Decision Weighted Adaptive Algorithms with Applications to Wireless Channel Estimation

Shane Martin Haas

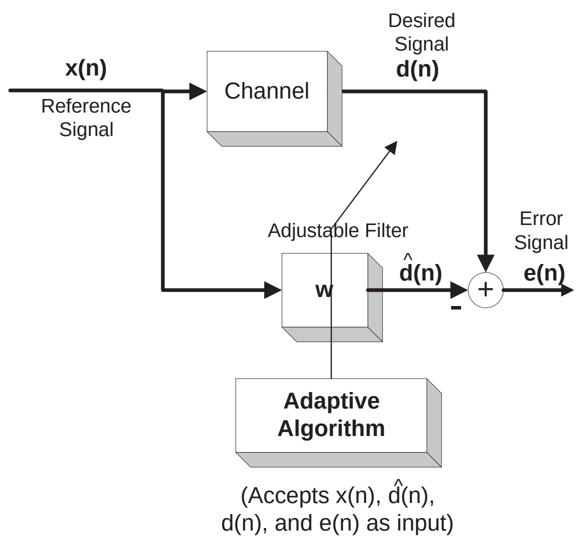
April 12, 1999

Thesis Defense for the Degree of
Master of Science in Electrical Engineering
Department of Electrical Engineering and Computer Science
University of Kansas

Introduction

⇒ What is an Adaptive Algorithm?

⇒ What is System Identification?



⇒ Benefits of Channel Estimation

⇒ Training Sequence Versus Blind Estimation

Presentation Overview

⇒ Theoretical Development

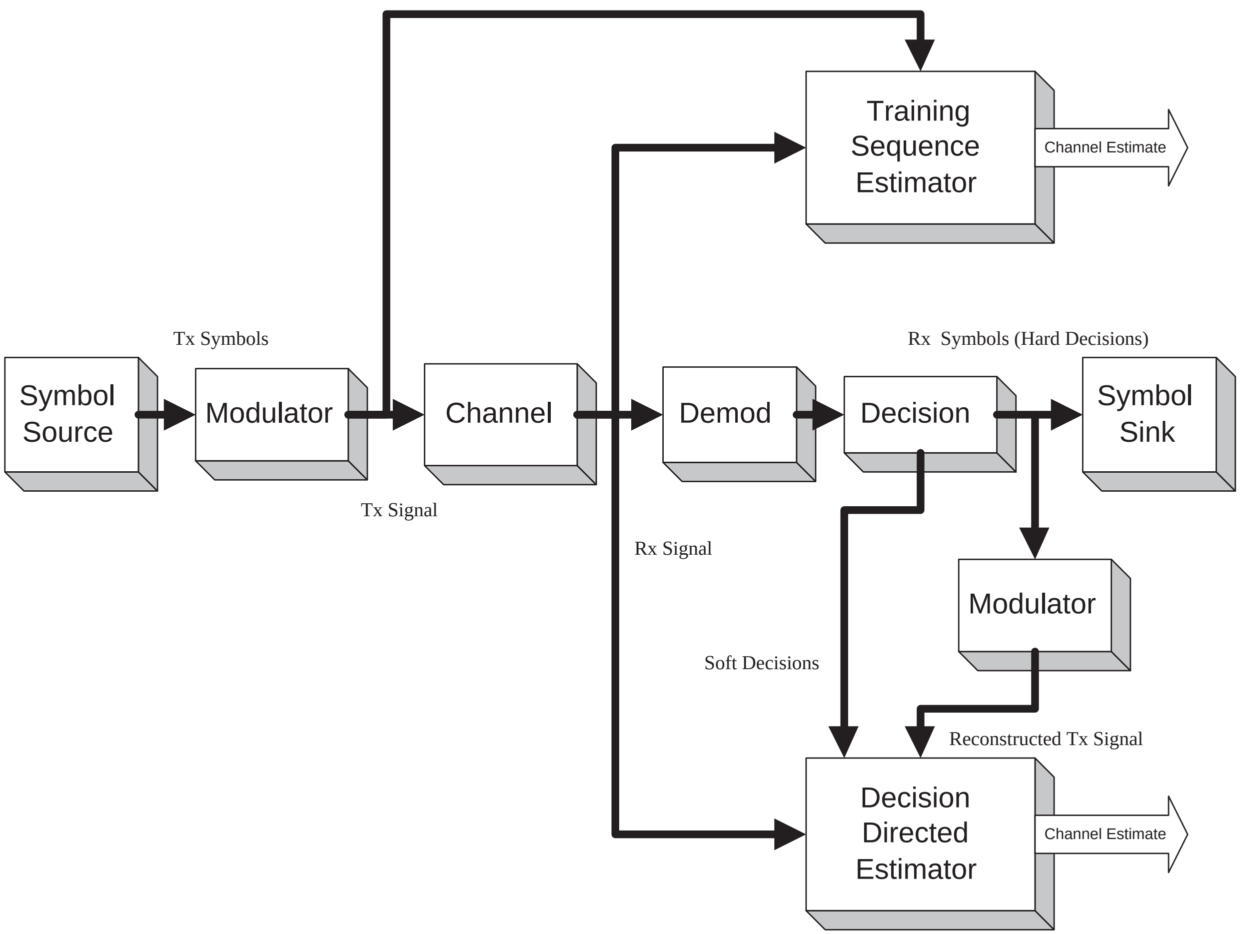
- Problem Formulation
- Multiple Phase Shift Keying
- Characterizing Wireless Communication Channels
- Bandpass to Low-Pass Conversion of Signals and Systems
- Adaptive Algorithms
  - Linear and LMS Estimation Algorithms
  - Properties of Decision Weighted Algorithms

⇒ Simulation Methodology

⇒ Simulation Results

⇒ Conclusions

Problem Formulation



Multiple Phase Shift Keying (MPSK)

