Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 54-20 (2021) 26–31

Fast Allan Variance (FAVAR) and Dynamic Fast Allan Variance (D-FAVAR) Algorithms for both Regularly and Irregularly Sampled Data ⋆

Satya Prasad Maddipatla * **Hossein Haeri** ** **Kshitij Jerath** *** **Sean Brennan** ****

* *The Pennsylvania State University, University Park, PA 16802, USA (e-mail: szm888@psu.edu).*
** *University of Massachusetts Lowell, Lowell, MA 01854, USA (email: hossein_haeri@uml.edu).*
*** *University of Massachusetts Lowell, Lowell, MA 01854, USA (email: kshitij_jerath@uml.edu).*
**** *The Pennsylvania State University, University Park, PA 16802, USA (e-mail: snb10@psu.edu)*

Abstract: This work develops algorithms demonstrating fast implementations of Allan variance (AVAR) for regularly and irregularly sampled signals. AVAR is a technique first developed to study the frequency stability of atomic clocks. Typical AVAR algorithms calculate changes in means between differently-sized groupings of data and thus are useful in many data aggregation processes: to select the appropriate window length or timescales for estimating a signal's moving average, to find the minimum variance of a signal, or to estimate the change in variance of a signal with complex noise contributions as a function of the number of collected data points. Unfortunately, AVAR typically involves very large signal lengths, yet the typical time required to compute AVAR increases quickly with the length of the time-series data. This paper presents a recursive algorithm inspired by the Fast Fourier Transform (FFT), specifically data organization into power-of-two groupings. This enables a fast AVAR implementation, called FAVAR, shown first for regularly sampled data. The results show a computational speed increase of three orders of magnitude versus typical AVAR calculations for data lengths often used with AVAR. Next, the FAVAR algorithm is extended to compute AVAR of irregularly sampled data by modeling these data as weighted but regularly sampled data clusters. Finally, this work analyzes Dynamic Allan variance implementations of FAVAR, called D-FAVAR, wherein AVAR is calculated at every timestep to capture window-varying statistical properties of the data stream. The recursion methods used in FAVAR, when extended to compute D-FAVAR, further increase computational speed by an additional factor of ten compared to computing the FAVAR at every timestep. They result in approximately four orders of magnitude speed improvements versus repeated calculation of AVAR with typical methods. These fast algorithms are demonstrated on signals that illustrate classical Allan variance curves, and the results agree with the classical AVAR formulations within computational accuracy.

Keywords: Allan variance, Dynamic Allan variance, Real-time algorithms.

1. INTRODUCTION

Allan variance (AVAR) is a technique first developed to study the frequency stability of atomic clocks, see Allan (1966). Typical AVAR algorithms calculate changes in means between differently-sized groupings of data and thus are useful in many data aggregation processes: to select the appropriate window length or timescales for estimating a signal's moving average, to find the minimum variance of a signal, or to estimate the change in variance of a signal with complex noise contributions as a function of the number of collected data points. The motivation of this work is to minimize AVAR computational times, particularly for signals with a large number of data points.

There are many variations in AVAR calculations. For example, AVAR for missing and irregularly sampled data is described in Sesia and Tavella (2008); Haeri et al. (2021). A detailed study on noise modeling and characterization of several noise types using AVAR was presented in Jerath et al. (2018). In some signals, the statistical properties of the noise change with time, and so Dynamic Allan variance (DAVAR) was developed; for example, DAVAR is used to

⋆ This material is based upon work supported by the National Science Foundation under grant no. CNS-1932509 "CPS: Medium: Collaborative Research: Automated Discovery of Data Validity for Safety-Critical Feedback Control in a Population of Connected Vehicles."

10.1016/j.ifacol.2021.11.148

represent the time-varying stability of a clock, see Galleani and Tavella (2003). In simple terms, DAVAR is the AVAR evaluated at every time-instant over a moving window, as in Galleani and Tavella (2009). DAVAR allows, for example, a dynamically-changing window to be calculated for data whose variance is a function of an operating condition such as in Haeri et al. (2021).

Evaluation of AVAR and DAVAR involves calculating means over different correlation intervals or window lengths. So the time to evaluate AVAR and DAVAR increase with data length. A recursive algorithm for assessing DAVAR of regularly sampled data was presented in Galleani (2010). It was observed that the recursive DAVAR is about 400 times faster than the standard computation for a data length of 10,000. There is little work on efficient computation of AVAR or DAVAR, especially for irregularly sampled data. This work fills this gap by presenting fast algorithms that compute AVAR and DAVAR of both regularly sampled and irregularly sampled data.

