Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 193 (2021) 285–294

Procedia Computer Science

www.elsevier.com/locate/procedia

10th International Young Scientists Conference on Computational Science

# Multi-objective closed-form algebraic expressions discovery approach application to the synthetic time-series generation

Mark Merezhnikov, Alexander Hvatov*

*National Centre for Cognitive Research, ITMO University, 197101, 49 Kronverksky pr., St Petersburg, Russia*

**Abstract**

Time-series modeling is a well-studied topic of classical analysis and machine learning. However, large datasets are required to obtain the model with a better prediction quality with the increasing model complexity. Therefore, some applications demand synthetic datasets that are preserving modeling-sensitive properties. Another application of synthetic data is data anonymization. The synthetic data generation algorithm may be split into two parts: the time-series modeling and the synthetic data generation parts. The model must be interpretable to obtain the synthetic data with good quality. The model parameter interpretation allows controlling generation by adding noise to different groups of parameters. In the paper, the evolutionary multi-objective closed-form algebraic expressions discovery approach that allows obtaining the model in the form that may be analyzed using the mathematics is proposed. The analysis allows the interpretation of the model parameters for the controllable generation of the synthetic data. The notion of synthetic data quality is discussed. The examples of the synthetic time-series generation based on two datasets with different properties are shown.

*Keywords:* model interpretation; evolutionary algorithm; synthetic data; algebraic equation discovery

## 1. Introduction

Time-series forecasting is a classical problem that arises in various areas. Starting from classical ARMA models [3] to the most advanced usage of the complex neural networks [7]. The simple and advanced models require observational data for parameter estimation. However, the amount of data is different. In particular, most modern machine learning methods require much data to show a good quality of prediction.

Moreover, some companies want to anonymize their data before passing it to the researchers. One of the possible solutions is to create the synthetic dataset based on the properties of the actual data [4], [1]. In this case, the properties essential for the machine learning methods such as statistical distributions or spectra should be preserved as much as possible.

* Corresponding author
*E-mail addresses:* mark.merezhnikov@mail.ru (Mark Merezhnikov)., alex˙hvatov@itmo.ru (Alexander Hvatov).

10.1016/j.procs.2021.10.029

Usually, time-series modeling is split into two parts. The first part of the generation of synthetic data is usually considered within the time-series modeling framework. Different models from classical trend-season-noise decomposition [10] to most advanced architectures of autoencoders [11] may be used for the modeling part. However, not every type of model may be properly used for time-series generation. One of the most important properties is the model parameters' mathematical stability, as the second input data stability. Many other properties increase the quality of the synthetics, such as interpretability and the ability to control the modeling and the synthetics generation process.

The properties of the chosen models are essential for the second part, which is the generation of synthetic data. The synthetics may be generated by adding noise to the parameters or the data [5]. Both approaches may be used simultaneously. However, it does not necessarily increase the quality of the results.

We propose the multi-objective approach that allows generating the algebraical expressions using the observations data. The first part is the time-series modeling in the form of algebraic equations. There are many modeling algorithms [2] that allow to end up with the mathematical expression based on a given input data. However, at this stage, additional objectives allow to tune the modeling process and obtain the Pareto frontier of the models analyzed by an expert and used to generate synthetic data.

One may note that there are a lot of different modeling problems in the applied fields that are solved using algebraic expression generation. There are exist both classical applications in hydrometeorology [13, 12] and modern examples in quantum physics [9]. However, in most cases, the form of the equation is known completely, so only coefficients are identified, or the discovery process is used to add data-specific terms to the known equations. In the previous work [8], we show that described in the paper algebraic equation discovery process does not have any a priori knowledge about the process. As additional input data, only simple building blocks, such as simple parameterized functions, are used.

One of the algebraic expression properties is the model interpretability since the form of the obtained model may be understood by a human, and every coefficient may be analyzed from a mathematical point of view. The possibility to interpret the system allows controlling both the modeling and synthetics generations process.

The paper is organized as follows. Section 2 describes the main principles of the synthetic time-series generation. Section 3 contains the description of the current algorithm realization. Section 4 contains two examples of the time-series generation on datasets with different properties. Section 5 outlines the paper.

## 2. The problem formulation

In the paper, we describe the approach that generates synthetic time-series data. Most of the time-series generation approaches can be separated into two independent parts: time-series modeling and time-series generation.

*Time-series modeling.* The input of an arbitrary time-series modeling algorithm is an observation of some continious process $P(t)$. It is assumed that $P(t)$ is defined only on a discrete time-steps $t_i$, i.e. $P(t_i) = p_i$. The resulting input for the time series modeling algorithm may written in form $\bar{P} = \{p_1, p_2, ...p_n\}$, where $n$ is the length of the time series.

