DOI: https://doi.org/10.15276/aait.09.2026.32
UDC 004.62:519.246.8

Streaming and distributed nonlinear parameter estimation by least squares in expansion coefficients

Oleksii O. Pysarchuk[1)]
ORCID: https://orcid.org/0000-0001-5271-0248; platinumpa2212@gmail.com. Scopus Author ID: 7801331277
Oleksandr A. Tuhanskykh[1)]
ORCID: https://orcid.org/0009-0007-9024-2570; ringanaro@gmail.com. Scopus Author ID: 59255570200
[1)] National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", 37, Beresteiskyi Ave. Kyiv, 03056, Ukraine

ABSTRACT

Relevance. Nonlinear least squares rereads every sample at every iteration, which suits neither streams nor distributed archives. Recursive filters need preset noise covariance's; distributed frameworks merge the statistics of linear models only. **Aim and objectives.** The aim is to develop a recursive and an additive form of least squares in expansion coefficients, the equal-areas criterion and least squares on Legendre coefficients, for nonlinear estimation on streams and archives. The objectives are to establish when the fit from an additive image of the data coincides with batch least squares, to build on it a window filter compared with the extended Kalman filter, to equip the filter for gross anomalies, missing samples and parameter changes, and to apply the additive form to large archives. **Methods used.** The data projection on a block or Legendre basis, its Gram matrix and the sample sums form an additive image, to which the model is fitted by whitened least squares. The filter is a damped information-form update with a noise level from the window, residual clipping and an innovation test on every sample. **Results.** The fit from the image coincides with batch least squares when the model's sensitivities lie in the span of the basis, and merged images reproduce it to rounding error. Under white noise the window filter is level with or ahead of a static Kalman filter on most plants and behind it on a sinusoid, keeps as many runs as a well-gated Kalman filter under gross anomalies and flags parameter jumps promptly, mostly by testing every sample. Public archives were reduced once and refitted from the images on a single-board computer, a nonlinear decay time included. **Conclusions.** One additive statistic serves streams, archives and small devices. The filter is comparable in accuracy to a well-tuned Kalman filter and costs more per step, but needs no noise covariance's, resists gross errors without a gate and tests for a change on every sample.

Keywords: parameter estimation; nonlinear model; data projection; recursive filter; change detection; distributed processing

***For citation*: Pysarchuk O. O., Tuhanskykh O. A. "Streaming and distributed nonlinear parameter estimation by least squares in expansion coefficients". *Applied Aspects of Information Technology.* 2026; Vol.9 No.4: 497–511. DOI: https://doi.org/10.15276/aait.09.2026.32**

INTRODUCTION

Monitoring, navigation and control systems describe what they measure with models nonlinear in a few physical parameters: a decay rate, a natural frequency, a time constant, seasonal amplitude. Nonlinear least squares [1] fits such a model by evaluating it and its derivatives at every sample on every iteration, so the whole record must reside in the memory of one machine, which neither of the two settings now dominating practice allows.

The first setting is the stream: a controller or a sensor node updates its estimate on every sample within a fixed time and memory. The established answer is the Kalman filter [2] and, for a nonlinear model, the extended Kalman filter used as a parameter estimator [3], [4]; its accuracy depends on a process noise, a measurement noise and an initial covariance chosen in advance [5], a single gross error moves its state directly, and change detection is a separate mechanism built on top of it [6].

The second setting is the archive: station networks and telemetry stores hold tens of gigabytes split over files and machines, and frameworks such as MapReduce [7] reward computations that reduce to sums over parts of the data. Linear least squares is one, because its normal equations are sums [8]; nonlinear least squares is not, because every iteration needs every sample again. The common way out, a generic approximator in place of the structured model [9], gives up the physical parameters the model was written for.

The authors' earlier work estimates the parameters of nonlinear models by least squares in expansion coefficients: the equal-areas criterion matches the areas under the data and under the model over sub-intervals of the record [10], the subdomain method with a piecewise-constant basis, and least squares on Legendre coefficients matches their coefficients on Legendre polynomials [11]; the model structure was chosen in the same framework [12]. Those works used the criteria as batch estimators that reread the record on every iteration. Computed from the samples, the criteria need from the data only sums, which over a sliding window can serve as the measurement of a recursive filter and over the parts of an archive can be computed where they are stored and then added. This paper develops these two forms of the criteria and confirms them experimentally.

© Pysarchuk O., Tuhanskykh O., 2026

This is an open access article under the CC BY license (https://creativecommons.org/licenses/by/4.0/deed.uk)

LITERATURE REVIEW AND PROBLEM STATEMENT

Recursive estimation of parameters Appending the parameters to the state of an extended Kalman filter is the standard recursive estimator of a nonlinear model [4]; Lung showed that it may converge to a biased value or diverge depending on the noise covariance's [3], and identifying them from the innovations has been a research line since Mehra [5]. Recent work infers them by variational inference [13], re-estimates them after an innovation test [14], derives them from a smoother [15] or adds a robust update to Sage-Husa estimation [16]; each such estimator has settings of its own [17]. Moving horizon estimation [18] re-solves an optimization over a window on every step, in its robust form with bounded errors under poor excitation [19], at the price of an iterative solver in the loop. None of these estimators carries a statistic that can be merged across machines.

Gross errors and changes. Robust statistics replaces the square of a residual with a function that grows more slowly [20]; in a recursive filter the usual counterpart is a gate on an improbably large innovation [21]. Recent robust filters use a correntropy kernel [22], [23] or M-estimation weights [24], and a chi-square test of the innovation rejects outliers and re-estimates the measurement noise [25]. Change detection rests on the cumulative sum test of Page [26] and its descendants [6], with the delay of an alarm against the rate of false alarms as the governing trade-off [27], set by how often and on what statistic the test is run. All these filters judge a measurement against a prediction made before it arrived, so an estimate moved by an error misjudges the samples that follow.

Projection on a basis. Least squares on the coefficients of orthogonal polynomials goes back to Forsythe [28]; with test functions equal to the basis functions it is a Galerkin condition. Weak-form methods [29], [30] project a differential equation on test functions to avoid differentiating noisy data, and the criteria named above project the record itself. In all of them the projection serves one batch fit; its additivity over parts of the data and its use as a recursive measurement are not exploited.

Additive statistics at scale. Chu et al. showed that learning algorithms in a summation form map onto MapReduce directly [8], and mergeable synopses [31] are summaries combined without the raw data; both cover linear models and counting statistics. Randomised iterative solvers reduce the memory movement of large linear least squares problems but are hard to stop [32]. Sums accumulated in reduced precision, which a small device imposes, lose accuracy unless the summation is compensated [33].

The unsolved part of the problem follows: least squares in expansion coefficients works only in a batch, the recursive estimators rest on noise covariance's and carry nothing that could be merged, and the distributed frameworks merge the statistics of linear models only. Missing is a form of the nonlinear criterion updated on every sample at a bounded cost and memory, robust to gross errors and reporting a change of the parameters without preset noise covariance's, and accumulated over the parts of a distributed record without loss of accuracy. The problem is thus to find a form of least squares in expansion coefficients whose data statistic is additive, and to establish under which conditions, at what cost and with what accuracy it serves recursive and distributed estimation.

RESEARCH AIM AND OBJECTIVES

The aim of the work is to develop a recursive and an additive form of least squares in expansion coefficients, the equal-areas criterion and least squares on Legendre coefficients, so that the parameters of a nonlinear model are estimated on a data stream at a bounded cost and memory per sample and over a distributed archive without rereading its samples.

The objectives are:

1) to write the criteria through an additive image of the data on a basis and to establish the conditions under which the fit from the image coincides with batch nonlinear least squares;

2) to build on the image of a sliding window a recursive filter with a bounded cost and memory per step whose noise level is estimated from the data, and to compare it with the extended Kalman filter;

3) to equip the filter for gross anomalies, missing samples and a change of the parameters, and to confirm each element;

4) to apply the additive form to the distributed processing of large archives and to confirm its exactness, memory and throughput on synthetic records, on two public archives and on a single-board computer.

MATERIALS AND METHODS

The method follows the order of the objectives (Fig. 1), and the experimental setting closes the section.

The fit from an additive image of the data

Let a record be samples of a model with an additive error,

$$y_i = f(t_i;\theta) + \varepsilon_i, \quad i = 1, \dots, n, \qquad (1)$$

where θ is a vector of m parameters. The earlier works compared the record with the model through its coefficients on basis functions $\varphi_0, \dots, \varphi_k$, stated as integrals over the span T of the record,

$$c_j(y) = \int_T \varphi_j(t)\, y(t)\, dt, \quad j = 0, \dots, k. \qquad (2)$$

With the indicator functions of $k+1$ equal subintervals, a block basis, the c_j are areas under the record, and equating them to the areas under the model is the equal-areas criterion, the subdomain method of weighted residuals; with Legendre polynomials mapped to T, the least squares distance between the coefficients of the record and of the model is least squares on Legendre coefficients. The batch form computed the coefficients of the record by a quadrature over the samples and those of the model separately, which leaves a quadrature error in the criterion and needs a complete record.

