LETTER

Estimation of the Matrix Rank of Harmonic Components of a Spectrogram in a Piano Music Signal Based on the Stein's Unbiased Risk Estimator and Median Filter

Seokjin LEE[†a)], *Member*

SUMMARY The estimation of the matrix rank of harmonic components of a music spectrogram provides some useful information, e.g., the determination of the number of basis vectors of the matrix-factorization-based algorithms, which is required for the automatic music transcription or in post-processing. In this work, we develop an algorithm based on Stein's unbiased risk estimator (SURE) algorithm with the matrix factorization model. The noise variance required for the SURE algorithm is estimated by suppressing the harmonic component via median filtering. An evaluation performed using the MIDI-aligned piano sounds (MAPS) database revealed an average estimation error of -0.26 (standard deviation: 4.4) for the proposed algorithm.
key words: *automatic music transcription, nonnegative matrix factorization, number of bases estimation, Stein's unbiased risk estimator*

1. Introduction

Automatic music transcription (AMT), which is the process of automatically converting a music signal into musical notation, is one of the key challenges in music information retrieval. Smaragdis and Brown introduced a transcription algorithm based on the nonnegative matrix factorization [1], which has been successfully implemented for polyphonic music transcription; several spectrogram-factorization-based algorithms have also been recently developed [2]–[4].

The spectrogram-factorization-based algorithms exhibit relatively good performance in solving multiple fundamental frequency estimation problems, but the performances of the algorithms are affected by the number of basis vectors [1]. If the number of basis vectors is set to a smaller or larger value than the actual number of note events (including notes and chords), the algorithm may fail and miss several notes. This can lead to errors, such as confusing the harmonic partials for additional notes [1], [4]. The factorization-based algorithm exhibits the best performance when the number of basis vectors and the number of note events in the music signal are identical [1]. Some algorithms that use a pre-trained frequency basis are immune to this problem, but the unsupervised algorithms still require the number of basis vectors. Recent matrix factorization meth-

ods control the number of basis vectors [5] or choose the relevant basis vectors [6], but the algorithms are designed for specific factorization algorithms.

The factorization-based AMT algorithm decomposes a magnitude spectrogram of harmonic components into multiplications of matrices and, hence, the number of basis vectors is related to the matrix rank of the harmonic components. In this paper, a method is developed for estimating the matrix rank of harmonic components of a spectrogram. The proposed algorithm is based on Stein's unbiased risk estimator (SURE) algorithm [7]. The SURE algorithm requires the noise variance as an *a priori* knowledge, so the proposed algorithm calculates the variance of non-harmonic components via median filtering [8].

The contributions of this work are two-fold: the SURE algorithm is applied to the NMF model; a noise variance estimation algorithm is proposed for the SURE method. The development of the SURE algorithm (Sect. 2.) is based on [7], but in our case Stein's lemma (4) is applied clearly. Although slight, this difference can help to extend the algorithm to non-Gaussian noise cases in the future.

2. Spectrogram Model and Rank Estimation

Lee and Seung [9] established the nonnegative matrix factorization (NMF) model, which is described by

$$\mathbf{V} = \mathbf{WH} + \mathbf{E} \triangleq \mathbf{V}_0 + \mathbf{E} \tag{1}$$

where $\mathbf{V}$, $\mathbf{W}$, and $\mathbf{H}$ are nonnegative matrices with sizes of $K \times N$, $K \times R$, and $R \times N$, respectively. According to Smaragdis and Brown [1], each basis vector of factorized matrices $\mathbf{W}$ and $\mathbf{H}$ corresponds to a note or a chord, as shown in Fig. 1, when the nonnegative matrix $\mathbf{V}$ is a magnitude of the spectrogram of a music signal. Therefore, the
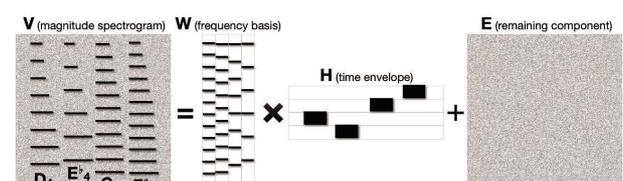
Fig. 1 An illustrative diagram for the NMF model in the music signal analysis.

number of basis vectors of the NMF model can be determined by estimating the rank of $\mathbf{V}_0$ in a noisy environment.

