A GENERALIZATION OF GRAM–SCHMIDT ORTHOGONALIZATION GENERATING ALL PARSEVAL FRAMES

PETER G. CASAZZA AND GITTA KUTYNIOK

Abstract. Given an arbitrary finite sequence of vectors in a finite–dimensional Hilbert space, we describe an algorithm, which computes a Parseval frame for the subspace generated by the input vectors while preserving redundancy exactly. We further investigate several of its properties. Finally, we apply the algorithm to several numerical examples.

1. introduction

Let $\mathcal{H}$ be a finite–dimensional Hilbert space. A sequence $(f_i)_{i=1}^n \subset \mathcal{H}$ forms a *frame*, if there exist constants $0 < A \leq B < \infty$ such that

$$A\|g\|^2 \leq \sum_{i=1}^{n} |\langle f_i, g\rangle|^2 \leq B\|g\|^2 \quad \text{for all } g \in \mathcal{H}. \tag{1}$$

Frames have turned out to be an essential tool for many applications such as, for example, data transmission, due to their robustness not only against noise but also against losses and due to their freedom in design [4, 6]. Their main advantage lies in the fact that a frame can be designed to be redundant while still providing a reconstruction formula. Since the *frame operator* $Sg = \sum_{i=1}^n \langle g, f_i\rangle f_i$ is invertible, each vector $g \in \mathcal{H}$ can be always reconstructed from the values $(\langle g, f_i\rangle)_{i=1}^n$ via

$$g = SS^{-1}g = \sum_{i=1}^{n} \langle g, f_i\rangle S^{-1}f_i.$$

However, the inverse frame operator is usually very complicated to compute. This difficulty can be avoided by choosing a frame whose frame operator equals the identity. This is one reason why *Parseval frames*, i.e., frames for which $S = Id$ or equivalently for which A and B in (1) can be chosen as $A = B = 1$, enjoy rapidly increasing attention. Another reason is that quite recently it was shown by Benedetto and Fickus [1] that in $\mathbb{R}^d$ as well as in $\mathbb{C}^d$ finite equal norm Parseval frames, i.e., finite Parseval frames whose elements all have the same norm, are exactly those sequences which are in equilibrium under the so–called frame force, which parallels a Coulomb potential law in electrostatics. In fact, they demonstrate that in this setting both orthonormal sets and finite equal norm Parseval frames arise from the same optimization problem.

Date: January 11, 2005.

1991 *Mathematics Subject Classification.* Primary 42C15; Secondary 46C99, 94A12.

Key words and phrases. Finite–dimensional Hilbert space, Gram–Schmidt orthogonalization, linear dependence, Parseval frame, redundancy.

Thus, in general, Parseval frames are perceived as the most natural generalization of orthogonal bases.

For more details on frame theory we refer to the survey article [3] and the book [5].

Given a finite sequence in a finite–dimensional Hilbert space, our driving motivation is to determine a Parseval frame for the subspace generated by this sequence. We could apply Gram–Schmidt orthogonalization, which would yield an orthonormal basis for this subspace together with a maximal number of zero vectors. But redundancy is the crucial part for many applications. Hence we need to generalize Gram–Schmidt orthogonalization so that it can also produce Parseval frames which are not orthonormal bases preferably while preserving redundancy exactly. In particular, the algorithm shall be applicable to sequences of linearly dependent vectors.

Our algorithm is designed to be iterative in the sense that one vector is added each time to an already modified set of vectors and then the new set is adjusted again. In each iteration it not only computes a Parseval frame for the span of the sequence of vectors already dealt with at this point, but also preserves redundancy in an exact way. Moreover, it reduces to Gram–Schmidt orthogonalization if applied to a sequence of linearly independent vectors and each time a linearly dependent vector is added, the algorithm computes the Parseval frame which is closest in l^2–norm to the already modified sequence of vectors.

The paper is organized as follows. In Section 2 we first state the algorithm and show that it in fact generates a special Parseval frame in each iteration. Additional properties of the algorithm such as, for example, the preservation of redundancy, are treated in Section 3. Finally, in Section 4 we first compare the complexity of our algorithm with the complexity of the Gram–Schmidt orthogonalization and then study the different steps of the algorithm applied to several numerical examples.

2. The algorithm

Throughout this paper let $\mathcal{H}$ denote a finite–dimensional Hilbert space. We start by describing our iterative algorithm. On input $n \in \mathbb{N}$ and $f = (f_i)_{i=1}^n \subset \mathcal{H}$ the procedure *GGSP (Generalized Gram–Schmidt orthogonalization to compute Parseval frames)* outputs a Parseval frame $g = (g_i)_{i=1}^n \subset \mathcal{H}$ for $\text{span}\{(f_i)_{i=1}^n\}$ with special properties (see Theorem 2.2).

procedure $GGSP(n, f; g)$

for $k := 1$ **to** n **do**

begin

  if $f_k = 0$ **then**

    $g_k := 0;$

  else

  begin

    $g_k := f_k - \sum_{j=1}^{k-1} \langle f_k, g_j \rangle \, g_j;$

if $g_k \neq 0$ **then**
$g_k := \frac{1}{\|g_k\|} g_k;$
else
begin
for $i := 1$ **to** $k-1$ **do** $g_i := g_i + \frac{1}{\|f_k\|^2}\left(\frac{1}{\sqrt{1+\|f_k\|^2}} - 1\right)\langle g_i, f_k\rangle f_k;$
$g_k := \frac{1}{\sqrt{1+\|f_k\|^2}} f_k;$
end;
end;
end;
end.

In the remainder of this paper the following notation will be used.

Notation 2.1. Let Φ denote the mapping $(f_i)_{i=1}^n \mapsto (g_i)_{i=1}^n$ of a sequence of vectors in $\mathcal{H}$ to another sequence of vectors in $\mathcal{H}$ given by the procedure *GGSP*. We will also use the notation $((f_i)_{i=1}^n, g) := (f_1, \ldots, f_n, g)$ for $(f_i)_{i=1}^n \subset \mathcal{H}$ and $g \in \mathcal{H}$.

The following result shows that the algorithm not only produces a Parseval frame for $\text{span}\{(f_i)_{i=1}^n\}$, but even in each iteration also produces a special Parseval frame for $\text{span}\{(f_i)_{i=1}^k\}$, $k = 1, \ldots, n$.

