

Stagewise Weak Gradient Pursuits

Part II: Theoretical Properties

Thomas Blumensath, *Member, IEEE*, Mike E. Davies, *Member, IEEE*

Abstract—In a recent paper [2] we introduced the greedy Gradient Pursuit framework. This is a family of algorithms designed to find sparse solutions to underdetermined inverse problems. One particularly powerful member of this family is the (approximate) conjugate gradient pursuit algorithm, which was shown to be applicable to very large data sets and which was found to perform nearly as well as the traditional Orthogonal Matching Pursuit algorithm. In Part I of this paper [3], we have further extended the Gradient Pursuit framework and introduced a greedier *stagewise weak* selection strategy that selects several elements per iteration. Combining the conjugate gradient update of [2] with this selection strategy led to a very fast algorithm, applicable to large scale problems, which was shown in [3] to perform well in a wide range of applications.

In this paper we study theoretical properties of the approach. In particular, we propose a novel fast recursion to calculate the conjugate gradient update direction and present a proof that shows that this update is guaranteed to be better than a simple gradient update.

The other contribution of this paper is to derive theoretic guarantees that give conditions under which the stagewise weak selection strategy can be used to exactly recover sparse vectors from a few measurements, that is, guarantees under which the algorithm is guaranteed to find the sparsest solution.

Index Terms—Sparse Representations/Approximations, Orthogonal Matching Pursuit, Weak Matching Pursuit, Gradient Pursuit, Stagewise Selection, Exact Signal Recovery, Compressed Sensing.

I. INTRODUCTION

In part I of this paper [3] we proposed a novel generalisation of the Gradient Pursuit family of algorithms, previously introduced in [2]. These algorithms were developed to solve large sparse inverse problems which can be formally stated as follows. Given an observation vector $\mathbf{x} \in \mathbb{R}^M$ and a matrix $\Phi \in \mathbb{R}^{M \times N}$, where $N > M$, find a *sparse* vector $\hat{\mathbf{y}} \in \mathbb{R}^N$, such that

$$\hat{\mathbf{x}} = \Phi \hat{\mathbf{y}},$$

is close to \mathbf{x} , that is the error $\mathbf{n} = \mathbf{x} - \hat{\mathbf{x}}$ is small. Whilst there are other possibilities, the most typical way to quantify

The authors are with IDCOM & Joint Research Institute for Signal and Image Processing, Edinburgh University, King's Buildings, Mayfield Road, Edinburgh EH9 3JL, UK (Tel.: +44(0)131 6505659, Fax.: +44(0)131 6506554, e-mail: thomas.blumensath@ed.ac.uk, mike.davies@ed.ac.uk).

This research was supported by EPSRC grant D000246/1. MED acknowledges support of his position from the Scottish Funding Council and their support of the Joint Research Institute with the Heriot-Watt University as a component part of the Edinburgh Research Partnership. The authors would like to thank Ian Marshall and Yuehui Tao for providing access to the MRI image data.

Some of the work presented here has previously been presented in [1].

© This work has been submitted to the IEEE for possible publication. Copyright may be transferred without notice, after which this version may no longer be accessible.

sparse of a vector is to count the number of elements that are non-zero. Even though this is not a norm, we denote this 'numerosity' as $\|\cdot\|_0$, informally called ℓ_0 norm.

A more thorough discussion of this problem, its application and different strategies for its solution can be found in part I of this paper [3].

In this part of the paper we present theoretical properties of the Stagewise Weak Conjugate Gradient Pursuit algorithm and address two questions. How fast does the algorithm approach a solution and how good is this solution?

The Stagewise Weak Conjugate Gradient Pursuit algorithm is a generalisation of Orthogonal Matching Pursuit (OMP) [4], which has strong convergence guarantees. Whilst the generality of the Gradient Pursuit framework of [2] does not allow general convergence statements, particular instances of the algorithmic framework, such as the algorithm based on gradient updates, could be shown to converge linearly [2]. In this paper we derive a convergence result for the (approximate) conjugate gradient pursuit algorithm of [2], which is at least as good as that for the gradient based scheme. Furthermore, we introduce a new recursion that allows the (approximate) conjugate gradient to be calculated with the same order of computations than the gradient itself, making this the preferable algorithm of the two.

The other question regards the quality of the solution found with a sparse approximation algorithm. Several surprising results in this direction were presented, for example, in the papers [5], [6], [7], [8] and [9], which could show that under certain conditions, solving a convex ℓ_1 problem will simultaneously solve the ℓ_0 problem. The recently developed theory of compressed sensing gave even more generous conditions on Φ [10], [11] that give similar guarantees. Comparable results have also been derived for OMP [12] and [13]. These results will be generalised here and are shown to also apply (with possible minor modifications) to the algorithms proposed in [2] and [3].

A. Paper Overview

Section II recalls the Gradient Pursuit framework of [2]. The companion paper [3] proposed the *Stagewise Weak* selection strategy which we quickly summarise in subsection II-B.

One member of the Gradient Pursuit family is the (approximate) Conjugate Gradient Pursuit algorithm, which will be discussed in subsection III-A, where we propose a novel fast implementation of the method. We will furthermore derive a novel convergence result for this approach (subsection III-C).

Theoretical properties of the stagewise weak methods will be studied in section IV where we derive several results that give guarantees on the recovery performance of the methods.

B. Notation

We here use the same notation as in [2]. In iteration n , $\Gamma^{[n]}$ is a set of indices, which is used to label a subset of columns from a matrix Φ . The matrix $\Phi_{\Gamma^{[n]}}$ is then the sub-matrix of Φ containing those columns of Φ with indices in $\Gamma^{[n]}$. The same convention is used for vectors. Superscripts in a subscript, as in $\hat{\mathbf{y}}_{\Gamma^{[n]}}$ reminds us that we are in iteration n , on occasion, however, we resort to using additional superscripts (e.g. $\hat{\mathbf{y}}^{[n]}$) to further clarify the current iteration. The Gram matrix is denoted by $\mathbf{G}_{\Gamma^{[n]}} = \Phi_{\Gamma^{[n]}}^T \Phi_{\Gamma^{[n]}}$. Lower case bold face characters represent vectors and upper case bold characters are used for matrices. Individual elements from a vector will be in standard type face with a subscript. For example \mathbf{g} will be used to refer to a negative gradient vector with g_i denoting the i^{th} element of this vector. Inner products between vectors will often be written using angled brackets, e.g. $\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^T \mathbf{y}$ and estimated quantities will be distinguished using the hat notation $\hat{\cdot}$.

II. STAGewise WEAK GRADIENT PURSUIT

A. Gradient Pursuits

Gradient Pursuit algorithms iterate through the following three steps.

- 1) Select columns from Φ based on the inner products $|\phi_i^T \mathbf{r}^{[n-1]}|$.
- 2) Update the estimate of $\hat{\mathbf{y}}$ using a directional update.
- 3) Update the residual $\mathbf{r}^{[n]} = \mathbf{x} - \Phi_{\Gamma^{[n]}} \hat{\mathbf{y}}_{\Gamma^{[n]}}^{[n]}$.

The directional update of $\hat{\mathbf{y}}^{[n-1]}$ is

$$\mathbf{y}^{[n]} = \mathbf{y}^{[n-1]} + \beta \mathbf{d}^{[n]},$$

where β is a step-size, that can be calculated explicitly to give the minimum mean squared error solution for a given update direction \mathbf{d} .

To keep the second part of this paper concise, we refer the reader to the first part of this paper [3] as well as to [2] for a more detailed discussion of this approach.

Whilst the updates in Matching Pursuit and Orthogonal Matching Pursuits also fall into this framework¹, different directions $\mathbf{d}^{[n]}$ can be beneficial. In [2], the gradient and an approximate conjugate gradient method were suggested. In the gradient based approach, the update direction is

$$\mathbf{d}_{\Gamma^{[n]}}^{[n]} = \mathbf{g}_{\Gamma^{[n]}}^{[n]} = \Phi_{\Gamma^{[n]}} (\mathbf{x} - \Phi_{\Gamma^{[n]}} \hat{\mathbf{y}}_{\Gamma^{[n]}}^{[n-1]}). \quad (1)$$

The (approximate)² conjugate gradient method uses the direction

$$\mathbf{d}_{\Gamma^{[n]}}^{[n]} = \mathbf{g}_{\Gamma^{[n]}}^{[n]} + \omega \mathbf{d}_{\Gamma^{[n]}}^{[n-1]}, \quad (2)$$

where ω is calculated such that $\langle \Phi_{\Gamma^{[n]}} \mathbf{d}_{\Gamma^{[n]}}^{[n]}, \Phi_{\Gamma^{[n]}} \mathbf{d}_{\Gamma^{[n]}}^{[n-1]} \rangle = 0$, that is, consecutive update directions are *conjugate*.

