PERFECT AND ALMOST PERFECT PULSE COMPRESSION CODES FOR RANGE SPREAD RADAR TARGETS

MARKKU LEHTINEN
Sodankylä Geophysical Observatory, University of Oulu
Tähteläntie 62, FIN-99600 Sodankylä, Finland

BAYLIE DAMTIE
Wasbera Geospace and Radar Science Laboratory
Bahir Dar University, P.O.Box, 79, Bahir Dar, Gojjam, Ethiopia

PETTERI PIIRONEN
Department of Mathematics and Statistics
University of Helsinki, FI-00014, Helsinki, Finland

MIKKO ORISPÄÄ
Sodankylä Geophysical Observatory, University of Oulu
Tähteläntie 62, FIN-99600 Sodankylä, Finland

(Communicated by Margaret Cheney)

Abstract. It is well known that a matched filter gives the maximum possible output signal-to-noise ratio (SNR) when the input is a scattering signal from a point like radar target in the presence of white noise. However, a matched filter produces unwanted sidelobes that can mask vital information. Several researchers have presented various methods of dealing with this problem. They have employed different kinds of less optimal filters in terms of the output SNR from a point-like target than that of the matched filter. In this paper we present a method of designing codes, called perfect and almost perfect pulse compression codes, that do not create unwanted sidelobes when they are convolved with the corresponding matched filter. We present a method of deriving these types of codes from any binary phase radar codes that do not contain zeros in the frequency domain. Also, we introduce a heuristic algorithm that can be used to design almost perfect codes, which are more suitable for practical implementation in a radar system. The method is demonstrated by deriving some perfect and almost perfect pulse compression codes from some binary codes. A rigorous method of comparing the performance of almost perfect codes (truncated) with that of perfect codes is presented.

1. Introduction. A radar system has emerged as one of the most powerful remote sensing instruments with numerous applications. Typical applications include air traffic control, weather monitoring, military surveillance, land mine detection, probing extraterrestrial objects and et cetera. The development of these diverse applications of a radar system has been accomplished in relatively a short span of time since the system was used mainly for the purpose of ranging and detecting very limited types of targets. The performance of a radar system has been continually enhanced by improving its hardware, the transmission waveform, the impulse response of the receiver filter and the associated data analysis methods. The advent

2000 Mathematics Subject Classification. Primary: 58F15, 58F17; Secondary: 53C35.
Key words and phrases. Radar waveform, comparison of experiments, Ambiguity function.
of modern computer and development in digital circuit technology provide unprecedented opportunities to implement different performance optimization techniques. The performance of different radar transmission waveforms, receiving systems and the associated data analysis methods can be easily evaluated.

A radio transmission from a radar system can be continuous or pulsed. Our investigations focus on a radar system that works by transmitting a short pulse of radio wave at specific intervals of time. This kind of radar system very often operates by transmitting a modulated radio wave into a target of interest and receiving the scattered signal. A considerable amount of research effort has been devoted to a radar waveform optimization problem of such kind of a radar system following the widely cited work of [22]. The majority of these investigations have focused on optimizing the transmitter waveform in conjugation with the associated filter in the receiving system [15].

In a radar remote sensing one usually wants to observe a target located over a long distance from the transmitter and also at the same time to be able to differentiate closely located targets within the observational volume. Long distance observations demand very high average radar power, which cannot be increased indefinitely in practice. This means one needs to use a long radar waveform in order to increase the average power of the radar. However, a short pulse is required in order to obtain a very high measurement resolution and thereby enabling the radar system to discern closely placed targets. This is a well known contradicting requirement in a radar system and it has been solved by means of pulse compression techniques [9].

Barker codes [1] are among the most widely used radar transmission waveforms in a pulse compression radar system and normally a matched filter is used in the analysis to enhance the output SNR. However, a matched filter creates unwanted sidelobes. These sidelobes can mask small targets which are illuminated simultaneously with large scatterers. Methods of suppressing these unwanted sidelobes has been developed e.g. [8]; but this method has an associated degradation in the output SNR. One remedy for the signal degradation problem was to carry out an exhaustive search for binary phase code and filter pairs that yield minimum possible SNR losses and these optimal pairs have been found by the present authors [10]. A similar investigation has been conducted for quadrature codes [3]. So far most of the investigations on optimizing the radar performance in the case of matched filtering seem to concentrate on designing different kinds of filters that can handle the limitations of a matched filter.

In this paper we present a method of designing optimal transmission radar waveforms that can be analyzed by means of a matched filter without creating unwanted sidelobes. In other words instead of looking for an appropriate mismatched filter, which we know from theory cannot yield the maximum possible output SNR attainable by a matched filter, we are searching for a radar transmission waveform with varying amplitude that can be decoded exactly by means of a matched filter. In practice such a waveform is derived from a given binary code by employing the discrete Fourier transform.

2. Radar waveform performance for the case of matched filtering. The performance of a radar system largely depends on the characteristics of the transmitted radar waveform. In general the characteristics of a radar waveform are usually quantified by means of the ambiguity function [12, 16]. There are different ways of presenting the ambiguity function depending on the information sought
from the radar and also the nature of the measurement scenario. For example, the theory of ambiguity functions, which is suitable for handling the analysis of an incoherent scatter radar measurement, has been formulated in [11] and the presentation is not receiver filter specific. Mathematical expressions for a more general atmospheric radar ambiguity function have been derived by [21]. The performance of a radar waveform in the case of matched filtering is usually quantified in terms of the matched-filter output as a function of the time delay (range) and frequency. The squared magnitude of this output is normally referred as the ambiguity function among the engineering community and we shall use this function in this study.