It is well–known that applying $S^{-\frac{1}{2}}$ to a sequence of vectors $(f_i)_{i=1}^n$ in $\mathcal{H}$ yields a Parseval frame, where S denotes the frame operator for this sequence (see [3, Theorem 4.2]). Moreover, Theorem 3.1 will show that the Parseval frame $(S^{-\frac{1}{2}} f_i)_{i=1}^n$ is the closest in l^2–norm to the sequence $(f_i)_{i=1}^n$. However, in general the computation of the operator $S^{-\frac{1}{2}}$ is not very efficient. In fact, in our algorithm we do not compute $S^{-\frac{1}{2}}((f_i)_{i=1}^n)$. Instead in each iteration when adding a vector, which is linearly dependent to the already modified vectors, we apply $S^{-\frac{1}{2}}$ to those vectors and the added one, where here S denotes the frame operator for this new set of vectors. This eases the computation in a significant manner, since the set of computed vectors already forms a Parseval frame, and nevertheless we compute the closest Parseval frame in each iteration. When we add a linearly independent vector, we orthogonalize this one vector by using a Gram–Schmidt step. Thus this algorithm is also a generalization of Gram–Schmidt orthogonalization.

Theorem 2.2. *Let $n \in \mathbb{N}$ and $(f_i)_{i=1}^n \subset \mathcal{H}$. Then, for each $k \in \{1, \ldots, n\}$, the sequence of vectors $\Phi((f_i)_{i=1}^k)$ is a Parseval frame for* $\text{span}\{(f_i)_{i=1}^k\} = \text{span}\{\Phi((f_i)_{i=1}^k)\}$.

In particular, for each $k \in \{1, \ldots, n\}$, the following conditions hold.

(i) *If $f_k \in \text{span}\{(f_i)_{i=1}^{k-1}\}$, then*

$$\Phi((f_i)_{i=1}^k) = (S^{-\frac{1}{2}}(\Phi((f_i)_{i=1}^{k-1}), f_k)),$$

where S is the frame operator for $(\Phi((f_i)_{i=1}^{k-1}), f_k)$.

(ii) *If* $f_k \notin \operatorname{span}\{(f_i)_{i=1}^{k-1}\}$, *then*

$$\Phi((f_i)_{i=1}^{k}) = (\Phi((f_i)_{i=1}^{k-1}), g_k), \quad g_k \in \mathcal{H}, \ \|g_k\| = 1$$

and

$$g_k \perp \Phi((f_i)_{i=1}^{k-1}).$$

Proof. We will prove the first claim by induction and meanwhile in each step we show that, in particular, the claims in (i) and (ii) hold. For this, let l denote the smallest number in $\{1, \ldots, n\}$ with $f_l \neq 0$. Obviously, for each $k \in \{1, \ldots, l-1\}$, the generated set of vectors g_k (see line 3 of *GGSP*) forms a Parseval frame for $\operatorname{span}\{(f_i)_{i=1}^{k}\} = \{0\}$ and also (i) is fulfilled. The hypothesis in (ii) does not apply here. Next notice that in the case $k = l$ we have $g_k := \frac{1}{\|f_k\|} f_k$ (line 8), which certainly is a Parseval frame for $\operatorname{span}\{(f_i)_{i=1}^{k}\} = \operatorname{span}\{f_k\}$. It is also easy to see that (i) and (ii) are satisfied.

Now fix some $k \in \{l+1, \ldots, n\}$ and assume that the sequence $(\tilde{g}_i)_{i=1}^{k-1} := \Phi((f_i)_{i=1}^{k-1})$ is a Parseval frame for $\operatorname{span}\{(f_i)_{i=1}^{k-1}\} = \operatorname{span}\{(\tilde{g}_i)_{i=1}^{k-1}\}$. We have to study two cases.

Case 1: The vector $g_k := f_k - \sum_{j=1}^{k-1} \langle f_k, \tilde{g}_j \rangle \tilde{g}_j$ computed in line 6 is trivial. This implies that

$$\operatorname{span}\{(f_i)_{i=1}^{k-1}\} = \operatorname{span}\{(\tilde{g}_i)_{i=1}^{k-1}\} = \operatorname{span}\{(f_i)_{i=1}^{k}\}, \tag{2}$$

since otherwise the Gram–Schmidt orthogonalization step would yield a non–trivial vector. In particular, only the hypothesis in (i) applies.

Now let P denote the orthogonal projection of $\mathcal{H}$ onto $\operatorname{span}\{f_k\}$. In order to compute $S^{-\frac{1}{2}}$, where S denotes the frame operator for $((\tilde{g}_i)_{i=1}^{k-1}, f_k)$, we first show that each $(I-P)\tilde{g}_i$, $i = 1, \ldots, k-1$ is an eigenvector for S with respect to the eigenvalue 1 or the zero vector. This claim follows immediately from

$$\begin{aligned} S(I-P)\tilde{g}_i &= \sum_{j=1}^{k-1} \langle (I-P)\tilde{g}_i, \tilde{g}_j \rangle \tilde{g}_j + \langle (I-P)\tilde{g}_i, f_k \rangle f_k \\ &= \sum_{j=1}^{k-1} \langle (I-P)\tilde{g}_i, \tilde{g}_j \rangle \tilde{g}_j \\ &= (I-P)\tilde{g}_i, \end{aligned}$$

since $(\tilde{g}_i)_{i=1}^{k-1}$ is a Parseval frame for $\operatorname{span}\{(\tilde{g}_i)_{i=1}^{k-1}\}$. Also f_k is an eigenvector for S, but with respect to the eigenvalue $1 + \|f_k\|^2$, which is proven by the following calculation:

$$Sf_k = \sum_{j=1}^{k-1} \langle f_k, \tilde{g}_j \rangle \tilde{g}_j + \langle f_k, f_k \rangle f_k = (1 + \|f_k\|^2) f_k.$$

Using f_k as an eigenbasis for $P(\operatorname{span}\{(\tilde{g}_i)_{i=1}^{k-1}\})$ and an arbitrary eigenbasis for $(I - P)(\operatorname{span}\{(\tilde{g}_i)_{i=1}^{k-1}\})$, we can diagonalize S to compute $S^{-\frac{1}{2}}$. This together with the fact that $(I-P)\tilde{g}_i$, $i = 1, \ldots, k-1$ is an eigenvector for S with respect to the eigenvalue 1 and that $S(I-P)f_k = 0$ yields

$$S^{-\frac{1}{2}} \tilde{g}_i = \frac{1}{\sqrt{1 + \|f_k\|^2}} P\tilde{g}_i + (I-P)\tilde{g}_i \quad \text{for} \quad 1 \le i \le k-1$$

and

$$S^{-\frac{1}{2}} f_k = \frac{1}{\sqrt{1+\|f_k\|^2}} f_k.$$

Comparing these equalities with line 11 and 12 of *GGSP* shows that in fact $\Phi((f_i)_{i=1}^k) = (S^{-\frac{1}{2}}((\tilde{g}_i)_{i=1}^{k-1}), f_k))$, which is (i). By [3, Theorem 4.2] and (2), this immediately implies that the sequence $\Phi((f_i)_{i=1}^k)$ is a Parseval frame for $\text{span}\{\Phi((f_i)_{i=1}^k)\} = \text{span}\{(f_i)_{i=1}^k\}$.

