

Single-Sensor Localization of Moving Sources Using Diffusion Kernels

Eran Zeitouni

Single-Sensor Localization of Moving Sources Using Diffusion Kernels

Research Thesis

In Partial Fulfillment of the Requirements for
the Degree of Master of Science in Electrical Engineering

Eran Zeitouni

Submitted to the Senate of the Technion - Israel Institute of Technology

Heshvan 5782

Haifa

November 2021

This research was done under the supervision of Prof. Israel Cohen, in the Andrew and Erna Viterbi Faculty of Electrical & Computer Engineering.

Acknowledgments

I would like to express my gratitude and appreciation to Professor Israel Cohen for the opportunity, his supervision, support and patience throughout the stages of this research.

I would like to thank Professor Ronen Talmon and Professor Yosi Keller for their review of my research.

Last but not least, I would like to thank my family and my friends for their constant love, faith, support and encouragement. This accomplishment would not have been possible without them.

The generous financial help of the Technion, ISF-NSFC joint research program (grant No. 2514/17), and the Pazy Research Foundation, is gratefully acknowledged.

Publications

The research thesis is based on the following publications:

1. E. Zeitouni and I. Cohen, “Single-sensor source localization of moving sources using diffusion kernels”, Submitted to Applied Acoustics, 2021.
2. E. Zeitouni and I. Cohen, “Single-sensor source localization of moving sources using diffusion kernels and Brownian motion model”, Submitted to IEEE International Conference on Communications, ICC-2022, Seoul, South Korea, May 16-22, 2022.

Contents

1	Introduction	29
1.1	Background and Motivation	29
1.2	Main Contribution	33
1.3	Research Overview	34
1.4	Organization	36
2	Scientific Background	39
2.1	Diffusion Maps Method	39
3	Localization of Moving Acoustic Sources	43
3.1	Introduction	43
3.2	Problem Formulation	44
3.3	Diffusion Kernel	46
3.3.1	From Observations to Feature Vectors	46
3.3.2	Manifold Structure and the Choice of an Affinity Measure	48
3.3.3	Diffusion Kernel Computation	49
3.4	Localization Based on Diffusion Mapping	50
3.4.1	Manifold Parameterization	50
3.4.2	Extension for New Observations	51
3.4.3	Recovery of the Controlling Independent Parameters	52
3.4.4	Accuracy Measure	54
3.5	Conclusion	54
4	Experimental Results	57
4.1	Introduction	57
4.2	Simulated Setup	58
4.3	Stationary Sources	58
4.3.1	One Dimensional Subcase	58
4.3.2	Two Dimensional Subcase	62
4.4	Deterministically Moving Sources	65
4.4.1	Sensitivity to Training Set Size	65

Contents (Continued)

4.4.2	Sensitivity to Number of Features	66
4.4.3	Sensitivity to Speed	68
4.4.4	Sensitivity to Direction	69
4.4.5	Sensitivity to Signal to Noise Ratio	72
4.4.6	Sensitivity to Reverberation Time	74
4.4.7	Sensitivity to Environmental Conditions Changes	75
4.4.8	Sensitivity to Frame Length	76
4.5	Brownian Motion Induced Moving Sources	80
4.6	Conclusions	86
5	Conclusions	91
5.1	Research Summary	91
5.2	Future Research	92
	Appendix A Single-Sensor Tracking Framework	95
A.1	Introduction	95
A.2	Problem Formulation	96
A.3	Manifold-Based Gaussian Process	97
A.3.1	Feature Vectors Extraction	97
A.3.2	Mapping Feature Vectors to Source Locations	98
A.4	Manifold-Based State-Space Model	99
A.4.1	Propagation (State) Model	99
A.4.2	Observation (Measurement) Model	100
A.4.3	Combined State-Space Representation	101
A.5	Tracking Scheme	102
A.6	Learning Hyperparameters	102
A.7	Summary	103
	Bibliography	105

List of Figures

1.1	Running fix technique for single-sensor source localization- a chronological order illustration.	32
2.1	Embedding of the observations as a function of the azimuth angle-previous single-sensor localization algorithm	42
4.1	Room setup of the stationary one-dimensional subcase	59
4.2	Comparison of distance measures, according to the azimuth angle, and subsequent embedding of observations.	62
4.3	Room setup of the stationary two-dimensional subcase	63
4.4	Diffusion mapping of the observations of the entire dataset into the embedded manifold	64
4.5	Polar plot of 35 randomly selected sources out of the test set	64
4.6	Room setup and the evolution of the locations of the sources through time of the training set size experiment	66
4.7	Performance of the proposed algorithm as a function of the training set size	67
4.8	Performance of the proposed algorithm as a function of the length of the feature vector	68
4.9	Performance of the proposed algorithm as a function of the speed of the sources	69
4.10	Comparison of diffusion mappings of the observations, according to the direction, between different speed scenarios.	70
4.11	Room setup and the midway point of the trajectories of the sources, with respect to all scenarios of the direction experiment	70
4.12	Performance of the proposed algorithm as a function of the direction of movement of the sources	72
4.13	Presence of spatial aliasing.	73
4.14	Performance of the proposed algorithm as a function of SNR	74
4.15	Performance of the proposed algorithm as a function of reverberation time and environmental conditions changes	76

List of Figures (Continued)

4.16	Room setup and the evolution of the locations of the sources through time for the experiment of the frames length	78
4.17	Performance of the proposed algorithm as a function of the frames length	79
4.18	Localization results of a moving source through time of each of the unknown parameters.	79
4.19	Brownian motion induced source trajectory example, for a coefficient of 0.8.	81
4.20	Performance of the proposed algorithm as a function of Brownian motion coefficient, for various intrinsic dimension cases	83
4.21	Embedding of the observations as a function of location and velocity- Brownian motion coefficient of 0.8	84
4.22	Localization results of a moving source through time of each of the unknown parameters- Brownian motion coefficient of 0.8. . .	86

List of Tables

4.1 One-Dimensional Stationary Case- Comparison	61
---	----

Abstract

Source localization is a common problem in various fields and has applications in both military and civil sectors. Localization of acoustic sources generally requires a few microphones, but it is also possible to use a single microphone and data that was prerecorded in the same environment. Unfortunately, existing single microphone localization methods are restricted to acoustic sources that have a fixed location.

In this research, we introduce a supervised method for estimating both the location and velocity of a moving acoustic source, using a single microphone, based on a manifold learning approach. Each source, one at a time, transmits a white Gaussian noise signal during its movement, which is received by the sensor. The received signal is divided into time frames. The trajectory, formed by the movement of the source during the frame, is approximated by a linear movement segment. Each frame is inspected individually for estimation of the location and velocity. The signal, received during the frame, is processed to extract a feature vector. In such a manner, we collect a training set of observations, generated from different sources. Using diffusion maps with a Euclidean distance-based diffusion kernel, we learn the nonlinear structure of the manifold of the data. The data is organized on the manifold according to the location and velocity values of the sources. The unknown location and velocity of a source can be recovered according to its observation's nearest training neighbors on the manifold. The recovery of the location and velocity is determined by the midway point of the segment. Based on that point, we minimize the approximation error of the true trajectory by the linear segment. The performance of the proposed method is examined by various simulations investigating the sensitivity to different hyperparameters (e.g., frames length), variables (e.g., speed and direction), and conditions (e.g., reverberation time).

Research findings indicate that localization of very slow sources results in good accuracy of the estimated location, at the expense of relatively low accuracy of the estimated direction. Localization of faster sources leads to an improvement of the accuracy of the estimated direction due to a more mean-

ingful movement of the sources, at the expense of the accuracy of the estimated location. The accuracy of the estimated direction starts deteriorating for fast sources, as the variations in their movement are too fast to be distinguished—resulting in a sparse manifold. The approximation by a linear segment is inaccurate for long frames, unless the source has constant velocity during the frame. Short frames are not good either, regardless the trajectory, as they are unable to capture the movement properly. The algorithm performs well in reverberant and noisy environments, yet is sensitive to environmental conditions changes. The results validate the necessity of reflections for yielding an accurate estimation. While the algorithm is designated for recovering the location and velocity of slow sources that change direction and speed gradually, it is even capable of successfully estimating the location and the average velocity of slow sources that change their velocity rapidly and randomly.

Notations

$ \cdot $	Absolute value
$\ \cdot\ $	Euclidean norm
\otimes	Kronecker product
$\mathbf{0}$	Vector of zeros
$\mathbf{1}$	Vector of ones
a	Coordinate/axis (i.e., \tilde{x} , \tilde{y} , or \tilde{z})
$a(\cdot)$	Deterministic function that maps a location to a velocity
$a(\mathbf{p}(t))$	Drift term
\mathbf{A}	Mahalanobis distance-based affinity matrix between all observations and the training observations
$\tilde{\mathbf{A}}$	Normalized affinity matrix
$\mathbf{b}(\cdot)$	Transition function
$\mathbf{B}(\mathcal{C}_{q,q-1})$	Transition matrix between subsequent time frame indices q and $(q-1)$
$\mathbf{B}_{3,q}$	Shortened notation of $\mathbf{B}_3(\mathcal{C}_{q,q-1})$
$\mathbf{B}_3(\mathcal{C}_{q,q-1})$	Extended version of the matrix $\mathbf{B}(\mathcal{C}_{q,q-1})$
$c_{\mathbf{x}_i}(j-l)$	Time-invariant autocorrelation function of $x_i(n)$, using two distinct discrete time indices
$c_{\mathbf{x}_i}(\tau)$	Time-invariant autocorrelation function of $x_i(n)$, using a single discrete time index (representing time difference)
$c_{\mathbf{y}_i}(n_1, n_2)$	Time-variant autocorrelation function of $y_i(n)$
$c_{\mathbf{y}_i}(\tau)$	Time-invariant autocorrelation function of $y_i(n)$

$c(\cdot)$	Nonlinear mapping function from the parametric space to the observable space
\mathbf{c}_i	Feature vector corresponding to an observation associated with the i th source (either labeled or unlabeled), in Chapter 2
\mathbf{c}_i	Feature vector corresponding to an observation associated with $\boldsymbol{\theta}_i(q)$, in Chapter 3
\mathbf{c}_i	Feature vector corresponding to an observation associated with \mathbf{p}_i , in Appendix A
\mathbf{c}_{i_j}	Feature vector corresponding to an observation associated with $\boldsymbol{\theta}_{i_j}$
$\mathbf{c}_i(q)$	First D elements of the autocorrelation function of the q th time frame
$\mathbf{c}_i^{(j)}$	j th entry of \mathbf{c}_i (in all chapters)
$\bar{\mathbf{c}}_i$	Feature vector corresponding to an observation associated with $\bar{\boldsymbol{\theta}}_i$, in Chapter 2
$\bar{\mathbf{c}}_i$	Feature vector corresponding to an observation associated with $\bar{\boldsymbol{\theta}}_i(q)$, in Chapter 3
$\bar{\mathbf{c}}_i$	Feature vector corresponding to an observation associated with $\bar{\mathbf{p}}_i$, in Appendix A
$\bar{\mathbf{c}}_{q_i}$	Training observation which is close to the observation of the moving source $\tilde{\mathbf{c}}(q)$
$\tilde{\mathbf{c}}(q)$	Feature vector of the moving source with respect to the unknown location at the q th time frame
$\tilde{\mathbf{c}}_i(q)$	Smoothed autocorrelation function of the i th source at the q th time frame
\mathcal{C}_q	Extended set which includes $\tilde{\mathbf{c}}(q)$ and the set of its \hat{k} -nearest training neighbors (i.e., $\bar{\mathcal{C}}_q$)
$\mathcal{C}_{q,q-1}$	Union of \mathcal{C}_q and \mathcal{C}_{q-1}
$\bar{\mathcal{C}}$	Set of the training feature vectors with respect to the known fixed locations
$\bar{\mathcal{C}}_q$	Set of the \hat{k} -nearest training neighbors of $\tilde{\mathbf{c}}(q)$
d	Intrinsic dimension

D	Length of the feature vector
\mathbf{D}	Normalization matrix, in Chapter 2
\mathbf{D}	Diagonal matrix, in Chapter 3
$D_{\text{DIFF}}(\mathbf{c}_i, \mathbf{c}_j)$	Diffusion distance between \mathbf{c}_i and \mathbf{c}_j
$D_{\text{EUC}}(\mathbf{c}_i, \mathbf{c}_j)$	Euclidean distance between \mathbf{c}_i and \mathbf{c}_j
$\mathbf{e}(\mathbf{c}_i)$	Normalized estimation error vector
e_i	Estimation error of the location (scalar)
$e_i^{(j)}$	j th element of $\mathbf{e}(\mathbf{c}_i)$
$\mathbb{E}[\cdot]$	Expected value
f_a	Nonlinear mapping function from the observable space to the parametric space
\mathbf{f}_q	Vector of all three-dimensional mapped locations of the observations in \mathcal{C}_q
$\mathbf{f}_{q,a}$	Vector of all corresponding mappings of the function $f_a(\cdot)$ over the observations in \mathcal{C}_q (a -coordinate only)
$\mathbf{f}_{q,\tilde{x}}$	Vector of all corresponding mappings of the function $f_a(\cdot)$ over the observations in \mathcal{C}_q (\tilde{x} -coordinate only)
$\mathbf{f}_{q,\tilde{y}}$	Vector of all corresponding mappings of the function $f_a(\cdot)$ over the observations in \mathcal{C}_q (\tilde{y} -coordinate only)
$\mathbf{f}_{q,\tilde{z}}$	Vector of all corresponding mappings of the function $f_a(\cdot)$ over the observations in \mathcal{C}_q (\tilde{z} -coordinate only)
$\hat{\mathbf{f}}_{q q}$	Estimated location at time frame index q , given all the observations available until time frame index q
$\hat{\mathbf{f}}_{q q-1}$	Estimated location at time frame index q , given all the observations available until time frame index $q - 1$
$\hat{\mathbf{f}}_{q-1 q-1}$	Estimated location at time frame index $q - 1$, given all the observations available until time frame index $q - 1$
f_s	Sampling frequency
$h_{\boldsymbol{\theta}_i(j)}(n, j)$	Acoustic impulse response at discrete time index n to an impulse transmitted at discrete time index j , between the i th source and the sensor, with respect to $\boldsymbol{\theta}_i(j)$

$h_{\theta_i}(n)$	Acoustic impulse response between the i th source and the sensor, at relative location θ_i
$h_{\theta_i}(\tau)$	Acoustic impulse response, associated with a time-invariant system, between the i th source and the sensor, corresponding to a parameters vector θ_i
$\mathbf{H}_{3,q}$	Extended version of the matrix \mathbf{H}_q
\mathbf{H}_q	Index matrix
H_{θ}	System, corresponding to the AIR, which depends on the parameters vector θ
$H_{\theta}\{\cdot\}$	Operator, applied on a transmitted signal, representing the operation of the system H_{θ}
\tilde{i}	Index for the new rows of the extended matrices and vectors
\mathbf{I}	Identity matrix
\mathcal{I}_q	All related information, available at frame index q
k	Number of nearest-neighbors of $\bar{\mathbf{c}}_j$ in $\bar{\Gamma}$, in the stage of the matrix \mathbf{W} creation
\tilde{k}	Number of nearest embedded training measurements $\{\Psi_d(\bar{\mathbf{c}}_j)\}_{j=1}^{\tilde{k}}$ of $\Psi_d(\mathbf{c}_i)$, according to the diffusion distance
\hat{k}	Number of nearest-neighbors of $\tilde{\mathbf{c}}(q)$ among all training observations
\mathbf{K}_q	Kalman gain at time frame index q
L	Number of small perturbations for each predefined relative location, in Chapter 2
L	Log-likelihood function of the training observations, in Appendix A
\mathbf{L}	Normalized graph-Laplacian matrix
m	Training set size
M	Test set size
\mathcal{M}	Manifold
<i>median</i>	Median value of the Gaussian function's numerator values associated with the non-zeros elements of the affinity matrix
n	Discrete time index

N	Total number of query points along the trajectory of the source (total number of time frames)
$\mathbf{n}(t)$	Brownian motion term
$n_{\tilde{x}}(t)$	Horizontal component of the Brownian motion term $\mathbf{n}(t)$
$n_{\tilde{y}}(t)$	Vertical component of the Brownian motion term $\mathbf{n}(t)$
$\mathcal{N}_{\tilde{z}}$	Set of the k -nearest-neighbors of \mathbf{c}_i in $\bar{\Gamma}$
$\bar{\mathcal{N}}_j$	Set of the k -nearest-neighbors of $\bar{\mathbf{c}}_j$ in $\bar{\Gamma}$
$\tilde{\mathcal{N}}_i$	Set of \tilde{k} -nearest embedded training measurements $\{\Psi_d(\bar{\mathbf{c}}_j)\}_{j=1}^{\tilde{k}}$ of $\Psi_d(\mathbf{c}_i)$, according to the diffusion distance
$\mathbf{p}(q)$	Three-dimensional absolute location at time frame q , in Appendix A
$\mathbf{p}(t)$	Absolute location of the source at time t , in Chapters 3 and 4
$p_a(q)$	a -coordinate of the estimated location at time frame q
\mathbf{p}_i	Three-dimensional absolute location of the i th source in Cartesian representation
$p_{i,\tilde{x}}(j)$	\tilde{x} coordinate of $\mathbf{p}_i(j)$
$p_{i,\tilde{y}}(j)$	\tilde{y} coordinate of $\mathbf{p}_i(j)$
$p_{i,\tilde{z}}(j)$	\tilde{z} coordinate of $\mathbf{p}_i(j)$
$\mathbf{p}_i(j)$	\mathbf{p}_i at discrete time index j
$\bar{\mathbf{p}}_a$	Vector of the a -coordinate of all training locations
$\bar{\mathbf{p}}_i$	Known fixed three-dimensional absolute location of the i th source in Cartesian representation
$\bar{p}_{i,a}$	a -coordinate ($a \in \{\tilde{x}, \tilde{y}, \tilde{z}\}$) of the measured known location $\bar{\mathbf{p}}_i$
$\bar{\mathbf{p}}_L$	Another notation for $\bar{\mathbf{p}}_a$. See $\bar{\mathbf{p}}_a$
$\bar{\mathbf{p}}_q$	Vector of length $3\tilde{k}$ of the three-dimensional measured training locations
$\bar{\mathbf{p}}_{q,a}$	Vector of length \tilde{k} of the a -coordinate of the training locations corresponding to the subset $\bar{\mathcal{C}}_q$

$\mathbf{P}_{q q}$	Covariance matrix of the estimation error at time frame index q , given all the observations available until time frame index q
$\mathbf{P}_{q q-1}$	Covariance matrix of the estimation error at time frame index q , given all the observations available until time frame index $q - 1$
$\mathbf{P}_{q-1 q-1}$	Covariance matrix of the estimation error at time frame index $q - 1$, given all the observations available until time frame index $q - 1$
$\dot{\mathbf{p}}(t)$	Velocity of moving source at time t
\mathbf{P}	Transition matrix
$\Pr(\bar{\mathbf{c}}_j \bar{\mathbf{c}}_i)$	Single-step transition probability from $\bar{\mathbf{c}}_i$ to $\bar{\mathbf{c}}_j$
q	Query point along the trajectory of the source, in Chapters 3 and 4
q	Time frame index, in Appendix A
q_i	Index ranging from 1 to m , for the subset of close training observations
$\mathbf{Q}(\mathcal{C}_{q,q-1})$	Variance of $\boldsymbol{\xi}(\mathcal{C}_{q,q-1})$ (process noise)
$\mathbf{Q}_3(\mathcal{C}_{q,q-1})$	Extended version of the matrix $\mathbf{Q}(\mathcal{C}_{q,q-1})$
$\mathbf{Q}_{3,q}$	Shortened notation of $\mathbf{Q}_3(\mathcal{C}_{q,q-1})$
RMSE	Root mean square error, in Chapter 2
RMSE	Root mean square error (generalized), in Chapters 3 and 4
RMSE	Vector containing the RMSE values of all query point along the trajectory of the source
$\overline{\text{RMSE}}$	Average RMSE along the trajectory of the source
$\mathbf{R}_{3,q}$	Extended version of the matrix \mathbf{R}_q
\mathbf{R}_q	Covariance matrix of \mathbf{u}_q
\mathbf{S}	Diagonal matrix
$s_i(j)$	Speed of the i th source
t	Continuous time value
T_{60}	Reverberation time
u	Measurement noise

\mathbf{u}_q	Measurement noise vector of length \dot{k}
$\mathbf{u}_{3,q}$	Extended version of the vector \mathbf{u}_q
$v_{\tilde{x}}$	Constant horizontal speed component of the i th source
$v_{\tilde{x}_i}(j)$	Horizontal speed component of the i th source at discrete time index j
$v_{\tilde{x}_{max}}$	Maximal horizontal speed drawn in an experiment
$v_{\tilde{y}}$	Constant vertical speed component of the i th source
$v_{\tilde{y}_i}(j)$	Vertical speed component of the i th source at discrete time index j
$v_{\tilde{y}_{max}}$	Maximal vertical speed drawn in an experiment
\mathbf{W}	Mahalanobis distance-based affinity matrix of size $m \times m$, in Chapter 2
\mathbf{W}	Euclidean distance-based affinity matrix of size either $m \times m$ (original matrix) or $(M + m) \times m$ (extended matrix), in Chapters 3 and 4
$x_i(n)$	Signal transmitted by the i th source
$\tilde{x}_i(j)$	Horizontal coordinate of the absolute location of the i th source at discrete time index j
\mathbf{y}_i	Signal received by the sensor, corresponding to the signal transmitted from the i th source, using a vector representation
$y_i(n)$	Signal received by the sensor, corresponding to $x_i(n)$
$\tilde{y}_i(j)$	Vertical coordinate of the absolute location of the i th source at discrete time index j
α_j	Coefficient that represents the significance of the individual estimation error corresponding to the j th physical quantity, with respect to the total estimation error
$\beta_i(j)$	Direction of movement of the i th source at discrete time index j
γ	Smoothing parameter
$\gamma_j(\mathbf{c}_i)$	j th interpolation coefficient of the i th unlabeled observation
Γ	Set of the feature vectors with respect to the unlabeled parameters in Θ

Γ_{LL}	Covariance matrix of $\bar{\mathbf{p}}_L$
$\bar{\Gamma}$	Set of the feature vectors with respect to the labeled parameters in $\bar{\Theta}$
$\delta(\tau)$	Delta function
$\tilde{\delta}$	Neighborhood radius
ε	Scale parameter of the affinity matrices \mathbf{W} and \mathbf{A} , in Chapter 2
ε	Scale parameter of the affinity matrix \mathbf{W} , in Chapter 3
ε	Scale parameter of the Gaussian kernel κ , in Appendix A
ε_{γ_i}	Scale parameter of all interpolation coefficients of \mathbf{c}_i
η	Brownian motion coefficient
Θ	Set of M unknown source locations, in Chapter 2
Θ	Set of M arbitrary unknown source locations and velocities, in Chapter 3
$\bar{\Theta}$	Set of m predefined source locations, in Chapter 2
$\bar{\Theta}$	Set of arbitrarily m known locations and velocities of the sources, in Chapter 3
θ_i	Relative location of the i th source- unlabeled, in Chapter 2
θ_i	Short representation of all location and velocity values of the i th source, varying during its movement, post-implementation of the assumptions, in Chapter 3
θ_{i_j}	j th slightly perturbed location of $\bar{\theta}_i$
$\theta_i(j)$	Location and velocity of the i th source at discrete time index j
$\theta_i(q)$	Location and velocity of the i th source at the query point
$\bar{\theta}_i$	Relative location of the i th source- labeled
$\bar{\theta}_j(q)$	Location and velocity of the i th source at the query point- labeled
$\hat{\theta}_i$	Estimated location of the i th source, in Chapter 2
$\hat{\theta}_i(q)$	Estimated location and velocity of the i th source at the query point, in Chapter 3
$\tilde{\theta}$	Set of the hyperparameters of the tracking model

κ	Gaussian kernel
$\tilde{\kappa}$	Covariance function
λ_j	j th eigenvalue of the Mahalanobis distance-based affinity matrix \mathbf{W} , in Chapter 2
λ_j	j th eigenvalue of the transition matrix \mathbf{P} , in Chapter 3
$\rho_i(j)$	Distance between the i th source and the sensor at discrete time index j
$\boldsymbol{\mu}_i$	Empirical mean of the local observations associated with $\bar{\mathbf{c}}_i$
ξ	Confidence factor
$\boldsymbol{\xi}(\mathcal{C}_{q,q-1})$	Process noise of the propagation equation
$\boldsymbol{\xi}_3(\mathcal{C}_{q,q-1})$	Extended version of the vector $\boldsymbol{\xi}(\mathcal{C}_{q,q-1})$
σ_u^2	Variance of the measurement noise
$\sigma_{\mathbf{x}_i}^2$	Variance of the signal transmitted by the i th source
σ_x^2	Horizontal component of the variance vector of the Brownian motion term
σ_y^2	Vertical component of the variance vector of the Brownian motion term
$\boldsymbol{\Sigma}_i$	Local covariance matrix of the i th observation from the training set
$\bar{\boldsymbol{\Sigma}}_{LL}$	$m \times m$ covariance matrix defined over the mapping function values at the labeled dataset $\bar{\mathcal{C}}$
$\bar{\boldsymbol{\Sigma}}_{Lq}$	$m \times 1$ covariance vector between the mapping function values at $\bar{\mathcal{C}}$ and $f_a(\tilde{\mathbf{c}}(q))$
$\bar{\boldsymbol{\Sigma}}_{qq}$	variance (a scalar) of $f_a(\tilde{\mathbf{c}}(q))$
$\tilde{\boldsymbol{\Sigma}}_{q,g}$	Covariance matrix in the size of $(\hat{k} + 1) \times (\hat{k} + 1)$, which each of its entries is computed by the covariance function
$\hat{\boldsymbol{\Sigma}}_i$	Empirical local covariance matrix of the i th observation from the training set
τ	Difference of discrete time indices

$\phi_i(j)$	Azimuth angle between the i th source and the sensor at discrete time index j
φ_j	j th original eigenvector of length m of the affinity matrix \mathbf{W}
Φ_d	Map of the training observations to their corresponding representation of independent controlling parameters in the embedded manifold
ψ_j	j th extended eigenvector (left singular vector of $\tilde{\mathbf{A}}$) of length $M+m$, in Chapter 2
ψ_j	j th eigenvector of the transition matrix \mathbf{P} - either of length m (original) or $m+M$ (extended), in Chapters 3 and 4
Ψ_d	Map of the observations to their corresponding representation of independent controlling parameters in the embedded manifold
$\Psi_d(\mathbf{c}_i)$	Vector of d elements, representing the mapping of the i th feature vector to the embedded manifold

Abbreviations

AIR	Acoustic Impulse Response
AWGN	Additive White Gaussian Noise
DF	Direction Finding
DOA	Direction of Arrival
GP	Gaussian Process
HF	High Frequency
IID	Independent and Identically Distributed
ML	Maximum Likelihood
MUSIC	Multiple Signal Classification
OOSE	Out of Sample Extension
PSD	Power Spectral Density
RF	Radio Frequency
RMSE	Root Mean Square Error
SLAM	Simultaneous Localization and Mapping
SNR	Signal to Noise Ratio
SRP-PHAT	Steered Response Phase Transformation
TDOA	Time Difference of Arrival
WGN	White Gaussian Noise
WSS	Wide Sense Stationary

Chapter 1

Introduction

1.1 Background and Motivation

Source localization problem has increasingly drawn attention since Hertz's discovery in the late 19th century regarding the directivity of antennas. Inspired by the entrance of digital technology, which was a major modern milestone in the development of the field, extensive research efforts have been dedicated for the last several decades. Source localization applications are tremendously broad and vary generally by the type of source's signal to be targeted (e.g., radio frequency or acoustic) and by the goals of the interested parties, such as:

- Government (e.g., spectrum enforcement of non-authorized transmitters),
- Military intelligence (e.g., location detection of hostile forces),
- Law enforcement and security services (e.g., reconnaissance of criminal organizations' radio-transmissions and tracking people at risk of wandering),
- Research and science (e.g., radio-astronomy and wildlife tracking),
- Commercially (e.g., providing better speech quality by acoustic source localization).

