CPGD: Cadzow Plug-and-Play Gradient Descent for Generalised FRI

Matthieu Simeoni, Member, IEEE, Adrien Besson, Associate Member, IEEE, Paul Hurley, Senior Member, IEEE, and Martin Vetterli, Fellow, IEEE

Abstract—Finite rate of innovation (FRI) is a powerful reconstruction framework enabling the recovery of sparse Dirac streams from uniform low-pass filtered samples. An extension of this framework, called generalised FRI (genFRI), has been recently proposed for handling cases with arbitrary linear measurement models. In this context, signal reconstruction amounts to solving a joint constrained optimisation problem, yielding estimates of both the Fourier series coefficients of the Dirac stream and its so-called annihilating filter, involved in the regularisation term. This optimisation problem is however highly non convex and non linear in the data. Moreover, the proposed numerical solver is computationally intensive and without convergence guarantee.

In this work, we propose an implicit formulation of the genFRI problem. To this end, we leverage a novel regularisation term which does not depend explicitly on the unknown annihilating filter yet enforces sufficient structure in the solution for stability reasons. The resulting optimisation problem is still non convex, but simpler since linear in the data and with less unknowns. We solve it by means of a provably convergent proximal gradient descent (PGD) method. Since the proximal step does not admit a simple closed-form expression, we propose an inexact PGD method, coined as Cadzow plug-and-play gradient descent (CPGD). The latter approximates the proximal steps by means of Cadzow denoising, a well-known denoising algorithm for sparse signals such as the prototypical sparse signal, namely the $T$-periodic stream of Diracs:

$$x(t) = \sum_{k' \in \mathbb{Z}} \sum_{k=1}^{K} x_k \delta(t - k - T k'), \quad \forall t \in \mathbb{R},$$

with $x_k \in \mathbb{C}$ and $t_k \in [0,T]$. In the FRI framework, the sparsity is measured in terms of its rate of innovation, defined as the number of degrees of freedom per unit of time. For instance, the Dirac stream (1) has $2K$ degrees of freedom $\{x_k,t_k\}_{k=1}^{K}$ per period $T$, yielding a finite rate of innovation of $\rho = 2K/T$. Intuitively, any lossless sampling scheme for (1) must therefore have a sampling rate at least as large as the rate of innovation $\rho$ or it will be impossible to fix all degrees of freedom.

Blu et al. described a sampling scheme achieving the second best sampling rate after the critical innovation rate, permitting to perfectly recover the signal innovations from the knowledge of any $2K + 1$ consecutive Fourier coefficients of $x$ [5]. Unfortunately, this scheme can be very sensitive to noise perturbations in the collected samples. This is because the recovery of the innovations $t_k$ relies on the resolution of a so-called annihilating equation which requires the Toeplitz matrix built from the Fourier coefficients to be rank deficient. While this structural constraint is guaranteed to hold in the case of noiseless recovery of Dirac streams, it can break down in the presence of noise, inevitable in practical applications.

As a remedy, Blu et al. proposed to denoise the collected samples prior to solving the annihilating equation. To this end, they leveraged the well-known Cadzow algorithm [6], which aims to retrieve the closest rank-deficient Toeplitz matrix to a high-dimensional embedding of the data via an alternating projection method. When upgraded with this extra denoising step, simulations results from Blu et al. revealed that the overall accuracy of the recovery procedure remains very good for signal-to-noise ratios (SNR) as low as 5 dB [5]. While Cadzow algorithm empirically provides accurate results after a few iterations, convergence in theory has however not been demonstrated to date, due to the non convex nature of the space of rank-deficient matrices. Condat and Hirabayashi [7] revisited Cadzow...
denoising as a structured low-rank approximation (SLRA) problem and proposed a Douglas-Rachford splitting algorithm to solve it [8], with higher accuracy than traditional Cadzow denoising. Unfortunately, the gain in accuracy comes at the price of significantly higher computational cost, the Douglas-Rachford splitting method requiring many more iterations to converge than Cadzow algorithm.

In addition to their somewhat heuristic nature, neither Cadzow denoising nor its upgrade can handle more general types of input measurements as considered in the generalised FRI (genFRI) framework introduced by Pan et al. in [9]. The latter extends FRI to very generic cases where the measurements are related to the unknown Fourier coefficients of signals satisfying the annihilating property by a linear map. In such configurations, both the Fourier coefficients and their corresponding annihilating filter are unknown and must be estimated from the data. Pan et al. proposed to perform this joint estimation task by solving a constrained optimisation problem which recovers the Fourier coefficients, required to minimise a quadratic data-fidelity term, and their corresponding annihilating filter coefficients. The annihilating equation linking the two unknowns is explicitly enforced as a constraint. This optimisation problem is highly non convex and non linear in the data. They suggested to solve it via an iterative alternating minimisation algorithm with multiple random initialisations [9]. The proposed algorithm however comes without convergence guarantees and is computationally intensive.

In this paper, we propose an implicit formulation of the genFRI problem in which only the Fourier coefficients to be annihilated are recovered. This formulation does not rely explicitly on the unknown annihilating filter but rather leverages a structured low-rank regularisation constraint based on a “Toeplitzification” linear operator, guaranteeing non-trivial solutions to the annihilating equation. The resulting optimisation problem is still non convex, but simpler to analyse and solve since it is linear in the data and with less unknowns. We solve the implicit genFRI problem via proximal gradient descent (PGD) [10], [11].

We first consider PGD with exact proximal steps which is shown to converge towards critical points of the implicit genFRI problem. The latter is however impractical since the proximal step involves at each iteration does not have a closed-form expression. We therefore consider an inexact PGD [12], with proximal steps approximated by means of alternating projections. In the case of injective forward matrices, the approximate proximal step is shown to reduce to Cadzow denoising. Such an approach is reminiscent of the plug-and-play (PnP) framework in which proximal operators involved in first-order iterative methods are replaced by generic denoisers [13], [14], [15]. For this reason, we name our reconstruction algorithm Cadzow PnP Gradient Descent (CPGD).¹ We demonstrate that CPGD converges locally towards fixed points of the update equation for injective forward matrices. Through simulations of irregular and noisy time sampling of periodic stream of Diracs we show that CPGD is almost always more accurate and more efficient than the procedure proposed by Pan et al. in [9], sometimes by several orders of magnitude.

The remainder of the paper is organised as follows:

- Preliminary concepts required for the understanding of the further sections are introduced in Section II.
- Section III describes the genFRI problem and details the proposed implicit formulation. The CPGD algorithm is introduced in Section IV.
- Experiments and results are detailed in Section V and concluding remarks are given in Section VI.

Finally, note that all experiments and simulations are fully reproducible using the benchmarking routines provided in our GitHub repository [16].

II. PRELIMINARIES

In this section we introduce a linear operator, baptised Toeplitzification operator,² which transforms a vector into a Toeplitz matrix. This operator will be used in the regularisation term of our implicit genFRI optimisation problem. We then briefly review the method of alternating projections [17] as well as the FRI [4] framework and Cadzow denoising [7].

A. Toeplitzification Operator

Assume that we are given an arbitrary vector \( x \in \mathbb{C}^N \), \( N = 2M + 1 \), with entries indexed as follows:

\[
x := [x_{-M}, x_{-M+1}, \ldots, x_{M-1}, x_M]^T.
\]

Then, for any \( P \leq M \), we can embed \( x \) into the space \( \mathbb{T}_P \) of Toeplitz matrices of \( \mathbb{C}^{(N-P)(P+1)} \) by means of the following Toeplitzification operator:

\[
\mathbb{T}_P : \mathbb{C}^N \rightarrow \mathbb{T}_P \subset \mathbb{C}^{(N-P)(P+1)}
\]

\[
x \mapsto [\mathbb{T}_P(x)]_{i,j} := x_{i-M+P+i-j}, (2)
\]

where \( i = 1, \ldots, N-P, \ j = 1, \ldots, P+1 \). Note from (2) that the value of an entry \([\mathbb{T}_P(x)]_{i,j}\) of the matrix \(\mathbb{T}_P(x)\) depends only on the distance \(i-j\) between the row and column indexes: \(\mathbb{T}_P(x)\) is therefore a Toeplitz matrix and the vector \(x\) is called its generator.

The Toeplitzification operator (2) can be used to implement linear convolutions. Indeed, it can be shown (see Appendix A available as supplementary material) that the multiplication of \(\mathbb{T}_P(x)\) with a vector \(u = [u_1, \ldots, u_{P+1}]^T \in \mathbb{C}^{P+1}\) returns the valid part³ of the convolution between the two zero-padded sequences \(\tilde{x} := \ldots, 0, x_{-M}, \ldots, x_0, \ldots \in \mathbb{C}^{2P+1}\) and \(\tilde{u} := \ldots, 0, u_1, \ldots, u_{P+1}, 0, \ldots \in \mathbb{C}^{2P+1}\). Similarly, the multiplication of \(\mathbb{T}_P(x)^H\) with a vector \(v = [v_1, \ldots, v_{N-P}]^T \in \mathbb{C}^{N-P}\) returns the valid part of the cross-correlation between \(\tilde{x}\) and the zero-padded sequence \(\hat{v} := \ldots, 0, v_1, \ldots, v_{N-P}, 0, \ldots \in \mathbb{C}^{2P}\).

¹An efficient Python implementation of CPGD is provided on our GitHub repository [16].

²The alternative appellation Toeplitzization was used in [7].

³See Appendix A for a formal definition of the valid part of a convolution between zero-padded sequences.
B. Inverse Toeplitzification Operator

The inverse Toeplitzification operator is the pseudoinverse of the Toeplitzification operator, mapping a Toeplitz matrix $H \in \mathbb{C}^{(N-P)\times(P+1)}$ to its generator $h \in \mathbb{C}^N$. As we shall prove in Proposition 2, inverse Toeplitzification is achieved by averaging across each diagonal of $T_P(x)$. It is interesting to note that this operation is also leveraged in Cadzow denoising as described in [5], in order to map back the data from its high dimensional Toeplitz embedding. The formal interpretation of this inverse map as the pseudoinverse of the Toeplitzification operator proposed hereafter is nevertheless not discussed in [5], nor anywhere else we may be aware of.

To compute the pseudoinverse of $T_P$, we first need an expression for its adjoint map, detailed in the proposition hereafter.

**Proposition 1** (Adjoint operator of $T_P$). The adjoint operator $T_P^*$ of $T_P$ defined in (2) is given by

$$T_P^* : \mathbb{C}^{(N-P)\times(P+1)} \to \mathbb{C}^N \quad \text{given by}$$

$$H \mapsto h_j = \sum_{i+k-j-1=-P}^{i+k-j-1=P} H_{ik}, \quad j = 1, \ldots, N. \quad (3)$$

**Proof.** The proof is described in Appendix B available as supplementary material of this manuscript. \qed

Note that the adjoint map $T_P^*$ proceeds by summing across each diagonal of the input matrix $H$. We are now ready to derive an expression for the (left) pseudoinverse of $T_P$, described in the proposition hereafter.

**Proposition 2** (Pseudoinverse of $T_P$). The pseudoinverse $T_P^+ : \mathbb{C}^{(N-P)\times(P+1)} \to \mathbb{C}^N$ of $T_P$ defined in (2) is given by

$$T_P^+ = \Gamma^{-1} T_P^* \quad (4)$$

where $\Gamma \in \mathbb{C}^{N \times N}$ is a diagonal matrix with diagonal entries given by

$$\Gamma_{i,j} = \min(i,P+1,N+1-i), \quad i = 1, \ldots, N. \quad (5)$$

**Proof.** The proof of this proposition is given in Appendix C available as supplementary material of this manuscript. \qed

Observe that the composition of $T_P^+$ and $\Gamma^{-1}$ in the expression of the pseudoinverse (4) indeed corresponds to a diagonal averaging since $T_P^*$ first sums across each diagonal of the matrix $H \in \mathbb{C}^{(N-P)\times(P+1)}$ and $\Gamma^{-1}$ then divides the sums by the number of elements on each diagonal.

C. The Method of Alternating Projections

In this section we briefly discuss the method of alternating projections (MAP) [17], central to Cadzow denoising. It is used in computational mathematics to approximate projections onto intersecting sets. In its simplest form proposed by von Neumann in 1933 [18], the MAP performs a cascade of $n$ projection steps onto subsets $\{M_1, \ldots, M_K\}$ of some Hilbert space $\mathcal{H}$, starting from a point $z \in \mathcal{H}$:

$$\hat{z} = \left[ \Pi_{M_K} \cdots \Pi_{M_1} \right]^n(z). \quad (6)$$

In (6), $\Pi_{M_k}$ denotes the orthogonal projection map onto $M_k$, defined for $k = 1, \ldots, K$ as

$$\Pi_{M_k} : \{H \mapsto M_k, z \mapsto \arg\min_{x \in M_k} \|z - x\|, \text{for some norm $\|\|$ on $\mathcal{H}$}.$$ In the case of closed linear subspaces $\{M_1, \ldots, M_K\}$, von Neumann and Halperin showed that [18], [19], [20]

$$\lim_{n \to \infty} \left\| \Pi_{M_K} \cdots \Pi_{M_1} (z) - \Pi_{\bigcap_{k=1}^K M_k}(z) \right\| = 0, \quad \forall z \in \mathcal{H}. \quad (7)$$

The MAP equation (6) can therefore be used to approximate the complex projection map $\Pi_{\bigcap_{k=1}^K M_k}$. For closed convex sets $\{M_1, \ldots, M_K\}$, Bregman [17] showed moreover the weak convergence of the MAP towards a point in the intersection $\bigcap_{k=1}^K M_k$. Strong convergence towards the actual projection was achieved by Dysktra’s MAP [21], one of the most popular variant to von Neumann’s original algorithm. In the case of non convex intersecting sets, the convergence of the MAP has only been established locally [22], [23], [24], [25]. For example, Anderson et al. considered in [24] the case of two (potentially non convex) finite-dimensional manifolds $M_1, M_2 \subset \mathcal{H}$ and showed the following local convergence result:

**Theorem 1.** [24, Theorem 1.6] Let $x \in M_1 \cap M_2$ be non-tangential, i.e. the angle between $M_1$ and $M_2$ at $x$ is positive.\(^4\) Then, for $z \in \mathcal{H}$ and $\epsilon > 0$, there exists $\delta \geq 0$ such that, if $\|x - z\| \leq \delta$,

1. $\left\| \Pi_{M_1} \Pi_{M_2} (z) - z_0 \right\| \to z_0 \in M_1 \cap M_2$,
2. $\left\| z_0 - \Pi_{M_1} \Pi_{M_2} (z) \right\| < \epsilon \left\| x - (\Pi_{M_1} \Pi_{M_2} (z)) \right\|.$

Roughly speaking, Theorem 1 states that if the starting point $z$ is close enough to a non-tangential point of $M_1 \cap M_2$ (which as explained in [24] are all but very exceptional points of $M_1 \cap M_2$), then the MAP converges to a point in $M_1 \cap M_2$. Moreover, the error $\|z_0 - \Pi_{M_1} \Pi_{M_2} (z)\|$ can be made arbitrarily small with respect to $\|x - \Pi_{M_1} \Pi_{M_2} (z)\|$. However, Theorem 1 is difficult to apply in practice since the value of $\delta$ guaranteeing a relative error below a given threshold $\epsilon$ is unknown. The MAP is hence often used as a heuristic in non convex settings with no convergence guarantees. This is notably the case of Cadzow denoising, discussed further in Section II-E.