No integral is needed if the coefficients are replaced by sums over the samples. The projection of the record on the basis and the Gram matrix of the basis on the sampling grid are

$$S = \Phi^T y, \quad G = \Phi^T \Phi, \quad \Phi_{ij} = \varphi_j(t_i), \qquad (3)$$

and together with the number of samples n, the sum of squares $y^T y$ and a descriptor of the grid they form the image of the record, $(S, G, n, y^T y, \text{grid})$. S is the discrete counterpart of (2), and G records how the basis functions overlap on the actual grid. Every part of the image is a sum over samples, so for any split of the record into parts c that share the basis and its span

$$S = \sum_c S_c, \quad G = \sum_c G_c. \qquad (4)$$

The image of a record is the sum of the images of its parts, in any order and on any machine. The model is compared with the image through the same sums:

$$\hat{\theta} = \arg\min_{\theta} e^T e,$$
$$e(\theta) = L^{-1}(S - \Phi^T f(t;\theta)), \quad G = LL^T. \qquad (5)$$

Three notions are kept apart. The projection is S. The image is an additive statistic holding everything criterion (5) and its residual variance need, whatever the model. It is a sufficient statistic in the statistical sense only for a model linear in its parameters whose repressors lie in the span of the basis, under Gaussian noise; for a nonlinear model it is sufficient for (5) and not for the likelihood of the samples.

Because $\Phi^T f$ is evaluated on the grid of the record, both sides of (5) carry the same discretisation and no quadrature error enters. What (5) leaves out follows from the projector $\Pi = \Phi G^{-1} \Phi^T$ on the span of the basis: with $r = y - f(t;\theta)$ criterion (5) equals $\|\Pi r\|^2$, whereas batch least squares minimises $\|r\|^2 = \|\Pi r\|^2 + \|(I-\Pi)r\|^2$, and with $J = \partial f / \partial \theta$ the two gradients differ by $2((I-\Pi)J)^T r$. If the columns of J lie in the span of the basis over the region the iteration searches, then $(I-\Pi)J = 0$, the discarded term does not depend on θ and the minima coincide exactly, as for a model linear in its parameters whose regressors the basis contains. For a sinusoid or an exponential a linearisation at the batch estimate θ_b, where $J^T r_b = 0$, gives the shift $\hat{\theta} - \theta_b \approx -(H^T H)^{-1}((I-\Pi)J)^T r_b$ with $H = L^{-1}\Phi^T J$, which falls with the share of the Jacobian outside the span as the order grows; the experiments measure that share and the difference of the estimates in standard errors.

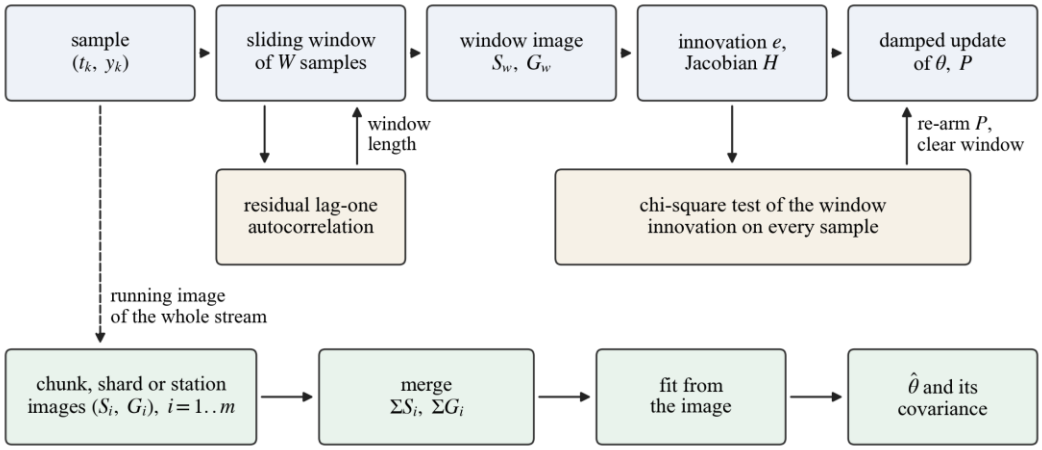


Fig. 1. **The recursive form of the criteria, a filter on the image of a sliding window, and the additive form, the reduction of an archive, both built on the image of the data**

***Source*: compiled by the authors**

The measured values enter the Gauss-Newton iteration only through S and $y^T y$. The model and its Jacobian are still evaluated at every grid position on every iteration, of the order of $nm(k+1)$ operations as on the samples; a uniform grid is stored as its ends and count, an irregular one as its positions. The image saves the pass over the measured values and their storage, not the pass over the grid.

The covariance of the estimate is $s^2(H^T H)^{-1}$, with $s^2 = \mathrm{rss}/(n-m)$, the residual sum of squares being recovered from the image as $\mathrm{rss} = y^T y - S^T \beta + d^T G^{-1} d$ with $\beta = G^{-1} S$ and $d = S - \Phi^T f$, exactly when the model lies in the span. G is nonsingular when the grid keeps the basis functions linearly independent, and $H^T H$ when the projected model identifies its parameters; neither is assumed. G is inverted through its pseudo-inverse and $H^T H$ through the singular value decomposition of H, a singular value below the largest one times the larger dimension of H times the machine precision counts as zero, and a parameter along such a direction is reported as not identified, with an infinite variance. In this form the block basis gives the equal-areas criterion and the Legendre basis least squares on Legendre coefficients, one estimator with two bases.

The window filter on the image of a sliding window

A recursive estimator has to update on every sample at a cost and memory independent of the length of the stream. Its measurement is the image of the last W samples on the window's own span, S_w and $G_w = L_w L_w^T$; the state is θ with the matrix P, and comparing the measurement with the image of the model on the window gives the whitened innovation $e_w = L_w^{-1}(S_w - \Phi_w^T f(t_w;\theta))$ and the whitened Jacobian H_w. One step is an information-form update

$$P^+ = (P^{-1} + H_w^T H_w / s^2)^{-1},$$
$$\theta^+ = \theta + \gamma P^+ H_w^T e_w / s^2, \quad P = P^+ + Q. \qquad (6)$$

Successive windows share no span, so (4) does not apply between them: S_w and G_w are recomputed from the buffer, and additivity serves where the span is fixed, in an archive and in a running image of a stream whose span is set in advance.

Two decisions in (6) replace settings that a Kalman filter asks for. A linearised step on a nonlinear model may overshoot, so γ starts at one and is halved until the whitened misfit $e_w^T e_w$ does not rise; at most nine lengths, down to $1/256$, are tried, and when none lowers the misfit the step is rejected with θ and P unchanged. The variance s^2 is an exponentially weighted average, with weight 0.05, of the mean square of the window residuals, without a correction for the degrees of freedom. What remains to be chosen is Q: close to zero the filter treats the parameters as constants, larger it tracks them.

Neighbouring windows share all but one sample, so adding $H_w^T H_w / s^2$ to P^{-1} on every step counts each sample up to W times: (6) is a damped Gauss-Newton step with a gain state P, not the Kalman update of a calibrated covariance. The covariance of the estimate comes from a separate fit of the current window by (5), which counts every sample once; with clipping on, it is that of the least squares fit to the clipped window, with s^2 from the same window. The experiments check both against the actual errors.

The cost of a step is fixed by W, k and m: imaging the window takes of the order of $W(k+1)^2$ operations and its Cholesky factor $(k+1)^3$, the model is evaluated on the window at most ten times and its Jacobian once, and the update inverts a matrix of order m; the state is the buffer, θ and P. The operations of a step are thus bounded independently of the length of the stream; its time is not guaranteed.

Gross anomalies, missing samples and a change of the parameters

Gross errors. Before the window is imaged, residuals of the current fit beyond 3×1.4826 median absolute deviations are clipped to that bound, Huber's estimator [20] applied to the window, so a residual is judged against the fit of the samples around it and not against a prediction made before they arrived. The clipping is recomputed on every step.

Missing samples need no element of their own: a window is imaged on the time stamps it received, so a gap is a thinner window and not a missed update.

