1st International Conference "From Scientific Computing to Computational Engineering"
1st IC-SCCE
Athens, 8-10 September, 2004

WINDOW DESIGN AND ENHANCEMENT USING CHEBYSHEV OPTIMIZATION

To Tran*, Mattias Dahl*, Ingvar Claesson*, and Thomas Lagö†

*Blekinge Institute of Technology
Department of Signal Processing
Box 520, SE-372 25 Ronneby, Sweden

†Svantek America Inc.
70 Himalaya Ct Alpine
Utah 84004, USA

Keywords: Frequency Analysis, Window, Flattop, Optimization, DNCA

Abstract. *This paper presents a new and versatile framework to window design based on a semi–infinite linear programming approach by using the Dual Nestled Complex Approximation (DNCA) algorithm. The paper considers a practical problem formulation, the general window design specification and the corresponding optimum solution. The DNCA linear programming algorithm is presented and several highly optimized window design examples included. Furthermore, the capability of the design method by enhancing the sidelobe attenuation of existing windows such as Flattop windows is illustrated as well. Flattop windows are commonly available in frequency analyzers for accurate amplitude measurements of harmonic tones in presence of noise. However, the design method can be directly applied to any other existing window, in purpose to enhance its performance. With the proposed method it is possible to design windows with more than 150 dB sidelobe attenuation. A relevant application where such high precision windows is needed are in today's 24bit equivalent 1bit-Sigma-Delta ADC:s, where high frequency noise is generated.*

1 INTRODUCTION

The Fast Fourier transform (FFT) algorithm is commonly used for frequency domain calculations in for example the field of sound and vibration analysis. The original version of the FFT algorithm as known today[1] was presented 1965 by J.W. Tukey and J.W. Cooley[2] and is often referred to as *the FFT algorithm*, but similar algorithms developed later also goes under the same name. Even though it is called Fast Fourier transform it is not a new transform itself, rather a computational efficient method to calculate the discrete Fourier transform (DFT) of a finite extent data block. Therefore, the terms FFT and DFT are often used interchanged in literatures.

The FFT algorithm theoretically requires that the finite extent data block to be analyzed is periodic with an exact integer number of cycles within the captured data block. This requirement can be traced back to the periodic extension property of the DFT[3]. However, this rarely occurs in reality. If the captured data block is not periodic with an exact integer number of cycles, discontinuities will appear at the boundaries when the captured (non-periodic) data block is periodically repeated according to the periodic extension property.

These discontinuities will gives rise to the so called spectral leakage problem, which can be explained as the power originating from one frequency bin has been spread out over the entire frequency spectra. One method to reduce or control the leakage is to smooth out the discontinuities by smoothly bring the endpoints of the captured data block to zero or close to zero. This can be performed by multiplying the captured data block with a window function in the time domain, i.e. windowing.

Many different window functions have been designed and investigated throughout the years[4, 5, 6]. The window functions for spectral analysis described by Harris[4] are well established and documented. Those windows are often referred as classic windows, which are characterized by a narrow mainlobe and moderate sidelobe attenuation and can be used in a majority of analysis situations.

Another class of windows, the so called flattop windows, are characterized by a wide and flat mainlobe and high sidelobe attenuation. The flattop windows are mainly designed for correct amplitude measurements and instrument calibration.

This paper presents a new symmetric (flattop) window design strategy using the Dual Nested Complex Approximation (DNCA) algorithm following the principle of window construction using the summation of shifted Dirichlet kernels. The DNCA optimization algorithm was originally designed to solve semi–infinite linear (or quadratic) programs, but as this paper indicates, it is also very efficient in solving optimization problems with a few variables (unknowns) and a huge number of constraints. The DNCA algorithm has been proven to be useful and effective for complex approximation with any filter having linear structure, such as Laguerre networks[7], narrowband as well as broadband beamformers[8, 9, 10] and digital channel equalizers[11]. The advantage of windows constructed with shifted Dirichlet kernels is that the obtained window coefficient set can be used to generate either a symmetric or periodic window, with arbitrary length. The main difference of designing symmetric windows than periodic windows[12], is that the phase is given *a priori* in the symmetric case, which implies that the optimization problem becomes real valued with a finite number of constraints.

