State-space partitioning via Granger causality for local particle filters

Andrea Quintanilla
School of Mathematics
University of Edinburgh
Edinburgh, UK
A.Quintanilla-Carranza@sms.ed.ac.uk

Víctor Elvira
School of Mathematics
University of Edinburgh
Edinburgh, UK
victor.elvira@ed.ac.uk

***Abstract*—Particle filtering (PF) is a widely used technique for approximating distributions in statistical signal processing, particularly in non-linear and non-Gaussian state-space models. However, its efficiency depends on the number of particles, which is significantly affected by the dimensionality of the latent state. As a result, PF has traditionally been less effective in high-dimensional settings. Local particle filters (LPFs) address this limitation by performing PF steps within smaller regions of the space, improving estimation accuracy even with fewer particles. However, their performance relies on designing an appropriate partitioning of the space. In this work, we propose a method to automatically identify structure within the state space based on Granger causality, enabling the creation of effective partitions. Our method can be applied to a large class of LPFs. We also discuss possible extensions and demonstrate the good performance of our approach in numerical experiments.**

***Index Terms*—Particle filtering, state-space models, local particle filters, state partitioning, Granger causality.**

I. Introduction

Particle filtering (PF) is a powerful sequential Monte Carlo method used for estimating hidden states in dynamic systems when traditional analytical solutions are infeasible. It addresses problems where uncertainty evolves over time, such as financial modelling and robotics, by approximating probability distributions with a set of weighted samples. Alternatives include the Kalman filter (KF) and its variants, but PF excels in handling non-linear and non-Gaussian scenarios [1], which makes it especially relevant to climate applications [2], [3].

Particle filtering faces significant challenges in high-dimensional settings, primarily due to the large number of particles required for accurate state estimation. Several strategies have been proposed to mitigate this issue, for example, using special proposals and modifying the resampling steps [4], [5].

Here we focus on the local particle filter (LPF) approach, which contains a large class of methods [6]. Following the divide-and-conquer strategy, in LPFs, several filters are applied, each to a smaller subset of the dimensions of the original state space; some examples can be found in [7]–[9].

LPFs are inspired by the idea that some regions evolve approximately independently over short periods, and that not all observations have information about all state dimensions. These methods have been shown to be effective in reducing weight degeneracy. However, existing approaches predominantly assume a spatial structure where state and observation components correspond to fixed geographic locations [6], and hence an idea of an adequate local area is known. The experimental results of relevant previous work, as in [10] and [11], indicate that the performance of these methods is highly sensitive to the chosen notion of locality. In this paper, we propose a novel method that leverages Granger causality to identify neighbourhood relationships, enabling automatic partitioning without the need for predefined spatial structures. Our approach learns partitions tailored to the underlying system dynamics, extending the applicability of LPFs beyond systems with known neighbourhood structure. We frame the proposed method in a more general conditional independence scheme that offers a promising way to formalize the connection between the the causes of weight degeneracy with the LPF framework.

The rest of the paper is structured as follows. Section II outlines the problem and reviews relevant LPF literature. Section III describes our method, tested in Section IV with numerical examples. Conclusions are drawn in Section V.

II. Problem statement and background

A. Problem statement

Consider a state-space model (SSM) defined as follows:

$$x_0 \sim p(x_0), \tag{1}$$

$$x_t \mid x_{t-1} \sim p(x_t \mid x_{t-1}), \tag{2}$$

$$y_t \mid x_t \sim p(y_t \mid x_t), \tag{3}$$

where $x_t \in \mathbb{R}^{d_x}$ is the state vector at time t, $y_t \in \mathbb{R}^{d_y}$ is the observation vector at time t. All model parameters that are considered to be known in this paper. The Bayesian filtering problem aims at computing the posterior distribution $p(x_t \mid y_{1:t})$, which in general SSMs can only be approximated through particle filters [1], [12] as

$$p(x_t \mid y_{1:t}) \approx \sum_{i=1}^{N_p} w_t^{(i)} \delta_{x_t^{(i)}}(x_t), \tag{4}$$

The work of A. Q. is supported by the Consejo Nacional de Humanidades, Ciencias y Tecnologías (CONAHCYT). The work of V. E. is supported by ARL/ARO under W911NF-22-1-0235 and by ARIA under PROTECT project.

where N_p is the number of particles; $x_t^{(i)}$ is the i-th particle, and $w_t^{(i)}$ is its associated importance weight, both at time t.