Case 2: The condition in line 7 applies, i.e., we have $g_k := (f_k - \sum_{j=1}^{k-1} \langle f_k, \tilde{g}_j\rangle \tilde{g}_j)/(\|f_k - \sum_{j=1}^{k-1} \langle f_k, \tilde{g}_j\rangle \tilde{g}_j\|) \neq 0$. Then we set $g_i := \tilde{g}_i$ for all $i = 1, \ldots, k-1$. Obviously, $\|g_k\| = 1$. Moreover, since by induction hypothesis $(\tilde{g}_i)_{i=1}^{k-1}$ forms a Parseval frame, for each $i = 1, \ldots, k-1$, we have

$$\left\langle g_i, f_k - \sum_{j=1}^{k-1} \langle f_k, \tilde{g}_j\rangle \tilde{g}_j \right\rangle = \langle g_i, f_k\rangle - \langle g_i, f_k\rangle = 0.$$

Thus g_k is normalized vector, which is orthogonal to $g_1, \ldots, g_{k-1}$. Hence (ii) is satisfied and, for all $h \in \text{span}\{(g_i)_{i=1}^k\}$, we obtain

$$\sum_{i=1}^{k} |\langle h, g_i\rangle|^2 = \sum_{i=1}^{k-1} |\langle (I-P)h, g_i\rangle|^2 + |\langle Ph, g_k\rangle|^2 = \|(I-P)h\|^2 + \|Ph\|^2 = \|h\|^2,$$

where P denotes the orthogonal projection of $\mathcal{H}$ onto $\text{span}\{g_k\}$. This proves that $(g_i)_{i=1}^k = \Phi((f_i)_{i=1}^k)$ is a Parseval frame for $\text{span}\{\Phi((f_i)_{i=1}^k)\}$. Moreover, we have $\text{span}\{\Phi((f_i)_{i=1}^k)\} = \text{span}\{(f_i)_{i=1}^{k-1}, f_k - \sum_{j=1}^{k-1} \langle f_k, g_j\rangle g_j\} = \text{span}\{(f_i)_{i=1}^k\}$. This finishes the proof, since the hypothesis in (i) does not apply in this case. □

The algorithm can be seen as a "Gram–Schmidt procedure backwards" in the sense that in each iteration, if the added vector is linearly dependent to the already computed vectors, not only this vector is modified, but also all the other vectors are rearranged with respect to the new vector so that the collection forms a Parseval frame. This way of computation will be demonstrated by several examples in Subsection 4.2.

3. Special Properties of the algorithm

In this section we first determine in general which Parseval frame is the closest to the initial sequence and study which properties of our algorithm this result implies. Then we investigate several additional properties of the procedure *GGSP*, in particular we characterize those sequences, which lead to orthonormal bases, and we show that Φ regarded as a map from finite sequences to Parseval frames is "almost" bijective. At last, we examine the redundancy of the generated Parseval frame.

Given a sequence $(f_i)_{i=1}^n$ with frame operator S, by [3, Theorem 4.2], the sequence $(S^{-\frac{1}{2}} f_i)_{i=1}^n$ always forms a Parseval frame. The following result shows that this sequence can in fact be characterized as the very same Parseval frame, which is the closest to $(f_i)_{i=1}^n$ in l^2–norm.

Theorem 3.1. *If $(f_i)_{i=1}^n \subset \mathcal{H}$, $n \in \mathbb{N}$ is any frame for $\mathcal{H}$ with frame operator S, then*

$$\sum_{i=1}^n \|f_i - S^{-\frac{1}{2}} f_i\|^2 = \inf\{\sum_{i=1}^n \|f_i - g_i\|^2 : (g_i)_{i=1}^n \text{ is a Parseval frame for } \mathcal{H}\}.$$

Moreover, $(S^{-\frac{1}{2}} f_i)_{i=1}^n$ is the unique minimizer.

Proof. Let $(e_j)_{j=1}^d$, $d := \dim \mathcal{H}$, be an orthonormal eigenvector basis for $\mathcal{H}$ with respect to S and respective eigenvalues $(\lambda_j)_{j=1}^d$. Then we can rewrite the left–hand side of the claimed inequality in the following way:

$$\begin{aligned}
\sum_{i=1}^n \|f_i - S^{-\frac{1}{2}} f_i\|^2 &= \sum_{i=1}^n \| \sum_{j=1}^d \langle f_i, e_j \rangle e_j - \tfrac{1}{\sqrt{\lambda_j}} \langle f_i, e_j \rangle e_j \|^2 \\
&= \sum_{i=1}^n \sum_{j=1}^d |\langle f_i, e_j \rangle|^2 \, |1 - \tfrac{1}{\sqrt{\lambda_j}}|^2 \\
&= \sum_{j=1}^d |1 - \tfrac{1}{\sqrt{\lambda_j}}|^2 \sum_{i=1}^n |\langle f_i, e_j \rangle|^2 \\
&= \sum_{j=1}^d |1 - \tfrac{1}{\sqrt{\lambda_j}}|^2 \lambda_j \\
&= \sum_{j=1}^d (\lambda_j - 2\sqrt{\lambda_j} + 1).
\end{aligned}$$

