Classification:
Public

Document Type:
Scientific Report

Document Reference:
PRJ-NICTA-PM-023

Status:
Final

Revision:
1.2

Date:
August 1, 2007

A joint venture between:
The University of Western Australia &
Curtin University of Technology

E-mail: ericl@watri.org.au
Tel.: +61 (0)8 6488 4642

Title:

Dynamics Models for Acoustic Speaker Tracking—Preliminary Results

Author(s):

Eric A. Lehmann and Anders M. Johansson

Document History:

Revision	Date	Comments
1.0	March 6, 2007	Initial draft
1.1	March 12, 2007	Corrections from internal review
1.2	August 1, 2007	Public classification status (paper version to appear at WASPAA'07)

Contents

Abstract

This report presents a study of various dynamics models (motion models) for an implementation in the frame of a particle filtering approach to acoustic speaker tracking. As a Bayesian filtering method, a particle filter relies on the definition of two main concepts, namely *i)* the measurement modality, and *ii)* the transition (dynamics) equation. Whereas a significant research effort has been devoted to improving the algorithm's performance by considering various types of observations (measurements), the influence of the dynamics formulation on the resulting tracking accuracy has received little attention so far. This work attempts to provide some insight into this secondary aspect of particle-filter design by considering several types of models, based on a Cartesian coordinates as well as polar coordinates definition of the state vector. A coarse calibration of the model parameters is carried out on the basis of real audio data recorded in a reverberant environment. The performance of the resulting particle filters is then assessed using extensive experimental simulations. An aspect of special interest considered in this work is related to the behaviour of the tracker during the silence gaps existing between utterances in the speech signal. This report demonstrates that the ability to achieve a reduced tracking error during periods of speech inactivity is influenced by both the chosen model as well as the specific tuning of its parameters.

1 Introduction

As a Bayesian filtering approach, the development of a particle filter (PF) for the acoustic speaker tracking (AST) problem requires the definition of two important concepts [1, 2]:

1) the measurement or observation PDF (probability density function), also known as the likelihood function, and

2) the transition PDF, based on a model describing the dynamics of the considered target.

In the literature currently available on the AST topic [3–5], a significant research effort is directed towards the development of improved measurement densities, and very little attention is given to the type of motion model implemented in the algorithm. Originally defined in [3], the so-called "Langevin" dynamics constitute the generic model of choice routinely implemented in these AST publications. However, no real justification can be found in the literature as to why this specific type of model was chosen in the first place. The research presented in this report is not meant to be a comprehensive review of existing dynamics models for AST. Instead, the aim is to provide some preliminary insight into the influence of the assumed dynamics on the overall tracking performance, something that has not been the object of significant research so far. Also, the results presented in this document are to be considered in the frame of a more in-depth research on the optimisation of various motion models, which is currently being carried out as a continuation of this work.

There exists a relatively large number of different motion models suitable for a potential implementation in relation to the AST problem definition [6]; an exhaustive list can be found, for instance, in a rigorous survey by Li and Jilkov in [7]. Most of the available motion dynamics are originally developed in relation to target tracking for military and defence systems. There is, however, no fundamental limitation that would preclude the application of such models in the context of particle filtering for AST. This work considers a subset of models which appear promising for such an implementation. In particular, we are interested in the specific behaviour of the resulting PF algorithm during the silence gaps existing between separate utterances in a typical speech signal. A PF method was recently proposed in [4] which takes into account the measurements obtained with a voice activity detection (VAD) scheme. This algorithm, denoted as PF-VAD, was developed on the basis of the usual Langevin dynamics, and as demonstrated in [8], this leads to the tracker effectively "freezing" its estimates and spreading the particles uniformly in all directions as soon as the speaker becomes inactive. In essence, this corresponds to the assumption that a person is equally likely to move in any direction following an interruption in the speech; with a uniform spreading of the particles, the algorithm hence blindly tracks *any* potential speaker motion while no observations are available.

This approach is however not fully representative of practical scenarios: typically, speakers moving in a given environment rarely exhibit abrupt changes in direction and velocity. In other words, it is more realistic to assume that during silence gaps, the speaker's motion will be similar to that displayed shortly before the break in the speech signal. Integrating this specific property of motion continuity within the tracking algorithm would hence lead to a superior tracking performance and robustness against disturbances (noise, competing speakers, etc.). As shown in this work, this can be achieved with a careful choice of dynamics model and an appropriate tuning of the model parameters.

This report is organised as follows. The next section briefly reviews the basic concepts of the PF-VAD method. Section 3 describes the various models considered in this work, and Section 4 discusses the optimisation method used in order to tune the different parameters for each model. Section 5 then presents the performance results obtained from experimental simulations of the resulting PF algorithms, and Section 6 finally concludes this report with a discussion of the presented developments and an overview of future research.

2 PF Algorithm Review

In this work, the considered dynamics models will be used in conjunction with the PF-VAD algorithm, a particle filter with VAD data integration, which was proposed in [4]. This section briefly reviews the operation of this tracking method.

2.1 PF Definitions

Assuming that a Cartesian coordinate system with known origin has been defined for the considered tracking setup, let $\boldsymbol{X}_k$ represent the state variable for time index k, corresponding to the position $\boldsymbol{\ell}_k = [x_k \;\; y_k]^{\mathrm{T}}$ of the target in the state space, as well as a model-dependent state sub-vector $\boldsymbol{S}_k$ of additional variables required by the system model (such as velocity, acceleration, etc.):

$$\boldsymbol{X}_k = [x_k \;\; y_k \;\; \boldsymbol{S}_k]^{\mathrm{T}}. \tag{1}$$

At any time step k, each microphone in the array delivers a frame of audio signal which is processed using a steered beamforming (SBF) principle. With $S_m(\omega) = \mathcal{F}\{s_m(t)\}$ the Fourier transform of the signal data from the m-th sensor, $m \in \{1, \ldots, M\}$, the output $\mathcal{P}(\boldsymbol{\ell})$ of a delay-and-sum beamformer steered to the location $\boldsymbol{\ell} = [x \;\; y]^{\mathrm{T}}$ is given as

$$\mathcal{P}(\boldsymbol{\ell}) = \int_{\Omega} \Big| \sum_{m=1}^{M} W_m(\omega)\, S_m(\omega)\, \mathrm{e}^{\,j\omega \|\boldsymbol{\ell} - \boldsymbol{\ell}_m\|/c} \Big|^2 \, \mathrm{d}\omega \,, \tag{2}$$

where c is the propagation velocity of sound waves, $\boldsymbol{\ell}_m = [x_m \;\; y_m]^{\mathrm{T}}$ is the known position of the m-th microphone, $W_m(\omega) = |S_m(\omega)|^{-1}$ is the phase-transform (PHAT) frequency weighting term, and Ω corresponds to the frequency range of interest, which is typically defined as $\Omega = \{\omega \,|\, 2\pi \cdot 300\mathrm{Hz} \leqslant \omega \leqslant 2\pi \cdot 3000\mathrm{Hz}\}$ for speech processing applications. Let the variable $\boldsymbol{Y}_k$ denote the observation (measurement), which corresponds to the localisation information resulting from this preprocessing of the audio signals.