Solutions to the source localization problem can be divided into two groups: active sensing and passive sensing. Classic and modern passive source localization methodologies mostly rely on exploiting variations of a single physical attribute between the received signals, which were originated from a particular source, such as: amplitude (e.g., Watson-Watt), frequency (e.g., Doppler), phase (e.g., Correlative Interferometer) or time (e.g., Time Difference of Arrival- TDOA).

Except the latter, utilization of these variations by a sensor outcomes in determination of the direction toward the signal source. This process is commonly known as direction finding (DF), where the determined direction toward the signal source is called either bearing or direction of arrival (DOA). Consequently, source location is determined as the point of intersection of bearings produced by a spatial array of sensors which form triangulation. Theoretically, the point of intersection (i.e., a fix) is determined by at least 3 bearings, but in practice, under sensors' formations which compatible to reasonable operational scenarios, 2 bearings are mostly sufficient. On the other hand, utilization of the acquisition time difference between the sensors results in hyperbola, where in order to determine the location of the source, at least three hyperbolas are needed for an unambiguous point of intersection. Thus, source localization by these passive methodologies is constrained to multiple sensors: either two sensors (Each composed of a spatial array of multiple inner elements, such as antennas or microphones) for all cases except TDOA, or three sensors for TDOA (Where each sensor involves a single element).

In recent years, advanced spatial array processing methods and algorithms for localization have been introduced, such as maximum likelihood (ML) based beamforming and subspace based methods. The former relies on optimizing the output power of the beamformer, according to a statistical model of the received signals. A popular algorithm which follows this approach is the steered response phase transformation (SRP-PHAT) [1, 2]. On the other hand, the subspace based approaches, such as the well-known multiple signal classification (MUSIC) algorithm [3], successfully tackle ML based algorithms fundamental flaws, alongside yielding high resolution results. These benefits of the subspace based methods are obtained even in the presence of noise, but they come at the price of high computational and storage resources. Common to all these algorithms is the requirement of a spatial array.

In contrast to the mostly well-posed source localization problem of a pair of sensors (where each is composed of a spatial array), under the constraint of a single sensor (e.g., a car speakerphone, operational scenarios where only a single sensor is available) this problem is mostly ill-posed/underdetermined using conventional localization methods. The reason for that is that system's transfer function (infamous multipath included) is mostly unknown- where in case it is known, alongside with the statistical model of the transmitted signal, the location can be recovered by ML based beamformer.

Another approach [4–6], [7, Ch. 7], [8, Ch. 7] for source localization using a single sensor (composed of a spatial array of multiple antennas) is restricted to the high frequency (HF) range, due to the modes of propagation of the signals in this range. The main propagation component in the HF range, skywave, is

reflected back from the ionosphere towards Earth surface. Thus, in addition to measuring the DOA, by recovering the elevation angle, a triangulation is achieved and the location can be estimated. However, in order to do so, the virtual height of the reflecting ionosphere layer must be known and since it depends in many variables (e.g., weather, season, time and solar activity), it is mostly impractical.

In addition to the previously mentioned single-sensor source localization methods, a useful technique named *running fix* [8, Ch. 7], can be applied for a stationary (or relatively slow) source, using a moving sensor, if the source is active long enough, as illustrated in Fig. 1.1. Note that the uncertainty regarding source's stationarity can be resolved by producing a third bearing from another location.

Talmon *et al.* [9] have introduced a supervised *manifold learning* based method, which was implemented explicitly by *diffusion kernels*. Instead of fitting the identified system to a predefined model as conventional system identification methods, this data-driven method focuses on revealing the underlying fundamental controlling parameters of the system based on a training set of measurements of signals from various known locations, such that no knowledge regarding system's transfer function is necessary. A diffusion kernel with a specially-tailored distance measure, which combines local estimates of covariance matrices of the measurements with global processing by spectral decomposition, allows parameterization of the measurements into a low-dimensional space. This low-dimensional space is also known as the *manifold*. Thus, since the location of the source is the only changing degree-of-freedom of the system (by assumption), the unknown location of each source can be estimated using this method, as its location is in fact represented by the recovered parameters. This method was later applied by Talmon *et al.* [10] for source localization of stationary (i.e., fixed location) acoustic sources by a sole microphone (i.e., single-sensor of single-element), in a small reverberant room, where their locations vary only by the azimuth angle. This method was further investigated in a series of works by Laufer-Goldshtein *et al.* for source localization of stationary acoustic sources by a pair of sensors [11–14], and moving sources using multiple pairs of sensors [15–17]. All of Laufer-Goldshtein's works were recently concluded in [18].

Another subspace based approach with a similar concept, called *fingerprinting*, is common in localization of RF emitters in highly scattering environments. As its name implies, this technique assumes a unique relation between the emitter location and the characteristics of the signals intercepted by the sensor (mostly multipath propagation). Thus, by extracting a fingerprint from the received signals and composing a database of fingerprints in the desired area

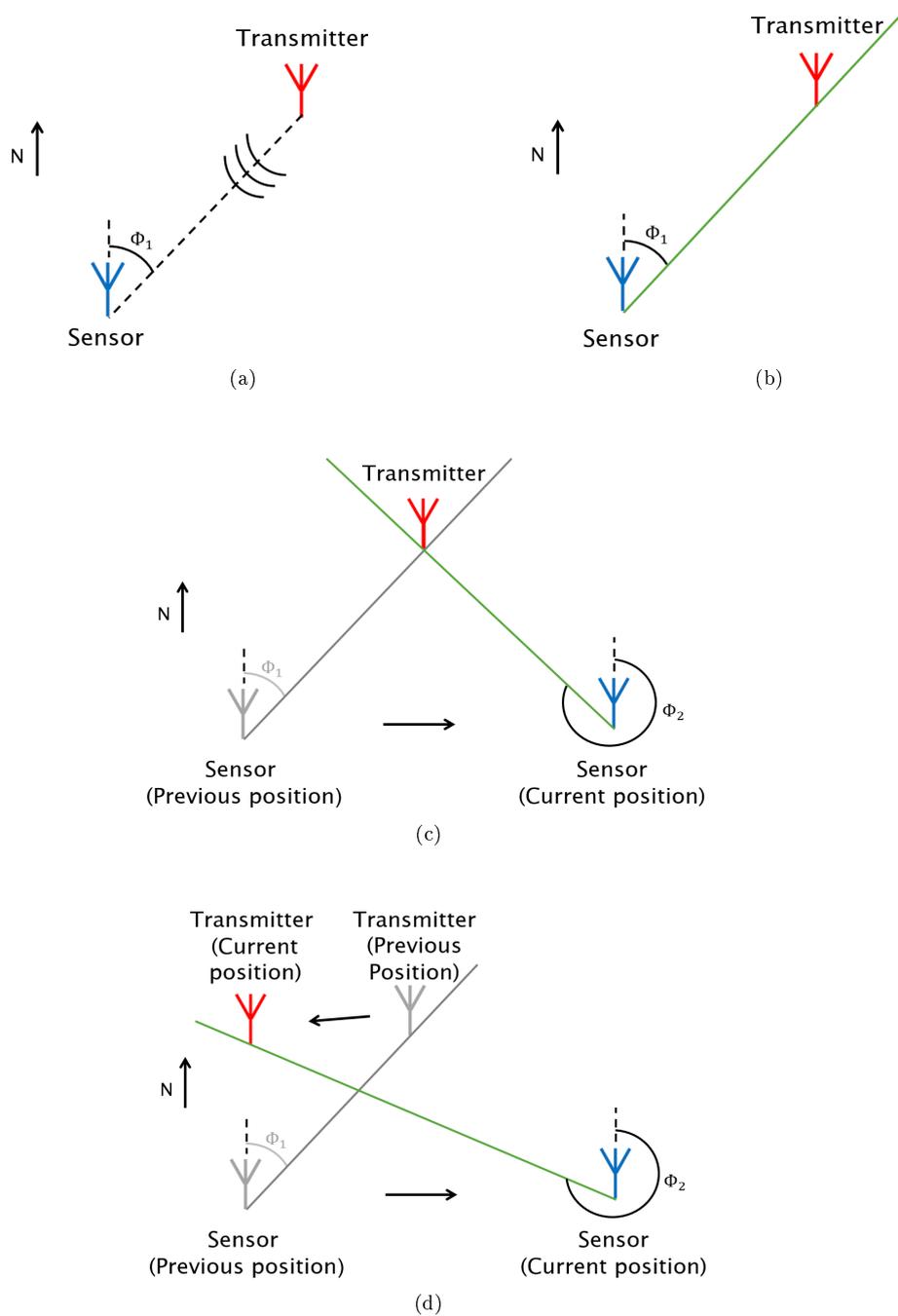


Figure 1.1: Running fix technique for single-sensor source localization- a chronological order illustration. (a) A signal is transmitted by the source. (b) The signal is intercepted by the DF sensor, which consequently produces a bearing of Φ_1 . Subsequently, in order to yield a fix, the location of the sensor is changed for producing another bearing, of Φ_2 with respect to the transmitting source, leading to two possible scenarios: (c) either a correct fix due to a stationary source, (d) or a false fix in case the source isn't stationary.

in advance, the location can be recovered by matching the fingerprint in hand to the fingerprints in the database. Several works have focused on localization using a single sensor [19–26] in recent years- all composed of a spatial array of multiple antennas. However, it is worth mentioning that one work, by Kuperstein *et al.* [19] has tried to determine the location using a single element as well- mostly with no success, in addition to significant performance degradation compared to multiple elements.

A different single-sensor localization approach determines the location using aids. Apart of the spatial array of multiple antennas composing the sensor, which provides a DOA according to the direct propagation path of the signal transmitted by the emitter, this technique exploits either a known dominant scatterer [27] or a set of known transponders [28, 29] available at the scene, in order to produce a fix.

In this research the paradigm of [10] is adopted and adapted to a more realistic setting, which corresponds to scenarios of moving acoustic sources in small reverberant environment, while the sensor remains stationary (for simplicity). The major differences between the algorithms are detailed in Section 1.2.

As scenarios where only a single-sensor is available are common, and the necessity for localization despite the lack of additional sensors is always relevant, is passive localization of moving source using only a single-sensor feasible? If so, is it feasible even for a more challenging setup involving a single-sensor of a single element as well? In case the location of the moving source can be estimated, is it possible to estimate its velocity as well? In case the answers to all previous questions are positive, what are the capabilities, limitations and tradeoffs? What will be the consequences when our assumptions are revoked? Can we improve the framework of previous works? In this thesis, we try to address these questions.

For clarity, we note that in the rest of this document, we refer to a sensor and a microphone interchangeably.

1.2 Main Contribution

In this research, as mentioned in Section 1.1, we follow and modify the paradigm of recovering a one-dimensional location of stationary sources by Talmon *et al.* [10], which is established upon a Mahalanobis distance-based diffusion kernel and its setting involves stationary sources and sensor. We seek to investigate a more realistic setting that corresponds to scenarios of moving acoustic sources in small reverberant environment, while the sensor remains stationary (for simplicity). Thus, the new setting has led to three major changes in the algorithm of [10], as follows.

1. In case of scenarios of sources that have a fixed location, an additional varying degree-of-freedom of the system is assigned for the distance of the source from the sensor, which assembles the location of the source along with the existing azimuth degree-of-freedom.

On the other hand, in case of scenarios of moving sources (deterministically and induced by Brownian motion), the sole degree-of-freedom is allocated for one of velocity's parameters: either speed or direction of movement. This velocity parameter dictates the location parameters of the sources throughout their movement.

2. Due to the movement of the sources, the system is time variant, by definition, which prevents the use of familiar and convenient convolution-based relations.
3. The Mahalanobis distance-based diffusion kernel is replaced by a customized Euclidean distance-based diffusion kernel, inspired by the study of the acoustic manifold by Laufer-Goldsthtein *et al.* [12] and its successful application in their following paper [13]- both focusing on multiple-sensors. However, the extension of the diffusion framework for the test set and its corresponding diffusion kernel in particular, do not follow neither [10] nor [12, 13].

1.3 Research Overview

The presented research addresses passive localization of moving acoustic sources by a single microphone (Sources that their position is fixed are of secondary interest). The primary goal is developing a method for an accurate estimation of the location and velocity of an acoustic source during its movement, despite the uncertainty inherent to localization by a single-sensor, and the absence of knowledge regarding system's transfer function. In order to do so, this method has to address another challenge: exploitation of the indirect propagation paths of the signal transmitted by the source.

In addition, the localization of moving sources is more challenging due to their movement, as the movement leads to time variance of the system. The system, which its input is the signal transmitted by the source and its output is the signal received by the sensor, is characterized by the acoustic impulse response (AIR). The time variance of the system is reflected by the AIR, such that witnessed changes in the AIR through time lead to dependency of the feature vector, representing the received signal, on the explicit time samples.

We propose a supervised manifold learning approach, specifically diffusion maps method, for identifying system's characteristics. Based on this data-driven

approach, we recover the latent physical parameters controlling the system-meaning, the location and velocity of the source.

Each source, one at a time, transmits a white Gaussian noise signal during its movement, which is received by the sensor. The received signal is divided into time frames. The trajectory, formed by the movement of the source during the frame, is approximated by a linear movement segment. Each frame is inspected individually for estimation of the location and velocity of the source. The signal, received during the frame, is processed in order to extract a feature vector. In such a manner, we collect a training set of observations, generated from different sources that their location and velocity values are known.

By restricting the movement to slow and gradually changing velocity components only, we introduce small changes to the AIR along time frames. As a result, the system can be approximately addressed as time invariant given sufficiently short time frames. Accordingly, the feature vector experiences small changes as well and its time differences depend purely on the variations of the AIR. These variations of the AIR are derived from the evolution of the only degrees-of-freedom of the system: the location and velocity of the source- meaning they can be inferred.

Using diffusion maps with a Euclidean distance-based diffusion kernel, for assessing the affinity between the feature vectors, we learn the nonlinear structure of the manifold of the data. The data is organized on the manifold according to the location and velocity values of the sources. In addition, we discuss popular alternatives for evaluation of the affinity (e.g., Mahalanobis distance) and explain our choice.

By executing an eigendecomposition of the affinity matrix of all training observations we find their mapping from the high-dimensional observable space to the embedded low-dimensional space. Given a set of observations generated from sources that their location and velocity values are unknown, we use Nystrom method for extension. Furthermore, we discuss flaws of suggested extension methods of former works, improve them and provide an accurate extended manifold structure.

The unknown location and velocity of a source can be recovered according to its observation's nearest training neighbors on the embedded low-dimensional space. The recovery of the location and velocity is determined according to the midway point of the segment. Based on that point, we minimize the approximation error of the true trajectory by the linear segment. The relevant accuracy measure, suggested in previous works, is modified and extended in order to quantify the estimation error for a moving source. In addition, we discuss the reasons for that modification and extension (e.g., combination of physical quantities of different units).

The capabilities and limitations of the proposed single-sensor localization algorithm are examined in three cases:

1. A baseline case of sources that have a fixed location, which consists of two subcases: one-dimensional (i.e., azimuth) and two-dimensional (i.e., radius and azimuth).
2. An extensive case of deterministically moving sources, composed of various experiments examining the sensitivity to different hyperparameters (e.g., frame length), variables (e.g., speed and direction of movement) and conditions (e.g., reverberation time). This case is the main contribution of our research.
3. A case of Brownian motion induced moving sources, which represents the influence of environmental conditions on the movement of the sources by rapid and random perturbations. This case examines the capability of the algorithm to cope with a violation of the fundamental assumption regarding the gradually changing velocity.

We establish a proof of concept proposing a state-of-the-art solution for the recovery of the location and velocity of a moving source, using a single microphone. The algorithm express its robustness to reverberant and noisy environments, yet its sensitivity to environmental conditions changes. The results verify the necessity of reflections in order to yield an accurate estimation. The results demonstrate the influence of the speed of the sources, resulting in a tradeoff between the accuracy of the estimated location and the accuracy of the estimated direction. In addition, we witness the influence of opposing forces such as the validity of our assumptions regarding the velocity attributes, and the time variance property of the system. The algorithm struggles to distinguish between speed values of sources, that move at the same direction. While the algorithm is designated for recovering the location and velocity of slow sources that change direction and speed gradually, it is even capable of successfully estimating the location and the average velocity of slow sources that change their velocity rapidly and randomly.

Regarding sources that have a fixed location, the algorithm provides state-of-the-art results for the recovery of one-dimensional location and two-dimensional location alike.

1.4 Organization

This thesis is organized as follows. In Chapter 2, we present the scientific background related to this work. In Chapter 3, we formulate the problem and

present proposed methodology and procedure for single-sensor source localization for moving acoustic sources based on diffusion kernels. In Chapter 4, we demonstrate the capabilities of the proposed algorithm using simulations. The proposed algorithm is examined in three cases (i.e., stationary sources, deterministically moving sources, and Brownian motion induced moving sources) through various experiments (e.g., sensitivity to different hyperparameters, variables and conditions), results are shown and accompanying tradeoffs are provided. In Chapter 5, conclusions regarding algorithm's performance are presented, followed by possible directions and opportunities for future research.

Chapter 2

Scientific Background

In this chapter, we provide background for reading this thesis and briefly describe an alternative method for single-sensor localization of sources that have a fixed location. Section 2.1 contains the required background for Chapters 3 and 4.

2.1 Diffusion Maps Method

Diffusion maps, introduced by Coifman and Lafon [30], is a method among the family of the geometry-based nonlinear dimensionality reduction methods of the manifold learning approach. The goal of the method is finding the intrinsic nonlinear structure of the low-dimensional manifold, on which the high-dimensional observable data lies. In this work, we focus on the application of diffusion maps method in the field of single-sensor source localization. Throughout this section, we review the fundamental of the framework of this method in localizing sources that their location is stationary using single-sensor, as introduced by Talmon *et al.* [10].

Let $h_{\theta_i}(n)$ denote an acoustic impulse response (AIR) between the i th source and the sensor, at relative location $\theta_i \in \mathbb{R}^d$. The signal received by the sensor, denoted by $y_i(n)$, is given by:

$$y_i(n) = h_{\theta_i}(n) * x_i(n), \quad (2.1)$$

where $x_i(n)$ is the signal transmitted by the i th source, assumed as a white Gaussian noise signal. We note that $y_i(n)$ is the output signal of a time-invariant system

In order to generate the training dataset, we select a set of m predefined locations of sources that their location is fixed $\bar{\Theta} = \{\bar{\theta}_1, \dots, \bar{\theta}_m\} \subset \mathbb{R}^d$. Let

$\{\boldsymbol{\theta}_{i_j}\}_{j=1}^L$ denote small perturbations of $\bar{\boldsymbol{\theta}}_i$. Let $\Theta = \{\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_M\} \subset \mathbb{R}^d$ be a set of M unknown source locations, corresponding to the M measurements of the test dataset.

From each observation, a feature vector is extracted based on an autocorrelation function of the observation. From (2.1), under the assumption of a white Gaussian noise (WGN) input signal, the autocorrelation function of $y_i(n)$ is given by:

$$c_{\mathbf{y}_i}(\tau) = h_{\boldsymbol{\theta}_i}(\tau) * h_{\boldsymbol{\theta}_i}(-\tau) * c_{\mathbf{x}_i}(\tau), \quad (2.2)$$

where $c_{\mathbf{x}_i}(\tau)$ and $c_{\mathbf{y}_i}(\tau)$ denote the autocorrelation functions of $x_i(n)$ and $y_i(n)$, respectively. Let $\mathbf{c}_i \in \mathbb{R}^D$ be a feature vector of length D which is composed of D autocorrelation elements, i.e.:

$$\mathbf{c}_i^{(j)} = c_{\mathbf{y}_i}(n, n+j) = \mathbb{E}[y_i(n)y_i(n+j)] \quad (2.3)$$

for $j = 0, \dots, D-1$. In such a manner, we extract a feature vector for each signal received by the sensor (including the received signals which are generated from the slightly perturbed known locations). Let $\{\mathbf{c}_i\}_{i=1}^{m+M}$ denote the set of all observations, and let $\{\bar{\mathbf{c}}_i\}_{i=1}^m$ denote its subset of observations corresponding to the labeled locations in $\bar{\Theta}$.

It is assumed that the high-dimensional observations lie on a low-dimensional manifold \mathcal{M} embedded in a D -dimensional Euclidean space, i.e. $\mathbf{c}_i \in \mathcal{M} \subset \mathbb{R}^D$. According to the diffusion maps method, the manifold can be referred as a graph, where the feature vectors are graph's nodes. In addition, the weights of the edges are defined according to a kernel function (Gaussian kernel typically) with a distance measure. There are two common choices for kernel's distance metric: Euclidean and Mahalanobis. The motivation for the Mahalanobis distance-based kernel, which the algorithm of Talmon *et al.* [10] is based on, is the direct relation between the observations and the latent parameters [31], as follows:

$$\|\bar{\boldsymbol{\theta}}_k - \bar{\boldsymbol{\theta}}_l\|^2 \approx (\bar{\mathbf{c}}_k - \bar{\mathbf{c}}_l)^T [\boldsymbol{\Sigma}_k + \boldsymbol{\Sigma}_l]^{-1} (\bar{\mathbf{c}}_k - \bar{\mathbf{c}}_l), \quad (2.4)$$

where the local covariance matrix of each training observations is estimated by:

$$\hat{\boldsymbol{\Sigma}}_i = \frac{1}{L} \sum_{j=1}^L (\mathbf{c}_{i_j} - \boldsymbol{\mu}_i) (\mathbf{c}_{i_j} - \boldsymbol{\mu}_i)^T, \quad (2.5)$$

where $\boldsymbol{\mu}_i = \frac{1}{L} \sum_{j=1}^L \mathbf{c}_{i_j}$.

The Mahalanobis distance-based affinity matrix $\mathbf{W} \in \mathbb{R}^{m \times m}$ is computed between all m training observations. According to [31, 32], its kl th element is given by:

$$\mathbf{W}^{(kl)} = \frac{\pi}{\mathbf{D}^{(kl)}} \exp \left\{ -\frac{(\bar{\mathbf{c}}_k - \bar{\mathbf{c}}_l)^T [\hat{\Sigma}_k + \hat{\Sigma}_l]^{-1} (\bar{\mathbf{c}}_k - \bar{\mathbf{c}}_l)}{\varepsilon} \right\}, \quad (2.6)$$

where ε is the kernel scale and d_{kl} is a normalization factor.

Let $\{\lambda_j\}_{j=0}^{m-1}$ and $\{\varphi_j\}_{j=0}^{m-1}$ be the eigenvalues (in descending order) and eigenvectors of length m of the affinity matrix \mathbf{W} , respectively. Accordingly, let Φ_d be the embedding of the training observations into the embedded low-dimensional space, given by:

$$\Phi_d : \bar{\mathbf{c}}_i \rightarrow [\varphi_1^{(i)}, \dots, \varphi_d^{(i)}]^T.$$

Φ_d represents the observations in terms of their latent parameters.

Let $\mathbf{A} \in \mathbb{R}^{(M+m) \times m}$ be an affinity matrix between all observations and the training observations, given by:

$$\mathbf{A}^{(kl)} = \exp \left\{ -\frac{(\mathbf{c}_k - \bar{\mathbf{c}}_l)^T \hat{\Sigma}_l^{-1} (\mathbf{c}_k - \bar{\mathbf{c}}_l)}{\varepsilon} \right\},$$

Let $\tilde{\mathbf{A}}$ be a normalized affinity matrix $\tilde{\mathbf{A}} = \mathbf{A} \mathbf{S}^{-1/2}$, where $\mathbf{S} = \text{diag} \{ \mathbf{A}^T \mathbf{A} \mathbf{1} \}$ is a diagonal matrix. The normalized affinity matrix $\tilde{\mathbf{A}}$ satisfies $\mathbf{W} = \tilde{\mathbf{A}}^T \tilde{\mathbf{A}}$. Therefore, the eigenvectors of \mathbf{W} are the right singular vectors of $\tilde{\mathbf{A}}$. The extended eigenvectors $\{\psi_j\}_{j=0}^{M+m-1}$ of length $M+m$, on the other hand, are the left singular vectors of $\tilde{\mathbf{A}}$, and are defined by a weighted linear combination, as follows:

$$\psi_j = \frac{1}{\sqrt{\lambda_j}} \tilde{\mathbf{A}} \varphi_j.$$

Thus, let Ψ_d be the embedding of all observations into the embedded low-dimensional space, given by:

$$\Psi_d : \mathbf{c}_i \rightarrow [\psi_1^{(i)}, \dots, \psi_d^{(i)}]^T.$$

Ψ_d represents all observations, and particular the those whose locations are unknown, in terms of their latent parameters.