B. Local particle filters (LPFs)

Localized Particle Filters (LPFs) leverage standard PF steps (propagation, weighting, resampling) but restrict computations to relevant state-space regions [6], [13], [14]. Locality is formalized by partitioning the state vector x_t into K disjoint blocks $\{B_k\}_{k=1}^K$, where each $B_k \subset \{1, \ldots, d_x\}$ and $\bigcup_{k=1}^K B_k = \{1, \ldots, d_x\}$. Block-specific notation is used (e.g., $w_{t,k}^{(i)}$ denotes the weight for block k of particle i at time t).

C. An LPF example with adaptive partitioning

We now describe a concrete example of LPF, in Algorithm 1, called the block PF [9]. We first sample from a prior distribution (line 1). Then, at each time step, we sample from the proposal distribution (line 3). Next, we create a partition of the state space (line 4). For each block in the new partition, we calculate the local weights $w_{t,k}^{(i)}$ and normalize them within each block (line 6). We then resample new particles $\widetilde{x}_{t,k}^{(i)}$ (line 7). Finally, we concatenate these particles to obtain global particles for the prediction step at $t+1$ (line 9). The step of creating a partition (line 4) is not proposed in the original block PF. But we added the adaptive partitioning, as in [10], where partitions that are adapted to the intrinsic system dynamics broaden the utility of a large spectrum of LPFs beyond systems with a predefined neighbourhood notion.

Algorithm 1 A LPF with adaptive partitioning

1: Initialize $\widetilde{x}_0^{(i)} \sim p(x_0)$ for $i = 1, \ldots, N_p$.
2: **for** $t = 1, 2, \ldots$ **do**
3: Sample particles $x_t^{(i)} \sim p(x_t \mid \widetilde{x}_{t-1}^{(i)})$, $i = 1, \ldots, N_p$.
4: **Obtain the state-space partition** $\{B_k\}_{k=1}^K$ with a partitioning algorithm.
5: **for** $k = 1, \ldots, K$ **do**
6: Compute block weights and normalize them
7: $w_{t,k}^{(i)} \propto p(y_{t,k} \mid x_{t,k}^{(i)})$.
8: Resample N_p particles $\widetilde{x}_{t,k}^{(i)}$, using the weights
9: $w_{t,k}^{(i)}$ of the block.
10: **end for**
11: Reassemble resampled particles and approximate the filtering distribution as in Eq. (4).
12: **end for**

D. LPF approaches

a) Propagation: in LPFs involves sampling particles either locally (by block) or globally. The local approach, as in [15], samples each block independently from $p(x_t^{(k)} \mid x_{t-1}^{(k)})$, requiring the transition to factorize across blocks as

$$p(x_t \mid x_{t-1}) = \prod_{k=1}^{K} p\left(x_{t,k} \mid x_{t-1,k}\right). \tag{5}$$

For systems where this factorization is unknown, a global approach, as illustrated in the Algorithm 1, can be used; it propagates full-state particles but requires reassembling local particles. This reassembling can lead to particles that are unlikely or even impossible under the model. For that reason, LPFs may include smoothing mechanisms to address these discontinuities [16]. Another option is to estimate the transitions $p(x_t^{(k)} \mid x_{t-1}^{(k)})$, as shown in [7].

b) Weighting: LPFs aim to mitigate weight collapse, a common phenomenon in PFs where a single particle accumulates nearly all the weight, by computing *local* weights. The local approach, as shown in the Algorithm 1, assumes factorized likelihoods,

$$p\left(y_t \mid x_t\right) = \prod_{k=1}^{K} p\left(y_{t,k} \mid x_{t,k}\right). \tag{6}$$

Then weights depend only on information of the block where, if the bootstrap proposal is used, they take the form $w_t^{(k)} \propto p(y_t^{(k)} \mid x_t^{(k)})$. Again, for non-factorizable models, approximations of local weights can be made, as proposed in [17]–[19], though performance degrades if the estimations are not adequate.

c) Resampling: in LPFs can be performed locally, as in Algorithm 1, where each block is resampled independently using its own weights. However, there are approaches that aim to resample global particles to enable estimations or global propagation. These methods aim to preserve the plausibility of the global particles while retaining the benefits of local weighting [4], [6], [20].

