ANTI-MODEL KL-SVM-NAP SYSTEM FOR NIST SRE 2012 EVALUATION

Hanwu Sun, Kong Aik Lee and Bin Ma

Institute for Infocomm Research (I^2R), A*STAR, Singapore 138632

{hwsun, kalee, mabin}@i2r.a-star.edu.sg

ABSTRACT

This paper presents an anti-model based speaker recognition system for NIST SRE 2012 evaluation, which is one of subsystems in IIR SRE12 submission. We apply the anti-model approach for the SRE12 evaluation. The KL-SVM-NAP based speaker recognition system is adopted to evaluate the performance. We present detailed comparison study of the classical KL-SVM-NAP based speaker recognition system and anti-model based KL-SVM-NAP system for NIST 2012 speaker recognition evaluation. The results are reported on in-house pre-SRE12 development set and NIST SRE12 core task. The clear advantages of the anti-model approach over that the traditional KL-SVM-NAP approach are presented and discussed.

Index Terms: speaker recognition, anti-model, Nuisance Attribute Projection

1. INTRODUCTION

The 2012 *Speaker Recognition Evaluation* (SRE12) is one of an ongoing series of benchmarking events conducted by NIST [1]. The basic recognition task specified in NIST SREs is speaker detection, i.e., to determine whether a specified target speaker is speaking during a given segment of speech.

The most recent SRE12 [1] is distinguished from previous NIST evaluations [2] by allowing the use of information of all target speakers in each detection trial. This differs from previous SREs, where the system is restricted to use only the knowledge of the target speaker specified in the detection trial [2]. This essentially changes the definition of the alternative hypothesis in the detection task, in which the test segment given in a trial can now be assumed to come from other speakers in the target set as well as some unseen non-target speakers. (Note: the null hypothesis being that the test segment is from a specific target speaker).

The definition of the alternative hypothesis as mentioned above has long been adopted in the *Language Recognition Evaluations* (LREs), also conducted by NIST [3]. For language detection task, where the target classed being the languages instead of speakers, the anti-model approach [4] has shown to be effective in separating near competitors. The results reported in [4, 5] demonstrated that the anti-model approach provides obvious advantages over conventional approach in language detection.

Though definition of the compound form of alternative hypothesis is always a matter of debate, it is interesting to study the feasibility of applying the anti-model approach on speaker detection task, using a large-scale evaluation platform as provided in SRE12. In particular, we apply the anti-model approach on the KL-SVM-NAP system [6, 7], and study the benefit of the anti-model-based KL-SVM-NAP compared to the classical KL-SVM-NAP on SRE12 evaluation set. In addition to the anti-model approach, we also show that the compound hypothesis could be established by simple manipulating of SVM scores.

The paper is organized as follows. In Section 2, we give an overview of the classical KL-SVM-NAP speaker recognition system. The anti-model KL-SVM-NAP system and the compound likelihood formulation are introduced in Section 3. In Section 4, the frontend feature processing for the SRE12 noisy data and the development of pre-SRE12 dataset are presented. The experimental results and analysis are reported also in Section 4. Finally, we conclude the paper in Section 5.

2. KL-SVM-NAP SPEAKER RECOGNITION SYSTEM

The speaker recognition system in this study is based on the support vector machine (SVM) using the *Kullback-Leibler* (KL) divergence kernel and the *nuisance attribute projection* (NAP) technique for channel compensation, in short, the KL-SVM-NAP as reported in [6, 7]. The fundamental idea here is to represent variable-length utterances, for training or test, as high- dimensional vectors referred to as the GMM supervectors. Channel compensation and speaker detection are then performed in the high-dimensional vector space. The discriminative nature of the SVM classifier allows a straightforward implementation of the anti-model approach as detailed in the next section.

