Applied Soft Computing 115 (2022) 108174

Contents lists available at ScienceDirect

Applied Soft Computing

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

Evolved Extended Kalman Filter for first-order dynamical systems with unknown measurements noise covariance

Leonardo Herrera [a], M.C. Rodríguez-Liñán [b], Eddie Clemente [c,*], Marlen Meza-Sánchez [d], Luis Monay-Arredondo [d]

[a] Mechanical and Aerospace Engineering Department, NPS, Monterey, CA, United States
[b] CONACYT-TecNM/I.T. Ensenada, Mexico
[c] Tecnológico Nacional de México/ I. T. Ensenada, Blvd. Tecnológico 150, Ex-ejido Chapultepec, Zip Code 22780, Ensenada, B.C., Mexico
[d] Tecnológico Nacional de México/ I. T. Tijuana, Blvd. Industrial s/n, Cd Industrial, Zip Code 22430, Tijuana, B.C., Mexico

ARTICLE INFO

Article history:
Received 19 May 2021
Received in revised form 9 October 2021
Accepted 10 November 2021
Available online 5 December 2021

Keywords:
Extended Kalman Filter
Analytic behaviors
Genetic programming
Nonlinear first-order dynamical systems
Logistic map system

ABSTRACT

The present work focuses on an open problem in the design of Extended Kalman filters: the lack of knowledge of the measurement noise covariance. A novel extension of the analytic behaviors framework, which integrates a theoretical formulation and evolutionary computing, has been introduced as a design methodology for the construction of this unknown parameter. The proposed methodology is developed and applied for the design of Evolved Extended Kalman Filters for nonlinear first-order dynamical systems. The proposed methodology applies an offline evolutionary synthesis of analytic nonlinear functions, to be used as measurement noise covariance, aiming to minimize the Kalman criterion. The virtues of the methodology are exemplified through a complex, highly nonlinear, first-order dynamical system, for which 2649 optimized replacements of the measurement noise covariance are found. Under different scenarios, the performance of the Evolved Extended Kalman Filter with unknown measurement noise covariance is compared with that of the conventional Extended Kalman Filter where the measurement noise covariance is known. The robustness of the Evolved Extended Kalman Filter is demonstrated through numerical evaluation.

1. Introduction

The Extended Kalman Filter (EKF) is a generalized version of the well-known state estimator Kalman Filter (KF) dedicated to nonlinear systems. The EKF has been proved to be useful from the Apollo moon landing project in the sixties (see [1,2]), and continues being relevant to date [3,4].

The functionality of the EKF is based on the covariance matrices that characterize the White Gaussian Noise (WGN) in the process and in the measurements. In the standard EKF, both covariance matrices are typically assumed constant values, being either known, proposed by an expert after analyzing the process and the measurements, or found by trial-and-error, among other techniques. Most of the time, the covariance matrices are difficult to be known, hence, the search for a replacement of these parameters is required.

The revision of literature shows several approaches to deal with the uncertainty in the covariance matrices. Some works aim to estimate, or identify, a numeric value for the unknown noise covariances by introducing mostly adaptive models or optimization techniques. Afterwards, the found value can be used either for the EKF, or to cancel the effect of noise, either in the process or in the measurement. In the early 70s, Mehra [5] classified the, so far known, adaptive filtering approaches into four categories: Bayesian, Maximum Likelihood, Correlation, and Covariance Matching. Among the new solutions found in the literature, are innovation-based methods [6–8]; fuzzy logic and/or neural networks [9–14]; stochastic or probabilistic approaches [15–18]; and metaheuristics [19], in particular Genetic Algorithms [20–22], and Differential Evolution [23]. Despite the large amount of research devoted to this particular aspect of Kalman Filters design, dating more than fifty years, the tuning of the filter's covariance matrices still remains an open problem [24].

This work focuses on the measurement noise covariance of the filter, i.e., it considers a configuration of noisy measurements coming from a deterministic process, as in [25–29]. A novel methodology that builds replacement functions for the typically unknown measurement noise covariance, which is an important tuning parameter in the EKF, is proposed. In contrast to the works found in the literature, the proposed methodology is built as an extension of the analytic behaviors framework originally

* Corresponding author.
E-mail addresses: leonardo.herrera@nps.edu (L. Herrera), mcrodriguez@conacyt.mx (M.C. Rodríguez-Liñán), eclemente@ite.edu.mx (E. Clemente), iliana.meza@tectijuana.edu.mx (M. Meza-Sánchez), jlmonay@ite.edu.mx (L. Monay-Arredondo).

https://doi.org/10.1016/j.asoc.2021.108174
1568-4946/

developed for the construction of nonlinear controllers [30–34]. The novelty of this framework resides in the integration of a theoretical formulation with an evolutionary process based on the Genetic Programming paradigm. Focused in our problem, it allows to find dynamic expressions that automatically adjust themselves to minimize the Kalman criterion, and can deal with parametric variations. A remarkable feature with respect to the literature, where the aim is to propose constant estimates of the unknown covariances, here the measurement noise covariance is replaced by an analytic function that depends on the filter's intrinsic variables and on the system's noisy measurement. Another important feature of the methodology is that it produces a large set of suitable solutions to the given problem. From these, the designer can choose the most appropriate analytic function replacement to be used online for real-world applications, such as navigation and control [35–37]. Tuning the EKF with the new replacement generates a new filter called Evolved Extended Kalman Filter (EEKF).

This document is organized as follows. Section 2 states the theoretical preliminaries for the EKF, a brief introduction of the analytic behaviors framework, and the description of the dynamics of the logistic map system employed as a testbed for the general methodology. The general form of the methodology developed for the construction of EEKFs, applied to nonlinear, First Order Dynamical Systems (FO-DS) is presented in Section 3. The application of the methodology for a particular FO-DS, logistic map system, is described in Section 4. The logistic map system is chosen since, despite its simplicity, its solutions are complex, varying from fixed points to chaos [38–40], depending on its bifurcation parameter. This system is commonly used to model the growth and decay of a population over time, the presence of turbulence in a fluid, the host–parasite problem, the double pendulum, as well as to describe the chaotic dynamics of phenomena within several research fields [38,41]. Finally, the conclusions are outlined in Section 5.

2. Preliminaries

For later use, the theoretical basis of EKF, an overview of the analytic behaviors framework, and the description of the logistic map employed to exemplify the proposed approach, are detailed.

2.1. Extended Kalman Filter

Consider a class of nonlinear systems described by the deterministic difference equation

$$\boldsymbol{x}_k = \boldsymbol{f}(\boldsymbol{x}_{k-1}), \tag{1}$$

where $\boldsymbol{x}_{k-1}$ represents the state vector, and $\boldsymbol{f}(\boldsymbol{x}_{k-1})$ is a nonlinear vector function that defines the dynamics of the systems. For (1), the outputs described by

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k) + \boldsymbol{v}_k, \tag{2}$$

are considered, where $\boldsymbol{h}(\boldsymbol{x}_k)$ is a function of the state vector, and the random sequence $\boldsymbol{v}_k$ corresponds to WGN normally distributed with $\mathcal{N}(0, \boldsymbol{R}_k)$, where $\boldsymbol{R}_k$ is the covariance matrix of $\boldsymbol{v}_k$ that is used for tuning the EKF.

The EKF, developed by Schmidt et al. [1], estimates the state vector of (1) subject to WGN; this is

$$\boldsymbol{x}_k = \boldsymbol{f}(\boldsymbol{x}_{k-1}) + \boldsymbol{w}_k, \tag{3}$$

where $\boldsymbol{w}_k$ are uncertainties of WGN distributed with $\mathcal{N}(0, \boldsymbol{Q}_k)$, with $\boldsymbol{Q}_k$ being the covariance matrix of $\boldsymbol{w}_k$. To optimally estimate $\boldsymbol{x}_k$, the filter uses the outputs (2) and the knowledge of (3). This filter is frequently found in the literature; see for instance, [42–44]. Since the present work considers (1) instead of (3), the standard EKF becomes

$$\hat{\boldsymbol{x}}_k^p = \boldsymbol{f}(\hat{\boldsymbol{x}}_{k-1}^{\boldsymbol{u}}), \tag{4}$$

$$\boldsymbol{P}_k^p = \boldsymbol{A}_{k-1}\boldsymbol{P}_{k-1}^{\boldsymbol{u}}\boldsymbol{A}_{k-1}^T, \tag{5}$$

$$\boldsymbol{K}_k = \boldsymbol{P}_k^p\boldsymbol{C}_k^T(\boldsymbol{C}_k\boldsymbol{P}_k^p\boldsymbol{C}_k^T + \boldsymbol{R}_k)^{-1}, \tag{6}$$

$$\hat{\boldsymbol{x}}_k^u = \hat{\boldsymbol{x}}_k^p + \boldsymbol{K}_k(\boldsymbol{y}_k - \boldsymbol{h}(\hat{\boldsymbol{x}}_k^p)), \tag{7}$$

$$\boldsymbol{P}_k^u = (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{C}_k)\boldsymbol{P}_k^p, \tag{8}$$

where the traditional EKF equation $\boldsymbol{P}_k^p = \boldsymbol{A}_{k-1}\boldsymbol{P}_{k-1}^{\boldsymbol{u}}\boldsymbol{A}_{k-1}^T + \boldsymbol{Q}_{k-1}$ has been modified to (5). In the above relations, $\hat{\boldsymbol{x}}_k^p$ is the estimation of the state $\boldsymbol{x}_k$, before considering $\boldsymbol{y}_k$; $\boldsymbol{P}_k^p$ is the predicted covariance matrix of the estimation error; $\boldsymbol{K}_k$ is the Kalman gain; $\hat{\boldsymbol{x}}_k^u$ describes, although locally, the optimal estimation of $\boldsymbol{x}_k$; and $\boldsymbol{P}_k^u$ is the updated covariance matrix of the estimation error. $\boldsymbol{R}_k$ is a symmetric and positive definite matrix capturing in its diagonal the variances of the uncertainties $\boldsymbol{v}_k$, while $\boldsymbol{P}_k^u$ and $\boldsymbol{P}_k^p$ are symmetric and positive definite matrices. The superindexes p and u indicate the prediction and update phases. The matrices

$$\boldsymbol{A}_{k-1} = \left.\frac{\partial \boldsymbol{f}(\boldsymbol{x}_{k-1})}{\partial \boldsymbol{x}_{k-1}}\right|_{\hat{\boldsymbol{x}}_{k-1}^u}, \quad \boldsymbol{C}_k = \left.\frac{\partial \boldsymbol{h}(\boldsymbol{x}_k)}{\partial \boldsymbol{x}_k}\right|_{\hat{\boldsymbol{x}}_k^p}$$

