)}, \mathbf{s}'(0))\}$
$[\Sigma''_k] = \texttt{cost}\left(\left\{\boldsymbol{\theta}_k'^{(1)} \dots \boldsymbol{\theta}_k'^{(N_S)}\right\}, \left\{\mathbf{x}_k'^{(1)} \dots \mathbf{x}_k'^{(N_S)}\right\}, \left\{\boldsymbol{\theta}_k^{(1)} \dots \boldsymbol{\theta}_k^{(N_P)}\right\}, \left\{w_k^{(1)} \dots w_k^{(N_P)}\right\}, \mathbf{s}'_k\right)$ (17)
$\texttt{while } \gamma \geq \gamma_{min}\texttt{ :}$
  $\texttt{if needdir:}$
    $[\nabla(\Sigma''_k)] = \texttt{dcost}\left(\left\{\boldsymbol{\theta}_k'^{(1)} \dots \boldsymbol{\theta}_k'^{(N_S)}\right\}, \left\{\mathbf{x}_k'^{(1)} \dots \mathbf{x}_k'^{(N_S)}\right\}, \left\{\boldsymbol{\theta}_k^{(1)} \dots \boldsymbol{\theta}_k^{(N_P)}\right\}, \left\{w_k^{(1)} \dots w_k^{(N_P)}\right\}, \mathbf{s}'_k, \text{px0}\right)$ (23)
  $\tilde{\mathbf{s}}'_k = \mathbf{s}'_k - \gamma \nabla^{\perp}_{\mathbf{s}'_k}(\Sigma''_k)$
  $\tilde{\mathbf{s}}'_k = \tilde{\mathbf{s}}'_k \sqrt{\text{PL}} / \sqrt{\tilde{\mathbf{s}}_k'^T \tilde{\mathbf{s}}'_k}$
  $\left[\tilde{\Sigma}''_k\right] = \texttt{cost}\left(\left\{\boldsymbol{\theta}_k'^{(1)} \dots \boldsymbol{\theta}_k'^{(N_S)}\right\}, \left\{\mathbf{x}_k'^{(1)} \dots \mathbf{x}_k'^{(N_S)}\right\}, \left\{\boldsymbol{\theta}_k^{(1)} \dots \boldsymbol{\theta}_k^{(N_P)}\right\}, \left\{w_k^{(1)} \dots w_k^{(N_P)}\right\}, \tilde{\mathbf{s}}'_k, \text{px0}\right)$ (17)
  $\texttt{if } \tilde{\Sigma}''_k < \Sigma''_k\texttt{:}$
    $\mathbf{s}'_k = \tilde{\mathbf{s}}'_k$
    $\boldsymbol{\Sigma}''_k = \tilde{\boldsymbol{\Sigma}}''_k$
    $\texttt{needdir} = \texttt{true}$
  $\texttt{else:}$
    $\gamma = \gamma/2$
    $\texttt{needdir} = \texttt{false}$

and

$$\nabla_{\mathbf{s}'_k}(\tilde{v}) = \sum_{m'=1}^{N_S} \frac{\nabla_{\mathbf{s}'_k}(p(\mathbf{x}_k'^{(m')}|\boldsymbol{\theta}_k'^{(m')}, \mathbf{s}'_k))}{p(\mathbf{x}_k'^{(m')}|\boldsymbol{\theta}_k'^{(m')}, \mathbf{s}'_k(0))}, \tag{26}$$

Finally, we express $p(\mathbf{x}'_k|\boldsymbol{\theta}_k, \mathbf{s}'_k)$, which is a circularly symmetric complex Gaussian PDF:

$$p(\mathbf{x}'_k|\boldsymbol{\theta}_k, \mathbf{s}'_k) = \frac{\exp(-(\mathbf{x}'_k - \mathbf{H}'_k\mathbf{s}'_k)^T \mathbf{R}_n^{-1}(\mathbf{x}'_k - \mathbf{H}'_k\mathbf{s}'_k))}{\sqrt{(2\pi)^{2N_R}\det(R_n)}}, \tag{27}$$

where $\det(.)$ is the determinant and $\mathbf{R}_n$ is the covariance of the noise, i.e.,

$$\mathbf{R}_n = \sigma_n^2 \mathbf{I}_{2N_R L}, \tag{28}$$

where $\mathbf{I}_{2N_R L}$ is the identity matrix of size $2N_R L$. From (27) we can express the gradient of $p(\mathbf{x}'_k|\boldsymbol{\theta}_k, \mathbf{s}'_k)$ with respect to $\mathbf{s}'_k$, as required in (25) and (26):

$$\begin{aligned}\nabla_{\mathbf{s}'_k}&(p(\mathbf{x}'_k|\boldsymbol{\theta}_k, \mathbf{s}'_k)) \\ &= 2p(\mathbf{x}'_k|\boldsymbol{\theta}_k, \mathbf{s}'_k)\left(\mathbf{H}_k'^T \mathbf{R}_n^{-1}\mathbf{x}'_k - \mathbf{H}_k'^T \mathbf{R}_n^{-1}\mathbf{H}'_k\mathbf{s}'_k\right).\end{aligned} \tag{29}$$

It is also necessary to take into account the maximum power constraint, defined in (6). The simplest way to do this is to assume that this maximum power constraint is always satisfied with equality, and thus the valid region for $\mathbf{s}'_k$ is a hyper-sphere. Thus, as long as our starting point is on the surface of this hyper-sphere, we can take the direction of descent as the directional derivative tangential to the surface of the hypersphere, and re-normalise after taking the step accordingly. Thus we express the component of $\nabla_{\mathbf{s}'_k}(\Sigma''_k)$ perpendicular to $\mathbf{s}'_k$:

$$\nabla^{\perp}_{\mathbf{s}'_k}(\Sigma''_k) = \nabla_{\mathbf{s}'_k}(\Sigma''_k) - \mathbf{s}'_k \frac{(\nabla_{\mathbf{s}'_k}(\Sigma''_k))^T \mathbf{s}'_k}{\mathbf{s}_k'^T \mathbf{s}'_k}, \tag{30}$$

which is the direction of maximum gradient, given the power constraint. A similar approach was taken by Ahmed *et al* to solve a related problem, with more restrictive power constraints [24].

