Chapter 6

An Auto-Calibrating Adaptive Array for Mobile Telecommunications

6.1 Introduction

Adaptive antenna arrays have a well documented ability to cancel interfering signals. Many high resolution algorithms which use information of the direction of arrival (DOA) of signals have been developed. To work well in a practical situation those methods need a very accurate characterization of the array, including antenna gain and phase response, positions of the antenna elements and receiver equipment effects, and so forth. This calls for accurate calibration methods. Some algorithms have been proposed which alleviate the sensitivity of these high resolution algorithms to calibration errors, see e.g. [39]. To avoid the problem with accurate calibration one can use a direct algorithm that combines the signals from the antenna elements to optimize some criterion such as the signal to interference plus noise ratio (SINR) [33]. These types of adaptive algorithms are insensitive to differences in amplitude loss and phase shifts in the antenna elements, transmission lines and sampling receivers for each individual antenna path.

Adaptive antennas are either using weighting of the antenna signals in a digital signal processor (DSP) or an analog weighting technique with analog attenuators and phase shifters that modulate the signals at radio frequency (RF). The advantage of the analog beamformer is that it can easily be concatenated to an ordinary base station. The output is just a spatially filtered RF signal that can be connected to the existing base station receiver. Normally the analog beamformer uses weights with larger quantization levels than its digital counterpart. The weights are also

calculated on signals at the analog-to-digital (A/D) converters but are applied to the signals at the analog weighting units. This is, of course, undesirable and consequently, there is a need for calibration to know the amplitude and phase shifts between the A/D converters and the weighting units for each signal path. This calibration is performed prior to start-up of the adaptive antenna and the calibration data is stored as look-up tables in the DSP. However, due to temperature drift in the receivers and weighting hardware, the look-up table becomes inaccurate and the performance of the adaptive antenna degrades.

The temperature drift has been studied for a weighting unit, in use in the adaptive antenna testbed described in Chapter 4. The phase and amplitude drifts were measured with a network analyzer for ten hours after a ”cold startup”. See Figure 6.1. After the initial two hours the drift is approximately 0.1 dB and 1 degree per hour. Another study of temperature effects in adaptive arrays are reported in [26].

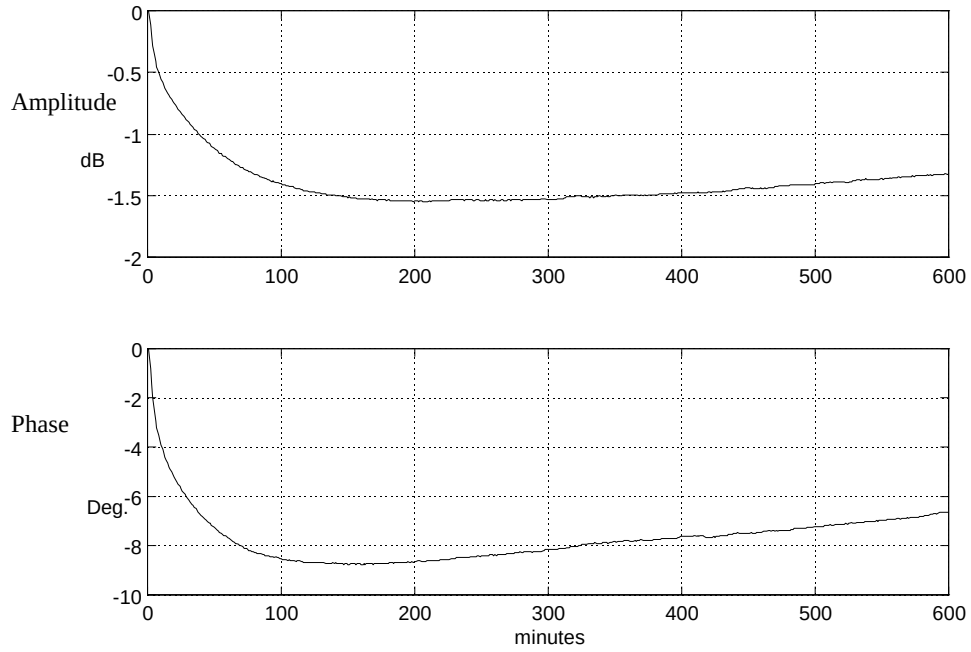

Fig. 6.1. Temperature drift of a hardware weight.

In this chapter, two methods for auto-calibration are proposed, where the weights are adjusted during operation in order to mitigate the temperature drift. In Section 6.2 the antenna model under study is introduced. Section 6.3 describes the auto-calibration algorithms and Section 6.4 presents some simulation results.