The Bayesian filtering approach also requires a model representing the transition dynamics of the state variable:

$$\boldsymbol{X}_k = g(\boldsymbol{X}_{k-1}, \boldsymbol{u}_k) \,, \tag{3}$$

where $\boldsymbol{u}_k$ is a noise variable. The specific definition of the transition function $g(\cdot)$, and the analysis of its influence on the tracking results, constitute the object of focus of the present work.

2.2 PF Algorithm

A Bayesian filtering approach to the tracking problem attempts to determine, for each time step k, the so-called posterior density $p(\boldsymbol{X}_k|\boldsymbol{Y}_{1:k})$, where $\boldsymbol{Y}_{1:k} = \{\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_k\}$ represents the concatenation of all measurements up to time k. From a statistical point of view, the posterior PDF contains all the information available regarding the current condition of the state variable $\boldsymbol{X}_k$, and an estimate $\widehat{\boldsymbol{X}}_k$ of the state then follows, for instance, as the mean or the mode of this density. Particle filtering is an approximation technique that solves the Bayesian filtering problem by representing the posterior PDF as a set of N samples $\boldsymbol{X}_k^{(n)}$ of the state space (particles) with associated weights $w_k^{(n)}$, $n \in \{1, \ldots, N\}$ [1]. The algorithm used in the development of the PF-VAD method [4] is based on the bootstrap PF, originally defined in [2]; the general iteration principle of this algorithm is described in Algorithm 1. In Step 2 of this PF iteration (update step), the likelihood function $p(\boldsymbol{Y}_k|\boldsymbol{X}_k)$ is defined as a mixture PDF:

$$p(\boldsymbol{Y}_k|\boldsymbol{X}_k) = q_{0,k} \cdot \mathcal{U}_{\mathcal{D}}(\boldsymbol{\ell}_k) + \gamma\,(1 - q_{0,k}) \cdot \big(\mathcal{P}(\boldsymbol{\ell}_k)\big)^2, \tag{4}$$

where γ is a normalisation constant, $\mathcal{U}_{\mathcal{D}}(\cdot)$ denotes the uniform distribution over the considered enclosure domain $\mathcal{D} = \{(x, y) \,|\, x_{\min} \leqslant x \leqslant x_{\max},\, y_{\min} \leqslant y \leqslant y_{\max}\}$, and $q_{0,k}$ represents the prior probability of a clutter SBF measurement, defined on the basis of the VAD output α_k as

$$q_{0,k} = 1 - \alpha_k \,. \tag{5}$$

Readers are referred to [4] for more information regarding this specific PF implementation.

2.3 Performance Assessment Parameters

The PF estimation error for the current frame is

$$\varepsilon_k = \big\|\boldsymbol{\ell}_{\mathrm{S},k} - \hat{\boldsymbol{\ell}}_k\big\| \,, \tag{6}$$

Assumption: at time $k-1$, assume that the set of particles $\boldsymbol{X}_{k-1}^{(n)}$ and weights $w_{k-1}^{(n)}$, $n \in \{1, \ldots, N\}$, is a discrete representation of the posterior $p(\boldsymbol{X}_{k-1}|\boldsymbol{Y}_{1:k-1})$.

Iteration: given the observation $\boldsymbol{Y}_k$ obtained during the current time k, update each particle $n \in \{1, \ldots, N\}$ as follows:

1. *Prediction:* propagate the particle through the transition equation,

$$\widetilde{\boldsymbol{X}}_k^{(n)} = g(\boldsymbol{X}_{k-1}^{(n)}, \boldsymbol{u}_k)\,.$$

2. *Update:* assign a likelihood weight to each new particle,

$$\tilde{w}_k^{(n)} = w_{k-1}^{(n)} \cdot p(\boldsymbol{Y}_k|\widetilde{\boldsymbol{X}}_k^{(n)})\,,$$

then normalize the weights:

$$w_k^{(n)} = \tilde{w}_k^{(n)} \cdot \Big(\sum_{i=1}^{N} \tilde{w}_k^{(i)}\Big)^{-1}.$$

3. *Resampling:* compute the effective sample size

$$N_{\text{eff}} = \Big(\sum_{n=1}^{N} \big(w_k^{(n)}\big)^2\Big)^{-1}.$$

If $N_{\text{eff}} \geqslant N_{\text{thr}}$, where N_{thr} is some pre-defined threshold, simply define $\boldsymbol{X}_k^{(n)} = \widetilde{\boldsymbol{X}}_k^{(n)}$, $\forall n$. Otherwise, draw N new samples $\boldsymbol{X}_k^{(n)}$ from the existing set of particles $\{\widetilde{\boldsymbol{X}}_k^{(i)}\}_{i=1}^N$ according to their weights $w_k^{(i)}$, then reset the weights to uniform values: $w_k^{(n)} = 1/N$, $n \in \{1, \ldots, N\}$.

Result: the new set $\{(\boldsymbol{X}_k^{(n)}, w_k^{(n)})\}_{n=1}^N$ is approximately distributed as the posterior density $p(\boldsymbol{X}_k|\boldsymbol{Y}_{1:k})$. An estimate of the target's location at time k can then be obtained as

$$\hat{\boldsymbol{\ell}}_k = \sum_{n=1}^{N} w_k^{(n)} \boldsymbol{\ell}_k^{(n)},$$

where $\boldsymbol{\ell}_k^{(n)}$ corresponds to the location information in the n-th particle vector: $\boldsymbol{X}_k^{(n)} = [\boldsymbol{\ell}_k^{(n)} \;\; \dot{\boldsymbol{\ell}}_k^{(n)}]^{\mathrm{T}}$.

Algorithm 1: Generic bootstrap PF algorithm.

where $\boldsymbol{\ell}_{\mathrm{S},k}$ is the ground-truth source position at time k. In order to assess the overall performance of the algorithm for a given sample of audio data, the average error is computed as the RMSE parameter (root-mean-square error)

$$\bar{\varepsilon} = \sqrt{\frac{1}{K}\sum_{k=1}^{K} \varepsilon_k^2}\,, \tag{7}$$

with K representing the total number of frames in the considered audio sample.

