Optimal Concurrent Estimation Method with Initial Value Search for Polynomial Kernel-based Nonlinear Observer Canonical Models

Jimei Li, Jan Swevers, and Feng Ding

Abstract— This paper presents a concurrent estimation method for a polynomial kernel-based nonlinear observer canonical model with a generic structure for nonlinear systems. The algorithm alternates the separable least squares parameter estimation algorithm and extended Kalman filtering. It has the characteristic of fast convergence to a local optimum which depends strongly on the initial values of parameters. To improve its convergence to the global optimum, a genetic algorithm-based concurrent estimation (GA-CE) method is proposed to search for the initial parameter set that minimizes the output estimation error. This optimal concurrent estimation method converges to the optimum or near-optimum solution without any trial-and-error steps for initial values. Validations on the Silverbox and Bouc-Wen hysteresis benchmarks show that the GA-CE method is able to find excellent solutions yielding models with superior predictive performance.

I. INTRODUCTION

The black box modeling approach equipped with state s-pace representation has the flexibility and capacity to capture different types of nonlinear phenomena, which can be applied to model a variety of mechanical and control systems [1]–[3]. Much research has been dedicated to the state space model identification to provide insights into various nonlinear dynamics such as the subspace identification [4], [5] and best linear approximation [6], [7]. Moreover, there has been an increasing interest in concurrent state and parameter estimation [8], [9] which allows for the efficient update of estimates using available measurements. During the estimation, the concurrent estimation method compares values of nearby points and quickly moves to a local optimal point. However, the non-convexity of nonlinear optimization problems makes the estimation results strongly dependent on the initial values of the parameters. So a long trial-and-error process is needed for a good local minimum. How to quickly search for a good initial parameter set is critical for concurrent estimation methods.

This work tackles the initial value sensitivity of concurrent estimation method by introducing a genetic algorithm (GA). As an intelligent bionic algorithm, the GA evolves in a manner analogous to the natural process of genetic evolution in living creatures [10]. It has the advantages of working with a random population and possessing great global optimization potential, which is able to help concurrent estimation

Jimei Li and Feng Ding are with the Key Laboratory of Advanced Process Control for Light Industry (Ministry of Education), School of Internet of Things Engineering, Jiangnan University, Wuxi 214122, China `jimeili0625@163.com, fding@jiangnan.edu.cn`

Jimei Li and Jan Swevers are with the MECO Research Team, Department of Mechanical Engineering, KU Leuven, Belgium, and with Flanders Make@KU Leuven, Belgium `jan.swevers@kuleuven.be`

methods find good initial values convergence to the global optimum. Random mutation also takes place in the GA as a means to escape entrapment in the local minimum. Moreover, GAs provide great flexibility to hybridize with various optimization approaches, yielding many efficient implementations, such as various enhanced genetic algorithms [11] and the genetic algorithm-based selection and prioritization for autonomous driving systems [12].

Inspired by these GA-based optimization methods, this work proposes a GA-based concurrent estimation (GA-CE) method. The GA randomly generates an initial population containing initial parameter sets, avoiding the trial process of finding an appropriate initialization. For each set of initial values generated by the GA, the concurrent estimation method alternates separable least squares parameter estimation algorithm and extended Kalman filtering, and calculates the root mean square output estimation error as the fitness. Using these initial value sets and their corresponding fitness values, the more fit initial sets are stochastically selected as candidates by the tournament selection strategy. Based on these candidates, simulated binary crossover and polynomial mutation operators generate other feasible initial sets, yielding a new population for the next iteration. After several optimization loops, the GA-CE determines the optimum or an accurate sub-optimal solution. The validity of the proposed GA-CE is tested using the Silverbox benchmark data [13] and Bouc-Wen hysteretic benchmark data [14].

This paper is organized as follows: Section II describes a polynomial kernel-based nonlinear observer canonical model for nonlinear systems. Section III proposes the GA-CE algorithm. Section IV provides two simulation examples. Finally, the conclusions are drawn in Section V.

II. K-NOCF MODEL

The considered polynomial kernel-based nonlinear observer canonical form (K-NOCF) model is a discrete-time state space model with following form:

$$\boldsymbol{x}_{t+1} = \boldsymbol{A}\boldsymbol{x}_t + \boldsymbol{\beta}u_t + \boldsymbol{E}\boldsymbol{f}(\boldsymbol{x}_t, u_t) + \boldsymbol{w}_t, \qquad (1)$$

$$y_t = \boldsymbol{C}\boldsymbol{x}_t + v_t, \qquad (2)$$

where $\boldsymbol{x}_t \in \mathbb{R}^n$ is the state vector, $y_t \in \mathbb{R}$ is the measured output, $u_t \in \mathbb{R}$ is the input, $\boldsymbol{w}_t \in \mathbb{R}^n$ and $v_t \in \mathbb{R}$ are the process noise vector and measurement noise, the system parameter matrix/vectors $\boldsymbol{A} \in \mathbb{R}^{n \times n}$, $\boldsymbol{\beta} \in \mathbb{R}^n$ and $\boldsymbol{c} \in \mathbb{R}^{1 \times n}$ have the observer canonical model structure,

which are defined as follows:

$$
\boldsymbol{A} := \begin{bmatrix} -\alpha_1 & 1 & 0 & \cdots & 0 \\ -\alpha_2 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -\alpha_{n-1} & 0 & 0 & \cdots & 1 \\ -\alpha_n & 0 & 0 & \cdots & 0 \end{bmatrix} \in \mathbb{R}^{n \times n},
$$
$$
\boldsymbol{\beta} := [\beta_1, \beta_2, \cdots, \beta_n]^{\mathrm{T}} \in \mathbb{R}^n,
$$
$$
\boldsymbol{c} := [1, 0, \cdots, 0] \in \mathbb{R}^{1 \times n}.
$$