Let $\Lambda = \{\ \omega_i, \mathbf{\mu}_i, \mathbf{\Sigma}_i; i = 1, 2, \ldots M\ \}$ be the parameters of the universal background model (UBM), where M is the number of mixture components, ω_i are the mixture weights, $\mathbf{m}_i$ are the mean vectors, and $\mathbf{\Sigma}_i$ are the covariance matrices assumed to be diagonal. The results generalize to the case of full-covariance matrices [8], which has shown to be useful for NIST detection tasks given rich amount of training data available in the order of hundreds hours. For a given utterance X_s, the Baum-Welch statistics are used to adapt the mean vectors of the UBM using the maximum *a posterior* (MAP) criterion. The adapted mean vectors are concatenated to form a GMM supervector, as follows:

$$\mathbf{m}(s) \equiv \left[\mathbf{m}_1^{\mathrm{T}}(s), \mathbf{m}_2^{\mathrm{T}}(s), \ldots, \mathbf{m}_M^{\mathrm{T}}(s)\right]^{\mathrm{T}}, \quad (1)$$

where T denotes transposition. The mean vectors are then normalized by its standard deviation and weighted by the squared root of the mixture weights:

$$\mathbf{m}_i'(s) = \sqrt{\omega_i}\,\mathbf{\Sigma}_i^{-1/2}\mathbf{m}_i(s),\ i = 1, 2, \ldots, M\ . \quad (2)$$

The normalization in (2) allows the similarity between two GMM supervectors to be computed by taking their inner product in accordance to the KL-divergence [6, 7].

Channel compensation is then applied on the normalized supervector via linear projection [6, 7]. Speaker recognition is then performed using the normalized and channel-compensated GMM supervectors with SVM. Typically, one SVM is trained for each target speaker using the one-versus-all strategy. Let Ω_k be the set of supervectors pertaining to the target speaker (i.e., the positive examples), and $\Re$ the set of supervectors pertaining to some background speakers. An SVM solver for the dual formulation [6, 7],

$$f_k \leftarrow \mathrm{SVM}\left(\Omega_k \,\|\, \Re\right), \tag{3}$$

returns the Lagrange multipliers α associated to all the supervectors in the training set and a bias parameter β, which essentially forms a linear model f_k for a target speaker, as follows:

$$f_k(\mathbf{m}') = \left[\sum\nolimits_{\Omega_k} \alpha_i \mathbf{m}'(i) - \sum\nolimits_{\Re} \alpha_j \mathbf{m}'(j)\right]^{\mathrm{T}} \mathbf{m}' + \beta\,. \tag{4}$$

Notice that, the same background set $\Re$ is used for all target speakers enrolled to the system. In the next section, we show that significant improvement could be obtained by changing the selection of the training set $\{\Omega_k, \Re\}$.

3. ANTI-MODEL APPROACH FOR SPEAKER RECOGNITION

In the following, we describe two approaches to deal with the compound, or composite, form of alternative hypothesis for the detection task as introduced in the recent NIST SRE12.

3.1. Anti-model KL-SVM-NAP System

In a speaker detection task, system performance is evaluated by presenting the system with a set of trials, each consisting of a test segment and a hypothesized identity. The system has to decide, for each trial, to accept or reject the hypothesized identity. Let *N* be the number of target speakers enrolled in the recognition system. The identity assumed in the null hypothesis (i.e., the hypothesized identity) is constrained to be one of the *N* target speakers enrolled in the system. It is customary to assume an alternative hypothesis that a test segment belongs to an unseen non-target speaker. This allows the target speakers to be treated independently in evaluating the system performance.

The definition of the alternative hypothesis as mentioned above has been a matter of debate. Different from previous SREs, one new challenge added to SRE12 is that it allows the use of the knowledge of all target speakers in each detection trial. This essentially changes the definition of the alternative hypothesis, in which the test segment given in a detection trial can now be assumed to come from other speakers in the target set in addition to the unseen non-target speakers.

As far as the SVM is concerned, these changes lead to the necessary modification on the negative training samples for a better modeling of the alternative hypothesis. To this end, the characteristic of the unseen non-target speakers can be learned from the background dataset $\Re$. The effect of other (N – 1) competing speakers from the target set can be taken into account using the anti-model approach [4]. In essence, we augment the background training set $\Re$ with the training data from other competing speakers. Following the same notation as above, the anti-model KL-SVM-NAP training is expressed as follows:

$$f_{k,\mathrm{anti}} \leftarrow \mathrm{SVM}\left(\Omega_k \,\|\, \Re \cup \bigcup\nolimits_{l=1,l\neq k}^{N} \Omega_l\right). \tag{5}$$

for k = 1, 2, …, N. Having the training sets as defined above, a better discrimination between speakers in the target set and with other unseen speakers can be obtained, as we shall demonstrate in Section 4.

