

Department of Mathematics

Sparsity Optimization in Design of Multidimensional Filter Networks

Mats Andersson, Oleg Burdakov, Hans Knutsson, Spartak Zikrin

LiTH-MAT-R--2013/16--SE



Linköpings universitet
TEKNISKA HÖGSKOLAN

Department of Mathematics
Linköping University
S-581 83 Linköping, Sweden.

Sparsity Optimization in Design of Multidimensional Filter Networks

Mats Andersson^a

Oleg Burdakov^{b,1}

Hans Knutsson^a

Spartak Zikrin^b

LiTH-MAT-R-2013/16-SE

Revised 22 November 2014

^aDepartment of Biomedical Engineering, Linköping University, SE-581 85 Linköping, Sweden

^bDepartment of Mathematics, Linköping University, SE-581 83 Linköping, Sweden

¹Corresponding author. E-mail address: oleg.burdakov@liu.se

Sparsity Optimization in Design of Multidimensional Filter Networks

Mats Andersson* Oleg Burdakov[†]
Hans Knutsson* Spartak Zikrin[†]

Abstract Filter networks are used as a powerful tool used for reducing the image processing time and maintaining high image quality. They are composed of sparse sub-filters whose high sparsity ensures fast image processing. The filter network design is related to solving a sparse optimization problem where a cardinality constraint bounds above the sparsity level. In the case of sequentially connected sub-filters, which is the simplest network structure of those considered in this paper, a cardinality-constrained multilinear least-squares (MLLS) problem is to be solved. Even when disregarding the cardinality constraint, the MLLS is typically a large-scale problem characterized by a large number of local minimizers, each of which is singular and non-isolated. The cardinality constraint makes the problem even more difficult to solve.

An approach for approximately solving the cardinality-constrained MLLS problem is presented. It is then applied to solving a bi-criteria optimization problem in which both the time and quality of image processing are optimized. The developed approach is extended to designing filter networks of a more general structure. Its efficiency is demonstrated by designing certain 2D and 3D filter networks. It is also compared with the existing approaches.

Keywords: Sparse optimization; Cardinality Constraint; Multicriteria Optimization; Multilinear Least-Squares Problem; Filter networks; Medical imaging.

1 Introduction

Linear filtering is widely used in solving applied problems. We consider multidimensional finite impulse response (FIR) filters employed in noise reduction, segmentation and image registration for medical image analysis (Hemmendorff et al. 2002; Estepar 2007; Westin et al. 2001). The design of such filters is aimed at, on one hand, providing a sufficiently high image quality and robustness of image processing and, on the other hand, reducing the image processing time. The quality and processing time are two important characteristics in filtering. From the viewpoint of the multicriteria optimization (Ehrgott 2005; Miettinen 1999), they can be considered as two conflicting objectives, because any improvement in the quality requires a longer processing time.

A FIR filter is represented in the spatial domain by a limited-size kernel defined by a finite number of coefficients. They are the filter parameters chosen to meet certain requirements. So-called *dense* filters are characterized by a complete set of coefficients. They ensure the best approximation to a desired frequency response, and hence image quality. The processing time is proportional to the number of operations in the convolution of an image with the kernel, which in turn is proportional to the number of non-zero coefficients. Multidimensional images are typical for medical image applications, where the number of filter coefficients grows exponentially with the image dimension, which dramatically increases the processing time in three-dimensional (3D) and four-dimensional (4D) cases. The processing time can be reduced by means of *sparse* filters, i.e., by decreasing the number of non-zero coefficients, but this increases the approximation error and degrades the image quality.

*Department of Biomedical Engineering, Linköping University, 58185 Linköping, Sweden. (mats.x.andersson, hans.knutsson@liu.se)

[†]Department of Mathematics, Linköping University, 58183 Linköping, Sweden. (oleg.burdakov, spartak.zikrin@liu.se)

The development of design techniques for sparse FIR filters has been active in the past several years in the signal processing society, see, e.g., Baran et al. (2010), Lu and Hinamoto (2011), Rusu and Dumitrescu (2012). However it was mainly limited to single 1D and 2D filters. Here we consider a much more difficult problem of designing filter networks.

An effective approach aimed at reducing the image processing time, while maintaining its reasonably high quality, was proposed in Andersson et al. (1999) and Knutsson et al. (1999). Instead of traditionally used dense filter, it is based on using a network of sparse filters with kernels of smaller size. They are referred to as sub-filters.

Filter network can be represented as a directed acyclic graph, whose nodes stand for sub-filters (Svensson et al., 2005b). Sequentially connected sub-filters act as a single dense filter defined by their convolution in the spatial domain. The key property here is the sparsity, because if to compare a single dense filter and a filter network that provide equally good approximation of the same desired frequency response, the latter has far fewer non-zero coefficients.

In medical image analysis, several filters are usually required to process a signal, with each of them approximating a certain desired frequency response. An important feature of filter networks is that they can be efficiently used instead of such sets of filters (Knutsson et al., 1999).

The design of filter networks is a nontrivial procedure comprised of the following stages.

1. Choose a *network structure*, i.e., a number of sub-filters and their connection.
2. Choose a *sparsity pattern*, i.e., a total number of non-zero coefficients and their placement for each sub-filter.
3. Choose *values* of the coefficients.

The choices to be made at these stages are related to solving some optimization problems. The multi-extremal and multi-criterial nature of the problems makes it difficult, and in the most cases practically impossible, to solve them using conventional optimization methods. For this reason, the successful network design was restricted to special types of filters, for instance those used for analyzing local structures (Svensson et al. 2005a; George et al. 2008; Norell et al. 2011). The lack of efficient optimization approaches was a bottleneck for the further progress in filter network design.

So far, the decisions on a network structure and its sparsity pattern were mainly based on the individual expertise of specialists in filtering and their intuition. Given a sparsity pattern, the choice of the coefficients at stage 3 is related to solving a weighted nonlinear least-squares problem (Andersson et al. 1999; Svensson et al. 2005b; Norell et al. 2011). Even in the case of sequentially connected filters, the resulting problem is of a multilinear least-squares (MLLS) type, which is a non-convex large-scale optimization problem. This is a very difficult global optimization problem that may have a large number of local minima, and each of them is singular and non-isolated. In Andersson et al. (2012), we developed a global search strategy for approximately solving the MLLS problem.

Here we develop practically efficient approaches for the problem of finding an optimal design at stages 2 and 3. This problem is related to sparse optimization, where the task is to find a solution of a low cardinality, which is the number of non-zero coefficients. Sparse optimization is an intensively developing research area (Tropp and Wright 2010; Sun et al. 2013). It should be emphasized that, in general, sparse optimization problems are NP-hard even in the case of a simple objective function like in the linear least-squares (Muthukrishnan, 2005). Our problems are much more difficult because of the multi-extreme nature of the objective functions. Therefore, any direct application of the existing sparse optimization methods has practically no chance to find an acceptable approximate solution for stages 2 and 3.