We define the time-series model as the map $M(\theta; p_{i-T}, ...p_{i-1}) - > p_i, ..., p_{i+H}$, where $\theta$ is the vector of the model parameters. We assume that parameters $\theta$ could be changed such that the error between the model prediction $\bar{M} = \{\bar{p}_T, ..., \bar{p}_n\}$ and observations $\bar{P}$ is minimal. We note that the method how the prediction itself (since there may be prediction overlaps) and prediction error is calculaed may be chosen in many ways.

*Mathematical expression model.* As the mathematical model, we understand the model that can be represented as a sum of products of integrable functions. Therefore, we do not stick to the given form of the equation. However, without loss of the generality, we assume that the resulting model has the form Eq.1.

$$M(t) = \sum_{i=1}^{i=L} c_i * a_t(t) \tag{1}$$

In Eq.1 coefficiencts $c_1$ are the constants, functions $a_1(x)$ are the products of the integrable functions. To reduce the problem, ususally the subclass $T$ of the functions is defined to form the prodcunts, $L$ is the maximum expression length.

We note that the subclass $T$ may be defined in a discrete manner, such that functions $\cos t$ and $\cos(2t)$ are different. Also, $T$ may be defined in a continuous way such that $\cos(\omega t)$ is one equivalence class. Even though it is an equivalence class, we assume that several class elements may present in the resulting model.

*Multi-objective optimization.* Definitions above may be used to formulate of the data-driven algebraic equation discovery problem. We use the part of the dataset $\bar{P}_{train} \subset \bar{P}$ to define functions $a_i(t)$ and conefficients $c_i$ in form Eq.1.

Usually, it is done as an optimization problem. Therefore, we introduce the "quality" metric. First we take rest of the dataset $\bar{P}_{test} = \bar{P} - \bar{P}_{train}$ and compute model at the same timesteps $\{t_{test}\}$ = as the $\bar{P}_{test}$, thus the set of the model responses $\bar{M}_{test}$ is obtained.

We formulate optimization problem as:

$$M(t) = \arg \min_{a_i \in T^L, c_i \in R_L} ||\bar{P}_{test} - \bar{M}_{test}|| \tag{2}$$

In Eq. 2 $T^L = T \times ... \times T$ is the Cartesian product of the sets of the possible equations, $|| \cdot ||$ is an arbitrary distance between the two discrete time-series.

The multi-objective formulation allows introducing more control to the optimization process and thus allows tuning the model in various ways. For example, for some problems, the time-series detailed reproduction precision leads to the "overfitting" of the model to the given observation data. Thus, the ability to balance model complexity and prediction quality may be useful for applications.

The first group of the objectives we refer to as "quality". For a given model $M_i$, the quality metric is the distance between observed and the predicted time-series (Eq.3)

$$Q(M_i) = ||\bar{P}_{test} - \bar{M}_{itest}|| \tag{3}$$

The second group of objectives we refer to as "complexity". For a given model $M_i$, the complexity metric Eq.4 is bound to the number of elements from the subclass $T$ in the model that is denoted as $\#(M)$

$$C(M_j) = \#(M_j) \tag{4}$$

The objective functions $Q(M), C(M)$ form the optimization space used to form the Pareto frontier with the multi-objective optimization algorithm.

## 3. The algorithm description

This section describes a multi-objective evolutionary optimization algorithm for time-series modeling and subsequent synthetic time-series generation. We follow the problem statement and describe the algorithm in two separate parts: building a generative model during time-series modeling and creating synthetics based on the generative model. First part can be implemented with single-objective (Sec. 3.1) based on [8] or multi-objective (Sec. 3.2) evolutionary optimizations based on [6]. Thereafter the synthetics is generated by varying the model parameters (Sec. 3.3)

### 3.1. Single-objective evolutionary optimization

Single-objective algorithm follows the formulation Eq. 2. The building algorithm is aimed to discover the mathematical expression $\bar{M}(t)$, closest to the initial time series. In the article, normalized variance $D$ of the difference vector Eq. 5 is used as the distance.

$$Q(M_i) = \frac{D(\bar{P} - \bar{M}_i(t))}{D(\bar{P})} \tag{5}$$

The algorithm detects the presence of trend components and periodic components and patterns using sets of the functions specified by the user. The sum of power functions forms a polynomial describing the trend or the lower-frequency component of the time series. Periodic functions such as pulses and trigonometric functions are used to describe seasonal or higher-frequency components. The result of the algorithm is a mathematical model, an analytical expression that determines components found in the time series.