6.2 Description of the Antenna Model

A block diagram of an analog beamformer is depicted in Figure 6.2. The signals from the antennas are split into a receiving path and a weighting path. The adaptive weights are calculated based on the sampled baseband signal in each receiving path, whereas the beamforming is performed on the RF signals of the weighting path, by means of hardware weights. The hardware weights may be composed of phase shifters and attenuators. Other approaches can also be taken, see [28].

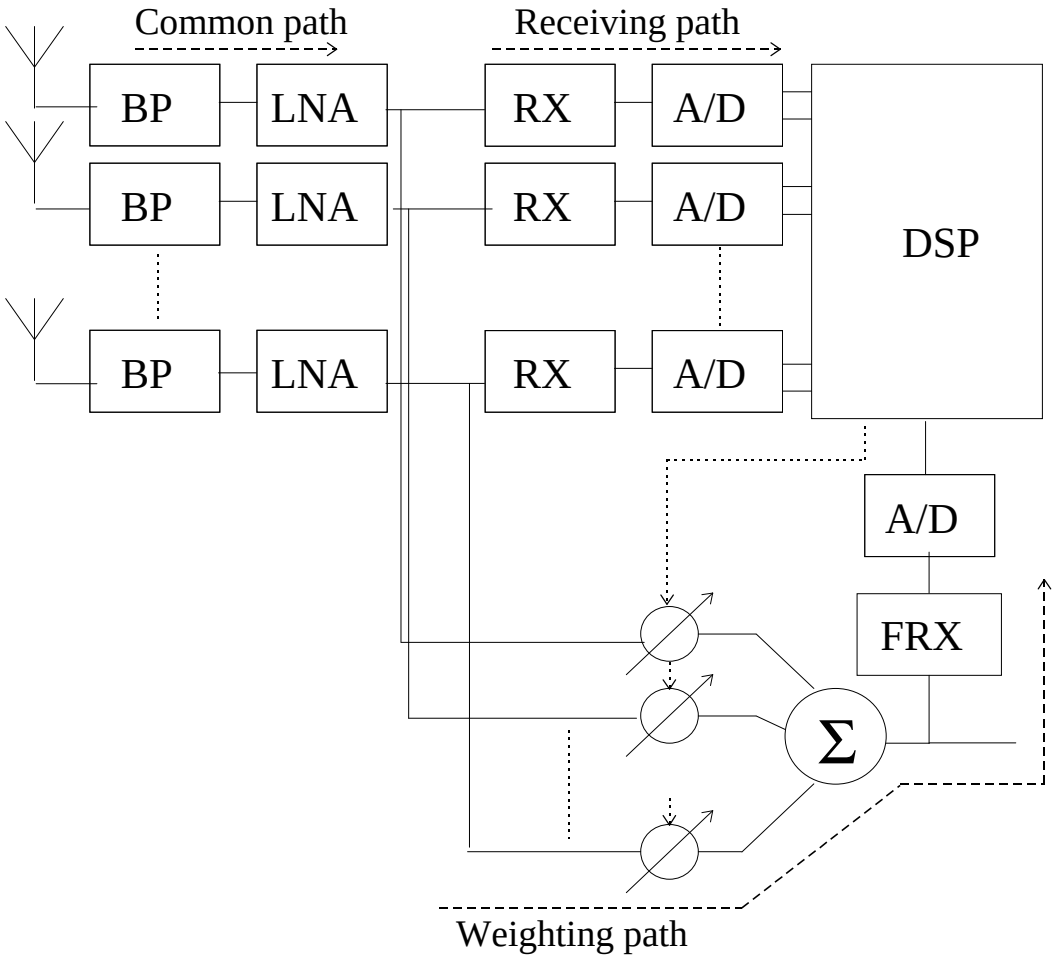

Fig. 6.2. Block diagram of an adaptive antenna utilizing analog beamforming with digitally calculated weights. (BP=band-pass filter, LNA=low-noise amplifier, RX=receiver, FRX=feedback receiver)

The example antenna presented here uses the Sample Matrix Inversion (SMI)-algorithm to calculate the adaptive antenna weights. A reference signal y present in each transmitted burst, e.g. the training sequence in Time Division Multiple Access (TDMA)-systems like GSM and IS-54, is utilized to form a lest-squares problem. The sampled baseband signals of

the antenna elements form a column vector denoted $\mathbf{x}(k)$. Consider Figure 6.3.