Algorithm 1 defines the gradient descent algorithm for MMSE waveform design according to this analysis.

VI. Numerical Example

We consider a MIMO radar system for our numerical example, a choice motivated not only by the fact that it provides a simple, clear and readily understandable demonstration of the principle of our method, but also because MIMO radar is one of the principal applications for MMSE adaptive waveform design. In particular, we consider a MIMO radar consisting of co-located linear transmit and receive arrays, each with five elements (i.e., $N_T = N_R = 5$) where the element spacing is equal to half of the wavelength.

Accordingly, we express our MIMO radar in the standard form [25]: the parameter space consists of the parameters associated with a fixed and known number of targets, Q', treated as point scatterers, thus we define $\boldsymbol{\theta}_k = [\boldsymbol{\phi}, |\boldsymbol{\alpha}|, \arg(\boldsymbol{\alpha})]$, where

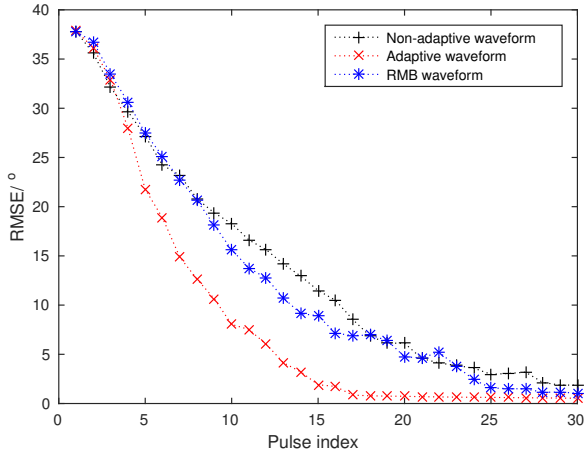

Fig. 1. RMSE for adaptive and orthogonal waveforms estimating the angle of one target, averaged over 500 trials.

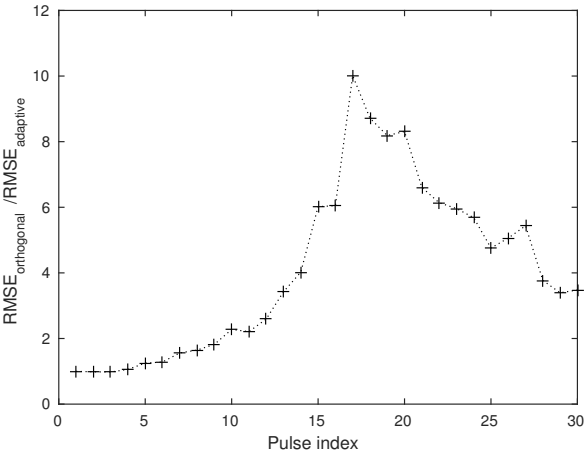

Fig. 2. Ratio of orthogonal waveform RMSE to adaptive waveform RMSE for one target.

$\boldsymbol{\phi}$ is a row vector of size Q' containing the angle of the targets, and $\boldsymbol{\alpha}$ is a row vector of size Q' containing the complex attenuation of the targets. Although our method applies to the case where $\boldsymbol{\theta}_k$ varies for each pulse, here we consider the case where $\boldsymbol{\theta}_k$ does not vary with k. This is because fixing k makes it easier to visualise the operation of the adaptive waveform design, i.e., as shown in the following results. To reduce the computational load, we make the assumption that $\boldsymbol{\alpha}$ is known. Even though such an assumption is not consistent with a real MIMO radar system, it does not negate the ability of the numerical example to demonstrate the principle of the proposed MMSE adaptive waveform design method.

These definitions enable us to express the MIMO radar in the form of (1):

$$\mathbf{X}_k = \sum_{q=1}^{Q'} \alpha_q \mathbf{a}_R(\phi_q)\mathbf{a}_T^T(\phi_q)\mathbf{S}_k + \mathbf{N}_k, \tag{31}$$

where $\mathbf{H}_k \triangleq \sum_{q=1}^{Q'} \alpha_q \mathbf{a}_R(\phi_q)\mathbf{a}_T^T(\phi_q)$, and $\mathbf{a}_R \in \mathbb{C}^{N_R \times 1}$ and $\mathbf{a}_T \in \mathbb{C}^{N_T \times 1}$ are the steering vectors associated with the receive and transmit arrays respectively. Note that for simplicity (without loss of generality) we've removed the Doppler shift and time delay.

A. Angle estimation for one target

As our intention is to demonstrate adaptive waveform design, rather than the operation of PFs in radar, we have chosen a simple PF implementation. Specifically, we consider the case where there is a single target, for which the radar must estimate the angle, and we initialise $N_P = 180$ equally spaced, equally weighted particles corresponding to target location at one degree intervals from -90^o to 89^o. As the target angle is known to be un-varying, there is no PF particle location updating and re-sampling, in the conventional sense, however we delete particles whose weight falls below 1% of their original value and re-normalise accordingly.

For our numerical example, we let $L = 1$, i.e., the case where waveform re-design happens for each transmitted snapshot, we also consider array signal to noise ratio (ASNR) of -3 dB, where ASNR $\triangleq |\alpha|^2 \mathrm{P} N_R L/(0.5\sigma_n^2)$ (where the factor 0.5 in the denominator is introduced owing to our definition of σ_n^2 as the noise variance for each of the real and imaginary components). Finally, we set $N_S = 250$, and position the target at -40^o.

