JOINT SPEAKER DIARISATION AND TRACKING IN SWITCHING STATE-SPACE MODEL

Jeremy H. M. Wong[1,2] *Yifan Gong*[2]

[1]Institute for Infocomm Research, A*STAR, Singapore [2]Microsoft, USA

ABSTRACT

Speakers may move around while diarisation is being performed. When a microphone array is used, the instantaneous locations of where the sounds originated from can be estimated, and previous investigations have shown that such information can be complementary to speaker embeddings in the diarisation task. However, these approaches often assume that speakers are fairly stationary throughout a meeting. This paper relaxes this assumption, by proposing to explicitly track the movements of speakers while jointly performing diarisation within a unified model. A state-space model is proposed, where the hidden state expresses the identity of the current active speaker and the predicted locations of all speakers. The model is implemented as a particle filter. Experiments on a Microsoft rich meeting transcription task show that the proposed joint location tracking and diarisation approach is able to perform comparably with other methods that use location information.

Index Terms— Switching state-space, location tracking, particle filter, diarisation, meeting transcription

1. INTRODUCTION

Speaker diarisation is the task of clustering segments of audio that are uttered by the same speaker. It can be used with Automatic Speech Recognition (ASR) to provide rich transcriptions of audio, expressing both words and speaker identities. The task of diarisation can be broken down into counting the number of clusters and clustering the audio segments. By treating these sub-tasks separately, the number of clusters can first be estimated by finding the maximum gap in a chosen statistic [1, 2], then the segments can be clustered using either k-means [3] or spectral clustering [4]. Alternatively, both sub-tasks can be performed in unison in the Agglomerative Hierarchical Clustering (AHC) framework [5, 6]. This iteratively performs greedy merging of clusters based on a measured affinity, until a stopping criterion is reached. A Hidden Markov Model (HMM) can capture information about the temporal nature of speech, which may be useful for diarisation. The HMM can either be used within AHC in the computation of the affinities [7], or on its own after being given an upper bound of the number of clusters [8, 9].

End-to-end neural network diarisation [10] has also been shown to perform competitively against the previously described unsupervised diarisation approaches when evaluated with few speakers. Location features can be used as inputs to the network [11]. However, the constraints of computational memory often limit the number of speakers that can be seen per-meeting during training. As such, these models may not generalise well to more speakers. This can be alleviated by only using the neural network to diarise short segments of audio, where it is reasonable to assume that only a few speakers are active [12]. Unsupervised diarisation, which may generalise better to having more speakers, is then used to combine these short segment hypotheses across the whole meeting. Thus, further investigations to improve unsupervised methods for diarisation are still relevant.

Diarisation is often performed using only speaker embeddings, extracted using models trained to discriminate between speakers through a speaker identification or verification task. Speaker location information may be complementary to the speaker embeddings, and is available when using a microphone array. In the HMM framework, location information can be incorporated by using the speaker embeddings together with either time-delay-of-arrival [13, 14] or Sound Source Localisation (SSL) [15] features as the observations. In these works, the HMM state only encodes information about the identities of the speakers, and does not keep track of where each speaker is at each point in time. This therefore may not explicitly model the movements of speakers, and may assume that speakers are fairly stationary throughout a meeting.

In the vision domain, multi-face tracking can be achieved using separate Kalman filters to track the movements of each face [16, 17]. When using a microphone array, localisation information from the audio has been shown to be complementary to visual information for face tracking [18]. In the audio-only scenario, challenges such as LOCATA [19] help to spur the development of audio localisation and tracking methods. Several of these methods also rely on Kalman or particle filtering techniques, to track the locations of a single [20, 21, 22] or multiple [23] audio sources. When tracking multiple audio sources, multi-target extensions of probabilistic data association provide a framework to estimate which observations belong to each of the targets being tracked [24]. However, when used with multiple speakers [25, 26, 27], these tracking methods often only rely on location information, and not speaker embeddings.

This paper proposes to track speaker movements jointly with performing diarisation, while also using speaker embeddings. It is hoped that explicitly modelling the movements of speakers may benefit the diarisation task. A switching state-space model [28] is proposed, which does joint modelling through a hidden state that encodes both information about the active speaker identity and also the current locations of each of the speakers. This model is implemented using a particle filter framework to accommodate for the forms of transition and emission likelihoods that are used. The model is referred to as the Switching State-space Particle Filter (SSPF).

2. JOINT CLUSTERING AND LOCATION TRACKING

HMM diarisation often encodes the active speaker as the hidden state. In [15], the observation sequence likelihood is computed as

$$p\left(\mathbf{D}_{1:T}, \mathbf{S}_{1:T}\right) \approx \sum_{\mathbf{q}_{1:T}} \prod_{t=1}^{T} p\left(\mathbf{d}_t|q_t\right) p\left(\mathbf{s}_t|q_t\right) p\left(q_t|q_{t-1}\right), \quad (1)$$

The work is done at Microsoft. Jeremy Wong is now employed at the Institute for Infocomm Research, A*STAR.

where $\mathbf{d}_t$ and $\mathbf{s}_t$ are the speaker embedding and SSL features respectively at frame t, T is the number of frames, and q_t is the discrete hidden state that encodes the active speaker identity. The initial state probability is omitted here for brevity. In this formulation, speaker location information is considered, but it is not possible to infer where each speaker is at each point in time. Thus, the model does not explicitly capture the movements of speakers.

Certain movement trajectories may be unlikely. For example, it may not be expected that a speaker should oscillate π radians around the microphone array at alternating frames. Explicitly modelling speaker movements may aid diarisation, by penalising hypotheses with unlikely movement trajectories. A probabilistic approach accommodates for noise and reverberation in the measured location features. To track speaker movements, this paper proposes to encode the current active speaker identity as well as the current locations of all speakers in the hidden state. Speech separation is first applied to the microphone array audio. This outputs N channels, without concurrent speakers in each. The number of separated channels, N, is a design choice that limits the maximum number of concurrent speakers allowed. The SSPF simultaneously models all N channels, thus explicitly accommodating for overlapping speech while preserving temporal information. In contrast, [15] merges the channels into a single stream. The SSPF hidden state is defined as $\mathbf{z}_t = \{\mathbf{q}_{t,1:N}, \boldsymbol{\theta}_{t,1:M}\}$, where the discrete $q_{t,n}$ represents the active speaker at frame t in channel n, $\theta_{t,m}$ represents the angular location in radians around the microphone array at frame t for speaker m, and M is the number of speakers. Using the same Markov assumptions as (1), the observation sequence likelihood is computed as