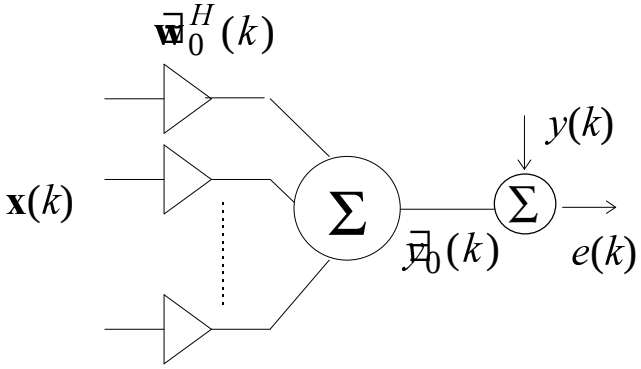

Fig. 6.3. Complex weighting of the antenna signals.

The optimal weight vector $\mathbf{w}_0$ minimizing the mean square error of difference between the reference signal $y(k)$ and the output $\hat{y}_0(k)$ is obtained by solving the Wiener-Hopf equations [20]

$$\mathbf{R}_{\mathbf{xx}}\mathbf{w}_0 = \mathbf{R}_{\mathbf{x}y} \tag{6.1}$$

where the covariance matrices are defined as

$$\begin{aligned} \mathbf{R}_{\mathbf{xx}} &= E\left\{\mathbf{x}\mathbf{x}^H\right\} \\ \mathbf{R}_{\mathbf{x}y} &= E\left\{\mathbf{x}y^*\right\} \end{aligned} \tag{6.2}$$

Here E denotes the expected value, ^{H} complex conjugate transpose and * complex conjugate. The covariance matrices may be estimated by

$$\begin{aligned} \hat{\mathbf{R}}_{\mathbf{xx}} &= \frac{1}{M}\sum_{k=1}^{M}\mathbf{x}(k)\mathbf{x}^H(k) \\ \hat{\mathbf{R}}_{\mathbf{x}y} &= \frac{1}{M}\sum_{k=1}^{M}\mathbf{x}(k)y^*(k) \end{aligned} \tag{6.3}$$

where M is the number of reference symbols. The estimate of the weight vector is then given by

$$\hat{\mathbf{w}}_0 = \hat{\mathbf{R}}_{\mathbf{xx}}^{-1}\hat{\mathbf{R}}_{\mathbf{x}y} \tag{6.4}$$

In what follows the differences in signal paths between the antennas and the split point (denoted common path in Figure 6.1) will be neglected. This is due to the choice of a direct algorithm such as SMI, where there is no need to know the array manifold. Phase and magnitude differences in the receiving path and weighting path corresponding to the same antenna element must however be compensated for, since the adaptive antenna weights are calculated based on the signals of the receiving paths, whereas the beamforming is performed on the RF-signals of the weighting paths. This problem is easily mitigated by calibration. A sinusoid with an offset in frequency compared to the carrier is injected in each of the antenna paths, one at a time. The attenuation and relative phase of the weighting path compared to the receiving path is then measured and stored in a matrix $\mathbf{D}_0$ to be used later when the calculated weights are to be steered out to the hardware weighting units. It should be noted that there is a relation between the magnitude and phase of a general hardware weight. If the magnitude is adjusted the phase will change, and vice versa. This non-linear behavior of the weights are in general treated by creating a large look-up table with all possible weight settings as entries.

For practical reasons a time consuming calibration can only be performed prior to operation of the adaptive antenna. Since the temperature drift in the receivers and weighting units etc. will degrade the performance of the antenna, it would be desirable to perform some calibration simultaneously with the main operation. The calibration must be fairly fast to execute, yet still accurate enough to maintain a good performance of the overall antenna system, for a long period of time. This is the topic of the next section.

6.3 The Auto-Calibration Algorithms

6.3.1 An LMS-approach

This section outlines the auto-calibration algorithm and presents the assumptions made concerning the prior-to-operation-calibration and thermal drift of the involved hardware. The SMI-weights $\vec{\mathbf{w}}_0$ together with the sampled signals of the receiving paths **x'** are utilized to form a reference signal $\vec{y}_0(k)$ in the DSP. The hardware weights are adjusted to minimize the squared difference between the reference signal $\vec{y}_0(k)$ and the sampled output of the adaptive antenna $y_0(k)$. For that purpose we

shall use an LMS-like (least mean square) algorithm. The aim of the algorithm is to maintain a low value of the error signal $e(k) = \bar{y}_0(k) - y(k)$. Consider Figure 6.4, where $\mathbf{n}_1$ models the noise received by the array, as well as noise generated in the front-end. Furthermore, $\mathbf{n}_2$ models the noise generated in the receivers and in the A/D converters. Finally n_3 models the noise generated in the active components of the hardware weighting units and in the feedback receiver and associated A/D converter.

