UNIMODULAR WAVEFORM DESIGN WITH LOW CORRELATION LEVELS: A FAST ALGORITHM DEVELOPMENT TO SUPPORT LARGE-SCALE CODE LENGTHS

Yongzhe Li, Chunxuan Shi, and Ran Tao

School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China
Emails: lyz@ieee.org/yongzhe.li@bit.edu.cn, chunxuanshi@bit.edu.cn, rantao@bit.edu.cn

ABSTRACT

We deal with the problem of unimodular waveform(s) design with low correlation levels for the case of large-scale code lengths that can reach tens of thousands. Our primary goals are to reduce the resulting complexity with high efficiency, and meanwhile, to ensure an integrated sidelobe level (ISL) or weighted ISL (WISL) of waveforms as low as possible. To this end, we study a generic model for the minimization of ISL/WISL, wherein the objective function is formulated to embed a Hadmard product into a high-order matrix norm. Our major contributions lie in the transformation of the objective into a proper form via multiple shift matrices and the reformulation of problem in order to use alternating direction method of multipliers (ADMM) technique. In particular, we introduce a virtual matrix to form an additional equality constraint for ADMM, whose augmented Lagrangian is elaborated to help derive a fast algorithm that iterates with closed-form solutions. Simulation results verify the superiority of our algorithm over existing state-of-the-art algorithms in different aspects.

Index Terms— Alternating direction method of multipliers, correlations, large code length, waveform design.

1. INTRODUCTION

Waveform design has been a research field that continuously attracts significant interest over the past several decades [1]–[4]. It plays an essential role in radar [5], [6], communications [7], and active sensing [8] since the high-quality transmission via waveform(s) is determinant to performance improvements at receivers [9]. When it comes to multiple-input multiple-output (MIMO) radar or sensing, the waveforms which constitute a set are commonly expected to have desirable characteristics such as good correlation or weighted correlation levels. Equivalently, they are expected to enable low integrated sidelobe level (ISL) or weighted ISL (WISL) [3], so that the mutual (quasi)-orthogonality between waveforms can be guaranteed to obtain various advantages. For example, the waveforms with low ISL/WISL can help to better target extraction [10], improve ambiguity properties [11], and increase system robustness [3], to name a few.

This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 61901041 and Grant 62171029.

There exist works on multiple-waveform design [12]–[19], and some of them focus on pursuing low ISL/WISL for waveforms [12]–[15]. In particular, the algorithms 'CAN' and 'WeCAN' [12] are generally considered as the benchmark methods. More recently, the framework of majorization-minimization (MM) [20] has been introduced for designing waveforms with good ISL/WISL performance, based on which 'MM-Corr' and 'MM-WeCorr' [13] are pioneering methods. Despite significantly faster than CAN/WeCAN verified by medium-scale code lengths, MM-Corr/WeCorr still costs heavy time consumption (reaching dozens of minutes) for large code lengths of tens of thousands (see Sec. 4 for evidence). The later MM-based developments such as 'ISLNew' and 'WISLNew' [14], however, are not experimented on extremely large code lengths. As for other relevant algorithms that are newly reported [15], [17]–[19], they rarely cope with large-scale code lengths.

In this paper, we research the problem of designing unimodular waveform(s) with low ISL/WISL for code lengths of large scale that can reach tens of thousands. We aim to reduce the heavy complexity with high efficiency, and meanwhile to guarantee an ISL/WISL performance as good as possible. To fulfill the goals, we cope with a generic model of ISL/WISL minimization whose objective however, takes a complex form with Hadmard product involved in a high-order norm. By introducing a set of shift matrices to the objective and also using the algebraic structures therein, we then transform and reformulate the generalized WISL minimization problem into a form that allows us to use the ADMM technique [21]. A virtual matrix is newly introduced to form an additional constraint for the ADMM, and the augmented Lagrangian of it is therefore elaborated. Based on this, we develop a fast algorithm that iterates with closed-form solutions to the waveform design problem. Simulation results show that our algorithm outperforms existing state-of-the-art algorithms in terms of different aspects.

Notations: $\circledast$, $\otimes$, $\odot$, $\|\cdot\|$, $|\cdot|$, $\mathrm{vec}\{\cdot\}$, $(\cdot)^*$, $(\cdot)^{\mathrm{T}}$, $(\cdot)^{\mathrm{H}}$ respectively denote the circular convolution, Kronecker product, Hadamard product, Euclidean norm, modulus, column-wise vectorization, conjugate, transpose, and conjugate transpose operations. In addition, $[\cdot]_{i,j}$ is the (i,j)th element of a matrix, $\mathcal{G}_P\{\cdot\}$ represents picking up the last P elements to form a new vector, $\mathcal{F}_P\{\cdot\}$ denotes the $(2P-1)$-point fast Fourier transform (FFT), and $\mathbf{0}_P$ is a P-sized vector of zeros, respectively.