come from the Taylor's linear approximations of (1) and (2) around the nominal values $\boldsymbol{x}_{k-1} = \hat{\boldsymbol{x}}_{k-1}^u$, $\boldsymbol{x}_k = \hat{\boldsymbol{x}}_k^p$, and $\boldsymbol{v}_k = 0$ (see [42]). These approximations are

$$\boldsymbol{x}_k \approx \boldsymbol{A}_{k-1}\boldsymbol{x}_{k-1} + \boldsymbol{u}_{k-1} \quad \text{and} \tag{9}$$

$$\boldsymbol{y}_k \approx \boldsymbol{h}(\hat{\boldsymbol{x}}_k^p) + \boldsymbol{C}_k\boldsymbol{x}_k - \boldsymbol{C}_k\hat{\boldsymbol{x}}_k^p + \boldsymbol{v}_k, \tag{10}$$

with $\boldsymbol{u}_{k-1} = \boldsymbol{f}(\hat{\boldsymbol{x}}_{k-1}^u) - \boldsymbol{A}_{k-1}\hat{\boldsymbol{x}}_{k-1}^u$. It should be noted that the EKF (4)–(8) is derived from the standard KF applied to estimate the state vector of (9), given the outputs

$$\boldsymbol{z}_k = \boldsymbol{C}_k\boldsymbol{x}_k + \boldsymbol{v}_k, \tag{11}$$

where $\boldsymbol{z}_k = \boldsymbol{y}_k - \boldsymbol{h}(\hat{\boldsymbol{x}}_k^p) + \boldsymbol{C}_k\hat{\boldsymbol{x}}_k^p$, (see [42]). The idealized initial conditions for the EKF are $\hat{\boldsymbol{x}}_0^u = E[\boldsymbol{x}_0]$ and $\boldsymbol{P}_0^u = \mathrm{var}[\boldsymbol{x}_0]$, where E and var are, respectively, the expectation and the variance operators. The EKF is locally optimal in the sense that locally minimize the Kalman criterion defined as the trace of the covariance of the estimation error, that is

$$\mathrm{trace}(\boldsymbol{P}_k^u) = \mathrm{trace}(E[(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^u)(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^u)^\top]). \tag{12}$$

2.2. Analytic behaviors framework

The analytic behaviors framework is a recent control design scheme for the automated synthesis of analytic nonlinear controllers proposed by [30], and extended in [31–34]. This approach, developed to address control problems, relates the behavior of artificial systems with the principles that govern the behaviors and the learning process in humans. It aims to automate and synthesize the process of acquiring new behaviors, and/or modifying existing ones. Such behaviors are translated into new skills for the system to solve a large class of problems. The general overview of this framework is presented in Fig. 1. The entire behavior of a system can be defined as the *natural* behavior, where three main basis behaviors have been considered and denoted as the *unforced*, the *forced*, and the *learned* behaviors. More behaviors can be introduced such as unmodeled dynamics, external disturbances from the environment, and/or parametric uncertainties, for instance, due to normal wear and tear in the case of physical systems. Additional phenomena can be added if the nature of the system requires it. The *unforced* behavior is defined as the model of the system representing its internal dynamics. This is what is known from the system. This behavior solely depends on initial conditions without applying any input or excitation

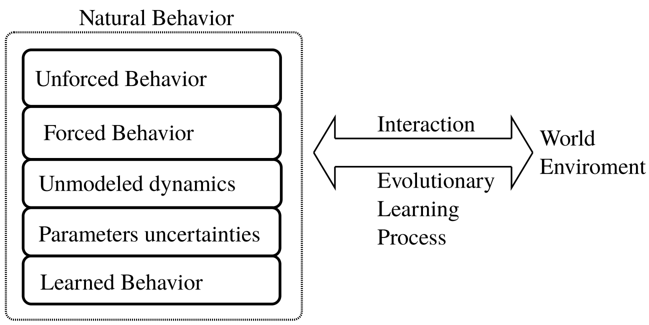


Fig. 1. Analytic behaviors framework.

signal. The *forced* behavior is related to the already acquired behaviors, either from a previous synthesis or by experience. And finally, the *learned* behaviors, produced by an evolutionary process, are dedicated to find nonlinear controllers that extend the capabilities of the system. The Genetic Programming (GP) paradigm is applied as the evolutive engine, since it has the ability to construct solutions using analytic functions.

The GP fits a solution for the task in hand following processes inspired by natural evolution. First, a solution is represented by a syntactic tree structure. Second, the initial population of solutions is randomly created. Third, each solution is evaluated and ranked according to a fitness function. Fourth, a selection method is performed to choose the fittest individuals for reproduction. Fifth, the selected individuals are mated and mutated to produce offspring. The old population is replaced with the new individuals. This cycle is repeated until a desired level of fitness is achieved [45].

In this work, the analytic behaviors framework is applied for the design of evolved versions of EKFs. This proposal solves the problem of estimating the state of a nonlinear first-order dynamical system whose output is corrupted by a noisy signal of unknown covariance value. The extension of the original framework is defined as the introduction of the theoretical formulation of the conventional EKF and a parallel structure of both the true system and the filter is set.

2.3. The logistic map system

The logistic map can be considered as one of the simplest nonlinear, first-order, difference systems, possessing a rich variety of behaviors ranging from stable fixed points to chaos. This system is given as [38]

$$x_k = \alpha x_{k-1}(1 - x_{k-1}). \tag{13}$$

For $0 \leq \alpha < 1$, the solution to this system is $x_k = 0$. Different values of α have different interpretations, for instance when modeling species, using $1 < \alpha < 4$ corresponds to the growth of a species, otherwise the population becomes extinct. Steady-state values are observed in the interval $1 \leq \alpha < 3$, and bifurcations in the form of period-doubling cascade appear from $\alpha \geq 3$ to $\alpha \approx 3.56995$. The chaotic regime appears for values of α greater than 3.56995 and $\alpha \leq 4$. Some islands of stability within this interval are also observed; this phenomenon is called *intermittency*. The complex behavior of this system can be graphically explained using its bifurcation map, presented in Fig. 2, for $1 \leq \alpha \leq 4$. The logistic map system has been widely employed in a vast number of science fields, and it is still attracting the focus of many researchers. For instance, see the works of [38,46–58], to mention a few.

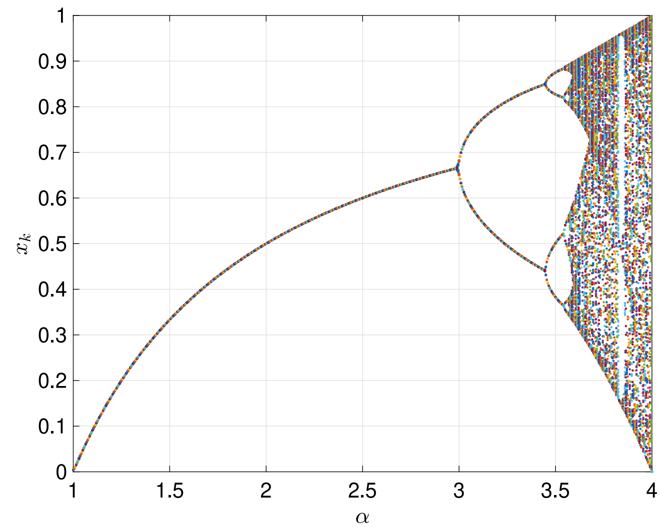


Fig. 2. Bifurcation diagram of the logistic map system (13).

3. Synthesis of analytic behaviors for EEKFs for nonlinear FO-DS

In this section, the proposed methodology for the construction of EEKFs applied to a nonlinear FO-DS is described in a general way. First, the problem statement is defined, and next, each stage of the approach is detailed.

3.1. Problem statement

Consider the EKF (4)–(8), with unknown tuning parameter $\boldsymbol{R}_k$, dedicated to the first-order version of (1) and (2). Given the order of the system, its matrices and the ones from the filter become scalars. The problem is thus defined as the search of a replacement of $\boldsymbol{R}_k$, expressed as a scalar analytic function R^{gp^i}. To solve the problem in question, a novel methodology based on the analytic behaviors approach is developed. This methodology is designed for the search of a set of functions R^{gp^i} that work as replacements of the typically unknown tuning parameter $\boldsymbol{R}_k$. These functions are found by means of evolutionary computation, hence the name Evolved EKF (EEKF). The implemented evolutionary process is guided by a fitness function constructed on the average of the Kalman criterion defined by (12).

3.2. The analytic behaviors approach for EEKF

This work proposes an extension of the analytic behavior-based framework for the construction of EEKFs. In this section, each stage of the methodology is detailed in order to build a suitable solution.

A general layout for the search of analytic solutions considering a nonlinear FO-DS is described in Fig. 3. The nonlinear model of the system and the EEKF are represented by the `NONLINEAR FO-DS`, and the `EVOLVED EXTENDED KALMAN FILTER` blocks, respectively. Note that the EEKF has the same structure as the conventional EKF presented in Section 2, where a function R^{gp^i} is introduced as a solution that substitutes the covariance matrix value R_k. Both systems, denoted by the `NONLINEAR FO-DS` and the `EVOLVED EXTENDED KALMAN FILTER` blocks, are running in parallel, where the EEKF's objective is to optimally estimate the state of the true system in the presence of the uncertainty introduced by the noise v_k.

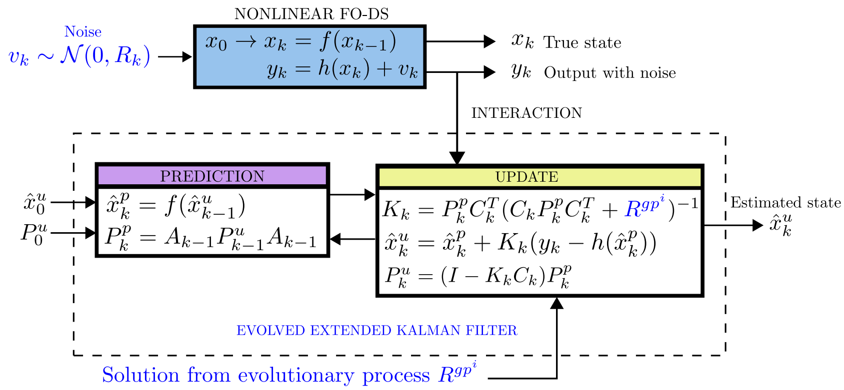


Fig. 3. Layout of the EEKF. The construction of the EEKFs is based on the analytic behaviors framework, that solves the estimation task in the presence of uncertainty introduced by the noise v_k. The solutions R^{gp^i}, $i = 1, \ldots, m$ are optimized analytic functions derived from an evolutionary process applied to the filter.

The first step of this methodology is the definition of the *basis* behaviors for both systems, where the dynamics and the interactions between them and the environment are described. The *basis* behaviors for the system's nonlinear model, denoted as NONLINEAR FO-DS in Fig. 3, are summarized as follows.