¹As an aside, it is interesting to note that the least angle regression algorithm (LARS) [14], which is a greedy version of convex relaxation methods, also falls into the Gradient Pursuit framework. In LARS, an update direction $\mathbf{d}^{[n]}$ and step size $\mu^{[n]}$ are chosen such that the inner products between the residual $\mathbf{r}^{[n]} = \mathbf{r}^{[n-1]} - \mu^{[n]} \mathbf{d}^{[n]}$ and the selected elements ψ_i all have the same magnitude as the largest inner product between $\mathbf{r}^{[n]}$ and a previously unselected element. See [14] for more details.

²Approximate, because the method only guarantees conjugacy of the current update direction to the previous update direction, but not to all previous update directions [2].

B. Stagewise Weak Selection Strategy

The Stagewise Weak selection strategy introduced in [3] was inspired by the *weak Matching Pursuit* algorithm [15]. Weak Matching Pursuit selects *any one* element such that

$$|g_j| \geq \alpha \max_i |g_i|.$$

Instead of selecting a *single* element satisfying the above condition, the stagewise weak algorithm selects *all* element that satisfy this condition. Using this approach, the set of indices is updated as

$$\Gamma^{[n]} = \Gamma^{[n-1]} \cup \{i : |g_i| \geq \alpha \max_j |g_j|\}. \quad (3)$$

Combining the stagewise weak selection strategy with the coefficient update of OMP, we get the Stagewise Weak Orthogonal Matching Pursuit algorithm (SWOMP) while combining the stagewise weak selection step with a conjugate gradient update, we get Stagewise Weak Conjugate Gradient Pursuit (SWCGP).

III. CONJUGATE GRADIENT PURSUIT

The algorithm based on update direction (2) was called Approximate Conjugate Gradient Pursuit in [2]. For simplicity, we will here drop the term approximate and use the name Conjugate Gradient Pursuit (CGP). This method is discussed in detail in this section, where we give a novel recursion that allows the method to be implemented more efficiently. We also present a convergence result for the approach.

A. Implementation

The following efficient implementation of Conjugate Gradient Pursuit is based on a novel recursion to calculate the update direction and uses auxiliary vectors \mathbf{w} and \mathbf{v} .

- Input: \mathbf{x} , Φ and stopping criterion
- Initialise: $\hat{\mathbf{y}}^{[0]} = \mathbf{0}$, $\Gamma^{[0]} = \emptyset$, $\mathbf{r}^{[0]} = \mathbf{x}$, $n = 1$
- iterate until stopping criterion is met:
 - 1) $\mathbf{g}^{[n]} = \Phi^T \mathbf{r}^{[n-1]}$
 - 2) Select a set of new elements \mathcal{I} .
 - 3) $\Gamma^{[n]} = \Gamma^{[n-1]} \cup \mathcal{I}$
 - 4) if $n = 1$

$$\begin{aligned} & - \mathbf{d}_{\Gamma^{[n]}}^{[n]} = \mathbf{g}_{\Gamma^{[n]}}^{[n]} \\ & - \mathbf{v}^{[n]} = \Phi_{\Gamma^{[n]}} \mathbf{d}_{\Gamma^{[n]}}^{[n]} \end{aligned}$$

else,

$$\begin{aligned} & - \mathbf{w}^{[n]} = \Phi_{\Gamma^{[n]}} \mathbf{g}_{\Gamma^{[n]}}^{[n]} \\ & - \omega^{[n]} = -\langle \mathbf{v}^{[n-1]}, \mathbf{w}^{[n]} \rangle / \kappa^{[n-1]} \\ & - \mathbf{d}_{\Gamma^{[n]}}^{[n]} = \mathbf{g}_{\Gamma^{[n]}}^{[n]} + \omega \mathbf{d}_{\Gamma^{[n]}}^{[n-1]} \\ & - \mathbf{v}^{[n]} = \mathbf{w}^{[n]} + \omega^{[n]} \mathbf{v}^{[n-1]} \end{aligned}$$

- 5) $\kappa^{[n]} = \|\mathbf{v}^{[n]}\|_2^2$
- 6) $\mu^{[n]} = \langle \mathbf{r}^{[n-1]}, \mathbf{v}^{[n]} \rangle / \kappa$
- 7) $\hat{\mathbf{y}}_{\Gamma^{[n]}}^{[n]} = \hat{\mathbf{y}}_{\Gamma^{[n]}}^{[n-1]} + \mu^{[n]} \mathbf{d}_{\Gamma^{[n]}}^{[n]}$
- 8) $\mathbf{r}^{[n]} = \mathbf{r}^{[n-1]} - \mu \mathbf{v}^{[n]}$
- 9) $n \mapsto n + 1$

- Output $\mathbf{r}^{[n-1]}$, $\Gamma^{[n-1]}$ and $\hat{\mathbf{y}}^{[n-1]}$

We have here deliberately left the element selection step ambiguous. In [2] we used the same strategy as in Matching Pursuit and Orthogonal Matching Pursuit, whilst in [3] the Stagewise Weak selection strategy was proposed for this task.

As discussed in [2], this approach is fundamentally different from a strategy that uses a conjugate gradient solver in each iteration of OMP as is done, for example, in Stagewise Orthogonal Matching Pursuit (StOMP) [16]. Whilst the conjugate gradient method tries to retain conjugacy between iterations, the method used in StOMP starts a new conjugate gradient optimisation in each iteration.

One important difference between the CGP strategy and the use of a standard conjugate gradient solver, restarted in each iteration and run for a fixed number of iterations, is that, using a single CGP update and allowing elements to be re-selected in consecutive iterations, CGP has a built in mechanism that senses how orthogonal the residual $\mathbf{r}^{[n]}$ is to all previously selected elements. If the residual is far from orthogonal, the inner products with already selected elements will be large and an element is re-selected. If on the other hand, a single update step leads to a solution close to the OMP optimum, in which case the residual becomes nearly orthogonal to all previously selected elements, the algorithm will not re-select elements.

B. Computation Cost per Iteration

An important property of the algorithm as outlined in subsection III-A is that it only requires the storage of vectors and scalars. The only exception is the required storage of the mapping Φ . The storage requirements are therefore low. In particular, \mathbf{g} has N elements, whilst \mathbf{v} , \mathbf{w} , \mathbf{x} and \mathbf{r} are of length M and Γ , $\hat{\mathbf{y}}_\Gamma$ and \mathbf{d}_Γ are at most of length K , where K is the number of non-zero elements in the solution.

The computational complexity is also low. The mapping Φ is only applied twice in each iteration, once in step 1, to calculate the inner products and once in step 4 to calculate the new update direction. Apart from the search required in the selection step, which is an $O(N)$ operation, all other computations are inner products or vector scalar multiplications. In total, there are three inner products between vectors of length M and 4 vector additions (two of M -vectors and two of K -vectors), all of which also involve a scalar product. The computational complexity is therefore of the same order as the application of the operator Φ or its adjoint. The method is thus particularly fast if these operators can be implemented efficiently, based, for example, on fast Fourier or fast wavelet transforms. In this case, the storage cost of Φ , which dominates the storage requirements, is also significantly reduced. Due to the new recursion, the computational requirements are slightly different from those reported in [2] and are therefore summarised in table I.

C. Convergence

Although the gradient based directional updates do not fully minimise the residual, it can be shown that under certain circumstances a single optimization step actually does a pretty good job. In [2] we have shown that a Gradient Pursuit

I: Computation and storage cost of CGP in iteration $n \geq 2$.

computation
2 applications of Φ (once using the adjoint)
3 inner products of M -vectors
2 M -vector additions (with scalar multiplication)
2 K -vector additions (with scalar multiplication)
storage
Φ
1 N -vector
4 M -vectors
3 K -vectors

algorithm with update (1) converges linearly. In particular we had

Theorem 1: There exists a constant $c < 1$, which only depends on Φ , such that the residual calculated with the gradient based Gradient Pursuit algorithm decays as

$$\|\mathbf{r}^{[n]}\|_2^2 \leq c \|\mathbf{r}^{[n-1]}\|_2^2.$$

The constant c can be expressed in terms of the following quantities of interest in the theoretical study of sparse signal recovery. The restricted isometry constant ζ_K [17] is a symmetric bound on the singular values of any sub-matrix of Φ with K (or less) elements and is defined as the smallest quantity such that

$$(1 - \zeta_K(\Phi)) \leq \frac{\|\Phi \mathbf{y}\|_2^2}{\|\mathbf{y}\|_2^2} \leq (1 + \zeta_K(\Phi)),$$

holds for all \mathbf{y} with no more than K non-zero elements.