Equation (1) can be rewritten as

$$\mathbf{v}(n) = \mathbf{W}\mathbf{h}(n) + \mathbf{e}(n) = \mathbf{v}_0(n) + \mathbf{e}(n) \qquad (2)$$

where $\mathbf{v}(n)$, $\mathbf{v}_0(n)$, $\mathbf{h}(n)$, and $\mathbf{e}(n)$ are the n-th columns of $\mathbf{V}$, $\mathbf{V}_0$, $\mathbf{H}$, and $\mathbf{E}$, respectively. The SURE method chooses the model order R, which is equal to the number of column vectors comprising matrix $\mathbf{W}$, which minimizes the risk function [7]

$$\begin{aligned}\mathfrak{R}_R &= E\|\mathbf{v}_0(n) - \hat{\mathbf{v}}_0(n)\|^2 \\ &= E\|\mathbf{e}(n) - \hat{\mathbf{e}}(n)\|^2 \\ &= E\|\hat{\mathbf{e}}(n)\|^2 - 2E(\hat{\mathbf{e}}^T(n)\mathbf{e}(n)) + K\sigma_e^2 \end{aligned} \qquad (3)$$

where $\hat{\mathbf{e}}(n) = \mathbf{v}(n) - \hat{\mathbf{v}}_0(n)$, $\hat{\mathbf{v}}_0(n)$ is the estimation of $\mathbf{v}_0(n)$, and σ_e^2 is the variance of $\mathbf{e}(n)$.

If we assume that $\mathbf{e}(n)$ has a zero-mean Gaussian distribution with a variance of $\sigma_e^2 I_K$, then we can use Stein's lemma. Stein's lemma states that random vector Y, which has Gaussian distributions, satisfies [10]

$$\mathrm{tr}\{\mathrm{Cov}(Y, f(Y))\} = \sigma_Y^2 E\left\{\mathrm{tr}\left(\frac{\partial f(Y)}{\partial Y}\right)\right\}, \qquad (4)$$

where $f(Y)$ is an arbitrary differentiable function. We use $\mathbf{e}$ and $\hat{\mathbf{e}}$ (omitting the time index n for convenience) instead of Y and $f(Y)$, respectively, then (4) becomes

$$\begin{aligned}E\left(\hat{\mathbf{e}}^T \mathbf{e}\right) &= \sigma_e^2 E\left\{\mathrm{tr}\left(\frac{\partial \hat{\mathbf{e}}}{\partial \mathbf{e}^T}\right)\right\} \\ &= \sigma_e^2\left[K - E\left\{\mathrm{tr}\left(\frac{\partial \hat{\mathbf{v}}_0}{\partial \mathbf{v}^T}\right)\right\}\right], \end{aligned} \qquad (5)$$

because $\mathbf{e} = \mathbf{v} - \mathbf{v}_0$ and $\hat{\mathbf{e}} = \mathbf{v} - \hat{\mathbf{v}}_0$.

