

# Quantum Operator Model for Data Analysis and Forecast

George Danko<sup>1,2</sup>

<sup>1</sup>Mackay School of Earth Science and Engineering, University of Nevada, Reno, USA

<sup>2</sup>Research Institute of Applied Earth Sciences, University of Miskolc, Miskolc, Hungary

Email: danko@unr.edu

**How to cite this paper:** Danko, G. (2021) Quantum Operator Model for Data Analysis and Forecast. *Applied Mathematics*, 12, 963-992.

<https://doi.org/10.4236/am.2021.1211064>

**Received:** September 21, 2021

**Accepted:** November 15, 2021

**Published:** November 18, 2021

Copyright © 2021 by author(s) and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

---

## Abstract

A new dynamic model identification method is developed for continuous-time series analysis and forward prediction applications. The quantum of data is defined over moving time intervals in sliding window coordinates for compressing the size of stored data while retaining the resolution of information. Quantum vectors are introduced as the basis of a linear space for defining a Dynamic Quantum Operator (DQO) model of the system defined by its data stream. The transport of the quantum of compressed data is modeled between the time interval bins during the movement of the sliding time window. The DQO model is identified from the samples of the real-time flow of data over the sliding time window. A least-square-fit identification method is used for evaluating the parameters of the quantum operator model, utilizing the repeated use of the sampled data through a number of time steps. The method is tested to analyze, and forward-predict air temperature variations accessed from weather data as well as methane concentration variations obtained from measurements of an operating mine. The results show efficient forward prediction capabilities, surpassing those using neural networks and other methods for the same task.

## Keywords

Time Series Analysis, Dynamic Operator, Quantum Vectors, Quantum Operator, Machine Learning, Forward Prediction, Real-Time Data Analysis

---

## 1. Introduction

Simulation, design, and process control tasks in engineering require the knowledge of the mathematical model of the controlled system. A dynamic model of a system may be created using analytical or numerical, computational simulation

tools. Complex problems involving coupled processes pose a challenge to set up an analytical or computational, dynamic model that is fast enough to evaluate, flexible enough to match experimental observations, and adjustable enough for corrective calibration. Analytical models may need precious skills to set up for sufficient details, while computational, system modeling tools are cumbersome to incorporate in real-time process control applications.

Artificial Intelligence (AI) and Machine Learning (ML) methods have arisen as a panacea for overcoming the model-building difficulties when the vast amount of monitored data is already available from the subject system. A systematic review of AI models for natural resource applications is given by Jung and Choi [1] albeit without any focus on applicability for real-time information processing. Neural Network (NN) models are regarded as the most universal model identification tools when large input and target data samples, as well as long training time, are available and acceptable, reviewed by Rojas [2]; Lin *et al.* [3]; Miao *et al.* [4]. However, for real-time data analysis, regression and autocorrelation methods may compare and compete favorably both in efficiency and evaluation time for dynamic, predictive model identification. For example, the training time for an LSTM type short-term forecasting NN model is reported by Dias [5] to take 30 minutes, while the same task is solved in one minute with the same accuracy using a first-order, time-series functional model.

The aims at the development of a new dynamic system model are: 1) data compression without information loss; 2) processing speed increase in model identification; and 3) accuracy improvement for short-term forecasting. Such demands (1)-(3) have arisen, e.g., for forward predicting and controlling atmospheric conditions in the hazardous workplace environment for workers' safety and health. The focus of the study, therefore, is to develop a fast, real-time evaluation of a method for the mass amount of data commonly monitored as environmental air parameters with the capabilities of forecasting.

The heat, mass, and momentum transport processes are dependent on the past and present input conditions involved in the outcome of process parameters of the atmospheric conditions such as air velocity, temperature, humidity, and contaminant gas species. Similarly, the expected, future, process parameters are governed by the past and present conditions and the general, self-similar system behavior, in addition to some recurring disturbances. Dynamic model identification is expected to recognize and account for these system characteristics for forward prediction applications. Once the systematic characteristics are matched, only the stochastic disturbances remain to be depressed using, for example, least-square fit matching during model training. The distraction caused by the "known unknowns" in the forecast of the process parameters will then be limited only to the extent of a random model fitting error.

Functional data analysis is a good starting point for dividing the input data into discrete time intervals within which the data in each time segment is characterized by some statistical parameters such as the median or mean values, e.g., in Horvath and Koloszka [6]. A classic time series analysis by Box and Jenkins

[7] applies Autoregressive Moving Average (ARMA) or Autoregressive Integrated Moving Average (ARIMA) models to best fit a time-series model to past values of the input time series. The goal of the presented work is the identification of a linear operator model, for which any unnecessary and nonlinear elements are avoided by design. Such nonlinear models are used e.g., by Milionis and Galanopoulos [8] in a univariate ARIMA model for analyzing economic time series in the presence of variance instability and outliers; or by Pam *et al.* [9], applying non-stationary time series analysis of energy intensity by an expanded ARIMA model with logarithmic terms; and Abebe [10] for annual rainfall analysis.

A similar approach is used in the presented work regarding the autoregressive concept but in a fundamentally new way in which any single time series of  $N$  members of data is broken into multivariate components in time compartments assigned to  $M$  number of designated time interval bins. A significant element is that the compartmentalized data to be processed are moved from time segment bin to bin step by step, moving with the progression of real-time. The dynamic model will then use the characteristic values of the groups of data as multivariate inputs kept in the time interval bins.

The quantum of data kept in bins serves as the fixed base of the  $M$ -dimensional operator (or functional) of the dynamic model. A similar approach is used in a previous work regarding operator representation of a system model, rendering an output function to an input function as a transformation, e.g., by matrix-vector multiplication, used by Danko [11]. The previous nomenclature is kept unchanged, referring to an operator as a “functional,” that is, a function-function as opposed to a function of scalar values.

The plan of the study is set up as follows. A data flow of  $X(t_1), X(t_2), \dots, X(t_N)$  is assumed from a single-channel sensor, acquired from the subject system at  $t_1, t_2, \dots, t_N$  time instants,  $X(t_N)$  being the most current. The past data are to be continuously stored in  $M$  number of bins, where  $M \ll N$  for data compression. Definitions are given for the time compartmentalization into bins; the data processing and distribution into bins; and transport of the quantum of data between the bins during step-by-step sliding from the most recent to the past time periods. Various data compression methods are shown for comparison of characteristics including the common, sliding time window averaging and a new property, named the “moving window quantum of data”. The moving window quantum value in each bin is defined from the contained  $X(t_i)$  data for constructing a set of base vectors of the dynamic operator of the system. For the model training of the matrix operator, a set of  $M$ -length quantum vectors is defined for setting up an over-determined set of equations for  $M < K$ . The  $M \times M$  matrix coefficients of the dynamic operator of the system are obtained by matching the model prediction to the data by the least-square (LSQ) error fit method. Application examples will complete the study to show the operator model’s performance to complement or surpass those of other ML techniques including NN.

## 2. Input Data Compression into Time Bin Compartments

**D1. Definition of time compartmentalization into bins.** Let  $t_1, t_2, \dots, t_N$  be the set of equidistant time divisions ( $[t_i, t_{i+1}] = \text{constant} = \Delta t$ ,  $t_i \in \mathbb{R}^1$ , e.g., minute, or day in seconds) for the acquisition of  $X(t_1), X(t_2), \dots, X(t_N)$  data samples ( $X(t_i) \in \mathbb{R}^1$ , e.g., temperature, or gas concentration) to be used simultaneously for operator model identification. A set of  $M$  time intervals with time divisions  $\tau_1, \tau_2, \dots, \tau_M$  where  $M \ll N$ , is defined for arranging the time divisions into bins over the same model input interval, that is,  $[0, t_N] = [0, \tau_M]$ . An arbitrary but strategical selection for the time bin intervals is defined to achieve monotonously and gradually widening division intervals from the most recent ( $\tau_M$ ) to the oldest ( $\tau_1$ ) time instant, that is,  $[0, \tau_1] \gg [\tau_{M-1}, \tau_M] = \Delta t_M$ , in such a way that the finest bin width equals the equidistant time divisions in  $t$ , that is,  $\Delta t_M = \Delta t$ . Consequently,  $X(t_M) = X(\tau_M)$  and  $X(t_{M-1}) = X(\tau_{M-1})$ . The width of each time-base bin is defined as  $\Delta \tau_k = \tau_k - \tau_{k-1}$ , for  $k = 1, \dots, M$  with  $\tau_0 = 0$  for the starting point of the first moving time window at  $k=1$ . Note that  $\Delta \tau_M = \Delta t_N = \Delta t$  by design.

### E1. Examples of bins selection.

**E1a.** Given is a time interval of  $N = 327$  days with 1-day increments as  $t_i = i$ , where  $i = 1, \dots, N$ . The number of bins is selected to be  $M = 50$ . The task is to find a smooth and monotonous function for the  $\tau_k$ ,  $k = 1, \dots, M$  division points for covering the entire 327 time period. A power series function is selected for  $\tau_k$  as follows:

$$\tau_k = a(1 - b^{-k}), \quad k = 1, \dots, M \quad (1)$$

where:

$$a = \frac{t_N - t_{N-1}}{b^{1-M} - b^{-M}}, \quad (2)$$

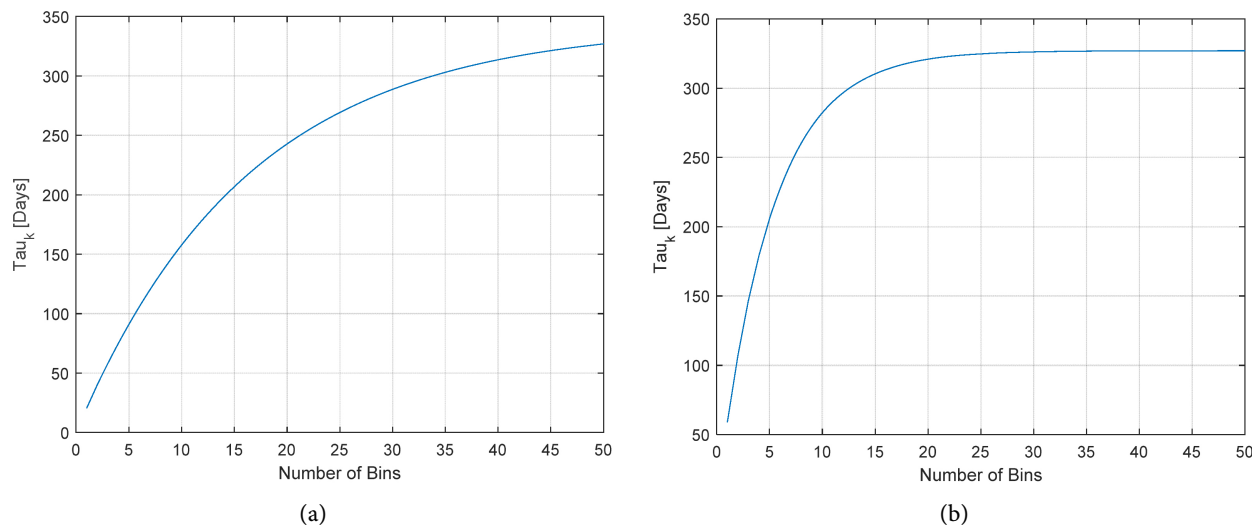
and:

$$b = \left( \frac{bt_N - t_{N-1}}{t_N - t_{N-1}} \right)^{\frac{1}{M}} \quad (3)$$

With  $t_i = i$  given, Equation (3) has to be solved first by iteration, that converges in 22 steps to  $1e-12$  absolute error, giving  $b = 1.0636$ . From Equation (2),  $a = 342.7324$ .

The  $\tau_k$  divisions from Equation (1) are plotted in **Figure 1(a)** against the number of bins. As shown, the division points of the bins are exponentially widening toward the oldest time instant from the latest, most recent  $\tau_M$  (or refining to the most recent from the oldest  $\tau_1$  until it equals the finest step of 1 day). The most recent four values for  $\tau_k$  are  $\tau_{47,48,49,50} = [323.8053, 324.9364, 326, 327]$ .

**E1b.** Given is a time interval of 327 days, each day to be further discretized to 5-minute intervals. This defines  $N = 327 \times 288 = 94176$  time intervals with 5-minute increments as  $t_i = i$ , where  $i = 1, \dots, N$ . The number of bins is selected



**Figure 1.** The  $\tau_k$  divisions from Equation (1) over the number of bins; (a) in E1a, the finest division is 1 day; (b) in E1b, the finest division is 0.0035 day (5 minutes).

to be  $M = 50$ . The task is to find the  $\tau_k$ ,  $k = 1, \dots, M$  division points for covering the entire 94,176 time period. From Equations (1)-(3),  $a = 327.0158$ ,  $b = 1.2199$ , and the  $\tau_k$  time division points are evaluated. The most recent four values for  $\tau_k$  are  $\tau_{47,48,49,50} = [323.8053, 324.9364, 326, 327]$ .