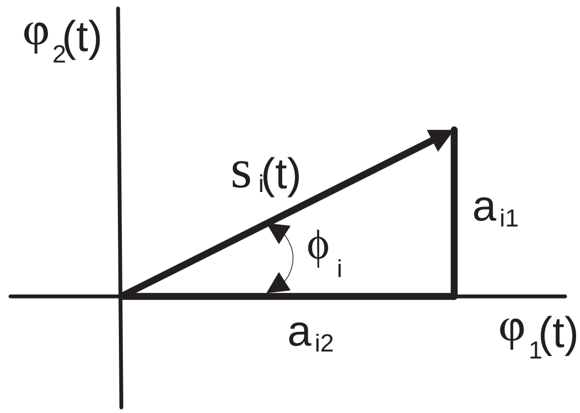
⇒ Modulation

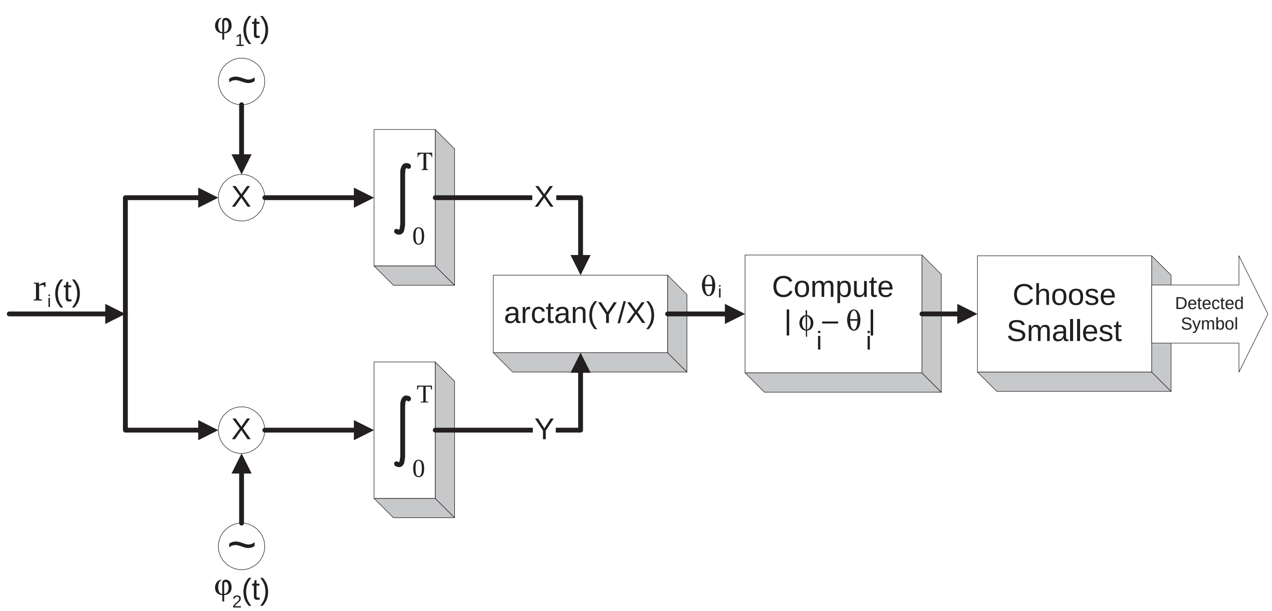
$$
\begin{aligned}
s_i(t) &= \sqrt{\frac{2E}{T}}\cos\left(2\pi f_c t + \frac{2\pi i}{M}\right) \qquad \text{for } 0 \le t < T \\
&= a_{i1}\psi_1(t) + a_{i2}\psi_2(t)
\end{aligned}
$$

with

$$
\begin{aligned}
\psi_1(t) &= \sqrt{\frac{2}{T}}\cos(2\pi f_c t) \\
\psi_2(t) &= -\sqrt{\frac{2}{T}}\sin(2\pi f_c t) \\
a_{i1} &= \sqrt{E}\cos(\phi_i) \\
a_{i2} &= \sqrt{E}\sin(\phi_i) \\
\phi_i &= \frac{2\pi i}{M}
\end{aligned}
$$

⇒ Demodulation

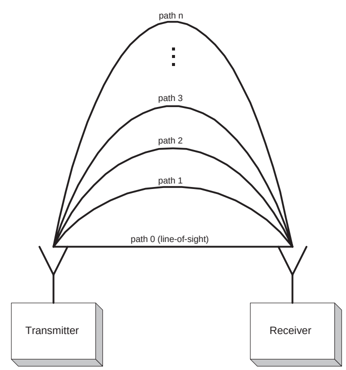




Characterizing Wireless Communication Channels

⇒ Multipath

$$y(t) = \sum_n \alpha_n(t) s(t - \tau_n(t))$$



⇒ Channel Models

* Radio Relay Three-Path (Rummler) Model
* Mobile Radio Channel Model

Bandpass to Low-Pass Conversion

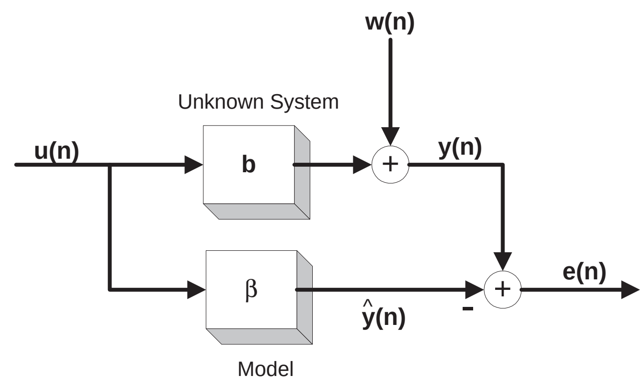
⇒ The Complex Envelope of the MPSK Signalling Waveform