This paper includes three window design examples illustrating the efficiency of using the DNCA optimization algorithm and the flexibility of the design procedure. It is worth to note that the design examples included in this paper might be solved using conventional optimization softwares, but with a considerably increase of computational complexity and memory consumption.

2 PROBLEM FORMULATION

Throughout the years, many different window functions have been constructed using the summation of shifted Dirichlet kernels. A symmetric and discrete-time window constructed in this way is defined as

$$w(n) = \sum_{k=0}^{K-1} (-1)^k a_k \cos\left(\frac{2\pi k n}{N-1}\right), \quad n = 0, \ldots, N-1 \tag{1}$$

where K denotes the number of window coefficients a_k and N denotes the length of the discrete time window. The Fourier transform of Eq. (1) is given by

$$W(\omega) = \sum_{n=0}^{N-1} \left(\sum_{k=0}^{K-1} (-1)^k a_k \cos\left(\frac{2\pi k n}{N-1}\right)\right) e^{-j\omega n} \tag{2}$$

By changing the summation order and rearranging the variables gives

$$W(\omega) = \sum_{k=0}^{K-1} (-1)^k a_k \sum_{n=0}^{N-1} \cos\left(\frac{2\pi k n}{N-1}\right) e^{-j\omega n} \tag{3}$$

Eq. (3) can be written as

$$W(\omega) = \sum_{k=0}^{K-1} (-1)^k a_k \sum_{n=0}^{N-1} \frac{1}{2} \left(e^{j\frac{2\pi k n}{N-1}} + e^{-j\frac{2\pi k n}{N-1}} \right) e^{-j\omega n} \tag{4}$$

$$= \sum_{k=0}^{K-1} (-1)^k \frac{a_k}{2} \left(\mathcal{D}\left(\omega - \frac{2\pi k}{N-1}\right) + \mathcal{D}\left(\omega + \frac{2\pi k}{N-1}\right)\right) \tag{5}$$

where $\mathcal{D}(\omega)$ is known as the Dirichlet kernel defined as

$$\mathcal{D}(\omega) = e^{-j\omega\left(\frac{N-1}{2}\right)} \frac{\sin\left(\omega \frac{N}{2}\right)}{\sin\left(\omega \frac{1}{2}\right)} \tag{6}$$

Equation (5) clearly shows that any window constructed using Eq. (1) has a Fourier Transform which consist of a summation of shifted Dirichlet kernels. Since $w(n)$ is symmetric, the frequency response of the window can be written as,

$$W(\omega) = A(\omega) e^{j\theta(\omega)} \tag{7}$$

where $A(\omega)$ is the real analytic amplitude function (with sign) and $\theta(\omega)$ is the continuous linear phase function given by

$$A(\omega) = \frac{1}{2}\sum_{k=0}^{K-1} a_k \left(\frac{\sin\left(\frac{N}{2}\left(\omega - \frac{2\pi k}{N-1}\right)\right)}{\sin\left(\frac{1}{2}\left(\omega - \frac{2\pi k}{N-1}\right)\right)} + \frac{\sin\left(\frac{N}{2}\left(\omega + \frac{2\pi k}{N-1}\right)\right)}{\sin\left(\frac{1}{2}\left(\omega + \frac{2\pi k}{N-1}\right)\right)} \right) \ , \ \theta(\omega) = -\left(\frac{N-1}{2}\right)\omega \tag{8}$$