A change of the parameters asks for a shorter memory and for an alarm. The lag-one autocorrelation of the window residuals is averaged exponentially with weight 0.1, and while it stays above 0.35, a sign that the model no longer describes the window, the window shrinks by one sample per step and otherwise grows back to W. The alarm tests the innovation with the statistic

$$\nu = e_w^T e_w / s^2 - b^T P^+ b, \quad b = H_w^T e_w / s^2. \qquad (7)$$

Table 1. **Test plants**

Plant	Model	True parameters	Starting guess	Samples, span	Windo w	Tracking Q
Damped oscillator	$Ae^{-\zeta\omega t}\sin(\omega\sqrt{1-\zeta^2}\,t)$	2.0; 2.5; 0.12	2.0; 2.0; 0.1	700, 12 s	60	$10^{-3}I$
Sinusoid with offset	$c + A\sin\omega t$	0.5; 2.0; 2.5	0.5; 1.5; 2.0	700, 12 s	60	$10^{-3}I$
First-order rise	$K(1 - e^{-t/\tau})$	3.0; 1.2	1.0; 1.0	600, 6 s	50	$10^{-2}I$
Accelerating trajectory	$c_0 + c_1 t + c_2 t^2$	1.0; 2.0; 0.5	0; 0; 0	600, 6 s	40	$10^{-2}I$

***Source*: compiled by the authors**

By the matrix inversion lemma (7) equals $e_w^T(s^2 I + H_w P H_w^T)^{-1} e_w$, the innovation normalised by its predicted covariance, and it follows a chi-square distribution with $k + 1$ degrees of freedom when the model is linear in θ over the error of the estimate, the noise is white and Gaussian with the variance s^2, and the error of the prior estimate is Gaussian with the covariance P and independent of the noise in the window. The filter meets the last two conditions only approximately: s^2 is estimated from the window, P is a gain state, the prior estimate rests on samples that overlap the window, and clipping truncates the residuals. The chi-square quantile is therefore a reference level and not the false-alarm rate, and the distribution of ν without a change is measured on long records. A flag is raised when ν exceeds the quantile of level $1 - \alpha$, $\alpha = 10^{-4}$, after a warm-up of three windows and with a cool-down of one, the test running on every sample; the filter is then re-armed by inflating P four times, by also collapsing the window to its minimum, or by clearing the window. The baseline is the detector such filters have carried so far [26], [6], consulted once per window after three windows of warm-up: a jump test of the innovation energy at level 10^{-3} and a two-sided cumulative sum of the window-mean innovation with a slack of 0.5 and a threshold of 5 standard deviations, both against baselines averaged with weight 0.15.

Distributed processing of large archives

For an archive the map step images each chunk, file or station on the common span and the reduce step adds the images by (4); the fit (5) runs once, on the merged image, and what is stored for a series is its image. Accumulation is in double precision; where single precision is imposed, the sum across chunks is compensated by Kahan's algorithm [33]. A running image reduces a stream the same way, in memory that does not depend on the length of the stream.

A series is imaged on a Legendre basis of order $max(16, \lceil 8T \rceil + 16)$ for a span of T years, capped at $n - 2$ and at $n/4$. Its fit from the stored image is accepted when the parameters agree with the raw fit to a relative 10^{-8}. A miss is put down to the image, reported apart and not counted as agreement, when the relative truncation error of the model's sensitivities at the image order exceeds 0.02 (undersampled) or when the miss stays within the least squares perturbation bound $\kappa_X^2(c + \epsilon\kappa_G)$ (ill-conditioned), with κ_X and κ_G the condition numbers of the model's design and of G, c the truncation error and ϵ the machine precision; any other miss is a failure.

Experimental setting

Accuracy is measured on four synthetic plants (Table 1) with 32 noise seeds; for a given seed every estimator receives the same noise realization, so the comparisons are paired. The noise is white Gaussian, five percent of the standard deviation of the clean signal unless stated otherwise, with a first-order autoregression of coefficient 0.5 or 0.8 for correlated noise; anomalies are spikes of eight signal standard deviations on a given share of samples, and dropout removes a given share. The score is the mean relative error of the final estimate, in percent,

$$\delta = \frac{100}{m} \sum_{j=1}^{m} \frac{|\hat{\theta}_j - \theta_j|}{|\theta_j|}, \qquad (8)$$

reported as the mean, standard deviation and median over the seeds; two estimators are compared by the ratio of their mean scores with a 95 % bootstrap interval over 10 000 resample's of the seed pairs.

The window filter runs with a Legendre basis of order five or as many blocks as parameters, matched to constant parameters ($Q = 10^{-8}I$, fixed window) or tracking (Q from Table 1, adaptive window). Every constant other than the window and the tracking Q is common: $P = 10I$ at the start, the weights and thresholds given above, nine step lengths and clipping at three scaled median absolute deviations. The filter thus asks for no noise covariance, while the window, the basis and the process noise remain settings. A whole record is fitted from its image at Legendre order 24.

The window follows from the mode. With constant parameters it is as long as the record

allows: over windows of 20 to 180 samples a fixed window of 180, the row of Table 2, is within 9 % of the best on every plant, while the windows of Table 1 cost up to 1.75 times the best, so the static rows of Table 2 below understate the filter. With moving parameters the window is chosen from the settling time allowed after a change: on the amplitude jump at its middle phase every run is flagged, on average within 0.3 s at any window, but the estimate settles in 0.80, 3.17 and 6.09 s at $Q = 10^{-2}I$ with windows of 30, 60 and 120, and a process noise raised from $10^{-4}I$ to $10^{-2}I$ shortens it by at most 1.29 s at a cost in accuracy on constant parameters.

The baseline is an extended Kalman filter with the parameters as its state over a grid of twelve settings: an initial covariance p_0 of 5 or 0.1, a measurement variance r of 0.5 or the true one, and a process noise q of 10^{-4}, 10^{-6} or 10^{-8}; the tracking setting ($p_0 = 5$, $r = 0.5$, $q = 10^{-4}$) and the static one ($q = 10^{-8}$) are the references. Under anomalies it also runs from $p_0 = 0.1$ in the static setting with a gate at three predicted standard deviations [21], with a re-acquisition that lifts the gate for one sample with an inflated covariance after ten consecutive rejections, and with a Huber weight of the innovation with the knee at 1.345, each with $r = 0.5$ or the true variance. The floor is batch nonlinear least squares [1] by SciPy's curve_fit, with tolerances of $1.49 \cdot 10^{-8}$ on the step and the cost, none on the gradient and at most 20 000 evaluations, against 10^{-8} on all three for the fit from the image; a refit of a sliding window by the same solver is the naive alternative to a recursive filter.

Change detection uses a sinusoid whose amplitude jumps from 2.0 to 3.5 at twelve phases of a window for each of eight seeds, and false alarms are counted over the distinct samples screened before the jump. The baseline detector and test (7) each run once per window, at their own thresholds, and on every sample, where both are set to the same false-alarm rate on records without a change. False alarms per unit time come from records without a change: eight of 10 000 samples for the sinusoid with and without offset, the trajectory and a straight line, and 64 of the plant's length for the damped oscillator and the rise.

The reduction is measured on synthetic records of up to a million samples and on two public archives, the daily station positions of the Nevada Geodetic Laboratory [34] and one year of the hourly Integrated Surface Database [35], each series fitted with a six-parameter model of level, trend and annual and semiannual harmonics [36]. This model is linear, so for stations near earthquakes of magnitude 7.0 and above with at least two years of post-event records a post-seismic term $A(1 - e^{-(t-t_e)/\tau})$ is added, its decay time τ fitted from the stored image of the post-event segment and from its raw samples.

The workstation has an AMD Ryzen 9 9950X3D processor (4.3 GHz base, 5.7 GHz boost, 32 logical cores) and 54.9 GiB of memory under WSL2, with Python 3.14.2, NumPy 2.4.1, SciPy 1.17.0 and OpenBLAS 0.3.30; the edge device is a Raspberry Pi 5 with four Cortex-A76 cores at up to 2.4 GHz and 7.7 GiB, with Python 3.12.14, NumPy 2.5.3, SciPy 1.18.1 and OpenBLAS 0.3.34. Linear algebra runs on one thread, and the archive runs use 16 worker processes on the workstation and four on the Pi. A filter step is timed around the update alone, as the mean, median and slowest step over five repeats; the reduction times include reading the files and writing the images. The Pi's load average of 2.08 at the start of its refit is its idle level; no other process ran. The microcontroller, an nRF52840 with a Cortex-M4F core, runs in single precision only the fixed-window update (6) for a two-parameter model linear in its parameters, without clipping, the adaptive window or the test (7).

RESEARCH RESULTS

The fit from an additive image of the data

Table 3 compares the two fits run by run, by the largest difference of the estimates in standard errors of the batch fit. The values are the largest over 32 seeds; the smallest order in the last row is taken over eight seeds.