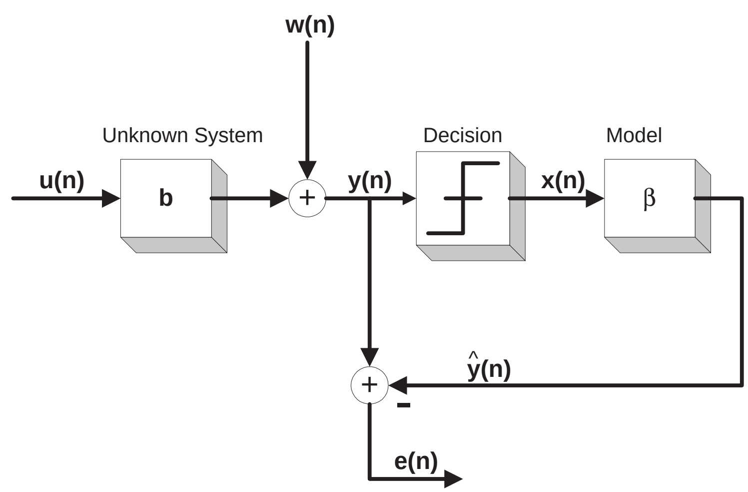
$$\widetilde{s}_i(t) = \sqrt{\frac{2E}{T}} \cos\left(\frac{2\pi i}{M}\right) + j\sqrt{\frac{2E}{T}} \sin\left(\frac{2\pi i}{M}\right)$$

⇒ Main Idea: Convolution of real bandpass signals is the same as the convolution of their complex envelope low-pass equivalents

Adaptive Algorithms

⇒ System Identification Problems





$\Rightarrow$ Linear Estimators

Define for the training sequence estimation problem:

$$\begin{aligned} y(n) &= b_1 u_1(n) + \cdots + b_M u_M(n) + w(n) \\ \widehat{y}(n) &= \beta_1 u_1(n) + \cdots + \beta_M u_M(n) \\ e(n) &= y(n) - \widehat{y}(n) = y(n) - (\beta_1 u_1(n) + \cdots + \beta_M u_M(n)) \end{aligned}$$

Define for the decision directed estimation problem:

$$\begin{aligned} y(n) &= b_1 u_1(n) + \cdots + b_M u_M(n) + w(n) \\ \widehat{y}(n) &= \beta_1 x_1(n) + \cdots + \beta_M x_M(n) \\ e(n) &= y(n) - \widehat{y}(n) = y(n) - (\beta_1 x_1(n) + \cdots + \beta_M x_M(n)) \end{aligned}$$

Observe the system for N sample periods and write

$$
\begin{aligned}
\mathbf{y} &= \begin{bmatrix} y(1) & y(2) & \cdots & y(N) \end{bmatrix}^T \\
\mathbf{w} &= \begin{bmatrix} w(1) & w(2) & \cdots & w(N) \end{bmatrix}^T \\
\mathbf{e} &= \begin{bmatrix} e(1) & e(2) & \cdots & e(N) \end{bmatrix}^T \\
\mathbf{b} &= \begin{bmatrix} b_1 & b_2 & \cdots & b_M \end{bmatrix}^T \\
\beta &= \begin{bmatrix} \beta_1 & \beta_2 & \cdots & \beta_M \end{bmatrix}^T \\
\mathbf{U} &= \begin{bmatrix} u_1(1) & \cdots & u_M(1) \\ \vdots & \ddots & \vdots \\ u_1(N) & \cdots & u_M(N) \end{bmatrix} \\
\mathbf{X} &= \begin{bmatrix} x_1(1) & \cdots & x_M(1) \\ \vdots & \ddots & \vdots \\ x_1(N) & \cdots & x_M(N) \end{bmatrix}
\end{aligned}
$$

The channel output is

$$\mathbf{y} = \mathbf{U}\mathbf{b} + \mathbf{w}$$

The error for the training sequence estimation problem is

$$\mathbf{e} = \mathbf{y} - \mathbf{U}\beta$$

While that for the decision directed estimation is

$$\mathbf{e} = \mathbf{y} - \mathbf{X}\beta$$

The error or loss function is

$$\mathbf{J}(\beta) = \mathbf{e}^{\mathbf{T}}\mathbf{R}\mathbf{e}$$

where $\mathbf{R}$ is a N x N matrix of weighting coefficients.

⇒ Linear Estimator

$$\widehat{\beta} = \left(\mathbf{U^T R U}\right)^{-1} \mathbf{U^T R y}$$

⇒ Recursive Weighted Least Squares Estimator

$$\mathbf{u}(n) = \begin{bmatrix} u_1(n) & u_2(n) & \cdots & u_M(n) \end{bmatrix}^T$$

$$\mathbf{R} = diag(\lambda^{n-1}a_1, \ldots, \lambda a_{n-1}, a_N) \qquad 0 < \lambda \leq 1$$

$$\begin{aligned} \widehat{\beta}_n &= \widehat{\beta}_{n-1} + a_n \mathbf{H}_n^{-1} \mathbf{u}(n) e(n) \\ \mathbf{H}_n &= \lambda \mathbf{H}_{n-1} + a_n \mathbf{u}(n) \mathbf{u}^T(n) \\ e(n) &= y(n) - \mathbf{u}(n)^T \widehat{\beta}_{n-1} \end{aligned}$$

⇒ Least Mean Squares (LMS) Estimator

$$\begin{aligned} \widehat{\beta}_n &= \widehat{\beta}_{n-1} + \mu\rho_n \mathbf{u}(n)e(n) \\ e(n) &= y(n) - \mathbf{u}(n)^T \widehat{\beta}_{n-1} \end{aligned}$$

⇒ Decision weighted estimators are decision directed estimators whose weights depend on the quality of the decisions.

* Ideal decision weighted estimators use knowledge of decision errors to calculate their weights. Specifically,

$$\mathbf{X^T R X} = \mathbf{X^T R U}$$

* Soft decision weighted estimators use receiver soft decisions to calculate their weights.

⇒ More on Ideal Decision Weighted Linear Estimators

* **Q**: How does one choose $\mathbf{R}$ such that $\mathbf{X^T R X} = \mathbf{X^T R U}$?
* **A**: If $\mathbf{X}$ and $\mathbf{U}$ differ in the jth *row*, choose the jth *column* of $\mathbf{R}$ orthogonal to each column in $\mathbf{X}$.
* **Q**: Are $\mathbf{X^T R X} = \mathbf{X^T R U}$ and $\mathbf{X^T R X}$ non-singular conflicting conditions?
* **A**: No, let $\mathbf{R}$ be an identity matrix with its jth column set to zero if $\mathbf{X}$ and $\mathbf{U}$ differ in the jth row. Under slightly more restrictive assumptions placed on $\mathbf{X}$ than in ordinary training sequence estimators, $\mathbf{X^T R X}$ is non-singular.