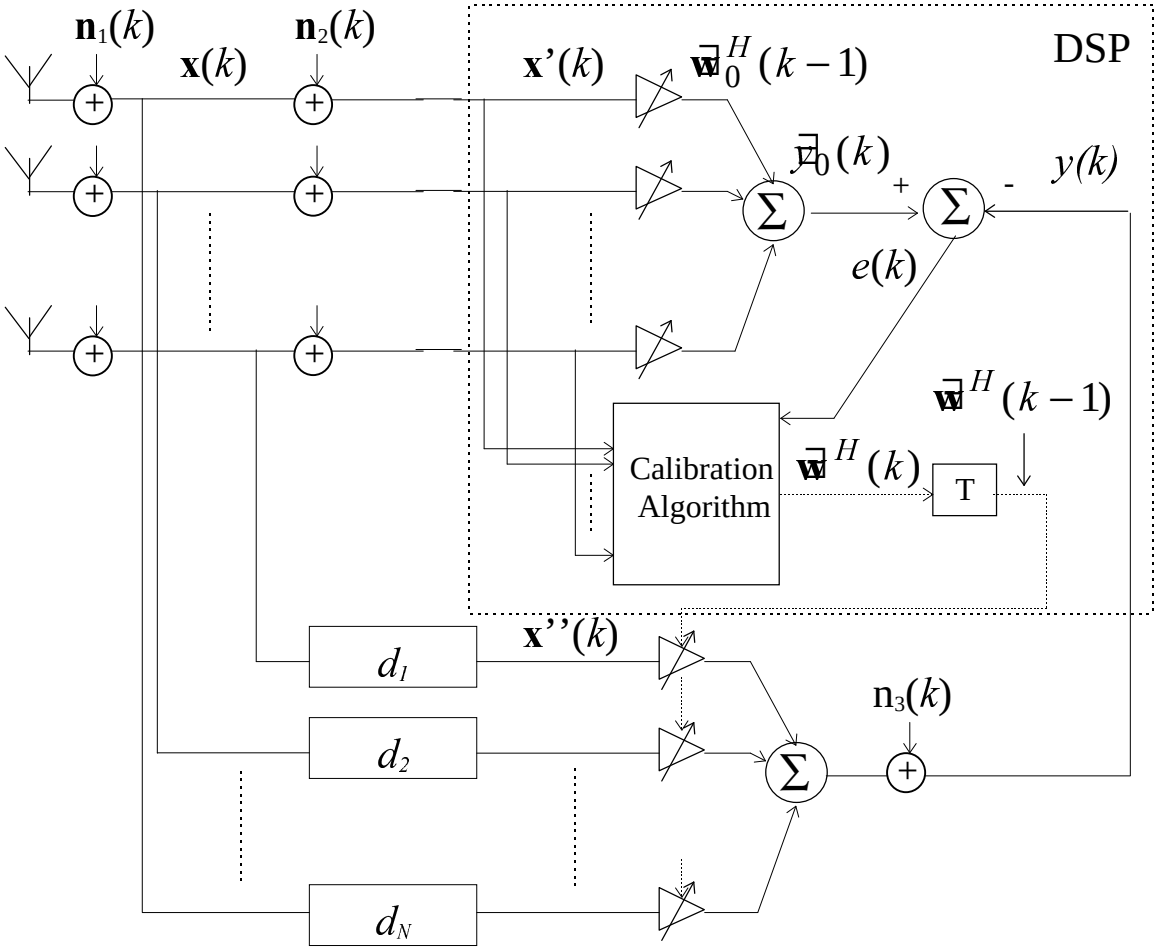

Fig. 6.4. Auto-calibration system. The dotted line defines the boundary of the DSP interface.

Let $\mathbf{x}''(k)$ be the input of the hardware beamformer. The signal vector $\mathbf{x}''(k)$ relates to $\mathbf{x}(k)$ according to

$$\mathbf{x}''(k) = \mathbf{D}\mathbf{x}(k) = \mathbf{D}\{\mathbf{x}'(k) - \mathbf{n}_2(k)\} \tag{6.5}$$

where $\mathbf{D}$ is a diagonal matrix with complex valued entries, given by

$$\mathbf{D} = \begin{bmatrix} d_1 & 0 & \Lambda & 0 \\ 0 & d_2 & \mathrm{O} & \mathrm{M} \\ \mathrm{M} & \mathrm{O} & \mathrm{O} & 0 \\ 0 & \Lambda & 0 & d_N \end{bmatrix} \tag{6.6}$$

If desired, leakage between the paths can be modeled by introducing off-diagonal elements in **D**. As mentioned in the previous section, the linear model of the relationship between **x**'' and **x** in (6.5) is an approximation, since **D** could actually be a function of $\vec{\mathbf{w}}$, i.e. $\mathbf{D} = \mathrm{f}(\vec{\mathbf{w}})$. However, in this study, we make the assumption that the coupling between the phase and magnitude part of the hardware weights is negligible, and therefore a weight-independent matrix **D** would be appropriate.

