THE MATCHING-MINIMIZATION ALGORITHM, THE INCA ALGORITHM AND A MATHEMATICAL FRAMEWORK FOR VOICE CONVERSION WITH UNALIGNED CORPORA.

Yannis Agiomyrgiannakis

Google

agios@google.com

ABSTRACT

This paper presents a mathematical framework that is suitable for voice conversion and adaptation in speech processing. Voice conversion is formulated as a search for the optimal correspondances between a set of source-speaker spectra and a set of target-speaker spectra under a transform that compensates speaker differences. It is possible to simultaneously recover a bi-directional mapping between two sets of vectors that is a parametric mapping (a transform) in one direction and a non-parametric mapping (correspondences) in the reverse direction. An algorithm referred to as Matching-Minimization (MM) is formally derived with proven convergence and an optimal closed-form solution for each step. The algorithm is closely related to the *asymmetric-1* variant of the well-known INCA algorithm [1] for which we also provide a proof within the same framework. The differences between MM and INCA are delineated both theoretically and experimentally. MM outperforms INCA in all scenarios. Like INCA, MM does not require parallel corpora. Unlike INCA, MM is suitable when only a few adaptation data are available.

Index Terms— INCA, voice-conversion, voice-transformation, matching-minimization, nearest-neighbour

1. INTRODUCTION

In voice conversion we have a *source speaker X* and a *target speaker Y* and we want to convert the voice of the source speaker to the voice of the target speaker. Assuming that the speech signal is parameterized by some vectors using, e.g. a vocoder [2], the problem effectively becomes one of predicting a sequence of Y-space vectors from a sequence of X-space vectors.

When parallel recordings are available, we can match X/Y-space sequences using a dynamic time warping algorithm [3]. For example, Stylianou et al. [4] proposed a *conversion function* that is closely related to a mixture of linear regressions. Given a GMM of X-space the conversion function is estimated using least-squares. Kain et al. [5] derived the parameters of the conversion function from a GMM of the joint source. Hui Ye et al. [6] proposed a mixture-of-linear-regressions function (MLR) that is quite similar to the conversion function [4] and estimated its parameters using a weighted error criterion.

In many applications it is not easy to obtain parallel recordings. Furthermore, it is not obvious how to use Mean-Squared-Error (MSE) criteria for voice conversion with non-parallel recordings, which has led some researchers to resort to heuristics [7, 8]. On the other hand, likelihood-based criteria are far more suitable for non-parallel corpora. Likelihood-based voice conversion resembles statistical adaptation techniques for Gaussian Mixture Models (GMM) [9], and Hidden Markov Models (HMM) [10]. Mouchtaris et al. [11], proposed a constrained speaker adaptation method that uses reference parallel recordings as anchors. Tokuda et al. [12, 13] use MLLR-based adaptation in the context of HMM-based speech synthesis [14]. Finally, Neural-Network-based speech synthesis [15], [16] can also benefit from adaptation [17–19].

A MSE-based algorithm that attempts to tackle voice conversion with non-parallel recordings is the INCA algorithm [20], [21]. The algorithm iterates three steps: a *nearest neightbor matching step*, a transformation function *training step* using e.g. Kain's et al. method [5] and a *transformation step*, until convergence. The experiments presented by the authors indicate that the algorithm performs similarly to training with parallel recordings. The insights one can get from the original INCA paper [20] are limited because the algorithm was not formally derived.

Some insight was provided by Benisty et al. [1] where an attempt was made to prove that the *symmetric-1* variant of INCA is an iterative minimization approach of the overall *matching distortion* of Y-space vectors to X-space vectors and vice-versa. However, the proof does not go beyond stating properties of alternating minimization [22] and a more formal proof is needed to answer questions like 1) are nearest neighbors the optimal solution for matching, 2) how to efficiently minimize simultaneously a function and its inverse. The later is a hard optimization problem that in practice limits the scope of the transformation function.

