Optimized Sinus Wave generation with Compressed Sensing for Radar Applications

A. Ens, A. Yousaf, T. Ostertag and L. M. Reindl
Laboratory for Electrical Instrumentation (EMP)
Department of Microsystems Engineering (IMTEK)
University of Freiburg, Germany
Email: alexander.ens@imtek.uni-freiburg.de

***Abstract*—In Radar applications it is important to generate a sinus signal with low phase distortion. High precision oscillators have a low level of efficiency, they are expensive and also require an analog power amplifier with linear characteristic. The sinus generation with pulse width modulation (PWM) is effective and economical, but adds additional noise to the signal.**

When the transmission channel is known, then the signal generation can be optimized to receive a sinus wave with lower noise. We introduce here a new algorithm to determine the coefficients for the PWM and compare the performance of signal generation with linearization method.

***Index Terms*—Compressed Sensing, BMP, PWM, Sinus, Binary, Chirp, Optimization**

I. Introduction

Sinus generation is an important topic in signal generation for different kinds of applications. One important application is Radar. In Radar a short pulse of a sinus wave is transmitted and reflected at the objects. The reflection is received by the Radar device and the distance can be calculated from the time delay of arrival and the velocity by estimating the frequency shift of the echo to the transmitted signal. This is mathematicaly done by correlation with a reference sinus signal. The signal-to-noise-ratio (SNR) for the parameter estimation increases when the received echo has nearly a perfect sinus form.

For low frequency radar, like ultrasonic distance mesaurement, to measure distances the signal is often generated by the microcontroller. The efficiency of PWM is better than an oscillator and a digital switch is needed as power amplifier and a low pass filter. This system is effective and economical. Furtermore, different signals can be generated, such as chirp signals or constant frequency signals.

The sampling frequency of the PWM generator is usually higher, than the sampling rate of the analog-to-digital converter (ADC). For one signal block the PWM vector consists of N elements and the measurement vector from the ADC of M. Consequently, $N > M$ and the unknown PWM vector is underdetermined. A common way to generate the PWM values is to generate the reference signal in the PWM domain and take a threshold on it (e.g. the sign function for bias free sinus).

For precise distance estimation, the phase of the received echo has also to be analyzed and the transmission channel. With the knowledge of the channel, the transmitted signal can be calculated from the reference signal and the channel to receive the expected signal [1]. But the result is a signal with real values and high amplitudes. An optimization for binary values result in a combinatorial problem, which is NP-complete. For short vectors, the calculation can be done in short time, but for large vectors, the computational complexity increases exponentially. Therefore, an approximation of the problem by the ℓ_1-norm reduces the computational complexity to polynomial order.

The rest of the paper is organized as follows: Section II presents the introduction of Compressed Sensing. In Section III we define the binary optimization problem for PWM generation. In Section IV we introduce a novel algorithm for binary optimization. Numerical simulation results and comparison wtih other optimization algorithms are discussed in Section V. The conlusions are discussed in Section VI.

II. Compressed Sensing

Important research to estimate frequencies in underdetermined systems was independently done by Candes [2] and Donoho [3] in 2006. The problem was to solve the equation $\boldsymbol{y} = \boldsymbol{Ax}$, where the unknown vector $\boldsymbol{x} \in \mathbb{R}^N$ have more elements than the measured values $\boldsymbol{y} \in \mathbb{R}^M$, $N > M$ and the unknown vector is sparse. A sparse vector has only $\|\boldsymbol{x}\|_{\ell_0} = K$, whereby $K < M$, elements that are unequal zero. Then the optimization problem is

$$\arg\min_{\boldsymbol{x}} \|\boldsymbol{x}\|_{\ell_0}, \text{ s.t.}\, \boldsymbol{y} = \boldsymbol{Ax} \tag{1}$$

To approximate problem (1) for sparse vector $\boldsymbol{x}$ using constraint linear optimization can be represented as [3][2]:

$$\arg\min_{\boldsymbol{x}} \|\boldsymbol{x}\|_{\ell_1}, \text{ s.t.}\, \boldsymbol{y} = \boldsymbol{Ax} \tag{2}$$

Then a correct solution can be calculated with very high probability by solving the approximation (2).

Instead of performing a combinatorial optimization (1) which is NP-hard, a linear optimization problem (2) with polynomial order of complexity is calculated.

