Development of a Smart-Antenna Test-bed, Demonstrating Software Defined Digital Beamforming

Taco Kluwer, Kees Slump, Roel Schiphorst, Fokke Hoeksema

University of Twente, Department of Electrical Engineering,
Laboratory of Signals and Systems,
P.O. box 217 - 7500 AE Enschede – The Netherlands
Phone: +31 628804388
E-mail: r.schiphorst@el.utwente.nl

Abstract—**This paper describes a smart-antenna test-bed consisting of 'common of the shelf' (COTS) hardware and software defined radio components. The use of software radio components enables a flexible platform to implement and test mobile communication systems as a real-world system. The test-bed is configured to demonstrate the concept of a smart antenna receiver using digital beamforming.**

The system consists of high-speed analog to digital converters (ADC), a digital signal processor (DSP) and a real-time operating system, combined with radio algorithms in software. Programmable signal generators generate signals at intermediate frequencies (IF), which are sampled by the ADCs. The sampled IF signal is transferred efficiently to the DSP where a software defined digital downconverter translates the signal to the baseband.

To demonstrate the test-bed for smart antennas a Constant Modulus (CM) algorithm is implemented. This algorithm recursively computes and updates the weight factors for the signals arriving at different antennas. The resulting optimal weight vector forms a spatial filter that can remove interfering signals arriving at an angle different from the desired signal.

keywords— **smart antennas, digital beamforming, software-defined radio**

I. INTRODUCTION

In a wireless communication system, desired signals can be distorted by interfering signals, that arrive at the antenna of the receiver. If the interfering signal is arriving at an angle different from the desired signal, the interference could be reduced if the antenna is insensitive in the direction of the interferer and more sensitive in the direction of the desired signal. This leads to the idea of the smart antenna. The smart antenna is able to change its sensitivity in a certain direction, using multiple antennas. The basic idea behind smart antennas is that multiple antennas, processed simultaneously, allow static or dynamical spatial processing with a fixed antenna topology. The pattern of the antenna in its totality, is now depending partly on its geometry and partly on the signal processing of the signals of the antennas. Smart antennas enable beamforming, to aim at targets or pattern modification to optimize the signal to noise performance or power consumption.

Many algorithms exist for smart antenna systems and a variety of applications are possible [1]-[4], [8]-[10]. To understand the quality of the various algorithms in a real-world system and with that, to estimate the potential of smart antennas, a test-bed is a good starting point for evaluation. This paper describes the development of a test-bed for smart antennas. A very detailed description of the development process is given in [21].

The paper starts with a description of the antenna array and its properties in section II. Adaptive beamforming with the CM algorithm is introduced in section III. The system design concept is based on the heterodyne receiver concept, which is shown in section IV. This concept is implemented with COTS hardware and real-time software, and described in section V. The test results that demonstrate the test-beds operation are presented in section VI.

II. ARRAY BASICS

Fig. 1 shows the concept of the smart antenna. The smart antenna is basically a set of receiving antennas in a certain topology. The signal $x(t)$ is received by the antenna elements and it is multiplied with the coefficients in a weight vector w, adjusting phase and amplitude. Summing up the weighted signals, results in the array output $y(t)$. The signals and weight factors are complex.
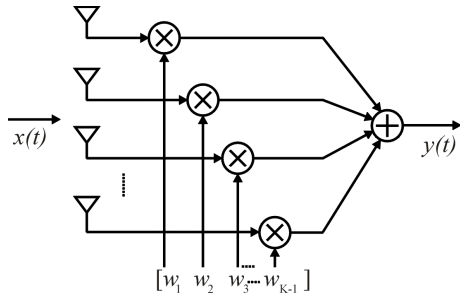
Fig. 1. Smart antenna concept for receiving antenna