$$p\left(\mathbf{D}_{1:T,1:N}, \mathbf{X}_{1:T,1:N}\right) \approx \sum_{\mathbf{Z}_{1:T}} \prod_{t=1}^{T} p\left(\mathbf{D}_{t,1:N}|\mathbf{z}_t\right) p\left(\mathbf{X}_{t,1:N}|\mathbf{z}_t\right) \times p\left(\mathbf{z}_t|\mathbf{z}_{t-1}\right), \quad (2)$$

where $\mathbf{X}_{t,1:N}$ is used as a placeholder to represent a location-based observation feature that can take several possible forms. Here, diarisation is performed after speech separation, and thus each frame has N unmixed observations, $\mathbf{D}_{t,1:N}$ and $\mathbf{X}_{t,1:N}$.

The transition probability is factorised for each state entity,

$$p\left(\mathbf{z}_t|\mathbf{z}_{t-1}\right) \approx \left[\prod_{n=1}^{N} P\left(q_{t,n}|q_{t-1,n}\right)\right]\left[\prod_{m=1}^{M} p\left(\theta_{t,m}|\theta_{t-1,m}\right)\right]. \quad (3)$$

This assumes that each separate $q_{t,n}$ and $\theta_{t,m}$ propagate independently over time. The speaker transition probability, $P\left(q_{t,n}|q_{t-1,n}\right)$, is an $M \times M$ matrix that is shared across all channels. The angular location transition likelihood is chosen to be a von Mises density function that is shared across all speakers,

$$p\left(\theta_{t,m}|\theta_{t-1,m}\right) = \frac{1}{2\pi I_0\left(\varsigma\right)} e^{\varsigma \cos\left(\theta_{t,m}-\theta_{t-1,m}\right)}, \quad (4)$$

where the concentration, ς, expresses how fast speakers tend to move, and $I_\nu\left(\varsigma\right)$ is the modified Bessel function of the first kind with order ν. The von Mises density function is chosen to abide by $\theta_{t,m}$ being bounded by $(-\pi, \pi]$ with a periodic boundary condition.

The initial state likelihood, which is omitted in (2) for brevity, is similarly factorised into each of the separate state entities,

$$p\left(\mathbf{z}_1\right) \approx \left[\prod_{n=1}^{N} P\left(q_{1,n}\right)\right]\left[\prod_{m=1}^{M} p\left(\theta_{1,m}\right)\right]. \quad (5)$$

Both the active speaker initial state probability, $P\left(q_{1,n}\right)$, and the initial location likelihood, $p\left(\theta_{1,m}\right)$, are set to be uniform, because the model has no information about the identity of the active speaker or the locations of the speakers, before any observation is made.

Similarly, the speaker embedding emission likelihood is also factorised into separate channels,

$$p\left(\mathbf{D}_{t,1:N}|\mathbf{z}_t\right) \approx \prod_{n=1}^{N} p\left(\mathbf{d}_{t,n}|\mathbf{z}_t\right), \quad (6)$$

which assumes that the emissions of the channels are independent of each other when given the state. The emission likelihood for each channel is chosen to be a von Mises-Fisher density function [15],

$$p\left(\mathbf{d}_{t,n}|\mathbf{z}_t\right) = \frac{\gamma^{\frac{\mathbb{D}}{2}-1}}{\left(2\pi\right)^{\frac{\mathbb{D}}{2}} I_{\frac{\mathbb{D}}{2}-1}\left(\gamma\right)} e^{\gamma \boldsymbol{\mu}_{q_{t,n}} \cdot \mathbf{d}_{t,n}}, \quad (7)$$

where $\mathbb{D}$ is the speaker embedding dimension, $\boldsymbol{\mu}_{q_{t,n}}$ represents the embedding centroid for speaker $q_{t,n}$, and γ is the concentration. The log of this likelihood is a cosine similarity between $\boldsymbol{\mu}_{q_{t,n}}$ and $\mathbf{d}_{t,n}$.

The location emission likelihood is also factorised per-channel,

$$p\left(\mathbf{X}_{t,1:N}|\mathbf{z}_t\right) \approx \prod_{n=1}^{N} p\left(\mathbf{x}_{t,n}|\mathbf{z}_t\right), \quad (8)$$

which again assumes that the observed locations in each channel are independent of each other when given the current state. Two forms of location features are considered. The first is the SSL vector, $\mathbf{s}_{t,n}$, representing a categorical distribution, where each dimension expresses the probability that the sound had originated from each angular bin around the microphone array,

$$s_{t,n,i} = P\left(\psi = i|\mathbf{x}_{t,n}\right), \quad (9)$$

where i is the angular bin index and ψ is the angular bin from which the frame $\mathbf{x}_{t,n}$ may have originated. This is computed using a complex angular central Gaussian model [29], using unmixing masks generated from an earlier speech separation step, as is described in [30]. The second form of location feature is the Direction-Of-Arrival (DOA), $\phi_{t,n}$, which is computed as the mode of the SSL,

$$\phi_{t,n} = b_j \quad , \text{ where } \quad j = \arg\max_i s_{t,n,i}, \quad (10)$$

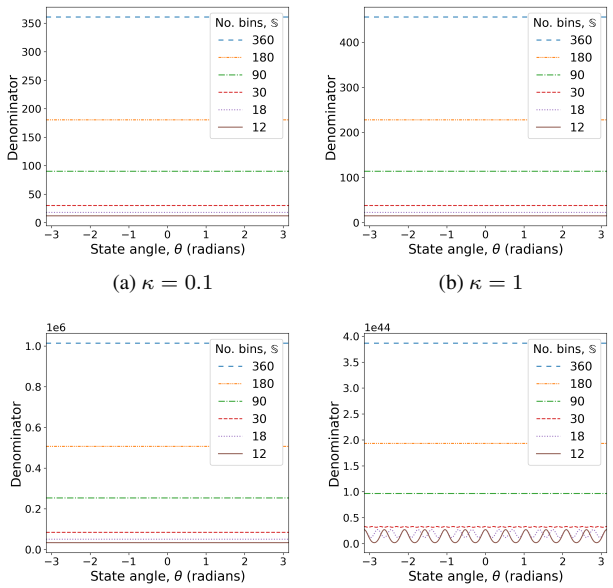
and b_j is the angle in radians of the jth bin. An alternative is to compute the DOA as the circular mean of the SSL, similarly to (16), instead of the mode, but initial tests did not suggest any significant performance difference between the two choices.