The analysis of a radar waveform performance by means of the ambiguity function is often simplified by considering the carrier frequency and the true time delay to be zero. This means that for the case of high SNR, the output of the matched filter denoted here by \( \chi(\tau, f) \) can be expressed as [17]

\[
\chi(\tau, f_D) = \int_{-\infty}^{\infty} \text{env}(t)\text{env}(t+\tau)e^{j2\pi f_D t}dt,
\]

Figure 1. Two-dimensional ambiguity function (top and middle panels) and the zero Doppler ambiguity function (bottom panel) of the 5-bit uncoded radar waveform with a 13 unit oversampling.
where \( \text{env}(t) \) denotes the radar transmission waveform, \( \tau \) and \( f_D \) indicate the time delay and frequency shift and the overhead bar is the complex conjugate. It is easy to see that (1) is basically the response of a point like radar target as a function of range and Doppler shift. In mathematical terms it is simply the correlation of the transmission code with the different frequency shifted version of itself. The relation takes the form of the well known convolution integral equation. The numerical approximation of the integral in (1) can be expressed by

\[
\chi(\tau, f_D) = \lim_{N \to \infty} \frac{1}{N} \sum_{n=1}^{N} \text{env}(n)\overline{\text{env}(n+\tau)}e^{j2\pi f_D n}.
\]

The calculation of the convolution sum in (2) can be carried out in practice using the discrete Fourier transform in order to shorten the calculation time. The corresponding ambiguity function is given by \( |\chi(\tau, f_D)|^2 \) and its characteristics curve can show the pros and cons associated to a given radar waveform. One usually seeks for \( |\chi(\tau, f_D)|^2 \) to have a single peak at the origin where \( \tau \) and \( f_D \) have zero values, which is hard to achieve in practice.

For example, FIGURE 1 displays the normalized ambiguity function (top panel), the corresponding contour view (middle panel) and the zero Doppler ambiguity function (bottom panel) of a 5-bit uncoded radar waveform. The well-known sinc-like function curve along the zero time delay and the triangular curve along the zero Doppler can be seen clearly from the two-dimensional figure of the ambiguity function. The contour in the middle panel shows the spread in time delay (range ambiguity) and also a relatively narrow distribution in frequency of the ambiguity function of the simple pulse radar waveform. A one dimensional cut of the ambiguity function along the zero Doppler, which is displayed in bottom panel, shows how the different time delays are weighted and essentially there is contributions from all the time delays. This means that there is range ambiguity. It is possible to minimize the range ambiguity associated to a simple pulse radar waveform by shortening its length. However, a short pulse creates poor frequency measurement accuracy. Also, the average radar power will be small and thereby limiting the observation space. Modulated radar waveforms have been used to solve this problem and there are several types of modulation techniques including Barker coding.

The radar ambiguity function of the 5-bit Barker code is portrayed in FIGURE 2. The top panel shows the two-dimensional ambiguity function and the middle panel depicts the corresponding contour view. One can see that the ambiguity function has been narrowed along the direction of the time delay and in other words there is now less range ambiguity than the result obtained from the corresponding simple pulse radar waveform discussed earlier. This can be easily seen from the curve in the bottom panel, which is just a cut of the two-dimensional ambiguity function along the zero Doppler. Normally, one considers the main peak as the only output of the matched filter neglecting the sidelobes. However, in some cases the contributions of the sidelobes can be substantial and thus creating bias in the measurement. In the following section we present a method of designing codes that do not create sidelobes at the corresponding matched-filter output.
Perfect and almost perfect

Figure 2. Two-dimensional ambiguity function (top and middle panels) and the zero Doppler ambiguity function (bottom panel) of the 5-bit Barker code with a 13 unit oversampling.

3. **Fourier transforms and their DFT approximation.** The Fourier transform of a signal \( z(t) \) is defined by

\[
\hat{z}(\omega) = \sum_{t=-\infty}^{\infty} e^{-i\omega t} z(t),
\]

and the inverse transform by

\[
z(t) = \frac{1}{2\pi} \int_{0}^{2\pi} e^{i\omega t} \hat{z}(\omega) d\omega.
\]

Numerical approximations of the Fourier transform are remarkably simple to carry out using the well known discrete fast Fourier transform (DFT) algorithm. The transformed data points then represent the values of the interval \([0, 2\pi]\) divided...
into \(N_{f_{\text{ft}}}\) equidistant points \(\omega_p = \frac{2n\pi}{N_{f_{\text{ft}}}}\). The DFT is defined by
\[
\hat{z}_n = \hat{z}(\omega_n) = \sum_{k=0}^{N_{f_{\text{ft}}}-1} z(k)e^{-ik2\pi/n_{f_{\text{ft}}}},
\]
and the corresponding inverse DFT becomes
\[
z(k) = \frac{1}{N_{f_{\text{ft}}}} \sum_{n=0}^{N_{f_{\text{ft}}}-1} \hat{z}_n e^{i2\pi kn/N_{f_{\text{ft}}}}.
\]
The inverse Fourier integral divided by \(2\pi\) is just replaced by the inverse DFT, which is the same as the forward DFT with negative exponent but divided by \(N_{f_{\text{ft}}}\). In the following sections we present different types of radar target models and then methods of constructing different types of perfect and almost perfect pulse compression codes are introduced.

4. A radar target model. A scattered signal from a singly spread radar target is modeled by the Itô integral equation [19]
\[
z(t) = \int_{-\infty}^{\infty} \text{env}(t-r)\mu(dr) + T^{1/2}\xi(t),
\]
where the structure function \(x(r)\) of the measure \(\mu(dr)\), \(\mathbb{E}(\mu(dr)\mu(dr')) = x(r)\delta(r-r')\) is the scattering power of the target that is given as a function of range. The range variable \(r\) is considered as the travel time of the signal to the target point and back. \(\xi(t)\) and \(T\) represent the receiver thermal noise (here it is modeled as white noise) and its power, respectively. In order to simplify our presentation the effects of the radar system parameters and range dependence of the transmitted power are included in the variable \(x(r)\). The high frequency carrier signal is not included in (7) and it is not needed for our purpose.

4.1. Discrete radar signal model (general case). It is clear that only the full continuous model in (7) can describe the scattering process completely in an accurate manner. However, in most practical cases all signals need to be sampled and the transmission envelopes are described by a finite number of subpulses. Hence, a more practical model is required, which can be obtained in the following manner. Let us choose a basic time unit \(\Delta t\) and define the transmission envelope as
\[
\text{env}(t) = \sum_n e_n p(t-n\Delta t) \text{ with } p(t) = \chi_{[0,\Delta t]}(t),
\]
where \(\chi_{[0,\Delta t]}(t)\) is the characteristic function of the interval \([0,\Delta t]\), defined by \(\chi_{[0,\Delta t]}(t) = 0\), if \(t < 0\) or \(t \geq \Delta t\) and \(\chi_{[0,\Delta t]}(t) = 1\), if \(0 \leq t < \Delta t\). Similarly, the discretized signal \(z_n\) can be defined as samples of the convolution \(q \ast z\) at equidistant points
\[
z_n = (q \ast z)(n\Delta t') \text{ with } q(t) = \chi_{[0,\Delta t']}(t)/\Delta t',
\]
In the simple case the discretization interval for the samples \(\Delta t'\) is the same as the transmission subpulse (baud) length \(\Delta t\). However, in order to avoid information loss in the discretization process, it is usually better to have \(\Delta t' < \Delta t\). If the sampling interval is an integer fraction of the baud length of the transmission envelope (\(\Delta t' = \Delta t/N_{f_{\text{rac}}}\)), the sampling method is called fractional sampling [6]. Instead of the characteristic function of the sample intervals, some other forms can be used for
the basic pulse $q(t)$ and the receiver impulse response $p(t)$ without affecting our conclusions.