The Least Mean Square (LMS)-approach would require both **x**''(*k*) and *e*(*k*) to update the weight vector $\vec{\mathbf{w}}(k)$. This is however not possible since **x**''(*k*) can not be measured. Our approach is therefore to use **x**'(*k*) instead, i.e. the noisy measurement of **x**(*k*), and the matrix $\mathbf{D}_0$ obtained from the off-line calibration process. The proposed algorithm can be stated as follows:

Initialization:

$$\vec{\mathbf{w}}(0) = \mathbf{D}_0^{-H}\vec{\mathbf{w}}_0(0)$$

Algorithm:

$$\begin{aligned} y_0(k) &= \vec{\mathbf{w}}_0^H(k-1)\mathbf{x}'(k) \\ e(k) &= y_0(k) - y(k) \\ \vec{\mathbf{w}}(k) &= \vec{\mathbf{w}}(k-1) + \mu\mathbf{D}_0\mathbf{x}'(k)e^*(k) \end{aligned} \tag{6.7}$$

Since we are studying adaptive arrays utilizing analog beamforming, the weights are assumed to be applied to the data of the next TDMA frame. This should be a valid assumption since otherwise the weights have to be calculated and steered out during a time less than a symbol interval. This also implies that the time step in the algorithm is one TDMA frame.

The fact that we are using the calibration data from the off-line calibration in (6.7) will only slightly affect the performance of the gradient method since the LMS-approximation of the gradient is in itself very crude. Our desired instantaneous gradient direction is given by

$$\mathbf{x}''(k)e^{*}(k) \tag{6.8}$$

but we are instead using the approximation

$$\mathbf{D}_0\mathbf{x}'(k)e^{*}(k). \tag{6.9}$$

This renders in a noise term in the gradient estimate, given by

$$\{\mathbf{D}-\mathbf{D}_0\}\mathbf{x}'(k)e^{*}(k)-\mathbf{D}\mathbf{n}_2e^{*}(k) \tag{6.10}$$

An exact analysis of the influence of the properties of this error (size of $\mathbf{D}$-$\mathbf{D}_0$, noise color etc.) on the convergence of the algorithm remains to be investigated.

However, simulation studies presented in Section 6.4 show that the performance of the proposed algorithm will be satisfactory in a stationary signal environment, i.e. with a slowly varying SMI-weight vector. When the signal environment is nonstationary, the long convergence time inherent in LMS-like algorithms can be expected to create problems. The performance will then be degraded.

6.3.2 Tracking approach

An alternative on-line calibration method is to track the temperature drift of d_l, l=1,2,…,N. This approach is not affected by the rapid variations in a typical signal environment with fast fading and multipath, as opposed to the method presented above. On the other hand, the input $\mathbf{x}(t)$ of the array must be persistently exciting of sufficiently high order, i.e. all ”spatial modes” of interest must be excited in order to identify the parameters d_l.