E. Partitions of the state space for LPF

Many LPFs require a state-space partition in addition to the usual inputs of a standard particle filter. It is often assumed a spatial structure in the SSM, where each state and observation component is linked to a spatial location, and a distance function exists between these locations [6]. Our proposed method, explained in Section III, explores alternative ways to define proximity when components are not tied to physical locations. In [11], a method for partitioning SSMs is presented; it is based on the structure of the transition distribution. Those partitioning methods are useful for SSMs where the component relationships are clearly defined and constant over time. Other approaches with adaptive partitioning are presented in [21], which employ multiple partitions with parallel local particle filter instances, and in [10], which constructs partitions based on covariance between dimensions.

III. Proposed method

We propose the *Granger constrained spectral clustering* (G-CSC) framework for Bayesian filtering in high-dimensional spaces. Our approach automatically finds a suitable partition of the dimensions in the state space based on Granger causality (GC). GC typically measures whether one time series improves the prediction of another one; it has been widely applied in signal processing and statistics [22], [23]. GC generally deals with observed time series; however, our G-CSC framework allows us instead to deal with both the observation and latent processes.

A. Granger causality for LPFs

Consider the trajectories of the true state, $\{x_t\}_t$, and observations, $\{y_t\}_t$. Given an index $0 \leq q \leq d_x$, we denote by $x(q) = \{x_{t,q}\}_t$ the time series formed only by the q-th dimension of the true state trajectory. Let $0 \leq q, r \leq d_x$, according to the definition of *strong Granger causality* [24], we say that the time series $x(r)$ is Granger non-causal for the time series $x(q)$ if

$$\forall t : p(x_{t,q} \mid x_{1:t-1,q}, x_{1:t-1,r}) = p(x_{t,q} \mid x_{1:t-1,q}), \tag{7}$$

or equivalently, $x_{t,q} \perp x_{1:t-1,r} \mid x_{1:t-1,q}$. Getting back to our purpose of partitioning the state, if up to a time t, the past values of $x(r)$ do not affect the present $x_{t,q}$, given the past of $x(q)$, then it is reasonable for the r-th and q-th dimensions of the particles at time t, to be in different blocks for the weighting step computations of Algorithm (1).

Note that in (7), we only used information from the state space. However, it can be beneficial to use the observation y_t to design the partition. For instance, it has been shown that the variability $\tau^2 = var\left[\ln p(y_t \mid x_t^{(i)})\right]$, in some settings, relates to the weight degeneracy and that τ^2 can be related to the dimension of the state space [25]. Consider then, the following conditional independence:

$$y_t \perp x_{t,r} \mid x_{t,q}. \tag{8}$$

The interpretation is that if, given $x_{t,q}$, y_t is independent of $x_{t,r}$, then the computation of the weight of the i-th particle can be done with $x_{t,r}^{(i)}$ and $x_{t,q}^{(i)}$ being in different blocks.

B. Approximating GC for LPFs

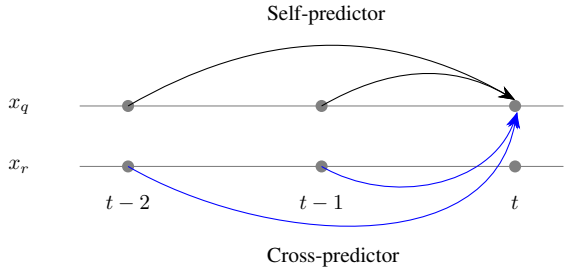
For practical implementation, we need to address that in the particle filter scenario we do not have access to the true states, and that the previous conditional independences can challenging to verify. Instead of using the true states to estimate GC up to a time t, we will use the particles $\{x_{t-m}\}_{m=1}^M$, where $M > 0$ is known as the predictor order. And as approximations of a measure of (7) and (8), we will use *pairwise Granger causality* (pairwise GC); this instead of the probability examines predictability by fitting an autoregressive model. In the following we describe how to do it for (7), the process to approximate (8) is analogous. With pairwise GC we compare the predictability of the present q-th dimension with: (a) a model that uses only information from the past of the q-th dimension, to the power of prediction of (b) a model that also uses the r-th past. To implement (a), we use the following linear estimation of $x_{t,q}$, known as the self-predictor model,