A closer look to the distortion criterion in INCA variants reveals that when X/Y-space datasets have substantially different sizes, the criterion is dominated by the matches of the big dataset to the small one. This can occurs when the source is a whole TTS corpus and the target is just a few adaptation utterances. In that case, a phone that exists in the big dataset but not in the small one will only have a bad match, polluting the criterion with bad matches that intuitively should not be used.

This paper presents a probabilistic framework that overcomes the aforementioned deficiency of INCA as a solution to the generic problem of matching datasets under a compensating transform, hereby referred to as Matching-Under-Transform (MUT), for a broad family of transformation functions. An iterative algorithm is formally derived with proven convergence: the Matching-Minimization algorithm (MM). In contrast to [1], a closed-form optimal solution is derived for every step of the iterative process and also provides a short proof of INCA.

MM is derived using deterministic annealing [23] to minimize a weighted MSE criterion. The algorithm recovers a set of hard associations (matches) in the sense that a Y-space vector is associated only with one X-space vector, while the reverse does not hold: an X-space vector is associated with zero or more Y-space vectors.

Section 2 presents a probabilistic formulation of the MUT problem, the inherent bi-directional mapping, the use of deterministic annealing and the MM algorithm. Section 3 clarifies the relationship between MM and INCA and provides a formal proof for the asymmetric-1 variant of INCA. Section 4 experimentally compares MM versus INCA and demonstrates the effect of having diverse size X/Y-space datasets. We report that MM significantly outperforms INCA when Y-space data size is much smaller than X-space data size.

2. THE MATCHING-UNDER-TRANSFORM PROBLEM

Assume that we have N samples from speaker X, $\vec{x}_n \in \Re^P$, $n = 1, ..., N$ and Q samples from speaker Y, $\vec{y}_q \in \Re^D$, $q = 1, ...Q$, sampled from the corresponding spaces, X and Y, respectively. X-space and Y-space vectors cannot be compared directly but only via a transformation function $\vec{y} = F(\vec{x})$ that converts an X-space vector to a Y-space vector. We want to find which X-space vectors $\vec{x}_n$ correspond to a Y-space vector $\vec{y}_q$ in the sense that $F(\vec{x}_n)$ is close to $\vec{y}_q$ in L2 norm. The problem is trivial when we know the transformation function $F(\cdot)$ or the correspondances but it is combinatorial when we don't. A brute-force solution would involve solving for the optimal transform for every possible mapping between X-space and Y-space vectors.

Let $d(\vec{y}_q, \vec{x}_n)$ be a distortion metric between $\vec{x}_n$ and $\vec{y}_q$:

$$d(\vec{y}_q, \vec{x}_n) = (\vec{y}_q - F(\vec{x}_n))^T W_q (\vec{y}_q - F(\vec{x}_n)), \qquad (1)$$

where W_q is a weighting matrix depending on the Y-space vector $\vec{y}_q$. The weighting matrix can be used - for example - to incorporate frequency weighting that provides better fit around Y-space formants as in [6] and/or bandlimiting. Let $p(\vec{y}_q, \vec{x}_n)$ be the joint probability of matching vectors $\vec{y}_q$ and $\vec{x}_n$. Then, the average distortion for all possible vector combinations is:

$$D = \sum_{n,q} p(\vec{y}_q, \vec{x}_n) d(\vec{y}_q, \vec{x}_n) = \sum_q p(\vec{y}_q) \sum_n p(\vec{x}_n | \vec{y}_q) d(\vec{y}_q, \vec{x}_n). \qquad (2)$$

The *association probabilities* $p(\vec{x}_n|\vec{y}_q)$ contain the requested mapping, while the Y-space probabilities are set to be uniformly distributed: $p(\vec{y}_q) = \frac{1}{Q}$. A uniform distribution is chosen as it implies no knowledge, but any prior could be used. Given the way that the distortion is formulated, for every Y-space vector there is at least one X-space vector, while the opposite does not hold, thus there might be X-space vectors that have no match in Y-space. This is a desirable property at least in two cases: a) in a typical intra/cross-lingual voice conversion scenario we have a lot of X-space vectors (i.e. a TTS corpus) and just a few Y-space vectors (i.e. a few utterances), b) in a cross-lingual voice conversion scenario, some X-space sounds may not have their Y-space equivalent.

