Nonlinear dynamic system identification with a cooperative population-based algorithm featuring a restart metaheuristic

C Brester\textsuperscript{1,2,3}, I Ryzhikov\textsuperscript{1,2}, V Stanovov\textsuperscript{2}, E Semenkin\textsuperscript{2}, and M Kolehmainen\textsuperscript{1}

\textsuperscript{1}Department of Environmental and Biological Sciences, University of Eastern Finland, Kuopio, Finland
\textsuperscript{2}Institute of Computer Science and Telecommunications, Reshetnev Siberian State University of Science and Technology, Krasnoyarsk, Russia
\textsuperscript{3}E-mail: christina.brester@gmail.com

Abstract. Dynamic system identification is commonly reduced to an optimization problem which is complex and multimodal. To find a global optimum of this problem, evolutionary algorithms are often applied. However, as it was shown in many studies, conventional evolution-based algorithms do not demonstrate the acceptable performance for this class of problems, therefore, some effective modifications have been proposed so far. In our study, we combine two approaches which were previously used in linear dynamic system identification and allowed their accurate identification. More specifically, we present a cooperative evolutionary algorithm with a restart metaheuristic and apply it for the parameter identification of a nonlinear cascaded system. The experimental results prove the effectiveness of the proposed evolution-based identification compared to other known solutions of this problem.

1. Introduction

System identification is a process of building mathematical models of dynamic systems based on measurements describing input and output signals. While identifying dynamic systems, differential equations are often chosen as a class of models enabling their accurate representation. On the one hand, there are some processes in biology, physics, and chemistry, which models have been already built in a parametric form [1]. In this case, identification implies finding optimal parameter values for a particular instantiation of a process or a system. On the other hand, for many more dynamic systems, a proper structure of differential equations needs identification as well. If first-order linear differential equations are not able to capture the complex system’s behaviour, more sophisticated model structures might be elaborated by using higher-order or nonlinear differential equations [2].

Generally, dynamic system identification is related to inverse mathematical modeling and might be reduced to an optimization problem. An optimized criterion usually reflects the difference between the measured data and the model output signal. In some studies, system identification was accomplished by solving a multi-objective optimization problem, which helped to introduce additional requirements for the model [3]. All in all, to find an optimum of this problem, either one-criterion or multi-criteria, stochastic population-based algorithms are often applied due to their helpful properties.

Previously, it has been shown that identification of linear dynamic systems requires advanced searching algorithms capable of escaping from multiple local optima. Otherwise, the algorithm converges quickly and never reaches the global optimum. Consequently, a number of modified heuristics have been proposed to cope with the premature convergence. As an example, cooperative...
population-based algorithms were successfully applied to build models for a single-input single-output (SISO) system as well as a multiple-input multiple-output (MIMO) system [3, 4]. The cooperation of different heuristics, working as parallel populations, allowed preserving a higher diversity within solutions and, as a result, provided the better models compared to the non-cooperative algorithms. Besides, in other studies, a restart metaheuristic was incorporated into evolutionary algorithms (EAs) as an additional operator to reinitialize a population trapped in a local optimum [5]. The restart metaheuristic showed its effectiveness in both single-objective and multi-objective EAs which were applied in identification of chemical reactions, specifically, the hexadecane disintegration reaction [6].

In this study, we present a modified EA which combines the useful properties of cooperative algorithms and restart metaheuristics and apply it for nonlinear dynamic system identification. The example considered is taken from systems biology, which encompasses a variety of complex modeling problems. Many parametric models have so far been developed to describe biological systems [1]. One of the most popular nonlinear models has a cascaded structure expressed as differential equations with the power-law nonlinearity. Cascaded mechanisms are inherent in various areas of biochemistry and physiology [7], therefore, many studies on identification are devoted to this model.

The rest of the paper is organized as follows. After the introduction, we provide the detailed description of the algorithm proposed in the Methods section. Then, we introduce the cascaded nonlinear dynamic system which has been chosen for this work. In the Results section, we present the solution obtained and compare it with other known solutions. Finally, we sum up our findings in the Conclusions section.