Our approach is aimed at solving, with a practically acceptable accuracy, the cardinality-constrained nonlinear least-squares problem associated with filter design. It exploits the special structure of the objective function that allows for reducing the problem to a sequence of relatively easier cardinality-constrained linear least-squares problems. An important feature of this paper is that the objective function here is, in contrast to the most of the publications on sparse optimization, non-convex and multi-extremal.

We also develop an approach for approximately solving the bi-criteria optimization problem (Ehrgott 2005; Miettinen 1999), where the approximation error and the number of non-zero coefficients are to be minimized. The resulting approximations of Pareto optimal solutions enable the practitioners to find at

stages 2 and 3 a trade-off between the image processing quality and the processing time. Our approach is based on the aforementioned techniques for solving the cardinality-constrained nonlinear least-squares problem.

The solution obtained for stages 2 and 3 allows for improving the choice of the network structure made at stage 1. This is done by removing the redundant connections between sub-filters, and even some of the sub-filters.

Our paper is organized as follows. In Sect. 2, we develop methods for finding sparse solution to the MLLS problem related to the design of sequential filter networks. These methods are extended further in Sect. 3 to the design of networks of a more general structure. Numerical results in design of two- and three-dimensional filter networks are presented in Sect. 4. They indicate the efficiency of our approach. Sect. 5 includes some conclusions and discussion of future work.

2 Design of sequentially connected sub-filters

Consider the case of L sequentially connected filters (Fig. 1) and the optimization problems related to the design of such filter networks.

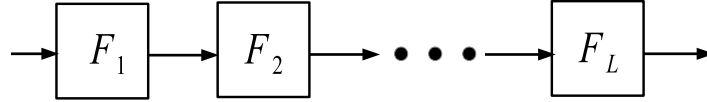


Figure 1: Sequential connection of L sub-filters

If $L = 1$, the kernel of a single filter of dimensionality d is determined by a finite number of coefficients $c = [c_1, \dots, c_l] \in C^l$ located in the discrete spatial positions $\{\xi_k\}_{k=1}^l$, where $\xi_k \in Z^d$ (Granlund and Knutsson 1995; Bigun 2006). For a frequency $u \in R^d$, the output of the filter is defined by the Fourier transform

$$F(u) = \sum_{k=1}^l c_k \exp(-iu^T \xi_k), \quad F : R^d \rightarrow C,$$

which depends linearly on c .

In the FIR filter design problem (Dudgeon and Mersereau, 1990), it is typical to sample the Fourier domain in m points u_1, \dots, u_m and split filter coefficients into real and imaginary parts. This allows for introducing the real-valued $x \in R^n$ and $A \in R^{m \times n}$ such that their product Ax represents the frequency response of the filter in the sampled points. Alternatively, one can impose symmetry constraints, when either real or imaginary part of c is used. In either case, the sparse x defines a sparse filter, and the number of its non-zero components is proportional to the non-zero filter coefficients.

Consider the more general case when $L \geq 1$. Let $x_i \in R^{n_i}$ and $A_i \in R^{m \times n_i}$ denote the individual characteristics of sub-filter i , for $i = 1, \dots, L$. By the convolution theorem (Bracewell and Bracewell, 1986), the frequency response of the sequentially connected sub-filters is defined by the multilinear operator

$$A(x) \equiv (A_1 x_1) \circ (A_2 x_2) \cdots \circ (A_L x_L), \quad (1)$$

where $x = [x_1^T, \dots, x_L^T]^T \in R^n$, $n = n_1 + \dots + n_L$ and \circ denotes the component-wise vector product. Given a desired frequency response $b \in R^m$, the design of a filter network at stages 2 and 3 requires seeking for a sparse approximation:

$$\text{Find sparse } x \quad \text{such that} \quad b \approx A(x). \quad (2)$$

In design of filters of dimensionality d , the most typical values are $m \in [10^d, 40^d]$, $L \in [2, 10]$, $n_i \in [3^d, 15^d]$. We shall refer to x_i as *segment i* of x . The number of non-zero components in x and x_i will be called their *sparsity level*.

In the following subsections, we formulate (2) as a cardinality-constrained optimization problem and present our approaches for solving such problems.

2.1 Cardinality-constrained MLLS

Suppose, the total number of non-zero components in the network is bounded above by a given value k , with $k \in [5d, 60d]$ being the most typical choice. Then we turn (2) into the following sparse optimization problem, in which the Euclidean norm of the approximation error is minimized subject to the cardinality constraint:

$$\min_{\|x\|_0 \leq k} \|b - A(x)\|^2 \equiv f(x). \quad (3)$$

Here $\|x\|_0$ stands for the number of non-zero components of x . In this paper, we consider the standard Euclidean norm of the approximation error for simplicity only, bearing in mind that our reasoning remains valid for any weighted Euclidean norm. Such weighted norms are usually used in practice.

If to disregard in (3) the cardinality constraint $\|x\|_0 \leq k$, the resulting unconstrained optimization problem is an MLLS problem. It is a non-convex, typically large-scale, optimization problem with a large number of local minimizers. Each of the local minimizers is singular and non-isolated.

If $L = 1$, problem (3) is a sparse linear least-squares problem, known also as *compressed sensing* and *compressive sampling* (Tropp and Wright, 2010). Since it is NP-hard (Muthukrishnan, 2005), the more general cardinality-constrained MLLS problem (3) with $L \geq 1$ is also NP-hard.

Our approach is based on the following observation. Given a feasible solution x for (3), consider the problem of finding a better sparsity pattern and values of the components for segment i , when the components of the other segments in x are fixed. Denoting $k_i = \|x_i\|_0$ for the given x_i , we can formulate it as the following cardinality-constrained linear least-squares problem for the new $x_i \in R^{n_i}$

$$\min_{\|x_i\|_0 \leq k_i} \|b - D_i A_i x_i\|^2, \quad (4)$$

where

$$D_i = \text{diag}((A_1 x_1) \circ \dots \circ (A_{i-1} x_{i-1}) \circ (A_{i+1} x_{i+1}) \circ \dots \circ (A_L x_L)).$$

From the theoretical point of view, problem (4) is still NP-hard. But in practice, the greedy pursuit algorithms, such as the orthogonal matching pursuit (OMP) and the orthogonal least-squares pursuit (OLS) (Chen et al. 1989; Pati et al. 1993) are able to produce reasonably good approximate solutions. They successively extend the basis composed of already selected columns of the matrix $D_i A_i$. At each iteration, a new basis element is selected of the remaining columns of this matrix following a greedy principle according to which its addition to the basis should decrease in the best way the approximation error in (4). We shall refer to the outlined algorithm of approximately solving problem (4) as $\text{GREEDY}(x, i, k_i)$. Its implementation could involve OMP or OLS. Algorithm $\text{GREEDY}(x, i, k_i)$ changes the segment x_i of the vector x to an approximate solution of (4) and returns the resulting vector x .

