

## Inverse Channel Estimation Based On Adaptive Inverse Control

Ge Liu<sup>1</sup>, Zhengxin Ma<sup>1</sup>, Yuhan Wang<sup>1</sup> and <sup>1</sup>Xiaowen Shu<sup>2</sup>

<sup>1</sup> Department of Electronic Engineering, Tsinghua University, Beijing 100084, China

<sup>2</sup> Military representative bureau of army aviation ,PLA, Beijing 100086, China

**Abstract.** A new algorithm of inverse channel estimation based on adaptive inverse control is proposed in this paper for the multipath fading wireless channel to compensate the channel distortion. The algorithm does not directly estimate the inverse channel, as compared with adaptive equalization. The algorithm estimates the channel firstly, and then use offline mode for inverse channel estimation. Simulation results show that, the new algorithm can obtain better performance in the same SNR, as compared with adaptive equalization.

**Keywords:** Inverse channel estimation, channel estimation, adaptive inverse control

### 1. Introduction

In wireless communications, wireless signal is affected by the environment, distance, propagation path and other uncertainties, resulting in multipath and fading phenomenon and system performance degradation. Accurate estimation of the inverse of a channel or inverse channel can transform an AWGN multipath fading channel into an AWGN channel, thus solve the wireless channel multipath and fading problem. The most commonly used method of inverse channel estimation is adaptive equalizer [1]. Adaptive equalizer is usually used to directly estimate the inverse channel. The value of direct inverse channel estimation deviates from the true value due to the presence of noise. When the signal noise ratio (SNR) is high, the inverse channel offset is smaller, and the impact on the system performance is smaller due to lower noise. When SNR is low, the inverse channel offset is greater, and impact on the system performance is greater.

Adaptive inverse control theory is put forward by B. Widrow in the 1980s. The basic idea is that, first directly set up a model for an unknown object, then use the inverse object model to control the dynamic characteristic of the object (1).

In this paper, starting from the model of wireless channel, in view of the AWGN multipath fading channel, we propose an inverse channel estimation algorithm based on adaptive inverse control. Then, Matlab is used in the simulation and analysis of this algorithm. Simulation results show that, by using this algorithm in AWGN multipath fading channel to estimate the inverse channel, the estimation offset due to noise is reduced.

### 2. Channel Model

Reference [3] points out, tapped delay line can be use in multipath fading channel modeling. Let's assume that the wireless channel is quasi static, multipath is fixed, each path loss and fading complies with probability distribution, then, the channel impulse response shall be:

$$h(t) = \sum_{k=0}^L a_k \delta(t - t_k) e^{j\theta_k} \quad (1)$$

Where, L is the number of multipath,  $a_k, t_k, \theta_k$  is respectively multipath fading coefficient, time delay and phase. Wireless channel model can be set up by using transverse filter [4], and therefore the inverse

---

<sup>+</sup> Corresponding author. Tel.: +86-010-62772751; fax: +86-010-62796691.  
E-mail address: lg10@mails.tsinghua.edu.cn.

channel model can be set up with transverse filter. The adaptive equalizer, as shown in Fig. 1, is the existing commonly used method for inverse channel estimation.

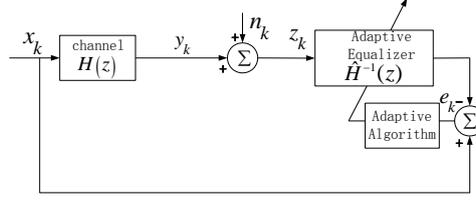


Fig. 1. Basic Block Diagram of Adaptive Equalizer

Where,  $x_k$  is the sending signal,  $H(z)$  is the channel transfer function,  $y_k$  is the signal after the sending signal pass through the channel,  $z_k$  is the receiving signal,  $\hat{H}^{-1}(z)$  is the inverse channel transfer function,  $n_k$  is the additive white Gaussian noise,  $e_k$  is the error.