Accordingly, the unknown location of the test observation can be estimated using its labeled neighbors on the manifold, by a weighted interpolation, as follows:

$$\hat{\boldsymbol{\theta}}_i = \sum_{j: \Psi_d(\bar{\mathbf{c}}_j) \in \tilde{\mathcal{N}}_i} \gamma_j(\mathbf{c}_i) \bar{\boldsymbol{\theta}}_j, \quad (2.7)$$

where $\tilde{\mathcal{N}}_i$ consists of the \tilde{k} -nearest embedded training measurements $\{\Psi_d(\bar{\mathbf{c}}_j)\}_{j=1}^{\tilde{k}}$

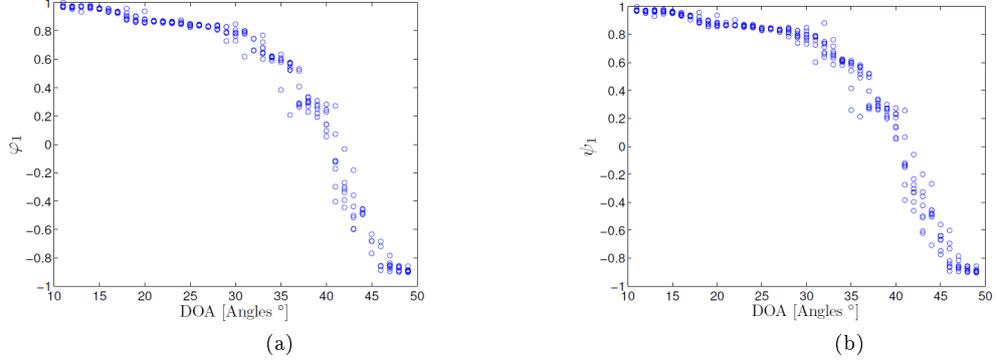


Figure 2.1: Embedding of the observations as a function of the azimuth angle by Talmon *et al.* [10]. (a) Values of the eigenvector φ_1 . (b) Values of the extended eigenvector ψ_1 .

of $\Psi_d(\mathbf{c}_i)$ according to the diffusion distance. In addition, the interpolation coefficients, denoted by $\{\gamma_j\}_{j=1}^{\bar{k}}$ and satisfying $\sum_{j=1}^{\bar{k}} \gamma_j(\mathbf{c}_i) = 1$, are proportional to the diffusion distance of the embedded test sample from its corresponding labeled neighbors on the manifold, as follows:

$$\gamma_j(\mathbf{c}_i) = \frac{\exp\left(-\frac{\|\Psi_d(\mathbf{c}_i) - \Psi_d(\bar{\mathbf{c}}_j)\|^2}{\varepsilon_{\gamma_i}}\right)}{\sum_{l: \Psi_d(\bar{\mathbf{c}}_l) \in \tilde{\mathcal{N}}_i} \exp\left(-\frac{\|\Psi_d(\mathbf{c}_i) - \Psi_d(\bar{\mathbf{c}}_l)\|^2}{\varepsilon_{\gamma_i}}\right)}, \quad (2.8)$$

where ε_{γ_i} is defined as the minimal distance between $\Psi_d(\mathbf{c}_i)$ and its nearest neighbor. Consequently, the estimation error of the location is defined as follows:

$$e_i = \|\boldsymbol{\theta}_i - \hat{\boldsymbol{\theta}}_i\|. \quad (2.9)$$

Accordingly, the root mean square error (RMSE) is given by:

$$\text{RMSE} = \sqrt{\frac{1}{M} \sum_{i=1}^M (e_i)^2}. \quad (2.10)$$

Using this method, Talmon *et al.* [10] have conducted a real-data experiment for estimation of a one-dimensional location (i.e., azimuth) of stationary sources. The results of this experiment, depicted in Fig. 2.1, demonstrate the monotonic behavior of the observations with respect to the DOA- meaning the azimuth angle (direction of arrival). We elaborate on these results and make a comparison to a similar one-dimensional stationary scenario in Section 4.3.

Chapter 3

Localization of Moving Acoustic Sources

3.1 Introduction

In this chapter we present the methodology and procedure for passive single-sensor source localization, in particular for moving acoustic sources, using diffusion kernels.

First, we formulate the problem for a generalized case which not only relates to movement of the sources in a deterministic fashion, but also to a more realistic movement model which involves spontaneous movement as well. In addition, we list the challenges the movement arise, such as the time variance of the system, and discuss the parameters affecting our system.

Subsequently, we look for a method which will allow us to recover the location and velocity of the sources, based on the observations acquired by our single microphone. Thus, we draw inspiration by the supervised manifold learning based method, introduced by Talmon *et al.* [9], which was implemented explicitly by diffusion kernels and was followed by its application in [10] for the recovery of an azimuth angle of stationary (i.e., fixed location) sources using a single sensor. However, we make substantial changes in order to adapt it to our settings, scenarios and assumptions. For example, rather than relying on Mahalanobis distance-based diffusion kernel, we use a customized version of Euclidean distance-based diffusion kernel, inspired by the study of the acoustic manifold by Laufer-Goldsthtein *et al.* [12] and its successful application in their following paper [13]- both focusing on multiple-sensors. Nevertheless, our extension of the diffusion framework for the test set, and its corresponding diffusion kernel in particular, do not follow neither [10] nor [12, 13].

We extract feature vectors from the observations. In addition, we discuss the hypothesis of a manifold, its structure and common affinity measures on the manifold, which may reflect accurately the physical vicinity between the locations and the velocities of the sources. Consequently, we compose a diffusion kernel between the feature vectors.

At last, based on the kernel we find a mapping from the high-dimensional observable space to the low-dimensional embedded space, which is governed by the parameters controlling the system, and resembles the original parametric space up to a monotonic distortion. Following the embedding of all training observations, we can estimate the unknown location and velocity of a new and unlabeled observation, based on its proximity to close training embeddings on the manifold. Accordingly, we discuss the relevant accuracy measure for the combination of physical quantities of different units.

The organization of the rest of the chapter is as follows. In Section 3.2, we formulate the problem. In Section 3.3, the computation of the diffusion kernel is presented. In Section 3.4, we present the proposed algorithm for single-sensor source localization, in particular for moving sources, based on diffusion maps. At last, the conclusion of the chapter in Section 3.5, summarizes the framework of the problem and the proposed method for the recovery of an unknown location and velocity of a source.

3.2 Problem Formulation

We consider a standard enclosure, such as a conference room. The velocity of moving source as a function of time, denoted by $\dot{\mathbf{p}}(t)$, is defined as a combination of a deterministic term and a random term, as follows:

$$\dot{\mathbf{p}}(t) = a(\mathbf{p}(t)) + \mathbf{n}(t), \quad (3.1)$$

where:

- $\mathbf{p}(t)$ is the absolute location of the source at time t ,
- $a(\mathbf{p}(t))$ is the drift term which describes the deterministic component of the velocity at location $\mathbf{p}(t)$,
- $a(\cdot)$ is a deterministic function which maps $\mathbf{p}(t)$ to a velocity, i.e.,
 $a : \mathbf{p}(t) \rightarrow \mathbb{R}^2$,
- $\mathbf{n}(t)$ is the Brownian motion term, which represents rapid and random perturbations that affect the movement of the sources as a result of environmental conditions (e.g., enclosure vibrations, vibrations of

source's mount, and wind). The Brownian motion term is described by white Gaussian noise.

Each source, one at a time, transmits a signal during its movement. We assume that the source signal is a zero-mean wide-sense stationary (WSS) process. Since many natural signals, such as speech and music, are WSS in short time frames (i.e., quasi-stationary process), we can even be satisfied with such a weaker assumption.

An acoustic impulse response (AIR), between a source and the microphone, is affected by several parameters, such as:

- Enclosure dimensions.
- Locations of the source and the microphone.
- Reflection coefficients (or reverberation time) of the walls, floor and ceiling.
- Presence of objects in the room (e.g., furniture, windows and doors).

Let $h_{\boldsymbol{\theta}_i(j)}(n, j)$ denote a real-valued AIR, which is defined as the response at discrete time index n to an impulse transmitted at discrete time index j , between the i th source and the microphone, with respect to the parameters vector $\boldsymbol{\theta}_i(j)$. The parameters vector of the i th source at discrete time index j , is defined as a combination of the relative location and the velocity of the source, i.e., $\boldsymbol{\theta}_i(j) = [\rho_i(j), \phi_i(j), s_i(j), \beta_i(j)]$, where:

$\rho_i(j)$ is the distance (i.e., radius) between the source and the microphone,

$\phi_i(j)$ is the bearing angle (also the azimuth angle in our setting),

$s_i(j)$ is the speed of the source,

$\beta_i(j)$ is the direction of movement (also known as the course) of the source.

It is assumed that the height difference between the source and the microphone is negligible- thus the elevation angle remains constant. Note that using Cartesian representation, $\boldsymbol{\theta}_i(j)$ can also be represented by $[\tilde{x}_i(j), \tilde{y}_i(j), v_{\tilde{x}_i}(j), v_{\tilde{y}_i}(j)]$, where $\tilde{x}_i(j)$ and $\tilde{y}_i(j)$ are the horizontal and vertical coordinates of the absolute location (\tilde{z} -axis is negligible), respectively. $v_{\tilde{x}_i}(j)$ and $v_{\tilde{y}_i}(j)$ denote the horizontal and vertical speed components, respectively.

The signal received by the sensor, denoted by $y_i(n)$, consists of the direct and indirect propagation paths of the transmitted signal, and is defined by:

$$y_i(n) = H_{\boldsymbol{\theta}} \{x_i(n)\} = \sum_{j=-\infty}^{\infty} h_{\boldsymbol{\theta}_i(j)}(n, j)x_i(j), \quad (3.2)$$

where $x_i(n)$ is the signal transmitted by the i th source. We note that $x_i(n)$ and $y_i(n)$ are the real-valued input and output signals, of finite length, of the system H_{θ} , which corresponds to the AIR and depends on the parameters vector. In addition, since the AIR is affected tremendously by the movement of the source, the system is time variant. We use a white Gaussian noise (WGN) signal as the source signal, since it fully excites the frequency response of the AIR. The received signal is saved and divided into time frames.

We assume the trajectory, formed by the movement of the source during the time frame, can be approximated by a linear movement segment. We inspect each time frame individually for estimation of the location and velocity. The goal of the proposed algorithm is to determine the unknown location and velocity of a source based on training dataset, which is available beforehand. For each time frame, we manage two datasets, based on the different signals received by the sensor: a training dataset and a test dataset. In order to generate the training dataset, we choose arbitrarily m known locations and velocities of the source $\bar{\Theta} = \{\bar{\theta}_1(q), \dots, \bar{\theta}_m(q)\} \subset \mathbb{R}^d$, where q is a query point along the trajectories of the sources, and d is the dimension of the parameters vector (i.e., the number of system's degrees-of-freedom). Let $\Theta = \{\theta_{m+1}(q), \dots, \theta_{m+M}(q)\} \subset \mathbb{R}^d$ be a set of M arbitrary unknown source locations and velocities, corresponding to the M measurements of the test dataset. We define the query point q as the midway point of the trajectory, such that the approximation error of the true trajectory by the linear segment is minimized. Note that the acoustic environment is fixed between training and test stages (i.e., room characteristics and microphone location remain unchanged), thus the only degrees-of-freedom of the controlling parameters of the AIR are the locations of the sources and their velocities.

3.3 Diffusion Kernel

In this section we compose a diffusion kernel between feature vectors, derived from the given observations, namely system's output signals.

3.3.1 From Observations to Feature Vectors

From each observation, a feature vector is extracted. We follow Talmon *et al.* [9, 10], and define our feature vector based on an autocorrelation function of the observation. The reason for that choice is that a second order statistics measure conveys the location better and less dependent on the specific random unknown transmitted signal, rather than using the raw observation. From (3.2), under the assumption of a WGN input signal, the autocorrelation function of $y_i(n)$, the output signal of a time-variant system, is given by:

$$\begin{aligned}
c_{\mathbf{y}_i}(n_1, n_2) &= \mathbb{E}[y_i(n_1)y_i(n_2)] = \\
&= \sum_{j, l=-\infty}^{\infty} h_{\boldsymbol{\theta}_i(j)}(n_1, j)h_{\boldsymbol{\theta}_i(l)}(n_2, l)c_{\mathbf{x}_i}(j-l), \tag{3.3}
\end{aligned}$$

where $c_{\mathbf{x}_i}(j-l)$ and $c_{\mathbf{y}_i}(n_1, n_2)$ denote the time-invariant autocorrelation function of the input signal $x_i(n)$ and the time-variant autocorrelation function of the output signal $y_i(n)$, respectively. $\mathbb{E}[\cdot]$ denotes an expected value. As implied by (3.3), we can represent the observation \mathbf{y}_i as a function of the controlling parameters $\boldsymbol{\theta}_i$ of the system. It is assumed that given a sufficiently short time interval, the first two moments (autocorrelation in particular) of the quasi-stationary input signal would not change along the interval. Indeed, the explicit time-invariant autocorrelation of a WGN input signal is $c_{\mathbf{x}_i}(\tau) = \sigma_{\mathbf{x}_i}^2 \delta(\tau)$, thus by substitution, we yield

$$c_{\mathbf{y}_i}(n_1, n_2) = \sigma_{\mathbf{x}_i}^2 \sum_{j=-\infty}^{\infty} h_{\boldsymbol{\theta}_i(j)}(n_1, j)h_{\boldsymbol{\theta}_i(j)}(n_2, j). \tag{3.4}$$

In fact, by considering an additional assumption (apart of short time intervals) of slow speed and gradually changing velocity (Brownian motion ignored), we introduce small changes to the AIR along the time frames. As a result, we can obtain the familiar convolution-based version of (3.3), as in [9, 10]:

$$c_{\mathbf{y}_i}(\tau) = h_{\boldsymbol{\theta}_i}(\tau) * h_{\boldsymbol{\theta}_i}(-\tau) * c_{\mathbf{x}_i}(\tau) = \sigma_{\mathbf{x}_i}^2 h_{\boldsymbol{\theta}_i}(\tau) * h_{\boldsymbol{\theta}_i}(-\tau). \tag{3.5}$$

As indicated by (3.5), the time differences of the autocorrelation function of the output signal depend purely on the variations of the AIR, i.e., on the evolution of the location and velocity of the i th source, $\boldsymbol{\theta}_i$. We remark that examining the short-term power spectral density (PSD) of the observation, rather than its autocorrelation function, will yield same results due to the Fourier transform relation between the two, but is less convenient.

Let c express the *nonlinear mapping* of the location and the velocity of the i th source, $\boldsymbol{\theta}_i \in \mathbb{R}^d$, to the first D elements of the autocorrelation function of the observation \mathbf{y}_i , defined as follows:

$$\mathbf{c}_i = c(\boldsymbol{\theta}_i), \tag{3.6}$$

where $\mathbf{c}_i \in \mathbb{R}^D$ is a vector of length D which is composed of the autocorrelation elements, i.e.:

$$\mathbf{c}_i^{(j)} = c_{\mathbf{y}_i}(n, n+j) = \mathbb{E}[y_i(n)y_i(n+j)] \quad (3.7)$$

for $j = 0, \dots, D-1$. In such a manner, we extract a *feature vector* for each signal received by the sensor. Note that the length of the feature vectors, D , should reflect the tradeoff between the length of the autocorrelation function (large value) and the latency and quasi-stationarity properties considerations (small value). Let $\Gamma = \{\mathbf{c}_i\}_{i=1}^M$ be the set of the feature vectors with respect to the unlabeled parameters in Θ . Accordingly, let $\bar{\Gamma} = \{\bar{\mathbf{c}}_i\}_{i=1}^m$ denote the set of the feature vectors with respect to the labeled parameters in $\bar{\Theta}$. We aim to recover the unknown parameters vectors based on the aforementioned feature vectors.

3.3.2 Manifold Structure and the Choice of an Affinity Measure

As pointed out previously in Sections 3.3.1 and 3.2, the autocorrelation function based feature vectors have a high-dimensional representation in \mathbb{R}^D , which resembles the high number of reflections from all surfaces characterizing the bounded environment- hence the AIR. On the other hand, the typical AIR, associated with our feature vector, is characterized by an exponentially decaying envelope. Moreover, the feature vectors are affected by a small set of parameters associated with the physical attributes of the enclosure (e.g., its size and geometry, surfaces' material, objects presence, locations of the sources and the microphone), in addition to the influence of the speech signal of the source which is quasi-stationary. Therefore, it is assumed that not only that the feature vectors, which were originated from a specific region of interest in the enclosure, do not spread uniformly in the entire space of \mathbb{R}^D , but are also restricted to more explicit and even compact structure. That structure, namely the manifold \mathcal{M} of dimension d , is significantly smaller than the dimension of the surrounding high-dimensional space (i.e., $d \ll D$). Thus, by applying the notation from Section 3.3.1, we define $c : \Theta \rightarrow \Gamma$ to be the nonlinear map between an unknown parametric manifold $\mathcal{M} \subseteq \Theta \subset \mathbb{R}^d$ and its corresponding observation-based feature vectors dataset $\Gamma \subset \mathbb{R}^D$.

Even though our setting involves moving sources, the hypothesis of such manifold can be rationalized by the combination of quasi-stationary input signal, slow speed and gradually changing velocity components of the source, stationary microphone and short time intervals. These assumptions allow us to experience small changes in the feature vector along the time intervals, and thus inferring the locations and velocities of the sources, which are the only degrees-of-freedom of the system. Therefore, we conclude that the feature vec-

tors can be represented by a low-dimensional manifold, which its embeddings are ruled by the location and velocity of the sources (i.e., parameters vectors).

The considered low-dimensional manifold is assumed to be a real nonlinear structure, but in practice it is locally linear in small areas. Indeed, the surface of the manifold is flat in the close neighborhood of each embedded point and coincide with the tangent plane to the manifold at this embedded point. This is in line with the implied assumption, based on which a small change in the physical parameters vector leads to a slight change in the corresponding feature vector, whereas a big variation results in a completely different feature vector. Therefore, the similarities between points that are located on the manifold in the vicinity of each other, can be measured reliably by using the Euclidean distance. Note that the Euclidean distance cannot assess reliably affinity for large scales. Instead, large scales shall be dealt with the geodesic distance, which is the locally shortest path along the manifold, in case that the structure of the manifold is known (the Euclidean distance is equal to the geodesic distance in case of flat/linear manifold only).

Another popular choice for measuring affinities on the manifold is Mahalanobis distance, which was the key element in several papers [9], [10], [11], since the affinities measured between feature vectors by this distance approximates the Euclidean distance between the corresponding physical parameters vectors [32]. However it holds several fundamental practical drawbacks: in order to estimate its local covariance matrices, several additional local slightly perturbed observations shall be generated for each training observation, which is in fact a resources-related burden in terms of both storage and implementation during a real-data experiment. In addition, these local covariance matrices are singular and thus not invertible, whereas computing their pseudo-inverse (their rank is d [9]) as an alternative is not necessarily trustworthy.

In this work, for complying with manifold’s nonlinear and unknown structure, affinities in local vicinities are measured by the Euclidean distance, whereas greater distances are omitted.

3.3.3 Diffusion Kernel Computation

In order to acquire the independent parameters controlling our system, which their availability is solely by the nonlinear feature vectors of the observations, we define an $m \times m$ affinity matrix \mathbf{W} between all the feature vectors in $\bar{\Gamma}$, related to the corresponding parameters vectors set $\bar{\Theta}$. The affinity matrix consists of a Gaussian kernel with a scale parameter ε , and following Section 3.3.2 its ij th element is calculated according to

$$\mathbf{W}^{(ij)} = \begin{cases} \exp\left\{-\frac{\|\bar{\mathbf{c}}_i - \bar{\mathbf{c}}_j\|^2}{\varepsilon}\right\} & , \text{if } \bar{\mathbf{c}}_i \in \bar{\mathcal{N}}_j \text{ or } \bar{\mathbf{c}}_j \in \bar{\mathcal{N}}_i \\ 0 & , \text{otherwise} \end{cases}, \quad (3.8)$$

where $\bar{\mathcal{N}}_j$ is the set of the k -nearest-neighbors of $\bar{\mathbf{c}}_j$ in $\bar{\Gamma}$. It is worth mentioning that the set $\bar{\mathcal{N}}_j$ is selected by the ordinary Euclidean distance between the feature vectors, since as mentioned previously it is trustworthy in close neighborhoods, whereas its issue in larger distances is irrelevant for this task.

The value of ε is determined in proportion to the median value of the Gaussian function's various numerator values associated with the non-zeros elements of the affinity matrix. That is reasoned by the assumption that the Euclidean distance is monotonic with respect to the parameters vector in small areas, as discussed in Section 3.3.2. The proportion is decided by an exhaustive search.

3.4 Localization Based on Diffusion Mapping

In this section we introduce a supervised single-sensor source localization algorithm for recovering the location and velocity of the sources by utilization of the feature vectors, based on the kernel of Section 3.3. Thanks to eigendecomposition of the kernel, we find the mapping from the observable space \mathbb{R}^D to the intrinsic manifold space of \mathbb{R}^d , which is ruled by the dominating parameters vector and resembles the original parametric space up to a monotonic distortion. Based on the mapping of all training observations into the manifold, the localization of a new observation of unknown parameters vector is estimated by exploiting its nearest labeled neighbors in the embedded space.

3.4.1 Manifold Parameterization

By constructing a diagonal matrix \mathbf{D} , defined as

$$\mathbf{D}^{(ii)} = \sum_{j=1}^m \mathbf{W}^{(ij)}, \quad (3.9)$$

we normalize the affinity matrix \mathbf{W} and obtain the transition matrix

$$\mathbf{P} = \mathbf{D}^{-1}\mathbf{W}, \quad (3.10)$$

which defines a Markov process, that is a discrete diffusion process over the training data set. Accordingly, $\mathbf{P}^{(ij)} \equiv \Pr(\bar{\mathbf{c}}_j | \bar{\mathbf{c}}_i)$ describes the single-step transition probability from $\bar{\mathbf{c}}_i$ to $\bar{\mathbf{c}}_j$. Let \mathbf{L} be the normalized graph-Laplacian [33], defined by

$$\mathbf{L} = \mathbf{I} - \mathbf{P}, \quad (3.11)$$

where \mathbf{I} is an identity matrix. It can be shown, under certain conditions, that the graph-Laplacian matrix \mathbf{L} converges to the Fokker-Planck operator on the manifold [34, 35], which describes a continuous diffusion process over the dataset.

By applying eigendecomposition of the transition matrix \mathbf{P} , the labeled feature vectors are nonlinearly mapped into a new embedded space, according to the parameterization of the manifold \mathcal{M} . In fact, the parameterization of the manifold constitutes an intrinsic representation of the labeled feature vectors. Let $\{\lambda_j\}_{j=0}^{m-1}$ and $\{\psi_j\}_{j=0}^{m-1}$ be the eigenvalues and eigenvectors of the transition matrix \mathbf{P} . Note that $\lambda_0 = 1$ and its corresponding eigenvector ψ_0 is a vector of ones [31]. The eigenvectors of \mathbf{P} are assumed to establish the reparameterization of the independent controlling parameters of the m training observations. Thus, let Ψ_d be the diffusion mapping of the training feature vectors into the embedded Euclidean space \mathbb{R}^d , which is spanned by d eigenvectors corresponding to the d largest eigenvalues (trivial case excluded). Ψ_d is defined as

$$\Psi_d : \bar{\mathbf{c}}_i \rightarrow \left[\lambda_1 \psi_1^{(i)}, \dots, \lambda_d \psi_d^{(i)} \right]^T. \quad (3.12)$$

Hence, Ψ_d maps the feature vectors of the observations to their corresponding representation of independent controlling parameters in the embedded manifold. In fact, this map can be regarded as an approximation of the inverse-map of the nonlinear function c , in addition to a monotonic distortion between the two parametric spaces (i.e., the original and the embedded manifold) [9, Fig. 1]. Note that the diffusion maps combines local relations by the construction of the affinity kernel with global processing by the spectral decomposition.

3.4.2 Extension for New Observations

Given an additional set of M new sequential observations, generated from unknown location and velocities of the sources, we seek to embed them as well in the low-dimensional manifold. However, in order to avoid another spectral decomposition, we tackle it according to the Nyström method [36] for out-of-sample extension (OOSE), by adding M new rows to the affinity matrix \mathbf{W} , as follows:

$$\mathbf{W}^{(\tilde{i}j)} = \begin{cases} \exp \left\{ -\frac{\|\mathbf{c}_i - \bar{\mathbf{c}}_j\|^2}{\varepsilon} \right\} & , \text{ if } \bar{\mathbf{c}}_j \in \mathcal{N}_i^{\tilde{c}} \\ 0 & , \text{ otherwise} \end{cases}, \quad (3.13)$$

where $\tilde{i} = m + i$ and $\mathcal{N}_i^{\tilde{c}}$ is the set of the k -nearest-neighbors of \mathbf{c}_i in $\bar{\Gamma}$. In contrast to the construction process of the symmetric affinity matrix \mathbf{W} of

(3.8), the affinity between the observations in (3.13) is carried out with respect to the training set only. Accordingly, the new entries of the transition matrix \mathbf{P} are given by:

$$\mathbf{P}^{(\tilde{i}j)} = \left(\sum_{j=1}^m \mathbf{W}^{(\tilde{i}j)} \right)^{-1} \mathbf{W}^{(\tilde{i}j)}. \quad (3.14)$$

The new entries of the extended eigenvectors of \mathbf{P} can be represented as a weighted linear interpolation of the original entries of the eigenvectors, as follows:

$$\psi_l^{(\tilde{i})} = \frac{1}{\lambda_l} \sum_{j=1}^m \mathbf{P}^{(\tilde{i}j)} \psi_l^{(j)}. \quad (3.15)$$

Subsequently, Ψ_d maps the new feature vectors of the unlabeled observations to their corresponding representation of independent dominating parameters in the embedded manifold:

$$\Psi_d : \mathbf{c}_i \rightarrow \left[\lambda_1 \psi_1^{(\tilde{i})}, \dots, \lambda_d \psi_d^{(\tilde{i})} \right]^T. \quad (3.16)$$

The suggested extensions of former works were based on either additional mathematical relations [9–11, 32] in case of Mahalanobis distance-based diffusion kernel, or straightforwardly [13, 18] in case of Euclidean distance-based diffusion kernel. Note that our extended entries of the eigenvectors are obtained following normalization by the transition matrix. In addition, in order to maintain the nonlinear structure of the manifold, our extended entries of the eigenvectors are acquired by restricting the computation of the entries of the new rows of the affinity matrix to those which are associated with the nearest training observations. In other words, our extension is free from the issues of its Euclidean distance-based counterpart suggested in [13, 18], which its combination of no normalization and affinity calculation with respect to all training observations (i.e., not restricted to close area only, in contrast to the gist of Section 3.3.2, as the Euclidean distance is accurate for larger distances in case of flat/linear manifold only) leads to a scaled and inaccurate extended manifold structure.

3.4.3 Recovery of the Controlling Independent Parameters

We can take advantage of the proximity of the unlabeled test observations to the labeled training observations in the embedded manifold space \mathbb{R}^d , which were mapped by Ψ_d , for estimating their parameters vectors.

As mentioned in Section 3.3.2 regarding the structure of the manifold, the

geodesic distance, which is the locally shortest path along the manifold, should be worked with in order to accurately measure affinities between feature vectors. The geodesic distance can be approximated by the diffusion distance, which is equal to the Euclidean distance in the embedded space when using all eigenvectors. The diffusion distance can be appropriately approximated by using merely the first d non-trivial eigenvectors [30], i.e.:

$$D_{\text{DIFF}}(\mathbf{c}_i, \mathbf{c}_j) \cong \|\Psi_d(\mathbf{c}_i) - \Psi_d(\mathbf{c}_j)\|, \quad (3.17)$$

where $\Psi_d(\mathbf{c}_i)$ is a vector of d elements, representing the mapping of the i th feature vector to the embedded manifold, i.e., $\Psi_d(\mathbf{c}_i) = \left[\lambda_1 \psi_1^{(i)}, \dots, \lambda_d \psi_d^{(i)} \right]^T$.