Many topologies exist for the smart antennas, but in this paper only the linear array, which is shown in Fig. 2 is discussed. In this figure, d is the distance between the antennas and θ is the angle at which the wave front arrives. The number of antenna elements is K and the index of an antenna element is denoted with k.
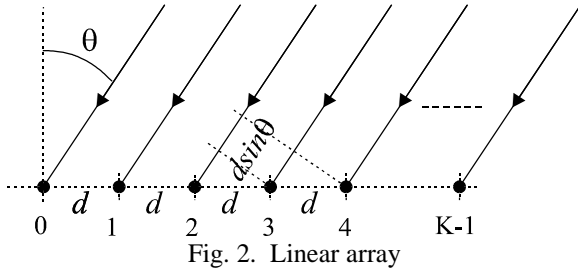

Fig. 2. Linear array

The mathematical foundations in this part of the paper on the linear array can be found in [1]. If the wavefront arrives at the array antenna as shown in Fig. 2, the wavefront arrives earlier at antenna element $k+1$ than element k. The difference in length between the paths is $d\sin\theta$. If the arriving signal is a harmonic signal, the signal arriving at antenna $k+1$ is leading in phase compared with antenna k. The signal that arrives at antenna element zero is considered to have a phase lead of zero. The signal that arrives at antenna k, leads in phase with $\xi kd\sin\theta$, where $\xi = 2\pi/\lambda$, where λ is the wavelength.

This leads to the so-called array propagation vector defined by:

$$v = \begin{bmatrix} 1 & e^{j\xi d\sin\theta} & \cdots & e^{j(K-1)\xi d\sin\theta} \end{bmatrix}^T \quad (1.1)$$

This vector contains the information of the arrival of the signal. The weight vector is defined by:

$$w = \begin{bmatrix} w_0 & w_1 & \cdots & w_{K-1} \end{bmatrix}^T \quad (1.2)$$

The array factor is defined by:

$$F(\xi,\theta) = \frac{y(\xi,\theta)}{x(\xi,\theta)} = w^T v \quad (1.3)$$

The array factor is the response of the signal arriving from angle θ; $y(\xi,\theta)$ and $x(\xi,\theta)$ are respectively the output and input of the beamforming array. If we consider ξ and d being fixed parameters of the antenna, chosen for a given frequency, combining (1.3) with (1.1) and (1.2) gives:

$$F(\theta) = \sum_{k=0}^{K-1} w_k e^{j\xi kd\sin\theta} \quad (1.4)$$

where k is the index for the antenna element.

As an example to demonstrate beamforming principle, assume the weight factor is defined by:

$$w_k = A_k e^{jk\alpha} \quad (1.5)$$

If a signal arrives at the antenna array at an angle θ_0, it is clear that the array response can be made maximal in that direction by adjusting the phase of the complex weight for:

$$\alpha = -\xi d\sin\theta_0 \quad (1.6)$$

Fig. 3 shows the effect of an array response of a linear array with 8 antennas with the beam steered to θ_0 for zero and 45 degrees. With 8 antennas a maximum of 7 nulls can be formed.
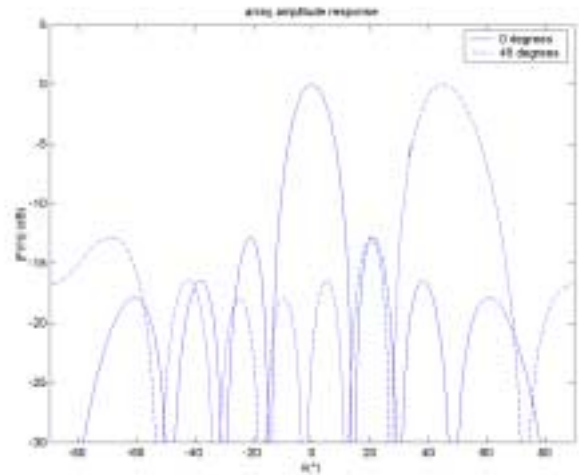