Other parameters to be estimated are the elements of the matrix $\boldsymbol{E} := [\boldsymbol{\rho}_1^{\mathrm{T}}, \boldsymbol{\rho}_2^{\mathrm{T}}, \cdots, \boldsymbol{\rho}_n^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{n \times a}$, where $\boldsymbol{\rho}_i \in \mathbb{R}^a$. The nonlinear vector $\boldsymbol{f}(\boldsymbol{x}_t, u_t)$ contains monomial combinations in the states and inputs up to a certain degree d, i.e., $\boldsymbol{f}(\boldsymbol{x}_t, u_t) := [x_{1,t}^2, x_{1,t}x_{2,t}, x_{1,t}u_t, \cdots, u_t^d]^{\mathrm{T}} \in \mathbb{R}^a$.

The success of the polynomial K-NOCF model comes from the fact that a large class of practical systems can be modeled in nonlinear state space form, and the polynomial kernel possesses universal approximation properties.

Our identification goal is to estimate parameter matrices $\boldsymbol{A}$ and $\boldsymbol{E}$, the vector $\boldsymbol{\beta}$ and the unmeasurable states based on the measurements without any trial-and-error steps for initial parameter values.

III. OPTIMAL CONCURRENT ESTIMATION METHOD

This section describes the genetic algorithm (GA) in detail, develops the extended Kalman filtering-separable least squares (EKF-SLS) algorithm for the concurrent estimation of the polynomial K-NOCF model, and shows how the GA helps the EKF-SLS search for initial parameter values.

A. GA

A GA has great global optimization potential and applies the concept of survival of the fittest to find optimal or near-optimal solutions. The operators of GAs include selection, crossover and mutation. The basic flow and operators are introduced below.

1) *Initial population:* The first step of GAs is to generate an initial population containing several strings. The number of strings can be artificially set. The initial population is generated randomly, but we can set an upper bound and a lower bound to determine a precise search space and to save optimization time. After creating an initial population, each string is then evaluated and assigned a fitness value.

2) *Selection:* Once we have the current population and the corresponding fitness values, the tournament selection strategy [15] is applied to create an intermediate population. The probability that strings in the current population are copied and placed in the intermediate generation is related to their fitness. We randomly select two individuals and run a tournament among them, only the fittest one is chosen and is passed on to the next generation. In this way, many such tournaments take place to get the desired amount of strings, and we have our final candidates that pass on to the next step as parent strings.

3) *Crossover:* Simulated binary crossover (SBX) [16] is applied to randomly paired strings. It is designed concerning the one-point crossover properties in binary-coded GA. The crossover operator starts with two randomly selected parent strings: $\boldsymbol{p}^1(p_1^1, p_2^1, \cdots, p_{n_s}^1)$ and $\boldsymbol{p}^2(p_1^2, p_2^2, \cdots, p_{n_s}^2)$, where n_s denotes the string length and also the dimension of the solution. If the uniform distribution random number $r_1 \in$ [0,1) is less than the crossover probability P_c, the values in offspring strings $\boldsymbol{o}^1(o_1^1, o_2^1, \cdots, o_{n_s}^1)$ and $\boldsymbol{o}^2(o_1^2, o_2^2, \cdots, o_{n_s}^2)$ are calculated as:

$$
o_i^1 = 0.5 \times [(1+\lambda)p_i^1 + (1-\lambda)p_i^2],
$$
$$
o_i^2 = 0.5 \times [(1-\lambda)p_i^1 + (1+\lambda)p_i^2],
$$

where λ is the spread factor, it is defined as

$$
\lambda = \begin{cases} (2r_2)^{\frac{1}{n_c+1}}, & \text{if } r_2 \leqslant 0.5, \\ (\frac{1}{2-2r_2})^{\frac{1}{n_c+1}}, & \text{if } r_2 > 0.5, \end{cases}
$$

$r_2 \in$ [0,1) is a random number. n_c denotes the spread factor distribution index, and a large value of n_c gives a high probability of creating "near-parent" solutions. The SBX operator performs well in local optimization searches and is widely used in high-dimensional target evolution algorithms.

4) *Mutation:* After the recombination, a polynomial mutation operator [17] is used to prevent all possible solutions from converging to a single local optimum. For each offspring value o_i updated by the SBX crossover, if the random number $r_3 \in$ [0,1) is less than the mutation probability P_m, the mutated value o_i' is calculated as

$$
o_i' = o_i + \mu(b_u - b_l), \quad i = 1, 2, \cdots, n,
$$

where μ is the perturbation factor, b_u and b_l is the upper and low bounds set during the initialization. Typically the mutation rate $P_m \leqslant 0.10$. Take a uniform distribution random number r_4 from [0,1). If $r_4 \leqslant 0.5$, the perturbation factor μ is taken as

$$
\mu = [2r_4 + (1 - 2r_4)(1 - \mu_1)^{n_m+1}]^{\frac{1}{n_m+1}} - 1,
$$

where $\mu_1 = (o_i - b_l)/(b_u - b_l)$, n_m is the user selected distribution index; if the random number $r_4 > 0.5$,

$$
\mu = 1 - [2(1 - r_4) + (2r_4 - 1)(1 - \mu_2)^{n_m+1}]^{\frac{1}{n_m+1}},
$$

where $\mu_2 = (b_u - o_i)/(b_u - b_l)$.

