Mechanical Systems and Signal Processing 216 (2024) 111455

Contents lists available at ScienceDirect

Mechanical Systems and Signal Processing

journal homepage: www.elsevier.com/locate/ymssp

Smooth least absolute deviation estimators for outlier-proof identification

Janusz Kozłowski, Zdzisław Kowalczuk [*,1]

Gdańsk University of Technology, Faculty of Electronics, Telecommunications and Informatics, Department of Robotics and Decision Systems, 80-233 Gdańsk, Poland

ARTICLE INFO

Communicated by Spilios Fassois

Keywords:
Least squares
Least absolute deviation
Linear models
Outliers
Smooth approximation
System identification

ABSTRACT

The paper proposes to identify the parameters of linear dynamic models based on the original implementation of least absolute deviation estimators. It is known that the object estimation procedures synthesized in the sense of the least sum of absolute prediction errors are particularly resistant to occasional outliers and gaps in the analyzed system data series, while the classical least squares procedure unfortunately becomes of little use for reliably identifying systems in the presence of destructive measurement errors. Bearing in mind that the classic task of minimizing the quality functional of absolute deviations encounters fundamental analytical problems, it is proposed to use a dedicated iterative estimator for off-line evaluation of the parameters of the analyzed process. In addition, a simplified recursive version of the absolute deviation estimation procedure was developed, which allows for practical on-line tracking of the evolution of variable parameters of non-stationary systems. Importantly, a novel refinement of the discussed absolute deviation estimators was proposed to effectively overcome some inconvenient numerical effects. We also present an interesting comparison of the improved (by non-linear modification) iterative absolute-deviation estimator with the classical Gauss-Newton gradient algorithm, which leads to constructive conclusions. Finally, using computer simulations, the operation of the developed iterative and recursive estimators minimizing the absolute deviation is illustrated. The work ends with an indication of directions for further research.

1. Introduction

The unprecedented development of science and technology, which is conventionally referred to as the „semiconductor revolution" (because it is associated with the invention of the transistor in 1947), has also radically influenced progress in numerous areas of human life. In parallel, these results also accelerated research in areas such as electronics, information technology, telecommunications, industrial robotics and automation. Especially in the domain of broadly understood automation, we observe great theoretical and practical progress in the field of diagnostics, process supervision and adaptive control [1,2], digital filtering and prediction [3,4], and system identification [5,6].

In many such practical professional implementations, effective ways of calculating the parameters of mathematical models turn out to be invaluable. This is because for industrial monitoring and/or supervision systems, such estimators are necessary when assessing

* Corresponding author.
E-mail addresses: jk23@eti.pg.edu.pl (J. Kozłowski), kova@eti.pg.edu.pl (Z. Kowalczuk).
[1] The authors contributed equally.

https://doi.org/10.1016/j.ymssp.2024.111455
Received 18 November 2023; Received in revised form 20 February 2024; Accepted 21 April 2024
Available online 3 May 2024

the correctness or quality of controlled production sub-processes. In the reliable mechanization of such procedures, proper modeling of the process dynamics is essential. Most often, operator models of transfer functions, difference or differential equations, or representations in the state space are taken into account here. In this context, the importance of the choice between a discrete-time description (instrumental, mathematical) and a continuous-time model (physically/ energetically motivated) should be raised.

Historically, as a result of the great progress in the production of digital computers, there was a tendency to uniformly treat numerically implemented algorithms and models of the analyzed systems in a common discrete domain, which was also manifested in the main directions of research undertaken in the field of parameter estimation. Users of discrete descriptions find measurable benefits here, because numerical implementations of such easy-to-use models are also significantly supported by advanced libraries of professional software. Though, a discrete representation is usually a purely mathematical description with unitless parameters, which rarely have a direct link with physical quantities (e.g. energy). What is more, the parameters of the discrete-time transfer function strongly depend on the sampling frequency. In general, the choice between continuous-time and discrete-time modeling is not a trivial problem and depends on many factors. For example, when system identification is aimed at evaluating significant and basic physical quantities, analog modeling is strongly recommended [7,8], while when the identified quantities constitute data needed for some specific processing (e.g. tuning of a digital controller), it is recommended to act in target discrete domain.

It is obvious that the quality of the processed measurements has a fundamental impact on the reliable identification of the monitored process. The signals recorded by measuring devices usually contain some undesirable additions or interferences such as additive (high frequency) noise, deterministic (systematic) bias or occasional deviations (outliers). Interfering noise (white or correlated) can usually be eliminated with auxiliary (low-pass) pre-filters. In turn, the phenomenon of bias can be reduced by proper calibration of the sensors or the use of dedicated compensation methods. Unfortunately, in the case of large measurement errors, the classical least squares (LS) method is unable to completely process data distorted by destructive errors. Therefore, it is recommended to use special estimation procedures that are insensitive to outliers [9,10].

There are basically two concepts for parameter estimation that are insensitive to outliers:

a) the use of dedicated detection procedures that allow for effective isolation of random outliers in measurement data, which in turn allows for safe identification using the least squares scheme,
b) the use of methods that are inherently insensitive to large measurement errors (such as the procedures presented here based on non-quadratic quality indices).

Concept (a) is clearly less practical, as it must rely on additional processing based on hypothesis verification methods (such as the Grubbs criterion) to detect off-line outliers in the recorded measurements. Unfortunately, this approach is reliable for eliminating single errors, but does not necessarily work for a sequence of several outliers in the recorded data set. Thus, any remaining outliers may easily impair the performance of LS estimators (which are not resistant to large errors).

Therefore, in our study, we consider concept (b), which aims to desensitize identification procedures to data with outliers [11,12]. For this purpose, we propose iterative-recursive estimation procedures with the least absolute deviation (LA). After discussing them, we present their practical performance in several computer simulations with obvious measurement errors and experimentally confirm the desirable feature of insensitivity to outliers.

These proprietary estimators, together with an inventive improvement that allows us to overcome specific implementation problems (e.g. scaling factors close to zero), are our contribution to the field of identification insensitive to measurement errors. In addition, the validity of the absolute deviation approach is undeniable due to the mathematically proven superiority of our modified iterative estimator over the estimation procedure based on the classical Gauss-Newton gradient algorithm. And to show the consistency of the developed methodology, the basic property of the original iterative LA estimator is briefly recalled in the appendix to this article (with reference to the rigorous proof given in [13]).

The above issues are presented in this article in the following order. In Section 2, derivations of the batch (off-line) and recursive (on-line) implementations of the basic least squares algorithm is outlined, which makes it easy to follow the reasoning associated with the target LA procedures. Then iterative (batch) and recursive (approximate) least absolute deviation procedures are discussed in Section 3. In turn, Section 4 proposes an inventive redefinition of the standard absolute-deviation functional that eliminates the cumbersome numerical problem of small divisors. In addition, Section 5 presents an instructive confrontation of the improved LA scheme with an estimation procedure implemented based on the well-known Gauss-Newton optimization pattern. Finally, Section 6 illustrates the unprecedented properties of the LA method using computer simulations where the processed data is contaminated with large measurement errors. In the summary of the work, in Section 7, the motivation for undertaking research is emphasized, the main results are briefly listed and directions for further research are outlined. In addition, Appendix A briefly explains an important property (convergence) of the basic iterative absolute-deviation procedure.

2. Batch and recursive LS algorithms

Consider the classical SISO regression model of a supervised process:

$$\chi(l) = \boldsymbol{\varphi}^{\mathrm{T}}(l)\boldsymbol{\theta} + e(l) \tag{1}$$

$$\boldsymbol{\varphi}(l) = [\varphi_1(l) \ \cdots \ \varphi_n(l)]^{\mathrm{T}} \tag{2}$$

$$\boldsymbol{\theta} = [\vartheta_1 \ \cdots \ \vartheta_n]^{\mathrm{T}} \tag{3}$$

where $\chi(l)$ stands for the measurement of the output, while $\boldsymbol{\varphi}(l)$ and $\boldsymbol{\theta}$ denote some regression and parameter vectors, respectively. The term $e(l)$, called equation error, prediction error (or residual bias), is also a good representation of the modeling inaccuracy or noise/disturbance component.

The model parameter vector $\boldsymbol{\theta}$ can be easily estimated using known identification algorithms, including the classical least squares procedure.

To facilitate our target consideration of the absolute deviation approach, we first briefly outline the typical reasoning leading to batch (algebraic) and recursive (on-line) implementations of the LS method.

The weighted least squares estimator follows directly from the minimization of the handy square index [14]

$$I(\boldsymbol{\theta}) = \frac{1}{2}\sum_{l=1}^{k}\gamma(l)e^2(l) = \frac{1}{2}\sum_{l=1}^{k}\lambda^{k-l}\left[\chi(l) - \boldsymbol{\varphi}^{\mathrm{T}}(l)\boldsymbol{\theta}\right]^2 \tag{4}$$

in which the sequence of positive weights $\gamma(l)$ takes the convenient exponential form $\gamma(l) = \lambda^{k-l}$. The weighting factor λ controlling the rate of forgetting (in the case of identifying a non-stationary/variant object) is usually in the range $[0.9 \ , \ 1]$. The coefficient λ is related to the coefficient $L = 1/(1-\lambda)$, which represents the 'effective number of observations'.

The strictly convex functional (4) reaches a minimum provided that its gradient is equal to zero

$$\nabla_{\boldsymbol{\theta}} I = -\sum_{l=1}^{k}\lambda^{k-l}\boldsymbol{\varphi}(l)\left[\chi(l) - \boldsymbol{\varphi}^{\mathrm{T}}(l)\boldsymbol{\theta}\right] = \mathbf{0} \tag{5}$$

Consequently, the LS estimation of vector $\boldsymbol{\theta}$ is given by

$$\widehat{\boldsymbol{\theta}} = \left[\sum_{l=1}^{k}\lambda^{k-l}\boldsymbol{\varphi}(l)\boldsymbol{\varphi}^{\mathrm{T}}(l)\right]^{-1}\left[\sum_{l=1}^{k}\lambda^{k-l}\boldsymbol{\varphi}(l)\chi(l)\right] \tag{6}$$

Finally, given the positive definite Hessian of $I(\boldsymbol{\theta})$,

$$\nabla_{\boldsymbol{\theta}}^2 I = \left[\sum_{l=1}^{k}\lambda^{k-l}\boldsymbol{\varphi}(l)\boldsymbol{\varphi}^{\mathrm{T}}(l)\right] > 0 \tag{7}$$

we conclude that (6) forms a unique minimum of (4).

The batch solution (6) is characterized by a tiresome matrix inversion, which, however, can be circumvented by using the matrix inversion lemma (MIL) [14]:

$$\left[\lambda\boldsymbol{M} + \boldsymbol{v}\boldsymbol{v}^{\mathrm{T}}\right]^{-1} = \frac{1}{\lambda}\left[\boldsymbol{M}^{-1} - \frac{\boldsymbol{M}^{-1}\boldsymbol{v}\boldsymbol{v}^{\mathrm{T}}\boldsymbol{M}^{-1}}{\lambda + \boldsymbol{v}^{\mathrm{T}}\boldsymbol{M}^{-1}\boldsymbol{v}}\right] \tag{8}$$

where $\boldsymbol{M}$ is an invertible square matrix, $\boldsymbol{v}$ is any column vector of compatible dimension, and λ is a positive number. By introducing the aggregated symbols $\boldsymbol{R}(k)$ and $\boldsymbol{s}(k)$ for the main two terms in (6), they can then be written in recursive form:

$$\boldsymbol{R}(k) = \sum_{l=1}^{k}\lambda^{k-l}\boldsymbol{\varphi}(l)\boldsymbol{\varphi}^{\mathrm{T}}(l) \tag{9}$$

$$\boldsymbol{R}(k) = \lambda\boldsymbol{R}(k-1) + \boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k) \tag{10}$$

$$\boldsymbol{s}(k) = \sum_{l=1}^{k}\lambda^{k-l}\boldsymbol{\varphi}(l)\chi(l) \tag{11}$$

$$\boldsymbol{s}(k) = \lambda\boldsymbol{s}(k-1) + \boldsymbol{\varphi}(k)\chi(k) \tag{12}$$

Thus, lemma (8) applied to (10) gives the inverse of $\boldsymbol{R}(k)$, denoted for convenience by the matrix $\boldsymbol{P}(k)$

$$\boldsymbol{P}(k) = \boldsymbol{R}^{-1}(k) = \left[\lambda\boldsymbol{R}(k-1) + \boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)\right]^{-1} = \frac{1}{\lambda}\left[\boldsymbol{P}(k-1) - \frac{\boldsymbol{P}(k-1)\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{P}(k-1)}{\lambda + \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{P}(k-1)\boldsymbol{\varphi}(k)}\right] \tag{13}$$

After simple rearrangements, the final estimate (6) is

$$\begin{aligned}\widehat{\boldsymbol{\theta}}(k) &= \boldsymbol{R}^{-1}(k)\boldsymbol{s}(k) = \boldsymbol{P}(k)\boldsymbol{s}(k) = \frac{1}{\lambda}\left[\boldsymbol{P}(k-1) - \frac{\boldsymbol{P}(k-1)\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{P}(k-1)}{\lambda + \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{P}(k-1)\boldsymbol{\varphi}(k)}\right]\left[\lambda\boldsymbol{s}(k-1) + \boldsymbol{\varphi}(k)\chi(k)\right] \\ &= \widehat{\boldsymbol{\theta}}(k-1) + \boldsymbol{P}(k)\boldsymbol{\varphi}(k)\left[\chi(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\widehat{\boldsymbol{\theta}}(k-1)\right]\end{aligned} \tag{14}$$

The above recursive LS procedure (13)-(14) starts with the adoption of the initial matrix (13) with the largest (realistically) values on the main diagonal of $\boldsymbol{P}(k)$, e.g. $\boldsymbol{P}(0) = \mathrm{diag}\left[10^5 \cdots 10^5\right]$.

