

CONJUGATE GRADIENT BASED SIGNAL SUBSPACE MOBILE USER TRACKING

P. Karttunen, R. Baghaie

Laboratory of Telecommunications Technology
Helsinki University of Technology
P.O.Box 3000, FIN-02015 HUT, Finland
e-mail: Petri.Karttunen@hut.fi, Ramin.Baghaie@hut.fi

Abstract - Adaptive antennas enable the increase in the system capacity through spatial interference suppression in the beamforming mode. All the spatial domain channel estimators must react fast to abrupt changes in the channel parameters. Additionally, they have to provide accurate enough tracking capabilities. In this paper, we construct the user tracking system by employing an efficient Conjugate Gradient (CG) method based on a step-by-step adaptation scheme. Furthermore, the paper investigates the tracking capability and the computational complexity issues of the proposed system. The simulation results confirm that the proposed CG method achieves better tracking performance than gradient methods both in the stationary and non-stationary signal scenario.

I INTRODUCTION

Adaptive array processing techniques can provide more system capacity by reducing co-channel interference. For this aim the channel information must be accurately acquired both in the stationary and non-stationary signal scenario. The channel parameter-tracking problem arises in the numerous situations, e.g., in the mobile user tracking where the spatial beamforming procedure must be carried out continuously for each user in order to maximise the signal powers of the desired users. In our paper a training signal based CG method and the gradient algorithm are implemented that adaptively in the sample-by-sample manner update the signal eigensubspace containing the relevant tracking parameters.

The paper is organised as follows: in Section 2, the signal model is formulated for the user tracking system. Section 3 presents the overall system model partitioning the user-tracking problem to the tracking, DOA extraction and the beamforming units. Section 4 discusses the computational complexity of the components of the tracking system. In Section 5, the numerical results comparing the gradient and the proposed CG methods are presented. Section 6 draws appropriate conclusions about the tracking performance of our adaptation schemes.

II SIGNAL MODEL

The observation vector $\mathbf{x}(t)$ of a one-tap channel model at the antenna receiver can be expressed as

$$\mathbf{x}(t) = \sum_{k=1}^K \sqrt{P_k} b_k(t) \mathbf{a}(\theta_k) + \mathbf{n}(t) \quad (1)$$

where M -sized column vectors $\mathbf{a}(\theta_k)$ and $\mathbf{n}(t)$ are the array response vector for the k th user ($k=1, \dots, K$) and the Additive White Gaussian Noise (AWGN) vector, respectively. The number of co-channel users is K . The array response vector is parameterised by θ_k which represents Direction-of-Arrival (DOA) on the azimuth plane. As a result, the user tracking problem becomes that of continuously updating the parameter θ_k . The transmitted signal $b_k(t)$ is modelled as a zero-mean Gaussian distributed process with the variance P_k . Similarly, $\mathbf{n}(t)$ is also drawn from a Gaussian distribution with the variance σ_n^2 . Signal-to-Noise Ratio (SNR) is defined to be $10 \log(P_k/\sigma_n^2)$ [1].

The quantity $\mathbf{a}(\theta_k)$ is the normalised array response vector. For the Uniform Linear Antenna (ULA) arrays it can be expressed as

$$\mathbf{a}(\theta_k) = \frac{1}{\sqrt{M}} [1 \exp(j\psi_k) \dots \exp(j(M-1)\psi_k)]^T \quad (2)$$

where the array factor ψ_k is defined to be $2\pi d \sin(\theta_k) f/c$. This factor includes the distance between elements d , the communication frequency f and the speed of light c .

In the matrix notation, Eq (1) can be expressed more compactly as

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{n}(t) \quad (3)$$

All the steering vectors are collected into a matrix \mathbf{A} , which can be expressed as

$$\mathbf{A} = [\mathbf{a}(\theta_1) \dots \mathbf{a}(\theta_k)] \quad (4)$$

The model correlation matrix for the antenna array can be expressed as

$$\mathbf{R} = \mathbb{E}[\mathbf{x}(t)\mathbf{x}(t)^H] = \mathbf{A}\mathbf{S}\mathbf{A}^H + \sigma_n^2\mathbf{I} \quad (5)$$

where $\mathbf{S} = \mathbb{E}[\mathbf{s}(t)\mathbf{s}(t)^H]$ is the signal correlation matrix and \mathbf{A} is a slowly varying antenna response matrix for all the users. The signal subspace updating schemes rely on the updating process of the adaptive algorithms to converge to the dominant signal subspace spanned by the full rank signal correlation matrix $\mathbf{A}\mathbf{S}\mathbf{A}^H$ [2].