It can be shown that [2]

$$c \leq \left(1 - \frac{\omega}{\|\Phi\|_2^2}\right),$$

where $\omega > 0$ is such that $\|\Phi^T \mathbf{x}\|_\infty^2 > \omega \|\mathbf{x}\|_2^2$, for all \mathbf{x} [15, pp. 422]. If $(1 - \zeta_K) > 0$ and if the algorithm selects only K elements, then we can write the constant in terms of the restricted isometry

$$c \leq \left(1 - \frac{1 - \zeta_K}{K(1 + \zeta_K)}\right).$$

An argument for the use of gradient based optimisation can also be given based on the restricted isometry. If the dictionary has a small restricted isometry constant ζ_K , then every subdictionary is very nearly orthogonal. Because eigenvalues of sub-matrices are nested between those of the full matrix [18, Theorem 7.3.9], for $n \leq K$, the condition number, κ , of the sub-dictionary's Gram matrix, $\mathbf{G}_\Gamma^{[n]}$ is bounded by

$$\kappa(\mathbf{G}_\Gamma^{[n]}) \leq \left(\frac{1 + \zeta_K}{1 - \zeta_K}\right)^2.$$

This can be used to explain the good performance of the gradient based updates. Using

$$f(\mathbf{y}_{\Gamma^n}) = \|\mathbf{x} - \Phi_{\Gamma^n} \mathbf{y}_{\Gamma^n}\|_2^2,$$

a worst case analysis of the gradient line search gives [19]:

$$\begin{aligned} \frac{f(\mathbf{y}_{\Gamma^n}^n) - f(\mathbf{y}_{\Gamma^n}^*)}{f(\mathbf{y}_{\Gamma^{n-1}}^{n-1}) - f(\mathbf{y}_{\Gamma^{n-1}}^*)} &\leq \left(\frac{\kappa - 1}{\kappa + 1}\right)^2 \\ &\leq \left(\frac{2\zeta}{(1 - \zeta)^2}\right)^2 \end{aligned} \quad (4)$$

where $\mathbf{y}_{\Gamma^n}^*$ denotes the least squares solution of $f(\mathbf{y}_{\Gamma^n})$. Hence for small ζ the convergence, even of a single gradient iteration, is good.

The convergence of the Conjugate Gradient Pursuit algorithm discussed above was not derived in [2]. The following theorem shows that the reduction in $f(\mathbf{y}_{\Gamma^n})$ is at least as good when using update (2) than when using update (1).

Theorem 2: Use ω and μ as defined in subsection III-A. In the Gradient Pursuit framework, using the update direction (2) reduces the ℓ_2 norm of the residual $\mathbf{r}^{[n]} = \mathbf{x} - \Phi\mathbf{y}^{[n]} = \mathbf{x} - \Phi(\mathbf{y}^{[n-1]} + \mu\mathbf{d}^{[n]})$ at least as much as using the update direction (1). Therefore, the convergence for the algorithm using the direction defined in (2) is at least as good as that in theorem 1.

The proof is basically that of [20]. However, in the Gradient Pursuit framework, the Gram matrix changes from iteration to iteration and care has to be taken to show that it still holds in this case. This can indeed be done, as shown in appendix I.

IV. RECOVERY ANALYSIS

Theoretical justification for the use of the stagewise weak selection step can be derived mirroring the results of Tropp [12] and Tropp and Gilbert [13]. These results will be stated for a quite general family of algorithms, which will be introduced first.

A. General weak MP algorithms

Some of the results derived here hold for all general weak Matching Pursuit algorithms, which we define slightly more generally than in [21]. A *general weak Matching Pursuit* algorithm is any algorithm that in each iteration

- 1) adds *any number* of indices from the set $\{i : |g_i| \geq \alpha \max_j |g_j|\}$ to the set of previously selected indices; and then
- 2) estimates \mathbf{y} deterministically by setting only elements with indices in the selected set to non-zero values.

Importantly, the exact value of the non-zero elements in step two is arbitrary as long as they depend deterministically on \mathbf{x} , $\mathbf{r}^{[n-1]}$ and Φ . Therefore, MP, OMP, CGP, SWOMP and SWCGP are all particular instances of general weak Matching Pursuits.

B. Analysis Based on Coherence: Noiseless Recovery

For the weakness factor $\alpha \leq 1$, define the exact recovery condition ($\text{ERC}_\alpha(\Gamma)$) [12] as

$$\max_{i \notin \Gamma} \|\Phi_\Gamma^\dagger \phi_i\|_1 < \alpha.$$

A proof very similar to that presented in [12] leads to the following theorem. The full proof is given in appendix II for completeness.

Theorem 3: Assume that $\mathbf{x} = \Phi\mathbf{y}$ and that \mathbf{y} is supported on the elements in Γ^* . If $\text{ERC}_\alpha(\Gamma^*)$ holds and if $\Gamma^{[n-1]} \subset \Gamma^*$, then

$$\{i : |g_i| \geq \alpha \max_j |\Phi^T(\mathbf{x} - \Phi_{\Gamma^{[n-1]}}\hat{\mathbf{y}}_{\Gamma^{[n-1]}})|\} \subset \Gamma^*$$

holds for all $\hat{\mathbf{y}}_{\Gamma^{[n-1]}}$.

The theorem guarantees that, as long as the residual $\mathbf{r}^{[n-1]}$ is in the span of Φ_{Γ^*} , the algorithm will only select correct elements, which in turn implies that the next residual will remain in the span of Φ_{Γ^*} . We therefore have the following corollary

Corollary 1: Assume that $\mathbf{x} = \Phi\mathbf{y}$ and that \mathbf{y} is supported on the elements in Γ^* . If $\text{ERC}_\alpha(\Gamma^*)$ holds, then a *general weak MP* algorithm will only select elements from the set Γ^* .

These results are in terms of the exact recovery condition, which is a function of the support set of the solution. A more general statement can be made using the coherence and cumulative coherence of Φ . The coherence of Φ is defined as

$$\mu = \max_{i \neq j} |\langle \phi_i, \phi_j \rangle|$$

and the cumulative coherence [12] is defined as

$$\mu_1(K) = \max_{\Gamma: |\Gamma|=K} \max_{i \notin \Gamma} \sum_{j \in \Gamma} |\langle \phi_i, \phi_j \rangle|,$$

where the outer maximum is taken over all sets Γ with K elements. Instead of using the cumulative coherence, the following bound in terms of the coherence (which is easily calculated) can be used [12]

$$\mu_1(K) \leq K\mu.$$

Without proof, we present the following result from [12] and [21].

Lemma 1: If $\mu_1(K) + \alpha\mu_1(K-1) < \alpha$, then $\text{ERC}_\alpha(\Gamma)$ holds for all Γ with $|\Gamma| \leq K$.

Therefore, combining the results of this subsection, if $\mu_1(K) + \alpha\mu_1(K-1) < \alpha$, if $\mathbf{x} = \Phi\mathbf{y}$ and if \mathbf{y} is K -sparse, then a *general weak Matching Pursuit* algorithm will pick up only elements from the support set of \mathbf{y} .

C. Analysis Based on Coherence: Approximation

The results in the previous subsection were for observations $\mathbf{x} = \Phi\mathbf{y}$, where \mathbf{y} was exact sparse, i.e. all but K elements were exactly zero. This is a quite restrictive assumption in practice. Furthermore, observations are often noisy. For this more general setting, we have the following result, which is again similar to the OMP result in [12].

Theorem 4: For any \mathbf{x} , let

$$\Gamma_K^* = \operatorname{argmin}_{\Gamma: |\Gamma| \leq K} \left\{ \min_{\hat{\mathbf{x}}: \hat{\mathbf{x}} = \Phi_\Gamma \hat{\mathbf{y}}_\Gamma} \|\mathbf{x} - \hat{\mathbf{x}}\|_2 \right\}$$

and let

$$\hat{\mathbf{x}}_K^* = \operatorname{argmin}_{\hat{\mathbf{x}}: \hat{\mathbf{x}} = \Phi_{\Gamma_K^*} \hat{\mathbf{y}}_{\Gamma_K^*}} \|\mathbf{x} - \hat{\mathbf{x}}\|_2$$

be the best K term approximation. For any $\hat{\mathbf{x}}^{[n]} = \Phi_\Gamma \hat{\mathbf{y}}_\Gamma$ with $\Gamma \subset \Gamma^*$, if in iteration $n+1$

$$\|\mathbf{x} - \hat{\mathbf{x}}^{[n]}\|_2 > \sqrt{1 + \frac{K(1 - \mu_1(K-1))}{(\alpha - (1 + \alpha)\mu_1(K))^2}} \|\mathbf{x} - \hat{\mathbf{x}}_K^*\|_2,$$

then any *general weak MP* algorithm will only pick elements from Γ_K^* .

The proof is presented in appendix III.

An important consequence of this result holds for *general weak MP* algorithms whose residual norm decreases.

Corollary 2: Any *general weak MP* algorithm for which $\|\mathbf{x} - \hat{\mathbf{x}}^{[n]}\|_2 > \|\mathbf{x} - \hat{\mathbf{x}}^{[n+1]}\|_2$ can calculate signal approximations of the form

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_2 \leq \sqrt{1 + \frac{K(1 - \mu_1(K-1))}{(\alpha - (1 + \alpha)\mu_1(K))^2}} \|\mathbf{x} - \hat{\mathbf{x}}_K^*\|_2,$$

where $\hat{\mathbf{x}} = \Phi_\Gamma \hat{\mathbf{y}}_\Gamma$ with $\Gamma \subset \Gamma^*$.

