Implementation of Adaptive Beamforming Based on QR Decomposition for CDMA

Zhaohui Liu   J. V. McCanny

DSiP Laboratories, Queen's University of Belfast, Belfast, N. Ireland

ABSTRACT

QR decomposition and inverse QR decomposition can be used for adaptive beamforming. These techniques can also be used in CDMA. Two different configurations based on chip interval and symbol interval are given and the computation throughput is compared. Some implementation details are also considered. It is shown that RLS algorithm based on QR decomposition can be implemented by DSP for adaptive beamforming. If further performance is required, a dedicated beamforming ASIC/FPGA chip is needed.

1. INTRODUCTION

3G mobile communications are based on CDMA. Adaptive beamforming can be used at a base station for the uplink and downlink (TDD-CDMA only) because of the following benefits: increasing the capacity, improving the quality, reducing the transmission power and etc.

Adaptive algorithms can be least mean squares (LMS), sample-matrix-inversion (SMI) or recursive least squares (RLS). The convergence rate of LMS may be slow and SMI may encounter numerical instability and so are not suitable for the case of fast fading channels. Based on QR decomposition, RLS can give fast convergence rate and have good numerical properties [8, 9].

Adaptive beamforming was originally used in radar system, and it is well known that RLS algorithm based on QR decomposition can be used in this case because *a posteriori* error can be found directly. However, for CDMA communications, the antenna weights must be given explicitly, which can be found by back-substitution in QR decomposition or directly by inverse QR decomposition.

In this paper, the canonical configuration of QR decomposition is given in section 2. And inverse QR decomposition for the computation of antenna weights is given in section 3. Implementation details related to 3G are discussed in the last section.

2. CANONICAL CONFIGURATION

Usually adaptive beamforming is implemented in the base station, and each user requires unique weightings. For CDMA, each user is assigned a unique signature waveform, which can be used to generate a reference signal. This is also called blind adaptation. When the training symbol or pilot symbol is available, it can be used as a reference signal and usually gives better performance than blind adaptation.

Assuming antenna input is $\mathbf{x} = [x^{(1)}\ x^{(2)} \cdots x^{(M)}]$, where M is the antenna number. The input is weighted by $\mathbf{w}_k = [w_k^{(1)}\ w_k^{(2)} \cdots w_k^{(M)}]$ and is then de-spread, we can find the tentative decision $\hat{b}_k$, where k is the k-th user. For high SIR case, $\hat{b}_k$ is regarded as correct decision and is re-spread, termed as z_k. z_k is in the chip interval and is used as a reference signal. The RLS algorithm can be mapped to a canonical triangular array [6], which is based on QR decomposition. This is shown in Figure 1.

The QR decomposition is related with RLS algorithm by Cholesky decomposition. Let $\mathbf{R}(n)$ be the Cholesky factor of the autocorrelation matrix of the input data, where $\mathbf{R}(n)$ is an upper triangular matrix. The updating equation at time n is given by [5]

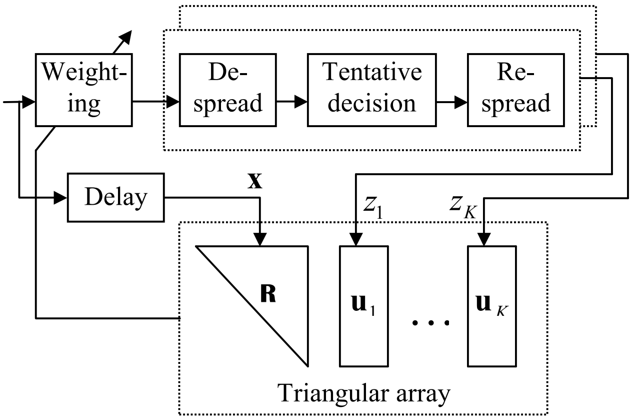

Figure 1. RLS algorithm mapped to a canonical triangular array (chip interval)

$$\mathbf{Q}(n)\begin{bmatrix}\sqrt{\beta}\mathbf{R}(n-1) & \mathbf{u}(n-1)/\sqrt{\beta} \\ \mathbf{x}^T(n) & z(n)\end{bmatrix} = \begin{bmatrix}\mathbf{R}(n) & \mathbf{u}(n) \\ \mathbf{0}^T & \eta(n)\end{bmatrix}$$

where $\mathbf{Q}(n)$ is orthogonal matrix.

The processing procedure can be divided into two steps. Firstly, the triangular array is in adaptive mode, and generates the updated triangular matrix $\mathbf{R}(n)$ and the corresponding vector $\mathbf{u}(n)$, and hence implicitly, the updated weight vector $\mathbf{w}(n)$; and secondly, the weight vector must be calculated by serial flushing out (for a systolic array) or by back-substitution (for DSP). It should be noted that the operation of weighting and QR decomposition is in chip interval, though the weights can be updated in a given number of symbol interval. The advantage of this structure is that the all users have the same input and so only one QR array is required. This is attractive for the case of large antenna numbers.