The Hessian $\boldsymbol{R}(k)$ found in (6), (7) and (9) can be loosely referred to as the 'information matrix', although strictly speaking it should be rescaled by the variance σ_e^2 of the residual error $e(k)$ to more appropriately represent information matrix, i.e. $\sigma_e^2\boldsymbol{R}(k)$. Its inverse, $\boldsymbol{P}(k)$ shown in the LS algorithm (13)-(14), can thus be called the 'covariance matrix', although the exact form is given here by $\boldsymbol{P}(k)/\sigma_e^2$.

Let us also recall the modification consisting in the introduction of an instrumental variable (IV) to the LS procedure. The desired IV solution can be obtained by replacing the column regression vector $\boldsymbol{\varphi}(k)$ with some 'noise-free' instrument $\boldsymbol{\xi}(k)$ when evaluating the gradient (5). As a result of a rearrangement similar to (9)-(14) we obtain a recursive procedure analogous to (13)-(14), where $\boldsymbol{\varphi}(k)$ is replaced by $\boldsymbol{\xi}(k)$ and $\boldsymbol{\varphi}^{\mathrm{T}}(k)$ remains unchanged.

In summary, we can introduce a common template for the LS and IV recursive procedures, in which the innovated covariance matrix $\boldsymbol{P}(k)$ together with the 'a priori' prediction error $\varepsilon(k)$ are used to update the estimate of the vector $\boldsymbol{\theta}$ [14]:

$$\varepsilon(k) = \chi(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\widehat{\boldsymbol{\theta}}(k-1) \tag{15}$$

$$\boldsymbol{P}(k) = \frac{1}{\lambda}\left[\boldsymbol{P}(k-1) - \frac{\boldsymbol{P}(k-1)\boldsymbol{\eta}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{P}(k-1)}{\lambda + \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{P}(k-1)\boldsymbol{\eta}(k)}\right] \tag{16}$$

$$\widehat{\boldsymbol{\theta}}(k) = \widehat{\boldsymbol{\theta}}(k-1) + \boldsymbol{P}(k)\boldsymbol{\eta}(k)\varepsilon(k) \tag{17}$$

where the auxiliary vector $\boldsymbol{\eta}(k)$ is $\boldsymbol{\varphi}(k)$ or $\boldsymbol{\xi}(k)$ for the LS and IV methods, respectively.

It is easy to show that the unweighted LS algorithm (at $\lambda = 1$) gives a consistent estimate of $\boldsymbol{\theta}$, provided that the prediction error $e(k)$ and the regression vector $\boldsymbol{\varphi}(k)$ have no correlation: $E\{\boldsymbol{\varphi}(k)e(k)\} = \boldsymbol{0}$. This takes place when the random variables $\{e(1), e(2)\cdots\}$ making the residual process $e(k)$ are zero-mean independent (the white noise sequence).

Since in practical situations the equation error $e(k)$ is most often a correlated process, the LS estimate of θ contains a systematic error. The IV variant of the LS procedure is a good method to avoid inconsistencies and asymptotic bias in the estimations, because the instrument $\boldsymbol{\xi}(k)$ is noise-free and does not correlate: $E\{\boldsymbol{\xi}(k)e(k)\} = \boldsymbol{0}$. Useful rules for generating the instrument $\boldsymbol{\xi}(k)$ can be found in [15,16].

3. Batch and recursive LA algorithms

The basic estimator of the smallest absolute deviation used to identify the model (1)-(3) results from the minimization of the non-quadratic functional

$$J(\boldsymbol{\theta}) = \sum_{l=1}^{k}\gamma(l)|e(l)| = \sum_{l=1}^{k}\lambda^{k-l}\left|\chi(l) - \boldsymbol{\varphi}^{\mathrm{T}}(l)\boldsymbol{\theta}\right| \tag{18}$$

with the weight sequence $\gamma(l)$, similarly as in (4).

Of course, the analytical minimization of such a non-square functional LA can be problematic. However, using an appropriate estimate of the prediction error $e(k)$ generated by an auxiliary estimator, a zero gradient of (18) can be achieved in the following approximate way

$$\nabla_{\boldsymbol{\theta}}J = -\sum_{l=1}^{k}\lambda^{k-l}\boldsymbol{\varphi}(l)\mathrm{sign}(e) = -\sum_{l=1}^{k}\lambda^{k-l}\boldsymbol{\varphi}(l)\frac{e(l)}{|e(l)|} \approx -\sum_{l=1}^{k}\lambda^{k-l}\frac{\boldsymbol{\varphi}(l)e(l)}{|\widehat{e}(l)|} = -\sum_{l=1}^{k}\lambda^{k-l}\frac{\boldsymbol{\varphi}(l)[\chi(l) - \boldsymbol{\varphi}^{\mathrm{T}}(l)\boldsymbol{\theta}]}{|\widehat{e}(l)|} = \boldsymbol{0} \tag{19}$$

where a helpful tautology was used (satisfied for $e \neq 0$):

$$\frac{d|e|}{de} = \mathrm{sign}(e) = \frac{e}{|e|} \tag{20}$$

Finally, based on (19), an approximate (LA) estimate of $\boldsymbol{\theta}$ can be expressed in batch (or algebraic) form as

$$\widehat{\boldsymbol{\theta}} = \left[\sum_{l=1}^{k}\lambda^{k-l}\frac{\boldsymbol{\varphi}(l)}{|\widehat{e}(l)|}\boldsymbol{\varphi}^{\mathrm{T}}(l)\right]^{-1}\left[\sum_{l=1}^{k}\lambda^{k-l}\frac{\boldsymbol{\varphi}(l)}{|\widehat{e}(l)|}\chi(l)\right] \tag{21}$$

Equation (21) coincides with (6) if only the vector $\boldsymbol{\varphi}(l)$ is scaled with the absolute value of the estimate of the equation error $e(l)$.

By implementing the above procedure in an iterative manner, the error estimation can be refined and thus the accuracy of parameter estimation can be improved. Namely, using the LS estimation of $\boldsymbol{\theta}$ given by (6) to approximate the successive values of the residual error $\widehat{e}(l) = \chi(l) - \boldsymbol{\varphi}^{\mathrm{T}}(l)\widehat{\boldsymbol{\theta}}$, (for $l = 1\cdots k$), we are able to initially calculate the estimate LA according to (21). Then, using the estimate of $\boldsymbol{\theta}$ thus obtained, the residual error $\widehat{e}(l)$ can be recalculated (updated) and reapplied in (21) to obtain the final (current) estimate of LA. This reasoning obviously overlaps with the conventional 'successive approximation' method and leads to the following iterative implementation of the batch LA procedure (for $r = 0,\ 1,\ \cdots$):

$$\widehat{e}^{[r]}(l) = \chi(l) - \boldsymbol{\varphi}^{\mathrm{T}}(l)\widehat{\boldsymbol{\theta}}^{[r]} \tag{22}$$

$$\widehat{\boldsymbol{\theta}}^{[r+1]} = \left[\sum_{l=1}^{k} \lambda^{k-l} \frac{\boldsymbol{\varphi}(l)\boldsymbol{\varphi}^{\mathrm{T}}(l)}{|\widehat{e}^{[r]}(l)|}\right]^{-1} \left[\sum_{l=1}^{k} \lambda^{k-l} \frac{\boldsymbol{\varphi}(l)\chi(l)}{|\widehat{e}^{[r]}(l)|}\right] \tag{23}$$

Note that by rearranging formula (22) to the regression form (for the output: $\chi(l) = \boldsymbol{\varphi}^{\mathrm{T}}(l)\widehat{\boldsymbol{\theta}}^{[r]} + \widehat{e}^{[r]}(l)$), we can show (23) as an innovation

$$\widehat{\boldsymbol{\theta}}^{[r+1]} = \widehat{\boldsymbol{\theta}}^{[r]} + [\boldsymbol{H}(k)]^{-1}\boldsymbol{\psi}(k) \tag{24}$$

$$\boldsymbol{H}(k) = \sum_{l=1}^{k} \lambda^{k-l} \frac{\boldsymbol{\varphi}(l)\boldsymbol{\varphi}^{\mathrm{T}}(l)}{|\widehat{e}^{[r]}(l)|} \tag{25}$$

$$\boldsymbol{\psi}(k) = \sum_{l=1}^{k} \lambda^{k-l} \frac{\boldsymbol{\varphi}(l)\widehat{e}^{[r]}(l)}{|\widehat{e}^{[r]}(l)|} = \sum_{l=1}^{k} \lambda^{k-l}\boldsymbol{\varphi}(l)\mathrm{sign}\left(\widehat{e}^{[r]}(l)\right) \tag{26}$$

where the error term $\widehat{e}^{[r]}(l)$ is given by (22), the vector $'-\boldsymbol{\psi}(k)'$ is the gradient and $\boldsymbol{H}(k)$ is the Hessian of (18).

In practice, the above iterative LA procedure is stopped when the decrease in successive values of criterion (18) becomes insignificant (relative to the numerical threshold Δ_{min}):

$$J\left(\widehat{\boldsymbol{\theta}}^{[r]}\right) - J\left(\widehat{\boldsymbol{\theta}}^{[r+1]}\right) < \Delta_{min} \tag{27}$$

The original concept of iterative LA processing [17], referred to as the "re-weighted least squares" method, has been further developed and applied to many practical problems [13,18,19]. It should be emphasized here that the iteratively calculated sequence of functional values (18) is proved to be decreasing [13]: $J\left(\widehat{\boldsymbol{\theta}}^{[r+1]}\right) < J\left(\widehat{\boldsymbol{\theta}}^{[r]}\right)$, which confirms the sense of test (27). Moreover, since each value of (18) is positive, the sequence $J\left(\widehat{\boldsymbol{\theta}}^{[r]}\right)$ has a lower bound. Thus, taking into account that every monotonic sequence bounded from below has a limit, it can be concluded that the LA algorithm (24)-(26) minimizes the quality index (18). The essence of the above-mentioned convergence property is briefly recalled in Appendix A.

Moreover, we should be aware that in the above LA procedure there is likely to be a problem of small divisors (factors $|\widehat{e}^{[r]}(l)|$). Thus, absolute values $|\widehat{e}^{[r]}(l)|$ that are very close to numerical zero can be replaced with a threshold value (e_{min}). This practical method of 'regularization' is commonly used in many numerical procedures.

Our objective function LA (18) given as a sum of absolute values is convex, but not in the strict sense. So it may happen that the LA functional adopts a 'flat minimum'. However, this is an extremely rare case where the actual gradient (26) becomes approximately zero $\|\boldsymbol{\psi}(k)\| = 0$, so we have stagnation circumstance $J\left(\widehat{\boldsymbol{\theta}}^{[r+1]}\right) = J\left(\widehat{\boldsymbol{\theta}}^{[r]}\right)$ and no innovation $\widehat{\boldsymbol{\theta}}^{[r+1]} = \widehat{\boldsymbol{\theta}}^{[r]}$. However, such a stagnation of the LA index is immediately detected by the stop test (27).

In conclusion, the numerical problem of divisors close to zero can be seen as an implementation flaw of the iterative batch LA approach presented above.

A recursive implementation of the LA method can be derived in a similar way as shown for the LS scheme. First, we introduce new symbols $\boldsymbol{R}(k)$ and $\boldsymbol{s}(k)$ for the key terms presented in (21) and represent them in a recursive way:

$$\boldsymbol{R}(k) = \sum_{l=1}^{k} \lambda^{k-l} \frac{\boldsymbol{\varphi}(l)}{|\widehat{e}(l)|}\boldsymbol{\varphi}^{\mathrm{T}}(l) \tag{28}$$

$$\boldsymbol{R}(k) = \lambda\boldsymbol{R}(k-1) + \frac{\boldsymbol{\varphi}(k)}{|\widehat{e}(k)|}\boldsymbol{\varphi}^{\mathrm{T}}(k) \tag{29}$$

$$\boldsymbol{s}(k) = \sum_{l=1}^{k} \lambda^{k-l} \frac{\boldsymbol{\varphi}(l)}{|\widehat{e}(l)|}\chi(l) \tag{30}$$

$$\boldsymbol{s}(k) = \lambda\boldsymbol{s}(k-1) + \frac{\boldsymbol{\varphi}(k)}{|\widehat{e}(k)|}\chi(k) \tag{31}$$

The MIL (8) applied to (29) yields the inverse of $\boldsymbol{R}(k)$:

$$\begin{aligned}\boldsymbol{P}(k) = \boldsymbol{R}^{-1}(k) &= \left[\lambda\boldsymbol{R}(k-1) + \frac{\boldsymbol{\varphi}(k)}{|\widehat{e}(k)|}\boldsymbol{\varphi}^{\mathrm{T}}(k)\right]^{-1} = \frac{1}{\lambda}\left[\boldsymbol{P}(k-1) - \frac{\boldsymbol{P}(k-1)\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{P}(k-1)}{\lambda|\widehat{e}(k)| + \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{P}(k-1)\boldsymbol{\varphi}(k)}\right] \\ &\approx \frac{1}{\lambda}\left[\boldsymbol{P}(k-1) - \frac{\boldsymbol{P}(k-1)\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{P}(k-1)}{\lambda|\varepsilon(k)| + \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{P}(k-1)\boldsymbol{\varphi}(k)}\right]\end{aligned} \tag{32}$$

After classical transformations, instead of the batch version (21), a more convenient recursive LA estimator can be used:

$$\widehat{\boldsymbol{\theta}}(k)=\boldsymbol{R}^{-1}(k)\boldsymbol{s}(k)=\boldsymbol{P}(k)\boldsymbol{s}(k)=\frac{1}{\lambda}\left[\boldsymbol{P}(k-1)-\frac{\boldsymbol{P}(k-1)\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{P}(k-1)}{\lambda|\widehat{e}(k)|+\boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{P}(k-1)\boldsymbol{\varphi}(k)}\right]\left[\lambda\boldsymbol{s}(k-1)+\frac{\boldsymbol{\varphi}(k)}{|\widehat{e}(k)|}\chi(k)\right]$$
$$=\widehat{\boldsymbol{\theta}}(k-1)+\boldsymbol{P}(k)\frac{\boldsymbol{\varphi}(k)}{|\widehat{e}(k)|}\left[\chi(k)-\boldsymbol{\varphi}^{\mathrm{T}}(k)\widehat{\boldsymbol{\theta}}(k-1)\right]\approx\widehat{\boldsymbol{\theta}}(k-1)+\boldsymbol{P}(k)\boldsymbol{\varphi}(k)\mathrm{sign}\left(\chi(k)-\boldsymbol{\varphi}^{\mathrm{T}}(k)\widehat{\boldsymbol{\theta}}(k-1)\right) \tag{33}$$

In the formula above, the 'a priori' prediction error $\varepsilon(k)=\chi(k)-\boldsymbol{\varphi}^{\mathrm{T}}(k)\widehat{\boldsymbol{\theta}}(k-1)$ was used as the best possible on-line measure of the unknown equation error ($|\varepsilon(k)|\approx|\widehat{e}(k)|$), and the sign tautology (20) was used to represent the ratio $\varepsilon(k)/|\varepsilon(k)|$. As before, the variables $\boldsymbol{R}(k)$ and $\boldsymbol{P}(k)$ can be loosely interpreted as information and covariance matrices, respectively (see explanations in Section 2). Formulas (32)-(33) are an approximate version of the recursive LA estimator.