Due to the partially random nature of PF implementations, statistical averaging over a large number D of algorithm runs is used in the results presentation. A parameter of particular interest to AST is the percentage of these runs for which the tracking algorithm completely loses track of the target during the simulation, typically due to significant silence gaps in the speech or an incorrect setting of the model parameters. For each simulation run $d \in \{1, \ldots, D\}$, a track loss parameter is thus defined as

$$\zeta_d = \begin{cases} 1 & \text{if } \big(\sum_{k=K-k^*}^{K} \varepsilon_{k,d}\big)/(k^*-1) > \delta\,, \\ 0 & \text{otherwise}\,, \end{cases} \tag{8}$$

with $k^* = \lceil 0.5/T \rceil$ and T representing the update time period (from time $k-1$ to k). The parameter ζ_d effectively checks whether the average estimation error over the last 0.5s of audio data is smaller than some threshold, set here to $\delta = 0.1$m, i.e., whether the algorithm is still correctly tracking the target at the end of the simulation run. The global track loss percentage (TLP) $\bar{\zeta}$ (expressed in %) for a given audio sample is then defined as

$$\bar{\zeta} = \frac{100}{D} \sum_{d=1}^{D} \zeta_d \,.$$

3 Dynamics Models

As mentioned previously, several dynamics models represent potential candidates for an implementation in the frame of AST [6, 7]. In the following, we investigate two main model types:

1. coordinate-uncoupled (CU) models, where the target's velocity is represented using a Cartesian coordinate setting; and
2. curvilinear (CL) models, which represent the target's velocity using a polar coordinate system.

Additionally, for any given state variable ξ, such as the target's velocity or acceleration for instance, the transition equation can be defined either as:

1. a purely random-walk (RW) process with variance σ_ξ^2:

$$\xi_k = \xi_{k-1} + \sigma_\xi \cdot u_k \,, \tag{9}$$
$$u_k \sim \mathcal{N}(0,1) \,, \tag{10}$$

   where $\mathcal{N}(\mu, \sigma^2)$ denotes a Gaussian PDF with mean μ and variance σ^2; or

2. a time-correlated (TC) process with variance σ_ξ^2 and correlation time constant $1/\beta_\xi$, whose discrete-time representation is

$$\xi_k = \mathrm{e}^{-\beta_\xi T} \cdot \xi_{k-1} + \sigma_\xi \sqrt{1 - \mathrm{e}^{-2\beta_\xi T}} \cdot u_k \,, \tag{11}$$
$$u_k \sim \mathcal{N}(0,1) \,, \tag{12}$$

   where T is the update time period (from time $k-1$ to k). This definition implicitly represents ξ as a zero-mean first-order stationary Markov process with autocorrelation

$$R_\xi(\tau) = \mathbb{E}\{\xi(t+\tau)\xi(t)\} = \sigma_\xi^2 \cdot \mathrm{e}^{-\beta_\xi |\tau|} \,, \tag{13}$$

   where $\mathbb{E}\{\cdot\}$ is the statistical expectation operator. The parameter β effectively represents the "manoeuvre" time constant (for the current state variable) and thus, the time-correlated process in (11)–(12) gives an indication of how long a target manoeuvre lasts. This representation has a wider coverage compared to a non-correlated random-walk process, which might be advantageous for modeling speakers in the frame of AST.

Finally, different representations also result depending on the considered model order, i.e., depending on whether the model requires the inclusion of the target's velocity or acceleration in the state vector. As mentioned in [6, p.202], experience seems to indicate that the use of an acceleration variable is often only of value when a velocity measurement is available. Since this is usually not the case in the context of AST, the following developments will focus mainly on first-order models which do not account for the target's acceleration.

The different combinations of the above choices, i.e., model type, model order, and whether each state variable is time-correlated or not, would lead to a prohibitively large number of different dynamics representations to be assessed. As a result, this work only considers a handful of models, whose implementation in relation to the AST problem setting was deemed promising or at least of some interest. The following subsections enumerate these different models in more detail. Unless otherwise stated, the generic noise variables u_k and u'_k used in the rest of this section are assumed to be zero-mean Gaussian variables with unit variance:

$$u_k, u'_k \sim \mathcal{N}(0,1) \,. \tag{14}$$

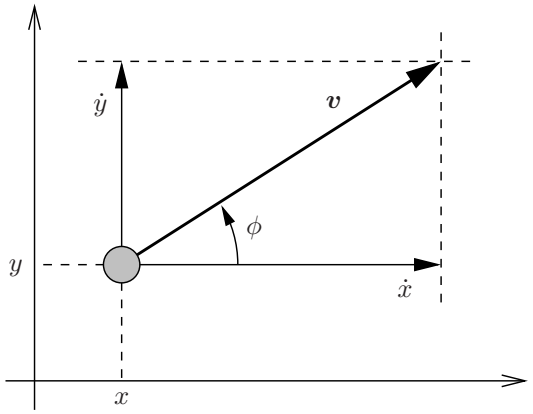

Figure 1: Definition of variables related to the chosen coordinate system, where $\boldsymbol{v}$ represents the target's velocity vector.

3.1 Coordinate-Uncoupled (CU) Models

Coordinate-uncoupled dynamics are characterised by the fact that the target's velocity is defined in a Cartesian coordinate system, as depicted in Figure 1. As defined in [4], which uses a CU dynamics model (Langevin), the state vector

$$\boldsymbol{X}_k = [x_k \;\; y_k \;\; \boldsymbol{S}_k]^{\mathrm{T}}, \tag{15}$$

follows from a definition of the model-dependent state sub-vector $\boldsymbol{S}_k$ given by

$$\boldsymbol{S}_k \triangleq [\dot{x}_k \;\; \dot{y}_k]\,. \tag{16}$$

By definition, this type of manoeuvre model assumes that the motion coupling between coordinates is negligible. Consequently, only one generic coordinate direction, chosen here to be the x variable, needs to be considered in the following derivations. The dynamics for the y dimension are defined in an identical manner.

3.1.1 CU-RWVE Model

This CU model is a first-order representation where the velocity (VE) in each dimension is defined as a random walk (RW) with variance σ_v^2, as follows:

$$\begin{bmatrix} x_k \\ \dot{x}_k \end{bmatrix} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_{k-1} \\ \dot{x}_{k-1} \end{bmatrix} + \begin{bmatrix} T/2 \\ 1 \end{bmatrix} \cdot \sigma_v \, u_k\,. \tag{17}$$

This formulation is also known as the "nearly-constant velocity" model.