D. FRI in a Nutshell

The classical FRI framework, introduced in [4], aims at estimating the innovations $\{(t_k, x_k), k = 1, \ldots, K\} \subset \mathbb{C} \times \mathbb{R}$, of a $T$-periodic stream of Diracs:

$$x(t) = \sum_{k \in \mathbb{Z}} \sum_{i=1}^K x_k \delta(t - t_k - Tk), \quad \forall t \in \mathbb{R}. \quad (8)$$

In standard FRI, the estimation procedure is divided into two stages. The locations $t_k$ are first estimated by a nonlinear method, and then arranged into a Vandermonde system whose solution yields the Dirac amplitudes [5]. The recovery of the locations $t_k$ relies on the so-called annihilating equation, dating

---

\(^4\)See [24, Definition 4.2] and [24, Definition 4.3] for a precise definition of the angle between two manifolds and the concept of non-tangentiality.
from Prony’s work [26], which cancels out the Fourier series coefficients of $x$ by convolving them with a particular filter, called the annihilating filter. The latter is defined as the finite-tap sequence $h = [-\cdots, 0, h_0, h_1, \ldots, h_K, 0, \cdots] \in \mathbb{C}^N$, with $z$-transform vanishing at roots $\{u_k := e^{-j2\pi n/T}, k = 1, \ldots, K\}$:

$$H(z) = \sum_{k=0}^{K} h_k z^{-k} = \prod_{k=1}^{K} (1 - u_k z^{-1}). \quad (8)$$

For such a filter, we have indeed

$$(\hat{x} * h)_m = \sum_{k=0}^{K} h_k \hat{x}_{m-k} = \sum_{k=1}^{K} x_k \left( \sum_{h=0}^{K} h_k u_k^m \right) u_k^m = 0, \ m \in \mathbb{Z}, \quad (9)$$

where $\hat{x}_m = \sum_{k=1}^{K} x_k u_k^m, m \in \mathbb{Z}$, are the Fourier coefficients of $x$ in (1). Notice that the roots $u_k$ of the $z$-transform $H(z)$ in (8) of $h$ are in one-to-one correspondence with the locations $t_k$. Recovering them amounts to estimating the coefficients $h = [h_0, \ldots, h_K] \in \mathbb{C}^{K+1}$ of $h$ from the annihilating equation (9). If for instance we have $N = 2M + 1$ consecutive Fourier coefficients of $x$, e.g. $x = [\hat{x}_M, \ldots, \hat{x}_1] \in \mathbb{C}^{2M+1}$, we can extract the $N - K$ equations from (9) corresponding to the convolution indices $m = M + K, \ldots, M$, and use the Toeplitzification operator$^5$ defined in (2) to form the following matrix equation:

$$T_K(x)h = 0_{N-K}, \quad \|h\| \neq 0. \quad (10)$$

Observe that any nontrivial element of the nullspace of $T_K(x)$ is a solution to (10). For $M \geq K$, it can be shown [5] that $T_K(x) \in \mathbb{C}^{(N-K) \times (K+1)}$ has rank $K$ and therefore has a nontrivial nullspace with dimension 1. Up to a multiplicative constant, the annihilating equation (10) admits hence a unique solution. The latter can be obtained numerically by means of total least-squares [5], which computes the eigenvector associated to the smallest$^6$ eigenvalue of $T_K(x)$. In the critical case $M = K$, the matrix $T_K(x)$ is square, while in the oversampling case $M > K$ it is rectangular and tall. As explained in [5], oversampling makes the estimation procedure more resilient to potential noise perturbations in the Fourier coefficients. In such cases, Blu et al. recommend moreover to perform Cadzow denoising on the Fourier coefficients $x$ (see Section II-E) as well as replace (10) by a more general annihilating equation:

$$T_P(x)\hat{h} = 0_{N-P}, \quad \|\hat{h}\| \neq 0. \quad (11)$$

with $K \leq P \leq M$, and $\hat{h} \in \mathbb{C}^{P+1}$. Again, it is possible to show that $T_P(x)$ has rank $K$, and hence a nontrivial nullspace with dimension $P + 1 - K$. Solutions to (11) are hence not unique in this case, but all are equally valid for practical purposes. Moreover, the increased nullspace dimension makes Cadzow denoising more efficient at filtering the noise component (see Section II-E hereafter). In practice, the case $P = M$ has been reported to yield the best empirical performance [5].

$^5$Remember the link between the Toeplitzification operator and convolution discussed in Section II-A.

$^6$An eigenvalue exactly equal to zero may in practice be impossible to obtain due to numerical inaccuracies.

E. Cadzow Denoising

For strong noise perturbations, the generalised annihilating equation (11) may fail to admit a nontrivial solution. Indeed, noisy generators $x$ can yield full column rank matrices $T_P(x)$ with trivial nullspace. As a potential cure, Blu et al. propose to denoise the Fourier coefficients $x$ prior to solving the annihilating equation. This denoising step attempts to transform $T_P(x)$ into a Toeplitz matrix with rank at most $K$, thus guaranteeing the existence of nontrivial solutions to (11). This operation is carried out by means of Cadzow denoising [7], an alternating projection method (see Section II-C) applied heuristically to the subspace $\mathcal{H}_P$ of Toeplitz matrices and the subset $\mathcal{H}_K$ of matrices with rank at most $K$:

$$\mathcal{H}_K := \left\{ M \in \mathbb{C}^{N-P \times (P+1)} \mid \text{rank} \ M \leq K \right\}. \quad (12)$$

Using the notation introduced in Sections II-A, II-B and II-C, Cadzow denoising can be seen as processing the noisy coefficients $x$ as follows:

$$\hat{x} = T_P^\dagger [\Pi_{\mathcal{H}_P}, \Pi_{\mathcal{H}_K}]^\dagger T_P(x), \quad (13)$$

for some suitable $n \in \mathbb{N}$. The inverse Toeplitzification operator $T_P^\dagger$ applied after the $n$ iterations of the alternating projection method is used to recover the denoised Fourier coefficients $\hat{x} \in \mathbb{C}^N$.

Equation (13) allows us to develop an intuitive understanding as to why Cadzow denoising tends to perform better in practice with values of $P$ close to $M$ in the generalised annihilating equation (11). Indeed, it is easy to see that the maximal rank of rectangular matrices in $\mathbb{C}^{(N-P) \times (P+1)}$ ranges in $[K+1, M+1]$ when $P$ ranges in $[K, M]$. Consequently, the subset $\mathcal{H}_K$ of matrices of rank at most $K$ becomes “smaller and smaller” relatively to the ambient space as $P$ increases towards $M$. This implies that the associated projection map $\Pi_{\mathcal{H}_K}$ is more selective$^8$ for large values of $M$, hence making Cadzow’s algorithm better at filtering the noise component.

Note that since $\mathcal{H}_K$ is a non convex set the convergence of the MAP in (13) is not guaranteed. Nevertheless, experimental results [5, 7] suggest that Cadzow denoising almost always converges after a few iterations (typically $n \leq 20$), which could theoretically$^9$ be explained by the local convergence result from Theorem 1. We conclude this section by providing closed-form expressions for the projection operators $\Pi_{\mathcal{H}_P}$ and $\Pi_{\mathcal{H}_K}$, needed in (13).

1) Projection onto $\mathcal{H}_P$: As shown in Appendix D of the supplementary material, the orthogonal projection operator onto the subspace $\mathcal{H}_P \subset \mathbb{C}^{(N-P) \times (P+1)}$ of rectangular Toeplitz matrices can be written in terms of the Toeplitzification operator and its pseudoinverse (see Section 2) as:

$$\Pi_{\mathcal{H}_P} = T_P T_P^\dagger = T_P \Gamma^{-1}_P T_P^\dagger.$$

$^7$For $n, m \in \mathbb{Z}, n < m$, we denote by $[n, m]$ the integer interval $[n, m] \cap \mathbb{Z}$.

$^8$This can be seen from the closed-form expression of $\Pi_{\mathcal{H}_K}$ provided in (14).

$^9$As explained in Section II-C, the assumptions of Theorem 1 are unfortunately very difficult to verify in practice.
2) Projection onto $\mathcal{H}_K$: The orthogonal projection operator onto the space $\mathcal{H}_K$ of matrices with rank at most $K$ is given by the Eckart-Young-Minsky theorem [27]. The latter states that the projection matrix

$$
\Pi_{\mathcal{H}_K}(X) = \arg\min_{H \in \mathcal{H}_K} \|X - H\|_F, \quad X \in \mathbb{C}^{(N-P)(P+1)},
$$

can be computed in closed-form as:

$$
\Pi_{\mathcal{H}_K}(X) = U_K A_K V_H^H, \quad X \in \mathbb{C}^{(N-P)(P+1)},
$$

where $X = UAV^H$ is the singular value decomposition of $X$, and $A_K$ is the diagonal matrix of sorted singular values truncated to the $K$ strongest ones. Note that the output of the projection map is unique as long as the $K$-th and $(K+1)$-th largest singular values are different. Fortunately, the space of matrices failing to verify this condition is very small –more precisely, it is thin, as discussed extensively in [24, Section 2].

In practice moreover, floating-point arithmetic makes it very unlikely that the $K$-th and $(K+1)$-th largest singular values be exactly identical. Thus, the projection map $\Pi_{\mathcal{H}_K}$ can be considered single-valued for practical purposes.

III. GENERALISED FRI AS AN INVERSE PROBLEM

A. Generalised FRI

In Section II-D, we have described a procedure for recovering the locations $t_k$ from consecutive Fourier coefficients of $x$. The issue of computing these Fourier coefficients from a collection of arbitrary linear measurements $y \in \mathbb{C}^L$ of $x$, $L \geq N$ now remains. Blu et al. [5] treated the simple scenario of measurements resulting from regular time sampling with ideal low-pass prefiltering. In such a case, they showed that, for a well chosen prefilter bandwidth, the Fourier coefficients could simply be obtained by applying a discrete Fourier transform to the measurements $y$. For more general measurement types, the situation is more complex, and the Fourier coefficients $x \in \mathbb{C}^N$ must in general be estimated by solving a linear inverse problem:

$$
y = Gx + n, \quad (15)
$$

where the forward matrix $G \in \mathbb{C}^{L \times N}$, $L \geq N$, is application dependent, and $n$ is additive noise, usually assumed to be a white Gaussian random vector. In [9], Pan et al. proposed the generalised FRI (genFRI) optimisation problem to deal with (15). The latter is a non convex constrained optimisation problem whose objective is to jointly recover the Fourier coefficients $x \in \mathbb{C}^N$—required to minimise a quadratic data-fidelity term—and their corresponding annihilating filter coefficients $h \in \mathbb{C}^{P+1}$. The annihilating equation linking the two unknowns is explicitly enforced as a constraint, yielding an optimisation problem of the form:

$$
\min_{x \in \mathbb{C}^N} \|Gx - y\|_2^2 \quad \text{subject to} \quad T_P(x) h = 0_{N-P}, \quad (h, h_0) = 1, \quad (16)
$$

where $h_0 \in \mathbb{C}^{P+1}$ is generated randomly according to the circularly-symmetric complex Gaussian distribution $\mathcal{CN}(0, I_{P+1})$. The normalisation constraint\footnote{In [9], the authors have also considered the more natural normalisation constraint $|h| = 1$. They claim however that this normalisation strategy is less successful experimentally.} $(h, h_0) = 1$ is used to exclude trivial solutions to the annihilating equation in (16) [9], [28].

To solve (16), Pan et al. consider the corresponding Lagrangian for a fixed vector $h \in \mathbb{C}^{P+1}$ and show that, when $G$ is full column rank, critical solutions $x \in \mathbb{C}^N$ can be expressed in terms of the annihilating filter $h$ as in [9, Appendix A]:

$$
x = \xi - (G^H G)^{-1} R_P(h)^H \Omega^{-1}_h R_P(h) \xi, \quad (17)
$$

where $\xi := (G^H G)^{-1} G^H y$, $\Omega_h := R_P(h)(G^H G)^{-1} R_P(h)^H$ and $R_P$ is the right-dual of the Toeplitzification operator $T_P$, defined as in [9, Definition 1]:

$$
R_P : \mathbb{C}^{P+1} \to \mathbb{C}^{(N-P) \times N}, \quad \forall x \in \mathbb{C}^N. \quad (18)
$$

By applying (18) to the canonical basis of $\mathbb{C}^N$, it is easy to show that:

$$
R_P(h) = \begin{bmatrix} h_P & h_{P-1} & \cdots & h_0 & 0 & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & h_P & h_{P-1} & \cdots & h_0 \end{bmatrix},
$$

for $h = [h_0, \cdots, h_P] \in \mathbb{C}^{P+1}$.