It is necessary to mention some other sparse optimization algorithms originally proposed for reconstructing sparse solution of underdetermined linear systems, namely, those based on convex relaxation of the cardinality constraint, such as GPSR (Figueiredo et al., 2007), SPGL1 (van den Berg and Friedlander, 2009), l1_ls (Kim et al., 2007), the l_1 -MAGIC package (Candès and Romberg, 2005), SOLVELASSO in the SparseLab toolbox (Donoho et al., 2007) and iterative thresholding algorithms, such as IHT (Blumensath and Davies, 2009), AIHT (Blumensath, 2012). They were developed mainly for solving compressed sensing problems in which the number of columns is vastly larger than the number of rows, while in our problem (4) n_i is a few times less than m . In our numerical experiments, these algorithms were often less successful in solving (4) in terms of the approximation accuracy than the fairly fast greedy algorithms.

Our approach for solving problem (3) leans upon the above mentioned observation. The approach consists in using approximate solution of (4) for decreasing the current value of the objective function $f(x)$. One of the ways is to improve the sparsity pattern of each segment of x . The other way is to try to decrease $f(x)$ by decreasing the number of non-zero components in one segment and increasing accordingly the number of non-zero components in the rest of segments. In the both cases, the corresponding component values of each segment of x are obtained by approximately minimizing $f(x)$.

This general approach can be implemented in various ways. One of such implementations is, first, outlined and then formally presented below by our cardinality-constrained alternating least-squares algorithm CCALS(x).

In CCALS(x), x is a given starting feasible solution. The algorithm returns an approximate solution of (3) whose cardinality does not exceed k . Its each major iteration consists of the two phases described below. The algorithm iterates over these phases until no improvement is achieved. In this process, it successively generates candidate solutions x^c and stores in x the best of them, i.e. the one whose objective function value is currently the lowest.

In phase one, the sparsity pattern of each segment i is individually improved using GREEDY(x, i, k_i) for alternating index i . In this phase, the number of non-zero components of each segment of x remains the same, while their locations and values may change.

In any of the phases, as soon as a new sparsity pattern is obtained, the candidate solution x^c is further refined by means of the described below algorithm ALS(x^c). It keeps the sparsity pattern unchanged. This algorithm implements a local optimization strategy called alternating least squares, known also as block-coordinate relaxation or nonlinear Gauss-Seidel algorithm (Ortega and Rheinboldt, 2000). Algorithm ALS(x^c) exploits the special structure of the objective function in the MLLS problem in the following way. It solves, for alternating index j , the linear least-squares problem that consists in minimizing $f(x^c)$ over the segment x_j^c while keeping fixed the rest of the segments of x^c . Whenever x^c provides in this process an improvement, which means $f(x^c) < f(x)$, x^c is stored in x .

In phase two, we try to improve the currently best approximate solution of (3) by redistributing the sparsity level between the segments. This is done in two loops running over the segments. In the outer loop, the sparsity level of segment i is decreased by one, and the corresponding sparsity is obtained by means of GREEDY($x, i, k_i - 1$). The resulting temporary solution x^t is then refined by ALS and used in the inner loop. The inner loop iterates over the segments $j \neq i$. At each inner iteration, GREEDY($x^t, j, k_j + 1$) is used with the aim to improve the temporary solution by increasing by one the sparsity level of segment j . The resulting candidate solution x^c is then refined using ALS as described above.

Consider the following alternative ways of implementing our approach. The outlined implementation can be extended to the case when one or more non-zero components of one segment are reallocated at a time in the other segments. This would make the new implementation more computationally demanding, but at the same time, this would have more chances to improve the objective function in (3), which is essentially the filter approximation quality. Another alternative is to improve the approximation for the new sparsity pattern by solving the corresponding MLLS problem using either the global search approach developed in Andersson et al. (2012) or by randomly generating starting points and refining each of them by ALS.

In the presentation of our approach, we use a specific local search algorithm, ALS, to minimize $\|b - A(x)\|^2$ for a given sparsity pattern. It should be underlined that any other appropriate local search, like the Levenberg-Marquardt algorithm (Nocedal and Wright, 2006), can be used instead of ALS. Similarly, problem (4) can be solved not necessary by greedy algorithms. The choice of the specific algorithms depends of the network structure and the ideal frequency response to be approximated. For the problem instances considered in this paper, the ALS and greedy algorithms were the most appropriate.


```

Algorithm CCALS( $x$ )
 $improvement \leftarrow true$ 
 $f \leftarrow f(x)$ 
while  $improvement$ 
  // phase 1:
  while  $improvement$ 
     $improvement \leftarrow false$ 
    for  $i = 1, \dots, L$ 
       $k_i \leftarrow \|x_i\|_0$ 
      if  $k_i > 0$ 
         $x^c \leftarrow \text{GREEDY}(x, i, k_i)$ 
         $x^c \leftarrow \text{ALS}(x^c)$ 
        if  $f(x^c) < f$ 
           $improvement \leftarrow true$ 
           $x \leftarrow x^c, f \leftarrow f(x^c)$ 
    end(for)
  end(while)
  // phase 2:
   $improvement' \leftarrow true$ 
  while  $improvement'$ 
     $improvement' \leftarrow false$ 
    for  $i = 1, \dots, L$ 
       $k_i \leftarrow \|x_i\|_0$ 
      if  $k_i - 1 > 0$ 
         $x^t \leftarrow \text{GREEDY}(x, i, k_i - 1)$ 
         $x^t \leftarrow \text{ALS}(x^t)$ 
        for  $j = 1, \dots, i - 1, i + 1, \dots, L$ 
           $k_j \leftarrow \|x_j^t\|_0$ 
          if  $k_j + 1 \leq n_j$ 
             $x^c \leftarrow \text{GREEDY}(x^t, j, k_j + 1)$ 
             $x^c \leftarrow \text{ALS}(x^c)$ 
            if  $f(x^c) < f$ 
               $improvement' \leftarrow true$ 
               $x \leftarrow x^c, f \leftarrow f(x^c)$ 
          end(for)
        end(for)
      if  $improvement'$ 
         $improvement \leftarrow true$ 
    end(while)
  end(while)

```

2.2 Bi-criteria optimization problem

The optimization problems related to the design of a filter network was so far considered for a given sparsity level of x . In this subsection, the sparsity level of x is allowed to vary.

Consider the following bi-criteria optimization problem:

$$\min_{x \in R^n} \{f(x), \|x\|_0\}, \quad (5)$$

in which both the approximation error $f(x)$ and the sparsity level $\|x\|_0$ are minimized.