Fig. 3. Antenna response with beam steered to 0 and 45 degrees

III. ADAPTIVE BEAMFORMING

An adaptive beamformer is able to automatically update the weight vector, in order to separate desired signals from interfering signals. Adaptive beamforming can be done in many ways. Many algorithms exist for many applications, varying in complexity. A generic adaptive beamformer is shown in Fig. 4. The weight vector w is calculated using the signal $x(t)$ received by multiple antennas. An adaptive processor will minimize the error $e(t)$ between a desired signal $d(t)$ and the array output
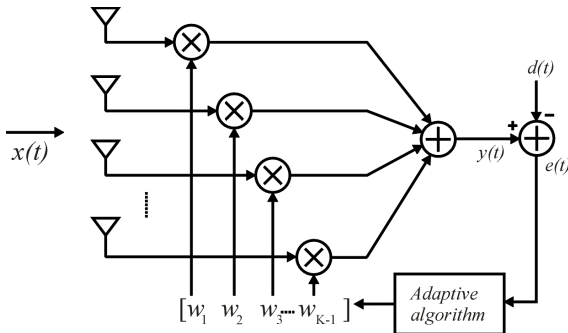
y(t).

Fig. 4. Concept of the adaptive beamforming

One of the most simple algorithms for adaptive processing is based on the Least Mean Square (LMS) error. Although the complexity of the algorithm is very low, its results are satisfying in many cases. The algorithm is very stable and it needs few computations, which is important for system implementation. The computational power of many systems is limited and should be managed wisely.

The LMS algorithm uses the steepest descend method to find the minimum error that is given by [1]:

$$\hat{w}(n+1) = \hat{w}(n) + \mu x(n)\varepsilon^*(n) \qquad (1.7)$$

Here, $x(n)$ is representing the signal $x(t)$ received by the antenna array elements. The estimation of optimal vector w, is done by updating the previous estimation with a factor which consists of a gain constant μ, the received signal $x(n)$ and the complex conjugate of error $\varepsilon(n)$ that is defined by:

$$\varepsilon(n) = d^*(n) - y(n) \qquad (1.8)$$

Array output $y(n)$ is defined by the $w^H x(n)$, where $(.)^H$ is the Hermetian transpose. The knowledge of the received signal (the reference signal $d^*(n)$) eliminates the need for beamforming, but the reference can also be a vector that is partly known, or correlated with the received signal. For example, the training sequence in the GSM standard, intended for channel equalization, could be used for beamforming. A disadvantage for an implementation of LMS in a system, is the fact that the algorithm requires the reference signal $d(t)$ and the received signal $y(t)$ to be aligned by a synchronization process.

When the adaptive algorithm is not using a reference, but statistic information of the signal, it is called blind beamforming. The Constant Modulus (CM) algorithm, for example, uses the knowledge that the modulus of the signal is constant. There are many modulation schemes where the modulus is kept constant, among which mini-

mum shift keying (MSK). MSK is used in Section VI to demonstrate the test-bed for beamforming.

The CM algorithm tries to minimize the cost function given by[6]:

$$J_p(W) = \frac{1}{2p} E\left\{\left(|y|^p - 1\right)^2\right\}, p \in \{1, 2, ...\} \quad (1.9)$$

where E{.} represents the statistical expectation. The algorithm to find the minimum, uses the same iterative relation as for the LMS algorithm. For $p = 1$ in (1.9), the error in (1.7) is given by [7]:

$$\varepsilon(n) = \frac{y(n)}{|y(n)|} - y(n) \qquad (1.10)$$

MATLAB simulations demonstrated satisfactory operation of the algorithm and therefore, the CM algorithm was used for the demonstration of the test-bed, which is described in the next section.