By defining a new variable

$$d(\omega, k) = \left(\frac{\sin\left(\frac{N}{2}\left(\omega - \frac{2\pi k}{N-1}\right)\right)}{\sin\left(\frac{1}{2}\left(\omega - \frac{2\pi k}{N-1}\right)\right)} + \frac{\sin\left(\frac{N}{2}\left(\omega + \frac{2\pi k}{N-1}\right)\right)}{\sin\left(\frac{1}{2}\left(\omega + \frac{2\pi k}{N-1}\right)\right)} \right) \tag{9}$$

and introduce the vector notations

$$\phi(\omega) = \begin{bmatrix} \frac{1}{2}d(\omega,0) \\ \frac{1}{2}d(\omega,1) \\ \vdots \\ \frac{1}{2}d(\omega,K-1) \end{bmatrix} \ , \quad \mathbf{a} = \begin{bmatrix} a_0 \\ a_1 \\ \dots \\ a_{K-1} \end{bmatrix} \tag{10}$$

the amplitude function $A(\omega)$ can be written as

$$A(\omega) = \phi^T(\omega)\mathbf{a} \tag{11}$$

3 THE WINDOW DESIGN SPECIFICATION

An essential analytical advantage with symmetric window design over periodic window design[12] is that the amplitude function $A(\omega)$ in Eq. (11) can be made real and linear with respect to the window coefficients a_k. Since $|A(\omega)| = |W(\omega)|$ and the phase function $\theta(\omega)$ is given a priori, only the amplitude function needs to be specified for the design. Consider the following window design specification

$$\begin{cases} \min ||\phi^T(\omega)\mathbf{a} - A_d(\omega)||_\infty, & \omega \in \Omega_s \\ |\phi^T(\omega)\mathbf{a} - A_d(\omega)| \le \sigma_p(\omega) & \omega \in \Omega_p \end{cases} \tag{12}$$

where Ω_p and Ω_p denotes the passband and the stopband respectively and $\sigma_p(\omega)$ the corresponding amplitude bound in the passband. The desired amplitude response $A_d(\omega)$ for a flattop window is given by

$$A_d(\omega) = \begin{cases} 1 & \omega \in \Omega_p = [0, \omega_p] \\ 0 & \omega \in \Omega_s = [\omega_s, \pi] \end{cases} \tag{13}$$

where ω_p denotes the passband edge and ω_s the stopband edge respectively. By introducing a new variable $\delta = \max\limits_{\omega\in\Omega_s} |\phi^T(\omega)\mathbf{a} - A_d(\omega)| = ||\phi^T(\omega)\mathbf{a} - A_d(\omega)||_\infty, \omega \in \Omega_s$, Eq. (12) can be rewritten as

$$\begin{cases} \min \delta \\ |\phi^T(\omega)\mathbf{a} - A_d(\omega)| \le \delta & \omega \in \Omega_s \\ |\phi^T(\omega)\mathbf{a} - A_d(\omega)| \le \sigma_p(\omega) & \omega \in \Omega_p \end{cases} \tag{14}$$

Note that the extra independent variable δ is active only in the stopband and the amplitude margin $\sigma_p(\omega)$ is only active in the passband. The non-linear constraints in Eq. (14) makes the problem difficult to solve as it stands. By using the fact that an absolute magnitude inequality constraint can be written as

$$|x| \le b \Leftrightarrow \begin{cases} x \le b \\ -x \le b \end{cases} , \tag{15}$$

and Eq. (14) becomes

$$\begin{cases} \min \delta \\ (\phi^T(\omega)\mathbf{a} - A_d(\omega)) & \le & \delta & \omega \in \Omega_s \\ -(\phi^T(\omega)\mathbf{a} - A_d(\omega)) & \le & \delta & \omega \in \Omega_s \\ (\phi^T(\omega)\mathbf{a} - A_d(\omega)) & \le & \sigma_p(\omega) & \omega \in \Omega_p \\ -(\phi^T(\omega)\mathbf{a} - A_d(\omega)) & \le & \sigma_p(\omega) & \omega \in \Omega_p \end{cases} \tag{16}$$