$$x_{t,q} = \sum_{m=1}^{M} \alpha_{qq}[m] x_{t-m,q} + \varepsilon_q[t];$$

to implement (b), we use the cross-predictor (see Figure III-B)

$$x_{t,r\to q} = \sum_{m=1}^{M} \beta_{qq}[m] x_{t-m,q} + \sum_{m=1}^{M} \beta_{rq}[m] x_{t-m,r} + \varepsilon_{r\to q}[t].$$

Fig. 1. Representation of the information used to predict $x_{t,q}$ in Granger causality: the self-predictor (using the past of signal x_q) and the cross-predictor (using also the past of x_r, shown in blue).



Where $\alpha_{qq}[m]$, and $\beta_{qq}[m], \beta_{rq}[m]$ are coefficients that are estimated using linear regressions; $\varepsilon_q[t]$ and $\varepsilon_{r\to q}[t]$ are the residual errors of the self-prediction and cross-prediction models, respectively. The particles are used to do the regression. Particles of the present, $x_{t,k}^{(i)}$, are the regression value and the corresponding past values, $\{x_{t-m,k}^{(i)}\}_{m=1}^M$, are the regressors. The pairwise Granger causality strength is then given by

$$G_{r\to q}[t] = \ln \frac{\text{Var}(\varepsilon_q[t])}{\text{Var}(\varepsilon_{r\to q}[t])}, \tag{9}$$

A causality strength matrix $G[t]$ is constructed, where each entry $G[t]_{r,q} = G[t]_{r\to q}$ represents the causality strength from the dimension r to the dimension q at time t; higher values provide stronger evidence that r Granger-causes q.

C. Partitioning the space with GC

We now describe a way to create a partition of the state space, $\{B_k\}_{k=1}^K$, based on GC, at each time t, as required in line 4 of Algorithm 1. To do this, we apply to the causality strength matrix $G[t]$ derived in (9) a constrained spectral clustering (CSC), as proposed in [10]. *Spectral clustering* connects to graphical approaches in SSMs [26], [27] as a graph-clustering method based on low-dimensional representations. The procedure is described in Algorithm 2 and requires particles $\{x_{t-m}^{(i)}\}_{m=0}^M$, the number of blocks K, the maximum block size ζ, and the number of eigenvalues to be retained ℓ. First, a pairwise Granger causality measure is computed for the state dimensions, as described previously, in Section III-A, and used to construct a similarity matrix (line 2). To ensure the matrix's symmetry and non-negativity, required for SC, the following steps are applied: the minimum value is subtracted from all entries; the diagonal is set to the maximum value,

Algorithm 2 G-CSC for partitioning the space

1: Approximate the Granger causality matrix G using $\{x_{t-m}^{(i)}\}_{m=0}^M$, as described in (9).
2: Compute the graph Laplacian L of G and store the eigenvectors associated with the ℓ smallest eigenvalues in a matrix $V \in \mathbb{R}^{n\times\ell}$.
3: Apply K-means clustering to the rows of V, subject to the block size constraint ζ, to obtain the partition $\{B_{t,k}\}_{k=1}^K$.

TABLE I
MSE PERFORMANCE OF DIFFERENT PARTICLE FILTERS IN THE LORENZ 96 MODEL WITH $d_x = 60$.

