# IMPLEMENTATION OF GRAM-SCHMIDT CONJUGATE DIRECTION AND CONJUGATE GRADIENT ALGORITHMS

Ricardo Insa Hernández, Ramin Baghaie and Kimmo Kettunen
Helsinki University of Technology
Laboratory of Telecommunications Technology
P.O. Box 3000
02015 HUT, Finland

## ABSTRACT

In this paper, we consider the implementation of two iterative methods: the Gram-Schmidt Conjugate Direction (GSCD) and Conjugate Gradient (CG) algorithm. We first study the convergence properties of the algorithms. Furthermore, the fixed-point implementation of both CG and GSCD are presented. For the simulations purposes, the algorithms are applied to a multiuser detection scheme in a DS-CDMA system. Finally, for different number of users, wordlengths, and number of iterations, the MSE resulting from the quantization is estimated.

## I. INTRODUCTION

The least mean-squares (LMS) and the recursive least-squares (RLS) algorithms are the most widely used adaptive algorithms. In order to solve finite linear equation systems of the type **Rw**=**b**, where **R** is a known semi-definite positive symmetric matrix, **b** is a known vector and **w** is an unknown vector, the RLS algorithm can be utilized. However, the RLS implies a high computational complexity and heavy matrix manipulations. Therefore, the RLS algorithm has a tendency to be numerically unstable. In order to avoid matrix inversions and stability problems other adaptive methods should be utilized.

The Conjugate type algorithms, such as CG method [1], are a family of iterative solvers of linear equation systems. Recently, they have been used in many applications, such as multiuser detection in Wideband Code Division Multiple Access (WCDMA) [2, 3] and mobile user tracking systems [4].

In this paper, the practical implementation of two adaptive Conjugate type algorithms, the GSCD and CG algorithms are studied and compared. For both algorithms, we study their convergence properties, their computational complexity and their fixed-point implementations. Fixed-point implementation schemes have some advantages in terms of increased speed, reduced power consumption and reduced hardware cost over the floating-point counter parts. However, as a result of the fixed-point implementation, quantization error will occur that will affect the convergence of the algorithms.

## II. THE ALGORITHMS

The two algorithms studied here are both based on the generic method of Conjugate Directions (CD) [5, 6]. The generic approach is to find a set of orthogonal search directions and to minimize the error along one search direction every iteration. Then, after at most $N$ steps, the search will converge to the solution of the equation system.

The generic CD method does not specify how the orthogonal set of search directions should be obtained. The different ways of obtaining the search directions determine the type of a particular CD algorithm.

### *A. Gram-Schmidt Conjugate Directions*

In the GSCD algorithm, a $R$-conjugate variant of the well-known Gram-Schmidt orthogonalization procedure [5] is used to find the $R$-orthogonal search directions $\mathbf{p}(n)$. The resulting algorithm is shown in Table I.

During each iteration step $n$, the new weight vector estimate $\mathbf{w}(n)$ is computed as the linear combination of the existing directions $\{\mathbf{p}(1), \ldots, \mathbf{p}(n)\}$, whose residual error $\mathbf{g}(n)$ is minimal. The next search direction is computed by applying the Gram-Schmidt orthogonalization step to the residual error vector. Since the $N$ search direction span the whole vector space this algorithm always converges at most in $N$ steps.

TABLE I
THE GSCD ALGORITHM

$\mathbf{w}(0)=\mathbf{0}$, $\mathbf{g}(0)=\mathbf{b}$, $\mathbf{p}(1)=\mathbf{g}(0)$ and $n$=1
while $n \le N$

$$\alpha(n)=\frac{\mathbf{p}(n)^{\mathrm{T}}\mathbf{g}(n-1)}{\mathbf{p}(n)^{\mathrm{T}}\mathbf{R}\mathbf{p}(n)}$$
$$\mathbf{w}(n)=\mathbf{w}(n-1)+\alpha(n)\mathbf{p}(n)$$
$$\mathbf{g}(n)=\mathbf{g}(n-1)-\alpha(n)\mathbf{R}\mathbf{p}(n)$$
$$\mathbf{p}(n+1)=\mathbf{g}(n)-\sum_{k=1}^{n}\frac{\mathbf{g}(n)^{\mathrm{T}}\mathbf{R}\mathbf{p}(k)}{\mathbf{p}(k)^{\mathrm{T}}\mathbf{R}\mathbf{p}(k)}\mathbf{p}(k)$$
$$n=n+1$$

end

### B. *Conjugate Gradients*

In the CG algorithm, the new search direction is still selected to be R-conjugate to the previous search directions. The generation of the new search direction has however a lower complexity when compared to the Gram-Schmidt orthogonalization step in the GSCD algorithm. Table II illustrates the CG algorithm [6].