Summarizing the discussion, an approximate recursive LA procedure (avoiding the problem of small divisors) can be presented in the following compact format, where the innovation covariance matrix $\boldsymbol{P}(k)$ and the 'a priori' prediction error $\varepsilon(k)$ are used to refresh the estimate of the vector $\boldsymbol{\theta}$ [18]:

$$\varepsilon(k)=\chi(k)-\boldsymbol{\varphi}^{\mathrm{T}}(k)\widehat{\boldsymbol{\theta}}(k-1) \tag{34}$$

$$\boldsymbol{P}(k)=\frac{1}{\lambda}\left[\boldsymbol{P}(k-1)-\frac{\boldsymbol{P}(k-1)\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{P}(k-1)}{\lambda|\varepsilon(k)|+\boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{P}(k-1)\boldsymbol{\varphi}(k)}\right] \tag{35}$$

$$\widehat{\boldsymbol{\theta}}(k)=\widehat{\boldsymbol{\theta}}(k-1)+\boldsymbol{P}(k)\boldsymbol{\varphi}(k)\mathrm{sign}(\varepsilon(k)) \tag{36}$$

This approximate recursive LA procedure coincides with the classical IV scheme (15)–(17), provided that the instrument $\boldsymbol{\eta}(k)$ is adopted as the regression vector scaled with the 'a priori' measure (34) of the prediction error, i.e. $\boldsymbol{\eta}(k)=\boldsymbol{\varphi}(k)/\varepsilon(k)$.

By performing the similar reasoning as in the LS case [14], and taking into account that $e(k)/|e(k)|=\mathrm{sign}[e(k)]$, we conclude that in the (unweighted) LA case the estimate $\widehat{\boldsymbol{\theta}}(k)$ converges asymptotically to $\boldsymbol{\theta}$, provided that $E[\boldsymbol{\varphi}(k)\mathrm{sign}(e(k))]=\boldsymbol{0}$. This issue, relating to the basic LA method, was discussed by the authors in [18].

Assuming, as before, that the residual process $e(k)$ is white noise, we conclude that the process 'sign$(e(k))$' has also white noise properties (this is because deterministic functions of independent random variables remain independent random variables). For this reason, the correlation $E[\boldsymbol{\varphi}(k)\mathrm{sign}(e(k))]$ of the signals $\boldsymbol{\varphi}(k)$ and sign$[e(k)]$ is zero, and therefore the LA estimate $\widehat{\boldsymbol{\theta}}(k)$ is asymptotically consistent ($k\to\infty$).

Because in most situations, the equation error $e(k)$ turns out to be a correlated process, so here it is also worth suggesting the approach of an instrumental variable (IV) as a good solution to the bias problem. Importantly, the methods of generating such an instrumental variable vector may be the same for both estimation schemes, i.e. LS and LA [15].

In contrast to LS, initialization of (34)-(36) based on a high-diagonal matrix $\boldsymbol{P}(k)$ cannot be recommended here. It is more rational to start the estimation with the LS procedure (15)-(17) in several steps, and then based on the obtained reliable LS estimate of $\boldsymbol{\theta}$, the matrix $\boldsymbol{P}(k)$ and the prediction error $|\widehat{e}(k)|$, one can proceed to LA processing according to (34)-(36).

4. Smoothly approximated LA algorithms

We now wish to propose another solution to the troublesome problem of analytic differentiation of the basic functional LA (18)

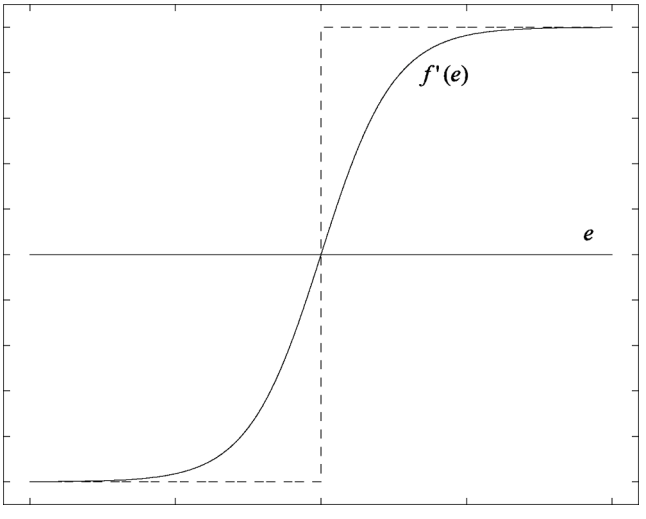


Fig. 1. Smooth approximation (39) of the sign function.

(starting with batch iterative).

The discussed batch-iterative method leading to the estimator of the smallest sum of absolute deviations in the innovation version (24)-(26) for the system (1)-(3) has the already mentioned disadvantages consisting of small scaling factors and the (less annoying) problem of flat minimum of the LA functional (18). These problems can also be solved in another way by using a 'sigmoid' function to reliably approximate the 'sign' function present in the gradient (19).

For this purpose, the previous quality functional (18) should be redefined in such a way that its derivative takes the desired sigmoid shape, which is represented, for example, by the analytically useful hyperbolic tangent shown in Fig. 1. After such modification, the new smooth and differentiable objective function takes the form

$$\widetilde{J}(\boldsymbol{\theta}) = \sum_{l=1}^{k} \gamma(l) f(e(l)) = \sum_{l=1}^{k} \lambda^{k-l} f\left(\chi(l) - \boldsymbol{\varphi}^{\mathrm{T}}(l)\boldsymbol{\theta}\right) \tag{37}$$

based on a smooth cost profile function $f(e)$, which is a smooth approximation of the absolute value (Fig. 2)

$$f(e) = \alpha^{-1}\ln(\cosh(\alpha e)) \tag{38}$$

Note that in the index (37) we have the weighting tool $\gamma(l)$ and in (38) we have a tuning factor α.

The applied smooth LA (SLA) cost profile $f(e)$, built on the hyperbolic cosine and logarithmic functions, is a sensible approximation to the absolute value ($f(e) \approx |e|$), as long as the tuning factor α is large enough ($\alpha \to \infty$).

It is easy to check that the profile $f(e)$ is locally quadratic: $fe) \approx 0.5\alpha e^2$ (for $e \approx 0$). Fig. 2 also shows that $f(e) < |e|$ (for $e \neq 0$). Also function $f(e)$ has (for large $|e|$) has skew asymptotes, namely: $f(e) \approx |e| - \alpha^{-1}\ln(2)$.

For our considerations, the forms of successive derivatives of $f(e)$ are also important:

$$f'(e) = \frac{df}{de} = \tanh(\alpha e) \tag{39}$$

$$f''(e) = \frac{d^2 f}{de^2} = \frac{\alpha}{\cosh^2(\alpha e)} \tag{40}$$

which are an odd function (strictly increasing) and an even function (greater than zero), respectively.

The shape of the derivative (39), representing the hyperbolic tangent (Fig. 1), explains our motivation for choosing the form of the profile function $f(e)$ (Fig. 2) used in the modified index (37). Here you can also see that the sigmoid curve approximates the sign function very well, as long as α is large enough.

According to the reasoning in Section 3, the index (37) can be approximately minimized, provided that estimates of the prediction error $e(l)$ are available.

Replicating the method of determining the gradient (19) and using the error shaping factor $\mu(e)$ relating e with the derivative (39) of the profile function $f(e)$:

$$\mu(e) = \frac{e}{f'(e)} = \frac{e}{\tanh(\alpha e)} \tag{41}$$

we can (equally) use $e/\mu(e) = f'(e)$ to express the gradient of (37)

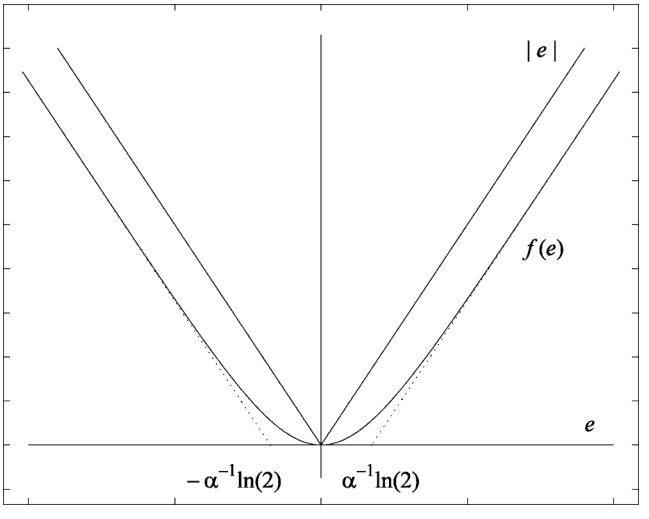


Fig. 2. Smooth approximation (38) of the absolute value.

$$\nabla_{\theta}\widetilde{J} = -\sum_{l=1}^{k}\lambda^{k-l}\boldsymbol{\varphi}(l)f^{'}(e) = -\sum_{l=1}^{k}\lambda^{k-l}\boldsymbol{\varphi}(l)\frac{e(l)}{\mu(e(l))} \approx -\sum_{l=1}^{k}\lambda^{k-l}\frac{\boldsymbol{\varphi}(l)[\chi(l)-\boldsymbol{\varphi}^{\mathrm{T}}(l)\boldsymbol{\theta}]}{\mu(\widehat{e}(l))} = \mathbf{0} \tag{42}$$

As before, in the denominator (42) there is an estimate $\widehat{e}(l)$ of $e(l)$. However, compared to (21), here we finally get an SLA estimate of $\boldsymbol{\theta}$, where the tool $\mu(e)$ takes over the role previously played by the absolute value:

$$\widehat{\boldsymbol{\theta}} = \left[\sum_{l=1}^{k}\lambda^{k-l}\frac{\boldsymbol{\varphi}(l)}{\mu(\widehat{e}(l))}\boldsymbol{\varphi}^{\mathrm{T}}(l)\right]^{-1}\left[\sum_{l=1}^{k}\lambda^{k-l}\frac{\boldsymbol{\varphi}(l)}{\mu(\widehat{e}(l))}\chi(l)\right] \tag{43}$$

Processing (43) can be again refined by iterations. With the appropriate LS estimate $\widehat{\boldsymbol{\theta}}^{[0]}$ of (6) initiating these iterations, the successively updated estimates $\widehat{\boldsymbol{\theta}}^{[r]}$ allow the sequence of prediction errors to be updated (for $l = 1\cdots k$). This leads to an iterative batch procedure of SLA similar to (22)-(23):

$$\widehat{e}^{[r]}(l) = \chi(l) - \boldsymbol{\varphi}^{\mathrm{T}}(l)\widehat{\boldsymbol{\theta}}^{[r]} \tag{44}$$

$$\widehat{\boldsymbol{\theta}}^{[r+1]} = \left[\sum_{l=1}^{k}\lambda^{k-l}\frac{\boldsymbol{\varphi}(l)\boldsymbol{\varphi}^{\mathrm{T}}(l)}{\mu(\widehat{e}^{[r]}(l))}\right]^{-1}\left[\sum_{l=1}^{k}\lambda^{k-l}\frac{\boldsymbol{\varphi}(l)\chi(l)}{\mu(\widehat{e}^{[r]}(l))}\right] \tag{45}$$

The innovation form of SLA is obtained by substituting the output $\chi(l) = \boldsymbol{\varphi}^{\mathrm{T}}(l)\widehat{\boldsymbol{\theta}}^{[r]} + \widehat{e}^{[r]}(l)$ from (44) to (45):

$$\widehat{\boldsymbol{\theta}}^{[r+1]} = \widehat{\boldsymbol{\theta}}^{[r]} + [\boldsymbol{H}(k)]^{-1}\boldsymbol{\psi}(k) \tag{46}$$

$$\boldsymbol{H}(k) = \sum_{l=1}^{k}\lambda^{k-l}\frac{\boldsymbol{\varphi}(l)\boldsymbol{\varphi}^{\mathrm{T}}(l)}{\mu(\widehat{e}^{[r]}(l))} \tag{47}$$

$$\boldsymbol{\psi}(k) = \sum_{l=1}^{k}\lambda^{k-l}\frac{\boldsymbol{\varphi}(l)\widehat{e}^{[r]}(l)}{\mu(\widehat{e}^{[r]}(l))} = \sum_{l=1}^{k}\lambda^{k-l}\boldsymbol{\varphi}(l)\tanh\left(\widehat{e}^{[r]}(l)\right) \tag{48}$$

where $'-\boldsymbol{\psi}(k)'$ is the gradient and $\boldsymbol{H}(k)$ is the Hessian of the minimized index (37).