When using the DOA as the observed location feature, $\mathbf{x}_{t,n}$ is substituted with $\phi_{t,n}$ in (8), and the location emission likelihood for each channel can be computed as a von Mises density function,

$$p\left(\phi_{t,n}|\mathbf{z}_t\right) = \frac{1}{2\pi I_0\left(\kappa\right)} e^{\kappa \cos\left(\phi_{t,n}-\theta_{t,q_{t,n}}\right)}, \quad (11)$$

where the concentration, κ, expresses the observation noise. This measures a similarity between the observed location, $\phi_{t,n}$, and the predicted location of the speaker that is estimated to be active on the channel, $\theta_{t,q_{t,n}}$. Alternatively, the full SSL vector can be used as the observed location feature, by substituting $\mathbf{x}_{t,n}$ with $\mathbf{s}_{t,n}$ in (8). For this feature, the location emission likelihood for each channel is computed using a continuous categorical density function [31],

$$p\left(\mathbf{s}_{t,n}|\mathbf{z}_t\right) = \frac{1}{C\left(\boldsymbol{\lambda}_{t,n}\right)} \prod_{i=1}^{\mathbb{S}} \lambda_{t,n,i}^{s_{t,n,i}}, \quad (12)$$



(a) $\kappa = 0.1$ (b) $\kappa = 1$

(c) $\kappa = 10$ (d) $\kappa = 100$

Fig. 1: Denominator term of discretised von Mises distribution (13)

where $\mathbb{S}$ is the number of discrete angular bins and $C\left(\boldsymbol{\lambda}_{t,n}\right)$ is the normalisation term defined in [31]. The continuous categorical bin probabilities are computed as a discretised von Mises distribution about a mean that represents the predicted location, $\theta_{t,q_{t,n}}$, of the current active speaker in the channel,

$$\lambda_{t,n,i} = \frac{e^{\kappa \cos\left(b_i - \theta_{t,q_{t,n}}\right)}}{\sum_{j=1}^{\mathbb{S}} e^{\kappa \cos\left(b_j - \theta_{t,q_{t,n}}\right)}}. \tag{13}$$

The equivalent log-likelihood of (12) is a KL divergence between the predicted SSL, $\boldsymbol{\lambda}_{t,n}$, and the measured SSL, $\mathbf{s}_{t,n}$, both representing discrete categorical distributions. Substituting (13) into (12) yields

$$p\left(\mathbf{s}_{t,n}|\mathbf{z}_t\right) = \frac{e^{\rho_{t,n} \cos\left(\eta_{t,n} - \theta_{t,q_{t,n}}\right)}}{C\left(\boldsymbol{\lambda}_{t,n}\right) \sum_{j=1}^{\mathbb{S}} e^{\kappa \cos\left(b_j - \theta_{t,q_{t,n}}\right)}}, \tag{14}$$

where

$$\rho_{t,n} = \kappa \sqrt{\sum_{i=1}^{\mathbb{S}} \sum_{j=1}^{\mathbb{S}} s_{t,n,i} s_{t,n,j} \cos\left(b_i - b_j\right)}, \tag{15}$$

$$\eta_{t,n} = \tan^{-1}\left(\frac{\sum_{i=1}^{\mathbb{S}} s_{t,n,i} \sin b_i}{\sum_{i=1}^{\mathbb{S}} s_{t,n,i} \cos b_i}\right). \tag{16}$$

This suggests that with the choice of location emission likelihood of (12) and (13), the SSL, $\mathbf{s}_{t,n}$, at each frame can be completely characterised by an equivalent concentration, $\rho_{t,n}$, and mean, $\eta_{t,n}$. The concentration may weigh the contribution of each frame to the total log-likelihood proportionally to the sharpness of the SSL.

However, the normalisation term, $C\left(\boldsymbol{\lambda}_{t,n}\right)$, is difficult to compute in a numerically stable manner [31]. As such, it is ignored in this paper. Furthermore, the $\sum_j e^{\kappa \cos\left(b_j - \theta_{t,q_{t,n}}\right)}$ term in the denominator of (14) is also ignored. Figure 1 plots $\sum_j e^{\kappa \cos\left(b_j - \theta\right)}$ as a function of θ, for various values of κ and $\mathbb{S}$. The plots suggest that $\sum_j e^{\kappa \cos\left(b_j - \theta\right)}$ is approximately independent of θ, except when both the concentration, κ, is large and the number of angular bins, $\mathbb{S}$, is small. The setup in this paper does not operate in this regime. Thus it seems reasonable to omit this term. Therefore, the location emission likelihood is computed as

$$p\left(\mathbf{s}_{t,n}|\mathbf{z}_t\right) \propto e^{\rho_{t,n} \cos\left(\eta_{t,n} - \theta_{t,q_{t,n}}\right)}, \tag{17}$$

which looks similar in form to a von Mises density function.

With N separated channels, there may not be N concurrent active speakers at every frame. If frame t in channel n does not have an observation, then the emission likelihoods are set to $p\left(\mathbf{d}_{t,n}|\mathbf{z}_t\right) = 1$ and $p\left(\mathbf{x}_{t,n}|\mathbf{z}_t\right) = 1$ for this frame and channel.

The joint speaker turn and location tracking model is illustrated graphically in Figure 2. This is reminiscent of the switching state-space model proposed in [28]. The N discrete chains that express the current active speakers switch the outputs between the M continuous chains that track the locations of each of the speakers, to generate the observed locations. The parameters of the model are the speaker embedding centroids, $\boldsymbol{\mu}_{1:M}$, the speaker transition probabilities, $P\left(q_{t,n}|q_{t-1,n}\right)$, and the concentrations, γ, ς, and κ.