⇒ More on Soft Decision Weighted Estimators

For MPSK modulation define a soft decision as

$$p_i = 1 - \frac{|\phi_i - \theta_i|}{\pi/S}$$

A possible choice for the soft decision weight is

$$a_n = p_n p_{n-1} \cdots p_{n-M+1}$$

⇒ Biasness of Decision Directed Linear Estimators

If $\mathcal{E}\{\mathbf{w}|\mathbf{X},\mathbf{U}\} = \mathbf{0}$ then

$$\mathcal{E}\left\{\widehat{\beta}\right\} = \mathcal{E}\left\{\left(\mathbf{X}^{\mathbf{T}}\mathbf{R}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathbf{T}}\mathbf{R}\mathbf{U}\right\}\mathbf{b}$$

For ideal decision weighted estimators $\mathbf{X}^{\mathbf{T}}\mathbf{R}\mathbf{X} = \mathbf{X}^{\mathbf{T}}\mathbf{R}\mathbf{U}$, and therefore the estimator is unbiased.

⇒ Covariance of Decision Directed Linear Estimators

If $\mathcal{E}\{\mathbf{w}|\mathbf{X},\mathbf{U}\} = \mathbf{0}$ then

$$\mathbf{cov}\left\{\widehat{\beta}\right\} = \mathbf{cov}\{\mathbf{S}\mathbf{U}\mathbf{b}\} + \mathcal{E}\left\{\mathbf{S}\mathbf{V}\mathbf{S}^{\mathbf{T}}\right\}$$

where $\mathbf{S} = \left(\mathbf{X}^{\mathbf{T}}\mathbf{R}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathbf{T}}\mathbf{R}$ and $\mathbf{V} = \mathcal{E}\left\{\mathbf{w}\mathbf{w}^{\mathbf{T}} \mid \mathbf{X},\mathbf{U}\right\}$. Notice for ideal decision weighted estimators $\mathbf{S}\mathbf{U} = \mathbf{I}$.

Simulation Methodology

⇒ Algorithm Summary

* **Training Sequence LMS (TLMS):** Uses training sequence LMS with $\rho_n = 1$
* **Blind LMS (BLMS):** Uses decision directed LMS with $\rho_n = 1$
* **Soft Decision Weighted LMS (SDWLMS):** Uses decision directed LMS with soft decision weights
* **Ideal Decision Weighted LMS (IDWLMS):** Uses decision directed LMS with $\rho_n = 1$ if $\mathbf{x}(\mathbf{n}) = \mathbf{u}(\mathbf{n})$ and zero otherwise
* **Training Sequence RLS (TRLS):** Uses training sequence WRLS with $a_n = 1$
* **Blind RLS (BRLS):** Uses decision directed WRLS with $a_n = 1$
* **Soft Decision Weighted RLS (SDWRLS):** Uses decision directed WRLS with soft decision weights

* **Ideal Decision Weighted RLS (IDWRLS):** Uses decision directed WRLS with $a_n = 1$ if $\mathbf{x}(\mathbf{n}) = \mathbf{u}(\mathbf{n})$ and zero otherwise
* **Modified Soft Decision Weighted RLS (MSDWRLS):** Uses decision directed WRLS with soft decision weights; however, we modify the matrix update $\mathbf{H}_n$ by removing the weight a_n, resulting in $\mathbf{H}_n = \lambda \mathbf{H}_{n-1} + \mathbf{x}(n)\mathbf{x}^T(n)$.
* **Modified Ideal Decision Weighted RLS (MSDWRLS):** Uses decision directed WRLS with $a_n = 1$ if $\mathbf{x}(\mathbf{n}) = \mathbf{u}(\mathbf{n})$ and zero otherwise; however, we modify the matrix update $\mathbf{H}_n$ by removing the weight a_n, resulting in $\mathbf{H}_n = \lambda \mathbf{H}_{n-1} + \mathbf{x}(n)\mathbf{x}^T(n)$.

⇒ General Methods for Delay Spread, SNR, and Doppler Frequency Tests

* **LMS Gain:** $\mu = 0.3$
* **RLS Forgetting Factor:** $\lambda = 0.99$
* **Sampling Rate:** 1 sample per second
* **Symbol Interval:** 4 samples per symbol
* **Modulation:** QPSK
* **Number of Symbols per Individual Simulation:** 300 symbols
* **Number of Individual Simulations to Perform per Test Point Iteration:** 20 simulations
* **Maximum Symbol Error Rate (SER):** 0.2
* **Number of Symbols to Skip Before Calculating Estimation Error (N_0):** 100 symbols
* **Initial Estimate:** the true response
* **Performance Criteria:** median average estimation error