The solution of problem (5) is a, so-called, *Pareto set* (Mietinen 1999; Ehrgott 2005). Point $x^* \in R^n$ is called *Pareto optimal* if there is no other point $x' \in R^n$ which would have a better objective function value in one of the two criteria and not worse in the other. In our case, the Pareto set is composed of the

pairs $\{f(x^*), \|x^*\|_0\}$ computed for all Pareto optimal points x^* . Note that the value of $f(x^*)$ is obviously monotonically decreasing with the increasing value of $\|x^*\|_0$.

In the simplest case, when the network consists of a single filter ($L = 1$) and $f(x) = \|b - Ax\|^2$, the Pareto set can be efficiently approximated by the algorithms like OMP and OLS. At each iteration of these algorithms the sparsity level is increased by one.

We present below an approach aimed at approximately solving problem (5) for $L \geq 1$. Note that, in contrast to $L = 1$, in the case of $L > 1$ there is practically no chance to find an exact minimizer of $f(x)$ for fixed sparsity pattern. Our approach is implemented in Pareto Set Approximation algorithm $\text{PSA}(x)$. The algorithm generates, for monotonically decreasing values of k , a sequence $x(k)$ such that $\|x(k)\|_0 = k$, and the set of pairs $\{f_k, k\}$, where $f_k \equiv f(x(k))$, approximates the Pareto set.

To construct a new approximation to the Pareto optimal solution of smaller cardinality, we try to decrease in the best way the sparsity level of the recently constructed one. We temporarily store the current solution x of cardinality k as x^t and iterate over the segments. For segment i , its sparsity level is decreased by δ using $\text{GREEDY}(x^t, k_i - \delta, i)$, and then the generated candidate solution x^c is refined by ALS. The best of the candidate solutions is further refined by CCALS and constitutes an approximation to the Pareto optimal solution of cardinality $k - \delta$.

In general, it may happen, that the new approximate solution dominates one or more of those already generated for larger cardinality. To preserve the monotonicity of the Pareto set, the new solution substitutes for those dominated. The substituted solutions can be improved by refining the dominating solution for the increasing cardinality.

Here the parameter δ is not specified. It can be either fixed or adjusted dynamically depending on the relations between f_k and $f_{k+\delta}$. For instance, a small relative change in the objective function indicates that one can choose a larger value of δ without any significant deviation from the Pareto set. In principle, it is possible to approximate a Pareto set using monotonically increasing values of k , but in our numerical experiments the reverse order was more successful.

Algorithm $\text{PSA}(x)$

```

 $x \leftarrow \text{CCALS}(x)$ 
 $K \leftarrow \|x\|_0, k \leftarrow K$ 
 $x(k) \leftarrow x$ 
 $f_k \leftarrow f(x)$ 
while  $\max\{\|x_i\|_0\} > \delta$ 
     $x^t \leftarrow x, f \leftarrow \infty$ 
    for  $i = 1, \dots, L$ 
         $k_i \leftarrow \|x_i^t\|_0$ 
        if  $k_i - \delta > 0$ 
             $x^c \leftarrow \text{GREEDY}(x^t, i, k_i - \delta)$ 
             $x^c \leftarrow \text{ALS}(x^c)$ 
            if  $f(x^c) < f$ 
                 $x \leftarrow x^c, f \leftarrow f(x^c)$ 
    end(for)
     $x \leftarrow \text{CCALS}(x)$ 
     $x(k) \leftarrow x, f_k \leftarrow f(x)$ 
end(while)
while  $k + \delta \leq K$ 
    if  $f_k < f_{k+\delta}$ 
         $x(k + \delta) \leftarrow x(k), f_{k+\delta} \leftarrow f_k$ 
     $k \leftarrow k + \delta$ 
end(while)

```

The developed approaches can be extended to design of a more general class of filter networks, where sub-filters are connected not only sequentially, but also in parallel.

3 Design of layered filter networks

Consider filter networks of acyclic structure in which sub-filters are grouped into layers according to the following principle (Svensson et al., 2005b). Arcs that begin in a sub-filter of one layer can end only in sub-filters of the lower layers, which means that there are no direct connections between the sub-filters of the same layer. In the special case, when layered network is composed of sequentially connected sub-filters, each layer consists of only one sub-filter.

Consider the frequency response of layered filter networks. It can be presented as a sum of terms of the type (1), with the number of terms being equal to the number of all directed paths in the acyclic graph from the starting to the terminal node. If sub-filters i and j belong to the same layer then, due to the layered structure, $A_i x_i$ and $A_j x_j$ never appear in the same term. This means that, for any given layer, if the coefficients of the sub-filters in the other layers are fixed, the network frequency response depends linearly on the coefficients of the sub-filters in the given layer. We will exploit this feature of layered filter networks.

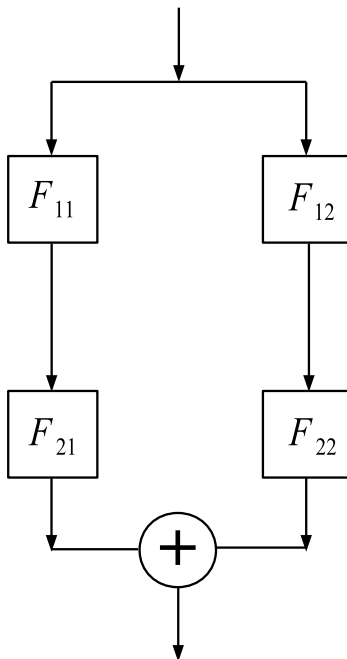


Figure 2: Filter network of parallel two-layers structure

For illustration, consider a simple filter network of a parallel structure with two layers (see Fig. 2). Here the first layer is composed of the sub-filters F_{11} and F_{12} and the second is composed of the sub-filters F_{21} and F_{22} . For more sophisticated layered filter networks, see, e.g., Svensson et al. (2005a) and George et al. (2008). Denote the individual characteristics of sub-filter F_{ij} , $i, j = 1, 2$, by $A_{ij} \in R^{m \times n_{ij}}$ and $x_{ij} \in R^{n_{ij}}$. Then the error of approximating the desired frequency response b takes the form

$$f(x) = \|b - (A_{11}x_{11}) \circ (A_{21}x_{21}) - (A_{12}x_{12}) \circ (A_{22}x_{22})\|^2, \quad (6)$$

where $x = [x_{11}^T, x_{21}^T, x_{12}^T, x_{22}^T]^T \in R^n$ and $n = n_{11} + n_{21} + n_{12} + n_{22}$. This approximation quality measure is the objective function in the general cardinality-constrained nonlinear least-squares problem

$$\min_{\|x\|_0 \leq k} f(x) \quad (7)$$

associated with the design of filter networks.