4.2. The simple discretized model. The simple discretized model is derived by setting $\Delta t' = \Delta t$. Let us define

\begin{equation}
\mu_n = (p * q * \mu)(n \Delta t) = \int (p * q)(t - n \Delta t) \mu(dr).
\end{equation}

Then the measurement equation (7) reduces to the discrete form

\begin{equation}
z_n = \sum_{n'} e_{n'} \mu_{n-n'} + \xi_n (T/\Delta t)^{1/2},
\end{equation}

where $\xi_n$ is the normalized discrete white noise process. Here the values of $\mu_n$ are considered as the scattering amplitude corresponding to different range gates $r_n = n \Delta t$ and are also treated as the unknowns variable in a radar measurement. In fact they are just Gaussian variables with zero mean and what one usually wants to know is their expectation values (i.e. $x_n = E[|\mu_n|^2]$).

We can simplify (11) further by ignoring the factor in front of the white noise error term to obtain

\begin{equation}
z_n = \sum_{n'} e_{n'} \mu_{n-n'} + \xi_n, \quad E(\xi_n \xi_{n'}) = \delta_{nn'}, \quad \sum_{n'} |e_{n'}|^2 = 1.
\end{equation}

Here we have assumed that the total power of the code is equal to that of a simple single pulse with unit power. This is in accordance with our goal of pulse compression and that is finding a method of spreading a unit power in time. This means that we require less peak power than the simple unit pulse with power 1.

5. Radar target estimators.

5.1. Matched filter estimator. We have discussed the basic principle of matched filtering using the radar ambiguity function in Section 2. Here we formulate a matched filter estimator in terms of the simple radar target model of equation (12).

If we assume $x_n = E[|\xi_n|^2] \neq 0$ at only one single range gate $n$, it is well known and also can easily be shown by simple inversion calculations that the matched filter $\hat{e}_n = e_{-n}$ gives us the value $\mu_n$ with the least possible variance caused by the thermal noise term as shown for example in [4]. The notation reflects the fact that flipping the filter coefficients is equivalent to a Fourier transform, followed by conjugation and again followed by the inverse Fourier transform. The matched filter estimator of the signal is thus given by

\begin{equation}
\hat{e} * z = A_e * \mu + \hat{e} * \xi,
\end{equation}

where $A_e$ is the transmission envelope autocorrelation function. At range gate $n$ this equation becomes

\begin{equation}
(\hat{e} * z)_n = \mu_n + (\hat{e} * \xi)_n.
\end{equation}

Here the variance of the noise term is equal to 1. Because of its simplicity and long history the matched filter estimator is widely used despite the fact that the sidelobes (i.e. the values $A_e(n) \neq 0$, for $n \neq 0$) cause bias in the results for all but point targets.
5.2. Adaptive filter estimator. In adaptive filtering techniques the combination of both bias and variance is minimized by considering the contributions of the unwanted sidelobes as additional noise [2]. The filter coefficients are then found by calculating the optimal filter coefficients for each range gate by taking this additional noise into account [2]. One particular problem with this method is that the variance \( x_n = \mathbb{E}[\mu_n]^2 \) is not known at the beginning. Thus, these kinds of analyses usually rely on iteration, the first iteration being usually just the matched filter and then using the results of the initial iteration to get some estimates for the variances \( x_n \) to get optimal filter coefficients for the next iteration and so on. As a result this method seems to remain somewhat heuristic in nature.

5.3. Inverse filter estimator. An inverse filter is based on a very simple application of the Fourier convolution theorem. We define the inverse filter \( \hat{e}^{-1} \) to be the inverse Fourier transform of the reciprocal of the Fourier transform of the transmission envelope denoted here by \( e_n \) and i.e.

\[
\hat{e}^{-1}_k = \frac{1}{2\pi} \int_0^{2\pi} e^{i\omega k} \left( \sum_n e^{-i\omega n} e_n \right)^{-1} d\omega.
\]

It is then clear that the convolution of the inverse filter \( \hat{e}^{-1} \) and the original code \( e_n \) is equal to the unit pulse and that is \( (\hat{e}^{-1} \ast e)_n = \delta_{n,0} \). By convolving the measurement equation in (12) with \( \hat{e}^{-1} \), we obtain

\[
(\hat{e}^{-1} \ast z)_n = \mu_n + (\hat{e}^{-1} \ast \xi)_n.
\]

This means that the values of \( \mu_n \) can be directly read from the convolution outputs. In this solution all values of \( \mu_n \) are independent of each other. However, there is dependence due to the non-white noise; but this is fixed and only dependent on the radar codes used and not on the target. An inverse filter estimator does not contain bias that affects both the matched and adaptive filter estimators [3]. There is one important drawback associated to the inverse filter estimator and that is the transformed noise \( (\hat{e}^{-1} \ast \xi) \) and it’s variance is generally larger than 1. We can state the following theorem:

**Theorem 5.1.** For the variance of the filtered noise with an inverse filter we have

\[
\mathbb{E}[(\hat{e}^{-1} \ast \xi)_n]^2 \geq 1
\]

with equality if and only if \( |\hat{e}(\omega)| = 1 \) \( \forall \omega \in [0, 2\pi) \).