1. The *natural* behavior is defined by the output of the system, where initial conditions and interactions with the environment are considered. This is given by the output y_k defined by the state x_k, and corrupted by additive noise, v_k. The uncertainty problem is given as the noise v_k being characterized by WGN with unknown covariance value R_k.
2. The *unforced* behavior denotes the output of the system when only the internal dynamics are considered. Thus, the state x_k, of the nonlinear first-order dynamical model (1), subject to some initial condition x_0, defines the *unforced* behavior of the true system.
3. The *forced* behavior of the system is defined by the output affected by external inputs, either from the environment or by interacting with other systems. Since the system is corrupted by an external input, denoted as the noise v_k, it is assumed that, in this setup, the *forced* behavior corresponds to the effect of the noise signal v_k over the output state x_k.

Notice that the *learned* behavior is not defined for the nonlinear system since its states are the desired target for the EEKF to achieve. Hence, the *learned* behavior is only defined for the EEKF through a learning stage implemented by an evolutionary process. The objective is that the EEKF recovers optimally the system state.

The EVOLVED EXTENDED KALMAN FILTER block is composed of the iterative procedure between the PREDICTION and the UPDATE stages. The *basis* behaviors for this block are described below.

1. The *unforced* behavior is denoted as the estimated outputs $\hat{x}_k^p$ and P_k^p from the PREDICTION block of the filter (see Fig. 3). These outputs depend on the estimated initial conditions $\hat{x}_0$ and P_0.
2. The *forced* behavior of the EEKF is given as the response to the effect of external inputs. This is the interaction with the true system by means of the input y_k. This interaction is described by (7), and it is used in the UPDATE stage of the filter (see Fig. 3).
3. Finally, our aim is the construction of an EEKF using a suitable function R^{gp^i}, whose dynamics converge to the states of the system's nonlinear model (1)–(2). Thus, the EEKF must *learn* to deal with the noise uncertainty, and to adapt in order to fulfill the estimation task. An optimization based on the GP paradigm is introduced to derive the analytic function R^{gp^i}, to be used in place of the R_k value. The restriction is that such function is only allowed to use information from the EKF and the noisy output, y_k, from the true system. This way, the found R^{gp^i} is a solution that will offer a straightforward implementation in a practical application. Note that, by using this methodology, it is possible to obtain hundreds or thousands of solutions R^{gp^i} that satisfy the restrictions with acceptable performance. In this case, the superindex i refers to a particular solution, from a set of m possible choices.

3.2.1. The learning stage

The description of the synthesis of suitable solutions, applying the GP paradigm, is given next. These solutions generate the *learned* behaviors of the EEKFs, and are meant to replace the unknown covariance value, R_k, in order to cope with the uncertainty. Based on the structure of the EEKF, note that the filter's *learned* behaviors propagate through the UPDATE stage shown in Fig. 3; they first appear in Eq. (6) in the computation of the filter's gain K_k. This gain is later applied for the calculation of the updated estimated state $\hat{x}_k^u$ in (7), and for the updated value of the covariance P_k^u in (8). Then, these values are iteratively used for the new prediction.

The computational description of the *learning* process, based on GP, is depicted in the flowchart in Fig. 4. This process requires the setting of

1. the numerical models of the nonlinear system and of the EEKF, as shown in Fig. 3;
2. the definition of the search space for the synthesis of the solution R^{gp^i};
3. the fitness function that evaluates the performance of each possible solution generated by the GP;
4. the setup of the scenarios or cases where the performance will be tested; and
5. the tuning stage of the GP parameters.

First, the definition of the numerical models is based on Eqs. (1)–(2) for the nonlinear system model, and on (4)–(8) for the EEKF. Second, the search space is defined as the pool of mathematical functions and arguments that can be used for the construction of the analytic solution R^{gp^i}. These expressions are known, respectively, as the Functions and Terminals sets used by the GP. The distinctive feature of the analytic behaviors framework is the use of an encoding of analytic nature. Arithmetic, trigonometric, and other special functions can be selected for the

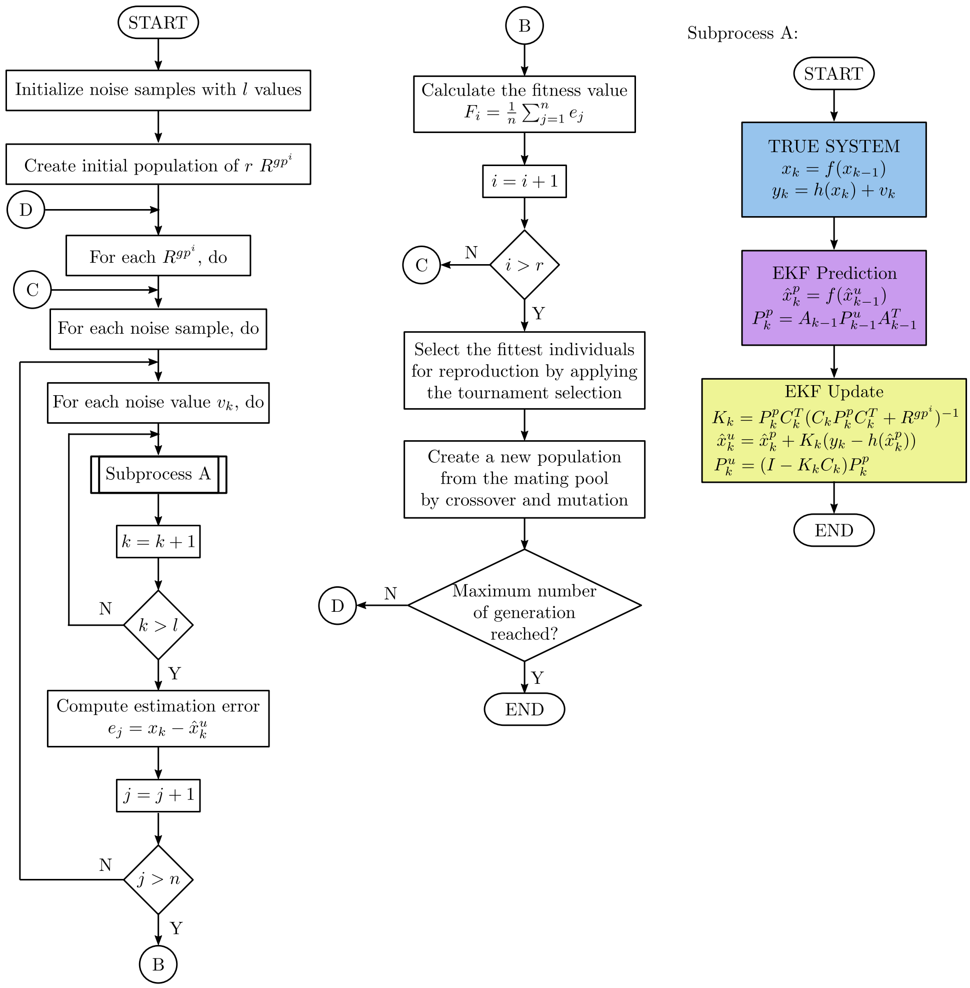


Fig. 4. Evolutionary process for the synthesis of optimized analytic functions R^{gp^i}, $i = 1, \ldots, m$ as estimations of the unknown covariance value R_k for the additive noise v_k. The analytic functions are applied into the structure of EEKFs aiming to solve the estimation task of a nonlinear FO-DS.

Functions set, see Table 1. From the nonlinear system model, only the noisy output y_k is selected to be a part of the set of Terminals, since it is the only piece of information available through measurements. The other arguments used, as part of the Terminals set, are variables from the EEKF, and a noise estimation error defined as $e_k^p = y_k - \hat{x}_k^p$; i.e. the difference between the noisy system output and the predicted estimated state, as well as its time derivative, $\dot{e}_k^p$. The variables that compose the set of Terminals are presented in Table 2.

The third setting corresponds to the design of a fitness function that guides the evolutionary process by ranking the performance of candidate solutions as replacement functions R^{gp^i} for the unknown value of R_k in the filter. The proposed methodology takes advantage of the Kalman criterion, which measures the local optimality of the filter. This criterion is defined as in Eq. (12), where the covariance is reduced to the variance and the trace operator is not needed, since we deal with first-order systems. Thus, the fitness function is defined as

$$F = \frac{1}{n}\sum_{j=0}^{n} e_{k_j}. \tag{14}$$

where $e_{k_j} = (E[(x_k - \hat{x}_k^u)^2])_j$, and n is the number of noisy samples $\boldsymbol{v}_k$ affecting the system's measurement $\boldsymbol{y}_k$. The evolutive algorithm performs an optimization process, which minimizes the fitness function (14). Hence, the definition of the optimization problem is expressed as

$$\min_{e_{k_j}} \quad \frac{1}{n}\sum_{j=0}^{n} e_{k_j}, \tag{15}$$

$$\begin{aligned} s.t. \quad & \boldsymbol{x}_k = \boldsymbol{f}(\boldsymbol{x}_{k-1}), \\ & \hat{\boldsymbol{x}}_k^u = \hat{\boldsymbol{x}}_k^p + \boldsymbol{K}_k(\boldsymbol{y}_k - \boldsymbol{h}(\hat{\boldsymbol{x}}_k^p)), \end{aligned} \tag{16}$$

where $\boldsymbol{x}_k$ denotes the state of the system, and $\hat{\boldsymbol{x}}_k^u$ corresponds to the local optimal estimation of $\boldsymbol{x}_k$, as established by the theory of the conventional EKF. Note that x_k, from the nonlinear model, is only used during the offline search performed by the evolutionary process. It is not used for the construction of the replacement functions R^{gp^i} as established by the defined set of Terminals listed in Table 2. Thus, the obtained solutions are completely independent of x_k.

A characteristic to be considered is that the number of selected scenarios has an impact on the increment of the computational load during the implementation of the evolutionary process. In addition, it is recommended to set the parameters of the system such that their dynamics provides valuable information to the learning stage.

The tuning stage of the GP process is based on a trial-and-error approach. Commonly, a large number of generations and individuals is used by the GP's evolutive process. However, an important feature of the analytic behaviors methodology is that the number of generations and individuals required to find suitable solutions

Table 1
Functions used in the evolutionary process to build the replacement functions R^{gp^i}.