To obtain a numerical value for the root mean squared error (RMSE) we average over 500 trials, as shown in Fig. 1. For comparison we also show the same plot for a non-adaptive orthogonal waveform (i.e., the transmitted waveform has uniform anglular spread). The orthogonal waveform in question consisted of simply transmitting all of the power from the first transmit array element, however additional results not included here showed that all orthogonal waveforms have the same RMSE performance (including where $L > 1$). In Fig. 1 we also include the same plot for our implementation of the Reuven-Messer bound (RMB) method proposed by Huleihel *et al* [4], denoted 'RMB' (we consider the RMB method as this is the better performing of the two methods proposed in [4]). To implement the method of Huleihel *et al* in a manner fit for fair comparison, we supplied the same resources as for the method proposed herein (i.e., the same PF particles) and chose the other parameters such that the simulation could be performed within a reasonable time-scale. It was necessary to set $L = 5$, owing to the workings of the RMB method in [4], however the ASNR was not changed, and thus the comparison presented is a valid one.

From Fig. 1, we can clearly see that our proposed adaptive waveform method outperforms the non-adaptive case, and also that it outperforms the RMB method. Nevertheless, we note that this *is* our implementation of the RMB method for our application, and thus in general the RMB method may remain appropriate for some applications. For this reason we provide the code for our method and our implementation of that of Huleihel *et al*, to facilitate future comparison between the two [26]. On a more general note, it is significant that both methods require non-convex optimisation, however the cost function we provide, (11), is the exact MMSE cost function whereas that of Huleihel *et al* is only approximate, therefore with sufficient

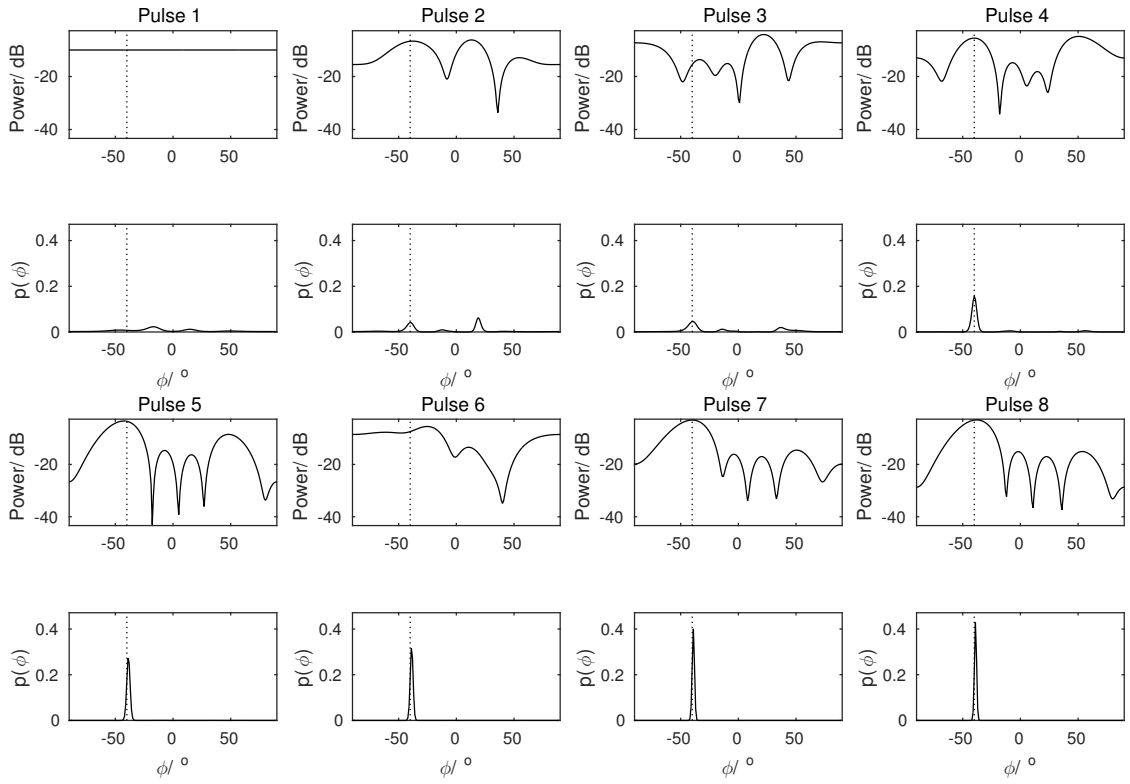

Fig. 3. Example of the first eight pulses of MMSE adaptive waveform design. Single target located at -40^o as indicated by the vertical dotted line.

computational resources (and using a more sophisticated non-convex optimisation technique) the method proposed herein can theoretically be used to find the global MMSE waveform design and thus will outperform any other including that of Huleihel *et al* [4].

Returning to the specific numerical example considered here, the aim of our adaptive waveform design is to locate the target within a shorter time than that of a non-adaptive waveform. To this end, we express the results of Fig. 1 as the ratio of the orthogonal waveform RMSE to the adaptive waveform RMSE in Fig. 2, which shows that at its peak the orthogonal waveform has an RMSE (which can be interpreted as a numerical approximation of the standard deviation) of approximately one tenth of that of the orthogonal, non-adaptive, waveform. Thus conclusively demonstrating the success of our MMSE adaptive waveform design method.

Turning our attention back to Fig. 1, we can see that between pulse 5 and pulse 10 there is a relatively large decrease in RMSE at each pulse for the adaptive waveform compared to that of the orthogonal waveform. This suggests that, on average, the adaptive method has located the target and is now decreasing the estimate variance, whereas the orthogonal method may still have a range of possible target locations.

In addition to Fig. 1 and Fig. 2, we include Fig. 3 to show how the waveform shape is adapted for the first eight pulses of one trial. Note that in Fig. 3 the transmit power shown on the y axis is defined as $(1/L)\mathbf{a}_T^H(\phi)(\mathbf{S}_k^*\mathbf{S}_k^T)\mathbf{a}_T(\phi)$. This further supports the above conclusion regarding the relatively large decrease in RMSE between pulse 5 and 10, as it can be seen that $p(\phi)$ is substantially uni-modal around -40^o from pulse 5 onwards. For higher values of pulse index in Fig. 1 the adaptive and orthogonal waveform RMSE plots come back together, owing to the fundamental limitation placed on the target location estimation by the finite number of particles in the PF.