N_p method	320	640	1280	2560
oracle	0.2369 (0.0052)	0.2252 (0.0035)	0.2203 (0.0019)	0.2175 (0.0017)
cov-CSC	0.2455 (0.0066)	0.2353 (0.0046)	0.2295 (0.0036)	0.2238 (0.0031)
Gx-CSC	0.2573 (0.0094)	0.2427 (0.0039)	0.2353 (0.0038)	0.2317 (0.0036)
Gy-CSC	0.2584 (0.0068)	0.246 (0.0061)	0.2383 (0.0046)	0.2359 (0.0023)
random	0.2689 (0.0094)	0.26 (0.0061)	0.2557 (0.0036)	0.2513 (0.0032)
bootstrap PF	11.3548 (1.8664)	10.0843 (1.8861)	9.2342 (1.5181)	7.9534 (1.6953)

TABLE II
ESS PERFORMANCE OF DIFFERENT PARTICLE FILTERS IN THE LORENZ 96 MODEL WITH $d_x = 60$.

N_p method	320	640	1280	2560
Gx-CSC	161.5159 (0.8282)	319.0632 (1.5842)	632.4754 (3.8463)	1260.6273 (6.5207)
Gy-CSC	157.3882 (1.0214)	309.7498 (2.0326)	611.8503 (3.7483)	1213.0253 (7.6419)
cov-CSC	156.7076 (0.9631)	306.1528 (1.3543)	604.0699 (3.402)	1189.0015 (7.8475)
oracle	139.645 (0.5169)	274.4733 (1.0842)	543.0379 (1.5278)	1078.6464 (1.7734)
random	135.5494 (1.0653)	267.5566 (1.6485)	530.5982 (2.1929)	1058.5751 (5.5388)
bootstrap PF	3.333 (0.3513)	4.4217 (0.4696)	5.5188 (0.8258)	7.2378 (0.9395)

and the matrix is symmetrized by adding it to its transpose. CSC is then applied to the resulting matrix following [10]: it calculates the Laplacian L of the matrix $G[t]$, along with its ℓ smallest eigenvalues (line 3); and finally, to determine the blocks and to prevent excessively large ones, it uses the constrained k-means algorithm [28] (line 4).

D. Discussion and future work

We have used the LPF [9] to provide a concrete method in the Algorithm 1. However, the partitions generated by our method could be used in any LPF that requires partitions as an input; some examples are shown in [6]. The partitions in the Algorithm 1 are updated at each time step, though they could be updated less frequently. One of our approaches uses only information from the states (7), so we expect our partitioning scheme to yield superior performance in methods that perform the prediction step by blocks. A promising avenue for future work is estimating non-linear Granger causality, which is ideal for typical PF applications.

IV. NUMERICAL EXPERIMENTS

We evaluate the proposed method G-CSC, with predictor order $M = 1$. To test, we use a stochastic version of the Lorenz 96 model, commonly employed in the PF literature to explore high-dimensional scenarios [29] and climate forecasting [30]. This model can exhibit chaotic behaviour, and its deterministic version is given by the following set of differential equations:

$$\frac{\mathrm{d}x_q}{\mathrm{d}t} = (x_{q+1} - x_{q-2})\, x_{q-1} - x_q + F, \quad q = 1 \dots d_x,$$

where the indices are to be understood with periodic boundary conditions: $x_{-1} = x_{d_x-1}, x_0 = x_{d_x}$, and $x_1 = x_{d_x+1}$, and where $d_x = 60$ and $F = 8$, which produce chaotic dynamics [31]. A fourth-order Runge-Kutta method with a time step of 0.05 is used for integration. The state-space model that we use for simulations is then given by

$$x_{t+1,q} = x_{t,q} + \Delta t \frac{\mathrm{d}x_{t,q}}{\mathrm{d}t} + w_{t+1,q} \tag{10}$$

$$y_{t+1} = x_{t+1} + v_t, \tag{11}$$

where $w_{t+1,q} \sim \mathcal{N}(0, \Delta t), v_t \sim \mathcal{N}(0, I)$. We compare the performance of bootstrap PF and block PF with different partition types: the proposed G-CSC partitions, the version based on (7), Gx-CSC, and the one based on (8), Gy-CSC; the method proposed in [10], that instead of a Granger matrix uses a covariance matrix, cov-CSC; an oracle method where consecutive indexes of the state space are conveniently grouped together in blocks of the same size and kept fixed for all time steps (called oracle); and finally, a method where indexes of the state space are randomly grouped in blocks of the same size (called random). They were applied with exponentially increasing numbers of N_p particles, $K = 12$ blocks for each partition and with $T = 100$ the total number of steps used in the SSM. The results of the simulations can be seen in Table I. The performance of the filters was evaluated using the mean squared error (MSE) as a metric, defined as