The ability to measure distances along the manifold thanks to the diffusion distance, provides us the option to determine the affinities between the feature vectors correctly. Thus, samples which are close to each other on the low-dimensional manifold are expected to be acquired from physically adjacent locations and hold similar velocities. Accordingly, the unknown parameters vector of the test observation can be estimated using its labeled neighbors on the manifold, by a weighted interpolation, as follows:

$$\hat{\boldsymbol{\theta}}_i(q) = \sum_{j: \Psi_d(\bar{\mathbf{c}}_j) \in \tilde{\mathcal{N}}_i} \gamma_j(\mathbf{c}_i) \bar{\boldsymbol{\theta}}_j(q), \quad (3.18)$$

where $\tilde{\mathcal{N}}_i$ consists of the \tilde{k}_i -nearest embedded training measurements $\{\Psi_d(\bar{\mathbf{c}}_j)\}_{j=1}^{\tilde{k}_i}$ of $\Psi_d(\mathbf{c}_i)$ according to the diffusion distance. In addition, the interpolation coefficients $\{\gamma_j\}_{j=1}^{\tilde{k}_i}$, satisfying $\sum_{j=1}^{\tilde{k}_i} \gamma_j(\mathbf{c}_i) = 1$, are given by (2.8), as defined in Chapter 2. Consequently, the normalized estimation error is defined as a vector of length d , one for each physical quantity due to its corresponding units, as follows:

$$\mathbf{e}(\mathbf{c}_i) = \left[e_i^{(1)}, \dots, e_i^{(d)} \right], \quad (3.19)$$

where its j th element is defined by:

$$e_i^{(j)} = \frac{|\boldsymbol{\theta}_i^{(j)}(q) - \hat{\boldsymbol{\theta}}_i^{(j)}(q)|}{|\boldsymbol{\theta}_i^{(j)}(q)|}. \quad (3.20)$$

Note that due to the movement of the sources, the estimation error measures suggested in former works, are incompatible (e.g., no reference to the number of dimensions and their units [32]; parameters are referred to as unitless [9]; one-dimensional scenarios with reference to units [10,13]; two-dimensional scenario, with reference to units, where both parameters share the same units, such as azimuth and elevation angles [11]) and thus we suggest an estimation error

measure. Our error measure should not be calculated using a norm as the physical quantities composing the parameters vector have different units. Thus, the estimation error is no longer a scalar and each coordinate of the estimation error is calculated individually- one for each of the physical quantities. As a result, the norm signs are redundant and are replaced by absolute values signs. In addition, in order to cancel out the units of each of the individual estimation errors, we normalize the errors relatively to the real values of the parameters, which also adds another virtue- larger absolute differences for a large real parameter value will not be prioritized any longer over smaller absolute differences for a small real parameter value. A complementary reference for calculating the total estimation error is given in the following Section.

We emphasize that the recovery of the unknown location and velocity of a source is determined according to the query point, as pointed out by (3.18). Based on that point, which we define as the midway point of the trajectory of the source, we minimize the approximation error of the true trajectory by the linear segment, as pointed out by (3.20).

3.4.4 Accuracy Measure

Following the considerations, mentioned in Section 3.4.3, of defining the estimation error as a vector of length d , we measure the accuracy of the algorithm as a linear combination of the root mean square error (RMSE) of each one of the elements of (3.19), given by:

$$\text{RMSE} = \sum_{j=1}^d \alpha_j \sqrt{\frac{1}{M} \sum_{i=1}^M \left(e_i^{(j)} \right)^2}, \quad (3.21)$$

where $\{\alpha_j\}$ are coefficients which represent the significance of the estimation error of each physical quantity, according to user's preference or optimization of the RMSE value (For simplicity, we define as $\{\alpha_j\} = \frac{1}{d}$). Note that in contrast to [9–11, 13, 18], both sources' movement and even more realistic stationary sources scenarios force us to deal with a combination of physical quantities of different units, thus we modify and extend the estimation framework by: defining the error as a vector (3.19), calculating each error component using absolute values (3.20), and at last a weighted summation of all individual RMSE values in (3.21).

3.5 Conclusion

The proposed methodology and the procedure for passive single-sensor source localization extends manifold learning techniques of former works, in order to

recover the unknown location and velocity of moving sources using diffusion kernels.

The challenges the movement arise, first and foremost the time variance of the system, can be tackled by assuming a slow speed and gradually changing velocity components, in addition to short time intervals and a stationary microphone. The combination of all these assumptions result in small changes to the AIR along the time frames. Furthermore, these assumptions, along with a quasi-stationary input signal, allow us to experience small changes in the feature vector along the time intervals. In practice, the time differences of the feature vector depend purely on the variations of the AIR- that is, on the evolution of the degrees-of-freedom of the system: the location and velocity of the source. Thus, we can represent the feature vectors by a low-dimensional manifold, which its embeddings are organized according to these controlling parameters, and consequently to recover the location and velocity.

The structure of the considered low-dimensional manifold is assumed to be a real and nonlinear. However, in practice it is locally linear in the vicinity of each embedded point and its surface is coincide with the tangent plane to the manifold at this point. This complies with our implied assumption- a small change in the physical parameters vector reflects, in turn, a slight change of the corresponding feature vector (whereas big variation in the values of the location and velocity leads to completely different feature vector).

Since the structure of the manifold is unknown, the geodesic distance, which is the locally shortest path on the manifold, cannot assess the affinity between the feature vectors. The Euclidean distance, which is equal to the geodesic distance for flat/linear manifolds only, cannot evaluate affinity reliably for large scales. Yet, Mahalanobis distance, which is a popular choice in several papers due to its virtue of approximating the Euclidean distance between the corresponding physical parameters vectors, suffers from several fundamental flaws. Thus, for complying with manifold's nonlinear and unknown structure, affinities in local vicinities are measured using Euclidean distance, whereas greater distances are omitted.

Accordingly, we compose an affinity matrix of all training feature vectors and execute eigendecomposition. The eigendecomposition provides us a mapping from the high-dimensional observable space to the low-dimensional embedded space. The low-dimensional embedded space is governed by the parameters controlling the system, and resembles the original parametric space up to a monotonic distortion.

Given a test set of unlabeled observations, we seek to find their low-dimensional embedding as well. However, rather than another spectral decomposition, we exploit the Nyström method for extension. The extension techniques of former

works were based on either additional mathematical relations in case of Mahalanobis distance-based diffusion kernel, or straightforwardly in case of Euclidean distance-based diffusion kernel. However, the latter leads to a scaled and inaccurate extended manifold structure due to a combination of affinity calculation with respect to all training observations (not close area only) and the absence of normalization. On the other hand, our extension is free from these issues.

At last, we determine the recovery of the location and velocity according to the midway point of the trajectory, which is formed by the movement of the source during the time interval. Based on the midway point, the approximation error of the trajectory of the source by a linear segment, is minimized. Consequently, the unknown location and velocity of the source are estimated as a weighted linear interpolation of the location and velocity values of all its nearest training neighbors on the embedded low-dimensional manifold.

Accordingly, the relevant accuracy measure, suggested in previous works, is modified and extended for the combination of physical quantities of different units, all the more so due to the movement of the sources.

Chapter 4

Experimental Results

4.1 Introduction

In this chapter we demonstrate the capabilities of the proposed passive single-sensor localization algorithm for recovering the location of an acoustic source, in three cases:

1. A baseline case of sources that have a stationary location, which consists of one-dimensional (i.e., azimuth) and two-dimensional (i.e., radius and azimuth) subcases.
2. An extensive case of deterministically moving sources, composed of various experiments, where we recover the velocity as well.
3. A case of Brownian motion induced moving sources, where we aim to recover the location and velocity of the sources in spite of a violation of a fundamental assumption regarding the velocity.

In the first stationary subcase, we confirm our choice of an affinity measure between the observations, followed by the recovery of the azimuth angle. In addition, we compare our results to the counterpart results of the prior diffusion maps based single-sensor source localization work [10]. In the second stationary subcase, we extend our experiment for estimation of the radius as well.

In the second case, which is the main contribution of this research, we allow deterministic movement of the sources and retrieve their location and velocity. We examine in detail the sensitivity of the proposed algorithm to different hyperparameters (e.g., training set size, number of features and frame length), variables (e.g., speed and direction) and conditions (e.g., signal to noise ratio, reverberation time and environmental conditions changes) by various experiments.

In the third case we introduce random and rapid perturbations to the movement of the sources, in a varying degree. We examine the capability of the proposed algorithm to deal with these perturbations, which represent the influence of environmental conditions on the movement of the sources.

The structure of the rest of this chapter is as follows. In Section 4.2, we describe the simulated setup used for conducting the experimental study. In Section 4.3, we present the baseline case of stationary sources and its results. In Section 4.4, we present the extensive case of deterministically moving sources, which is the main contribution of this research, and its results. In Section 4.5, we present the case of Brownian motion induced sources and its results. At last, the conclusions of the chapter, regarding the highlights of the experimental results, are presented in Section 4.6.

4.2 Simulated Setup

We describe the simulated setup used for conducting the experimental study, using an efficient implementation [37] of the image method [38]. In all experiments room dimensions were set to $6 \times 5.8 \times 3$ [m³], and an omnidirectional microphone was located at (3, 1, 1.8) m. The reverberation time of the room was defined as $T_{60} = 0.3$ sec (Except otherwise stated), simulating moderate reverberation conditions. In each location of the source, 1 sec (Unless else noted) long signal of a zero-mean and unit-variance (for neglecting system's gain) WGN, sampled at $f_s = 16$ kHz, is transmitted from the source, and after its convolution with the AIR it is measured at the microphone. Consequently, we acquire total of $m + M$ observations, where m out of them are randomly selected for the training set, while the remaining M samples are allocated for the test set. The corresponding autocorrelation-based feature vector of each observation consists of $D = 800$ lags (If not otherwise specified. The choice of its noted value will be justified later).

4.3 Stationary Sources

4.3.1 One Dimensional Subcase

Former single-sensor source localization results [10] have been achieved using Mahalanobis distance-based diffusion kernel, focusing on one-dimensional stationary scenario. On the other hand, various Euclidean distance-based diffusion kernels have been exploited for source localization by multiple-sensors [12, 13, 18]. Therefore, first of all and prior to the movement case, we have to test our setting and customized choice of Euclidean distance-based diffusion kernel in a station-

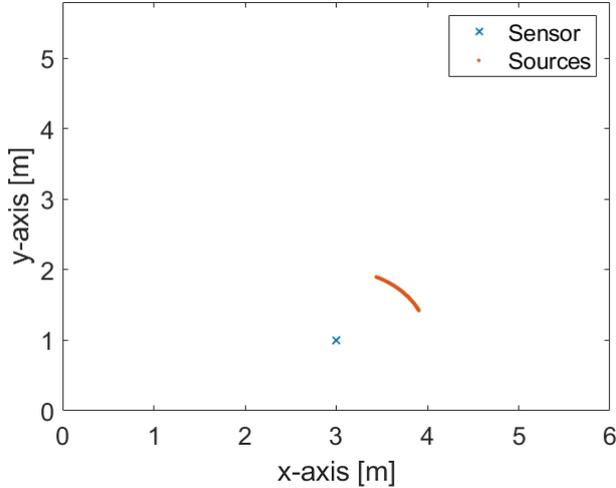


Figure 4.1: Room setup of the stationary one-dimensional subcase (top view). The blue cross represents the location of the sensor, whereas the red dots denote the location of the sources.

ary case. We aim to validate the ability of the Euclidean distance-based diffusion kernel to organize the observations according to their DOA (i.e., azimuth) values, in comparison to the results of [10]. In order to do so, a similar stationary experiment has been conducted by positioning all sources at a radius of 1 m from the microphone, and at azimuth angles which were drawn according to a uniform distribution $U[25, 64]^\circ$. The allocation of the locations of the sources has formed an arc, as depicted in Fig. 4.1. The experiment was carried out by generating training and test sets of 720 observations each (i.e., $m = M = 720$).

For considering a metric between the feature vectors which reflects their physical adjacency properly, we compare between two optional distance measures: Euclidean distance and diffusion distance (3.17). The Euclidean distance is defined by:

$$D_{\text{EUC}}(\mathbf{c}_i, \mathbf{c}_j) = \|\mathbf{c}_i - \mathbf{c}_j\|. \quad (4.1)$$

Note the examined diffusion distance (3.17) is derived from the customized Euclidean distance-based diffusion kernel.

Fig. 4.2(a) illustrates a comparison of the Euclidean distance and the diffusion distance according to the azimuth angle, which were measured among all feature vectors with respect to a reference feature vector, representing the minimal azimuth angle (25°). The depicted Euclidean distance is normalized by 1000, for a clear presentation of the behavior of both graphs together. By inspecting the Euclidean distance, we notice it lacks the mandatory property of

monotonicity with respect to the azimuth angle, and thus it cannot be used as a distance function between our feature vectors. However, since the Euclidean distance holds monotonicity in the range of approximately 1.85° from the reference angle, we deduce it is adequate for short arcs. Furthermore, in a broader perspective, we can conclude the adequacy of the Euclidean distance is limited to the vicinity of each feature vector, only. Nevertheless, the Euclidean distance can be exploited as part of our diffusion kernel, from two reasons. First, by restricting the computation of the Euclidean distance to small neighborhoods only we can benefit from a reliable affinity measure. Second, the Gaussian kernel is characterized by an inherent locality nature due to its scaling parameter.

In contrast to the Euclidean distance, the diffusion distance is characterized by monotonic behavior throughout the entire azimuth range, thus proving it is a suitable metric for quantifying affinity between our feature vectors. In order to produce the diffusion distance, we use $k = 20$ nearest-neighbors and a scaling parameter of $\varepsilon = 1.45 \cdot \text{median}$ for constructing the diffusion kernel. We assume that only the first eigenvector of the embedding is sufficient (i.e., $d = 1$), since the azimuth is the only varying controlling parameter of the system- this decision will be later validated.

The aforementioned comparison results confirm the nonlinearity of the manifold, which is characterized by a relatively flat surface in the vicinity of each observation, such that this surface looks like a linear Euclidean space. Moreover, these results comply with the counterparts of [13,18], which have been acquired by a similar Euclidean distance-based diffusion kernel for a corresponding stationary one-dimensional scenario. However, the results of [13,18] have been obtained by using dual-sensors, a different feature vector and a dissimilar extension for the eigenvector, as described in detail in Section 3.4.2.

Fig. 4.2(b) elaborates on the diffusion mapping of the observations by illustrating the first eigenvector ψ_1 as a function of the azimuth angle. In other words, it shows a comparison of observations representations between two parametric spaces: the one-dimensional location of the sources (i.e., azimuth) and their corresponding independent controlling parameter in the embedded manifold (i.e., ψ_1). We witness that the diffusion mapping follows the azimuth successfully (even linearly for most of the range), up to a monotonic distortion. As a result, the diffusion mapping is capable of accurately discovering the underlying independent parameter dominating the system, that is the location of the source. Moreover, since only the first eigenvector of the embedding is sufficient for monotonic organization of the observations with respect to the azimuth, the choice of $d = 1$ for estimation of the diffusion distance is approved. Hence, the diffusion distance, which approximates the geodesic distance, is accomplished for measuring the distances along the manifold correctly, as well as quantifying

Table 4.1: One-Dimensional Stationary Case- Comparison

Parameter	Ours	Talmon <i>et al.</i> [10]
Diffusion Kernel Type	Euclidean distance-based	Mahalanobis distance-based
Training Set Size	720	420 (5220, including local observations)
Angles Range [°]	39	60
Azimuth Angles Allocation	Uniform distribution over the entire range	60 Predefined angles, 1° apart.
RMSE [°], by (2.9)	0.094	1.2

the physical adjacency between the locations of the sources. These results support the embeddings results, originated from similar one-dimensional scenarios (yet following the aforementioned fundamental framework differences) that were presented in [10, 12, 13, 18] and are depicted in Fig. 2.1- all reflect monotonicity with respect to the azimuth. Note that the latter three, which are based on multiple-sensors, do not refer to the extended entries of the eigenvector in their provided embedding results. Our embedding results, on the other hand, prove our suggested extension is accurate, following the arguments mentioned in Section 3.4.2.

Encouraged by the successful revelation of the latent independent governing parameter of the system by the diffusion maps, we head to estimation of the locations of the unlabeled test set. By interpolating the location of the $\tilde{k} = 3$ nearest training neighbors on the manifold, we have established a minimal RMSE of 0.094° . For comparison, the single-sensor source localization algorithm presented in [10], which consists of a Mahalanobis distance-based diffusion kernel, has obtained minimal RMSE of approximately 1.2° in attempt to restore the location out of 60 possible predefined (up to perturbations) angles by using 420 training observations (Note that each training observation was equipped with 10 additional observations for estimation of the local covariance matrix). Both RMSE values were calculated according to (2.9), as proposed in [10] and mentioned in Section 2.1- in fact, this is the unnormalized version of (3.20) (the norm signs are equivalent to the absolute value for a one-dimensional location). We summarize our results in Table 4.1 and compares them to the counterpart by [10], along with fundamental settings differences.

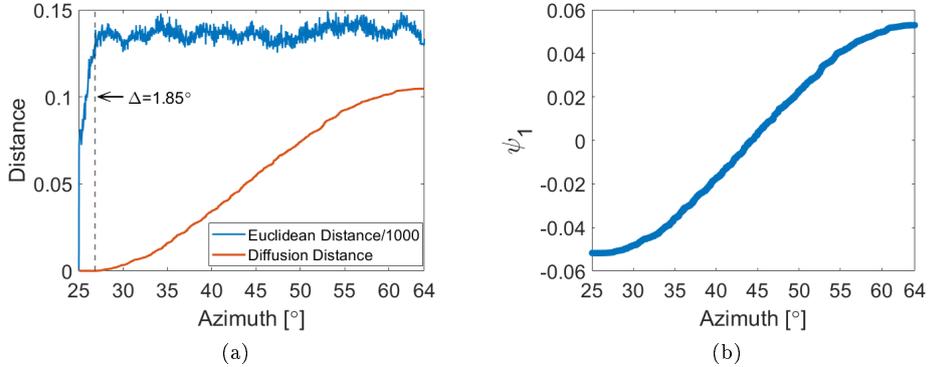


Figure 4.2: (a) Comparison of the Euclidean distance (blue) and the diffusion distance (red) according to the azimuth angle. The distances are measured among all feature vectors with respect to a reference feature vector, representing the minimal angle (25°). The dashed line represents the range of monotonicity maintained by the Euclidean distance. (b) Embedding of the observations as a function of the azimuth angle.

4.3.2 Two Dimensional Subcase

Following the confirmation of the choice of an affinity measure between the observations, the successful azimuth angle recovery and the comparison to known counterpart results from the literature [10], we can add a second degree-of-freedom to our system. We aim to challenge the proposed algorithm in recovering the two-dimensional location of stationary sources, where both their radius and azimuth are unknown. In order to do such an extension, we first examine the feasibility of the diffusion framework in organizing the observations monotonically, according to the values of the two hidden independent controlling parameters of the system: the radius and the azimuth. Similarly to the one-dimensional subcase, we arrange our stationary sources at azimuth angles according to a uniform distribution $U[25, 64]^\circ$. However, in contrast to the former subcase, we allow variability of the radius values according to a uniform distribution $U[1, 1.3]\text{m}$, as shown in Fig. 4.3. The experiment was carried out by acquiring training and test sets of 2880 observations each.

With an eye towards generation of the diffusion maps, we use $k = 10$ nearest-neighbors and a scaling parameter of $\varepsilon = 1.38 \cdot \text{median}$ for constructing the kernel. Since we have added a second varying independent controlling parameter to the system, we assume that only the first two eigenvectors of the embedding are adequate (i.e., $d = 2$). Figure 4.4 depicts the diffusion mapping of the observations of the entire dataset into the embedded manifold. The coloring patterns of the embedded observations, color-coded according to their radius (a) and az-

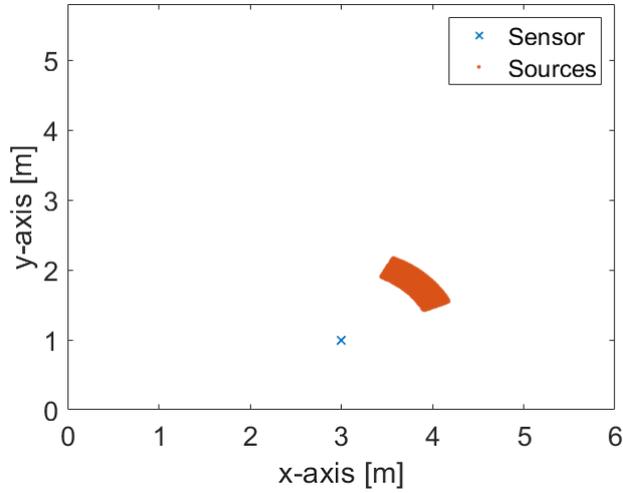


Figure 4.3: Room setup of the stationary two-dimensional subcase (top view). The blue cross represents the location of the sensor, whereas the red dots denote the location of the sources.

imuth (b) values, behave monotonically by each of the location coordinates, thus implying the diffusion mapping perceives the latent parameters dominating the system. Similar results have been obtained for Cartesian location coordinates.

Subsequently, we recover the unknown location of the test observations, by interpolation, using $\tilde{k} = 2$ nearest training neighbors on the manifold. We have established a minimal RMSE of 0.0105, which consists of 0.0041 for radius and 0.017 for azimuth. Furthermore, the localization performance of the proposed algorithm is portrayed in Fig. 4.5, by a polar plot of 35 sources, randomly picked from the test set.

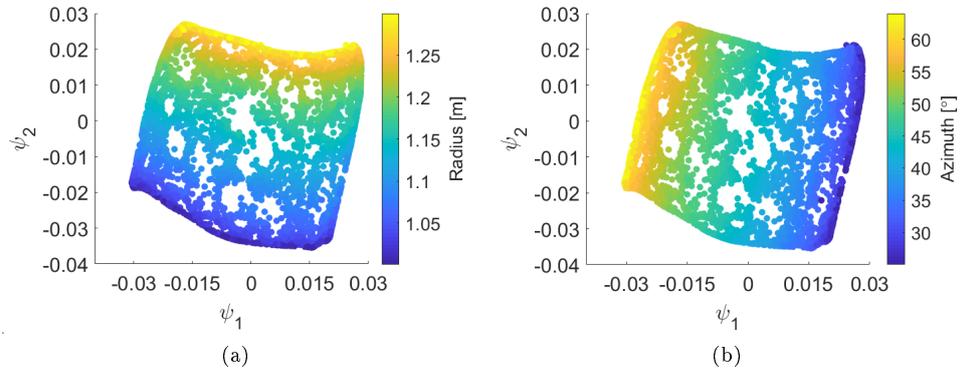


Figure 4.4: Diffusion mapping of the observations of the entire dataset into the embedded manifold, color-coded according to: (a) radius values and (b) azimuth values.

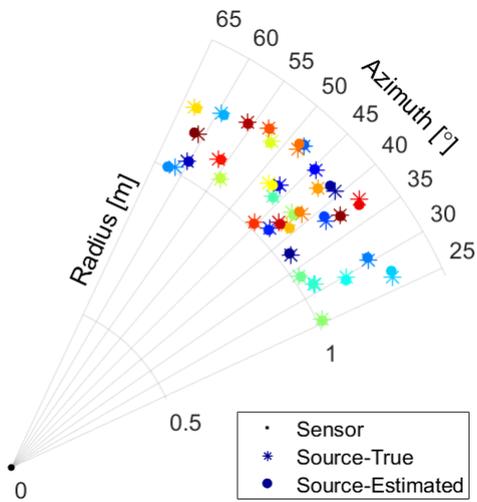


Figure 4.5: Polar plot of 35 randomly selected sources out of the test set. The black dot denotes the location of the sensor, the colored asterisk and disk pairs denote the true and estimated locations, respectively, of each source.

4.4 Deterministically Moving Sources

Following the stationary case, composed of the one-dimensional subcase as a baseline and the two-dimensional subcase as an extension, we are capable of proceeding to the main goal of our research- recovery of the location and velocity of deterministically moving sources.

In this section we delve into details and describe by various experiments the different hyperparameters (e.g., training set size, number of features and frames length), variables (e.g., speed and direction) and conditions (e.g., signal to noise ratio, reverberation time and environmental conditions changes) influencing the performance of the proposed algorithm for the case of moving sources (free of Brownian motion).

For isolating the different impacts of the elements composing the movement of the sources, as well as the factors affecting the performance of the localization algorithm under various circumstances, we simplify the scenarios by focusing on a sole degree-of-freedom system in all of our movement scenarios (i.e., $d=1$). Not only it supports conclusions which are relevant regardless the amount of degrees-of-freedom and even viable for more sophisticated cases, but it also allows us working with significantly smaller dataset compared to scenarios of higher degrees-of-freedom. Scenarios where all parameters of the location and velocity are independent are out of the scope of this work.

Even though only a single controlling parameter is independent, it dictates variations of another two parameters along the movement of the sources, such that it results in an estimation error in each of them. Since the relation between the errors of the independent parameter and the dependent parameters is nonlinear, the errors of the dominated parameters are calculated explicitly, and consequently are taken into consideration at the calculation of the total RMSE.

We assume, for simplicity, that the sources move linearly- meaning the drift term is constant, i.e., $a(\mathbf{p}(t)) = \begin{bmatrix} v_{\tilde{x}} \\ v_{\tilde{y}} \end{bmatrix}$.

In order not to exhaust the reader with dozens values of k, \tilde{k} and ε , due to several simulations composing each movement experiment, we note that both k and \tilde{k} range from 10 to 50 neighbors ($\tilde{k} \leq k$), while the proportion of ε to the median ranges from 1.41 to 1.45.

4.4.1 Sensitivity to Training Set Size

We examine the performance of the proposed localization algorithm with respect to the size of the training set. For that purpose, we position all sources at a distance of 1 m from the microphone and at an azimuth angle of 45° . Their movement is initialized with a speed of 0.5m/sec, at directions of movement

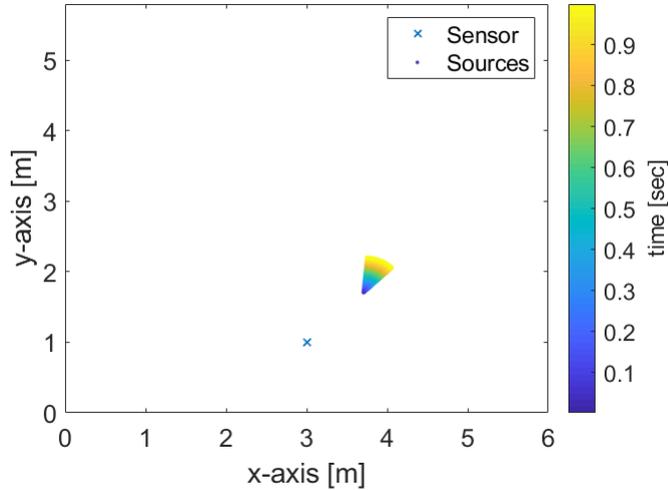


Figure 4.6: Room setup and the evolution of the locations of the sources through time of the training set size experiment (top view). The blue cross represents the location of the sensor, whereas the color coded dots denote the location of the sources. The color represents the time. Produced using training and test sets of 720 observations each.

which are drawn according to a uniform distribution $U [45, 85]^\circ$. Figure 4.6 illustrates the setup and the evolution of the locations of the sources through time, using training and test sets of 720 observations each, for instance. The simulation is repeated each time with a different amount of training observations, which varies from 120 to 2880. We maintain a ratio of 1 between the sizes of the test set and the training set throughout the entire experiment.