The ability to ignore some portions of X-space may also become handy in cases where X-space contains noisy or irrelevant information (i.e. silences).

2.1. Understanding the bi-directional mapping

The association probabilities $p(\vec{x}_n|\vec{y}_q)$ and the transformation function $\vec{y} = F(\vec{x})$ operate in different directions. This bidirectional mapping is illustrated in Figure 1: a parametric mapping in the *forward direction* (X → Y) via function $F(\cdot)$ and a non-parametric mapping in the *backward direction* (Y → X) via the association probabilities $p(\vec{x}_n|\vec{y}_q)$. Qualitatively, the overall operation is balanced in the sense that backward mapping counteracts the forward mapping and vice-versa. This is a key property of the formulation of MUT that ensures convergence to a meaningful solution.

To understand the importance of balancing the mappings, let us examine the case where both operate in the forward direction. This is formulated by expressing the *reverse* average distortion formula as:

$$D_{\text{REV}} = \sum_n p'(\vec{x}_n) \sum_q p'(\vec{y}_q|\vec{x}_n) d(\vec{y}_q, \vec{x}_n), \qquad (3)$$

and keeping $p'(\vec{x}_n) = \frac{1}{N}$ constant. There are Q zero distortion solutions for this formulation that are degenerate because they map all $\vec{x}_n$ vectors to one $\vec{y}_{q'}$ vector (i.e. $p'(\vec{y}_{q'}|\vec{x}_n) = 1.0$) with the constant transform: $F(\vec{x}_n) = \vec{y}_{q'}$. Thus, the balanced mapping prevents the existence of degenerate solutions.

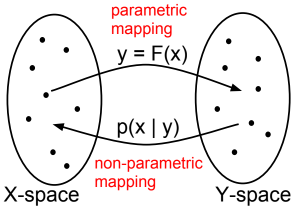

Fig. 1. The bidirectional mapping: X-space is mapped to Y-space via a parametric mapping, while Y-space is mapped back to X-space via a non-parametric mapping.

MUT's dual (parametric/non-parametric) nature makes it a versatile tool. Depending on the application, one may choose to use it in a parametric manner or a non-parametric manner.

2.2. Deterministic Annealing

Minimizing the average distortion D simultaneously for the transformation function and the association probabilities is a non-trivial optimization problem. From the deterministic annealing perspective [23], the associations between X-space and Y-space are always probabilistic and their joint entropy $H(Y, X)$ expresses the fuzziness of the matching. Zero entropy means that we are absolutely sure of an association. Higher entropy indicates our uncertainty on whether an association exist or not. The association entropy can be expressed as $H(Y, X) = H(Y) + H(X|Y)$. The term $H(Y)$ is fixed because in the formulation we made in the previous section we asserted that Y-space probabilities are fixed $p(\vec{y}_q) = \frac{1}{Q}$ to ensure that all Y-space vectors were equally taken into account. The level of the uncertainty of the associations is typically a design parameter that reflects the level of trust one has on the associations.

Deterministic annealing simultaneously minimizes the average distortion and the association entropy to find a solution that takes into account both the distortion and the uncertainty of the associations. This is made by augmenting the average distortion with the association entropy $H(Y, X) = H(Y) + H(X|Y)$ or equivalently by $H(X|Y)$ since $H(Y)$ is assumed to be constant. The entropic term fuzzifies the optimal association probabilities so that a Y-space vector can be mapped to more than one X-space vectors. Following [23] we define the composite minimization criterion D' as:

$$D' = D - \lambda H(X|Y), \qquad (4)$$

where the entropy Lagrangian λ is related to the annealing temperature.