Crossover and mutation operators are applied to create the next population, and new strings should belong to the bounds set earlier. The process of going from the current population to the next population constitutes one generation in the execution of GAs. After several generations, the algorithm converges to the best initial set, which hopefully generates the optimum or an accurate sub-optimal solution to the problem.

B. EKF-SLS algorithm

Here we derive a concurrent parameter and state estimation scheme for the K-NOCF model by using the separable least squares and extended Kalman filtering methods.

First, we use the property of the unit backward shift operator, transform the K-NOCF model in (1)–(2) into a parameter estimation model: $y_t = \boldsymbol{\varrho}_t^{\mathrm{T}}\boldsymbol{\gamma} + \boldsymbol{\varsigma}_t^{\mathrm{T}}\boldsymbol{\beta} + \eta_t + v_t$, where

the information vectors $\boldsymbol{\varrho}_t$ and $\boldsymbol{\varsigma}_t$, the parameter vector $\boldsymbol{\gamma}$ and the cumulative process noise η are defined as

$$
\begin{aligned}
\boldsymbol{\varrho}_t := [&-x_{1,t-1}, -x_{1,t-2}, \cdots, -x_{1,t-n}, \boldsymbol{f}^{\mathrm{T}}(\boldsymbol{x}_{t-1}, u_{t-1}), \\
&\boldsymbol{f}^{\mathrm{T}}(\boldsymbol{x}_{t-2}, u_{t-2}), \cdots, \boldsymbol{f}^{\mathrm{T}}(\boldsymbol{x}_{t-n}, u_{t-n})]^{\mathrm{T}} \in \mathbb{R}^m, \\
&m := na + n, \\
\boldsymbol{\varsigma}_t := [&u_{t-1}, u_{t-2}, \cdots, u_{t-n}]^{\mathrm{T}} \in \mathbb{R}^n, \\
\boldsymbol{\gamma} := [&\alpha_1, \alpha_2, \cdots, \alpha_n, \boldsymbol{\rho}_1, \boldsymbol{\rho}_2, \cdots, \boldsymbol{\rho}_n]^{\mathrm{T}} \in \mathbb{R}^m, \\
\eta_t := &w_{1,t-1} + w_{2,t-2} + \cdots + w_{n,t-n} \in \mathbb{R}.
\end{aligned}
$$

The hierarchical identification principle is introduced to develop a computationally efficient parameter estimation method for the K-NOCF model. The principle is to divide the model into two sub-models $y_{1,t} = \boldsymbol{\varrho}_t^{\mathrm{T}} \boldsymbol{\gamma} + \eta_t + v_t$ and $y_{2,t} = \boldsymbol{\varsigma}_t^{\mathrm{T}} \boldsymbol{\beta} + \eta_t + v_t$, each of which contains partial parameters to be identified, to reduce the scale of the optimization problem.

Let $\hat{\boldsymbol{\gamma}}_t$ and $\hat{\boldsymbol{\beta}}_t$ be the parameter estimates of $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ at sampling instant t, and $\hat{\boldsymbol{x}}_t$ denotes the state estimate. The steps of the EKF-SLS algorithm are summarized here.

1) *SLS method for parameter estimation:*

$$\hat{\boldsymbol{\gamma}}_t = \hat{\boldsymbol{\gamma}}_{t-1} + \boldsymbol{L}_{\gamma,t} e_t, \tag{3}$$

$$\boldsymbol{L}_{\gamma,t} = \boldsymbol{P}_{\gamma,t-1} \hat{\boldsymbol{\varrho}}_t [1 + \hat{\boldsymbol{\varrho}}_t^{\mathrm{T}} \boldsymbol{P}_{\gamma,t-1} \hat{\boldsymbol{\varrho}}_t]^{-1}, \tag{4}$$

$$\boldsymbol{P}_{\gamma,t} = [\boldsymbol{I}_m - \boldsymbol{L}_{\gamma,t} \hat{\boldsymbol{\varrho}}_t^{\mathrm{T}}] \boldsymbol{P}_{\gamma,t-1}, \tag{5}$$

$$\hat{\boldsymbol{\beta}}_t = \hat{\boldsymbol{\beta}}_{t-1} + \boldsymbol{L}_{\beta,t} e_t, \tag{6}$$

$$\boldsymbol{L}_{\beta,t} = \boldsymbol{P}_{\beta,t-1} \boldsymbol{\varsigma}_t [1 + \boldsymbol{\varsigma}_t^{\mathrm{T}} \boldsymbol{P}_{\beta,t-1} \boldsymbol{\varsigma}_t]^{-1}, \tag{7}$$

$$\boldsymbol{P}_{\beta,t} = [\boldsymbol{I}_n - \boldsymbol{L}_{\beta,t} \boldsymbol{\varsigma}_t^{\mathrm{T}}] \boldsymbol{P}_{\beta,t-1}, \tag{8}$$