The  $\tau_k$  divisions in day units from Equation (1) are plotted in **Figure 1(b)** against the number of bins. The most recent four values for  $\tau_k$  are  $\tau_{47,48,49,50} = [326.9871, 326.9923, 326.9965, 327]$ . The last time period is  $\tau_{50} - \tau_{49} = 0.00347222222229\dots$ , equaling 5 minutes in day unit.

The focus is on the real-time evaluation of a continuous, discretized data stream. The time-base bins are designed to hold the newest sample unchanged, and the characteristics of past data compressed, representative to the acquisition time of the cluster relative to the last, current time instant. There are several, known ways to characterize past data using some methods of averaging. For example, the conventional, daily average of minute-acquired temperatures use the integral mean value of the measured data, the integral approximated by the Riemann sum of the definite integral for each day. Following this example, and assuming for simplicity a continuous, piecewise-linear function,  $Xp(t)$ , for representing the discretized data  $X(t_i)$ , that is,  $Xp(t_i) = X(t_i)$ , for  $i = 1, \dots, N$ , the average data,  $\overline{Xp}_k(t)$ , belonging to each time bin may be defined as:

$$\overline{Xp}_k(t) = \frac{1}{\Delta\tau_k} \int_{\tau_{k-1}}^{\tau_k} Xp(t) dt, \quad t \in [\tau_{k-1}, \tau_k] \quad (4)$$

There are difficulties in using Equation (4) for discretized data  $X(t_i)$  directly. The  $\tau_k$  bin division points do not coincide with the  $t_i$  time divisions except for bin  $k$  at  $k = M - 1$  and  $k = M$ , therefore,  $Xp(t)$  cannot simply be replaced by the  $X(t_i)$  values within the time intervals  $[\tau_{k-1}, \tau_k]$  to avoid rounding errors. In addition, linear interpolation function fitting for  $Xp(t)$  is necessary, albeit not practical, as the storage of all original data,  $X(t_i)$ , is needed for  $Xp(t)$

that alone contradicts data compression. Therefore,  $\overline{Xp}_k(t)$  is not practical as defined in Equation (4) but re-written in its moving boundaries form for accepting a constant  $\Delta t = \Delta \tau_M$  time step change to account for the moving time window. The transition from  $\Delta \tau_k$  bin at  $\overline{Xp}_k(t)$  average to  $\Delta \tau_{k+1}$  bin at  $\overline{Xp}_{k+1}(t)$  average adds an  $\int_{\tau_{k+1}}^{\tau_{k+1} + \Delta \tau_M} Xp(t) dt$  difference value, while leaves behind a  $-\int_{\tau_k}^{\tau_k + \Delta \tau_M} Xp(t) dt$  difference value of the integral  $\overline{Xp}_k(t) \Delta \tau_k$ . The sliding window expression for  $\overline{Xp}_k(t + \Delta t)$  is:

$$\overline{Xp}_k(t + \Delta t) = \overline{Xp}_k(t) + \int_{\tau_{k+1}}^{\tau_{k+1} + \Delta \tau_M} Xp(t) dt \frac{\Delta t}{\Delta \tau_k} - \int_{\tau_k}^{\tau_k + \Delta \tau_M} Xp(t) dt \frac{\Delta t}{\Delta \tau_k}, t \in [\tau_{k-1}, \tau_k] \tag{5}$$

For evaluating  $\overline{Xp}_k(t + \Delta t)$  at the next time step, the two integrals in Equation (5) still need additional data at the beginning and the end time bin  $k$  to store, but at least the many data already averaged inside bin  $k$  will need not be individually stored as their previous average value is reused in  $\overline{Xp}_k(t)$ . The shortcomings in using  $\overline{Xp}_k$  may be alleviated by modifications of its content, leading to a different property of the sliding-averaged data. The modifications to be made to the expression in Equation (5) to make it less cumbersome to use are first introduced as approximations to replace the  $Xp(t)$  kernel functions with their integral mean values for the respective time bins in the integrals:

$$\int_{\tau_k}^{\tau_k + \Delta \tau_M} Xp(t) dt = \int_{\tau_k}^{\tau_k + \Delta \tau_M} \overline{Xp}_k(t) dt + \varepsilon_1 = \overline{Xp}_k(t) \Delta t + \varepsilon_1 \tag{6a}$$

$$\int_{\tau_{k+1}}^{\tau_{k+1} + \Delta \tau_M} Xp(t) dt = \int_{\tau_{k+1}}^{\tau_{k+1} + \Delta \tau_M} \overline{Xp}_{k+1}(t) dt + \varepsilon_2 = \overline{Xp}_{k+1}(t) \Delta t + \varepsilon_2 \tag{6b}$$

Indeed, substituting Equations (6a) and (6b) into (5) gives an approximate expression for  $\overline{Xp}_k(t + \Delta t)$  that is easy to evaluate and effective in data compression, but includes the sum of two error terms,  $\varepsilon_1 + \varepsilon_2$ :

$$\overline{Xp}_k(t + \Delta t) = \overline{Xp}_k(t) - \overline{Xp}_k(t) \frac{\Delta t}{\Delta \tau_k} + \overline{Xp}_{k+1}(t) \frac{\Delta t}{\Delta \tau_k} + \varepsilon_1 + \varepsilon_2, t \in [\tau_{k-1}, \tau_k] \tag{7}$$

The need for a new, useful, average-type characteristics of the data stored in bin  $k$  is inspired by Equation (7) together with the goal of eliminating the  $\varepsilon_1 + \varepsilon_2$  error term. The new data property is called the quantum of data in a time interval bin in sliding window coordinates, leading to the definition of the quantum of data.

**D2. Definition of quantum of data in a time-base bin**

Definition of the quantum of data,  $Q_k(t)$ , in time-base bin  $k$  is given in a finite difference equation form as follows:

$$\frac{\Delta Q_k(t)}{\Delta \tau} = \frac{Q_{k+1}(t) - Q_k(t)}{\Delta \tau_k} \text{ where } \Delta Q_k(t) = Q_k(t + \Delta \tau) - Q_k(t) \tag{8}$$

The quantum definition in Equation (8) expresses that the rate of change in quantum  $Q_k$  at any time,  $t$ , over the finest time step,  $\Delta \tau = \Delta t$ , is proportional

to the rate of change of quantum differences between the upstream,  $Q_{k+1}(t)$ , and downstream,  $Q_k(t)$  quantum neighbors. It is straightforward to use Equation (8) step-by-step, starting from the  $Q_{k+1}(t) = Q_M(t)$  quantum that is known as it is always equated with the last, constant, sampled value of the data stream.

Applying the definition in Equation (8) for a discrete data series yields:

$$Q_k(i+1) = Q_k(i) \left( 1 - \frac{\Delta\tau}{\Delta\tau_k} \right) + Q_{k+1}(i) \frac{\Delta\tau}{\Delta\tau_k} \quad \text{for } k = 1, \dots, M-1 \quad (9)$$

The quantum property in Equation (9) is an improvement over the sliding window property in Equation (7) as the ambiguous error term,  $\varepsilon_1 + \varepsilon_2$ , is eliminated due to the modified definition. The sliding window *average* is not a convenient property to use in comparison to the sliding window *quantum* of data property. By definition and design,  $\overline{Xp}_k(i\Delta t) \neq Q_k(i)$ , but their value may be close to each other. The essential difference, however, is that  $Q_k(i)$  is efficiently calculated with superior data compression while serves well the purpose of a reliable data characteristics for system model application with large data.

Note that the definition in Equation (9) is recursive and the  $Q_k(i+1)$  quantum value at bin  $k$  at time  $(i+1)\Delta t$  is defined by the weighted quantum value of  $Q_k(j)$  at a previous time  $j\Delta t$ , and the quantum value of the upstream neighbor bin,  $Q_{k+1}(i)$ . The newest quantum value at  $k = M-1$  is  $Q_M(i)$ , that is the single origin of filling all bins downward with their quantum content according to Equation (9).  $Q_M(t)$  may be selected as the original data,  $X(t)$ , taken at  $t = i\Delta t$ . This way, the quantum of data will retain everywhere the physical unit of the original data.

A straightforward way to give closed formulas of  $Q_k(t)$ ,  $k = 1, \dots, M-1$  for evaluating the quantum of data directly in each bin from the original data stream may be obtained by repeatedly applying Equation (9) starting from the known, new value of  $Q_M(i\Delta t) = X(i\Delta t)$  toward  $Q_1(i\Delta t + \Delta t)$ . However, a simple, matrix-vector equation is more convenient for numerical evaluation as shown in the following example.

## E2. Example of bin-to-bin quantum of data transformation using matrix-vector calculation

Let the values of quantum  $Q_k(i+1)$  and  $Q_k(i)$  for  $k = 1, \dots, M$  be organized into column vectors  $\mathcal{Q}^{i+1} = [\mathcal{Q}_k^{i+1}]$  and  $\mathcal{Q}^i = [\mathcal{Q}_k^i]$ , respectively. Using Equation (8), new vector elements  $\mathcal{Q}_k^{i+1}$  for  $(i+1)\Delta t$  time for  $k = 1, \dots, M-1$  can be expressed with the previous vector elements  $\mathcal{Q}_k^i$  for  $i\Delta t$  time for  $k = 1, \dots, M$  in a matrix-vector equation:

$$\begin{bmatrix} \mathcal{Q}_1^{i+1} \\ \mathcal{Q}_2^{i+1} \\ \vdots \\ \mathcal{Q}_{M-1}^{i+1} \end{bmatrix} = A \begin{bmatrix} \mathcal{Q}_1^i \\ \mathcal{Q}_2^i \\ \vdots \\ \mathcal{Q}_M^i \end{bmatrix} \quad (10)$$

where  $A$  is a sparse  $(M-1) \times M$  matrix with zero elements everywhere ex-

cept for non-zero elements only in the main diagonal and in the first, upper off-diagonal:

$$\left. \begin{aligned} A(k, j) &= 0 \text{ for } k = 1, \dots, M - 1; j = 1, \dots, M, \text{ but } j \neq i \text{ and } j \neq i + 1 \\ A(k, k) &= 1 - \frac{\Delta\tau}{\Delta\tau_i} \text{ for } k = 1, \dots, M - 1 \\ A(k, k + 1) &= \frac{\Delta\tau}{\Delta\tau_i} \text{ for } k = 1, \dots, M - 2 \end{aligned} \right\} \quad (11)$$

The last element of vector  $Q^{i+1}$  for  $k = M$ , not included in Equation (10), is defined by the new data, that is,  $Q_M^{i+1} = X(i\Delta t)$ .

**E3. Example of quantum of data vectors for a harmonic signal**

A continuous, sinusoidal data stream of 327 days sampled at regular 5-minute time intervals is processed into 50-element quantum vectors. A synthetic data stream is selected in the example to model daily and yearly temperature variations superimposed according to

$$X(i) = \frac{1}{2} \left[ \sin\left(\frac{327}{365} 2\pi i/N\right) + \sin(2\pi 327i/N) \right],$$