ID	Expression	Definition	ID	Expression	Definition
1	$(\cdot)^2$	Square	20	$sc(\cdot)$	Secant
2	$(\cdot)^3$	Cubic	21	$ct(\cdot)$	Cotangent
3	$\mathrm{sr}(\cdot)$	Square root	22	s_i	Inverse sine
4	$s_{hi}(\cdot)$	Inverse hyperbolic sine	23	c_i	Inverse cosine
5	$c_{hi}(\cdot)$	Inverse hyperbolic cosine	24	$\mathrm{atanr}(\cdot)$	Inverse tangent of the real part of the argument
6	$t_{hi}(\cdot)$	Inverse hyperbolic tangent	25	$e^{(\cdot)}$	Exponential
7	$cs_{hi}(\cdot)$	Inverse hyperbolic cosecant	26	$\ln(\cdot)$	Natural logarithm
8	$sc_{hi}(\cdot)$	Inverse hyperbolic secant	27	$\mathrm{Re}(\cdot)$	Real part of the argument
9	$ct_{hi}(\cdot)$	Inverse hyperbolic cotangent	28	$\|\| \cdot \|\|$	Euclidean norm
10	$s_h(\cdot)$	Hyperbolic sine	29	$\mathrm{abs}(\cdot)$	Absolute value
11	$c_h(\cdot)$	Hyperbolic cosine	30	$\mathrm{sgn}(\cdot)$	Signum function
12	$t_h(\cdot)$	Hyperbolic tangent	31	$+$	Addition
13	$cs_h(\cdot)$	Hyperbolic cosecant	32	$-$	Subtraction
14	$sc_h(\cdot)$	Hyperbolic secant	33	$/$	Division
15	$ct_h(\cdot)$	Hyperbolic cotangent	34	$*$	Multiplication
16	$c(\cdot)$	Cosine	35	$(\cdot)^{(\cdot)}$	Exponentiation
17	$s(\cdot)$	Sin	36	$\max(\cdot,\cdot)$	Maximum
18	$t(\cdot)$	Tangent	37	$\min(\cdot,\cdot)$	Minimum
19	$cs(\cdot)$	Cosecant			

Table 2
Terminals used by the GP in the learning stage.

ID	Term	Description	ID	Term	Description
1	$\hat{x}_k^u$	Updated estimated state	5	e_k^p	Noise estimation error
2	$\hat{x}_k^p$	Predicted estimated state	6	$\dot{e}_k^p$	Time derivative of noise estimation error
3	$\dot{\hat{x}}_k^p$	Time derivative of the predicted estimated state	7	$R_{k-1}^{gp^i}$	Past value of the ith proposed solution
4	y_k	Noisy measurement of the nonlinear system			

for a well formulated problem is reduced. This implies that, in general, optimal solutions can be found with a relatively small number of generations, and the increment of this number will only result in the growth of the syntactic tree (this is, an increase in the size of the solutions' expressions). Since, intrinsic to the methodology, a lower number of generations are required, the proposed approach reduces the computational load when compared to the traditional applications of the GP approach reported in the literature, [59]. A solution can be considered as suitable if its fitness is comparable to a target value. For the addressed problem in this work, this target value can be established in terms of a suitable score calculated with respect to the optimality Kalman criterion.

3.2.2. Post-processing stage

The post-processing stage is composed of two tasks. The first, is the statistical analysis of the learning stage performed by the GP. Notice that, an evolutive process is a non deterministic technique, therefore, it is a good practice to realize several runs of the learning stage, with the objective of obtaining a measurement of its average performance. Thus, the statistical analysis includes the performance of the best fitness, the average best fitness, and the average fitness over a set of runs, as well as their standard deviation. The score for the diversity of the solutions, and the frequency of appearance of each function and variable, taken respectively from the defined sets of Functions and Terminals, can be further analyzed.

Once the analysis of the evolutionary process is finished, the solutions from all the runs are ranked according to their performance with respect to the fitness criterion F, see Eq. (14). The found optimal analytic functions R^{gp^i}, $i = 1, \ldots, m$ constitute a large set of generated learned behaviors in the EEKF that address the formulated problem. The complexity of the solutions found by the GP can be constrained by specifying the maximum permitted depth of the syntactic trees. A decreased depth in the syntactic trees will also result in a reduced search space. A compromise between the complexity of the solutions and the size search space must be reached. Further mathematical analysis of the found solutions can also be done as part of the post-processing stage to simplify and reduce the size of large expressions.

The post-processing stage is important since it is the backbone of the process of selecting a particular solution based on the suitability of its features for a particular system or setup. For instance, the solutions can be classified according to

- having a similar or better fitness value, F, than a traditional or ideal setup;
- their structural complexity or the size of the expressions;
- the variables used, either from the filter or the noisy measurements of the true system;
- the geometrical or mathematical properties of the functions used in the expressions;
- the hardware and computational capabilities available for the implementation of the EEKFs.

The above are some practical assumptions that can constrain the use of certain solutions in specific problems and applications.

3.2.3. Discussion

From the proposed methodology, the following remarkable features can be outlined.

First, the core of the analytic behaviors framework is the formulation of an emulation of the natural learning process [30,32,34]. This means that it is a scalable, iterative method, where new desirable characteristics can be incrementally added to the solutions. Translated into our uncertainty problem, this implies that the analysis of the statistical results can give insight towards improving the scores of the solutions generated by the evolutionary process. This can be realized through the introduction of either a reshaped fitness function, a modified set of Functions, and/or a manual tuning of the GP parameters.

Second, the proposed methodology is an optimization strategy applied to the EKF. Even though, according to Kalman theory, the EKF is an optimal estimation of a process under zero-mean WGN, in practice, this type of noise is not truly achieved. The WGN, either in simulation or experimentally, rarely is a true zero-mean noise; there are always small deviations from the mean. This slightly diminishes the optimal estimation performed by the EKF. The proposed methodology uses the output measurement's affected by the flawed zero-mean WGN to minimize the Kalman cost function and to estimate the state. Hence, the EEKF can lead to results where its equals or exceeds the performance of the EKF.

Third, the generality and robustness of the solutions discovered by the proposed methodology can be numerically evaluated. The tests can include the introduction of model uncertainty, evaluation of local stability conditions considering an initial estimation error larger than that used during the learning stage, and the variation of the initial conditions of the true system.

In the following section, the implementation of the proposed methodology is exemplified for the construction of an EEKF for a particular complex, highly nonlinear, first-order dynamical system. The system under study is the logistic map described in Section 2.3.

4. A motivating example: the logistic map system

The logistic map system is a nonlinear, first-order dynamical system selected to exemplify our methodology due the richness of its behaviors, which range from stable fixed points to chaos. This particular system (13) can be rewritten in the form (1)–(2) with f and h given as

$$f(x_{k-1}) = \alpha\, x_{k-1}(1 - x_{k-1}), \quad h(x_k) = x_k + v_k, \tag{17}$$

where $f(x_{k-1})$ is the logistic map function, and $h(x_k)$ is the logistic map state corrupted by a noise signal v_k. Then, the construction of the EKF (4)–(8) for (17) is specified with

$$A_{k-1} = \alpha\,(1 - 2\,\hat{x}^u_{k-1}), \quad C_k = 1. \tag{18}$$

The parameter $\alpha = 3.7$ is selected such that the system operates in the chaotic regime; this is done to enrich the dynamics of the system during the learning stage. The initial condition $x_0 = 0.6$ is also considered during this process. In addition, a set of 50 training WGN samples v_k of amplitude $a = 0.3$ are generated, these are generated with the randn MATLAB function, which creates normally distributed random numbers with "zero" mean. Each v_k, with 400 elements each, defines a scenario for the learning process. The initial conditions for the EEKFs are chosen as $\hat{x}^u_0 = x_0 + 0.1$ and $P_0 = 0.01$.

The search space is defined by the set of Terminals and Functions to be used in the construction of the analytic functions R^{gp^i}, which is an estimation of the unknown covariance R_k. The Terminals set, described in Table 2, is defined as per the methodology in Section 3, while the Functions set is listed in Table 1, where only the real values of the functions are considered. The fitness function that evaluates the performance of each individual, as proposed by the learning stage, is defined as in (14). Finally, the parameters used to define the learning process are indicated in Table 3.

Table 3
Parameters use in the evolutionary process.

Parameter	Value
Runs	30
Number of generations	100
Population size	400
Crossover rate	80%
Mutation rate	20%
Maximum tree depth	12
Sampling	Lexicographic
Elitism	Keep best

4.1. Statistical analysis of the learning process

The first post-processing stage consists in the statistical analysis of the minimization process based on the minimization of the standard optimality criteria of the EKF, defined by the fitness function, F, as in (14). Given that the GP is a non-deterministic process, a criterion to determine its convergence is to repeat the method several times. The validity of our methodology is based on the execution of 30 runs using the same settings for each one (see Table 3).

The statistic results from the evolutionary process applied to the system defined by Eqs. (17) are presented in Fig. 5. The best fitness average, its standard deviation, and the best fitness over the 30 runs are depicted in Fig. 5(a). From this Figure, two features of the optimization process are remarked. First, there are not significant changes after the 60th generation. Implying that a local optimal score for the proposed setting has been found; in consequence, increasing the number of generations will not significantly improve the fitness score F of the solutions. Second, similar performance is observed around the 20th generation, according to the evaluated EEKF's optimal criterion given in (14), in comparison with the conventional EKF using the covariance value R_k. This means that the problem is well formulated, since the results of the found solutions are comparable to the ideal setting, where the covariance value R_k is known.

An important feature of this methodology is the generation of many solutions, and the differences between them is referred to as the diversity of the solutions. An analysis over the diversity of the solutions with respect to the number of generations is shown in Fig. 5(b). There is a slight reduction in the diversity along the evolutionary process, see Fig. 5(b), but a high score of the fitness is preserved at the last generation (see Fig. 5(a)). The increment in size of the expressions, i.e., complexity, can be seen through a graph of depth and number of nodes of the syntactic tree that defines each solution. An analysis of the complexity along the runs executed by the GP is shown in Figs. 5(c) and 5(d).

The analysis of the variables and functions used by the found solutions is based on their usage rate. This is shown in Fig. 6. Using the indexes from Tables 2, the three terminals that are used the most are the noisy measurement of the nonlinear system y_k, followed by the time derivative of the proposed error function $\dot{e}^p_k$ and the predicted estimated state $\hat{x}^p_k$. Similarly, according to the IDs of the Functions set, listed in Table 1, the top four most frequently used functions, in descending order, are the maximum, the square root, the exponentiation, and the hyperbolic cosine functions. As part of the learning process, the GP discarded some functions from the set, due to their poor performance with respect to the optimal criterion of fitness. A new evolutionary process could be proposed by redefining the Functions set if a new focused GP process is required or desirable.