2. PROBLEM FORMULATION

Let us consider designing a set of M unimodular waveforms, each of which has a code length P. We denote the waveform matrix as $\mathbf{Y} \triangleq [\mathbf{y}_1, ..., \mathbf{y}_M] \in \mathbb{C}^{P\times M}$, whose mth column $\mathbf{y}_m \triangleq \left[e^{j\psi_m(1)}, \ldots, e^{j\psi_m(P)}\right]^{\mathrm{T}} \in \mathbb{C}^{P\times 1}$ consists of constant-modulus elements with arbitrary phase values $\psi_m(p) \in [-\pi, \pi), p = 1, \ldots, P$.

We focus on the classic ISL/WISL minimization-based waveform design that aims to synthesize waveforms with good correlation or weighted correlation properties. Given the waveform matrix $\mathbf{Y}$, the WISL of it can be expressed as [12]

$$\zeta = \sum_{m=1}^{M} \sum_{\substack{p=-P+1 \\ p\neq 0}}^{P-1} \gamma_p^2 |r_{mm}(p)|^2 + \sum_{m=1}^{M} \sum_{\substack{m'=1 \\ m'\neq m}}^{M} \sum_{p=-P+1}^{P-1} \gamma_p^2 |r_{mm'}(p)|^2 \tag{1}$$

where we use $r_{mm'}(p) \triangleq \sum_{k=p+1}^{P} y_m(k) y_{m'}^*(k-p)$ to denote the cross-correlation between the mth and m'th waveforms, and $\{\gamma_p\}_{p=-P+1}^{P-1}$ to be the real-valued weights for controlling the auto- and cross-correlation levels, both at the pth time lag. When all the weights equal 1, the WISL boils down to the ISL of waveforms. Based on this model, the unimodular waveform design problem can written as

$$\min_{\mathbf{Y}} \zeta \ \text{ s.t.} \ |[\mathbf{Y}]_{m,p}| = 1, \ m = 1, \ldots, M; \ p = 1, \ldots, P \tag{2}$$

whose constraints guarantee the constant-modulus property of all elements in the waveform matrix.

3. UNIMODULAR WAVEFORM DESIGN VIA ADMM

The design problem (2) can be rewritten as (see (42) in [14])

$$\min_{\mathbf{Y}} \quad \sum_{p=1}^{2P} \left\| \mathbf{Y}^{\mathrm{H}} \left((\mathbf{a}_p \mathbf{a}_p^{\mathrm{H}}) \odot \mathbf{\Gamma} \right) \mathbf{Y} \right\|^2 \tag{3a}$$

$$\text{s.t.} \quad |[\mathbf{Y}]_{m,p}| = 1, \ m = 1, \ldots, M; \ p = 1, \ldots, P \tag{3b}$$

where $\mathbf{\Gamma} \in \mathbb{R}^{P\times P}$ is a Toeplitz matrix whose upper and lower triangular parts are constructed by the weights $\{\gamma_p\}_{p=0}^{P-1}$ and $\{\gamma_{-p}\}_{p=0}^{P-1}$, respectively, and $\mathbf{a}_p \triangleq [1, e^{-j\omega_p}, ..., e^{-j(P-1)\omega_p}]^{\mathrm{T}}$ with $\omega_p = \frac{2\pi}{2P}(p-1)$.

Let $\mathbf{A}_p \triangleq \mathbf{a}_p \mathbf{a}_p^{\mathrm{H}} \in \mathbb{C}^{P\times P}$, using also $\mathbf{Y} = [\mathbf{y}_1, ..., \mathbf{y}_M]$, the objective (3a), denoted hereafter as $\tilde{\zeta}$, can be expressed as

$$\tilde{\zeta} = \sum_{m'=1}^{M} \sum_{m=1}^{M} \mathbf{y}_{m'}^{\mathrm{H}} \mathbf{Z}_m \mathbf{y}_{m'} \tag{4}$$

where $\mathbf{Z}_m \triangleq \sum_{p=1}^{2P} (\mathbf{A}_p \odot \mathbf{\Gamma}) \mathbf{y}_m \mathbf{y}_m^{\mathrm{H}} \left(\mathbf{A}_p^{\mathrm{H}} \odot \mathbf{\Gamma}^{\mathrm{H}} \right) \in \mathbb{C}^{P\times P}$.

Note that $\tilde{\zeta}$ in (4) is complicated since $\mathbf{Z}_m$ involves both the outer product of $\mathbf{y}_m$ and the Hadamard product of $\mathbf{A}_p$ and $\mathbf{\Gamma}$. To reformulate $\tilde{\zeta}$ into an easily tractable form, a proper transformation of $\mathbf{Z}_m$ is necessary. To this end, we take its column-wise vectorization, denoted by $\mathbf{z}_m \triangleq \mathrm{vec}\{\mathbf{Z}_m\}$, into account for finding an equivalent expression of $\mathbf{Z}_m$.