TABLE II
THE CG ALGORITHM

$\mathbf{w}(0)=\mathbf{0}$, $\mathbf{g}(0)=\mathbf{b}$, $\rho(1)=\mathbf{g}(0)^{\mathrm{T}}\mathbf{g}(0)$,
$\mathbf{p}(1)=\mathbf{g}(0)$ and $n$=1
while $n \le N$

$$\alpha(n)=\frac{\rho(n)}{\mathbf{p}(n)^{\mathrm{T}}\mathbf{R}\mathbf{p}(n)}$$
$$\mathbf{w}(n)=\mathbf{w}(n-1)+\alpha(n)\mathbf{p}(n)$$
$$\mathbf{g}(n)=\mathbf{g}(n-1)-\alpha(n)\mathbf{R}\mathbf{p}(n)$$
$$\rho(n+1)=\mathbf{g}(n)^{\mathrm{T}}\mathbf{g}(n)$$
$$\beta(n)=\frac{\rho(n+1)}{\rho(n)}$$
$$\mathbf{p}(n+1)=\mathbf{g}(n)+\beta(n)\mathbf{p}(n)$$
$$n=n+1$$

end

In these algorithms, $\alpha(n)$ is the step size which minimizes the cost function along the search direction $\mathbf{p}(n)$ and it is used in the update of the weight vector $\mathbf{w}(n)$, $\mathbf{g}(n)$ is the residual of the function and points to the direction of the steepest descent. The new weight vector $\mathbf{w}(n)$ is computed as a linear combination of the previous weight vector and the search direction. In the CG algorithm, factor $\beta(n)$ ensures that the $R$-orthogonality is preserved between the new search directions. Due to non-linear processing, the Polack-Ribière formula [5], shown in (1), can be utilized in the computation of $\beta(n)$, for resetting purposes.

$$\beta(n)=\max\left\{\frac{(\mathbf{g}(n)-\mathbf{g}(n-1))^{\mathrm{T}}\mathbf{g}(n)}{\mathbf{g}(n-1)^{\mathrm{T}}\mathbf{g}(n-1)},0\right\} \qquad (1)$$

## III. THE CONVERGENCE PROBLEMS

Despite the fact that theoretically the CG algorithms must converge in at most $N$ steps, in practice, due to the round-off errors sometimes this may not be the case. Based on the numerical results in this section, we will demonstrate that for large $N$ the CG algorithm does not always converge after $N$ steps. Furthermore, depending on how close the initial conditions are from the solution of the linear equation system the same situation may happen and the algorithm will not converge in $N$ steps.

### A. *Numerical Results*

To demonstrate the above mentioned problems, we have used random symmetric positive-definite matrices and element-wise independent random vectors $\mathbf{w}_{opt}$ with zero mean and variance var($\mathbf{w}_{opt}$), where $\mathbf{w}_{opt}$ approximates the exact solution of $\mathbf{Rw}=\mathbf{b}$. Since the random vector $\mathbf{w}_{opt}$ has a zero mean, the mean-squared length of the vector is $N$var($\mathbf{w}_{opt}$). Thus, the distance of the $\mathbf{w}_{opt}$ from the origin can be controlled through the variance of $\mathbf{w}_{opt}$. In Fig. 1, is it shown that when the optimal solution, $\mathbf{w}_{opt}$, is close to the initial value of $\mathbf{w}$, $\mathbf{w}(0)$, in other words when the variance is rather small, the CG algorithm converges. As the variance of $\mathbf{w}_{opt}$ increases, more iterations are required to achieve an acceptable mean squared error (MSE).
As a result, in such cases, for an acceptable MSE more iteration steps are needed as can be

seen from Fig. 3. Of course, using initial conditions close to the optimal value is one solution to this convergence problem. In many applications, however, this may not be practical. On the other hand, the GSCD method always converges in $N$ steps, as observed in Fig. 2.