We will now show how to adapt algorithms CCALS and PSA to solving problems (7) and (5), respectively, when the objective function $f(x)$ is defined by (6). For this purpose, denote $x_i = [x_{i1}^T, x_{i2}^T]^T \in \mathbb{R}^{n_{i1}+n_{i2}}$ and $D_{ij} = \text{diag}(A_{sj}x_{sj})$, where $i, s \in \{1, 2\}$ and $i \neq s$. Let x be a given feasible solution for (7) such that $\|x_i\|_0 = k_i$. Consider the problem of finding an optimal sparsity pattern and values for the components of x that correspond to the sub-filters in layer $i \in \{1, 2\}$, provided that the values of the other components are kept unchanged. It can be written as the following cardinality-constrained linear least-squares problem in the new values of the corresponding components $x_i \in \mathbb{R}^{n_i}$:

$$\min_{\|x_i\|_0 \leq k_i} \|b - [D_{i1}A_{i1}, D_{i2}A_{i2}] x_i\|^2. \quad (8)$$

Observe that this problem resembles to (4). This implies that the filter network on Fig. 2 can be treated by algorithms CCALS and PSA as a sequential connection of two sub-filters, with each of them corresponding to one layer.

It can be easily verified, that the problem of improving the sparsity of each layer in layered filter networks of a general structure can be reduced to solving a cardinality-constrained linear least-squares problem like (8). This is the key observation that permits adopting algorithms CCALS and PSA for the purpose of designing layered filter networks.

4 Numerical Experiments

Our numerical experiments were related to designing 2D and 3D filters of the monomial class (Knutsson et al., 2011), whose frequency response has the form

$$R(\rho)D(\hat{u}),$$

where $u = \rho\hat{u}$ and $\rho = \|u\|_2$. Here $R(\rho)$ is the lognormal radial part (Knutsson 1982; Granlund and Knutsson 1995) and $D(\hat{u})$ is the directional part along the coordinate axes. Such filters are efficiently used for estimating local structure in medical imaging. Depending on the properties of the frequency response, we distinguish band-pass (BP) and high-pass (HP) filters of various order and direction.

The numerical experiments were performed on a Linux machine with a 6-core 2.27GHz Intel Xeon E5520 processor. All algorithms were implemented in MATLAB R2011b and accelerated by parallelizing the for-loops in CCALS and GREEDY by means of the Matlab routine PARFOR. Our implementation of algorithm GREEDY was based on OLS, because it was most often producing solutions of a higher accuracy than OMP. We used procedure GREED-OLS from the package SPARSIFY by Blumensath (2011).

4.1 Sequential filter network

In the first series of experiments, we considered 3D filter networks composed of sequentially connected sub-filters. Procedure CCALS was applied to solving the cardinality-constrained MLLS problem (3). For assessing the approximation quality, we used the relative error calculated by the formula

$$\varepsilon = \sqrt{f(x)}/\|b\|. \quad (9)$$

In practice, the approximation is considered as acceptable when $\varepsilon \leq 0.2$.

Our test problems were related to the following choice of parameters in problem (3):

$$m = 1330, \quad L = 5, \quad n = 217, \quad N = (63, 63, 63, 14, 14), \quad k = 20,$$

where $N = (n_1, \dots, n_L)$.

We choose the initial solution in the way that all 20 non-zero components are distributed in the first sub-filter. This means that the sparsity level of each of the other sub-filters is zero. Our convention here is to view such sub-filters as if they are not involved in the actual signal processing by acting as identity operators.

The initial location of the nonzero components in the first sub-filter and their values are found using OLS. The resulting relative error denoted by ε_0 is presented in Table 1. It refers to a single sparse

Filter	ε_0	ε_{sp}	ε_d	$\Delta\varepsilon \cdot 100\%$	sub-filters sparsity
BP	0.22	0.06	0.21	72.39	[9,0,0,4,7]
HP	0.19	0.04	0.19	80.31	[9,0,4,0,7]
BP, 1:st order, x	0.08	0.04	0.07	45.69	[9,0,8,3,0]
HP, 1:st order, y	0.17	0.02	0.16	85.30	[1,9,4,6,0]
BP, 2:nd order, x^2	0.30	0.08	0.26	73.91	[6,5,0,0,9]
BP, 2:nd order, xy	0.21	0.07	0.19	68.02	[8,3,9,0,0]
HP, 2:nd order, x^2	0.26	0.09	0.23	63.66	[2,8,5,5,0]
HP, 2:nd order, xy	0.24	0.14	0.23	42.11	[16,2,2,0,0]

Table 1: Performance of algorithm CCALS in designing 3D filter networks composed of sequentially connected sub-filters

filter used instead of a filter network. The sparsity of the filter network was then improved by means of algorithm CCALS. The resulting relative error is denoted by ε_{sp} . The relative improvement is calculated by the formula

$$\Delta\varepsilon = \frac{\varepsilon_0 - \varepsilon_{sp}}{\varepsilon_0}.$$

The last column in Table 1 presents the distribution of non-zero components among the sub-filters. One can see that not all of the sub-filters are used, typically, just a half of them. This information is of great importance for improving the network structure. For reference, we report the approximation error ε_d for the single dense filter whose number of components is equal to 63. It should be emphasized that this dense filter has approximately three times more coefficients than in our networks. This is indicative of the speed-up in the signal processing time provided by filter networks. The presented approximation quality illustrates one more advantage of filter networks over single filters, even if they are dense.

4.2 Parallel filter network

The second series of experiments is related to designing a 2D filter network of the parallel structure presented by Fig. 2. We compared our algorithm CCALS with the genetic algorithm tailored by Langer et al. (2005) to solving problem (7) with $f(x)$ defined by (6). The choice of the parameters involved in defining problem (7) was the following:

$$m = 1225, \quad n = 968, \quad n_{ij} = 242 \text{ for } i, j = 1, 2 \text{ and } k = 88.$$

The solution time reported by Langer (2004) was 48 hours on a computer with a 900 Mhz processor on a SunFire 6800 server. Our algorithm CCALS was run from 50 randomly generated starting points of the given cardinality. It produced a better approximation of the same desired frequency response, and this took only a few minutes of the CPU time. We mention here the CPU time just for information because in our case a modern PC was used. The approximation error is presented in Table 2, where the first three rows reproduce the results reported in Langer et al. (2005). In one of these rows, the low-rank approximation corresponds to the case when all sub-filters in the network are one-dimensional. This approach is based on the singular value decomposition (Andersson et al. 1999; Ansari and Hou 2012). The expert filter design mentioned in the table corresponds to the sparsity pattern suggested by experts in designing filter networks. Our results are reported in the last row.

The sparsity pattern produced by CCALS is presented in Fig. 3. In total, 44 complex-valued coefficients are placed, which corresponds to 88 non-zero real-valued decision variables in (7). One can see that only one of the two parallel branches of the network is used, which implies that a network of sequential structure might be a better choice for designing the considered filter than the parallel structure in Fig. 2. This illustrates one more important feature of our approach, namely, its ability to improve not only the network sparsity pattern, but also its structure.