We describe subclass $T$ of the functions used for the optimization in terms of tokens. The tokens are essentially the building blocks for the resulting mathematical expression, as shown in Eq. 1. We define tokens in the form of a parameterized mathematical function. The example of the token is shown in Eq. 6.

$$a_1(p_1, p_2, p_3, t) = p_1 * \sin(p_2 * t + p_3) \tag{6}$$

The number of parameters $p_i$ within the token is pre-defined and dependent on the type of functions included in the token. We note that amplitudes $c_i$ from Eq. 1 are used as the token's parameter. It is assumed that the resulting expression is the sum of the products of tokens.

Therefore, the problem is described as follows, let $\bar{M}(\bar{p}, t) = \sum\limits_I \prod\limits_J a_{i,j}(p_1^{(i,j)}, ..., p_k^{(i,j)}, t)$, where is the resulting model, where $I = 1, 2, 3, ...$ and $J = 1, 2, 3, ...$ multi-indices, that determine the token type, $k$ is the fixed number of the parameters for the given token type and $\bar{p} = (p_1^{(1,1)}, ...)$ is the model multi-parameter. Using the model $\bar{M}(\bar{p}, t)$ definition we can formulate the optimization problem Eq. 7

$$M(t) = \arg \min_{a_{i,j} \in T, \bar{p}} Q(\bar{M}(\bar{p}, t)) \tag{7}$$

It is assumed that the number of tokens $a_{i,j} \in T$ used in the model is varied during the optimization problem that means that the token multiparameter $\bar{p}$ dimensions are varying.

The building algorithm can be divided into three parts. The first part is token optimization, which is the preliminary step for evolutionary optimization and proper fitness function calculation. The second part is evolutionary optimization, which allows changing the number of tokens using the mutation and cross-over operators. Finally, the third part is the regularization that allows reducing expression by canceling out the insignificant tokens.

*Token parameter optimization.* A critical part of the algorithm is the numerical optimization of all tokens' parameters. Parameters form the chromosomes of individuals in the evolutionary algorithm. Every chromosome that is passed after the evolutionary part should obtain the optimal set of parameters. The quality of the parameters' set affects the probability of removing or preserving a token from the chromosome during regularization, and as a result, it affects the growth rate of individuals' chromosomes and the convergence rate of the algorithm.

We will not provide the details for brevity and state that every token may require different algorithms to optimize its parameters (more detailed they are described in [8]). Below we describe two methods used as part of the algorithm within the tokens' parameters optimization procedure.

The periodic tokens are optimized using the spectral analysis of the initial time series. If the token has a frequency parameter, it gets a frequency value from one of the harmonics with a large amplitude. Other parameters are optimized using different optimizing methods such as Gradient Descent or Differential Evolution. Also, the spectral analysis prevents appearing of unexpectable tokens in the chromosome. For example, there will be no trend token if there are no low-frequency significant harmonics in the spectrum.

Impulse tokens have the opportunity to approximate close to periodic components. To do this, every single pulse in the token undergoes additional parameters optimization on the allocated time interval, which usually is equal to the token period. It allows to approximate non-periodic peaks. That sort of token is critically important in some cases and called complex impulse.

*Evolutionary part.* Another proposed time-series modeling algorithm is an evolutionary algorithm with a population consisting of individuals with arrays of tokens as chromosomes. A token is a gene in a chromosome. The sum of the array elements forms the final analytical expression. Thus, individuals are competing in the train data representation quality. We define the fitness of an individual as the Eq. 5. Overall it is the evolutionary approach to solve single-objective optimization problem Eq. 7. Consistent approaching optima can solve it. Thereby individuals in the initial population have short chromosomes. Influenced by genetic operators, chromosomes are expanded with new tokens.

The cross-over operator allows two selected parents to exchange random tokens among themselves or expand their chromosomes. The mutation operator similarly acts on the selected individual – randomly generated tokens replace some tokens in the chromosome or expand it. Selection is implemented using roulette wheel selection and elitism.

The intensity of genetic operators affecting the number of tokens involved in the operation and the probability of chromosome expansion are meta-parameters.

Qualitative optimization of token parameters in individual chromosomes is likely to produce the same results for different individuals. Therefore, it is expected that diversity is reduced to a minimum. In addition, the population size is preferred to be small to eliminate an overhead associated with solving similar problems of optimizing token parameters.

The maximum number of evolution iterations and the critical fitness value is used as the stop criterion.

*Regularization part.* During the evolutionary part of the algorithm, excessive chromosome expansion may appear. It means that the tokens can start approximating the noise component of the initial time series or appeared as a result of an inaccurate approximation on previous steps. Thus, chromosome length regularization is necessary to avoid model so-called overfitting.