2. Methods
Since the identification problem and its reduction to the optimum seeking problem are complex, multimodal and with the large amount of attraction basins, it requires some specific techniques. We suggest applying cooperative algorithms with search controlling heuristics. The cooperation of different optimization algorithms commonly outperforms the distinct algorithms which form it. There are many different types of cooperation and many ways providing the computational resource allocation.

Let us consider the way the cooperation works. In this study, we applied an island-based cooperation, where islands were conventional optimization algorithms: evolutionary strategies (ES) [8], particle swarm optimization (PSO) [9], and differential evolution (DE) [10]. Specifically, there are three different DE algorithms with different parameters, one PSO, and one ES, which have been implemented in the PyGMO library [11]. All the islands are evolving independently. At each \( N_{migration} \)-th iteration, there is a local search performed for a randomly chosen solution from each island. After the local search, islands exchange their solutions (it is called migration). Each island’s worst solution gets randomly replaced with one of the best solutions from other islands. After that, we check if a restart condition is met.

It is well-known that global search algorithms have a risk to stagnate in local optimum basins. The restart operator detects stagnation of the search and reacts to it by manipulating algorithms. In the current case, it reinitializes populations. The stagnation detection is based on the fitness value improvement of the best alternative found during the algorithm run. We assume that the search is attracted to a local optimum if the best fitness value does not change noticeably for a number of iterations. If stagnation is detected, the restart adds the best found alternative to a solution’s collection, consequently, by the end of optimization, one has a number of different solutions of the problem.

Here we describe the way how the stagnation detection works. Let the best fitness value of the \( i \)-th generation be denoted as \( f_i^{best} \). This best value is updated after each migration, i.e. after each \( N_{migration} \) generations. A queue \( Q \) contains \( n_d \) differences \( d_k^f \) of two
consecutive best fitness values: \( d_k^f = \left| f_i^\text{best} - f_i^{(i-N_{\text{migration}})} \right| , \quad k = 1, \ldots, n_q \). The restart takes place when all \( d_k^f \) in the queue \( Q \) do not exceed a threshold \( d \):

\[
d_k^f < d, \quad \forall k, \quad k = 1, \ldots, n_q,
\]

\( n_q \in \mathbb{N} \) and \( d \in \mathbb{R} \) are the restart operator parameters.

The pair of restart parameters controls the operator sensitivity. The queue size \( n_q \) controls the number of generations in which there should be the required improvement. The larger this number is, the less sensitive to stagnation the restart operator becomes. The second parameter \( d \) controls the balance between exploration and exploitation. The smaller it is, the more the searching algorithm works on improving the accuracy of the optimization problem solution. Decreasing the queue size and increasing the limitation parameter will lead to an increase of the restart rate.

When restarting, all the populations of the algorithms in the cooperation are randomly generated and the restart operator queue is reset to an empty one.

The pseudocode of the approach proposed is given below:

The island-based optimization with migration, the local search, and the restart metaheuristic

1. initialize populations
2. initialize the restart operator
3. \( \text{migration\_counter} = 1 \)
4. while any computation resources do:
   4.1. migration\_counter++
   4.2. produce the next generations for all the islands
   4.3. if \( \text{mod(migration\_counter, N_{migration})} = 0 \) then
      4.3.1. randomly choose one solution from each island for the local search, 10 steps
      4.3.2. overwriting the worst solutions by randomly chosen best solutions from other islands
      4.3.3. update the restart queue
      4.3.4. if the restart condition is met then go to 1

In this pseudocode listing, one can see that the search continues while there are computational resources available (step 4). When the algorithm starts, we initialize all the islands with their populations and the restart operator: steps 1 and 2. Here we use \( \text{migration\_counter} \) (step 3) to show when we perform migration and improving solutions with the local search if the condition at step 4.3 is met. All the interactions and additional search heuristics take place at steps 4.3.1 – 4.3.2 while updating the restart operator and checking stagnation are at step 4.3.4.