Simulation Results

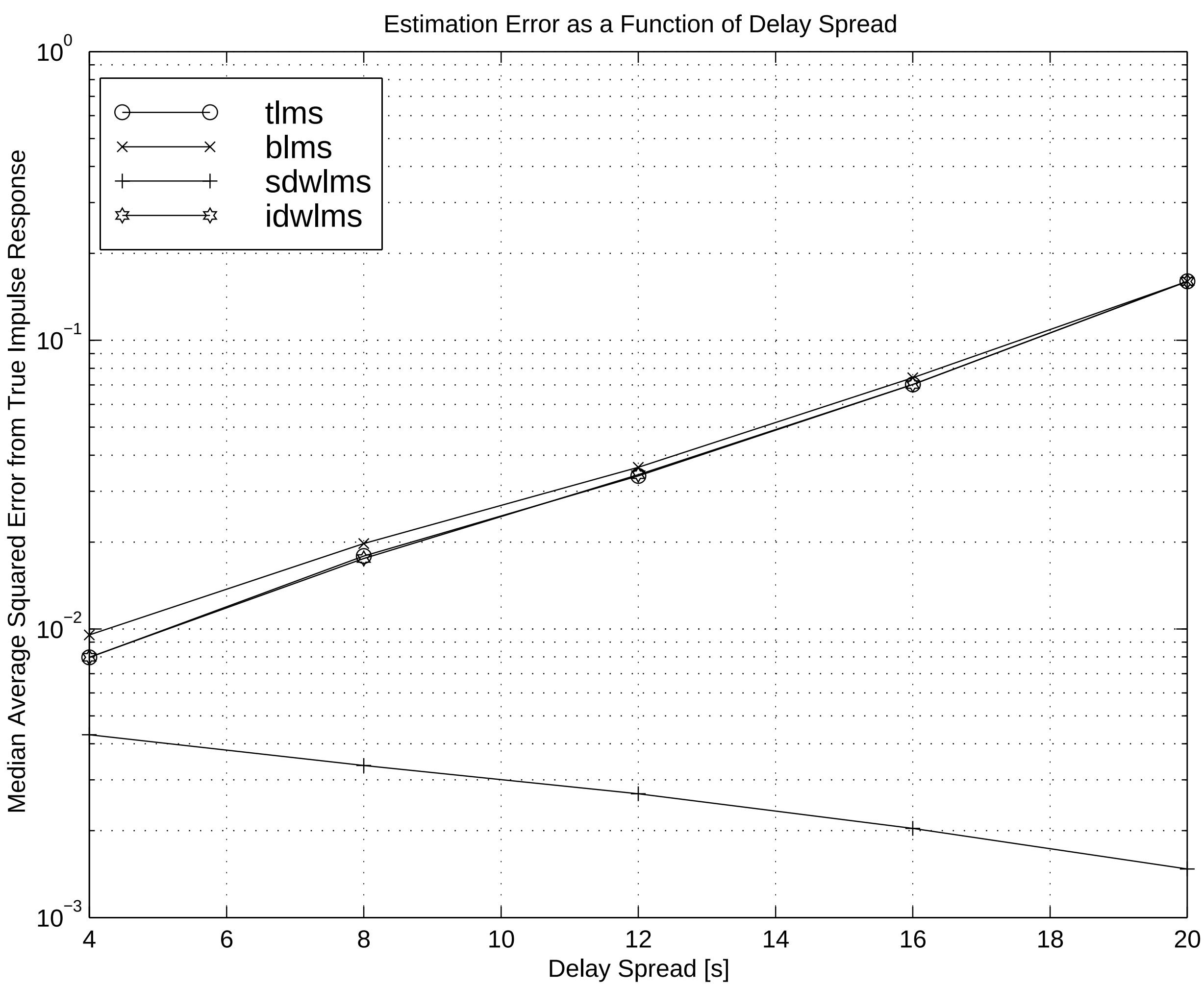


Figure 1: Median of the average squared error of LMS algorithms as a function of delay spread (SNR = 10 dB)

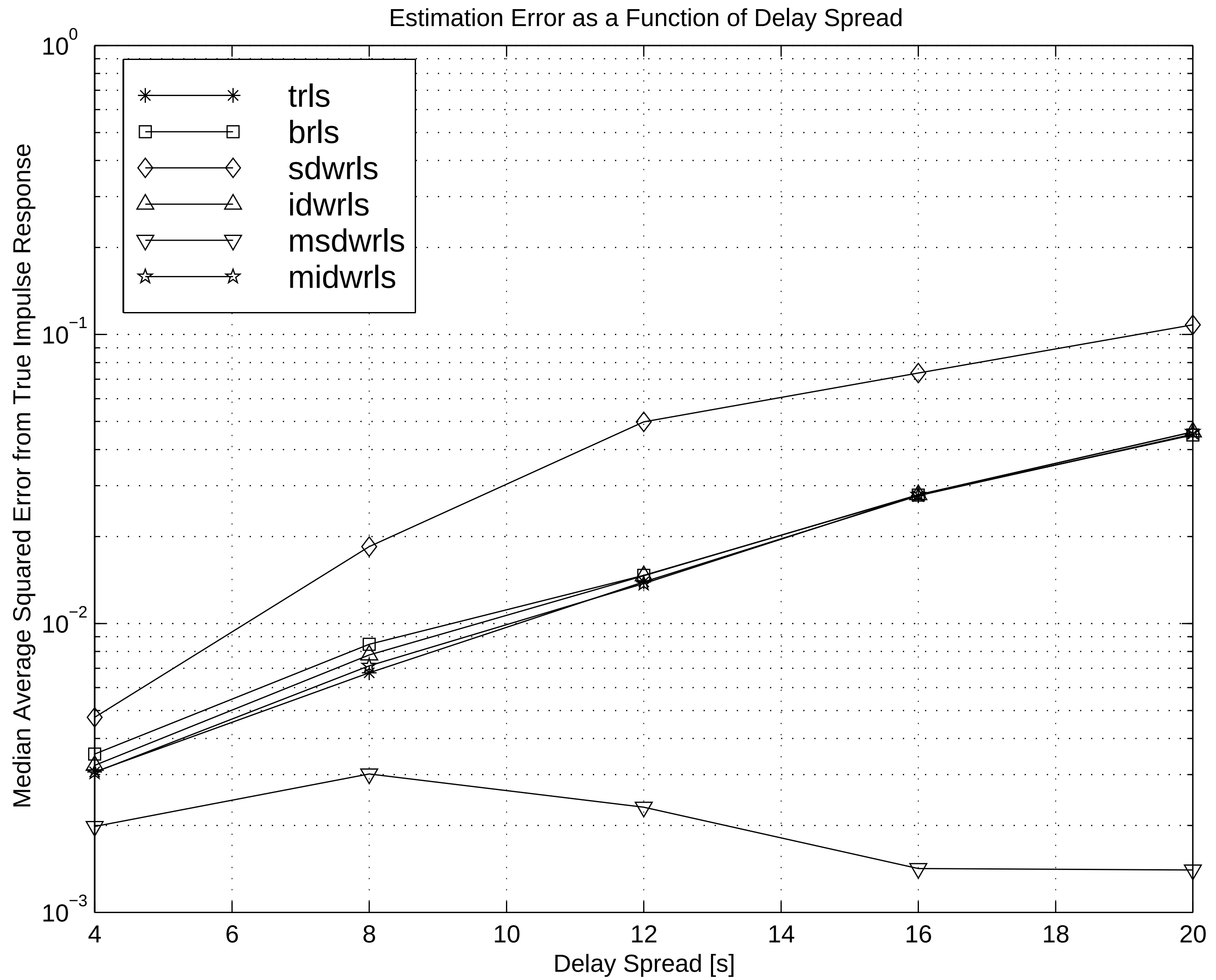


Figure 2: Median of the average squared error of RLS algorithms as a function of delay spread (SNR = 10 dB)