Regarding Fig. 3, there are a few interesting features in additional to already identified the uni-modal nature of $p(\phi)$ from pulse 5 onwards. For example, we can see that the received measurement after pulse 3 leads to peaks of $p(\phi)$ at around -40^o, and 40^o, which in turn leads to corresponding peaks in the waveform design for pulse 4. However, the measurement after pulse 4 leads to a much lower peak of $p(\phi)$ at around 40^o compared to that around -40^o, and the design of the waveform for pulse 5 is adapted accordingly. We can see that this behaviour of adapting the waveform to transmit the majority of the power in the direction(s) of high probability of target location is typical of the behaviour of the MMSE adaptive waveform design method, as would be expected.

However, whilst it is reassuring that these features align with our intuitive expectation for how the adaptive waveform

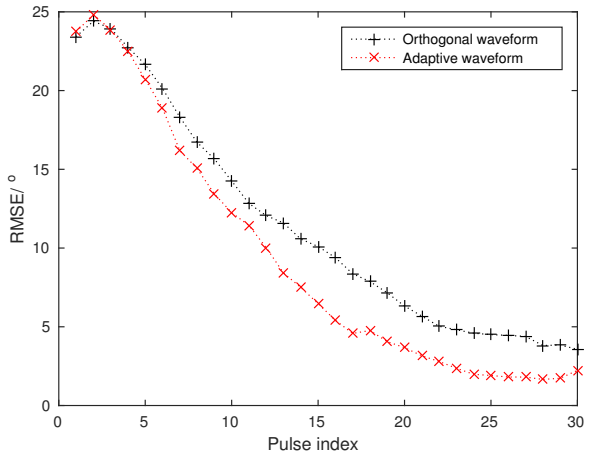

Fig. 4. RMSE for adaptive and orthogonal waveforms estimating the angle of two targets, averaged over 1000 trials.

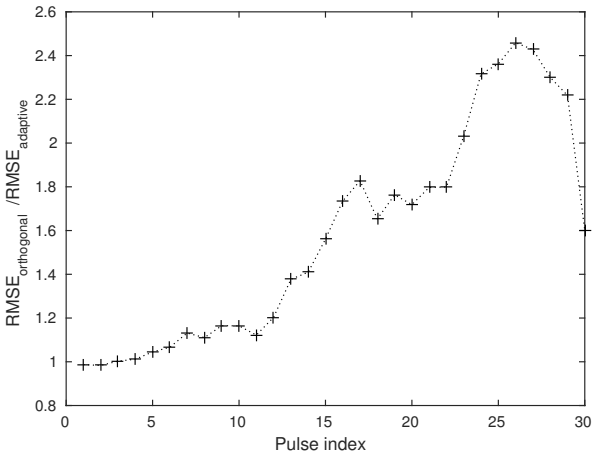

Fig. 5. Ratio of orthogonal waveform RMSE to adaptive waveform RMSE, for two targets.

design ought to behave, we acknowledge that this is not exclusively the case. In particular, it is important to recognise that each trial has independent simulated noise, and so will have a different received signal $\mathbf{X}_k$, and consequently the designed waveform will differ from trial to trial. In particular, we note that the local nature of the optimisation sometimes leads to adaptive waveform designs that we do immediately recognise as obviously good, for example pulse 6 in Fig. 3. The specific waveform design presented in this paper (i.e., in Fig. 3) has been chosen as a good illustration of the designed waveform, and typical of those that we have seen throughout our simulations. It is also worth highlighting that, although we only present this up to the eighth pulse, we continued to observe a similar waveform shaping for latter pulses. Finally, this waveform shaping still occurs when the SNR is varied, however at lower SNR it takes a larger number of pulses for the target position PDF to become substantially non-uniform, and therefore it takes a larger number of pulses for waveforms with significantly non-uniform angular spread to be designed.

B. Angle estimation for two targets

In addition to the one target scenario, we also consider a scenario where the MIMO radar must estimate the angle of two targets. The ASNR was adjusted to be 0 dB, equally split between the two targets, i.e., the attenuation of each target is equal to that of the lone target in the first simulations. Additionally, the number of particles in the PF was increased to cover the joint distribution of the two target angles (i.e., the parameter space), and again the particles were placed on a grid, this time with spacing of 3^o (the reduction in resolution from 1^o to 3^o the innevitable result of requiring many more particles to jointly estimate the locations of two targets). Again, we used $N_S = 250$ and the targets were located at -40^o and 20^o.

The results of the two target simulation are shown in Figs. 4, 5 and 6 for the RMSE plot, RMSE ratio and single trial example respectively (again chosen as it is a good illustration of the waveform design, and representative of the general behaviour we observed). To get reasonable results, it was necessary to increase the number of trials from the 500 used for the single target case to 1000. As for the single target case, we can see in Fig. 4 that the adaptive waveform outperforms the non-adaptive orthogonal waveform. Even though the relative improvement of the adaptive waveform design (in terms of RMSE) is not as great as for the single target case, we can see from Fig. 5 that at its peak, the RMSE is for the adaptive waveform is less than a half that of the non-adaptive waveform, which would represent a significant gain in performance if applied in an actual MIMO radar system.

We also observe similar behaviour to the one target case for the single trial of the two target simulation shown in Fig. 6, where the black solid plot corresponds to the probability of the target with lower value of ϕ, and the red dashed plot corresponds to the probability of the target with the higher value of ϕ. That is, from pulse 4 onwards, two peaks occur near the actual target angles, and again the power is steered predominantly towards the directions of high probability density.