Reference (2) put forward the Wiener solution for  $\hat{H}^{-1}(z)$ , as shown in equation (2):

$$\begin{aligned} \hat{H}^{-1}(z) &= \frac{\Phi_{zx}(z)}{\Phi_{zz}(z)} = \frac{\Phi_{yx}(z)}{\Phi_{yy}(z) + \Phi_{nn}(z)} \\ &= \frac{\Phi_{yx}(z)}{[\Phi_{yy}(z) + \Phi_{nn}(z)]H(z)} \end{aligned} \quad (2)$$

Where,  $\Phi_{zx}(z)$  is the Z transformation of  $z_k$  and  $x_k$  cross-correlation function,  $\Phi_{zz}(z)$  is the Z transformation of  $z_k$  autocorrelation function,  $\Phi_{yx}(z)$  is the Z transformation of  $x_k$  and  $y_k$  cross-correlation function,  $\Phi_{yy}(z)$  is the Z transformation of  $y_k$  autocorrelation function,  $\Phi_{nn}(z)$  is the Z transformation of  $n_k$  autocorrelation function.  $n_k, x_k, z_k$  are not related, so  $\Phi_{zx}(z) = \Phi_{yx}(z)$ .

As seen in equation (2), due to the existence of  $\Phi_{nn}(z)$ ,  $\hat{H}^{-1}(z)$  is not the inverse of  $H(z)$ , thus produce a deviation. With the increase of noise power, the adaptive equalizer performance becomes worse and worse.

In order to reduce the deviation due to noise, this paper proposes a new inverse channel estimation algorithm based on adaptive inverse control theory. The core idea is, firstly carry out channel estimation, then estimate the inverse of the channel in offline mode, finally, the inverse channel estimation is used as a controller to compensate for the channel.

### 3. Inverse Channel Estimation Algorithm Based On Adaptive Inverse Control

Because direct inverse channel estimation may be affected by noise, three steps are taken to estimate the channel inverse:

#### 3.1. Channel Estimation

The basic block diagram for channel estimation is shown in Fig. 2.

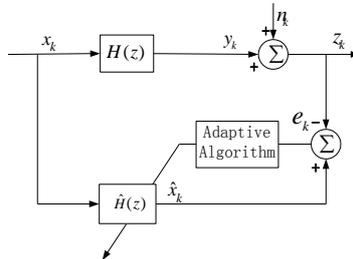


Fig. 2. Basic Block Diagram of Channel Estimation

Where,  $\hat{H}(z)$  is the estimation of channel transfer function.

Reference (2) put forward the Wiener solution for  $\hat{H}(z)$ , as shown in equation (3):

$$\hat{H}(z) = \frac{\Phi_{zz}(z)}{\Phi_{xx}(z)} = \frac{\Phi_{yy}(z)}{\Phi_{xx}(z)} = \frac{\Phi_{yy}(z)H(z)}{\Phi_{xx}(z)} = H(z) \quad (1)$$

Where,  $\Phi_{zx}(z)$  is the Z transformation of  $x_k$  and  $z_k$  cross-correlation function,  $\Phi_{xy}(z)$  is the Z transformation of  $x_k$  and  $y_k$  cross-correlation function,  $\Phi_{xx}(z)$  is the Z transformation of  $y_k$  autocorrelation function, Because  $n_k, x_k, z_k$  are not related,  $\Phi_{xz}(z) = \Phi_{xy}(z)$ .

As seen in equation (3), when there is a  $n_k$ , as long as  $n_k$  and  $x_k$  are not correlated, then  $\hat{H}(z)$  can accurately estimate  $H(z)$ .

In order to follow the time-varying characteristics of the channel, adaptive filter is used in  $h(n)$  modeling estimation. Due to the existence of noise, the error  $e_k$  cannot be zero, so the method of recursive least square (RLS) of the variable forgetting factor (VFF) [5] is used in channel estimation. The purpose of variable genetic factor  $\lambda$  is to enable  $\hat{x}_k \approx x_k$  and  $e_k \approx n_k$ . When  $\lambda$  is small,  $e_k$  tends to zero. When  $\lambda > 1$ , significant error will be introduced. In order to avoid a large imbalance,  $\lambda$  must be limited inside the closed interval of around 1. The adaptive filter will take a known sending training sequence  $x(n) = x_k$  as the input, the desired signal  $d(n) = z_k$ , the recursive equation of VFF-RLS algorithm is as follows:

$$\begin{aligned} k(n) &= \frac{\lambda(n)^{-1}P(n-1)x(n)}{1 + \lambda(n)^{-1}x^H(n)P(n-1)x(n)}, \varepsilon(n) = d(n) - w^H(n-1)x(n), \hat{w}(n) = \hat{w}(n-1) + k(n)\varepsilon^*(n), \\ P(n) &= \lambda(n)^{-1}(1 - k(n)x^H(n))P(n-1), y(n) = \hat{w}^H(n)x(n), e(n) = d(n) - \hat{w}^H(n)x(n), \\ q(n) &= x^H(n)P(n-1)x(n), \sigma_e^2(n+1) = \alpha\sigma_e^2(n) + (1-\alpha)\varepsilon^2(n), \sigma_q^2(n+1) = \alpha\sigma_q^2(n) + (1-\alpha)q^2(n), \\ \sigma_e^2(n+1) &= \beta\sigma_e^2(n) + (1-\beta)e^2(n), \lambda(n) = \frac{\sigma_q(n)\sigma_e(n)}{[\sigma_e^2(n) - \sigma_e^2(n)]^2}, \lambda(n) = \min[\max(\lambda_{\min}, \lambda(n), \lambda_{\max})]. \end{aligned} \quad (2)$$

$$\alpha = 1 - \frac{1}{K_\alpha N}, \beta = 1 - \frac{1}{K_\beta N}, K_\beta < K_\alpha < 1, N \text{ is the length of the filter. When the VFF-RLS converges to a fixed}$$

value,  $\hat{h}(n) = \hat{w}(n)$ .

### 3.2. Channel Estimation

The inverse channel stability depends on whether the channel transfer function is at minimum phase or not. A stable inverse channel can be obtained in the following two cases:

#### 3.2.1 Minimum Phase

When the channel transfer function is at minimum phase, the transfer function zeros are within the unit circle, that is to say, its inverse poles are also inside the unit circle, the inverse channel is stable. The basic block diagram of inverse channel estimation is shown in Fig. 3.

Wiener solution for  $\hat{H}^{-1}(z)$  is shown in equation (5):

$$\hat{H}^{-1}(z) = \frac{\Phi_{yx}(z)}{\Phi_{yy}(z)} = \frac{\Phi_{xy}(z)}{\Phi_{xx}(z)H(z)} = \frac{1}{\hat{H}(z)} \quad (3)$$

As seen in equation (5), offline inverse modeling can accurately estimate the inverse  $\hat{H}^{-1}(z)$  of  $\hat{H}(z)$ .

#### 3.2.2 Non-Minimum Phase

When the channel transfer function is at non-minimum phase, the transfer function zeros are outside the unit circle, that is to say, its inverse poles are also outside the unit circle, the inverse channel is not stable. The delay channel inverse channel estimation under this condition is shown in Fig. 4.

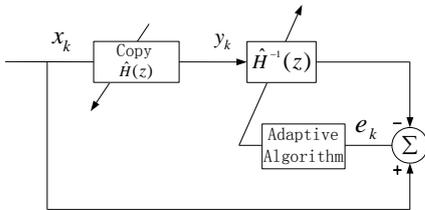


Fig. 3. Basic Block Diagram of Inverse Channel Estimation

Where,  $z^{-\Delta}$  is the delay transfer function of  $\Delta$ .

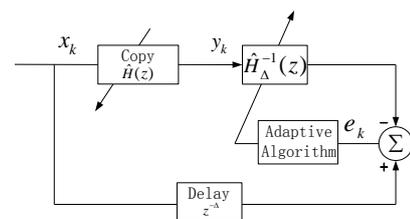


Fig. 4. Basic Block Diagram of Delay Inverse Channel Estimation

The purpose of using  $z^{-\Delta}$  to eliminate the zero points of channel transfer function outside the unit circle, that is multiply  $\hat{H}(z)$  and  $\hat{H}_{\Delta}^{-1}(z)$ , the result is the best fitting for  $z^{-\Delta}$ . The selecting of  $\Delta$  value is very important. If the value selected is too small, the transfer function cannot be stable. If the value is too big, a more ideal time delay inverse can be obtained, but the time delay of the output signal will be bigger. Therefore, the commonly used  $\Delta$  empirical value is half the length of the filter. As for the minimum phase system, delay  $\Delta = 0$  is a good option.