Previously [8], we use LASSO regression to reduce model complexity and thus filter out the insignificant components. Overall we may use an arbitrary regularization operator to remove tokens with close frequencies and tokens that make the lowest contribution to an individual's fitness (initial time series approximation).

As it turned out, the LASSO regularization is not universal. It removes high correlated features, but tokens with close frequencies have high correlation only on short time intervals. It removes features whose contribution to the target approximation does not justify their amplitude. Nevertheless, it means that, for example, narrow and sharp impulses with a high amplitude have fewer chances to stay in the chromosome than wide impulses with a low amplitude. It is necessary to evaluate the token's utility by contributing to the individual's fitness without considering its amplitude because tokens can take a wide range of forms.

Moreover, the regularization coefficient is the LASSO meta-parameter. There is no unambiguous and understandable relationship between it and the scale of the tokens' amplitude removed. Small changes in the coefficient can give substantial changes in the resulting model and vice versa. The magnitude of the regularization coefficient depends on the maximal model length and the input data properties. Thereby, the expert has to do some experiments to define the regularization coefficient's interval in which acceptable results are obtained. Usually, it is a computationally expensive procedure.

We use an alternative approach for regularization. First, for every gene of the chromosome, its contribution to individual fitness is evaluated. Then genes are sorted, and part of the less important genes are removed. The ratio of the genes removed is the new meta-parameter. Also, we perform the regularization procedure for genes with close frequencies. Finally, tokens with less contribution are removed. This approach is easier to implement and lacks all of the above disadvantages. It is also better suited for multi-objective optimization, as we discuss this topic in the corresponding section.

### 3.2. Multi-objective evolutionary optimization

The best model in both the modeling of the original series and the generation of the synthetic is not the same. Therefore, for the synthetic generation application, single-objective evolutionary optimization is not always the proper choice. Multi-objective evolutionary optimization may control the optimization process. As a result of the better control, we may simultaneously obtain good models and good synthetic data. The Pareto frontier within the quality-complexity objective space gives the set of models, which may be used either for modeling or synthetics generation.

Our realization of a multi-objective evolutionary optimization algorithm for closed-form algebraic expression discovery is based on a single-objective evolutionary algorithm (Sec. 3.1). However, individuals' chromosomes are extended with an isolated gene that contains the regularization parameter. Individual fitness consists of two objectives. The first one is the approximation quality Eq. 5 as before. The second one is the tokens number in the individual chromosome Eq. 4. Tokens parameter optimization and regularization remain unchanged.

Cross-over and mutation operators are expanded by the corresponding operations with the regularization parameter gene. Individuals can exchange this gene among themselves during cross-over. In addition, the gene can change its value in specified ranges during mutation.

The work with population is based on an evolutionary multi-optimization algorithm based on dominance and decomposition [6]. This approach gives the Pareto frontier of models along objective axes and the opportunity to choose non-overfitted models with the best quality metrics of the approximation and synthesis.

Here, the inappropriateness of using the LASSO regularization operator becomes more transparent. Furthermore, due to the high sensitivity of the chromosome length to the regularization parameter and its wide working range, many iterations of the influence of evolutionary operators on the regularization gene will be required to obtain a high population diversity in terms of the second objective. On the other hand, the maximum length of the chromosome as the regularization parameter does not have this problem. So it is chosen as the regularization gene in this work.

The final algorithm is an evolutionary algorithm, in which the population consists of individuals with arrays of tokens as chromosomes. The algorithm scheme is shown in Fig. 1. For each individual, all tokens are optimized during iterations. There is a sequential growth of individual chromosomes due to genetic operators. Excessive growth of chromosomes is restrained using regularization. The sum of tokens in the chromosome of the fittest individual is the final mathematical model describing all components and patterns found in the considered time series.

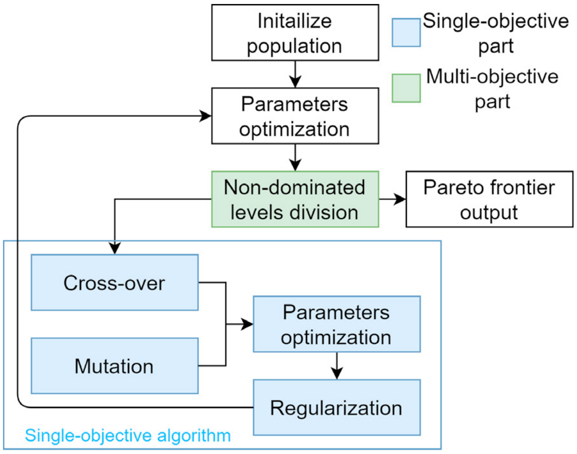


Fig. 1: The modeling algorithm's resulting scheme

### 3.3. Synthetic generation algorithm

Based on the above approaches, a generative model can be constructed. The model in the form of an algebraic expression can give a wide range of synthetic data by varying tokens parameters. There are two synthetics generation algorithms, the first one for Complex impulses and the second one for other tokens described in the first part. The choice of the quality metric to assess the generated synthetics is described in the second part.