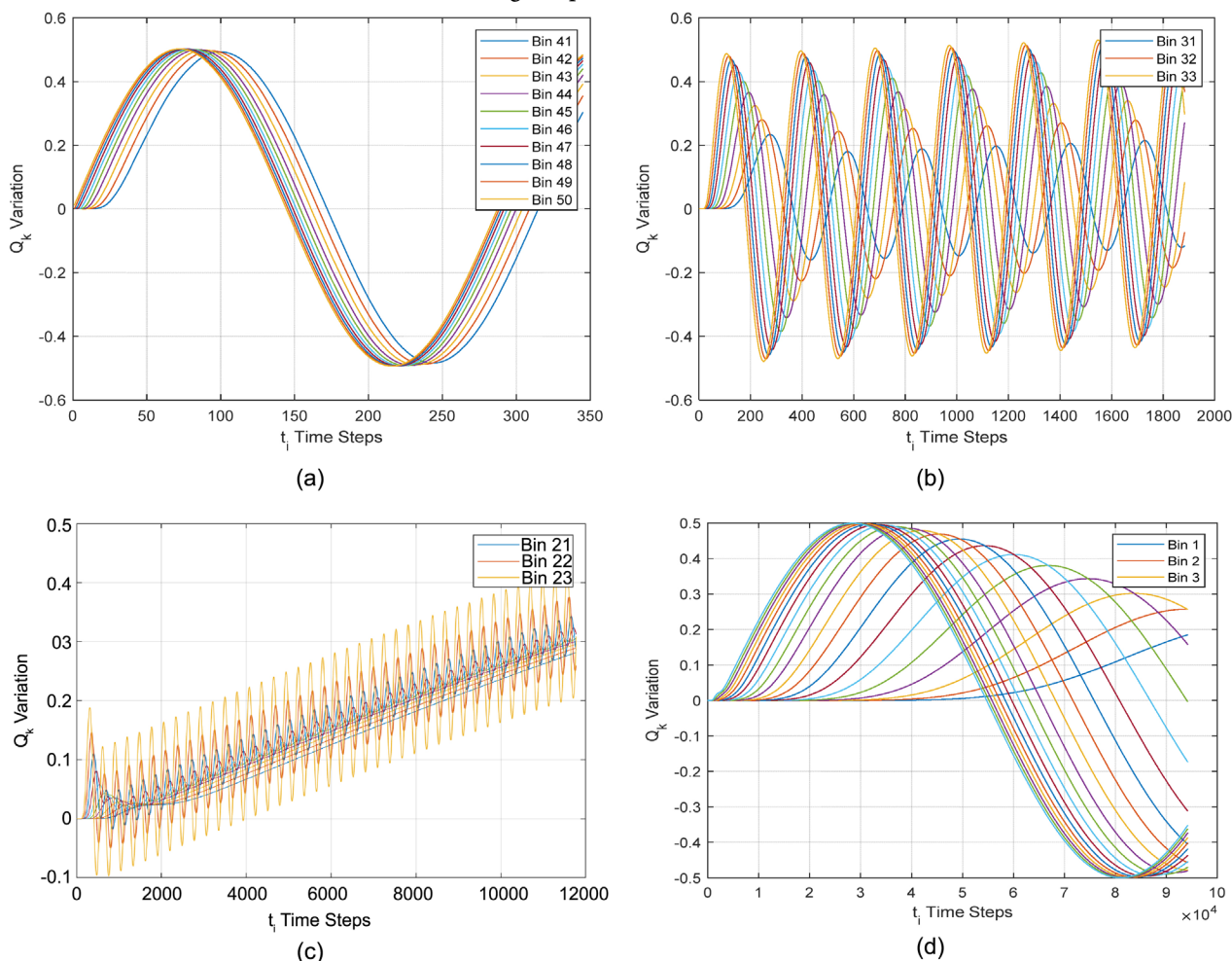
where the real-time vector is  $i = [1, \dots, N]$ , the series of time divisions. The time compartmentalization in E2 into 50 bins is used for the  $X(i) \rightarrow Q^i = [Q_k^i]$  transformation according to Equation (11). The 50 components of the  $Q^i$  vectors are shown in **Figures 2(a)-(d)** with an arbitrary bin and time interval selection for best visualization. The selected time steps and  $Q_k^i$  elements are shown in: **Figure 2(a)** for  $k \in [41, 50]$  (with all 10 bins marked); **Figure 2(b)** for  $k \in [31, 40]$  (with only 3 bins marked); **Figure 2(c)** for  $k \in [21, 30]$  (with only 3 bins marked); and **Figure 2(d)** for  $k \in [1, 20]$  (with only 3 bins marked). Note that the  $Q_{50}^i$  component in Bin 1 equals  $X(i)$ , albeit shown only for little over one day. As shown in **Figure 2(c)**, the daily, periodic temperature variation gradually disappears almost entirely between Bins 21 through 30 ( $k \in [21, 30]$ ). This implies that the most relevant elements of the  $Q_k^i$  vectors are those for  $k > 21$  for modeling signal variations on the fine, 5-minute scale. However, if the seasonal variations are of interest, all bins from  $k = 1$  must be used; furthermore, longer observed time period than 1 year is needed as  $Q_1^i$  in the first bin (with about 59 days width) has not yet established periodicity within one year.

**E4. Example of quantum of data vectors for measured data**

A true, outside temperature data stream of 327 days sampled at regular 5-minute time intervals is accessed from a commercial weather data vendor for Northern Nevada, USA. The data is processed into 50-element quantum vectors using the same process described in E3. The time compartmentalization in E2 into 50 bins is used for the  $X(i) \rightarrow Q^i = [Q_k^i]$  transformation according to Equation (11). The 50 components of the  $Q^i$  vectors are shown in **Figures 3(a)-(d)** with an arbitrary bin and time interval selection for best visualization as before. As shown in **Figure 3(c)**, the daily, periodic temperature variation gradually disappears almost entirely between Bins 21 through 30 ( $k \in [21, 30]$ ), implying that the most relevant elements of the  $Q_k^i$  vectors are those for  $k > 21$



for modeling temperature variations on the fine, 5-minute scale.



**Figure 2.** (a)-(d) Variation of the  $Q^i$  vectors with time for synthetic input data series; (a):  $k \in [41, 50]$  (with all 10 bins marked); (b):  $k \in [31, 40]$  (with only 3 bins marked); (c):  $k \in [21, 30]$  (with only 3 bins marked); and (d):  $k \in [1, 20]$  (with only 3 bins marked).

### 3. DQO Model Building of a System for Time Series Analysis and Forecast

It is straightforward to expand the concept of the autoregressive (AR) model into a dynamic operator. The AR model of order  $p$  is defined following Shumway and Stoffer [12], and Kun [13]:

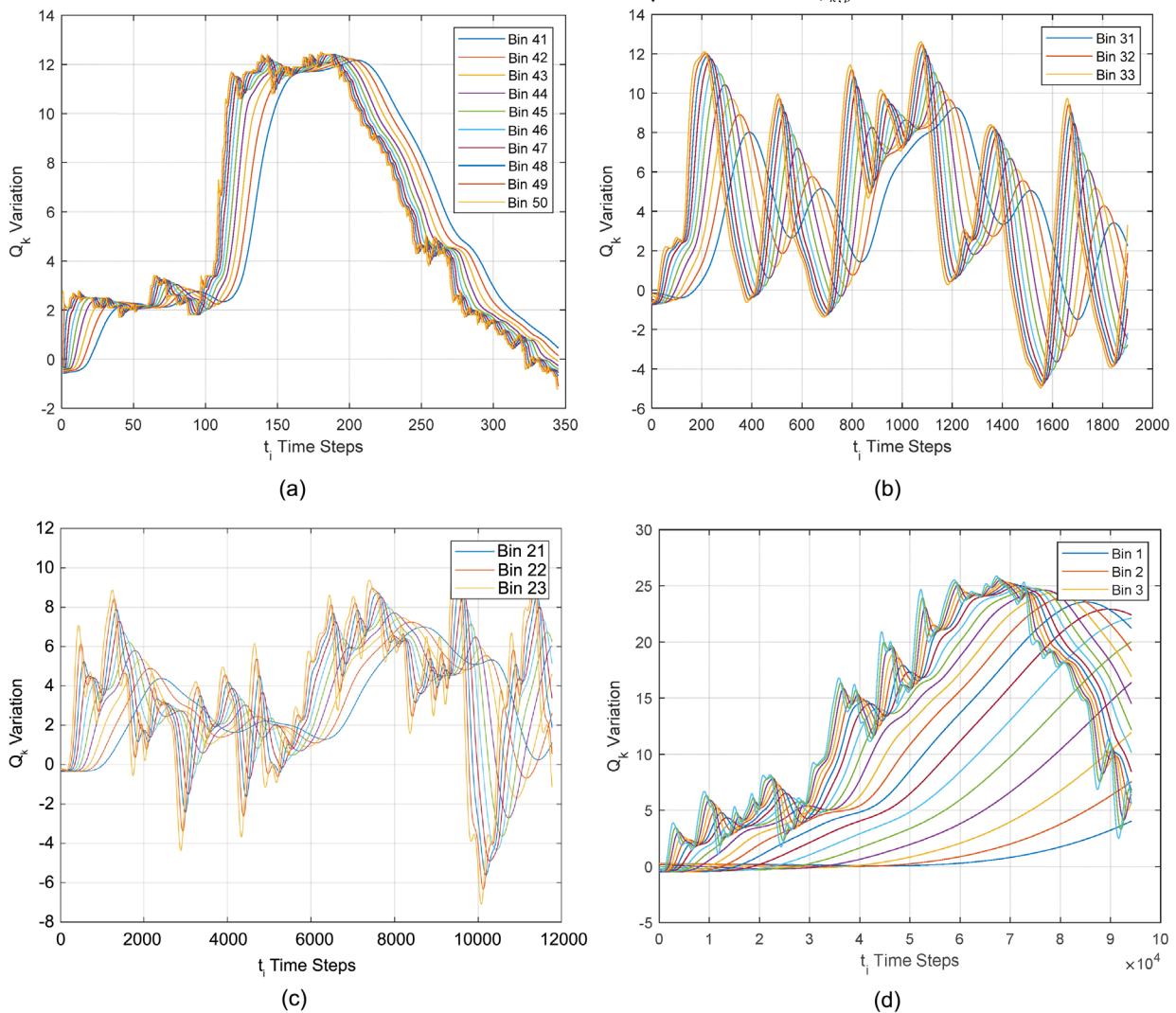
$$X(i) = c + \sum_{j=1}^p \varphi_j X(i-j) + \varepsilon(i) \quad (12)$$

where  $c$  is a constant,  $\varphi_j$  are constant coefficients, and  $\varepsilon(i)$  is noise. Applying the AR concept to the quantum of data with low-pass-filtered components instead of the original time series and absorbing  $c$  into the  $\varphi_j$  coefficients leads the definition of the dynamic operator.

#### D3. Definition of the z-step dynamic operator

The z-step dynamic operator of the system,  $\phi^{i,z}$ , is defined by its matrix. The

matrix of DQO is defined by the set of its  $\phi_{k,p}^{i,z}$  coefficients, which satisfies the



**Figure 3.** (a)-(d) Variation of the  $Q^i$  vectors with time for input data series from measurement; (a):  $k \in [41, 50]$  (with all 10 bins marked); (b):  $k \in [31, 40]$  (with only 3 bins marked); (c):  $k \in [21, 30]$  (with only 3 bins marked); and (d):  $k \in [1, 20]$  (with only 3 bins marked).

simultaneous, AR model fit for all elements of the  $Q_k^i$  modeled quantum vector,  $Q_{k,m}^i$ , to the measured origin,  $Q_k^i$  with a minimized  $\varepsilon(i)$  fitting error for a set of quantum vector samples  $i \in S$  for all  $k \in M$  elements:

$$Q_k^i = \sum_{p=1}^M \phi_{k,p}^{i,z} Q_p^{i-z} + \varepsilon(i) \tag{13}$$

where  $\varepsilon(i) = \min \left[ \sqrt{\sum_S \left[ \sum_k (Q_k^i - Q_m^i)^2 \right]} \right]$ ,  $i \in [1, N]$ ,  $k \in M$ , and  $S \subset [1, N]$ .

The  $Q_k^i$  modeled quantum vector component on the left side and the  $Q_p^{i-z}$  quantum vector component given at a shifted time instant by  $z$  number of time

steps on the right side are the inputs of the model fitting procedure, derived from measured data. The  $[\varphi_{k,p}^{i,z}]$  coefficients on the right side of Equation (13) are to be evaluated by best fitting the model prediction,  $\mathcal{Q}_{k,m}^i$  to  $\mathcal{Q}_k^i$  input with minimum error for all  $k$  components.

Time step shift  $z$  is a parameter of choice to forward predict future outcome from previous measured values of the time series. Equation (13) must be applied for all  $k$  components simultaneously. Using a matrix notation for the dynamic operator of the system at time index  $i$  as  $\phi^{i,z} = [\varphi_{k,p}^{i,z}]$ , Equation (13) for all  $k$  components reads:

$$\mathcal{Q}^i = \phi^{i,z} \mathcal{Q}^{i-z} + \varepsilon(i), \quad (14)$$

where  $\varepsilon(i) = \min \left[ \sqrt{\sum_S \left[ \sum_k (\mathcal{Q}^i - \mathcal{Q}_m^i)^2 \right]} \right]$ ,  $i \in [1, N]$ ,  $k \in M$ , and  $S \subset [1, N]$ .

The  $[\varphi_{k,p}^{i,z}]$  coefficients of the  $\phi^{i,z}$  operator on the right side of Equation (14) must be evaluated from the measured data and subsequently process  $\mathcal{Q}^i$  and  $\mathcal{Q}^{i-z}$  quantum vectors using an optimization procedure for minimizing the error of fit,  $\varepsilon(i)$ .