As seen in Fig. 4.7, the total accuracy increases with respect to additional training observations, up to marginal improvement for significantly larger sizes. Moreover, an insufficient training set size leads to sparse manifold, which fails to reliably represent the widely spanned controlling parameter, and as a result ends in a considerable error. Note the presented RMSE is an average of the individual RMSE measures of the controlling parameter (i.e., direction) and the dominated parameters (i.e., radius and azimuth).

4.4.2 Sensitivity to Number of Features

We test the sensitivity of the proposed algorithm to the length of the feature vector. In order to do so, we repeat the settings of the previous experiment, except two adjustments. First, instead of a varying training set size, we generate a training set and test set of 720 observations each. Second, we modify the number of features in each simulation, in the range of 50 to 5600 dimensions.

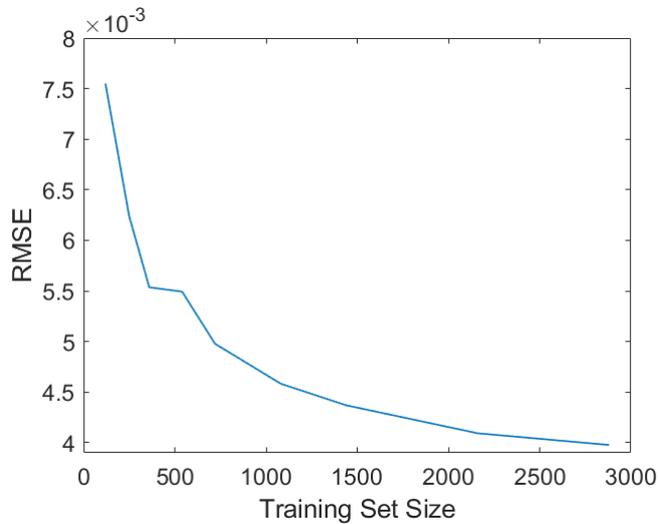


Figure 4.7: Performance of the proposed algorithm as a function of the training set size

Fig. 4.8 illustrates the total estimation error in dB units, in order to emphasize the behavior of the graph despite the enormous value achieved for small number of features, which deviates significantly from the other values, all the more so being bigger than the lowest error in more than an order-of-magnitude. These estimation error results reflect the infamous curse of dimensionality, and particularly the peaking phenomenon [39]. First, the poor accuracy obtained for small number of features is caused by an insufficient resolution which prevents us from distinguishing between feature vectors referring to different locations and velocities. As the dimensionality of the feature vectors hyper-space increases, the additional features allows us to describe the location and velocity in finer details, thus the recovery improves. However, at some point the dimensionality increase becomes counterproductive and the performance of the estimation accuracy starts deteriorating, since the volume of the high dimensional space grows exponentially with respect to the dimensionality. Consequently, the given, fixed-sized, training set turns to sparse, which questions the feasibility of finding the nearest neighbors reliably at the stage of the construction of the affinity matrix. Generally, by increasing the size of the training set, its sparsity in the hyper-space can be mitigated- all the more so, infinite-sized training set will result in gradual improvement with respect to the dimensionality. Note that the optimal results are accomplished for $D = 800$ lags, thus justifying our choice of this value as the length of the feature vectors in all other experiments.

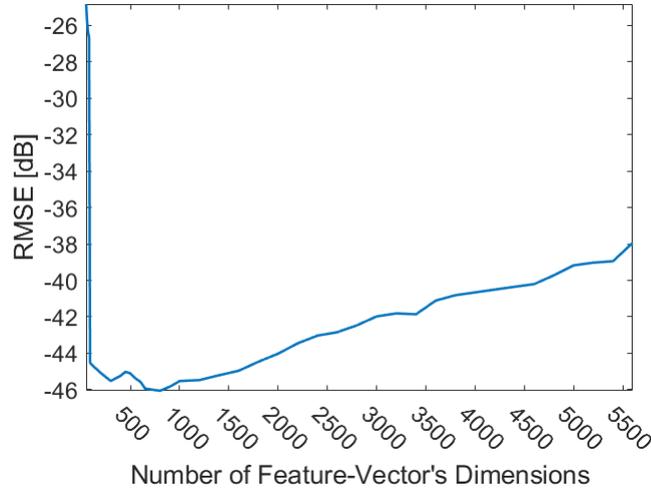


Figure 4.8: Performance of the proposed algorithm as a function of the length of the feature vector

4.4.3 Sensitivity to Speed

We present the accuracy of the proposed localization algorithm with respect to the speed of the moving sources. For that objective, we repeat the settings of the previous experiment with a slight change: the length of the feature vector is fixed to $D = 800$ lags, and the speed of the sources is changed in each simulation, in the range of 0.0625 to 1 m/sec.

Fig. 4.9 depicts the total estimation error, as well as the individual estimation errors of the controlling parameter of the scenario (i.e., direction) and the dominated ones (i.e., radius and azimuth). Due to the high variability of the direction its estimation error dictates the behavior of the total error throughout the entire experiment. We yield a high estimation error of the direction for very slow speed values thanks to the struggle of perceiving variations between the different directions during a bare movement. That struggle, in turn, results in a more scattered embedding (e.g., as can be seen in Fig. 4.10(a) of the scenario of 0.0625 m/sec, as opposed to a “fine” embedding as in Fig. 4.10(b) of the scenario of 0.5 m/sec. Further details and aspects will be discussed in the next experiment) and thus in a challenge for distinguishing correctly between the various direction values. As the speed gets faster, the accuracy of the estimated direction improves due to a more meaningful movement by the sources, up to a point where the accuracy starts decreasing. The accuracy of the estimated direction starts deteriorating for fast sources, due to two factors. First, greater speeds lead to sparser, yet clustered manifolds. That, in turn, makes

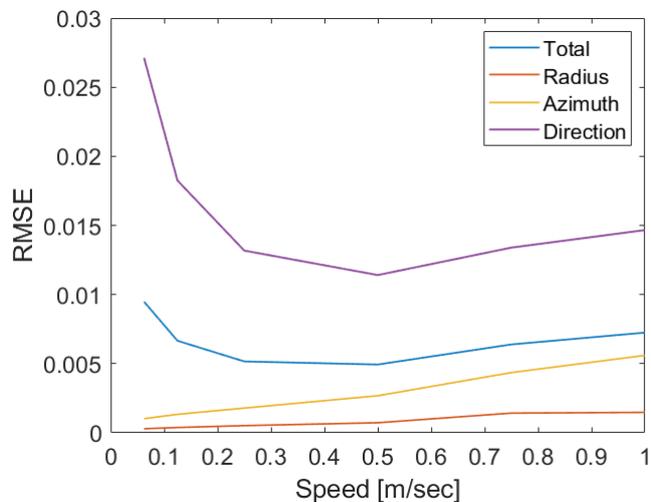


Figure 4.9: Performance of the proposed algorithm as a function of the speed of the sources

the estimation of the direction, based on interpolation of the nearest neighbors, more challenging- meaning the variations in the movement of the sources are too fast to be distinguished. In addition, the quasi-stationarity assumption is revoked, hence our autocorrelation-based feature vector, implemented by MATLAB's 'xcorr' function, is no longer valid as the function implicitly assumes WSS signal.

Regarding the radius and the azimuth, as speed gets faster, a wider range of possible location values is obtained, which consequently results in larger estimation errors.

4.4.4 Sensitivity to Direction

We examine the accuracy of the proposed localization algorithm with respect to the direction of movement of the sources. For that purpose, we position all sources at a distance of 1 m from the microphone and at an azimuth angle of 45° . The movement of the sources is initialized with speed values which are drawn according to a uniform distribution $U [0.25, 0.5]$ m/sec (based on the range which has produced the best accuracy results in the previous experiment), at the examined direction value. The simulation is repeated each time with a different direction of movement of all sources, varying from 5° to 90° , using training and test sets of 720 observations each. Figure 4.11 illustrates the setup and the midway point of the trajectories of the sources, with respect to all scenarios of this experiment.

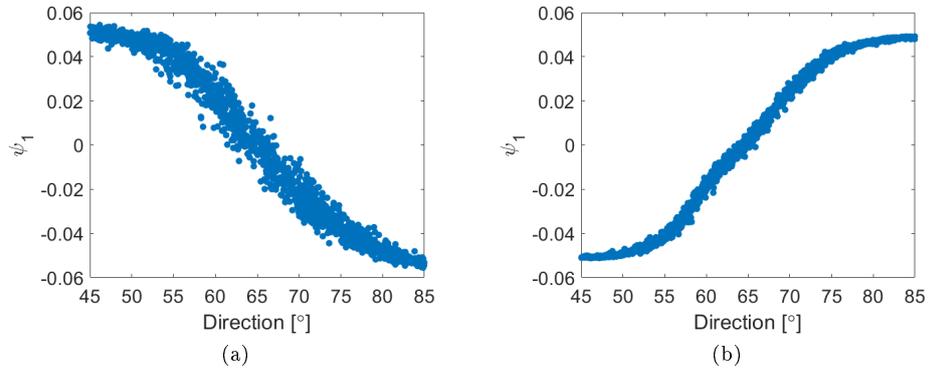


Figure 4.10: Comparison of diffusion mappings of the observations, according to the direction, between different speed scenarios. (a) 0.0625 m/sec scenario. (b) 0.5 m/sec scenario.

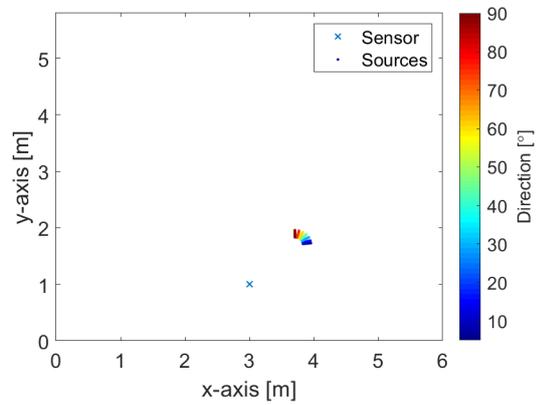


Figure 4.11: Room setup and the midway point of the trajectories of the sources, with respect to all scenarios of the direction experiment (top view). The blue cross represents the location of the sensor, whereas the color coded dots denote the midway point location of the sources. The color represents the scenario (i.e., direction).

Fig. 4.12 describes the total estimation error, as well as the individual estimation errors of the controlling parameter of the scenario (i.e., speed) and the dominated ones (i.e., radius and azimuth). Not only the speed dictates the behavior of the total estimation error throughout the entire experiment due to its high variability, but its estimation error graph is also followed by the radius estimation error.

As for the azimuth, its estimation error is minimal for a direction of 45° since the azimuth is identical to the direction through the entire movement by the sources (The initial azimuth angle, prior movement, is 45°), which leads to a degenerated scenario where there is no variability in the azimuth values. As the direction gets away from 45° , the range of possible azimuth values becomes bigger and thus its estimation error aggravates with respect to the gap from the direction of the degenerated scenario.

The inconsistent behavior of speed's estimation error can be explained by a varying extent of spatial aliasing throughout the experiment, due to the changing direction and the symmetry of the room, as illustrated in Fig. 4.13. Figures 4.13(a),(c),(e) present the original rows of the affinity matrix, the additional rows of the affinity matrix and the interpolation coefficients matrix, respectively- all associated with a typical case of spatial aliasing from this experiment, represented by the scenario of the direction of 45° . Both rows and columns of these matrices are sorted according to the speed of the sources. Whereas the first matrix represents the nearest neighbors involved in the stage of parameterization of the manifold (using training set only), the second matrix represents the nearest training neighbors associated with the stage of manifold extension for the test observations. In addition, the third matrix represents the nearest neighbors in the embedded manifold involved in the interpolation for recovering the unknown controlling parameter. All matrices are characterized in an utterly scattered pattern, meaning the nearest neighbors are allocated arbitrarily and not according to the true physical adjacency. This spread allocation of nearest neighbors, or in other words spatial aliasing, translates into the difficulty of distinguishing between sources, which move at different rates but share the same direction.

On the other hand, Figs. 4.13(b),(d),(f) demonstrate the original rows of the affinity matrix, the additional rows of the affinity matrix and an interpolation coefficients matrix, respectively- all associated with a typical case free of spatial aliasing, represented by the scenario of the speed of 0.5m/sec from the previous experiment. Both rows and columns of these matrices are sorted according to the direction of the sources, which is the controlling parameter of the previous experiment. As opposed to the spatially aliased case, these matrices are characterized by well defined patterns, which consist of physically adjacent neighbors

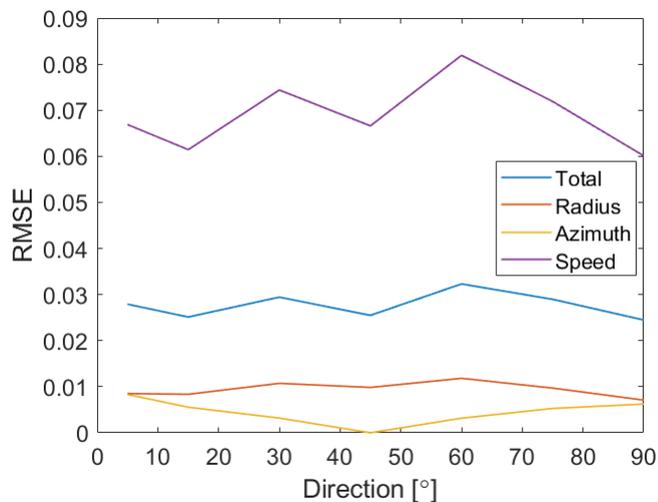


Figure 4.12: Performance of the proposed algorithm as a function of the direction of movement of the sources

and thus resemble thick streaks.

Furthermore, the outcome of the spatial aliasing is expressed by significantly higher estimation errors, compared to the other experiments, and can be slightly mitigated by a considerable addition of training observations (e.g., a larger training set which its size is 5 times the original size, has improved the accuracy of speed and the total accuracy by nearly 11%).

4.4.5 Sensitivity to Signal to Noise Ratio

We examine the performance of the proposed localization algorithm with respect to the signal to noise ratio (SNR). In order to do so, we repeat the setting of the speed experiment with a slight change, by fixing the speed of all sources to 0.5m/sec. In addition, in each individual simulation we introduce additive white Gaussian noise (AWGN) of a specific variance value to all signals received by the microphone. The simulation is repeated each time with a varying degree of SNR, ranging from 0 to 30 dB, followed by a scenario free of noise, or in other words an infinite SNR.

Fig. 4.14 illustrates the total accuracy of the proposed algorithm for various SNR conditions. As expected, as the conditions get harsher, the estimation error grows. Note that although the estimation error is significantly high for SNR of 0 dB, it is not an order-of-magnitude higher compared with the other conditions. The reason for that is the use of WGN as a speech signal, thus all frequencies of AIR's transfer function come to realize in the signals received by

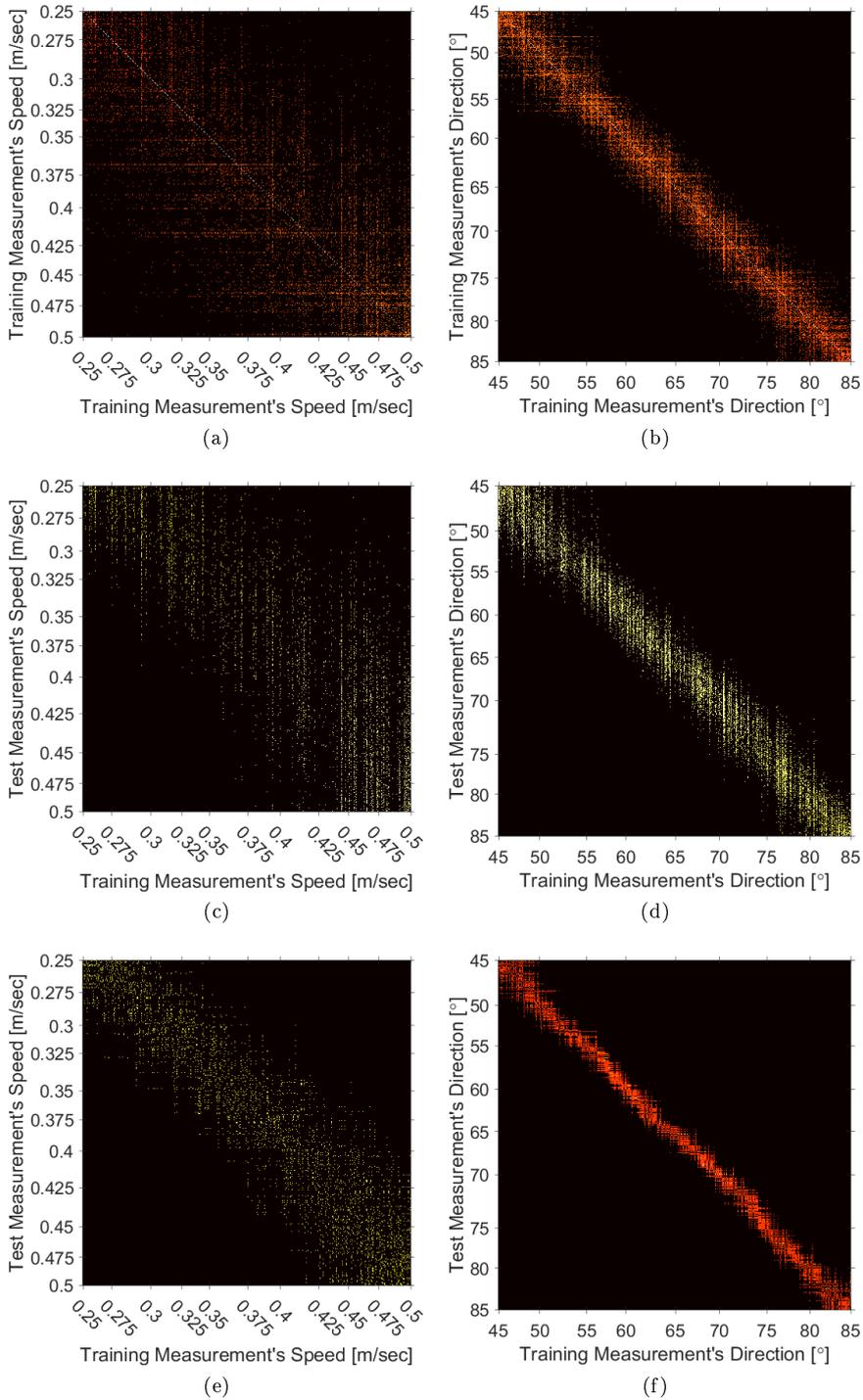


Figure 4.13: Presence of spatial aliasing. Original rows of the affinity matrix (training set) of: (a) spatially aliased case, (b) non-spatially aliased case. Additional rows of the affinity matrix (test set) of: (c) spatially aliased case, (d) non-spatially aliased case. Interpolation coefficients matrix of: (e) spatially aliased case, (f) non-spatially aliased case.

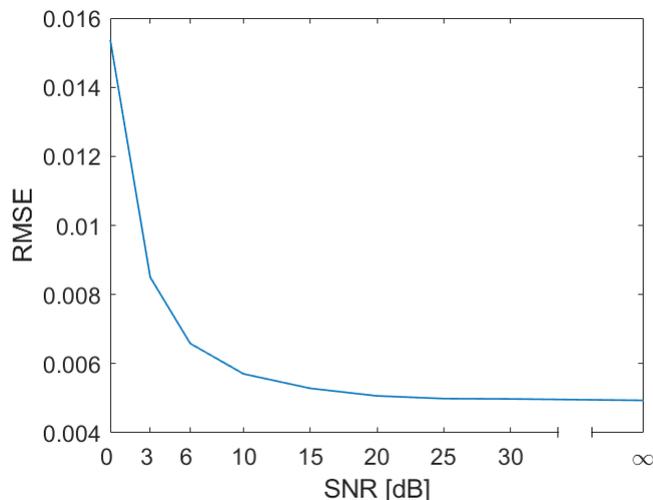


Figure 4.14: Performance of the proposed algorithm as a function of SNR

the microphone, and as a result there is no a fundamental difference between the embedded manifolds of the various conditions. In addition, from SNR of 20 dB onward, the accuracy improves insignificantly and tends to the accuracy of the case of no noise at all (i.e., $\text{SNR} \rightarrow \infty$).

4.4.6 Sensitivity to Reverberation Time

We examine the performance of the proposed single-sensor localization algorithm with respect to the reverberation time. For this purpose, we repeat the setting of the speed experiment with a slight change, by fixing the speed of all sources to 0.5m/sec. The simulation is executed each time with different reverberation time, ranging from 0.128 to 1 sec.

Fig. 4.15 depicts the total estimation error of the algorithm for various reverberation time values, in the blue graph. The estimation errors are displayed in dB units for emphasizing the small yet significant differences between close estimation error values. For nearly no reverberation the estimation error is colossal. As the reverberation time gets longer the accuracy improves gradually, up to a point of moderate reverberation (0.4 sec), where the performance of the algorithm starts deteriorating steadily. These results validate the implied hypothesis of our work. In contrast to source localization using multiple sensors, where the recovery of the location is based on the direct propagation path of the signal and the reflections are in fact its Achilles' heel, these reflections are essential for source localization using a sole sensor. In addition, by relying solely on the direct direction of arrival, the single-sensor algorithm is incapable of dis-

tinguishing between the different locations and velocities of the sources, due to lack of information needed for an accurate estimation. Traditionally, the lack of information is resolved by adding sensors, which provide all together a precise estimation. However, by exploiting these reflections we can tackle this inadequacy and compensate for the inherent flaw of the single-sensor. On the other hand, similarly to source localization using multiple sensors, the performance of a sole sensor suffers from over-reverberation, but more robustly.

4.4.7 Sensitivity to Environmental Conditions Changes

An outcome of the last experiment provides us the opportunity to inspect the influence of environmental conditions changes, that take place between the training and the test stages, on the performance of the proposed algorithm, as can be seen in the red graph of Fig. 4.15. In the first stage we generate a training set with fixed reverberation time of 0.4 sec, followed by generation of several test sets, where each one is according to different reverberation time. As a result, two consequences are obtained. First, as the reverberation level of the test stage gets away from reverberation time of 0.4 sec, which has prevailed during the training stage, the estimation error worsens. The second finding derives from a comparison between the estimation errors obtained in this experiment and the estimation errors acquired when ideally both training and test stages are held under the same reverberation conditions (as in the previous experiment). We notice that as the absolute difference in the reverberation time between the stages grows, the error deviates from the values achieved ideally- unless the reflections do not play a role in practice (e.g., as at 0.128 sec).

From manifold point of view, small changes in reverberation time results in moderate influence on the manifold structure in general, whereas considerable changes are associated with significant impact on the structure of the manifold. According to the extension, the mapping of the test observations to the embedded manifold is based upon the combination of the manifold structure, established by the training observations, and the nearest training feature vectors. Thus, these environmental conditions changes between the training and test stages introduce ambiguities to the extended manifold to an extent. These ambiguities consequently sabotage the efforts for an accurate recovery of the unknown locations and velocities of the test set, as achieved when both stages share the same reverberation time. In addition, opposite variations in the reverberation time at the test stage affect significantly different on the estimation results since the single-sensor is more vulnerable to gradual absence of reflections than presence of additional ones. That behavior is reasoned by prevention of essential information for the single-sensor that was previously available dur-

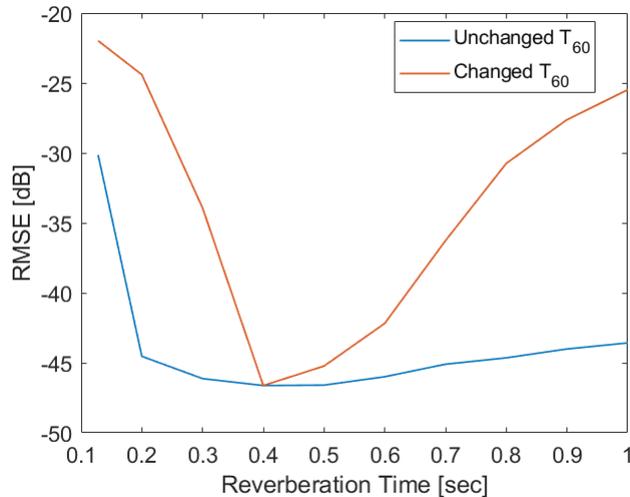


Figure 4.15: Performance of the proposed algorithm as a function of reverberation time: when both training and test stages share the same T_{60} (blue), and when trained with $T_{60} = 0.4$ sec (red).

ing the ideal training stage, as a result of environmental conditions changes. That absent information is responsible for establishing a manifold structure according to an increasing dominance of the direct propagation path component of the received signals. On the other hand, additional reflections following the environmental conditions changes do not necessarily provide us more crucial information, compared to what was achieved at the ideal training stage. The reason for that behavior is that reverberation time of 0.4 sec is mostly sufficient (in the sense of taking advantage of reflections) for training for single-sensor source localization.

However, note that when the localization is based mostly on direct propagation path of the signal, as at 0.128 sec, a conflicting outcome arises where the deviation in the estimation error due to environmental conditions changes is smaller despite the distinct trend. That outcome can be explained by an extensively substantial ambiguity in the original manifold formed ideally, when both training and test stages have shared the same reverberation time. Thus, this significantly ambiguous behavior initially, along with the incremental ambiguity due to environmental conditions changes, leads eventually to a relatively smaller impact on the performance.

4.4.8 Sensitivity to Frame Length

We examine the performance of the single-sensor localization algorithm with respect to the length of the time frame.

All previous experiments refer to a single frame. Thus, the recovery of the location and velocity of the source in these experiments is determined according to a single query point along source's path, following the discussion and derivation in Section 3.3.2 and in (3.21). Rather than executing the proposed localization algorithm once along the whole trajectory of the source, and consequently carrying an extrapolation throughout the path based on a long linear movement segment (Even though it may work well for a source that has a constant velocity), we can exploit calculus approach for better estimation results. According to this approach, the whole trajectory of the source, formed by its movement which is characterized by a slow speed and a gradually changing velocity, can be approximated by short linear movement segments. As a result, it allows obtaining a smaller average estimation error by an iterative execution of the algorithm in each frame.

Since the algorithm is repeated and executed in different points (i.e., query points) along the trajectory of the source, each point represents a non-overlapping frame (overlapping frames are out of the scope of this work, except Appendix A) which is associated with a unique short linear movement segment. Thus, each segment may hold different optimal hyperparameters (i.e., k , \tilde{k} and ε) values for each query point. We define the average RMSE along the trajectory, as an average of the different RMSE values (obtained in each of the frames) over the frames, as follows:

$$\overline{\mathbf{RMSE}} = \frac{1}{N} \sum_{r=1}^N \mathbf{RMSE}^{(r)}, \quad (4.2)$$

where $\mathbf{RMSE}^{(r)}$ is the RMSE of the r -th query point along the path, as defined in (3.21), and N is the total number of query points along the path (i.e., the total number of frames).