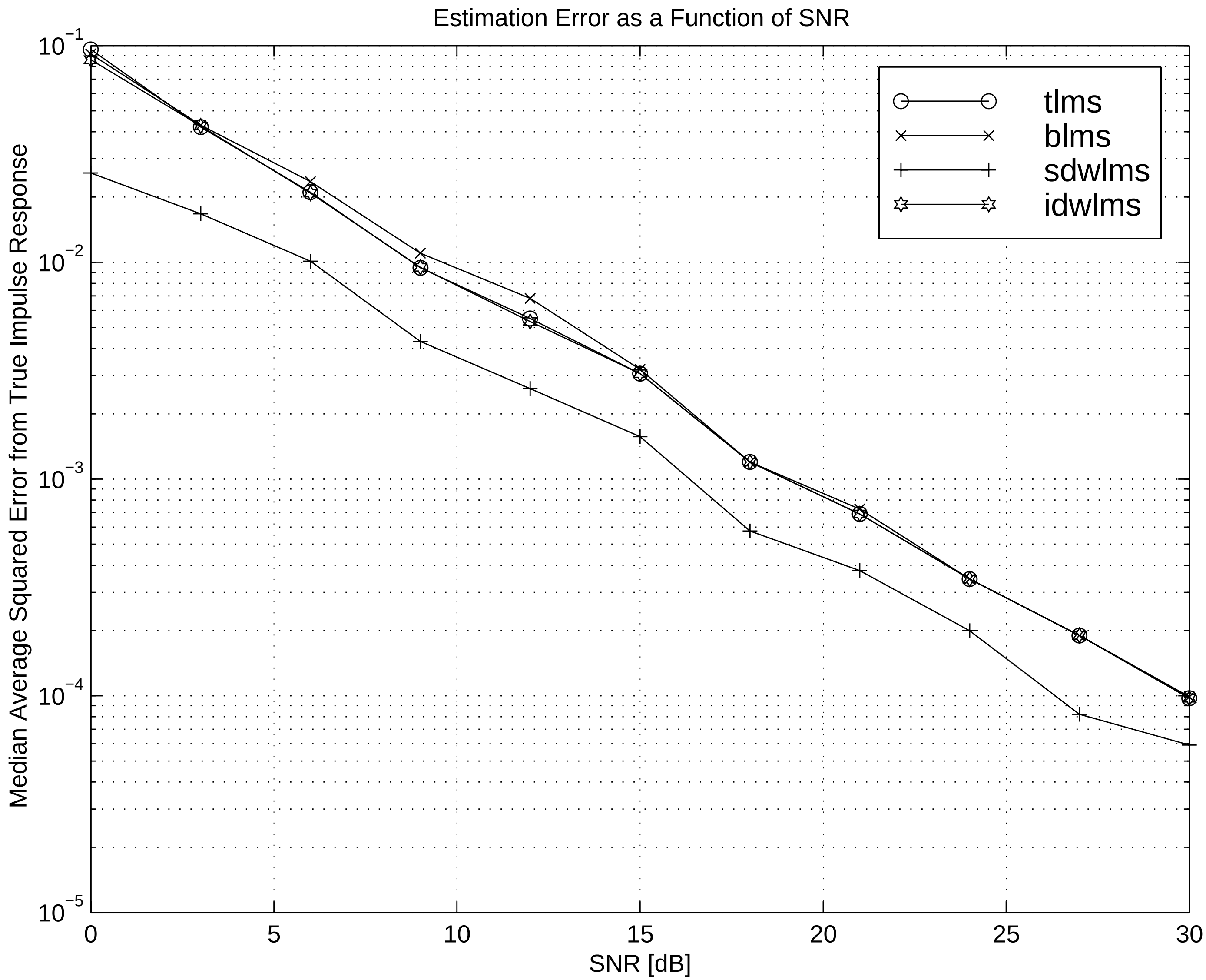


Figure 3: Median of the average squared error of LMS algorithms as a function of SNR (Delay Spread = 1 symbol interval)