3.1.2 CU-TCVE Model

This model results from a definition of the target's velocity as a time-correlated (TC) process with variance σ_v^2 and rate constant β_v, resulting in

$$\begin{bmatrix} x_k \\ \dot{x}_k \end{bmatrix} = \begin{bmatrix} 1 & (1-\mathrm{e}^{-\beta_v T})/\beta_v \\ 0 & \mathrm{e}^{-\beta_v T} \end{bmatrix} \cdot \begin{bmatrix} x_{k-1} \\ \dot{x}_{k-1} \end{bmatrix} + \begin{bmatrix} T \\ 1 \end{bmatrix} \cdot \sigma_v \sqrt{1-\mathrm{e}^{-2\beta_v T}} \cdot u_k\,. \tag{18}$$

3.1.3 CU-TCAC Model

As stated earlier, including an acceleration (AC) component in the state vector might not necessarily be an advantage if no velocity observations are available. In order to determine the relevance of this statement for AST, this work also considers a second-order version of the previous model. The CU-TCAC model hence extends the state sub-vector as follows:

$$\boldsymbol{S}_k \triangleq [\dot{x}_k \;\; \dot{y}_k \;\; \ddot{x}_k \;\; \ddot{y}_k]\,, \tag{19}$$

with the acceleration modeled as a time-correlated process with variance σ_a^2 and rate constant β_a [7]:

$$\begin{bmatrix} x_k \\ \dot{x}_k \\ \ddot{x}_k \end{bmatrix} = \begin{bmatrix} 1 & T & (\beta_a T - 1 - \mathrm{e}^{-\beta_a T})/\beta_a^2 \\ 0 & 1 & (1 - \mathrm{e}^{-\beta_a T})/\beta_a \\ 0 & 0 & \mathrm{e}^{-\beta_a T} \end{bmatrix} \cdot \begin{bmatrix} x_{k-1} \\ \dot{x}_{k-1} \\ \ddot{x}_{k-1} \end{bmatrix} + \begin{bmatrix} T^2/2 \\ T \\ 1 \end{bmatrix} \cdot \sigma_a \sqrt{1 - \mathrm{e}^{-2\beta_a T}} \cdot u_k \,. \tag{20}$$

3.1.4 CU-LAN Model

The Langevin (LAN) dynamics model is also part of the CU class. The definition originally provided in [3] is very close to the CU-TCVE model given in Section 3.1.2, with only a slight variation in the definition of the transition matrix:

$$\begin{bmatrix} x_k \\ \dot{x}_k \end{bmatrix} = \begin{bmatrix} 1 & T \cdot \mathrm{e}^{-\beta_v T} \\ 0 & \mathrm{e}^{-\beta_v T} \end{bmatrix} \cdot \begin{bmatrix} x_{k-1} \\ \dot{x}_{k-1} \end{bmatrix} + \begin{bmatrix} T \\ 1 \end{bmatrix} \cdot \sigma_v \sqrt{1 - \mathrm{e}^{-2\beta_v T}} \cdot u_k \,. \tag{21}$$

Note that the term $T \cdot \mathrm{e}^{-\beta_v T}$ tends towards T for $\beta_v \to 0$, and towards 0 for $\beta_v \to \infty$, just like the term $(1 - \mathrm{e}^{-\beta_v T})/\beta_v$ does in the definition of CU-TCVE. The CU-LAN model is included as a benchmark in the current analysis, since it has been used in most of the publications of the AST literature so far.

3.2 Curvilinear (CL) Models

The curvilinear class comprises dynamics models which define the target's velocity vector $\boldsymbol{v}$ in a polar coordinate system, i.e., in terms of its magnitude $v = \|\boldsymbol{v}\|$ and its orientation (turn) angle ϕ, see Figure 1. With this approach, the state vector is typically defined as

$$\boldsymbol{X}_k = [x_k \;\; y_k \;\; \boldsymbol{S}_k]^{\mathrm{T}}, \tag{22}$$

$$\boldsymbol{S}_k \triangleq [v_k \;\; \phi_k] \,, \tag{23}$$

with the target's position at time k then resulting indirectly as

$$\begin{bmatrix} x_k \\ y_k \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_{k-1} \\ y_{k-1} \end{bmatrix} + T \cdot v_k \cdot \begin{bmatrix} \cos(\phi_k) \\ \sin(\phi_k) \end{bmatrix} \,, \tag{24}$$

3.2.1 CL-RWVE-RWTU Model

The CL-RWVE-RWTU model defines the velocity as a random-walk process with variance σ_v^2, and the turn angle (TU) as a RW process with variance σ_ϕ^2:

$$\begin{bmatrix} v_k \\ \phi_k \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} v_{k-1} \\ \phi_{k-1} \end{bmatrix} + \begin{bmatrix} \sigma_v \, u_k \\ \sigma_\phi \, u'_k \end{bmatrix} . \tag{25}$$

3.2.2 CL-TCVE-TCTU Model

This model defines each state variable as a time-correlated process, resulting in:

$$\begin{bmatrix} v_k \\ \phi_k \end{bmatrix} = \begin{bmatrix} \mathrm{e}^{-\beta_v T} & 0 \\ 0 & \mathrm{e}^{-\beta_\phi T} \end{bmatrix} \cdot \begin{bmatrix} v_{k-1} \\ \phi_{k-1} \end{bmatrix} + \begin{bmatrix} \sigma_v \sqrt{1 - \mathrm{e}^{-2\beta_v T}} \cdot u_k \\ \sigma_\phi \sqrt{1 - \mathrm{e}^{-2\beta_\phi T}} \cdot u'_k \end{bmatrix} . \tag{26}$$

3.2.3 CL-RWVE-RAAC Model

A different variant of curvilinear model is given by considering the dynamics of the turn angle ϕ_k through the target's normal acceleration a_k, which corresponds to the projection of the acceleration vector onto an axis perpendicular to the velocity vector. This type of models hence use the following representation of the state sub-vector:

$$\boldsymbol{S}_k \triangleq [v_k \;\; a_k] \,, \tag{27}$$

and the current angle then results from the following relationship:

$$\phi_k = \phi_{k-1} + \frac{T \cdot a_k}{v_k} \,, \tag{28}$$

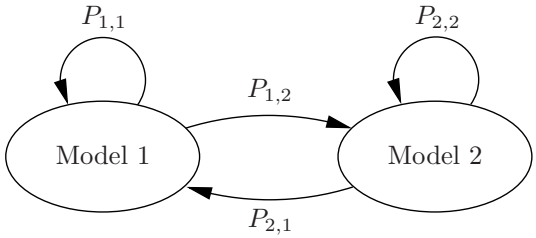

Figure 2: Two-state Markov process for switching-dynamics PF.

with the target's current position defined as given by (24). The relation in (28) is a discrete-time formulation resulting from the kinematics of a uniform circular motion, where $a = v \cdot \dot{\phi}$. It implicitly implements a mechanism that decreases the rate of angle change as the target's velocity increases, which is a behaviour that can be expected from a speaker moving in a typical room. This model however involves a singularity when $v_k = 0$ (stationary target), which means that this type of definition might potentially be inappropriate for AST.