For practical purpose of the implementation of the algorithm, Ω_p and Ω_s are assumed to be finite subsets of the continuous frequency domain $\Omega = [0, \pi]$. In the discrete frequency domain ω_i, where $i = 1, \ldots, I$, the design problem can be formulated as

$$\begin{cases} \min \delta \\ (\phi^T(\omega_i)\mathbf{a} - A_d(\omega_i)) & \leq & \delta & \omega_i \in \Omega_s \\ -(\phi^T(\omega_i)\mathbf{a} - A_d(\omega_i)) & \leq & \delta & \omega_i \in \Omega_s \\ (\phi^T(\omega_i)\mathbf{a} - A_d(\omega_i)) & \leq & \sigma_p(\omega_i) & \omega_i \in \Omega_p \\ -(\phi^T(\omega_i)\mathbf{a} - A_d(\omega_i)) & \leq & \sigma_p(\omega_i) & \omega_i \in \Omega_p \end{cases} \tag{17}$$

Equation (17) can now be formulated as a minimax optimization program in standard form, which can be solved using conventional available linear programming software. However, for large number of constraints, conventional linear programming software are both time consuming and memory extensive.

Therefore, this paper suggests that Eq. (14) should be solved by using the DNCA algorithm, described in the next section. The DNCA algorithm was originally developed to solve semi-infinite optimization problems, but it is also very efficient in solving finite-dimensional optimization problems with a large number of constraints.

On the a discrete frequency domain, Eq. (14) can be written as

$$\begin{cases} \min \delta \\ |\phi^T(\omega_i)\mathbf{a} - A_d(\omega_i)| - \delta \cdot t(\omega_i) \leq \sigma_p(\omega_i) \quad \omega_i \in \Omega_p \bigcup \Omega_s \end{cases} \tag{18}$$

where

$$t(\omega_i) = \begin{cases} 1 & \omega_i \in \Omega_s \\ 0 & \omega_i \in \Omega_p \end{cases} \tag{19}$$

emphasizes that the extra independent variable δ is only active in the stopband.

Equation (18) is now expressed in a form which is in accordance with the DNCA formulation. Even though the constraints are real valued.

4 DUAL NESTED COMPLEX APPROXIMATION ALGORITHM

The Dual Nested Complex Approximation (DNCA) algorithm has shown to be very efficient in solving semi-infinite linear and quadratic programs[11, 13], i.e. optimization problem with a finite number of variables (unknowns) and infinite number of constraints. Even though the optimization problem formulation in this paper is not semi-infinite, the DNCA algorithm is still useful. The DNCA algorithm is capable to reduce the computation time significantly and is less memory extensive compared to conventional optimization software, which can be used to solve Eq. (17).

This section will give a general description of the DNCA algorithm, which yields for conventional or semi-infinite, linear or quadratic, optimization problems.

4.1 The Approximation Problem

A general optimization problem can be formulated as

$$(P) \quad \begin{cases} \min\limits_{\mathbf{x}} f(\mathbf{x}), \\ g_\alpha(\mathbf{x}) \leq 0, \qquad \alpha \in \mathcal{A} \subset R^k \\ \mathbf{x} \in \mathcal{X} \subset R^n \end{cases} \tag{20}$$

where $\mathbf{x}$ is an $N \times 1$ variable vector, $f(\mathbf{x})$ a convex continuous function, $\mathcal{X}$ a convex restriction set, $\mathcal{A}$ an infinite or large index set as a compact subset of Euclidean k-space, and $g_\alpha(\mathbf{x})$ a continuous constraint function which is convex for any fixed index α.

4.2 The DNCA-LP Optimization Algorithm

The Dual Nested Complex Approximation Linear Programming algorithm to solve Eq. (20) is outlined below. Let $\mathcal{A}^{(k)}$ denote a sequence of subsets of the index set $\mathcal{A}$ and initialize the algorithm with the subset $\mathcal{A}^{(0)}$.

1. Given $\mathcal{A}^{(k)} \subset \mathcal{A}$, solve the subproblem

$$\left(P^{(k)}\right) \quad \begin{cases} \min\limits_{\mathbf{x}} f(\mathbf{x}), \mathbf{x} \in \mathcal{X} \\ g_{\alpha_k}(\mathbf{x}) \leq 0, \qquad \alpha_k \in \mathcal{A}^{(k)} \end{cases} \tag{21}$$

   yielding the solution vector $\mathbf{x}_k$ and the Lagrange multiplier vector λ_k.