The work herein exploits the significant similarities between the calculation of the fast Fourier transform and Allan variance. Specifically, the results of this paper utilize the power-of-2 structure of the Fast Fourier Transform (FFT) to identify improvements in AVAR calculations. To illustrate the similarities, note that the FFT evaluates the Fourier transform using a recursive process of multiplication followed by addition, thereby considerably decreasing the computation time (Cooley and Tukey (1965); Brigham and Morrow (1967)). The results of this paper show that AVAR calculations can employ a similar recursive pattern of addition followed by division/multiplication across window lengths.

This paper is motivated by a project requiring real-time signal analysis for autonomous vehicles aggregating information in space and time via 2-dimensional AVAR. The 2-d application of AVAR is nearly identical to 1-d AVAR. For simplicity, this article focuses on 1-d implementation using example signals with a random walk and white noise properties that are pervasive in real systems. This article is organized as follows: Section 2 presents fast algorithms to calculate AVAR and DAVAR of both regularly and irregularly sampled data. In Section 3, the fast algorithms are compared with classical AVAR methods to demonstrate the significant improvements in computational speed and accuracy.

2. ALGORITHMS

Four fast algorithms to evaluate AVAR and DAVAR of both regularly and irregularly sampled data are presented in this section.

2.1 Fast AVAR of regularly sampled data

AVAR of regularly sampled data $\{y_i\}$ $(i = 1, 2, \ldots, n)$ as a function of correlation interval or window length, m, is given by equation (1) (Allan (1966)).

$$\sigma_A^2[m] = \frac{1}{2}\mathbb{E}[(\bar{y}_k - \bar{y}_{k-m})^2] \quad (1)$$

where $\bar{y}_k$ is given by equation (2).

$$\bar{y}_k = \frac{1}{m}\sum_{i=k-m}^{k-1} y_i \quad (2)$$

The expectation operator in equation (1) may be approximated, so that the AVAR can be evaluated using equation (3) (Allan (1987)).

$$\sigma_A^2[m] = \frac{1}{2(N-2m)}\sum_{k=2m+1}^{N}(\bar{y}_k - \bar{y}_{k-m})^2 \quad (3)$$

where N is the data length.

Algorithm 1, FAVAR, is a fast algorithm to evaluate the AVAR of regularly sampled data. It is implemented by utilizing $\bar{y}_k$ for a window length, m, to calculate $\bar{y}_k$ for a window length, $2m$, thereby reducing the total number of computations. It is implemented at eighth line of the FAVAR algorithm. The recursion across window lengths for a data length of seven is shown in Fig. 1.

Algorithm 1 FAVAR

1: Truncate the data $\{y_i\}$ $(i = 1, 2, \ldots, n)$ to one greater than the nearest integer power of 2 to yield data length $N = 2^p + 1$ $(p \in \mathbb{Z}^+)$.
2: Set the initial window length $m = 2$.
3: Initialize the vector $\mathbf{v}$ and its length l_v:
4: $\mathbf{v} = [y_{n-N+1}\ y_{n-N+2}\ \ldots\ y_{n-1}]^T$
5: $l_v = N - 1$
6: **while** $m \leq 2^{p-1}$ **do**
7: Update $\mathbf{v}$ and l_v:
8: $\mathbf{v} \leftarrow 0.5 \cdot ([v_1\ v_2\ \ldots\ v_{l_v - \frac{m}{2}}]^T + [v_{\frac{m}{2}+1}\ v_{\frac{m}{2}+2}\ \ldots\ v_{l_v}]^T)$
9: $l_v \leftarrow l_v - \frac{m}{2}$
10: Update the vectors $\mathbf{v}^f$ and $\mathbf{v}^b$:
11: $\mathbf{v}^f = [v_{m+1}\ v_{m+2}\ \ldots\ v_{l_v}]^T$
12: $\mathbf{v}^b = [v_1\ v_2\ \ldots\ v_{l_v - m}]^T$
13: Calculate AVAR for a window length m:
14: $\sigma_A^2[m] = \frac{1}{2(N-2m)}\sum_{j=1}^{l_v - m}(v_j^f - v_j^b)^2$
15: Update window length:
16: $m \leftarrow 2m$
17: **end while**

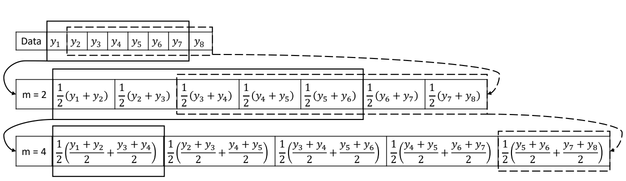

Fig. 1. FAVAR algorithm: Recursion across window lengths.