Method	Normalized error
low-rank approximation	0.119
expert filter design	0.109
genetic algorithm by Langer et al. (2005)	0.054
CCALS	0.049

Table 2: Design of the 2D filter network of parallel structure

4.3 Pareto set approximation

In the third series of experiments, we evaluated the efficiency of algorithm PSA aimed at approximating the Pareto set in problem (5). It was applied in designing a 2D BP filter of order 1 along axis x with $m = 360$. We considered several network structures with the following parameters presented in Table 3.

Structure	Parameters
a) sequential	$L = 3, n = 30, N = (13, 13, 4)$
b) sequential	$L = 2, n = 37, N = (25, 12)$
c) parallel (Fig. 4)	$n = 56, n_{ij} = 13, i, j = 1, 2, n_{31} = 4$
d) single	$n = 40$

Table 3: List of network structures and their parameters

Recall that in algorithm PSA, δ is the step of decreasing the sparsity level. We observed in practice that for some, typically large, values of k the optimal objective function value changes slowly with k . In such cases, larger values of the step δ look more suitable for uniformly approximating the Pareto set by algorithm PSA. Therefore, to make our implementation of this algorithm more time efficient, the value of δ was chosen adaptively for each k . A proper value of δ was chosen in the process of running algorithm GREEDY in the cycle for i of algorithm PSA.

The results of Pareto set approximation in problem (5) are presented in Fig. 5 and Fig. 6 in the form of a set of non-dominated pairs (k, ε_k) , where ε_k is the relative error calculated by formula (9) for the sparsity level equal to k .

Note that algorithm CCALS is able to approximate the Pareto set if to run it for various values of sparsity level. For each k , we run CCALS from 30 randomly generated starting points and then choose the best objective function value. In Fig. 5, these results are presented along with those produced by algorithm PSA for designing filter network of structure a) in Table 3. They speak in favor of the efficiency of PSA in terms of its filter approximation accuracy. Its success is explained by the fact that, for the current value of k , a good starting point for solving (3) can be produced from the previously generated solution for the larger cardinality.

Fig. 6 presents the Pareto set approximations produced by algorithm PSA for the filter networks of the structures a)–d) in Table 3. One can see that the best design is obtained for the filter network of the parallel two-layers structure. Based on its Pareto set approximation, filter network developers can choose one of the non-dominated solutions, depending on the design preferences. For instance, one can either use 24 non-zero components with the relative error 0.06 or improve the image processing time by using only 13 components. However, in the latter case, the image quality is slightly worse, because the corresponding relative error is 0.07.

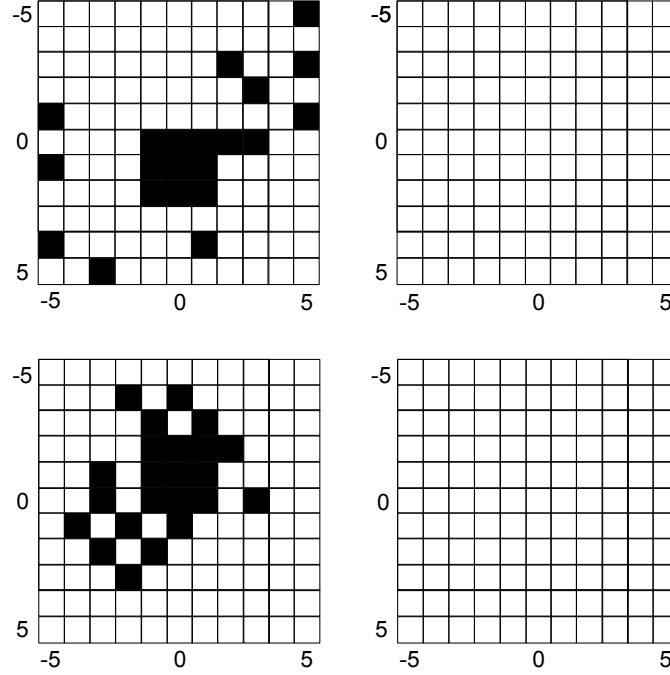


Figure 3: The sparsity pattern produced by CCALS for the filter network in Fig. 2. Sparsity pattern for sub-filters F_{11} (top left), F_{12} (top right), F_{21} (bottom left) and F_{22} (bottom right), where each cell corresponds to a spatial position associated with certain non-zero coefficient in Z^2 . The black cells refer to the non-zero coefficients.

5 Conclusions

The main aim of this paper was not in designing certain filter networks, but rather in developing optimization tools that can be used for this purpose. We have developed an efficient approach for optimizing sparsity pattern of filter networks. Its ability to improve network structure allows for automating some stages of designing filter networks. The Pareto set approximation offers practitioners a means of finding a proper trade-off between the image processing quality and time.

The numerical experiments were conducted with the purpose of demonstrating some of the high potentials of our generic approach. The presented results refer to some of its possible implementations. There are strong grounds for believing that, in the future, more refined implementations will be able to demonstrate in full scale the efficiency of our approach. These implementations improve, within an acceptable computation time, the approximation accuracy produced by the existing algorithms, while keeping the same sparsity.

The range of possible applications goes beyond the design of filter networks. Similar problems occur, e.g., in factor analysis, chemometrics, psychometrics (Leardi et al. 2000; Leurgans and Ross 1992; Lopes and Menezes 2003; Paatero 1997; Wang et al. 2003). Our approach may be extended to solving such problems.

Acknowledgements

This work was supported by the Swedish Research Council; the Linnaeus Center for Control, Autonomy, and Decision-making in Complex Systems (CADICS); the Swedish Foundation for Strategic Research (SSF) Strategic Research Center (MOVIII); the Linköping University Center for Industrial Information Technology (CENIIT).