The  $\phi^{i,z}$  operator is assigned to time index  $i$ , where  $t(i)$  is the current (most recent) time step. Each  $\phi^{i,z}$  operator is determined over a subset of sampled time steps,  $S$ , as well as over  $M$  quantum vector components to incorporate past history data. Each  $\phi^{i,z}$  operator characterizes the changing system with respect to time variation, focusing on to  $z$ -step forward prediction. While operator  $\phi^{i,z}$  has constant matrix coefficients, it may be considered as the sampled element of a dynamic, time variable operator,  $\phi^z(t)$ . Each  $\phi^{i,z}$  operator has an inherent matching error originating from the stochasticity of the data, processed into quantum vectors  $\mathcal{Q}_m^i$ , obtained from the unknown system, and the mismatch between the temporal characteristics of the system and the AR operator model that enforces an autoregressive behavior.

#### D4. Definition of forward prediction from the dynamic operator of the system.

Equation (14) may be directly used for forward-predicting an expected, modeled quantum vector,  $\mathcal{Q}_m^i$ , at time  $t(i)$  from a previous quantum vector,  $\mathcal{Q}_m^{i-z}$ , processed from measured data at  $t(i-z)$ . Likewise, assuming the continuity of the  $\phi^{i,z}$  operator, forecast estimate may be written, jumping  $z$  steps from the most recent time  $t(i)$ , as:

$$\mathcal{Q}_m^{i+z} = \phi^{i,z} \mathcal{Q}_m^i \quad (15)$$

Alternatively, choosing  $z_{\text{train}} = 1$  in identifying operator  $\phi^{i,1}$  during model training, a  $z$ -step forward prediction estimate may be written as follows, repeatedly using  $z$ -times Equation (15), each step resulting in increasing the power index of  $\phi^{i,1}$  by one until the power of the required forward steps,  $z_{\text{predict}} = z$  are reached:

$$\mathcal{Q}_m^{i+z} = (\phi^{i,1})^z \mathcal{Q}^i \quad (16)$$

#### D5. Definition of a training data set for the solution of the dynamic operator of the system.

A training data set  $i \in S$  must be selected from the set of the  $\mathcal{Q}_m^i$  quantum vectors for identifying the unknown  $\phi^{i,z} = [\phi_{k,p}^{i,z}]$  coefficients in Equation (14). Set  $S$  is defined by the requirement for a unique solution for the elements of matrix  $\phi^i$ .

From elementary algebra, a minimum of  $M$  equations are needed for the solution of  $M$  unknown coefficients in an  $M$ -variable equation. For example, assuming a zero error term, hypothetically, for  $M = 50$  and  $z = 1$ ,  $S = [1, 51]$  quantum data set were sufficient to fill the left and right sides of Equation (14) and set 50 equations for the evaluation of the coefficients:

$$[\mathcal{Q}^{51} \mathcal{Q}^{50} \dots \mathcal{Q}^2] = \phi^1 [\mathcal{Q}^{50} \mathcal{Q}^{49} \dots \mathcal{Q}^1] \quad (17)$$

The solution, provided that the inverse matrix  $[\mathcal{Q}^{50} \mathcal{Q}^{49} \dots \mathcal{Q}^1]^{-1}$  exists, is:

$$\phi^1 = [\mathcal{Q}^{51} \mathcal{Q}^{50} \dots \mathcal{Q}^2] [\mathcal{Q}^{50} \mathcal{Q}^{49} \dots \mathcal{Q}^1]^{-1} \quad (18)$$

In reality, for the effective minimization of the fitting error term, a much larger input quantum set  $S$  is required. A least-square fit minimization scheme is devised by selecting a subset of time series input data,  $j \in S$ ,  $S \subset [1, N]$  as follows:

$$[\mathcal{Q}^j] = \phi^z [\mathcal{Q}^{j-z}], \quad j \in S \subset [1, N] \quad (19)$$

where  $[\mathcal{Q}^j]$  and  $[\mathcal{Q}^{j-z}]$  are  $M \times j$  matrices,  $j \gg M$ .

Multiplying Equation (19) on both sides from the right by the  $[\mathcal{Q}^{j-z}]^T$  transpose matrix; and again, multiplying the result from the right by the inverse of the square matrix  $\{[\mathcal{Q}^{j-z}] [\mathcal{Q}^j]^T\}^{-1}$  gives the LSQ solution for the over-determined set of equation, provided that the inverse exists:

$$\phi^z = [\mathcal{Q}^j] [\mathcal{Q}^{j-z}]^T \{[\mathcal{Q}^{j-z}] [\mathcal{Q}^{j-z}]^T\}^{-1}, \quad j \in S, \quad S \subset [1, N] \quad (20)$$

The  $\phi^z$  is a matrix representation of the linear operator of the system applicable for dynamic, time-series analysis and prediction.

The solvability of Equation (20) defines the necessary training data set for the determination of the  $\phi^z$  DQO model. The solvability depends on the existence of the  $\{[\mathcal{Q}^{j-z}] [\mathcal{Q}^{j-z}]^T\}^{-1}$  inverse matrix.

#### E5. Illustrative example of a DQO model fit and forward prediction for weather

A true, outside temperature data stream of 327 days sampled at regular 5-minute time intervals is used in its quantum-processed form discussed in E4 for a model fitting and prediction exercise. At each of the  $i = 1$  to  $327 \times 288$  time steps, a separate DQO model is built using four days with sliding window width,  $w = 8 \times 288 = 2304$  as set  $S$ . The goals of the exercise are to check the quality of: 1) the DQO model fit for each time step, measured by the normalized

absolute error between input data and model prediction at each time step; and 2) the DQO forward prediction steps of  $z = 12$  steps ahead at each time step, measured by the normalized absolute error between the known (but yet unused) input data at  $i + z$  and the model forward prediction at  $i + z$  time step. The sliding time window moves from  $i = 1$ , starting from an initial assumption of all zero history quantum values. The DQO model is trained to match the last 20 quantum components only (for  $k \in [31, 50]$ ) as just a short memory of the system is needed to learn for a  $z = 12$ -step forward prediction.

After the 400 coefficients of the  $\phi^{i,z}$  matrix of the DQO model of Equation (14) are determined with the LSQ solution of Equation (20) at each  $i$  time step over the  $w = 2304$ -step training window, the model prediction,  $Q_m^{i+z}$ , is calculated for quality check from the quantum-processed input data  $Q^i$  taken at real-time instants as:

$$Q_m^{i+z} = \phi^{i,z} Q^i \quad (21)$$

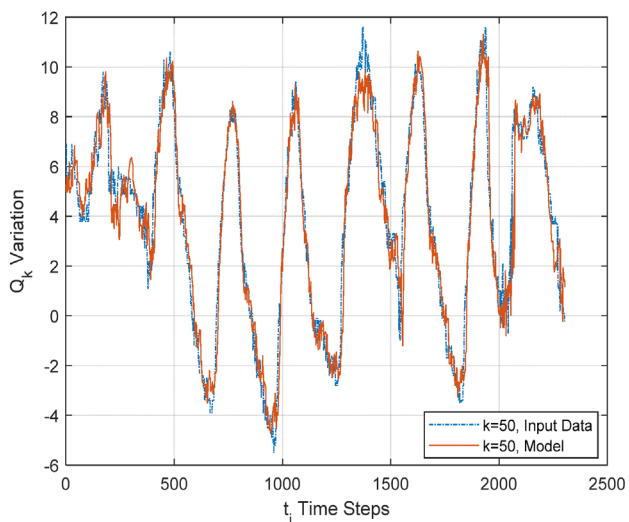
The variation of the  $Q_m^i$  and  $Q^i$  quantum vector components for the  $k \in [31, 50]$  components for the last moving window segment for  $i \in S$  are shown in **Figures 4(a)-(h)**, ( $i$  being used instead of  $j$  in the notation in the figure). The components of the  $Q_m^i$  and  $Q^i$  vectors with time are shown in (a)-(g) for  $k \in [44, 50]$  (with each individual pair and  $k$  marked); and in (h) for  $k \in [31, 43]$  (with only each  $k$  marked as no difference between  $Q_m^i$  and  $Q^i$  can be seen). Note that **Figure 4(a)** shows the DQO model match to the 5-minute data as the quantum vector for  $k = 50$  equals the un-processed input data. As shown in **Figures 4(a)-(h)**, the match between the DQO model's output results,  $Q_m^i$  and the input data,  $Q^i$ , is gradually improving toward slower frequency components at decreasing  $k$  values.

The forward-predicting capability of the DQO model is tested by evaluating forecasted outputs,  $Q_m^{i+z}$  from previous known values,  $Q^i$ . Using Equation (21), the model's output,  $Q_m^{i+z}$ , is calculated at each future time by  $z = 12$  time steps outside the training time window, while using a 12-step-old DQO. The forecasted results,  $Q_m^{i+z}$  are compared with the known future values,  $Q^{i+z}$ , not used in the DQO model training.

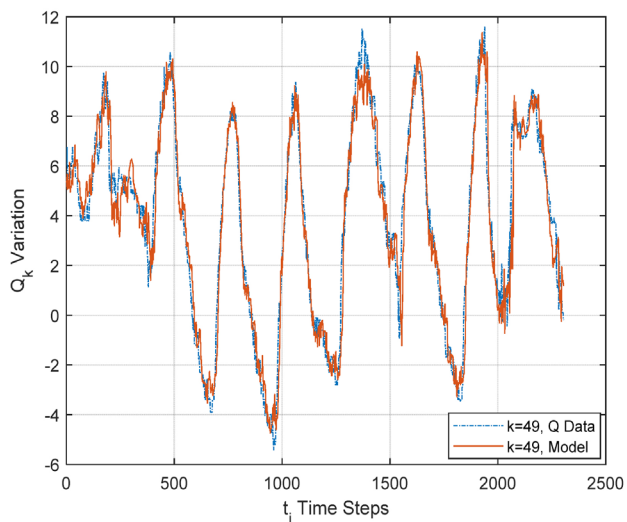
**Figure 5(a)** and **Figure 5(b)** show the variation of selected  $Q_m^{i+z}$  and  $Q^{i+z}$  vectors over the entire last training window forward predicted by  $z$  time steps, compared with input data series from measurement. The components of the  $Q^{i+z}$  and  $Q_m^{i+z}$  vectors with time are shown in (a) for  $k = 50$  (with marked pairs of  $Q^{i+z}$  and  $Q_m^{i+z}$ ); and in (b) for  $k \in [31, 44]$  (with only each  $k$  marked as no difference between model and data can be seen). As shown in **Figure 5(a)** and **Figure 5(b)**, the match between the DQO model's output results and the input data is about as good as the match for model training, indicating that input data have a learnable trend that holds well for about an hour ahead.

The absolute error of the model fit for each time step, normalized by dividing it with the average of the absolute values over each sliding window of  $w = 2304$  is calculated as  $E(i)$  for  $i = 1$  to  $327 \times 288$  time step (327 days):

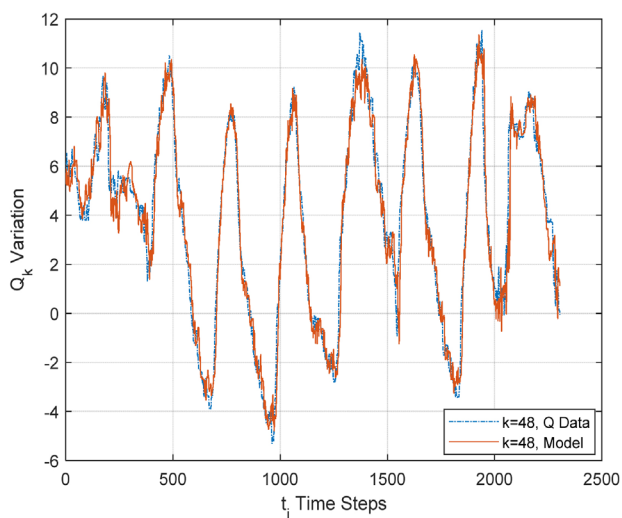
$$E(i) = \frac{w(\mathcal{Q}_m^i - \mathcal{Q}^i)}{\sum_{j=1}^{j=w} |\mathcal{Q}^{i-j+1}|} \times 100 [\%] \quad (22)$$



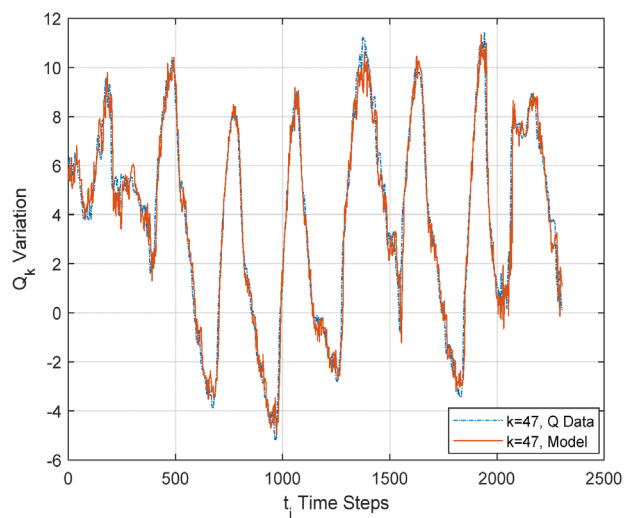
(a)