3.2. Compound Likelihood

For a given trial *t*, the output score of the SVM classifier as given in (4) can be interpreted as a log-likelihood ratio, scaled and shifted by the factors, *a* and *b*, respectively, as follows:

$$f_k(t) = a\left[\log s_k(t)\right] + b = \log\left[\exp(b)\, s_k^a(t)\right]. \tag{6}$$

Here, $s_k(t)$ is the canonical likelihood ratio without the effect of scaling and shifting for the *k*-th target speaker

$$s_k(t) = \frac{P(\mathrm{X}_t \mid H_k)}{P(\mathrm{X}_t \mid H_{N+1})}\text{, for k = 1, 2, …, } N, \tag{7}$$

where H_{N+1} is the hypothesis that the test segment X_t is from some unseen speakers excluding other (*N-1*) speakers from the target set. Taking these competing speakers into the alternative hypothesis we form the following compound likelihood-ratio [9]:

$$\tilde{s}_k(t) = \frac{P^a(\mathrm{X}_t \mid H_k)}{\frac{P_{\mathrm{Known}}}{N-1}\sum\limits_{l=1,l\neq k}^{N} P^a(\mathrm{X}_t \mid H_l) + (1-P_{\mathrm{Known}})\, P^a(\mathrm{X}_t \mid H_{N+1})}, \tag{8}$$

where P_{Known} gives a proper weight between the two categories of known and unknown (or unseen) speakers. Using (6) and (7) in (8) we arrive at the following compound log-likelihood ratio:

$$\log\left[\tilde{s}_k(t)\right] = f_k(t) - \log\left\{\frac{P_{\mathrm{Known}}}{N-1}\sum_{l=1,l\neq k}^{N} \exp\left[f_l(t)\right] + (1-P_{\mathrm{Known}})\right\}. \tag{9}$$

The above equation copes with the compound alternative hypothesis by manipulating the output scores of SVMs.

4. SPEAKER RECOGNITION EXPERIMENTS

The experiments were conducted on the NIST SRE12 and a development set designed for pre-SRE12 evaluation. In the following, we give a brief description on the front-end feature processing and the development set designed to evaluate system robustness against noisy test segments. We then focus on the analysis of the KL-SVM-NAP with and without the anti-model training.

4.1. Frontend Feature Processing for Noisy Speech

We use the MFCC feature in this study. In particular, a 16-dimenison MFCC features were generated for each speech frame with a window of 30ms and a frame shift of 12.5ms. By including the 16-dimension first and second derivatives, the resulting MFCC feature vector consists of 48 elements.

One new challenge added to NIST SRE12 is noisy test segments. These could be clean speech segments corrupted by crowd noise or HVAC-like noise (note: HVAC is shorthand for *heating, ventilation, and air conditioning*) [10]. There are also some test segments collected under noisy environment, from which

the Lombard effect could be observed. As shown in Fig. 1, the noisy speech is first processed with an ETSI based Wiener filter [11] prior to feature extraction. A second stage of noise reduction is then performed using spectral subtraction technique [12] to assist the voice activity detection (VAD) in selecting useful speech frames [13]. It is worth mentioning that the spectral subtracted signal is used for frame selection, while the MFCC features are directly derived from the Wiener filter. The MFCC feature vectors are then processed by RASTA filtering [14] and followed by *mean-and-variance* normalization (MVN). Notice that, in Fig. 1, MVN is performed on the selected frames while RASTA filtering is perform on the whole sequence.

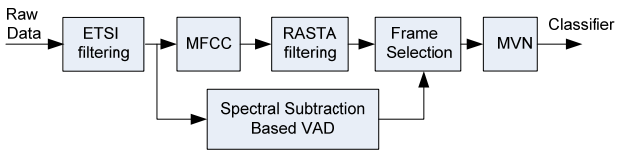

Fig 1. Frontend MFCC feature extraction.

4.2. Development, Train, and System Configuration

We designed a pre-SRE12 development set using speech segments drawn from SRE06, SRE08 and SRE10. The development set consists of two disjoint partitions, DEV and EVAL, each consisting of a train and test sets. The number of train segments in EVAL-train is about twice the number of segments in DEV-train. Considering noisy segments in actual SRE12 [1], we added two type noises to DEV-test and EVAL-test signals by using the *FaNT* [15] tool. The noise types used are HVAC [16] and crowd noises [17]. Test segments are corrupted at 6-dB and 15-dB SNR level.