Using the property $\mathrm{vec}\{\mathbf{B_1 B_2 B_3}\} = (\mathbf{B}_3^{\mathrm{T}} \otimes \mathbf{B}_1)\mathrm{vec}\{\mathbf{B}_2\}$ by enabling $\mathbf{B}_1 \triangleq \mathbf{A}_p \odot \mathbf{\Gamma}$, $\mathbf{B}_2 \triangleq \mathbf{y}_m \mathbf{y}_m^{\mathrm{H}}$, and $\mathbf{B}_3 \triangleq \mathbf{A}_p^{\mathrm{H}} \odot \mathbf{\Gamma}^{\mathrm{H}}$, the vector $\mathbf{z}_m$ can be rewritten as follows

$$\boldsymbol{z}_m = \sum_{p=1}^{2P} \left(\mathbf{A}_p^* \otimes \mathbf{A}_p \right) \odot (\mathbf{\Gamma}^* \otimes \mathbf{\Gamma}) \mathrm{vec}\left\{ \mathbf{y}_m \mathbf{y}_m^{\mathrm{H}} \right\} \tag{5}$$

where the elementary properties of Kronecker and Hadamard products have been used for the derivations. To proceed with $\mathbf{z}_m$, we present the following Lemma.

Lemma 1. *For a set of rank-1 matrices* $\{\mathbf{F}_n = \mathbf{f}_n \mathbf{f}_n^{\mathrm{H}} \mid \mathbf{f}_n \triangleq [1, \ldots, e^{-j(N-1)\pi/N}]^{\mathrm{T}}\}$*, the equality* $\sum_{n=1}^{2N} \mathbf{F}_n^* \otimes \mathbf{F}_n = 2N \sum_{p=-N+1}^{N-1} \mathbf{J}_p \otimes \mathbf{J}_p$ *holds if* $\{\mathbf{J}_p\}_{p=-N+1}^{N-1}$ *are shift matrices (Positive/nagative* p *for upper/lower shift, and* $\mathbf{J}_0 = \mathbf{I}_N$*).*

Proof. We consider $\overline{\mathbf{F}} \triangleq \sum_{n=1}^{2N} \mathbf{F}_n^* \otimes \mathbf{F}_n$ as a block matrix and denote its N^2 blocks by $\overline{\mathbf{F}}_{k_1 k_2} \in \mathbb{C}^{N\times N}, k_1, k_2 \in \{1, ..., N\}$. Since $[\mathbf{F}_n]_{k_3,k_4} = e^{j(k_4-k_3)\omega_p}$, $k_3, k_4 \in \{1, \ldots, N\}$, thus $[\overline{\mathbf{F}}_{k_1 k_2}]_{k_3,k_4}$ equals $2N$ when $k_2 - k_1 = k_4 - k_3$, and otherwise it equals 0. This proves that $\overline{\mathbf{F}}_{k_1 k_2} = 2N \mathbf{J}_{k_2-k_1}$, which leads to the equality in Lemma 1. The proof is complete. □

Applying Lemma 1 to the first component of (5) via enabling $N = P$ and $\mathbf{F}_n = \mathbf{A}_p$, after some straightforward derivations, we can finally rewrite $\mathbf{z}_m$ as

$$\begin{aligned} \mathbf{z}_m &= 2P \sum_{p=-P+1}^{P-1} (\mathbf{J}_p \otimes \mathbf{J}_p) \odot (\mathbf{\Gamma}^* \otimes \mathbf{\Gamma}) \ \mathrm{vec}\left\{ \mathbf{y}_m \mathbf{y}_m^{\mathrm{H}} \right\} \\ &= 2P \sum_{p=-P+1}^{P-1} \gamma_p^2 (\mathbf{J}_p \otimes \mathbf{J}_p) \mathrm{vec}\left\{ \mathbf{y}_m \mathbf{y}_m^{\mathrm{H}} \right\}. \end{aligned} \tag{6}$$

Before recovering $\mathbf{z}_m$ in (6) into an equivalent expression of $\mathbf{Z}_m$, we present the following results.

Lemma 2. *For any vector* $\mathbf{w} \in \mathbb{C}^{N^2\times 1}$ *that takes the from* $\mathbf{w} = (\mathbf{U} \otimes \mathbf{U}) \ \mathrm{vec}\{\mathbf{V}\}$ *via matrices* $\mathbf{U} \in \mathbb{C}^{N\times N}$ *and* $\mathbf{V} \in \mathbb{C}^{N\times N}$*, the matrix form constructed from* $\mathbf{w}$ *column-wisely is given by* $\mathbf{W} = \mathbf{U}\mathbf{V}\mathbf{U}^{\mathrm{T}} \in \mathbb{C}^{N\times N}$*.*

Proof. The $((k_1-1)N + k_2)$th element of $\mathbf{w}$ equals the (k_1, k_2)th element of $\mathbf{W}$, i.e., $\mathbf{w}((k_1-1)N + k_2) = [\mathbf{W}]_{k_1,k_2} = \sum_{p_1=1}^{N} \sum_{p_2=1}^{N} [\mathbf{U}]_{k_2,p_1} [\mathbf{U}]_{k_1,p_2} [\mathbf{V}]_{p_2,p_1}$. Therefore, the lemma holds. The proof is complete. □