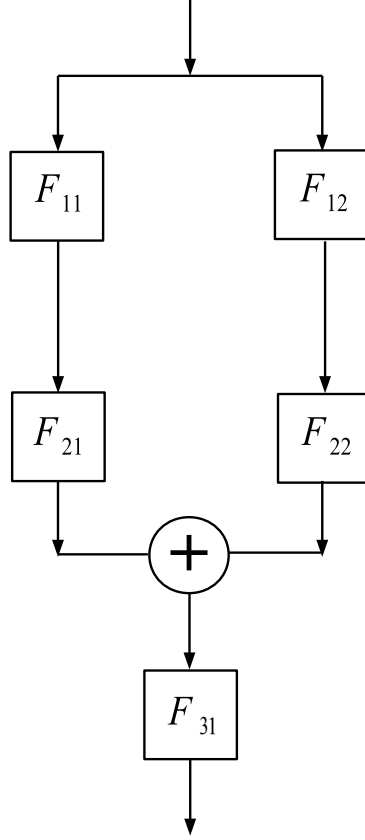


Figure 4: Filter network of parallel three-layers structure

References

- Andersson M, Wiklund J, Knutsson H (1999) Filter networks. In: Namazi N (ed) Signal and Image Processing (SIP), Proceedings of the IASTED International Conferences, October 18-21, 1999, Nassau, The Bahamas, IASTED/ACTA Press, pp 213 – 217
- Andersson M, Burdakov O, Knutsson H, Zikrin S (2012) Global search strategies for solving multilinear least-squares problems. SQU J Sci 17(1):12-21
- Ansari N, Hou E (2012) Computational intelligence for optimization. Springer Publishing Company, Inc.
- Baran T, Wei D, Oppenheim AV (2010) Linear programming algorithms for sparse filters design. IEEE Trans Signal Process, 58(3):1605–1617
- van den Berg E, Friedlander M (2009) Probing the Pareto frontier for basis pursuit solutions. SIAM J Sci Comput 31(2):890–912
- Bigun J (2006) Vision and direction. Springer, Heidelberg
- Blumensath T (2011) Sparsify. <http://users.fmrib.ox.ac.uk/~tblumens/sparsify/sparsify.html>. Accessed 1 May 2012
- Blumensath T (2012) Accelerated iterative hard thresholding. Signal Process 92(3):752 – 756

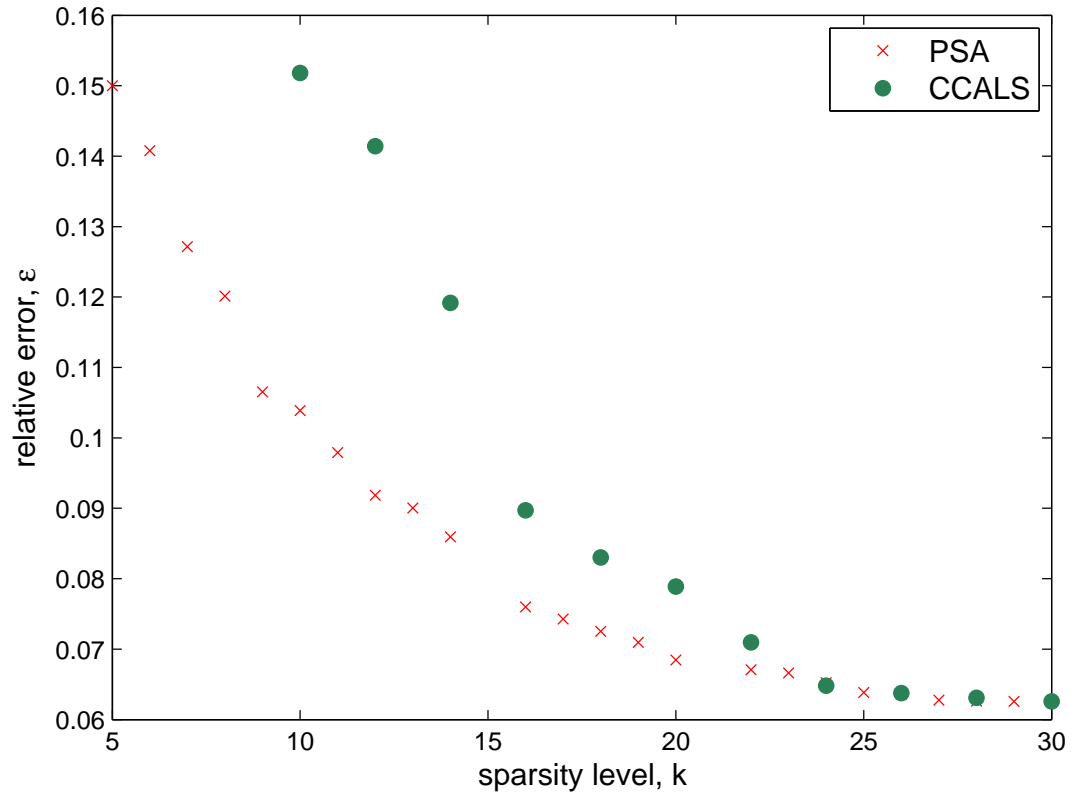


Figure 5: Pareto set approximations generated by PSA and CCALS for the sequential filter network of structure a) (see Table 3). The desired frequency response is defined by the 2D BP filter of order 1 along axis x .

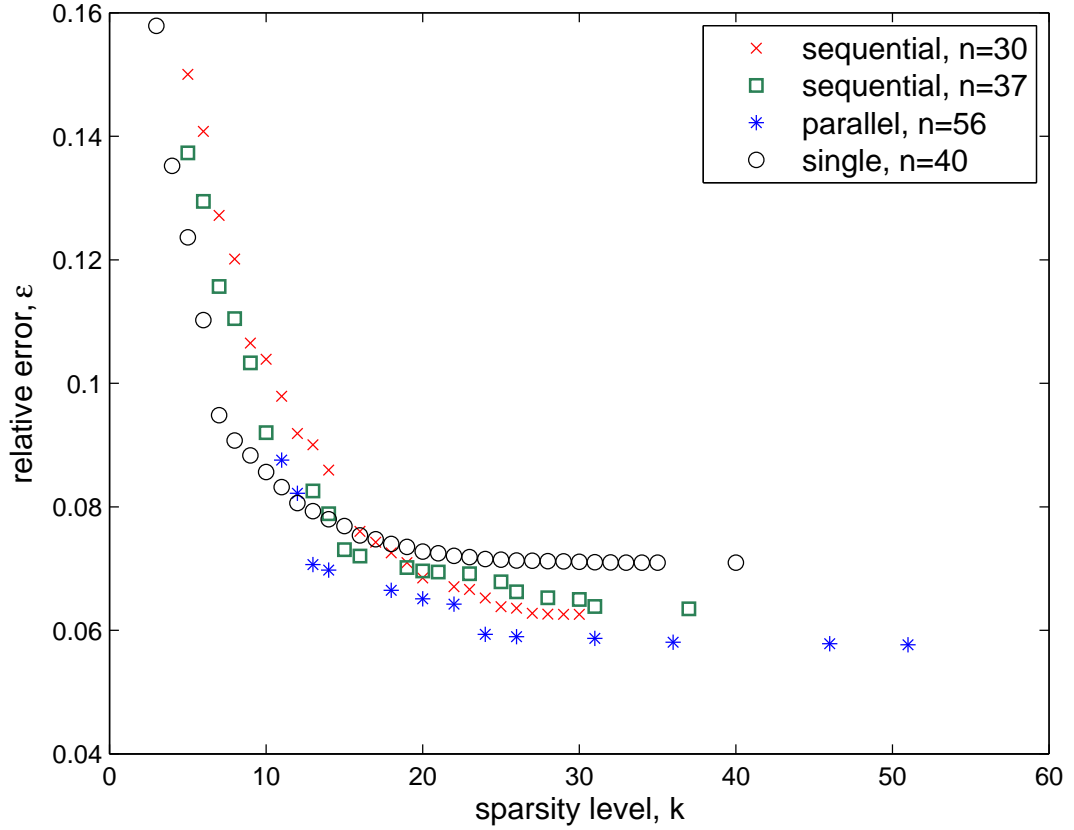


Figure 6: Pareto set approximations generated by PSA for four network structures (see Table 3). The desired frequency response is defined by the 2D BP filter of order 1 along axis x