2.2 Fast AVAR of irregularly sampled data

AVAR of irregularly sampled data can be evaluated using equation (4) (Haeri et al. (2021)).

$$\sigma_A^2[\tau] = \frac{1}{2\sum_S \omega_t}\sum_S \omega_t(\bar{y}_t - \bar{y}_{t-\tau})^2 \quad (4)$$

where the summation is performed on a finite set of instances $t \in S$. $\bar{y}_t$ and ω_t are given by equation (5a) and (5b) respectively.

$$\bar{y}_t = \begin{cases} \dfrac{1}{|C_t|} \displaystyle\sum_{y_{t_i} \in C_t} y_{t_i}, & |C_t| \neq 0 \\ 0, & |C_t| = 0 \end{cases} \tag{5a}$$

$$\omega_t = |C_t|\ |C_{t-\tau}| \tag{5b}$$

where $C_t = \{y_{t_i} : t - \tau \leq t_i < t\}$ and $|C_t|$ is cardinal number of the set C_t.

It can be shown that irregularly sampled data can be modeled as weighted regularly sampled data. Let us consider that the interval $[t-\tau, t)$ contains p evenly spaced intervals and the i_{th} interval contain w_i data points. Then equation (5a) can be rewritten as equation (6) and $|C_t| = \sum_{i=1}^{p} w_i$.

$$\bar{y}_t = \frac{1}{\sum_{i=1}^{p} w_i} \sum_{i=1}^{p} \sum_{j=1}^{w_i} y_j^i = \frac{1}{\sum_{i=1}^{p} w_i} \sum_{i=1}^{p} w_i \bar{\theta}_i \tag{6}$$

where y_j^i is the j_{th} data point in the i_{th} interval and $\{\bar{\theta}_i, w_i\}$ represents processed irregularly sampled data.

And $\bar{\theta}_i$ is given by equation (7).

$$\bar{\theta}_i = \begin{cases} \dfrac{1}{w_i} \displaystyle\sum_{j=1}^{w_i} y_j^i, & w_i \neq 0 \\ 0, & w_i = 0 \end{cases} \tag{7}$$

Algorithm 2, FAVAR-I, is an extension of FAVAR to calculate AVAR of irregularly sampled data by modeling the data as regularly sampled weighted data.

2.3 *Fast DAVAR of regularly sampled data*

DAVAR for regularly sampled data can be evaluated using equation (8) (Galleani and Tavella (2009)).

$$\sigma_A^2[n, m] = \frac{1}{2(N - 2m)} \sum_{k=n-N+2m+1}^{n} (\bar{y}_k - \bar{y}_{k-m})^2 \tag{8}$$

where n is the instant at which DAVAR is evaluated, and N is the horizon length or data length for calculating AVAR.

Change in DAVAR between two successive instances is given by equation (9).

$$\sigma_A^2[n, m] = \sigma_A^2[n-1, m] + \frac{0.5}{(N - 2m)}((\bar{y}_n - \bar{y}_{n-m})^2 - (\bar{y}_{n-N+2m} - \bar{y}_{n-N+m})^2) \tag{9}$$

Algorithm 3, D-FAVAR, shows the computation of DAVAR of regularly sampled data. DAVAR is initialized using FAVAR and the recursion across window lengths is implemented using equations (10a), (10b).

$$\bar{y}_{n,2m} = \frac{1}{2}(\bar{y}_{n,m} + \bar{y}_{n-m,m}) \tag{10a}$$

$$\bar{y}_{n-N+2m,2m} = \frac{1}{2}(\bar{y}_{n-N+2m,m} + \bar{y}_{n-N+m,m}) \tag{10b}$$

where $\bar{y}_{k,m}$ is $\bar{y}_k$ for a window length m.