Applying Lemma 2 to (6) by enabling $N = P$, $\mathbf{U} = \mathbf{J}_p$, and $\mathbf{V} = \mathbf{y}_m \mathbf{y}_m^{\mathrm{H}}$, we can recover $\mathbf{Z}_m$ and then rewrite $\tilde{\zeta}$ into the following form

$$\tilde{\zeta} = 2P \sum_{m=1}^{M} \sum_{m'=1}^{M} \sum_{p=-P+1}^{P-1} \gamma_p^2 \mathbf{y}_m^{\mathrm{H}} \mathbf{J}_p \mathbf{y}_{m'} \mathbf{y}_{m'}^{\mathrm{H}} \mathbf{J}_p^{\mathrm{H}} \mathbf{y}_m. \tag{7}$$

Hence, the design problem in (3) can be rewritten as

$$\min_{\mathbf{Y}} \quad \sum_{m'=1}^{M} \sum_{m=1}^{M} \sum_{p=-P+1}^{P-1} \gamma_p^2 \left| \mathbf{y}_{m'}^{\mathrm{H}} \mathbf{J}_p \mathbf{y}_m \right|^2 \tag{8a}$$

$$\text{s.t.} \quad |[\mathbf{Y}]_{m,p}| = 1, \ m = 1, \ldots, M; \ p = 1, \ldots, P. \tag{8b}$$

To solve (8), we introduce a virtual auxiliary matrix $\mathbf{X} \in \mathbb{C}^{P\times M}$ to transform it into the following form

$$\min_{\mathbf{X}} \quad \sum_{m'=1}^{M}\sum_{m=1}^{M}\sum_{p=-P+1}^{P-1} \gamma_p^2 \left|\mathbf{y}_{m'}^{\mathrm{H}}\mathbf{J}_p\mathbf{y}_m\right|^2 \tag{9a}$$

$$\text{s.t.} \quad |[\mathbf{X}]_{m,p}| = 1,\ m = 1,\ldots,M;\ p = 1,\ldots,P \tag{9b}$$

$$\mathbf{Y} = \mathbf{X} \tag{9c}$$

to which we apply the ADMM technique [21], [22] for finding the solution via iterations. To this end, the first-order Taylor expansion at $\mathbf{Y}^{(k)}$ for the kth iteration is employed to construct the augmented Lagrangian given as follows

$$\begin{aligned}\mathcal{L}_\rho(\mathbf{Y},\mathbf{X},\boldsymbol{\lambda}_{m'}) =& \sum_{m'=1}^{M}\sum_{m=1}^{M}\sum_{p=-P+1}^{P-1} \gamma_p^2\left|(\mathbf{y}_{m'}^{(k)})^{\mathrm{H}}\mathbf{J}_p\mathbf{y}_m^{(k)}\right|^2 \\ &+ 4\sum_{m'=1}^{M}\sum_{m=1}^{M}\sum_{p=-P+1}^{P-1} \gamma_p^2(\mathbf{y}_{m'}^{(k)})^{\mathrm{H}}\mathbf{J}_p\mathbf{y}_m^{(k)}(\mathbf{y}_m^{(k)})^{\mathrm{H}}\mathbf{J}_p^{\mathrm{H}}\mathbf{y}_{m'} \\ &+ \sum_{m'=1}^{M}\boldsymbol{\lambda}_{m'}^{\mathrm{H}}(\mathbf{y}_{m'}-\mathbf{x}_{m'}) + \rho\sum_{m'=1}^{M}||\mathbf{y}_{m'}-\mathbf{x}_{m'}||^2 \end{aligned} \tag{10}$$

where $\{\boldsymbol{\lambda}_{m'}\}_{m'=1}^{M} \in \mathbb{C}^{P\times 1}$ are Lagrange multipliers and ρ is the penalty parameter. The resulting ADMM consists of the following iterations

$$\mathbf{x}_{m'}^{(k+1)} = \exp\left\{i\cdot\arg\left(\mathbf{y}_{m'}^{(k)} + \frac{1}{\rho}\boldsymbol{\lambda}_{m'}^{(k)}\right)\right\} \tag{11a}$$

$$\mathbf{y}_{m'}^{(k+1)} = \mathbf{x}_{m'}^{(k+1)} - \frac{1}{\rho}\boldsymbol{\lambda}_{m'}^{(k)} - \mathbf{g}_{m'}^{(k)} \tag{11b}$$

$$\boldsymbol{\lambda}_{m'}^{(k+1)} = \boldsymbol{\lambda}_{m'}^{(k)} + \rho\left(\mathbf{y}_{m'}^{(k+1)} - \mathbf{x}_{m'}^{(k+1)}\right) \tag{11c}$$