$$e_t = y_t - \boldsymbol{\varsigma}_t^{\mathrm{T}} \hat{\boldsymbol{\beta}}_{t-1} - \hat{\boldsymbol{\varrho}}_t^{\mathrm{T}} \hat{\boldsymbol{\gamma}}_{t-1}, \tag{9}$$

where $\boldsymbol{P}_{\gamma,t} \in \mathbb{R}^{m \times m}$ and $\boldsymbol{P}_{\beta,t} \in \mathbb{R}^{n \times n}$ are covariance matrices, $\boldsymbol{L}_{\gamma,t} \in \mathbb{R}^m$ and $\boldsymbol{L}_{\beta,t} \in \mathbb{R}^n$ are gain vectors, e_t is the innovation.

2) *Prediction steps of the EKF:*

Read parameter estimates from the vectors $\hat{\boldsymbol{\gamma}}_t$ and $\hat{\boldsymbol{\beta}}_t$, and form the matrices $\hat{\boldsymbol{A}}_t$ and $\hat{\boldsymbol{E}}_t$.

$$\hat{\boldsymbol{x}}_{t|t-1} = \hat{\boldsymbol{A}}_t \hat{\boldsymbol{x}}_{t-1} + \hat{\boldsymbol{\beta}}_t u_{t-1} + \hat{\boldsymbol{E}}_t \boldsymbol{f}(\hat{\boldsymbol{x}}_{t-1}, u_{t-1}), \tag{10}$$

$$
\begin{aligned}
\hat{\boldsymbol{\Psi}}_{t|t-1} = &[\hat{\boldsymbol{A}}_t + \hat{\boldsymbol{E}}_t \boldsymbol{F}(\hat{\boldsymbol{x}}_{t-1}, u_{t-1})] \hat{\boldsymbol{\Psi}}_{t-1} \\
&\times [\hat{\boldsymbol{A}}_t + \hat{\boldsymbol{E}}_t \boldsymbol{F}(\hat{\boldsymbol{x}}_{t-1}, u_{t-1})] + \boldsymbol{Q},
\end{aligned} \tag{11}
$$

where $\hat{\boldsymbol{x}}_{t|t-1} \in \mathbb{R}^n$ and $\hat{\boldsymbol{\Psi}}_{t|t-1} \in \mathbb{R}^{n \times n}$ denote the predicted state and covariance estimates at sample instant t given the previous estimates $\hat{\boldsymbol{x}}_{t-1}$ and $\hat{\boldsymbol{\Psi}}_{t-1}$, $\boldsymbol{F}(\hat{\boldsymbol{x}}_{t-1}, u_{t-1}) \in \mathbb{R}^{a \times n}$ denotes the partial derivative of the nonlinear mapping $\boldsymbol{f}(\hat{\boldsymbol{x}}_{t-1}, u_{t-1})$ with respect to $\hat{\boldsymbol{x}}_{t-1}$, $\boldsymbol{Q}$ is the covariance of the process noise vector $\boldsymbol{w}_t$.

3) *Correction steps of the EKF:*

$$S_t = \boldsymbol{c} \hat{\boldsymbol{\Psi}}_{t|t-1} \boldsymbol{c}^{\mathrm{T}} + R, \tag{12}$$

$$\boldsymbol{K}_t = \hat{\boldsymbol{\Psi}}_{t|t-1} \boldsymbol{c}^{\mathrm{T}} S_t^{-1}, \tag{13}$$

$$\hat{\boldsymbol{x}}_t = \hat{\boldsymbol{x}}_{t|t-1} + \boldsymbol{K}_t (y_t - \boldsymbol{c} \hat{\boldsymbol{x}}_{t|t-1}), \tag{14}$$

$$\hat{\boldsymbol{\Psi}}_t = (\boldsymbol{I}_n - \boldsymbol{K}_t \boldsymbol{c}_t) \hat{\boldsymbol{\Psi}}_{t|t-1}, \tag{15}$$

where $S_t \in \mathbb{R}$ is the innovation covariance, $R \in \mathbb{R}$ is the covariance of the measurement noise, $\boldsymbol{K}_t \in \mathbb{R}^n$ is the gain vector, $\hat{\boldsymbol{\Psi}}_t$ is the updated covariance estimate.

By executing these steps in Equation (3)–(15) iteratively, the EKF-SLS algorithm performs local search by a convergent stepwise procedure, which has been demonstrated in previous work [18]. The right of Fig. 1 in red shows the schematic diagram of the EKF-SLS algorithm. A global optimum can be found only if the problem possesses certain convexity properties. Otherwise, we need to find good initial values for precise estimation by trial and error, which is tricky for various nonlinear systems.