Pan et al. then use (17) to saturate (16) with respect to $x$, obtaining an optimisation problem in terms of $h$ only:

$$
\min_{h \in \mathbb{C}^{P+1}} \langle \Omega_h^{-1} T_P(\xi) h, T_P(\xi) h \rangle \quad \text{subject to} \quad (h, h_0) = 1. \quad (19)
$$

Finally, they propose to solve (19) via an heuristic alternating minimisation algorithm. At each iteration, the annihilating filter is updated as the solution to the constrained quadratic minimisation problem:

$$
\arg\min_{h \in \mathbb{C}^{P+1}} \langle \Omega_h^{-1} T_P(\xi) h, T_P(\xi) h \rangle \quad \text{subject to} \quad (h, h_0) = 1, \quad n \geq 0. \quad (20)
$$

Observe that (20) corresponds to a relaxation of (19) where the unknown matrix $\Omega_h$ has been replaced by the fixed matrix $\Omega_{h_0}$, constructed from the previous estimate $h_0$. Pan et al. show moreover in [9, Appendix B] that the iterates can be computed efficiently by solving, at each iteration, a linear system of size $2(N+1) \times 2(N+1)$:

$$
\begin{bmatrix} 0 & T_P(\xi)^H & 0 & h_0 & h_{n+1} \\ T_P(\xi) & 0 & -R_P(h_n) & 0 & \ell \\ 0 & -R_P(h_n)^H & G^H G & v & 0 \\ h_n^H & 0 & 0 & 0 & \lambda \end{bmatrix} = \begin{bmatrix} 0 \\ \ell \\ v \\ 0 \\ 1 \end{bmatrix},
$$

where $\ell \in \mathbb{C}^{N-P}, v \in \mathbb{C}^N$ and $\lambda \in \mathbb{C}$ are auxiliary variables. In practice, the algorithm is stopped when the data mismatch $\|Gx_n - y\|_2$ falls below a certain threshold $\epsilon$ (typically the noise level), where $x_n \in \mathbb{C}^N$ is the estimate of the Fourier series coefficients at iteration $n$, obtained by plugging $h_n$ in (17). For greater numerical stability, Pan et al. recommend to compute $x_n$ by solving a linear system with size $(2N - P) \times (2N - P)$, equivalent to (17) [9, Appendix B]:

$$
\begin{bmatrix} G^H G & R_P(h_n)^H \\ R_P(h_n) & 0 \end{bmatrix} \begin{bmatrix} x_n \\ \ell \end{bmatrix} = \begin{bmatrix} G^H y \\ 0 \end{bmatrix}. \quad (22)
$$
Note that this heuristic iterative procedure comes without any strong or weak convergence guarantee: it is neither known if the sequence \( \{ (h_n, x_n), n \in \mathbb{N} \} \subset \mathbb{C}^{p+1} \times \mathbb{C}^N \) converges to a critical point of \((16)\), nor if \( \{ h_n, n \in \mathbb{N} \} \subset \mathbb{C}^{p+1} \) converges to a fixed-point of \((20)\). Moreover, the proposed stopping criterion requires knowing the noise level, unknown in practice. When it is unknown, Pan et al. recommend performing the reconstruction for a fixed and arbitrary number of iterations (typically fifty). For optimal performance, they moreover suggest to run the algorithm for multiple random initialisations of \( h_0 \in \mathbb{C}^{p+1} \) (typically fifteen). The overall reconstruction procedure can therefore be computationally intensive, since each iteration has complexity \( O(8(N+1)^3 + (2N - P)^3) = O(N^3) \) –the cost of solving the linear systems \((21)\) and \((22)\).

Remark (Choice of \( P \) and numerical stability). While theoretically well-defined for \( K \leq P \leq M \), the iterative procedure described above was only tested for \( P = K \) in \([9]\). This conservative choice was most likely motivated by numerical stability considerations: the inversion step in \( \xi = (G^H G)^{-1} G^H y \) can indeed be numerically unstable when \( G^H G \) is ill-conditioned, which is less likely to be the case for tall matrices \( G \) obtained with \( P = K \). We will touch on this again later.

### B. Implicit Generalised FRI

The annihilating equation constraint in \((16)\) can be thought of as regularising the genFRI problem. Indeed, minimising the quadratic term \( \| Gx - y \|_2^2 \) alone in the presence of noise would not necessarily yield Fourier coefficients \( x \) with nontrivial annihilating filter, which the annihilating constraint enforces explicitly. Unfortunately, this regularisation also complicates significantly the optimisation procedure. Indeed, it requires the introduction of an extra unknown variable with non linear dependency on the data, namely the annihilating filter \( h \). Moreover, the non linear constraint \( T_P(x) h = 0_{N-P} \) is highly non convex, and state-of-the-art algorithms, such as alternating minimisation or gradient descent \([10]\), may suffer from getting trapped in local minima\(^{11}\) \([29]\). To circumvent these issues, we propose the following implicit formulation of the genFRI problem, in which only the Fourier coefficients are recovered:

\[
\min_{x \in \mathbb{C}^N} \| Gx - y \|_2^2 \quad \text{subject to} \quad \begin{cases} \text{rank } T_P(x) \leq K, \\ \| x \|_F \leq \rho, \end{cases} \tag{23}
\]

where \( K \leq P \leq M , \rho \in [0, +\infty) \), and \( \| x \|_F \) is the norm induced by the diagonal and positive definite matrix \( \Gamma \in \mathbb{C}^{N \times N} \) in \((5)\):

\[
\| x \|_F := \sqrt{\Gamma \hat{X}}, \quad \forall x \in \mathbb{C}^N. \tag{24}
\]

Similarly to \((16)\), the quadratic term \( \| Gx - y \|_2^2 \) in \((23)\) is used to guarantee high fidelity of the recovered coefficients to the observed data. Unlike \((16)\), \((23)\) leverages a regularising rank constraint on \( T_P(x) \) which does not explicitly involve the unknown annihilating filter. As already discussed in Section II-E in the context of Cadzow denoising, requiring \( T_P(x) \) to be of rank at most \( K \) is indeed a sufficient condition for the generalised annihilating equation \((11)\) to admit nontrivial solutions. This implicit regularisation greatly simplifies the genFRI problem, since it decouples the problem of estimating the Fourier coefficients from the problem of estimating the annihilating filter. The normalisation constraint \( \| x \|_F \leq \rho \) enforces finite weighted energy \((24)\) to the recovered Fourier coefficients. As shall be seen in Section IV, it can be relaxed when the forward matrix \( G \) is injective by setting \( \rho = +\infty \). Indeed, it is only used to ensure coercivity in underdetermined cases where the forward matrix \( G \) has a nontrivial null space. Coercivity is indeed a key assumption \([30]\) for the convergence of the proximal gradient descent method envisioned in Section IV-A. We conclude this section by noting that the choice of \( \Gamma \) as weighting matrix in the energy normalisation constraint is arbitrary and purely motivated by computational considerations. Indeed, any choice of positive definite weighting matrix in \((24)\) would have been suitable for the sole purpose of making the objective functional coercive. As explained in Section IV-B however, defining the weighting matrix as \( \Gamma \) greatly simplifies the computations involved at each iteration of the numerical solver proposed in Section IV-A.

Remark (On the choice of \( P \)). With similar developments to Section II-E for Cadzow denoising, it is possible to show that the set of matrices in \( \mathbb{C}^{(N-P)\times (P+1)} \) of rank at most \( K \) becomes “smaller and smaller” with respect to the ambient space as \( P \) grows towards \( M \). As a result, the rank constraint in \((23)\) is more selective for values of \( P \) close to \( M \), hence enforcing a stronger regularisation. We can hence expect \((23)\) to perform better in practice for \( P = M \). This is in contrast with the explicit generalised FRI problem \((16)\), whose equality constraint is equally stringent for different values of \( P \).

Remark (Case \( G = I \)). When \( G = I \), the optimisation problem \((23)\) becomes a simple denoising problem, which could therefore be used as an alternative to Cadzow denoising or its upgrade \([7]\).

### IV. Optimisation Algorithm

#### A. Non Convex Proximal Gradient Descent

The optimisation problem \((23)\) can be rewritten in unconstrained form as:

\[
\min_{x \in \mathbb{C}^N} \| Gx - y \|_2^2 + \iota_{\mathcal{H}_K}(T_P(x)) + \iota_{\mathcal{J}_\rho}(x), \tag{25}
\]

where \( \mathcal{H}_K \) is the non convex set of matrices with rank lower than \( K \) defined in \((12)\), \( \mathcal{J}_\rho := \{ x \in \mathbb{C}^N : \| x \|_F \leq \rho \} \) is the \( \Gamma \)-ball with radius \( \rho > 0 \), and \( \iota_{\mathcal{H}_K} : \mathbb{C}^{(N-P)\times (P+1)} \rightarrow [0, +\infty) \), \( \iota_{\mathcal{J}_\rho} : \mathbb{C}^N \rightarrow [0, +\infty) \) are indicator functions with domains \( \mathcal{H}_K \) and \( \mathcal{J}_\rho \), respectively. Observe that the unconstrained optimisation problem \((25)\) can be written as a sum between a convex and differentiable quadratic term

\[
F(x) := \| Gx - y \|_2^2, \quad x \in \mathbb{C}^N,
\]

and a non convex and non differentiable term

\[
H(x) := \iota_{\mathcal{H}_K}(T_P(x)) + \iota_{\mathcal{J}_\rho}(x), \quad x \in \mathbb{C}^N.
\]

It is moreover easy to see that the gradient of \( F \)

\[
\nabla F(x) = 2G^H(Gx - y), \quad x \in \mathbb{C}^N, \tag{26}
\]
is $\beta$-Lipschitz continuous with respect to the $\Gamma$-norm (24), with Lipschitz constant given by

$$\beta = 2\|G^H G\|_F$$

$$= \sup \{ 2 \|G^H Gx\|_F : x \in \mathbb{C}^N, \|x\|_F = 1 \}$$

$$= \sup \{ \|r^{1/2} G^H GT^{-1/2} \|_2 : \hat{x} \in \mathbb{C}^N, \|\hat{x}\|_2 = 1 \}$$

$$= 2\|r^{1/2} G^H GT^{-1/2} \|_2.$$  (27)

It is hence possible to optimise (25) by means of proximal gradient descent (PGD) [10], an iterative method alternating between gradient and proximal steps according to the following update equation:

$$x_{k+1} \in \text{prox}_{\tau H}^\Gamma (x_k - \tau \nabla F(x_k)),$$  (28)

for $k \geq 0$, $x_0 \in \mathbb{C}^N$, $\tau > 0$ and $\text{prox}_{\tau H}^\Gamma$ defined in (29). Given a current estimate $x_k \in \mathbb{C}^N$, the update equation (28) decreases the value of the objective function (25) by selecting a proximal point [10] –with respect to $H$—of a target located at a distance $\tau$ from $x_k$ along the direction of steepest descent $-\nabla F(x_k)$. The operator mapping a point $x \in \mathbb{C}^N$ to its proximal points with respect to $H$ is called proximal operator, and is defined as [10]

$$\text{prox}_{\tau H}^\Gamma (x) : \{ \mathbb{C}^N \rightarrow 2^\mathbb{C}^N \},$$

$$x \mapsto \arg \min_{z \in \mathbb{C}^N} \frac{1}{2\tau} \|x - z\|^2_2 + \tau H(z),$$  (29)

where $\mathbb{P}(\mathbb{C}^N)$ is the power set of $\mathbb{C}^N$, and $\tau > 0$ controls the relative importance of $H$ with respect to the squared distance to $x$ measured in terms of the $\Gamma$-norm (24). The function $H$ being non convex, the proximal operator (29) will in general return multiple proximal points, which can all be used interchangeably in (28). The convergence of the sequence $\{x_k\}_{k \in \mathbb{N}}$ of PGD iterates (28) towards critical points of (25) is established by the following theorem.

**Theorem 2 (Convergence of PGD for Arbitrary $G$). Assume that $\rho \in [0, +\infty]$ in (25), and $\tau < 1/\beta$ with $\beta$ defined in (28). Then, any limit point $x_*$ of the sequence $\{x_k\}_{k \in \mathbb{N}}$ generated by (28) is a local minimum of (25).**

**Proof.** The proof of this theorem is adapted from [30, Theorem 1] and given in Appendix E available as supplementary material of this manuscript. □

As stated by Theorem 3 hereafter, the convergence of PGD furthermore extends to the case $\rho = +\infty$, at least for injective forward matrices $G$. Setting $\rho = +\infty$ in (23) is equivalent to dropping the energy normalisation constraint, since $\|x\|_F \leq +\infty$ is trivially verified and hence the associated indicator function $\iota_{\mathbb{B}_\rho}$ in (25) is always null.

**Theorem 3 (Convergence of PGD for Injective $G$). Assume that $\rho = +\infty$ in (25), $\tau < 1/\beta$ with $\beta$ defined in (28), and $G \in \mathbb{C}^{L \times N}$ in (25) is injective, i.e. ker($G$) = $\{0_N\}$. Then, any limit point $x_*$ of the sequence $\{x_k\}_{k \in \mathbb{N}}$ generated by (28) is a local minimum of (25).**

**Proof.** The proof of this theorem is given in Appendix E available as supplementary material of this manuscript. □

A practical implication of Theorem 3 is that, for injective forward matrices $G$, PGD applied to the following relaxed implicit genFRI problem is convergent:

$$\min_{x \in \mathbb{C}^N} \|Gx - y\|^2_F + \iota_{H_k} (T_P (x)),$$  (30)

where $F(x) := \|Gx - y\|^2_F$, and $H(x) := \iota_{H_k} (T_P (x))$. As discussed in Section IV-B, (30) should always be favoured over (25) for injective forward matrices $G$, since solving it via PGD requires less computations at each proximal step.

**B. Cadzow PnP Gradient Descent**

As seen in the previous section, PGD requires the computation of the proximal operator (29) at each iteration, which amounts to finding a minimiser to the following non convex optimisation problem:

$$\tilde{x} \in \arg \min_{z \in \mathbb{C}^N} \left( \frac{1}{2\tau} \|x - z\|^2_F + \iota_{H_k} (T_P (z)) + \iota_{\mathbb{B}_\rho} (z) \right),$$  (31)

for some input $x \in \mathbb{C}^N$. Observe that the proximal step (31) can be seen as a generalised projection step, aiming to find a point $\tilde{x}$ as close as possible—in terms of the $\Gamma$-norm— from $x$ while verifying some convex and non convex constraints specified by the indicator functions. This is formalised by Proposition 3, which shows that solutions to (31) can be identified with those of an orthogonal projection problem:

**Proposition 3. The proximal operator (29) of $H(x) := \iota_{H_k} (T_P (x)) + \iota_{\mathbb{B}_\rho} (x)$, for $\rho \in [0, +\infty]$ and $K \leq P \leq M$ is given by**

$$\text{prox}_{\tau H}^\Gamma (x) = T_P \Pi_{T_P \cap \mathcal{H}_K \cap \mathbb{B}_\rho} (x), \quad \forall x \in \mathbb{C}^N,$$  (32)

where $\mathbb{B}_\rho := \{ X \in \mathbb{C}^{N-P \times (P+1)} : \|X\|_F \leq \rho \}$ and $\Pi_{T_P \cap \mathcal{H}_K \cap \mathbb{B}_\rho}$ is the orthogonal projection operator onto $T_P \cap \mathcal{H}_K \cap \mathbb{B}_\rho$ with respect to the Frobenius norm:

$$\Pi_{T_P \cap \mathcal{H}_K \cap \mathbb{B}_\rho} (X) : \{ \mathbb{C}^{(N-P) \times (P+1)} \rightarrow 2^{\mathbb{C}^{(N-P) \times (P+1)}} \},$$

$$X \mapsto \arg \min_{H \in \mathbb{B}_\rho \cap \mathcal{H}_K} \|X - H\|_F.$$

**Proof.** The proof of this proposition is given in Appendix F available as supplementary material of this manuscript. □

Equation (32) provides us with a practical way of computing the proximal set (29) associated to a point $x \in \mathbb{C}^N$. Unfortunately, the orthogonal projection operator $\Pi_{T_P \cap \mathcal{H}_K \cap \mathbb{B}_\rho}$ admits no simple closed-form expression. We therefore propose to approximate it by the method of alternating projections (MAP) (see Section II-C):

$$\Pi_{T_P \cap \mathcal{H}_K \cap \mathbb{B}_\rho} \approx \left[ \Pi_{T_P} \Pi_{H_k} \Pi_{\mathbb{B}_\rho} \right]^n,$$  (33)

for some $n \in \mathbb{N}$. Observe that when $\rho = +\infty$ (which is possible for injective matrices $G$, see Theorem 3) we have $\Pi_{\mathbb{B}_\rho} = \text{Id}$ and the right-hand side of (33) simplifies to $\left[ \Pi_{T_P} \Pi_{H_k} \right]^n$. Note that since $\mathcal{H}_K$ is non convex, the convergence as $n$ grows to infinity of the product $\left[ \Pi_{T_P} \Pi_{H_k} \Pi_{\mathbb{B}_\rho} \right]^n$ towards the actual projection map $\Pi_{T_P \cap \mathcal{H}_K \cap \mathbb{B}_\rho}$ is not guaranteed in general (see discussion

12Observe that the weighting matrix $\Gamma$ puts more emphasis on the coefficients that appear more often in the Toeplitz matrix $T_P(x)$.  