Although this is a crude signal model for a typical radio communication channel, it is assumed that the multipath components essentially arrive from the same direction and thus cannot be resolved by the beamformer. Furthermore, the possible resolved signal components are coherent and the coherency removing methods like Spatial Smoothing (SS) are needed to restore the rank of the rank deficiency matrix \mathbf{R} .

III SYSTEM MODEL

The block diagram of Figure 1 describes the overall system model for our tracking system. Next, the system components will be described in detail.

Tracking Unit

The signal subspace tracking unit deals with the adaptation schemes for updating the signal subspace \mathbf{A}_n . The parameter estimation problem will be formulated both for a gradient and especially for a CG based method. As an adaptive implementation, we propose a CG method based on the step-by-step update scheme of [3] which has been implemented to update the signal subspace with every new incoming sample by using the given cross-correlation information. In [4], we have utilised a gradient like method with an adaptive step size for the user tracking. The adaptive step size method was based on the block processing form. N samples were gathered for better performance and one update iteration was performed. For two reasons this turns out to be disadvantageous. First, although good convergence properties could be achieved but in the practical systems we favour step-by-step update mode rather than instantaneous block update. Secondly, the non-stationarity in the collected samples will result in inaccurate estimates. Therefore, we implement the CG algorithm in the step-by-step update mode although with

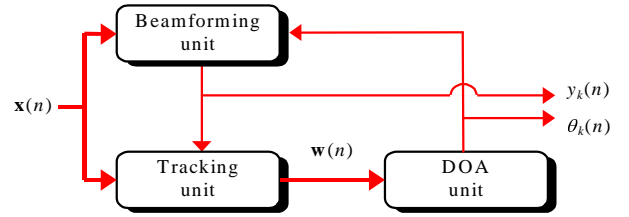


Figure 1 Overall system block diagram for the signal subspace user tracking (adaptation update step n).

the computational complexity increase as opposed to the gradient method.

The general cost function of the gradient method can be expressed as an unconstrained minimisation problem [5] as

$$\mathbf{J} = \mathbb{E} \left[\left\| \mathbf{x} - \mathbf{A} \left[\mathbf{A}^H \mathbf{A} \right]^{-1} \mathbf{A}^H \mathbf{x} \right\|^2 \right] \quad (6)$$

This can be interpreted as the projection of the observation vector \mathbf{x} into the noise subspace. As an adaptive implementation, the gradient based inflation like update scheme with a step size μ can be invoked for updating the signal subspace \mathbf{A} directly. In [4], we modified the same cost function with the adaptive step size control for faster convergence and lower misadjustment.

The quadratic cost function of our CG problem for the k th user can be formulated as follows:

$$J = \mathbf{a}(\theta_k)^H \mathbf{R} \mathbf{a}(\theta_k) + \mathbf{a}(\theta_k)^H \mathbf{b} \quad (7)$$

where \mathbf{R} and \mathbf{b} are the model correlation matrix and the model cross-correlation vector, respectively. This quadratic cost function will be used as the minimisation criterion of interference and noise components [5,6]. In our tracking system, computationally complex Gram-Schmidt orthonormalisation step can be eliminated, which is needed in the ordinary adaptive eigensubspace algorithms.

In our system derivation, initially unknown antenna response vector $\mathbf{a}(\theta_k)$ will be replaced with a weight vector estimate $\mathbf{w}(n)$. The weight vector $\mathbf{w}(n)$ converges to the desired steering vector, which correlates best the desired user signal. The sample autocorrelation matrix $\mathbf{R}(n)$ and the sample cross-correlation vector $\mathbf{b}_k(n)$ are rank one update with a forgetting factor λ_f . They are estimated by using exponentially decaying data windows with a forgetting factor λ_f as expressed in Eq. (8) where $d(n)$ is the reference signal. The forgetting factor λ_f will reflect the amount of signal non-stationarity in the correlation matrix estimate.

$$\begin{aligned}\mathbf{R}(n) &= \lambda_f \mathbf{R}(n-1) + \mathbf{x}(n)\mathbf{x}^H(n) \\ \mathbf{b}(n) &= \lambda_f \mathbf{b}(n) + d(n)\mathbf{x}(n)\end{aligned}\quad (8)$$

Residual vector points to the direction of the steepest descent, i.e., the negative gradient of the cost function and can be expressed as

$$\mathbf{g}(n) = -\nabla_w J = -\mathbf{R}(n)\mathbf{w}(n) - \mathbf{b}(n) \quad (9)$$