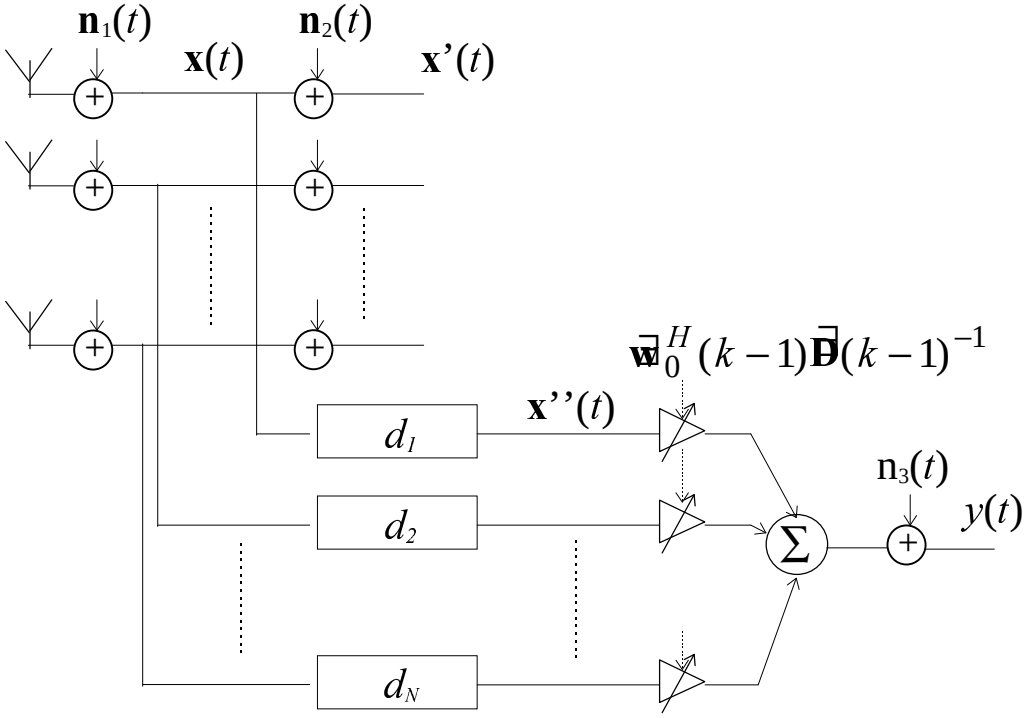

Fig. 6.5. Auto-calibration system using a tracking approach

The SMI-weights compensated by the inverse of the tracked diagonal matrix $\bar{\mathbf{D}}$ are steered to the hardware weighting units to eliminate the temperature drift. The identification of $\mathbf{D}$ is performed by forming a least-squares identification problem utilizing the noisy measurements of $\mathbf{x}(t)$, i.e. $\mathbf{x}'(t)$ and the sampled output $y(t)$. In order to identify the temperature drift the model in Figure 6.6 is used. The signal $\mathbf{x}'(t)$, the weight vector $\bar{\mathbf{w}}_0(k-1)$ and the off-line calibration data $\mathbf{D}_0$ are known, and can therefore be used to form the regressor $\mathbf{z}(t)$, given by $\mathbf{z}(t) = \begin{bmatrix} z_1(t) & z_2(t) & \Lambda & z_N(t) \end{bmatrix}^T$. This is possible since the system is linear.

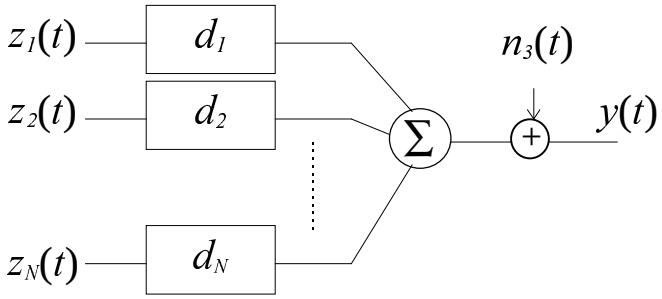

Fig. 6.6 Identification model.

The element $z_l(t)$ in the regressor vector is given by

$$z_l(t) = x_l'(t)\bar{w}_{0,l}^*(k-1) / \bar{d}_l(k-1) \tag{6.11}$$

and the least-squares estimated temperature drift by

$$\hat{\mathbf{d}}^{*} = \hat{\mathbf{R}}_{zz}^{-1}\hat{\mathbf{R}}_{zy} \tag{6.12}$$

where

$$\hat{\mathbf{d}} = \begin{bmatrix} d_1 & d_2 & \mathrm{K} & d_N \end{bmatrix}^T \tag{6.13}$$

and

$$\hat{\mathbf{R}}_{zz} = \frac{1}{M}\sum_{t=1}^{M}\mathbf{z}(t)\mathbf{z}^H(t) \tag{6.14}$$

$$\hat{\mathbf{R}}_{zy} = \frac{1}{M}\sum_{t=1}^{M}\mathbf{z}(t)y^*(t) \tag{6.15}$$

Above, M is the number of symbols used for the estimation of the covariance matrices. The estimated **d**:s are *low-pass filtered* in order to introduce a memory in the algorithm. This will reduce the variance of the regressors $z_l(t)$ and therefore improve the tracking of **d.** Experiences from the simulations show that the tracking ability is improved.

Using this indirect approach instead of the direct approach of Subsection 6.3.1, we are not restricted by the length of the training sequence, since we have samples of the input and output for the complete time slot. Mobile communication systems are often said to be interference limited and not noise limited. This means that the noise in **x** (and also in **z**) will be small as compared to the signal levels, and $\hat{\mathbf{R}}_{zz}$ will be badly conditioned if the number of *incoherent* rays impinging on the array is small. The tracking of the temperature drift will therefore not succeed. This can however be mitigated by using a few number of samples from several frames to estimate the covariance matrices. This is possible because the variations in **D** due to temperature is typically slow compared to the length of a frame. This approach is therefore likely to succeed if the signal scenario is time varying.