with $\mathbf{g}_{m'}^{(k)} = \frac{2}{\rho}\sum_{m=1}^{M}\sum_{p=-P+1}^{P-1}\gamma_p^2\mathbf{J}_p\mathbf{y}_m^{(k)}(\mathbf{y}_m^{(k)})^{\mathrm{H}}\mathbf{J}_p^{\mathrm{H}}\mathbf{y}_{m'}^{(k)} \in \mathbb{C}^{P\times 1}$. For simplicity, we combine (11a), (11b), and (11c) and then write them into the form given as follows

$$\mathbf{y}_{m'}^{(k+1)} = \exp\left\{i\cdot\arg\left(\mathbf{y}_{m'}^{(k)} - \mathbf{g}_{m'}^{(k)}\right)\right\}. \tag{12}$$

Using the equalities $(\mathbf{y}_m^{(k)})^{\mathrm{H}}\mathbf{J}_p^{\mathrm{H}}\mathbf{y}_{m'}^{(k)} = \sum_{l=p+1}^{P}\mathbf{y}_m^{*(k)}(l)\,\mathbf{y}_{m'}^{(k)}(l-p) = r_{mm'}^{*(k)}(p) = r_{m'm}^{(k)}(-p)$, the explicit expression of $\mathbf{g}_{m'}^{(k)}$ in (12) can be obtained as

$$\mathbf{g}_{m'}^{(k)} = \frac{2}{\rho}\sum_{m=1}^{M}\sum_{p=-P+1}^{P-1}\gamma_p^2 r_{m'm}^{(k)}(-p)\mathbf{J}_p\mathbf{y}_m^{(k)} \triangleq \frac{2}{\rho}\sum_{m=1}^{M}\mathbf{h}_{m'm}^{(k)} \tag{13}$$

where $\mathbf{h}_{m'm}^{(k)} \triangleq \sum_{p=-P+1}^{P-1}\gamma_p^2 r_{m'm}^{(k)}(-p)\mathbf{J}_p\mathbf{y}_m^{(k)} \in \mathbb{C}^{P\times 1}$ can further be expressed via the circular convolution given by

$$\mathbf{h}_{m'm}^{(k)} = \mathcal{G}_P\left(\widetilde{\mathbf{r}}_{m'm}^{(k)} \circledast \widetilde{\mathbf{y}}_m^{(k)}\right) \tag{14}$$

with $\widetilde{\mathbf{r}}_{m'm}^{(k)} \in \mathbb{C}^{(2P-1)\times 1}$ and $\widetilde{\mathbf{y}}_m^{(k)} \in \mathbb{C}^{(2P-1)\times 1}$ defined as

$$\widetilde{\mathbf{r}}_{m'm}^{(k)} \triangleq \left[\gamma_{-P+1}^2 r_{m'm}^{(k)}(-P+1),\ldots,\gamma_{P-1}^2 r_{m'm}^{(k)}(P-1)\right]^{\mathrm{T}} \tag{15}$$

$$\widetilde{\mathbf{y}}_m^{(k)} \triangleq \left[(\mathbf{y}_m^{(k)})^{\mathrm{T}}, \mathbf{0}_{P-1}^{\mathrm{T}}\right]^{\mathrm{T}} \tag{16}$$

Algorithm 1 The proposed waveform design algorithm.

1: Initialization: $\mathbf{Y}^{(0)}$, ρ, and $k \leftarrow 0$
2: **repeat**
3: Calculate $\widetilde{\mathbf{r}}_{m'm}^{(k)}$ via (18), $\forall m, m' \in \{1,\ldots,M\}$
4: Calculate $\mathbf{g}_{m'}^{(k)}$ via (19), $\forall m'$
5: $\mathbf{y}_{m'}^{(k+1)} = \exp\left\{i\cdot\arg\left(\mathbf{y}_{m'}^{(k)} - \mathbf{g}_{m'}^{(k)}\right)\right\}, \forall m'$
6: $k \leftarrow k+1$
7: **until** convergence

respectively. The effect of $\{\mathbf{J}_p\}_{p=-P+1}^{P-1}$ on shifting elements of $\mathbf{y}_m^{(k)}$ with zeros inserted is employed in deriving to (14).

Substituting (14) to (13), we obtain

$$\mathbf{g}_{m'}^{(k)} = \frac{2}{\rho}\sum_{m=1}^{M}\mathcal{G}_P\left(\widetilde{\mathbf{r}}_{m'm}^{(k)} \circledast \widetilde{\mathbf{y}}_m^{(k)}\right) \tag{17}$$

which can further be implemented via the FFTs with respect to $\mathbf{y}_m^{(k)}$ and $\mathbf{y}_{m'}^{(k)}$, i.e.,

$$\widetilde{\mathbf{r}}_{m'm}^{(k)} = \mathcal{F}_P^{-1}\left\{\mathcal{F}_P\{\mathbf{y}_{m'}^{(k)}\} \odot \mathcal{F}_P^{*}\{\mathbf{y}_m^{(k)}\}\right\} \odot \boldsymbol{\gamma} \tag{18}$$