C. GA-based concurrent estimation method

The optimal concurrent estimation method proposed in this work benefits from the random initial value sets provided by GAs and the fast local search capability of the EKF-SLS method. The schematic diagram of the genetic algorithm-based concurrent estimation (GA-CE) method is shown in Fig. 1. Because this work is dedicated to obtaining accurate estimates, the objective function is defined as the root mean square error (RMSE) of the estimated output. The proposed GA-CE aims to finding initial values $\boldsymbol{\gamma}_0$, $\boldsymbol{\beta}_0$ which minimize the RMSE

$$\min_{\boldsymbol{\gamma}_0, \boldsymbol{\beta}_0} \sqrt{\frac{1}{L} \sum_{t=1}^{L} [\hat{y}_t - y_t]^2},$$

where L is the length of estimation data, y_t denotes the measurements, and $\hat{y}_t$ represents the estimated output obtained by the EKF-SLS. Since the objective function is non-negative, we directly use the objective function as the fitness function. The flow of the GA-CE algorithm is that:

1) Randomly generate N strings to form a finite ground set (a population) within the upper and lower bounds using the GA.
2) Take N strings as N sets of initial values, introduce these sets to the EKF-SLS for concurrent estimation, and then get N RMSEs (fitnesses) of estimated outputs.
3) Select candidates using the tournament selection strategy based on these N fitnesses, then process these candidates using SBX crossover and polynomial mutation operators in turn yielding the next population.
4) Start the next optimization loop with new N feasible solutions until the number of generation reaches the maximal generation G.
5) Output the best initial set and the corresponding RMSE.

IV. EXPERIMENTS AND RESULTS

This section evaluates the polynomial K-NOCF model and GA-CE method on two benchmarks in nonlinear system identification, which are the Silverbox system [13] and Bouc-Wen hysteretic system [14].

A. Silverbox system

The nonlinear model for the Silverbox system is following continuous-time expression:

$$m_s \ddot{y}_t + \zeta \dot{y}_t + k_1 y_t + k_2 y_t^3 = u_t, \tag{16}$$

where the input u_t is the force applied to the mass m_s, the output y_t is its displacement, the parameters k_1 and k_2
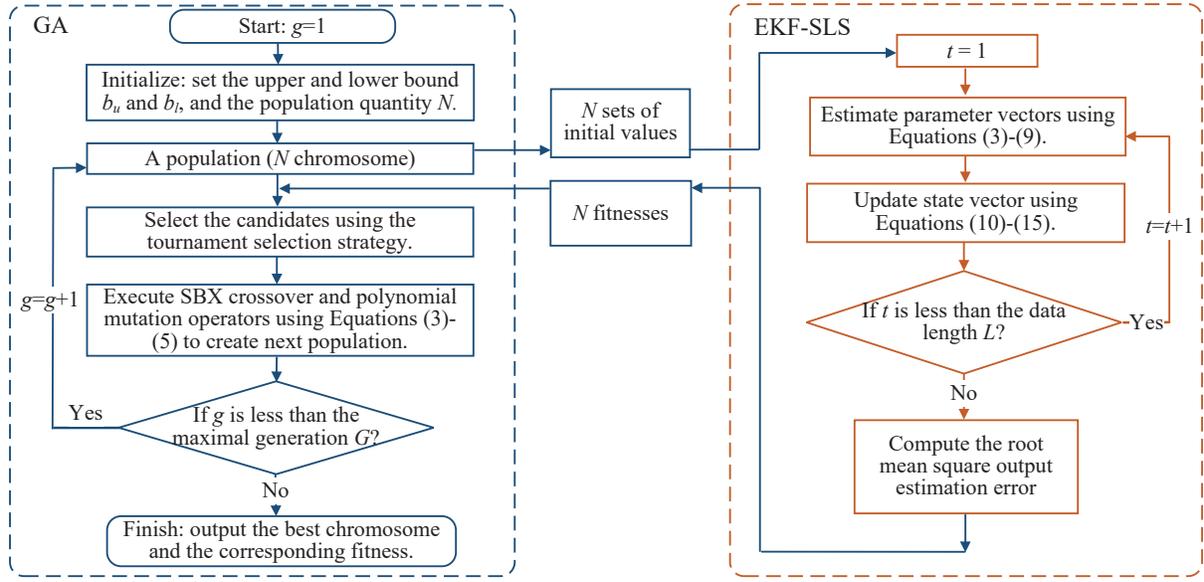
Fig. 1. Schematic diagram of the GA-CE method.

describe the behavior of the spring, and ζ is the damping of the system. The input data for the system is a random phase multi-sine excitation with maximum frequency approximately equal to 200 Hz. The data of the benchmark has been collected at a sampling frequency of 610.35 Hz.

Fig. 2 shows the measured excitation and response data as presented in [13]. The first part of the signals consists of 40000 samples which are used for verifying the model obtained by the estimated process. This part of the excitation shows a linearly increasing amplitude which exceeds the amplitude of the second part of the excitation signal, hereby enabling to test the extrapolation capabilities of the estimated model. The remaining data is used for the model parameter estimation, so we have $L = 87000$.
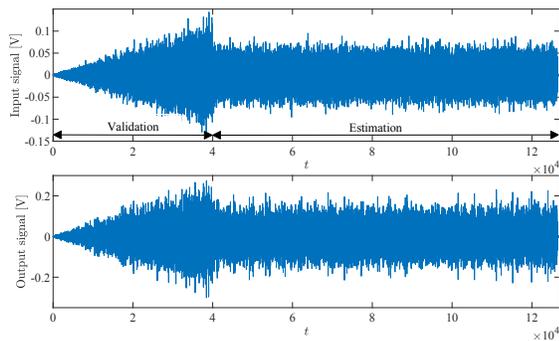

Fig. 2. Excitation and measurement signals of the Silverbox system

The second-order polynomial K-NOCF model proposed to identify the Silverbox model is as follows:

$$\boldsymbol{x}_{t+1} = \boldsymbol{A}\boldsymbol{x}_t + \boldsymbol{\beta}u_t + \boldsymbol{E}\boldsymbol{f}(\boldsymbol{x}_t, u_t) + \boldsymbol{w}_t,$$
$$y_t = \boldsymbol{c}\boldsymbol{x}_t + v_t,$$
$$\boldsymbol{f}(\boldsymbol{x}_t, u_t) = [x_{1,t}x_{2,t}^2, x_{1,t}^3]^{\mathrm{T}}.$$

So there are 8 parameters to be identified in this problem.