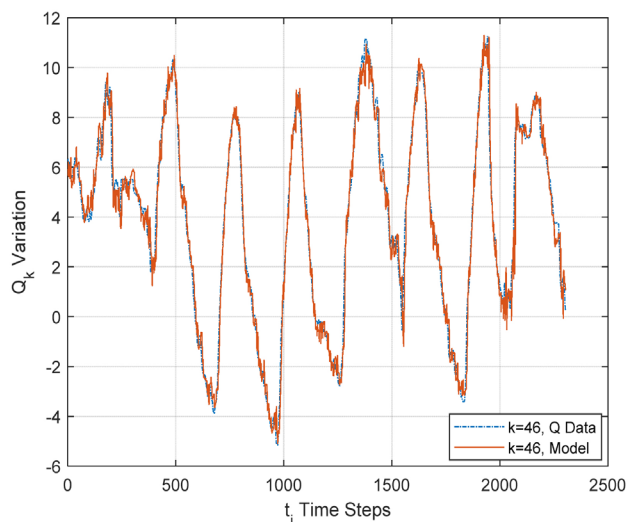
(b)



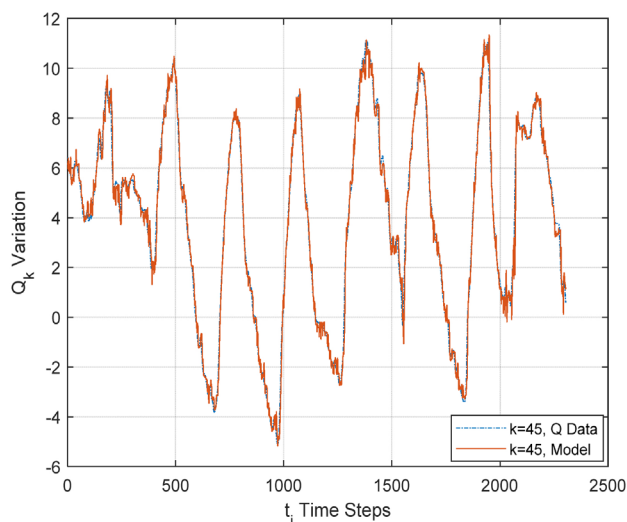
(c)



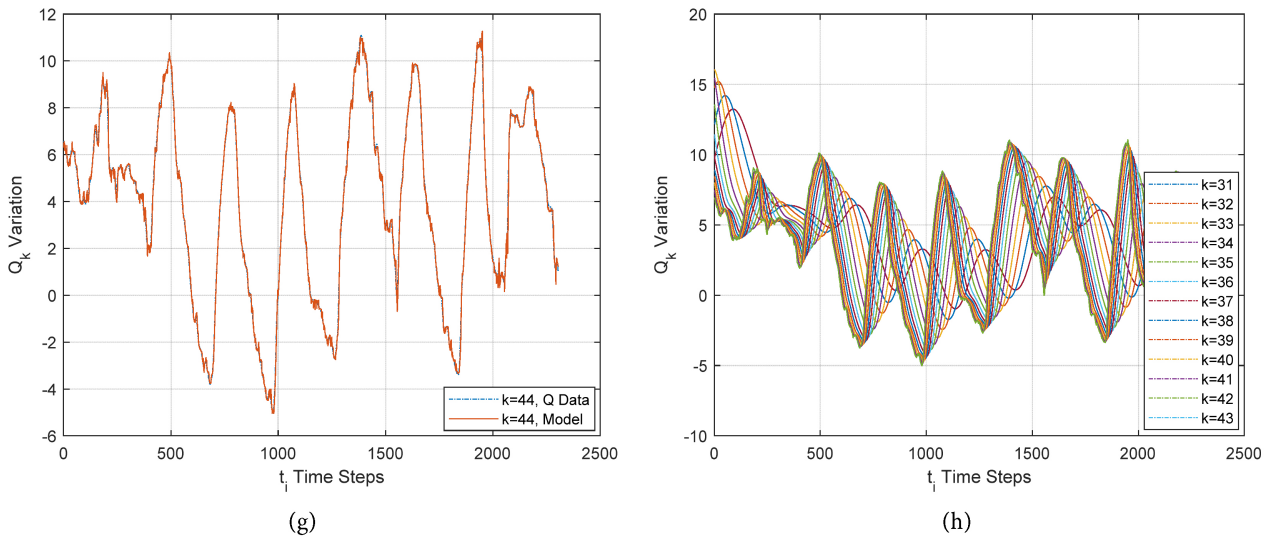
(d)



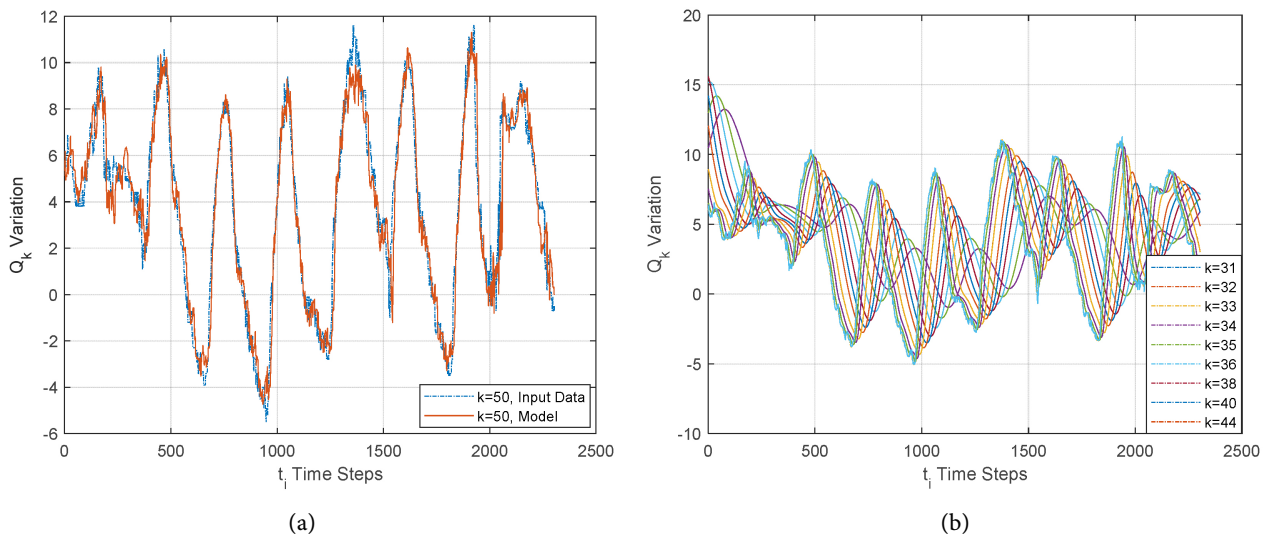
(e)



(f)



**Figure 4.** (a)-(h) Variation of the  $Q_M^i$  and  $Q_D^i$  vectors with time for input data series from measurement; (a)-(g):  $k \in [44, 50]$  (with each individual pair and  $k$  marked); (h):  $k \in [31, 43]$  (with only each  $k$  marked as no difference I between  $Q_D^i$  and  $Q_M^i$  can be seen).

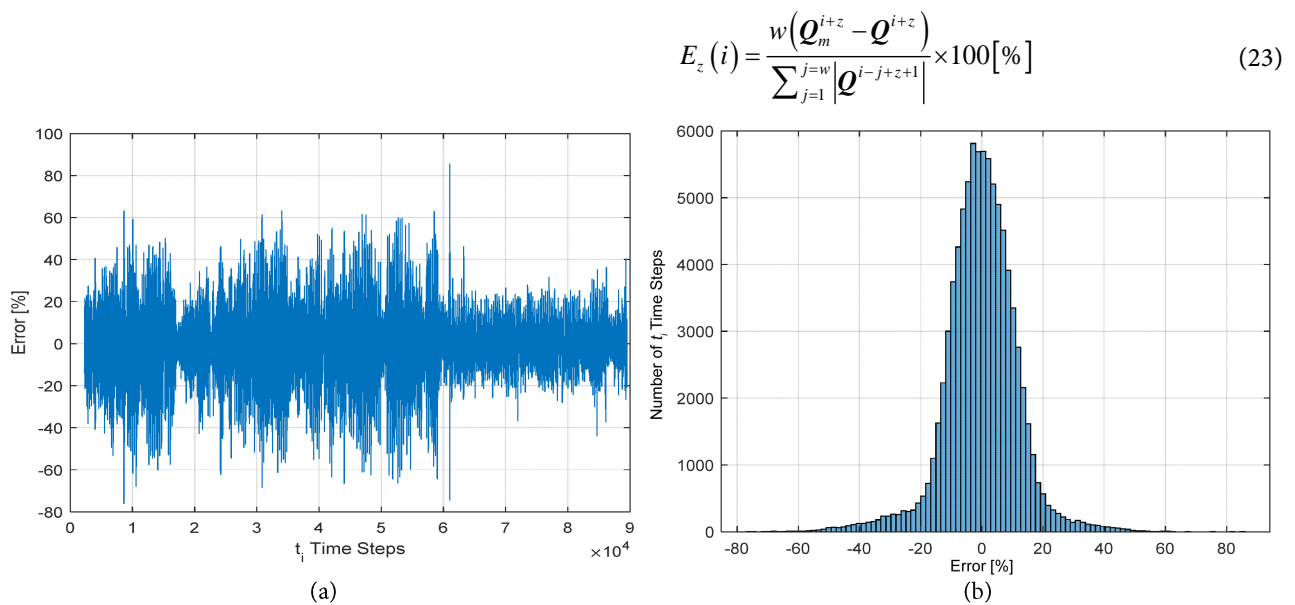


**Figure 5.** (a), (b) Variation of selected  $Q_M^{i+z}$  and  $Q_D^{i+z}$  vectors forward predicted by  $z = 12$  time steps, compared with input data series from measurement; (a):  $k = 50$  (with marked pairs of  $Q_D^{i+z}$  and  $Q_M^{i+z}$ ); (b):  $k \in [31, 44]$  (with only each  $k$  marked as no difference between model and data can be seen).

The variation of  $E(i)$  over the  $327 \times 288$  time steps is shown in **Figure 6(a)**. The histogram of the variation is depicted in **Figure 6(b)**.

The normalized absolute error of the model fit at forward predicted instances by  $z$  time steps for each time step over each sliding window  $w = 2304$  is calculated as  $E_z(i)$ :





**Figure 6.** (a), (b) Variations of normalized absolute DQO model error; (a) Model error over the training time window,  $E(i)$ ; and (b) Histogram of the model error.

The graph of  $E_z(i)$  and its histogram are shown in **Figure 7(a)** and **Figure 7(b)**, respectively.

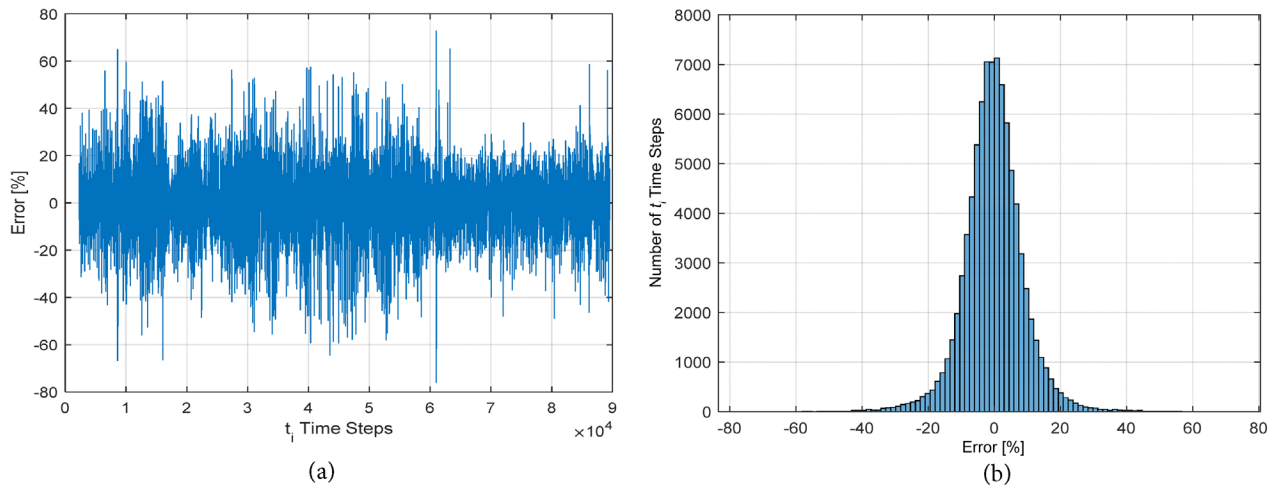
A comparison between **Figure 6** and **Figure 7** indicates a steady or overall better error performance in forward prediction application relative to that in model identification, an observation that should be considered coincidental, due to generally improving regularity in the input data stream with time in the example. Nevertheless, a steady DQO model performance up to 12 forward-step forecast in the example makes the method appealing, especially in comparison to published results for LSTM NN models with poorer forward prediction performance [5]. The typical running time for model DQO identification and forward prediction at each time step in E5 takes 18 milliseconds using a laptop computer.