$$\mathbf{g}_{m'}^{(k)} = \frac{2}{\rho}\sum_{m=1}^{M}\mathcal{G}_P\left\{\mathcal{F}_P^{-1}\left\{\mathcal{F}_P\{\mathbf{y}_m^{(k)}\} \odot \mathcal{F}_P\{\widetilde{\mathbf{r}}_{m'm}^{(k)}\}\right\}\right\} \tag{19}$$

with $\boldsymbol{\gamma} \triangleq [\gamma_{-P+1}^2,\ldots,\gamma_{P-1}^2]^{\mathrm{T}} \in \mathbb{C}^{(2P-1)\times 1}$.

The iteration procedures for the waveform design are summarized in Algorithm 1, for which the Steffensen-type acceleration [23], [24] with fixed-point iterations can be applied.

4. SIMULATION RESULTS

We compare our proposed algorithm with algorithms that include CAN/WeCAN [12], MM-Corr/MM-WeCorr [13], ISLNew/WISLNew [14], in terms of ISL/WISL performances.[1] Throughout simulations, we generate unimodular sequences with random phase values as the initialization, and we use the same initialized waveform(s) for each comparison. The same acceleration scheme (i.e., SQUAREM [25]) is used to speed up the compared algorithms except CAN and WeCAN. The stopping criterion is defined as the absolute ISL/WISL difference between two neighboring iterations normalized by the initial ISL/WISL, whose tolerance is set to 10^{-9}. All results are are obtained over 30 independent trials based on the same hardware and software configurations.

Example 1: ISL comparisons. We compare the ISL performances of the tested algorithms in terms of the following characteristics: i) the minimum and average ISL values obtained after convergence; ii) the averaged overall time consumption; and iii) the average number of iterations. For this example, we test multiple code lengths of large scale for fixed numbers of waveforms $M = 2$ and 3, wherein $P = 2^{13}, 2^{14}$, and 2^{15} are experimented. The ISL weights are all switched to 1 here.

[1]The FFT has been used here if algorithms allow for fast implementation.

Table 1. ISL performance comparisons versus different numbers of waveforms and code lengths of large scale.

	$P=8192, M=1$				$P=8192, M=2$				$P=16384, M=1$				$P=16384, M=2$				$P=32768, M=1$				$P=32768, M=2$			
	Min.[a]	Ave.[b]	Time[c]	Iter.[d]	Min.	Ave.	Time	Iter.	Min.	Ave.	Time	Iter.	Min.	Ave.	Time	Iter.	Min.	Ave.	Time	Iter.	Min.	Ave.	Time	Iter.
CAN	65.35	65.44	64.76	9696	81.28	81.28	590.79	23615	71.38	71.52	93.74	8131	87.30	87.30	1238.74	24797	77.36	77.43	299.16	12200	93.32	93.32	2318.22	24693
MM-Corr	64.51	64.60	17.27	5147	81.28	81.28	78.44	16397	70.54	70.59	154.18	24054	87.30	87.30	313.31	30439	76.55	76.61	1315.91	158867	93.32	93.32	1105.74	57173
ISLNew	64.48	64.59	8.07	1502	81.28	81.28	7.72	805	70.49	70.56	26.51	2169	87.30	87.30	21.79	978	76.49	76.57	66.05	2694	93.32	93.32	47.08	1151
Proposed	64.43	64.57	4.46	1124	81.28	81.28	5.81	727	70.47	70.55	12.64	1546	87.30	87.30	13.79	798	76.41	76.53	37.32	2450	93.32	93.32	26.82	842

[a] Min.: Minimum ISL value (in dB). [b] Ave.: Average ISL value (in dB). [c] Time: Average time consumption (in seconds). [d] Iter.: Average number of conducted iterations.

Table 2. WISL performance comparisons versus different numbers of waveforms and code lengths.

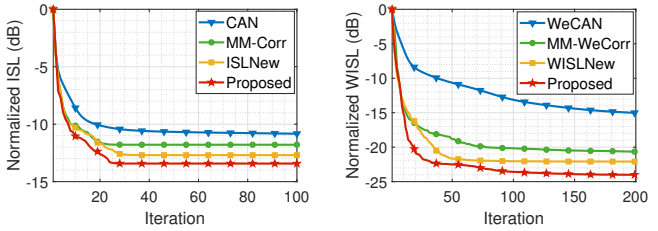
	$P=128, M=3$				$P=128, M=4$				$P=128, M=5$				$P=128, M=6$				$P=128, M=7$				$P=128, M=8$			
	Min.[a]	Ave.[b]	Time[c]	Iter.[d]	Min.	Ave.	Time	Iter.	Min.	Ave.	Time	Iter.	Min.	Ave.	Time	Iter.	Min.	Ave.	Time	Iter.	Min.	Ave.	Time	Iter.
WeCAN	22.49	23.12	41.11	2791	27.27	27.97	100.35	5347	34.89	35.25	176.42	7703	41.07	41.29	202.32	7126	45.71	45.76	304.11	9750	48.78	48.86	358.36	10110
MM-WeCorr	-22.52	-14.24	6.89	284	21.75	23.16	45.48	997	34.10	34.52	56.10	850	40.36	40.62	134.56	1373	45.20	45.26	153.29	1182	47.81	47.90	200.13	1164
WISLNew	-29.54	-22.81	2.51	182	21.77	22.94	22.58	711	33.81	33.98	30.98	685	40.32	40.60	56.25	889	45.04	45.15	95.96	1134	47.72	47.81	167.24	1444
Proposed	-47.22	-26.12	0.05	111	21.56	22.93	0.36	462	33.01	33.94	0.98	524	40.24	40.60	1.69	745	44.92	45.15	2.19	808	47.67	47.80	3.26	1024