For examining the optimal frames length, we position all sources at a distance of 1 m from the microphone and at azimuth angles, which their values are drawn according to a uniform distribution $U [45, 85]^\circ$. The movement of the sources is initialized with a speed of 0.5 m/sec, at a direction of 45° , as illustrated in Fig. 4.16. The duration of the signal transmitted by each source is set to 2 sec (twice the duration in all previous experiments). The signals received by the sensor in each simulation are divided into time frames. The simulation is repeated each time with a different number of non-overlapping frames, varying from a one long frame (the whole received signals) to 20 short frames- meaning the frame length varies from 2 sec to 0.1 sec. The experiment is carried out by acquiring training and test sets of 720 observations each.

Fig. 4.17 depicts the average total estimation error of the algorithm for

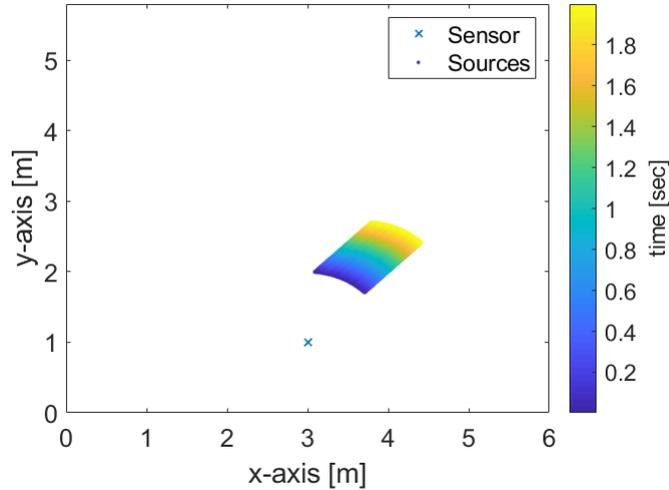


Figure 4.16: Room setup and the evolution of the locations of the sources through time for the experiment of the frame length (top view). The blue cross represents the location of the sensor, whereas the color coded dots denote the location of the sources. The color represents the time.

various duration values of the segments and their corresponding amount, in dB units. On the one hand, the approximation by a linear segment is inaccurate for long frames, unless the source has a constant velocity during the frame. This inaccuracy is expressed by an incremental error, which becomes worse as the frame gets longer- all the more so, when the trajectory of the source is nonlinear. Note that this result is obtained for sources that move linearly, which is the ideal case for such an approximation, and is evident even in such an ideal case. On the other hand, short frames are not good either, regardless the trajectory of the source, as they are unable to capture the movement properly, and thus incapable of distinguishing between the various azimuth values. We observe the optimal recovery accuracy is achieved by frames of 0.5 sec.

Figure 4.18 depicts the successful localization results through time of the unknown azimuth (i.e., controlling parameter) and radius (i.e., dominated parameter) of an arbitrary source from the test set, using the optimal frame length. Note that the speed and direction are degenerated in this experiment. In each frame, the linear segments of the estimated radius and azimuth are derived as an extrapolation, based on the estimated value of the location in the midway point of the segment. In contrast to the estimated radius which is completely precise through time, the estimated azimuth is slightly less accurate due to its high variability. Similar results have been received for all sources from the test set.

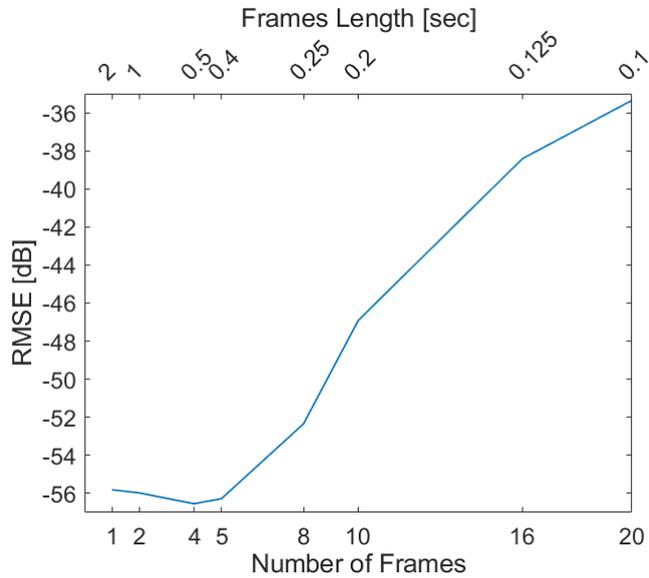


Figure 4.17: Performance of the proposed algorithm as a function of the frames length

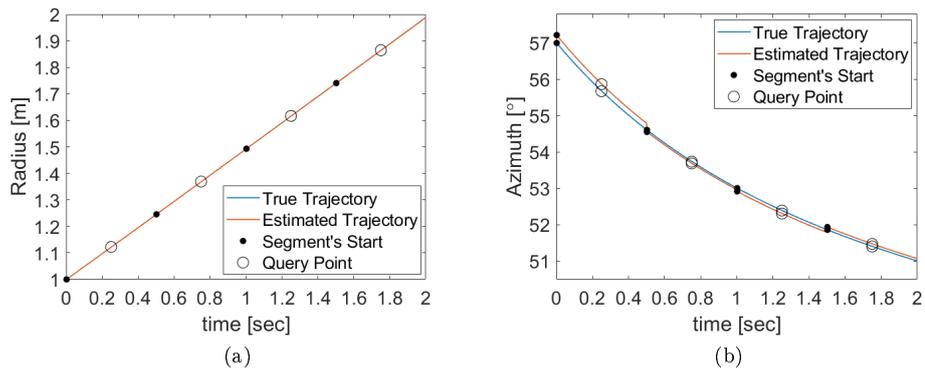


Figure 4.18: Localization results of a moving source through time of each of the unknown parameters. (a) Radius and (b) azimuth. Blue represents true values, red represents the estimated values. The black dot denotes the beginning of each segment, the black circle denotes each of the query points.

4.5 Brownian Motion Induced Moving Sources

In the last case, deterministically moving sources, we have examined the different hyperparameters, variables, conditions and limitations affecting the performance of the proposed algorithm by various experiments- all under the assumption of slow and gradually changing velocity components. Following the last case, which has ignored the Brownian motion term, we introduce it to the movement of the sources, according to (3.1).

We remind that these velocity restrictions, along with all the other assumptions characterizing the AIR and the problem setting, as fully noted in Section 3.3.2, are associated with small changes in the feature vectors along the time intervals. These small changes allow the hypothesis of a low dimensional manifold, and thus the recovery of locations and velocities of the sources.

In this section we examine the impact of violating the fundamental assumption of gradual changes of the velocity through time, by inspecting slow sources that change their velocity rapidly and randomly. In order to do so, we execute the worst-case scenario by using the same algorithm framework- for emphasis, the algorithm is executed just once (i.e., a single frame).

We recall that the Brownian motion is characterized by rapid and random perturbations, representing environmental conditions which present throughout sources movement (e.g., enclosure vibrations, source's mount vibrations, and wind). The introduction of Brownian motion term to sources movement describes a more realistic setting. In addition to the definition, mentioned in Section 3.2, the Brownian motion term is described by white Gaussian noise with a variance vector $\begin{bmatrix} \sigma_{\tilde{x}}^2 \\ \sigma_{\tilde{y}}^2 \end{bmatrix}$. The directional standard deviations are defined in direct proportion to the Brownian motion coefficient, as follows:

$$\begin{bmatrix} \sigma_{\tilde{x}} \\ \sigma_{\tilde{y}} \end{bmatrix} = \begin{bmatrix} \frac{v_{\tilde{x}max}}{\xi} \eta \\ \frac{v_{\tilde{y}max}}{\xi} \eta \end{bmatrix}, \quad (4.3)$$

where $v_{\tilde{x}max}$ and $v_{\tilde{y}max}$ are the maximal horizontal and vertical speed components drawn in the experiment, respectively, η is the Brownian motion coefficient, and ξ is the confidence factor. We define $\xi = 4$, such that under the constraint of four-sigma confidence level, 99.994% of the realizations are within the following range:

$$\begin{cases} -\eta \cdot v_{\tilde{x}max} \leq n_{\tilde{x}}(t) \leq \eta \cdot v_{\tilde{x}max} \\ -\eta \cdot v_{\tilde{y}max} \leq n_{\tilde{y}}(t) \leq \eta \cdot v_{\tilde{y}max} \end{cases}, \quad (4.4)$$

where $n_{\tilde{x}}(t)$ and $n_{\tilde{y}}(t)$ are the horizontal and vertical components of the Brow-

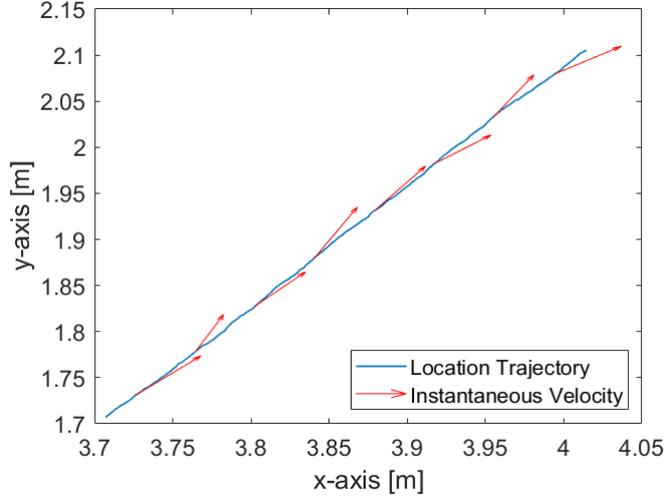


Figure 4.19: Brownian motion induced source trajectory example, for a coefficient of 0.8. Blue denotes the trajectory through time, the red vectors denotes the instantaneous speed and direction of movement of few arbitrary points along the trajectory.

nian motion term $\mathbf{n}(t)$, respectively, i.e., $\mathbf{n}(t) = \begin{bmatrix} n_{\tilde{x}}(t) \\ n_{\tilde{y}}(t) \end{bmatrix}$.

In order to examine the influence of the Brownian motion on the performance of the proposed algorithm, we repeat the setting of the speed experiment with a slight change, by fixing the speed of all sources to 0.5m/sec. In addition, in each individual simulation we introduce a Brownian motion term to the movement of all sources. The simulation is repeated each time with a different Brownian motion coefficient value, ranging from 0.05 to 0.8, followed by a scenario free of Brownian motion (as in the previous experiments). Figure 4.19 depicts the Brownian motion induced trajectory of a source, with respect to Brownian motion coefficient of 0.8, and the instantaneous speed and direction of few arbitrary points along the trajectory.

Note that the comment regarding the various values of the hyperparameters (i.e., k , \tilde{k} and ε), mentioned in Section 4.4, applies for the following analysis as well.

In all previous movement experiments, discussed in Section 4.4, we have focused in a single degree-of-freedom due to variability of a single random independent parameter. However, the introduction of the Brownian motion term to source's movement leads to the question what is the true number of intrinsic dimensions. It can be interpreted as either one (i.e., direction), as before. On the other hand, it can also be interpreted as either two (i.e., direction and speed) or

three (i.e., direction, radius and azimuth)- depending by the attribution of the additional variability to either the location or the velocity. In other words, the addition of Brownian motion term, according to (3.1), may lead to independent and random variability in another two parameters: the radius and the azimuth. However, since that can also be interpreted as variability in the velocity parameters instead, one may claim of just an additional degree of freedom- the speed (The direction is already a degree-of-freedom, due to the experiment setting).

Thus, Fig. 4.20 illustrates the total estimation error of the proposed algorithm for various Brownian motion conditions, with respect to three intrinsic dimension d cases: one, two or three. As expected, as the Brownian motion term gets more dominant, the estimation error grows, for all cases. We note two examples for emphasizing the colossal estimation error. First, we can witness that the estimation error for Brownian motion coefficient of 0.425 is an order-of-magnitude higher compared to the ideal case, where no Brownian motion term is introduced. Second, for Brownian motion coefficient of 0.8, the error is 2400% higher than in the ideal case. The reason for the high errors is the revoke of our assumption of approximating the trajectories by short linear segments, due to the rapid and random changes in the movement of the sources. These changes prevent us from collecting a sufficiently long enough observation and consequently a meaningful feature vector during each time frame.

Note the presented RMSE is an average of the individual RMSE measures of the controlling parameter (i.e., direction) and the dominated parameters (i.e., radius and azimuth) for the Brownian motion free scenario, whereas it is an average of all individual RMSE measures for the Brownian motion induced scenarios.

Regarding the determination of the intrinsic dimension value based on the various intrinsic dimensions cases, we note that the examined scenario is unambiguous upon the parameters vector. In addition, the Brownian motion adds small variability compared to the significant variability triggered by the original degree-of-freedom (direction). Moreover, there is no case which dominates all the others- all the more so, the difference between the results is negligible, as depicted in Fig. 4.20. Thus, the two additional degrees-of-freedom are redundant, and we determine $d=1$ as the dimension of the low-dimensional embedded manifold for yielding the embeddings and the estimation results.

For emphasizing the impact of the Brownian motion, we elaborate on the scenario of Brownian motion coefficient of 0.8. Figure 4.21 describes the diffusion mapping of the observations of the entire dataset, obtained from that scenario, into the embedded manifold, by illustrating the first eigenvector ψ_1 as a function of all four parameters vector coordinates: the radius (a), azimuth angle (b), speed (c) and direction (d). We witness that despite the significant interference

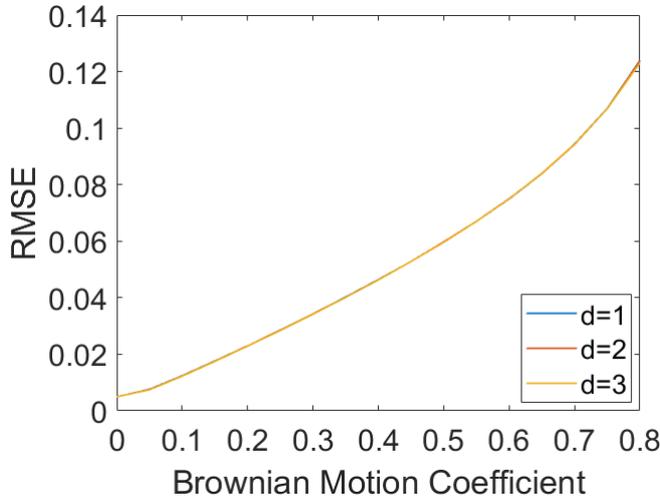


Figure 4.20: Performance of the proposed algorithm as a function of Brownian motion coefficient, for various intrinsic dimension cases

by the Brownian motion, the resulting small perturbations along the trajectory have minor influence on the location from two reasons. First, the diffusion mapping follows the azimuth successfully. Second, the diffusion map of the radius encounters minor ambiguities, due to the small range of possible radius values. On the other hand, the instantaneous velocity is heavily affected in each infinitesimal step of the source by the random and rapid nature of the Brownian motion. All the more so, the velocity is influenced by the significant variability of the direction (the degree-of-freedom). Consequently, both diffusion mappings of the speed and direction fail to maintain monotonic organization of the observations- meaning the instantaneous speed and direction cannot be accurately recovered (even for higher number of intrinsic dimensions).

We note that the individual RMSE measures, composing the total RMSE, correspond well with the embeddings depicted in Fig. 4.21. The individual RMSE values that are associated with the velocity (23.99% for speed, and 25.03% for direction) are colossal, where the ones of the location (0.24% for radius, and 0.32% for azimuth) are negligible.

Note that for small Brownian motion coefficients, the direction preserves monotonicity and unambiguous behavior successfully, whereas the speed is not- similarly to the conclusion from the direction experiment of Section 4.4.4: the algorithm struggles to distinguish between sources which share the same direction, but have different speed values, due to spatial aliasing. In addition, similar conclusions are deduced according to comparison of the embedding of the observations to their Cartesian location and velocity coordinates.

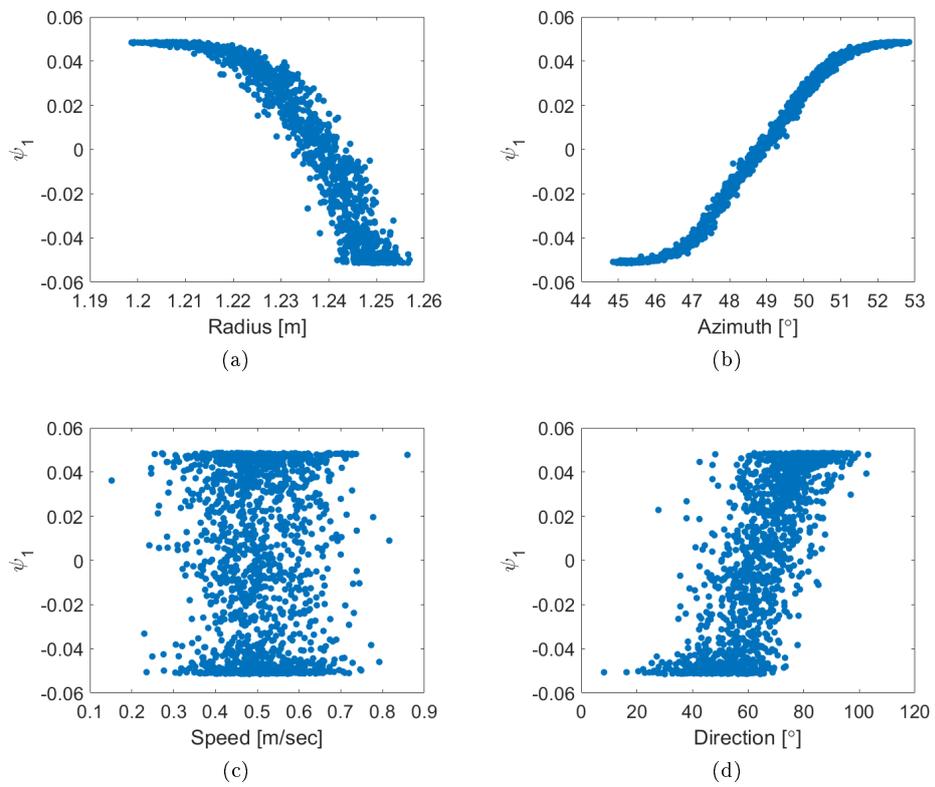


Figure 4.21: Embedding of the observations, obtained from the scenario of Brownian motion coefficient of 0.8, as a function of: (a) radius, (b) azimuth, (c) speed and (d) direction.

As a representative example of the aforementioned scenario of Brownian motion coefficient of 0.8, we focus on an arbitrary source, for demonstrating the localization results of the location and velocity through time. The source is characterized by a speed of 0.5 m/sec (common to all) and a direction of about 52° (drawn). Figure 4.22 depicts a comparison through time between the true and the estimated trajectories of all four unknown parameters of the source: the radius (a), azimuth angle (b), speed (c) and direction (d). The true trajectories of all location and velocity parameters describe fluctuations in the location and velocity values through time, which reflect the random and rapid nature of the Brownian motion.

The estimated trajectories are divided into 3 cases- common to all is the nearest training neighbors in the manifold, which are determined according to the midway point. As already implied by the embeddings and the individual RMSE values, the estimated trajectories of the radius and the azimuth recover the true values well despite the Brownian motion, whereas the efforts for recovering the instantaneous speed and direction of the source do not succeed. The first estimated trajectory, denoted in Fig. 4.22 as LM, is executed according to the linear model that was described in detail in Chapter 3. This method demonstrates fair estimation results of the average velocity.

In case that the whole trajectories of the nearest training observations are known, these localization results can be improved, as suggested by the remaining two estimation methods. The second estimated trajectory, denoted in Fig. 4.22 as WA, is produced by a weighted average of the trajectories of the nearest training neighbors. This method cancels out the explicit random fluctuations in the location and velocity values in each step. However, the weighted average fairly succeeds to estimate the average speed and direction values through time. Another improvement can be achieved by combination of the two, by estimating the trajectories according to a linear model. The constant velocity is determined according to the mean value of the velocity trajectories of the weighted average method. This method, denoted in Fig. 4.22 as MV, improves both the accuracy of the estimated location and the estimated average velocity. It provides us accuracy of less than one percent, with respect to the average velocity, which is in practice the Brownian-motion-free instantaneous velocity. To conclude, despite the failure in recovering the instantaneous velocity of the source through time, the algorithm succeeds to estimate the average velocity through time- either fairly or well, depending the data available to us. Similar results have been received for all test sources- all the more so for Cartesian location and velocity coordinates.

We note the recovery of the average velocity can be improved by advanced denoising techniques, as the Brownian motion term, which is defined as WGN,

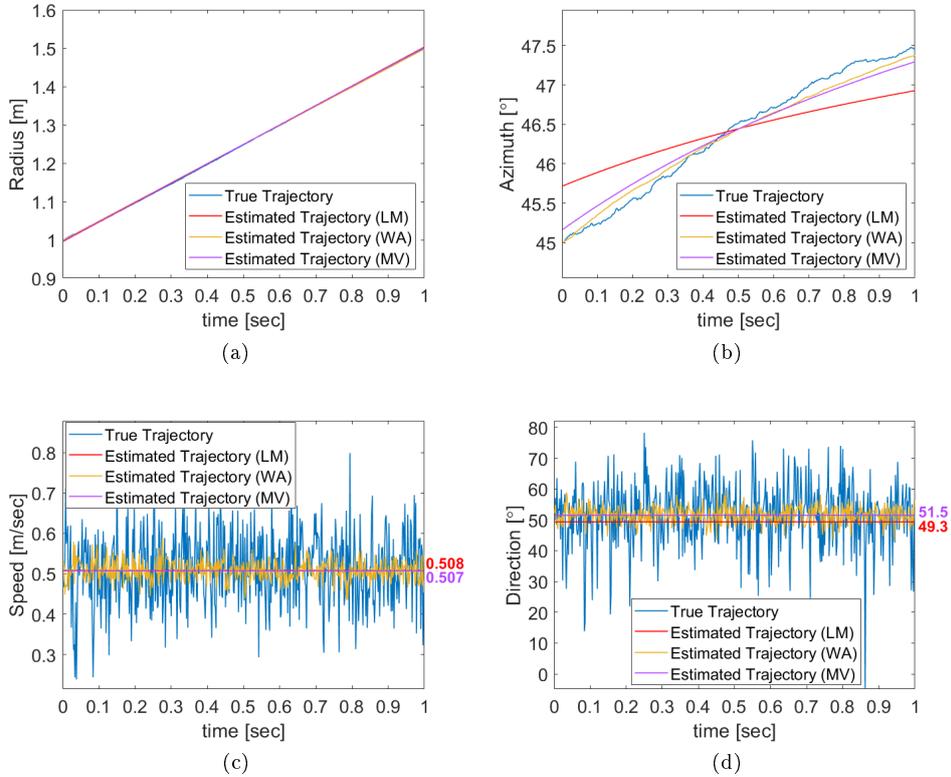


Figure 4.22: Localization results of a moving source through time of each of the unknown parameters for the scenario of Brownian motion coefficient of 0.8. (a) Radius, (b) azimuth, (c) speed and (d) direction. Blue represents true values, red represents the estimated values according to the linear model (LM), yellow represents the estimated values according to a weighted average of the trajectories (WA), and purple represents the estimated values according to a mean velocity value (MV).

is considered as an environmental interference with respect to the original movement of the source.

4.6 Conclusions

Following the proposed algorithm, the experimental results demonstrate state-of-the-art localization results for moving sources and stationary sources alike, using a single sensor.

In the first case, we address stationary sources since prior diffusion maps based single-sensor source localization work, by Talmon *et al.* [10], has focused on one-dimensional stationary scenario. It consists of the one-dimensional sub-

case as a baseline and the two-dimensional subcase as an extension, as follows:

1. One-dimensional subcase- our choice of an affinity measure between the observations and its corresponding extension for the test observations is confirmed, followed by the recovery of the azimuth angle- achieving RMSE of 0.094° - better than the results of [10].
2. Two-dimensional subcase- our experiment is extended for estimation of the two-dimensional location. First, we show that the embedded observations, mapped into the low-dimensional manifold, are arranged monotonically by each of the location coordinates. Subsequently, we recover the radius and azimuth of the sources- obtaining RMSE of 0.0105, which consists of 0.0041 for radius and 0.017 for azimuth.

In the second case, which is the main contribution of this research, we allow deterministic movement of the sources and retrieve their location and velocity. We examine in detail the capabilities of the proposed algorithm by 8 various experiments inspecting the sensitivity to different hyperparameters, variables and conditions by 8 various experiments:

1. Training set size- the total estimation accuracy increases with respect to additional training observations, up to marginal improvement for significantly larger sizes, as expected.
2. Number of features- the estimation error results reflect the peaking phenomenon. Small number of features yields poor accuracy due to an insufficient resolution. As the dimensionality increases, the additional features allows us to describe the location and velocity in finer details, thus the recovery improves. However, at some point the dimensionality increase becomes counterproductive and the performance of the estimation accuracy starts deteriorating. Optimal results are accomplished for $D = 800$ lags.
3. Speed- due to the high variability of the direction, its estimation error dictates the behavior of the total error throughout the entire experiment. We yield a high estimation error of the direction for very slow speed values thanks to the struggle of perceiving variations between different direction during a bare movement. As the speed gets faster, the accuracy of the direction improves due to a more meaningful movement by the sources, up to a point where the accuracy starts decreasing. Regarding the radius and the azimuth, as speed gets faster, their estimation errors get bigger.
4. Direction- due to the high variability of the speed, its estimation error dictates the behavior of the total estimation error and the radius estima-

tion error, throughout the entire experiment. The speed estimation error is characterized in inconsistent behavior as a result of a varying extent of spatial aliasing. As for the azimuth, as the direction gets away from the degenerated scenario, its estimation error aggravates.

5. SNR- the estimation error grows as the SNR conditions get harsher. Moreover, from SNR of 20 dB onward the accuracy improves insignificantly and tends to the accuracy of the case of no noise at all. The algorithm indicates robustness to noisy environments.
6. Reverberation time- the accuracy results of our single-sensor algorithm validate the implied hypothesis of our work, which is unique for single-sensor. The infamous reflections, which sabotage the efforts for a reliable location recovery using multiple sensors, are vital for accurate source localization using a sole sensor. The single-sensor algorithm is incapable of distinguishing between the different sources by relying solely on the direct direction of arrival, as expected. Similarly to source localization using multiple sensors, the performance of a sole sensor suffers from over-reverberation, but more robustly.
7. Environmental conditions changes- as the reverberation level of the test stage gets away from the reverberation time of the training stage, the estimation error worsens. Moreover, by comparing to the estimation errors acquired ideally, we can notice that as the absolute difference in the reverberation time between the stages grows, the error deviates from the values achieved ideally- unless the reflections do not play a role in practice.
8. Frame length- approximation of the trajectory of the source, formed by its movement during the frame, using a linear segment is inaccurate for long frames, unless the source moves at a constant velocity during the frame. Short frames are not useful either, regardless the trajectory, since they are unable to perceive the movement correctly. The optimal recovery accuracy is achieved by an iterative execution of the algorithm every 0.5 sec.