Therefore, the risk function $\mathfrak{R}_R$ becomes

$$\begin{aligned}\mathfrak{R}_R &= E\|\hat{\mathbf{e}}\|^2 - 2E(\hat{\mathbf{e}}^T \mathbf{e}) + K\sigma_e^2 \\ &= E\|\hat{\mathbf{e}}\|^2 - 2\left[K\sigma_e^2 - \sigma_e^2 E\left\{\mathrm{tr}\left(\frac{\partial \hat{\mathbf{v}}_0}{\partial \mathbf{v}^T}\right)\right\}\right] + K\sigma_e^2 \\ &= E\|\hat{\mathbf{e}}\|^2 + 2\sigma_e^2 E\left\{\mathrm{tr}\left(\frac{\partial \hat{\mathbf{v}}_0}{\partial \mathbf{v}^T}\right)\right\} - K\sigma_e^2. \end{aligned} \qquad (6)$$

If we assume that the processes are ergodic, we may use the time average instead of the ensemble average as

$$\mathfrak{R}_R = \frac{1}{N}\sum_{n=1}^{N}\|\mathbf{v}(n) - \hat{\mathbf{v}}_0(n)\|^2 + \frac{2\sigma_e^2}{N}\sum_{n=1}^{N}\mathrm{tr}\left(\frac{\partial \hat{\mathbf{v}}_0(n)}{\partial \mathbf{v}^T(n)}\right) - K\sigma_e^2 \qquad (7)$$

Because R is identical to the rank of $\mathbf{V}_0$ as shown in Fig. 1, we apply the rank estimation method in the noisy principal component analysis (PCA). According to previous studies considering the noisy PCA, the R-rank components $\mathbf{v}_0$ can be estimated by R eigenvectors as follows [7]

$$\hat{\mathbf{v}}_0(n) = \sum_{j=1}^{R} \mathbf{p}_j \frac{l_j - \hat{\sigma}_R^2}{l_j}\mathbf{p}_j^T \mathbf{v}(n) \qquad (8)$$

where l_j is the j-th eigenvalue, and $\mathbf{p}_j$ is the j-th eigenvector of $(1/N)\sum_{n=1}^{N}\mathbf{v}(n)\mathbf{v}^T(n)$. $\hat{\sigma}_R^2$ can be calculated as [7]

$$\hat{\sigma}_R^2 = \frac{1}{K - R}\sum_{j=R+1}^{K} l_j. \qquad (9)$$

The partial derivative of (8) is calculated as

$$\begin{aligned}\frac{\partial \hat{\mathbf{v}}_0(n)}{\partial \mathbf{v}^T(n)} &= \sum_{j=1}^{R}\left(\frac{\partial \mathbf{p}_j}{\partial \mathbf{v}^T(n)}d_j \mathbf{p}_j^T \mathbf{v}(n) + \mathbf{p}_j \frac{\partial d_j}{\partial \mathbf{v}^T(n)}\mathbf{p}_j^T \mathbf{v}(n)\right. \\ &\quad \left. + \mathbf{p}_j d_j \mathbf{p}_j^T + \mathbf{p}_j d_j \mathbf{v}(n)\frac{\partial \mathbf{p}_j}{\partial \mathbf{v}^T(n)}\right) \end{aligned} \qquad (10)$$

where $d_j = (l_j - \hat{\sigma}_R^2)/l_j$.

According to the theorem regarding the derivatives of eigenvectors, the partial derivatives of the eigenvectors and eigenvalues are calculated as [7]:

$$\frac{\partial \mathbf{p}_j}{\partial \mathbf{v}^T(n)} = \frac{1}{N}\sum_{i \neq j}\frac{\left(\mathbf{p}_j^T \mathbf{z}(n)\right)\mathbf{p}_i \mathbf{p}_i^T + \left(\mathbf{p}_i^T \mathbf{z}(n)\right)\mathbf{p}_i \mathbf{p}_j^T}{l_j - l_i}, \qquad (11)$$

$$\frac{\partial l_j}{\partial \mathbf{v}^T(n)} = \frac{2}{N}\mathbf{v}^T(n)\mathbf{p}_j \mathbf{p}_j^T, \qquad (12)$$

$$\mathrm{tr}\left(\frac{\partial \mathbf{p}_j}{\partial \mathbf{v}^T(n)}\right) = \frac{1}{N}\sum_{i \neq j}\frac{\mathbf{p}_j^T \mathbf{v}(n)}{l_j - l_i}. \qquad (13)$$