2. Reduce the subset by the inactive constraints

$$\mathcal{A}_R^{(k)} = \mathcal{A}^{(k)} \setminus \left\{\alpha_k \in \mathcal{A}^{(k)} | (\lambda_k) = 0\right\} \tag{22}$$

   where the Lagrange multipliers $\lambda_k = 0$ indicates the inactive constraints in the reference set.

3. Define the entering index $\hat{\alpha}_k$ and add it to the subset

$$\hat{\alpha}_k = \arg \max_{\alpha \in \mathcal{A}} g_\alpha(\mathbf{x}_k), \tag{23}$$

$$\mathcal{A}^{(k+1)} = \mathcal{A}_R^{(k)} \bigcup \{\hat{\alpha}_k\} \tag{24}$$

   and return to Step 1.

A practical stopping criteria is when $|g_{\hat{\alpha}_k}(\mathbf{x}_k)| \leq \varepsilon(\mathbf{x}_k)$, where $\varepsilon(\mathbf{x}_k) > 0$ is a tolerance parameter which may depend on the current solution $\mathbf{x}_k$. If the tolerance is pre-defined, then $\varepsilon(\mathbf{x}_k)$ may be substituted with the scalar ε.

The convergence of the algorithm to the optimal solution is non-trivial to show. A theoretical framework to prove the global convergence of the DNCA algorithm is shown in[14].

5 DESIGN EXAMPLES

In this section, the flexibility of the design method using the DNCA algorithm is illustrated by numerical examples. The design examples corresponds to the problem formulation, design specification and optimization described in Section 1–4. All figures presented in this section are normalized in the frequency domain according to $|W(0)| = 1$.

5.1 Enhanced MATLAB Flattop

This example considers the design of an enhanced version of MATLAB flattop window (MF). The MF window is actually the flattop window Brüel & Kjær have implemented in their Dual Channel Signal Analyzers Type 2032 and 2034 and is generated from five ($K = 5$) window coefficients, see table 1. The objective in this example is to design an Enhanced-MATLAB flattop (E-MF) window with higher stopband attenuation than the original MF window while preserving the desired passband properties. To be able to do the enhancement, a design specification has to be defined.

The design specification used in this example is derived as follows; First define the passband edge as the angular frequency ω_p where,

$$W_{MF}(\omega_p) = 2 - \max_{\omega \in [0,\pi]} |W_{MF}(\omega)| \tag{25}$$

Then, define the allowed amplitude deviation in the passband $\sigma_p(\omega)$ as

$$\sigma_p(\omega) = \begin{cases} 0 & \omega = 0 \\ \max\limits_{\omega \in [0,\pi]} |W_{MF}(\omega)| - 1 & \omega \in]0, \omega_p] \end{cases} \tag{26}$$

which ensures that $W_{MF}(0) = 1$. The stopband edge ω_s is defined as the frequency on the mainlobe where the amplitude is equal to the peak sidelobe level. Given $\sigma_p(\omega)$, ω_p and ω_s, the E-MF window coefficients is obtained by solving Eq.(18) using the optimization method described in sections 4. The obtained E-MF window coefficients are listed in Table 1. Figure 1 shows the frequency response of the original MF window and the E-MF window for $\omega \in [0, \pi]$. As seen, the E-MF window have much lower sidelobes compared with the original MF window. Figure 2, compares the behavior of the MF window and the E-MF window in the defined passband and around the stopband edge. The E-MF window is slightly flatter in the passband compared with the original MF window (not visible in figure). The peak sidelobe level for the MF window is approximately -69dB while the E-MF window peak sidelobe is -84.5dB, which implies an improvement of ~ 15.5dB.

	MF	E-MF	ISO	E-ISO	HA-150dB
a_0	1	1	1	1	1
a_1	1.93	1.93008089169324	1.933	1.932947271643817	1.96353981283797
a_2	1.29	1.29395556869826	1.286	1.283398773617144	1.58091907696028
a_3	0.388	0.39510455056118	0.388	0.381224493749515	0.82673271845542
a_4	0.032	0.03131265625958	0.0322	0.029314745653344	0.24058413387128
a_5	–	–	–	–	0.03261529646652
a_6	–	–	–	–	0.03262152875336
a_7	–	–	–	–	0.00138976869451
a_8	–	–	–	–	0.00000308667600

Table 1. Normalized window coefficients.