Now let $(g_i)_{i=1}^n$ be an arbitrary Parseval frame for $\mathcal{H}$. Using again the eigenbasis and its eigenvalues, we obtain

$$\begin{aligned}
\sum_{i=1}^n \|f_i - g_i\|^2 &= \sum_{i=1}^n \| \sum_{j=1}^d \langle f_i, e_j \rangle e_j - \langle g_i, e_j \rangle e_j \|^2 \\
&= \sum_{i=1}^n \sum_{j=1}^d | \langle f_i, e_j \rangle - \langle g_i, e_j \rangle |^2 \\
&= \sum_{j=1}^d \sum_{i=1}^n \left(|\langle f_i, e_j \rangle|^2 + |\langle g_i, e_j \rangle|^2 - 2\mathrm{Re}\left[\langle f_i, e_j \rangle \overline{\langle g_i, e_j \rangle} \right] \right) \\
&= \sum_{j=1}^d \left(\sum_{i=1}^n |\langle f_i, e_j \rangle|^2 + \sum_{i=1}^n |\langle g_i, e_j \rangle|^2 - 2\mathrm{Re}\left[\sum_{i=1}^n \langle f_i, e_j \rangle \overline{\langle g_i, e_j \rangle} \right] \right) \\
&= \sum_{j=1}^d \left(\lambda_j + 1 - 2\mathrm{Re}\left[\sum_{i=1}^n \langle f_i, e_j \rangle \overline{\langle g_i, e_j \rangle} \right] \right).
\end{aligned}$$

Moreover, we have

$$\begin{aligned}\sum_{j=1}^{d} \mathrm{Re}\left[\sum_{i=1}^{n} \langle f_i, e_j\rangle \overline{\langle g_i, e_j\rangle}\right] &\leq \sum_{j=1}^{d}\sum_{i=1}^{n} |\langle f_i, e_j\rangle|\,|\langle g_i, e_j\rangle| \\ &\leq \sum_{j=1}^{d} \sqrt{\sum_{i=1}^{n} |\langle f_i, e_j\rangle|^2}\sqrt{\sum_{i=1}^{n} |\langle g_i, e_j\rangle|^2} \\ &= \sum_{j=1}^{d} \sqrt{\lambda_j}.\end{aligned}$$

Combining this estimate with the computations above yields

$$\sum_{i=1}^{n} \|f_i - g_i\|^2 \geq \sum_{j=1}^{d} (\lambda_j - 2\sqrt{\lambda_j} + 1) = \sum_{i=1}^{n} \|f_i - S^{-\frac{1}{2}} f_i\|^2.$$

Since $(S^{-\frac{1}{2}} f_i)_{i=1}^n$ is a Parseval frame for $\mathcal{H}$, the first claim follows.

For the moreover part, suppose that $(g_i)_{i=1}^n$ is another minimizer. Then, by the above calculation, for each $k \in \{1, \dots, n\}$, we have

$$\mathrm{Re}\,\langle f_k, e_j\rangle \overline{\langle g_k, e_j\rangle} = |\langle f_k, e_j\rangle|\,|\langle g_k, e_j\rangle| \tag{3}$$

and, for each $j \in \{1, \dots, d\}$,

$$\sum_{k=1}^{n} |\langle f_k, e_j\rangle|\,|\langle g_k, e_j\rangle| = \sqrt{\sum_{k=1}^{n} |\langle f_k, e_j\rangle|^2}\sqrt{\sum_{k=1}^{n} |\langle g_k, e_j\rangle|^2}. \tag{4}$$

Now let $r_{k,j}, s_{k,j} > 0$ and $\theta_{k,j}, \psi_{k,j} \in [0, 2\pi)$ be such that $\langle f_k, e_j\rangle = r_{k,j} e^{i\theta_{k,j}}$ and $\langle g_k, e_j\rangle = s_{k,j} e^{i\psi_{k,j}}$. We compute

$$\mathrm{Re}\left[\langle f_k, e_j\rangle \overline{\langle g_k, e_j\rangle}\right] = r_{k,j} s_{k,j} \mathrm{Re}\left[e^{i(\theta_{k,j} - \psi_{k,j})}\right] = r_{k,j} s_{k,j} \cos(\theta_{k,j} - \psi_{k,j}).$$

Hence (3) implies that

$$r_{k,j} s_{k,j} \cos(\theta_{k,j} - \psi_{k,j}) = r_{k,j} s_{k,j},$$

which in turn yields $\theta_{k,j} = \psi_{k,j}$. Thus $\langle g_k, e_j\rangle = t_{k,j} \langle f_k, e_j\rangle$ for some $t_{k,j} > 0$ for all $k \in \{1, \dots, n\}$, $j \in \{1, \dots, d\}$. By (4), for each $j \in \{1, \dots, d\}$ there exists some $u_j > 0$ such that

$$u_j |\langle f_k, e_j\rangle| = |\langle g_k, e_j\rangle| = t_{k,j} |\langle f_k, e_j\rangle|.$$

This implies $t_{k,j} = u_j$ for all $k \in \{1, \dots, n\}$. Hence, for each $k \in \{1, \dots, n\}$ and $j \in \{1, \dots, d\}$, we obtain the relation

$$\langle g_k, e_j\rangle = u_j \langle f_k, e_j\rangle. \tag{5}$$

Since $(g_i)_{i=1}^n$ is a Parseval frame for $\mathcal{H}$, we have

$$1 = \sum_{k=1}^{n} |\langle g_k, e_j\rangle|^2 = u_j^2 \sum_{k=1}^{n} |\langle f_k, e_j\rangle|^2 = u_j^2 \lambda_j.$$

This shows that $u_j = \frac{1}{\sqrt{\lambda_j}}$. Thus, using (5) and the definition of $(e_j)_{j=1}^d$ and $(\lambda_j)_{j=1}^d$, it follows that $g_k = S^{-\frac{1}{2}} f_k$ for all $k \in \{1, \ldots, n\}$. □

This result together with Theorem 2.2 (i) implies the following property of our algorithm.

Corollary 3.2. *In each iteration of GGSP, in which a linearly dependent vector is added, the algorithm computes the unique Parseval frame, which is closest to the frame consisting of the already computed vectors and the added one.*

Next we characterize those sequences of vectors applied to which the algorithm computes an orthonormal basis. The proof will show that this is exactly the case, when only the steps of the Gram–Schmidt orthogonalization are carried out.

Proposition 3.3. *Let $(f_i)_{i=1}^n \subset \mathcal{H}$, $n \in \mathbb{N}$. The following conditions are equivalent.*

(i) *The sequence $\Phi((f_i)_{i=1}^n)$ is an orthonormal basis for* $\text{span}\{(f_i)_{i=1}^n\}$.
(ii) *The sequence $(f_i)_{i=1}^n$ is linearly independent.*

Proof. If (ii) holds, only line 6–8 of *GGSP* will be performed and these steps coincide with Gram–Schmidt orthogonalization, hence produce an orthonormal system.