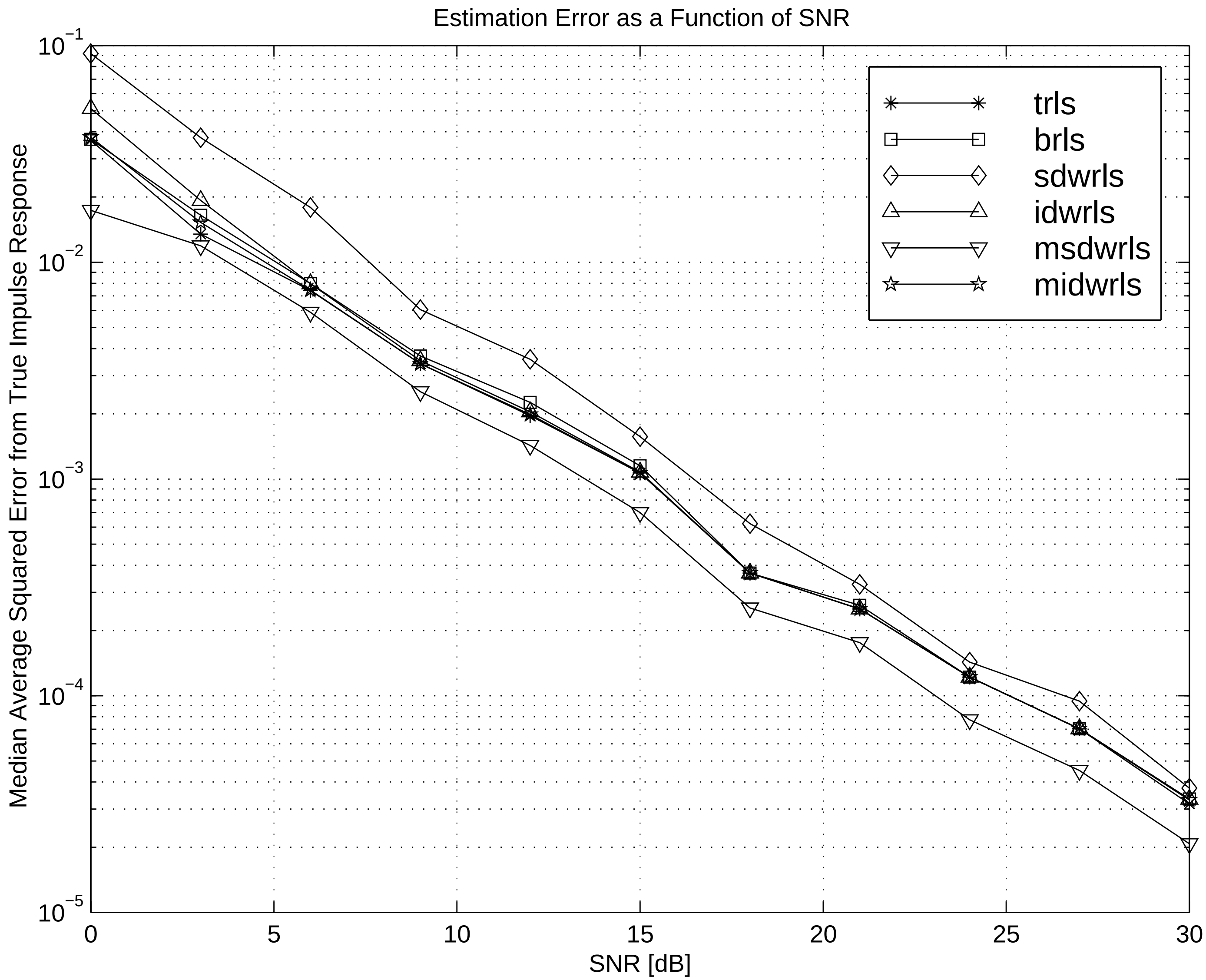


Figure 4: Median of the average squared error of RLS algorithms as a function of SNR (Delay Spread = 1 symbol interval)

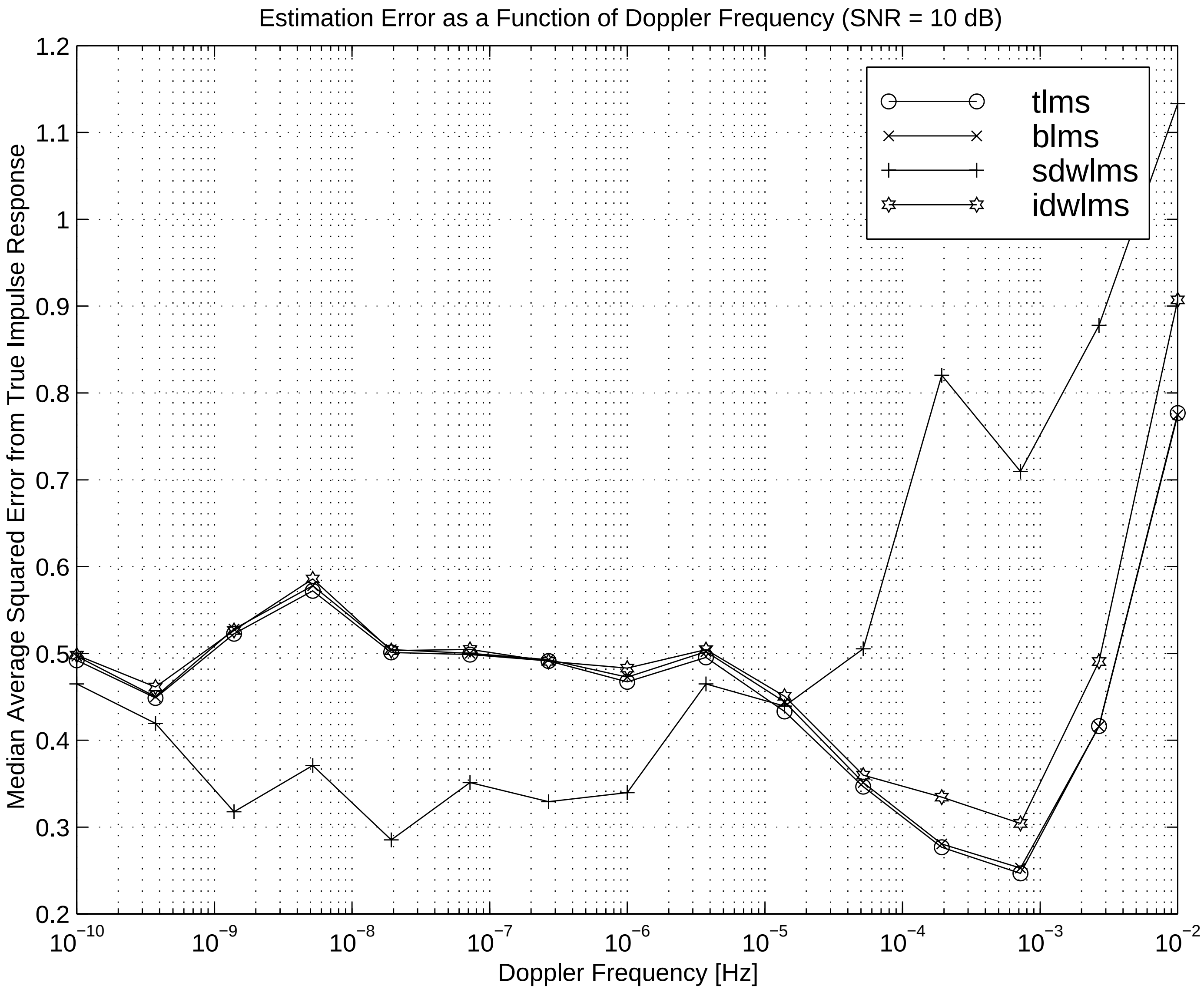


Figure 5: Median of the average squared error of LMS algorithms as a function of Doppler frequency (SNR = 10 dB)