Using (10), (12), and (13), (7) becomes (see the details in the appendix)

$$\begin{aligned}\mathfrak{R}_R &= (K - R)\hat{\sigma}_R^2 + \sum_{j=1}^{R}\frac{\hat{\sigma}_R^4}{l_j} + 2\sigma_e^2 R - 2\sigma_e^2\hat{\sigma}_R^2\sum_{j=1}^{R}\frac{1}{l_j} \\ &\quad + \frac{4\sigma_e^2\hat{\sigma}_R^2}{N}\sum_{j=1}^{R}\frac{1}{l_j} + \frac{2\sigma_e^2}{N}\sum_{j=1}^{R}d_j\sum_{i \neq j}\frac{l_j + l_i}{l_j - l_i} - K\sigma_e^2. \end{aligned} \qquad (14)$$

By dropping the parts that do not depend on R, we obtain

$$\begin{aligned}\mathfrak{R}_R &= (K - R)\hat{\sigma}_R^2 + \sum_{j=1}^{R}\frac{\hat{\sigma}_R^4}{l_j} + 2\sigma_e^2 R - 2\sigma_e^2\hat{\sigma}_R^2\sum_{j=1}^{R}\frac{1}{l_j} \\ &\quad + \frac{4\sigma_e^2\hat{\sigma}_R^2}{N}\sum_{j=1}^{R}\frac{1}{l_j} + \frac{2\sigma_e^2}{N}\sum_{j=1}^{R}d_j\sum_{i \neq j}\frac{l_j + l_i}{l_j - l_i}. \end{aligned} \qquad (15)$$

Every variable on the right-hand side of (15) can be calculated from the eigenvalues except for the noise variance σ_e^2. The σ_e^2 can be estimated from the random matrix theory [7], but this theory considers the general noise, rather than the error of the NMF model for AMT. Therefore, a new method is proposed for estimating σ_e^2 in the next section.

Unfortunately, the assumption that $\mathbf{e}$ has a Gaussian distribution is not strictly satisfied. The distribution of $\mathbf{e}$ is more similar to the Laplacian than the Gaussian. However, direct development of the SURE function for the Laplacian is quite difficult. Therefore, we use the Gaussian assumption like the other SURE-based method [7]. The SURE-based method based on the Laplacian will be considered in future
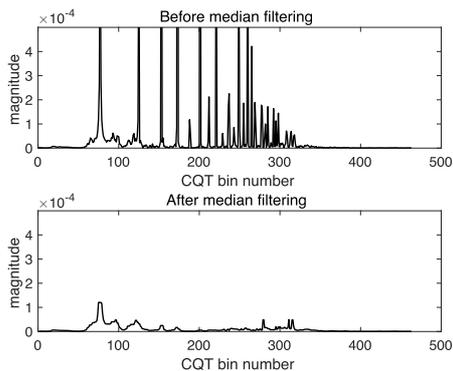
Fig. 2 Magnitude spectrum of a frame from a piano music signal. Upper half: original spectrum. Lower half: spectrum after median filtering.

Table 1 Summary of the proposed algorithm.

Summary of the estimation algorithm
1) Estimate the noise variance from (16) and (17).
2) Calculate the covariance matrix $\mathbf{R_V} = \mathbf{VV}^T/N$ for all of the data.
3) Calculate the eigenvalues of $\mathbf{R_V}$ via the eigenvalue decomposition.
4) Calculate $\Re_R$ with the eigenvalues of $\mathbf{R_V}$ using (15).
5) Determine the rank $\hat{R}$ by $\hat{R} = \arg\min_R \Re_R.$

work.