A modified MNLMS algorithm proposed in reference (6) can be used in inverse channel estimation.. Adaptive filter can take  $x(n) = y_k$  as the input, where the parameter  $\hat{h}(n)$  can copy the estimation value of the above step, the desired signal is  $d(n) = x_k$ , the recursive equation for MNLMS algorithm is as follows:

$$y(n) = \hat{w}^H(n)x(n), e(n) = d(n) - \hat{w}^H(n)x(n)$$

$$\hat{w}(n+1) = \hat{w}(n) + \frac{\mu}{\left( \frac{\beta}{\lambda + \sum_{i=1}^n (e_i)^2} \right) + (1-\beta)\|x(n)\|^2} e(n)x(n) \quad (6)$$

Where,  $\mu$  is the step length,  $\beta > 0, 0 < \lambda < 1$ . When MNLMS is converged to a fixed value,  $\hat{h}_{\Delta}^{-1}(n) = \hat{w}(n)$ .

### 3.2.3 Adaptive Inverse Control of the Channel

The basic block diagram of the channel adaptive inverse control is shown in Fig. 5.

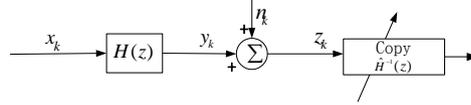


Fig. 5. Basic Block Diagram of the Channel Adaptive Inverse Control

After the inverse channel estimation  $\hat{h}_{\Delta}^{-1}(n)$  is obtained, the estimation can be used as a controller connected in series to the receiver front-end, so as to compensate the influence of channel on the signal.

## 4. Simulation Results

Simulation parameter settings are as follows: QPSK modulation is adopted, signal symbol rate is 2.5 Mbps, sampling rate is 16, roll-off factor of forming and matched filtering is 0.35, channel impulse response is  $h(n) = [0.6708 \ 0.5 \ 0.3873 \ 0.3162 \ 0.2236]$ , channel noise is additive white Gaussian noise. VFF-RLS algorithm is used in channel estimation,  $K_{\alpha} = 0.5$ ,  $K_{\beta} = 0.05$ ,  $\lambda_{\min} = 0.9965$ ,  $\lambda_{\max} = 0.999999$ . Both the inverse channel estimation and adaptive channel equalization adopts the MNLMS algorithm,  $\mu = 0.2$ ,  $\beta = 0.8, \lambda = 0.4$ .

Fig. 6 shows the signal amplitude difference in the case of SNR = 0 db. In this case, the adaptive inverse control and adaptive equalization are used respectively in inverse channel estimation. Receiving signal with no noise respectively pass through the inverse channel estimator, producing a signal that is different from the sending signal. The figure shows that the receiving signal passed through the adaptive inverse control estimation or the channel inverse filter is better than the signal of the adaptive equalizer, namely closer to the send signal, therefore it can better compensate the influence of channel on the signal.

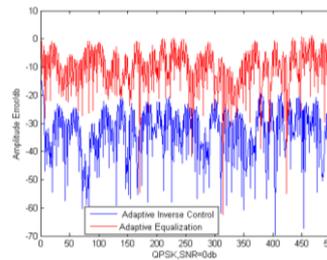


Fig. 6. Signal Amplitude Difference in Case of SNR=0db

Fig. 7 and Fig. 8 respectively show the frequency response in the case of SNR = 0 db and SNR = 20 db. The adaptive inverse control and adaptive equalization are used respectively in inverse channel estimation. After the inverse channel estimation is obtained, we can see the combined frequency response of the channel and the inverse channel. Ideally, the system with the channel and the inverse channel connected in series is an all-pass system. As shown in the two figures, with the same SNR, using adaptive inverse control in inverse channel estimation, the system will be a near all-pass system. The system bandwidth is wider than that of the system which uses the adaptive equalization algorithm for inverse channel estimation. It also has a

better in-band flatness and better basic linear phase. As SNR gradually degrading, the performance of inverse channel estimation by using adaptive equalization algorithm deteriorates. But the performance of inverse channel estimation by using adaptive inverse control algorithm will not be seriously impacted.