3. Non-linear dynamic system and its identification

Cascaded systems have been known since 1960-1970 when researchers considered their role as amplifiers for biological signals [7]. Within the processes described with cascaded systems, there are the phosphorylase activation cascade in muscles, the bicyclic glutamine synthase cascade, and many other biochemical and physiological reactions. The generic representation of these systems includes a number of differential equations with power-low terms:

\[
X_i(t) = \alpha_i \prod_{j=1}^{n} X_j^{g_{ij}}(t) - \beta_i \prod_{j=1}^{n} X_j^{h_{ij}}(t),
\]

where \( X \) is a vector of \( n \) dependant or independent variables, \( X \in \mathbb{R}^n \), \( \alpha \) and \( \beta \) are vectors of \( n \) non-negative constants, \( \alpha, \beta \in \mathbb{R}_{\geq 0}^n \), \( g \) and \( h \) are matrices of powers (kinetic orders), \( g, h \in \mathbb{R}^{nn} \). In
practice, many \( g_{ij} \) and \( h_{ij} \) equal 0, and the structure identification simply means finding non-zero elements of matrices \( g \) and \( h \).

In this work, we start with some known cascaded structure and define its parameters. The system considered was presented earlier in [12] and it is given below:

\[
X'_1 = \alpha_1 X^g_{41} X^g_{31} X^g_{41} - \beta_1 X^h_{11}, \\
X'_2 = \alpha_2 X^g_{11} - \beta_2 X^h_{22}, \\
X'_3 = \alpha_3 X^g_{21} - \beta_3 X^h_{33},
\]

(3)

where \( X_1, X_2, X_3 \) are independent variables and \( X_4 \) is an independent variable. The true constants \( \alpha, \beta \) and kinetic orders \( g, h \) are presented in table 1:

**Table 1.** The system parameters to be identified.

| \( \alpha_j \) | \( \beta_j \) | \( g_{i1} \) | \( g_{i2} \) | \( g_{i3} \) | \( g_{i4} \) | \( h_{i1} \) | \( h_{i2} \) | \( h_{i3} \) |
|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| 10.0        | 5.0         | -0.1        | -0.05       | 1.0         | 0.5         |             | 0.5         |             |
| 2.0         | 1.44        | 0.5         |             |             |             |             |             |             |
| 3.0         | 7.2         | 0.5         |             |             |             |             |             |             |

In figure 1, the system considered is shown. There is a three-step cascade with feedbacks: the precursor (independent variable) \( X_4 \) activates the process and the metabolites (dependant variables) \( X_1, X_2, X_3 \) are synthesized. Also, metabolites at the earlier steps influence metabolites synthesis later in the cascade.

![Figure 1. The cascaded system considered in this study.](image)

Since we consider the dynamic system, we need to solve the Cauchy problem: Equation (3) and some known initial value \( X_1^0, X_2^0, X_3^0, X_4^0 \), so it gives the numerical solution that we can compare with the observation data. It is possible that for a single initial value there are many different parameters for which the model solution is the same (to some extent) as the observation data. To lessen the amount of possible solutions, we increase the number of samples by adding the observation data for different initial values. As a result, we have the number of samples \( n_{\text{samples}} \), each sample consists of 4 time series (3 for the dependent variables and 1 for the independent one) of the size \( n_{\text{obs}} \) each.

We can evaluate the solution adequacy by the criterion:

\[
I(\theta) = \frac{1}{n_{\text{samples}}} \sum_{k=1}^{n_{\text{samples}}} \max_{1 \leq i \leq 3, 1 < j \leq n_{\text{obs}}} \left| \hat{X}_{k,i}(\theta, t_j) - Y_{k,i}(t_j) \right|,
\]

(4)
where $\hat{X}_{k,i}(\theta, t_j)$ is the solution of the Cauchy problem of System (3) and the $k$-th initial value for parameters $\theta = (\alpha, \beta, g, h) \in \mathbb{R}^{14}$, and $Y_{k,i}(t_j)$ is the observations of the $i$-th output of the $k$-th sample at the $t_j$ time point.

In this study, we used two different approaches for solving the initial value problem: the Runge-Kutta 4-th order numeric integration scheme [13] and the collocation approximation approach described in [14]. We used the collocation approximation approach in the global search and the local search before migration. After all, the final solutions were improved by the local search, where the criterion was calculated on the basis of the Runge-Kutta integration scheme.

The next section provides the experimental results obtained while identifying parameters of this cascaded system.