The Lagrangian can be used to control the type of backward mapping. When λ is zero, the mapping between Y-space and X-space is many-to-one (many Y-space vectors may be mapped to one X-space vector). When λ is higher, the mapping becomes many-to-many. Thus, by controlling λ we can move between many-to-1 and many-to-many mappings.

The minimization of D' is made iteratively using two steps: the first step minimizes D' with respect to the association probabilities and the second step minimizes D' with respect to the transform. Convergence is guaranteed because each step minimizes a convex function.

2.3. Association/Matching Step

This step minimizes D' with respect to the association probabilities under the constraint that $p(\vec{x}_n|\vec{y}_q)$ behave like a probability:

$$\sum_n p(\vec{x}_n|\vec{y}_q) = 1,\ q = 1,\ ...,\ Q. \tag{5}$$

Since D' is convex on $p(\vec{x}_n|\vec{y}_q)$, the solution can be obtained by equating $\frac{\partial D'}{\partial p(\vec{x}_n|\vec{y}_q)} = 0$, which yields the Gibbs distribution [23]:

$$p(\vec{x}_n|\vec{y}_q) = \frac{\exp\{-\frac{1}{\lambda}d(\vec{y}_q,\vec{x}_n)\}}{\sum_i \exp\{-\frac{1}{\lambda}d(\vec{y}_q,\vec{x}_i)\}}. \tag{6}$$

The solution is valid as a probability because it is non-negative. When the annealing temperature $\lambda \to 0$, the above probabilities tend to be either 0 or 1, effectively corresponding to a minimum distance selection. In that case, this step can be replaced by a *nearest neightbor search* for the nearest X-space vector in terms of the distance function $d(\vec{y}_q,\vec{x}_n)$:

$$I(q) = \operatorname*{argmin}_n\{d(\vec{y}_q,\vec{x}_n)\} \tag{7}$$

$$p(\vec{x}_n|\vec{y}_q) = \begin{cases} 1, & n = I(q) \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

To discriminate between the two cases, this step is referred to as an *association step* when equation (6) is used and as a *matching step* when equation (8) is used.

2.4. Minimization Step

At this stage we can define the transform function $F(\cdot)$ and solve for its optimal parameters given the associations. This section proves the minimization step for a broad family of transformation functions, including *context* as in [1]. Let $F(\vec{x}_n)$ be a mixture-of-linear-regressions function

$$F(\vec{x}_n) = \sum_{k=1}^{K} p(k|\vec{x}_n)[\vec{\mu}_k + \Sigma_k\vec{x}_n], \tag{9}$$

where $\vec{\mu}_k \in \Re^D$ is the bias vector, $\Sigma_k \in \Re^{D\times P}$ is a linear transformation matrix of the k-th class and $p(k|\vec{x}_n)$ is the probability that $\vec{x}_n$ belongs to the k-th class. As class probabilities $p(k|\vec{x}_n)$ we may use a Gaussian Mixture Model (GMM) estimated from source-speaker spectra, as in [4]. The GMM allows us to avoid audible abrupt transitions when switching between different regression functions. Alternatively, we may use a vector quantizer, i.e. [23].

Having the parameters Σ_k in a matrix form is not notationally convenient, so we will reformulate the matrix-vector multiplication using the vector operator $vec\{\cdot\}$ and the Kronecker product:

$$\Sigma_k\vec{x}_n = vec\{\Sigma_k\vec{x}_n\} = (\vec{x}_n^T \otimes I_D)vec\{\Sigma_k\} = (\vec{x}_n^T \otimes I_D)\vec{\sigma}_k, \tag{10}$$

where $\vec{\sigma}_k \equiv vec\{\Sigma_k\} \in \Re^{DP}$ is the vectorized transformation matrix and $I_D \in \Re^{D\times D}$ the identity matrix. Note that the operator $vec\{\cdot\}$ is simply rearranging the parameters by stacking together the columns of the matrix.