The trajectory is the exact cause of the condition stated with (5): a quadratic lies in the span from order two, and the fits coincide at every order and span. The oscillating plants are the approximate case: at order 24 their estimates differ by at most $5.5 \cdot 10^{-4}$ of a standard error in any of the 128 runs, while at order five they move by hundreds of standard errors, and the order needed grows with the span. Equal areas on as many blocks as parameters is an exactly determined matching, not least squares, and differs from the batch fit by up to 306 standard errors. On the one-second filter window of the oscillating plants order five leaves $1.3 \cdot 10^{-4}$ of the Jacobian outside the span. Additivity holds in floating point: eight chunk images of a record of a million samples, merged in either order, reproduce the parameters of the whole record to a relative $2.1 \cdot 10^{-15}$, and without one chunk the parameters move by 0.009 and 0.018 of their standard errors, a loss of data and not an error of the method.

Table 2. **Final parameter error in percent**

Estimator	Damped oscillator	Sinusoid	First-order rise	Trajectory
Window filter, Legendre basis	0.59 (0.37) / 0.50	0.34 (0.19) / 0.33	0.29 (0.44) / 0.18	2.69 (1.65) / 2.76
Window filter, block basis	0.62 (0.46) / 0.49	0.34 (0.18) / 0.33	0.45 (0.53) / 0.24	2.69 (1.64) / 2.80
Window filter, window of 180 samples	0.34 (0.25) / 0.27	0.32 (0.18) / 0.30	0.21 (0.15) / 0.17	2.57 (1.68) / 2.11
Window filter, tracking tuning	7.00 (7.89) / 3.84	1.69 (1.26) / 1.54	16.2 (9.60) / 16.5	24.3 (17.5) / 20.4
Kalman filter, tracking setting	1.00 (0.45) / 0.89	0.50 (0.29) / 0.39	0.67 (0.26) / 0.60	4.67 (3.50) / 3.26
Kalman filter, static setting	0.54 (0.19) / 0.53	0.20 (0.14) / 0.16	1.31 (0.28) / 1.27	2.32 (1.76) / 1.59
Kalman filter, best of twelve settings	0.54 (0.19) / 0.53	0.20 (0.14) / 0.17	0.35 (0.21) / 0.35	2.17 (1.54) / 1.70
Sliding-window refit	48.4 (14.0) / 56.1	1.95 (1.43) / 1.53	38.8 (13.5) / 44.1	417 (49) / 432
Batch least squares	0.17 (0.10) / 0.14	0.20 (0.14) / 0.15	0.17 (0.14) / 0.14	2.34 (1.78) / 1.53
Ratio to the static Kalman setting	1.10 [0.89, 1.33]	1.68 [1.26, 2.25]	0.22 [0.14, 0.36]	1.16 [0.97, 1.42]
Ratio to the best Kalman setting	1.10 [0.89, 1.33]	1.64 [1.24, 2.19]	0.85 [0.50, 1.42]	1.24 [1.05, 1.50]
Ratio to best Kalman, window of 180	0.63 [0.50, 0.76]	1.55 [1.21, 2.01]	0.61 [0.47, 0.77]	1.19 [1.03, 1.38]
Ratio to batch least squares	3.52 [2.92, 4.27]	1.70 [1.27, 2.28]	1.70 [1.10, 2.63]	1.15 [0.96, 1.42]

***Source*: compiled by the authors**

Table 3. **Fit from the image against batch least squares**

Measure	Damped oscillator	Sinusoid	First-order rise	Trajectory
Difference, standard errors, Legendre order 24	$5.5 \cdot 10^{-4}$	$3.4 \cdot 10^{-4}$	$1.0 \cdot 10^{-5}$	$6.5 \cdot 10^{-7}$
The same, Legendre order 5	758	2084	0.040	$6.5 \cdot 10^{-7}$
The same, as many blocks as parameters	306	20.8	1.08	2.19
Relative excess of the batch criterion, order 24	$5.3 \cdot 10^{-10}$	$2.0 \cdot 10^{-10}$	$4.4 \cdot 10^{-16}$	$4.4 \cdot 10^{-16}$
Jacobian outside the span, window, order 5, median	$1.3 \cdot 10^{-4}$	$1.3 \cdot 10^{-4}$	$1.5 \cdot 10^{-9}$	$7.6 \cdot 10^{-11}$
Smallest order for 0.1 standard error, half / whole / double span	10 / 24 / over 24	10 / 24 / over 24	3 / 5 / 6	2 / 2 / 2

***Source*: compiled by the authors**

The window filter against the extended Kalman filter

Table 2 gives the final error over the 32 paired seeds and the ratios the comparison rests on. Each cell holds the mean error in percent with the standard deviation in parentheses and, after the slash, the median; a ratio is that of the mean error of the window filter on the Legendre basis to the mean error of another estimator, with the 95 % bootstrap interval over the paired seeds.

The Legendre and block forms of the filter are level, so the two criteria give one filter. Against the static Kalman setting the paired intervals give parity on the damped oscillator and the trajectory, a lead on the first-order rise, where the filter's error is 0.22 of the Kalman filter's, and a deficit of 1.7 times on the sinusoid; the best of the twelve Kalman settings, chosen after the fact, closes the gap on the rise and is 1.24 times ahead on the trajectory.

Neither recursion reaches the batch floor on a plant with a transient: passing over each sample once, the filter is 3.5 times behind it on the damped oscillator, in all 32 seeds, and 1.7 times on the sinusoid and the rise. The medians agree except on the trajectory, where the filter's median is 1.74 times the static Kalman filters, so parity there holds for the means only. In its tracking tuning the filter is 3.4 to 24 times behind the Kalman tracking setting on these constant parameters. The sliding-window refit, which carries nothing over, is the worst throughout.

How the Kalman filter is set decides most of this comparison (Fig. 2a): over twelve settings its error spans a factor of 465 on the damped oscillator and 12 to 233 on the other plants, and the most even setting is within 2.3 times the best on every plant. The noise sweep (Fig. 2b) bounds parity with the static setting: the filter is ahead at two percent noise, level at five and behind by 1.5 and 1.2 times at ten and twenty percent.

A filter step costs eight to nine times a Kalman step (Table 4), little of it in the step search, since 99.8 percent of 154 607 steps took the first length, the most was five and none was rejected; the step

grows by a third from $W = 20$ to 240, not measurably with the order, by a fifth from two to five parameters and by half with a tenth of the samples missing. The refit is comparable only in the mean, it's median being that of the steps between refits, one step in twenty, while a refit itself reaches 15 ms because its iterations are not fixed. In Table 4, measured on the Python reference implementation, the ranges of the first three rows are over the four plants, the rows in the window and the order are for the window filter on the damped oscillator, and the row in the number of parameters is on a polynomial model. What the filter keeps is small: the state of a three-parameter filter takes 212 bytes in single precision with a window of 15 and 492 bytes with 50, a step allocates 5.3 and 7.6 kB beside the 210 MB of the interpreter process, and on the microcontroller the fixed-window update of a two-parameter linear model holds 152 bytes against 184 predicted and takes 104 us.

Table 5 gives the calibration of the window filter under the static tuning over 32 seeds: the median of $e^T C^{-1} e$ for an estimate error e and covariance C, then the shares of the errors within one and within two standard errors. The estimate is scored every W samples from the third window on and at the last sample, 320 points (416 on the trajectory); 16 window fits on the rise are excluded, having no finite covariance.

The gain state P is overconfident on every plant: its median normalised squared error $e^T P^{-1} e$ is 11 to 20 times the chi-square median and 0.20 to 0.28 of the errors fall within one of its standard errors, as a matrix that counts each sample up to W times must; it sets the gain and is not reported. The covariance of the window fit is close to the reference on the sinusoid, the rise and the trajectory but not on the damped oscillator, where its median of 38.4 exceeds that of P and 0.90 of the errors fall within two standard errors.