SIMEONI, BESSON, HURLEY & VETTERLI: CADZOW PNP GRADIENT DESCENT FOR GENERALISED FRI 7
Algorithm 1 Cadzow PhP Gradient Descent (CPGD)

Require: \( y, G, T_p, x_0, K \leq P, \tau < 1/\beta, n \in \mathbb{N}, \rho > 0 \)

\( k:=0 \)

repeat

\( z_{k+1} := x_k - 2\tau G^H (G x_k - y) \)

if \( \rho = +\infty \) then

\( x_{k+1} := T_p \left[ \Pi_{T_p} \Pi_{H_K} \right]^n T_p (z_{k+1}) \)

else

\( x_{k+1} := T_p \left[ \Pi_{T_p} \Pi_{H_K} \Pi_{\beta_p} \right]^n T_p (z_{k+1}) \)

end if

\( k \leftarrow k + 1 \)

until a stopping criterion is satisfied

return \( x^{(k)} \)

in Section II-C). For the specific case \( \rho = +\infty \) however, it is possible to apply Theorem 1 to show the local convergence of the MAP (33):

**Corollary 1.** Let \( Z \in H_K \cap T_p \) be a non tangential point [24, Definition 4.3]. Then, for \( x \in \mathbb{C}^{(N-P)\times(P+1)} \) and \( \varepsilon > 0 \), there exists \( \delta > 0 \) such that, if \( \|X - Z\| \leq \delta \),

1. \( \|X - \Pi_{T_p} \Pi_{H_K} \| \to X_{\infty} \in H_K \cap T_p \)
2. \( \|X_{\infty} - \Pi_{H_K} \Pi_{T_p} \| \leq \varepsilon \|X - \Pi_{H_K} \Pi_{T_p} \| \)

Proof. Similarly to the proof of [31, Theorem 7], Corollary 1 is obtained by applying Theorem 1 to the manifolds \( M_1 = \mathbb{R}_K \) of matrices with rank exactly \( K \)--which is dense in \( H_K \) [24, Proposition 2.1]--and \( M_2 = T_p \). For more details, see the proof of [31, Theorem 7], which discusses the local convergence of the MAP for \( H_K \cap T_p \) where \( T_p \) denotes the space of rectangular Hankel matrices. Since Hankel matrices are just reflected Toeplitz matrices, the analysis extends to the case of Toeplitz matrices.

Roughly speaking, Corollary 1 states that, if applied to a matrix \( X \) close enough to a non tangential point of \( T_p \cap H_K \) (as discussed in [24] for the case of Hankel matrices are all but very exceptional matrices of \( T_p \cap H_K \)), the MAP (33) converges to a point in \( T_p \cap H_K \). Moreover, the error \( \|X_{\infty} - \Pi_{H_K} \Pi_{T_p} \| \) can be made arbitrarily small with respect to \( \|X - \Pi_{H_K} \Pi_{T_p} \| \). While difficult to verify in practice, the local convergence result Corollary 1 reassures us on the well-foundness of approximation (33).

Plugging (33) into (32) finally yields the following approximate proximal step:

\[ \text{prox}_T^G(x) = T_p \left[ \Pi_{T_p} \Pi_{H_K} \Pi_{\beta_p} \right]^n T_p (x), \forall x \in \mathbb{C}^n, \]

for some \( n \geq 0 \). The PGD algorithm with approximate proximal step (34) is provided in Algorithm 1. Observe that when \( \rho = +\infty \), (34) reduces to Cadzow denoising (13). The effect of heuristic (33) is hence to replace the proximal step in the PGD iterations by a generic denoising step. Such an approach is reminiscent of the plug-and-play (PhP) framework [15, 14] from image processing, which leverages generic denoisers to approximate complex projection or proximal operators [13]. For this reason, we baptise our algorithm *Cadzow PhP Gradient Descent (CPGD)*. In the next section, we study the convergence of Algorithm 1.

### C. Local Fixed-Point Convergence of CPGD

In Section IV-A, we established Theorems 2 and 3 which show the convergence of PGD towards critical points of (25). However, such results required the computation of exact proximal steps (29) in the PGD iterations, and do not apply to CPGD which leverages the inexact proximal step (34). Convergence of PGD in non convex setups with inexact proximal steps was studied in [12, 32]. The results established in both papers require the proximal step approximation errors incurred at each iteration to be decreasing and summable, which may not necessarily be the case for the MAP approximation (33). It is nevertheless possible to demonstrate that the iterations of CPGD are locally contractive, and therefore locally convergent towards a fixed point using the Banach contraction principle. Such a result is stated in Theorem 4 hereafter.

**Theorem 4 (CPGD is a Local Contraction).** Let \( R_K \subset \mathbb{C}^{(N-P)\times(P+1)} \) be the set of matrices of rank exactly \( K \leq P \) and \( U_{\tau,n} : \mathbb{C}^n \to \mathbb{C}^n \) the update CPGD map

\[ U_{\tau,n}(x) := H_n (x - \nabla \Phi (x)), \quad x \in \mathbb{C}^n, \]

with \( H_n (x) := T_p \left[ \Pi_{T_p} \Pi_{H_K} \Pi_{\beta_p} \right]^n T_p (x) \). Let \( G \in \mathbb{C}^{L \times N} \) be injective, and \( \Gamma \) be the diagonal and positive definite matrix defined in (5). Define

\[ \alpha := 2 \lambda_{\min} \left( \Gamma^{1/2} G^H G \Gamma^{-1/2} \right), \]

\[ \beta := 2 \lambda_{\max} \left( \Gamma^{1/2} G^H G \Gamma^{-1/2} \right), \]

where \( \lambda_{\min}(M) \) and \( \lambda_{\max}(M) \) denote the minimum and maximum eigenvalues of a matrix \( M \) respectively.

Then, \( U_{\tau,n} \) is locally well-defined (single-valued) and Lipschitz continuous with respect to the \( \Gamma \)-norm

\[ \|U_{\tau,n}(x) - U_{\tau,n}(x)\|_{\Gamma} \leq L_{\tau} \|x - z\|_{\Gamma}, \]

for all \( x, z \in \mathbb{C}^n \) such that \( T_p(x) \) and \( T_p(z) \) are in some neighbourhood of some matrix \( R \in R_K \). The Lipschitz constant \( L_{\tau} \) is given by

\[ L_{\tau} = \max \left( \|1 - \tau \alpha |, |1 - \tau \beta \| \right). \]

Moreover, \( U_{\tau,n} \) is contractive, i.e. \( 0 \leq L_{\tau} < 1 \), for \( 0 < \tau < 2/\beta \), and \( L_{\tau} \) is minimised for \( \tau = 2/(\alpha + \beta) \).

Proof. The proof of this theorem is given in Appendix G available as supplementary material of this manuscript.

The following corollary shows the local convergence of CPGD towards a fixed-point of the update map (35):

**Corollary 2 (CPGD Converges Locally).** With the same notations as in Theorem 4, assume that all CPGD iterates \( \{x_k\}_{k \in \mathbb{N}} \) are such that

\[ \{T_p(x_{k+1}), T_p(x_k)\} \subset U_k, \quad \forall k \in \mathbb{N}, \]

for some neighbourhood \( U_k \) of some matrix \( R_k \in R_K \). Assume further that \( 0 < \tau < 2/\beta \). Then, \( x_k \overset{k \to \infty}{\to} x_* \) where \( x_* \in \mathbb{C}^n \) is a fixed-point of \( U_{\tau,n} \), i.e. \( U_{\tau,n}(x_*) = x_* \). Moreover, we have

\[ \|x_* - x_k\| \leq \frac{L_{\tau}}{1 - L_{\tau}} \|x_1 - x_0\|, \quad \forall k \geq 1. \]
Corollary 2

\[ \text{From such a case, we have:} \]

enough step sizes, large enough for an illustration). This condition, although difficult to verify estimates to travel from one neighbourhood of the manifold is much less stringent than requiring the entire lifted path speaking, Corollary 2 other hand, show the global convergence of PGD with exact optimisation problem

Remark (Speed of Convergence). From Theorem 4 and (2), we see that the sequence \( \{x_k\}_{k \in \mathbb{N}} \) converges the fastest when \( L_r \) is minimised, i.e. \( \tau = 2/(\alpha + \beta) \).

Remark (Fixed Points vs. Critical Points). Note that Corollary 2 is a much weaker result than Theorems 2 and 3. Indeed, Corollary 2 only shows the local convergence of CPGD towards fixed points of \( \mathcal{U} \), which may not necessarily be critical points of the optimisation problem (25). Theorems 2 and 3 on the other hand, show the global convergence of PGD with exact proximal step towards critical points of (25). This is however the price to pay for computing the proximal step (31) efficiently in practice.

Remark (Geometric Interpretation of Condition (36)). Roughly speaking, Corollary 2 guarantees the convergence of CPGD towards a fixed point of the update map (35), provided that the forward matrix \( \mathbf{G} \) is injective, and that any two consecutive lifted estimates \( T_p(x_k), T_p(x_{k+1}) \), are in a common neighbourhood \( \mathcal{U}_k \) of some matrix \( \mathbf{R}_k \in \mathcal{R}_K \). Note that this is much less stringent than requiring the entire lifted path \( \{T_p(x_k)\}_{k \in \mathbb{N}} \) to belong to some neighbourhood \( \mathcal{U} \) of some fixed matrix \( \mathbf{R} \in \mathcal{R}_K \). Indeed, condition (36) allows the lifted estimates to travel from one neighbourhood of the manifold \( \mathcal{R}_K \) to another, provided that every visited neighbourhood contains at least two consecutive lifted estimates (see Figure 1 for an illustration). This condition, although difficult to verify in practice, seems however likely to hold for \( \rho = +\infty \), small enough step sizes, large enough \( n \) and \( x_0 = 0_N \). Indeed, in such a case, we have:

- \( T_p(x_0) \in \mathcal{H}_K \) is in some neighbourhood of \( \mathcal{R}_K \) since \( \mathcal{R}_K \) is dense in \( \mathcal{H}_K \).
- For \( n \) large enough, \( T_p(x_k) \) is very likely to be in some neighbourhood of \( \mathcal{R}_K \), since the denoising step in the update map (35) makes \( T_p(x_k) \) close to be in the intersection \( \mathcal{H}_K \cap T_p \) (see Corollary 1).
- For a small enough step size \( \tau \), \( T_p(x_k) \) and \( T_p(x_{k+1}) \) are likely to belong to the same neighbourhood of \( \mathcal{R}_K \).

V. Experiments and Results

In this section we validate the CPGD method numerically, considering as a testbed the scenario of irregular time sampling from [9, Section IV.A]. We assess both the reconstruction accuracy and the computational complexity of the method, and compare it to the state-of-the-art.

Remark (Reproducibility). Special care has been taken into making the experiments and simulations of this section fully reproducible. To reproduce the results, the reader is referred to the routines provided in our GitHub repository [16].

A. Reconstruction Accuracy

We define a 1-periodic stream of \( K = 9 \) Diracs (see Fig. 2):

\[
x(t) = \sum_{m \in \mathbb{Z}} \sum_{k=1}^{K} x_k \delta(t - t_k - m), \quad \forall t \in \mathbb{R},
\]

where the amplitudes \( x_k \in \mathbb{R}^+ \) and locations \( t_k \in [0, 1] \) are random, with log-normal and uniform distributions respectively. We then generate \( L = 73 \) noisy samples as

\[
y_l = \sum_{m=-M}^{M} \hat{x}_m e^{j2\pi mt_l} + \epsilon_l, \quad l = 1, \ldots, L,
\]

where \( \hat{x}_m = \sum_{k=1}^{K} x_k \exp(-j2\pi mt_k), \ m = -M, \ldots, M, \) are the Fourier coefficients of the Dirac stream \( x, \theta_l \in [0, 1] \) are chosen uniformly at random, and \( \epsilon_l \in \mathbb{R} \) are independent realisations of a Gaussian random variable \( \mathcal{N}(0, \sigma^2) \), for \( l \in \{1, L\} \).