Now suppose that (ii) does not hold. This is equivalent to $\dim(\text{span}\{(f_i)_{i=1}^n\}) < n$. By Theorem 2.2, we have $\text{span}\{(f_i)_{i=1}^n\} = \text{span}\{\Phi((f_i)_{i=1}^n)\}$. This in turn implies $\dim(\text{span}\{\Phi((f_i)_{i=1}^n)\}) < n$. Thus $\Phi((f_i)_{i=1}^n)$ cannot form an orthonormal basis for $\text{span}\{(f_i)_{i=1}^n\}$. □

The mapping Φ given by the procedure *GGSP* of a finite sequence in $\mathcal{H}$ to a Parseval frame for a subspace of $\mathcal{H}$ is "almost" bijective in the following sense.

Proposition 3.4. *Let Φ be the mapping defined in the previous paragraph. Then Φ satisfies the following conditions.*

(i) *Φ is surjective.*
(ii) *For each Parseval frame $(g_i)_{i=1}^n \subset \mathcal{H}$, the set $\Phi^{-1}((g_i)_{i=1}^n)$ equals*

$$\left\{ (f_i)_{i=1}^n : f_i = \left\{ \begin{array}{cl} \tilde{f}_i, & \text{if } \text{span}\{(\tilde{f}_j)_{j=1}^{i-1}\} = \text{span}\{(\tilde{f}_j)_{j=1}^{i}\}, \\ \lambda \tilde{f}_i + \varphi,\ \lambda \in \mathbb{R}^+, & \\ \varphi \in \text{span}\{(\tilde{f}_j)_{j=1}^{i-1}\}, & \text{otherwise} \end{array} \right. \right\} \tag{6}$$

for some $(\tilde{f}_i)_{i=1}^n \in \Phi^{-1}((g_i)_{i=1}^n)$.

Proof. It is easy to see that each step of the procedure *GGSP* is reversable which implies (i).

To prove (ii) we first show that the set (6) is contained in $\Phi^{-1}((g_i)_{i=1}^n)$. For this, let $(f_i)_{i=1}^n$ be an element of the set (6). Notice that, by definition of $(f_i)_{i=1}^n$, we have $\text{span}\{(f_i)_{i=1}^k\} = \text{span}\{(\tilde{f}_i)_{i=1}^k\}$ for all $k \in \{1, \ldots, n\}$. Since $(\tilde{f}_i)_{i=1}^n \in \Phi^{-1}((g_i)_{i=1}^n)$, we only have to study the case $\text{span}\{(f_i)_{i=1}^{k-1}\} \neq \text{span}\{(f_i)_{i=1}^k\}$ for some $k \in \{1, \ldots, n\}$. But then line 8 of *GGSP* will be performed. Let $\Phi((f_i)_{i=1}^{k-1})$ be denoted by $(\tilde{g}_i)_{i=1}^{k-1}$. By

Theorem 2.2, the sequence $(\tilde{g}_i)_{i=1}^{k-1}$ forms a Parseval frame for $\text{span}\{(f_i)_{i=1}^{k-1}\}$. Hence

$$\begin{aligned}\frac{\lambda\tilde{f}_k+\varphi-\sum_{j=1}^{k-1}\langle\lambda\tilde{f}_k+\varphi,\tilde{g}_j\rangle\tilde{g}_j}{\|\lambda\tilde{f}_k+\varphi-\sum_{j=1}^{k-1}\langle\lambda\tilde{f}_k+\varphi,\tilde{g}_j\rangle\tilde{g}_j\|} &= \frac{\lambda\tilde{f}_k-\sum_{j=1}^{k-1}\langle\lambda\tilde{f}_k,\tilde{g}_j\rangle\tilde{g}_j}{\|\lambda\tilde{f}_k-\sum_{j=1}^{k-1}\langle\lambda\tilde{f}_k,\tilde{g}_j\rangle\tilde{g}_j\|}\\ &= \frac{\tilde{f}_k-\sum_{j=1}^{k-1}\langle\tilde{f}_k,\tilde{g}_j\rangle\tilde{g}_j}{\|\tilde{f}_k-\sum_{j=1}^{k-1}\langle\tilde{f}_k,\tilde{g}_j\rangle\tilde{g}_j\|},\end{aligned}$$

which proves the first claim.

Secondly, suppose $(f_i)_{i=1}^n \subset \mathcal{H}$ is not an element of (6). We claim that $\Phi((f_i)_{i=1}^n) \neq \Phi((\tilde{f}_i)_{i=1}^n)$, which finishes the proof. Let $k \in \{1,\dots,n\}$ be the largest number such that f_k does not satisfy the conditions in (6). We have to study two cases.

Case 1: Suppose that $f_k \neq \tilde{f}_k$, but $\text{span}\{(f_i)_{i=1}^{k-1}\} = \text{span}\{(f_i)_{i=1}^{k}\}$. Then in the kth iteration line 12 will be performed and we obtain

$$h_k := \frac{1}{\sqrt{1+\|f_k\|^2}} f_k \neq \frac{1}{\sqrt{1+\|\tilde{f}_k\|^2}}\tilde{f}_k =: \tilde{h}_k,$$

since $f_k \neq \tilde{f}_k$. Thus $\Phi((f_i)_{i=1}^k) \neq \Phi((\tilde{f}_i)_{i=1}^k)$. If in the following iterations the condition in line 7 always applies, we are done, since h_k and $\tilde{h}_k$ are not changed anymore. Now suppose that there exists $l \in \{1,\dots,n\}$, $l > k$ with $\text{span}\{(f_i)_{i=1}^{l-1}\} = \text{span}\{(f_i)_{i=1}^{l}\}$. Then in the lth iteration h_k and $\tilde{h}_k$ are modified in line 11. Since $f_l = \tilde{f}_l$ by choice of k, using a reformulation of line 11, we still have

$$h_k := \frac{1}{\sqrt{1+\|f_l\|^2}} P h_k + (I-P)h_k \neq \frac{1}{\sqrt{1+\|\tilde{f}_l\|^2}} P\tilde{h}_k + (I-P)\tilde{h}_k =: \tilde{h}_k,$$

where P denotes the orthogonal projection onto $\text{span}\{f_l\}$.