*Token parameters varying.* We add noise or some time-depended function to their parameters to obtain synthetic data for most of the tokens. Amplitude, frequency, and phase may be perturbed directly by adding noise components to the corresponding parameters. We note that high-frequency components are susceptible to frequency and phase variation. Therefore we may add noise only to the low-frequency components' parameters like a phase in periodic tokens to get a synthetic with higher time-series diversity.

The complex impulse is a token with a sequence of single pulses with different parameters. The main difference of complex impulse tokens that besides varying their parameters, there is the ability to vary their order in the sequence. For the synthetics generation purpose, single pulses from each complex impulse in the model are clustered by their forms and time gaps between them. Pulses from all Complex impulses in the model additionally are clustered according to their belonging to complex impulses. Thus, in total single pulses within the complex one are clustered in three-dimensional space.

Based on the sequence of states in the model, the Markov chain trains and can predict the next state based on the previous state. Synthetic data are obtained by decoding states generated by the chain. The state decoding gives a single random pulse from the corresponding cluster. The cluster size is the generating algorithm meta-parameters. The smaller the size of the clusters, the more deterministic the chain is and the less variability of the synthetic. Therefore, we obtain additional control over the synthetics generation process by varying meta-parameters.

*Quality metric.* It is not easy to assess the quality of synthetic data in an automated manner. On the one hand, we require synthetics to preserve the properties of the original series, but at the same time, we require as many differences as possible to achieve diversity.

During the work satisfying the several metrics were tested. We chose the Fourier specter proximity of the synthetic data to the original data (as the proximity measure, we chose variance of the difference, L2-norm of the difference, correlation). The spectrum was chosen because it not only contains information about the main structure of the series (trend, seasonal, and noise components) but, unlike time and frequency-time representations, is invariant to the non-stationarity of the series if the trend is removed correctly. Specter also allows one to create synthetics for the non-stationary processes since we may add noise to the spectral part without changing the non-stationary part.

However, using only the specter does not reflect the quality of the synthetics since the generating algorithm perturbing the specter. Therefore, specter quality metric and synthetics diversity are in contradiction. We consider initial and synthetic time-series as the random variables and compare values probability density distribution as another method. In addition to the quality metric and synthetics diversity contradiction, a uniqueness problem arises. It means that the entirely different time series might have similar distributions, and the first requirement would not be met.

As a result, the proximity to the initial data by itself is not the best measure for the quality of the synthetic data. The data predicted by the model almost always had a spectrum closer to the original one than the synthetics based on it. Therefore, from the point of view of the chosen metric, it would be more efficient not to vary the model parameters to get the highest quality synthetics. We modify the quality metric by adding a value reflecting the dissimilarity of the synthetic and original data in the temporal area.

Since we have already used the Eq. 5 to measure the proximity of the model and the original series in the time domain, a logical step is to apply the same formula to the spectra for an equivalent contribution of each part to the final assessment of the quality of synthetics. The resulting metric $Q$ is the ratio of the normalized variance of the difference between the synthetic $\bar{X}$ and the original data $\bar{P}$ spectra $S$ to the normalized variance of their difference in the temporal area. It is shown in Eq. 8. So the smaller the spectrum difference and the larger the time series difference, the smaller the metric and the better the synthetics.

$$Q(\bar{X}) = [\frac{D(S(\bar{P}) - S(\bar{X}))}{D(S(\bar{P}))}][\frac{D(\bar{P} - \bar{X})}{D(\bar{P})}]^{-1} \tag{8}$$

The synthetics quality metric Eq. 8 provides the best synthetics data based on expert analysis.