At last, in the third case, we introduce Brownian motion to the movement of the sources, in a varying degree, in order to describe a more realistic setting involving influence of environmental conditions. We examine the impact of violating the fundamental assumption of gradual changes of the velocity through time and the capability of the proposed algorithm to cope with it. As expected, as the Brownian motion term gets more dominant, the estimation error grows significantly. Despite the failure in recovering the instantaneous velocity of the

source through time, due to the rapid and random nature of the Brownian motion, the algorithm is capable of estimating successfully the average velocity, namely the drift term.

Chapter 5

Conclusions

5.1 Research Summary

An unconventional supervised approach for the familiar and mostly ill-posed/underdetermined problem of single-sensor source localization has been presented, using diffusion maps. The proposed algorithm extends manifold learning techniques of former works and demonstrates a proof of concept proposing a state-of-the-art solution for the recovery of the location and velocity of a moving source, using a single microphone. In contrast to common model-based source localization approaches, the proposed algorithm implements a data-driven approach which utilizes prior data for learning room's acoustic manifold.

Moreover, unlike former manifold learning-based localization works, the proposed algorithm deals with time variant system as a result of the movement of the sources. This movement calls for modification and expansion of the estimation formulation for physical quantities of different units. In addition, the proposed algorithm tackles their framework drawbacks such as an impractical kernel and a scaled and inaccurate extension. Furthermore, the algorithm is also examined for scenarios violating its fundamental assumption regarding the characteristics of the velocity.

Simulation results demonstrate the sensitivity of the algorithm to variations in speed, direction and frames length, and the accompanying tradeoffs, such as the tradeoff between the accuracy of the estimated location and the accuracy of the estimated direction, as a result of the speed of the sources. In addition, we witness the influence of opposing forces, such as the validity of our assumptions regarding the velocity attributes, and the time variance property of the system. The algorithm have difficulty with distinguishing between speed values of sources that move at the same direction. The reflections, which sabotage the efforts for a reliable location recovery using multiple sensors, are vital for accurate

source localization using a single sensor- these results validate the underlying hypothesis of our work. The algorithm performs well in reverberant and noisy environments, yet is sensitive to environmental conditions changes. While the algorithm is designated for recovering the location and velocity of slow sources that change direction and speed gradually, it is even capable of successfully estimating the location and the average velocity of slow sources that change their velocity rapidly and randomly.

These promising results and consequences establish a proof of concept, which contributes to the research endeavors in the field, and may benefit a scenario, where only a single sensor is available. By focusing on scenarios of moving sources related to one varying degree-of-freedom at a time, we have isolated each individual factor and laid the foundation for future work, involving scenarios where all parameters are varying randomly and independently.

At last, it is anticipated that in addition to successful validation of the simulated localization results by a real experiment, the results of the latter will be even superior, since the implementation [37] of the mirror model, for simulating the measurements of the received signal, is strict as it assumes a pure diffuse noise model, where in practice it is not.

5.2 Future Research

The single-sensor source localization for moving sources algorithm, proposed in this research, lays the foundation and paves the way for further research and experiments.

The results and the conclusions, derived from the examined single degree-of-freedom scenarios, are relevant regardless the number of degrees-of-freedom. However, it is highly recommended to verify them in scenarios where all four location and velocity parameters are independent, using significantly larger datasets. In addition, the localization results, obtained by simulations, can be validated by a real-data experiment.

As another step towards a realistic model, it is suggested to expand the algorithm for more sophisticated movement models, where the drift term is not constant.

The suggested supervised approach obliges us to provide labels for all training observations, which may be a cumbersome task for a real-data experiment, all the more so for scenarios involving larger datasets, or by learning according to labeled moving sources. Thus, following Laufer-Goldshtein *et al.* [13,18], we can exploit unlabeled observations and even observations that their unknown locations and velocities were previously estimated. These observations can be utilized for system adaptation and subsequently yielding better localization re-

sults, by developing a semi-supervised approach, which will be designated for single-sensor source localization.

As one of the outcomes of the proposed algorithm is favoring an iterative execution of the algorithm at each time frame, it can be inefficient for long and sophisticated trajectories. Thus, a major progress can be done by development of a data-driven tracking filter (e.g., Kalman filter) for single-sensor localization purposes, such that by exploiting its correction term the flaws of the current algorithm will be addressed. In addition, not only the tracking scheme will allow to exploit the semi-supervised approach, but also to learn based on stationary sources only, rather than moving sources. Such a single-sensor tracking filter can be developed following the works by Wang *et al.* [40] and Laufer-Goldshtein *et al.* [16–18], as elaborated in Appendix A- introducing a suggested single-sensor tracking framework.

Another major and challenging step towards real-world scenarios involves movement of the sensor as well. However, since the acquired manifold depends on the location of the sensor, each movement by the sensor results in a completely different manifold. Thus, all the learning process, based on the previous location of the sensor and prior its movement, is irrelevant. In order to overcome this challenge, a combination of two steps is advised to be examined. First, developing an extension for the aforementioned data-driven tracking filter, which its mechanism includes updates of the location of the sensor, similarly to simultaneous localization and mapping (SLAM) algorithms. Subsequently, domain adaptation methods, such as optimal transport and parallel transport, may adapt the acquired manifolds through time.

An additional virtue of the aforementioned domain adaptation is the possibility of successfully coping with significant environmental changes between the training and test stages. As a result, a flaw of the proposed algorithm, which struggles with moderate changes in the reverberation time following the training stage, can be addressed- along with additional setting changes.

Since single-sensor scenarios are common for RF, an interesting direction for expansion would be source localization of RF emitters at environments which are characterized in presence of a high level of multipath- such as the sea. Such localization at the sea, for instance, must involve integration of spherical trigonometry (explicitly Haversine formula) for accurate localization results, as one of the expansion phases. In order to do so, a reliable feature vector must be derived. The RF emitters arise new challenges, such as localization amidst of different noisy environments (where except unintentional interference sources-jammers are common), various frequency bands (HF, particularly, will allow comparison to the unique approach mentioned in [4], [7, Ch. 7], [8, Ch. 7]), various types of frequency ranges transmissions (e.g., continuous wave, narrow

band, wide band), and spread spectrum sources (e.g., chirps, hoppers).

One of the major flaws of the proposed algorithm, contradicting real world scenarios, is its restriction to a single transmitting source at a time. Thus, finding a new localization framework, which incorporates speaker diarization techniques, is highly suggested for localization of multiple sources simultaneously (Note that at the RF case, several sources which transmit simultaneously in different frequencies can be dealt by exploiting the narrowband filter of the DF sensor).

At last, since all velocities on the discussed setting are significantly slow in comparison to the speed of sound, Doppler effect has been neglected in this contribution, but it is recommended to be taken into consideration in several scenarios in future work. Note that in the RF case, even fighter aircrafts cause a barely noticeable frequency shift, which is already within the bandwidth of the narrowband receiver.

Appendix A

Single-Sensor Tracking Framework

A.1 Introduction

In the following appendix we present a single-sensor tracking framework, as mentioned in Section 5.2. The tracking framework is based on modeling the function mapping the observations to locations by a Gaussian process. The Gaussian process allows exploiting Bayesian inference approaches for estimation of the mapping function. In particular, it provides us the opportunity to use tracking filters for an accurate location estimation of moving sources during their movement. By using a data-driven Kalman filter we incorporate smoothness with respect to the geometric structure of the manifold with the movement of the sources through time. This model is designated to tackle major flaws of the diffusion kernel-based model presented in Chapter 3, and is characterized by the following attributes:

1. Small training set size. Moreover, training set is based on stationary sources only, where each one represents a unique location. Thus, better applicable for real-life applications, as the process of generating a labeled training set is a cumbersome task.
2. Tracking capability- not only it mitigates the inefficiency issues of the diffusion kernel-based model for long and sophisticated trajectories, but it also provides us a correction term that is exploited in each iteration.
3. Allows a mixture of both unlabeled and labeled observations for the learning process (i.e., semi-supervised learning), such that the recovery of the location is more accurate due to better understanding of the structure of

the manifold. We note that in the derivation of the following framework we focus on labeled training set only (i.e., supervised learning).

4. Robustness- similarly to the diffusion kernel-based model, the tracking model is expected to provide robustness to noise and challenging environmental conditions, such as reverberation time.
5. Inability to estimate the current velocity of the moving source- the capability to recover the velocity of a source in each time frame, following the diffusion kernel-based model, is based on a supervised learning using moving sources. In the suggested tracking framework, on the other hand, the learning process is based on stationary sources. Thus, the velocity of the current time frame cannot be estimated. We note that the velocity can be estimated indirectly by subtraction of the sequential estimated locations of the source (followed by division by the time difference), but in fact it is the velocity of the previous time frame- not the current.

The following data-driven tracking framework is inspired by both the observation (measurement) and propagation (state) models by Wang *et al.* [40], and the multiple-sensors-based propagation models by Laufer-Goldshtein *et al.* [16–18].

We note that experiments examining the performance of the single-sensor tracking framework are out of scope of this work.

The structure of the rest of this appendix is as follows. In Section A.2, we formulate the problem. In Section A.3, we describe the feature vectors and the mapping function from the observations to locations. In Section A.4, we present the state-space model, including the propagation and observation equations. In Section A.5, we present an iterative scheme for tracking the location of the moving source based on the state-space representation. At last, in Section A.6 we describe how to determine the values of the hyperparameters of the data-driven tracking filter.

A.2 Problem Formulation

We consider a standard enclosure. The velocity of moving source as a function of time, denoted by $\dot{\mathbf{p}}(t)$, is defined as in (3.1).

We consider m sources that their location is fixed and unique. In addition, we consider a moving source. Each source, one at a time, transmits a signal (The moving source transmits the signal during its movement). The signal received by the sensor, denoted by $y_i(n)$, is defined as in (3.2). For clarity and avoiding confusion with the definitions in Chapters 2 and 3, in the rest of this appendix we refer to the parameters vector $\boldsymbol{\theta}_i$ as \mathbf{p}_i . \mathbf{p}_i denotes the three-dimensional

absolute location of the i th source in Cartesian representation- meaning it can be described at discrete time index j by: $\boldsymbol{\theta}_i(j) \equiv \mathbf{p}_i(j) = [p_{i,\bar{x}}(j), p_{i,\bar{y}}(j), p_{i,\bar{z}}(j)]^T$. The received signal is recorded and divided into overlapping time frames.

The movement of the moving source can be expressed by the following Markovian relation:

$$\mathbf{p}(q) = \mathbf{b}(\mathbf{p}(q-1), \mathcal{I}_q), \quad (\text{A.1})$$

where q is the frame index, $\mathbf{b}(\cdot)$ represents a transition function and \mathcal{I}_q denotes all related information, available at frame index q (e.g., prior knowledge regarding the enclosure).

A.3 Manifold-Based Gaussian Process

A.3.1 Feature Vectors Extraction

From each observation, a feature vector is extracted based on the autocorrelation function of the signal received by the sensor. Following the derivation and assumptions discussed in Section 3.3.1, we yield the time-variant autocorrelation function of $y_i(n)$ according to (3.3) and (3.4). Subsequently, following the slow speed and gradually changing velocity assumption, we yield the time-invariant version of the autocorrelation function according to (3.5). Accordingly, we define the nonlinear mapping of the location, denoted by c , and the first D elements of the autocorrelation function of the observation \mathbf{y}_i , denoted by \mathbf{c}_i , according to (3.6). In addition, we define the j th entry of \mathbf{c}_i , denoted by $\mathbf{c}_i^{(j)}$, according to (3.7).

Let $\mathbf{c}_i(q)$ denote the first D elements of the autocorrelation function of the q th time frame. By smoothing the autocorrelation function through time, we form a relation between the sequential overlapping time frames, such that the smoothed autocorrelation function of the q th frame is given by:

$$\tilde{\mathbf{c}}_i(q) = \gamma \mathbf{c}_i(q) + (1 - \gamma) \tilde{\mathbf{c}}_i(q-1), \quad (\text{A.2})$$

where $0 \leq \gamma \leq 1$ is the smoothing parameter.

Let $\bar{\mathcal{C}} = \{\bar{\mathbf{c}}_i\}_{i=1}^m$ be the set of the training feature vectors with respect to the known fixed locations $\{\bar{\mathbf{p}}_i\}_{i=1}^m$, that are generated from m unique locations. Let $\tilde{\mathbf{c}}(q)$ denote the feature vector of the moving source with respect to the unknown location at the q th time frame. We note that since the locations of the training sources are fixed, it is not necessary to extract their corresponding feature vectors according to (A.2)- meaning their feature vectors can be computed according to (3.5), using the whole observations (prior the division to

time frames).

A.3.2 Mapping Feature Vectors to Source Locations

Let f_a be a nonlinear mapping function from the observable space to the parametric space, i.e. $f_a : \mathcal{M} \rightarrow \mathbb{R}$. In other words the purpose of the mapping function is matching an autocorrelation feature vector to its corresponding source location (i.e., $p_a(q) = f_a(\tilde{\mathbf{c}}(q))$, $a \in \{\tilde{x}, \tilde{y}, \tilde{z}\}$). The mapping function is modeled by a Gaussian process (GP) [41], which is a nonparametric model defined entirely by its second moment and is widely used for regression problems. We define the mapping function by a zero-mean GP:

$$f_a(\cdot) \sim \mathcal{GP}(0, \tilde{\kappa}), \quad (\text{A.3})$$

where $\tilde{\kappa} : \mathcal{M} \times \mathcal{M} \rightarrow \mathbb{R}$ is its manifold-based covariance function. The covariance function, also known as the kernel function, reflects the similarity measure between feature vectors (similarly to the purpose of the affinity matrix mentioned in Section 3.3.3), as follows:

$$\tilde{\kappa}(\mathbf{c}_r, \mathbf{c}_l) = \sum_{i=1}^m \kappa(\mathbf{c}_r, \bar{\mathbf{c}}_i) \kappa(\mathbf{c}_l, \bar{\mathbf{c}}_i), \quad (\text{A.4})$$

where $\kappa : \mathcal{M} \times \mathcal{M} \rightarrow \mathbb{R}$ is the standard kernel function, which in our case is chosen to be the Gaussian kernel:

$$\kappa(\mathbf{c}_i, \mathbf{c}_j) = \exp \left\{ -\frac{\|\mathbf{c}_i - \mathbf{c}_j\|_2^2}{\varepsilon} \right\}, \quad (\text{A.5})$$

where ε is a scaling factor. Note that in contrast to the affinity matrix (3.8) mentioned in Section 3.3.3, the computation (A.4) of the covariance function between two feature vectors involves their relations to the entire training set. In such a manner, we yield an affinity measure which respects the manifold structure. Two feature vectors are considered close if their relations to all the training observations are similar- meaning their corresponding locations of the sources are physically adjacent as well (and vice versa). Despite the Gaussian kernel (A.5) is based on a standard Euclidean distance, which is reliable for close distance only as discussed in Section 3.3.2, the combination of the scaling factor and the multiplication in another Gaussian kernel in (A.4), preserves locality. We also note that the affinity kernel (3.8) mentioned in Section 3.3.3 can be another useful choice for either the Gaussian kernel (A.5) or the covariance function (A.4).

The GP model assumption is intended for guaranteeing the smoothness of the solution for the regression problem, while the pairs $\{\bar{\mathbf{c}}_i, \bar{\mathbf{p}}_i\}_{i=1}^m$ supply anchor

points.

For each frame of the observation of the moving source $\tilde{\mathbf{c}}(q)$, we define a subset of nearby training observations $\{\bar{\mathbf{c}}_{q_i} \mid \|\tilde{\mathbf{c}}(q) - \bar{\mathbf{c}}_{q_i}\| < \tilde{\delta}, q_i \in \{1, \dots, m\}\}$, where $\tilde{\delta}$ is the neighborhood radius. For acquiring fixed-size subsets, we limit ourselves to the \tilde{k} -nearest neighbors among the specified subset. Accordingly, we denote the \tilde{k} -nearest neighbors of $\tilde{\mathbf{c}}(q)$ as $\bar{\mathcal{C}}_q = \{\bar{\mathbf{c}}_{q_i}\}_{i=1}^{\tilde{k}}$. Note that the neighbors are determined according to the Euclidean distance between corresponding feature vectors, according to considerations mentioned in Section 3.3.3. Let $\mathcal{C}_q = \tilde{\mathbf{c}}(q) \cup \bar{\mathcal{C}}_q$ denote an extended set of size $\tilde{k} + 1$. While the former contains the feature vector of the current time frame of the moving source and its \tilde{k} -nearest feature vectors of training sources, the latter contains the corresponding locations of the sources. Moreover, let $\mathbf{f}_{q,a} \equiv f_a(\mathcal{C}_q) = [f_a(\tilde{\mathbf{c}}(q)), f_a(\bar{\mathbf{c}}_{q_1}), \dots, f_a(\bar{\mathbf{c}}_{q_{\tilde{k}}})]^T$ denote a vector of all corresponding mappings of the function $f_a(\cdot)$ over the observations in \mathcal{C}_q , yielding representation of the their corresponding locations.

A.4 Manifold-Based State-Space Model

A.4.1 Propagation (State) Model

The relation between the mappings of the successive $(q - 1)$ th and q th subsets, which is described by the state equation and is governed by the GP (A.3), represents the Markovian transition of the hidden function values. Since both $\mathbf{f}_{q,a}$ and $\mathbf{f}_{q-1,a}$ are Gaussian vectors, they have a joint Gaussian distribution with a zero-mean and a covariance matrix, which is based on the terms of the covariance function (A.4):

$$\begin{bmatrix} \mathbf{f}_{q,a} \\ \mathbf{f}_{q-1,a} \end{bmatrix} | \mathcal{C}_{q,q-1} \sim \mathcal{N} \left(\mathbf{0}_{2(\tilde{k}+1)}, \begin{bmatrix} \tilde{\Sigma}_{q,q} & \tilde{\Sigma}_{q,q-1} \\ \tilde{\Sigma}_{q,q-1}^T & \tilde{\Sigma}_{q-1,q-1} \end{bmatrix} \right), \quad (\text{A.6})$$

where $\mathcal{C}_{q,q-1} = \mathcal{C}_q \cup \mathcal{C}_{q-1}$, $\mathbf{0}_{2(\tilde{k}+1)}$ is a zeros vector of length $2(\tilde{k} + 1)$, and:

$$\tilde{\Sigma}_{q,g} = \begin{bmatrix} \tilde{\kappa}(\tilde{\mathbf{c}}(q), \tilde{\mathbf{c}}(g)) & \tilde{\kappa}(\tilde{\mathbf{c}}(q), \bar{\mathbf{c}}_{g_1}) & \cdots & \tilde{\kappa}(\tilde{\mathbf{c}}(q), \bar{\mathbf{c}}_{g_{\tilde{k}}}) \\ \tilde{\kappa}(\bar{\mathbf{c}}_{q_1}, \tilde{\mathbf{c}}(g)) & \tilde{\kappa}(\bar{\mathbf{c}}_{q_1}, \bar{\mathbf{c}}_{g_1}) & \cdots & \tilde{\kappa}(\bar{\mathbf{c}}_{q_1}, \bar{\mathbf{c}}_{g_{\tilde{k}}}) \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{\kappa}(\bar{\mathbf{c}}_{q_{\tilde{k}}}, \tilde{\mathbf{c}}(g)) & \tilde{\kappa}(\bar{\mathbf{c}}_{q_{\tilde{k}}}, \bar{\mathbf{c}}_{g_1}) & \cdots & \tilde{\kappa}(\bar{\mathbf{c}}_{q_{\tilde{k}}}, \bar{\mathbf{c}}_{g_{\tilde{k}}}) \end{bmatrix}. \quad (\text{A.7})$$

Hence, the conditional distribution of $\mathbf{f}_{q,a}$ given $\mathbf{f}_{q-1,a}$ is also a Gaussian distribution:

$$\Pr(\mathbf{f}_{q,a}|\mathbf{f}_{q-1,a},\mathcal{C}_{q,q-1}) = \mathcal{N}(\mathbf{B}(\mathcal{C}_{q,q-1})\mathbf{f}_{q-1,a}, \mathbf{Q}(\mathcal{C}_{q,q-1})), \quad (\text{A.8})$$

where

$$\mathbf{B}(\mathcal{C}_{q,q-1}) = \tilde{\Sigma}_{q,q-1}\tilde{\Sigma}_{q-1,q-1}^{-1}, \quad (\text{A.9})$$

$$\mathbf{Q}(\mathcal{C}_{q,q-1}) = \tilde{\Sigma}_{q,q} - \tilde{\Sigma}_{q,q-1}\tilde{\Sigma}_{q-1,q-1}^{-1}\tilde{\Sigma}_{q-1,q}^T. \quad (\text{A.10})$$

Thus, according to the conditional Gaussian distribution (A.8), we can conclude that the state equation is governed by a linear dependence between the positions of the current set $\mathbf{f}_{q,a}$ to the positions of the previous set $\mathbf{f}_{q-1,a}$. Therefore, the propagation of the source locations in (A.1) can be expressed by a linear equation between subsequent time frames, with an additive WGN $\boldsymbol{\xi}(\mathcal{C}_{q,q-1}) \sim \mathcal{N}(\mathbf{0}_{k+1}, \mathbf{Q}(\mathcal{C}_{q,q-1}))$:

$$\mathbf{f}_{q,a} = \mathbf{B}(\mathcal{C}_{q,q-1})\mathbf{f}_{q-1,a} + \boldsymbol{\xi}(\mathcal{C}_{q,q-1}). \quad (\text{A.11})$$

We can deduce that using $\mathcal{I}_q \equiv \mathcal{C}_{q,q-1}$ and the relations between the corresponding feature vectors, the model parameters (i.e., state-transition matrix $\mathbf{B}(\mathcal{C}_{q,q-1})$ and variance of the process noise $\mathbf{Q}(\mathcal{C}_{q,q-1})$) are calculated. As a result, the problem of nonlinear regression in \mathbb{R}^D (i.e., the high-dimensional space of the observations) is reduced to a linear time-varying state equation of the movement of the source.

To conclude, the complete state equation for the three-dimensional location, denoted by $\mathbf{f}_q = [\mathbf{f}_{q,\tilde{x}}^T, \mathbf{f}_{q,\tilde{y}}^T, \mathbf{f}_{q,\tilde{z}}^T]^T$, is provided by:

$$\mathbf{f}_q = \mathbf{B}_3(\mathcal{C}_{q,q-1})\mathbf{f}_{q-1} + \boldsymbol{\xi}_3(\mathcal{C}_{q,q-1}), \quad (\text{A.12})$$

where $\mathbf{B}_3(\mathcal{C}_{q,q-1}) = \mathbf{I}_3 \otimes \mathbf{B}(\mathcal{C}_{q,q-1})$, $\boldsymbol{\xi}_3 \sim \mathcal{N}(\mathbf{0}_{3(k+1)}, \mathbf{Q}_3(\mathcal{C}_{q,q-1}))$ with $\mathbf{Q}_3(\mathcal{C}_{q,q-1}) = \mathbf{I}_3 \otimes \mathbf{Q}(\mathcal{C}_{q,q-1})$, \otimes is the Kronecker product, and \mathbf{I}_3 is a 3×3 identity matrix.

A.4.2 Observation (Measurement) Model

The goal of the nonlinear regression (nonlinear with respect to the observation) is learning a function $f_a(\bar{\mathbf{c}}_i)$ following the noisy relationship between a feature vector input $\bar{\mathbf{c}}_i \in \mathbb{R}^D$ and its corresponding output $\bar{p}_{i,a} \in \mathbb{R}$, which is the a -coordinate ($a \in \{\tilde{x}, \tilde{y}, \tilde{z}\}$) of the measured known location $\bar{\mathbf{p}}_i$. We assume that $\bar{p}_{i,a}$ is a noisy version of the corresponding coordinate of the actual known location, as a result of imperfections in the measurement of the location while

obtaining the training observation:

$$\bar{p}_{i,a} = f_a(\bar{\mathbf{c}}_i) + u, \quad (\text{A.13})$$

where u is a measurement noise represented by an AWGN $u \sim \mathcal{N}(0, \sigma_u^2)$.

Let $\bar{\mathbf{p}}_a = [\bar{p}_{1,a}, \dots, \bar{p}_{m,a}]^T$ denote a vector of the a -coordinate of all training locations. Let $\bar{\mathbf{p}}_{q,a} = [\bar{p}_{q_1,a}, \dots, \bar{p}_{q_k,a}]^T$ denote a vector of length \hat{k} of the a -coordinate of the training locations corresponding to the subset $\bar{\mathcal{C}}_q = \{\bar{\mathbf{c}}_{q_i}\}_{i=1}^{\hat{k}}$. The observation equation, which is the relationship between the latent mapping function values and the outputs (i.e., measured known positions) is given by:

$$\bar{\mathbf{p}}_{q,a} = \mathbf{H}_q \mathbf{f}_{q,a} + \mathbf{u}_q, \quad (\text{A.14})$$

where $\mathbf{H}_q = [\mathbf{0}_{\hat{k}} \quad \mathbf{I}_{\hat{k}}]$ is an index matrix such that the each of the measured locations in $\bar{\mathbf{p}}_{q,a}$ is only related to its corresponding feature vector in $\bar{\mathcal{C}}_q$, $\mathbf{0}_{\hat{k}}$ is a zeros vector of size $\hat{k} \times 1$, $\mathbf{I}_{\hat{k}}$ is a $\hat{k} \times \hat{k}$ identity matrix, and \mathbf{u}_q is an independent and identically distributed (iid) WGN. The covariance matrix of \mathbf{u}_q is given by $\mathbf{R}_q = \sigma_u^2 \mathbf{I}_{\hat{k}}$.

Let $\bar{\mathbf{p}}_q = [\bar{\mathbf{p}}_{q,\hat{x}}^T, \bar{\mathbf{p}}_{q,\hat{y}}^T, \bar{\mathbf{p}}_{q,\hat{z}}^T]^T$ denote the three-dimensional measured training locations (vector of length $3\hat{k}$). Accordingly, we can define the complete observation equation for the three-dimensional measured training locations as follows:

$$\bar{\mathbf{p}}_q = \mathbf{H}_{3,q} \mathbf{f}_q + \mathbf{u}_{3,q}, \quad (\text{A.15})$$

where $\mathbf{H}_{3,q} = \mathbf{I}_3 \otimes \mathbf{H}_q$, and $\mathbf{u}_{3,q} \sim \mathcal{N}(\mathbf{0}_{3\hat{k}}, \mathbf{R}_{3,q})$ with $\mathbf{R}_{3,q} = \mathbf{I}_3 \otimes \mathbf{R}_q = \sigma_u^2 \mathbf{I}_{3\hat{k}}$.