In the algorithm, we choose the step size that minimises the cost function J through the line search procedure along the search direction $\mathbf{p}(n-1)$. As a result, the step size can be expressed as

$$\alpha(n) = \eta \frac{\mathbf{p}^H(n-1)\mathbf{g}(n-1)}{\mathbf{p}^H(n-1)\mathbf{R}(n)\mathbf{p}(n-1)} \quad (10)$$

where η is an auxiliary step size parameter. The new weight vector is computed as a sum of previous weight vector and the scaled direction vector $\mathbf{p}(n-1)$.

$$\mathbf{w}(n) = \mathbf{w}(n-1) + \alpha(n)\mathbf{p}(n-1) \quad (11)$$

The key point in the sample-by-sample processing is that we update the residual vector of Eq (9) by incorporating the estimates of the correlation matrix and cross-correlation vector in Eq (8) into the residual vector [3].

$$\begin{aligned}\mathbf{g}(n) &= \lambda_f \mathbf{g}(n-1) - \alpha(n)\mathbf{R}(n)\mathbf{p}(n-1) \\ &+ \mathbf{x}(n)(d(n) - \mathbf{x}(n)^H \mathbf{w}(n-1))\end{aligned}\quad (12)$$

In the conventional CG, the residuals will be used as search directions. Therefore, in the conventional Block Conjugate Gradient (BCG) algorithm the convergence could be achieved in M steps. The new search directions are forced to be \mathbf{R} -orthogonal to all the previous search directions. The factor $\beta(n)$ will be used for ensuring that this \mathbf{R} -orthogonality is preserved. In the sample-by-sample processing the direction vectors lose orthogonality and periodic resetting will be eventually needed. Due to this non-linear sample-by-sample processing, Polak-Ribiere method with resetting in the computation of the $\beta(n)$ factor will be used for providing faster convergence.

$$\beta(n) = \max \left\{ \frac{(\mathbf{g}(n) - \mathbf{g}(n-1))^H \mathbf{g}(n)}{\mathbf{g}^H(n-1)\mathbf{g}(n-1)}, 0 \right\} \quad (13)$$

The new search directions can be evaluated as

$$\mathbf{p}(n) = \mathbf{g}(n) + \beta(n)\mathbf{p}(n-1) \quad (14)$$

DOA Unit

The weight vector of the tracking unit converges toward a steering vector, which maximises the signal power for the desired user. The DOA extraction unit updates the estimates for the tracking angles $\theta_k(n)$ due to the spatial movements of users by using linear regression on the samples from the array manifold. The DOAs are to be extracted from the steering vector representation of Eq (2). As an extraction method, Least Square (LS) performance criterion will be employed. The LS criterion is based on the linear model, which can be expressed as

$$\mathbf{x} = \mathbf{H}\theta_k + \mathbf{n} \quad (15)$$

where θ_k is the parameter of interest to be estimated, \mathbf{H} is the $M \times 1$ observation vector and \mathbf{n} is the model noise vector. The LS estimator can be expressed as

$$\hat{\theta}_k^{(LS)}(n) = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{z}_k(n) \quad (16)$$

where $\mathbf{z}_k(n)$ consists of M samples of θ_k from the array manifold.

The observation vector \mathbf{H} contains the known quantities from the steering vector. In the general case, these values depend on the array structure. For the linear array configuration they can be defined as $2\pi d(m-1)\sin(\theta_k)/c$, where m ($m=1, \dots, M$) is an index of the antenna element. Therefore, for the ULA antenna the observation vector \mathbf{H} can be expressed as

$$\mathbf{H} = [0 \ \pi \ 2\pi \ \dots \ (M-1)\pi]^T \quad (17)$$

The parameter θ_k inside the exponent in the steering vector representation is extracted through the log-operation. Because of the linear estimation of the parameter θ_k from $\sin(\theta_k)$ some error will emerge. This error is tolerable when the small changes in the estimated parameter θ_k can be assumed. The LS estimation is done on the array manifold difference of the CG estimated weight vector $\mathbf{w}(n)$ and $\mathbf{a}(\theta_k(n-1))$ estimated in the previous iteration round that can be specified as

$$\mathbf{z}_k(n) = \text{Im}(\log(\mathbf{w}(n) \div \mathbf{a}(\theta_k(n-1)))) \quad (18)$$

where \div is defined to be elementwise vector division. This kind of difference computation is applied because otherwise, sudden changes in the parameter value can not accurately be tracked through the linear model. The new DOA $\theta_k(n)$ for the k th user is based on the presented LS estimation and can be expressed as

$$\theta_k(n) = \theta_k(n-1) + \hat{\theta}_k^{(LS)}(n) \quad (19)$$

It should be noted that as the weight vector estimate of the tracking unit has converged the DOA extraction unit does not provide any further estimation errors.