## IV. Computational Complexity

In this Section the computational complexities of the CG and GSCD are studied and compared. In these calculations, one division has the same complexity as one multiplication. Note that for estimating the complexities, we have only considered the number of multiplications. This is due to the fact that multiplications are more complex than additions [7]. As compared to the CG algorithm, in the GSCD, computations of the new search direction $\mathbf{p}(n)$ are more complex and require a higher number of vector inner products. The results are shown in Table III.

TABLE III
COMPARISON OF COMPUTATIONAL COMPLEXITIES OF THE ALGORITHMS

| *Algorithm* | *Number of multiplications* |
|---|---|
| CG | $I(N^2+5N+2)-2N-1$ |
| GSCD | $I(N^2+2N+I(2N+1))-N$ |

$I$: Number of iterations.

When calculating the computational complexities of both algorithms, one should note that in the last iteration for updating the filter coefficients the only computation required is the calculation of the step size $\alpha$. These complexities are calculated supposing that the initial guess $\mathbf{w}(0)$ is equal to zero. In both algorithms, if the initial weight vector is nonzero, in other words, if we have an initial guess, the residue will be calculated as (2) and the computational cost will increase in $N^2$ multiplications.

$$\mathbf{g}(0) = \mathbf{b} - \mathbf{R}\mathbf{w}(0) \qquad (2)$$

The computational complexity depends on the number of iterations $I$ in both methods. Thus, as the calculation of the new search direction vector $\mathbf{p}(n)$ in the GSCD requires the existing search directions $\{\mathbf{p}(1), \ldots, \mathbf{p}(n)\}$, the dependence with the number of iterations $I$ is larger than in the CG method.