Despite this, the CL-RWVE-RAAC model uses the above definitions in conjunction with a random-walk velocity, and a purely random (RA) acceleration process:

$$\begin{bmatrix} v_k \\ a_k \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} v_{k-1} \\ a_{k-1} \end{bmatrix} + \begin{bmatrix} \sigma_v\, u_k \\ \sigma_a\, u'_k \end{bmatrix}. \tag{29}$$

3.2.4 CL-RWVE-RWAC Model

This model uses the same definitions as formulated previously in (27) and (28), and defines both the velocity and normal acceleration as random-walk processes:

$$\begin{bmatrix} v_k \\ a_k \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} v_{k-1} \\ a_{k-1} \end{bmatrix} + \begin{bmatrix} \sigma_v\, u_k \\ \sigma_a\, u'_k \end{bmatrix}. \tag{30}$$

3.2.5 CL-TCVE-TCAC Model

Finally, the CL-TCVE-TCAC model also uses the formulation of Section 3.2.3, with both the velocity and acceleration defined as time-correlated processes:

$$\begin{bmatrix} v_k \\ a_k \end{bmatrix} = \begin{bmatrix} \mathrm{e}^{-\beta_v T} & 0 \\ 0 & \mathrm{e}^{-\beta_a T} \end{bmatrix} \cdot \begin{bmatrix} v_{k-1} \\ a_{k-1} \end{bmatrix} + \begin{bmatrix} \sigma_v \sqrt{1 - \mathrm{e}^{-2\beta_v T}} \cdot u_k \\ \sigma_a \sqrt{1 - \mathrm{e}^{-2\beta_a T}} \cdot u'_k \end{bmatrix}. \tag{31}$$

3.3 Switching-Dynamics PF

A potential extension of the PF principle described in Section 2 can be achieved by considering a switching-dynamics approach. Similar to the definition of a mixture likelihood function in (4), a mixture PDF could be implemented in Step 1 of Algorithm 1 as follows:

$$p(\boldsymbol{X}_k|\boldsymbol{X}_{k-1}) = P_1 \cdot p_1(\boldsymbol{X}_k|\boldsymbol{X}_{k-1}) + (1 - P_1) \cdot p_2(\boldsymbol{X}_k|\boldsymbol{X}_{k-1}), \tag{32}$$

where $p_1(\cdot)$ and $p_2(\cdot)$ are the transition PDFs associated with two different dynamics models, and P_1 is the probability of the target being in the first state. This principle can be easily implemented with a simple two-state Markov model, as depicted in Figure 2, and by extending the state vector with an additional variable representing the type of dynamics exhibited by the target (model 1 or 2). The particles would jump from one model to the other according to the transition probabilities, therefore keeping track of the target's dynamics state. With this approach, improved tracking results could potentially be achieved with the particle filter switching, for instance, between a model specialised in tracking constant-velocity motion, and another developed specifically for accelerated motion; or between a CU-based model and a CL-based one. Note that this switching-dynamics approach has already received significant attention in the literature, see, e.g., [9, 10].

Here too, the transition probabilities $P_{i,j}$ could draw on specific information obtained from a voice activity detector, for instance allowing the PF to use a normal tracking model during periods of speech activity, and switch to a purely constant-velocity model during silences in order to remain "on track" based on the target's last known velocity and heading direction. As shown in the following, however, some of the previously defined models are able achieve this behaviour without having to specifically switch to a different dynamics model. Consequently, the switching-dynamics approach does not really provide any advantage in the context of the current AST application, and this concept was hence not pursued further in this work.

A case of potential interest for this approach might be for mostly-stationary targets, as encountered in a teleconferencing environment for instance. A normal dynamics model could be used during periods of speech activity in order to keep track of slight variations in position (e.g., speaker shifting in their seat), and then switch to nearly-stationary dynamics when the speech signal is interrupted, in order to render the PF unable diverge quickly as a result of background noise.

4 Parameter Optimisation

Each of the dynamics models presented in Sections 3.2.5 to 3.1.1 contains at least one free parameter that requires to be optimised. This corresponds to the various variances σ and rate constants β for either the velocity, acceleration or turn angle. In the frame of AST, these parameters should ideally result from an optimisation process defined in the following fashion:

$$(\sigma_{\text{opt}}, \beta_{\text{opt}}) = \arg \min_{\sigma,\beta>0} \overline{\text{RMSE}}\,, \tag{33}$$

where the average error $\overline{\text{RMSE}}$ is computed by considering a large number of speakers, rooms, array setups and speaker trajectories in various environments with different SNR and reverberation levels. The acquisition of such a large amount of real-world data constitutes a significant practical challenge, and ongoing research is currently assessing the feasibility of carrying out this optimisation process using software simulations rather than real recordings.

A simplified approach is used in the present report. As previously mentioned, the main goal of the present document is to provide some preliminary insight on whether the use of a specific dynamics model can improve the tracking accuracy of a PF algorithm. Particular attention is given to the tracker's behaviour when the speech stops, and determining whether the best thing to do in such circumstances is to simply stop tracking, as in the original PF implementation of [4]. For this purpose, the model parameters were coarsely optimised using a series of audio samples recorded in a reverberant environment with reverberation time $T_{60} \approx 0.27$s and approximate background noise level SNR ≈ 20dB (white noise). Note that the optimisation was performed manually and independently for each model parameter, which might have lead to sub-optimal tuning of the parameters; a more accurate method is currently being developed for joint optimisation of all model parameters simultaneously, and is expected to lead to superior results.

The results obtained from this parameter-tuning process are summarised in Table 1, which lists the value of the parameters of interest for each of the considered models. The model denoted CU-LAN-ORIG corresponds to the Langevin formulation as given in Section 3.1.4, but using the parameter settings originally found in most of the current AST literature (i.e., non-optimised Langevin dynamics); this specific definition is included here as a benchmark for comparison with other models.

5 Experimental Simulations

5.1 Simulation Setup

The simulation results presented in this section were generated using a series of multi-channel audio recordings obtained with the following setup. An array of $M = 8$ omnidirectional microphones were placed at a constant height of 1.51m in a room with dimensions 3.5m $\times$ 3.1m $\times$ 2.2m, in a square fashion with one sensor pair centered on each side of the square, as depicted in Figure 3. The distance between the sensors in each pair is 0.8m, and the area spanned by the array is 2.52m $\times$ 2.52m. The walls in this environment were partially covered with acoustic foam, leading to a practical reverberation time

Model	σ_v	β_v	σ_a	β_a	σ_ϕ	β_ϕ
CU-RWVE	0.05	-	-	-	-	-
CU-TCVE	0.7	0.3	-	-	-	-
CU-TCAC	-	-	2	30	-	-
CU-LAN	0.7	0.2	-	-	-	-
CU-LAN-ORIG	0.8	10	-	-	-	-
CL-RWVE-RWTU	0.7	-	-	-	2	-
CL-TCVE-TCTU	0.9	0.2	-	-	3	2
CL-RWVE-RAAC	0.07	-	1.5	-	-	-
CL-RWVE-RWAC	0.3	-	2	-	-	-
CL-TCVE-TCAC	3	0.15	3	1	-	-

Table 1: Optimised values for each model parameter; β values are given in Hz, σ_v values in m/s, σ_a values in m/s^2, σ_ϕ values in rad.