Blumensath T, Davies ME (2009) Iterative hard thresholding for compressed sensing. *Appl Comput Harmon Anal* 27(3):265 – 274

Bracewell RN, Bracewell RN (1986) *The Fourier transform and its applications*, vol 31999. McGraw-Hill New York

Candès E, Romberg J (2005) l_1 -MAGIC: Recovery of sparse signals via convex programming. Tech. rep., California Institute of Technology, Pasadena, CA

Chen S, Billings SA, Luo W (1989) Orthogonal least squares methods and their application to non-linear system identification. *Int J Control* 50(5):1873–1896

Donoho D, Stodden V, Tsaig Y (2007) Sparselab. <http://sparselab.stanford.edu/>. Accessed 1 May 2012

Dudgeon DE, Mersereau RM (1990) *Multidimensional digital signal processing*. Prentice Hall Professional Technical Reference

Ehrgott M (2005) *Multicriteria optimization*. Springer-Verlag New York, Inc., Secaucus, NJ, USA

- Estepar RSJ (2007) Local Structure Tensor for Multidimensional Signal Processing. Applications to Medical Image Analysis. Presses universitaires de Louvain, Louvain-la-Neuve, Belgium
- Figueiredo MAT, Nowak RD, Wright SJ (2007) Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems. *IEEE J Sel Top Signal Process* 1(4):586–597
- George J, Indu SP, Rajeev K (2008) Three dimensional ultrasound image enhancement using local structure tensor analysis. In: *TENCON 2008 - 2008 IEEE Region 10 Conference*, pp 1–6
- Granlund G, Knutsson H (1995) Signal processing for computer vision. Kluwer, Dordrecht
- Hemmendorff M, Andersson M, Kronander T, Knutsson H (2002) Phase-based multidimensional volume registration. *IEEE Trans Med Imag* 21(12):1536–1543
- Kim SJ, Koh K, Lustig M, Boyd S (2007) An efficient method for compressed sensing. In: *Proceedings of IEEE International Conference on Image Processing*, vol 3, pp 117–120
- Knutsson H (1982) Filtering and reconstruction in image processing. Dissertation, Linköping University
- Knutsson H, Andersson M, Wiklund J (1999) Advanced filter design. In: *Proceedings of the 11th Scandinavian Conference on Image Analysis : Greenland, SCIA*, pp 185–193
- Knutsson H, Westin CF, Andersson M (2011) Representing local structure using tensors II. In: Heyden A, Kahl F (eds) *Image Analysis, Lecture Notes in Computer Science*, vol 6688, Springer, Berlin / Heidelberg, pp 545–556
- Langer M (2004) Design of fast multidimensional filters by genetic algorithms. Master thesis, Linköping University
- Langer M, Svensson B, Brun A, Andersson M, Knutsson H (2005) Design of fast multidimensional filters using genetic algorithms. In: Rothlauf F, Branke J, Cagnoni S, Corne D, Drechsler R, Jin Y, Machado P, Marchiori E, Romero J, Smith G, Squillero G (eds) *Applications of Evolutionary Computing, Lecture Notes in Computer Science*, vol 3449, Springer Berlin / Heidelberg, pp 366–375
- Leardi R, Armanino C, Lanteri S, Alberotanza L (2000) Three-mode principal component analysis of monitoring data from venice lagoon. *J Chemom* 14(3):187–195
- Leurgans S, Ross RT (1992) Multilinear models: applications in spectroscopy. *Stat Sci* 7(3):289–310
- Lopes JA, Menezes JC (2003) Industrial fermentation end-product modelling with multilinear PLS. *Chemom Intell Lab Syst* 68(1):75–81
- Lu WS, Hinamoto T (2011) Two-dimensional digital filters with sparse coefficients. *Multidimens Syst Signal Process* 22(1-3):173–189
- Miettinen K (1999) Nonlinear multiobjective optimization. Kluwer Academic Publishers, Boston, USA
- Muthukrishnan S (2005) Data streams: algorithms and applications. Now Publishers, Boston
- Nocedal J, Wright SJ (2006) Numerical Optimization. 2nd edn, Springer, New York.
- Norell B, Burdakov O, Andersson M, Knutsson H (2011) Approximate spectral factorization for design of efficient sub-filter sequences. Tech. Rep. LiTH-MAT-R-2011-14, Department of Mathematics, Linköping University
- Ortega JM, Rheinboldt WC (2000) Iterative solution of nonlinear equations in several variables, *Classics in Applied Mathematics*, vol 30, reprint of the 1970 original edn. SIAM, Philadelphia, PA

- Paatero P (1997) Least squares formulation of robust non-negative factor analysis. *Chemom Intell Lab Syst* 37(1):23 – 35
- Pati YC, Rezaiifar R, Krishnaprasad P (1993) Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition. In: *Proceedings of the 27th Annual Asilomar Conference on Signals, Systems, and Computers*, vol 1, pp 40–44
- Rusu C, Dumitrescu B (2012) Iterative reweighted l_1 design of sparse FIR filters. *Signal Process* 92(4):905–911
- Sun X, Zheng X, Li D (2013) Recent advances in mathematical programming with semi-continuous variables and cardinality constraint. *J Oper Res Soc China* 1(1):55–77
- Svensson B, Andersson M, Knutsson H (2005a) Filter networks for efficient estimation of local 3-d structure. In: *Proceedings of the IEEE International Conference on Image Processing*, vol 3, pp 573–576
- Svensson B, Andersson M, Knutsson H (2005b) A graph representation of filter networks. In: Kalviainen H, Parkkinen J, Kaarna A (eds) *Image Analysis, Lecture Notes in Computer Science*, vol 3540, Springer Berlin Heidelberg, pp 1086–1095
- Tropp JA, Wright SJ (2010) Computational methods for sparse solution of linear inverse problems. In: *Proceedings of the IEEE*, vol 98(6), pp 948–958
- Wang JH, Hopke PK, Hanczewicz TM, Zhang SL (2003) Application of modified alternating least squares regression to spectroscopic image analysis. *Anal Chim Acta* 476(1):93–109
- Westin CF, Wigström L, Looock T, Sjöqvist L, Kikinis R, Knutsson H (2001) Three-dimensional adaptive filtering in magnetic resonance angiography. *J Magn Reson Imaging* 14(1):63–71