III. Problem formulation

To generate a sinus signal, a bipolar signal with $x^* \in \{-1, 1\}$ generates first a rectangle signal form. The rectangle in the time domain is a $\frac{\sin x}{x}$ in the frequency domain. In the second step the rectangle signal form is filtered by a band pass filter. In our environment the bandpass filter is a piezo electrical ultrasonic transmitter. The microcontroller generates a PWM with binary unipolar values of $\{0, 1\}$ which transformed to bipolar values $\{-1, 1\}$ by inverting the signal.

The reference signal $\boldsymbol{y}_{\mathrm{Ref}} \in \mathbb{R}^M$ is usually a sinus or a chirp signal. So the mathematical description of the signal generation is

$$\boldsymbol{y}_{\mathrm{Ref}} = \boldsymbol{\Phi}\boldsymbol{\Psi}\boldsymbol{x}^*$$

where $\boldsymbol{x}^* \in \mathbb{P}^N$, $\mathbb{P} = \{-1, 1\}$ is the bipolar signal. The vector elements can be transformed to a binary set $\mathbb{B} = \{0, 1\}$ for the PWM generation vector by

$$\boldsymbol{x} = 2\boldsymbol{x}^* - 1_N$$

Then with binary signal generation it follows

$$\begin{aligned} 2\boldsymbol{y}_{\mathrm{Ref}} - \mathbf{1}_N &= \boldsymbol{\Phi}\boldsymbol{\Psi}\boldsymbol{x} \\ \boldsymbol{y} &= \boldsymbol{\Phi}\boldsymbol{\Psi}\boldsymbol{x} \end{aligned}$$

with the dimensions of the vector $\boldsymbol{x} \in \mathbb{B}^N$ and the measurement vector $\boldsymbol{y} \in \mathbb{R}^M$ with the sensing or selection matrix

$$\boldsymbol{\Phi} \in \mathbb{R}^{M \times N}$$

and the channel matrix

$$\boldsymbol{\Psi} \in \mathbb{R}^{N \times N}$$

The measurement vector is rewritten as

$$\boldsymbol{y} = 2\boldsymbol{y}_{\mathrm{Ref}} - \mathbf{1}_M$$

and the compression can be expressed as

$$\boldsymbol{A} = \boldsymbol{\Phi}\boldsymbol{\Psi}, \qquad \in \mathbb{R}^{M \times N}$$

The selection matrix selects the rows of the channel matrix for the measurement. This can be constructed by generating a diagonal matrix and omitting the rows without a one:

$$\boldsymbol{\Phi} = \mathrm{diag}\begin{pmatrix} 1 & 0 & \cdots & 1 & 0 \end{pmatrix}$$

The generation of the values for the bipolar signal are usually constructed by generating a reference signal $\boldsymbol{x}_{\mathrm{Ref}} \in \mathbb{R}^N$ in the PWM domain with the sample rate of the PWM generator

$$\boldsymbol{x}^*_{\mathrm{Ref}} = 2\mathrm{sign}\left(\boldsymbol{x}_{\mathrm{Ref}}\right) - \mathbf{1}_N$$

The PWM signal $\boldsymbol{x}$ include about $K \approx N/2$ ones and zeros. Therefore the PWM vector is sparse subsequently the following optimization problem can be solved:

$$\arg\min_{\boldsymbol{x}} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_{\ell_2} \qquad \text{s.t.}\, x_n \in \{0, 1\}$$

with the unrecoverable error $\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_{\ell_2} = \epsilon$. This is a combinatorial problem and can be solved in deterministic time. The complexity is exponentiel $\mathcal{O}\left(2^N\right)$ to the amount of PWM values. For large vectors with big values of N the computational complexity is to high to compute the solution in affordable time. Therefore a linear approximation of the result by CS algorithms decrease the computation time to polynomial complexity. The half PWM signal is usually with ones $\|\boldsymbol{x}\|_{\ell_0} = N/2$. Than the optimization problem is:

$$\|\boldsymbol{x}\|_{\ell_0} = N/2 \qquad \text{s.t}\, \boldsymbol{y} = \boldsymbol{A}\boldsymbol{x},\; x_n \in [0, 1]$$

With the theory of CS for sparse vector $\boldsymbol{x}$ and low compression, the problem can be relaxed to ℓ_1 minimization with high probability of correct result. Therefore the combinatorial minimization problem is relaxed to:

$$\arg\min_{\boldsymbol{x}} \|\boldsymbol{x}\|_{\ell_1} \qquad \text{s.t.}\, \boldsymbol{y} = \boldsymbol{A}\boldsymbol{x},\; x_n \in [0, 1]$$

A threshold analysis of the ℓ_1 minimization as shown in [4] states that it is possible to recover the signal for $\frac{M}{N} = 2$.