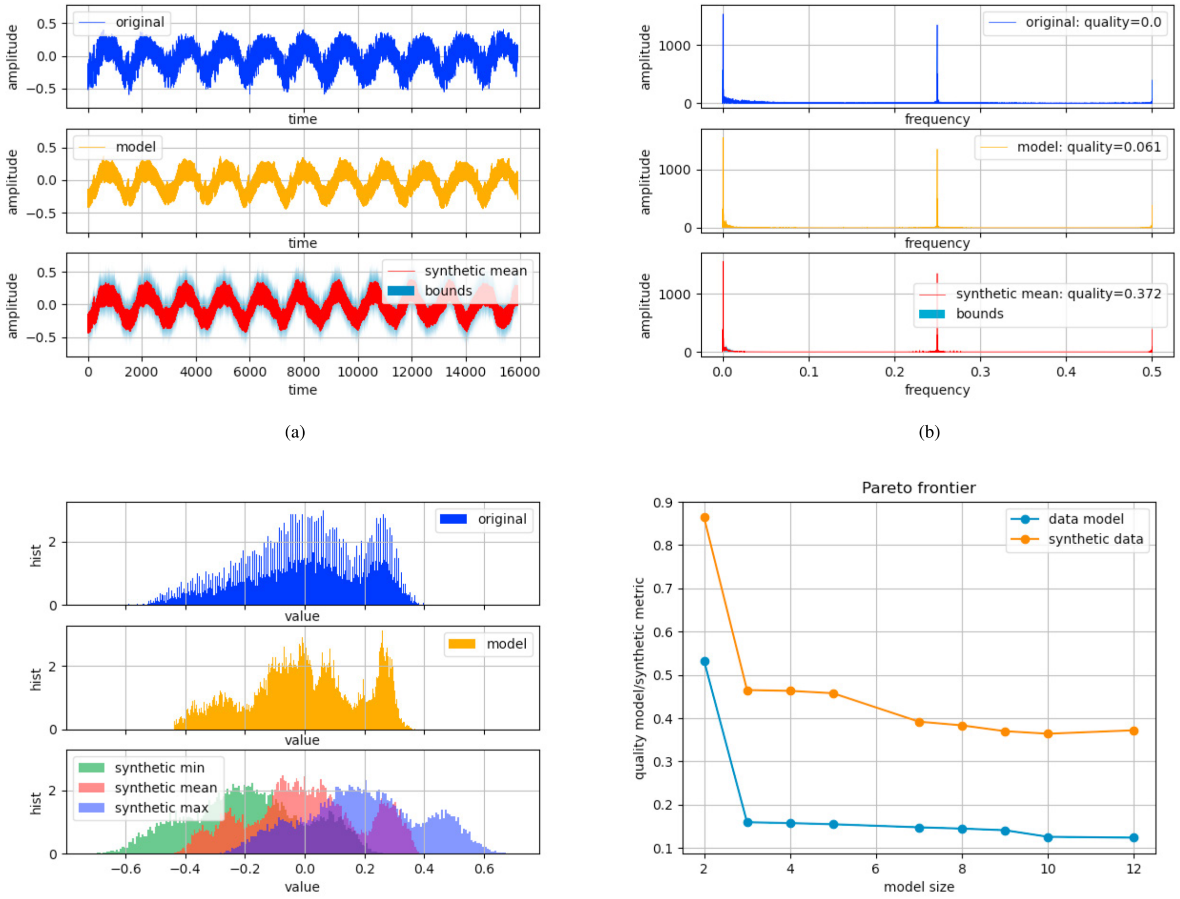
## 4. Experiments

We chose two different time series representing two different real-world processes to generate synthetic datasets. The first one is tropical temperature changes during some time and has periodic structure (Sec. 4.1). The second one is the electrical power consumption of some areas and does not have a clear periodic structure. It is needed to show Complex impulse tokens capabilities in particular (Sec. 4.2).

### *4.1. Time-series with significant seasonality*

The temperature variability time-series can be accurately approximated by periodic tokens such as sine and cosine functions. However, we cannot assess how complex the optimal model may be without preliminary analysis of the series. Therefore the chromosome size should not be limited by regularization. Using a single-objective algorithm with the regularization coefficient that allows for a considerable chromosome length, the generative model with a size of 12 tokens is obtained. The 50 synthetic samples are generated based on it. The resulting model and synthetics data are shown in Fig. 2.

As we can see, the original time series has three main spectral harmonics, but the model has a lot more tokens. This means that most tokens have minimal amplitude, approximate noise and unnecessarily increase the complexity of the model. As a result, the synthetic data has a slight variation due to the high determinism of the series and the low contribution of most tokens. Thus, the model is not optimal in its complexity and the quality of its approximation and generation. To obtain a more optimal model using a single-objective model, we have to restart the algorithm with the changed meta-parameters based on the conclusions made.