IV. DESIGN CONCEPT

With digital beamforming in a wireless communication system, the received signals must be available as complex digital data. Therefore a radio receiver must convert the received RF signals to digital baseband signals, for every antenna. A suitable receiver architecture is the heterodyne receiver. The heterodyne receiver is characterized by multiple stages for downconversion. The heterodyne receiver as part of the smart antenna receiver is visible in Fig. 5. In this case the first stage converts the incoming RF signal to an IF signal by using an analog receiver. In Fig. 5, this is represented schematically by the mixers, bandpass filters and local oscillators. The necessary RF filters and amplifiers have been left out for simplicity of the figure. The second stage is formed by digital hardware, where the IF signals are converted to the baseband, represented by an in-phase and a quadrature component.
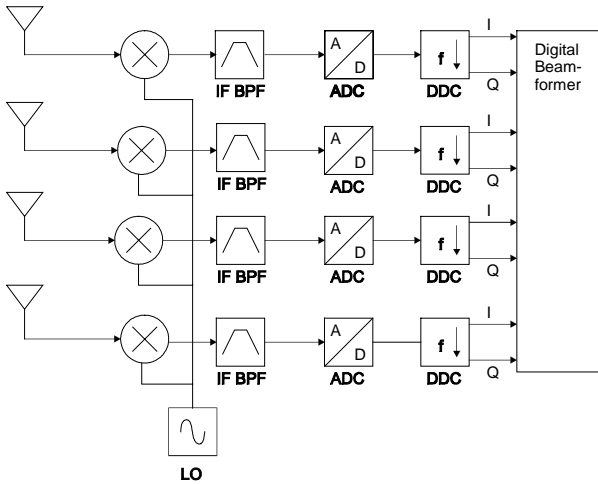
Fig. 5. Design concept of the digital beamformer

The design was focussed on the digital part, to avoid the difficult analog RF hardware. Therefore the analog front-end is replaced by programmable function generators, which are able to simulate the intermediate frequencies. This simulation is done by uploading arbitrary waveforms that represent the IF signals of the front-end receivers. An advantage of this set-up is that the testing of the digital hardware can be done for many aspects by simply uploading different IF signals.

V. SYSTEM IMPLEMENTATION

The system that is implemented can be split up into hardware and software components. The hardware consists of programmable function generators, a DSP board and a compatible multi-channel ADC. The software objects are implemented as real-time components.

A. Hardware Components

The AD converters in Fig. 5 are implemented by an AD converter of Texas Instruments (TI), the THS1206. Specifications of the THS1206 can be found in [13]-[15]. The digital downconverter and the beamform algorithm are implemented in software, running on an evaluation module for the TI TMS320C6711 DSP. The AD converter and DSP are compatible. The configuration is shown in Fig. 6.

The THS1206 is a quad channel ADC, that samples the IF signals from the HP33120A function generators by the sample and hold units simultaneously. These values are then multiplexed and sampled by the AD converter. The DSP's EDMA controller is triggered by the external interrupt coming from the ADC. On the interrupt the EDMA controller copies the data in burst mode from the AD converter's FIFO to an assigned memory block. When this buffer is full, the EDMA controller generates an interrupt, which indicates that the data in

the buffer is ready for processing by the software objects. To prevent reading and writing to the buffer at the same time, the buffer is implemented based on the ping-pong principle. The data from the ADC is successively copied to the ping and pong buffer. The buffer in which the EDMA controller is not writing is available to be processed by the algorithms on the DSP. The EDMA controller will indicate to the software which buffer is allowed to be processed. This way continuous operation on the incoming data can be guaranteed. The EDMA controller relieves the processor from the intensive data acquisition task. Details on the DSP and peripheral hardware can be found in [16] and [17]. The overall performance is represented in Table I.
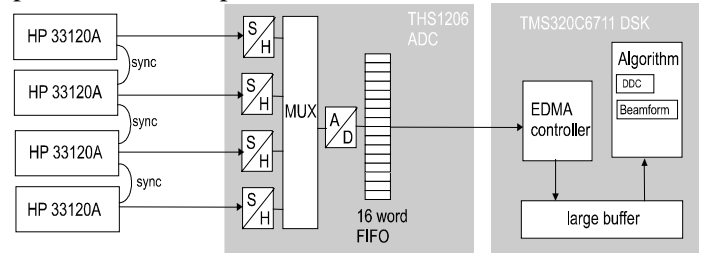