A.4.3 Combined State-Space Representation

To summarize, the proposed state-space model, for the problem of single-sensor source localization of a moving source, is fully specified by the following set of linear equations:

$$\mathbf{f}_q = \mathbf{B}_3(\mathcal{C}_{q,q-1}) \mathbf{f}_{q-1} + \boldsymbol{\xi}_3(\mathcal{C}_{q,q-1}) \quad (\text{A.16a})$$

$$\bar{\mathbf{p}}_q = \mathbf{H}_{3,q} \mathbf{f}_q + \mathbf{u}_{3,q} \quad (\text{A.16b})$$

A.5 Tracking Scheme

Since both the propagation and the measurement equations are linear, a standard Kalman filter is sufficient (i.e., unscented or extended Kalman filters are unnecessary) for recursively solving (A.16). By substituting values following (A.16), Kalman equations can be expressed by two phases:

1. The predict phase:

$$\hat{\mathbf{f}}_{q|q-1} = \mathbf{B}_{3,q} \hat{\mathbf{f}}_{q-1|q-1}, \quad (\text{A.17})$$

where $\mathbf{B}_{3,q} \equiv \mathbf{B}_3(\mathcal{C}_{q,q-1})$ and $\mathbf{Q}_{3,q} \equiv \mathbf{Q}_3(\mathcal{C}_{q,q-1})$

$$\mathbf{P}_{q|q-1} = \mathbf{B}_{3,q} \mathbf{P}_{q-1|q-1} \mathbf{B}_{3,q}^T + \mathbf{Q}_{3,q}, \quad (\text{A.18})$$

2. The update phase:

$$\hat{\mathbf{f}}_{q|q} = \hat{\mathbf{f}}_{q|q-1} + \mathbf{K}_q (\bar{\mathbf{p}}_q - \mathbf{H}_{3,q} \hat{\mathbf{f}}_{q|q-1}), \quad (\text{A.19})$$

$$\mathbf{P}_{q|q} = (\mathbf{I} - \mathbf{K}_q \mathbf{H}_{3,q}) \mathbf{P}_{q|q-1}, \quad (\text{A.20})$$

where $\mathbf{P}_{q|q-1}$ is the predicted error covariance, $\mathbf{P}_{q|q}$ is the updated error covariance, and \mathbf{K}_q is Kalman gain which is given by:

$$\mathbf{K}_q = \mathbf{P}_{q|q-1} \mathbf{H}_{3,q}^T (\mathbf{H}_{3,q} \mathbf{P}_{q|q-1} \mathbf{H}_{3,q}^T + \mathbf{R}_{3,q})^{-1}. \quad (\text{A.21})$$

A.6 Learning Hyperparameters

Let $\tilde{\boldsymbol{\theta}} = \{\varepsilon, \sigma_u^2\}$ denote the set of the hyperparameters of the tracking model. The value of $\tilde{\boldsymbol{\theta}}$ can be derived from the data by optimization of the likelihood function of the labeled dataset. For clarity in the following section, we denote the vector of the a -coordinate of all training locations as $\bar{\mathbf{p}}_L$, i.e. $\bar{\mathbf{p}}_a \equiv \bar{\mathbf{p}}_L$. Following (A.13), we emphasize that the random vector $\bar{\mathbf{p}}_L$ has a multivariate Gaussian distribution. Similarly to the derivation of (A.6), we can describe the joint Gaussian distribution of the training dataset and the estimated a -coordinate of the location of the moving source at time frame q (i.e., $f_a(\tilde{\mathbf{c}}(q))$):

$$\begin{bmatrix} \bar{\mathbf{p}}_L \\ f_a(\tilde{\mathbf{c}}(q)) \end{bmatrix} | \bar{\mathcal{C}} \sim \mathcal{N} \left(\mathbf{0}_{m+1}, \begin{bmatrix} \bar{\boldsymbol{\Sigma}}_{LL} + \sigma_u^2 \mathbf{I}_m & \bar{\boldsymbol{\Sigma}}_{Lq} \\ \bar{\boldsymbol{\Sigma}}_{Lq}^T & \bar{\boldsymbol{\Sigma}}_{qq} \end{bmatrix} \right), \quad (\text{A.22})$$

where $\bar{\boldsymbol{\Sigma}}_{LL}$ is a $m \times m$ covariance matrix defined over the mapping function values at the labeled dataset $\bar{\mathcal{C}}$, $\bar{\boldsymbol{\Sigma}}_{Lq}$ is a $m \times 1$ covariance vector between the mapping function values at $\bar{\mathcal{C}}$ and $f_a(\tilde{\mathbf{c}}(q))$, and $\bar{\boldsymbol{\Sigma}}_{LL}$ is the variance (a scalar) of

$f_a(\tilde{\mathbf{c}}(q))$ - all are calculated according to (A.4). Accordingly, the log-likelihood function of the training observations is given by:

$$L = \ln \Pr(\bar{\mathbf{p}}_L | \bar{\mathbf{c}}; \tilde{\boldsymbol{\theta}}) = -\frac{1}{2} \bar{\mathbf{p}}_L^T (\bar{\boldsymbol{\Sigma}}_{LL} + \sigma_u^2 \mathbf{I}_m)^{-1} \bar{\mathbf{p}}_L - \frac{1}{2} \ln |\bar{\boldsymbol{\Sigma}}_{LL} + \sigma_u^2 \mathbf{I}_m| - \frac{m}{2} \ln(2\pi). \quad (\text{A.23})$$

According to [41], the first term assesses the correspondence of the parameters to the given training dataset, while the second term demonstrates model's complexity (by the determinant of the covariance matrix). The optimization procedure is based on the computation of the gradients of the log-likelihood function with respect to each parameter. The partial derivative with respect to ε is given by:

$$\begin{aligned} \frac{\partial L}{\partial \varepsilon} &= -\frac{1}{2} \text{trace} \left\{ \boldsymbol{\Gamma}_{LL} \frac{\partial \bar{\boldsymbol{\Sigma}}_{LL}}{\partial \varepsilon} \right\} + \frac{1}{2} \bar{\mathbf{p}}_L^T \boldsymbol{\Gamma}_{LL} \frac{\partial \bar{\boldsymbol{\Sigma}}_{LL}}{\partial \varepsilon} \boldsymbol{\Gamma}_{LL} \bar{\mathbf{p}}_L = \\ &= \frac{1}{2} \text{trace} \left\{ \left[(\boldsymbol{\Gamma}_{LL} \bar{\mathbf{p}}_L) (\boldsymbol{\Gamma}_{LL} \bar{\mathbf{p}}_L)^T - \boldsymbol{\Gamma}_{LL} \right] \frac{\partial \bar{\boldsymbol{\Sigma}}_{LL}}{\partial \varepsilon} \right\}, \quad (\text{A.24}) \end{aligned}$$

where $\boldsymbol{\Gamma}_{LL} \equiv (\bar{\boldsymbol{\Sigma}}_{LL} + \sigma_u^2 \mathbf{I}_m)^{-1}$. Similarly, the partial derivative with respect to σ_u^2 can be expressed by:

$$\frac{\partial L}{\partial \sigma_u^2} = \frac{1}{2} \text{trace} \left\{ (\boldsymbol{\Gamma}_{LL} \bar{\mathbf{p}}_L) (\boldsymbol{\Gamma}_{LL} \bar{\mathbf{p}}_L)^T - \boldsymbol{\Gamma}_{LL} \right\}. \quad (\text{A.25})$$

Note that since the optimization is executed according to the log-likelihood of the labeled data set, the optimized values of the hyperparameters cannot assure optimality with respect to the observations of the moving source. Thus, this optimization procedure can provide an initialized value to the hyperparameters, such that by exploiting complementary methods (for instance cross-validation based methods) their values may later be fine-tuned.

A.7 Summary

The single-sensor tracking algorithm is summarized in Algorithm A.1.

Algorithm A.1 Single-Sensor Tracking Algorithm

Input:

- A training set consisting of m observations $\bar{\mathcal{C}} = \{\bar{\mathbf{c}}_i\}_{i=1}^m \subset \mathbb{R}^D$, generated from sources located at m known fixed and unique locations $\{\bar{\mathbf{p}}_i\}_{i=1}^m \subset \mathbb{R}^d$, which form a grid with a predefined resolution.
Note: All computations required in the training stage, are performed ahead (i.e., offline). The calculation of the feature vectors of the training set is according to the previous scheme.
- A time frame of new test measurements, generated from a moving source.

Output:

- Estimated location of the moving source at time frame q .

Step 1:

Learning the optimal hyperparameters' values given the training set, by applying a gradient-descent based optimizer on the marginal likelihood $\Pr(\bar{\mathbf{p}}_L | \bar{\mathcal{C}}; \boldsymbol{\theta})$.

Step 2:

For each (overlapping) time frame:

1. Estimate $\tilde{\mathbf{c}}(q)$ according to (A.2).
 2. Search for the \hat{k} nearest neighbors of $\tilde{\mathbf{c}}(q)$ among all training observations, and accordingly form the set $\bar{\mathcal{C}}_q = \{\bar{\mathbf{c}}_{q_i}\}_{i=1}^{\hat{k}}$.
 3. Form the sets $\mathcal{C}_q = \tilde{\mathbf{c}}(q) \cup \bar{\mathcal{C}}_q$ and $\mathcal{C}_{q,q-1} = \mathcal{C}_q \cup \mathcal{C}_{q-1}$.
 4. Compute all the covariance terms between the sets \mathcal{C}_q and \mathcal{C}_{q-1} (A.7), according to (A.5) and (A.4).
 5. Compute matrices $\mathbf{B}_3(\mathcal{C}_{q,q-1})$ and $\mathbf{Q}_3(\mathcal{C}_{q,q-1})$, using (A.9), (A.10) and Kronecker product.
 6. Compute matrices $\mathbf{H}_{3,q}$ and $\mathbf{R}_{3,q}$, according to Section A.4.2.
 7. Estimate $\bar{\mathbf{p}}_q$ by (A.16b).
 8. Apply a standard Kalman Filter recursion:
 - (a) Predict phase according to (A.17) and (A.18).
 - (b) Kalman gain calculation according to (A.21).
 - (c) Update phase according to (A.19) and (A.20).
-

Bibliography

- [1] M. Omologo and P. Svaizer, "Use of the crosspower-spectrum phase in acoustic event location," *IEEE Transactions on Speech and Audio Processing*, vol. 5, no. 3, pp. 288–292, 1997.
- [2] J. H. DiBiase, H. F. Silverman, and M. S. Brandstein, "Robust localization in reverberant rooms," in *Microphone Arrays*, pp. 157–180, Springer, 2001.
- [3] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276–280, 1986.
- [4] H. Höring, "Comparison of the fixing accuracy of single-station locators and triangulation systems assuming ideal shortwave propagation in the ionosphere," in *Proc. IEE Proceedings F (Radar and Signal Processing)*, vol. 137, pp. 173–176, IET, 1990.
- [5] H. H. Jenkins, *Small-Aperture Radio Direction-Finding*. Artech House on Demand, 1991.
- [6] R. L. Johnson, Q. Black, and A. G. Sonsteby, "Hf multipath passive single site radio location," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 30, no. 2, pp. 462–470, 1994.
- [7] R. Poisel, *Electronic Warfare Target Location Methods*. Artech House, 2012.
- [8] D. Adamy, *EW 103: Tactical Battlefield Communications Electronic Warfare*. Artech House, 2008.
- [9] R. Talmon, D. Kushnir, R. R. Coifman, I. Cohen, and S. Gannot, "Parametrization of linear systems using diffusion kernels," *IEEE Transactions on Signal Processing*, vol. 60, no. 3, pp. 1159–1173, 2011.

- [10] R. Talmon, I. Cohen, and S. Gannot, “Supervised source localization using diffusion kernels,” in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pp. 245–248, IEEE, 2011.
- [11] B. Laufer, R. Talmon, and S. Gannot, “Relative transfer function modeling for supervised source localization,” in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pp. 1–4, IEEE, 2013.
- [12] B. Laufer-Goldshtein, R. Talmon, and S. Gannot, “A study on manifolds of acoustic responses,” in *Proc. International Conference on Latent Variable Analysis and Signal Separation*, pp. 203–210, Springer, 2015.
- [13] B. Laufer-Goldshtein, R. Talmon, and S. Gannot, “Semi-supervised sound source localization based on manifold regularization,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 8, pp. 1393–1407, 2016.
- [14] B. Laufer-Goldshtein, R. Talmon, and S. Gannot, “Manifold-based bayesian inference for semi-supervised source localization,” in *Proc. 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6335–6339, IEEE, 2016.
- [15] B. Laufer-Goldshtein, R. Talmon, and S. Gannot, “Semi-supervised source localization on multiple manifolds with distributed microphones,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 7, pp. 1477–1491, 2017.
- [16] B. Laufer-Goldshtein, R. Talmon, and S. Gannot, “Speaker tracking on multiple-manifolds with distributed microphones,” in *Proc. International Conference on Latent Variable Analysis and Signal Separation*, pp. 59–67, Springer, 2017.
- [17] B. Laufer-Goldshtein, R. Talmon, and S. Gannot, “A hybrid approach for speaker tracking based on tdoa and data-driven models,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 4, pp. 725–735, 2018.
- [18] B. Laufer-Goldshtein, R. Talmon, S. Gannot, *et al.*, “Data-driven multi-microphone speaker localization on manifolds,” *Foundations and Trends in Signal Processing*, vol. 14, no. 1–2, pp. 1–161, 2020.

- [19] E. Kupershtein, M. Wax, and I. Cohen, "Single-site emitter localization via multipath fingerprinting," *IEEE Transactions on signal processing*, vol. 61, no. 1, pp. 10–21, 2012.
- [20] A. Jaffe and M. Wax, "Single-site localization via maximum discrimination multipath fingerprinting," *IEEE Transactions on Signal Processing*, vol. 62, no. 7, pp. 1718–1728, 2014.
- [21] A. Khalajmehrabadi, N. Gatsis, and D. Akopian, "Modern wlan fingerprinting indoor positioning methods and deployment challenges," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1974–2002, 2017.
- [22] S. Wielandt and L. D. Strycker, "Indoor multipath assisted angle of arrival localization," *Sensors*, vol. 17, no. 11, p. 2522, 2017.
- [23] X. Sun, X. Gao, G. Y. Li, and W. Han, "Fingerprint based single-site localization for massive mimo-ofdm systems," in *Proc. GLOBECOM 2017-2017 IEEE Global Communications Conference*, pp. 1–7, IEEE, 2017.
- [24] X. Sun, X. Gao, G. Y. Li, and W. Han, "Single-site localization based on a new type of fingerprint for massive mimo-ofdm systems," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 7, pp. 6134–6145, 2018.
- [25] R. Zhang, G. Chen, Q. Zeng, and L. Shen, "Single-site positioning method based on high-resolution estimation in vanet localization," *IEEE Access*, vol. 6, pp. 54674–54682, 2018.
- [26] L. Chen, X. Yang, P. X. Liu, and C. Li, "A novel outlier immune multipath fingerprinting model for indoor single-site localization," *IEEE Access*, vol. 7, pp. 21971–21980, 2019.
- [27] M. S. Nikoo and F. Behnia, "Single-site source localisation using scattering data," *IET Radar, Sonar & Navigation*, vol. 12, no. 2, pp. 250–259, 2017.
- [28] O. Bar-Shalom and A. J. Weiss, "Transponder-aided single platform geolocation," *IEEE Transactions on Signal Processing*, vol. 61, no. 5, pp. 1239–1248, 2012.
- [29] O. Bar-Shalom and A. J. Weiss, "Emitter geolocation using single moving receiver," *Signal Processing*, vol. 105, pp. 70–83, 2014.

- [30] R. Coifman and S. Lafon, "Diffusion maps," *Applied and computational harmonic analysis*, vol. 21, no. 1, pp. 5–30, 2006.
- [31] A. Singer and R. R. Coifman, "Non-linear independent component analysis with diffusion maps," *Applied and Computational Harmonic Analysis*, vol. 25, no. 2, pp. 226–239, 2008.
- [32] D. Kushnir, A. Haddad, and R. Coifman, "Anisotropic diffusion on sub-manifolds with application to earth structure classification," *Applied and Computational Harmonic Analysis*, vol. 32, no. 2, pp. 280–294, 2012.
- [33] F. R. Chung and F. C. Graham, *Spectral graph theory*. No. 92, American Mathematical Soc., 1997.
- [34] B. Nadler, S. Lafon, R. R. Coifman, and I. G. Kevrekidis, "Diffusion maps, spectral clustering and reaction coordinates of dynamical systems," *Applied and Computational Harmonic Analysis*, vol. 21, no. 1, pp. 113–127, 2006.
- [35] B. Nadler, S. Lafon, I. Kevrekidis, and R. R. Coifman, "Diffusion maps, spectral clustering and eigenfunctions of fokker-planck operators," in *Proc. Advances in neural information processing systems*, pp. 955–962, 2006.
- [36] E. J. Nyström *et al.*, "Über die praktische auflösung von integralgleichungen mit anwendungen auf randwertaufgaben," *Acta Mathematica*, vol. 54, pp. 185–204, 1930.
- [37] E. A. Habets, "Room impulse response generator," *Technische Universiteit Eindhoven, Tech. Rep*, vol. 2, no. 2.4, p. 1, 2006.
- [38] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *The Journal of the Acoustical Society of America*, vol. 65, no. 4, pp. 943–950, 1979.
- [39] S. Theodoridis, K. Koutroumbas, *et al.*, "Pattern recognition," *IEEE Transactions on Neural Networks*, vol. 19, no. 2, p. 376, 2008.
- [40] Y. Wang and B. Chaib-Draa, "A knn based kalman filter gaussian process regression," in *Proc. Twenty-Third International Joint Conference on Artificial Intelligence*, Citeseer, 2013.
- [41] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes in Machine Learning*. MIT Press, 2006.

**איכון מטרות נעות על ידי חיישן בודד
באמצעות גרעיני דיפוזיה**

ערן זיתוני

איכון מטרות נעות על ידי חיישן בודד באמצעות גרעיני דיפוזיה

חיבור על מחקר

לשם מילוי חלקי של הדרישות
לקבלת התואר מגיסטר למדעים בהנדסת חשמל

ערן זיתוני

הוגש לסנט הטכניון - מכון טכנולוגי לישראל

מחקר זה נעשה תחת הנחייתו של פרופסור ישראל כהן, בפקולטה להנדסת חשמל ומחשבים
ע"ש אנדרו וארנה ויטרבי.

תודות

ברצוני להביע את תודתי והערכתי לפרופסור ישראל כהן על מתן ההזדמנות, הנחייתו, תמיכתו
וסבלנותו לאורך שלבי המחקר.
אבקש להודות לפרופסור רונן טלמון ופרופסור יוסי קלר על סקירת המחקר ובחינתו.
לסיום, ברצוני להודות למשפחתי וחבריי על אהבתם, אמונם, תמיכתם והעידוד. הישג זה
אינו היה אפשרי בלעדיהם.

אני מודה לטכניון, לתכנית המחקר המשותפת של הקרן הלאומית למדע והקרן הלאומית
למדעי הטבע של סין (מענק מס' 2514/17), ולקרן פזי, על תרומתם הכספית הנדיבה
שאפשרה את מחקר זה.

פרסומים

החיבור על המחקר מבוסס על הפרסומים הבאים:

E. Zeitouni and I. Cohen, "Single-sensor source localization of moving sources using diffusion kernels", Submitted to Applied Acoustics, 2021.

E. Zeitouni and I. Cohen, "Single-sensor source localization of moving sources using diffusion kernels and Brownian motion model", Submitted to IEEE International Conference on Communications, ICC-2022, Seoul, South Korea, May 16-22, 2022.

תקציר

מחקר זה מתמקד באיכון פסיבי של מקורות אקוסטיים נעים, באמצעות מיקרופון יחיד (המחקר עוסק באיכון מקורות נייחים באופן משני). המטרה המרכזית של המחקר היא פיתוח שיטה לטובת שיערוך מדוייק של מיקום ווקטור מהירות של מקור אקוסטי בעת תנועתו, על אף אי הוודאות שאיכון באמצעות חיישן יחיד טומן בחובו וכן היעדר ידע על פונקציית התמסורת של המערכת. לשם כך, שיטה זו צריכה להתמודד עם אתגר נוסף: ניצול נתיבי ההתפשטות הבלתי-ישירים של האות המשודר על ידי המקור.

בנוסף, איכון של מקורות נעים מוסיף נדבך נוסף של קושי הנובע מעצם תנועתם, מאחר שתנועה זו מובילה לתלות המערכת בזמן. המערכת, שכניסתה היא האות המשודר על ידי המקור ומוצאה היא האות הנקלט על ידי החיישן, מאופיינת באמצעות תגובת ההלם האקוסטית. תלות המערכת בזמן מתבטאת באמצעות תגובת ההלם האקוסטית, כך ששינויים בתגובת ההלם האקוסטית מובילים לתלות של וקטור המאפיינים (המייצגים את האות הנקלט) בזמנים מפורשים.

אנו מציעים גישת למידת יריעות מונחית, ובפרט מפות דיפוזיה, לשם זיהוי מאפיינייה של המערכת. בהתבסס על גישה זו, המונעת על פי הנתונים, אנו משחזרים את הפרמטרים הפיזיקליים החבויים, השולטים במערכת - כלומר, המיקום ווקטור המהירות של המקור.

כל מקור, בתורו, משדר אות רעש לבן גאוסי במהלך תנועתו, הנקלט על ידי החיישן. האות הנקלט מחולק למסגרות זמן. המסלול שנוצר באמצעות תנועת המקור בעת מסגרת הזמן, מקורב באמצעות מקטע של תנועה לינארית. כל מסגרת זמן נבחנת באופן פרטני לשם שיערוך המיקום ווקטור המהירות של המקור. האות אשר נקלט בעת מסגרת הזמן עובר עיבוד לטובת הפקת וקטור מאפיינים. באופן זה, אנו אוספים סט אימון של תצפיות שמוצאן במקורות שונים, אשר ערכי המיקום ווקטור המהירות שלהם ידועים.

באמצעות הגבלת התנועה למהירות איטית ווקטור מהירות המשתנה באופן הדרגתי, אנו מביאים לשינויים קלים בתגובת ההלם האקוסטית לאורך מסגרות הזמן השונות. כתוצאה מכך, ניתן להתייחס למערכת כקבועה בזמן בקירוב, בהינתן מסגרות זמן קצרות דיו. על כן, וקטור המאפיינים חווה שינויים קלים אף הוא, והפרשי הזמן בערכיו תלויים בשינויים של תגובת ההלם האקוסטית בלבד. שינויים אלו של תגובת ההלם האקוסטית נובעים מהתפתחותן של דרגות החופש היחידות של המערכת: המיקום ווקטור המהירות של המקור - כלומר, ניתן לשחזרם.

באמצעות שימוש במפות דיפוזיה בעזרת גרעין דיפוזיה המבוסס על מרחק אוקלידי, לטובת אומדן הקירבה בין וקטורי המאפיינים, אנו לומדים את המבנה הלא לינארי של

יריעת הנתונים. הנתונים מאורגנים על גבי היריעה על פי ערכי המיקום וקטור המהירות של המקורות. בנוסף, אנו דנים בנוגע לחלופות נפוצות להערכת הקירבה (למשל, מרחק מהלנוביס) ומסבירים את בחירתנו.

באמצעות ביצוע של פירוק ספקטרלי של מטריצת הקירבה של כל תצפיות האימון, אנו מוצאים את ההעתקה שלהן מהמרחב הרב-מימדי (בו ניתן לחזות בהן) למרחב בעל מספר מועט של מימדים. בהינתן סט תצפיות שמוצאן במקורות, אשר ערכי המיקום וקטור המהירות שלהם אינם ידועים, אנו משתמשים בשיטת נייסטרום לטובת הרחבה. יתרה מזו, אנו דנים בחסרונות של שיטות ההרחבה המוצעות בעבודות קודמות, משפרים אותן ומספקים מבנה מדויק ליריעה המורחבת.

המיקום והמהירות של המקור ניתנים לשחזור באמצעות תצפיות האימון הקרובות ביותר לתצפית המקור במרחב בעל מספר המימדים המועט. שחזור המיקום וקטור המהירות נקבע על פי נקודת האמצע של של המקטע. בהתבסס על נקודה זו, אנו ממזערים את שגיאת הקירוב של המסלול האמיתי באמצעות מקטע לינארי. האומדן הרלוונטי למדידת דיוק השערוך, המוצע בעבודות קודמות, מותאם ומורחב על מנת לכמת את שגיאת השערוך עבור מקור נע. בנוסף, אנו דנים בנוגע לסיבות להתאמה והרחבה זו (למשל, שילוב של גדלים פיזיקליים בעלי יחידות שונות).

היכולות והמגבלות של האלגוריתם המוצע לאיכון באמצעות סנסור יחיד נבחנות בשלושה מקרים:

1. מקרה בסיס של מקורות נייחים, אשר מורכב משני תתי-מקרים: חד-מימדי (אזימוט) ודו-מימדי (רדיוס ואזימוט).

2. מקרה נרחב של מקורות הנעים באופן דטרמיניסטי, המורכב ממספר ניסויים הבוחנים את הרגישות לגורמים שונים, כגון: היפרפרמטרים (למשל, אורך מסגרת הזמן), משתנים (למשל, מהירות וכיוון התנועה) ותנאים (למשל, זמן רברברציה). מקרה זה הוא התרומה המרכזית של מחקרנו.

3. מקרה של מקורות הנעים בהשפעת תנועה בראונית, אשר מייצגת את השפעתם של תנאים סביבתיים על תנועתם של מקורות באמצעות תזוזות מהירות ואקראיות. מקרה זה בוחן את היכולת של האלגוריתם להתמודד עם הפרת הנחה בסיסית הנוגעת לשינויים ההדרגתיים של וקטור המהירות.

אנו מבססים הוכחת היתכנות המציעה את הפתרון המתקדם ביותר לשחזור המיקום וקטור המהירות של מקור נע באמצעות חישן יחיד. האלגוריתם מבטא את חוסנו כלפי סביבות רועשות ורוויות ברברציות, אך מאידך גם את רגישותו כלפי שינויים בתנאים הסביבתיים. התוצאות מאמתות את נחיצות החזרי האות מהסביבה לשם הפקת שערוך מדויק. התוצאות ממחישות את השפעת המהירות של המקורות, המתבטאת בשקלול תמורות בין הדיוק של המיקום המשוערך לבין הדיוק של הכיוון המשוערך. בנוסף, אנו חוזים בהשפעת של כוחות סותרים כגון התוקף של ההנחות שלנו בנוגע למאפייני וקטור המהירות, אל מול תכונות התלות בזמן של המערכת. האלגוריתם מתקשה להבחין בין מהירויות שונות של מקורות הנעים באותו הכיוון. בעוד האלגוריתם נועד לשחזור המיקום וקטור המהירות של מקורות איטיים שמשנים את וקטור המהירות בצורה הדרגתית, האלגוריתם אף מסוגל לשערך בהצלחה את

המיקום ווקטור המהירות הממוצעת של מקורות איטיים אשר משנים את וקטור המהירות שלהם בצורה מהירה ואקראית.
בנוגע למקורות נייחים, האלגוריתם מפיק את התוצאות המוצלחות ביותר בתחום עבור שערך של מיקום חד-מימדי ומיקום דו-מימדי, כאחד.