The size of a population N is set as 50, and the maximal generation is taken as $G = 100$. Upper and lower bounds for all parameters are set as $b_u = 2$ and $b_l = -2$. The crossover probability P_c and distribution index n_c are taken as $P_c = 0.90$ and $n_c = 20$. The mutation rate P_m and distribution index n_m are taken as $P_m = 0.10$ and $n_m = 20$. For comparison, we run the proposed GA-CE, GA, GA-SLS (GA combined with SLS) and GA-EKF (GA combined with EKF) methods under the same settings.
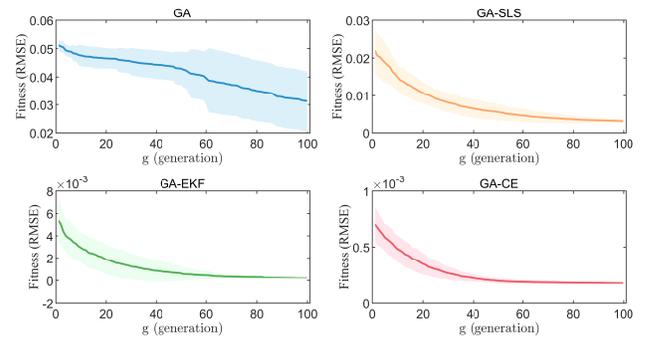

Fig. 3. Means and Stds of the best fitness obtained by 10 repeated experiments in each generation

Because the initial population of these four algorithms are generated stochastically, we repeat these experiments 10 times to verify the effectiveness of the proposed algorithm on more statistical spaces. For each experiment, the four algorithms are executed with $N = 50$ different initial strings. The result with the lowest RMSE of the estimated output is selected. Fig. 3 shows the mean (solid line) and the standard deviation (shadow) of these selected best results over the 10 repeated experiments, as a function of the number of generations. Table I shows the minimum, mean and maxi-

mum training and prediction RMSEs for the four algorithms under 10 repeated experiments in the $100th$ generation. The combination of the GA with the SLS and EKF algorithms reduces the degree of dispersion of results, and results in more accurate solutions than simple GA. The GA-CE gets the lowest training and validation RMSEs among these four algorithms. The good performance of this combination of the GA and EKF-SLS is due to the fact that the GA-CE finds the best initial parameter set among these four methods in the first generation, and outperforms the other combinations in the whole estimation process.

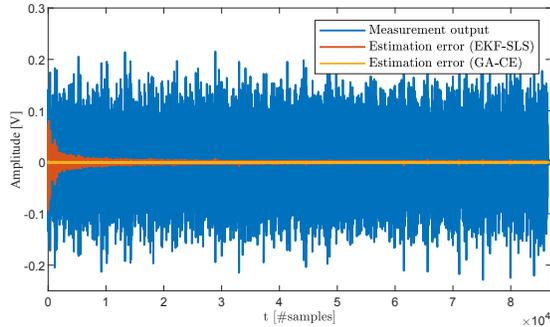| Approaches	Training RMSE [mV]			Validation RMSE [mV]		
	Minimum	Mean	Maximum	Minimum	Mean	Maximum
GA	10.87	31.29	46.69	15.78	32.27	45.71
GA-SLS	2.60	3.12	4.57	2.94	3.56	4.69
GA-EKF	0.17	0.23	0.39	0.18	0.31	0.67
GA-CE	**0.17**	**0.18**	**0.20**	**0.17**	**0.19**	**0.25**

Fig. 4. Estimation error obtained using the EKF-SLS and GA-CE on the estimation set of the silverbox benchmark

Fig. 4 compares the estimation performance of the EKF-SLS algorithm and the GA-CE (the best result obtained in 10 repeated experiments). Initial values of the EKF-SLS algorithm are set as $\hat{\boldsymbol{\gamma}}_0 = \mathbf{1}_m/p_0$ and $\hat{\boldsymbol{\beta}}_0 = \mathbf{1}_n/p_0$, $p_0 = 10^6$. We compute the root mean square output estimation error of these two algorithms. $\text{RMSE}_{\text{EKF-SLS}} = 1.43$ mV, $\text{RMSE}_{\text{GA-CE}} = 0.17$ mV. The above results indicate that the GA-CE significantly outperforms the EKF-SLS algorithm. The estimation error of the EKF-SLS algorithm mainly comes from the initial estimation part (as shown in Fig. 4). The GA-CE with the global searching ability improves the initial error fluctuation problem of the EKF-SLS method, and gets low error in the whole estimation process.