For voice conversion it is beneficial to pack three source-speaker (X-space) vectors together to provide a better understanding of the spoken sound [1], [24] or to stack together the parameters and their delta and delta-delta values. In any case, the same transform has to be applied to all three vectors with a block-diagonal matrix Σ_k.

Any X-space structure can be incorporated in the above equation with a repetition matrix R:

$$\Sigma_k\vec{x}_n = (\vec{x}_n^T \otimes I_D)R\vec{\sigma}'_k = X_n\vec{\sigma}'_k, \tag{11}$$

where $R \in \Re^{DP\times L}$ contains only zeros and ones at the appropriate locations, $\vec{\sigma}'_k \in \Re^L$ contains only the free parameters of the structured matrix and $X_n \equiv (\vec{x}_n \otimes I_D)R \in \Re^{D\times L}$ is the *X-space data matrix* that contains the recasted information of $\vec{x}_n$. The latter matrix can be very sparse.

We constrain the linear transform matrix Σ_k to be a block-transform matrix as follows:

$$\Sigma_k = \begin{bmatrix} \Sigma'_k & 0 & 0 \\ 0 & \Sigma'_k & 0 \\ 0 & 0 & \Sigma'_k \end{bmatrix},$$

where, $\vec{\sigma}'_k \equiv vec\{\Sigma'_k\}$ and $L = \frac{DP}{9}$. Deriving R is a simple exercise that is omitted due to space restrictions.

Now, we may express the mapping function $F(\vec{x}_n)$ as a simple linear regression:

$$F(\vec{x}_n) = \Delta_n\vec{\mu} + B_n\vec{\sigma} = \begin{bmatrix} \Delta_n & B_n \end{bmatrix} \begin{bmatrix} \vec{\mu} \\ \vec{\sigma} \end{bmatrix} = \Gamma_n\vec{\gamma}, \tag{12}$$

where

$$\Delta_n = \begin{bmatrix} p(k=1|\vec{x}_n)I_D & ... & p(k=K|\vec{x}_n)I_D \end{bmatrix} \in \Re^{D\times KD}, \tag{13}$$

$$\vec{\mu} = \begin{bmatrix} \vec{\mu}_1^T & \vec{\mu}_2^T & ... & \vec{\mu}_K^T \end{bmatrix}^T \in \Re^{KD}, \tag{14}$$

$$B_n = \begin{bmatrix} p(k=1|\vec{x}_n)X_n & ... & p(k=K|\vec{x}_n)X_n \end{bmatrix} \in \Re^{D\times KL}, \tag{15}$$

$$\vec{\sigma} = \begin{bmatrix} \vec{\sigma}_1'^T & \vec{\sigma}_2'^T & ... & \vec{\sigma}_K'^T \end{bmatrix}^T \in \Re^{KL}. \tag{16}$$

Since D' is convex on the parameters, the optimal $\vec{\gamma}$ can be obtained by equating the corresponding partial derivative to zero

$$\frac{\partial D'}{\partial \vec{\gamma}} = 0, \tag{17}$$

which yields the following unique solution:

$$\vec{\gamma} = -\left(\sum_q p(\vec{y}_q)\sum_n p(\vec{x}_n|\vec{y}_q)\Gamma_n^T W_q \Gamma_n\right)^{-1} \left(\sum_q p(\vec{y}_q)\sum_n p(\vec{x}_n|\vec{y}_q)\Gamma_n^T W_q \vec{y}_q\right). \tag{18}$$

2.5. The Matching-Minimization algorithm

The Matching-Minimization (MM) algorithm is derived as the limit case of equations (6) and (18) when the annealing temperature reaches zero. In that case, the association step (6) becomes a matching step (8) and the minimization (18) considers only matched pairs of vectors. The algorithm is iterative and alternatively minimizes the matching and the conversion function, hence it's name. As shown in the previous sections, both steps are optimal and in closed form. As expected, MM needs to start from an appropriate initialization point. For the conversion of spectral envelopes [4], [20], [24], it is sufficient to search for a linear frequency warping transform [20]. Summarizing, the *Matching-Minimization* algorithm is:

1. Initialization
2. Matching Step
3. Minimization Step
4. Repeat from step 2 until convergence.

Theoretically, one could use the deterministic annealing approach to avoid getting stuck in weak local minima but in practice it is very hard to find the optimal annealing schedule. Given the formulation in this paper, it is straightforward to derive the *Association-Minimization* (AM) algorithm which is the deterministic annealing counterpart of MM, but this algorithm is omitted due to space limitation. It worth reporting that the AM algorithm does converge to a degenerate solution once D_{REV} is minimized instead of D.

3. RELATION TO ASYMMETRIC-1 INCA AND A PROOF

There is a direct relation between MM and the *asymmetric-1 INCA variant*. In fact, the latter minimizes a composite distortion that consists of the forward and the reverse average distortions (2), (3):

$$D_{\text{INCA}} = D + D_{\text{REV}} = g_y \sum_{n,q} p(\vec{x}_n|\vec{y}_q) d(\vec{y}_q, \vec{x}_n) + g_x \sum_{n,q} p'(\vec{y}_q|\vec{x}_n) d(\vec{y}_q, \vec{x}_n), \tag{19}$$

if we constrain the forward/backward association probabilities to be hard (either 0 or 1). Hard association probabilities can be replaced with appropriate index mappings $I(\cdot)$ and $I'(\cdot)$ so that:

$$D_{\text{INCA}} = g_y \sum_{q} d(\vec{y}_q, \vec{x}_{I(q)}) + g_x \sum_{n} d(\vec{y}_{I'(n)}, \vec{x}_n), \tag{20}$$

where g_x, g_y are the constant probabilities of a X- and Y-space vectors respectively. Asymmetric-1 INCA requires $g_x = g_y$.

The association probabilities $p(\vec{x}_n|\vec{y}_q)$, $p'(\vec{y}_q|\vec{x}_n)$ are independent and correspond to two different non-parametric mappings, a forward mapping from X-space to Y-space via $p'(\vec{y}_q|\vec{x}_n)$ and a backward mapping via $p(\vec{x}_n|\vec{y}_q)$. Therefore, we can use the probabilistic formulation from Section 2 to also prove the convergence and the optimality of the individual steps of the *asymmetric-1* INCA, as follows: 1) augment D_{INCA} with association entropies: $D'_{\text{INCA}} = D_{\text{INCA}} + \lambda_1 H(Y|X) + \lambda_2 H(X|Y)$, 2) fix $F(\cdot)$ and solve for $p(\vec{x}_n|\vec{y}_q)$, $p'(\vec{y}_q|\vec{x}_n)$, 3) fix $p(\vec{x}_n|\vec{y}_q)$, $p'(\vec{y}_q|\vec{x}_n)$ and solve for $F(\cdot)$, 4) take the limits $\lambda_1 \to 0$, $\lambda_2 \to 0$ to obtain hard association probabilities (matchings). The details of the proof are trivial and omitted due to space restrictions. In relation to [1], this proof shows the convergence of the algorithm while it provides optimal, closed-form solutions for a broad range of transformation functions.

4. EXPERIMENT

It is interesting to investigate how INCA is effected by the distortion term D_{REV} that corresponds to the degenerate solution. Further, we can expect that performance degrades when there are vectors in X-space that cannot be reliably matched to a Y-space vector under the transform; i.e. having a TTS corpus from the source speaker and a few utterances from the target speaker.

We conducted an experiment using 5893 15-dimensional spectral vectors from each of two female speakers A & B respectively. 70% of this dataset was used for training and the rest for testing. The vectors correspond to HMM state means. INCA is used to estimate the two component distortions: D, D_{REV}, MM is used to minimize distortion D and a special version of MM that uses the forward mapping and thus minimizes the reverse distortion D_{REV} was used for reference. The experiment is conducted using randomly selected subsets with 100%, 50%, 20%, 10% and 5% of the original 5893 vectors for Y-space. Both algorithms use a simple conversion function $F(\vec{x}) = \vec{\mu} + \Sigma\vec{x}$, where Σ is a matrix, and 15 iterations. All algorithms were initialized using the identity matrix and zero bias.