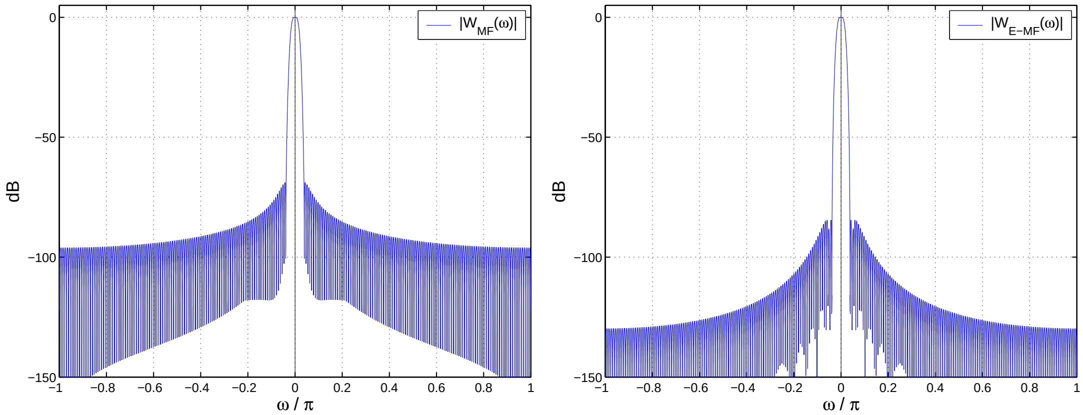

Fig. 1. Frequency domain representation of the MATLAB flattop window (left) $W_{MF}(\omega)$ and the Enhanced-MATLAB flattop window $W_{E-MF}(\omega)$ (right) for length $N = 256$.

5.2 High Accuracy window design

With the proposed design method it is possible to design specialized windows, such as high accuracy windows which are needed in today's 24bit equivalent 1bit-Sigma-Delta ADC:s, where high frequency noise is generated. To suppress the high frequency noise, a window with more than 150dB sidelobe attenuation is required. When such high attenuation is required, the Kaiser window is normally the used. However, by using the proposed method, it is possible to design high accuracy windows by using the technique of shifted Dirichlet kernels. In this design example, the windows is constructed with $K = 8$ coefficients. The window has much lower passband ripple than most of the commonly used flattop windows (~ 0.0002 dB), the mainlobe is wider and the peak sidelobe level is below 150dB, see Fig. 3 and Fig. 4. The High Accuracy (HA) window coefficients are listed in Table 1.

5.3 Enhanced ISO Flattop Window

In year 2000, the International Organization for Standardization (ISO) initiated the work of ISO 18431-2, *Mechanical Vibration and Shock - Signal Processing Part 2 - Rectangular, Hanning, and Flattop Windows for Fourier Transform Analysis*. The goal of the project is to standardize the three most commonly used window functions for frequency analysis. This example aims to design an Enhanced ISO (E-ISO) flattop window, in terms of higher sidelobe attenuation. The same design strategy as used for enhancing the MF window have been applied to the upcoming ISO 18431-2 flattop window. The original ISO window and the enhanced E-ISO window coefficients are listed in Table 1. Note that the coefficients for the ISO window are very similar to the MF window coefficients. The main differences between these windows are that the ISO window has a wider, less flat passband and higher stopband attenuation than the MF window.

The peak sidelobe level for the ISO window is approximately -84dB while the E-ISO window peak sidelobe is -89dB, which implies an improvement of ~ 15.5dB. The new E-ISO window coefficients are listed in Table 1.