(a)

(b)

(c)

(d)

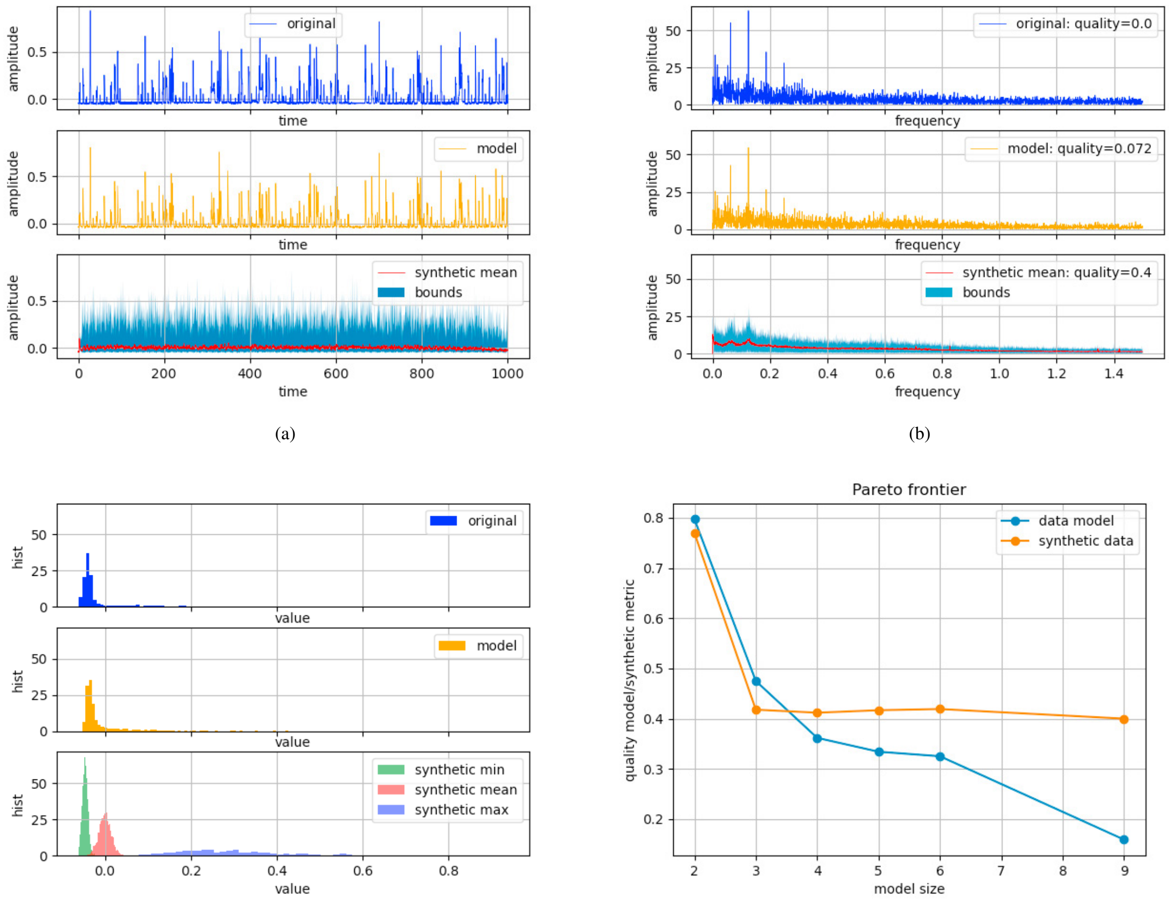
Fig. 2: Tropical temperature variability dataset results: (a) temperature variability dataset (blue), obtained by single-objective optimization model (model size 12, orange) and averaged generated synthetics (red) with minimum and maximum bounds for 50 generated samples; (b) corresponding spectra with the computed quality metric Eq. 8; (c) corresponding disributions for the source data; (d) multi-objective optimization Pareto frontier (metric Eq. 5, blue) and synthetic data (metric Eq. 8, orange)

Multi-objective optimization copes well with these difficulties. The Pareto frontier of models allows one to draw more profound conclusions and immediately choose the best model. The result obtained by the multi-objective algorithm is shown in Fig. 2d. From this graph, it is clear that the best model is the model with the size of 3 tokens, which approximate the basic structure of the series. Other tokens do not significantly contribute to either the approximation or the variability and quality of synthetic data due to their low amplitudes. Models including them have excessive complexity.

### 4.2. *Time-series without significant seasonality*

The previous experiment is repeated on the series that cannot be customarily approximated only with the help of trend and periodic tokens. The primary working tokens, in this case, are Complex impulses. The result of using the single-objective algorithm is shown in Fig. 3.

In contrast to the previous case, the spectrum has a more complex shape. Thus, it is impossible to say unequivocally what size of the model will be sufficient for high-quality approximation and generation of synthetics. For the resulting model with nine tokens, the very variable synthetic is obtained due to the generation algorithm based on complex



(a)

(b)

(c)

(d)

Fig. 3: Households electricity usage dataset results: (a) households electricity usage dataset (blue), obtained by single-objective optimization model (model size 9, orange) and averaged generated synthetics (red) with minimum and maximum bounds for 50 generated samples; (b) corresponding spectra with the computed quality metric Eq. 8; (c) corresponding disributions for the source data; (d) multi-objective optimization Pareto frontier (metric Eq. 5, blue) and synthetic data (metric Eq. 8, orange)

tokens. The shaded area of the synthetic bounds indicates possible permutations of the sequence of single pulses. It turns out that a weakly deterministic series generates highly variable synthetics, as it should be. In this case, it is not known with which meta-parameters a better model will be obtained, and multi-objective optimization will be even more helpful here. The result of its use is shown in Fig. 3d.