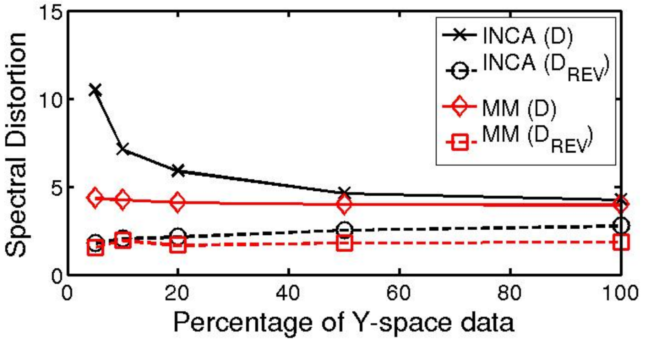

Fig. 2. Matching distortions for several Y-space sizes.

We use spectral distortion as the evaluation criterion and we present the average distortion for each distortion term independently. The results are shown in Figure 2. We observe that: 1) MM has lower distortion D than INCA for all Y-space percentages, 2) MM has significantly lower distortion D for the backward mapping, 3) MM behaves consistently in all percentages and, 4) The forward mappings (D_{REV}) have substantially lower distortion than the backward mappings (D). The first two observations are easy to explain considering that INCA minimizes an additional term than MM. The third observation states that MM is consistent and reliable. The forth observation is harder to explain but we suspect that it is due to the fact that D_{REV} has at least Q degenerate solutions with zero distortion that lower the distortion functional. The latter may also render the obtained solution to be undesirable, but a detailed investigation of this phenomenon is beyond the scope of this paper. Never-theless, the fact that D and D_{REV} have significantly different mapping distortions raises questions.

5. CONCLUSION

A probabilistic deterministic annealing framework was used to formally derive the Matching-Minimization algorithm and the asymmetric-1 variant of the INCA algorithm. It is shown that the MM algorithm is closely related to the INCA variant by augmenting the matching distortion of the former with an error term that corresponds to degenerate solutions. Both algorithms converge with each step of the iteration being optimal. MM outperforms INCA in all settings, and significantly so for the backward mapping when the adaptation data are less than 50% of the source-speaker data.

In [25] we demonstrate how to use MM to algorithmically generate new TTS voices with similar or even higher quality than the original voice.

6. REFERENCES

[1] Hadas Benisty, David Malah, and Koby Crammer, “Non-parallel voice conversion using joint optimization of alignment by temporal context and spectral distortion.,” in *ICASSP*, 2014.

[2] Yannis Agiomyrgiannakis, “Vocaine the vocoder and applications in speech synthesis,” in *ICASSP*, 2015.

[3] H Valbret, Eric Moulines, and Jean-Pierre Tubach, “Voice transformation using PSOLA technique,” *Speech Communication*, vol. 11, no. 2, pp. 175–187, 1992.

[4] Yannis Stylianou and Eric Moulines, “Continuous probabilistic transform for voice conversion,” *IEEE Transactions on Speech and Audio Processing*, vol. 6, pp. 131–142, 1998.

[5] Alexander Kain and Michael W Macon, “Spectral voice conversion for text-to-speech synthesis,” in *ICASSP*. IEEE, 1998, vol. 1, pp. 285–288.

[6] Hui Ye and Steve Young, “Perceptually weighted linear transformations for voice conversion,” in *Proc. of the Eurospeech’03*, 2003.

[7] David Sündermann, A. Bonafonte, Hermann Ney, and Harald Höge, “A first step towards text-independent voice conversion,” in *Proc. of the ICSLP’04*, 2004.