**Proof.** First we note that \( \mathbb{E}[(\hat{e}^{-1} \ast \xi)_n]^2 = \frac{1}{2\pi} \int_0^{2\pi} |\hat{e}^{-1}(\omega)|^2 d\omega \) and then use the Cauchy-Schwartz inequality to see that

\[
\frac{1}{2\pi} \int_0^{2\pi} |\hat{e}^{-1}(\omega)|^2 d\omega \frac{1}{2\pi} \int_0^{2\pi} |\hat{e}(\omega)|^2 d\omega \geq \left( \frac{1}{2\pi} \int_0^{2\pi} |\hat{e}^{-1}(\omega)||\hat{e}(\omega)| d\omega \right)^2.
\]

The inequality follows by noting that \( \hat{e}^{-1}(\omega) \) and \( \hat{e}(\omega) \) are reciprocals making the right side equal to 1 and that the second integral on the left side is also equal to 1. The “only if” part is a consequence of the “only if” part of the Cauchy-Schwartz inequality stating that equality exists only if the two functions \( (|\hat{e}^{-1}(\omega)| \) and \( |\hat{e}(\omega)| \) \) are linearly dependent. Being reciprocals they can only be linearly dependent if they are both equal to 1 (almost) everywhere. \( \square \)

The autocorrelation function of the transmission code \( e_n \) is the inverse Fourier transform of \( |\hat{e}(\omega)|^2 \), which leads us to the corollary:
Corollary 1. There is no performance penalty in terms of increased noise when using inverse filters if and only if the autocorrelation of the code has no sidelobes.

Also, we have the following fact:

Theorem 5.2. The autocorrelation function of a finitely long code (longer than one pulse) cannot have zero sidelobes.

Proof. Consider the lag corresponding to the separation of the first and last non-zero pulses in the code. Since the corresponding autocorrelation value is the simple product of these first and last pulses, it cannot be zero. □

The inverse filter is also called mismatched filter and we have also adopted this term in our earlier publications [3]. However, the term inverse filter is more appropriate, because as a mathematical operation it is the inverse, not just something that is different from a matched filter.

6. Perfect and almost perfect pulse compression codes. Let us assume that we can find codes with desired properties \(|\hat{e}(\omega)| \equiv 1\) or, equivalently \(A_e(n) = \delta_{0,n}\). Then the convolution of our measurement equation with the inverse filter would lead to an equivalent equation where data would be represented by the scattering values \(\mu_n\) directly, with normalized additive white noise and i.e.

\[
(\hat{e}^{-1} \ast z)_n = \mu_n + \xi'_n.
\]

This would be the ideal solution to our pulse compression problem. However, the convolved noise \(\xi'_n\) is not sample-wise the same as the original white normalized noise. But it is just another white normalized noise and thus statistically equivalent to the original noise \(\xi_n\). In this manner we have been able to get rid of the convolution with the code in the measurement equation and the new equation is the same as the measurement equation of a single pulse experiment with unit power and normalized noise. In order to obtain this result we just need to convolve our measurements with the inverse filter, which in this case is also equal to the matched filter. This is because if \(|\hat{e}(\omega)| \equiv 1\) the reciprocal in the Fourier side is the same as complex conjugation, which in turn is the same as time-inverse of the filter coefficients.

The question is then whether such optimal codes can be found in practice. We have proved that they do not exist as finitely long. However, it is actually very simple to construct such (infinitely long) codes and also codes with close enough properties to perfectness that are finitely long.

6.1. Infinitely long perfect codes. Let us consider any known pulse compression code, for example a Barker code \(e_n\) or some of the codes found by the present authors [10] that minimizes the noise associated to the inverse filtering technique. One needs to realize that most probably these kinds of codes (when normalized so that the total power equals one) are already close to becoming perfect codes. This means that we are likely to get \(|\hat{e}_n(\omega)| \approx 1\), where \(\hat{e}_n(\omega)\) is the Fourier transform of the code. We can then define

\[
d_n = \frac{1}{2\pi} \int e^{i\omega n} \frac{\hat{e}_n(\omega)}{|\hat{e}_n(\omega)|} d\omega,
\]

in order to obtain a radar code \(e_n\) with the absolute value of its Fourier transform equals to one and this is basically similar to the code we started with. This code is of course infinitely long, but the subpulses go to zero very rapidly:
Theorem 6.1. Assume \( \hat{e}_n(\omega) \neq 0 \) for \( \omega \in [0, 2\pi) \). The subpulses \( e_n \) of the code defined above approach zero faster than any (negative) power of \( n \) as \( n \to \pm\infty \).

Proof. Let us study the polynomial \( P(c) = \sum_n e_n c^n \) along the unit circle of the complex plane \( c \in \mathbb{C} \). With \( c = e^{i\omega} \), it is actually the Fourier transform \( \hat{e}_n(\omega) \) and being a finite polynomial it is infinitely differentiable and the same is true for the Fourier transform of \( d_n \) because of our assumption. This proves the claim.

One may be concerned with the possibility of having zero values for the Fourier transform of these codes. Let us assume that we pick the finite number of coefficients \( e_n \) in a random manner. From the analysis in [7, 13] it follows immediately that the expected number of zeros on the unit circle is zero but as the degree \( n \) of the polynomial \( P(c) \) approaches infinity then almost all of the zeros (of order \( n - (2/\pi) \ln n \)) approaches the unit circle distributed uniformly in angle except at the real points \( \pm 1 \). The number of zeros concentrated to these points is of order \( (2/\pi) \ln n \). Thus, while the probability of having a zero on the unit circle is zero, this also shows that very good codes with all zeroes far away from the unit circle are very hard to find by chance. In practice it is straightforward to check the absence of zeroes by numerical calculations.

Infinitely long codes cannot be used in practice and hence one needs to truncate them. Truncating the codes is not a problem because of the theorem above. Since their amplitude goes to zero rapidly, any numerical accuracy used in the calculations or the analog-to-digital conversion of the signals is reached very fast. This means that the truncation error will be completely lost within these kinds of errors that exist in any practical system. In this manner the perfect codes are in practice as good as the finite codes. The authors feel that a more significant problem is the practical implementation of this kind of amplitude modulation on the outgoing pulses. Since this modulation is not necessary with most of the traditional methods, many existing radars are not capable of handling it.

6.1.1. Examples of (truncated) infinite perfect codes. We use the 11-bit and 13-bit Barker codes for illustration of the pulse compression methods. FIGURE 3 shows the performance of the 11-bit Barker code using matched and inverse filtering methods. One can see that the autocorrelation function of this code contains sidelobes and this means that matched filtering can create bias. These sidelobes disappear when the analysis is carried out by using the corresponding inverse filter as shown in the convolution result at the bottom of the left panel of this figure. One can see from its formulation in (15) that the inverse filter has theoretically infinitely long impulse response. However, in practice it decays rapidly to zero as shown in the middle of the left panel of FIGURE 3 and thus we can consider its length to be finite for practical purpose.