3. Noise Variance Estimation

As shown in Fig. 1, $\mathbf{V}_0$ and $\mathbf{E}$ consist of the harmonic component and remaining component, respectively; therefore, $\mathbf{E}$ can be roughly estimated by removing the harmonic component. Most of the harmonic component in a music signal is a line spectrum that can be removed by a median filter [8].

Figure 2 shows a magnitude spectrum before and after the median filtering: the upper and lower graphs show the spectrums acquired by the constant-Q transform (CQT) [11] before and after filtering, respectively. The median filter can remove the line spectrum without removing the non-harmonic components.

In the proposed noise variance estimation method, the element of the remaining component at the k-th row and n-th column $\hat{e}_{med}(k,n)$ is estimated by the median filtering applied to each spectrum as
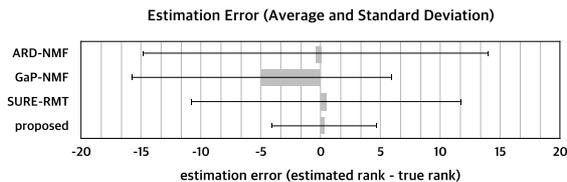
$$\hat{e}_{med}(k,n) = \text{med}\left\{ v(i,n), k - \frac{M-1}{2} \geq i \geq \left(k + \frac{M-1}{2}\right) \right\} \quad (16)$$

where med$\{x\}$ is the median value of x, and the median filter length M is an odd number. Finally, σ_e^2 is estimated as

$$\sigma_e^2 = \frac{1}{KN}\sum_{k=1}^{K}\sum_{n=1}^{N}|\hat{e}_{med}(k,n)|^2 - \left\{ \frac{1}{KN}\sum_{k=1}^{K}\sum_{n=1}^{N}\hat{e}_{med}(k,n) \right\}^2 \quad (17)$$

4. Experiment

The proposed algorithm was evaluated by performing an ex-

Fig. 3 Average and standard deviation of estimation errors.

periment with 30 piano signals from 'AkPnCGdD' of the MIDI-aligned piano sounds (MAPS) database [12]. Each signal was trimmed to 30 seconds long and transformed using CQT [11] into 24 frequency bins per octave from 27.5 Hz to 22050 Hz (half of the sampling rate). The magnitudes of the CQT spectrograms were used as input values. The ground truth data were calculated as follows: obtain the pitch and onset/offset information from the MIDI file, extract the harmonic components of the spectrogram by removing the energy of the CQT bins that does not correspond to harmonics of f0, and calculate the matrix rank of the harmonic components. The number of note events of each sample ranged from 11 to 51.

The conventional SURE with the random matrix theory (SURE-RMT) [7] and the NMF rank estimation methods, gamma process NMF (GaP-NMF) [5], NMF with automatic relevance determination (ARD-NMF) [6], were compared with the proposed algorithm. The hyperparameters a, b, and α of the GaP-NMF were set to 0.5, 0.1, and 1.0, and the model parameters a and ϕ of the ARD-NMF were set to 25.0 and 1.0, respectively. The median-filter length of the proposed algorithm was set to 3. Each basis of the GaP-NMF is considered an active basis when $\theta_r > \tau$, where θ_r is a gain of the r-th basis in the GaP-NMF and τ is a threshold of 10^{-5}. The input data of the comparison methods were pre-processed to extract the harmonic components, as described in [8]. The median-filter lengths of time frame (l_{perc}) and frequency slice (l_{harm}) were 9 and 501, respectively, for the fully rasterized CQT spectrogram provided by the CQT toolbox [11].

Figure 3 shows the average and standard deviation of the estimation error that is the estimated value minus the ground truth of each algorithm. The horizontal bars denote the average error, and the error bars denote the range between the average ± standard deviation values. The average errors of the GaP-NMF, ARD-NMF, and SURE-RMT were −4.93, −0.4, and 0.5 with standard deviations of 10.89, 14.42, and 11.3, respectively; the average error of the proposed algorithm was −0.26 with a standard deviation of 4.4, which indicates a better performance than the other tested algorithms.