In this presented work, we can not reconstruct the signal without an error, so the optimization is adopted to

$$\arg\min_{\boldsymbol{x}} \|\boldsymbol{x}\|_{\ell_1} \qquad \text{s.t.}\; \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_{\ell_2} \le \epsilon,\; x_n \in [0, 1]$$

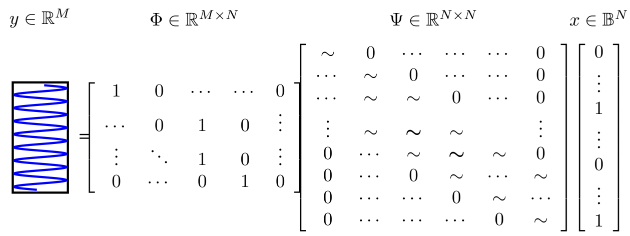

Figure 1. Block diagram of the PWM optimization problem

IV. Algorithms

For our optimization we use greedy algoritms that improve the result in each iteration. Therefore we analyze the Matching Pursuit (MP) algorithm from [5] Mallat and the extended version Orthogonal Matching Pursuit (OMP) from [6] Pati. Further the Compressed Sampling Matching Pursuit (CoSaMP) from [7] from Needell is observed.

The applied algorithms are the Orthogonal Matching Pursuit (OMP) and the Constraint BY Linearization Algorithm (COBYLA) for constraint optimization from Powell [8]. The CoSaMP algorithm is not applicable to our problem because it usese $2K$ Elements for optimization and for PWM there are about $K = N/2$ values that are not zero and therefore all elements have to be used for calculation. The OMP algorithm uses the pseudo inverse to estimate the amplitudes of the data for the used idices. The Matching Pursuit algorithm adds at every iteration a new index to the unknown vector.

Our algorithm is based on the MP algorithm by setting the highest value to one and therefore it is named Binary Matching Pursuit (BMP). In the next iteration the value for the determined indices is set to infinity or to a small value, that the maximum search discard this value and find a new index with a maximum value. The algorithm add at every iteration an indice and after K iterations the algorithm finishes the for loop. A detail functional description is shown in algorithm 1. The complexity of the algorithm increases qudratic to the

Algorithmus 1 Binary Matching Pursuit (BMP) algorithm

Require: sensing matrix $\boldsymbol{A}$, measerement vector $\boldsymbol{y}$, coefficients count K

$\hat{\boldsymbol{x}} \leftarrow \mathbf{0}$, $\boldsymbol{r}_0 \leftarrow \boldsymbol{y}$, $\Lambda_0 \leftarrow \emptyset$

for $i \leftarrow 1; i \leftarrow i+1$ until $i > K$ **do**

  $\boldsymbol{g} \leftarrow \boldsymbol{A}^T \boldsymbol{r}_{i-1}$ ▷ Estimate signal from the residue

  $\boldsymbol{g}_{\Lambda_{i-1}} \leftarrow -\infty$ ▷ Disable the estimated indices

  $\lambda \leftarrow \operatorname{supp}\left(H_1\left(\boldsymbol{g}\right)\right)$ ▷ Find index of greatest value

  $\Lambda_i \leftarrow \Lambda_{i-1} \cup \lambda$ ▷ Add the new index to the set

  $\hat{\boldsymbol{x}}_i|_{\Lambda_i} \leftarrow 1$ ▷ Set the values to one

  $\hat{\boldsymbol{x}}_i|_{\Lambda_i^c} \leftarrow 0$ ▷ All other values to zero

  $\boldsymbol{r}_i \leftarrow \boldsymbol{y} - \boldsymbol{A}\hat{\boldsymbol{x}}_i$ ▷ Calculate the residue

end for

Ensure: Coefficient vector $\hat{\boldsymbol{x}}$

number of unknowns in $\boldsymbol{x}$, $\mathcal{O}\left(MN^2\right)$. The BMP algorithm minimizes the problem formulation

$$\arg\min_{\boldsymbol{x}} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_{\ell_2} \qquad \text{s.t. } \|\boldsymbol{x}\|_{\ell_0} = K,\ x_n \in \{0,\,1\}$$