VII. Computational Complexity

TABLE I
Computational Complexity

Equation	Number of operations
(18)	$\mathcal{O}(N_S)$
(17)	$\mathcal{O}(N_c N_S N_p (Q + LN_T N_R))$
(23)	$\mathcal{O}(N_d N_S N_p (LN_T Q + L^2 N_T^2 N_R))$

As well as demonstrating the principle of our MMSE adaptive waveform design method using a numerical example, it is also of practical concern to establish the computational load of the method. To this end, we express in Table I the contribution of the evaluation of the various functions in Algorithm 1 to the overall computational complexity. This is expressed in terms of how the number of floating point operations grows with each of the degrees of freedom. By definition, N_c is the number of times the cost function is evaluated, and N_d is the number of times the gradient of the cost function is evaluated. Note

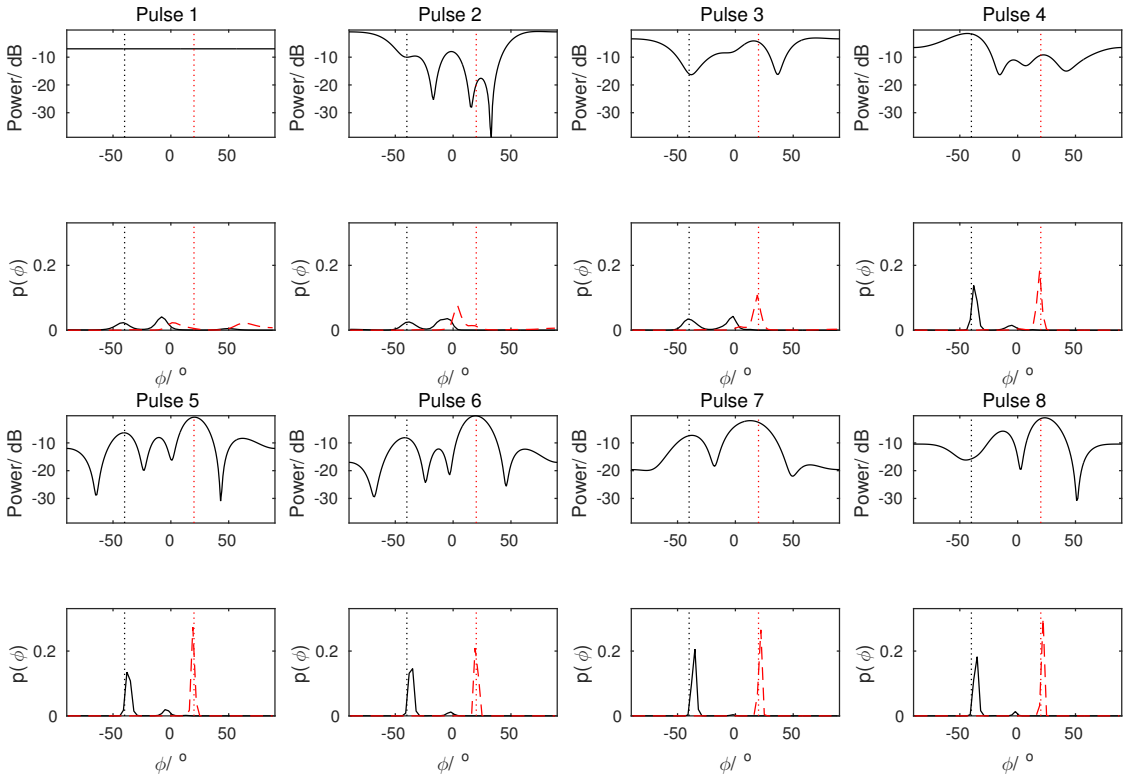

Fig. 6. Example of the first eight pulses of MMSE adaptive waveform design. Two targets located at -40^o and 20^o as indicated by the vertical dotted lines.

that for the simple implementation of gradient descent given in Algorithm 1 it may be possible to combine these two as they can differ only by a constant (i.e., the number of times the step size can be decreased) however it is worth leaving them as independent as this facilitates later discussion on the complexity of more sophisticated implementations, and indeed other forms of optimisation.

Examining the three functions in turn, (18) simply requires $\mathcal{O}(N_S)$ samples to be drawn, hence its complexity is $\mathcal{O}(N_S)$. Regarding (17), this consists of N_S evaluations of a sum which is dominated (in terms of computational complexity) by the transpose of a Q element column vector multiplied by itself, hence $\mathcal{O}(N_S Q)$ operations. For each of the N_S terms of the sum, other sums are required, each with N_P terms, and each term of which requires the evaluation of a multivariate Gaussian PDF with N_R elements. As each element of this multivariate Gaussian is independent, there is no need to invert a covariance matrix, and instead it can be treated as the product of N_R univariate Gaussians. To do so, however, still requires the evaluation of $(\mathbf{x}'_k - \mathbf{H}'_k\mathbf{s}'_k)$ in the exponent of the multivariate Gaussian, in order to separate it into its elements. Construction of $\mathbf{H}_k$ requires $N_T \times N_R$ multiplications (from which the artificial construct, $\mathbf{H}'_k$, used only to simplify the expressions in the analysis derives). Calculation of $\mathbf{H}'_k\mathbf{s}'_k$ given $\mathbf{H}_k$ requires $\mathcal{O}(N_T N_R)$ multiplications plus $\mathcal{O}(N_T N_R)$ additions, repeated L times. Hence the evaluation of $(\mathbf{x}'_k - \mathbf{H}'_k\mathbf{s}'_k)$ requires $\mathcal{O}(N_T N_R L)$ which is greater than the $\mathcal{O}(N_T N_R)$ required to construct $\mathbf{H}_k$, and the $\mathcal{O}(N_R)$ multiplications required to obtain the joint probability of the independent components. Finally, for each term in the sum over N_P particles, a further Q multiplications are required to scale the vector $\boldsymbol{\theta}_k^{(j)}$ (i.e., in the sum in the numerator only). This dominates the Q multiplications previously stated. Including the N_c times that the cost function is evaluated, and putting this together with the above yields overall complexity, $O(N_c(N_S N_P Q + N_S N_P N_T N_R L))$, equivalent to that stated in Table I.