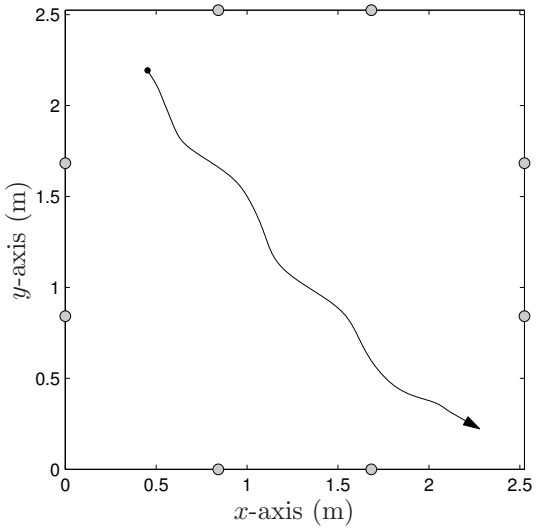

Figure 3: Array setup for real-audio recordings. Circles show the microphone positions, the line represents one of the speaker trajectories.

$T_{60} \approx 0.27$s (frequency-averaged up to 24kHz). Audio samples of background noise (white noise) were recorded separately from the speech signals, and used in the simulation phase with a variable level to generate specific values of SNR.

The availability of clean speech signals also allowed the precise measurement of the speaker trajectory directly from the audio data in each scenario, using the method proposed in [8]. The microphone signals were processed with a high-definition beamformer delivering accurate localisation estimates, with outliers easily discarded based on the approximate knowledge of the source trajectory combined with the output of some VAD scheme. Figure 3 displays an example of such a ground-truth measurement of the speaker trajectory in the above environment, based on a 7.8s speech sample from the speaker.

5.2 Tracking Example

The results presented in this section were obtained as follows. Each of the considered dynamics model was implemented in the framework of the PF algorithm described in Section 2, with $N = 75$ particles and $N_{\text{thr}} = 56.25$. These different algorithms were subsequently simulated using a tracking scenario example, which corresponds to part of the trajectory depicted in Figure 3, and with an approximate SNR level of 20dB (white noise). An example of audio signal recorded with one of the array sensors is shown in Figure 4, together with the output from the VAD scheme for this recording. In order to avoid unpredictable convergence effects, the particle set was initialised around the correct speaker trajectory at the start of each simulation.

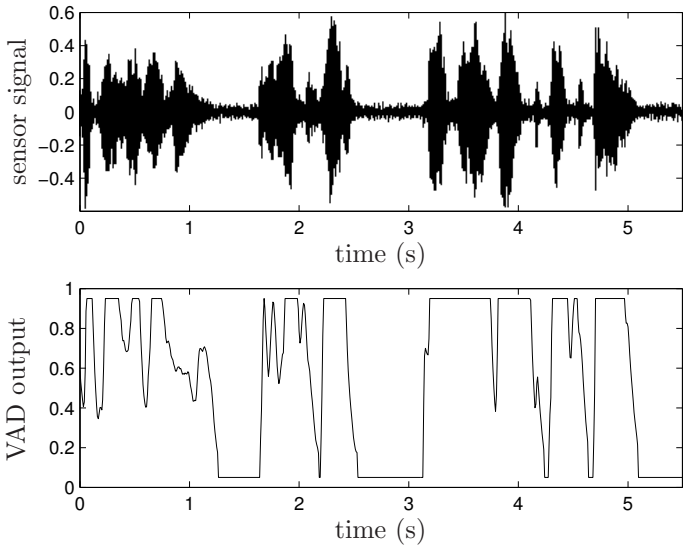

Figure 4: Example of sensor signal (top plot) and resulting VAD output (bottom plot) for simulations leading to the results of Figure 5.

Figure 5 shows the tracking results obtained with each of the considered models. The plots display the target's x-location estimate obtained from the PF versus time, averaged over 50 simulation runs for each model, as well as lines indicating the standard deviation of the particle set, i.e., the average particle spread in the x dimension.

The results of Figure 5 clearly illustrate that some of the considered models are able to achieve the desired behaviour: most of the CU models, as well as CL-RWVE-RAAC, maintain a proper heading angle and velocity when the speaker is inactive, i.e., when no observations are available. Based on the assumption that the speaker is unlikely to exhibit abrupt direction and velocity changes, this consequently results in a superior tracking performance.

It is also interesting to compare the results obtained for the Langevin dynamics with the original parameter values (CU-LAN-ORIG) and the optimised values (CU-LAN). The corresponding plots in Figure 5 demonstrate that the non-optimised model CU-LAN-ORIG in essence stops tracking whenever no measurements are available, as previously observed in [4, 8]; on the other hand, the optimised version of the Langevin model CU-LAN achieves a near-perfect tracking throughout the simulation. This consequently leads to the important observation that an improved tracking behaviour may not depend solely on the type of dynamics model used, but also relies on the specific tuning of its parameters. The following fact must therefore be specifically emphasised at this point: a failure by any model to achieve the desired behaviour in Figure 5 does not necessarily result from a poor formulation of the target's dynamics. It may well be the case that the considered model is fully relevant for AST, but the coarse optimisation process described in Section 4 failed to achieve appropriate values for the model parameters. The results from a more rigorous optimisation process, currently being developed, will be able to determine whether some of the considered dynamics models really are inappropriate for the current AST application.

Finally, an aspect of particular importance to consider in Figure 5 is the resulting standard deviation of the particle set during silence periods, as this represents a crucial factor for a successful PF tracking. The process of spreading the particles when no observations are available is what allows the algorithm to successfully resume tracking once the speech becomes active again, even in the eventuality that the target has slightly changed its course during the silence period. It can be seen from Figure 5 that most models achieve a satisfactory performance from this point of view, which suggests that the various model parameters in Table 1 were set to meaningful values; setting these values too tightly might result in the PF algorithm not being able to spread the particles fast enough.