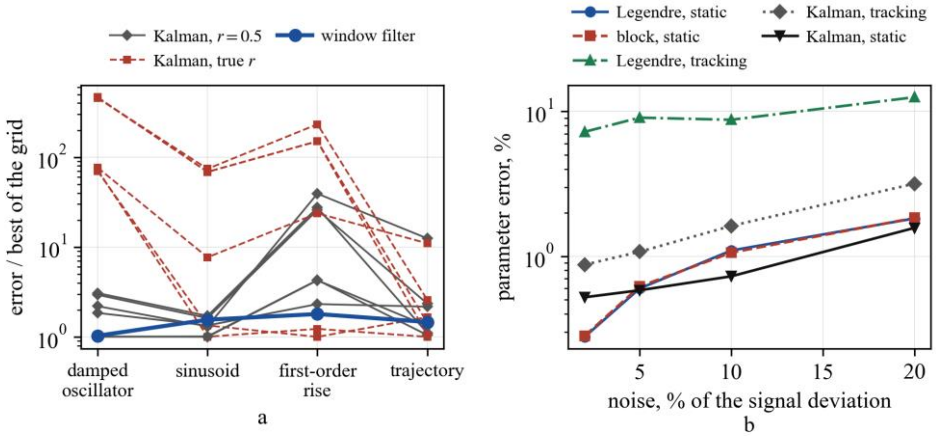


Fig. 2. **Error of the extended Kalman filter over its settings and of the window filter, means over eight seeds:**
a – **error at each of twelve Kalman settings relative to the best setting for the plant;**
b – final parameter error on the damped oscillator against the noise level
***Source*: compiled by the authors**

Table 4. **Cost of one step in microseconds**

Estimator	Mean	Median	Slowest
Window filter	82 to 116	74 to 99	225 to 345
Kalman filter	9.8 to 13.1	9.0 to 12.3	51 to 82
Sliding-window refit	29 to 121	0.19 to 0.20	801 to 15 212
W 20; 60; 240	101; 107; 133	90; 94; 114	312; 544; 561
Order 3; 9	107; 107	94; 94	487; 605
m 2; 3; 5	76; 79; 91	67; 71; 81	380; 312; 367
10 % missing	164	133	552

***Source*: compiled by the authors**

Table 5. **Calibration of the window filter**

Plant	Gain state P	Window fit	Calibrated reference
Damped oscillator	25.9; 0.28; 0.49	38.4; 0.69; 0.90	2.37; 0.68; 0.95
Sinusoid with offset	36.2; 0.22; 0.40	2.18; 0.69; 0.95	2.37; 0.68; 0.95
First-order rise	19.5; 0.25; 0.42	1.43; 0.75; 0.96	1.39; 0.68; 0.95
Trajectory	46.6; 0.20; 0.37	2.32; 0.65; 0.94	2.37; 0.68; 0.95

***Source*: compiled by the authors**

Gross anomalies, missing samples and a change of the parameters

Table 6 sets the clipped filter against the defences a Kalman filter has. Each cell gives the median error in percent over 32 seeds and, after the semicolon, the number of the 32 runs ending below five percent.

Without a defence the Kalman filter fails at the first spikes. Gated with the fixed variance $r = 0.5$, it keeps as many runs as the clipped filter at three to five times its median error, a variance far above the true one damping every update. Given the true variance the gated filter is level with the window filter; re-acquisition recovers one run at most, the Huber weight does worse, and the gated tracking setting keeps only 14, 12 and 3 runs. The window filter needs neither a gate nor the true variance, because a residual is judged against the fit of the samples around it; on clean data clipping changes its error by 0.004 points. Missing samples, up to a fifth, cost no estimator more than two runs, and the window filter, at medians of 0.43 to 0.60 percent, stays level with the static setting's 0.53 to 0.58.

Table 7 separates the test from how often it is consulted. Each entry gives the value with a fixed and, after the slash, an adaptive window over 96 runs (twelve phases, eight noise realizations); false alarms are counted in 1400 distinct pre-jump samples and settling is to within five percent of the new value. On every sample, with both detectors set to the same false-alarm rate on records without a change, two alarms in 80 000 samples, the baseline detector flags 85 of 96 jumps 0.08 s after them and test (7) all 96 within 0.4 s; once per window, at their own thresholds, they flag 76 and 80 at 3.27 and 3.05 s. The polling rate thus accounts for nearly all of the delay (Fig. 3a), and the test adds the jumps the baseline misses. No per-sample arrangement raised a false alarm among the 1400 distinct screened samples, which bounds the rate below 21.4 per 10 000 at 95 percent; Table 8 measures it.

The re-arming decides the return to the new value (Fig. 3b): inflating *P* leaves the samples from before the jump in the window, so the fixed window needs 5.26 s and the adaptive one 2.37 s, while clearing it brings both to 0.78 s.

Without a change, statistic (7) stays below its chi-square reference in mean and quintiles, with a lag-one autocorrelation above 0.7 from the overlapping windows (Table 8), so α bounds the false-alarm rate, 0 to 0.25 per 10 000 samples, for white noise only: under a first-order autoregression with coefficient 0.5 and 0.8 the rate rises to 36 and 124, and the error of the filter on the damped oscillator grows to 1.74 and 2.43 times that of the static Kalman filter. The records of Table 8 are white noise unless an autoregression coefficient is given.

Table 8. **Statistic (7) on records without a change**

Record	Mean of (7); lag-one correlation	99.99 % quantile of (7)	False alarms per 10^4 samples
Damped oscillator	3.99; 0.74	22.5	0.22
Sinusoid with offset	3.29; 0.74	21.3	0
First-order rise	4.35; 0.71	22.3	0
Trajectory	5.20; 0.75	23.9	0.13
Sinusoid	4.22; 0.78	25.7	0.25
Sinusoid, coef. 0.5	11.0; –	–	35.9
Sinusoid, coef. 0.8	25.6; –	–	124.4
Reference	6; 0	27.9	1

***Source*: compiled by the authors**

Distributed processing of large archives

On synthetic records double-precision accumulation stays below $2 \cdot 10^{-15}$ for any number of chunks, while naive single precision reaches $1.3 \cdot 10^{-6}$ at 1024 chunks and a compensated sum gains a factor of 24 (Fig. 4a); streaming holds memory flat, 80 MiB at a million samples of 32 channels against 336 MiB resident (Fig. 4b).

A file is attempted when it is read and reduced when its image is written (Table 9); those that failed on input had too few epochs or valid temperature rows for the model or, for 324 hourly files, a read block with none. Every reduced series was fitted; the 2817 and 966 not accepted are those the gate flags. One worker reduced 76 000 daily epochs of three components a second, or 600 000 on twelve workers, at a resident peak of 670 MiB, 50 MiB of it the method's own, flat over a tenfold number of files.

The sampling grid decides the saving: the hourly archive, on a uniform grid, shrinks 180 times, the irregular daily positions 1.18 times, since the image keeps an irregular grid and its order grows with the span: the median ratio falls from 2.4 at one to three years to 1.0 at twenty to thirty-five (Fig. 4c).

The Raspberry Pi 5, given only the images, refitted all 12 677 station-years on four workers (Table 9); every accepted fit agrees with the workstation to $7.3 \cdot 10^{-8}$, and the 233 beyond 10^{-6} are all flagged by the gate. Of the 1447 station components whose decay time the raw fit puts two standard errors from zero, 1441 pass the gate, and on these τ, A and the offset agree to within a hundredth of their standard errors.

Table 6. **Damped oscillator under gross anomalies and missing samples**

Estimator	No anomalies	5 %	10 %	20 %
Window filter, clipped	0.50; 32	0.35; 32	0.43; 31	0.55; 29
Kalman filter, static setting, no gate	0.53; 32	348; 5	395; 1	494; 0
Kalman filter, gated, $r = 0.5$	1.70; 32	1.80; 32	1.87; 30	2.85; 28
Kalman filter, gated, true variance	0.44; 31	0.45; 31	0.46; 30	0.48; 30
The same with re-acquisition	0.44; 32	0.45; 32	0.46; 31	0.48; 30
Kalman filter, Huber weight, true variance	0.43; 31	0.58; 23	1.17; 19	1.79; 19
Kalman filter, gated, tracking setting	0.89; 32	7.03; 14	11.5; 12	316; 3
Missing samples				
Window filter, clipped	–	0.43; 32	0.56; 32	0.60; 32
Kalman filter, static setting, no gate	–	0.53; 32	0.56; 32	0.58; 32
Kalman filter, gated, $r = 0.5$	–	1.76; 32	1.76; 32	1.78; 32
Kalman filter, gated, true variance	–	0.46; 31	0.46; 30	0.53; 31
The same with re-acquisition	–	0.46; 31	0.46; 31	0.53; 31
Kalman filter, Huber weight, true variance	–	0.44; 31	0.41; 31	0.46; 31
Kalman filter, gated, tracking setting	–	0.87; 32	0.79; 32	1.06; 32