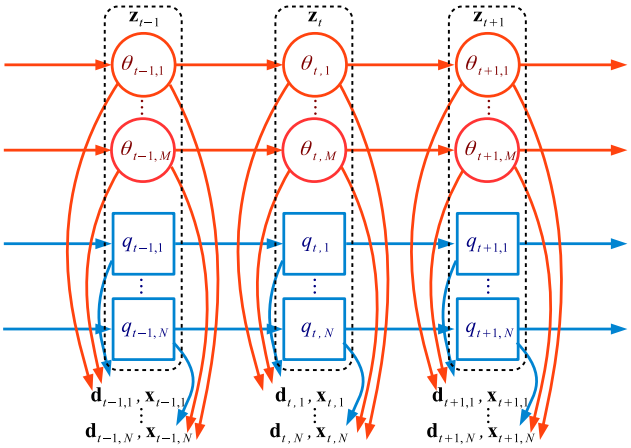


Fig. 2: Joint modelling of discrete speaker turns (squares) and continuous locations (circles) using a switching state-space model

3. PARTICLE FILTER IMPLEMENTATION

Clustering by decoding the model requires the forward recursion of

$$p\left(\mathbf{z}_t|\mathbf{O}_{1:t,1:N}\right) = \int p\left(\mathbf{z}_{t-1}|\mathbf{O}_{1:t-1,1:N}\right) p\left(\mathbf{D}_{t,1:N}|\mathbf{z}_t\right) \times p\left(\mathbf{X}_{t,1:N}|\mathbf{z}_t\right) p\left(\mathbf{z}_t|\mathbf{z}_{t-1}\right) d\mathbf{z}_{t-1}, \tag{18}$$

where $\mathbf{O}_{1:t,1:N}$ is used to concisely represent the pair of observations, $\{\mathbf{D}_{1:t,1:N}, \mathbf{X}_{1:t,1:N}\}$. The choice of emission and transition likelihoods in Section 2 are not closed under the multiplication and convolution operations in (18). This makes it difficult to implement the model exactly, in a manner analogous to a Kalman filter or HMM. In this paper, the model is implemented as a particle filter [32]. This does not require the likelihoods to be closed under the forward pass operations, and instead performs a Monte Carlo simulation of the propagation of density functions along the forward pass.

The sequential importance resampling algorithm [33] is used. At each frame in the forward pass, the prediction step samples particles from either the initial state likelihood, $P(\mathbf{z}_1)$, for the first frame or from the transition likelihood, $P(\mathbf{z}_t|\mathbf{z}_{t-1})$, at subsequent frames, when given the particles or resampled particles from the previous frame. The factorised forms of (3) and (5) allow each state entity to be sampled separately. The collection of particles represents an approximation of the prediction likelihood,

$$p(\mathbf{z}_t|\mathbf{O}_{1:t-1,1:N}) \approx \sum_{r=1}^{\mathbb{R}} \breve{\omega}_{t-1}^{(r)} \delta\left(\mathbf{z}_t, \widehat{\mathbf{z}}_t^{(r)}\right), \tag{19}$$

where $\widehat{\mathbf{z}}_t^{(r)}$ is the rth particle, $\mathbb{R}$ is the number of particles, $\breve{\omega}_{t-1}^{(r)}$ are the importance weights after resampling from the previous frame, and the Dirac delta function is defined as

$$\delta(\mathbf{y}_1, \mathbf{y}_2) = \begin{cases} \infty & , \text{ if } \mathbf{y}_1 = \mathbf{y}_2 \\ 0 & , \text{ otherwise} \end{cases}. \tag{20}$$

After sampling the particles, the update step then computes the importance sampling weights as

$$\omega_t^{(r)} = \frac{\breve{\omega}_{t-1}^{(r)} p\left(\mathbf{D}_{t,1:N}\middle|\widehat{\mathbf{z}}_t^{(r)}\right) p\left(\mathbf{X}_{t,1:N}\middle|\widehat{\mathbf{z}}_t^{(r)}\right)}{\sum_{r'=1}^{\mathbb{R}} \breve{\omega}_{t-1}^{(r')} p\left(\mathbf{D}_{t,1:N}\middle|\widehat{\mathbf{z}}_t^{(r')}\right) p\left(\mathbf{X}_{t,1:N}\middle|\widehat{\mathbf{z}}_t^{(r')}\right)}. \tag{21}$$

These together with the particles approximate the update likelihood,

$$p(\mathbf{z}_t|\mathbf{O}_{1:t,1:N}) \approx \sum_{r=1}^{\mathbb{R}} \omega_t^{(r)} \delta\left(\mathbf{z}_t, \widehat{\mathbf{z}}_t^{(r)}\right). \tag{22}$$

Sequential Monte Carlo simulation methods may suffer from the importance weights attenuating to zero for many particles as the forward pass progresses, because these are computed recursively as a product of previous importance weights in (21). This limits the exploration of the state-space support. Resampling [33] aims to alleviate this at the expense of an increase in the variance of the estimates. A new collection of resampled particles are sampled with replacement from the original particles, $\widehat{\mathbf{z}}_t^{(r)}$, with each original particle being resampled with a probability equal to its importance weight, $\omega_t^{(r)}$. The systematic method [34] is used here to perform resampling. After resampling, the new resampled importance weights are set uniformly, $\breve{\omega}_t^{(r)} = \frac{1}{\mathbb{R}}$. Resampling is only performed at a frame if the effective sample size [35], $[\sum_r \omega_t^{(r)2}]^{-1}$, falls below a threshold.

4. DECODING

Clustering can be performed by decoding the model. Only the active speakers, $\mathbf{q}_{t,1:N}$, are of interest to the diarisation task, while the speaker locations, $\boldsymbol{\theta}_{t,1:M}$, can be marginalised over. One approach to estimate the active speaker sequence is a Viterbi-style decoding,

$$\mathbf{Q}_{1:T,1:N}^* = \underset{\mathbf{Q}_{1:T,1:N}}{\arg\max} \int p(\mathbf{O}_{1:T,1:N}, \mathbf{Q}_{1:T,1:N}, \boldsymbol{\theta}_{1:T,1:M})\, d\boldsymbol{\theta}_{1:T,1:M}. \tag{23}$$

However, it may not be trivial to develop an efficient algorithm for this when the hidden state contains continuous variables. Furthermore, in the diarisation setup used in this paper, the objective is to hypothesise a speaker identity for each word, which may not be perfectly matched with finding the most likely speaker sequence.