[a] Min.: Minimum WISL value (in dB). [b] Ave.: Average WISL value (in dB). [c] Time: Average time consumption (in seconds). [d] Iter.: Average number of conducted iterations.

The corresponding results are shown in Table 1. It can be seen from Table 1 that our proposed algorithm outperforms the other three counterparts in terms of all aspects evaluated, and such advantages become more significant for extremely large code length (see the case for $P = 2^{15}$). Therein lies the considerable improvement of our algorithm reducing the time consumption to 26.83 seconds via 842 iterations when $M = 2$, while MM-Corr and CAN cost around 19 and 39 minutes with iterations more than several thousands. The algorithm ISLNew behaves as the second best, but it still costs about 2 and 1.5 times consumption of time and iterations more than ours, respectively. The cases with single waveform (i.e., $M = 1$) tested show visible advances of our algorithm on the obtained minimum and average ISL after convergence, even if the values are rounded off to two digits after the decimal point.

Example 2: WISL comparisons. We conduct the WISL performance comparisons between the aforementioned algorithms in terms of the characteristics: i) the minimum and average WISL values obtained after convergence; ii) the averaged overall time consumption; and iii) the average number of iterations. Different from *Example 1*, the code length for evaluation is fixed to $P = 128$ here,[2] but the multiple numbers of waveforms from $M = 3$ to 8 are studied. The ISL controlling weights are $\gamma_p = 1$ when $|p| \leq 20$, while the others are zeros. The corresponding results are shown in Table 2. It can be seen from Table 2 that our proposed algorithm again outperforms the other three counterparts in terms of all aspects, and its superiority on the overall consumption of time and iterations become significant especially for a larger number of waveforms (see the case of $M = 8$). The time consumption therein has been reduced around 51 and 61 times (3.26 versus 167.24 and 200.13 seconds) compared to WISLNew and MM-WeCorr, respectively, also with less iterations.

Example 3: ISL/WISL convergence speed. We compare the convergence speeds of our proposed algorithm versus numbers of iterations with the aforementioned algorithms for both ISL and WISL evaluations, respectively. The parameters $M = 1$ and $M = 2$ are respectively chosen for the experiments on ISL and WISL convergence speeds, and the code length of $P = 64$ is tested for both. Other parameters used here are the same as in *Examples 1* and *2* (respectively for ISL and WISL cases). We normalize the obtained ISL/WISL values by the initial ISL/WISL value, whose corresponding results are shown in Fig. 1. It can be seen from Figs. 1(a) and 1(b) that our newly proposed algorithm shows faster convergence speeds compared to all other counterparts tested, for both the ISL and WISL cases. It gives around 0.8 dB, 1.7 dB, and 2.6 dB improvements on normalized ISL values after 100 iterations compared to ISLNew, MM-Corr, and CAN, respectively. As for the WISL convergence speed after 200 iterations, it shows around 2.0 dB, 3.5 dB, and 9.0 dB improvements accordingly.



(a) Convergence speed: ISL case. (b) Convergence speed: WISL case.

Fig. 1. ISL and WISL values versus numbers of iterations.

5. CONCLUSION

We have studied the problem of designing unimodular waveform(s) with low ISL/WISL for large code lengths up to tens of thousands. Starting from a generic WISL minimization model with a complex objective, we have transformed the WISL into a proper form via a set of shift matrices. Based on this, we have then reformulated the problem into a form that allows us to use ADMM techniques. We have introduced a virtual matrix to form an additional constraint for the ADMM, and have also elaborated its augmented Lagrangian to develop a fast algorithm. Simulation results have verified the superiority of our proposed algorithm in terms of different aspects.

[2] Note that the large code length with partial non-zero weights for WISL is relatively much easier for algorithms and therefore is not presented here.

6. REFERENCES

[1] N. Levanon and E.Mozeson, *Radar Signals*. Hoboken, NJ, USA: Wiley, 2004.

[2] M. C. Wicks, E. L. Mokole, S. D. Blunt, R. S. Schneible, and V. J. Amuso, *Principles of Waveform Diversity and Design*. Raleigh, NC, USA: SciTech Publishing, 2010.