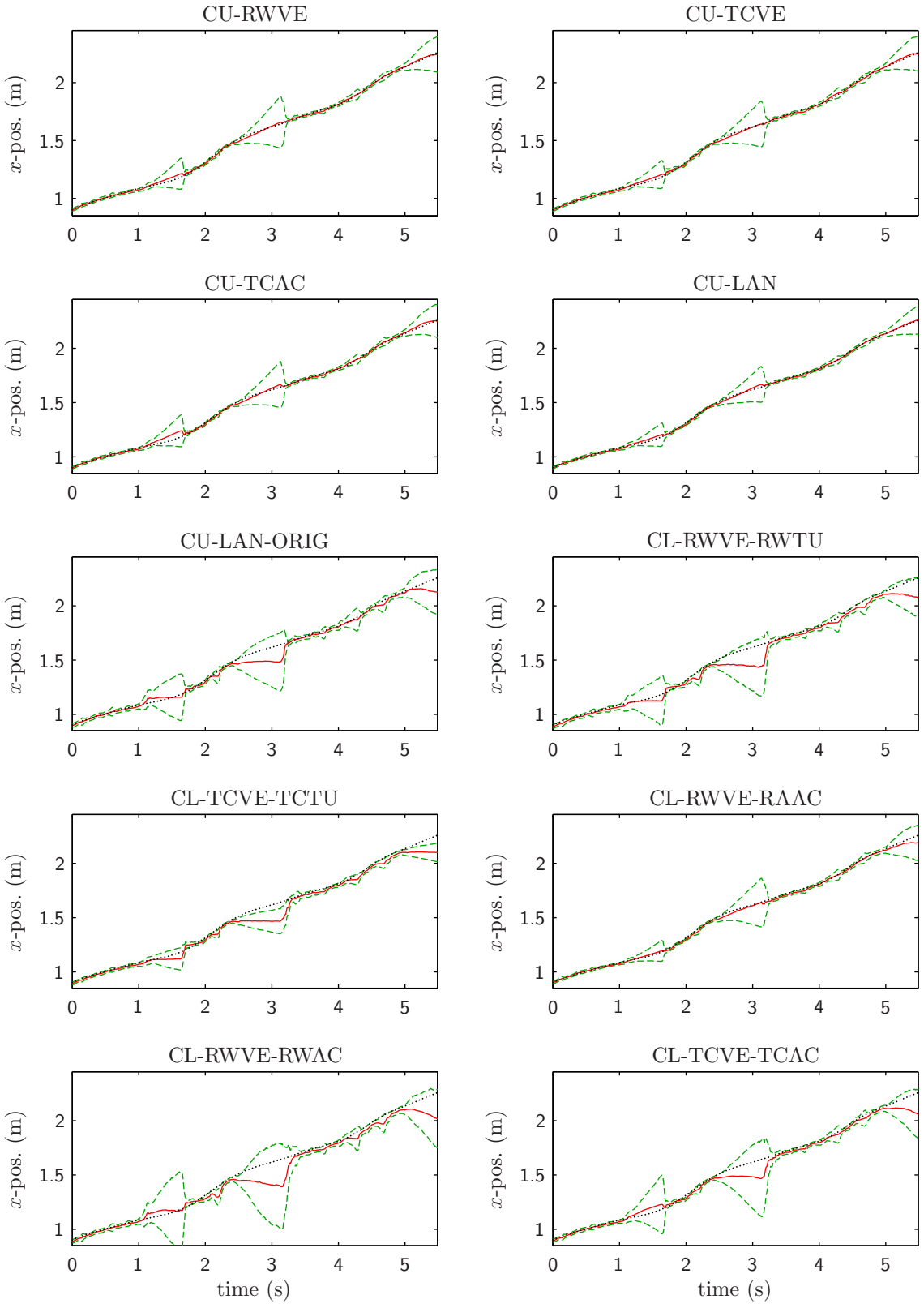

Figure 5: Example of tracking results for each considered model, showing the PF's x-position estimate (results in the y dimension are similar). Dotted lines correspond to the true source location, solid lines show the PF estimates, and dashed lines represent plus/minus one standard deviation of the particle set.

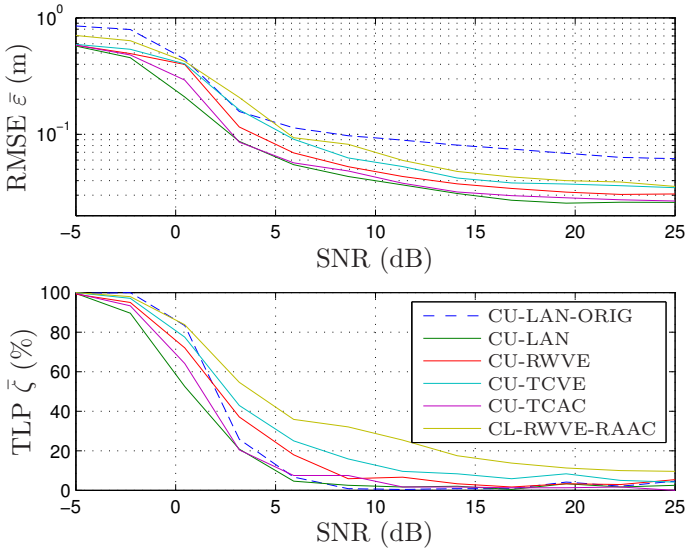

Figure 6: Tracking performance versus SNR level.

5.3 Average Performance vs. SNR

This section only considers the dynamics models which demonstrate a satisfactory behaviour in the simulations of Section 5.2, that is, all of the CU models as well as CL-RWVE-RAAC. While these models have shown to achieve successful tracking results at 20dB SNR, a risk exists that the chosen parameter values restricts their performance when the SNR decreases. Figure 6 hence presents the performance results for each of these models as a function of the SNR level; the benchmark model CU-LAN-ORIG is here also included for comparison purposes. For each SNR value, these results were averaged over a total of 240 simulation runs, corresponding to 60 runs carried out for each of four different audio recordings representing different speaker trajectories and speech signals.

Figure 6 shows a similar trend for all the considered models, with a breakdown of the tracking performance as the SNR decreases below approximately 5dB. This suggests that none of these methods suffers from a drastically erroneous setting of its parameters; the non-optimised Langevin model CU-LAN-ORIG presents a decreased overall tracking performance (larger RMSE results) as a result of an increased tracking error during periods of speech inactivity. Among all the other models, the most promising appears to be the one based on Langevin dynamics (optimised model version), which achieves the lowest average error and track-loss percentage overall. The CU-TCAC model also proves to be quite efficient, which is somewhat against the expectations for a second-order formulation of the target's dynamics. Still in comparison with CU-LAN, the CU-TCVE model demonstrates a significantly decreased performance, despite the fact that both models are defined in a very similar fashion (see Sections 3.1.2 and 3.1.4) and have almost identical parameter settings. Once again, however, these observations must be put in the perspective of a potentially sub-optimal setting of the model parameters.

5.4 Performance vs. Reverberation Time

In a manner similar to Section 5.3, this section presents the performance results obtained for the models of interest for two different values of reverberation time. The room setup described in Section 5.1 provides an environment with approximate reverberation time $T_{60} \approx 0.27$s. By removing some of the acoustic foam panels in this original setup, a second reverberant environment was considered with a level of reverberation $T_{60} \approx 0.34$s (24kHz average). A total of five different audio samples were recorded in this environment, for different speaker trajectories and source signals. Table 2 shows the performance results achieved by the considered models for these two different settings, in terms of the average RMSE and TLP parameters. Each of the values reported in Table 2 represents an average over 60 runs of the algorithms for each of the five samples of audio data.