Algorithm 2 FAVAR-I

Truncate the processed data $\{\bar{\theta}_i, w_i\}$ $(i = 1, 2, \ldots, n)$ to one greater than the nearest integer power of 2 to yield data length $N = 2^p + 1$ $(p \in \mathbb{Z}^+)$.
Set the initial window length $m = 2$.
Initialize the vectors $\mathbf{v}$, $\mathbf{w}$ and their length l_v:
$\mathbf{v} = [\bar{\theta}_{n-N+1}\ \bar{\theta}_{n-N+2}\ \ldots\ \bar{\theta}_{n-1}]^T$
$\mathbf{w} = [w_{n-N+1}\ w_{n-N+2}\ \ldots\ w_{n-1}]^T$
$l_v = N - 1$
while $m \leq 2^{p-1}$ **do**
    Update $\mathbf{w}$, l_v, and $\mathbf{v}$:
    $\mathbf{s} = [w_1 v_1\ w_2 v_2\ \ldots\ w_{l_v - \frac{m}{2}} v_{l_v - \frac{m}{2}}]^T +$ $[w_{\frac{m}{2}+1} v_{\frac{m}{2}+1}\ w_{\frac{m}{2}+2} v_{\frac{m}{2}+2}\ \ldots\ w_{l_v} v_{l_v}]^T$
    $\mathbf{w} \leftarrow [w_1\ w_2\ \ldots\ w_{l_v - \frac{m}{2}}]^T +$ $[w_{\frac{m}{2}+1}\ w_{\frac{m}{2}+2}\ \ldots\ w_{l_v}]^T$
    $l_v \leftarrow l_v - \frac{m}{2}$
    $\mathbf{v} = [\frac{s_1}{w_1}\ \frac{s_2}{w_2}\ \ldots\ \frac{s_{l_v}}{w_{l_v}}]^T$
    Update the vectors $\mathbf{v}^f$, $\mathbf{w}^f$, $\mathbf{v}^b$, and $\mathbf{w}^b$:
    $\mathbf{v}^f = [v_{m+1}\ v_{m+2}\ \ldots\ v_{l_v}]^T$
    $\mathbf{w}^f = [w_{m+1}\ w_{m+2}\ \ldots\ w_{l_v}]^T$
    $\mathbf{v}^b = [v_1\ v_2\ \ldots\ v_{l_v - m}]^T$
    $\mathbf{w}^b = [w_1\ w_2\ \ldots\ w_{l_v - m}]^T$
    Calculate total weights and AVAR for a window length m:
    $w[m] = \sum_{j=1}^{l_v - m} w_j^f w_j^b$
    $\sigma_A^2[m] = \frac{1}{2w[m]} \sum_{j=1}^{l_v - m} w_j^f w_j^b (v_j^f - v_j^b)^2$
    Update window length:
    $m \leftarrow 2m$
end while

2.4 *Fast DAVAR of irregularly sampled data*

Algorithm 4, D-FAVAR-I, shows the evaluation of DAVAR of irregularly sampled data. DAVAR is initialized using FAVAR-I, and the recursion across window lengths is implemented similar to D-FAVAR.

3. RESULTS

A summary analysis showing the accuracy and computational speed of all the four fast algorithms is presented below. FAVAR and D-FAVAR algorithms are tested on both regularly and irregularly sampled signals each of length $2^{18} + 1$. A regularly sampled parent signal in Fig. 2 is generated, at a higher frequency of 10 times faster, by corrupting random walk signal with white noise. The first half of the signal has different noise properties compared to the second half of the signal. Random walk and white noise are generated as presented in Jerath et al. (2018). A regularly sampled signal to test FAVAR and D-FAVAR algorithms is obtained by down-sampling the parent signal at regular intervals. An irregularly sampled signal to test FAVAR-I and D-FAVAR-I algorithms is generated by down-sampling the parent signal at irregular intervals. FAVAR algorithms are tested on the random walk and white noise but are applicable for other types of noise like flicker noise, quantization noise, and rate random walk.