Turning our attention now to $\nabla_{\mathbf{s}''_k}(\Sigma'(\mathbf{s}''_k))$, as defined in (23). Each of the $2N_T L$ elements in the vector equation (23) is a sum over N_S terms. Using the same analysis as that above, it can be shown that evaluation of each of the component terms has the following complexity: v_m requires $\mathcal{O}(LN_T N_R)$ operations; $\tilde{v}$ requires $\mathcal{O}(N_S LN_T N_R)$ operations (but only requires evaluation once, rather than for each term in the sum, and thus the extra N_S is the complexity later cancels); $\nabla_{\mathbf{s}''_{k,n}}(v_m)$ requires $\mathcal{O}(LN_T N_R)$ operations for each of its $2N_T L$ elements; $\nabla_{\mathbf{s}''_{k,n}}(\tilde{v})$ requires $\mathcal{O}(N_S LN_T N_R)$ operations for each of its $2N_T L$ elements (but again only needs evaluating once, and not for each of the N_S terms in the sum in (23)); $\mathbf{u}_m^T\mathbf{u}_m$ requires $\mathcal{O}(N_P(Q + LN_T N_R))$ operations; and $\partial(\mathbf{u}_m^T\mathbf{u}_m)/\partial\mathbf{s}''_{k,n}$ requires $\mathcal{O}(N_P(Q + LN_T N_R))$ operations. Putting these together, the number of floating

point operations required for the numerical evaluation of (23) is $\mathcal{O}(N_d N_S N_P (LN_T Q + L^2 N_T^2 N_R))$. Table I includes this result, multiplied by the number of times that (23) is called, defined as N_d.

The number of floating point operations required by the algorithm is therefore $\mathcal{O}(N_c N_S N_p (Q + LN_T N_R) + N_d N_S N_p (LN_T Q + L^2 N_T^2 N_R))$. In most cases we expect the complexity to be dominated by the evaluation of the gradient, simplifying and rewriting the complexity accordingly as $\mathcal{O}((N_d)(N_S N_P)(N_T L)(Q + LN_T N_R))$ reveals that the algorithm is, in some sense, computationally efficient. Taking the bracketed terms in turn, the first, N_d, corresponds to the iterations of the gradient descent, and the second, $N_S N_P$, corresponds to the double integration, that we previously argued to be irreducible, evaluated numerically as a double sum. The third term, $N_T L$, corresponds to the $N_T L$ elements of $\mathbf{S}_k$ that must be optimised for, i.e., the number of elements of the gradient that must be evaluated at each iteration, leaving the fourth term, $Q + LN_T N_R$, corresponding to the evaluation of the gradient itself. From this analysis, it can be seen that the evaluation of the gradient grows only proportionally to each of the degrees of freedom of the system, hence making it a scalable algorithm.

We do, however, observe that the complexity associated with evaluation of the gradient of the cost function is relatively large compared to evaluation of the cost function itself, which raises two further points. Firstly, it can be seen that a smarter adaptive step-size technique may be beneficial rather than the simple example given in Fig. 1. Secondly, along similar lines, it is of interest to explore whether a meta-heuristic optimisation, such as *Particle Swarm Optimisation* [27], which does not require evaluation of the gradient of the cost function at all, can optimise the cost function equally well or even better at lower computational complexity.

Owing to the number of degrees of freedom, and in particular their lack of correspondence to those of other methods, it is not possible to make general statements about the absolute advantage of our algorithm in terms of computational complexity. It is, however, possible to make a comparison between our embodiment of the adaptive waveform method proposed in this paper, and our embodiment of the method proposed by Huleihel *et al* [4], in terms of computation time. For this we performed a single run of each method using *Matlab* on an Intel Core 1.9 GHz i3 4030U processor with memory of 4 GB 1600 MHz DDR3L SSDRAM (note that it is not possible to provide an average computation time for the whole ensemble of trials for each method, as this was computed on a distributed system). For comparison we also show the computation time for the non-adaptive case, where just the computation required for updating the PF is required. Fig. 7 shows the computation times for the three methods.

From Fig. 7, we can see that both the adaptive waveform proposed in this paper, and the RMB waveform of Huleihel *et al* [4] have significantly longer computational time than the non-adaptive case, as would be expected. It should be noted that the absolute magnitude of these computation times is not relevant, as we do not specify a system upon which the MIMO radar processing would run, instead it is the relative

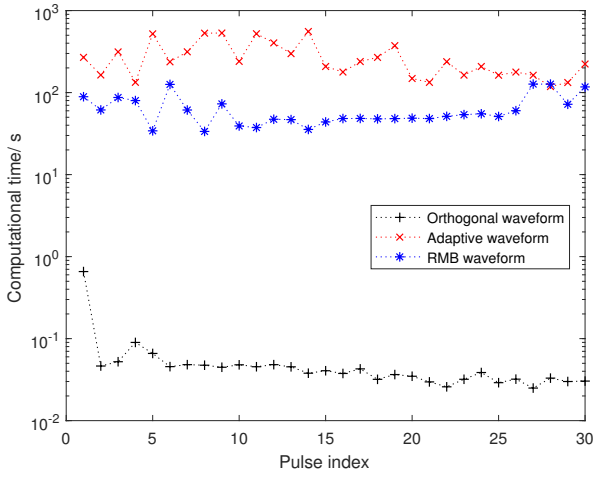

Fig. 7. Computational time for the three methods.

performance that is important. To this end, it is pertinent to note that the code used for the two adaptive design methods was not optimised at all, and a reduction in computation time of at least an order of magnitude should be achievable. Nevertheless, it is undeniably the case that implementing either of these adaptive waveform design methods is computationally expensive compared to simply estimating the target parameters, and this provides motivation for a future investigation into whether MMSE adaptive waveform design can be achieved, or approximately achieved using a method that does not recursively sum over a large number of particles, as do both the method proposed herein and that of Huleihel *et al*.

Considering now a specific comparison between these two adaptive waveform design methods, from Fig. 7 we can see that the computation times for the two are broadly comparable, which is to be expected given that it was our declared intention to implement the method of Huleihel *et al* in a manner which provides a fair comparison to our method. Indeed, we chose the parameter value $J = 10$ (as defined in [4]) as this appeared to require a similar computational time for cost function evaluation as does the method we propose in this paper, with the parameters as specified. Nevertheless, it is the case that the mean computational time of the RMB method is approximately 1/4 of that for the method we propose, and this may mean that the RMB method is preferable in some circumstances, even though it performs relatively poorly.