5. Conclusion

In this paper, we report the development of a method for estimating the rank of the harmonic components of a spectrogram based on the SURE algorithm. The proposed algorithm estimates the standard deviation of remaining compo-

nents for the SURE by suppressing the harmonic component via the median filtering. Simulation results using MAPS data show that the average error of the proposed algorithm was -0.26, with a standard deviation of 4.4 which delivers a better performance than that of the conventional algorithms.

Acknowledgments

References

[1] P. Smaragids and J.C. Brown, "Non-Negative Matrix Factorization for Polyphonic Music Transcription," IEEE Workshop on Appl. Sig. Proc. Audio. Acoust. (WASPAA 2003), pp.177–180, IEEE, 2003.

[2] E. Vincent, N. Bertin, and R. Badeau, "Adaptive Harmonic Spectral Decomposition for Multiple Pitch Estimation," IEEE Trans. Audio. Speech. Lang. Proc., vol.18, no.3, pp.528–537, 2010.

[3] K. Ochiai, H. Kameoka, and S. Sagayama, "Explicit Beat Structure Modeling for Non-negative Matrix Factorization-based Multi-pitch Analysis," Int. Conf. Acoust. Speech. Sig. Proc. (ICASSP), pp.133–136, IEEE. 2012.

[4] S.H. Park, S. Lee, and K.-M. Sung, "Polyphonic Music Transcription by Nonnegative Matrix Factorization with Harmonicity and Temporal Criteria," IEICE Trans. Fundamentals, vol.E95-A, no.9, pp.1610–1614, 2012.

[5] D.M. Biel, P.R. Cook, and M. Hoffman, "Bayesian nonparametric matrix factorization for recoded music," Proc. 27th Int. Conf. Machine Learning (ICML-10), pp.439–446, IMLS, 2010.

[6] V.Y.F. Tan and C. Févotte, "Automatic Relevance Determination in Nonnegative Matrix Factorization with the β-Divergence," IEEE Trans. Pattern Anal. Mach. Intell., vol.35, no.7, pp.1592–1605, 2013.

[7] M.O. Ulfarsson and V. Solo, "Dimension Estimation in Noisy PCA with SURE and Random Matrix Theory," IEEE Trans. Signal Process., vol.56, no.12, pp.5804–5816, 2008.

[8] D. Fitzgerald, "Harmonic/percussive separation using median filtering," 13th Intern. Conf. Digit. Audio Effects (DAFX10), 2010.

[9] D.D. Lee and H.S. Seung, "Algorithms for non-negative matrix factorization," Adv. Neural Inform. Process. Sys., pp.556–562, 2001.

[10] Y.C. Eldar, "Generalized Sure for Exponential Families: Applications to Regularization," IEEE Trans. Signal Process., vol.57, no.2, pp.471–481, 2009.

[11] C. Schörkhuber, A. Klapuri, N. Holighaus, and M. Dörfler, "A Matlab toolbox for efficient perfect reconstruction time-frequency transforms with log-frequency resolution," 53rd AES Int. Conf. Seman. Audio, AES, 2014.

[12] V. Emiya, R. Badeau, and B. David, "Multipitch Estimation of Piano Sounds using a New Probabilistic Spectral Smoothness Principle," IEEE Trans. Audio, Speech, Lang. Process., vol.18, no.6, pp.1643–1654, 2010.