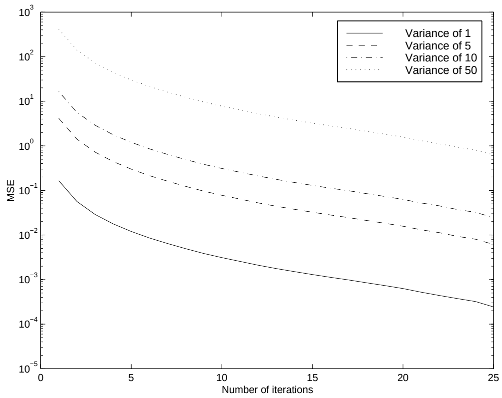

Fig. 1. MSE versus the var($w_{opt}$) for the CG

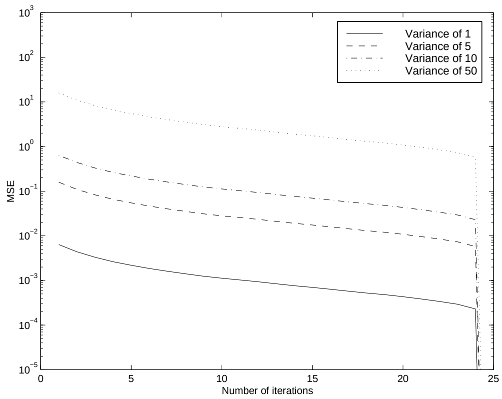

Fig. 2. MSE versus the var($w_{opt}$) for the GSCD

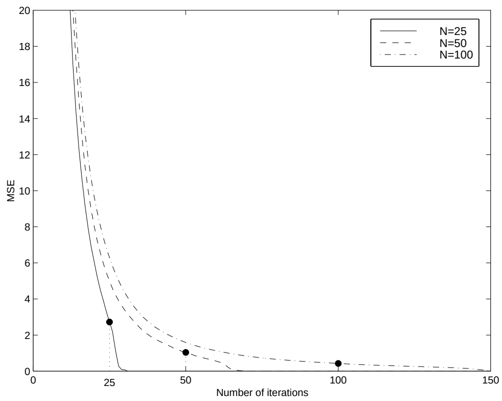

Fig. 3. Convergence of the CG algorithm for longer iteration steps

## V. FIXED-POINT IMPLEMENTATION

In this section, we discuss the fixed-point implementation of the algorithms and evaluate the resultant quantization error. A fixed-point number can be represented with $b_i$ bits for the integer part and $b_f$ bits for the fractional part as it is shown in Fig. 4. The parameter $b_i$ determines the dynamic range while the parameter $b_f$ determines the precision of the problem. In the following section, throughout extensive simulations we estimate the optimum the wordlength, $b_i+b_f$, for both the CG and the GSCD algorithms, when using a multiuser detection scheme.

### *A. Numerical Results*

The simulator is written in Matlab and it is assumed that all the arithmetic operations have the same input and output wordlength. For more realistic results, the algorithm was applied to a multiuser detection scheme in a DS-CDMA system [2].

In these simulations, a fix number of 10 bits plus a sign bit, were assigned to the integer part of the parameters, the fractional part varied between 3 to 21 bits, varying the wordlength between 14 to 32. The number of users $K$, was 33 and 65, the signal-to-noise ratio was 8 dB, the data block length, $N_b$, was 1, the number of chips per symbol $N_c$ are 31 and 63, respectively. The number of samples per chip, $N_{sc}$, was set to be 2 in order to assure $K<N_s$, where $N_s$ is the number of samples per symbol and is defined as $N_s=N_cN_{sc}$.

Fig. 5 and Fig. 6 illustrate the results of these simulations after four iterations when initial guess is zero or it is the output of the matched filter, respectively. Fig. 7 represents the same results as in Fig. 6, but after six iterations. As can be observed from Fig. 5 and Fig. 6, by utilizing the initial guess in the simulations, the MSE will reduce. However longer wordlengths are required to reach the same MSE level as when using floating point arithmetic.

In Fig. 8, for different number of iterations, the fixed-point performance of the algorithms is demonstrated. In these simulations $b_i$ and $b_f$ were 10 and 6, respectively. As can be observed from this figure, simply with the

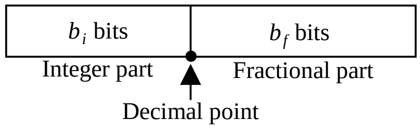


Fig. 4. Fixed-point representation

wordlength of 16 bits we may not reach the same MSE as with the floating-point arithmetic and more bits should be utilized.

## VI. CONCLUSION

In this paper, practical implementations of the Gram-Schmidt Conjugate Direction and Conjugate Gradient algorithms were studied and compared. We illustrated that although theoretically the CG algorithm must converge in at most $N$ steps, in practice for large $\mathbf{R}$ it may not happen. As a result, due to round-off errors different resetting schemes and more iterations were required. On the other hand, we demonstrated that the GSCD method always converges at most in $N$ steps.

The fixed-point implementation of the above mentioned algorithms were also studied and presented. For more realistic results, the algorithms were applied to a multiuser detection scheme in a DS-CDMA system. In these simulations, for different number of users, different number of iterations and different wordlengths the MSE errors were calculated.

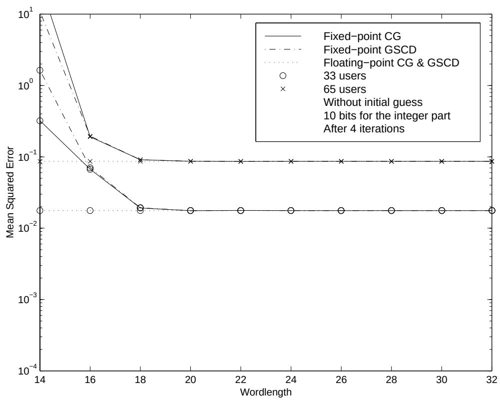


Fig. 5. Fixed-point performance of GSCD and CG algorithms without initial guess
($I$=4, $K$=33 and 65, $SNR$=8dB, $N_b$=1)