6.4 Simulation Study

A simulation study of the LMS calibration approach in Subsection 6.3.1 has been performed based on the models described in Section 6.3. Two mobiles with binary phase shift keying (BPSK) modulated signals of equal power but with different training sequences are simulated to impinge on an

eight element uniform linear array (ULA). The length of the training sequence is 26 symbols in order to make the simulation GSM/DCS-1800-like. The SMI-algorithm is used for the beamforming. In the simulations the temperature drift in magnitude and phase are generated as independent integrated random-walk processes in order to get a ”smooth” drift. The elements of the diagonal matrix **D** can be written as

$$d_l = A_l e^{j\varphi_l} \tag{6.16}$$

where d_l is the complex ratio of the l:th weighting path and the l:th receiving path. The quantities A_l and φ_l where generated according to

$$\begin{aligned} A_l(k+1) &= A_l(k) + \frac{1}{1-q^{-1}} e_{A,l}(k) \\ \varphi_l(k+1) &= \varphi_l(k) + \frac{1}{1-q^{-1}} e_{\varphi,l}(k) \end{aligned} \tag{6.17}$$

where $e_{A,l}(k)$ and $e_{\varphi,l}(k)$ are white noise sequences of appropriate variance. A_l and φ_l are initialized by the values from the off-calibration. This gives a temperature drift with statistical properties similar to that of the example in Figure 6.1.

To reduce the condition number (or eigenvalue spread) of the covariance matrix $\hat{\mathbf{R}}_{\mathbf{xx}}$ in (6.4), regularization is used to give a condition number of the order of 10^2. This regularization method is described in [40]. The covariance matrix can therefore be inverted without numerical problems, and the weight vector calculated by the SMI-algorithm will be approximately the same from frame to frame, under steady state conditions. As mentioned above, it is *not* likely for the weights to be calculated and the hardware weights to be updated at intervals less than a symbol period, due to the incapability of delaying the signal in the analog weighting path.

Since the LMS-algorithm proposed in Subsection 6.3.1 is a closed loop algorithm, it is not possible to update the weights more frequently than once per frame; we need to see the ”response” of the old weight in order to calculate a new one. This means that the time unit of the algorithm is one TDMA frame (or 4.615 ms in GSM/DCS-1800). The noise variances of the integrated random walk processes (6.17) that model the temperature drift of the magnitude and phase sets the time scale of the simulation. Here it is

chosen unrealistically high to reduce the simulation time. It is therefore not necessary to simulate more than 500 frames which corresponds to 2.3 seconds in GSM/DCS-1800 (!). The signal-to-interference ratio (SIR) of the adaptive array utilizing the auto-calibration algorithm presented in Subsection 6.3.1 is plotted as a function of frame number in Figure 6.5. In this particular simulation the angles of the two mobiles have been chosen to 15° and 43° relative to broadside of the ULA, and the step size μ in the LMS-like algorithm to 0.005. The variances of the noise sources $\mathbf{n}_1$, $\mathbf{n}_2$ and n_3 were chosen 20 dB, 40 dB and 30 dB below the signal level, respectively.

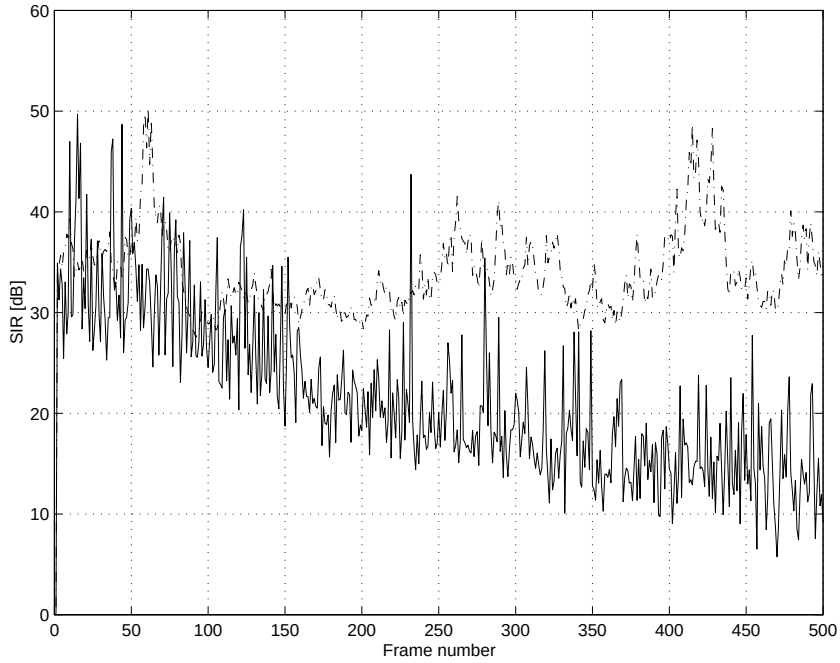

Fig. 6.5. SIR as a function of frame number. Dash-dotted: auto-calibration using LMS, solid: traditional SMI.

The SIR of the adaptive array only utilizing off-line calibration is also depicted in Figure 6.5. It is evident that the auto-calibration algorithm is able to maintain the SIR on a high level, whereas the performance of the traditional SMI adaptive array is severely degraded as the magnitude and phase drift away.