VIII. Conclusions

In this paper we have derived a general cost function for MMSE adaptive waveform design. This has been expressed as a double integral over the parameter space and the space containing the forthcoming measurement. We argue that there is an irreducible need for a double integral, and moreover it is only because we can express an intermediate step in the derivation as an integral over a sum of Dirac delta functions and hence apply the sifting property that the need for integration over three spaces is negated. Thus we conclude that our cost function expression is in a suitably simple form.

Noting that the double integration cannot, in general, be solved analytically, and that the underlying state is likely to be estimated by a PF, we also provide analysis leading to a method for applying the MMSE cost function using particle filtering, MC integration and optimisation by gradient descent (noting that, in general, the cost function is not convex). We provide analysis for the computational complexity of this method, in terms of how the number of floating point operations grows with the value of the various operating parameters.

In addition to these theoretical contributions, we also provide a numerical example, based on a simplified model for MIMO radar. The numerical example shows that our MMSE adaptive waveform method outperforms the non-adaptive case where a waveform with uniform angular spread is transmitted, thus demonstrating the principle of our analysis. Moreover, MIMO radar is known to be one of the foremost applications of adaptive waveform design, and the MIMO radar MMSE waveform design method implementation described herein may suffice for some applications. However, we do recognise that gradient descent, as used in the implementation of our method described herein may not be sufficient to perform the required non-convex optimisation of the cost function in all cases. Furthermore, our computational complexity analysis demonstrates that the computational load associated with calculation of the gradient is relatively heavy, compared to that associated with evaluated of the cost function itself. Thus it is of future interest to investigate whether the theoretical cost function proposed in this paper can be better optimised for various real world applications by alternative optimisation methods.

APPENDIX A
A COUNTER EXAMPLE TO THE PROPOSITION THAT THE COST FUNCTION IS CONVEX

To show that the optimisation function is not in *general* convex, it suffices to find a single example of non-convexity. Specifically, we will show a case where the cost function is not convex along a specified (linear) direction as a counter-example to the proposition that the cost function is convex in general.

To do this we consider the one-target scenario and set $\mathbf{S}_0 = [0.3166, 0, 0, 0, 0]^T$, and randomly choose a direction to step in: $\Delta\mathbf{S} = [0.0273 - 0.1349i, 0.1673 - 0.2676i, -0.0721 - 0.0815i, 0.3774 + 0.3043i, -0.0988 + 0.1351i]^T$, where $i = \sqrt{-1}$. Thus, to demonstrate non-convexity, we show that the cost function in the direction of $-\Delta\mathbf{S}$ is not convex. That is, we evaluate $\Sigma_0''(\mathbf{S}_0 - a\Delta\mathbf{S}_0)$ according to (17) for $a \in \mathbb{Z}_{\geq 0}$. This yields the result plotted in Fig. 8 which demonstrates that the cost function surface is indeed not, in general, convex.

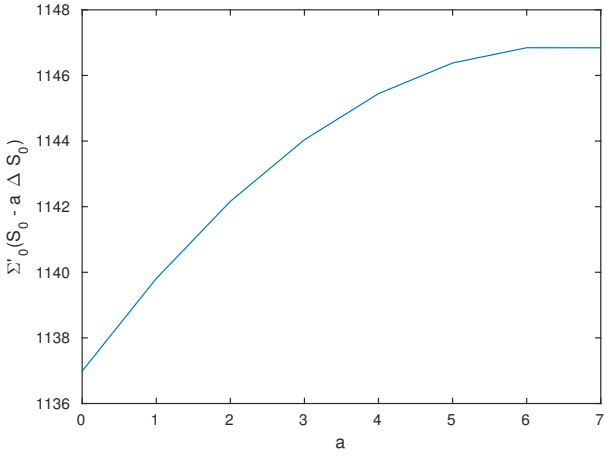

Fig. 8. Special case to show that the optimisation surface is not convex

REFERENCES

[1] S. Haykin, "Cognitive radar: a way of the future," *IEEE Signal Processing Magazine*, vol. 23, no. 1, pp. 30–40, Jan 2006.

[2] J. Ender and S. Brüggenwirth, "Cognitive radar - enabling techniques for next generation radar systems," in *2015 16th International Radar Symposium (IRS)*, June 2015, pp. 3–12.

[3] S. P. Sira, Y. Li, A. Papandreou-Suppappola, D. Morrell, D. Cochran, and M. Rangaswamy, "Waveform-agile sensing for tracking," *IEEE Signal Processing Magazine*, vol. 26, no. 1, pp. 53–64, Jan 2009.

[4] W. Huleihel, J. Tabrikian, and R. Shavit, "Optimal adaptive waveform design for cognitive MIMO radar," *IEEE Transactions on Signal Processing*, vol. 61, no. 20, pp. 5075–5089, Oct 2013.

[5] Y. Li, W. Moran, S. P. Sira, A. Papandreou-Suppappola, and D. Morrell, "Adaptive waveform design in rapidly-varying radar scenes," in *2009 International Waveform Diversity and Design Conference*, Feb 2009, pp. 263–267.

[6] M. R. Bell, "Information theory and radar waveform design," *IEEE Transactions on Information Theory*, vol. 39, no. 5, pp. 1578–1597, Sep 1993.

[7] Y. Jin, J. M. F. Moura, and N. O'Donoughue, "Time reversal in multiple-input multiple-output radar," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 1, pp. 210–225, Feb 2010.

[8] S. Sen and A. Nehorai, "OFDM MIMO radar with mutual-information waveform design for low-grazing angle tracking," *IEEE Transactions on Signal Processing*, vol. 58, no. 6, pp. 3152–3162, June 2010.

[9] A. Turlapaty and Y. Jin, "Bayesian sequential parameter estimation by cognitive radar with multiantenna arrays," *IEEE Transactions on Signal Processing*, vol. 63, no. 4, pp. 974–987, Feb 2015.

[10] R. A. Romero and N. A. Goodman, "Waveform design in signal-dependent interference and application to target recognition with multiple transmissions," *IET Radar, Sonar Navigation*, vol. 3, no. 4, pp. 328–340, August 2009.