[3] H. He, J. Li, and P. Stoica, *Waveform design for active sensing systems: A computational Approach*. Cambridge, UK: Cambridge University Press, 2012.

[4] F. Gini, A. De Maio, and L. Patton, *Waveform design and diversity for advanced radar systems*. London, UK: Institution Eng. Tech., 2012.

[5] P. Stoica, J. Li, and Y. Xie, "On probing signal design for MIMO radar," *IEEE Trans. Signal Process.*, vol. 55, no. 8, pp. 4151–4161, Aug. 2007.

[6] S. D. Blunt and E. L. Mokole, "Overview of radar waveform diversity," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 31, no. 11, pp. 2–42, Nov. 2016.

[7] D. Tse and P. Viswanath, *Fundamentals of wireless communication*. Cambridge, UK: Cambridge University Press, 2005.

[8] P. Stoica, H. He, and J. Li, "Optimization of the receive filter and transmit sequence for active sensing," *IEEE Trans. Signal Process.*, vol. 60, no. 4, pp. 1730–1740, Apr. 2012.

[9] J. Li, P. Stoica, and X. Zheng, "Signal synthesis and receiver dsign for MIMO radar imaging," *IEEE Trans. Signal Process.*, vol. 56, no. 8, pp. 3959–3968, Aug. 2008.

[10] A. Aubry, A. De Maio, Y. Huang, and M. Piezzo, "Robust design of radar Doppler filters," *IEEE Trans. Signal Process.*, vol. 64, no. 22, pp. 5848–5860, Nov. 2016.

[11] Y. Li, S. A. Vorobyov, and V. Koivunen, "Ambiguity function of the transmit beamspace-based MIMO radar," *IEEE Trans. Signal Process.*, vol. 63, no. 17, pp. 4445–4457, Sep. 2015.

[12] H. Hao, P. Stoica, and J. Li, "Designing unimodular sequences sets with good correlations—Including an application to MIMO radar," *IEEE Trans. Signal Process.*, vol. 57, no. 11, pp. 4391–4405, Nov. 2009.

[13] J. Song, P. Babu, and D. P. Palomar, "Sequence set design with good correlations properties via Majorization-Mimimization," *IEEE Trans. Signal Process.*, vol. 64, no. 11, pp. 2866–2879, Jun. 2016.

[14] Y. Li and S. A. Vorobyov, "Fast algorithms for designing multiple unimodular waveforms with good correlation properties," *IEEE Trans. Signal Process.*, vol. 66, no. 5, pp. 1197–1212, Mar. 2018.

[15] J. Liang, H. C. So, J. Li, and A. Farina, "Unimodular sequence design based on alternating direction method of multipliers," *IEEE Trans. Signal Process.*, vol. 64, no. 20, pp. 5367–5381, 2016.

[16] G. Cui, H. Li, and M. Rangaswamy, "MIMO radar waveform design with constant modulus and similarity constraints," *IEEE Trans. Signal Process.*, vol. 62, no. 2, pp. 343–353, Jan. 2014.

[17] Z. Cheng, Z. He, B. Liao, and M. Fang, "MIMO radar waveform design with papr and similarity constraints," *IEEE Trans. Signal Process.*, vol. 66, no. 4, pp. 968–981, 2017.

[18] A. Bose and M. Soltanalian, "Constructing binary sequences with good correlation properties: An efficient analytical-computational interplay," *IEEE Trans. Signal Process.*, vol. 66, no. 11, pp. 2998–3007, Jun. 2018.

[19] S. P. Sankuru and P. Babu, "Designing unimodular sequence with good auto-correlation properties via block majorization-minimization method," *Signal Process.*, vol. 176, pp. 1–9, Jan. 2020.

[20] D. R. Hunter and K. Lange, "A tutorial on MM algorithms," *Amer. Statist.*, vol. 58, no. 1, pp. 30–37, Feb. 2004.

[21] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, *Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers*. Delft, The Netherlands: now Publisher Inc., Jan. 2011.

[22] H. Ouyang, N. He, L. Tran, and A. Gray, "Stochastic alternating direction method of multipliers," in *Proc. Int. Conf. Machine Learning*, Jun. 2013, pp. 80–88.

[23] M. Raydan and B. F. Svaiter, "Relaxed steepest descent and cauchy-barzilai-borwein method," *Comput. Optim. Appl.*, vol. 21, no. 2, pp. 155–167, Feb. 2002.

[24] A. Cordero, J. L. Hueso, E. Martínez, and J. R. Torregrosa, "Steffensen type methods for solving nonlinear equations," *J. Comput. Appl. Math.*, vol. 236, pp. 3058–3064, Jun. 2012.

[25] R. Varadhan and C. Roland, "Simple and globally convergent methods for accelerating the convergence of any EM algorithm," *Scand. J. Statist.*, vol. 35, no. 2, pp. 335–353, 2008.