Case 2: Suppose that $f_k \neq \lambda\tilde{f}_k + \varphi$ for each $\lambda \in \mathbb{R}^+$ and $\varphi \in \text{span}\{(f_i)_{i=1}^{k-1}\}$, and also $\text{span}\{(f_i)_{i=1}^{k-1}\} \neq \text{span}\{(f_i)_{i=1}^{k}\}$. Let $(h_i)_{i=1}^{k-1}$ and $(\tilde{h}_i)_{i=1}^{k-1}$ denote $\Phi((f_i)_{i=1}^{k-1})$ and $\Phi((\tilde{f}_i)_{i=1}^{k-1})$, respectively. If $(h_i)_{i=1}^{k-1} = (\tilde{h}_i)_{i=1}^{k-1}$, the computation in line 8 in the kth iteration yields

$$h_k := \frac{f_k-\sum_{j=1}^{k-1}\langle f_k,h_j\rangle h_j}{\|f_k-\sum_{j=1}^{k-1}\langle f_k,h_j\rangle h_j\|} \neq \frac{\tilde{f}_k-\sum_{j=1}^{k-1}\langle\tilde{f}_k,\tilde{h}_j\rangle\tilde{h}_j}{\|\tilde{f}_k-\sum_{j=1}^{k-1}\langle\tilde{f}_k,\tilde{h}_j\rangle\tilde{h}_j\|} =: \tilde{h}_k.$$

If $(h_i)_{i=1}^{k-1} \neq (\tilde{h}_i)_{i=1}^{k-1}$, then there exists some $l \in \{1,\dots,k-1\}$ with $h_l \neq \tilde{h}_l$. In both situations these inequalities remain valid as it was shown in the preceding paragraph. □

An important aspect of our algorithm is the redundancy of the computed frame. Hence it is desirable to know in which way redundancy is preserved throughout the algorithm. For this, we introduce a suitable definition of redundancy for sequences in a finite–dimensional Hilbert space.

Definition 3.5. Let $(f_i)_{i=1}^n \subset \mathcal{H}$, $n \in \mathbb{N}$. Then the *redundancy* $\mathrm{red}((f_i)_{i=1}^n)$ of this set is defined by

$$\mathrm{red}((f_i)_{i=1}^n) = \frac{n}{\dim(\mathrm{span}\{(f_i)_{i=1}^n\})},$$

where we set $\frac{n}{0} = \infty$.

Indeed in each iteration our algorithm preserves redundancy in an exact way.

Proposition 3.6. *Let $(f_i)_{i=1}^n \subset \mathcal{H}$, $n \in \mathbb{N}$. Then*

$$\mathrm{red}(\Phi((f_i)_{i=1}^n)) = \mathrm{red}((f_i)_{i=1}^n).$$

Proof. By Theorem 2.2, we have $\mathrm{span}\{(f_i)_{i=1}^n\} = \mathrm{span}\{\Phi((f_i)_{i=1}^n)\}$. From this, the claim follows immediately. □

4. Implementation of the algorithm

We will first compare the numerical complexities of the Gram–Schmidt orthogonalization and of *GGSP*. In a second part the procedure *GGSP* will be applied to several numerical examples in order to visualize the modifications of the vectors while performing the algorithm.

4.1. **Numerical complexity.** In this subsection it will turn out that the numerical complexity of our algorithm coincides with the numerical complexity of the Gram–Schmidt orthogonalization. In particular, we will analyze the complexity of the Gram–Schmidt step in lines 6 and 8 of *GGSP* with the complexity of the step in lines 6, 11, and 12 of *GGSP*, which is performed if the added vector is linearly dependent to the already modified vectors.

For this, suppose the computation of a square root of a real number with some given precision requires S elementary operations (additions, subtractions, multiplications, and divisions). Let d denote the dimension of $\mathcal{H}$, and let $k \in \{1, \ldots, n\}$. An easy computation shows that the number of elementary operations in lines 6 and 8 of *GGSP* equals $3dk+S$, and the number of elementary operations in lines 6, 11, and 12 of *GGSP* equals $10dk+(S+4)k-7d-3$. Hence in both cases the numerical complexity is $O(dk)$. Only the constants are slightly larger in the new step, which is performed in case of linear dependency. Thus both the Gram–Schmidt orthogonalization and *GGSP* possess the same numerical complexity of $O(dn^2)$.

4.2. **Graphical examples.** In order to give further insight into the algorithm, in this subsection we will study the different steps of *GGSP* for three examples. The single steps of each example are illustrated by a diagram. In each of these the first image in the uppermost row shows the positions of the vectors of the input sequence. Then in the following images the remaining original vectors and the modified vectors are displayed after each step of the loop in line 0 of *GGSP*. The original vectors are always marked by a circle and the already computed new vectors are indicated by a filled circle. The vector, which will be dealt with in the next step, is marked by a square. Recall that, by Theorem 2.2, in each step the set of vectors marked with a filled circle forms a Parseval frame for their linear span.