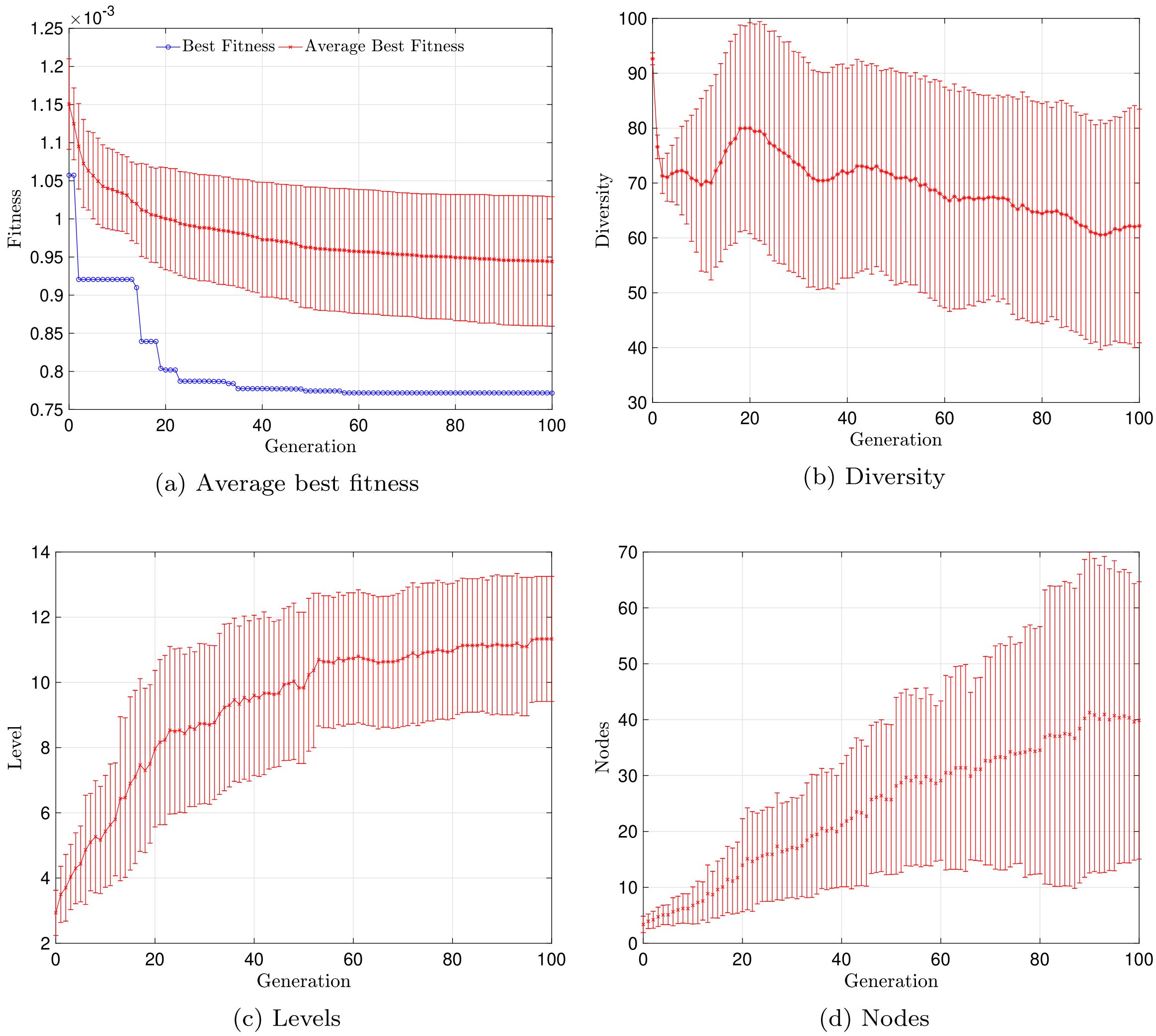


Fig. 5. Statistics of the evolutionary GP process.

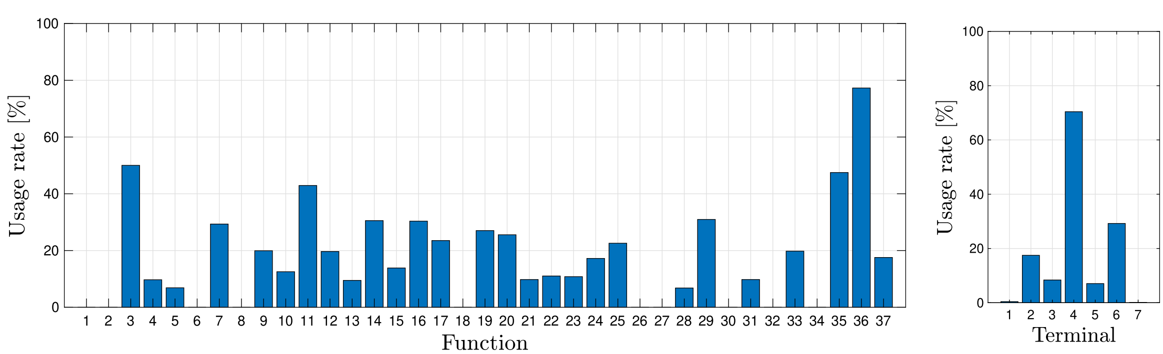


Fig. 6. Usage rate of Functions and Terminals by the evolutionary process within the discovered solutions. The indexes of the Terminals and Functions sets are defined as shown in Tables 1 and 2, respectively.

4.2. The discovered behaviors

The proposed methodology applied to the logistic difference equation system (13) and its output, gave a set of 2649 suitable analytic functions to be used in place of the covariance value R_k. Such functions are those having a fitness score less than 1×10^{-3}. This threshold is chosen according to the actual fitness of the conventional EKF ($F = 0.001544$). For every solution, the fitness F is computed using the same noisy signal v_k, see (14).

From the large set of solutions found by the analytic behaviors approach, described in Section 3, a list of ten solutions R^{gp^i}, $i = 1, \ldots, 10$, have been selected due to their interesting properties, in terms of their performance, domain, structure, and the variables used, (see Table 4). Two tests were performed by randomly varying the WGN samples affecting the system's output. The filters are tested for noise samples different from those used in the learning stage. The results are shown in the fourth and fifth columns of Table 4. The fourth column describes the average error variance for a first test with 10 different noise samples. The size of

Table 4
From the evolutionary process, using parameters $\alpha = 3.7$, $a = 0.3$, $x_0 = 0.6$, $\hat{x}_0^u = x_0 + 0.1$ and $P_0 = 0.01$, a set of ten solutions R^{gp^i} are shown with their respective training score, as well as their performance under two different scenarios, Test 1 and Test 2. The performance of the conventional EKF is also computed for comparison purposes. The scores are calculated using the fitness function (14) and shown with a scaling factor of 1×10^{-3} units.

Sol	Expression	Training	Test 1	Test 2
R^{gp^1}	$\max(\max(\mathrm{sr}(\|\|c_h(\mathrm{sr}(c_h(s(sc_h(ct_{hi}(e_k^p))))))\|\|),$ $\mathrm{sr}(c_h(\mathrm{sr}(\mathrm{sr}(\mathrm{sr}(c_h(\mathrm{sr}(sc_h(ct_{hi}(e_k^p)))))))))),$ $\max(sc_h(ct_{hi}(\|\|\mathrm{Re}(ct_{hi}(e_k^p))\|\|)),$ $\|\|c_h(\mathrm{sr}(c_h(s(sc_h(ct_{hi}(\mathrm{Re}(ct_{hi}(e_k^p))))))))\|\|))$	0.722	0.728	0.941
R^{gp^2}	$\max(\max(sc_h(ct_{hi}(\|\|\mathrm{Re}(e_k^p)\|\|)),$ $\|\| c_h(c_h(s(sc_h(ct_{hi}(e_k^p)))))\|\|), sc_h(\hat{x}_k^p))$	0.802	0.723	0.983
R^{gp^3}	$c_h(ct_h(ct_h(c_h(c_h(y_k)^{y_k})))^{y_k} + c_{hi}(R_{k-1}^{gp^3}))$	0.907	1.740	1.345
R^{gp^4}	$sc_h(ct_{hi}(e_k^p))$	0.921	0.730	0.962
R^{gp^5}	$\max(t_h(\mathrm{sr}(ct_{hi}(y_k))), s_i(e^{ct_{hi}(y_k)}))$	0.922	2.443	1.421
R^{gp^6}	$s_i(e^{ct_{hi}(y_k)})$	0.925	2.486	1.430
R^{gp^7}	$\mathrm{abs}(ct(\dot{e}_k^p)) + \mathrm{abs}(\mathrm{abs}(\mathrm{abs}(c_i(\mathrm{abs}(\dot{e}_k^p)))))+$ $\mathrm{abs}(c_i(ct(\mathrm{abs}(\mathrm{abs}(c_i(\mathrm{abs}(ct(\dot{e}_k^p)))))) + c_i(\hat{x}_k^p)))$	0.92	1.705	1.260
R^{gp^8}	$e^{ct_{hi}(\mathrm{atanr}(t_h(\max(\hat{x}_{k-1}^u, s_{hi}(sc(y_k))))))}$	0.941	1.666	1.366
R^{gp^9}	$\frac{cs_h(s_h(\hat{x}_{k-1}^u))}{\|\|\|\| \dot{e}_k^p \|\|\|\|}$	0.961	1.261	1.264
$R^{gp^{10}}$	$c_h(\max(\hat{x}_k^p, ct_{hi}(t_h(\mathrm{sr}(c_h(y_k))))))$	0.966	1.653	1.365
R_k	$\mathrm{var}(v_k)$	1.544	1.834	1.335

this set corresponds to the 20% of the number of noise samples used in the training stage. From the observed results, a second test is presented with a larger number of noise samples. Hence, the fifth column shows the average error variance value taking into account 500 new different noise samples. In both tests, the samples were randomly generated with the randn MATLAB function. These tests were performed using the same parameters as in the learning stage, for the logistic map, these are $\alpha = 3.7$, $a = 0.3$, $x_0 = 0.6$, $\hat{x}_0^u = x_0 + 0.1$ and $P_0 = 0.01$.