Proof: The proof is by induction. In iteration 1, $\hat{\mathbf{x}} = \mathbf{0}$. If the error bound already holds, we would be done, if not, theorem 4 applies and the algorithm selects elements from Γ^* . Assume that up to iteration n , the algorithm has only selected elements from Γ^* . Either the assumption of the corollary holds or the algorithm will select only elements from the set Γ^* . Running the algorithm until the inequality in the corollary is reached, we see that we can only have picked correct elements on the way. ■

The corollary guarantees that the algorithm will only pick elements from Γ^* and will find an approximation which is ‘close’ in some sense to the best possible approximation. However, the corollary falls short of showing that we can in fact recover all of the set Γ^* .

D. Analysis Based on Coherence: Estimation

Whilst the above result gave guarantees in terms of the approximation accuracy achievable with *general weak Matching Pursuit* algorithms, in this section we ask the question, how good the estimation of the coefficient vector \mathbf{y} is with these methods. This is an important property in applications such as compressed sensing, where the signal of interest is the sparse vector \mathbf{y} (or an orthogonal transform thereof). We here present a result based on coherence of Φ in the case in which the algorithm has a given convergence guarantee. This condition is fulfilled for example for Matching Pursuit [15] (and therefore also for Orthogonal Matching Pursuit) as well as for the Conjugate Gradient algorithm and the Stagewise Weak Conjugate Gradient algorithm. The proof is given in appendix IV.

Theorem 5: Let $\hat{\mathbf{y}}^{[n]}$ be the estimation of any \mathbf{y} produced by any *general weak MP* algorithm that satisfies $\|\mathbf{x} - \hat{\mathbf{x}}^{[n]}\|_2 \leq \omega \|\mathbf{x} - \hat{\mathbf{x}}^{[n-1]}\|_2$ for any n and $0 \leq \omega < 1$. Let n^* be no larger than the smallest iteration count n for which the condition in corollary 2 holds. Further assume that in iteration n^* the algorithm has selected no more than K elements and that Φ and K are such that $1 - \mu_1(K-1) > 0$ and that $c = \omega^2(1 + \mu_1(K+1))/(1 - \mu_1(K+1)) < 1$, then

$$\frac{\|\hat{\mathbf{y}}^{[n]} - \mathbf{y}^*\|_2^2}{\|\mathbf{y}^*\|_2^2} \leq c^{n^*}.$$

where $\mathbf{y}^* = \Phi_{\Gamma^*}^\dagger(\Phi \mathbf{y} + \mathbf{n})$.

This theorem tells us that we can use *general weak MP* algorithms and do nearly as well as an ‘oracle’ algorithm that knew the set Γ^* of the K largest elements.

Unfortunately, the theorem tells us neither what n^* is nor, what the best K would be. Increasing K will increase c , whilst also increasing n^* . A compromise therefore needs to be struck.

Whilst an estimate of c can be based on the convergence guarantees for a particular update direction, estimation of n^* can be done following an approach similar to the argument that led to Theorem 4 in [21]. We will leave this development for a future publication and instead turn to an analysis of exact recovery based on random matrix ideas.

E. Analysis Based on Random Matrixes: Noiseless Case

The results in the previous section are universal, that is, if Φ has a given coherence, then any stagewise weak algorithm will select only correct atoms. These are the best results attainable using coherence. However, following Tropp and Gilbert [13], better bounds can be given if we draw the matrix Φ randomly from a suitable distribution, independently from a given \mathbf{y} . For a single \mathbf{y} , OMP was shown in [13] to be able to recover the correct support of \mathbf{y} with high probability when M is of the order of $K \ln(N)$. The results require Φ to be an admissible matrix, i.e. a matrix that satisfies [13]

M1 The columns of Φ are statistically independent.

M2 $E\{\|\phi_i\|_2^2\} = 1$.

M3 For any set of L vectors $\mathbf{r}_i \in \mathbb{R}^M : \|\mathbf{r}_i\|_2 \leq 1$, which are statistically independent from Φ_B and any column ϕ of Φ_B ,

$$P\{\max_i |\langle \phi, \mathbf{r}_i \rangle| \leq \epsilon\} \geq 1 - 2Le^{-c_1 \epsilon^2 M}. \quad (5)$$

M4 Any $M \times K$ sub-matrix of Φ has the k^{th} singular value σ_K larger than 0.5 with probability

$$P\{\sigma_K > 0.5\} \geq 1 - e^{-c_2 M}.$$

As shown in [13], if the entries in Φ are drawn i.i.d. from a normal or Bernoulli distribution (with the appropriate variance to satisfy M2), then Φ will satisfy the above conditions. The proof in [13] can be easily adapted to include a stagewise weak selection step. However, we develop a more general result from which an analogous result for SWOMP will follow as a corollary.

Theorem 6: (general weak MP with random measurements) Suppose that \mathbf{y} is an arbitrary K -sparse signal in \mathbb{R}^N and draw a random $M \times N$ admissible matrix Φ independent from the signal. Given the data $\mathbf{x} = \Phi \mathbf{y}$ any general weak MP algorithm will select only correct elements within the first L iterations with probability more than

$$1 - (4L(N-K))e^{-c\alpha^2 M/K}.$$

In particular, if $L \leq K$, then choosing $M \geq 2/c\alpha^2 L \log(N/\sqrt{\delta})$ guarantees that any general weak MP algorithm will only pick correct atoms during the first L iterations with probability at least $1 - \delta$. The constant is $c = \frac{1}{4} \max\{c_1, c_2\}$, where c_1 and c_2 are the constants in M3 and M4.

The proof is very similar to the one in [13], but takes account of the weakness parameter. For completeness, this proof can be found in appendix V.

The difference to the result in [13] is that we are no longer guaranteed that, if there are K non-zero elements in \mathbf{y} and if the algorithm selects only correct elements in each iteration,

the algorithm will actually have selected K different elements after K iterations. Whilst for the algorithms described in this paper, this was indeed observed experientially in many cases, this does not seem to be guaranteed in general. Therefore, setting $L = K$ in the last theorem does not guarantee that the algorithm extracts *all* of the correct elements. We can nevertheless state the following “result dependent” corollary.

Corollary 3: (general weak MP with random measurements) Suppose that \mathbf{y} is an arbitrary K -sparse signal in \mathbb{R}^N and draw a random $M \times N$ admissible matrix Φ independent from the signal. Given the data $\mathbf{x} = \Phi\mathbf{y}$ and choosing $M \geq c\alpha^{-2}K \log(N/\sqrt{\delta})$. If a general weak MP algorithm has selected K atoms in at most K iterations, then it has found the support of the signal \mathbf{y} with probability at least $1 - \delta$.

If we use an exact orthogonalisation as in OMP and SWOMP, then the residual $\mathbf{r}^{[n]}$ is orthogonal to all previously selected elements. This guarantees that the general weak selection step will pick at least one new element in each iteration. If the algorithm only picks up correct atoms, it is guaranteed to converge in at most K iterations and the above theorem gives the following corollary.

Corollary 4: (SWOMP with random measurements) Suppose that \mathbf{y} is an arbitrary K -sparse signal in \mathbb{R}^N and draw a random $M \times N$ admissible matrix Φ independent from the signal. Given the data $\mathbf{x} = \Phi\mathbf{y}$ and choosing $M \geq c\alpha^{-2}K \log(N/\sqrt{\delta})$, then SWOMP can reconstruct the signal \mathbf{y} with probability at least $1 - \delta$.

V. DISCUSSION AND CONCLUSION

Underdetermined inverse problems with a sparsity constraint on the solution are found in many areas of modern signal processing. In this paper, we have presented additional theoretical properties for several greedy strategies. The Conjugate Gradient Pursuit algorithm was introduced in an earlier paper [2]. In this paper, we have presented an additional recursion to speed up the algorithm and derived an additional convergence result.

Using the Stagewise Weak selection strategy introduced in [3] together with an OMP coefficient update does not necessarily offer computational advantages, however, strong theoretical performance guarantees for this method could be derived. The main computational advantage of the Stagewise Weak strategy was achieved when combining it with the conjugate gradient update. The coherence based performance guarantees derived here still hold in this case and similar results for random dictionaries were demonstrated. Importantly, the performance guarantees derived here are similar to those derived for OMP in [12], [13] and [21]. In conclusion, we could show that the Stagewise Weak Conjugate Gradient Pursuit algorithm is more efficient than OMP, whilst retaining similar theoretical performance guarantees.

APPENDIX I

PROOF OF THEOREM 2

A. Notation and Prerequisites

Let us first simplify notation. In this proof we will use the shorthand $\mathbf{G}_n = \Phi_{\Gamma^{[n]}}^T \Phi_{\Gamma^{[n]}}$. Also, instead of keeping track

of the size of vectors, we will assume the size to be clear from the context. For example, we use $\mathbf{d}^{[n]}$ to mean $\mathbf{d}_{\Gamma^{[n]}}^{[n]}$ as well as $\mathbf{d}_{\Gamma^{[n+1]}}^{[n]}$ (note the change in the set of elements used!). In this notation, if a vector such as $\mathbf{d}^{[n]}$ has more than n elements, all those elements not in $\Gamma^{[n]}$ will always be zero.