It is easy to see that codes that are close to perfect codes the values of the square of the modulus of their Fourier transform will be close to 1. We denote the minimum and the maximum value of the square of the modulus of the Fourier transform of a code by \( P_{\min} \) and \( P_{\max} \), respectively. As we shall see these values can be understood as a pair of single-pulse powers, which define an interval for experiment comparison making our code better than a single pulse with power \( P_{\min} \) and worse than one with power \( P_{\max} \). In the case of 11-bit Barker code \( P_{\min} \) and \( P_{\max} \) are, respectively, 0.0909 and 1.314 as shown in the bottom of the left panel of FIGURE 3.
inverse filtering method degrades the output SNR and the noise enhancement is usually quantified using a factor $R$, which is defined as the ratio of SNR at the output of an inverse filter to that of the corresponding matched filter. This means that $1/R$ can be considered as a factor specifying the apparent noise increase associated with inverse filtering and in the case of 11-bit Barker code it has a value of 1.4078 as shown in the top of the right panel of FIGURE 3.

Similarly, the performance of the 13-bit Barker code when different pulse compression methods are employed and other important characteristics of this code is shown in FIGURE 4. One can see that the noise enhancement factor $1/R$ is close to one and thereby making the 13-bit Barker code a suitable code for inverse filtering without substantial SNR losses. The values of $P_{\min}$ and $P_{\max}$ are closer to 1 than the corresponding values obtained from 11-bit Barker code. The value of the Fourier transform at DC is greater than 1. The other curves have similar characteristics to that of the 11-bit Barker code including the sidelobes in its autocorrelation and their suppression by means of an inverse filter.
Almost perfect codes have been derived from 11-bit and 13-bit Barker codes using equation (18). FIGURE 5 shows characteristics of the perfect code calculated from 11-bit Barker code after truncating the resulting code at $10^{-3}$ (top panel) and $2 \times 10^{-5}$ (bottom panel). This means that subpulses with amplitude less than $10^{-3}$ and $2 \times 10^{-5}$, respectively, are left out. One can see that the values of $P_{\text{min}}$ and $P_{\text{max}}$ are now very close to 1. There is no noise enhancement in this case (the noise enhancement factor is 1 within the numerical accuracy of our calculation) and this is because the impulse response of the inverse filter and that of the associated matched filter become almost the same. Similarly, FIGURE 6 shows the behavior of the perfect code derived from 13-bit Barker code by leaving out subpulses whose amplitude is less than $10^{-3}$ (top panel) and $2 \times 10^{-5}$ (bottom panel). In this case as well the values of $P_{\text{min}}$ and $P_{\text{max}}$ are nearly 1 and also the code converges faster to perfect code than the 11-bit Barker code, which implies that the 13-bit Barker code as such is already close to perfect code. It is clear that the convolution of these almost perfect codes with the correspond matched filter gives a single peak without sidelobes.

7. A heuristic algorithm for finding almost perfect codes. We have shown that it is possible to derive a perfect code from almost traditional codes such as binary codes or optimized quadripase codes [3] using equation (18) and truncating the resulting sequence in an appropriate manner. Another method of finding
optimal radar waveforms has been presented in [20]. It is also possible to directly search for almost perfect finite codes using a Heuristic Algorithm for Finding almost Perfect Codes (HAFPC henceforth) we have developed. The HAFPC gives a code with the square of the modulus of its Fourier transform is close to a constant value. Also, a code with some other good properties like, for example, having a constant amplitude for the most of the subpulses in the middle and with a small number of subpulses having small correction amplitude at both ends can be found. In addition very long radar codes can be obtained by employing this algorithm. The calculation procedures in HAFPC may be summarized as follows:

1. Start with an arbitrary binary code with $n$ length such that

$$e_n = (a_1, a_2, \ldots, a_n), \quad a_{n'} \in \{-1, 1\}, \quad n' = 1, 2, \ldots, n.$$  

2. Calculate the FFT of the code $e_n$ using sufficient length $N$. Since the original code is real (consisting only either -1 and 1), the real part of it’s FFT will be even and the imaginary part will be odd. Denote the FFT of the code by $\hat{e}_n$.

3. Scale $\hat{e}_n$ so that $|\hat{e}_n^{(n')}| = 1$, for $n' = 1, 2, \ldots, N$ using

$$\hat{e}_n^{(n')} = \frac{\hat{e}_n^{(n')}}{|\hat{e}_n^{(n')}|}.$$
Figure 6. Perfect code made from the 13-bit Barker code. In the left panel it is truncated by leaving out pulses less than $10^{-3}$ (the largest pulse is equal to 1). In the right panel truncation is made at level $2 \times 10^{-5}$. In both cases the minimum and maximum values ($P_{\text{min}}$ and $P_{\text{max}}$) of the Fourier transform are very close to 1. In both panels the numbers displayed in a vertical direction show the scaled value of the amplitude of the code just below it for easy visualization of the decaying rate.

Note that still after the scaling the real and imaginary parts of $\hat{e}_n$ will be even and odd, respectively.

4. Calculate the inverse FFT of the scaled FFT of the code $\hat{e}_n$ and denote the result by $e_n^{(p)}$. Now $e_n^{(p)}$ is a $N$-vector representing a perfect code and its amplitudes are no longer constants. It typically has about $n$ subpulses which have high amplitude and the rest of them have low amplitude. Choose some threshold amplitude value $A$ and classify the subpulses to high baud HB (placed in the middle of the code) and low baud LB according to the values of their amplitude using the chosen threshold value.

5. Set all the values of HB $a_n'$ to either 1 or $-1$ and i.e.

$$a_n' = \text{sgn}(a_n'),$$

and leave the LB as they are. Denote this new $N$-binary code by $e_n^{(1)}$. In most cases a binary phase code having about $n$ number of HB is obtained.

6. Truncate the $N$ length code obtained in earlier step by including only a suitable number of LB from both sides of the HB and denote this number by $l_b$. It is obvious that the perfectness of the code can be increased by increasing $l_b$. The shortest possible codes are generally desired for practical applications. Next find the first ($n_{\text{min}}'$) and the last ($n_{\text{max}}'$) indices of the HB sequence and then denote the truncated code by $e_n^{(t)}$, which can be described by

$$e_n^{(t)} = (a_{n_{\text{min}}'-l_b}, a_{n_{\text{min}}'-l_b+1}, \ldots, a_{n_{\text{max}}'+l_b}).$$

7. Choose a low amplitude value $A_l$ for the LB and then scale the amplitude of the HB to 1 and the amplitude of LB to $A_l$. 
8. Calculate the FFT of the scaled code obtained in step 8 and then scale the resulting amplitude to 1 and lastly determine its inverse FFT.