Decoding is instead performed by first computing the per-frame speaker state posteriors, marginalising over the location states,

$$P(\mathbf{q}_{t,1:N}|\mathbf{O}_{1:T,1:N}) = \int p(\mathbf{q}_{t,1:N}, \boldsymbol{\theta}_{t,1:M}|\mathbf{O}_{1:T,1:N})\, d\boldsymbol{\theta}_{t,1:M}. \tag{24}$$

The speaker for each word is then estimated by choosing the most probable speaker from the aggregated speaker state posteriors over the frames within the word. Aggregation of the state posteriors can be done either as a sum (25), product (26), or majority voting (27),

$$\overline{q}_l^* = \underset{\overline{q}_l}{\arg\max} \sum_{t=\tau_l^{\text{start}}}^{\tau_l^{\text{end}}} P(q_{t,n_l} = \overline{q}_l|\mathbf{O}_{1:T,1:N}), \tag{25}$$

$$\overline{q}_l^* = \underset{\overline{q}_l}{\arg\max} \prod_{t=\tau_l^{\text{start}}}^{\tau_l^{\text{end}}} P(q_{t,n_l} = \overline{q}_l|\mathbf{O}_{1:T,1:N}), \tag{26}$$

$$\overline{q}_l^* = \underset{\overline{q}_l}{\arg\max} \sum_{t=\tau_l^{\text{start}}}^{\tau_l^{\text{end}}} \partial\left[\overline{q}_l, \underset{q}{\arg\max}\, P(q_{t,n_l} = q|\mathbf{O}_{1:T,1:N})\right], \tag{27}$$

where $\overline{q}_l$ is the speaker identity of the lth word, τ_l^{start} and τ_l^{end} are the start and end frame indexes of the word respectively, n_l is the channel that the word is detected on, the Kronecker delta function is

$$\partial(i,j) = \begin{cases} 1 & , \text{ if } i = j \\ 0 & , \text{ otherwise} \end{cases}, \tag{28}$$

and $P(q_{t,n_l}|\mathbf{O}_{1:T,1:N})$ is computed by marginalising over the other channels in $P(\mathbf{q}_{t,1:N}|\mathbf{O}_{1:T,1:N})$. The product combination in (26) is related to a maximum probability interpretation, as the probability for the speaker of a word should be computed as a joint probability of the same speaker over all of the frames within the word.

The state posterior in (24) can be estimated through Forward Filtering-Backward Smoothing (FFBS) [36],

$$p(\mathbf{q}_{t,1:N}, \boldsymbol{\theta}_{t,1:M}|\mathbf{O}_{1:T,1:N}) \approx \sum_{r=1}^{\mathbb{R}} \dot{\omega}_t^{(r)} \delta\left(\mathbf{z}_t, \widehat{\mathbf{z}}_t^{(r)}\right). \tag{29}$$

The backward recursion computes the backward importance weights,

$$\dot{\omega}_t^{(r)} = \omega_t^{(r)} \sum_{i=1}^{\mathbb{R}} \dot{\omega}_{t+1}^{(i)} \frac{p\left(\widehat{\mathbf{z}}_{t+1}^{(i)}\middle|\widehat{\mathbf{z}}_t^{(r)}\right)}{\sum_{j=1}^{\mathbb{R}} \omega_t^{(j)} p\left(\widehat{\mathbf{z}}_{t+1}^{(i)}\middle|\widehat{\mathbf{z}}_t^{(j)}\right)}. \tag{30}$$

An exact computation of the backward importance weights in (30) is used, which has a computational cost that scales as $\mathcal{O}\left(\mathbb{R}^2\right)$. This can be expensive when using many particles. Many particles may be required to sufficiently explore the state-space. A kernel density approximation [37, 38] can be used to speed up the computation to scale as $\mathcal{O}(\mathbb{R}\log\mathbb{R})$, but this requires that the transition likelihoods represent monotonic kernels [39], which may limit the form of the allowed speaker transition probabilities, $P(q_{t,n}|q_{t-1,n})$, to matrices with a probability attenuating monotonically away from the diagonal. As opposed to this, the forward recursion has a computational cost that scales as $\mathcal{O}(\mathbb{R})$. Therefore, the computational cost can be reduced by decoding using only the forward pass, by replacing $\mathbf{O}_{1:T,1:N}$ with $\mathbf{O}_{1:t,1:N}$ in the conditional dependencies in (24), (25), (26), and (27). However, this foregoes information from the future context when making the decoding decisions. An alternative method to reduce the computational cost is to uniformly sub-sample

the particles after the forward pass, when performing the backward pass. The exploration of the state-space in the FFBS algorithm is primarily achieved during the sampling of particles in the prediction step of the forward pass. Therefore, having a large number of particles is more important for the forward pass than the backward pass.

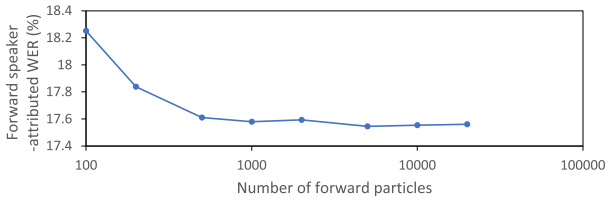
Decoding for diarisation is done per word. Thus, it seems reasonable to restrict the state transitions to only allow speaker changes at the word boundaries. In the forward pass, this can be achieved by setting $P\left(q_{t,n}|q_{t-1,n}\right)$ to the identity matrix when sampling in the prediction step at frames that are not at word boundaries. In the backward pass, the same restricted speaker transition probabilities can be used to compute the backward importance weights in (30).