As explained in [9, Section IV.A], the measurements \( y_l \) correspond to noisy samples of the low-pass filtered Dirac stream \( x \) at irregular times \( \theta_l \) (see Fig. 2), where the low-pass filter is chosen as an ideal low-pass filter with bandwidth \( 2M + 1 \leq L \). Using the formalism of Section III, we can rewrite (39) in vector notation as

\[
y = \mathbf{G}x + \epsilon,
\]

where \( y = [y_1, \ldots, y_L] \in \mathbb{C}^L, \ x = [\hat{x}_M, \ldots, \hat{x}_M] \in \mathbb{C}^{N = 2M + 1}, \ \epsilon = [\epsilon_1, \ldots, \epsilon_L] \in \mathbb{C}^L, \) and \( \mathbf{G} \in \mathbb{C}^{L \times N} \) is given by

\[
\mathbf{G} = \begin{bmatrix}
e^{-j2\pi M \theta_0} & \cdots & 1 & \cdots & e^{j2\pi M \theta_0} \\
e^{-j2\pi M \theta_1} & \cdots & 1 & \cdots & e^{j2\pi M \theta_1} \\
\vdots & \ddots & \vdots & \ddots & \vdots \\
e^{-j2\pi M \theta_{L-1}} & \cdots & 1 & \cdots & e^{j2\pi M \theta_{L-1}} \\
e^{-j2\pi M \theta_L} & \cdots & 1 & \cdots & e^{j2\pi M \theta_L}
\end{bmatrix}
\]

Note that from the periodicity of complex exponentials, it is possible to flip the columns of \( \mathbf{G} \) so as to rewrite it as a Vandermonde matrix [9]. This shows that \( \mathbf{G} \) is injective since \( 2M + 1 \leq L \) and the irregular time samples are all distinct. From the samples \( y \) and the data model (40), we consider recovering the Fourier coefficients \( x \in \mathbb{C}^N \) by means of three algorithms:

- The CPGD algorithm 1 with \( P = M, \ \rho = +\infty \) (since \( \mathbf{G} \) is injective) and step size \( \tau = 1.5/\beta \) (where \( \beta \) is set as described in Theorem 4).
- The state-of-the-art algorithm of Pan et al. [9], described in Section III-A and referred to hereafter as GenFRI.

13To avoid degenerate cases, the Diracs are required to have a minimum separation distance of 1% of the total period.

14To avoid degenerate cases, the sampling locations are required to have a minimal separation distance of 0.5% of the total period.

15The noise level is defined as the standard deviation \( \sigma \) of the Gaussian distribution.
For smooth integration, the Python 3 implementation of GenFRI provided by Pan et al. on their official Github repository [33] was included in our own algorithmic interface. Since the noise level is assumed to be unknown, we set \( \sigma \) as recommended in [9, Section III-C]—the number of inner iterations and random initialisations to their default values, 50 and 15 respectively.

The baseline method, referred to hereafter as LS-Cadzow, which consists in applying Cadzow denoising to the least-squares estimate of the Fourier coefficients

\[
\begin{aligned}
\{x_{LS}\} &= \underset{x \in \mathbb{C}^N}{\text{argmin}} \|Gx - y\|_2^2, \\
\{x_{LS, \text{Cadzow}}\} &= T_P^\dagger \left[ \Pi_I \Pi_H K \right] T_P (x_{LS}).
\end{aligned}
\] (41)

We solve the least-squares optimisation problem in (41) by means of the `linalg` function in the Python 3 package `scipy` [34], with cut-off ratio \( \text{cond} = 10^{-4} \).

For CPGD, we fix the maximum number of iterations\(^ {16}\) to 500 and consider that convergence is reached if the iterate norm is changed by less than 0.01\% between two iterations. For Cadzow denoising, we fix the number of iterations to 10 for both LS-Cadzow and CPGD. The reconstruction accuracy is assessed by matching the true Dirac locations \( t_k \) to the recovered ones, denoted by \( \omega_k \), for \( k \) between 1 and \( K \). To do so, we proceed as explained in Section II-D and infer the Dirac locations \( \omega_k \) from the z-transform roots of the annihilating filter associated to the Fourier coefficients estimated by each method.\(^ {17}\) Then, we solve the following matching problem by means of the Hungarian algorithm\(^ {18}\) [35]

\[
\begin{aligned}
\min_{j_1, \ldots, j_K \in \{1, \ldots, K\}} \left\{ \frac{1}{K} \sum_{k=1}^{K} d(t_k, \omega_{j_k}) \right\},
\end{aligned}
\] (42)

where \( d(t, \omega) = \min \{|t - \omega|, 1 - |t - \omega|\} \). \( \forall t, \omega \in [0, 1], \) is the canonical distance on the periodised interval [0,1]. Finally, we report the average positioning error, corresponding to the value of the cost function \( \sum_{k=1}^{K} d(t_k, \omega_{j_k})/K \) for the indices \( \{i_1, \ldots, i_K\} \) solutions to the matching problem (42). This metric is computed for 192 noise realisations, various cutoff frequencies \( M = \beta K \) with the oversampling factor \( \beta \in \{1, 2, 3, 4\} \) (see Figs. 2a, 2b, 2c and 2d respectively) and various noise levels

\[
\sigma = \max_{k=1, \ldots, K} |x_k| \times \exp \left( -\frac{\text{PSNR}}{10} \right),
\]

where the peak signal to noise ratio \( \text{PSNR} \) ranges from \(-30\) dB to \(30\) dB. The conditioning numbers of the matrix \( G^H G \) for the different values of the oversampling parameter \( \beta \) are provided in Table I. The results of the experiments are displayed on Figs. 3 and A—available as supplementary material of this manuscript. In Fig. 3 we plot, for different oversampling factors and PSNR, the median and inter-quartile region of the empirical distribution

\(^ {16}\)In practice this upper bound is never reached: CPGD almost always converges in less than 150 iterations.

\(^ {17}\)See [7, Fig. 2] for additional details on the procedure used to recover the Dirac locations from the annihilating filter coefficients.

\(^ {18}\)The Hungarian algorithm is implemented in the `linear_sum_assignment` function from the Python 3 package `scipy` [34].
we plot, for each source, different oversampling factors and
Table I: Conditioning number of
can be drawn from the results are the following:
conclusions that
methods against the true source location. The conclusions that
distribution of the source location as estimated by the three
PSNR, the median and inter-quartile region of the empirical
in the directory
Figure 3: Positioning error 
• 
−
Figure 3: Positioning error 
(a) $\beta = 1$, cutoff frequency $M = 9$
(b) $\beta = 2$, cutoff frequency $M = 18$
(c) $\beta = 3$, cutoff frequency $M = 27$
(d) $\beta = 4$, cutoff frequency $M = 36$

![Figure 3: Positioning error](image)

Figure 3: Positioning error (42) (in percent of period and log-scale) for LS-Cadzow, CPGD and GenFRI, various oversampling
parameters $\beta \in \{1, 2, 3, 4\}$ and a PSNR in $\{-30, -20, -10, 0, 10, 20, 30\}$ dB. For each case, plain lines and shaded areas represent
respectively the median and inter-quartile region of the positioning error’s empirical distribution obtained from 192 independent
noise realisations. These results can be reproduced using the Python script `reproduce_simulation_results.py` located
in the directory `/benchmarking/` of our GitHub repository [16].

| $\beta$ | $1$ | $2$ | $3$ | $4$
|---|---|---|---|---|
| $\kappa(G^H G)$ | 5.89 | 1.7×10$^2$ | 1.9×10$^5$ | 1.55×10$^{16}$

Table I: Conditioning number of $G^H G$ for various values of
the oversampling parameter $\beta$.

of the average positioning error of the three methods. In Fig. A,
we plot, for each source, different oversampling factors and
PSNR, the median and inter-quartile region of the empirical
distribution of the source location as estimated by the three
methods against the true source location. The conclusions that
can be drawn from the results are the following:

- Figs. 3a, 3a, Ab and Ac reveal that without oversampling
in the Fourier domain (i.e. $\beta = 1$ and a minimal cutoff
frequency of $M = K = 9$), the three algorithms CPGD,
GenFRI and LS-Cadzow perform similarly throughout
the entire PSNR range. The average positioning error –
almost indistinguishable for the three algorithms– goes
from approximately 10% of the period for PSNRs of
$-30$ dB to 1% of the period for PSNRs of $30$ dB.

- Figs. 3b, Ad, Ae and Af reveal that with an oversampling
of $\beta = 2$ –yielding a cutoff frequency of $M = 18$– the
three algorithms CPGD, GenFRI and LS-Cadzow start
behaving differently for PSNRs larger than 0 dB: CPGD
has the lowest positioning error, followed by LS-Cadzow
and finally GenFRI. The inter-quartile regions of the
positioning errors distribution are however overlapping,
which means that the differences in performance are not
statistically significant. For PSNRs larger than 0 dB, all
algorithms have a lower positioning error than in the case
$\beta = 1$. For high PSNRs, this improvement can be as high
as one and a half order of magnitude.

- Figs. 3c, Ag, Ah and Ai reveal that with an oversampling
of $\beta = 3$ –yielding a cutoff frequency of $M = 27$, the
three algorithms CPGD, GenFRI and LS-Cadzow
again behave differently for PSNRs larger than 0 dB:
CPGD has the lowest positioning error, followed by
GenFRI and finally LS-Cadzow. For high PSNRs, the
differences in performance among the three algorithms
become statistically significant: the inter-quartile regions
of the positioning error’s empirical distribution do not
overlap anymore. CPGD moreover reaches a positioning
error as low as 0.01% of the period, which is up to two
orders of magnitude smaller than the minimal positioning
error of GenFRI or LS-Cadzow in this scenario. For
PSNRs greater than 10 dB, CPGD improves its positioning
error with respect to the case $\beta = 2$ by a bit less than half
an order of magnitude. This is not the case for GenFRI
and LS-Cadzow which both underperform with respect
to the case $\beta = 2$ –and even with respect to the case $\beta = 1$ for LS-Cadzow. This can be explained by the large conditioning number of the matrix $G^H G$ in this case (see Table I), affecting the numerical stability of both algorithms (see remark in Section III-A).

- Figs. 3a, Aj, Ak and Al reveal that with an oversampling of $\beta = 4$ –yielding a critical bandwidth of $2M + 1 = 73$ equal to the number of measurements $L$, CPGD is superior to GenFRI which is itself superior to the baseline method LS-Cadzow in nearly all cases, with the exception of very low PSNRs ($\sim -30 \text{dB}$), where the three methods have comparable reconstruction accuracy. For PSNRs larger than $-20 \text{dB}$, the differences in performance are statistically significant. CPGD is more accurate than GenFRI and LS-Cadzow by a few orders of magnitude (from 1 to 3 orders of magnitude for PSNRs larger than $-10 \text{dB}$), reaching a minimal positioning error as low as 0.005\% of the period. For PSNRs greater than $-20 \text{dB}$, CPGD improves its positioning error with respect to all previous cases $\beta \in \{1, 2, 3\}$. Again, this is not the case for GenFRI and LS-Cadzow which both perform as good as or worse than the case $\beta = 1$. This can be explained by the (very) large conditioning number of the matrix $G^H G$ in this case (see Table I), which severely affects the numerical stability of both algorithms.

In conclusion, in these simulations CPGD is better at leveraging oversampling in the Fourier domain to improve the reconstruction accuracy by several orders of magnitude with respect to the non oversampled case. In particular, CPGD performs best when the bandwidth of the low-pass filter is chosen as large as the number of measurements. As explained in Section II-E, Cadzow denoising exhibits a similar behaviour. This similarity is not fortuitous: both algorithms leverage a similar rank constraint which becomes more and more selective as the oversampling parameter increases. In contrast, GenFRI and LS-Cadzow are negatively affected by large oversampling parameters, due to numerical stability issues. For the critical bandwidth $2M + 1 = L$, CPGD notably outperforms GenFRI and LS-Cadzow by one to three orders of magnitude, and this even for PSNRs as low as $-20 \text{dB}$.

B. Computational Complexity

As explained in Section III-A, solving the linear systems (21) and (22) are the most expensive operations performed at each iteration of GenFRI, yielding an overall computational complexity of $O(N^3)$. For CPGD, the computational cost of each iteration is dominated by the successive projections onto $\mathcal{H}_K$ in the approximate proximal step, which are computed via a SVD –see Algorithm 1 and (14). At a cursory glance, it may seem that the overall complexity of CPGD is somewhat comparable to the one of GenFRI, since computing the SVD of a matrix with size $(N-P)\times(P+1)$ has in general a computational complexity of $O((N-P)^2(P+1)+(P+1)^3)$ which reduces to $O(N^3)$ when $P = M$. In practice however, projecting onto $\mathcal{H}_K$ does not require to perform a complete SVD since only the $K$ strongest eigenvalues and their associated eigenvectors are needed. This truncated SVD can be performed very efficiently by means of the implicitly restarted Arnoldi method (IRAM) [37], or the implicitly restarted Lanczos method for Hermitian matrices. When $K \ll P + 1$, such methods are obviously much more efficient than a wasteful standard SVD. IRAM is moreover a matrix-free method [38]: it does not need the processed matrix to be stored in memory but simply requires an algorithm for performing matrix/vector products. In our context, since the truncated SVDs are exclusively performed on Toeplitz matrices, such matrix/vector products can be efficiently implemented by means of FFTs thanks to the convenient links between Toeplitz matrices and convolutions outlined in Section II-A.

The CPGD implementation provided in our Github repository [16] leverages all these computational tricks. In Fig. 4, we show that our implementation of CPGD is considerably faster than GenFRI. Fig. 4 reports the reconstruction times of CPGD and GenFRI for $K = 9$ and bandwidth $N = 2M + 1 = L$ with $\beta$ ranging from 1 to 300. To save computational time, the reconstruction times were scaled from the execution time of a single iteration of CPGD and GenFRI, assuming a typical number of iterations of 100 and $15 \times 50 = 750$ respectively. We observe that CPGD is always faster than GenFRI, sometimes by two orders of magnitudes. Moreover, regressions performed in log-log scale reveal that CPGD scales as $N^{2.11}$ while GenFRI scales as $N^{2.79}$. Note that this difference in scaling behaviour is overlooked by the complexity analysis above.

VI. CONCLUSION

We propose an implicit version of the generalised finite-rate-of-innovation (genFRI) problem for the recovery of the Fourier series coefficients of sparse Dirac streams with arbitrary linear sensing. This formulation relies on a novel regularisation term which enforces the annihilation of the recovered Fourier series coefficients without explicitly involving the unknown annihilating filter. The resulting non convex optimisation problem is consequently simpler and linear in the data. To solve it, we suggest a proximal gradient descent (PGD) algorithm which we prove converges towards a critical point of the objective function. We further introduce an inexact PGD method, coined

![Figure 4: Reconstruction times for CPGD and GenFRI for various bandwidth sizes $N$ and $K = 9$. The reported times are for a MacBook Pro (16-inch, 2019), Intel Core i7 (6C/12T) @ 2.6GHz with 32GB RAM. These results can be reproduced using the Python script reproduce_execution_times.py located in the directory /benchmarking/ of our GitHub repository [16].](image)
Cadzow plug-and-play gradient descent (CPGD), where the intractable proximal steps involved in PGD are approximated by means of alternating projections, akin to the popular Cadzow denoising algorithm. We outline the resemblance of CPGD to PnP methods used in image processing and prove its local fixed-point convergence under relatively weak assumptions. Considering the traditional irregular time sampling testbed, we demonstrate empirically that CPGD outperforms by several orders of magnitude the state-of-the-art GenFRI algorithm, both in terms of accuracy and reconstruction time.