Fig. 5 shows the GA-CE and EKF-SLS output prediction errors for the 40000 validation data samples. The root mean square predicted output errors are $\text{RMSE}_{\text{EKF-SLS}} = 0.37$ mV and $\text{RMSE}_{\text{GA-CE}} = 0.18$ mV. The proposed method has excellent prediction performance, and the prediction error of the GA-CE is smaller than that of the EKF-SLS. The K-NOCF model performs well during the extrapolation, which
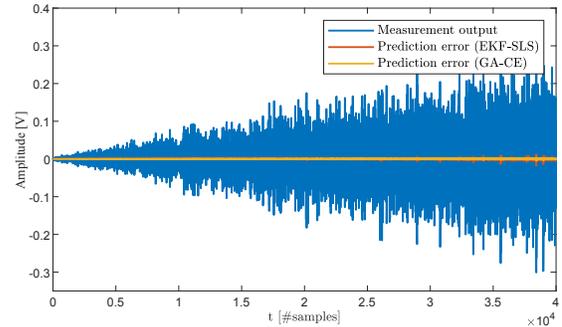

Fig. 5. Prediction error obtained using the estimated K-NOCF model on the validation set of the silverbox benchmark

starts at sample $t = 30000$ approximately. This is due to the similarity between the K-NOCF model with polynomial kernel and the internal polynomial structure of the Silverbox analytic model (16).

B. Bouc-Wen hysteretic system

The second example uses data from the Bouc-Wen benchmark system, described in detail in [14]. The Bouc-Wen system is a hysteretic system featuring a dynamic nonlinearity, which is governed by the second-order differential equation:

$$m_b\ddot{y}_t + c_b\dot{y}_t + k_b y_t + z(y_t, \dot{y}_t) = u_t, \quad (17)$$

where u_t is the external force, y_t is the resulting displacement, m_b, k_b and c_b are the mass constant, stiffness coefficient and viscous damping coefficient, respectively. The hysteretic force $z(y_t, \dot{y}_t)$ obeys the first-order differential equation:

$$\dot{z}(y_t, \dot{y}_t) = \eta_1\dot{y}_t - \eta_2(k_3|\dot{y}_t||z_t|^{\epsilon-1} + k_4\dot{y}_t|z_t|^{\epsilon}), \quad (18)$$

where the parameters η_1, η_2, k_3, k_4 and ϵ determine the shape and smoothness of the system hysteretic loop.

The training data are generated by integrating Equations (17)–(18) using the Newmark integration at a sampling rate of 15000 Hz. The data are then low-pass filtered and downsampled to 750Hz. We use five periods of the random phase multi-sine input signal, which has 40960 samples.

The third-order polynomial K-NOCF model proposed to identify the Bouc-Wen system is as follows:

$$\boldsymbol{x}_{t+1} = \boldsymbol{A}\boldsymbol{x}_t + \boldsymbol{\beta}u_t + \boldsymbol{E}\boldsymbol{f}(\boldsymbol{x}_t, u_t) + \boldsymbol{w}_t,$$
$$y_t = \boldsymbol{c}\boldsymbol{x}_t + v_t,$$
$$\boldsymbol{f}(\boldsymbol{x}_t, u_t) = [x_{1,t}^2 x_{2,t}, x_{1,t}x_{2,t}]^{\mathrm{T}}.$$

Hence there are 12 parameters that need to be estimated. For comparison, we identify the Bouc-Wen hysteretic system using the EKF-SLS and GA-CE methods. Initial settings of these methods are the same as the Silverbox example. For the GA-CE, we run the GA-CE algorithm 10 times, each time with a different population of $N = 50$ initial strings.

Then the estimated models and the multi-sine and swept sine validation data sets provided with the benchmark [14] are used for the prediction. For the multi-sine data set, two

periods are used to make sure that the model output is in a steady state, and the RMSE is calculated in the second period. For the swept sine data, the first 2000 samples are ignored when calculating the RMSE to allow the transient to decay.
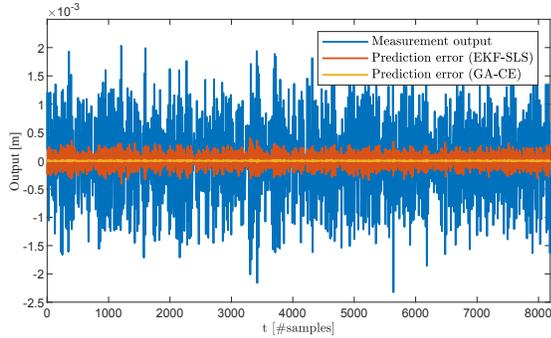

Fig. 6. Prediction error obtained using the estimated K-NOCF model on the multi-sine validation set of the Bouc-Wen benchmark
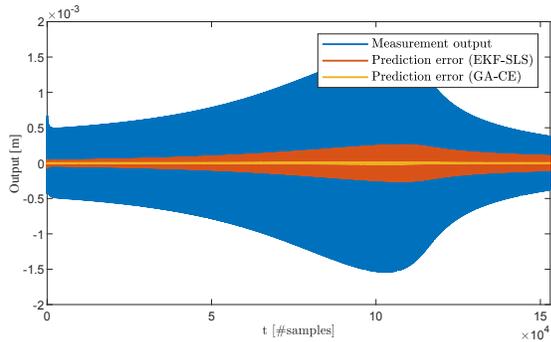

Fig. 7. Prediction error obtained using the estimated K-NOCF model on the swept sine validation set of the Bouc-Wen benchmark

Figs. 6–7 show the GA-CE (the minimum error in 10 repeat experiments) and EKF-SLS output prediction errors for these two validation data sets. For the multi-sine validation, we have $\text{RMSE}_{\text{EKF-SLS}} = 1.15 \times 10^{-4}$ and $\text{RMSE}_{\text{GA-CE}} = 3.60 \times 10^{-6}$. For the swept sine validation, we have $\text{RMSE}_{\text{EKF-SLS}} = 1.05 \times 10^{-4}$ and $\text{RMSE}_{\text{GA-CE}} = 2.90 \times 10^{-6}$. It is apparent from these results that the estimated model obtained by the GA-CE gets low errors in prediction experiments. Compared with the EKF-SLS, the proposed GA-CE searches for a better solution for the Bouc-Wen system.