***Source*: compiled by the authors**

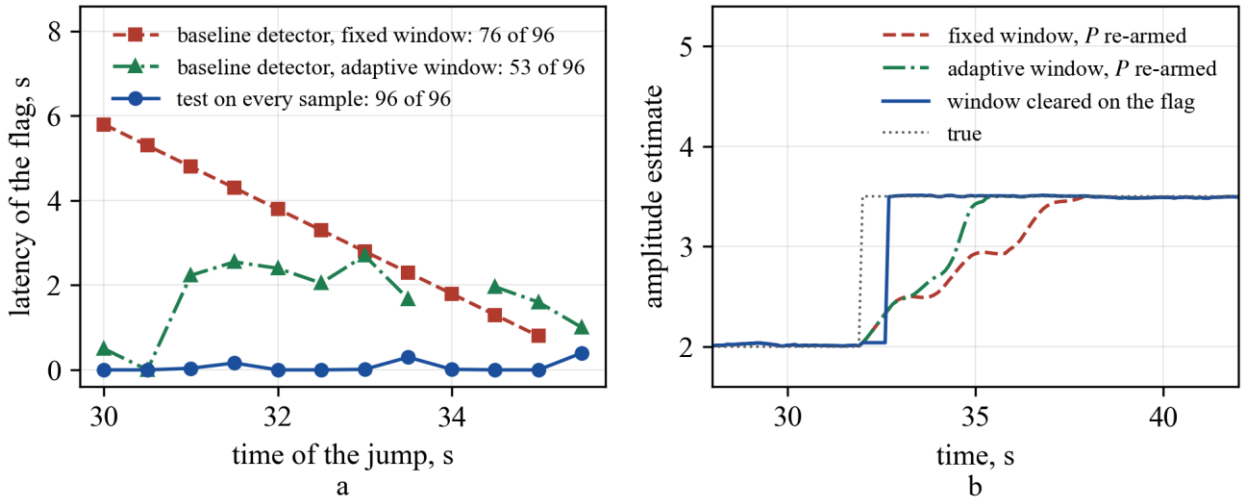


Fig. 3. **Alarm and recovery after a jump in amplitude:**
a – delay of the alarm against the position of the jump within the window;
b – amplitude estimate around a jump under three ways of re-arming the filter
***Source*: compiled by the authors**

Table 7. **Detection of a jump in amplitude**

Alarm, polling and re-arming	Flagged	Mean delay, s	False alarms	Settling, s
Baseline detector, once per window	76 / 53	3.27 / 1.35	1 / 0	5.26 / 2.36
Baseline detector, every sample, matched rate	85 / 85	0.08 / 0.08	0 / 0	5.26 / 2.37
Test (7), once per window	80 / 48	3.05 / 1.18	0 / 0	5.26 / 2.36
Test (7), every sample, P inflated	96 / 96	0.08 / 0.08	0 / 0	5.26 / 2.37
Test (7), every sample, window cleared	96 / 96	0.08 / 0.08	0 / 0	0.78 / 0.78

***Source*: compiled by the authors**

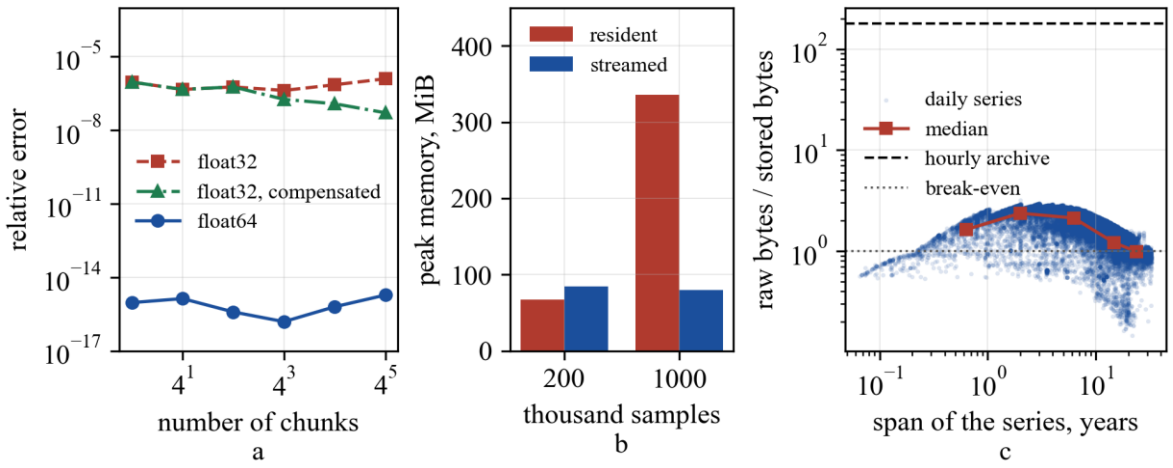


Fig. 4. **The reduction at scale:**
a – accumulation error against the number of chunks; b – peak memory against the record length; c – ratio of raw to stored size against the span of a series for the daily positions, with the hourly archive and the break-even line for reference
***Source*: compiled by the authors**

Table 9. **Processing of the two public archives**

Quantity	Daily positions	Hourly weather
Files attempted; failed on input	23 769; 594	13 345; 668
Files reduced; series fitted	23 175; 69 525	12 677; 12 677
Series accepted at 10^{-8}	66 708	11 711
Ill-conditioned; undersampled	1275; 1542	835; 131
Raw; stored size, GB	17.00; 14.42	51.59; 0.287
Reduction, 16 workers, s	133.3	49.1
Refit, s: workstation; Pi	–	18.75; 373.7
Post-seismic fitted; accepted	2541; 1441	–
Largest difference of τ; A; offset, standard errors	0.0083; 0.0075; 0.0044	–

***Source*: compiled by the authors**

DISCUSSION OF RESULTS

The discrete form of the criteria brings nonlinear least squares into the summation form of Chu et al. [8]: criterion (5) needs from the data only a sum, and the image is a mergeable synopsis [31] for nonlinear models.

Over a sliding window the same sum gives a recursive filter that at the windows of Table 1 is not ahead of a well-tuned extended Kalman filter in accuracy and with the longest window is ahead of the best Kalman setting on two plants and behind on two (Table 2); it changes the tuning, since the sensitivity analysed by Ljung [3] spans a factor of up to 465 across the Kalman grid and the filter needs no noise covariance. Adaptive Kalman filters [13], [15] and moving horizon estimation [18], [19] carry no mergeable statistic and were not compared here.

The findings are limited to synthetic plants, one spike model and one amplitude step, process noise set by hand per plant, windows of at most 180 samples, white noise for the rankings and false-alarm rates, no per-sample test on Kalman innovations, timings of a Python reference, a microcontroller build of the fixed-window update alone, a fixed order rule for long irregular series, and a running image that needs the span of the stream in advance.

CONCLUSIONS

1) Least squares in expansion coefficients has one additive statistic, the image of the data on a block or a Legendre basis, and the fit from it coincides with batch nonlinear least squares when the model's sensitivities lie in the span of the basis: at Legendre order 24 the two differ by at most $5.5 \cdot 10^{-4}$ of a standard error, and merged images reproduce the whole-record fit to rounding error. At a low order or with equal areas the fit is an approximation.

2) A recursive filter on the image of a sliding window has a bounded cost and state per step and needs no noise covariance. At the windows of Table 1 it is not ahead of a well-tuned extended Kalman filter and 1.7 times behind the static setting on the sinusoid; with a window of 180 samples (Table 2) it is ahead of the best Kalman setting on two plants and behind on two, at paired ratios of 0.63 and 0.61 on the damped oscillator and the rise and 1.55 and

1.19 on the sinusoid and the trajectory, no interval containing one, and a step costs eight to nine times a Kalman step; its gain state is overconfident, and the covariance of the window fit describes its errors on three of the four plants.

3) Clipping the window residuals keeps as many runs below five percent under gross anomalies as a Kalman filter gated with the fixed variance $r = 0.5$, at a third to a fifth of its median error, and stays level with one gated at the true variance; missing samples up to a fifth leave the filter level with the Kalman filter. The innovation test on every sample flagged all 96 jumps within 0.4 s at 0 to 0.25 false alarms per 10 000 samples under white noise, its lead in delay coming mostly from the polling rate.

4) The additive form carries over to large archives: two public archives of 17 GB and 52 GB were reduced once in memory that does not grow with the record, every series the gate passed is refitted from its image to 10^{-8} and, on a Raspberry Pi 5, to $7.3 \cdot 10^{-8}$ of the workstation's fit, and the post-seismic decay time to within 0.0083 of its standard error.

The scientific novelty is a form of least squares in expansion coefficients whose one additive statistic is both the data statistic of a distributed nonlinear fit and the measurement of a recursive filter with a change test on every sample; its practical value is an estimator for streams, archives and small devices without noise covariances. Further work concerns correlated noise, a compiled filter step and a rule for the basis order on long irregular series.

USE OF ARTIFICIAL INTELLIGENCE

While preparing this article, the authors used Google Gemini to verify the logical consistency of the text to improve its readability and ensure that bibliographic descriptions comply with international standards. The authors bear sole responsibility for the content of this publication.