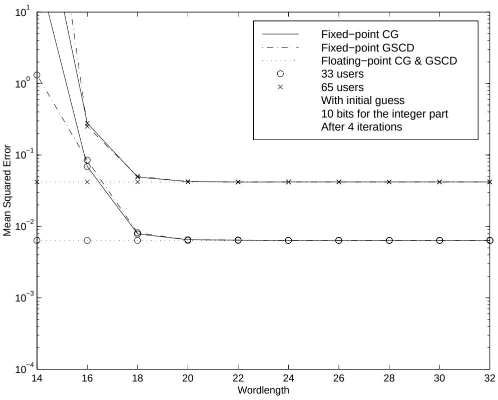


Fig. 6. Fixed-point performance of GSCD and CG algorithms with initial guess ($I$=4, $K$=33 and 65, $SNR$=8dB, $N_b$=1)

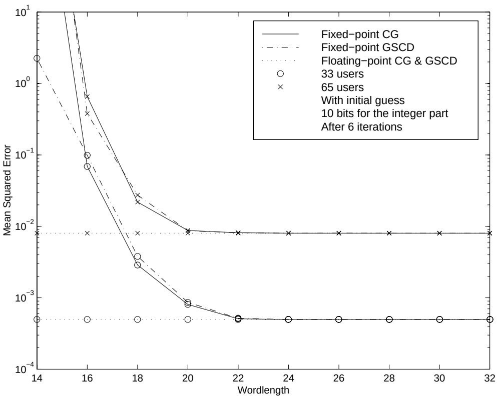


Fig. 7. Fixed-point performance of GSCD and CG algorithms with initial guess ($I$=6, $K$=33 and 65, $SNR$=8dB, $N_b$=1)

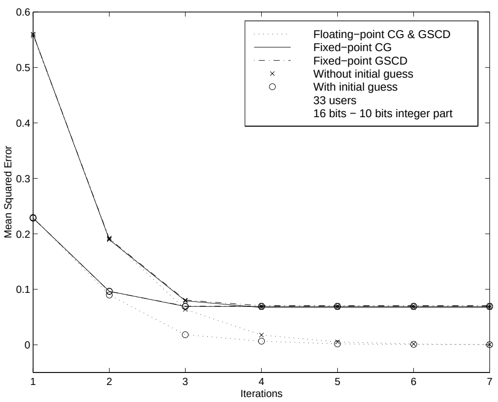


Fig. 8. Fixed-point performance of the CG and GSCD algorithms for different number of iterations ($K$=33, $SNR$=8dB, $N_b$=1)

## ACKNOWLEDGEMENT

The authors would like to thank Dr. Markku Juntti for his valuable comments.

This research was founded in part by the Academy of Finland, Technology Development Center (TEKES), NOKIA Research Center, Sonera Ltd. and the Helsinki Telephone Company.

## REFERENCES

[1] M. R. Hestenes and E. Stiefel, “Methods of conjugate gradients for solving linear systems,” *Journal of Research of the National Bureau of Standards*. Vol.49, No. 6, pp. 409-436, December 1952.

[2] M. Juntti, B. Aazhang and J. O. Lilleberg, “Iterative implementation of linear multiuser detection for dynamic asynchronous CDMA systems,” *IEEE Trans. on Communications*, Vol. 46, No. 4, pp. 503-508, April 1998.

[3] S. Das, J. R. Cavallaro and B. Aazhang, “Computationally efficient multiuser detectors,” in *Proc. IEEE Int. Symp. Personal, Indoor and Mobile Radio Commun.*, Vol. 1, pp. 62-67, Helsinki, Finland, Sept. 1-4, 1997.

[4] P. Karttunen, R. Baghaie, “Conjugate gradient based signal subspace mobile user tracking,” to appear in *Proc. VTC*, May 1999.

[5] J. R. Shewchuk, “An introduction to the conjugate gradient method without the agonising pain,” *http://www.cs.cmu.edu/~jrs/jrspapers.html*, August 1994.

[6] G. H. Golub, C. F. Van Loan, *Matrix Computations*. 3rd edition, The Johns Hopkins University Press, 1996.

[7] T. Callaway and E. Swartzlander, “Optimizing arithmetic for signal processing,” in *Proc. IEEE Workshop on VLSI Signal Processing, V*, Napa Valley, California, pp. 91-100, October 1992.