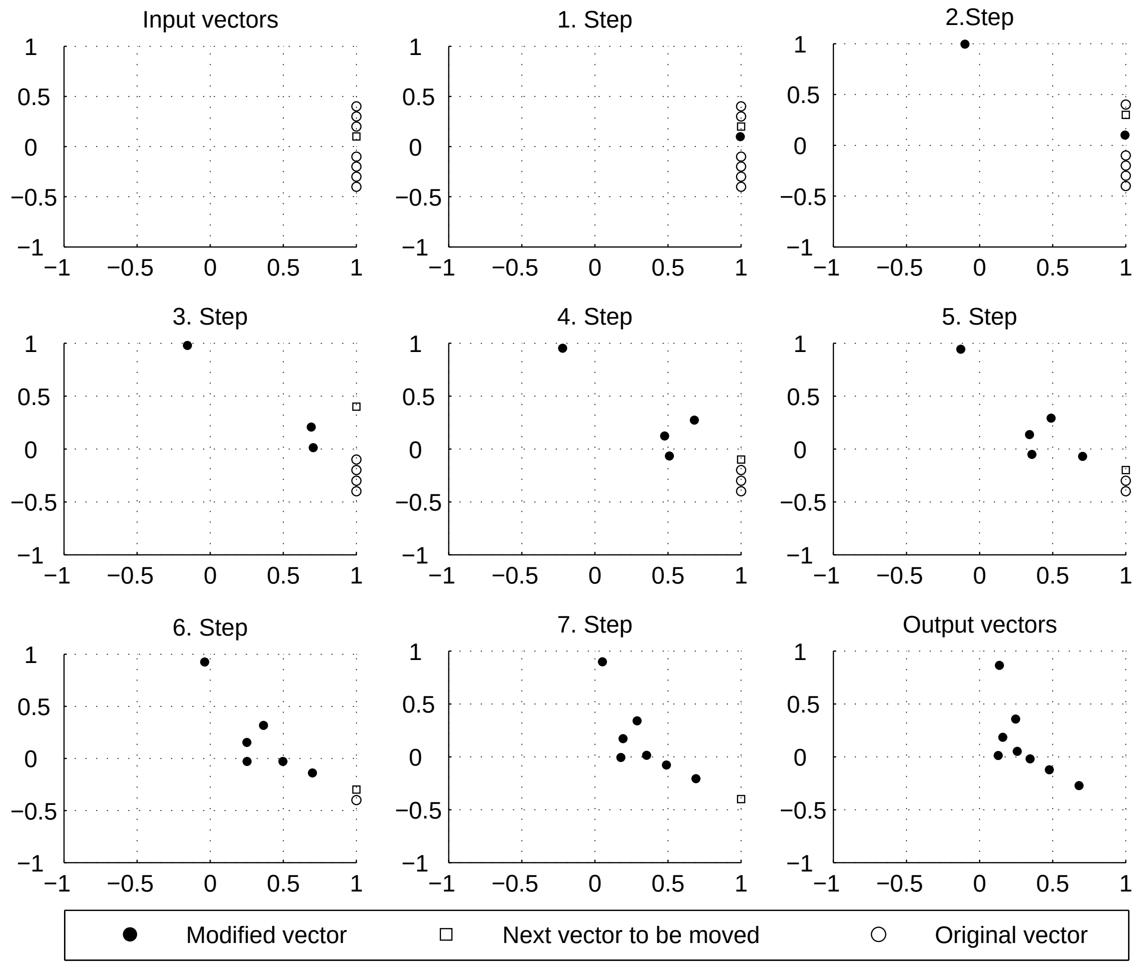

FIGURE 1. *GGSP* applied to the sequence of vectors $((1, 0.1), (1, 0.2), (1, 0.3), (1, 0.4), (1, -0.1), (1, -0.2), (1, -0.3), (1, -0.4))$

In the first example we consider the sequence of vectors $((1, 0.1), (1, 0.2), (1, 0.3), (1, 0.4), (1, -0.1), (1, -0.2), (1, -0.3), (1, -0.4))$. Figure 1 shows the modifications of the vectors while performing the *GGSP*. The Gram–Schmidt orthogonalization, which is performed in line 6–8 of *GGSP*, applies twice. In all the following steps the added vector is linearly dependent to the already modified vectors. Therefore we have to go through line 11 and 12, and the vectors already dealt with are newly rearranged in each step.

Figure 2 shows the same example with a different ordering of the vectors. It is no surprise that the generated Parseval frame is completely different from the one obtained in Figure 1, since already the Gram–Schmidt orthogonalization is sensitive to the ordering of the vectors.

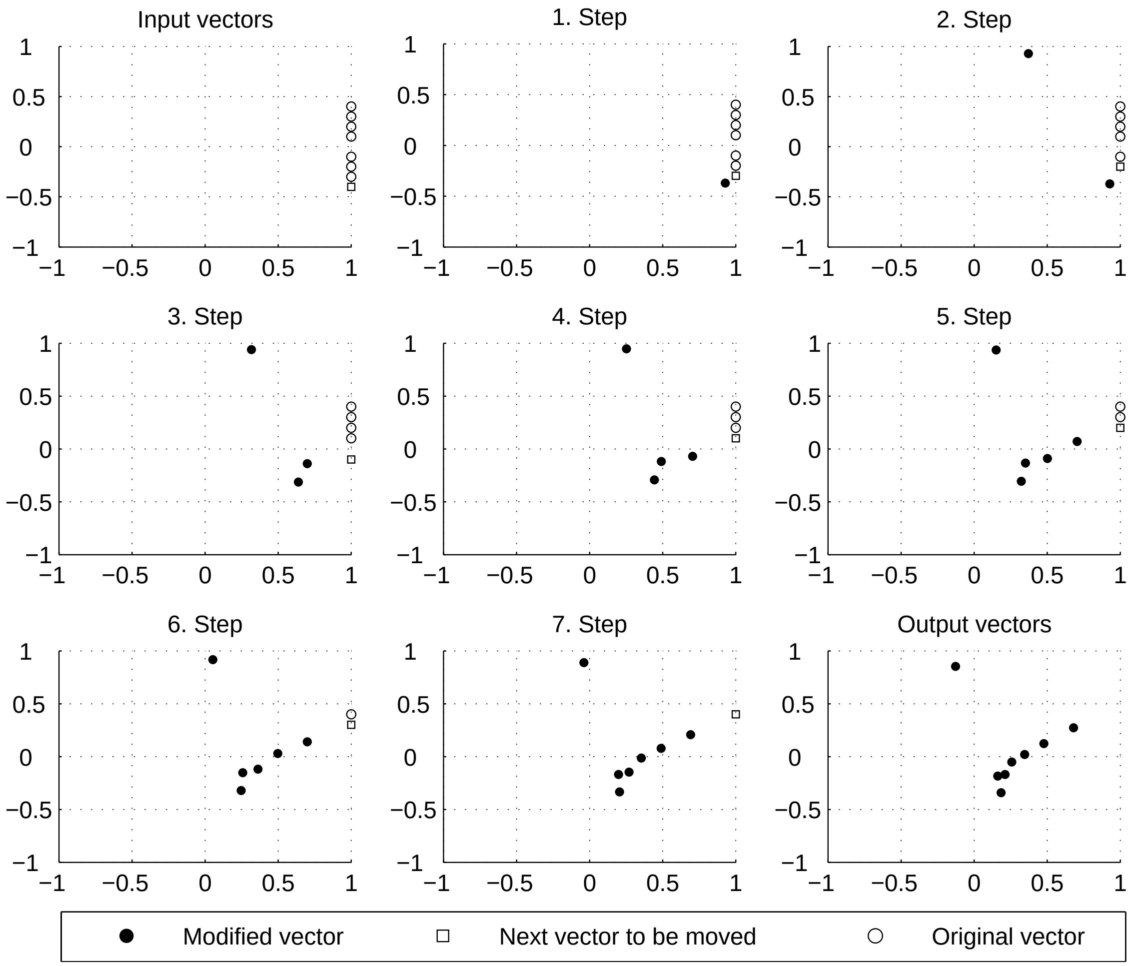

FIGURE 2. *GGSP* applied to the sequence of vectors $((1,-0.4)$, $(1,-0.3)$, $(1,-0.2)$, $(1,-0.1)$, $(1,0.1)$, $(1,0.2)$, $(1,0.3),(1,0.4))$

Both generated Parseval frames have in common that the first components of the vectors are almost all positive. Intuitively this is not astonishing, since already all vectors of the input sequence possess a positive first component.