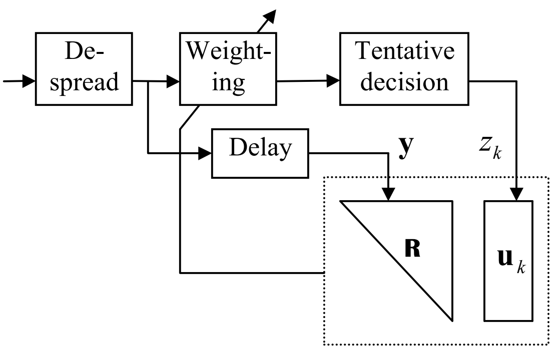

Figure 2. RLS algorithm mapped to a canonical triangular array (symbol interval)

Figure 2 gives another configuration. The antenna input is de-spread first, then all of the operation is in the symbol interval and the re-spread is not needed. It also requires the two operation steps. The adaptive antenna array is usually combined with the Rake receiver, which includes channel estimation and maximum-ratio-combination (MRC). After being de-spread, the input of the QR array is different for each user and so each user has its own QR array. However, because most arithmetic operations run at symbol interval and re-spreading is not needed, it is not computationally intensive. So this structure is more suitable than Figure 1 when the antenna number is small.

3. INVERSE QR DECOMPOSITION

Both configurations in section 2 are based on QR decomposition and so the flushing or back-substitution operation cannot be avoided. When there are many users, this will take a long time. In fact, an algorithm based on inverse QR decomposition can be used for calculating weights directly [4, 5]. This approach is based on time-recursive updating of $R^{-T}(n)$, the Cholesky factor of the inverse of the data autocorrelation matrix. The QR updating equation at time n is given by [5]

$$\begin{cases}\mathbf{a}(n) = \mathbf{R}^{-T}(n-1)\mathbf{x}(n)/\sqrt{\beta} \\ \mathbf{Q}(n)\begin{bmatrix}\mathbf{a}(n) & \mathbf{R}^{-T}(n-1)/\sqrt{\beta} \\ 1 & \mathbf{0}^T\end{bmatrix} = \begin{bmatrix}\mathbf{0} & \mathbf{R}^{-T}(n) \\ \mu(n) & \mathbf{v}^T(n)\end{bmatrix}\end{cases}$$

and then the Kalman gain vector $\mathbf{k}(n)$ and the weights $\mathbf{w}(n)$ are updated according to

$$\begin{cases}\mathbf{k}(n) = \mathbf{v}(n)/\mu(n) \\ \mathbf{w}(n) = \mathbf{w}(n-1) + e(n)\mathbf{k}(n)\end{cases}$$

where $e(n)$ is the error signal.

In this approach, there is a matrix-vector product operation before the QR updating. After the product $\mathbf{a}(n)$ is obtained, the inverse QR decomposition can also be mapped to a canonical triangular array. And then each user's weights can be updated recursively according to $\mathbf{k}(n)$

The inverse-QR decomposition can also be configured in the chip interval or symbol interval as in section 2. When operating in chip interval, $\mathbf{R}^{-T}(n)$ is the Cholesky factor of $\mathbf{R}_{xx}{}^{-T}(n)$, where $\mathbf{R}_{xx}{}^{-T}(n)$ is the time averaged autocorrelation matrix of the input data $\mathbf{x(n)}$. The reference signal is generated from the re-spreading operation and all the users share the same inverse matrix $R^{-T}(n)$. On the other hand, if it operates in the symbol level, i.e., the antenna input is de-spread first, then $\mathbf{R}^{-T}(n)$ is different for each user. Here $\mathbf{R}^{-T}(n)$ is the Cholesky factor of $\mathbf{R}_{yy}{}^{-T}(n)$, where $\mathbf{R}_{yy}{}^{-T}(n)$ is the autocorrelation matrix of the de-spread data $\mathbf{y}(n)$. And the reference signal does not need to be re-spread.

4. IMPLEMENTATION for WCDMA

For WCDMA uplink, control channel (including pilot signal) and data are code multiplexed to quadrature and in-phase. Control channel and data channel are spread by orthogonal code, however, multipath propagation degrades the orthogonality. The power ratio between the control channel and the data channel is adjustable. It is reasonable to assume that the power of the pilot signal is large at the beginning of transmission and decreased after the adaptive weighting convergence. So the reference signal can be the pilot signal at the beginning and then switched to the tentative decision, i.e., in decision directed mode. Usually, the antenna number is small [2, 3] and so the (inverse) QR decomposition operating in symbol interval is more suitable.

Square-root-free algorithms for QR decomposition are often used and here the SGR algorithm [7, 10] with complex input is considered as an example. This algorithm required floating-point representation and division operation is also involved. It can be implemented in ASIC/FPGA or DSP. For a general DSP, division operation is calculated recursively and we assume it to be equal to 15 multiplication/addition (MA) operations. Then for the antenna number M, a summary of arithmetic operations is given in Table 1 where the back-substitution and inverse-QR decomposition are also considered.