9. Repeat the calculation in steps 7, 8 and 9 until the result converges or blows up. This will take anything from couple of hundreds to couple of thousands of iteration cycles. Iterations will either converge to an almost perfect code with a very small noise enhancement factor $\frac{1}{R}$ or it may fail. The performance can be monitored by making use of the resulting noise enhancement factor in each iteration.

HAFPC is completely a heuristic algorithm and it does not always create almost perfect codes. However, it is possible to use this algorithm in a try-and-see manner in order to obtain almost perfect codes, which can be more suitable for practical purpose than the ones obtained using equation (18) directly.

![Almost perfect code derived by means of HAFPC from the binary code \{1, -1, -1\} (top panel) and \{1, 1, 1, -1\} (bottom panel).](image)

**Figure 7.** Almost perfect code derived by means of HAFPC from the binary code \{1, -1, -1\} (top panel) and \{1, 1, 1, -1\} (bottom panel).

### 7.1. Examples of almost perfect codes obtained by using HAFPC.

We have carried out an extensive search for perfect codes by employing HAFPC and different types of almost perfect codes were found. **FIGURE 7** displays two of such almost perfect codes obtained via HAFPC as an example. The vertically displayed numbers in both panels are the scaled values of the amplitude of the code displayed directly beneath it. The convolution of these codes with the corresponding matched filter yields a single peak without sidelobes. Also, one can see that the values of $P_{\text{max}}$ and $P_{\text{min}}$ are nearly one, which is a requirement for perfectness in our formulation. These codes have relatively small number of different amplitude sizes than the ones obtained by using equation (18), which is a highly desirable property for simplifying their practical implementation. One may also enhance this useful property further by manually adjusting the amplitude to the nearest suitable values. **FIGURE 8** shows the codes and their performance obtained by hand tuning the amplitude values of the corresponding codes displayed in **FIGURE 7**. The results show that the performance of these codes are similar to the ones obtained...
Figure 8. Almost perfect code obtained by means of HAFPC and then hand tuning the amplitude to suitable values using the binary code \{1, -1, -1\} (top panel) and from the binary code \{1, 1, 1, -1, 1\} (bottom panel). In both panels the numbers displayed in a vertical direction show the scaled value of the amplitude of the code just below it for easy visualization of the decaying rate.

using 18, which shows that HAFPC in combination with hand tuning can indeed yield almost perfect codes that are suitable for practical implementation.

In FIGURE 9 we show the ambiguity function of the code displayed in the top panel of FIGURE 8 and we see that there is no range spreading effect for zero Doppler (note the width of the main peak is determined by baud length of the code and here we apply oversampling). In other words the unwanted sidelobes that appear in matched filtering as indicated in FIGURE 2, for example, are suppressed in this case as shown in the bottom panel. The contour display in the middle panel shows that this code is sensitive to Doppler shift and this is a problem that also exists in other pulse compression methods. HAFPC can also be used to find very long almost perfect codes, which may be needed in some applications. FIGURE 10 displays two such long codes and their performance. One can see that these codes have very good performance with the values of \(P_{\text{max}}\) and \(P_{\text{min}}\) approaching to 1.

Lists of other almost perfect codes obtained by employing the HAFPC algorithm together with the corresponding performance parameters are presented in Table 1.

8. Comparison of codes. In this section we want to introduce a rigorous way of comparing perfect codes and almost perfect codes to each other. While it has been implied above that there is not much difference between a perfect code with \(|\hat{e}_n(\omega)| = 1\) and an almost perfect code with \(|\hat{e}_n(\omega)| \approx 1\), this remains a vague statement and one may suspect the non-white nature of the filtered noise in the latter case might have some greater significance anyway. To make this more rigorous we state the following theorem:

**Theorem 8.1.** Let us denote the minimum and maximum values of the spectrum of the code by \(P_{\text{min}} = \min_{\omega \in [0, 2\pi]} |\hat{e}_n(\omega)|^2\) and \(P_{\text{max}} = \max_{\omega \in [0, 2\pi]} |\hat{e}_n(\omega)|^2\), respectively. It is possible to add independent simulated noise to the measurement.
Table 1. Almost perfect codes obtained by means of HAFPC

| Length | Amplitude                                                                 | $P_{\text{max}}$ | $P_{\text{min}}$ | $1/R$  |
|--------|----------------------------------------------------------------------------|-------------------|-------------------|--------|
| 7      | 86, -414, 1000, -1000, -1000, -414, -86                                    | 1.0047            | 0.9956            | 1.000010 |
| 9      | -10, 92, -404, 1000, -1000, -404, -92, -10                                 | 1.0006            | 0.9994            | 1.000000 |
| 17     | 35, -48, 0, 294, -361, 177, 1000, -1000, 1000, 1000, -177, -361, -294, 0, 48, 35 | 1.0027            | 0.9982            | 1.000001 |
| 23     | 50, -18, 91, -248, 283, -361, 23, -255, 999, -1000, 999, 999, -999, -1000, 999 | 1.0111            | 0.9884            | 1.000045 |
| 35     | 47, 26, 50, 19, 71, -71, -216, -486, -172, 164, 1000, 922, 882, -916, -882, 922, -1000, 164, 172, -486, 216, -71, -71, 19, -50, 26, -47 | 1.0286            | 0.9686            | 1.000265 |
| 37     | 21, -29, 8, 161, -178, 143, -103, 265, -27, 12, -386, 203, 997, -1000, 998, 1000, 997, -203, -386, -12, -27, -265, -103, -143, -178, -161, 8, 29, 21 | 1.0201            | 0.9772            | 1.000092 |
| 48     | -6, -45, -35, -20, 30, -24, 28, 129, 44, -158, -15, -129, 8, 218, 332, 301, 588, -980, -978, -981, -975, 1000 | 1.0365            | 0.9627            | 1.000107 |
| 66     | -40, 5, -83, 41, -54, 37, -81, 40, -104, 190, -5, 179, -111, 140, -104, 86, 25, 80, -201, -125, -504, 13, -169, 882, -908, 838, -908, 905, 930, 978, 867, -909, 886, 945, 968, 920, -990, -977, 868, -959, -1000, 970, 988, 99, 249, 137, -245, -461, -391, 81, 44, -79, 181, 146, -193, 100, 149, -89, -11, 81, -38, -74, -30, 5, 31, 11 | 1.0392            | 0.9648            | 1.000242 |
| 81     | 16, 17, -11, 31, 53, -59, -15, 60, -98, -48, 47, -36, -6, 36, 129, 92, -178, 127, 47, -371, -9, 124, -252, -30, 304, 102, 170, -282, 812, -804, -868, 756, -830, -778, 751, 807, -820, 983, -880, 819, -1000, 819, -880, -983, -820, -807, 751, 778, -830, -756, -868, 804, 812, 282, 170, -102, 304, 30, -252, -124, -9, 371, 47, -127, -178, -92, 129, -36, -6, 36, 47, 48, -98, -60, -15, 59, 53, -31, -11, -17, 16 | 1.0344            | 0.9624            | 1.000251 |
| 115    | -1, -4, 4, 4, 4, 27, 45, 31, 35, 12, -36, -32, -31, -109, -110, -15, -43, -16, 28, -39, -32, 22, -23, -90, -152, -187, -127, 25, 27, -5, 99, 1, -50, 191, -44, -8, 104, -189, 157, 884, 859, 892, 877, 805, 491, 890, 831, -775, -589, -895, -858, 850, 472, 376, 831, -884, -877, 766, -765, 825, 862, -835, 774, -838, 898, -896, 886, -883, -265, 922, 922, -1000, -864, 863, 930, -871, -290, 77, 98, 155, -59, -44, -181, 122, 91, 30, 36, -173, 17, 116, -4, 27, 12, -148, -55, 214, 112, -279, -53, 150, 55, -48, 19, -39, -71, 83, 91, -92, -83, 43, 77, -1, -44, -23, 18 | 1.0380            | 0.9657            | 1.000141 |