#### 4. DQO Model Application for Safety and Health Analysis and Forecast

The DQO model is developed for analyzing and controlling atmospheric conditions for safety and health in working and living. As demonstrated in E5, a DQO model can be identified and used for forecasting with minimum cost and efforts, adding values for the raw data. The hypothesis is that precious, and quite significant time may be saved for preventive interventions to alleviate impending hazard conditions at any monitored, living or working place. The hypothesis is tested in a mine safety and health application example.

Atmospheric conditions are obtained from in situ, monitored data from an operating mine for 327 days under normal operating conditions. The monitored parameters are air flow rate in the face drift ( $Qa$ ), incoming Methane ( $CH_4$ ) gas concentration at the main gate ( $c_{MG}$ ), and exiting Methane concentration at the

tail gate ( $c_{TG}$ ). A synthetic data modification is introduced in Day 322 by an added Methane source ( $qms$ ) surge that increases the  $CH_4$  concentration above the allowable threshold value of 2%. The goal is to forecast the effect of the  $qms$



**Figure 7.** (a), (b) Variations of normalized absolute DQO model error; (a) Model error at 12 time steps (60 minutes) forward,  $E_{fps}(i)$ ; and (b) Histogram of the model error.

gas inburst as well as the resulting  $CH_4$  concentration by the DQO model for preventive intervention before the condition for a fatal explosion may happen.

#### E6. Illustrative example of a DQO model fit and forecast using large forward steps

The monitored parameters of air flow rate,  $Qa$ , incoming Methane gas concentration at the main gate,  $c_{MG}$ , and exiting concentration at the tail gate,  $c_{TG}$ , are inter-related. A transport model is used first for back calculating the root-cause gas source term,  $qm$ , from the observed incoming and exiting gas concentrations. A simplified Methane mass balance transport equation is used for the working drift:

$$c_{TG} = c_{MG} + 100qm/qa \quad [\%] \quad (24)$$

From the monitored data of the  $c_{MG}$ ,  $c_{TG}$ , and  $qa$  variables, the  $qmp = 100qm$  term is processed for DQO model building and forward prediction. The sampled values of  $qmp$  are first processed into quantum vectors. The DQO model is built for each 5-minute time step of  $288 \times 322$  time intervals. Note that a 2-day interval is reserved beyond the model-building time period of 360 missing the last two days for model. At each of the  $i = [1, 327 \times 288]$  time steps, a separate DQO model is built using three days for sliding window width,  $w = 3 \times 288 = 864$  as set  $S$ . The goals of the exercise are to check: 1) the quality of DQO model fit for each time step, measured by the normalized absolute error between input data and model prediction at each time step; 2) the quality of DQO forward prediction at steps of  $z = 36$  ahead, measured by the normalized absolute error between the known (but yet unused) input data at  $i + z$  and the model forward prediction at  $i + z$  time step; and 3) the hypothesis that  $CH_4$  threshold crossing

can be detected many time steps ahead of the real-time occurrence from previous measured data. The sliding time window moves from  $i = 1$ , starting from an initial assumption of all zero history quantum values. The DQO model is trained to match the last 20 quantum components only (for  $k \in [31, 50]$ ) as just a short memory of the system is needed to learn for a  $z = 36$ -step forward prediction.

The DQO model of Equation (14) are determined with the LSQ solution of Equation (20) at each  $i$  time step over the  $w = 864$ -step training window. The model prediction,  $Q_m^i$ , is calculated for quality check from the quantum-processed input data  $Q^{i-z}$  according to Equation (21), using  $z = 36$ . The variation of the  $Q_m^i$  and  $Q^i$  quantum vector components for the  $k \in [31, 50]$  components for the last moving window segment for  $i \in S$  are shown in **Figures 8(a)-(h)**, ( $i$  being used instead of  $j$  in the notation in the figure). The components of the  $Q_m^i$  and  $Q^i$  vectors with time are shown in (a)-(g) for  $k \in [44, 50]$  (with each individual pair and  $k$  marked); and in (h) for  $k \in [31, 43]$ . As shown in **Figures 8(a)-(h)**, the match between the DQO model's output results,  $Q_m^i$  and the input data,  $Q^i$ , is gradually improving toward slower frequency components at decreasing  $k$  values.

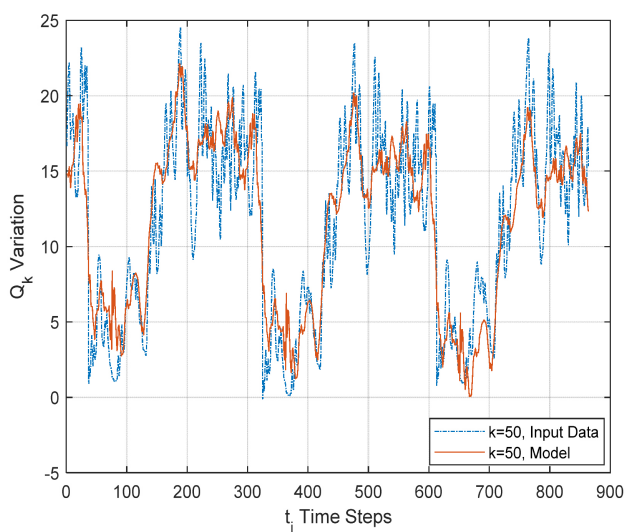
The forward-predicting capability of the DQO model is tested by evaluating forecasted outputs,  $Q_m^{i+z}$  from previous known values,  $Q^i$ . Using Equation (15), the model's output,  $Q_m^{i+z}$ , is calculated at each of future time by  $z = 36$  time steps outside the training time window, while using a 36-step-old DQO. The forecasted results,  $Q_m^{i+z}$  are compared with the known future values,  $Q^{i+z}$ , not used in the DQO model training. **Figure 9(a)** and **Figure 9(b)** show the variation of selected  $Q_m^{i+z}$  and  $Q^{i+z}$  vectors over the entire last training window, forward predicted by  $z$  time steps, compared with input data series from measurement. The components of the  $Q^{i+z}$  and  $Q_m^{i+z}$  vectors with time are shown in **Figure 9(a)** for  $k = 50$  (with marked pairs of  $Q^{i+z}$  and  $Q_m^{i+z}$ ); and in **Figure 9(a)** for  $k \in [31, 44]$ . As shown in **Figure 9(a)** and **Figure 9(b)**, the match between the DQO model's output results and the input data is close to the match for model training.

The absolute error of the model fit,  $E(i)$ , for each of the  $327 \times 288$  time step, normalized according to Equation (22), is shown in **Figure 10(a)**. The histogram of the variation is depicted in **Figure 10(b)**. The absolute error of the model fit at forward-predicted instances by  $z = 36$  time steps,  $E_z(i)$ , normalized according to Equation (23), is shown in **Figure 11(a)** of which the histogram is given in **Figure 11(b)**. A steady DQO model performance in forward prediction by 36 forward-step forecast makes the method appealing for safety application (**Figure 12**).

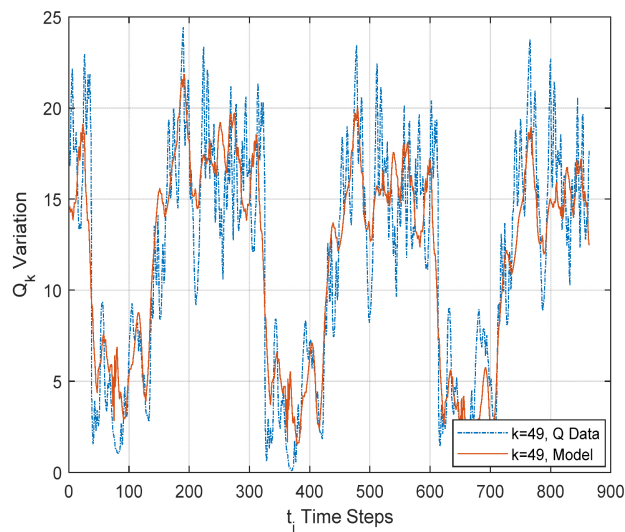
The time gain by using the 36-step forward predicting DQO model against the real-time input data is directly evaluated. The temporal Methane concentration variation,  $c_{TG}$  in quantum vector form is back-calculated from the modeled  $Q_m^{i+36}$  prediction for the  $100qm$  Methane source term. The comparison between the  $Q_m^{i+36}$  (forward modeled) and  $Q^i$  (measured) concentrations are depicted in

**Figure 13**, showing an actual time gain of 140 minutes for forecasting a future threshold crossing event at 2% against the real-time data.

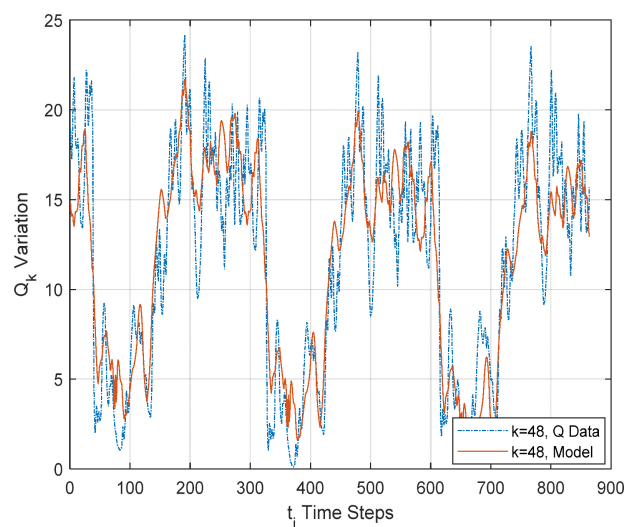
**E7. Illustrative example of a DQO model fit and forecast using repeated, small forward steps**



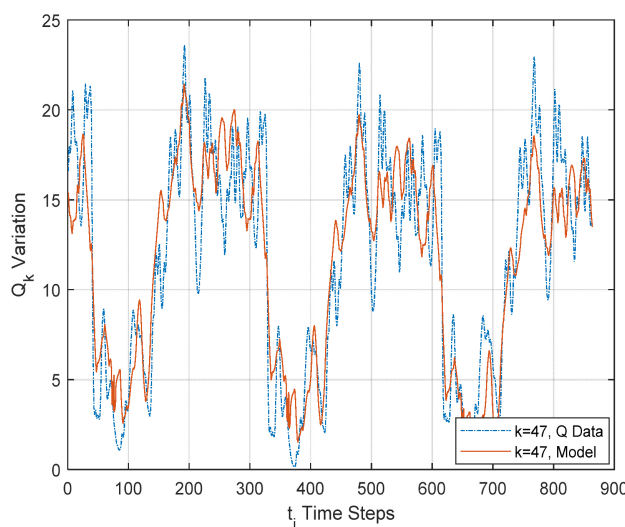
(a)



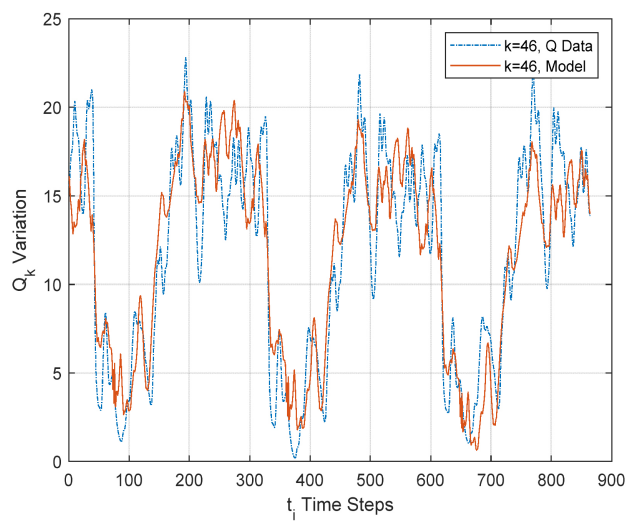
(b)