To simplify the expression of the cost function, we introduce the vector $\mathbf{z} = \Phi_{\Gamma^{[n]}}^\dagger \mathbf{x} - \hat{\mathbf{y}}_{\Gamma^{[n]}}$. This allows us to write the cost function at iteration n as

$$\begin{aligned} & \|\mathbf{x} - \Phi_{\Gamma^{[n]}} \hat{\mathbf{y}}_{\Gamma^{[n]}}\|_2^2 \\ & \propto -2\mathbf{x}^T \Phi_{\Gamma^{[n]}} \hat{\mathbf{y}}_{\Gamma^{[n]}} + \hat{\mathbf{y}}_{\Gamma^{[n]}}^T \mathbf{G}_n \hat{\mathbf{y}}_{\Gamma^{[n]}} \\ & \propto \mathbf{z}^T \mathbf{G}_n \mathbf{z}, \end{aligned}$$

Minimisation of the original problem is therefore equivalent to minimisation of $\frac{1}{2}\mathbf{z}^T \mathbf{G}_n \mathbf{z}$. \mathbf{G} is assumed to be strictly positive definite and therefore invertible. We abbreviate the update direction for the Conjugate Gradient approach by $\mathbf{d}_{CG}^{[n]}$ and the direction for the gradient based approach by $\mathbf{d}_G^{[n]}$.

The proof uses the following equalities, which are either definitions or easily proven.

- E1 $\langle \mathbf{d}^{[n]}, \mathbf{G}_m \mathbf{d}^{[n-1]} \rangle = \langle \mathbf{d}^{[n]}, \mathbf{G}_n \mathbf{d}^{[n-1]} \rangle$, whenever $n \leq m$.
- E2 $\mathbf{g}^{[n+1]} = \mathbf{G}_{n+1} \mathbf{z}^{[n]}$ or $\mathbf{z}^{[n]} = \mathbf{G}_{n+1}^{-1} \mathbf{g}^{[n+1]}$.
- E3 $\langle \mathbf{z}^{[n-1]}, \mathbf{G}_{n+1} \mathbf{d}^{[n]} \rangle \stackrel{E1}{=} \langle \mathbf{G}_n \mathbf{z}^{[n-1]}, \mathbf{d}^{[n]} \rangle \stackrel{E2}{=} \langle \mathbf{g}^{[n]}, \mathbf{d}^{[n]} \rangle$.
- E4 Define: $\mathbf{d}_{CG}^{[n]} = \mathbf{g}^{[n]} = \Phi_{\Gamma^{[n]}} (\mathbf{x} - \Phi \mathbf{y}^{[n-1]})$.
- E5 Define: $\mathbf{d}_{CG}^{[n]} = \mathbf{g}^{[n]} + \omega \mathbf{d}_{CG}^{[n-1]}$.
- E6 ω is chosen such that conjugacy holds

$$\omega = -\frac{\langle \mathbf{g}^{[n]}, \mathbf{G}_n \mathbf{d}_{CG}^{[n-1]} \rangle}{\langle \mathbf{d}_{CG}^{[n-1]}, \mathbf{G}_n \mathbf{d}_{CG}^{[n-1]} \rangle} \stackrel{E1}{=} -\frac{\langle \mathbf{g}^{[n]}, \mathbf{G}_{n+1} \mathbf{d}_{CG}^{[n-1]} \rangle}{\langle \mathbf{d}_{CG}^{[n-1]}, \mathbf{G}_{n+1} \mathbf{d}_{CG}^{[n-1]} \rangle}.$$

E7 The optimum step size is

$$\mu = \frac{\langle \mathbf{g}^{[n]}, \mathbf{d}^{[n]} \rangle}{\langle \mathbf{d}^{[n]}, \mathbf{G}_n \mathbf{d}^{[n]} \rangle} \stackrel{E1}{=} \frac{\langle \mathbf{g}^{[n]}, \mathbf{d}^{[n]} \rangle}{\langle \mathbf{d}^{[n]}, \mathbf{G}_{n+1} \mathbf{d}^{[n]} \rangle}.$$

- E8 $\mathbf{y}^{[n]} = \mathbf{y}^{[n-1]} + \mu \mathbf{d}^{[n]}$ implies $\mathbf{z}^{[n]} = \mathbf{z}^{[n-1]} - \mu \mathbf{d}^{[n]}$.
- E9 In subsection I-D we show that $\langle \mathbf{g}^{[n]}, \mathbf{d}^{[n-1]} \rangle = 0$.
- E10 Conjugacy and E1 imply $\langle \mathbf{d}_{CG}^{[n]}, \mathbf{G}_{n+1} \mathbf{d}_{CG}^{[n-1]} \rangle = 0$.

B. Proof Outline

The proof shows the following

P1 For any update direction $\mathbf{d}^{[n]}$, the cost is proportional to

$$\langle \mathbf{z}^{[n-1]}, \mathbf{G}_{n+1} \mathbf{z}^{[n-1]} \rangle - \frac{\langle \mathbf{g}^{[n]}, \mathbf{d}^{[n]} \rangle^2}{\langle \mathbf{d}^{[n]}, \mathbf{G}_{n+1} \mathbf{d}^{[n]} \rangle}.$$

P2 Using $\mathbf{d}_G^{[n]}$, this becomes trivially

$$\langle \mathbf{z}^{[n-1]}, \mathbf{G}_{n+1} \mathbf{z}^{[n-1]} \rangle - \frac{\langle \mathbf{g}^{[n]}, \mathbf{g}^{[n]} \rangle^2}{\langle \mathbf{g}^{[n]}, \mathbf{G}_{n+1} \mathbf{g}^{[n]} \rangle}.$$

P3 Using $\mathbf{d}_{CG}^{[n]}$, it can be shown that the cost is

$$\langle \mathbf{z}^{[n-1]}, \mathbf{G}_{n+1} \mathbf{z}^{[n-1]} \rangle - \frac{\langle \mathbf{g}^{[n]}, \mathbf{g}^{[n]} \rangle^2}{\langle \mathbf{d}_{CG}^{[n]}, \mathbf{G}_{n+1} \mathbf{d}_{CG}^{[n]} \rangle}.$$

P4 It therefore remains to show that

$$\langle \mathbf{d}_{CG}^{[n]}, \mathbf{G}_{n+1} \mathbf{d}_{CG}^{[n]} \rangle \leq \langle \mathbf{g}^{[n]}, \mathbf{G}_{n+1} \mathbf{g}^{[n]} \rangle,$$

so that the cost in P3 is not more than that in P2.

C. Details

To show P1 we write

$$\begin{aligned}
& \langle \mathbf{z}^{[n]}, \mathbf{G}_{n+1} \mathbf{z}^{[n]} \rangle \\
\stackrel{E2}{=} & \langle \mathbf{g}^{[n+1]}, \mathbf{G}_{n+1}^{-1} \mathbf{g}^{[n+1]} \rangle \\
\stackrel{E2, E8}{=} & (\mathbf{G}_{n+1} \mathbf{z}^{[n-1]} - \mu \mathbf{G}_{n+1} \mathbf{d}^{[n]})^T \mathbf{G}_{n+1}^{-1} \\
& \cdot (\mathbf{G}_{n+1} \mathbf{z}^{[n-1]} - \mu \mathbf{G}_{n+1} \mathbf{d}^{[n]}) \\
= & \langle \mathbf{z}^{[n-1]}, \mathbf{G}_{n+1} \mathbf{z}^{[n-1]} \rangle \\
& - 2\mu \langle \mathbf{z}^{[n-1]}, \mathbf{G}_{n+1} \mathbf{d}^{[n]} \rangle \\
& + \mu^2 \langle \mathbf{d}^{[n]}, \mathbf{G}_{n+1} \mathbf{d}^{[n]} \rangle \\
\stackrel{E3}{=} & \langle \mathbf{z}^{[n-1]}, \mathbf{G}_{n+1} \mathbf{z}^{[n-1]} \rangle \\
& - 2\mu \langle \mathbf{g}^{[n]}, \mathbf{d}^{[n]} \rangle \\
& + \mu^2 \langle \mathbf{d}^{[n]}, \mathbf{G}_{n+1} \mathbf{d}^{[n]} \rangle \\
\stackrel{E7}{=} & \langle \mathbf{z}^{[n-1]}, \mathbf{G}_{n+1} \mathbf{z}^{[n-1]} \rangle \\
& - \frac{\langle \mathbf{g}^{[n]}, \mathbf{d}^{[n]} \rangle^2}{\langle \mathbf{d}^{[n]}, \mathbf{G}_{n+1} \mathbf{d}^{[n]} \rangle}.
\end{aligned}$$

P2 follows trivially from the definition E4.