Conflicts of Interest. The authors declare that they have no conflict of interest regarding this study, including financial, personal, authorship or other, which could influence the results presented in this article

Funding. The study was carried out without external funding

Accessibility data. The results are reproducible from the public repository and two public archives

Data and code availability. The code of the method, the notebooks that produce every table and figure of this article (notebooks 21, 22, 24 and 35 and the rig notebook) and the tables they export are in github.com/ringavirda/science-nonline. The Nevada Geodetic Laboratory records are at geodesy.unr.edu, and the Integrated Surface Database is at www.ncei.noaa.gov/products/land-based-station/integrated-surface-database

REFERENCES

1. Nocedal, J. & Wright, S. J. "Numerical Optimization". *2nd ed. Springer*. 2006, https://www.scopus.com/record/display.uri?eid=2-s2.0-85098056778&origin=resultslist. DOI: https://doi.org/10.1007/978-0-387-40065-5.

2. Kalman, R. E. "A new approach to linear filtering and prediction problems". *Journal of Basic Engineering*. 1960; 82 (1): 35–45, https://www.scopus.com/record/display.uri?eid=2-s2.0-85024429815&origin=resultslist. DOI: https://doi.org/10.1115/1.3662552.

3. Ljung, L. "Asymptotic behavior of the extended Kalman filter as a parameter estimator for linear systems". *IEEE Transactions on Automatic Control*. 1979; 24 (1): 36–50. https://www.scopus.com/record/display.uri?eid=2-s2.0-0018435192&origin=resultslist. DOI: https://doi.org/10.1109/TAC.1979.1101943.

4. Simon, D. "Optimal State Estimation: Kalman, H-infinity, and Nonlinear Approaches". *Wiley*. 2006, https://www.scopus.com/record/display.uri?eid=2-s2.0-84889830739&origin=resultslist. DOI: https://doi.org/10.1002/0470045345.

5. Mehra, R. K. "On the identification of variances and adaptive Kalman filtering". *IEEE Transactions on Automatic Control*. 1970; 15 (2): 175–184, https://www.scopus.com/record/display.uri?eid=2-s2.0-0014764781&origin=resultslist. DOI: https://doi.org/10.1109/TAC.1970.1099422.

6. Gustafsson, F. "Adaptive Filtering and Change Detection". *Wiley*. 2000. DOI: https://doi.org/10.1002/0470841613.

7. Dean, J. & Ghemawat, S. "MapReduce: Simplified data processing on large clusters". *Communications of the ACM*. 2008; 51 (1): 107–113, https://www.scopus.com/record/display.uri?eid=2-s2.0-37549003336&origin=resultslist. DOI: https://doi.org/10.1145/1327452.1327492.

8. Chu, C.-T., Kim, S. K., Lin, Y.-A., Yu, Y., Bradski, G., Ng, A. Y. & Olukotun, K. “Map-reduce for machine learning on multicore”. In *Advances in Neural Information Processing Systems 19*. Vancouver, Canada. 2006. p. 281–288, https://www.scopus.com/record/display.uri?eid=2-s2.0-56049109090&origin=resultslist. DOI: https://doi.org/10.7551/mitpress/7503.003.0040.

9. Lim, B. & Zohren, S. “Time-series forecasting with deep learning: A survey”. *Philosophical Transactions of the Royal Society A*. 2021; 379 (2194): 20200209, https://www.scopus.com/record/display.uri?eid=2-s2.0-85101380645&origin=resultslist. DOI: https://doi.org/10.1098/rsta.2020.0209.

10. Tuhanskykh, O., Baran, D. & Pysarchuk, O. “Method for statistical evaluation of nonlinear model parameters in statistical learning algorithms”. In: *Proceedings of the International Conference on Intelligent Computing and Information Technologies. Lecture Notes in Networks and Systems*. *Springer*. Singapore. 2024; Vol. 1013: 265–274, https://www.scopus.com/record/display.uri?eid=2-s2.0-85200974277&origin=resultslist. DOI: https://doi.org/10.1007/978-981-97-3559-4_21.

11. Pysarchuk, O. O. & Tuhanskykh, O. A. “Statistical evaluation of parameters in nonlinear models using integral-form of least squares method and differential non-Taylor transformations”. *Information, Computing and Intelligent Systems*. 2025; (6): 118–131. DOI: https://doi.org/10.20535/2786-8729.6.2025.332702.

12. Pysarchuk, O. O., Tuhanskykh, O. A. & Baran, D. R. “Method for determining the structure of nonlinear models for time series processing”. *Radio Electronics, Computer Science, Control*. 2025; 1: 53–62. DOI: https://doi.org/10.15588/1607-3274-2025-1-5.

13. Xu, H., Duan, K., Yuan, H., Xie, W. & Wang, Y. “Black box variational inference to adaptive Kalman filter with unknown process noise covariance matrix”. *Signal Processing*. 2020; 169: 107413, https://www.scopus.com/record/display.uri?eid=2-s2.0-85076170426&origin=resultslist. DOI: https://doi.org/10.1016/j.sigpro.2019.107413.

14. Zheng, B., Fu, P., Li, B. & Yuan, X. “A robust adaptive unscented Kalman filter for nonlinear estimation with uncertain noise covariance”. *Sensors*. 2018; 18 (3): 808, https://www.scopus.com/record/display.uri?eid=2-s2.0-85043461665&origin=resultslist. DOI: https://doi.org/10.3390/s18030808.

15. Kruse, T., Griebel, T. & Graichen, K. “Adaptive Kalman filtering: Measurement and process noise covariance estimation using Kalman smoothing”. *IEEE Access*. 2025; 13: 11863–11875, https://www.scopus.com/record/display.uri?eid=2-s2.0-85214850240&origin=resultslist. DOI: https://doi.org/10.1109/ACCESS.2025.3528348.

16. Fan, Y., Qiao, S., Wang, G. & Zhang, H. “An improved Sage-Husa variational robust adaptive Kalman filter with uncertain noise covariances”. *IEEE Sensors Journal*. 2024; 24 (18): 28921–28930, https://www.scopus.com/record/display.uri?eid=2-s2.0-85198247475&origin=resultslist. DOI: https://doi.org/10.1109/JSEN.2024.3421271.

17. Khodarahmi, M. & Maihami, V. “A review on Kalman filter models”. *Archives of Computational Methods in Engineering*. 2023; 30 (1): 727–747, https://www.scopus.com/record/display.uri?eid=2-s2.0-85139212368&origin=resultslist. DOI: https://doi.org/10.1007/s11831-022-09815-7.

18. Rao, C. V., Rawlings, J. B. & Mayne, D. Q. “Constrained state estimation for nonlinear discrete-time systems: Stability and moving horizon approximations”. *IEEE Transactions on Automatic Control*. 2003; 48 (2): 246–258, https://www.scopus.com/record/display.uri?eid=2-s2.0-0037291591&origin=resultslist. DOI: https://doi.org/10.1109/TAC.2002.808470.

19. Schiller, J. D. & Müller, M. A. “Nonlinear moving horizon estimation for robust state and parameter estimation”. *Automatica*. 2026; 185: 112790, https://www.scopus.com/record/display.uri?eid=2-s2.0-105025152050&origin=resultslist. DOI: https://doi.org/10.1016/j.automatica.2025.112790.

20. Huber, P. J. “Robust estimation of a location parameter”. *The Annals of Mathematical Statistics*. 1964; 35 (1): 73–101. DOI: https://doi.org/10.1214/aoms/1177703732.

21. Bar-Shalom, Y., Li, X. R. & Kirubarajan, T. “Estimation with Applications to Tracking and Navigation”. *Wiley*. 2001. DOI: https://doi.org/10.1002/0471221279.

22. Huang, H. & Zhang, H. “Student’s t-kernel-based maximum correntropy Kalman filter”. *Sensors*. 2022; 22 (4): 1683, https://www.scopus.com/record/display.uri?eid=2-s2.0-85124990066&origin=resultslist. DOI: https://doi.org/10.3390/s22041683.

23. Li, S., Shi, D., Zou, W. & Shi, L. “Multi-kernel maximum correntropy Kalman filter”. *IEEE Control Systems Letters*. 2022; 6: 1490–1495, https://www.scopus.com/record/display.uri?eid=2-s2.0-85115684735&origin=resultslist. DOI: https://doi.org/10.1109/LCSYS.2021.3114137.

24. Liu, C., Wang, G., Guan, X. & Huang, C. “Robust M-estimation-based maximum correntropy Kalman filter”. *ISA Transactions*. 2023; 136: 198–209, https://www.scopus.com/record/display.uri?eid=2-s2.0-85141980502&origin=resultslist. DOI: https://doi.org/10.1016/j.isatra.2022.10.025.