V. CONCLUSIONS

This work proposes a concurrent estimation method for the polynomial K-NOCF model structure which extends the EKF-SLS method with a GA driven inital value estimation. The excellent estimation and prediction performance of the proposed GA-CE is demonstrated on the Silverbox and Bouc-Wen benchmarks. Making full use of the advantages of both algorithms, the proposed method searches for the optimum or near-optimum solution without using trial and

error for the initial values of parameters. Future work includes extending the proposed optimization method to the systems with multiple inputs and multiple outputs.

REFERENCES

[1] S.B. Cooper, K. Tiels, B. Titurus, and D. Di Maio, Polynomial nonlinear state space identification of an aero-engine structure, Comput. Struct., vol. 238, Article number: 106299, Oct. 2020.

[2] K. Wang, J.H. Chen, Z.H. Song, Y.L. Wang, and C.H. Yang, Deep neural network-embedded stochastic nonlinear state-space models and their applications to process monitoring, IEEE Trans. Neural Netw. Learn. Syst., vol. 33, no. 12, pp. 7682-7694, Dec. 2022.

[3] J. Decuyper, T. De Troyer, M.C. Runacres, K. Tiels, and J. Schoukens, Nonlinear state-space modelling of the kinematics of an oscillating circular cylinder in a fluid flow, Mech. Syst. Signal Process., vol. 98, pp. 209-230, Jan. 2018.

[4] W. Favoreel, B. De Moor, and P. Van Overschee, Subspace identification of bilinear systems subject to white inputs, IEEE Trans. Autom. Control, vol. 44, no. 6, pp. 1157-1165, Jun. 1999.

[5] A. Chiuso, The role of vector autoregressive modeling in predictor-based subspace identification, Automatica, vol. 43, no. 6, pp. 1034-1048, Jun. 2007.

[6] J.P. Noel, A.F. Esfahani, G. Kerschen, and J. Schoukens, A nonlinear state-space approach to hysteresis identification, Mech. Syst. Signal Process., vol. 84, pp. 171-184, Feb. 2017.

[7] J. Paduart, L. Lauwers, J. Swevers, K. Smolders, J. Schoukens, and R. Pintelon, Identification of nonlinear systems using polynomial nonlinear state space models, Automatica, vol. 46, no. 4, pp. 647-656, Apr. 2010.

[8] S.Y. Liu, X. Zhang, L. Xu, and F. Ding, Expectation-maximization algorithm for bilinear systems by using the Rauch-Tung-Striebel smoother, Automatica, vol. 142, Article Number: 110365, Aug. 2022.

[9] V. Stojanovic and N. Nedic, Joint state and parameter robust estimation of stochastic nonlinear systems, Int. J. Robust Nonlinear Control, vol. 26, no. 14, pp. 3058-3074, Sep. 2016.

[10] D. Whitley, A genetic algorithm tutorial, Stat. Comput., vol. 4, no. 2, pp. 65-85, Jun. 1994.

[11] P.F. Guo, X.Z. Wang, and Y.S. Han, The enhanced genetic algorithms for the optimization design, 3rd International Conference on Biomedical Engineering and Informatics, vol. 1-7, pp. 2990-2994, Oct. 2010.

[12] C.J. Lu, H.H. Zhang, T. Yue, and S. Ali, Search-based selection and prioritization of test scenarios for autonomous driving systems, 13th Symposium on Search-Based Software Engineering, vol. 12914, pp. 41-55, Oct. 2021.

[13] T. Wigren and J. Schoukens, Three free data sets for development and benchmarking in nonlinear system identification, European Control Conference (ECC), pp. 2933-2938, Jul. 2013.

[14] M. Schoukens and J.P. Noel, Three benchmarks addressing open challenges in nonlinear system identification, IFAC PapersOnLine, Vol. 50, no. 1, pp. 446-451, 2017.

[15] B.L. Miller and D.E. Goldberg, Genetic algorithms, selection schemes, and the varying effects of noise, Evol. Comput., vol. 4, no. 2, pp. 113-131, 1996.

[16] K. Deb and R. Agrawal, Simulated binary crossover for continuous search space, Comput. Syst., vol. 9, no. 2, pp. 115-148, 1995.

[17] K. Deb and M. Goyal, A combined genetic adaptive search (GeneAS) for engineering design, Comput. Sci. and Inform., vol. 26, pp. 30-45, 1996.

[18] J.M. Li and F. Ding, "Synchronous optimization schemes for dynamic systems through the kernel-based nonlinear observer canonical form," *IEEE Trans. Instrum. Meas.*, vol. 71, Article number: 6503213, 2022.