Algorithm 3 D-FAVAR

Initialize AVAR using FAVAR algorithm for data length equal to the horizon length N. The horizon length is of the form $2^p + 1$ ($p \in \mathbb{Z}^+$).
Truncate the data $\{y_i\}$ ($i = 1, 2, \ldots, n$) to one greater than the horizon length $N + 1$.
Set the initial window length $m = 2$.
Initialize the vectors $\mathbf{v}^o$ and $\mathbf{v}^e$:
$\mathbf{v} = 0.5 \cdot ([y_{n-N}\ y_{n-N+1}\ \ldots\ y_{n-2}]^T + [y_{n-N+1}\ y_{n-N+2}\ \ldots\ y_{n-1}]^T)$
$\mathbf{v}^o = [v_1\ v_3\ \ldots\ v_{N-2}]^T$
$\mathbf{v}^e = [v_{N-1}\ v_{N-3}\ \ldots\ v_2]^T$
while $m \leq 2^{p-1}$ **do**
  if $2 == m$ **then**
    $v^{b-} = v^o_1$
    $v^{f-} = v^o_2$
    $v^{f+} = v^e_1$
    $v^{b+} = v^e_2$
  else
    $v^{b-} \leftarrow 0.5(v^{f-} + v^{b-})$
    $v^{f-} = \frac{1}{0.5m} \sum_{j=\frac{m}{2}+1}^{m} v^o_j$
    $v^{f+} \leftarrow 0.5(v^{f+} + v^{b+})$
    $v^{b+} = \frac{1}{0.5m} \sum_{j=\frac{m}{2}+1}^{m} v^e_j$
  end if
  Update DAVAR for a window length m:
  $\sigma^2_A[n, m] = \sigma^2_A[n-1, m] + \frac{1}{2(N-2m)}((v^{f+} - v^{b+})^2 - (v^{f-} - v^{b-})^2)$
  Update window length:
  $m \leftarrow 2m$
end while

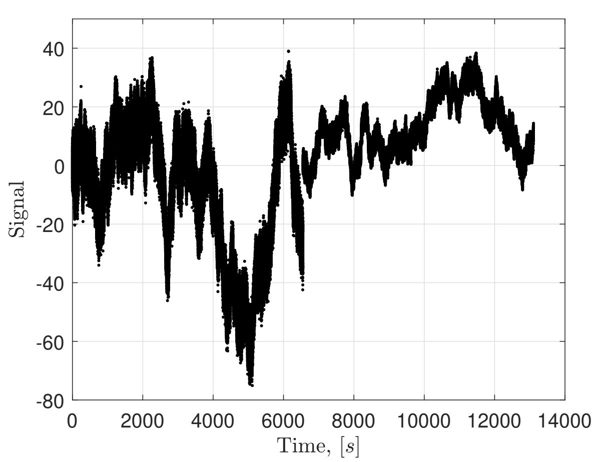

Fig. 2. Regularly sampled parent signal corrupted with white noise.

The absolute difference between AVAR of a regularly sampled signal evaluated using standard and FAVAR algorithms is within the computational accuracy as in Fig. 3(b). A similar trend in computational accuracy is observed with FAVAR-I, D-FAVAR, and D-FAVAR-I algorithms, each showing absolute errors less than 10^{-11} relative to