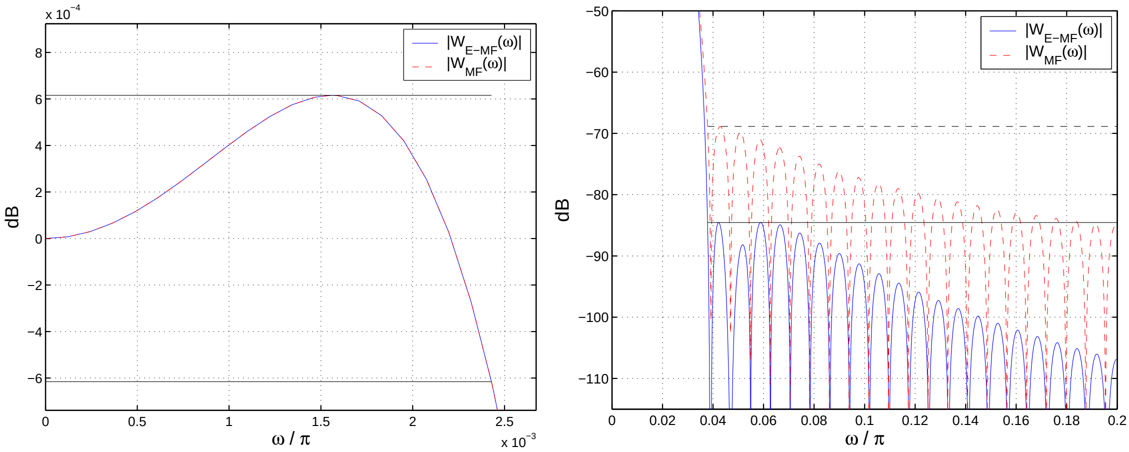

Fig. 2. Frequency domain representation of the E-MF window $|W_{E-MF}(\omega)|$ and the MF window $|W_{MF}(\omega)|$ for length $N = 256$ in the passband (left) and around the stopband edge (right).

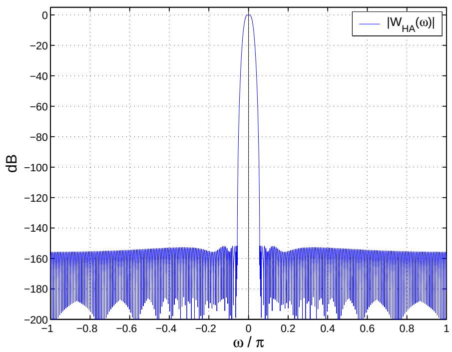

Fig. 3. Frequency domain representation of the HA window $|W_{HA}(\omega)|$ for length $N = 256$.

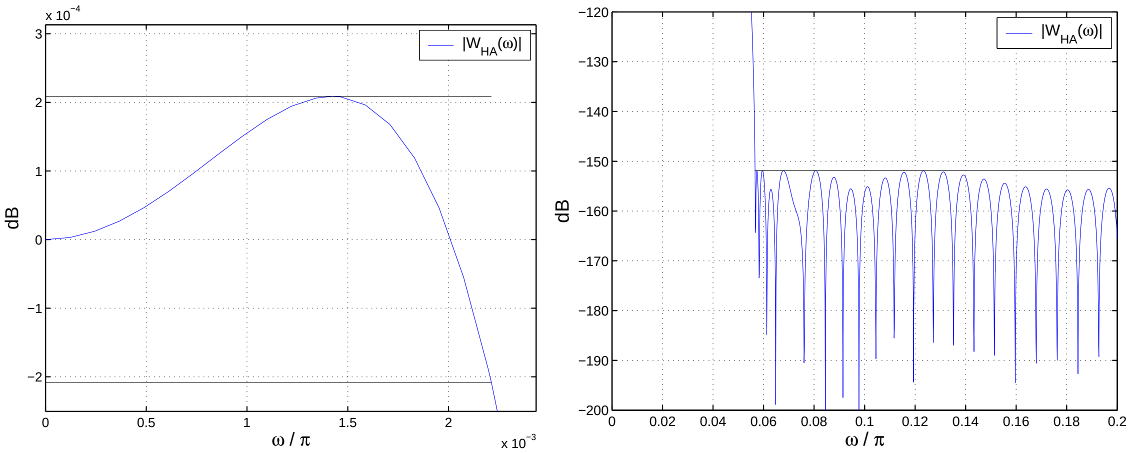

Fig. 4. Frequency domain representation of the HA-150dB window $|W_{HA}(\omega)|$ for length $N = 256$ in the passband (left) and around the stopband edge.