$$\mathrm{MSE}(\widehat{x}) = \frac{1}{d_x}\frac{1}{T}\sum_{t=1}^{T}\sum_{q=1}^{d_x}\left(\hat{x}_{q,t} - x_{q,t}\right)^2,$$

where $\widehat{x} \in \mathbb{R}^T$ is the mean of the trajectories of the estimated particles. The MSE results were averaged over $S = 25$ independent runs for each system. We also show the MSE standard deviation -between parentheses- over runs, to be able to evaluate statistical significance. Finally, in Table II, we display the mean effective sample size (ESS) across blocks,

$$\mathrm{ESS} = \frac{1}{K}\frac{1}{T}\sum_{t=1}^{T}\sum_{k=1}^{K}\frac{1}{\sum_{i=1}^{N} w_{t,k}^{(i)}}.$$

We can see that the bootstrap PF has a poor performance in this complicated high-dimensional model. The block approaches help mitigate the curse of dimensionality in all partitioning approaches, including random partitioning. Our methods, Gx-CSC and Gy-CSC, achieve an MSE not far from the oracle version but higher than cov-CSC. They also outperform the random partitioning approach. Finally, we find that the gain in ESS of our methods with respect to all other approaches is statistically significant.

V. CONCLUSION

In this work, we have proposed a framework for performing inference in high-dimensional state-space models. We designed a method based on Granger causality to automatically identify structure within the latent space. The framework, implemented in a block particle filter for concreteness in the explanation, can be easily adapted to operate with generic local particle filters and different proposal choices of the filters. We also discussed possible extensions and demonstrated the effectiveness of our framework through a challenging numerical experiment using the Lorenz 96 model.

REFERENCES

[1] S. Särkkä and L. Svensson, *Bayesian filtering and smoothing*, vol. 17, Cambridge university press, 2023.

[2] M. Bocquet, C. A. Pires, and L. Wu, "Beyond Gaussian Statistical Modeling in Geophysical Data Assimilation," *Monthly Weather Review*, Aug. 2010.

[3] P. J. Van Leeuwen, "Particle filtering in geophysical systems," *Monthly Weather Review*, vol. 137, no. 12, pp. 4089–4114, 2009.

[4] S. Reich, "A nonparametric ensemble transform method for bayesian inference," *SIAM Journal on Scientific Computing*, vol. 35, no. 4, pp. A2013–A2024, 2013.

[5] O. Kviman, N. Branchini, V. Elvira, and J. Lagergren, "Variational resampling," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2024, pp. 3286–3294.

[6] A. Farchi and M. Bocquet, "Comparison of local particle filters and new implementations," *Nonlinear Processes in Geophysics*, vol. 25, no. 4, pp. 765–807, 2018.

[7] P. M. Djuric, T. Lu, and M. F. Bugallo, "Multiple particle filtering," in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*. IEEE, 2007, vol. 3, pp. III–1181.

[8] P. Closas and M. F. Bugallo, "Improving accuracy by iterated multiple particle filtering," *IEEE Signal Processing Letters*, vol. 19, no. 8, pp. 531–534, 2012.

[9] P. Rebeschini and R. Van Handel, "Can local particle filters beat the curse of dimensionality?," *Ann. Appl. Probab.*, pp. 2809–2866, 2015.

[10] R. Min, C. Garnier, F. Septier, and J. Klein, "State space partitioning based on constrained spectral clustering for block particle filtering," *Signal Processing*, vol. 201, pp. 108727, 2022.

[11] M. Iloska and M. F. Bugallo, "State-space partitioning schemes in multiple particle filtering for improved accuracy," in *2022 30th European Signal Processing Conference (EUSIPCO)*. IEEE, 2022.

[12] P. M. Djuric, J. H. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M. F. Bugallo, and J. Miguez, "Particle filtering," *IEEE signal processing magazine*, vol. 20, no. 5, pp. 19–38, 2003.