Algorithm 4 D-FAVAR-I

Initialize AVAR using FAVAR-I algorithm for data length equal to the horizon length N. The horizon length is of the form $2^p + 1$ ($p \in \mathbb{Z}^+$).
Truncate the processed data $\{\bar{\theta}_i, w_i\}$ ($i = 1, 2, \ldots, n$) to one greater than the horizon length $N + 1$.
Set the initial window length $m = 2$.
Initialize the vectors $\mathbf{v}^o$, $\mathbf{w}^o$, $\mathbf{v}^e$, and $\mathbf{w}^e$:
$\mathbf{w} = [w_{n-N}\ w_{n-N+1}\ \ldots\ w_{n-2}]^T + [w_{n-N+1}\ w_{n-N+2}\ \ldots\ w_{n-1}]^T$
$\mathbf{s} = [w_{n-N}\bar{\theta}_{n-N}\ w_{n-N+1}\bar{\theta}_{n-N+1}\ \ldots\ w_{n-2}\bar{\theta}_{n-2}]^T + [w_{n-N+1}\bar{\theta}_{n-N+1}\ w_{n-N+2}\bar{\theta}_{n-N+2}\ \ldots\ w_{n-1}\bar{\theta}_{n-1}]^T$
$\mathbf{v} = [\frac{s_1}{w_1}\ \frac{s_2}{w_2}\ \ldots\ \frac{s_{N-1}}{w_{N-1}}]^T$
$\mathbf{v}^o = [v_1\ v_3\ \ldots\ v_{N-2}]^T$
$\mathbf{w}^o = [w_1\ w_3\ \ldots\ w_{N-2}]^T$
$\mathbf{v}^e = [v_{N-1}\ v_{N-3}\ \ldots\ v_2]^T$
$\mathbf{w}^e = [w_{N-1}\ w_{N-3}\ \ldots\ w_2]^T$
while $m \leq 2^{p-1}$ **do**
  if $2 == m$ **then**
    $w^{b-} = w^o_1$
    $v^{b-} = v^o_1$
    $w^{f-} = w^o_2$
    $v^{f-} = v^o_2$
    $w^{f+} = w^e_1$
    $v^{f+} = v^e_1$
    $w^{b+} = w^e_2$
    $v^{b+} = v^e_2$
  else
    $s^{b-} = w^{f-}v^{f-} + w^{b-}v^{b-}$
    $w^{b-} \leftarrow w^{f-} + w^{b-}$
    $v^{b-} = \frac{s^{b-}}{w^{b-}}$
    $w^{f-} = \sum_{j=\frac{m}{2}+1}^{m} w^o_j$
    $v^{f-} = \frac{\sum_{j=\frac{m}{2}+1}^{m} w^o_j v^o_j}{w^{f-}}$
    $s^{f+} = w^{f+}v^{f+} + w^{b+}v^{b+}$
    $w^{f+} \leftarrow w^{f+} + w^{b+}$
    $v^{f+} = \frac{s^{f+}}{w^{f+}}$
    $w^{b+} = \sum_{j=\frac{m}{2}+1}^{m} w^e_j$
    $v^{b+} = \frac{\sum_{j=\frac{m}{2}+1}^{m} w^e_j v^e_j}{w^{b+}}$
  end if
  Update total weights and DAVAR for a window length m:
  $w[n, m] = w[n-1, m] + w^{f+}w^{b+} - w^{f-}w^{b-}$
  $\sigma^2_A[n, m] = \frac{1}{w[n,m]}(w[n-1, m]\sigma^2_A[n-1, m] + 0.5(w^{f+}w^{b+}(v^{f+} - v^{b+})^2 - w^{f-}w^{b-}(v^{f-} - v^{b-})^2))$
  Update window length:
  $m \leftarrow 2m$
end while

signals larger than 10^{-1}. The AVAR plots for both regularly and irregularly sampled signals are shown in Fig. 3(a) and Fig. 4 respectively.

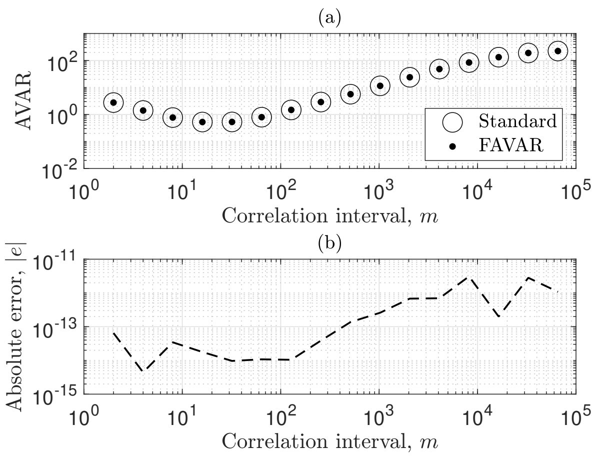

Fig. 3. (a) AVAR of regularly sampled data calculated using standard and FAVAR algorithms. (b) Absolute difference between AVAR calculated using standard and FAVAR algorithms.

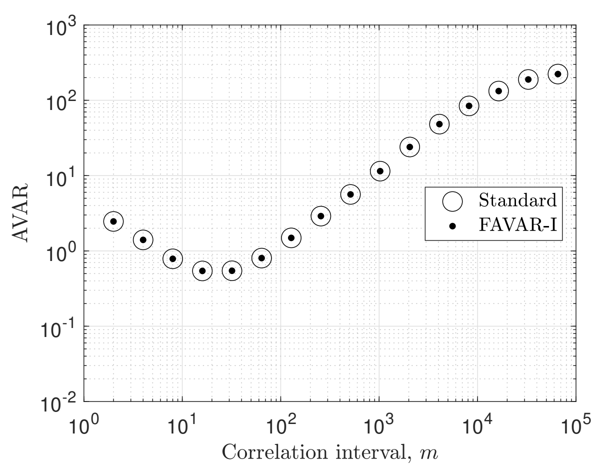

Fig. 4. AVAR of irregularly sampled data calculated using standard and FAVAR-I algorithms.