5. MEETING TRANSCRIPTION SETUP

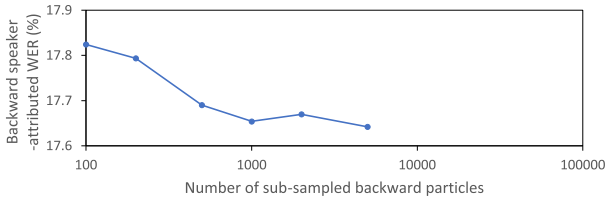
The proposed approach was evaluated on a rich meeting transcription task, with the setup that was initially described in [30], and used again in [15]. Speech separation was used to separate microphone array audio into multiple channels. Voice activity detection and hybrid ASR were run on each channel. Speaker change detection was used to find segments with speaker purity, by applying a threshold to the cosine similarity of the speaker embeddings computed using the model described in [40]. A two-pass approach was used for diarisation. In the first pass, an initial diarisation hypothesis was computed using AHC. This clustered together all of the segments from all of the channels that belonged to the same speaker, by greedily merging clusters with the highest speaker embedding cosine similarity, until the maximum similarity fell below a threshold. The Hungarian algorithm [41] was then used to find the optimal mapping between the AHC hypothesised clusters and the enrolled speakers. This AHC diarisation hypothesis was used to initialise the parameters of either a HMM or SSPF model. For the SSPF, the concentrations were estimated using parameter sweeps on the *dev* data. This influences which movement trajectories are considered as likely. The speaker embedding centroids and speaker transition probabilities were estimated by maximising the likelihood of the AHC hypothesis. Uniform smoothing was interpolated into the speaker transition probabilities to improve generalisation. As with [15], the HMM parameters here were fine-tuned for each test meeting using Expectation-Maximisation (EM). As opposed to this, the SSPF parameters were not modified after initialisation, to limit the computational cost. In the second pass, the HMM or SSPF was decoded to refine the clusters. The maximum possible number of speaker states, M, was equal to the number of AHC clusters. As such, the second pass can only hypothesise fewer than or equal to the number of speakers from the first pass. In [15], Hungarian speaker tagging was performed after HMM clustering. However here, HMM or SSPF clustering was performed after Hungarian tagging, to isolate the experimental trends arising from the HMM and SSPF methods, and ignore the trends due to interactions between clustering and tagging. Following [15], the HMM here also used a segment of one or more words as a frame, and all channels were merged into one stream. Uniform time segmentation with a duration and shift of 0.4s was instead used for the SSPF over parallel channels, to capture speakers' temporal movements.

6. EXPERIMENTS

Experiments used audio data collected from internal English meetings, with an average of 7 and maximum of 18 active participants per meeting, lasting up to 1 hour each. The *dev* set comprised 51 meetings making up 23 hours, while the *eval* set comprised 60 meetings making up 35 hours. The model described in [40] was used



(a) Number of forward particles



(b) Number of backward particles sub-sampled from 20000 forward particles

Fig. 3: Performance on the *dev* set with various numbers of forward and sub-sampled backward particles

to extract 128-dimensional d-vector speaker embeddings. The SSL dimension was 360. The HMM used downsampled 18-dimensional SSL vectors, as these yielded gains in initial tests. The SSPF used the full 360 dimensions, to preserve the resolution for accurate location tracking. The speaker-attributed Word Error Rate (WER) [30] was used to measure the performance. This first measured the WER separately for each speaker, then averaged the WERs over all speakers. The speaker-attributed WER assesses both the speaker diarisation and ASR performances. Both are important for the rich transcription task. The diarisation error rate is another commonly used measure. This computes the error per unit time, while the speaker-attributed WER computes per word. In rich transcription, the hypothesis is often displayed per-word. Therefore, the speaker-attributed WER may be more indicative of the user's perceived performance.

The first experiment assesses the influence of the number of particles. This affects the exploration of the state-space during the forward pass. Decoding of the SSPF can be done using only a forward pass or by using both the forward and backward passes. Figure 3a assesses the impact on the *dev* set of the number of particles in the forward pass, by decoding using only the forward pass. DOA features were used with sum aggregation, without state transition restrictions. The speaker-attributed WER can be seen to degrade when fewer than 1000 particles are used. It may not be possible to effectively explore the support of the state-space with so few particles. In the remaining experiments, 20000 particles were used in the forward pass to ensure adequate exploration of the state-space. Going beyond 20000 particles required more than the available CPU memory.

Forward pass decoding ignores future context information. Such information can be utilised by decoding using FFBS. In the backward pass, the computational cost can be reduced by sub-sampling from the 20000 forward pass particles. Figure 3b assesses how the number of sub-sampled particles in the backward pass affects the performance on the *dev* set, showing improvements with more sub-sampled particles. Comparing the two passes, a forward pass with 20000 particles yields a speaker-attributed WER of 17.56%, while a backward pass with 5000 sub-sampled particles yields 17.64%. The backward pass performance may eventually surpass that of the forward pass when given sufficient sub-sampled particles, but going beyond 5000 sub-sampled particles was already infeasible in the

Table 1: Location observation feature type

Observations	*dev* speaker-attributed WER (%)
d-vector	17.65
d-vector + DOA	17.56
d-vector + SSL	17.55

current implementation. Unless otherwise stated, the remaining experiments perform decoding using only the forward pass.

The next experiment investigates the benefit of tracking the speaker locations, for the diarisation task. Setting $\kappa = 0$ makes the SSPF only use speaker embeddings. Joint location tracking and diarisation can also be performed, by using location features in the form of either the DOA with an emission likelihood of (11), or the SSL with an emission likelihood of (17). A comparison of these features on the *dev* set is shown in Table 1, suggesting that both DOA and SSL features may yield small gains over using only d-vectors. Thus, jointly performing speaker tracking and clustering may aid the diarisation task. The results agree with [15] in suggesting that location features may be complementary to speaker embeddings for diarisation. SSL features do not show any significant gain over DOA features. In the remaining experiments, the SSL features were used.

As is described in Section 4, the speaker for each word can be chosen through different methods of aggregating the per-frame state posteriors within each word. The *dev* set speaker-attributed WERs of the sum, product, and majority voting are 17.55, 17.56, and 17.56% respectively, suggesting no significant difference.

Table 2: Restricting speaker transitions to word boundaries

Restrict in		*dev* speaker-attributed WER (%)	
forward	backward	forward	backward
no	no	17.55	17.72
yes	no	17.60	17.78
yes	yes	17.65	17.77

Section 4 also describes the possibility of restricting the speaker transitions, such that speaker changes are only allowed at word boundaries. This restriction can be applied in either or both of the forward and backward passes. Table 2 assess these restrictions on the *dev* set. Here, the backward pass used only 1000 sub-sampled particles for faster experimentation. The results suggest that there may not be any significant gain yielded by enforcing these restrictions. The 17.60% and 17.65% forward pass speaker-attributed WERs for when speaker transitions are restricted in the forward pass differ because of the stochasticity of the SSPF model.