[13] P. J. Van Leeuwen, H. R. Künsch, L. Nerger, R. Potthast, and S. Reich, "Particle filters for high-dimensional geoscience applications: A review," *Quarterly Journal of the Royal Meteorological Society*, vol. 145, no. 723, pp. 2335–2365, 2019.

[14] P. J. Van Leeuwen, Y. Cheng, S. Reich, and P. J. van Leeuwen, *Nonlinear Data Assimilation for high-dimensional systems: -with geophysical applications*, Springer, 2015.

[15] P. M. Djurić and M. F. Bugallo, "Multiple particle filtering with improved efficiency and performance," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2015, pp. 4110–4114, ISSN: 2379-190X.

[16] J. Poterjoy, "A localized particle filter for high-dimensional nonlinear systems," *Monthly Weather Review*, vol. 144, no. 1, pp. 59–76, 2016.

[17] M. F. Bugallo, T. Lu, and P. M. Djuric, "Target tracking by multiple particle filtering," in *2007 IEEE aerospace conference*. 2007, pp. 1–7, IEEE.

[18] B. Ait-El-Fquih and I. Hoteit, "A variational Bayesian multiple particle filtering scheme for large-dimensional systems," *IEEE Transactions on Signal Processing*, vol. 64, no. 20, pp. 5409–5422, 2016, Publisher: IEEE.

[19] L. Úbeda Medina, A. F. García-Fernández, and J. Grajal, "Sigma-point multiple particle filtering," *Signal Processing*, vol. 160, pp. 271–283, 2019, Publisher: Elsevier.

[20] P. J. Van Leeuwen, "Particle filtering in geophysical systems," *Monthly Weather Review*, vol. 137, no. 12, pp. 4089–4114, 2009.

[21] R. Min, C. Garnier, F. Septier, and J. Klein, "Parallel block particle filtering," in *2021 IEEE Statistical Signal Processing Workshop (SSP)*. IEEE, 2021, pp. 86–90.

[22] C. W. Granger, "Investigating causal relations by econometric models and cross-spectral methods," *Econometrica: journal of the Econometric Society*, pp. 424–438, 1969.

[23] D. Luengo, G. Rios-Munoz, V. Elvira, C. Sanchez, and A. Artes-Rodriguez, "Hierarchical algorithms for causality retrieval in atrial fibrillation intracavitary electrograms," *IEEE Journal of Biomedical and Health Informatics*, vol. 23, no. 1, pp. 143–155, 2018.

[24] A. Shojaie and E. B. Fox, "Granger causality: A review and recent advances," *Annual Review of Statistics and Its Application*, vol. 9, no. 1, pp. 289–319, 2022.

[25] C. Snyder, T. Bengtsson, P. Bickel, and J. Anderson, "Obstacles to High-Dimensional Particle Filtering," *Monthly Weather Review*, Dec. 2008.

[26] V. Elvira and É. Chouzenoux, "Graphical inference in linear-gaussian state-space models," *IEEE Transactions on Signal Processing*, vol. 70, pp. 4757–4771, 2022.

[27] E. Chouzenoux and V. Elvira, "Sparse graphical linear dynamical systems," *Journal of Machine Learning Research*, vol. 25, no. 223, pp. 1–53, 2024.

[28] P. S. Bradley, K. P. Bennett, and A. Demiriz, "Constrained k-means clustering," Tech. Rep. 20.0, Microsoft Research, Redmond, 2000.

[29] B. Cox, E. Chouzenoux, and V. Elvira, "Graphgrad: Efficient estimation of sparse polynomial representations for general state-space models," *arXiv preprint arXiv:2411.15637*, 2024.

[30] D. S. Wilks, "Effects of stochastic parametrizations in the Lorenz '96 system," *Quarterly Journal of the Royal Meteorological Society*, vol. 131, no. 606, pp. 389–407, 2005, _eprint: https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1256/qj.04.03.

[31] S. Pérez-Vieites, H. Molina-Bulla, and J. Míguez, "Nested smoothing algorithms for inference and tracking of heterogeneous multi-scale state-space systems," *Foundations of Data Science*, pp. 0–0, Feb. 2025, Publisher: Foundations of Data Science.