Complex impulses are good approximators for the data without a significant seasonal component. Therefore, the quality of the model approximation increases with the number of such tokens in the model. An increasing amount of the tokens leads to a more accurate noise component approximation. However, the quality of synthetics quickly ceases to grow because, with an abundance of complex impulses, the variability of synthetic time series increases. At the same time, they lose the structure of the original series, which is reflected in the proximity of their spectra. Thus, excessive variability is penalized. The interpretable optimization results confirm the correctness of the synthetic quality metric choice. Based on the Pareto frontier, the best generative model would be the model with three tokens.

## 5. Conclusion

In the paper, we develop the evolutionary multi-objective approach to generate synthetic time series. It involves building a model in the form of a closed-form algebraic expression and creating synthetic data based on the model. As the advantages, we can state:

- In contrast to single-objective optimization, multi-objective evolutionary optimization allows getting a Pareto frontier of models from which the expert can choose the optimal model for current tasks.
- The interpretability of the model allows to fully control the process of creating synthetics and surpasses existing approaches in the context of this task. Some of the possible variants of creation are presented in the form of synthesizing algorithms in this paper.
- The presented algorithm can work with the different types of time-series without significant tuning

However, the current realization algorithm requires significant computational time to obtain the model. Therefore, we will work on the code quality to speed up the optimization process in the future. Moreover, as future work, we propose advanced models that allow splitting the data to different scales.

## Code and data avaliability

Data and code to reproduce the experiments described in the paper are available at Nature Systems Simulation (NSS) lab repository https://github.com/ITMO-NSS-team/EPDE/tree/main/examples/multi_objective_algebraic_expression

## Acknowledgements

This research is financially supported by The Russian Science Foundation, Agreement #21-11-00296

## References

[1] Alzantot, M., Chakraborty, S., Srivastava, M., 2017. Sensegen: A deep learning architecture for synthetic sensor data generation, in: 2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), IEEE. pp. 188–193.

[2] Arnaldo, I., O'Reilly, U.M., Veeramachaneni, K., 2015. Building predictive models via feature synthesis, in: Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation, pp. 983–990.

[3] Contreras, J., Espinola, R., Nogales, F.J., Conejo, A.J., 2003. Arima models to predict next-day electricity prices. IEEE transactions on power systems 18, 1014–1020.

[4] Esteban, C., Hyland, S.L., Rätsch, G., 2017. Real-valued (medical) time series generation with recurrent conditional gans. arXiv preprint arXiv:1706.02633 .

[5] Keylock, C., 2012. A resampling method for generating synthetic hydrological time series with preservation of cross-correlative structure and higher-order properties. Water Resources Research 48.

[6] Li, K., Deb, K., Zhang, Q., Kwong, S., 2014. An evolutionary many-objective optimization algorithm based on dominance and decomposition. IEEE Transactions on Evolutionary Computation 19, 694–716.

[7] Lim, B., Zohren, S., 2020. Time series forecasting with deep learning: A survey. arXiv preprint arXiv:2004.13408 .

[8] Merezhnikov, M., Hvatov, A., 2020. Closed-form algebraic expressions discovery using combined evolutionary optimization and sparse regression approach. Procedia Computer Science 178, 424–433.

[9] Murari, A., Peluso, E., Lungaroni, M., Gelfusa, M., Gaudio, P., 2015. Application of symbolic regression to the derivation of scaling laws for tokamak energy confinement time in terms of dimensionless quantities. Nuclear Fusion 56, 026005.

[10] Taylor, S.J., Letham, B., 2018. Forecasting at scale. The American Statistician 72, 37–45.

[11] Yang, J., Zhang, S., Xiang, Y., Liu, J., Liu, J., Han, X., Teng, F., 2020. Lstm auto-encoder based representative scenario generation method for hybrid hydro-pv power system. IET Generation, Transmission & Distribution 14, 5935–5943.

[12] Young, P., 2003. Top-down and data-based mechanistic modelling of rainfall–flow dynamics at the catchment scale. Hydrological processes 17, 2195–2217.

[13] Young, P.C., 1986. Time-series methods and recursive estimation in hydrological systems analysis, in: River flow modelling and forecasting. Springer, pp. 129–180.