Processing (44)-(45) or (46)-(48) ends with the relative change test (27), which step by step analyzes the change in criterion (37) whether it falls below the given threshold Δ_{min}. This solves the problem of possible zero divisors, since the factor (41) is always positive: $\mu(\widehat{e}^{[r]}(l)) > 0$.

In addition, the calculation of the shaping factor (41) itself is numerically well-conditioned (no division by zero), which can be demonstrated using two rapidly converging power series for the components of the hyperbolic 'sinh' and 'cosh' functions, used to construct the 'tanh' function in $\mu(e)$ (see also Fig. 3):

$$\mu(e) = \frac{e}{\tanh(\alpha e)} = \frac{1}{\alpha}\frac{\cosh(\alpha e)}{\sinh(\alpha e)}(\alpha e) = \frac{1}{\alpha}\frac{1+\frac{(\alpha e)^2}{2!}+\frac{(\alpha e)^4}{4!}+\frac{(\alpha e)^6}{6!}+\cdots}{1+\frac{(\alpha e)^2}{3!}+\frac{(\alpha e)^4}{5!}+\frac{(\alpha e)^6}{7!}+\cdots} \tag{49}$$

The redefined SLA criterion (37) based on $f(e)$ resembles the standard LA shape (18), but (37) is square in the small neighborhood

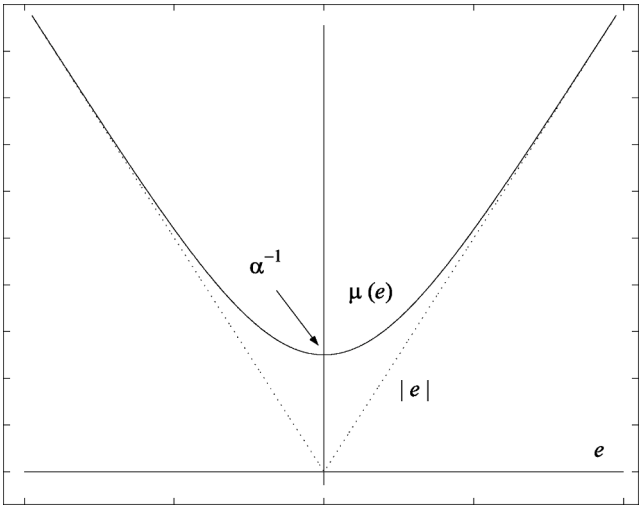


Fig. 3. Smooth error shaping function μ(e).

of each kink point (now smoothed). The strictly convex SLA index (37) also overcomes the 'flat minimum' problem in index (18).

An approximate recursive implementation of the smooth SLA scheme can be obtained by the method shown in Section 3. Thus, introducing the new symbols $\boldsymbol{R}(k)$ and $s(k)$ for the expressions given in (43), we get

$$\boldsymbol{R}(k) = \sum_{l=1}^{k} \lambda^{k-l} \frac{\boldsymbol{\varphi}(l)}{\mu(\widehat{e}(l))} \boldsymbol{\varphi}^{\mathrm{T}}(l) \tag{50}$$

$$\boldsymbol{R}(k) = \lambda \boldsymbol{R}(k-1) + \frac{\boldsymbol{\varphi}(k)}{\mu(\widehat{e}(k))} \boldsymbol{\varphi}^{\mathrm{T}}(k) \tag{51}$$

$$\boldsymbol{s}(k) = \sum_{l=1}^{k} \lambda^{k-l} \frac{\boldsymbol{\varphi}(l)}{\mu(\widehat{e}(l))} \chi(l) \tag{52}$$

$$\boldsymbol{s}(k) = \lambda \boldsymbol{s}(k-1) + \frac{\boldsymbol{\varphi}(k)}{\mu(\widehat{e}(k))} \chi(k) \tag{53}$$

The MIL formula (8) can be applied to invert (51):

$$\boldsymbol{P}(k) = \boldsymbol{R}^{-1}(k) = \left[\lambda \boldsymbol{R}(k-1) + \frac{\boldsymbol{\varphi}(k)}{\mu(\widehat{e}(k))} \boldsymbol{\varphi}^{\mathrm{T}}(k) \right]^{-1} \approx \frac{1}{\lambda} \left[\boldsymbol{P}(k-1) - \frac{\boldsymbol{P}(k-1)\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{P}(k-1)}{\lambda \mu(\varepsilon(k)) + \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{P}(k-1)\boldsymbol{\varphi}(k)} \right] \tag{54}$$

As a result, the estimate (43) can be shown as

$$\begin{aligned} \widehat{\boldsymbol{\theta}}(k) &= \boldsymbol{R}^{-1}(k)\boldsymbol{s}(k) = \boldsymbol{P}(k)\boldsymbol{s}(k) = \frac{1}{\lambda} \left[\boldsymbol{P}(k-1) - \frac{\boldsymbol{P}(k-1)\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{P}(k-1)}{\lambda \mu(\widehat{e}(k)) + \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{P}(k-1)\boldsymbol{\varphi}(k)} \right] \left[\lambda \boldsymbol{s}(k-1) + \frac{\boldsymbol{\varphi}(k)}{\mu(\widehat{e}(k))} \chi(k) \right] \\ &\approx \widehat{\boldsymbol{\theta}}(k-1) + \boldsymbol{P}(k)\boldsymbol{\varphi}(k) \frac{\chi(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\widehat{\boldsymbol{\theta}}(k-1)}{\mu(\chi(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\widehat{\boldsymbol{\theta}}(k-1))} \end{aligned} \tag{55}$$

In the above argument, the 'a priori' prediction error $\varepsilon(k) = \chi(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\widehat{\boldsymbol{\theta}}(k-1)$ was consistently used as an estimate of the equation error ($|\varepsilon(k)| \approx |\widehat{e}(k)|$).

Finally, the smooth (approximate) SLA recursive estimator can be presented in the standard form, including the determination of the 'a priori' prediction error $\varepsilon(k)$, the innovation of the covariance matrix $\boldsymbol{P}(k)$, and the final update of the estimation vector:

$$\varepsilon(k) = \chi(k) - \boldsymbol{\varphi}^{\mathrm{T}}(k)\widehat{\boldsymbol{\theta}}(k-1) \tag{56}$$

$$\boldsymbol{P}(k) = \frac{1}{\lambda} \left[\boldsymbol{P}(k-1) - \frac{\boldsymbol{P}(k-1)\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{P}(k-1)}{\lambda \mu(\varepsilon(k)) + \boldsymbol{\varphi}^{\mathrm{T}}(k)\boldsymbol{P}(k-1)\boldsymbol{\varphi}(k)} \right] \tag{57}$$

$$\widehat{\boldsymbol{\theta}}(k) = \widehat{\boldsymbol{\theta}}(k-1) + \boldsymbol{P}(k)\boldsymbol{\varphi}(k) \frac{\varepsilon(k)}{\mu(\varepsilon(k))} \tag{58}$$

The above (56)-(58) coincides with the IV scheme (15)–(17), where the auxiliary variable is $\boldsymbol{\eta}(k) = \boldsymbol{\varphi}(k)/\mu(\varepsilon(k))$ and not $\boldsymbol{\eta}(k) = \boldsymbol{\varphi}(k)/\varepsilon(k)$.

Principles of practical initialization of algorithm (56)-(58) are described in Section 3.

In conclusion, it is worth mentioning here the basic advantages of smooth execution of the LA strategy. First, in the iterative scheme 46–48 we overcome the problem of divisors close to zero ($|\widehat{e}^{[r]}(l)| \approx 0$) at no cost, because the applied error shaping factor $\mu(e)$ always remains positive (since $\mu(0) = \alpha^{-1}$). Second, the smooth LA functional represented by the sum of strictly convex profile functions $f(e)$ is also strictly convex, which guarantees that (37) has a unique minimum. Third, the calculation of $\mu(e) = e/\tanh(\alpha e)$ based on the power series expansions (49) is numerically safe (i.e. does not require division by small numbers).

It is worth adding that the primary iterative procedure LA (24)-(26) has already been shown to minimize the original absolute deviation quality factor (Appendix A). Unfortunately, an analogous rigorous proof for the iterative SLA algorithm (46)-(48) cannot yet be provided. On the other hand, the iterative implementation of the SLA algorithm (46)-(48) with a sufficiently large tuning parameter α is actually numerically 'very close' to the primary LA implementation (see Figs. 2 and 3). Hence, one can derive a practically justified belief that the smooth LA method also converges.

5. Alternative approaches to LA estimation

In practical situations, linear programming methods (simplex) are usually the first choice for optimizing an 'analytically difficult' functional [20]. Still, adapting such procedures to support the non-linear index (18) requires the implementation of additional variables $v(l)$, such that $|y(l) - \boldsymbol{\varphi}^{\mathrm{T}}(l)\boldsymbol{\theta}| \le v(l)$. $J(\boldsymbol{\theta}) = v(1) + \cdots + v(k)$ is apt for simplex minimization under the linear constraints $-v(l) + \boldsymbol{\varphi}^{\mathrm{T}}(l)\boldsymbol{\theta} \le y(l)$ and $-v(l) - \boldsymbol{\varphi}^{\mathrm{T}}(l)\boldsymbol{\theta} \le -y(l)$.

This approach is rather impractical for several reasons. First, a large number of measurements ($k = 100$, see Section 4, part A)

involves the use of (k) new variables and at least ($2 \times k$) constraints. Thus, even when we identify only a few parameters (6 or 4, see Section 6), the appearance of many additional variables radically increases the computational effort.

Secondly, in the worst case of the exponential complexity of the simplex scheme, the data processing time can be really long (up to 30 times longer compared to the execution time of the LA procedure). As a result, it is unrealistic to attempt to run the simplex routine on-line (as can be the case with LA and SLA).

With this in mind, we attempt an instructive comparison of the smoothed LA procedure with the Gauss-Newton (gradient descent) algorithm adapted to minimize the smooth LA criterion. By comparing the important properties of the considered estimators (e.g. "information content" expressed in the Hessian matrices), we show the relevance of the SLA approach. Besides, we show the Huber's loss function as an alternative to our smooth approximation (38).

Therefore, against the background of the disadvantages of the classical concept, the proposed SLA estimation method shows its strong advantages.

5.1. Gauss-Newton optimization

From a theoretical point of view, it is instructive that the improved/smooth iterative method LA (46)-(48) can be related to the known Gauss-Newton (GN) optimization algorithm. This classical steepest descent scheme, which uses the gradient $\nabla_{\boldsymbol{\theta}}$ and Hessian $\nabla^2_{\boldsymbol{\theta}}$ of the fundamental criterion function $\Im(\boldsymbol{\theta})$, is given by (with $r = 0,\ 1, \cdots$)

$$\widehat{\boldsymbol{\theta}}^{[r+1]} = \widehat{\boldsymbol{\theta}}^{[r]} + \left\{ \left[\nabla^2_{\boldsymbol{\theta}} \Im(\boldsymbol{\theta}) \right]^{-1} \left[-\nabla_{\boldsymbol{\theta}} \Im(\boldsymbol{\theta}) \right] \right\} \Big|_{\boldsymbol{\theta}=\widehat{\boldsymbol{\theta}}^{[r]}} \tag{59}$$

Theoretically, the criterion $\Im(\boldsymbol{\theta})$ present in (59) can take quite any form, such as those used above: $I(\boldsymbol{\theta})$, $J(\boldsymbol{\theta})$ or $\widetilde{J}(\boldsymbol{\theta})$. Of course, applying the GN method to the square index (4) is pointless, because by definition formula (59) simply coincides with the exact LS solution (6) after one iteration. On the other hand, in the case of the basic criterion LA (18), the direct application of (59) is unrealistic, because it is impossible to determine the derivative of the sign function (as we obtain an impermissible Dirac function in the Hessian).