For better comparison we use the Constraint BY Linearization Algorithm (COBYLA) to calculate the optimal PWM values. This algorithm minimizes constraint optimization problems with $x \geq 0$. The constraint are adjusted, that the algorithm minimizes the following problem formulation

$$\arg\min_{\boldsymbol{x}} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_{\ell_2} \qquad \text{s.t.}\, 0 \leq x_n \leq 1$$

To calculate the PWM values a threshold with 0.5 is applied to the optimized values

$$x_n = \begin{cases} 0 & x_n < 0.5 \\ 1 & x_n \geq 0.5 \end{cases}$$

The initial values for COBYLA are the reference PWM values by generating from a reference sinus wave, multiplied by the channel. To avoid additional error by wrong phase, the phase of the reference signal is adjusted to get the minimum error ϵ.

V. Simulation

Numeric simulation are performed for a channel that consist of a ultrasonic transmitter and receiver with the resonance frequency $f_{\text{Res}} \approx 40$ kHz. This channel has been measured for a distance of 1 m using a vector analyzer. The metric for comparison between the different PWM generation methods is the unrecoverable error ϵ. This is calculated for every value and method. The reference PWM signal is generated by generating different sinus singals with different phase shifts, multiplied with the channel and search the minimum error ϵ at different phases. The reference signal $\boldsymbol{y}$ is a sinus signal with 40 kHz. The break condition for the BMP is the iteration count. COBYLA stops when the reduction of the error is smaller than $1 \cdot 10^{-7}$.

The first simulation in figure 2 shows the error ϵ for different pulse length for an ADC sampling frequency of 100 kHz and a PWM sampling frequency of 200 kHz. The graphs shows, that the optimization algorithms outperform primitve PWM generation for higher time durations. For the primitive PWM generation and the COBYLA optimizad PWM there is a linear up trend. Whereby the BMP optimized PWM converged to a constant error.

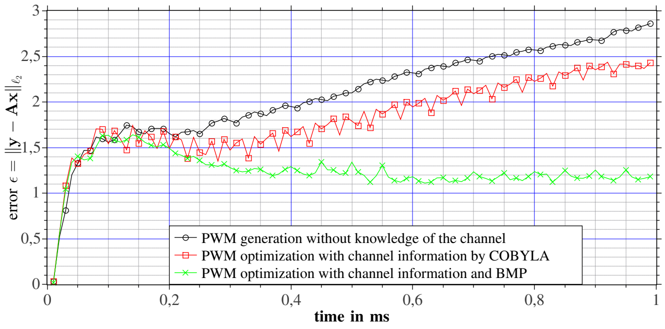

Figure 2. Error ϵ for different time length of the PWM pulse for $N/M = 2$

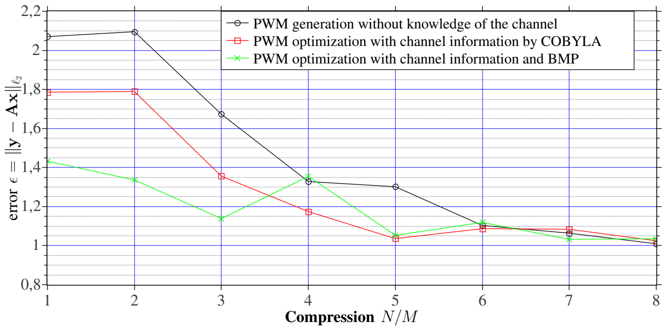

Figure 3. Error ϵ for different compression factors with ADC sampling frequency 100 kHz

A further simulation shows in figure 3 the error ϵ for different compression factors N/M at an constant ADC sampling frequency of 100 kHz and a pulse length of 0.5 ms. For small compression factors and therefore for low PWM frequencies the optimization generates better performance than primitive PWM generation. For larger compression the error between the methods gets smaller and the difference is not significant.

To see the difference of the generated signals the difference of the optimized vector $\boldsymbol{x}$ is calculated

$$\boldsymbol{v} = \left|\operatorname{diff}\left(\boldsymbol{x}\right)\right|$$

Then a set $\Omega_{\text{Ones}} = \left\{\operatorname{supp}\left(\boldsymbol{v}\right)|v_i = 1\right\}$ contains the indices where $v = 1$ and this set is formed into a sorted vector w, the difference is taken and the step is the cumulative sum

$$s_p = \sum_{j=0}^{p} \operatorname{diff}\left(w\right)_j$$

with the step iteration p. The results for the three methods are shown in figure 4. The primitive PWM has always the same pulse width between the steps. Whereby the BMP optimized PWM differ in the pulse width between the steps. This is the reason for smaller error ϵ.