Referring to Table 4, the following can be observed. The solution R^{gp^1} has the largest expression but it is the function with the best fitness F from the whole evolutionary process. All the solutions are listed as they were synthesized by the learning stage. Several expressions can be simplified by inspection, while others may require further mathematical analysis. Notice that solutions R^{gp^1} and R^{gp^4} only rely on the value of the error, e_k^p. Such solutions can be compared to R^{gp^2} and $R^{gp^{10}}$ since they consider y_k and $\hat{x}_k^p$, which define this error. The noisy measurements y_k and the updated estimated state $\hat{x}_k^u$ are considered by R^{gp^8}. The time derivative of the state estimation error, $\dot{e}_k^p$, is used in R^{gp^7} and R^{gp^9}. In the first one, the derivative of the error is paired with the predicted estimated state $\hat{x}_k^p$, and in the second one, with the updated estimated state $\hat{x}_k^u$. Solution R^{gp^9} possess a singularity point in $\dot{e}_k^p = 0$ that never occurred during the learning process. The expressions that only use the noisy measurements y_k, and no other variable, are R^{gp^5} and R^{gp^6}. This may be an indicative of solutions that are specific for the logistic map system and/or for the particular scenarios used in the learning stage. A recursive function is identified in solution R^{gp^3} as it relies on the values of the noisy measurements y_k and on the past value of R^{gp^3}, this is, $R_{k-1}^{gp^3}$. Solutions R^{gp^3}, R^{gp^4}, R^{gp^7}, R^{gp^8}, and $R^{gp^{10}}$ can be straightforwardly concluded as positive definite, since functions $\cosh(\cdot)$, $\mathrm{sech}(\cdot)$, $\mathrm{abs}(\cdot)$, and $e^{\mathrm{acoth}(\cdot)}$ are positive by definition. Solution R^{gp^2} is also a positive function, since it switches between the maximum values of two positive valued functions: the norm and the $\mathrm{sech}(\cdot)$ functions. Thus, the fulfilling of $R_k > 0$ condition of the EKF can be easily concluded for all of them. On the other hand, solutions R^{gp^1}, R^{gp^5}, R^{gp^6} and R^{gp^9} require further numerical and mathematical analysis to test if the $R_k > 0$ condition holds globally, if they constitute local solutions for the problem, or if they are particular solutions for the logistic map system under study. In addition, piece-wise analysis may be required for some of the found solutions.

Fig. 7 shows the error variance for each noise sample obtained from Tests 1 and 2. For clarity purposes, only 4 of the 10 selected solutions have been plotted. These include the solution with the best overall performance, solutions with low structural complexity, and solutions that are interesting due to the functions and variables they use. Fig. 7(a) shows the results for Test 1, while Fig. 7(b) presents the error variance obtained from Test 2. A direct comparison against the performance of R_k shows that the performance of the found functions is better, or at least similar, to that of R_k.

4.3. Validation stage: numeric analysis of performance robustness

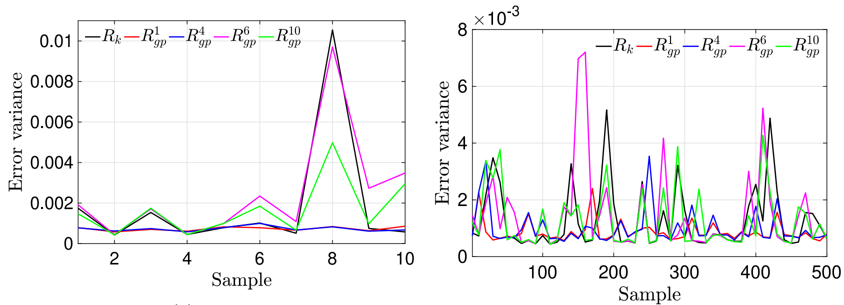
The performance of the found solutions are now numerically evaluated on scenarios different from those used in the learning stage. The performance robustness of the EEKFs using solutions

$$R^{gp^4} = \mathrm{sech}(\coth^{-1}(e_k^p)), \tag{19}$$

$$R^{gp^6} = \sin^{-1}(e^{\coth^{-1}(y_k)}), \tag{20}$$

$$R^{gp^{10}} = \cosh(\max(\hat{x}_k^p, \mathrm{arcoth}(\tanh(\sqrt{\cosh(y_k)})))), \tag{21}$$

from Table 4, are evaluated and compared against the conventional EKF. Recall that only the real domain of the solutions is considered. Function R^{gp^6} may be a particular solution as it corresponds to a positive valued function that is zero in the interval $(-1, 1)$ of the argument. On the other hand, the hyperbolic secant and the hyperbolic cosine functions from R^{gp^4} and $R^{gp^{10}}$, respectively, are nonzero positive valued functions by definition. These features of the studied solutions fulfill the required condition $R^{gp^i} > 0$ that guarantees the internal stability of the EEKF. These solutions have been numerically verified to satisfy this condition, in all of the testing scenarios. The test settings to evaluate the robustness are summarized as follows.



(a) Test 1

(b) Test 2

Fig. 7. Error variance for each noise sample in Test 1 7(a) and Test 2 7(b). In Fig. 7(a) note that the highest fluctuation in sample 8 is due to the natural response of the filter to that particular noise sample. Despite this, the performance produced by the conventional EKF is not as good as the one provided by our proposal.

Table 5
Numeric evaluation of the robustness performance of the EEKF with R^{gp^4}, R^{gp^6}, and $R^{gp^{10}}$ compared with the conventional EKF using the real covariance matrix R_k. The minimum average error variances are indicated in bold. All values have a factor of 1×10^{-3} units.

x_0		α					
		3.5	3.6	3.7	3.8	3.9	4.0
0.50	R_k	**0.1099**	**0.8621**	1.4765	1.0347	**0.8590**	**0.9619**
	R^{gp^4}	0.1118	0.9759	**1.1410**	**0.8469**	1.1825	1.0818
	R^{gp^6}	0.1103	0.8813	1.3832	2.2463	2.0795	0.9832
	$R^{gp^{10}}$	0.1103	0.9390	1.5846	2.9782	1.0864	0.9622
0.55	R_k	**0.1106**	1.0025	1.9989	1.6892	1.9363	2.3064
	R^{gp^4}	0.1167	1.2484	**0.8240**	1.9203	**1.4550**	**1.0421**
	R^{gp^6}	0.1109	1.1118	2.8391	**1.1616**	1.9469	1.9139
	$R^{gp^{10}}$	0.1109	**0.9986**	2.9489	1.5768	2.0586	1.6087
0.60	R_k	**0.1089**	0.8558	1.9137	**0.9789**	1.2451	1.5855
	R^{gp^4}	0.1230	0.8664	**1.2150**	1.0868	1.1863	1.0563
	R^{gp^6}	0.1092	0.9652	2.5609	2.0708	1.2319	**0.9786**
	$R^{gp^{10}}$	0.1093	**0.8340**	1.7277	2.3314	**1.1548**	1.4326
0.65	R_k	**0.1215**	**0.8795**	2.2495	1.1745	**1.6433**	1.7116
	R^{gp^4}	0.1439	0.9881	**1.0512**	**0.8851**	1.6835	**1.2044**
	R^{gp^6}	0.1218	1.3189	2.8768	1.9720	1.7553	1.8738
	$R^{gp^{10}}$	0.1218	0.9454	3.2340	1.3722	1.7570	1.6578
0.70	R_k	**0.1117**	**0.7186**	2.0154	**0.9697**	1.4833	1.5496
	R^{gp^4}	0.1894	0.8806	**1.4192**	1.4488	**1.1337**	**0.9419**
	R^{gp^6}	0.1121	0.8230	2.8065	2.3579	2.3279	1.6800
	$R^{gp^{10}}$	0.1121	0.8080	2.4548	2.2542	1.9687	1.7466

(a) Model uncertainties are introduced in the form of variations in α. For the learning process, this parameter was set to $\alpha = 3.7$. For the performance testing, this value is ranged through the chaotic regime $\alpha = \{3.5, 3.6, 3.7, 3.8, 3.9, 4.0\}$.

(b) The Local stability of the EEKF is evaluated by considering larger initial estimation errors $x_0 - \hat{x}_0^u$ between the real system and the filter. The initial estimation error for the learning stage was set at -0.1 since $x_0 = 0.6$ and $\hat{x}_0^u = x_0 + 0.1$ were selected. For the numerical study, the initial estimation error is set to -0.2, i.e., $\hat{x}_0^u = x_0 + 0.2$.

(c) The dynamics of the system within the chaotic regime is tested by considering variations in its initial conditions. The robustness of the filter is tested with variations in the logistic map's initial conditions, operating in chaotic regime with $x_0 = \{0.50, 0.55, 0.60, 0.65, 0.70\}$, in contrast to $x_0 = 0.6$, which was the initial condition used in the learning stage.

The robustness is assessed by comparing the average error variance, (14), of the EEKFs using R^{gp^4}, R^{gp^6}, $R^{gp^{10}}$, and the EKF with the conventional (known) R_k. For each combination of x_0–α, the system is simulated with the same noise samples as in Test 1, see figure 8. For every scenario, it has been numerically verified that the condition $K > 0$ for the Kalman gain holds, which means that the estimation is not relying solely on the predicted estimated state but also on the noisy measurements, y_k, from the system. It is important to note that while the conventional EKF assumes knowledge about the covariance value R_k, the proposed EEKFs do not possess any information about it.

Table 5 shows a quantitative comparison of the average error variances between the chosen solutions, and the traditional R_k. The best performances are highlighted in bold for each α and for each initial condition x_0. In all cases, the evolved solutions have similar or better performance than the conventional EKF. Note that for $\alpha = 3.7$, R^{gp^4} offers minimum average error variance, since the learned behavior was trained for this value of α.

The robustness analysis confirms that the EEKFs provide an accurate estimation of logistic map system's state, regardless

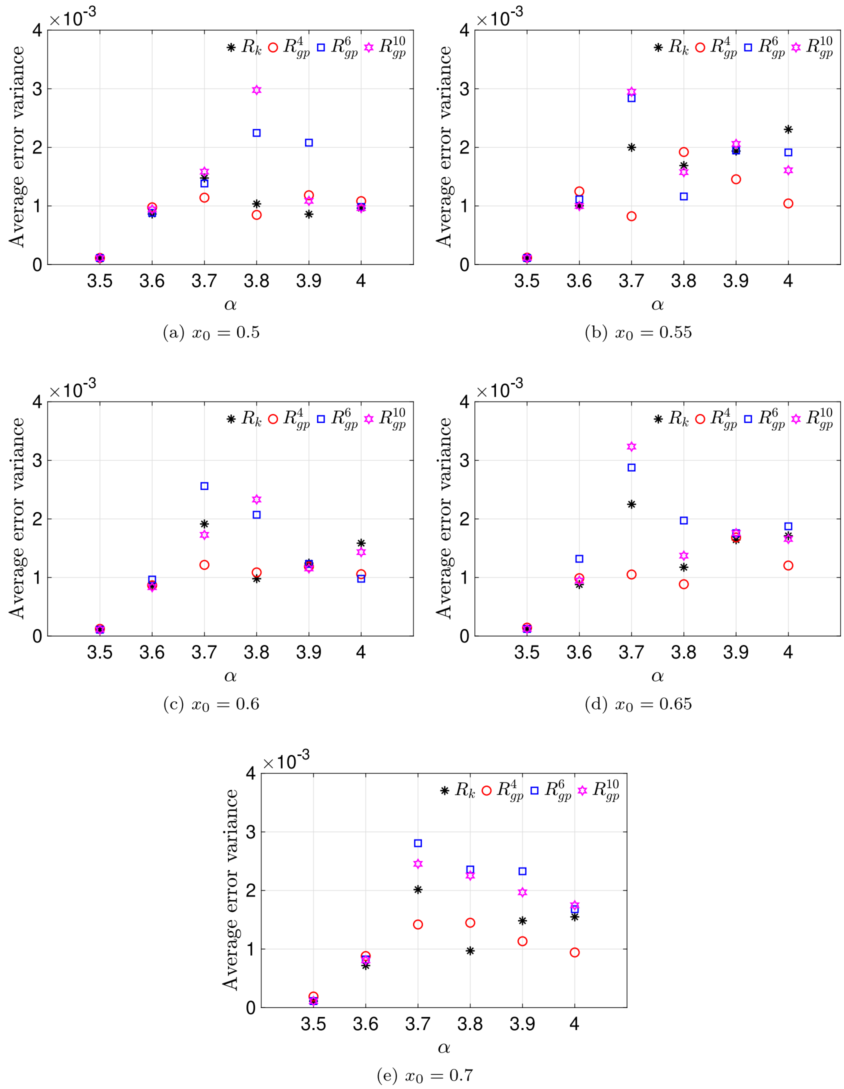


Fig. 8. Robustness analysis of the discovered solutions against the conventional EKF. Each solution is tested with five different initial conditions and for six different values of parameter α. The average error variance is shown in each case.

of the unknown measurements noise covariance. These results are numerically comparable to the conventional EKF using the known covariance matrix of the noise. The strength and capabilities of the proposed synthesis, using the analytic behaviors approach, applied to a particular complex, highly nonlinear, first-order dynamical system has been shown.