Table 3 compares the SSPF against the HMM from [15], on both the *dev* and *eval* sets. Here, the meetings were categorised into those with and without speaker movements. A meeting was considered to have movement if it had at least one speaker, such that it was possible to find two disjoint angular arcs of at least $\frac{\pi}{6}$ radians (i.e. 30°) each, and that speaker spent at least 30s of active speech in each of the two remaining regions not covered by these two arcs, based on manually transcribed locations from video. The results suggest that the SSPF may improve the speaker-attributed WER over the HMM for meetings with movement. Although the gains may be small for each of the *dev* and *eval* sets, they are consistent across both data sets. However, the SSPF seems to degrade the performance of stationary meetings compared to the HMM. If speakers are fairly stationary throughout a meeting, then their static location information may be particularly useful for the diarisation task. This scenario may

Table 3: Effect of explicitly modelling movement

Test set	Model	Speaker-attributed WER (%)		
		stationary	moving	average
dev	HMM	16.59	18.19	17.53
	SSPF	16.68	18.14	17.55
eval	HMM	19.45	15.26	16.02
	SSPF	19.54	15.17	16.00

fit well with the assumptions of the HMM, which does not explicitly model changes in the speaker locations. It is shown in [15] that EM fine-tuning of the initial state and state transition probabilities on the current test meeting yields improvement for the HMM. It is difficult to perform per-meeting fine-tuning of the analogous parameters in the SSPF in a computationally feasible manner, and these were instead only initialised from the AHC hypothesis. Despite this, the SSPF is able to perform comparably with the HMM on average.

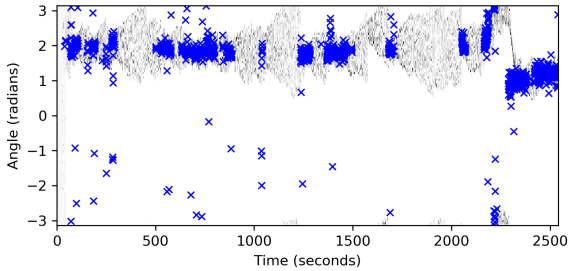


Fig. 4: Example prediction of a speaker's location. Blue crosses represent the DOA observations, while the heat map shows the weighted distribution of the particles, where darker means higher probability

An advantage of the SSPF over the HMM is that the SSPF can yield the estimated locations of each of the speakers as they move, through the duration of the meeting. An example of such a predicted location trace after the forward pass is illustrated in Figure 4. The location estimation continues, even when the speaker is silent. The particles express growing uncertainty about the speaker's location, as the duration of silence increases. The SSPF tracks the location, even with a movement of 1 radian ($\approx 60°$), near the end of the meeting.

Even though the speaker-attributed WER may not yield significant improvements in the proposed SSPF approach, the contributions of this paper may still be useful. The ability of the SSPF to hypothesise the movement trajectories of speakers, even when they are not speaking, may be useful to downstream tasks. Furthermore, this paper has demonstrated that is it possible to apply a switching state-space model implemented as a particle filter in a realistic scenario, while achieving comparable performance to alternative approaches.

7. CONCLUSION

This paper has proposed a framework to jointly perform diarisation and speaker location tracking. A switching state-space model is implemented as a particle filter, with discrete chains that represent speaker turns, which are used to switch between continuous chains that express speaker locations. This model performs comparably with a previously proposed HMM diarisation approach that models static speaker locations.

8. REFERENCES

[1] R. Tibshirani, G. Walther, and T. Hastie, "Estimating the number of clusters in a data set via the gap statistic," *Journal of the Royal Statistical Society B*, vol. 63, no. 2, pp. 411–423, 2001.

[2] T. J. Park, K. J. Han, M. Kumar, and S. Narayanan, "Auto-tuning spectral clustering for speaker diarization using normalized maximum eigengap," *IEEE Signal Processing Letters*, vol. 27, pp. 381–385, Dec 2019.

[3] S. Shum, N. Dehak, E. Chuangsuwanich, D. Reynolds, and J. Glass, "Exploiting intra-conversation variability for speaker diarization," in *Interspeech*, Florence, Italy, Aug 2011, pp. 945–948.

[4] H. Ning, M. Liu, H. Tang, and T. Huang, "A spectral clustering approach to speaker diarization," in *ICSLP*, Pittsburgh, USA, Sep 2006, pp. 2178–2181.

[5] M. A. Siegler, U. Jain, B. Raj, and R. M. Stern, "Automatic segmentation, classification and clustering of broadcast news audio," in *DARPA Speech Recognition Workshop*, Chantilly, USA, Feb 1997, pp. 97–99.

[6] H. Jin, F. Kubala, and R. Schwartz, "Automatic speaker clustering," in *DARPA Speech Recognition Workshop*, Chantilly, USA, Feb 1997.

[7] J. Ajmera and C. Wooters, "A robust speaker clustering algorithm," in *ASRU*, St. Thomas, US Virgin Islands, Nov 2003, pp. 411–416.

[8] M. Diez, L. Burget, S. Wang, J. Rohdin, and H. Černocký, "Bayesian HMM based x-vector clustering for speaker diarization," in *Interspeech*, Graz, Austria, Sep 2019, pp. 346–350.

[9] F. Landini, S. Wang, M. Diez, L. Burget, P. Matějka, K. Žmolíková, L. Mošner, A. Silnova, O. Plchot, O. Novotný, H. Zeinali, and J. Rohdin, "BUT system for the second DIHARD speech diarization challenge," in *ICASSP*, Barcelona, Spain, May 2020, pp. 6529–6533.

[10] S. Horiguchi, Y. Fujita, S. Watanabe, Y. Xue, and P. García, "Encoder-decoder based attractors for end-to-end neural diarization," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 1493–1507, Mar 2022.

[11] N. Zheng, N. Li, J. W. Yu, C. Weng, D. Su, X. Y. Liu, and H. Meng, "Multi-channel speaker diarization using spatial features for meetings," in *ICASSP*, Singapore, May 2022, pp. 7337–7341.

[12] S. Horiguchi, S. Watanabe, P. García, Y. Xue, Y. Takashima, and Y. Kawaguchi, "Towards neural diarization for unlimited numbers of speakers using global and local attractors," in *ASRU*, Cartagena, Colombia, Dec 2021, pp. 98–105.

[13] J. M. Pardo, X. Anguera, and C. Wooters, "Speaker diarization for multiple-distant-microphone meetings using several sources of information," *IEEE Transactions on Computers*, vol. 56, no. 9, pp. 1212–1224, Sep 2007.