4. Results
We conducted an experiment according to the conditions described in [12] to compare the result obtained in the current study with other known results for the cascaded system shown in figure 1 with the parameters from table 1. First, the system dynamics was simulated to collect measurements of dependent and independent variables. The set of 10 random initial values was generated from intervals: $X_1^0, X_2^0, X_3^0 \in [0.1,0.6]$ and $X_4^0 \in [0.6,0.9]$. The Runge-Kutta numerical integration scheme was applied to generate the data: 100 points in the time interval $[0,10.0]$ with the step of 0.1. Thus, the collection of samples was created and used to fit Model (3).

The following boundaries of the parameter values were defined for the searching algorithm: $\mathcal{A}_i$ and $\mathcal{B}_i \in [0.0,20.0]$, $g_{ij}$ and $h_{ij} \in [-4.0,4.0]$. In general, there are 27 parameters which require identification. However, in this article, we assume that the system structure is known, therefore, 14 parameters should be found. There are studies devoted to the structure identification of System (3) but we consider this problem as the next step of our research.

The cooperative EA with the restart metaheuristic was provided with 250 000 function evaluations. Each population size was equal to 20. Every 5-th generation, the local search (sequential quadratic programming) and migration were applied as well as the restart criterion was checked. The restart operator had the following parameters: $d = 0.001$ and $n_g = 5$. The following settings of the algorithms included in the cooperation were used (table 2):

| Algorithm                                      | Settings                                      |
|------------------------------------------------|-----------------------------------------------|
| 1 Differential Evolution (DE)                  | $F = 0.3$, $CR = 0.9$, the rest is default values of pygmo |
| 2 Differential Evolution (DE)                  | $F = 0.6$, $CR = 0.9$, the rest is default values of pygmo |
| 3 Differential Evolution (DE)                  | $F = 0.9$, $CR = 0.9$, the rest is default values of pygmo |
| 4 Evolutionary Strategies                      | default values of pygmo                      |
| 5 PSO                                           | default values of pygmo                      |

After the global search performed with the cooperative EA, all the solutions found were improved by the local search (sequential quadratic programming) using the Runge-Kutta integration scheme. The local search stopped if the criterion $|a_k - a_{k-1}| < \delta \cdot |a_k|$ was met for all the parameters estimated, where $a_{k-1}$ and $a_k$ are the parameter values in two consecutive steps. The total number of function evaluations in the local search was equal to 2924. Finally, the best solution was selected for testing on validation samples generated with two sets of initial values: $[0.5,0.2,0.5,0.9]$ and $[0.72,0.08,0.72,1.05]$. The first initial values were from the intervals used for training, whereas the second initial values did not belong to the training intervals.
In Table 3, we present the final best solution achieved in the current study (model I). Also, there are two more solutions introduced in [12]: these parameter values were obtained by Hybrid Differential Evolution using collocation approximation (model II) and slope approximation (model III) in the first and second cases respectively. These two solutions were reached with the global search (314 022 function evaluations) and additionally improved with the local search. For the first set of initial values, Criterion (4) was equal to 0.0174 for model I, 0.0496 for model II, and 1.0262 for model III. Then, for the second set of initial values, it was equal to 0.0513 for model I, 0.0663 for model II, and 1.4365 for model III. Figures 2, 3, and 4 illustrate the difference between the cascaded system output and the output of models I, II, and III for two sets of initial values. As we can see, models I and II fit with the system output much better than model III does.

Table 3. The solution found in this study and two other known solutions.

| Approach                      | $\alpha_i$ | $\beta_i$ | $g_{i1}$ | $g_{i2}$ | $g_{i3}$ | $g_{i4}$ | $h_{i1}$ | $h_{i2}$ | $h_{i3}$ |
|-------------------------------|------------|-----------|----------|----------|----------|----------|----------|----------|----------|
| Cooperative EA with restart   | 9.26       | 4.19      | -0.12    | -0.05    | 1.11     | 0.56     |          |          |          |
| HDE with collocation          | 10.06      | 4.98      | -0.107   | -0.045   | 1.00     | 0.50     | 0.48     |          | 0.54     |
| HDE with slope approximation  | 1.25       | 8.05      | 1.06     | 1.06     | 1.06     | 1.06     |          |          |          |
| HDE with slope approximation  | 12.04      | 4.58      | -0.261   | 0.107    | 1.00     | 0.486    | 0.576    |          | 0.483    |
|                               | 1.65       | 1.12      | 0.569    |          |          |          |          |          |

Figure 2. The output of the cascaded system (solid) and the output of the model built in this study (dashed). The left figure corresponds to the first set of initial values and the right figure corresponds to the second set of initial values.
In spite of the fact that in the current study, we compare the model built only with the results presented in [12], a number of other experiments with non-cooperative EAs and no restarting were conducted as well. However, all the results attained were much worse than ones from [12], therefore, the comparison includes only the best other known solution.