[8] Arun Kumar and Ashish Verma, “Using phone and diphone based acoustic models for voice conversion: a step towards creating voice fonts,” in *Multimedia and Expo, 2003. ICME’03. Proceedings. 2003 International Conference on*. IEEE, 2003, vol. 1, pp. I–393.

[9] Vassilios V Digalakis, Dimitry Rtischev, and Leonardo G Neumeyer, “Speaker adaptation using constrained estimation of gaussian mixtures,” *Speech and Audio Processing, IEEE Transactions on*, vol. 3, no. 5, pp. 357–366, 1995.

[10] Vassilis D Diakoloukas and Vassilios V Digalakis, “Maximum-likelihood stochastic-transformation adaptation of hidden markov models,” *Speech and Audio Processing, IEEE Transactions on*, vol. 7, no. 2, pp. 177–187, 1999.

[11] Athanasios Mouchtaris, Jan Van der Spiegel, and Paul Mueller, “Non-parallel training for voice conversion by maximum likelihood constrained adaptation,” in *ICASSP*. IEEE, 2004, vol. 1, pp. I–1.

[12] Masatsune Tamura, Takashi Masuko, Keiichi Tokuda, and Takao Kobayashi, “Adaptation of pitch and spectrum for HMM-based speech synthesis using MLLR,” in *ICASSP*. IEEE, 2001, vol. 2, pp. 805–808.

[13] Keiichi Tokuda, Heiga Zen, and Alan W Black, “An HMM-based speech synthesis system applied to english,” in *Speech Synthesis, 2002. Proceedings of 2002 IEEE Workshop on*. IEEE, 2002, pp. 227–230.

[14] Takayoshi Yoshimura, Keiichi Tokuda, Takashi Masuko, Takao Kobayashi, and Tadashi Kitamura, “Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis,” in *Proc. Eurospeech*, 1999, pp. 2347–2350.

[15] Orhan Karaali, Gerald Corrigan, and Ira A. Gerson, “Speech synthesis with neural networks,” *CoRR*, vol. cs.NE/9811031, 1998.

[16] Heiga Zen, Andrew Senior, and Mike Schuster, “Statistical parametric speech synthesis using deep neural networks,” in *ICASSP*. IEEE, 2013, pp. 7962–7966.

[17] Zhizheng Wu, Pawel Swietojanski, Christophe Veaux, Stephen Renals, and Simon King, *A study of speaker adaptation for DNN-based speech synthesis*, International Speech Communication Association, 2015, Date of Acceptance: 01/06/2015.

[18] Toru Nakashika, Ryoichi Takashima, Tetsuya Takiguchi, and Yasuo Ariki, “Voice conversion in high-order eigen space using deep belief nets.,” in *Interspeech*. 2013, pp. 369–372, ISCA.

[19] Yuchen Fan, Yao Qian, F.K. Soong, and Lei He, “Multi-speaker modeling and speaker adaptation for DNN-based TTS synthesis,” in *ICASSP*, April 2015, pp. 4475–4479.

[20] Daniel Erro, Asunción Moreno, and A. Bonafonte, “INCA algorithm for training voice conversion systems from nonparallel corpora,” *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 18, no. 5, pp. 944–953, 2010.

[21] Daniel Erro and Asunción Moreno, “Frame alignment method for cross-lingual voice conversion,” in *Interspeech*, 2007.

[22] Asela Gunawardana and William Byrne, “Convergence theorems for generalized alternating minimization procedures,” *Journal of Machine Learning Research*, December 2005.

[23] Kenneth Rose, “Deterministic annealing for clustering, compression, classification, regression, and related optimization problems,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2210–2239, 1998.

[24] Hanna Silén, Jani Nurminen, Elina Helander, and Moncef Gabbouj, “Voice conversion for non-parallel datasets using dynamic kernel partial least squares regression,” in *Interspeech*, 2013.

[25] Yannis Agiomyrgiannakis, “Voice Morphing that improves TTS quality using an Optimal Dynamic Frequency Warping-and-Weighting transform,” in *ICASSP*, 2016.