For future work, we plan on investigating acceleration techniques for CPGD, such as approximate sketching-based eigenvalue decomposition methods [39], [40], more computationally efficient for large-scale problems. Applications of CPGD to acoustics and radio astronomy will also be investigated.

REFERENCES
Cadzow Plug-and-Play Gradient Descent for Generalised FRI

Supplementary Material

Matthieu Simeoni, Member, IEEE, Adrien Besson, Associate Member, IEEE, Paul Hurley, Senior Member, IEEE, and Martin Vetterli, Fellow, IEEE

CONTENTS

Appendix A: The Toeplitzification Operator and Convolutions
Appendix B: Proof of Proposition
Appendix C: Proof of Proposition
Appendix D: Projection on the Subspace of Toeplitz Matrices
Appendix E: Proofs of Theorems and
Appendix F: Proof of Proposition
Appendix G: Proof of Theorem
Appendix H: Proof of Corollary

References

APPENDIX A
THE TOEPLITZIFICATION OPERATOR AND CONVOLUTIONS

Consider the Toeplitzification operator defined in (2). When multiplied with a vector \( \mathbf{u} = [u_1, \cdots, u_{P+1}] \in \mathbb{C}^{P+1} \), the matrix \( T_P(x) \) returns the valid part of the convolution between the two zero-padded sequences:

\[
\hat{x} := \begin{bmatrix} \ldots, 0, x_{-M}, \ldots, x_0, \ldots, x_M, 0, \ldots \end{bmatrix} \in \mathbb{C}^Z
\]

and

\[
\hat{u} := \begin{bmatrix} \ldots, 0, u_1, \ldots, u_{P+1}, 0, \ldots \end{bmatrix} \in \mathbb{C}^Z.
\]

Indeed,

\[
(\hat{x} \star \hat{u})[k] := \sum_{j \in \mathbb{Z}} \hat{x}_{k-j} \hat{u}_j = \sum_{j=1}^{P+1} \hat{x}_{k-j} u_j, \quad k \in \mathbb{Z}.
\]

The valid part corresponds to the indices \( k \in \mathbb{Z} \) for which all the terms in the summation are nonzero. This is the case when

\[
k \in \{-M + P + 1, \ldots, M + 1\}.
\]

Consider \( i = k + M - P \) we get that the valid part of the convolution is given by

\[
(\tilde{x} \star \tilde{u})[i - M + P] = \sum_{j=1}^{P+1} x_{-M + P + i - j} u_j, \quad i = 1, \ldots, N - P,
\]

which corresponds precisely to \( T_P(x) \mathbf{u} \).

Using similar arguments, it is easy to show that the multiplication of \( T_P(x)^H \in \mathbb{C}^{(P+1) \times (N-P)} \) with a vector \( \mathbf{v} = [v_1, \ldots, v_{N-P}]^T \in \mathbb{C}^{N-P} \) returns the valid part of the cross-correlation

\[
\tilde{x} \star \tilde{v}[k] := \sum_{j \in \mathbb{Z}} \tilde{x}_{k-j} \tilde{v}_j, \quad k \in \mathbb{Z},
\]

between \( \tilde{x} \) and the zero-padded sequence

\[
\tilde{v} := \begin{bmatrix} \ldots, 0, v_1, \ldots, v_{N-P}, 0, \ldots \end{bmatrix} \in \mathbb{C}^Z.
\]

This time however, the valid part corresponds to the indices:

\[
k \in \{M + 1 - P, \ldots, M + 1\}.
\]

\[ \Box \]

APPENDIX B
PROOF OF PROPOSITION

Consider a matrix \( \mathbf{H} \in \mathbb{C}^{(N-P) \times (P+1)} \) and the following Frobenius inner product

\[
\langle T_P(x), \mathbf{H} \rangle_F = \text{tr}\left(T_P^H(x)\mathbf{H}\right)
\]

\[
= \sum_{i=1}^{N-P} \sum_{k=1}^{P+1} T_P(x)_{ik} H_{ik}
\]

\[
= \sum_{i=1}^{N-P} \sum_{k=1}^{P+1} \tilde{x}_{-M + P + i - k} H_{ik}
\]

\[
= \sum_{i=1}^{N-P} \sum_{s=0}^{N-1} x_{M + s} \left( \sum_{i=k+s-P}^{i+s} H_{ik} \right)
\]

The term \( \sum_{i=k+s-P}^{i+s} H_{ik} \) sums the elements of \( \mathbf{H} \) along lines with equation \( i = k + (s - P) \). These lines have slope 1 and intercept \( b = s - P \). Notice that these lines have nonnull intersection with
we have trivially:

\[ \Gamma = T^\dagger P \Gamma T \]  

\[ \Gamma^{-1}P \Gamma = (T^\dagger P)^{-1}T \Gamma T \]

The operator \( T^\dagger P \) is diagonal with diagonal entries given by:

\[ \Gamma_{i,j} = \begin{cases} i & \text{for } i \leq P \\ P + 1 & \text{for } P < i \leq N - P \\ N + 1 - i & \text{for } N - P < i \leq N. \end{cases} \]  

(\text{C.1})

The operator \( T^\dagger P \) is left inverse for \( T_P \):

\[ T^\dagger_P T_P = \Gamma^{-1}P \Gamma = (T^\dagger_P T^\dagger P)^{-1}T^\dagger P T_P = I_N. \]  

(\text{C.2})

Indeed, the two extreme cases occur when the lines hit the points \((1, N - P)\) and \((P + 1, 1)\). This happens respectively when \( 1 + b = N - P \Rightarrow b = N - P - 1 \) and \( P + 1 + b = 1 \Rightarrow b = -P \).

Since \( s \in [0, N - 1] \), the intercept \( b \) varies indeed in the range \([-P, N - P - 1]\) and each term in the summation is nonnull. The summation \( \sum_{i=k+1-P}^{N} H_{ik} \) corresponds then to summing across each diagonal of \( H \). We finally get:

\[ (T_P(x), H) = (x, T_P \Gamma (H)) \]

with

\[ T^\dagger_P : \mathbb{C}^{N-P} \times (P+1) \rightarrow \mathbb{C}^N \]

\[ H \mapsto h_j = \sum_{i=k+j-1-P}^{N} H_{ik}, \quad j = 1, \ldots, N. \]

\[ \Box \]

\section*{Appendix C}

\section{Proof of Proposition 2}

From (3) and the definition of \( T_P \), it is straightforward to observe that the operator \( \Gamma = T^\dagger_P T_P : \mathbb{C}^N \rightarrow \mathbb{C}^N \) is a diagonal matrix, with diagonal entries given by:

\[ \Gamma_{i,j} = \begin{cases} i & \text{for } i \leq P \\ P + 1 & \text{for } P < i \leq N - P \\ N + 1 - i & \text{for } N - P < i \leq N. \end{cases} \]

The operator \( T^\dagger_P \) is \( \Gamma^{-1} \) of \( T_P \) is hence a left inverse for \( T_P \):

\[ T^\dagger_P T_P = \Gamma^{-1}P \Gamma = (T^\dagger_P T^\dagger P)^{-1}T^\dagger P T_P = I_N. \]

Moreover, the latter is actually the pseudoinverse of \( T_P \). Indeed, we have trivially:

\[ T^\dagger_P T_P T_P = T_P, \quad T^\dagger_P T_P T^\dagger_P = T^\dagger_P, \quad (T^\dagger_P T_P)^* = T^\dagger_P T_P. \]

Finally, we have

\[ T^\dagger_P T_P \Gamma = T_P \Gamma^{-1} \Gamma = T_P \Gamma \]

since \( \Gamma \) is diagonal and hence symmetric. \( T^\dagger_P \) verifies thus the definition of the pseudoinverse of \( T_P \).

\[ \Box \]

\section*{Appendix D}

\section{Projection on the Subspace of Toeplitz Matrices}

The operator \( T_P \) is actually a surjection onto the subspace \( T_P \subset \mathbb{C}^{(N-P)\times(P+1)} \) of rectangular Toeplitz matrices with size \((N - P) \times (P + 1)\). Indeed, it is easy to see that every such matrix can be written as in (2) for some generator \( x \in \mathbb{C}^N \).

Moreover, we have from (C.2) that \( T^\dagger_P T_P = I_N \) and hence from [1, Theorem 2.29], \( T^\dagger_P T^\dagger_P \) is a projection operator onto the range \( T \) of \( T_P \). Since \( T^\dagger_P T^\dagger_P \) is moreover self-adjoint from (C.3), it is actually an orthogonal projection operator, which achieves the proof.

\[ \Box \]

\[ 1 \]

\[ \text{For } n, m \in \mathbb{Z}, n < m, \text{ we denote by } \mathbb{Z}^{n,m} \text{ the } 1D \text{ lattice } [n, m] \cap \mathbb{Z}. \]
Since $\tau < 1/\beta$ we have hence $(1/2\tau - \beta/2) \geq 0$ and 
\[ \Phi(x_{k+1}) = \Phi(x_k) = \Phi(x_0), \quad \forall k \geq 1. \]
The sequence $\{\Phi(x_k)\}_{k \in \mathbb{N}}$ is hence bounded and since $\Phi$ is 
coercive so is $\{x_k\}_{k \in \mathbb{N}}$. The sequence $\{x_k\}_{k \in \mathbb{N}}$ admits hence 
limit points. Moreover, since $\Phi(x_k)$ is decreasing and bounded from below, it takes the same value $\Phi_*$ at all of these 
limit points. Summing (E.4), we obtain hence:
\[ \left( \frac{1}{2\tau} - \frac{\beta}{2} \right) \sum_{k=0}^{\infty} \|x_{k+1} - x_k\|^2 \leq \Phi(x_0) - \Phi_* < +\infty. \]
Since $\tau < 1/\beta$, we have necessarily $\sum_{k=0}^{\infty} \|x_{k+1} - x_k\|^2 < +\infty$, which yields
\[ \lim_{k \to +\infty} \|x_{k+1} - x_k\| = 0. \quad (E.5) \]
From the optimality condition (E.3) and Items 1 and 3 of 
Proposition 1 of the supplementary material of [2], we have moreover
\[ 0_n \in \nabla F(x_k) + \frac{1}{\tau}(x_{k+1} - x_k) + \partial H(x_{k+1}) \]
\[ = \partial \Phi(x_{k+1}) - \nabla F(x_{k+1}) + \nabla F(x_k) + \frac{1}{\tau}(x_{k+1} - x_k), \quad (E.6) \]
where $\partial H : \mathbb{R}^n \to \mathcal{P}(\mathbb{R}^n)$ and $\partial \Phi : \mathbb{R}^n \to \mathcal{P}(\mathbb{R}^n)$ denote the (set-valued) subdifferential operators of $H$ and $\Phi$ respectively 
(see Definition 2 of the supplementary material of [2]).
Equation (E.6) can moreover be rewritten as
\[ \nabla F(x_{k+1}) - \nabla F(x_k) - \frac{1}{\tau}(x_{k+1} - x_k) \in \partial \Phi(x_{k+1}). \]
Furthermore, from the Lipschitz continuity of $F$, we have
\[ \|\nabla F(x_{k+1}) - \nabla F(x_k) - \frac{1}{\tau}(x_{k+1} - x_k)\| \leq \left( \beta + \frac{1}{\tau} \right) \|x_{k+1} - x_k\|, \]
and hence from (E.5):
\[ \lim_{\|x\| \to +\infty} \|\nabla F(x_{k+1}) - \nabla F(x_k) - \frac{1}{\tau}(x_{k+1} - x_k)\| = 0. \quad (E.7) \]
Let $(x_k)_{k \in \mathbb{N}}$ be a convergent subsequence of $(x_k)_{k \in \mathbb{N}}$, with 
limit $x_*$. Then, we have from (E.7) and Item 2 of Proposition 
1 of the supplementary material of [2]:
\[ 0_n \in \lim_{j \to +\infty} \partial \Phi(x_k) = \partial \Phi(x_*), \]
which completes the proof. \( \square \)
We now show Theorems 2 and 3 by applying Lemma 1 to 
The implicit genFRI problem in unconstrained form (25)
\[ \min_{x \in \mathbb{C}^N} \|Gx - y\|^2_{\mathbb{C}^L} + \iota_H^L(T_p(x)) + \iota_{\mathbb{C}^L}(x). \quad (E.8) \]
To do so, we must first convert (E.8) into an optimisation 
problem of the form (E.1), defined over $\mathbb{R}^n$ for some $n \in \mathbb{N}$. 
We achieve this by proceeding as in [3, Section 7.8] and 
identifying $\mathbb{C}^N$ with $\mathbb{R}^{2N}$ (respectively $\mathbb{C}^L$ with $\mathbb{R}^{2L}$) in the 
canonical way
\[ x \in \mathbb{C}^N \leftrightarrow \hat{x} := \begin{bmatrix} \Re(x) \\ \Im(x) \end{bmatrix} \in \mathbb{R}^{2N}, \]
where $\Re$ and $\Im$ denote the real and imaginary parts respectively. 
Such an identification makes the canonical inner products and 
norms on $\mathbb{C}^N$ and $\mathbb{R}^{2N}$ (respectively $\mathbb{C}^L$ and $\mathbb{R}^{2L}$) consistent 
with one another, i.e. for all $x, z \in \mathbb{C}^N$, we have
\[ \langle x, z \rangle_{\mathbb{C}^N} = z^H x = \Re(z)^T \Re(x) + \Im(z)^T \Im(x) = (\hat{x})^T \hat{z} \in \mathbb{R}^{2L^2}, \]
and
\[ \|x\|_{\mathbb{C}^N} = \sqrt{x^H x} = \sqrt{\|\Re(x)\|_{\mathbb{R}^L}^2 + \|\Im(x)\|_{\mathbb{R}^L}^2} = \|\hat{x}\|_{\mathbb{R}^{2L^2}}. \]
Still following [3, Section 7.8], we moreover identify the linear map $G : \mathbb{C}^N \to \mathbb{C}^L$ with a linear map $\hat{G} : \mathbb{R}^{2N} \to \mathbb{R}^{2L}$ with 
matrix representation:
\[ \hat{G} := \begin{bmatrix} \Re(G) \\ \Im(G) \end{bmatrix} \in \mathbb{R}^{2L \times 2N}. \]
Again, it is easy to show that the two operators are consistent, 
in the sense that
\[ \hat{G} x = \hat{G} \hat{x}, \quad \text{and} \quad \hat{G}^H x = \hat{G}^T \hat{x}, \quad \forall x \in \mathbb{C}^N. \]
Similarly, the Toeplitization operator $T_p : \mathbb{C}^N \to \mathbb{C}^{(N-P)\times (P+1)}$ is identified with the linear operator $\hat{T}_p : \mathbb{R}^{2N} \to \mathbb{R}^{2L \times (P+1)}$ defined as
\[ \hat{T}_p(x) := T_p(\Re(x)) + jT_p(\Im(x)), \quad \forall x \in \mathbb{C}^N, \]
where $j$ is the complex 2-root of unity. From the linearity of $T_p$, this definition yields indeed $T_p(x) = \hat{T}_p(\hat{x})$. Finally, the 
Gamma-ball $\Gamma_{\rho}^N \subset \mathbb{C}^N$ is identified with
\[ \hat{\Gamma}_{\rho}^N := \{ \hat{x} \in \mathbb{R}^{2N} : \|\hat{x}\|_{\mathbb{R}^L} \leq \rho \}, \]
where $\hat{\Gamma} \in \mathbb{R}^{2N \times 2N}$ is a positive definite and diagonal matrix defined as
\[ \hat{\Gamma} := \begin{bmatrix} \Gamma & 0_{N \times N} \\ 0_{N \times N} & \Gamma \end{bmatrix}. \]
Again, we trivially have $\|\hat{x}\| = \|x\|_{\mathbb{R}^L}$ and hence $x \in \hat{\Gamma}_{\rho}^N \iff \hat{x} \in \Gamma_{\rho}^N$ for all $x \in \mathbb{C}^N$.
In summary, the optimisation problem (E.8) is hence equivalent to 
the following optimisation problem with search space $\mathbb{R}^{2N}$:
\[ \min_{x \in \mathbb{R}^{2N}} \|\hat{G} x - y\|^2_{\mathbb{R}^{2L}} + \iota_{\hat{\Gamma}^N}(\hat{T}_p(\hat{x})) + \iota_{\mathbb{R}^{2L}}(x). \quad (E.9) \]
Letting $\hat{F}(\hat{x}) := \|\hat{G} \hat{x} - y\|^2_{\mathbb{R}^{2L}}$ and $\hat{H}(\hat{x}) := \iota_{\hat{\Gamma}^N}(\hat{T}_p(\hat{x})) + \iota_{\mathbb{R}^{2L}}(\hat{x})$ we have $\hat{F} : \mathbb{R}^{2N} \to \mathbb{R}_+$ and $\hat{H} : \mathbb{R}^{2N} \to \{0, +\infty\}$, so that (E.9) is indeed of the form (E.1). We must now verify assumptions (A), (B) and (C) of Lemma 1:
(A) $\hat{F}$ is proper, differentiable and $\nabla \hat{F}$ Lipschitz continuous. $\hat{F}$ is proper since
\[ \nabla \hat{F}(\hat{x}) = 2 \hat{G}^T(\hat{G} \hat{x} - y) = \nabla \hat{F}(x), \quad \hat{x} \in \mathbb{R}^{2N}. \quad (E.10) \]
The gradient (E.10) is moreover \( \hat{\beta} \)-Lipschitz continuous with respect to the norm \( \| \cdot \|_F \) on \( \mathbb{R}^{2N} \), and its Lipschitz constant is given by:

\[
\begin{align*}
\hat{\beta} &= 2\| \hat{G}^T \hat{G} \|_F \\
&= \sup \left\{ 2\| \hat{G}^T \hat{G} \hat{x} \|_F : \hat{x} \in \mathbb{R}^{2N}, \| \hat{x} \|_F = 1 \right\} \\
&= \sup \left\{ 2\| \hat{G}^T \hat{G} \hat{x} \|_F : x \in \mathbb{C}^N, \| x \|_F = 1 \right\} \\
&= \beta < +\infty.
\end{align*}
\]  

(B) \( \hat{H} \) is proper and lower semicontinuous. \( \hat{H} \) is proper since for all \( \rho > 0 \), and \( K \in \mathbb{N} \),

\[
\hat{H}(\theta_{2N}) = \iota_{\mathcal{H}_K}(\theta_{(N-P_{\infty}(P_{1}))}) + \iota_{\mathbb{B}_\rho}(\theta_{2N}) = 0 < +\infty.
\]

The indicator functions are moreover lower semicontinuous since the sets \( \mathcal{H}_K \) and \( \mathbb{B}_\rho \) are both closed. Since \( T_P \) is a bounded linear operator, it is continuous and hence \( \hat{H} \) is indeed lower semicontinuous as composition between continuous and lower semicontinuous functions.

(C) \( \hat{\Phi} = \hat{F} + \hat{H} \) is coercive. It is easy to see that \( \hat{\Phi} = \hat{F} + \hat{H} \geq 0 \). To show that \( \hat{\Phi} \) is coercive, it is hence sufficient to show that

\[
\lim_{\| \hat{x} \|_F \to \infty} \hat{\Phi}(\hat{x}) = +\infty.
\]

To this end, we distinguish two cases, which correspond respectively to the assumptions of Theorems 2 and 3:

1) \( \rho \in [0, +\infty) \): in this case \( \hat{\Phi} \) is trivially coercive since

\[
t_{\rho, \hat{x}}(\hat{x}) = +\infty, \quad \forall \| \hat{x} \|_F \geq \rho.
\]

2) \( \rho = +\infty \) and \( \hat{G} \) injective: When \( \rho = +\infty \), the term \( t_{\rho, \hat{x}} \) is always null and \( \hat{\Phi} \) simplifies to

\[
\hat{\Phi}(\hat{x}) = \left\| \hat{G} \hat{x} - \hat{y} \right\|_{2L}^2 + \iota_{\mathcal{H}_K}(T_P(\hat{x})), \quad \hat{x} \in \mathbb{R}^{2N}.
\]

From [3, Section 7.8], we have moreover that

\[
\det \left( \hat{G}^T \hat{G} \right) = | \det(H^H G) |^2 \neq 0,
\]

since \( \hat{G} \) is injective by assumption. From the reverse triangle inequality, we have hence

\[
\left\| \hat{G} \hat{x} - \hat{y} \right\|_{2L} \geq \sigma_{\min} \| \hat{x} \|_{2N} - \| \hat{y} \|_{2L}, \quad \forall \hat{x} \in \mathbb{R}^{2N},
\]

where \( \sigma_{\min} = \sqrt{\lambda_{\min}(\hat{G}^T \hat{G})} > 0 \) is the square root of the eigenvalue of \( \hat{G}^T \hat{G} \) with lowest magnitude, which is non null from (E.12). From the equivalence of norms in finite dimensions, there exist moreover \( c_1, c_2 > 0 \) such that

\[
c_1 \| \hat{x} \|_F \leq \| \hat{x} \|_{2N} \leq c_2 \| \hat{x} \|_F, \quad \forall \hat{x} \in \mathbb{R}^{2N}.
\]

This yields

\[
\left\| \hat{G} \hat{x} - \hat{y} \right\|_{2L} \geq \sigma_{\min} c_1 \| \hat{x} \|_F - \| \hat{y} \|_{2L}, \quad \forall \hat{x} \in \mathbb{R}^{2N},
\]

and hence

\[
\lim_{\| \hat{x} \|_F \to +\infty} \left\| \hat{G} \hat{x} - \hat{y} \right\|_{2L} \geq \lim_{\| \hat{x} \|_F \to +\infty} \sigma_{\min} c_1 \| \hat{x} \|_F = +\infty,
\]

which shows that \( \hat{\Phi} \) is indeed coercive.

We can hence apply Lemma 1, to show that the iterates \( \{ \hat{x}_k \}_{k \in \mathbb{N}} \subset \mathbb{R}^{2N} \) generated by PGD applied to (E.9):

\[
\hat{x}_{k+1} \in \text{prox}_{\tau \hat{H}}^F \left( \hat{x}_k - \tau \nabla \hat{F}(\hat{x}_k) \right),
\]

with \( \tau < 1 / \hat{\beta} \) and \( \hat{x}_0 \in \mathbb{R}^{2N} \), are bounded. Moreover, any limit point \( \hat{x}_* \) of \( \{ \hat{x}_k \}_{k \in \mathbb{N}} \) is a critical point of (E.9), i.e. \( \theta_{2N} \in \partial \hat{\Phi}(\hat{x}_*) \).

Observe finally, that the iterations (E.13) can be rewritten in complex form as

\[
x_{k+1} \in \text{prox}_{\tau \hat{H}}^F (x_k - \tau \nabla F(x_k)),
\]

which follows trivially from the previous identifications. By identification and equivalence between the real and complex optimisation problems (E.9) and (E.8), we can hence conclude that limit points of the iterates \( \{ x_k \}_{k \in \mathbb{N}} \subset \mathbb{C}^N \) generated by (E.14) are critical points of (E.8), which achieves the proof.

\[\square\]

APPENDIX F

PROOF OF PROPOSITION 3

Recall the definition of the proximal set associated to a point \( x \in \mathbb{C}^N \):

\[
\text{prox}_{\tau \hat{H}}^F(x) = \arg \min_{z \in \mathbb{C}^N} \left\{ \frac{1}{2\tau} \| x - z \|_F^2 + \iota_{\mathcal{H}_K}(T_P(z)) + t_{\rho, \hat{x}}(z) \right\}.
\]

(F.1)

When mapped via the Toeplitzification operator \( T_P \), the proximal set (F.1) becomes

\[
T_P \left( \text{prox}_{\tau \hat{H}}^F(x) \right) = \\
= \left\{ T_P(\hat{x}) : \hat{x} \in \text{prox}_{\tau \hat{H}}^F(x) \right\} \\
= \left\{ \hat{X} \in T_P, T_P^{-1}(\hat{X}) \in \text{prox}_{\tau \hat{H}}^F(x) \right\} \\
= \arg \min_{z \in T_P^{-1}(\mathcal{H}_K)} \left\{ \frac{1}{2\tau} \| T_P^{-1}(z) \|_F^2 + \iota_{\mathcal{H}_K}(T_P^{-1}(z)) + t_{\rho, \hat{x}} \left( T_P^{-1}(z) \right) \right\} \\
= \arg \min_{z \in T_P^{-1}(\mathcal{H}_K)} \left\{ \frac{1}{2\tau} \| T_P^{-1}(z) \|_F^2 + t_{\rho, \hat{x}} \left( T_P^{-1}(z) \right) \right\}, \quad (F.2)
\]
where we have used the fact that \( T_p^1 T_p(z) = z \) for all \( z \in \mathbb{C}^N \). We have moreover, \( \forall Z \in T_p^1 \):

\[
\| T_p^1(Z) - x \|^2 = \langle \Gamma_{p}^{-1} T_p(Z) - x, \Gamma_{p}^{-1} T_p(Z) - x \rangle
\]

\[
= ( \Gamma_{p}^{-1} T_p(Z), (\Gamma_{p}^{-1} T_p(Z))_F ) + \langle \Gamma(x), x \rangle
\]

\[
- ( \Gamma_{p}^{-1} T_p(Z), x_T ) - (\Gamma, \Gamma_{p}^{-1} T_p(Z))_F
\]

\[
= (\mathcal{Z}, T_p \Gamma_{p}^{-1} T_p(Z)_F ) + \langle \mathcal{T}(x), T_p(x)_F \rangle - (\mathcal{Z}, T_p(x)_F ) - (\mathcal{T}(x), \mathcal{Z}_F)
\]

\[
= (\mathcal{Z}, \Pi_{\gamma p}(Z))_F + \| T_p(x) \|^2_F
\]

\[
- (\mathcal{Z}, T_p(x)_F ) - (T_p(x), \mathcal{Z}_F)
\]

\[
= \| T_p(x) \|^2_F - \| \mathcal{Z} - T_p(x) \|^2_F,
\]

where we have used the fact that \( \Gamma = \Gamma^H = T_p^1 T_p \) and \( T_p \Gamma_{p}^{-1} T_p = \Pi_{\gamma p} \) (see Appendices C and D). With similar arguments, we have \( \forall Z \in T_p^1 \):

\[
\| T_p^1(Z) \|_F \leq \rho \Leftrightarrow \sqrt{\langle \Gamma_{p}^{-1} T_p(Z), (\Gamma_{p}^{-1} T_p(Z))_F \rangle} \leq \rho
\]

\[
\Leftrightarrow \sqrt{\langle \mathcal{Z}, \mathcal{Z}_F \rangle} \leq \rho
\]

\[
\Rightarrow \| \mathcal{Z} \|_F \leq \rho.
\]

so that

\[
\bigcup_{\rho} \{ T_p^1(Z) \} = \bigcup_{\rho} \{ \mathcal{Z} \} \subset T_p^1,
\]

where \( \mathcal{Z} \subset \mathbb{C}^{(N-P) \times (P+1)} : \| \mathcal{Z} \|_F \leq \rho \}. \) Plugging (F.3) and (F.4) into (F.2) hence yields

\[
T_p \left( \operatorname{prox}_{\Gamma^H}(x) \right)
\]

\[
= \arg \min_{Z \in T_p^1 \cap \mathbb{H}_k} \left\{ \frac{1}{2\tau} \| T_p^1(Z) - x \|^2 + \frac{\| \mathcal{Z} \|_F}{2} \right\}
\]

\[
= \arg \min_{Z \in T_p^1 \cap \mathbb{H}_k} \left\{ \frac{1}{2\tau} \| Z - T_p(x) \|^2 + \frac{\| \mathcal{Z} \|_F}{2} \right\}
\]

\[
= \arg \min_{Z \in T_p^1 \cap \mathbb{H}_k \cap \mathbb{B}_\rho} \left\{ \frac{1}{2\tau} \| Z - T_p(x) \|^2 \right\}
\]

\[
= \Pi_{T_p^1 \cap \mathbb{H}_k \cap \mathbb{B}_\rho}(x).
\]

Using the fact that \( T_p^1 T_p = I_N \) we can finally rewrite (F.5) as

\[
\operatorname{prox}_{\Gamma^H}(x) = T_p \Pi_{T_p^1 \cap \mathbb{H}_k \cap \mathbb{B}_\rho}(x),
\]

which completes the proof.

\( \square \)

**APPENDIX G**

**PROOF OF THEOREM 4**

The proof of Theorem 4 relies on the four lemmas hereafter. The first lemma shows that gradient descent is Lipschitz continuous, and exhibits step size ranges for which it is also a contraction. This is a famous result in optimisation [4], [5], traditionally stated in terms of the \( \ell_2 \) canonical norm. Lemma 2 in contrast assumes the \( \Gamma \)-norm as underlying norm, since the latter is more natural for our particular problem.