Fig. 6. Hardware set-up of the test-bed

TABLE I
PERFORMANCE SPECIFICATION

Maximum bandwidth 1 channel (Nyquist)	3 MHz
Maximum bandwidth 2 channels (Nyquist)	1.5 MHz
Maximum bandwidth 3 channels (Nyquist)	1 MHz
Maximum bandwidth 4 channels (Nyquist)	750 kHz
Resolution	12 bit
System clock	150 MHz
Maximum input frequency	54 MHz
Input signal level	-1V to 1V
Floating point operations per second	900 MFLOPS

B. Software Components

The digital algorithms for the demonstration of the test-bed are defined in software objects. The digital downconverter and the beamform algorithm are configured to work within a real time operating system. For the development of the software objects Code Composer Studio and the real-time kernel DSP/BIOS II is used. Details on the development environment can be found in [18]-[20]. This environment provides tools to configure, monitor and debug real-time software.
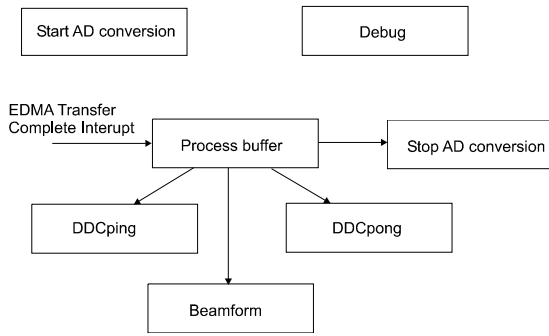
Fig. 7. Overview of the software objects

The software objects that are used in the test-bed can be found in

Fig. 7 and are described by:

1) Start AD conversion

This object configures the AD converter and performs the programming of the EDMA controller. The EDMA controller is programmed to copy data from the AD converter and to generate and interrupt when ready.

2) Process buffer

This object is triggered by an external interrupt, generated by the EDMA controller. The EDMA controller is configured to use two buffers, called a ping and a pong buffer. The 'Process buffer' object is a hardware interrupt service routine, and its functionality is restricted. The routine checks whether the ping or pong buffer is ready and depending on this it will start the DDC and beamform objects.

3) DDC ping/pong

This object performs the actual downconversion of the channels. It operates either on the ping or the pong buffer. The digital downconversion consists of digital mixers and FIR filters. The input is either the ping or pong buffer, and its output consists of the I/Q signals for different channels.

4) Beamform

This object will perform the actual beamforming algorithm. For this the CM algorithm is used. The input consists of the I/Q signals for different channels. Its output is one I/Q signal representing the array output.

5) Debug

The 'Debug' object is a thread that can be used for various debug functions. It can be used to halt the system, by placing breakpoints in the thread. The other objects can call the debug object when an error occurs, and the system is halted.

VI. TEST RESULTS

The beamform algorithm is implemented for two channels and therefore one interferer can be suppressed.

Convergence of the CM algorithm will be depending on the initial weight vector. This is proven by experiments, using two MSK signals that arrive at different angles. As stated before the received signals are simulated at IF by programmable function generators. The following test set-up is used:

- Two function generators, simulating the IF signals of two channels
- Two channels, each sampled at 521 kS/s
- ADC speed operating at 1042 kS/s
- IF frequency selected at 130250 Hz
- MSK modulated test vectors, 130250 bits/s

A. Experiment 1

Two signals are generated to simulate the two antennas and receivers which are a combination of the signals specified in Table II. The initial weight vectors are chosen so that initial array response is slightly steered towards the desired signal. This enables the CM algorithm to convert to a local minimum, by suppressing the interferer.

After beamforming, the resulting weight factors are used in MATLAB to plot the antenna response pattern. Fig. 8 shows that the weight coefficients, that are found, suppress the interferer at around 18 degrees. The suppression is around 12 dB. This makes the signal to interference ratio 18.5 dB as the desired signal at zero degrees has a gain of 6.5 dB.