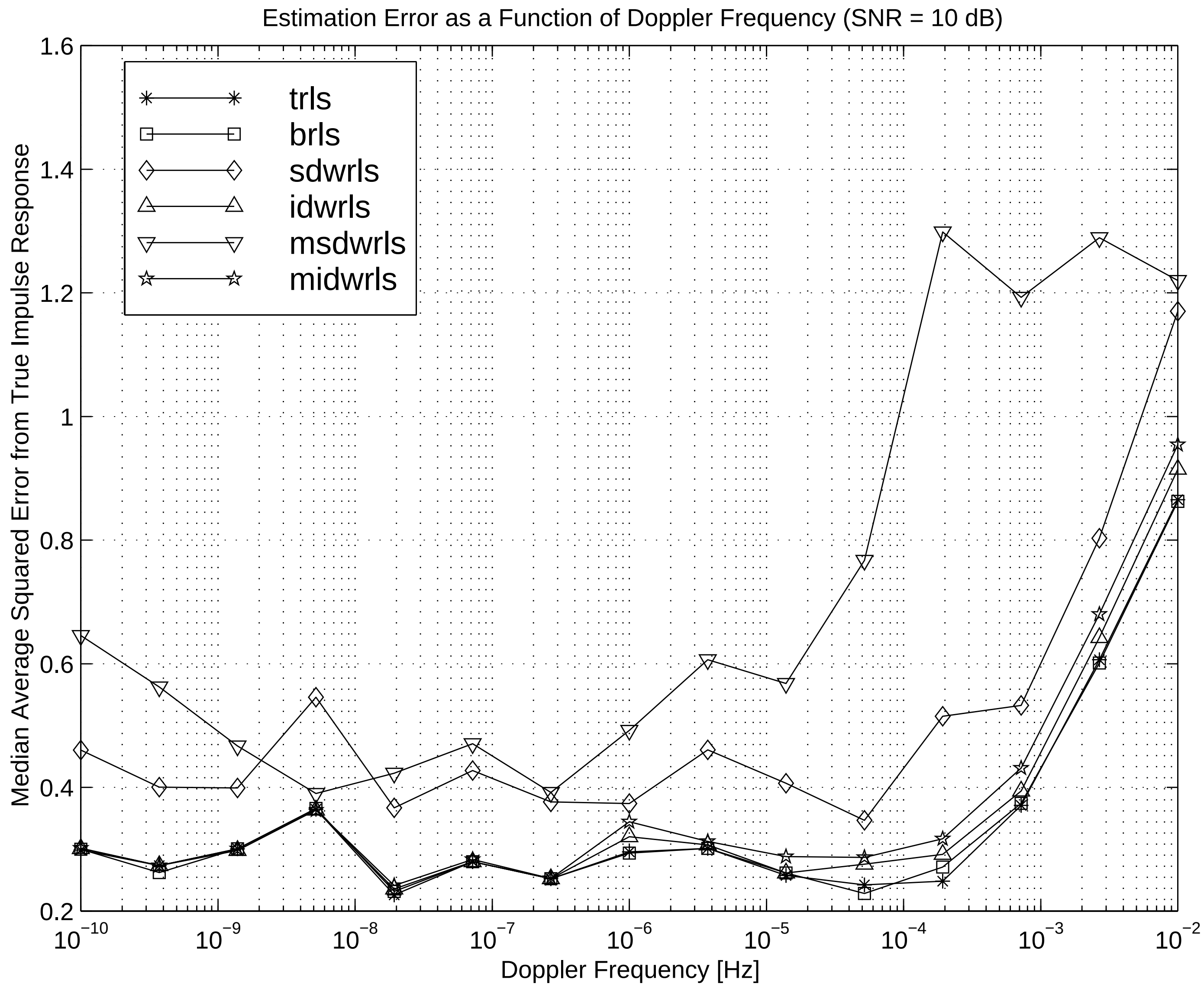


Figure 6: Median of the average squared error of RLS algorithms as a function of Doppler frequency (SNR = 10 dB)

Conclusions

⇒ Summary of Performance Test Results

- Soft decision weighted LMS (SDWLMS) performed *better* than the other LMS algorithms in delay spread (by a factor of 2 to 100, Figure 1) and SNR (by a factor of 2, Figure 3) tests
- Soft decision weighted RLS (SDWRLS) performed *worse* than the other RLS algorithms in delay spread (by a factor of 2, Figure 2) and SNR (by a factor of 3, Figure 4) tests
- Modified soft decision weighted RLS (MSDWRLS) performed *better* than the other RLS algorithms in delay spread (by a factor of 2 to 20, Figure 2) and SNR (by a factor of 2, Figure 2) tests
- SDWLMS performed *better* at normalized Doppler frequencies less than 10^{-5} and *worse* at higher Doppler frequencies than the other LMS algorithms (Figure 5)
- SDWRLS and MSDWRLS performed *worse* over all Doppler frequencies than the other RLS algorithms (Figure 6)

- * Ideal decision weighted LMS and RLS (IDWLMS and IDWRLS) performed similar to their training sequence versions in all tests, and generally better than their ordinary decision-directed counterparts (Figures 1 through 6).

⇒ General Conclusions

- * Decision weighted estimators defined, analyzed, and simulated
- * SDWLMS shows most promise for implementation
- * SDWRLS performed poorly, but MSDWRLS performed well
- * Ideal decision weighted algorithms performed similar to training sequence algorithms