P3 follows from E9

$$\begin{aligned}
\langle \mathbf{g}^{[n]}, \mathbf{d}_{CG}^{[n]} \rangle & \stackrel{E5}{=} \langle \mathbf{g}^{[n]}, \mathbf{g}^{[n]} \rangle + \omega \langle \mathbf{g}^{[n]}, \mathbf{d}_{CG}^{[n-1]} \rangle \\
& \stackrel{E9}{=} \langle \mathbf{g}^{[n]}, \mathbf{g}^{[n]} \rangle.
\end{aligned}$$

Finally to show P4,

$$\begin{aligned}
& \langle \mathbf{d}_{CG}^{[n]}, \mathbf{G}_{n+1} \mathbf{d}_{CG}^{[n]} \rangle \\
\stackrel{E5, E10}{=} & \langle \mathbf{g}^{[n]}, \mathbf{G}_{n+1} \mathbf{d}_{CG}^{[n]} \rangle \\
\stackrel{E5}{=} & \langle \mathbf{g}^{[n]}, \mathbf{G}_{n+1} \mathbf{g}^{[n]} \rangle \\
& + \omega \langle \mathbf{g}^{[n]}, \mathbf{G}_{n+1} \mathbf{d}_{CG}^{[n-1]} \rangle \\
\stackrel{E6}{=} & \langle \mathbf{g}^{[n]}, \mathbf{G}_{n+1} \mathbf{g}^{[n]} \rangle \\
& - (\langle \mathbf{g}^{[n]}, \mathbf{G}_{n+1} \mathbf{d}_{CG}^{[n-1]} \rangle^2 / \langle \mathbf{d}_{CG}^{[n-1]}, \mathbf{G}_{n+1} \mathbf{d}_{CG}^{[n-1]} \rangle).
\end{aligned}$$

D. Proof of $\langle \mathbf{g}^{[n]}, \mathbf{d}^{[n-1]} \rangle = 0$

$$\begin{aligned}
\mathbf{g}^{[n]} & \stackrel{E2}{=} \mathbf{G}_n \mathbf{z}^{[n-1]} \\
& \stackrel{E8}{=} \mathbf{G}_n \mathbf{z}^{[n-2]} - \mu \mathbf{G}_n \mathbf{d}^{[n-1]} \\
& \stackrel{E7}{=} \mathbf{G}_n \mathbf{z}^{[n-2]} \\
& - \frac{\langle \mathbf{g}^{[n-1]}, \mathbf{d}^{[n-1]} \rangle}{\langle \mathbf{d}^{[n-1]}, \mathbf{G}_n \mathbf{d}^{[n-1]} \rangle} \mathbf{G}_n \mathbf{d}^{[n-1]}.
\end{aligned}$$

With this the inner product becomes

$$\begin{aligned}
& \langle \mathbf{g}^{[n]}, \mathbf{d}^{[n-1]} \rangle \\
= & \langle \mathbf{d}^{[n-1]}, \mathbf{G}_n \mathbf{z}^{[n-2]} \rangle \\
& - \frac{\langle \mathbf{g}^{[n-1]}, \mathbf{d}^{[n-1]} \rangle}{\langle \mathbf{d}^{[n-1]}, \mathbf{G}_n \mathbf{d}^{[n-1]} \rangle} \langle \mathbf{d}^{[n-1]}, \mathbf{G}_n \mathbf{d}^{[n-1]} \rangle \\
= & \langle \mathbf{z}^{[n-2]}, \mathbf{G}_n \mathbf{d}^{[n-1]} \rangle - \langle \mathbf{g}^{[n-1]}, \mathbf{d}^{[n-1]} \rangle \\
\stackrel{E3}{=} & 0.
\end{aligned}$$

APPENDIX II PROOF OF THEOREM 3

We prove that for any $\mathbf{r} \in \text{span}\{\Phi_{\Gamma^*}\}$, $\text{ERC}_\alpha(\Gamma^*)$ implies that $\{i : |g_i| \geq \alpha \max_j |\Phi^T \mathbf{r}^{[n]}|\} \subset \Gamma^*$, that is as long as $\mathbf{r}^{[n]}$ lies in the span of the correct elements, the algorithm will select more of the correct elements. $\mathbf{r}^{[0]} = \mathbf{x} \in \text{span}\{\Phi_{\Gamma^*}\}$ is trivially true. By induction, if up to iteration n we only selected correct elements, then, whatever the vector $\hat{\mathbf{y}}_{\Gamma^{[n-1]}}$, $\mathbf{r}^{[n-1]} = \mathbf{x} - \Phi_{\Gamma^{[n-1]}} \hat{\mathbf{y}}_{\Gamma^{[n-1]}}$ will also be in the span of Φ_{Γ^*} . It therefore remains to show that if $\text{ERC}_\alpha(\Gamma^*)$ holds for any vector \mathbf{r} in the span of Φ_{Γ^*} , then all indices i for which $\phi_i^T \mathbf{r} \geq \alpha \max_j |\Phi^T \mathbf{r}^{[n]}|$ are in the set Γ^* .

To show this, split the matrix Φ into two sub-matrices. Let $\Phi_{\Gamma^*} = \Phi_G$ be the *good* matrix with those elements in the true support set Γ^* and let Φ_B be the *bad* matrix with the remaining columns. All we need to show is that for any \mathbf{r} in the span of Φ_G

$$\frac{\|\Phi_B^T \mathbf{r}\|_\infty}{\|\Phi_G^T \mathbf{r}\|_\infty} < \alpha.$$

Because $\mathbf{r} = \Phi_G \mathbf{y}_G$ for some \mathbf{y}_G and using the identity $\Phi_G = (\Phi_G^\dagger)^T \Phi_G^T \Phi_G$, we have

$$\begin{aligned}
\frac{\|\Phi_B^T \mathbf{r}\|_\infty}{\|\Phi_G^T \mathbf{r}\|_\infty} & = \frac{\|\Phi_B^T (\Phi_G^\dagger)^T \Phi_G^T \mathbf{r}\|_\infty}{\|\Phi_G^T \mathbf{r}\|_\infty} \\
& \leq \|\Phi_B^T (\Phi_G^\dagger)^T\|_\infty \\
& = \|\Phi_G^\dagger \Phi_B\|_1 = \text{ERC}_\alpha(\Gamma^*) \\
& < \alpha.
\end{aligned}$$

APPENDIX III PROOF OF THEOREM 4

Let Γ^* be defined as

$$\Gamma^* = \underset{\Gamma: |\Gamma| \leq 0}{\text{argmin}} \min_{\mathbf{y}_\Gamma} \|\mathbf{x} - \Phi_\Gamma \mathbf{y}_\Gamma\|_2, \quad (6)$$

that is, Γ^* is the K element set which allows the best approximation to a signal \mathbf{x} . Let $\mathbf{y}_\Gamma^* = \min_{\mathbf{y}_\Gamma} \|\mathbf{x} - \Phi_\Gamma \mathbf{y}_\Gamma\|_2$ and

$$\mathbf{r}^* = \mathbf{x} - \Phi_{\Gamma^*} \mathbf{y}_{\Gamma^*}^*$$

In iteration n , assume the algorithm has recovered a set $\Gamma^{[n]}$ of elements from the set Γ^* . Let the residual be $\mathbf{r}^{[n]} = \mathbf{x} - \Phi_{\Gamma^{[n]}} \mathbf{y}_{\Gamma^{[n]}}^{[n]} = \mathbf{x} - \hat{\mathbf{x}}^{[n]}$. Let $\hat{\mathbf{x}}^* = \Phi \mathbf{y}^*$ be the best K term error. Again, let the sets $G = \Gamma^*$ and $B = G^\perp$ be the good and bad sets. We then have

$$\begin{aligned}
\frac{\|\Phi_B^T \mathbf{r}^{[n]}\|_\infty}{\|\Phi_G^T \mathbf{r}^{[n]}\|_\infty} & = \frac{\|\Phi_B^T (\mathbf{x} - \hat{\mathbf{x}}^*) + \Phi_B^T (\hat{\mathbf{x}}^* - \hat{\mathbf{x}}^{[n]})\|_\infty}{\|\Phi_G^T (\mathbf{x} - \hat{\mathbf{x}}^*) + \Phi_G^T (\hat{\mathbf{x}}^* - \hat{\mathbf{x}}^{[n]})\|_\infty} \\
& \leq \frac{\|\Phi_B^T (\mathbf{x} - \hat{\mathbf{x}}^*)\|_\infty}{\|\Phi_G^T (\hat{\mathbf{x}}^* - \hat{\mathbf{x}}^{[n]})\|_\infty} + \frac{\|\Phi_B^T (\hat{\mathbf{x}}^* - \hat{\mathbf{x}}^{[n]})\|_\infty}{\|\Phi_G^T (\hat{\mathbf{x}}^* - \hat{\mathbf{x}}^{[n]})\|_\infty}.
\end{aligned}$$