SRE04 data was used to form the background dataset $\Re$ and also for training the UBM (gender dependent and 1024 mixtures). Meanwhile, the NAP matrix was trained with data drawn from SRE04, SRE06, SRE08, SRE08-followup and SRE10, but it did not include neither utterances involving in Dev-test or Eval-test data, as well as Eval-train utterances. The rank of NAP was set to be 60 in the experiments. SVMTorch [18] was used to train SVM models. TZnorm was used for score normalization [19], where SRE05 data was used for training the cohort models for Tnorm while SRE04 data was used as imposture utterances for Znorm.

4.3. Results and Analysis

Experiments were conducted on the pre-SRE12 development sets and SRE12 core task to investigate the effectiveness of the anti-model training approach and conversion of log-likelihood ratio based on the compound alternative hypothesis as in (8) and (9).

4.3.1. Results on pre-SRE12 DEV and Eval sets

The experimental results on DEV set under clean, 6-dB noisy, and 15-dB noisy conditions are shown in Table 1. The clean and 6-dB tasks' DET plots are also illustrated in Fig. 2.

From Table 1, we can see that the anti-model approach has overall better performances in both EER and DCF when *simple log-likelihood ratio* (Simp) is used. There are 35% and 45% relative improvements on the EER and Minimum DCF, respectively, in average across clean, 6-dB-noisy and 15-dB-noisy conditions.

When the compound log-likelihood ratio (Comp), as in (9), was applied to both scores, the EER and minimum DCF significantly reduce. After the compound log-likelihood conversion, the EER for both systems were almost identical. However, the minimum DCF are still better for KL-SVM with anti-model training for all three conditions. There are about 15% to 20% relative improvements in terms of minimum DCF for all the three test conditions. The DET curves in Fig. 2 illustrate the obvious advantages of anti-model training (solid line) over the case conventional approach (dashed line) at two DCF points (shown in green dot and red dot).

Table 1. EER and minimum DCF on the pre-SRE12 DEV set for the KL-SVM system with and without anti-model training, and/or conversion to compound log-likelihood ratio.

Score Type	Test Cond.	KL-SVM		KL-SVM-anti	
		EER%	DCF	EER%	DCF
Simp	Clean	0.721	0.089	0.427	0.039
	15dB	1.366	0.139	0.849	0.069
	6dB	3.348	0.261	2.508	0.175
Comp	Clean	0.296	0.032	0.312	0.026
	15dB	0.558	0.062	0.505	0.050
	6dB	1.797	0.163	1.710	0.138

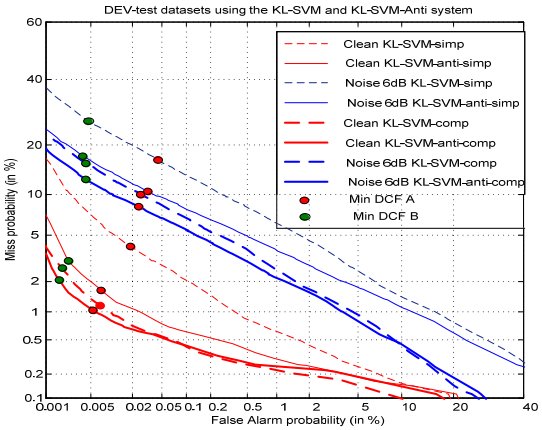

Fig 2. Pre-SRE12 DEV set under clean and 6-db-noisy conditions for the KL-SVM system with and without anti-model training, and/or conversion to compound log-likelihood ratio.

Table 2. EER and min DCF on the pre-SRE12 EVAL set.

Score Type	Test Cond.	KL-SVM		KL-SVM-anti	
		EER%	DCF	EER%	DCF
Simp.	Clean	0.577	0.093	0.532	0.050
	15dB	1.202	0.203	1.057	0.130
	6dB	3.680	0.429	3.041	0.294
Comp.	Clean	0.343	0.045	0.469	0.038
	15dB	0.704	0.126	0.733	0.107
	6dB	2.439	0.311	2.361	0.256

Similar experiments were conducted on the pre-SRE12 EVAL set. The results are shown in Table 2. Similar results as in the DEV set are observed on the EVAL set. Anti-model training improves

the performance. Conversion to the compound log-likelihood ratio improves further the performance especially on the DCF points where the decision thresholds are relatively higher.