The following example gives further evidence for the claim that the generated Parseval frame inherits the geometry of the input sequence in a particular way. Here the vectors of the input sequence are located on the unit circle, in particular we consider the sequence of vectors $((1,0)$, $(\sqrt{0.5},\sqrt{0.5})$, $(0,1)$, $(-\sqrt{0.5},\sqrt{0.5})$, $(-1,0)$, $(-\sqrt{0.5},-\sqrt{0.5})$, $(0,-1)$, $(\sqrt{0.5},-\sqrt{0.5}))$. While performing the *GGSP* the vectors almost keep the geometry of a circle and the final Parseval frame is located on a slightly deformed circle (see Figure 3). Notice that in the second step of the algorithm the second vector is moved to the position of the third vector (0,1). Hence in all the following computations these two vectors remain indistinguishable.

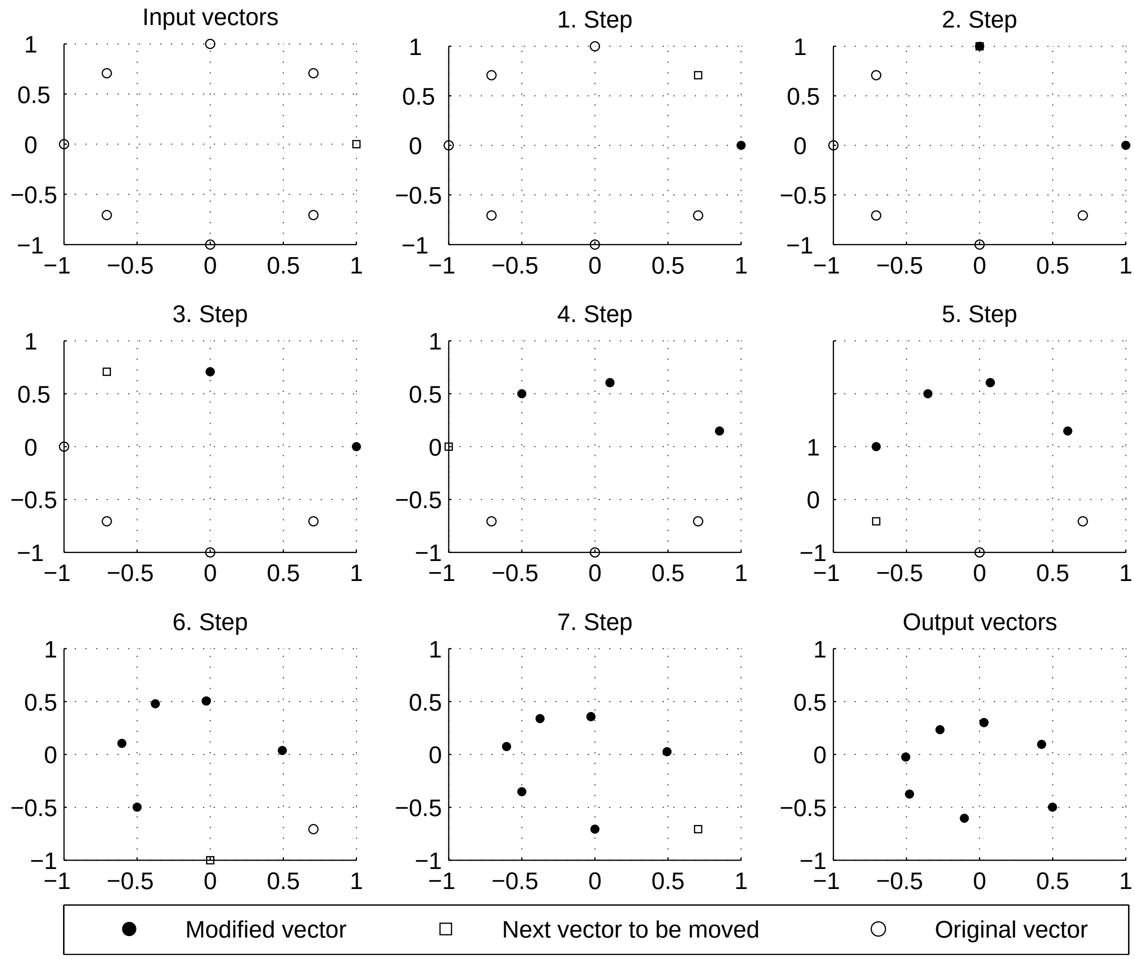

FIGURE 3. *GGSP* applied to the sequence of vectors $((1,0), (\sqrt{0.5}, \sqrt{0.5}), (0,1), (-\sqrt{0.5}, \sqrt{0.5}), (-1,0), (-\sqrt{0.5}, -\sqrt{0.5}), (0,-1), (\sqrt{0.5}, -\sqrt{0.5}))$

The graphical examples seem to indicate that to a certain extent output sequences inherit their geometry from the input sequence. For applications it would be especially important to characterize those input sequences, which generate equal norm Parseval frames or more generally "almost" equal norm Parseval frames (compare [2, Problem 4.4]).

Acknowledgments

The majority of the research for this paper was performed while the first author was visiting the Department of Mathematics at the University of Paderborn. This author thanks this department for its hospitality and support during this visit. The first author also acknowledges support from NSF DMS 0405376.

The second author acknowledges support from DFG grant KU 1446 and Forschungspreis der Universität Paderborn 2003.

The authors are indebted to the referee for valuable comments and suggestions which improved the paper.

References

[1] J.J. Benedetto and M. Fickus, Finite normalized tight frames, Adv. Comput. Math. 18 (2003), 357–385.
[2] P.G. Casazza, Custom building finite frames, In Wavelets, frames and operator theory, 61–86, Contemp. Math., 345, Amer. Math. Soc., Providence, RI, 2004.
[3] P.G. Casazza, The art of frame theory, Taiwanese J. of Math. 4 (2000), 129–201.
[4] P.G. Casazza and J. Kovaĉević, Equal-norm tight frames with erasures, Adv. Comput. Math. 18 (2003), 387–430.
[5] O. Christensen, *An Introduction to Frames and Riesz Bases*, Birkhäuser, Boston, 2003.
[6] V.K. Goyal, J. Kovačević, and J.A. Kelner, Quantized frame expansions with erasures, Appl. Comput. Harmon. Anal. 10 (2001), 203–233.

Department of Mathematics, University of Missouri, Columbia, Missouri 65211, USA
E-mail address: pete@math.missouri.edu

Mathematical Institute, Justus–Liebig–University Giessen, 35392 Giessen, Germany
E-mail address: gitta.kutyniok@math.uni-giessen.de