By using back-substitution, the weights can be updated. The QR array is updated whenever new symbol data is input, but the antenna weights can be updated in a given number of symbol intervals.

For the inverse QR decomposition, we can assume SGR can again be used. The required number of weights updating operations is less than the back-substitution, but the matrix-vector multiplication requires an extra $2M(M+2)$ operation as shown in Table 1. So the inverse QR decomposition is computation intensive because matrix-vector multiplication is required whenever new symbol data is input.

In WCDMA, the chip rate is 3.84M chips/second and the spreading gain ranges from 4 to 256, so the symbol rate ranges from 960k to 15k symbols/second. Given the antenna number M, user number K, path number L, symbol rate B and weights updating rate D, Table 2 gives the required performance for different cases.

From Table 2, it is clear that the required throughput of QR decomposition is less than that of inverse QR decomposition. Currently, floating-point DSPs with performance up to 1 GFlops are available, e.g., TMS320C6713. Furthermore, a dedicated division unit implemented in FPGA can be used as an accelerator because the computation load of the DSP can be reduced by nearly 30 percent, as is shown in the last column of Table 2. However, this is only the case of moderate load. For the heavy load case, for example, the user number is 60 and the baud rate is still 60k, then it will require multiple DSPs for adaptive beamforming. In this case, a dedicated beamforming chip is a good choice.

In summary, an RLS algorithm based on QR decomposition can be implemented using a DSP (or with the aid of an FPGA) for adaptive beamforming. If further performance is required, a dedicated beamforming ASIC/FPGA chip will be needed.

REFERENCES

1. J. Litva and T. K. Lo, Digital beamforming in wireless communications. Boston, MA: Artech House, 1996.
2. S. Tanaka, A. Harada and M. Sawahashi, “Experiments on coherent adaptive antenna array diversity for

wideband DS-CDMA mobile radio," IEEE J.SAC, vol. 18, no. 8, pp. 1495—1503, Aug. 2000
3. Y. S. Song, H. M. Kwon and B. J. Min, "Computationally efficient smart antennas for CDMA wireless communications," IEEE Trans. Veh. Technol., vol. 50, no. 6, pp. 1613—1627, Nov. 2001
4. S. T. Alexander and A. L. Ghirnikar, "A method for recursive least squares filtering based upon an inverse QR decomposition," IEEE Trans. Signal Processing, vol. 41, no. 1, Jan. 1993
5. H. V. Poor and X. Wang, "Code-aided interference suppression for DS/CDMA communications – part II: parallel blind adaptive implementations," IEEE Trans. Commun. Vol. 45, no. 9, pp. 1112 –1122, Sept. 1997
6. J. G. McWhirter, "Recursive least-square minimization using a systolic array," in Proc. SPIE Real Time Signal Processing VI, pp. 105–112., Aug. 1983
7. G. Lightbody, R. Walke, R. Woods and J. McCanny, "Linear QR architecture for a single chip adaptive beamformer," Journal of VLSI signal processing system. Vol. 24, pp. 67–81, 2000
8. Jun Ma, K. K. Parhi, and E. F. Deprettere, "Annihilation-reordering look-ahead pipelined CORDIC-based RLS adaptive filters and their application to adaptive beamforming," IEEE Trans. on Signal Processing, vol. 48, no. 8, Aug. 2000, pp. 2414-2431
9. Shang-Chieh Liu and E. Geraniotis, "The blind QRD-DMS beamformer and its VLSI systolic designs for DS/CDMA systems," Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing, vol. 6, 1998, pp. 3189-3192
10. E. N. Frantzeskakis and K. J. R. Liu, "A class of square root and division free algorithms and architectures for QRD-based adaptive signal processing," IEEE Trans. on Signal Processing, vol. 42, no. 9, Sept. 1994, pp. 2455-2469

Table 1. Summary of arithmetic operations related with the antenna number M
(Counted according to real operation)

	Arithmetic operations	Equivalent operations ($1\ Division = 15\ MAs$)
QR decomposition — N_{QR}	M divisions and $12M + 9M(M+1)$ MAs	$9M(M+4)$
Back-substitution — N_{SB}	$2M$ divisions and $4M^2 + 5M$ MAs	$4M^2 + 35M$
Equation (3) — N_{kw} ($\mathbf{k}(n)$ and $\mathbf{w}(n)$ updating)	$2M$ divisions and $17M$ MAs	$47M$
Matrix-vector multiplication N_{MV}	$2M(M+2)$	

Table 2. Required performance for different cases

	M	L	K	B	D	Equivalent MA	Pure MA only (no division)
QR decomposition $(N_{QR}B + N_{SB}D)LK$	4	3	30	60k	16k	1850 MFlops	1350 MFlops
	4	3	2	960k	6k	1670 MFlops	1320 MFlops
Inverse QR $((N_{QR} + N_{MV})B + N_{kw}D)LK$	4	3	30	60k	16k	2085 MFlops	1590 MFlops