Beamforming Unit

In the beamforming unit the normalisation procedure is carried out by replacing the weight vector estimate $\mathbf{w}(n)$ by the normalised antenna response vector $\mathbf{a}(\theta_k(n))$. This updated vector is fed back to the tracking unit for the next iteration cycle. In our system, we utilise the conventional beamforming method [2] for computing the beamforming signal $y_k(n)$ that can be expressed as

$$y_k(n) = (\mathbf{w}^H(n)\mathbf{w}(n))^{-1} \mathbf{w}^H(n)\mathbf{x}(n) \quad (20)$$

IV COMPLEXITY OF THE SYSTEM

In this section, the order of complexity of different units are estimated and compared. In the tracking, DOA, and beamforming units, the most computationally intensive operations are Eq (10), Eq (16) and Eq (20), respectively. When estimating the computational complexity of the DOA unit, the log function of Eq (18) was not considered. This was due to the fact that for the implementation of the log function, different techniques such as table oriented methods, iterative methods, and polynomial approximations can be utilized. In fact in [7], it was shown that $\log(x)$, can be evaluated to n significant bits in $O(M_u(n)\log(n))$ steps, where $M_u(n)$ is units of time required to multiply n -bit numbers. In Table 1, we have calculated the order of complexity for different units of Figure 1. In many cases $K \ll M$, therefore, the complexities in Table 1 can still be reduced by a factor of K .

Table 1 Comparison of computational complexities

Unit	Equation Number	Order of complexity
Tracking	(10)	$O(KM^2)$
DOA	(16)	$O(KM)$
Beamforming	(20)	$O(KM)$

As can be seen from Table 1, the tracking unit has the highest order of complexity. It is clear that the core of this unit is the sample-by-sample CG algorithm. Therefore, in the following section, we will discuss the complexity of the algorithm in more details.

Computational Complexity of the CG algorithm

As compared to the conventional CG algorithm, the computation of the residual vector $\mathbf{g}(n)$ and the factor $\beta(n)$ of the sample-by-sample CG algorithm are more complex and require a higher number of vector inner products. Therefore, in this section the computational complexities of the sample-by-sample Conjugate Gradient algorithm and the conventional Conjugate Gradient also referred to as Block Conjugate Gradient (BCG) algorithms are studied and compared. In these comparisons all the parameters are complex and 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. The results are shown in Table 2.

Table 2 Comparison of computational complexities of different CG algorithms

Algorithm	Number of complex multiplications
BCG	$I(M^2 + 5M + 2) - 2M - 1$
Sample-by-sample CG	$M^2 + 10M + 3$

I : Maximum number of iterations for a block

When calculating the computational complexity of the BCG, one should note that in the last iteration for updating the filter coefficients the only necessary computation required is the calculation of the step size α . It is clear that the computational complexity of the BCG depends on the number of iterations I . Thus, for large M the computational complexity of the BCG is I times of the complexity of the sample-by-sample CG algorithm.

In the CG algorithm, the dominating operation is the matrix-vector product of Eq (10). Furthermore, for every new sample in the tracking unit one should estimate the correlation matrix according to Eq (8). Therefore, by considering the practical aspects of the system, for the tracking unit adequate hardware should be developed. In our future work, we will discuss the implementation of our tracking system, and will focus on developing efficient VLSI array processors that are suitable for real time applications.

V NUMERICAL RESULTS

The simulation system setting assumes the base station or similar configuration where $M=8$ linearly arranged

antenna elements with $\lambda/2$ spacing is applied. The users introduced into the cell area have moderate SNR=20 dB and their initial location estimates possess the 5° pointing errors. As the performance measure, the mean DOA error criterion has been utilised and the results have been averaged over 2000 independent realisations. In the case of a stationary signal scenario, the users are at the fixed locations whereas in the non-stationary signal scenario the users are moving with the constant angular speed of 0.025 samples/deg. In this kind of the simulation setting both adaptive algorithms can cope with the specified angular speed without divergence by using only one update step for the signal subspace on every incoming sample.