In computational speed, FAVAR is about 1000 times faster than the standard AVAR algorithm (eq. (3)) for a data length greater than 10,000, as shown in Fig. 5(a). And FAVAR-I is about 3500 times faster, as shown in Fig. 5(b). Irregularly sampled data need to be processed before using FAVAR-I. Even with the pre-processing, FAVAR-I performs markedly faster than the standard algorithm for long data lengths. However, the additional FAVAR steps and pre-processing for FAVAR-I creates overhead that penalized very short signals; it is found that the standard AVAR algorithms actually perform better than FAVAR algorithms for data with a length less than 50, especially for irregularly sampled data.

The AVAR of the first and second halves of the signal, when performed separately, reveals more information about noise properties than the AVAR of the complete signal, as shown in Fig. 6, 7. D-FAVAR is able to estimate

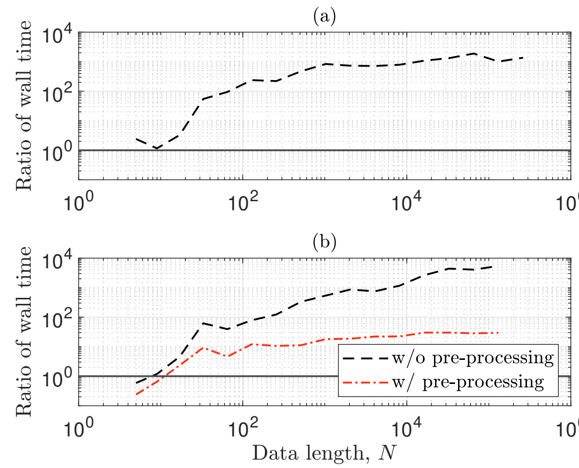

Fig. 5. Ratio of wall time is the time taken for calculating AVAR using standard algorithms divided by the time taken by FAVAR algorithms. (a) Regularly sampled data. (b) Irregularly sampled data.

this change in noise behavior; this stresses the importance of DAVAR and the need for fast algorithms to evaluate DAVAR, such as D-FAVAR and D-FAVAR-I.

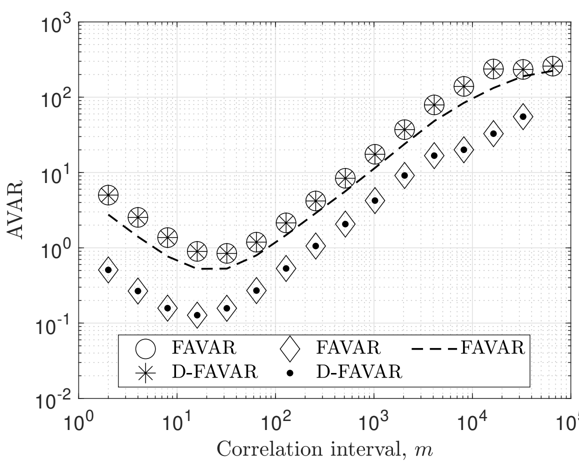

Fig. 6. The circles and asterisks represent the AVAR of the first half of the signal whereas diamonds and points represent the AVAR of the second half of the signal. The dashed line represents the AVAR of the complete signal.

Both D-FAVAR and D-FAVAR-I were found to be 10 times faster than evaluating AVAR using FAVAR/FAVAR-I at every time-instant; the results are shown in 8. FAVAR algorithms do perform better than D-FAVAR algorithms for small data lengths, for horizon lengths less than 1000. FAVAR algorithms estimate AVAR/DAVAR for both regularly and irregularly sampled data faster with the same memory used by the standard algorithms. All four fast algorithms were implemented in MATLAB R2019b on Windows 10. The computer configuration was an i7-1065G7 processor with a 1.3GHz clock and 8.00 GB RAM. MATLAB implementation of all the four algorithms presented above is available at `https://github.com/ForgetfulDatabases/FDB_AVAR_Based_Algorithms.git`.