**Lemma 2** (Contractive Gradient Descent). Let \( G \in \mathbb{C}^{L \times N} \) be injective, and \( \Gamma \) be the diagonal and definite positive matrix defined in (C.1). Define

\[
\alpha := 2\lambda_{\min} \left( \Gamma^{1/2} G^H G \Gamma^{-1/2} \right),
\]

\[
\beta := 2\lambda_{\max} \left( \Gamma^{1/2} G^H G \Gamma^{-1/2} \right),
\]

where \( \lambda_{\min}(G) \) and \( \lambda_{\max}(G) \) denote the minimum and maximum eigenvalue of a matrix \( G \) respectively. Let \( \tau \in \mathbb{R} \) be a positive constant and consider the linear map

\[
D_{\tau} : \mathbb{C}^N \to \mathbb{C}^N,
\]

\[
x \mapsto x - 2\tau G^H (Gx - y),
\]

for some \( y \in \mathbb{C}^L \). Then, \( D_{\tau} \) is Lipschitz continuous with respect to the norm induced by \( \Gamma \):

\[
\| D_{\tau}(x) - D_{\tau}(z) \|_\Gamma \leq L_\tau \| x - z \|_\Gamma, \quad \forall x, z \in \mathbb{C}^N,
\]

with Lipschitz constant:

\[
L_\tau := \max \{ |1 - \tau \alpha|, |1 - \tau \beta| \}.
\]

Moreover, \( D_{\tau} \) is contractive, i.e. \( 0 \leq L_\tau < 1 \), for \( 0 < \tau < 2/\beta \), and \( L_\tau \) is minimised for \( \tau = 2/(\alpha + \beta) \).

**Proof.** We have

\[
\| D_{\tau}(x) - D_{\tau}(z) \|_\Gamma = \left\| \left( I_N - 2\tau G^H G \right)(x - z) \right\|_\Gamma
\]

\[
\leq \left\| I_N - 2\tau G^H G \right\|_\Gamma \| x - z \|_\Gamma,
\]

\[
= L_\tau \| x - z \|_\Gamma
\]

where the Lipschitz constant \( L_\tau := \| I_N - 2\tau G^H G \|_\Gamma > 0 \) is the operator norm of \( I_N - 2\tau G^H G \) induced by the \( \Gamma \)-norm on \( \mathbb{C}^N \):

\[
\left\| I_N - 2\tau G^H G \right\|_\Gamma = \sup_{\| x \|_\Gamma = 1} \left\| \left( I_N - 2\tau G^H G \right)x \right\|_\Gamma
\]

\[
= \sup_{\| x \|_\Gamma = 1} \left\| \Gamma^{1/2} \left( I_N - 2\tau G^H G \right)x \right\|_2
\]

\[
= \sup_{\| x \|_2 = 1} \left\| \Gamma^{1/2} \left( I_N - 2\tau G^H G \right) \right\|_{\Gamma^{-1/2}} \| x \|_2
\]

\[
= \left\| I_N - 2\tau G^H G \right\|_{\Gamma^{-1/2}}.
\]

Note that since \( G \) is injective, \( G^H G \) is positive definite and hence we easily get [5] that the eigenvalues of \( I_N - 2\tau G^H G \Gamma^{-1/2} \) are contained in the interval \( [1 - \tau \beta, 1 - \tau \alpha] \), where \( \beta \geq \alpha > 0 \) are defined in (G.1) and (G.2) respectively. Its spectral norm is hence given by:

\[
\left\| I_N - 2\tau G^H G \right\|_{\Gamma^{-1/2}} = \max \{ |1 - \tau \alpha|, |1 - \tau \beta| \},
\]

which proves (G.4). Finally, the restriction on \( \tau \) for \( L_\tau \) to be smaller than one follows from basic algebra, and is discussed in [4].

\( \square \)

The second lemma states that in a Hilbert space, orthogonal projection maps onto closed convex sets are non-expansive. This is a known result of approximation theory [6], [7].
Lemma 3 (Non-Expansiveness of Closed Convex Projections). Let \( \mathcal{H} \) be some Hilbert space with some inner-product norm \( \| \cdot \| \) and \( C \subset \mathcal{H} \) a closed, convex set. Then the orthogonal projection map onto \( C \), defined as
\[
\Pi_C(x) = \arg \min_{z \in C} \| x - z \|, \quad \forall x \in \mathcal{H},
\]
is non-expansive, i.e.
\[
\| \Pi_C(x) - \Pi_C(z) \| \leq \| x - z \|, \quad \forall x, z \in \mathcal{H}.
\]

Proof. Lemma 3 is proven in [6, Theorem 5.5].

The third lemma states that the singular value projection map \( \Pi_{\mathcal{H}_k} \) is locally non-expansive in every neighbourhood of the manifold of matrices with rank exactly \( k \).

Lemma 4 (Local Non-Expansiveness of the Singular Value Projection). Let \( \mathcal{H}^{\text{mxn}} \) be the space of complex-valued rectangular matrices of size \( m \times n \), and \( \mathcal{H}_k \subset \mathcal{H}^{\text{mxn}} \) be the sets of matrices with rank at most and exactly \( k \leq \min(m,n) \) respectively. Denote further by \( \Pi_{\mathcal{H}_k} \) the orthogonal projection map onto \( \mathcal{H}_k \) given in (14). Then, for every \( R \in \mathcal{R}_k \), the map \( \Pi_{\mathcal{H}_k} \) is well-defined (single-valued) and locally non-expansive,
\[
\| \Pi_{\mathcal{H}_k}(X) - \Pi_{\mathcal{H}_k}(Z) \| \leq \| X - Z \|, \quad \forall X, Y, Z \in \mathcal{U},
\]
for some neighbourhood \( \mathcal{U} \ni R \).

Proof. Since \( \mathcal{R}_k \) is dense in \( \mathcal{H}_k \) [8, Proposition 2.1], we have \( \Pi_{\mathcal{H}_k} = \Pi_{\mathcal{R}_k} \) in a neighbourhood \( W \) of every \( R \in \mathcal{R}_k \) (see [9, Example 2.3] for a detailed proof of this fact). Moreover, [10, Lemma 3] tells us that, for every \( R \in \mathcal{R}_k \), \( \Pi_{\mathcal{R}_k} \) is, in a neighbourhood \( \mathcal{U} \ni R \) such that \( \mathcal{U} \subset W \), well-defined (single-valued), continuous and differentiable, with gradient given by:
\[
\nabla \Pi_{\mathcal{R}_k} = \Pi_{\mathcal{R}_k}(R) \text{ where } \Pi_{\mathcal{R}_k}(R) \subset \mathcal{H}^{\text{mxn}} \text{ is the tangent plane of the manifold } \mathcal{R}_k \text{ in } R \text{ (see [9, Example 2.2])}.
\]

Since \( \Pi_{\mathcal{R}_k}(R) \) is by definition a linear subspace of \( \mathcal{H}^{\text{mxn}} \), the orthogonal projection operator \( \Pi_{\mathcal{H}_k}(R) \) is bounded with unit spectral norm. The map \( \Pi_{\mathcal{H}_k} \equiv \Pi_{\mathcal{R}_k} \) is consequently 1-Lipschitz continuous (i.e. non-expansive) with respect to the Frobenius norm in the neighbourhood \( \mathcal{U} \ni R \). □

The last lemma finally, makes use of Lemmas 3 and 4 to show that the denoising operator \( H_n(x) = T_P^\dagger \Pi_{\mathcal{H}_n} \Pi_{\mathcal{R}_n} T_P(x) \) is locally non-expansive with respect to the Frobenius norm:

Lemma 5 (Local Non-Expansiveness of Denoiser). Let \( \mathcal{H}^{(N-P)\times(P+1)} \) be the space of complex-valued rectangular matrices of size \( (N - P) \times (P + 1) \), \( P \leq \lfloor N/2 \rfloor \), and \( \mathcal{H}_k \subset \mathcal{H}^{(N-P)\times(P+1)} \), \( \mathcal{R}_k \subset \mathcal{H}^{(N-P)\times(P+1)} \) be the sets of matrices with rank at most and exactly \( k \leq P \times (P + 1) \) respectively. Let
\[
H_n(x) := T_P^\dagger \Pi_{\mathcal{H}_n} \Pi_{\mathcal{R}_n} \Pi_{\mathcal{R}_n} T_P(x), \quad \forall x \in \mathcal{H}^{N \times N},
\]
be the approximate proximal operator (34). Then, \( H_n \) is locally well-defined (single-valued) and non-expansive with respect to \( \| \cdot \|_F \),
\[
\| H_n(x) - H_n(z) \|_F \leq \| x - z \|_F,
\]
for all \( x, z \in \mathcal{H}^{N \times N} \) such that \( T_P(x), T_P(z) \) are in some neighbourhood of some matrix \( R \in \mathcal{R}_K \).

Proof. First, we have, for all \( x, z \in \mathbb{C}^N \):
\[
\| H_n(x) - H_n(z) \|_F = \| T_P^\dagger (D_n T_P(x) - D_n T_P(z)) \|_F = (G.6)
\]
where \( D_n = [\Pi_{\mathcal{T}_P} \Pi_{\mathcal{H}_K} \Pi_{\mathcal{R}_L}]^n \). Notice that for \( X \in \mathcal{T}_P \), we have
\[
\| T_P^\dagger (X) \|_F^2 = (TT_P^\dagger (X), TT_P^\dagger (X))_2 = (TT_P^\dagger (X), TT_P^\dagger (X))_F = (X, TT_P^\dagger (X))_F = (X, X)_F = \| X \|_F^2.
\]
Since the range of \( D_n \) is \( \mathcal{T}_P \), (G.6) becomes:
\[
\| H_n(x) - H_n(z) \|_F = \| D_n T_P(x) - D_n T_P(z) \|_F.
\]
Assuming now that \( T_P(x) \) and \( T_P(z) \) are in some neighbourhood of some point \( R \in \mathcal{R}_K \), we can invoke Lemmas 3 and 4 recursively to obtain:
\[
\| D_n T_P(x) - D_n T_P(z) \|_F \leq \| T_P(x) - T_P(z) \|_F = \| x - z \|_F,
\]
where we have used: \( \| T_P(x) \|_F^2 = (T_P(x), T_P(x))_F = (T_P(x), T_P(x)) = \| x \|_F^2 \) for all \( x, z \in \mathbb{C}^N \) such that \( T_P(x), T_P(z) \) are in some neighbourhood of some matrix \( R \in \mathcal{R}_K \).

We are now ready to show Theorem 4. Let
\[
U_{\tau,n}(x) := H_n(x - \tau \nabla F(x)) = H_n(D\tau(x)), \quad x \in \mathbb{C}^N.
\]
Then, for every \( x, z \in \mathbb{C}^N \) such that \( T_P(x), T_P(z) \) are in some neighborhood of some matrix \( R \in \mathcal{R}_K \), \( U_{\tau,n} \) is locally Lipschitz continuous (i.e. non-expansive) with respect to the Frobenius norm in the neighborhood \( \mathcal{U} \ni R \).

Proof of Corollary 2

First, we note that from Theorem 4, we have under the assumptions of the corollary that
\[
\| x_{k+1} - x_k \|_R \leq \| U_{\tau,n}(x_k) - U_{\tau,n}(x_{k-1}) \|_R \leq L_{\tau} \| x_k - x_{k-1} \|_R\quad \forall k \geq 1.
\]

APPENDIX H

Proof of Corollary 2

First, we have under the assumptions of the corollary that
\[
\| x_{k+1} - x_k \|_R \leq \| U_{\tau,n}(x_k) - U_{\tau,n}(x_{k-1}) \|_R \leq L_{\tau} \| x_k - x_{k-1} \|_R\quad \forall k \geq 1.
\]
By assumption $0 < \tau < 2/\beta$ and therefore $0 < L_\tau < 1$. We deduce hence from (H.1) that \( \{x_k\}_{k \in \mathbb{N}} \) is a Cauchy sequence. Let \( j, k \in \mathbb{N} \) with \( j > k \):

\[
\|x_j - x_k\| \leq \sum_{m=k}^{j-1} \|x_{m+1} - x_m\| \leq \sum_{m=k}^{j-1} L_\tau^m \|x_1 - x_0\| \leq \|x_1 - x_0\| L_\tau^k \sum_{m=0}^{j-1-k} L_\tau^m \leq \|x_1 - x_0\| L_\tau^k \sum_{m=0}^{\infty} L_\tau^m = \frac{L_\tau^k}{1 - L_\tau} \|x_1 - x_0\|.
\]

(H.2)

For every \( \epsilon > 0 \), we can choose a \( J \in \mathbb{N} \) such that

\[
L_\tau^J < \frac{\epsilon (1 - L_\tau)}{\|x_1 - x_0\|},
\]

and hence for all \( j > k > J \)

\[
\|x_j - x_k\| < \epsilon.
\]

The sequence \( \{x_k\}_{k \in \mathbb{N}} \) is hence a Cauchy sequence, and since \( \mathbb{C}^N \) is complete, it converges towards a limit point \( x_* \in \mathbb{C}^N \). We have moreover, since \( U_{\tau,n} \) is continuous

\[
x_* = \lim_{n \to \infty} x_k = \lim_{n \to \infty} U_{\tau,n}(x_{k-1}) = U_{\tau,n} \left( \lim_{n \to \infty} x_{k-1} \right) = U_{\tau,n}(x_*),
\]

and hence \( x_* \) is a fixed-point of \( U_{\tau,n} \). Note moreover that, from (H.2) we get

\[
\|x_* - x_k\| = \lim_{j \to \infty} \|x_j - x_k\| \leq \lim_{j \to \infty} \|x_1 - x_0\| L_\tau^k \sum_{m=0}^{j-1-k} L_\tau^m \leq \|x_1 - x_0\| L_\tau^k \sum_{m=0}^{\infty} L_\tau^m = \frac{L_\tau^k}{1 - L_\tau} \|x_1 - x_0\|,
\]

which proves (37) of Corollary 2.

\[\square\]

REFERENCES

Figure A: Actual vs. recovered Dirac locations for LS-Cadzow, CPGD and GenFRI, various oversampling parameters $\beta \in \{1, 2, 3, 4\}$, a PSNR in $\{-30, 0, 30\}$ dB. For each case and each source (denoted with the same colours as in Fig. 2), the markers and horizontal lines represent respectively the median and inter-quartile region of the estimated locations’ empirical distribution obtained from 192 noise realisations. The closer a marker is from the line $y = x$ (in dark grey), the better the recovery.