Figure 6.6 show the SIR resulting from using the tracking approach presented in Subsection 6.3.2, compared to the traditional SMI-array. All simulation parameters are identical to those in the LMS-study. Also in this case the SIR of the auto-calibration algorithm is maintained on a constant high level as the phase and magnitude is subjected to temperature drift. The variance of the SIR is higher in this case compared to the LMS-approach. This can be explained by the smoothing effect of the recursive LMS-algorithm, as opposed to the batch oriented approach taken in the

identification method (the tracking-compensated SMI-weights are used to form the beam). In this case we are also able to study the tracking ability of the algorithm. Tracking of the temperature drift for the fifth antenna element is presented in Figure 6.7, and it can be seen that both magnitude and phase are well estimated.

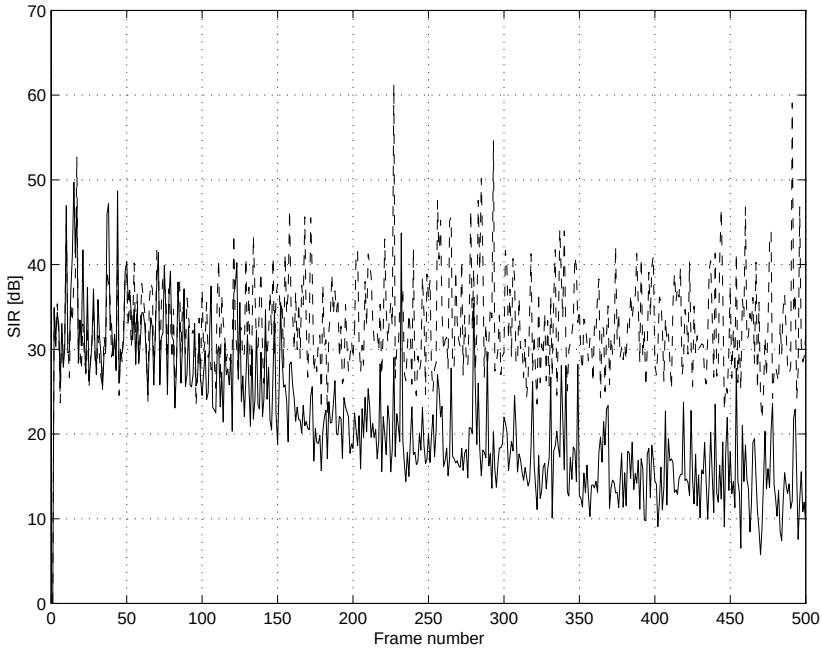

Fig. 6.6. SIR as a function of frame number. Dashed: auto-calibration using the identification approach, solid: traditional SMI.

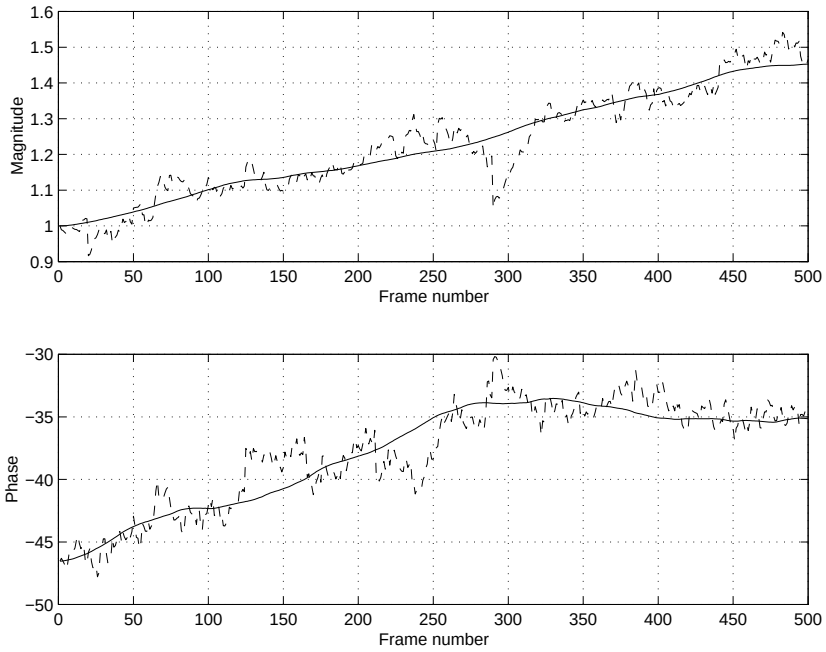

Fig. 6.7. Tracking of variations in magnitude and phase for the fifth antenna path obtained from the identification method. Dashed: tracked, solid: true.