Appl. Comput. Harmon. Anal. 41 (2016) 852–866

Contents lists available at ScienceDirect

Applied and Computational Harmonic Analysis

www.elsevier.com/locate/acha

Letter to the Editor

# Rescaled pure greedy algorithm for Hilbert and Banach spaces ☆

Guergana Petrova

*Department of Mathematics, Texas A&M University, College Station, TX 77843, USA*

ARTICLE INFO

*Article history:*
Received 14 May 2015
Received in revised form 27 September 2015
Accepted 20 October 2015
Available online 26 October 2015
Communicated by Leslie F. Greengard

*MSC:*
41A25
41A46

*Keywords:*
Greedy algorithms
Rates of convergence

ABSTRACT

We show that a very simple modification of the Pure Greedy Algorithm for approximating functions by sparse sums from a dictionary in a Hilbert or more generally a Banach space has optimal convergence rates.



## 1. Introduction

Greedy algorithms have been used quite extensively as a tool for generating approximations from redundant families of functions, such as frames or more general dictionaries $\mathcal{D}$, see [8,10,3,2,15]. Given a Banach space $X$, a dictionary is any set $\mathcal{D}$ of norm one elements from $X$ whose span is dense in $X$. The most natural greedy algorithm in a Hilbert space is the Pure Greedy Algorithm (**PGA**), which is also known as Matching Pursuit, see [3] for the description of this and other algorithms. The fact that the **PGA** lacks optimal convergence properties has led to a variety of modified greedy algorithms such as the Relaxed Greedy Algorithm (**RGA**), the Orthogonal Greedy Algorithm (**OGA**), and their weak versions. There are also analogues of these, developed for approximating functions in Banach spaces, see [15].

The central issues in the study of these algorithms is their ease of implementation and their approximation power, measured in terms of convergence rates. If $f_m$ is the output of a greedy algorithm after $m$ iterations, then $f_m$ is a linear combination of at most $m$ dictionary elements. Such linear combinations are said to be

☆ This research was supported by the Office of Naval Research Contract ONR N00014-11-1-0712, and by the NSF Grants DMS 1222715 and DMS 1521067.

*E-mail address:* gpetrova@math.tamu.edu.

http://dx.doi.org/10.1016/j.acha.2015.10.008
1063-5203/

sparse of order $m$. The quality of the approximation is measured by the decay of the error $\|f - f_m\|$ as $m \to \infty$, where $\|\cdot\|$ is the norm in the Hilbert or Banach space, respectively. Of course, the decay rate of this error is governed by properties of the target function $f$. The typical properties imposed on $f$ are that it is sparse, or more generally, that it is in some way compressible. Here, compressible means that it can be written as a (generally speaking, infinite) linear combination of dictionary elements with some restrictions on the coefficients. The most frequently applied assumption on $f$ is that it is in the unit ball of the class $\mathcal{A}_1(\mathcal{D})$, that is the set of all functions which are a convex combination of dictionary elements (provided we consider symmetric dictionaries). It is known that the elements in this class can be approximated by $m$ sparse vectors to accuracy $\mathcal{O}(m^{-1/2})$, see Theorem 2.1, and so this rate of approximation serves as a benchmark for the performance of greedy algorithms.

It has been shown in [3] in the case of Hilbert space that whenever $f \in \mathcal{A}_1(\mathcal{D})$, the output $f_m$ of the **PGA** satisfies

$$\|f - f_m\| = \mathcal{O}(m^{-1/6}), \quad m \to \infty.$$

Later results gave slight improvements of the above estimate. For example, in [9], the rate $\mathcal{O}(m^{-1/6})$ was improved to $\mathcal{O}(m^{-11/62})$. Based on the method from the latter paper, Sil'nichenko [14] then showed a rate of $\mathcal{O}(m^{-\frac{s}{2(s+2)}})$, where $s$ solves a certain equation, and that $\frac{s}{2(s+2)} > 11/62$. Similar estimates for the weak versions of the **PGA** can be found in [15]. Estimates for the error from below have also been provided, see [12,11].

The fact that the **PGA** does not attain the optimal rate for approximating the elements in $\mathcal{A}_1(\mathcal{D})$ has led to various modifications of this algorithm. Two of these modifications, the Relaxed and the Orthogonal Greedy Algorithm were shown to achieve the optimal rate $\mathcal{O}(m^{-1/2})$, see [3].

The purpose of the present paper is to show that a very simple modification of the **PGA**, namely just rescaling $f_m$ at each iteration, already leads to the improved convergence rate $\mathcal{O}(m^{-1/2})$ for functions in $\mathcal{A}_1(\mathcal{D})$ and the rate $\mathcal{O}(m^{-\theta/2})$ for functions from the interpolation space $[H, \mathcal{A}_1(\mathcal{D})]_{\theta,\infty}$, $0 < \theta < 1$. The rescaling we suggest is simply the orthogonal projection of $f$ onto $f_m$. We call this modified algorithm a Rescaled Pure Greedy Algorithm (**RPGA**) and prove optimal convergence rates for its weak version in Hilbert and Banach spaces. In a subsequent paper, see [5], we show that this strategy can also be applied successfully for developing an algorithm for convex optimization.

The paper is organized as follows. In Section 2, we spell out our notation and recall some simple known facts related to greedy algorithms. In Section 3, we present the **RPGA** for a Hilbert space and prove the above convergence rates. The remaining parts of this paper consider a modification of this algorithm for Banach spaces and weak versions of this algorithm.

## 2. Notation and preliminaries

We denote by $H$ a Hilbert space and by $X$ a Banach space with $\|\cdot\|$ being the norm in these spaces, respectively. A set of functions $\mathcal{D} \subset H$(or $X$) is called a dictionary if $\|\varphi\| = 1$ for every $\varphi \in \mathcal{D}$ and the closure of span$(\mathcal{D})$ is $H$(or $X$). An example of a dictionary is any Shauder basis for $H$(or $X$). However, the main idea behind dictionaries is to cover redundant families such as frames. A common example of dictionaries is the union of several Shauder bases.

We denote by $\Sigma_m(\mathcal{D})$ the set, consisting of all $m$-sparse elements with respect to the dictionary $\mathcal{D}$, namely

$$\Sigma_m := \Sigma_m(\mathcal{D}) = \{g : \ g = \sum_{\varphi \in \Lambda} c_\varphi \varphi, \ \Lambda \in \mathcal{D}, \ |\Lambda| \le m\}.$$

Here, we use the notation $|\Lambda|$ to denote the cardinality of the index set $\Lambda$. For a general element $f$ from $H$(or $X$), we define the error of approximation