25. Gao, G., Gao, S., Hong, G., Peng, X. & Yu, T. “A robust INS/SRS/CNS integrated navigation system with the chi-square test-based robust Kalman filter”. *Sensors*. 2020; 20 (20): 5909, https://www.scopus.com/record/display.uri?eid=2-s2.0-85093677167&origin=resultslist.
DOI: https://doi.org/10.3390/s20205909.

26. Page, E. S. “Continuous inspection schemes”. *Biometrika*. 1954; 41 (1–2): 100–115. DOI: https://doi.org/10.1093/biomet/41.1-2.100.

27. Aminikhanghahi, S. & Cook, D. J. “A survey of methods for time series change point detection”. *Knowledge and Information Systems*. 2017; 51 (2): 339–367, https://www.scopus.com/record/display.uri?eid=2-s2.0-84986255561&origin=resultslist.
DOI: https://doi.org/10.1007/s10115-016-0987-z.

28. Forsythe, G. E. “Generation and use of orthogonal polynomials for data-fitting with a digital computer”. *Journal of the Society for Industrial and Applied Mathematics*. 1957; 5 (2): 74–88. DOI: https://doi.org/10.1137/0105007.

29. Messenger, D. A. & Bortz, D. M. “Weak SINDy: Galerkin-based data-driven model selection”. *Multiscale Modeling and Simulation*. 2021; 19 (3): 1474–1497, https://www.scopus.com/record/display.uri?eid=2-s2.0-85118117006&origin=resultslist.
DOI: https://doi.org/10.1137/20M1343166.

30. Bortz, D. M., Messenger, D. A. & Dukic, V. “Direct estimation of parameters in ODE models using WENDy: Weak-form estimation of nonlinear dynamics”. *Bulletin of Mathematical Biology*. 2023; 85 (11): 110. https://www.scopus.com/record/display.uri?eid=2-s2.0-85173745043&origin=resultslist.
DOI: https://doi.org/10.1007/s11538-023-01208-6.

31. Cormode, G., Garofalakis, M., Haas, P. J. & Jermaine, C. “Synopses for massive data: Samples, histograms, wavelets, sketches”. *Foundations and Trends in Databases*. 2011; 4 (1–3): 1–294. https://www.scopus.com/record/display.uri?eid=2-s2.0-84858061988&origin=resultslist.
DOI: https://doi.org/10.1561/1900000004.

32. Pritchard, N. & Patel, V. “Towards practical large-scale randomized iterative least squares solvers through uncertainty quantification”. *SIAM/ASA Journal on Uncertainty Quantification*. 2023; 11 (3): 996–1024, https://www.scopus.com/record/display.uri?eid=2-s2.0-85174683307&origin=resultslist. DOI: https://doi.org/10.1137/22M1515057.

33. Kahan, W. “Pracniques: Further remarks on reducing truncation errors”. *Communications of the ACM*. 1965; 8 (1): 40, https://www.scopus.com/record/display.uri?eid=2-s2.0-84943347456&origin=resultslist. DOI: https://doi.org/10.1145/363707.363723.

34. Blewitt, G., Hammond, W. C. & Kreemer, C. “Harnessing the GPS data explosion for interdisciplinary science”. *Eos*. 2018. p. 99. DOI: https://doi.org/10.1029/2018EO104623.

35. Smith, A., Lott, N. & Vose, R. “The Integrated Surface Database: Recent developments and partnerships”. *Bulletin of the American Meteorological Society*. 2011; 92 (6): 704–708, https://www.scopus.com/record/display.uri?eid=2-s2.0-79961046329&origin=resultslist.
DOI: https://doi.org/10.1175/2011BAMS3015.1.

36. Bevis, M. & Brown, A. “Trajectory models and reference frames for crustal motion geodesy”. *Journal of Geodesy*. 2014; 88 (3): 283–311, https://www.scopus.com/record/display.uri?eid=2-s2.0-84894353002&origin=resultslist. DOI: https://doi.org/10.1007/s00190-013-0685-5.

Received 02.09.2026
Received after revision 21.10.2026
Accepted 30.10.2026

DOI: https://doi.org/10.15276/aait.09.2026.32
УДК 004.62:519.246.8

Потокове та розподілене оцінювання параметрів нелінійних моделей методом найменших квадратів за коефіцієнтами розкладу

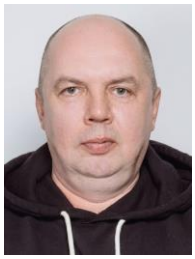
Писарчук Олексій Олександрович[1)]
ORCID: https://orcid.org/0000-0001-5271-0248; platinumpa2212@gmail.com. Scopus Author ID: 7801331277
Туганських Олександр Антонович[1)]
ORCID: https://orcid.org/0009-0007-9024-2570; ringanaro@gmail.com. Scopus Author ID: 59255570200
[1)] Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», пр. Берестейський, 37. Київ, 03056, Україна

АНОТАЦІЯ

Актуальність. Оцінювання нелінійної моделі методом найменших квадратів перечитує кожен відлік на кожній ітерації, що не підходить ані для потоку даних, ані для розподіленого архіву. Рекурентні фільтри потребують наперед заданих коваріацій шумів, а засоби розподіленої обробки об'єднують статистики лише лінійних моделей. **Мета і завдання.** Метою є розроблення рекурентної та адитивної форм методу найменших квадратів за коефіцієнтами розкладу, критерію рівних площ і методу найменших квадратів за коефіцієнтами Лежандра, для нелінійного оцінювання на потоці з обмеженими витратами та на архіві без повторного читання. Завдання: встановити умови, за яких оцінка за адитивним зображенням даних збігається з пакетною оцінкою найменших квадратів, побудувати на ній віконний фільтр і порівняти його з розширеним фільтром Калмана, оснастити фільтр засобами для грубих аномалій, пропущених відліків і змін параметрів та застосувати адитивну форму до великих архівів. **Використані методи.** Проєкція даних на блоковий базис або базис Лежандра, її матриця Грама та суми відліків утворюють адитивне зображення, за яким модель оцінюють вибіленим методом найменших квадратів. Фільтр є демпфованим оновленням в інформаційній формі з рівнем шуму, оціненим за вікном, обмеженням залишків і перевіркою інновації на кожному відліку. **Результати.** Оцінка за зображенням збігається з пакетною оцінкою найменших квадратів, коли чутливості моделі лежать у просторі базису, а об'єднані зображення відтворюють її з точністю до округлення. За білого шуму віконний фільтр рівний статичному фільтру Калмана або випереджає його на більшості об'єктів і поступається йому на синусоїді, за грубих аномалій зберігає стільки ж прогонів, скільки фільтр Калмана з добре налаштованим стробуванням, і швидко виявляє стрибки параметрів, головно завдяки перевірці на кожному відліку. Відкриті архіви один раз зведено до зображень і оцінено за ними на одноплатному комп'ютері, зокрема нелінійну сталу часу згасання. **Висновки.** Одна адитивна статистика обслуговує потік, архів і малий обчислювач. Фільтр порівнянний за точністю з добре налаштованим фільтром Калмана і дорожчий на кожен крок, але не потребує коваріацій шумів, стійкий до грубих похибок без стробування і перевіряє зміни на кожному відліку.

Ключові слова: оцінювання параметрів; нелінійна модель; проєкція даних; рекурентний фільтр; виявлення змін; розподілене оброблення

ABOUT THE AUTHORS



Oleksii O. Pysarchuk - Doctor of Engineering Sciences, Professor, Department of Computer Engineering. National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", 37, Beresteiskyi Ave. Kyiv, 03056, Ukraine
ORCID: https://orcid.org/0000-0001-5271-0248; platinumpa2212@gmail.com. Scopus Author ID: 7801331277
Research field: analysis and synthesis of complex information systems; mathematical modelling; information processing; data science; machine learning; decision support systems

Писарчук Олексій Олександрович - доктор технічних наук, професор кафедри Обчислювальної техніки. Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського", проспект Берестейський, 37. Київ, 03056, Україна
Галузь досліджень: аналіз і синтез складних інформаційних систем; математичне моделювання; оброблення інформації; наука про дані; машинне навчання; системи підтримки прийняття рішень



Oleksandr A. Tuhanskykh - PhD student, Assistant, Department of Computer Engineering. National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", 37, Beresteiskyi Ave. Kyiv, 03056, Ukraine
ORCID: https://orcid.org/0009-0007-9024-2570; ringanaro@gmail.com. Scopus Author ID: 59255570200
Research field: models and methods of nonlinear time-series analysis

Туганських Олександр Антонович - аспірант, асистент кафедри Обчислювальної техніки. Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського", пр. Берестейський, 37. Київ, 03056, Україна
Галузь досліджень: моделі та методи аналізу нелінійних часових рядів