TABLE II
PROPERTIES OF THE RECEIVED SIGNALS FOR EXP. 1

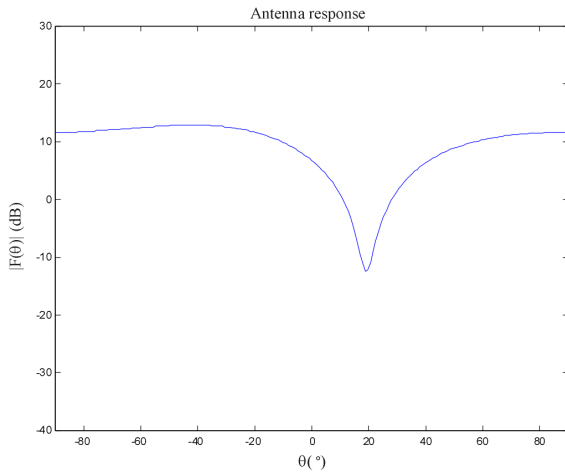| Desired Signal	Interfering signal
MSK modulated signal	MSK modulated signal
150 bit test vector	150 bit test vector
Long lengths with same bits	Frequent changes of bits
Arriving at zero degrees	Arriving at 18 degrees

Fig. 8. Antenna response, showing suppression of the interferer at 18 degrees after beamforming

The same experiment is repeated for an interfering signal arriving at –37 degrees. The results are represented in Fig. 9. The suppression seems better then the previous experiment. The minimum of the antenna response is not exactly at –37 degrees, which reduces suppression to 21 dB. Note that the signal at zero degrees has a 6.5 dB gain which makes the signal to interference ratio 27.5 dB. This indicates that even a small mismatch of the minimum of the antenna response, can have strong influence on the suppression.
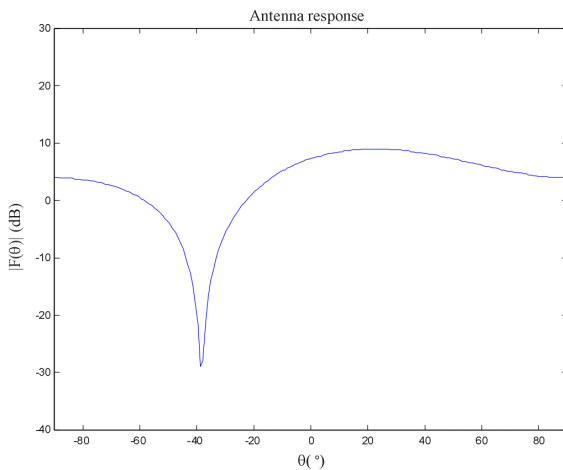

Fig. 9. Antenna response, showing suppression of the interferer at –37 degrees after beamforming

B. Experiment 2

For the suppression of a signal that arrives close to zero degrees, the initial weight vectors are chosen such that the initial antenna response is slightly steered towards the desired signal at –37 degrees.

Table III
PROPERTIES OF THE RECEIVED SIGNALS FOR EXP. 2

Desired Signal	Interfering signal
MSK modulated signal	MSK modulated signal
150 bit test vector	150 bit test vector
Long lengths with same bits	Frequent changes of bits
Arriving at –2.5 degrees	Arriving at –37 degrees

After beamforming, the weight vectors that are found are used to plot the antenna response pattern, shown in Fig. 10. The antenna pattern has its minimum at –4 degrees, and at –2.5 degrees the gain is around -10 dB. Signal to interference ratio is 16.5 dB to the gain at –37 degrees.

Both experiments indicate proper operation of the beamforming. With these experiments the concept of beamforming is demonstrated in a real system.

VII. CONCLUSIONS

To be able to test algorithms and systems for wireless communications in general and smart antennas in particular, a flexible test-bed was designed. The test-bed was formed by general purpose hardware, combined with a flexible software development environment. The real-time operating system enables evaluation of the behavior of digital algorithms and real-world systems.
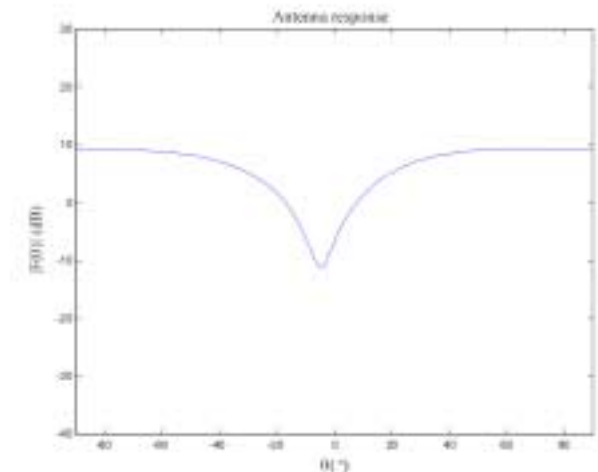