Note that due to the optimality of $\hat{\mathbf{x}}^*$, the error $(\mathbf{x} - \hat{\mathbf{x}}^*)$ is orthogonal to all elements in Φ_G . The term on the right hand side can be bounded in the same way as was done in appendix II

$$\frac{\|\Phi_B^T (\hat{\mathbf{x}}^* - \hat{\mathbf{x}}^{[n]})\|_\infty}{\|\Phi_G^T (\hat{\mathbf{x}}^* - \hat{\mathbf{x}}^{[n]})\|_\infty} \leq \max_{i \in B} \|\Phi^\dagger \phi_i\|_1. \quad (7)$$

The other term can be bounded as

$$\begin{aligned} \frac{\|\Phi_B^T(\mathbf{x} - \hat{\mathbf{x}}^*)\|_\infty}{\|\Phi_G^T(\hat{\mathbf{x}}^* - \hat{\mathbf{x}}^{[n]})\|_\infty} &\leq \frac{\max_{\phi_i, i \in B} \|\phi_i\|_2 \|\mathbf{x} - \hat{\mathbf{x}}^*\|_2}{\sqrt{K-1} \|\Phi_G(\hat{\mathbf{x}}^* - \hat{\mathbf{x}}^{[n]})\|_2} \\ &\leq \frac{\sqrt{K} \|\mathbf{x} - \hat{\mathbf{x}}^*\|_2}{\sigma_K(\Phi_G) \|\hat{\mathbf{x}}^* - \hat{\mathbf{x}}^{[n]}\|_2}. \end{aligned}$$

In the first line we have used the Cauchy-Schwarz inequality to bound the maximum inner product in the numerator and standard norm inequalities for the denominator. In the second line we then used $\sigma_K(\Phi_G)$ to be the K^{th} singular value of Φ_G . In order for this bound to make sense, this needs to be non-zero, so we require Φ_G to be of full rank.

Therefore, to guarantee correct element selection, it is sufficient that

$$\frac{\|\Phi_B^T \mathbf{r}^{[n]}\|_\infty}{\|\Phi_G^T \mathbf{r}^{[n]}\|_\infty} \leq \frac{\sqrt{K} \|\mathbf{x} - \hat{\mathbf{x}}^*\|_2}{\sigma_K(\Phi_G) \|\hat{\mathbf{x}}^* - \hat{\mathbf{x}}^{[n]}\|_2} + \max_{i \in B} \|\Phi^\dagger \phi_i\|_1 < \alpha.$$

To state this in terms of the cumulative coherence, we note that [22]

$$\sigma_K(\Phi_G) \geq \sqrt{1 - \mu_1(K-1)}.$$

To bound $\max_{i \in B} \|\Phi^\dagger \phi_i\|_1$, note that [22]

$$\max_{i \in B} \|\Phi^\dagger \phi_i\|_1 \leq \frac{\mu_1(K)}{1 - \mu_1(K-1)}.$$

With these inequalities, we see that it is sufficient that

$$\|\hat{\mathbf{x}}^* - \hat{\mathbf{x}}^{[n]}\|_2 > \frac{\sqrt{K}(1 - \mu_1(K-1))}{\alpha - (1 + \alpha)\mu_1(K)} \|\mathbf{x} - \hat{\mathbf{x}}^*\|_2$$

or, using orthogonality (i.e. $\|\mathbf{x} - \hat{\mathbf{x}}^{[n]}\|_2^2 = \|\hat{\mathbf{x}}^* - \hat{\mathbf{x}}^{[n]}\|_2^2 + \|\mathbf{x} - \hat{\mathbf{x}}^*\|_2^2$),

$$\|\mathbf{x} - \hat{\mathbf{x}}^{[n]}\|_2 > \sqrt{1 + \frac{K(1 - \mu_1(K-1))}{(\alpha - (1 + \alpha)\mu_1(K))^2}} \|\mathbf{x} - \hat{\mathbf{x}}^*\|_2.$$

APPENDIX IV PROOF OF THEOREM 5

Let \mathbf{y}^* be the orthogonal projection of $\mathbf{x} = \Phi \mathbf{y} + \mathbf{n}$ onto the span of Φ_{Γ^*} , where Γ^* is the set of K elements, for which $\|\mathbf{y}_{\Gamma^*}\|_2$ is largest. Let $\mathbf{e} = \mathbf{x} - \Phi \mathbf{y}^*$. Note that \mathbf{e} is orthogonal to $\Phi \mathbf{y}^*$ by definition. By the assumption of linear convergence of the algorithm, there is an $\omega < 1$, such that

$$\|\Phi(\hat{\mathbf{y}}^{[n]} - \mathbf{y}^*) - \mathbf{e}\|_2^2 \leq \omega^2 \|\Phi(\hat{\mathbf{y}}^{[n-1]} - \mathbf{y}^*) - \mathbf{e}\|_2^2.$$

By corollary 2 and the assumption on n^* , the algorithm will only have selected elements from Γ^* , that is, the support of $\hat{\mathbf{y}}^{[n]} \subset \Gamma^*$. Therefore $\Phi(\hat{\mathbf{y}}^{[n]} - \mathbf{y}^*)$ and \mathbf{e} are orthogonal and we can use Pythagoras to decompose $\|\Phi(\hat{\mathbf{y}}^{[n]} - \mathbf{y}^*) - \mathbf{e}\|_2^2$ and $\|\Phi(\hat{\mathbf{y}}^{[n-1]} - \mathbf{y}^*) - \mathbf{e}\|_2^2$, which gives

$$\begin{aligned} \|\Phi(\hat{\mathbf{y}}^{[n]} - \mathbf{y}^*)\|_2^2 + \|\mathbf{e}\|_2^2 &\leq \\ \omega^2 \|\Phi(\hat{\mathbf{y}}^{[n-1]} - \mathbf{y}^*)\|_2^2 + \omega^2 \|\mathbf{e}\|_2^2 & \end{aligned}$$

Using the fact that the largest and K^{th} squared singular values of Φ_{Γ^*} are bounded by $(1 + \mu_1(K+1))$ and $(1 - \mu_1(K+1))$, we have

$$\begin{aligned} (1 - \mu_1(K-1)) \|\hat{\mathbf{y}}^{[n]} - \mathbf{y}^*\|_2^2 &\leq \\ \omega^2 (1 + \mu_1(K-1)) \|\hat{\mathbf{y}}^{[n-1]} - \mathbf{y}^*\|_2^2 + (\omega^2 - 1) \|\mathbf{e}\|_2^2 & \end{aligned}$$

Because $\omega < 1$ this implies

$$\|\hat{\mathbf{y}}^{[n]} - \mathbf{y}^*\|_2^2 \leq c \|\hat{\mathbf{y}}^{[n-1]} - \mathbf{y}^*\|_2^2,$$

where we use $c = \omega^2(1 + \mu_1(K+1))/(1 - \mu_1(K+1))$. (Note that we could state this constant equally well in terms of the restricted isometry constant.) By induction we therefore have

$$\|\hat{\mathbf{y}}^{[n]} - \mathbf{y}^*\|_2^2 \leq c^{n^*} \|\mathbf{y}^*\|_2^2.$$

APPENDIX V PROOF OF THEOREM 6

Proof: We use the notation $\Phi_G = \Phi_{\Gamma^*}$, where Γ^* is the support set of \mathbf{y} and define the *bad* set Φ_B to be the sub-matrix of Φ containing only elements not in Γ^* . Assume the algorithm is successful, i.e it only selects elements from Φ_G . In this case, let $\mathbf{r}_G^{[n]}$ be the residual in iteration n . This is exactly the same sequence of residuals produced as if we run the algorithm using Φ_G only, that is the sequence of residuals generated by the algorithm only depends on Φ_G and *not* on Φ_B . Therefore, the sequence $\mathbf{r}_G^{[n]}$ is independent from Φ_B given that the algorithm succeeds.

We now argue by induction. Assume the algorithm has selected only correct elements up to iteration n . The residual $\mathbf{r}_G^{[n]}$ is the same residual we would have found if we had run the algorithm with Φ_G only. $\mathbf{r}_G^{[n]}$ is therefore independent from Φ_B . For the algorithm to select only elements from Φ_G , we require that

$$\frac{\|\Phi_B^T \mathbf{r}^{[n]}\|_\infty}{\|\Phi_G^T \mathbf{r}^{[n]}\|_\infty} < \alpha.$$

If we run the algorithm for L iterations, we want to bound the probability that all of these fractions are below α , that is

$$P \left\{ \bigcup_{n \in \{1, 2, \dots, L\}} \left(\frac{\|\Phi_B^T \mathbf{r}^{[n]}\|_\infty}{\|\Phi_G^T \mathbf{r}^{[n]}\|_\infty} < \alpha \right) \right\}.$$

We can bound this probability from below by the probability

$$P \left\{ \bigcup_{n \in \{1, 2, \dots, L\}} \left(\frac{\|\Phi_B^T \mathbf{r}^{[n]}\|_\infty}{\|\Phi_G^T \mathbf{r}^{[n]}\|_\infty} < \alpha \right) \text{ and } \Sigma \right\}$$

where Σ is the event that Φ_G has the K^{th} singular value larger than 0.5. This probability can be written using the definition of conditional probability

$$P \left\{ \bigcup_{n \in \{1, 2, \dots, L\}} \left(\frac{\|\Phi_B^T \mathbf{r}^{[n]}\|_\infty}{\|\Phi_G^T \mathbf{r}^{[n]}\|_\infty} < \alpha \right) \middle| \Sigma \right\} P\{\Sigma\}.$$

Condition M4 bounds $P\{\Sigma\}$. It therefore remains to bound the conditional probability. Using a standard norm inequality we have

$$\frac{\|\Phi_B^T \mathbf{r}^{[n]}\|_\infty}{\|\Phi_G^T \mathbf{r}^{[n]}\|_\infty} \leq \sqrt{K} \frac{\|\Phi_B^T \mathbf{r}^{[n]}\|_\infty}{\|\Phi_G^T \mathbf{r}^{[n]}\|_2} \leq 2\sqrt{K} \frac{\|\Phi_B^T \mathbf{r}^{[n]}\|_\infty}{\|\mathbf{r}^{[n]}\|_2} \quad (8)$$

where the last inequality comes from the condition Σ , which states that the K^{th} singular value of Φ_G is greater than 0.5.