Fortunately, the gradient and Hessian can be determined analytically for the smooth criterion LA (37). The resulting GN formula with $\Im(\boldsymbol{\theta}) = \widetilde{J}(\boldsymbol{\theta})$ is

$$\widehat{\boldsymbol{\theta}}^{[r+1]} = \widehat{\boldsymbol{\theta}}^{[r]} + \left[\boldsymbol{H}_{\mathrm{GN}}(k) \right]^{-1} \boldsymbol{\psi}(k) \tag{60}$$

On the other hand, the iterative procedure of SLA (46)-(48) with shaping factor (49) can be shown as

$$\widehat{\boldsymbol{\theta}}^{[r+1]} = \widehat{\boldsymbol{\theta}}^{[r]} + \left[\boldsymbol{H}_{\mathrm{SLA}}(k) \right]^{-1} \boldsymbol{\psi}(k) \tag{61}$$

Using (39)-(40), the Hessian matrices can be given as

$$\boldsymbol{H}_{\mathrm{GN}}(k) = \sum\nolimits_{l=1}^{k} \lambda^{k-l} \boldsymbol{\varphi}(l) \boldsymbol{\varphi}^{\mathrm{T}}(l) \frac{\alpha}{\cosh^2(\alpha \widehat{e}_r(l))} \tag{62}$$

$$\boldsymbol{H}_{\mathrm{SLA}}(k) = \sum\nolimits_{l=1}^{k} \lambda^{k-l} \boldsymbol{\varphi}(l) \boldsymbol{\varphi}^{\mathrm{T}}(l) \frac{\tanh\left(\alpha \widehat{e}^{[r]}(l)\right)}{\widehat{e}^{[r]}(l)} \tag{63}$$

while the gradient component $'-\boldsymbol{\psi}(k)'$ is identical in both formulas (60) and (61) and

$$\boldsymbol{\psi}(k) = \sum\nolimits_{l=1}^{k} \lambda^{k-l} \boldsymbol{\varphi}(l)\ \tanh\left(\alpha \widehat{e}^{[r]}(l)\right) \tag{64}$$

Note that both the GN and SLA algorithms have the same search direction (64), while the gain terms (i.e. inverted Hessian matrices) are apparently different. It is easy to check that for errors close to zero ($e \approx 0$) quantities (62) and (63) are practically equivalent: $\tanh(\alpha e)/e \approx \alpha/\cosh^2(\alpha e) \approx \alpha$. As a result, for small errors the Hessian matrices are also equal ($\boldsymbol{H}_{\mathrm{GN}}(k) \approx \boldsymbol{H}_{\mathrm{SLA}}(k)$), so both solutions (60) and (61) converge.

However, there are some indications that the SLA scheme is superior to GN. This conclusion can be reached by comparing the square forms of the corresponding Hessian matrices. Since Hessian has a sense of information matrix (see the explanatory discussion in Section 2), we can evaluate these solutions in terms of related 'information content'. Equivalently, by comparing the inverse matrices (interpreted as covariance matrices $\boldsymbol{F}(k) = \boldsymbol{H}^{-1}(k)$), we can confront the corresponding dispersion of estimates.

For this purpose, using (62) and (63), the corresponding quadratic forms (with a compatible vector $\|\boldsymbol{v}\| \neq 0$) can be represented as

$$\boldsymbol{v}^{\mathrm{T}} \left[\boldsymbol{H}_{\mathrm{GN}}(k) \right] \boldsymbol{v} = \sum\nolimits_{l=1}^{k} \lambda^{k-l} \left[\boldsymbol{v}^{\mathrm{T}} \boldsymbol{\varphi}(l) \right]^2 \frac{\alpha}{\cosh^2(\alpha \widehat{e}_r(l))} \tag{65}$$

$$\boldsymbol{v}^{\mathrm{T}} \left[\boldsymbol{H}_{\mathrm{SLA}}(k) \right] \boldsymbol{v} = \sum\nolimits_{l=1}^{k} \lambda^{k-l} \left[\boldsymbol{v}^{\mathrm{T}} \boldsymbol{\varphi}(l) \right]^2 \frac{\tanh\left(\alpha \widehat{e}^{[r]}(l)\right)}{\widehat{e}^{[r]}(l)} \tag{66}$$

Based on the following easily verifiable relationships

$$\begin{cases} \dfrac{\tanh(\alpha e)}{e} > \dfrac{\alpha}{\cosh^2(\alpha e)} & \text{for} \quad e \neq 0 \\ \dfrac{\tanh(\alpha e)}{e} \approx \dfrac{\alpha}{\cosh^2(\alpha e)} \approx \alpha & \text{for} \quad e \approx 0 \end{cases} \tag{67}$$

we conclude that in the sense of positive definiteness of the appropriate information matrices $\boldsymbol{H}(k)$ and the covariance matrices $\boldsymbol{F}(k) = \boldsymbol{H}^{-1}(k)$ the following occurs:

$$\boldsymbol{H}_{\mathrm{SLA}}(k) \geq \boldsymbol{H}_{\mathrm{GN}}(k) \tag{68}$$

$$\boldsymbol{F}_{\mathrm{SLA}}(k) \leq \boldsymbol{F}_{\mathrm{GN}}(k) \tag{69}$$

The above clearly proves the superiority of the SLA procedure (61) over the GN version (60).

Notably, for large α, the SLA cost (37) can match LA (18) well enough, so the convergence theorem (Appendix A) is likely to hold also for the smoothed LA estimator (46)-(48). It should also be noted that there is no such guarantee for the GN approach, where convergence is only ensured provided that the initial value of $\boldsymbol{\theta}$ is 'close enough' to the actual minimum.

5.2. Application of Huber's loss function

In an alternative concept of adapting the gradient routine to minimize the piecewise linear criterion (18), the Huber loss function (HL) can be applied [21]. Then such a redefined LA indicator can be presented as

$$\Im(\boldsymbol{\theta}) = \sum\nolimits_{l=1}^{k} \gamma(l)\hbar(e(l)) = \sum\nolimits_{l=1}^{k} \lambda^{k-l}\hbar\big(y(l) - \boldsymbol{\varphi}^{\mathrm{T}}(l)\boldsymbol{\theta}\big) \tag{70}$$

where the cost profile function takes the Huber form

$$\hbar(e) = \begin{cases} 0.5e^2 & \text{for} \quad |e| \leq \rho \\ \rho\,(\,|e| - 0.5\rho\,) & \text{for} \quad |e| > \rho \end{cases} \tag{71}$$

On the one hand, the above Huber loss function (locally quadratic in the ρ-neighborhood of the minimum of (71)) eliminates the non-differentiable kink points of the modified criterion (70).

On the other hand, as this function is linear outside the user-defined "shaping interval" $[\,-\rho\ ,\ \rho]$, criterion (70) must remain weakly convex. Thus, unlike with the unimodal SLA functional (37), the Huber-based index (70) may result in the so-called 'flat minima'.

Moreover, minimizing the absolute deviation index based on the Huber approximation is inconvenient. Strictly speaking, the Gauss-Newton scheme based on the Huber criterion (70) leads to

$$\widehat{\boldsymbol{\theta}}^{[r+1]} = \widehat{\boldsymbol{\theta}}^{[r+1]} + [\boldsymbol{H}_{\mathrm{HL}}(k)]^{-1}\varsigma(k) \tag{72}$$

$$\boldsymbol{H}_{\mathrm{HL}}(k) = \sum\nolimits_{l=1}^{k} \lambda^{k-l}\boldsymbol{\varphi}(l)\boldsymbol{\varphi}^{\mathrm{T}}(l)\hbar^{''}\big(e^{[r]}(l)\big) \tag{73}$$

$$\varsigma(k) = \sum\nolimits_{l=1}^{k} \lambda^{k-l}\boldsymbol{\varphi}(l)\hbar^{'}\big(\widehat{e}^{[r]}(l)\big) \tag{74}$$

Unfortunately, the derivative of the spline function (71) has undesirable, non-differentiable kink points at the ends of the shaping interval $[\,-\rho\ ,\ \rho]$ and in the middle of this interval it has $\hbar^{'}(e) = e$ (with the tangent equal to one), and some second derivative $\hbar^{''}(e)$ of the function (71), while in any reliable approximation of the sign function this tangent (slope) should be as large as possible (Fig. 1).

To overcome this issue, paradoxically, we should use a specific smooth and classically differentiable function that is another approximation of the 'problematically smooth' Huber loss profile. Bearing in mind this clear weakness of the implementation of the Huber estimator, one can appreciate the appropriateness of (39) used in the SLA method (46)-(48) discussed here.

As mentioned, the simplex method has exponential complexity. Batch methods (LS, LA, SLA, GN) are characterized by complexity $O(n^3)$, which results from the applied matrix inversion. In turn, the recursive algorithms (LS, LA and SLA) have complexity $O(n^2)$.

In the next section, the iterative and recursive LA methods discussed will be shown in action.

5.3. Solutions based on machine learning

Of the many effective algorithms often considered when solving optimization problems (such as estimating system parameters), the expectation minimization (EM) procedure is very effective when identifying systems based on an incomplete measurement set (with missing data). However, just like simplex and Gauss-Newton schemes, this procedure is also only off-line. In addition, the rate of convergence is very low, especially for large data sets. In contrast, our LA/SLA procedures can be implemented in a recursive (approximate) format on-line. These procedures also have the unique feature of high insensitivity to destructive measurement errors (which can be observed both in the case of 'isolated' sporadic outliers and the entire sequence of such measuring errors).

In addition, possible implementation of the EM algorithm for processing data contaminated with outliers is quite problematic,

because it would require detection, isolation and elimination of all erroneous measurements from the data set before running the EM procedure. Therefore, the EM solution in the case of identification including outliers does not constitute any special competition or counterweight to the approach (S)LA.

The Alternating Direction Multiplier Method (ADMM) is a machine learning procedure (often considered a more efficient version of the stochastic gradient descent algorithm). On the one hand, ADMM is fast and easy to implement. On the other hand, the method is not always convergent. Whereas, the LA-iterative procedure guarantees convergence while minimizing the underlying quality indicator with the nature of 'absolute deviation' (the proof is in the appendix).

Undoubtedly, ADMM is practical, effective and worth your attention. However, we focus our research on analytical methods, for which you can mathematically demonstrate the convergence or asymptotic behavior. We value modern optimization methods and use them in the evolutionary optimization of multi-criteria in multidimensional parameters and when designing control and diagnostic systems. However, in this work we want to maintain the concise and consistency of the theoretical and analytical content provided, without mixing various areas (i.e. analytical approaches and machine learning techniques).

6. Simulation experiments

One of the basic advantages of the LA method is its insensitivity to large measurement errors, called outliers. Thus, in this section we intend to demonstrate this unique property of LA and its methodological advantages over the classical LS approach. In the literature in this field, you can find practical examples demonstrating the claimed properties of the basic LA procedures [13,15]). The numerical tests included below illustrate the use of smoothed LA estimators to identify various regression models in the presence of destructive errors in processed I/O data (in supervisory or automation systems). The iterative LA algorithm was used for polynomial approximation of measurement data, while tracking the parameters of a non-stationary system was performed with the recursive LA method.

6.1. Polynomial signal reconstruction

Among the various tasks considered in industrial diagnostics, polynomial reconstruction of measurement signals based on sometimes unreliable records of sampled data is used. Determining the Newton-Lagrange interpolation polynomial seems like a simple solution, but it can end in failure when processing infected measurement data. Moreover, the resulting polynomial may oscillate unacceptably between the interpolation nodes, which in practice disqualifies this type of data reconstruction. The polynomial degree equal to the number of nodes (minus 1) is also impractical. Therefore, a low-degree polynomial approximation of measurements is usually more useful.

Considering the polynomial of degree n

$$\chi(t) = c_0 + c_1 t + c_2 t^2 + \cdots + c_n t^n \tag{75}$$

at discrete moments of time ($t = lT$), a classical regression model [13] can be used as

$$\chi(l) = \boldsymbol{\varphi}^{\mathrm{T}}(l)\boldsymbol{\theta} + e(l) \tag{76}$$

$$\boldsymbol{\varphi}(l) = \left[1 \;\; lT \;\; (lT)^2 \;\; \cdots \;\; (lT)^n\right]^{\mathrm{T}} \tag{77}$$

$$\boldsymbol{\theta} = \left[c_0 \;\; c_1 \;\; c_2 \;\; \cdots \;\; c_n\right]^{\mathrm{T}} \tag{78}$$

with the regression vector $\boldsymbol{\varphi}(l)$ containing deterministic quantities (successive powers of the discrete-time variable (lT) and the vector $\boldsymbol{\theta}$ holding the coefficients of polynomial (75).

In the test, we reconstructed the polynomial (75) of the fifth degree ($n = 5$), using the following continuous signal as the primary data source

$$y(t) = \Omega \exp(-\zeta t)\sin(\omega t) \tag{79}$$

parameterized with $\Omega = 15$, $\zeta = 1.5$ and $\omega = 6.4$ rad/s.

With a sampling period of $T = 0.01$ s, a sequence of one hundred discrete measurements was recorded: $y(l) = y(t)_{|t=lT}$, $l = 1 \cdots k$ ($k = 100$). To make the test realistic, the quantization effect of the AD converter was simulated using additive white noise $w(l)$ with a

Table 1
Estimating the Parameters of an Approximating Polynomial using four Methods.

	LS	SLA	LA	simplex
c_0	–0.5753	–0.7214	–0.7347	–0.7317
c_1	151.4144	123.8066	123.9808	123.8682
c_2	–844.3762	–411.6575	–413.9981	–412.9612
c_3	1805.7320	308.3843	316.6720	313.5116
c_4	–1767.9299	115.5361	104.9884	108.7346
c_5	658.1648	–135.1215	–130.6854	–132.2009

uniform distribution and variance $\sigma_w^2 = 10^{-4}$, which results in

$$\chi(l) = y(l) + w(l) \tag{80}$$

It was assumed that large errors in the processed signal can be imitated by the phenomenon of loss of measurement data. Thus, the correct values of $y(l)$, obtained at sampling moments $l = 25\cdots34$ and $l = 57\cdots66$, were replaced with zeros, which is equal to the loss of analog data in two time intervals of 0.1 s.

The batch estimate of $\boldsymbol{\theta}$ was obtained using the LS algorithm (6) and the iterative SLA procedure (46)-(48) with $\alpha = 20$ and $\Delta_{min} = 10^{-3}$. In this stationary case, exponential forgetting was disabled in both algorithms ($\lambda = 1$).

The estimated coefficients (78) are given in Table 1. The LS and SLA reconstructions of the signal (79) observed according to the polynomial (75) are shown in Fig. 4. The stopping condition (27) of the SLA procedure worked after 13 iterations.