4.3.2. Results on SRE12 Core Task

The actual SRE12 core task results are shown in Table 3 and Fig. 3. In SRE12 core task [1], five common conditions (or subtasks) were defined based on the quality of the test segments, namely, *clean-interview*, *clean-telephone*, *interview with added noise*, *telephone with added noise*, and *telephone collected in noisy room*.

Table 3. The EER and minimum DCF for the five common conditions of the SRE12 core task.

Score Type	Test Cond.	KL-SVM		KL-SVM-anti	
		EER%	DCF	EER%	DCF
Simp.	Int-clean	4.559	0.347	3.516	0.268
	Tel-clean	2.469	0.2943	2.409	0.232
	Int-noise	4.276	0.283	3.622	0.188
	Tel-noise	2.813	0.376	2.587	0.281
	Tel-room-noise	3.117	0.339	2.926	0.262
Comp.	Int-clean	3.964	0.278	3.357	0.253
	Tel-clean	1.768	0.209	1.991	0.198
	Int-noise	4.173	0.247	3.458	0.187
	Te.-noise	2.056	0.299	2.035	0.245
	Tel-room-noise	2.190	0.238	2.388	0.219

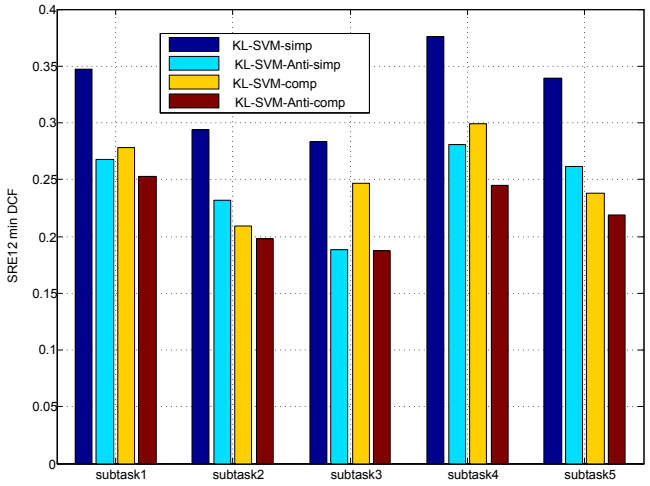

Fig 3. Minimum DCF across five common conditions in SRE12 core task.

For all the five subtasks, both EER and DCF improve significantly when anti-model training is applied. In particular, 10% and 25% of relative improvement can be observed on the EER and minimum DCF, respectively. For the case of compound log-likelihood ratio, anti-model gives 13% relative improvement on DCF though no significant improvement on EER can be observed. These results are consistent with that observed on the pre-SRE12 DEV and EVAL sets. Fig. 3 illustrates clearly the advantage of anti-model training and compound log-likelihood conversion in terms of minimum DCF values.

4.3.3. Results on SRE12 Core Task for Known and Unknown Non-target Trials

Since anti-model training uses different background data in training each speaker model, it is important to check how the system performs on the "known" and "unknown" non-target trials in the SRE12 core task [1]. Briefly, imposter trials whereby the test segments not belong to any of the target speakers are referred to as the "unknown" non-target trials. We analyze the relative EER and DCF changes between known and unknown tasks in SRE12 core test. Fig. 4 shows the results for the five subtasks (using compound log-likelihood). From the figure, we can see that both EER and DCF exhibit similar trends across all five test conditions. This result indicates that anti-model training (solid line) works well for both "known" and "unknown" non-target trials.

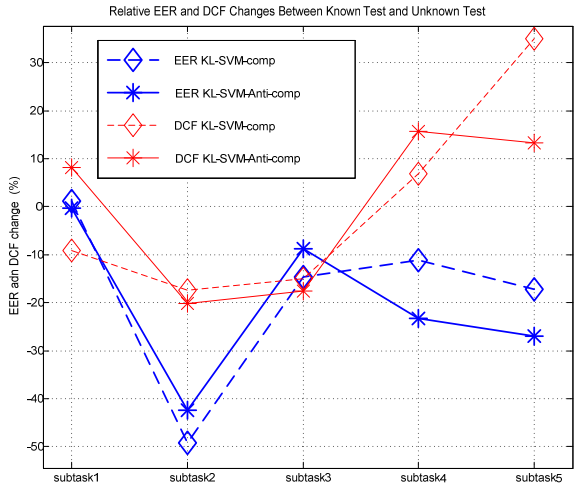

Fig 4. EER and DCF relative changes under five subtests in SRE12 core test for classical and KL-SVM-anti systems.