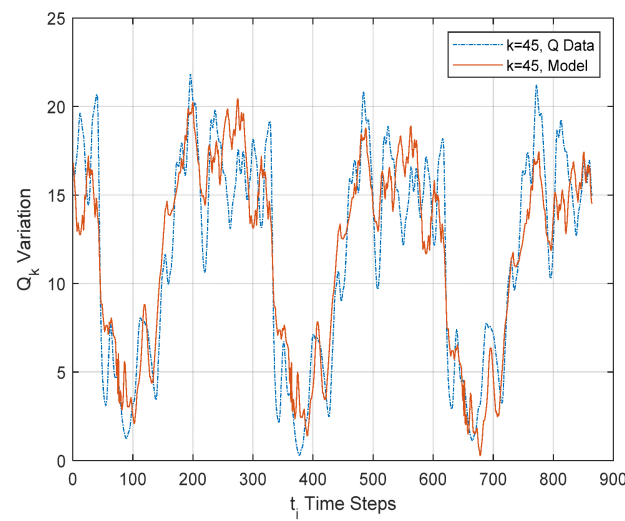
(c)



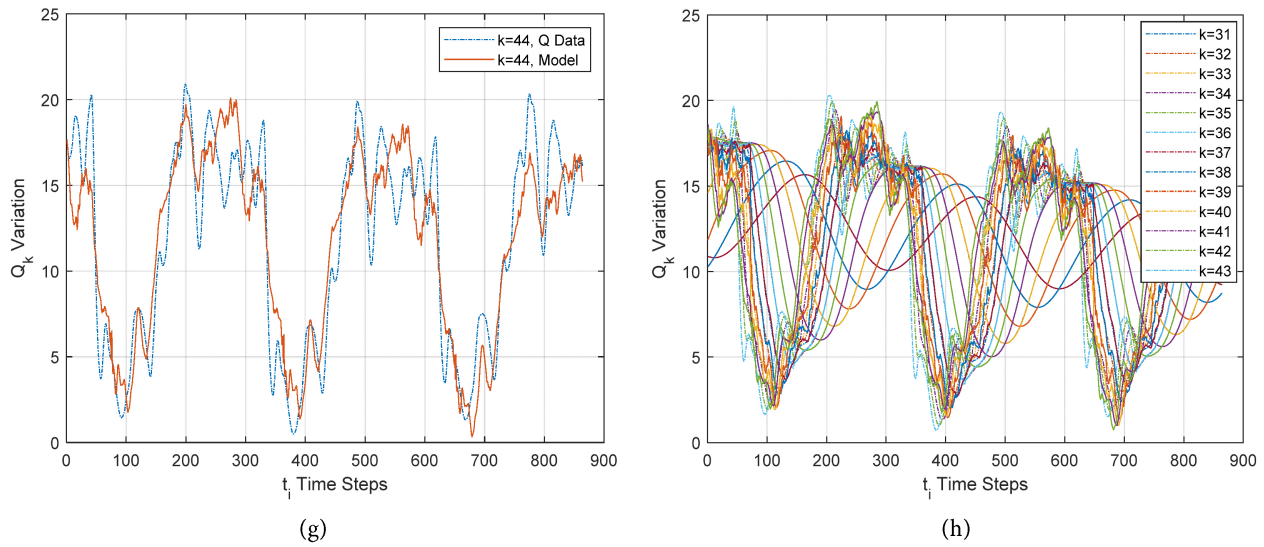
(d)



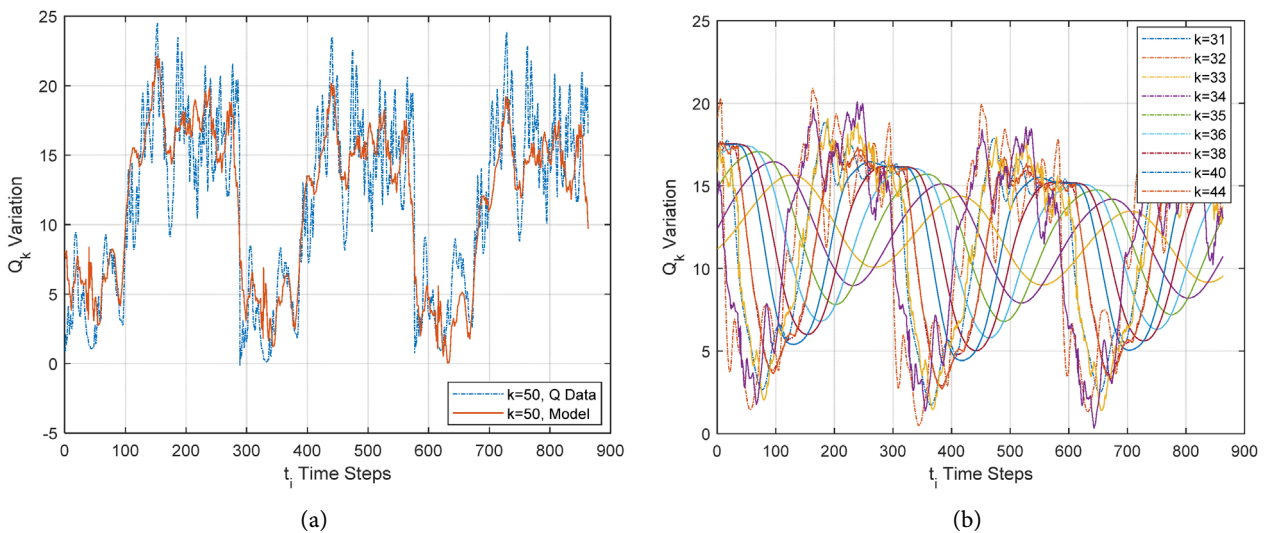
(e)



(f)



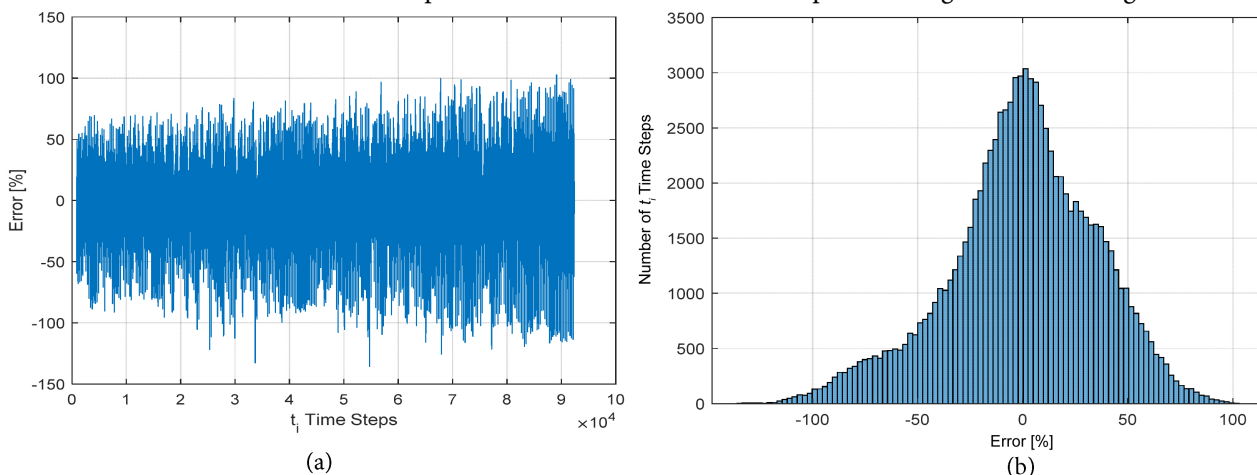
**Figure 8.** (a)-(h) Variation of the  $Q_M^i$  and  $Q_D^i$  vectors with time for CH4 source input data series from measurement; (a)-(g):  $k \in [44,50]$  (with each individual pair and  $k$  marked); (h):  $k \in [31,43]$  (with only each  $k$  marked as no difference I between  $Q_D^i$  and  $Q_M^i$  can be seen).



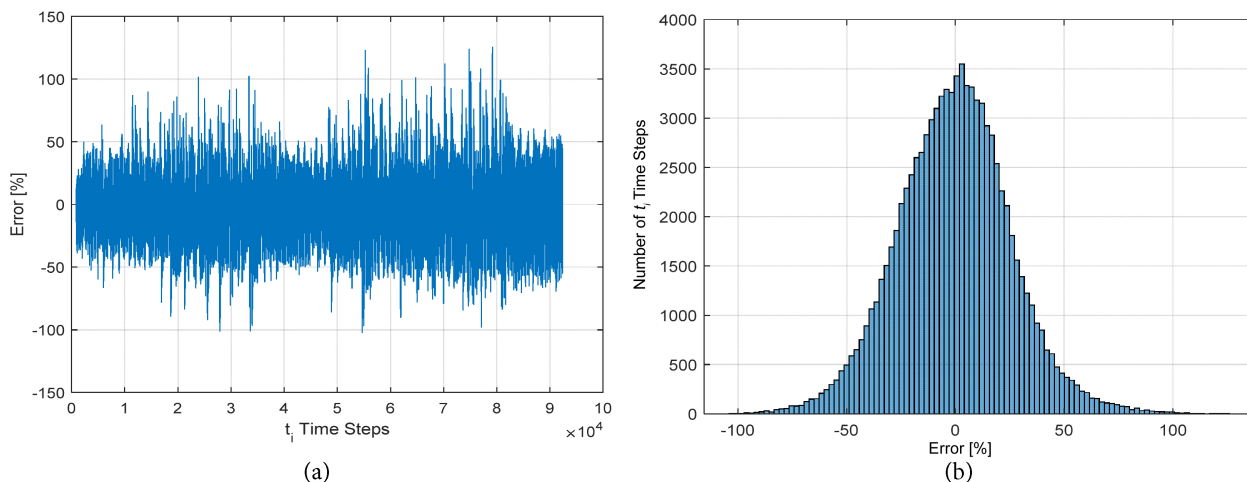
**Figure 9.** (a), (b) Variation of selected  $Q_M^{i+z}$  and  $Q_D^{i+z}$  vectors forward predicted by  $z = 36$  time steps, compared with input data series from CH4 source measurement; (a):  $k = 50$  (with marked pairs of  $Q_D^{i+z}$  and  $Q_M^{i+z}$ ); (b):  $k \in [31,44]$  (with only each  $k$  marked as no difference between model and data can be seen).

The same input data and contaminant gas transport system is used in a demonstrational example for the same task but with the application of refined forward prediction steps to  $z = 1$ , applying Equation (16). The goal is to forecast the effect of the  $qms$  gas inburst by the DQO model for preventive intervention before condition for a fatal explosive condition may happen. The DQO model training steps is reduced to  $z_{train} = 1$  step, while the  $z_{predict} = z = 20$  is used by experimentation for forward prediction for achieving a similar result to the ex-

ample in E6. The shortest forward step in training allows reducing the training



**Figure 10.** (a), (b) Variations of normalized absolute DQO model error for CH<sub>4</sub> source (last two days disturbed time period excluded); (a) Model error over the training time window,  $E(i)$ ; and (b) Histogram of the model error.



**Figure 11.** (a), (b) Variations of normalized absolute DQO model error for CH<sub>4</sub> source (last two days disturbed time period excluded); (a) Model error in  $z = 36$  time steps (180 minutes) forward prediction,  $E_z(i)$ ; and (b) Histogram of the model error.

window to  $w = 360$  without destabilizing model training. As before, the DQO model is trained to match only the last 20 quantum components amid the short-lived memory of the gas transport system.