We now use the normalisation

$$\mathbf{u}^{[n]} = \frac{\mathbf{r}^{[n]}}{\|\mathbf{r}^{[n]}\|_2}.$$

Inserting $\mathbf{r}^{[n]} = \mathbf{u}^{[n]} \|\mathbf{r}^{[n]}\|_2$ into the right hand side of inequality (8), we get

$$\frac{\|\Phi_B^T \mathbf{r}^{[n]}\|_\infty}{\|\Phi_G^T \mathbf{r}^{[n]}\|_\infty} \leq 2\sqrt{K} \|\Phi_B^T \mathbf{u}^{[n]}\|_\infty.$$

As argued above, $\mathbf{r}^{[n]}$ and therefore $\mathbf{u}^{[n]}$ are independent from Φ_B so that M3 holds for the L vectors $\mathbf{u}^{[n]}$. The conditional probability is therefore lower bounded by

$$\begin{aligned} & P \left\{ \bigcup_{n \in \{1, 2, \dots, L\}} \left(\frac{\|\Phi_B^T \mathbf{r}^{[n]}\|_\infty}{\|\Phi_G^T \mathbf{r}^{[n]}\|_\infty} < \alpha \right) \middle| \Sigma \right\} \\ & \geq P \left\{ \bigcup_{n \in \{1, 2, \dots, L\}} \left(\|\Phi_B^T \mathbf{u}^{[n]}\|_\infty < \frac{\alpha}{2\sqrt{K}} \right) \middle| \Sigma \right\} \\ & = P \left\{ \bigcup_{n \in \{1, 2, \dots, L\}} \bigcup_{i \notin \Gamma^*} \left(|\langle \phi_i, \mathbf{u}^{[n]} \rangle| < \frac{\alpha}{2\sqrt{K}} \right) \middle| \Sigma \right\} \\ & = \prod_{i \notin \Gamma^*} P \left\{ \bigcup_{n \in \{1, 2, \dots, L\}} \left(|\langle \phi_i, \mathbf{u}^{[n]} \rangle| < \frac{\alpha}{2\sqrt{K}} \right) \middle| \Sigma \right\} \end{aligned}$$

In the second to last line, we have used the fact that in order for $\|\Phi_B^T \mathbf{u}^{[n]}\|_\infty < \frac{\alpha}{2\sqrt{K}}$ it is necessary that $|\langle \phi_i, \mathbf{u}^{[n]} \rangle| < \frac{\alpha}{2\sqrt{K}}$ for all $i \notin \Gamma^*$. We have then interchanged the unions and used the independence of the ϕ_i to split the probability into the product in the last line.

There are $N - K$ elements not in Γ^* . Using property M3, we then have

$$\begin{aligned} & P \left\{ \bigcup_{n \in \{1, 2, \dots, L\}} \left(\frac{\|\Phi_B^T \mathbf{r}^{[n]}\|_\infty}{\|\Phi_G^T \mathbf{r}^{[n]}\|_\infty} < \alpha \right) \middle| \Sigma \right\} \\ & \geq \left[1 - 2Le^{-c\alpha^2 M/(4K)} \right]^{N-K} \end{aligned}$$

Overall, the probability of success is larger than

$$\left[1 - 2Le^{-c\alpha^2 M/(4K)} \right]^{N-K} \left[1 - e^{-cM} \right].$$

This can be cleaned up using the inequality $(1 - x)^k \geq 1 - kx$, valid for $k > 1$ and $x < 1$. Dropping one positive term, we then get a probability of success of more than

$$\begin{aligned} & 1 - 2L(N - K)e^{-c\alpha^2 M/(4K)} - e^{-cM} \\ & \geq 1 - (4L(N - K))e^{-c\alpha^2 M/(4K)} \end{aligned}$$

If $L \leq K$, then we can use $L(N - K) \leq K(N - K) \leq N^2/4$ and get a success with probability more than

$$1 - N^2 e^{-c\alpha^2 M/K}$$

Therefore, if $L \leq K$, the failure probability is $\delta \leq N^2 e^{-c\alpha^2 M/K}$. Choosing $M \geq c\alpha^{-2} K \ln(N/\delta)$ therefore is sufficient to guarantee a failure probability of less than δ . ■

REFERENCES

- [1] M. E. Davies and T. Blumensath, "Faster & greedier: algorithms for sparse reconstruction of large datasets," in *Proc. of the third IEEE International Symposium on Communications, Control, and Signal Processing*, (St Julians, Malta), March 2008.
- [2] T. Blumensath and M. Davies, "Gradient pursuits," *IEEE Transactions on Signal Processing*, vol. 56, pp. 2370–2382, June 2008.
- [3] T. Blumensath and M. E. Davies, "Stagewise weak gradient pursuits. Part I: Fundamentals and numerical studies," *submitted for publication*, 2008.
- [4] Y. C. Pati, R. Rezaifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition," in *27th Asilomar Conf. on Signals, Systems and Comput.*, Nov. 1993.
- [5] R. Gribonval and M. Nielsen, "Sparse decompositions in "incoherent" dictionaries," in *Proc. IEEE Intl. Conf. on Image Proc. (ICIP'03)*, (Barcelona, Spain), September 2003.
- [6] J.-J. Fuchs, "On sparse representations in arbitrary redundant bases," *IEEE Transactions on Information Theory*, vol. 50, no. 6, pp. 1341–1344, 2004.
- [7] J.-J. Fuchs, "Recovery of exact sparse representations in the presence of bounded noise," tech. rep., Institut de Recherche en Informatique et Systemes Aléatoires, 2004.
- [8] D. L. Donoho and M. Elad, "Optimally-sparse representation in general (non-orthogonal) dictionaries via ℓ_1 minimization," *Proc. Nat. Acc. Sci.*, vol. 100, pp. 2197–2202, 2003.
- [9] D. L. Donoho, M. Elad, and V. Temlyakov, "Stable recovery of sparse overcomplete representations in the presence of noise," *IEEE Transactions On Information Theory*, February 2004.
- [10] D. Donoho, "Compressed sensing," *IEEE Trans. on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.
- [11] E. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information.," *IEEE Transactions on information theory*, vol. 52, pp. 489–509, Feb 2006.
- [12] J. A. Tropp, "Greed is good: algorithmic results for sparse approximation," *IEEE Transactions on Information Theory*, vol. 50, no. 10, pp. 2231–2242, 2004.
- [13] J. A. Tropp and A. C. Gilbert, "Signal recovery from partial information via orthogonal matching pursuit," *Submitted for publication.*, 2006.
- [14] B. Efron, T. Hastie, I. Johnston, and R. Tibshirani, "Least angle regression," *The Annals of Statistics*, vol. 32, no. 2, pp. 407–499, 2004.
- [15] S. Mallat, *A Wavelet Tour of Signal Processing*. Academic Press, 1999.
- [16] D. Donoho, Y. Tsaig, I. Drori, and J. Starck, "Sparse solutions of underdetermined linear equations by stagewise orthogonal matching pursuit," 2006.
- [17] E. J. Candès and T. Tao, "Decoding by linear programming," *IEEE Transactions on Information Theory*, vol. 51, pp. 4203–4215, 2004.
- [18] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge University Press, 1985.
- [19] J. R. Shewchuk, "An introduction to the conjugate gradient method without the agonizing pain," tech. rep., School of Computer Science, Carnegie Mellon University, 1994.
- [20] H. Crowder and P. Wolfe, "Linear convergence of the conjugate gradient method," *Numerical Computation*, vol. 16, no. 4, pp. 431–433, 1972.
- [21] R. Gribonval and P. Vandergheynst, "On the exponential convergence of matching pursuits in quasi-incoherent dictionaries," *IEEE Transactions on Information Theory*, vol. 52, no. 1, pp. 255–261, 2006.
- [22] J. Tropp, "Just relax: Convex programming methods for identifying sparse signals in noise," *IEEE Transactions on Information Theory*, vol. 51, no. 3, pp. 1031–1051, 2006.