Figure 2 illustrates the simulation results. Figure 2a) presents a tracking realisation of the CG based method with the maximum allowable $M=8$ users that can still reliably be tracked. In Figure 2b) and c) the performance comparison of the tracking capability of the gradient and the CG based methods has been carried out. The CG based method provides faster convergence and smaller misadjustment. When the number of users increases and especially when they are closely located the CG method can more steadily track them as compared to the gradient method. Furthermore, the location estimates provided by the CG method behave smoother. The performance of the gradient method is greatly sensitive to the proper choice of the step size parameter μ . Therefore, the fixed step sizes selected were a trade-off between the convergence speed and the final misadjustment.

VI CONCLUSIONS

The step-by-step update scheme of the CG method was implemented for the user tracking system. The paper compared the tracking performance of the gradient and CG method in the stationary and non-stationary signal scenario. For the proposed CG method the simulation results indicated the better tracking performance in terms of the faster and smoother convergence and the smaller misadjustment. As for further research, we will discuss the implementation of our tracking system in the noise subspace and also focus on developing efficient VLSI array processors for real time applications.

ACKNOWLEDGEMENTS

This work is part of a research project of the Institute of Radio Communication (IRC) funded by the Technology Development Center (TEKES), NOKIA Research Center, Sonera and the Helsinki Telephone Company.

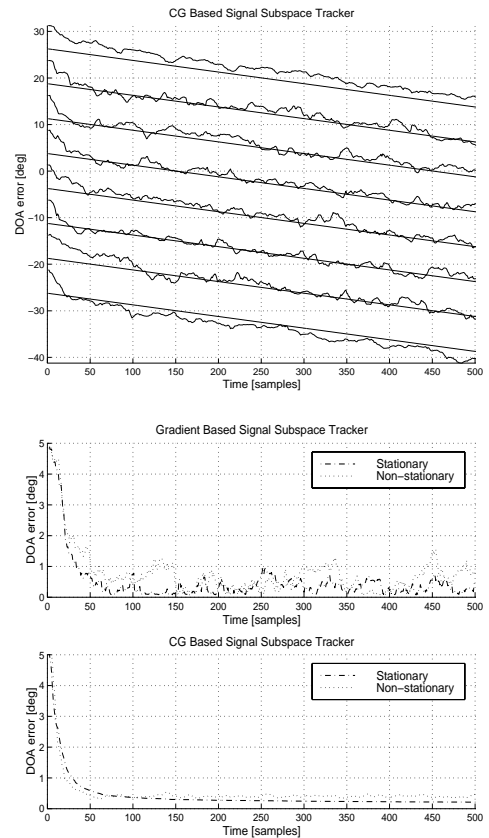


Figure 2 a) Sample tracking trajectory for the CG based signal subspace method with $M=8$ users. b) Mean DOA error for the gradient method with parameters $\mu=0.004$ (stationary) and $\mu=0.005$ (non-stationary). c) Mean DOA error for CG based method with parameters $\lambda_f=1$, $\eta=0.4$ (stationary) and $\lambda_f=0.8$, $\eta=0.6$ (non-stationary).

REFERENCES

- (1) C. W. Therrien, *Discrete Random Signals and Statistical Signal Processing*. Englewood Cliffs, NJ: Prentice Hall, 1992.
- (2) R. T. Compton, *Adaptive Antennas*. Englewood Cliffs, NJ: Prentice Hall, 1988.
- (3) S. P. Chang and A.W. Willson, "Adaptive filtering using modified conjugate gradient," in *Proc. 38th Midwest Symposium on Circuits and Systems*, Rio de Janeiro, Brazil, pp. 243-246, August 1995.
- (4) P. T. Karttunen, T. I. Laakso, and J. Lilleberg, "Tracking of mobile users in a mobile communications system using adaptive convergence parameter," in *Proc. International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'97)*, pp. 989-993, Helsinki, Sept. 1-4, 1997.
- (5) G. Mathew, V. U. Reddy, and S. Dasgupta, "Adaptive estimation of eigensubspace," *IEEE Trans. on Signal Processing*, vol. 43, no. 2, pp. 401-411, Feb. 1995.
- (6) Z. Fu and E. M. Dowling, "Conjugate gradient projection subspace tracking," *IEEE Trans. on Signal Processing*, vol. 45, no. 6, pp. 1664-1668, June 1997.
- (7) R. P. Brent, "Fast multiple-precision evaluation of elementary function," *Journal of the Association for Computing Machinery*, vol. 23, no. 2, pp. 242-251, April 1976.