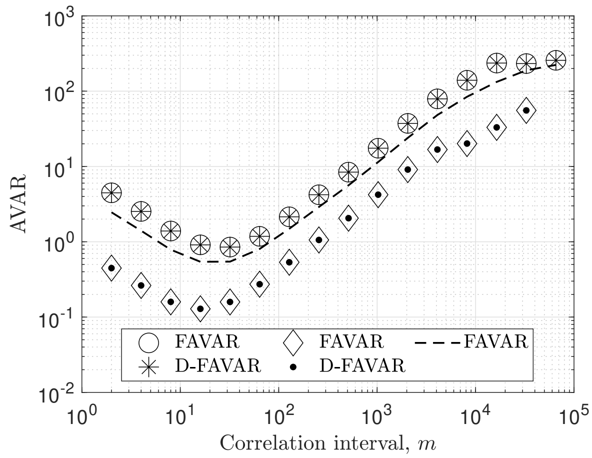

Fig. 7. The circles and asterisks represent the AVAR of the first half of the signal whereas diamonds and points represent the AVAR of the second half of the signal. The dashed line represents the AVAR of the complete signal.

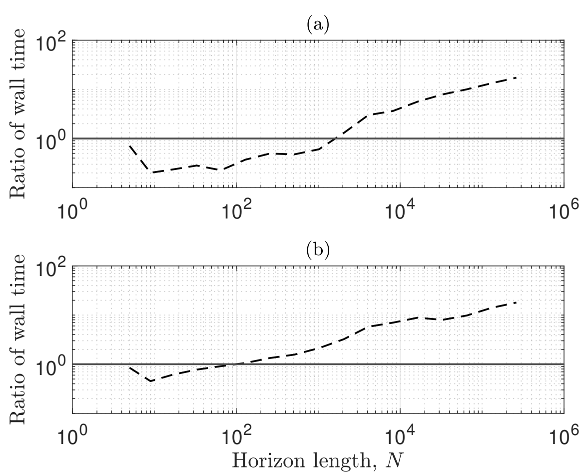

Fig. 8. Ratio of wall time is the time taken for calculating DAVAR using FAVAR algorithms divided by the time taken by D-FAVAR algorithms. (a) Regularly sampled data. (b) Irregularly sampled data.

4. CONCLUSIONS

The computational cost for evaluating AVAR and DAVAR typically increases significantly with data length. Four fast algorithms are presented in this work to evaluate AVAR and DAVAR of both regularly and irregularly sampled data. These fast algorithms are three to four orders faster than the classical formulations of AVAR/DAVAR for data lengths of about four orders or more. And the absolute errors between fast and standard algorithms are within the computational accuracy. For an illustration of the significant impacts of algorithm improvement, the standard AVAR algorithm applied to irregularly sampled data of length $2^{18} + 1$ took 2.68 hours, whereas FAVAR methods took 0.53 seconds.

REFERENCES

Allan, D.W. (1966). Statistics of atomic frequency standards. *Proceedings of the IEEE*, 54(2), 221–230. doi:10.1109/PROC.1966.4634.

Allan, D.W. (1987). Time and frequency (time-domain) characterization, estimation, and prediction of precision clocks and oscillators. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 34(6), 647–654. doi:10.1109/T-UFFC.1987.26997.

Brigham, E.O. and Morrow, R.E. (1967). The fast fourier transform. *IEEE Spectrum*, 4(12), 63–70. doi:10.1109/MSPEC.1967.5217220.

Cooley, J. and Tukey, J.W. (1965). An algorithm for the machine calculation of complex fourier series. *Mathematics of Computation*, 19, 297–301.

Galleani, L. (2010). The dynamic allan variance ii: a fast computational algorithm. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 57(1), 182–188. doi:10.1109/TUFFC.2010.1396.

Galleani, L. and Tavella, P. (2003). The characterization of clock behavior with the dynamic allan variance. In *IEEE International Frequency Control Symposium and PDA Exhibition Jointly with the 17th European Frequency and Time Forum, 2003. Proceedings of the 2003*, 239–244. doi:10.1109/FREQ.2003.1275096.

Galleani, L. and Tavella, P. (2009). The dynamic allan variance. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 56(3), 450–464. doi:10.1109/TUFFC.2009.1064.

Haeri, H., Beal, C.E., and Jerath, K. (2021). Near-optimal moving average estimation at characteristic timescales: An allan variance approach. *IEEE Control Systems Letters*, 5(5), 1531–1536. doi:10.1109/LCSYS.2020.3040111.

Jerath, K., Brennan, S., and Lagoa, C. (2018). Bridging the gap between sensor noise modeling and sensor characterization. *Measurement*, 116, 350–366. doi:https://doi.org/10.1016/j.measurement.2017.09.012.

Sesia, I. and Tavella, P. (2008). Estimating the allan variance in the presence of long periods of missing data and outliers. *Metrologia*, 45(6), S134–S142. doi:10.1088/0026-1394/45/6/s19.