Additionally, we should admit that although our solution outperformed models II and III in the current experiment, the future investigations should include not only the system parameter identification but also the structure identification. All in all, the results obtained prove that cooperative algorithms and restarting are effective not only for linear dynamic system identification, as it was previously shown, but also for identification of nonlinear dynamic systems such as cascaded systems.

5. Conclusions
This study presented the results of the cascaded system identification with the cooperative evolutionary algorithm featuring the restart metaheuristic. The idea of using these techniques was transferred from the studies devoted to linear dynamic system identification, where cooperative EAs and restarting were successfully applied.
We compared the model built with two known solutions and found that it could outperform them. Generally, the experimental results are promising in relation to the use of cooperative EAs and restarting in nonlinear dynamic system identification.

However, in the future studies, we should accomplish not only the system parameter identification but also the structure identification to make the comparative analysis thorough.

Acknowledgements
The reported research was funded by Russian Foundation for Basic Research and the government of the region of the Russian Federation, grant № 18-41-243007.

References
[1] Gennemark P and Wedelin D 2009 Benchmarks for identification of ordinary differential equations from time series data Bioinformatics 25(6) pp 780–786 doi:10.1093/bioinformatics/btp050
[2] Ryzhikov I, Semenkin E and Panfilov I 2016 Evolutionary optimization algorithms for differential equation parameters, initial value and order identification Proceedings of the 13th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2016) 1 pp 168–176
[3] Ryzhikov I, Brester Ch and Semenkin E 2017 A novel linear time invariant systems order reduction approach based on a cooperative multi-objective genetic algorithm Advances in Swarm Intelligence pp 23–30 doi: 10.1007/978-3-319-61833-3_6
[4] Ryzhikov I, Semenkin E and Akhmedova Sh 2016 LTI system order reduction approach based on asymptotical equivalence and the Co-operation of biology-related algorithms IOP Conference Series: Materials Science and Engineering 173 1 doi:10.1088/1757-899X/173/1/012014
[5] Brester Ch, Ryzhikov I and Semenkina O 2018 Generic scheme of a restart meta-heuristic operator for multi-objective genetic algorithms International Journal on Information Technologies & Security 2(10) pp 101–110
[6] Ryzhikov I, Brester Ch and Semenkin E 2017 Multi-objective dynamical system parameters and initial value identification approach in chemical disintegration reaction modelling. Proceedings of the 14th International Conference on Informatics in Control, Automation and Robotics pp 497–504 doi: 10.5220/0006431504970504
[7] Voit E O 2000 Computational Analysis of Biochemical Systems Cambridge University Press, Cambridge, UK
[8] Schwefel H-P 1995 Evolution and Optimum Seeking Wiley
[9] Kennedy J and Eberhart R 1995 Particle Swarm Optimization Proceedings of IEEE International Conference on Neural Networks pp 1942–48
[10] Storn R and Price K 1997 Differential evolution—a simple and efficient heuristic for global optimization over continuous space Journal of Global Optimization 11 pp 341–359 doi: 10.1023/A:1008202821328
[11] Biscani F, Izzo D and Märtens M 2018 Pagmo 2.7 doi: 10.5281/zenodo.1217831
[12] Tsai K Y and Wang F S 2005 Evolutionary optimization with data collocation for reverse engineering of biological networks Bioinformatics 21(7) pp 1180–88 doi:10.1093/bioinformatics/bti099
[13] Virtanen P, Gommers R and Oliphant T E, et al. 2019 SciPy 1.0 Contributors SciPy 1.0—Fundamental Algorithms for Scientific Computing in Python preprint arXiv:1907.10121
[14] Wang F S and Sheu J W 2000 Multiobjective parameter estimation problems of fermentation processes using a high ethanol tolerance yeast Chemical Engineering Science 55 pp 3685–95