In the case under consideration, it is easy to provide mechanical motivation for the proposed polynomial approximation of an oscillating signal (79). Such a case can be used for reliable diagnostics of the vehicle at general car diagnostic stations to check the quality of mechanical suspension. In a typical diagnostic procedure, the tested wheel is first subjected to vibrations. Then the vibration platform on which the wheel is placed is blocked, the resulting exponentially-decaying oscillation signal (representing the free response of the suspension) is recorded and evaluated for the efficiency of the suspension components (i.e. spring and shock absorber – Fig. 5). Once a polynomial model (75) approximating the oscillatory signal has been identified, the actual 'suspension parameters' ω, ζ, and Ω can be reliably reproduced from the appropriate analytical formulas. Namely, the zero crossing points ($t_1, \chi(t_1) = 0$; $t_2, \chi(t_2) = 0$) and the extreme values (t_{min}, $\chi(t_{min})$; t_{max}, $\chi(t_{max})$) can be established by numerically solving the algebraic equations: $\chi(t) = c_0 + c_1 t + c_2 t^2 + c_3 t^3 + c_4 t^4 + c_5 t^5 = 0$ and $d\chi(t)/dt = c_1 + 2c_2 t + 3c_3 t^2 + 4c_4 t^3 + 5c_5 t^4 = 0$. Hence, the desired 'suspension parameters' can be determined as: $\omega = \pi/|t_2 - t_1|$, $\Gamma = \ln[\sin(\omega t_{min})/\sin(\omega t_{max})]$, $\aleph = \ln[\chi(t_{min})/\chi(t_{max})]$, $\zeta = (\Gamma - \aleph)/(t_{min} - t_{max})$ and $\Omega = \chi(t_{max})/[\exp(-\zeta t_{max})\sin(\omega t_{max})]$.

The obtained suspension parameters (TABLE 2) can be used to obtain the target values: Hooke's spring coefficient (K) of the spring and Stokes' drag/resistance coefficient (B) of the shock absorber (Fig. 5).

Given the effectiveness of this approach, direct assessment of diagnostic parameters (ω, ζ, Ω) from raw sample data may prove surprisingly ineffective. This is expected because trying to measure the period of damped oscillations ω based on the detection of zero crossings is rather unreliable, especially when the sampled data is contaminated with additive noise ('environmental' disturbances or unavoidable quantization noise). The assessment of the remaining constants (ζ, Ω) will also be a big problem if we do not use a reliable model of the recorded process $y(t)$.

Other interesting examples, such as fault-insensitive dynamic weighing of trucks [13] and practical diagnostics of vehicle mechanical parts [15] are available in the literature.

The test realistically illustrates the impact of large measurement errors on the accuracy of parameter estimation. As expected, the LS estimator is very sensitive to simulated large errors, while the SLA algorithm is extremely insensitive to unwanted events such as measurement outliers or data loss.

Obviously, since the LS method fails to identify the system in the presence of destructive faults, we implemented the original (not smoothed) LA method (24)-(26) and the simplex procedure (Section 5) to solve the signal reconstruction problem. As we expected, the SLA, LA, and simplex scores were actually very similar (Table 1). This is because for the applied tuning factor $\alpha = 20$, which is quite a large value of this 'smoothing parameter', the smooth approximation (41) of the error shaping factor $\mu(e)$ (Fig. 3) used in the SLA

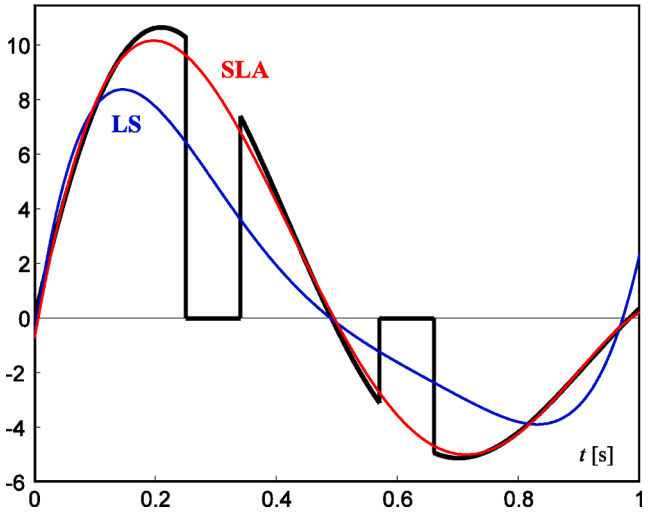


Fig. 4. Measurement data signal (thick line) with data corruption/loss (within vertical lines) and its off-line/batch LS and SLA reconstructions shown as solid blue and red lines, respectively.

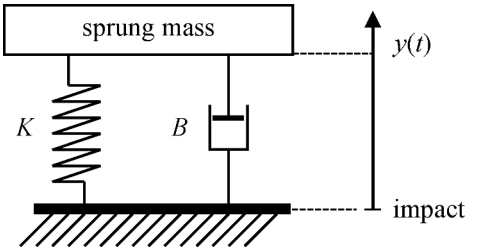


Fig. 5. Mechanical suspension of a vehicle, where the reconstructed signaly represents a system free response (79).

Table 2
Reconstruction of Continuous-Signal Parameters using the LS and SLA Methods.

	LS	SLA
ω (6.4)	6.53	6.43
ζ (1.5)	1.17	1.46
Ω (15)	12.22	14.23

procedure (46)-(48) is sufficiently close to the pure absolute value ($|e|$) used in the simple LA procedure (24)-(26). This means that both SLA and LA schemes should also behave very similarly.

Additionally, we verified the operation of the numerical simplex procedure (described in Section 5) implemented to minimize the LA index (18) (the whole thing can be referred to as 'simplex LA', i.e. LA estimation in the simplex sense). Also in this case, the obtained estimates did not differ significantly from the LA and SLA results (Table 1). However, it should be emphasized that the LA/SLA procedures have the same complexity $O(n^3)$, while the simplex scheme is characterized by (in the worst case) exponential complexity.

It is obvious that the choice of the polynomial degree (n) affects the accuracy of the resulting polynomial reconstruction of the underlying signal (79). On the other hand, it is best to implement models (75) with as few parameters as possible. To resolve this trade-off, the polynomials $\chi(t)$ of different degrees were fitted to the undisturbed signal $y(t)$. Taking into account that the extreme values of $y(t)$ are y_{max} and y_{min}, the relative error $e_\chi = max|y(t) - \chi(t)|/(y_{max} - y_{min})$ was considered a reasonable measure for the effective evaluation of a polynomial and its degree. Preferring the lowest possible degree of the polynomial and considering the value of $e_\chi \approx 5\%$ as acceptable, we decided to assume $n = 5$ (which gives the relative error $e_\chi \approx 4.26\%$).

6.2. Identification of a Non-Stationary system

In supervisory and control systems, on-line monitoring of the evolution of industrial processes is needed. This requires adopting an appropriate model of the supervised process. The parameters of this model are then assessed using appropriate procedures. Diagnostic information can be derived from the analysis of estimated parameters or other aggregate measures (e.g. spectral density function or cumulative periodogram).

To illustrate such a situation, the recursive SLA algorithm was used to identify on-line the following autoregressive discrete-time object dynamics model:

$$y(l) = -\sum_{i=1}^{n} a_i y(l-i) + \sum_{i=1}^{m} b_i u(l-i) + e(l) \tag{81}$$

where $e(l)$ is the error of the equation, $u(l)$ and $y(l)$ are the sampled input and output, respectively. As before (80), contamination of the output with white noise $w(l)$ of uniform distribution and variance $\sigma_w^2 = 10^{-2}$ was taken into account.

The standard form of regression [14] for (81) is

$$\chi(l) = \boldsymbol{\varphi}^{\mathrm{T}}(l)\boldsymbol{\theta} + e(l) \tag{82}$$

$$\boldsymbol{\varphi}(l) = [\,-\chi(l-1)\ \cdots\ -\chi(l-n)\ \ u(l-1)\ \cdots\ u(l-m)\,]^{\mathrm{T}} \tag{83}$$

$$\boldsymbol{\theta} = [a_1\ \cdots\ a_n\ \ b_1\ \cdots\ b_m]^{\mathrm{T}} \tag{84}$$

where the parameter vector $\boldsymbol{\theta}$ contains the model coefficients, and the delayed samples of the input and output signals are stored in the regression vector $\boldsymbol{\varphi}(l)$.

The following deterministic excitation was used:

$$u(l) = \Omega_1 \sin(\omega_1 l) + \Omega_2 \sin(\omega_2 l) + \Omega_3 \sin(\omega_3 l) \tag{85}$$

with amplitudes $\Omega_1 = \Omega_2 = \Omega_3 = 25$ and frequencies $\omega_1 = 0.3$, $\omega_2 = 0.7$, $\omega_3 = 0.9$. The non-stationary behavior of the observed

system (81) with low order ($n = 2, m = 2$) and three constants ($a_2 = 0.3, b_1 = 1.0, b_2 = -0.9$) was associated with a gradual change of only one parameter $a_1 = -0.2 \cdots -0.4$ in the time interval $l = 50 \cdots 250$. The measurements sequence $y(l)$ recorded (with noise) at the computational moments $l = 1 \cdots k$ ($k = 300$) was then locally reset to zero $y(l) = 0$ on two intervals (for $l = 95 \cdots 99$ and for $l = 195 \cdots 199$).

The modeled system (81) was identified using the recursive procedures LS (15)-(17) and SLA (56)-(58) with $\alpha = 20$. The weighting mechanism ($\lambda = 0.98$) was used to track the variability of the non-stationary dynamics parameters.

The LS algorithm was initiated using the diagonal matrix $\boldsymbol{P}(0) = \mathrm{diag}\left[10^5 \cdots 10^5\right]$, while the SLA routine was activated at time $l = 20$ based on the LS results (see description in Section 3). The quality of estimation is illustrated in Figs. 6-7.

As expected, the weighting mechanism serves well to identify non-stationary dynamics, but LS shows strong sensitivity to occasional outliers, while SLA has great robustness to large errors. It can therefore be seen that, despite its approximate nature, the recursive SLA procedure can be recommended as a reliable tool for identifying dynamic systems on-line without additional measures applied to the case of outliers.

Additionally, both experiments presented in this section were performed for different levels of additive measurement noise. The simulated noise here mainly represents quantization effects occurring in the sampling devices. Since such devices (AD converters) perform rounding-off on sampled data with different values, the resulting residual 'quantization noise' will certainly be uniformly distributed zero-mean white noise. Bearing in mind that by increasing the resolution of the converters, the variance of this noise (proportional to the square of the quantization step) can be dramatically reduced, we also devoted some attention to the problems caused by additive measurement noise.

However, in order to numerically analyze the impact of quantization noise on the estimation results, we repeated the simulation tests with a significantly increased (even one hundred times) level of noise variance σ_w^2. It was observed that the SLA procedure remains insensitive to such simulated outliers. At the same time, the accuracy of SLA estimation suffered from the inevitably increased 'local variance' of individual estimates, which (to some extent) reduced the quality of parametric identification under these conditions.

It is obvious that we can improve the estimation accuracy by increasing the AD resolution. Moreover, the concept of instrumental variables is an effective tool for improving consistency in estimation (as suggested in Section 2).

The basic property of the studied batch and recursive SLA estimators is the desired insensitivity to large measurement errors. Important issues of polynomial signal reconstruction and dynamic system identification have been deliberately addressed to demonstrate the practical attractiveness of the method. Other practical implementations of the LA approach can be found in the literature [13,22].

7. Conclusion

In this study, practical estimation algorithms in the sense of the least sum of absolute deviations have been discussed and verified in practical tests. From a theoretical point of view, it is important that, thanks to the demonstrated convergence of the LA estimator (Appendix A), the concept of parametric estimation proposed here has a solid mathematical justification.

In general, the analyzed LA estimators when confronted with LS procedures turn out to be exceptionally effective in identifying systems in the presence of destructive measurement errors (in the form of random outliers or other errors represented by sequences of erroneous or zero values in measurement records). Importantly, our recursive implementation of the LA and SLA methods is uncomplicated and similar to the classical LS procedure, while the Gauss-Newton estimator can only be run off-line (with explicit matrix inversion).

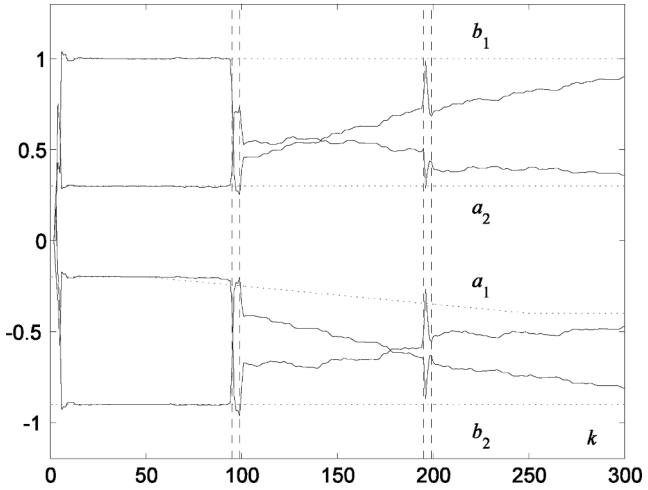


Fig. 6. Trajectories of the parameters of the non-stationary system (thin dotted lines) and the evolution of four LS estimates (solid lines), where the intervals with outliers are marked with dashed vertical lines.