Inverse Problems and Imaging Volume 3, No. 3 (2009), 465–486
values so that the modified measurements are equivalent to measurements using a simple pulse of any power \( P \leq P_{\min} \). Also, it is possible to add independent simulated noise to a single-pulse measurement of power \( P \geq P_{\max} \) so that the modified measurement is equivalent to the radar measurement coded with \( e_n \).

**Proof.** In the first case, consider a Gaussian stationary noise process \( \xi_1 \) independent of all the other processes considered. Adding this to our measurement values gives

\[
(19) \quad z = e \ast \mu + \xi + \xi_1,
\]

which is equivalent to

\[
(20) \quad \sqrt{P\hat{e}^{-1} \ast z} = \sqrt{P}(\mu + \hat{e}^{-1} \ast \xi + \hat{e}^{-1} \ast \xi_1).
\]

The spectrum of the noise \( P(|\hat{\xi}_1(\omega)|^2 + 1)|\hat{e}(\omega)|^{-2} \) is equal to a constant 1, if

\[
(21) \quad |\hat{\xi}_1(\omega)|^2 = P^{-1}|\hat{e}(\omega)|^2 - 1.
\]

This is non-negative and thus specifies a Gaussian stationary process \( \xi_1 \) for any

\[
(22) \quad P \leq \min_{\omega \in [0,2\pi)} |\hat{e}(\omega)|^2.
\]
With this choice the modified measurement equation corresponds to that of a single pulse of power $P$ and normalized additive white noise, which proves the first part of the theorem.

Similarly in the second case, adding a Gaussian stationary noise process $\xi_2$ independent of all the other processes considered to a simple-pulse measurement model, one gets

$$(23) \quad z' = \sqrt{P} \delta_{\omega, 0} \ast \mu + \xi + \xi_2,$$

which is equivalent to

$$(24) \quad \frac{1}{\sqrt{P}} e \ast z' = e \ast \mu + \frac{1}{\sqrt{P}} (e \ast \xi + e \ast \xi_2).$$

The spectrum of the noise $\frac{1}{P} |\epsilon(\omega)|^2 (1 + |\hat{\xi}_2(\omega)|^2)$ is equal to 1, if

$$(25) \quad |\hat{\xi}_2(\omega)|^2 = \frac{P}{|\hat{\epsilon}(\omega)|^2} - 1.$$

With this choice for the value of $\xi_2$, equation 24 corresponds to the radar measurement coded with $\epsilon_n$ and normalized additive white noise and that proves the second part of the theorem. \(\square\)

The theorem above is a very strong statement about comparison of goodness of the different radar experiments. Having the values produced by the better experiment, it directly shows the way of producing (for example by some computer software) modified measurement values, which are distributed in the same way as the measurement values produced by the inferior experiment. Thus, it becomes impossible to make the betterness relationship questionable and it is also independent of any ways of analysis methods possibly applied: just transform the better measurement values first to what is proven possible by the theorem and then obviously any analysis method will produce equivalent results.
Formal theories for experiment comparison have been developed in [14, 18]. Using the notation of [14] (assuming the variances $x_i$ of the scattering factors $\mu_i$ are our unknown variables with any prior) we can write the following betterness relation

\[
P_{\min}^{1/2} \delta_{e,0} \preceq x \preceq P_{\max}^{1/2} \delta_{e,0},
\]

where a single pulse code with unit power is denoted by with $\delta_{e,0}$. The proof is a direct application of our theorem to the definition as presented in [14].

9. **Generalization from the simple discrete model.** We have used the simple discrete model in [17] to introduce the perfect codes. This does not mean that they only work in that case. It is actually easy to see that for example in the fractional sampling model, convolution of the fractional samples with a filter where the inverse filter coefficients $\hat{e}^{-1}$ are spaced $N_{frac}$ samples apart with zeroes in between will reduce our measurements to what would be produced by a single pulse transmission. If the code is perfect this filtering does not change the spectrum of the fractionally sampled signal at all. Thus, everything goes as in the simple sampling case.

In addition, if we consider a continuous (non-sampled) signal and a perfect code defined by $\Delta t$, baud form $q(t)$ and a subpulse coefficients $e_i$, the proper inverse filter for the continuous signal is defined by the sum

\[
\sum_i \hat{e}^{-1} \delta(t - i\Delta t).
\]

Again, convolving our original signal with this does not change the thermal white noise process and changes the response to that of a single pulse of the form $q(t)$, see equation in [9]. Experiment comparison also easily generalizes to both of these cases.