Remark 1. As expected, there are cases where the performance of the EEKFs exceed the performance of the conventional EKF. In our case, the WGN produced in software, MATLAB, does not comply with the zero mean condition.

Remark 2. It has been numerically evaluated that studied solutions R^{gp^4}, R^{gp^6}, and $R^{gp^{10}}$ fulfill the required internal stability condition $R^{gp^i} > 0$. In addition, $K_k > 0$ has also been numerically verified, which indicates that the EEKF relies on the measurements of the real system, despite the assumption of absence of process noise.

4.4. Discussion

In the previous subsections, the proposed methodology is illustrated to estimate the state of a logistic map system in its chaotic regime under noisy measurement conditions. Through the application of the analytic behaviors framework, 2649 R^{gp^i}'s, in the form of analytic functions, were found. These functions can be used in place of the EKF measurements noise covariance matrix, R_k; this results in an EEKF. To build the R^{gp^i} functions, the Genetic Programming technique was used. Then, for the training stage, a data set of 50 normally distributed random noise samples, and a fixed set of system parameters (initial conditions and describing parameters) were proposed. Once this learning stage is over, some R^{gp^i}'s were analyzed in order to study the robustness of the methodology. Thus, the selected R^{gp^i}'s were tested with a new set of 10 different normally distributed random noise samples (Test 1). This process allows us to compare the performance of the obtained EEKFs against the conventional EKF. To show that there is no overfitting, a second set of 500 normally distributed random noise samples is now considered for a new test (Test 2). Furthermore, robustness was verified considering variations in the system bifurcation parameter, the EEKF initial condition, and the system initial conditions, as shown in the previous subsection.

The robustness analysis demonstrates the performance of the constructed EEKFs for accurately estimating the state of the logistic map system. Notice that the EEKF does not assume WGN; thus, the function R^{gp^i} adjusts the covariance value in terms of the predicted noise value, and it does not need the real noise covariance of the measurement to tune the filter. This is, the EEKF dynamically self-tunes the measurement noise covariance parameter. From the numerical results, it can be observed that, in some cases, the EEKF provides better performance than the EKF. This can be explained since the EKF, by construction, assumes ideal conditions like WGN with zero mean. However, these theoretical conditions are not perfectly satisfied, since experimentally, or even in simulation, a WGN with zero mean is not truly accomplished.

In order to show that our methodology allows to achieve a suitable solution in average, the proposed procedure is repeated 30 times. Afterwards, a statistical analysis of performance, complexity, diversity, and frequency of functions and terminals, is realized. From this analysis the convergence of the proposed optimization problem is concluded.

5. Conclusions

An extension of the analytic behaviors framework has been developed for the optimal computation of a replacement of the unknown measurement noise covariance matrix, R_k, conventionally used to tune the EKF. Traditionally, the estimation of R_k is a challenge, since, in general, the nature of the noise is not known, and there is not an analytic method to compute it. In this approach, the covariance matrix is replaced by an analytic function. This expression is given in terms of the noisy output, the only available information from the system, and in terms of the filter's variables. Applying any one of the solutions, found by means of an evolutionary process, the resulting EKF structure becomes an Evolved EKF (EEKF).

The proposed extension of the analytic behaviors is applied using two parallel dynamical systems: the model of the system, and the EEKF. The *basis* behaviors are defined for each system, and the optimized search, by means of evolutionary computation, is embedded into the EEKF using the noisy outputs from the nonlinear FO-DS as an interaction between both systems.

By using the analytic behaviors framework, many different solutions with similar performance can be found for the construction of EEKFs. The optimality criteria of the standard EKF formulation is used to define an aptitude function to guide the search of solutions and to assess the suitability of the EEKFs.

Since the found functions depend only on available information from the system and on the variables of the filter itself, our proposal can be used in any application that requires an EKF, with the advantage that it does not need the known measurement noise covariance to tune the filter. Besides, even when the search for solutions is realized offline, the produced results are analytic functions which are easy to implement online. Moreover, the robustness analysis showed that the found EEKFs maintain their adequate response even in the presence of variations in parameters and/or initial conditions.

Inspired by the results presented in this work, the analytic behaviors methodology can be applied to propose specialized EEKFs for particular applications in nonlinear first-order dynamical systems. Extension to nonlinear higher order dynamical systems and the introduction of further considerations about the process noise can be also evaluated as future work.

CRediT authorship contribution statement

Leonardo Herrera: Conceptualization, Methodology, Formal analysis, Software, Writing – original draft, Writing – review & editing. **M.C. Rodríguez-Liñán:** Conceptualization, Formal analysis, Writing - original draft, Writing – review & editing. **Eddie Clemente:** Conceptualization, Formal analysis, Software, Visualization, Writing – original draft, Writing – review & editing. **Marlen Meza-Sánchez:** Conceptualization, Methodology, Formal analysis, Software, Writing – original draft, Writing – review & editing. **Luis Monay-Arredondo:** Validation, Software.

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.

Acknowledgments

Eddie Clemente thanks TecNM the support given through the project 11391.21-P. L. Monay-Arredondo thanks CONACYT and TecNM for the support given under scholarship No. 406498.

References

[1] L.A. McGee, S.F. Schmidt, Technical Memorandum: Discovery of the Kalman Filter as a Practical Tool for Aerospace and Industry, (Tech. Rep. NASA TM-86847) National Aeronautics and Space Administration, Washington, DC 20546, 1985, p. 24.
[2] M.S. Grewal, A.P. Andrews, Applications of Kalman filtering in aerospace 1960 to the present [Historical Perspectives], IEEE Control Syst. Mag. 30 (3) (2010) 69–78, http://dx.doi.org/10.1109/MCS.2010.936465.
[3] C.T. Fraser, S. Ulrich, Adaptive extended Kalman filtering strategies for spacecraft formation relative navigation, Acta Astronaut. 178 (2021) 700–721, http://dx.doi.org/10.1016/j.actaastro.2020.10.016.
[4] M. Zhu, W. Ouyang, Y. Wu, Orientation estimation by partial-state updating Kalman filter and vectorial magnetic interference detection, IEEE Trans. Aerosp. Electron. Syst. (2021) 1, http://dx.doi.org/10.1109/TAES.2021.3050657.
[5] R. Mehra, Approaches to adaptive filtering, IEEE Trans. Automat. Control 17 (5) (1972) 693–698, http://dx.doi.org/10.1109/TAC.1972.1100100.
[6] E. Zerdali, Adaptive extended Kalman filter for speed-sensorless control of induction motors, IEEE Trans. Energy Convers. 34 (2) (2019) 789–800, http://dx.doi.org/10.1109/tec.2018.2866383.
[7] L.V. Eykeren, Q. Chu, J. Mulder, Sensor fault detection and isolation using adaptive extended Kalman filter, IFAC Proc. Vol. 45 (20) (2012) 1155–1160, http://dx.doi.org/10.3182/20120829-3-mx-2028.00195.
[8] Y. Tian, H. Suwoyo, W. Wang, D. Mbemba, L. Li, An AEKF-SLAM algorithm with recursive noise statistic based on MLE and EM, J. Intell. Robot. Syst. 97 (2) (2020) 339–355.
[9] A.J. Barragán, B.M. Al-Hadithi, A. Jiménez, J.M. Andújar, A general methodology for online TS fuzzy modeling by the extended Kalman filter, Appl. Soft Comput. 18 (2014) 277–289, http://dx.doi.org/10.1016/j.asoc.2013.09.005.
[10] N. Bouzera, M. Oussalah, N. Mezhoud, A. Khireddine, Fuzzy extended Kalman filter for dynamic mobile localization in urban area using wireless network, Appl. Soft Comput. 57 (2017) 452–467, http://dx.doi.org/10.1016/j.asoc.2017.04.007.
[11] S. Yazdkhasti, J.Z. Sasiadek, Multi sensor fusion based on adaptive Kalman filtering, in: Advances in Aerospace Guidance, Navigation and Control, Springer International Publishing, 2017, pp. 317–333, http://dx.doi.org/10.1007/978-3-319-65283-2_17.
[12] V. Pesce, S. Silvestrini, M. Lavagna, Radial basis function neural network aided adaptive extended Kalman filter for spacecraft relative navigation, Aerosp. Sci. Technol. 96 (2020) 105527, http://dx.doi.org/10.1016/j.ast.2019.105527.
[13] A. Odry, I. Kecskes, P. Sarcevic, Z. Vizvari, A. Toth, P. Odry, A novel fuzzy-adaptive extended Kalman filter for real-time attitude estimation of mobile robots, Sensors 20 (3) (2020) 803, http://dx.doi.org/10.3390/s20030803.
[14] A. Assad, W. Khalaf, I. Chouaib, Radial basis function Kalman filter for attitude estimation in GPS-denied environment, IET Radar Sonar Navig. 14 (5) (2020) 736–746, http://dx.doi.org/10.1049/iet-rsn.2019.0467.
[15] C. Sun, Y. Zhang, G. Wang, W. Gao, A new variational Bayesian adaptive extended Kalman filter for cooperative navigation, Sensors 18 (8) (2018) 2538.
[16] S.-Y. Wang, C. Yin, S.-K. Duan, L.-D. Wang, A modified variational Bayesian noise adaptive Kalman filter, Circuits Systems Signal Process. 36 (10) (2017) 4260–4277, http://dx.doi.org/10.1007/s00034-017-0497-6.