5. CONCLUSIONS

We presented an anti-model training approach to SVM based speaker recognition system. We also showed the effectiveness of compound log-likelihood ratio for the case when the training data from all target speakers can be used for each trials. Results on the pre-SRE12 and the actual SRE12 core test sets show the significant advantage of the anti-model training approach in both EER and DCF. For the case of compound log-likelihood ratio, anti-model training shows obvious benefit on DCF points (where the decision thresholds are relatively higher), though no significant improvement on EER can be observed. In addition, the anti-model approach also works well for "unknown" non-target trials. As future works, we will look into applying anti-model training for i-vector based classifier.

6. REFERENCES

[1] NIST 2012 Speaker Recognition Evaluation Plan, http://nist.gov/itl/iad/mig/upload/NIST_SRE12_evalplan-v17-r1.pdf.

[2] NIST 2008 Speaker Recognition Evaluation Plan, http://www.itl.nist.gov/iad/mig//tests/sre/2008/sre08_evalplan_release4.pdf.

[3] The 2011 NIST Language Recognition Evaluation Plan , http://www.nist.gov/itl/iad/mig/upload/LRE11_EvalPlan_releasev1.pdf.

[4] P. Matejka, P. Schwarz, L. Burget, and J. Cernocky, "Use of anti-models to further improve state-of-art PRLM language recognition system," in Proc. ICASSP, 2006.

[5] X. Yang, M. Siu, H. Gish, B. Mak "Boosting with Anti-models for Automatic Language Identification", in Proc. *Interspeech*, pp342-345, Belgium, 2007.

[6] W.M. Campbell, A. Solomonoff and I Boardman, "Advances in Channel Compensation for SVM Speaker Recognition**"**. in *Proc. ICASSP*, pp. 18-23 Philadelphia, 2005.

[7] W. M. Campbell, D. E. Sturim, D. A. Reynolds, and A. Solomonoff, "SVM based speaker verification using a GMM supervector kernel and NAP variability compensation," in *Proc. ICASSP*, pp. 97–100, 2006.

[8] P. Kenny, "A Small Footprint i-Vector Extractor", in *Speaker and Language Recognition Workshop , Odyssey* , 2012.

[9] https://sites.google.com/site/bosaristoolkit/sre12.

[10] C. Greenberg, A. Martin and M. Przybocki, "The 2011 BEST Speaker Recognition Interim Assessment", in *Speaker and Language Recognition Workshop , Odyssey* , 2012.

[11] ETSI standard document, "Speech Processing, Transmission and Quality aspects (STQ); Distributed *speech recognition; Advan*ced Front-end feature extraction algorithm; Compression algorithm", *ETSI ES 202 050 v1.1.3 (2003-11)*, Nov. 2003.

[12] R. Martin "Spectral Subtraction Based on Minimum Statistics," in *Proc. EUSPICO*, vol. 2, pp.1182–1185, 1994.

[13] H. Sun, B. Ma and H. Li, "An Efficient Feature Selection Method for Speaker Recognition," in *Proc. ISCSLP*, pp. 181–184, 2008.

[14] H. Hermansky and N. Morgan, "RASTA Processing of Speech," *IEEE Transactions on Speech and Audio Processing*, vol. 2, no. 4, pp. 578–589, 1994.

[15] G. Hirsch, "Fant," http://dnt.kr.hs-niederrhein.de/download.html.

[16] http://www.freesound.org/.

[17] NOISEX-92, http://spib.rice.edu/spib/select_noise.html.

[18] R. Collobert and S. Bengio, "SVMTorch: support vector machines for large-scale regression problems," Journal of Machine Learning Research, vol. 1, pp. 143-160, 2001.

[19] R. Auckenthaler, M. Carey, and H. Lloyd-Thomas, "Score Normalization for Text-independent Speaker Verification Systems," Digital Signal Processing, vol. 10, no 1-3, pp. 42–54, Jan 2000.