[11] G. Rossetti, A. Deligiannis, and S. Lambotharan, "Waveform design and receiver filter optimization for multistatic cognitive radar," in *2016 IEEE Radar Conference (RadarConf)*, May 2016, pp. 1–5.

[12] G. Rossetti and S. Lambotharan, "Waveform optimization techniques for bi-static cognitive radars," in *2016 IEEE 12th International Colloquium on Signal Processing Its Applications (CSPA)*, March 2016, pp. 115–118.

[13] ——, "Coordinated waveform design and receiver filter optimization for cognitive radar networks," in *2016 IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM)*, July 2016, pp. 1–5.

[14] Y. Yang and R. S. Blum, "MIMO radar waveform design based on mutual information and minimum mean-square error estimation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 1, pp. 330–343, January 2007.

[15] ——, "Radar waveform design using minimum mean-square error and mutual information," in *Fourth IEEE Workshop on Sensor Array and Multichannel Processing, 2006.*, July 2006, pp. 234–238.

[16] ——, "Minimax robust MIMO radar waveform design," *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, no. 1, pp. 147–155, June 2007.

[17] D. Guo, S. Shamai, and S. Verdu, "Mutual information and minimum mean-square error in gaussian channels," *IEEE Transactions on Information Theory*, vol. 51, no. 4, pp. 1261–1282, April 2005.

[18] N. Sharaga, J. Tabrikian, and H. Messer, "Optimal cognitive beamforming for target tracking in MIMO radar/sonar," *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, no. 8, pp. 1440–1450, Dec 2015.

[19] M. C. Wicks, K. M. Magde, W. M. Moore, and J. D. Norgard, "Adaptive multi-sensor waveform design for RF tomographic sensors," in *2007 International Conference on Electromagnetics in Advanced Applications*, Sept 2007, pp. 423–426.
[20] F. Gini, A. D. Maio, and L. Patton, Eds., *Waveform Design and Diversity for Advanced Radar Systems*, ser. Radar, Sonar &; Navigation. Institution of Engineering and Technology, 2012. [Online]. Available: http://digital-library.theiet.org/content/books/ra/pbra022e
[21] J. Liu, H. Li, and B. Himed, "Joint optimization of transmit and receive beamforming in active arrays," *IEEE Signal Processing Letters*, vol. 21, no. 1, pp. 39–42, Jan 2014.
[22] P. Stoica, H. He, and J. Li, "Optimization of the receive filter and transmit sequence for active sensing," *IEEE Transactions on Signal Processing*, vol. 60, no. 4, pp. 1730–1740, April 2012.
[23] A. Aubry, A. D. Maio, and M. M. Naghsh, "Optimizing radar waveform and doppler filter bank via generalized fractional programming," *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, no. 8, pp. 1387–1399, Dec 2015.
[24] S. Ahmed, J. S. Thompson, Y. R. Petillot, and B. Mulgrew, "Unconstrained synthesis of covariance matrix for MIMO radar transmit beampattern," *Signal Processing, IEEE Transactions on*, vol. 59, no. 8, pp. 3837 –3849, Aug. 2011.
[25] I. Bekkerman and J. Tabrikian, "Target detection and localization using MIMO radars and sonars," *IEEE Transactions on Signal Processing*, vol. 54, no. 10, pp. 3873–3883, Oct 2006.
[26] "Matlab code for adaptive waveform design." [Online]. Available: https://github.com/sjh227/mmse_waveform_design
[27] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *International Conference on Neural Networks*, 1995, pp. 1942–1948.

Steven Herbert received the M.A. (cantab) and M.Eng. degrees from the University of Cambridge Department of Engineering in 2010, and the Ph.D. degree from the University of Cambridge Computer Laboratory in 2015. Following the submission of his Ph.D. thesis he remained at the University of Cambridge Computer Laboratory as a Research Assistant from May to June 2014, and remained a visiting researcher until December 2016. From January 2015 until July 2016 he worked at Blu Wireless Technology, Bristol, during which time he was a co-inventor of five patent applications (all currently under review). He started his current position, Research Associate at the University of Edinburgh, in September 2016, where he is working on adaptive waveform design for active sensing. His primary research interests lie in statistical signal processing and information theory, and his other interests include quantum information theory. He also has significant experience in mesh network organisation and routing.

James R. Hopgood received the M.A., M.Eng. degree in Electrical and Information Sciences in 1997 and a Ph.D. in July 2001 in Statistical Signal Processing, part of Information Engineering, both from the University of Cambridge, England. He was then a Post-Doctoral Research Associate for the year after his Ph.D within the same group, at which point he became a Research Fellow at Queens College continuing his research in the Signal Processing Laboratory in Cambridge. He joined the University of Edinburgh in April 2004, and is currently a Senior Lecturer in the Institute for Digital Communications, within the School of Engineering, at the University of Edinburgh, Scotland. His research specialisation include model-based Bayesian signal processing, speech and audio signal processing in adverse acoustic environments, including blind dereverberation and multi-target acoustic source localisation and tracking, single channel signal separation, distant speech recognition, audio-visual fusion, medical imaging, blind image deconvolution, and general statistical signal and image processing. Since September 2011, he is Editor-in-Chief for the IET Journal of Signal Processing.

Professor Bernard (Bernie) Mulgrew (FIEEE, FREng, FRSE, FIET) received his B.Sc. degree in 1979 from Queen's University Belfast. After graduation, he worked for 4 years as a Development Engineer in the Radar Systems Department at Ferranti, Edinburgh. From 1983-1986 he was a research associate in the Department of Electrical Engineering at the University of Edinburgh. He was appointed to lectureship in 1986, received his Ph.D. in 1987, promoted to senior lecturer in 1994 and became a reader in 1996. The University of Edinburgh appointed him to a Personal Chair in October 1999 (Professor of Signals and Systems). His research interests are in adaptive signal processing and estimation theory and in their application to radar and sensor systems. Prof. Mulgrew is a co-author of three books on signal processing.