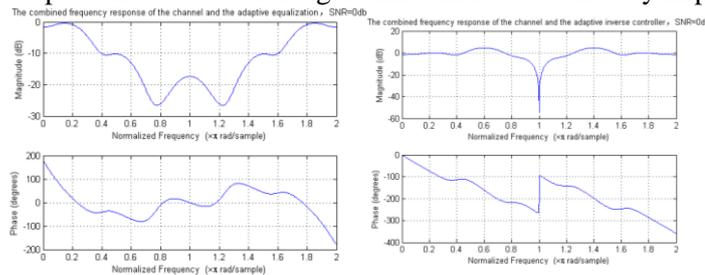


Fig. 7. Frequency Response in the Case of SNR=0db

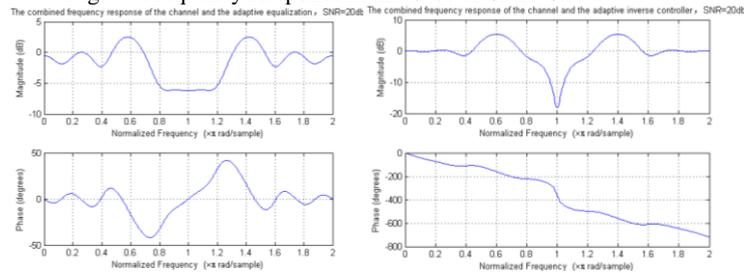


Fig. 8. Frequency Response in the Case of SNR=20db

Fig. 9 shows the system bit error rates (BER) in the cases of different SNRs. As shown in the figure, with the same SNR, the BER in the case of using the inverse channel estimation algorithm based on adaptive inverse control is lower than that of using the adaptive equalization algorithm.

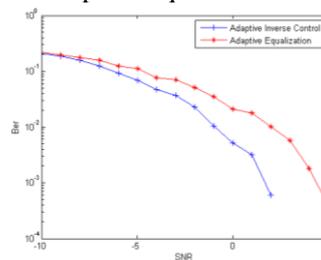


Fig. 9. Bit Error Rates

## 5. Conclusions

This paper studies the problem of inverse channel estimation for the multipath fading channel in wireless communication. The inverse channel estimation algorithm based on adaptive inverse control is proposed. Using the algorithm can transform an AWGN multipath fading channel into an AWGN channel. Theoretical analysis results show that this algorithm can suppress noise influence on inverse channel estimation. Modeling and simulation and analysis are conducted on this algorithm. As compared with the traditional adaptive equalization algorithm, this algorithm is less affected by noise, can more accurately estimate the channel inverse. With the same SNR, the new algorithm has lower bit error rates.

## 6. References

- [1] Chen T, Zakharov Y V, Liu C. "Low-Complexity Channel-Estimate Based Adaptive Linear Equalizer," *Signal Processing Letters, IEEE*, 2011, 18(7): 427-430.
- [2] B. Widrow, "Adaptive Inverse Control", Adaptive System in Control and Signal Processing, Sweden, 1986.
- [3] Zhan Yi. "Variable step size LMS adaptive equalization algorithm and its implementation on DSP," Chengdu: University of Electronic Science and Technology of China. 2010.11
- [4] Heping, Xu bingxiang, Zhanghai, Feng maoguan, "Adaptive Equalization Technique Application to Time-Variant Fading Channels," *Aota Electronica Sinica*, 1993, 4: 85-89.
- [5] TIAN Wen-ke, WANG Jian, SHAN Xiu-ming, "Adaptive Equalization Technique Application to Time-Variant Fading Channels," *Telecommunication Engineering*, 2011, 9: 78-82.
- [6] HAN Hua, LUO An, "Modified NLMS Algorithm and Its Simulation Research in Adaptive Inverse Control," *Measurement & Control Technology*, 2008, 5: 74-77.