[17] R.M. Watson, J.N. Gross, C.N. Taylor, R.C. Leishman, Enabling robust state estimation through measurement error covariance adaptation, IEEE Trans. Aerosp. Electron. Syst. 56 (3) (2020) 2026–2040, http://dx.doi.org/10.1109/taes.2019.2941103.
[18] K. Xiong, C. Wei, H. Zhang, Q-learning for noise covariance adaptation in extended Kalman filter, Asian J. Control (2020) http://dx.doi.org/10.1002/asjc.2336.
[19] H. Nobahari, A. Sharifi, A hybridization of extended Kalman filter and ant colony optimization for state estimation of nonlinear systems, Appl. Soft Comput. 74 (2019) 411–423, http://dx.doi.org/10.1016/j.asoc.2018.10.010.
[20] M. Sabet, A. Fathi, H.M. Daniali, Optimal design of the own ship maneuver in the bearing-only target motion analysis problem using a heuristically supervised extended Kalman filter, Ocean Eng. 123 (2016) 146–153, http://dx.doi.org/10.1016/j.oceaneng.2016.07.028.
[21] N. Rezaei, H. Kordabadi, A. Elkamel, A. Jahanmiri, An optimal extended Kalman filter designed by genetic algorithms, Chem. Eng. Commun. 196 (5) (2008) 602–615, http://dx.doi.org/10.1080/00986440802483996.
[22] D.Y. Yun, T. Hong, D.-E. Lee, H.S. Park, Structural damage identification with a tuning-free hybrid extended Kalman filter, Struct. Eng. Int. (2020) 1–15, http://dx.doi.org/10.1080/10168664.2020.1797614.
[23] D. Panigrahy, P.K. Sahu, Extended Kalman smoother with differential evolution technique for denoising of ECG signal, Australas. Phys. Eng. Sci. Med. 39 (3) (2016) 783–795, http://dx.doi.org/10.1007/s13246-016-0468-4.
[24] L. Zhang, D. Sidoti, A. Bienkowski, K.R. Pattipati, Y. Bar-Shalom, D.L. Kleinman, On the identification of noise covariances and adaptive Kalman filtering: A new look at a 50 year-old problem, IEEE Access 8 (2020) 59362–59388, http://dx.doi.org/10.1109/ACCESS.2020.2982407.
[25] M. Boutayeb, H. Rafaralahy, M. Darouach, Convergence analysis of the extended Kalman filter used as an observer for nonlinear deterministic discrete-time systems, IEEE Trans. Automat. Control 42 (4) (1997) 581–586, http://dx.doi.org/10.1109/9.566674.
[26] K. Reif, R. Unbehauen, The extended Kalman filter as an exponential observer for nonlinear systems, IEEE Trans. Signal Process. 47 (8) (1999) 2324–2328, http://dx.doi.org/10.1109/78.774779.
[27] Q. Zhang, Nonlinear system identification with output error model through stabilized simulation, IFAC Proc. Vol. 37 (13) (2004) 501–506, http://dx.doi.org/10.1016/S1474-6670(17)31273-9, 6th IFAC Symposium on Nonlinear Control Systems 2004 (NOLCOS 2004), Stuttgart, Germany, 1-3 September, 2004.
[28] L. Li, T. Wang, Y. Xia, N. Zhou, Trajectory tracking control for wheeled mobile robots based on nonlinear disturbance observer with extended Kalman filter, J. Franklin Inst. B 357 (13) (2020) 8491–8507, http://dx.doi.org/10.1016/j.jfranklin.2020.04.043.
[29] M. Holland, An Introduction To the Extended Kalman Filter, in: Mathematics research developments, Nova Science Publishers, 2020.
[30] E. Clemente, M. Meza-Sánchez, E. Bugarin, A.Y. Aguilar-Bustos, Adaptive behaviors in autonomous navigation with collision avoidance and bounded velocity of an omnidirectional mobile robot, J. Intell. Robot. Syst. 92 (2) (2018) 359–380, http://dx.doi.org/10.1007/s10846-017-0751-y.
[31] M. Meza-Sánchez, E. Clemente, R. Villalvazo, G. Olague, Bounding velocity in tracking control of unicycle mobile robots with genetic programming, in: 2018 XX Congreso Mexicano de RobÓtica (COMRob), 2018, pp. 1–6, doi:10.1109/COMROB.2018.8689409.
[32] M. Meza-Sánchez, E. Clemente, M.C. Rodríguez-Liñán, G. Olague, Synthetic-analytic behavior-based control framework: Constraining velocity in tracking for nonholonomic wheeled mobile robots, Inform. Sci. 501 (2019) 436–459, http://dx.doi.org/10.1016/j.ins.2019.06.025.
[33] O.P. naloza Mejía, E. Clemente, M. Meza-Sánchez, C. Pérez, F. Chavez, GP-based motion control design for the double-integrator system subject to velocity constraint, in: GECCO '17 Companion, Berlin, Germany, 2017.
[34] O.P. naloza Mejía, E. Clemente, M. Meza-Sánchez, C.B. Pérez, Evolving behaviors for bounded-flow tracking control of second-order dynamical systems, Eng. Appl. Artif. Intell. 78 (2019) 12–27, http://dx.doi.org/10.1016/j.engappai.2018.10.001.
[35] J. Farrell, Aided Navigation: GPS with High Rate Sensors, McGraw-Hill, Inc., 2008.
[36] M.S. Grewal, L.R. Weill, A.P. Andrews, Global Positioning Systems, Inertial Navigation, and Integration, John Wiley & Sons, 2007.
[37] L. Herrera, R. Hernández, F. Jurado, Control and extended Kalman filter based estimation for a ballbot robotic system, in: 2018 XX Congreso Mexicano de RobÓTica (COMRob), IEEE, 2018, pp. 1–6.
[38] R. May, Simple mathematical models with very complicated dynamics, Nature 261 (1978) 459–467, http://dx.doi.org/10.1038/261459a0.
[39] R.L. Kraft, Chaos, cantor sets, and hyperbolicity for the logistic maps, Amer. Math. Monthly 106 (5) (1999) 400–408, http://dx.doi.org/10.1080/00029890.1999.12005062.
[40] M.F. Maritz, A note on exact solutions of the logistic map, Chaos 30 (3) (2020) 033136, http://dx.doi.org/10.1063/1.5125097.
[41] R.M. May, G.F. Oster, Bifurcations and dynamic complexity in simple ecological models, Amer. Nat. 110 (974) (1976) 573–599.
[42] D. Simon, Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches, John Wiley & Sons, 2006.
[43] P. Kim, Kalman Filter for Beginners: With MATLAB Examples, CreateSpace, 2011.
[44] M.S. Grewal, A.P. Andrews, Kalman Filtering: Theory and Practice with MATLAB, John Wiley & Sons, 2014.
[45] R. Poli, W.B. Langdon, N.F. McPhee, a Field Guide To Genetic Programming, 2008, Published via http://lulu.com and freely available at http://www.gp-field-guide.org.uk, (With contributions by J. R. Koza).
[46] A. Ferretti, N. Rahman, A study of coupled logistic map and its applications in chemical physics, Chem. Phys. 119 (2) (1988) 275–288, http://dx.doi.org/10.1016/0301-0104(88)87190-8.
[47] D.L. Turcotte, Fractals and Chaos in Geology and Geophysics, second ed., Cambridge University Press, 1997, http://dx.doi.org/10.1017/CBO9781139174695.
[48] J.R. Groff, Exploring dynamical systems and chaos using the logistic map model of population change, Amer. J. Phys. 81 (10) (2013) 725–732, http://dx.doi.org/10.1119/1.4813114.
[49] A. Margaris, E. Kotsialos, N. Kofidis, M. Roumeliotis, M. Adamopoulos, Logistic map neural modelling: A theoretical foundation, Int. J. Comput. Math. 82 (9) (2005) 1055–1072, http://dx.doi.org/10.1080/00207160500113033.
[50] M. Ausloos, M. Dirickx (Eds.), The Logistic Map and the Route to Chaos, in: Understanding Complex Systems, Springer-Verlag Berlin Heidelberg, 2006, http://dx.doi.org/10.1007/3-540-32023-7.
[51] P. Jhansi Rani, M.S. Rao, S. Durga Bhavani, Design of secure chaotic hash function based on logistic and tent maps, in: D.C. Wyld, M. Wozniak, N. Chaki, N. Meghanathan, D. Nagamalai (Eds.), Advances in Network Security and Applications, Springer Berlin Heidelberg, Berlin, Heidelberg, 2011, pp. 43–52.
[52] M. Lawnik, Generalized logistic map and its application in chaos based cryptography, J. Phys. Conf. Ser. 936 (2017) 12–17, http://dx.doi.org/10.1088/1742-6596/936/1/012017.
[53] A. Sallam, E. El-Rabaie, O. Faragallah, Efficient HEVC selective stream encryption using chaotic logistic map, Multimedia Syst. 24 (2018) 419–437, http://dx.doi.org/10.1007/s00530-017-0568-3.
[54] R.L.C.J.C. do Nascimento, G.L. Damasceno, de Oliveira, R.V. Ramos, Quantum-chaotic key distribution in optical networks: from secrecy to implementation with logistic map, Quantum Inf. Process. 17 (2018) 329, http://dx.doi.org/10.1007/s11128-018-2097-1.
[55] A. Girdhar, V. Kumar, A reversible and affine invariant 3D data hiding technique based on difference shifting and logistic map, J. Ambient. Intell. Human. Comput. 10 (2019) 4927–4961, http://dx.doi.org/10.1007/s12652-019-01179-4.
[56] K.K. Raghuvanshi, S. Kumar, S. Kumar, A data encryption model based on intertwining logistic map, J. Inform. Secur. Appl. 55 (2020) 102622, http://dx.doi.org/10.1016/j.jisa.2020.102622.
[57] X.-H. Song, H.-Q. Wang, S.E. Venegas-Andraca, A.A. Abd El-Latif, Quantum video encryption based on qubit-planes controlled-xor operations and improved logistic map, Physica A 537 (2020) 122660, http://dx.doi.org/10.1016/j.physa.2019.122660.
[58] L. Moysis, E. Petavratzis, C. Volos, H. Nistazakis, I. Stouboulos, A chaotic path planning generator based on logistic map and modulo tactics, Robot. Auton. Syst. 124 (2020) 103377, http://dx.doi.org/10.1016/j.robot.2019.103377.
[59] R. Riolo, T. McConaghy, E. Vladislavleva (Eds.), Genetic Programming Theory and Practice VIII, Springer New York, 2011, http://dx.doi.org/10.1007/978-1-4419-7747-2.