Fig. 10. Antenna response, showing suppression of the interferer at–2.5 degrees after beamforming

The test-bed demonstrates proper operation of the CM algorithm used for digital beamforming, combined with digital downconversion. It was shown that a small mis-

match in the minimum of the antenna response has strong influence on the suppression of the interferer. Furthermore, selection of one of two MSK signals is possible by choosing correct initial weight vectors.

REFERENCES

[1] J.Litva. "Digital Beamforming in Wireless Communications", 1996

[2] L.C.Godara, "Applications of Antenna Arrays to Mobile Communications, part I: Performance improvement, Feasibility, and System Considerations", *Proc. IEEE*, Vol.85, No. 7, pp. 1031-1060, July 1997.

[3] L.C. Godara, "Application of Antenna Arrays to Mobile Communications, Part II: Beam-Forming and Direction of Arrival Considerations", *Proc. IEEE*, Vol 85, No. 8, pp. 1195-1245, August 1997.

[4] B. Widrow, P.E. Mantey, L.J. Griffiths, B.B. Goode, "Adaptive Antenna Systems", *Proc IEEE*, Vol. 55, No.12, pp.2143-2159, December 1967

[5] A. van der veen, "An Analytical Constant Modulus Algorithm*", IEEE Trans. Signal Processing*, Vol. 44, No. 5, pp. 1136-1155, May 1996.

[6] M. Rupp, A. H. Sayed, "On the Convergence of Blind Adaptive Equalizers for Constant Modulus Signals", *IEEE Trans, Commun.*, Vol. 48, No. 5, May 2000.

[7] S. Gummadi, B.L. Evans, "Co-Channel Signal Separation In fading Channels Using A Modified Constant Modulus Array", University of Texas at Austin.

[8] Wireless Personal Communications for the 21st Century: European Technological Advances in Adaptive Antennas", *IEEE Commun. Mag.* pp. 102-109, September 1997.

[9] S. Ponnekanti, "An Overview of Smart Antenna Technology for Heterogeneous Networks", *IEEE Commun. Surveys*, Vol. 2, No. 4, pp. 14-23, Fourth Quarter 1999.

[10] Z. Kostic, S. Seetharaman, "Digital Signal Processors in Cellular Racio Communications", *IEEE Commun. Mag.* Vol. 35, No. 12, pp. 22-35, December 1997.

[11] M.E. Frerking, "Digital Signal Processing in Communication Systems", 1994

[12] C.B. Papadias, D.T.M.Slock, "New Adaptive Blind Equalization Algorithms for Constant Modulus Constellations", Institute EURECOM

[13] Texas Instruments, "THS1206, THS12082, THS 10064, THS10082 Evaluation module", may 2000

[14] Texas Instruments, "Designing With the THS1206 High-speed Data Converter", April 2000

[15] Texas Instruments, "THS1206 12-bit 6MSPS, Simultaneous sampling analog-to-digital converters", May 1999

[16] Texas Instruments, "TMS320C6000 Peripherals Reference Guide", February 2001

[17] Texas Instruments, "TMS320C6000 Technical Brief", February 1999

[18] Texas Instruments, "TMS320C6000 DSP/BIOS User's Guide", May 1999

[19] Texas Instruments, "TMS320C62x/67x Programmer's Guide", May 1999

[20] Texas Instruments, "Code Composer Studio, User's Guide", May 1999

[21] T. Kluwer, "Development of a Smart Antenna Test-bed, Using Digital Beamforming", University of Twente, M.Sc. Thesis, 2001.