10. **Conclusions.** A radar waveform is one of the key components of a radar system that has attracted the attentions of several scientists and engineers. The inventions of many kinds of radar waveforms and their implementations in a practical radar measurement have not only widened the applications of a radar system but also improved its performance substantially. An excellent review on the development of the most widely known radar waveforms and their implementations has been published by [5]. A radar waveform may be modulated in various manner including frequency and phase modulations. The most widely known radar waveforms includes the linear frequency modulation chirp radar waveform [9] and the binary phase radar pulse [1]. The modulation of chirp waveform is carried out by sweeping the frequency linearly and binary phase coding is done by shifting the phase at an appropriate time intervals. These waveforms create unwanted sidelobes when they are decoded by means of a matched filter in order to maximize the SNR.

In the present paper we have presented a new method of finding infinitely long (but fast decaying) pulse compression codes, which do not create unwanted sidelobes via matched filtering. We have also found a technique to compare the truncated versions of such codes to single pulse measurements of slightly higher and slightly lower powers than the total power of the coded pulse. We have shown that a truncated version can be compared from both above and below with such single pulse experiment with only a very small difference between them, thus making them in every practical sense equivalent to a single pulse experiment. Also, we have presented a heuristic algorithm that can be used to design similar codes with even
Perfect and almost perfect 485

more desired properties (a constant amplitude value for the main subpulses and other amplitude values for small number of subpulses on tail for correction).

The impulse response of the inverse filter for this kind of perfect or almost perfect code is same as that of the matched filter and there is no performance penalty via increased thermal noise in using these codes. Thus these codes may be used in a radar measurement where the sidelobes in matched filtering is a substantial problem. Also, when using these kinds of codes complicated adaptive filter algorithms are not necessary as they do not bring any more accuracy to the results. While our results and motivations have been derived for the case of radar, it is obvious that these kinds of codes may also be useful in dealing with spread spectrum in communications. Our conclusions rely on the time-stationary range spread target model. Thus sidelobes will develop with a moving target as shown in the last figure in very much the same way as the matched filter sidelobes get worse for non-stationary targets. If the Doppler is point-like it is possible to correct it by multiplying the signal with the correct compensating phase terms. However, for Doppler-spread targets the perfect and almost perfect codes cannot be free from sidelobes.

Acknowledgments. This work was supported by the Academy of Finland (application number 213476, Finnish Programme for Centres of Excellence in Research 2006-2011).

REFERENCES

[1] R. H. Barker, Group synchronizing of binary digital systems, in communications Theory, edited by Jackson, 39 (1953), 273–287.
[2] S. D. Blunt and K. Gerlach, Adaptive pulse compression via MMSE estimation, IEEE Trans. Aerospace and Electronic Systems, 42 (2006), 572–584.
[3] B. Damtie, M.S. Lehtinen, M. Orispää and J. Vierinen, Mismatched filtering of aperiodic quadrature codes, IEEE Trans. Inf. Theory, 42 (2008), 1742–1749.
[4] B. Damtie and M. S. Lehtinen, Comparison of the performance of different radar pulse compression techniques in an incoherent scatter radar measurement, Ann. Geophysicae (submitted), 2008.
[5] H. D. Griffiths, L. Vinagre and W. K. Lee, Developments in radar waveform design, Microwaves and Radar, 1998. MIKON ’98., 12th International Conference on, 4 (1998), 56–76.
[6] A. Huuskonen, M. S. Lehtinen and J. Pirttilä, Fractional lags in alternating codes: Improving incoherent scatter measurements by using lag estimates at noninteger multiples of baud length, Radio Sci., 31 (1996), 245–261.
[7] I. Ibragimov and O. Zeitouni, On roots of random polynomials, Trans. Amer. Math. Soc., 349 (1997), 2427–2441.
[8] E. L. Key, E. N. Fowl and R. D. Haggarty, “A Method of Sidelobe Suppression in Phase Coded Pulse Compression Systems,” MIT, Lincoln Laboratory, Report No.Tech Report, 209, 1959.
[9] J. R. Klauder, A. C. Darlington and W. J. Albersheim, The theory and design of chirp radars, Bell Syst. Tech. J., 39 (1960), 745–809.
[10] M. S. Lehtinen, B. Damtie and T. Nygrén, Optimal binary phase codes and sidelobe-free decoding filters with application to incoherent scatter radar, Ann. Geophysicae, 22 (2004), 1623–1632.
[11] M. S. Lehtinen and A. Huuskonen, General incoherent scatter analysis and GUISDAP, J. Atmos. Terr. Phys., 58 (1996), 435–452.
[12] N. Levanon and E. Mozeson, “Radar Signals,” Wiley, NY, 2004
[13] M. Kac, On the average number of real roots of a random algebraic equation, Bull. Amer. Math. Soc., 49 (1943), 314–320.
[14] P. Piiroinen, “Statistical Measurements, Experiments and Applications,” PhD thesis, Helsinki University, Academia Scientiarum Fennica, 2005.
[15] H. Rohling and W. Plagge, Mismatched-filter design for periodical binary phased signals, IEEE Trans. Aerospace and Electron. Syst., 25 (1989), 890–896.
[16] A. I. Sinsky and C. P. Wang, *Standardization of the definition of the radar ambiguity function*, IEEE Trans. Aerospace and Electron. Syst., 10 (1974), 532–533.

[17] M. I. Skolnik, “Introduction to Radar Systems,” 3rd edition, McGraw-Hill, New York, 2001.

[18] E. Torgersen, “Comparison of Statistical Experiments,” Cambridge University Press, Cambridge, 1991.

[19] H. L. V. Trees, “Detection, Estimation, and Modulation Theory,” Part III, John Wiley & Sons, New York, 2001.

[20] J. Vierinen, M. S. Lehtinen, M. Orispää and B. Damtie, *General radar transmission codes that minimize measurement error of a static target*, arXiv:physics/0612040v1, 2006.

[21] R. F. Woodman, *A general statistical instrument theory of atmospheric and ionospheric radars*, J. Geophys. Res, 96 (1991), 7911–7928.

[22] P. M. Woodward, *Theory of radar information*, IEEE Transactions on Information Theory, 1 (1953), 108–113.

Received October 2008; revised February 2009.

E-mail address: Markku.lehtinen@sgo.fi
E-mail address: bayliedamtie@yahoo.com
E-mail address: petteri.piirinen@helsinki.fi
E-mail address: mikko.orispaa@oulu.fi