Appendix A: Derivation of the Risk Function

Using (13) and $(1/N) \sum_{n=1}^{N} (\mathbf{v}^T(n)\mathbf{p}_j)^2 = l_j$, the time average of the trace of the first term of (10) is [7]

$$\frac{1}{N} \sum_{n=1}^{N} \text{tr} \left(\sum_{j=1}^{R} \frac{\partial \mathbf{p}_j}{\partial \mathbf{v}^T(n)} d_j \mathbf{p}_j^T \mathbf{v}(n) \right)$$

$$= \frac{1}{N} \sum_{n=1}^{N} \sum_{j=1}^{R} \text{tr} \left(\frac{\partial \mathbf{p}_j}{\partial \mathbf{v}^T(n)} \right) d_j \mathbf{p}_j^T \mathbf{z}(n)$$

$$= \frac{1}{N} \sum_{j=1}^{R} \sum_{i \neq j} \frac{d_j}{l_j - l_i} l_j. \tag{A·1}$$

Furthermore, the partial derivative of d_j is calculated using (12) as

$$\frac{\partial d_j}{\partial \mathbf{v}^T(n)} = \frac{\partial}{\partial \mathbf{v}^T(n)} \left(1 - \frac{\hat{\sigma}_R^2}{l_j} \right)$$

$$= \left(\frac{\hat{\sigma}_R^2}{l_j^2} \right) \frac{2}{N} \mathbf{v}^T(n) \mathbf{p}_j \mathbf{p}_j^T. \tag{A·2}$$

Therefore, the time average of the trace of the second term of (10) is calculated as

$$\frac{1}{N} \sum_{n=1}^{N} \text{tr} \left(\sum_{j=1}^{R} \mathbf{p}_j \frac{\partial d_j}{\partial \mathbf{v}^T(n)} \mathbf{p}_j^T \mathbf{v}(n) \right)$$

$$= 2 \frac{1}{N} \sum_{j=1}^{R} \frac{\hat{\sigma}_R^2}{l_j^2} \frac{1}{N} \sum_{n=1}^{N} \left(\mathbf{p}_j^T \mathbf{v}(n) \right)^2$$

$$= 2 \frac{1}{N} \sum_{j=1}^{R} \frac{\hat{\sigma}_R^2}{l_j}. \tag{A·3}$$

Similarly, the time averages of the traces of the third term and the fourth term of (10) are calculated as

$$\frac{1}{N} \sum_{n=1}^{N} \text{tr} \left(\sum_{j=1}^{R} \mathbf{p}_j d_j \mathbf{p}_j^T \right) = R - \hat{\sigma}_R^2 \sum_{j=1}^{R} \frac{1}{l_j} \tag{A·4}$$

and

$$\frac{1}{N} \sum_{n=1}^{N} \text{tr} \left(\sum_{j=1}^{R} \mathbf{p}_j d_j \mathbf{v}^T(n) \frac{\partial \mathbf{p}_j}{\partial \mathbf{v}^T(n)} \right) = \frac{1}{N} \sum_{j=1}^{R} \sum_{i \neq j} \frac{d_j}{l_j - l_i} l_i \tag{A·5}$$

respectively. Finally, the first term of (7) is calculated as

$$\frac{1}{N} \sum_{n=1}^{N} \|\mathbf{v}(n) - \hat{\mathbf{v}}_0(n)\|^2$$

$$= \frac{1}{N} \sum_{n=1}^{N} \|\mathbf{v}(n) - \sum_{j=1}^{R} \mathbf{p}_j d_j \mathbf{p}_j^T \mathbf{v}(n)\|^2$$

$$= \sum_{j=1}^{K} l_j + \sum_{j=1}^{R} \left(-2 l_j d_j + d_j^2 l_j \right)$$

$$= (K - R)\hat{\sigma}_R^2 + \sum_{j=1}^{R} \frac{\hat{\sigma}_R^4}{l_j}, \tag{A·6}$$

because $(1/N) \sum_{n=1}^{N} \|\mathbf{v}(n)\|^2 = \sum_{j=1}^{K} l_j$ and $(1/N) \sum_{n=1}^{N} (\mathbf{v}^T(n)\mathbf{p}_j)^2 = l_j$. The detailed derivation can be also found in [7].

Substituting (A·1), (A·3), (A·4), (A·5), and (A·6) instead of each term of (7), the risk function (14) is obtained.