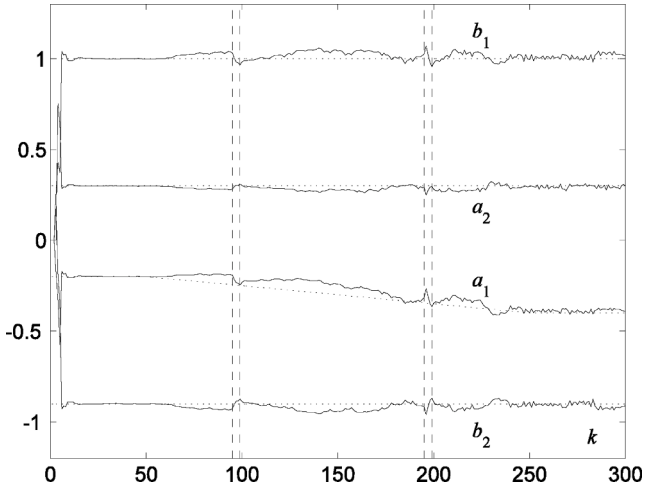


Fig. 7. Trajectories of the parameters of the non-stationary system (thin dotted lines) and the evolution of four SLA estimates (solid lines), where the intervals with outliers are marked with dashed vertical lines.

The numerous simulation results mentioned and included in this article show that the practical forgetting mechanism used in the conventional realization of the weighted LS routine also works in the recursive implementation of the LA and SLA methods.

7.1. Reported contribution

The original modification of the primary iterative LA method based on the sigmoid function (39), resulting in the minimum uniqueness of the redefined (strictly convex) LA index (37), appears to be an innovative remedy to the problem of minimization of the primary LA criterion (18) with problematic non-differentiable 'kink points'.

The recursive implementations (34)-(36) and (56)-(58) of the weighted LA estimator, which can be run on-line, allow for reliable system identification irrespective of their approximate nature. Importantly, an interesting confrontation of the improved iterative LA procedure and the classical Gauss-Newton gradient algorithm (Section 5) proves the predominance of our innovative SLA solution over the standard Gauss-Newton implementation (i.e. in terms of positive definiteness of the information and covariance matrices).

It is instructive from the interpretation point of view that the recursive LA procedures can be treated as particular realizations of the IV method (15)-(17) with the instrumental variable taking form of the regression vector modified using special factors (i.e. the profile functions of the prediction error, including the original absolute value function, as well as the shaping function (49)), used for the primary and smoothed LA realizations. It is also important that the proposed numerical evaluation (49) of the error shaping factor $\mu(e)$ perfectly overcomes the problem of small divisors.

The usefulness of the developed smooth LA concept is proved by means of simulation tests, in which the announced 'genetic insensitivity' to destructive faults (random outliers or sequences of faulty data) is evident.

In practice, it may be useful to implement a complex iterative-recursive LA algorithm that uses both an iterative (precise) LA and recursive (approximate) LA schemes. This can be achieved by performing iterative processing of LA (24)-(26) between successive sampling moments. However, there is Hessian's inversion in the iterative formula (24), which increases the processing time. To meet the time constraints of such an LA solution (i.e. stop iterations before the next sampling moment), one can relax the constraints imposed on the estimation accuracy (27) by increasing the threshold value Δ_{min}.

7.2. Further research

From a scientific point of view, it is advisable to further develop the discussed error-insensitive methods and apply them in many areas, such as process supervision, adaptive filtering and intelligent control.

1. Improved system identification, error tolerant and effective for 'highly non-stationary' systems: To further immunize LA procedures against additive high noise, it is practical to use dedicated, noise-free LA instruments strengthen by the instrumental variable method. Promising results can be found in [15,16]. Moreover, it is possible to reliably track systems with variable parameters and 'strong non-stationarity' by using the concept of parallel identification [23] based on LA/SLA estimators.
2. Error-insensitive identification of non-trivial objects: In many practical situations, the superiority of continuous-time models over discrete-time ones is obvious, mainly because continuous-time models are physically motivated and thus retain the proper physical meaning of the identified parameters [5,8]. The tasks of identifying non-linear objects, infinite-dimensional models, or delay systems are considered to be complex technical challenges. There are many such practical or industrial examples available in the

literature, which concern objects with static non-linearities [15,24], partial differential equations [25] and systems with unknown input delay [26,27].

3. Modern methods of system identification: Currently, some non-classical solutions are also popular in the practical implementation of process parameter estimation. In general, these approaches can be derived from a large family of artificial intelligence methods such as genetic algorithms [28] and (deep) neural networks [29]. Some interesting implementations of this type can already be found in the literature (e.g. [30]).

CRediT authorship contribution statement

Janusz Kozłowski: Writing – original draft, Visualization, Validation, Resources, Methodology, Formal analysis, Data curation. **Zdzisław Kowalczuk:** Writing – review & editing, Supervision, Methodology, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Appendix A

Convergence of the LA Algorithm

A detailed proof of the convergence property can be found in [13]. Here we present the essence of analytical reasoning so as to draw constructive conclusions.

A. Theorem: The sequence $J^{[r]} = J\left(\widehat{\boldsymbol{\theta}}^{[r]}\right)$ of the LA functional (18) determined in the successive iterations along with the primary LA estimate (24)-(26) is decreasing ($r = 0,\ 1,\ \cdots$): $J^{[r+1]} < J^{[r]}$.

B. Proof: As the sum of absolute values (18) is positive, the target inequality is equivalent to: $\left(J^{[r+1]}\right)^2 < \left(J^{[r]}\right)^2$.

The LA functional (18) can be shown as

$$J(\boldsymbol{\theta}) = \sum_{l=1}^{k} \gamma(l)|e(l)| = \sum_{l=1}^{k} \gamma(l)\left|\chi(l) - \boldsymbol{\varphi}^{\mathrm{T}}(l)\boldsymbol{\theta}\right| \tag{A.1}$$

with the weighting mechanism $\gamma(l) > 0$ represented by any summable sequence ($\gamma(1) + \gamma(2) + \cdots < \infty$), including the exponential function $\gamma(l) = \lambda^{k-l}$. The LA estimates for $r = 0,\ 1,\ \cdots$ that optimize (18) are

$$\widehat{\boldsymbol{\theta}}^{[r+1]} = \widehat{\boldsymbol{\theta}}^{[r]} + [\boldsymbol{H}(k)]^{-1}\boldsymbol{\psi}(k) \tag{A.2}$$

where the prediction error $\widehat{e}^{[r]}(l)$, the gradient $'-\boldsymbol{\psi}(k)'$ and the Hessian $\boldsymbol{H}(k)$ are expressed as

$$\widehat{e}^{[r]}(l) = \chi(l) - \boldsymbol{\varphi}^{\mathrm{T}}(l)\widehat{\boldsymbol{\theta}}^{[r]} \tag{A.3}$$

$$\boldsymbol{H}(k) = \sum_{l=1}^{k} \gamma(l)\frac{\boldsymbol{\varphi}(l)\boldsymbol{\varphi}^{\mathrm{T}}(l)}{|\widehat{e}^{[r]}(l)|} \tag{A.4}$$

$$\boldsymbol{\psi}(k) = \sum_{l=1}^{k} \gamma(l)\frac{\boldsymbol{\varphi}(l)\widehat{e}^{[r]}(l)}{|\widehat{e}^{[r]}(l)|} \tag{A.5}$$

Now assume that in subsequent iterations the prediction error is non-zero: $\widehat{e}^{[r]}(l) \neq 0$, for $l = 1\cdots k$. This results in the positive definiteness of the Hessian $\boldsymbol{H}(k)$, where for any non-zero vector of compatible size ($\|\boldsymbol{v}\| \neq 0$)

$$\boldsymbol{v}^{\mathrm{T}}[\boldsymbol{H}(k)]\boldsymbol{v} = \sum_{l=1}^{k} \gamma(l)\frac{[\boldsymbol{v}^{\mathrm{T}}\boldsymbol{\varphi}(l)]^2}{|\widehat{e}^{[r]}(l)|} > 0 \tag{A.6}$$

By introducing the covariance matrix $\boldsymbol{F}(k) = \boldsymbol{H}^{-1}(k)$, formula (A.2) can be developed step by step as follows

$$\boldsymbol{\varphi}^{\mathrm{T}}(l)\widehat{\boldsymbol{\theta}}^{[r+1]} = \boldsymbol{\varphi}^{\mathrm{T}}(l)\left[\widehat{\boldsymbol{\theta}}^{[r]} + \boldsymbol{F}(k)\boldsymbol{\psi}(k)\right] \tag{A.7}$$

$$\chi(l) - \boldsymbol{\varphi}^{\mathrm{T}}(l)\widehat{\boldsymbol{\theta}}^{[r+1]} = \chi(l) - \boldsymbol{\varphi}^{\mathrm{T}}(l)\left[\widehat{\boldsymbol{\theta}}^{[r]} + \boldsymbol{F}(k)\boldsymbol{\psi}(k)\right] \tag{A.8}$$

$$\left|\widehat{e}^{[r+1]}(l)\right| = \left|\widehat{e}^{[r]}(l) - \boldsymbol{\varphi}^{\mathrm{T}}(l)\boldsymbol{F}(k)\boldsymbol{\psi}(k)\right| \tag{A.9}$$

$$\sum_{l=1}^{k}\gamma(l)\left|\widehat{e}^{[r+1]}(l)\right| = \sum_{l=1}^{k}\gamma(l)\left|\widehat{e}^{[r]}(l) - \boldsymbol{\varphi}^{\mathrm{T}}(l)\boldsymbol{F}(k)\boldsymbol{\psi}(k)\right| \tag{A.10}$$

Using the simplifying notation $J^{[r+1]} = J\left(\widehat{\boldsymbol{\theta}}^{[r+1]}\right)$, we get

$$J^{[r+1]} = \sum_{l=1}^{k}\gamma(l)g(l)h(l) \tag{A.11}$$

$$g(l) = \sqrt{\left|\widehat{e}^{[r]}(l)\right|} \tag{A.12}$$

$$h(l) = \frac{\left|\widehat{e}^{[r]}(l) - \boldsymbol{\varphi}^{\mathrm{T}}(l)\boldsymbol{F}(k)\boldsymbol{\psi}(k)\right|}{\sqrt{\left|\widehat{e}^{[r]}(l)\right|}} \tag{A.13}$$

Then we apply the 'weighted' Cauchy-Schwarz theorem (with the weighting sequence $\gamma(l) > 0$)

$$\left[\sum_{l=1}^{k}\gamma(l)g(l)h(l)\right]^2 \le \left[\sum_{l=1}^{k}\gamma(l)g^2(l)\right]\left[\sum_{l=1}^{k}\gamma(l)h^2(l)\right] \tag{A.14}$$

Therefore, based on (A.11)-(A.13), the above (A.14) takes the form

$$\left(J^{[r+1]}\right)^2 \le \left[\sum_{l=1}^{k}\gamma(l)g^2(l)\right]\left[\sum_{l=1}^{k}\gamma(l)h^2(l)\right] \tag{A.15}$$

Still, the expressions in parentheses can be shown as

$$\sum_{l=1}^{k}\gamma(l)g^2(l) = \sum_{l=1}^{k}\gamma(l)\left|\widehat{e}^{[r]}(l)\right| = J^{[r]} \tag{A.16}$$

$$\begin{aligned}
\sum_{l=1}^{k}\gamma(l)h^2(l) &= \sum_{l=1}^{k}\gamma(l)\frac{\left[\widehat{e}^{[r]}(l) - \boldsymbol{\varphi}^{\mathrm{T}}(l)\boldsymbol{F}(k)\boldsymbol{\psi}(k)\right]^2}{\left(\sqrt{\left|\widehat{e}^{[r]}(l)\right|}\right)^2} \\
&= \sum_{l=1}^{k}\gamma(l)\frac{\left(\widehat{e}^{[r]}(l)\right)^2}{\left|\widehat{e}^{[r]}(l)\right|} - 2\left(\sum_{l=1}^{k}\gamma(l)\frac{\widehat{e}^{[r]}(l)\boldsymbol{\varphi}^{\mathrm{T}}(l)}{\left|\widehat{e}^{[r]}(l)\right|}\right)\boldsymbol{F}(k)\boldsymbol{\psi}(k) + \sum_{l=1}^{k}\gamma(l)\frac{\left[\boldsymbol{\varphi}^{\mathrm{T}}(l)\boldsymbol{F}(k)\boldsymbol{\psi}(k)\right]^2}{\left|\widehat{e}^{[r]}(l)\right|} \\
&= \sum_{l=1}^{k}\gamma(l)\left|\widehat{e}^{[r]}(l)\right| - 2\left(\sum_{l=1}^{k}\gamma(l)\frac{\boldsymbol{\varphi}(l)\widehat{e}^{[r]}(l)}{\left|\widehat{e}^{[r]}(l)\right|}\right)^{\mathrm{T}}\boldsymbol{F}(k)\boldsymbol{\psi}(k) + \sum_{l=1}^{k}\gamma(l)\frac{\left[\boldsymbol{\psi}^{\mathrm{T}}(k)\boldsymbol{F}^{\mathrm{T}}(k)\boldsymbol{\varphi}(l)\right]\left[\boldsymbol{\varphi}^{\mathrm{T}}(l)\boldsymbol{F}(k)\boldsymbol{\psi}(k)\right]}{\left|\widehat{e}^{[r]}(l)\right|} \\
&= J^{[r]} - 2\boldsymbol{\psi}^{\mathrm{T}}(k)\boldsymbol{F}(k)\boldsymbol{\psi}(k) + \boldsymbol{\psi}^{\mathrm{T}}(l)\boldsymbol{F}^{\mathrm{T}}(k)\left(\sum_{l=1}^{k}\gamma(l)\frac{\boldsymbol{\varphi}(k)\boldsymbol{\varphi}^{\mathrm{T}}(l)}{\left|\widehat{e}^{[r]}(l)\right|}\right)\boldsymbol{F}(k)\boldsymbol{\psi}(k) \\
&= J^{[r]} - 2\boldsymbol{\psi}^{\mathrm{T}}(k)\boldsymbol{F}(k)\boldsymbol{\psi}(k) + \boldsymbol{\psi}^{\mathrm{T}}(k)\boldsymbol{F}^{\mathrm{T}}(k)\boldsymbol{F}^{-1}(k)\boldsymbol{F}(k)\boldsymbol{\psi}(k) = J^{[r]} - \boldsymbol{\psi}^{\mathrm{T}}(k)\boldsymbol{F}(k)\boldsymbol{\psi}(k)
\end{aligned} \tag{A.17}$$