[14] D. Vijayasenan and F. Valente, "Speaker diarization of meetings based on large TDOA feature vectors," in *ICASSP*, Kyoto, Japan, Mar 2012, pp. 4173–4176.

[15] J. H. M. Wong, X. Xiao, and Y. Gong, "Hidden Markov model diarisation with speaker location information," in *ICASSP*, Toronto, Canada, Jun 2021, pp. 7158–7162.

[16] Z. Shaik and V. Asari, "A robust method for multiple face tracking using Kalman filter," in *AIPR*, Washington DC, USA, Oct 2007, pp. 125–130.

[17] J. Foytik, P. Sankaran, and V. Asari, "Tracking and recognizing multiple faces using Kalman filter and ModularPCA," *Procedia Computer Science*, vol. 6, pp. 256–261, 2011.

[18] I. D. Gebru, S. Ba, G. Evangelidis, and R. Horaud, "Tracking the active speaker based on a joint audio-visual observation model," in *ICCVW*, Santiago, Chile, Dec 2015, pp. 702–708.

[19] C. Evers, H. W. Löllmann, H. Mellmann, A. Schmidt, H. Barfuss, P. A. Naylor, and W. Kellermann, "The LOCATA challenge: acoustic source localization and tracking," *IEEE/ACM Transactions on Audio, Speech, and Language processing*, vol. 28, pp. 1620–1643, Apr 2020.

[20] D. Bechler, M. Grimm, and K. Kroschel, "Speaker tracking with a microphone array using Kalman filtering," *Advances in Radio Science*, vol. 1, pp. 113–117, May 2003.

[21] D. Salvati, C. Drioli, and G. L. Foresti, "Localization and tracking of an acoustic source using a diagonal unloading beamforming and a Kalman filter," in *LOCATA Challenge Workshop*, Tokyo, Japan, Sep 2018.

[22] I. Marković and I. Petrović, "Speaker localization and tracking with a microphone array on a mobile robot using von Mises distribution and particle filtering," *Robotics and Autonomous Systems*, vol. 58, no. 11, pp. 1185–1196, Nov 2010.

[23] C. Segura, A. Abad, J. Hernando, and C. Nadeu, "Multispeaker localization and tracking in intelligent environments," in *CLEAR2007 and RT2007*, Baltimore, USA, May 2007, pp. 82–90.

[24] T. Gehrig and J. McDonough, "Tracking multiple speakers with probabilistic data association filters," in *CLEAR*, Southampton, UK, Apr 2006, pp. 137–150.

[25] M. Murase, S. Yamamoto, J.-M. Valin, K. Nakadai, K. Yamada, K. Komatani, T. Ogata, and H. G. Okuno, "Multiple moving speaker tracking by microphone array on mobile robot," in *Interspeech*, Lisbon, Portugal, Sep 2005, pp. 249–252.

[26] J. McDonough, K. Kumatani, T. Arakawa, K. Yamamoto, and B. Raj, "Speaker tracking with spherical microphone arrays," in *ICASSP*, Vancouver, Canada, May 2013, pp. 3981–3985.

[27] A. Plinge and G. A. Fink, "Multi-speaker tracking using multiple distributed microphone arrays," in *ICASSP*, Florence, Italy, May 2014, pp. 614–618.

[28] Z. Ghahramani and G. E. Hinton, "Variational learning for switching state-space models," *Neural Computation*, vol. 12, no. 4, pp. 831–864, Apr 2000.

[29] N. Ito, S. Araki, and T. Nakatani, "Complex angular central Gaussian mixture model for directional statistics in mask-based microphone array signal processing," in *EUSIPCO*, Budapest, Hungary, Aug 2016, pp. 1153–1157.

[30] T. Yoshioka, I. Abramovski, C. Aksoylar, Z. Chen, M. David, D. Dimitriadis, Y. Gong, I. Gurvich, X. Huang, Y. Huang, A. Hurvitz, L. Jiang, S. Koubi, E. Krupka, I. Leichter, C. Liu, P. Parthasarathy, A. Vinnikov, L. Wu, X. Xiao, W. Xiong, H. Wang, Z. Wang, J. Zhang, Y. Zhao, and T. Zhou, "Advances in online audio-visual meeting transcription," in *ASRU*, Singapore, Dec 2019, pp. 276–283.

[31] E. Gordon-Rodriguez, G. Loaiza-Ganem, and J. P. Cunningham, "The continuous categorical: a novel simplex-valued exponential family," in *ICML*, Jul 2020, pp. 3637–3647.

[32] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *IEE Proceedings-F*, vol. 140, no. 2, pp. 107–113, Apr 1993.

[33] D. B. Rubin, "A noniterative sampling/importance resampling alternative to the data augmentation algorithm for creating a few imputations when fractions of missing information are modest: the SIR algorithm," *Journal of the American Statistical Association*, vol. 82, no. 398, pp. 543–546, Jun 1987.

[34] J. Carpenter, P. Clifford, and P. Fearnhead, "An improved particle filter for non-linear problems," *IEE Proceedings - Radar, Sonar and Navigation*, vol. 146, no. 1, pp. 2–7, Feb 1999.

[35] A. Kong, J. S. Liu, and W. H. Wong, "Sequential imputations and Bayesian missing data problems," *Journal of the American Statistical Association*, vol. 89, no. 425, pp. 278–288, Mar 1994.

[36] A. Doucet, S. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and Computing*, vol. 10, pp. 197–208, Jul 2000.

[37] J. H. Friedman, J. L. Bentley, and R. A. Finkel, "An algorithm for finding best matches in logarithmic expected time," *ACM Transactions on Mathematical Software*, vol. 3, no. 3, pp. 209–226, Sep 1977.

[38] A. G. Gray and A. W. Moore, "'N-body' problems in statistical learning," in *NIPS*, Denver, USA, Nov 2000, pp. 521–527.

[39] M. Klaas, M. Briers, N. de Freitas, A. Doucet, S. Maskell, and D. Lang, "Fast particle smoothing: if I had a million particles," in *ICML*, Pittsburgh, USA, Jun 2006, pp. 481–488.

[40] T. Zhou, Y. Zhao, and J. Wu, "ResNeXt and Res2Net structures for speaker verification," in *SLT*, Shenzhen, China, Jan 2021, pp. 301–307.

[41] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, no. 1-2, pp. 83–97, Mar 1955.