6 SUMMARY AND CONCLUSION

A new and versatile window design method using the Dual Nested Complex Approximation (DNCA) optimization algorithm is presented. One design example illustrates the capability of the design method in terms of enhancing existing windows like the MATLAB flattop window. Results shows that the peak sidelobe level for the MATLAB flattop window can be lowered from -69dB to -85.5dB without degrading the passband and mainlobe properties. The same strategy can be applied to enhance all existing flattop windows, such as the Hewlett Packard flattop window series P-301 and P-401. Another design example illustrates the flexibility by designing a specialized windows with more than 150dB sidelobe attenuation. During the optimization process, constraints such as lower passband ripple, null response at certain frequencies, etc. can be taken under consideration.

7 REFERENCES

[1] Claesson, I., Lagö, T. (2004), "A historical perspective of the FFT algorithm", *Proc. of the XXII International Modal Analysis Conference, IMAC-XXII, Dearborn, MI USA, January.*

[2] Cooley, J.W., Tukey, J.W. (1965), "An algorithm for the machine calculation of the complex Fourier series", Mathematics of Computation, Vol. 19, pp. 297-301.

[3] Oppenheim, A.V., Schafer, R.W. and Buck, J.R. (1998), *Discrete-time signal processing, second edition*, Prentice Hall, New Jersey, ISBN 0-13-754920-2.

[4] Harris, F.J. (1978), "On the use of windows for harmonic analysis with the discrete Fourier transform", Proc. of the IEEE, Vol. 66, pp. 51-83.

[5] Nuttall, A.H. (1981), "Some windows with very good sidelobe behavior", IEEE Trans. on Acoustics, Speech, and Signal Processing, Vol. 29, no. 1, pp. 84-91.

[6] Geckinli, N.C., Yavuz, D. (1978), "Some novel windows and a Concise tutorial comparison of window families", IEEE Trans. on Acoustics, Speech, and Signal Processing, Vol. 26, no. 6, pp. 501-507.

[7] Wahlberg, B. (1991), "System identification using Laguerre models", IEEE Trans. on Automatic Control, Vol. 36, no. 5, pp. 551-562.

[8] Van Veen, B.D., Buckley, K.M. (1988), "Beamforming: a versatile approach to spatial filtering", IEEE ASSP Magazine, Vol. 5, pp. 4-24.

[9] Nordebo, S., Claesson, I., and Nordholm, S. (1994), "Weighted chebyshev approximation for the design of broadband beamformers using quadratic programming", IEEE Signal Processing Letters, Vol. 1, no. 7, pp. 103-105.

[10] Nordholm, S., Rehbock, V., Teo, K.L., and Nordebo, S. (1998), "Chebyshev optimization for the design of broadband beamformers in the near field", IEEE Trans. on Circuits and Systems II, Vol. 45, no. 1, pp. 141-144.

[11] Tran, T., Dahl, M., and Claesson, I. (2003), "A semi-infinite quadratic programming algorithm with applications to channel equalization", *Proc. of the 7th International Symposium on Signal Processing and its Applications, ISSPA2003, Paris France, July*, Vol. 1, pp. 653-656.

[12] Tran, T., Dahl, M., Claesson, I., and Lagö T. (2004), "High accuracy windows for todays 24bit ADCs", *Proc. of the XXII International Modal Analysis Conference, IMAC-XXII, Dearborn, MI, USA, January.*

[13] Dahl, M., Claesson, I., and Nordebo, S. (2003), "Antenna array design using dual nested complex approximation", *Proc. of the 7th International Symposium on DSP for Communication Systems, DSPCS2003, Coolangatta, Australia, December.*

[14] Nordebo, S., Dahl, M., and Claesson, I. (2001), "Complex approximation with applications to antenna array pattern synthesis", *Proc. of Electromagnetic Computations EMB01, Uppsala, Sweden, November.*