These results lead to the same observations with respect to T_{60} as for the results of Figure 6 with respect to the SNR level. In essence, a distinct improvement in the tracking accuracy can be observed for

		SNR = 5dB		SNR = 10dB	
		$T_{60} \approx 0.27$s	$T_{60} \approx 0.34$s	$T_{60} \approx 0.27$s	$T_{60} \approx 0.34$s
CU-LAN-ORIG	error $\bar{\varepsilon}$	0.127	0.232	0.094	0.140
	TLP $\bar{\zeta}$	11.2	88.7	0.0	58.0
CU-LAN	error $\bar{\varepsilon}$	0.063	0.104	0.040	0.070
	TLP $\bar{\zeta}$	11.2	38.7	1.7	22.0
CU-RWVE	error $\bar{\varepsilon}$	0.082	0.199	0.047	0.098
	TLP $\bar{\zeta}$	18.3	62.3	8.3	36.7
CU-TCVE	error $\bar{\varepsilon}$	0.099	0.284	0.054	0.180
	TLP $\bar{\zeta}$	34.6	69.0	9.2	50.0
CU-TCAC	error $\bar{\varepsilon}$	0.070	0.128	0.042	0.073
	TLP $\bar{\zeta}$	12.9	46.0	2.1	23.3
CL-RWVE-RAAC	error $\bar{\varepsilon}$	0.118	0.359	0.063	0.233
	TLP $\bar{\zeta}$	40.8	82.7	27.1	61.7

		SNR = 15dB		SNR = 20dB	
		$T_{60} \approx 0.27$s	$T_{60} \approx 0.34$s	$T_{60} \approx 0.27$s	$T_{60} \approx 0.34$s
CU-LAN-ORIG	error $\bar{\varepsilon}$	0.077	0.113	0.068	0.100
	TLP $\bar{\zeta}$	0.0	40.3	4.6	33.3
CU-LAN	error $\bar{\varepsilon}$	0.029	0.052	0.027	0.045
	TLP $\bar{\zeta}$	0.4	19.0	0.0	19.3
CU-RWVE	error $\bar{\varepsilon}$	0.034	0.073	0.033	0.059
	TLP $\bar{\zeta}$	2.5	22.0	1.2	19.7
CU-TCVE	error $\bar{\varepsilon}$	0.040	0.090	0.036	0.070
	TLP $\bar{\zeta}$	7.5	31.0	5.8	22.0
CU-TCAC	error $\bar{\varepsilon}$	0.031	0.060	0.028	0.049
	TLP $\bar{\zeta}$	0.8	14.3	0.8	11.3
CL-RWVE-RAAC	error $\bar{\varepsilon}$	0.048	0.179	0.039	0.106
	TLP $\bar{\zeta}$	15.0	52.7	12.5	43.7

Table 2: Tracking performance results versus reverberation time: average estimation error $\bar{\varepsilon}$ (m) and track loss percentage $\bar{\zeta}$ (%).

CU-LAN when compared to the non-optimised version of this model, i.e., CU-LAN-ORIG. And among the considered models, the results here also show that the CU-TCAC model (with its specific parameter setting!) achieves a level of performance versus T_{60} comparable to CU-LAN.

6 Conclusions and Future Research

The work presented in this report provides a preliminary study on the use of various dynamics models in the frame of acoustic source tracking. Several types of models were considered and simulated under various levels of noise and reverberation. It was shown that the dynamics model implemented in the tracking algorithm can play a potentially significant part in achieving superior tracking results. The choice of a relevant model, together with the proper optimisation of its parameters, can lead to a more targeted tracking of the speaker, especially during periods of speech inactivity. Rather than simply freezing its estimate, the tracking algorithm can be made to "blindly" track the speaker when no measurements are avalailable; this ultimately leads to a decreased chance of track loss when the speech resumes, and consequently, an improved robustness against noise and reverberation, and potentially also against the influence of other sound sources in the case of multi-target tracking.

On the basis of a coarse optimisation of the model parameters, some important conclusions can be

drawn from the experimental simulation results obtained in this work:

1) both the coordinate-uncoupled as well as the curvilinear model types have the potential to provide satisfactory tracking results, and should hence be considered as possible candidates for acoustic source tracking;

2) as far as the target's dynamics are concerned, the successful implementation of a tracking algorithm does not solely rely on choosing the correct model type and order; the process of optimising the model parameters also plays a crucial part in the accuracy of the resulting algorithm. As shown with the CU-LAN and CU-LAN-ORIG examples, a given model can be made to work more or less accurately depending on the specific setting of its free parameters.

3) a comparison between CU-LAN and CU-TCVE shows that two seemingly very similar models (with similar parameter settings) can potentially lead to significant differences in the tracking performance results. This suggests that many different dynamics models should be investigated with respect to a minimisation of the algorithm's tracking accuracy;

4) the relatively good performance results obtained with CU-TCAC, a second-order model incorporating the target's acceleration, somewhat refutes the statement in [6, p.202] that the use of an acceleration variable is only of value when a velocity measurement is available;

As one of the main outcomes from this study, the process of optimising model parameters appears as a task of significant importance in the overall tracker design. The crucial issue here is to obtain a set of parameter values achieving a robust tracking performance on one hand, while being able to deal successfully with a wide range of target motions on the other; a tradeoff might have to be found between these two factors in practice. In an attempt to deal with this specific issue, ongoing research will require the development of two main components:

1) a trajectory simulator, able to generate a range of typical speaker motions, which can be used for the purpose of creating a potentially large amount of input data for the optimisation and simulation of the various dynamics models;

2) a suitable technique for the joint optimisation of model parameters on the basis of typical input data.

The availability of such blocks will allow for a rigorous and efficient assessment of several dynamics models in the design of an algorithm for acoustic source tracking.

Acknowledgments

This work was supported by National ICT Australia (NICTA). NICTA is funded through the Australian Government's Backing Australia's Ability initiative, in part through the Australian Research Council.

References

[1] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188, February 2002.

[2] N. J. Gordon, D. J. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings on Radar and Signal Processing*, 140(2):107–113, April 1993.

[3] J. Vermaak and A. Blake. Nonlinear filtering for speaker tracking in noisy and reverberant environments. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 5, pages 3021–3024, Salt Lake City, UT, USA, may 2001.

[4] E. A. Lehmann and A. M. Johansson. Particle filter with integrated voice activity detection for acoustic source tracking. *EURASIP Journal on Advances in Signal Processing*, 2007. Article ID 50870, 11 pages.

[5] D. Ward and R. Williamson. Particle filter beamforming for acoustic source localization in a reverberant environment. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 1777–1780, Orlando, FL, USA, May 2002.

[6] S. Blackman and R. Popoli. *Design and analysis of modern tracking systems*. Artech House, Boston, 1999.

[7] X. R. Li and V. P. Jilkov. Survey of maneuvering target tracking. Part I: dynamic models. *IEEE Transactions on Aerospace and Electronic Systems*, 39(4):1333–1364, October 2003.

[8] E. A. Lehmann and A. M. Johansson. Experimental performance assessment of a particle filter with voice activity data fusion for acoustic speaker tracking. In *Proceedings of the IEEE Nordic Signal Processing Symposium*, pages 126–129, Reykjavik, Iceland, June 2006.

[9] S. McGinnity and G. Irwin. Multiple model bootstrap filter for maneuvering target tracking. *IEEE Transactions on Aerospace and Electronic Systems*, 36(3):1006–1012, July 2000.

[10] Y. Boers and J. Driessen. Interacting multiple model particle filter. *IEE Proceedings on Radar, Sonar and Navigation*, 150(5):344–349, October 2003.