The above transformations use the symmetry of the matrix $\boldsymbol{F}(k)$ and the product transposition property

$$\boldsymbol{F}^{\mathrm{T}}(k) = \boldsymbol{F}(k) = \boldsymbol{H}^{-1}(k) \tag{A.18}$$

$$\boldsymbol{\varphi}^{\mathrm{T}}(l)\boldsymbol{F}(k)\boldsymbol{\psi}(k) = \boldsymbol{\psi}^{\mathrm{T}}(k)\boldsymbol{F}^{\mathrm{T}}(k)\boldsymbol{\varphi}(l) \tag{A.19}$$

Based on (A.16)-(A.17), we get inequality (A.15) as

$$\left(J^{[r+1]}\right)^2 \le \left(J^{[r]}\right)^2 - J^{[r]}\boldsymbol{\psi}^{\mathrm{T}}(k)\boldsymbol{F}(k)\boldsymbol{\psi}(k) \tag{A.20}$$

C. Discussion: Since the Hessian $\boldsymbol{H}(k)$ (i.e. the information matrix) is positive definite (A.6), its inverse $\boldsymbol{F}(k) = \boldsymbol{H}^{-1}(k)$ (the covariance matrix) is also positive definite. Therefore, for a non-zero value of the gradient ($\|\boldsymbol{\psi}(k)\| \neq 0$) we have $\boldsymbol{\psi}^{\mathrm{T}}(k)\boldsymbol{F}(k)\boldsymbol{\psi}(k) > 0$. Since the LA quality index is always positive ($J^{[r]} > 0$), the term $(-J^{[r]}\boldsymbol{\psi}^{\mathrm{T}}(k)\boldsymbol{F}(k)\boldsymbol{\psi}(k))$ in (A.20) is less than zero, so the target relation must be sharp: $\left(J^{[r+1]}\right)^2 < \left(J^{[r]}\right)^2$.

In the case of zero gradient ($\|\boldsymbol{\psi}(k)\| = 0$) the LA algorithm (24)-(26) gives $\widehat{\boldsymbol{\theta}}^{[r+1]} = \widehat{\boldsymbol{\theta}}^{[r]}$, then the iterations become inefficient ($J^{[r+1]} = J^{[r]}$), therefore the terminal condition (27) stops this loop with $\widehat{\boldsymbol{\theta}}_{\mathrm{LA}} = \widehat{\boldsymbol{\theta}}^{[r]}$. This solves the problem of the (rather unlikely) 'flat minimum' of the LA functional (18).

The 'monotonic convergence' theorem states that every descending sequence bounded from below converges. Bearing in mind that the sequence $J^{[r]}$ for $r = 0,\ 1,\ \cdots$ is decreasing and the values of index (18) are bounded from below ($J^{[r]} > 0$), we conclude that the sequence $J^{[r]}$ is convergent. This proves that the iterative routine (24)-(26) minimizes the LA index (18).

D. Comment: It should be recalled that in the basic LA procedure (24)-(26) the residual errors $\widehat{e}^{[r]}(l)$ for $l = 1 \cdots k$ must be non-zero. In fact, the vanishing errors ($|\widehat{e}^{[r]}(l)| \approx 0$), inevitably leading to 'approximating zero' divisors in $\boldsymbol{H}(k)$, are replaced by the threshold e_{min} (see the discussion in Section 3).

Yet, we would like to emphasize that using the smoothed LA method (46)-(48), which is based on the minimization of criterion (37), the problem of divisors close to zero naturally disappears, because for $|\widehat{e}^{[r]}(l)| \approx 0$ the error shaping factor $\mu(\widehat{e}^{[r]}(l))$ is always positive and never decreases below α^{-1}.

Finally, regarding the SLA procedure, since in the case of large α both quality criteria LA (18) and SLA (37) are 'virtually identical', there is a reasonable premise that the iterative SLA (46)-(48) also converges in light of the proof for the presented LA convergence.

References

[1] W. Byrski, M. Drapała, J. Byrski, An adaptive identification method based on the modulating functions technique and exact state observers for modeling and simulation of a nonlinear MISO glass melting process, Int. J. Appl. Math. Comput. Sci. 29 (4) (2019) 739–757, https://doi.org/10.2478/amcs-2019-0055.

[2] P. Suchomski, Z. Kowalczuk, Analytical design of stable delta-domain generalized predictive control, Opt. Control Appl. Methods 23 (5) (2002) 239–273, https://doi.org/10.1002/oca.712.

[3] R.H. Middleton and G.C. Goodwin, *Digital Control and Estimation. A Unified Approach*, Prentice-Hall, Upper Saddle River, NJ, USA, 1990.

[4] J. Schoukens, Modeling of continuous time systems using a discrete time representation, Automatica 26 (3) (1990) 579–583, https://doi.org/10.1016/0005-1098(90)90029-H.

[5] Z. Kowalczuk, J. Kozłowski, Continuous-time approaches to identification of continuous-time systems, Automatica 36 (8) (2000) 1229–1236, https://doi.org/10.1016/S0005-1098(00)00033-9.

[6] H. Unbehauen, G.P. Rao, Continuous-time approaches to system identification – a survey, Automatica 26 (1) (1990) 23–35, https://doi.org/10.1016/0005-1098(90)90155-B.

[7] R. Johansson, Identification of continuous-time models, IEEE Trans. Signal Process. 42 (4) (1994) 887–897, https://doi.org/10.1109/78.285652.

[8] S. Sagara, Z.J. Yang, K. Wada, Identification of continuous systems using digital low-pass filters, Int. J. Syst. Sci. 22 (7) (1991) 1159–1176, https://doi.org/10.1080/00207729108910693.

[9] J.E. Gentle, Least absolute values estimation: an introduction, Commun. Stat. – Simul. Comput. 6 (4) (1977) 313–328, https://doi.org/10.1080/03610917708812047.

[10] K.B. Janiszowski, "Towards estimation in the sense of the least sum of absolute errors," *IFAC Proceedings Volumes* (ISSN 14746670), vol. 31, no. 20, pp. 605–610, 1998, doi: 10.1016/S1474-6670(17)41862-3.

[11] S. Eakambaram, A. Rex Irudhaya Raj, Robust Regression using Least Absolute Deviations Method, Int. J. Mech. Eng. 7 (5) (2022) 53–57.

[12] F.H. Thanoon, Robust Regression by Least Absolute Deviations Method, Int. J. Statist. Appl. 5 (3) (2015) 109–112, https://doi.org/10.5923/j.statistics.20150503.02.

[13] J. Kozłowski and Z. Kowalczuk, "Robust to measurement faults parameter estimation algorithms in issues on systems diagnosis," in Z. Kowalczuk and B. Wiszniewski, eds., *Automation and Informatics: Information technologies – Diagnostics*, Gdańsk, Poland: PWNT, 2007, pp. 221–240.

[14] L. Ljung, *System Identification: Theory for the User*, Upper Saddle River, NJ, Prentice-Hall, USA, 1987.

[15] J. Kozłowski, Z. Kowalczuk, Discrete identification of continuous non-linear and non-stationary dynamical systems that is insensitive to noise correlation and measurement outliers, Archiv. Control Sci. 33 (2) (2023) 391–411, https://doi.org/10.24425/acs.2023.146281.

[16] T. Soderstrom, P. Stoica, Comparison of some instrumental variable methods – consistency and accuracy aspects, Automatica 17 (1) (1981) 101–115, https://doi.org/10.1016/0005-1098(81)90087-X.

[17] E. Schlossmacher, An iterative technique for absolute deviations curve fitting, J. Am. Stat. Assoc. 68 (344) (1973) 857–865, https://doi.org/10.1080/01621459.1973.10481436.

[18] Z. Kowalczuk, J. Kozłowski, Non-quadratic quality criteria in parameter estimation of continuous-time models, IET Control Theory Appl. 5 (13) (2011) 1494–1508, https://doi.org/10.1049/iet-cta.2010.0310.

[19] G. Zhang, Y. Shi, Y. Sheng, Least absolute deviation estimation for uncertain vector autoregressive model with imprecise data, Int. J. Uncertainty, Fuzziness and Knowledge-Based Systems 31 (3) (2023) 353–370, https://doi.org/10.1142/S0218488523500186.

[20] V. Cerda, J.L. Cerda, A.M. Idris, Optimization using the gradient and simplex methods, Talanta(int. J. Pure and Applied Analytical Chem.) 148 (2016) 641–648, https://doi.org/10.1016/j.talanta.2015.05.061.

[21] X. Cai, L. Xue, F. Lu, Robust estimation with a modified Huber's loss for partial functional linear models based on splines, J. Korean Stat. Soc. 49 (4) (2020) 1214–1237, https://doi.org/10.1007/s42952-020-00052-x.

[22] J. Kozłowski, Z. Kowalczuk, Identification of continuous systems – practical issues of insensitivity to perturbations, in: J.M. Kościelny, M. Syfert, A. Sztyber (Eds.), Advanced Solutions in Diagnostics and Fault Tolerant Control (advances in Intelligent Systems and Computing), Springer IP AG, Cham, Switzerland, 2018, pp. 180–191, https://doi.org/10.1007/978-3-319-64474-5_15.

[23] Z. Kowalczuk, Competitive identification for self-tuning control: robust estimation design and simulation experiments, Automatica 28 (1) (1992) 193–201, https://doi.org/10.1016/0005-1098(92)90021-7.

[24] J. Schoukens, L. Ljung, Nonlinear system identification: a user-oriented road map, IEEE Control Syst. Mag. 39 (6) (2019) 28–99, https://doi.org/10.1109/MCS.2019.2938121.

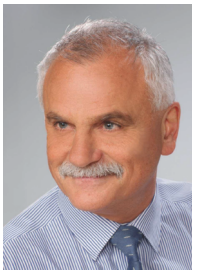
[25] S. Sagara and Z.Y. Zhao, "Identification of system parameters in distributed parameter systems," *Proc. 11th IFAC World Congress*, Tallinn, Estonia, 1990, pp. 471–476, doi: 10.1016/S1474-6670(17)51960-6.

[26] J. Kozłowski, Z. Kowalczuk, On-line parameter and delay estimation of continuous-time dynamic systems, Int. J. Appl. Math. Comput. Sci. 25 (2) (2015) 223–232, https://doi.org/10.1515/amcs-215-0017.

[27] Z.Y. Zhao, S. Sagara, Consistent estimation of time delay in continuous-time systems, Trans. Soc. Instrum. Control Engineers 27 (1) (1991) 64–69, https://doi.org/10.9746/sicetr1965.27.64.

[28] D. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Reading, MA, Addison-Wesley, USA, 1989.
[29] D.T. Pham, X. Liu, *Neural Networks for Identification, Prediction and Control*, London, Springer-Verlag, UK, 1995.
[30] D. Uciński, M. Patan, Sensor network design for the estimation of spatially distributed processes, Int. J. Appl. Math. Comput. Sci. 20 (3) (2010) 459–481, https://doi.org/10.2478/v10006-010-0034-2.

Janusz Kozłowski obtained the title of M.Sc. Eng. (1990) in automation and Ph.D. (2000) in electronics, both at the Gdańsk University of Technology. He gained industrial experience working for ABB Stromberg (Finland), where his activities focused on designing communication software and implementing communication protocols. Currently, he works at the Faculty of Electronics, Telecommunications and Computer Science of the GUT, where his main research focuses on modeling and identification of continuous-time systems, signal processing and adaptive control.



Zdzisław Kowalczuk (Senior Member, IEEE) has been associated with the Faculty of Electronics, Telecomm. and Informatics at the Gdańsk University of Technology since 1978, where he is a Full Professor in automatic control and robotics (Prof. DSc PhD MScEE: 2003, 1993, 1986, 1978), and the Chair of the Dept. of Robotics and Decision Systems. He held visiting appointments at University of Oulu (1985), Australian National University (1987), Technische Hochschule Darmstadt (1989), and at George Mason University (1990-1991). Main scientific interests include robotics, control theory, adaptation, system modeling, identification and estimation, diagnostics, failure detection, signal processing, artificial intelligence, control engineering and computer science. He has authored and co-authored about 30 books (incl. WNT 2002, PWNT 2007-2021, Springer 2004-2023), over 120 journal papers (over 50 on JCR) and over 350 conference publications and book chapters. His citation index in Google Scholar exceeded 33 hundred with a Hirsh index of 21. He is the President of the Polish Consultants Society (TKP) and of the Polish Society for Measurements, Automatic Control and Robotics (POLSPAR, the NMO of IFAC), and a member of the Automation and Robotics Committee of the Polish Academy of Sciences. Since 2003 professor Kowalczuk is the founder and chief editor of the publishing house PWNT – the Pomeranian Science and Technology Publishers. He is also a winner of the Awards of the Minister of National Education for outstanding research achievements for 1990 and 2003 and other state awards and medals, as well as the Award of the Foundation of Polish Science for 1999 in the field of automation and the Medal of the Association of Polish Electrical Engineers (SEP) for 2014 named after Professor Paweł Jan Nowacki (co-founder of IFAC).