To see the enhancement of the signals at the receiver, another simulation has been done with an ADC sampling

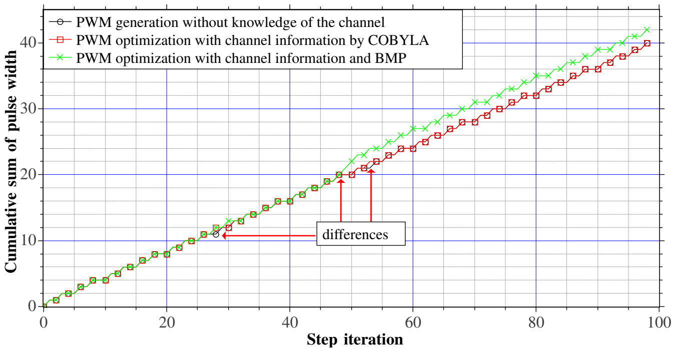

Figure 4. Step difference of the PWM width for primitive and optimized PWM

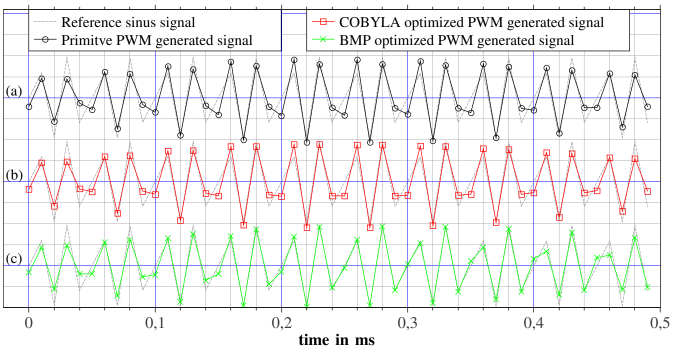

Figure 5. Sinus signal generated by PWM and multiplied with the channel for different methods. (a) primitive PWM generation, (b) optimized PWM generation by COBYLA, (c) optimized PWM generation by BMP

frequency of 100 kHz and a PWM sampling frequency of 200 kHz. The generated PWM signals are folded with the channel and the resulting sinus signals are shown in figure 5. The sinus signal shaded in gray is the reference signal when generated in the ADC sampling frequency domain. By comparing the prmitive generated sinus signal in figure 5a with the BMP optimized sinus in figure 5c, the BMP optimized sinus obviously fits better the reference signal than the primitive generated sinus signal. Especially in the middle of the sequence, the BMP optimized sinus signals fits nearly perfect the reference signal.

VI. Conclusion

We introduce for binary optimization especially for PWM generation an adopted Binary Matching Pursuit (BMP) algorithm. The simulation results shows, that the algorithm outperforms the primitive and COBYLA optimized PWM generation. Especially for small compression factors, the BMP can improve the signal by optimizing the PWM. For higher compression factors and therefore higher PWM sampling frequencies their is no significant improvements of the signal by the optimization algorithms.

Acknowledgement

We gratefully acknowledge financial support from "Spitzencluster MicroTec Suedwest" and German Federal Ministry of Education and Research (BMBF).

References

[1] P. A. Höher, *Grundlagen der digitalen Informationsübertragung Von der Theorie zu Mobilfunkanwendungen*. Wiesbaden: Springer Vieweg, 2013.

[2] E. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[3] D. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[4] M. Stojnic, "Recovery thresholds for l1 optimization in binary compressed sensing," in *2010 IEEE International Symposium on Information Theory Proceedings (ISIT)*, 2010, pp. 1593–1597.

[5] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, 1993.

[6] Y. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition," in *1993 Conference Record of The Twenty-Seventh Asilomar Conference on Signals, Systems and Computers, 1993*, 1993, pp. 40–44 vol.1.

[7] D. Needell and J. Tropp, "CoSaMP: iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, May 2009.

[8] M. J. D. Powell, "Direct search algorithms for optimization calculations," *Acta Numerica*, vol. 7, pp. 287–336, 1998.