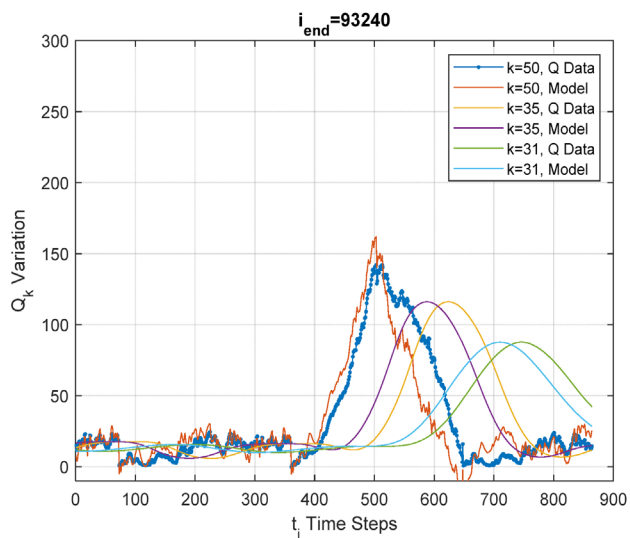
The DQO model are determined with the LSQ solution of Equation (20) at each  $i$  time step over the  $w = 360$ -step training window. The model prediction,  $Q_m^i$ , is calculated for quality check from the quantum-processed input data  $Q^{i-1}$ , applying the power index formula in Equation (16) for model forecast, using  $z = 20$ :

$$Q_m^{i+z} = (\phi^{i,1})^z Q^i \tag{25}$$

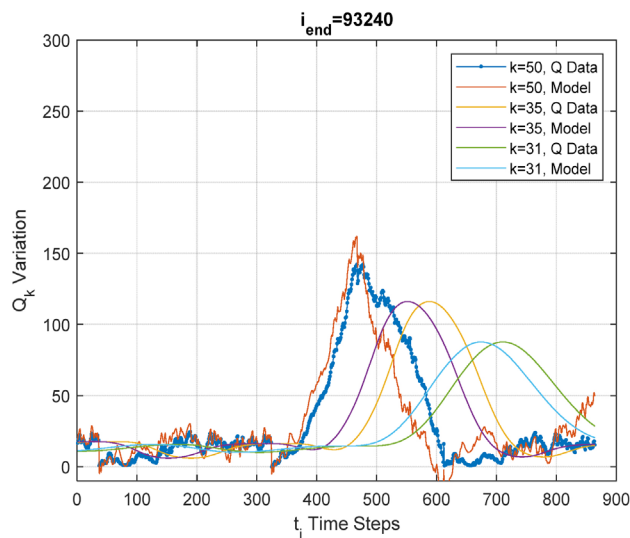
The variation of the  $Q_m^i$  and  $Q^i$  quantum vector components for the  $k \in [31, 50]$  components for the last moving window segment for  $i \in S$  are shown

in **Figures 14(a)-(h)**, ( $i$  being used instead of  $j$  in the notation in the figure).

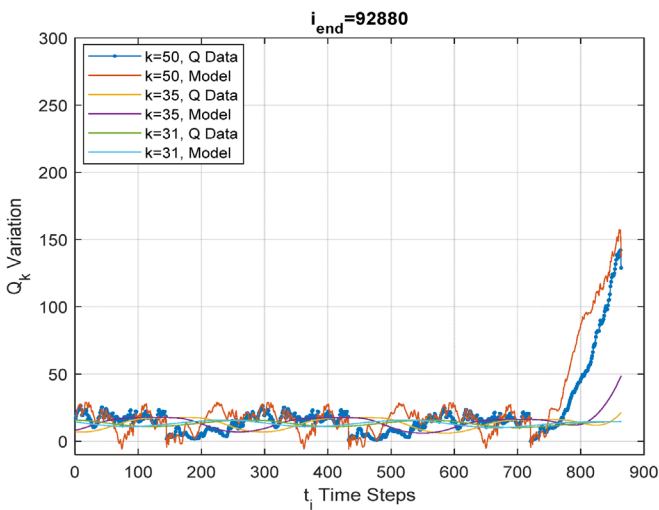




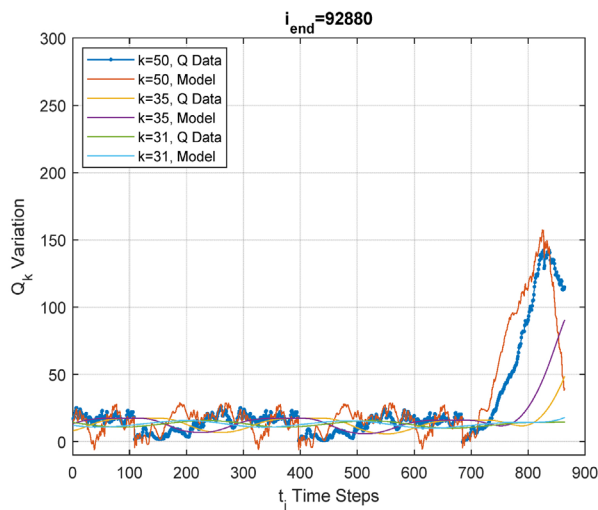
(a)



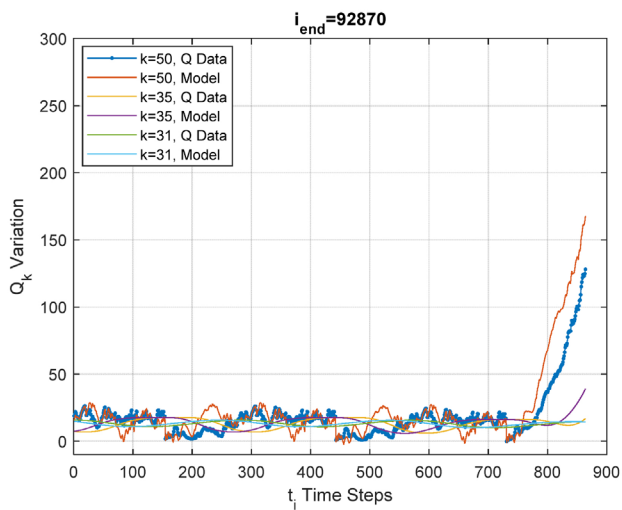
(b)



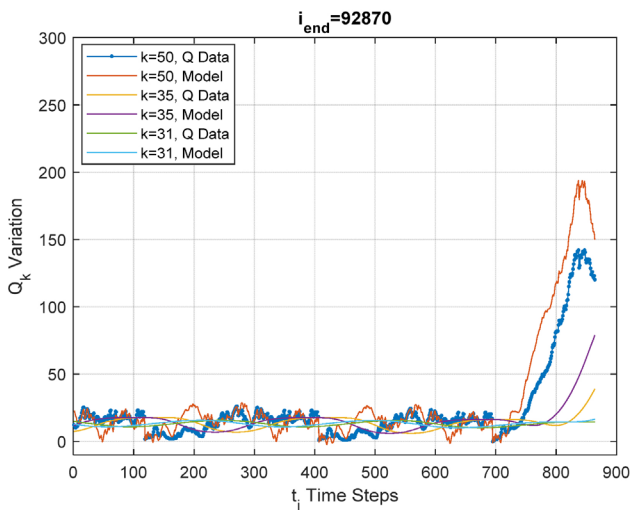
(c)



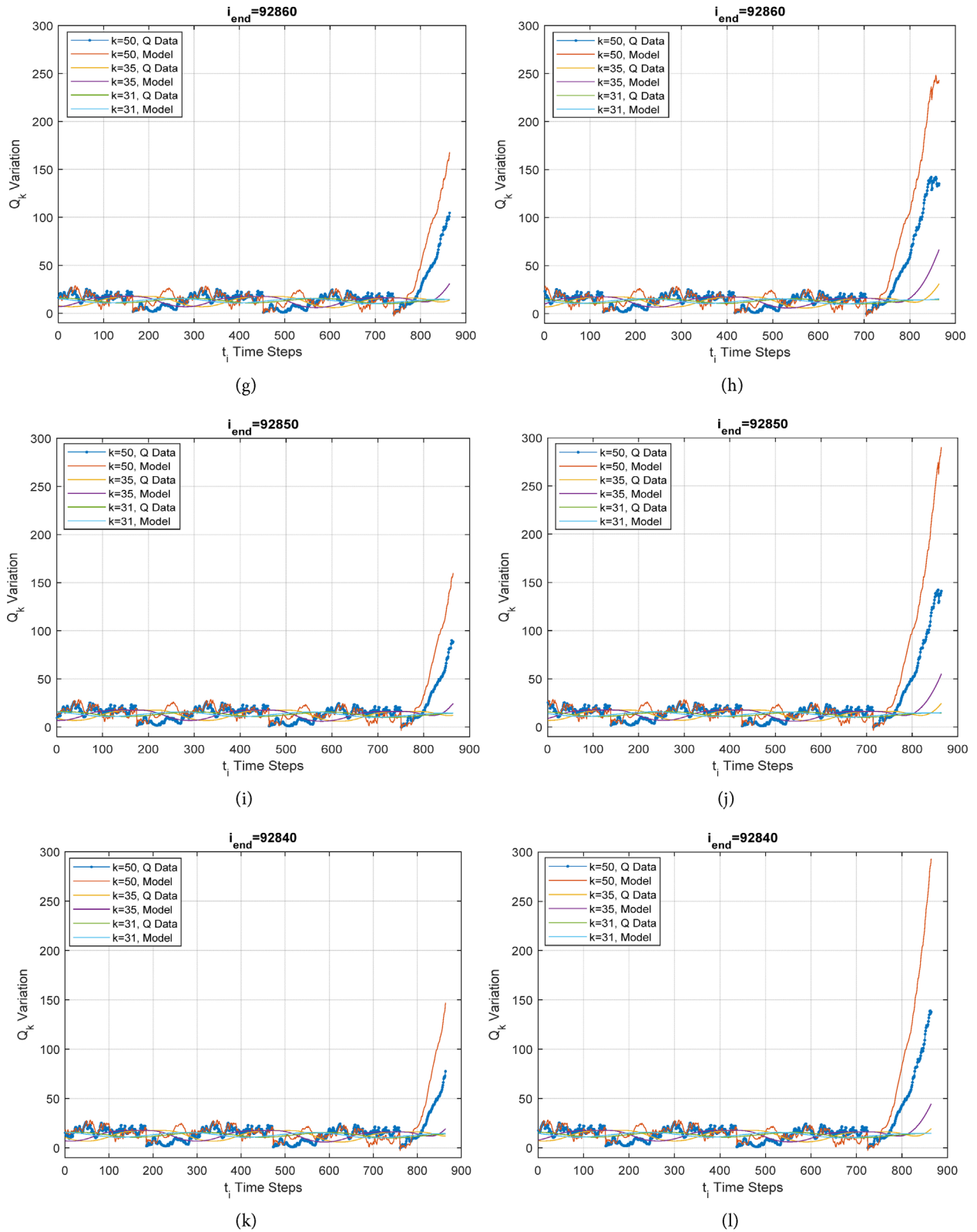
(d)



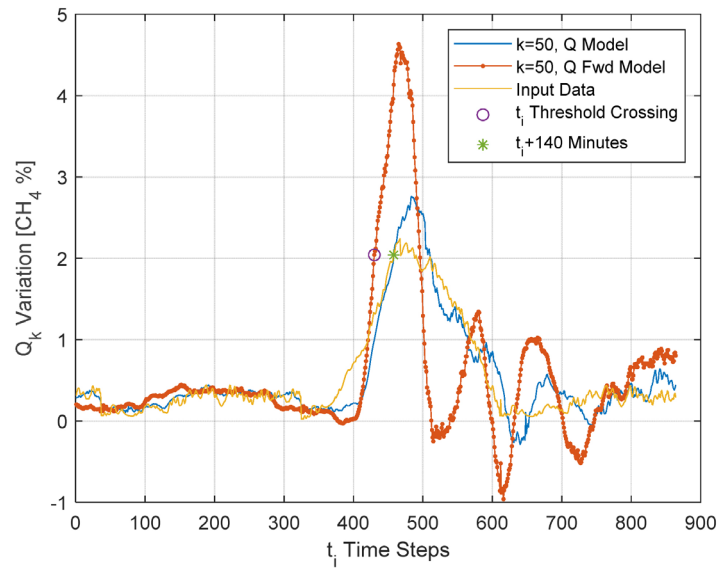
(e)



(f)



**Figure 12.** (a)-(l) Records of DQO CH<sub>4</sub> source model fit and DQO forward prediction over a sliding 3-day time period with changing  $t(i_{end})$  end date; ((a), (c), (e), (g), (i), (k)): DQO model fit for part of a disturbed day; ((b), (d), (f), (h), (j), (l)): 36-step forward prediction from the DQO model.

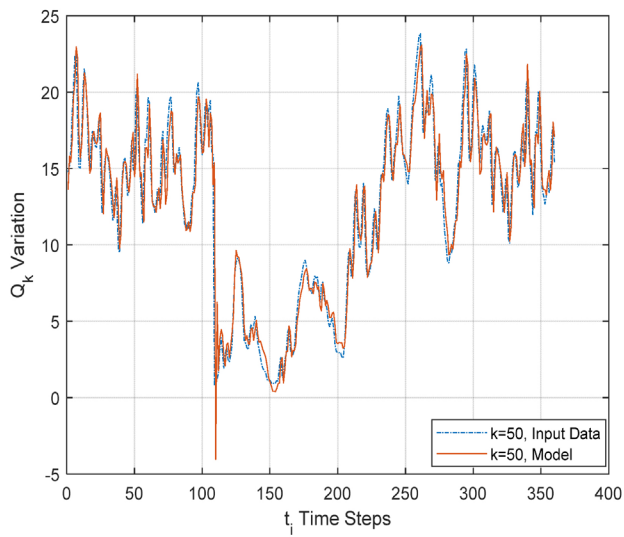


**Figure 13.** DQO model fit and a 36-step CH<sub>4</sub> concentration forward prediction for a disturbed day; an actual 28-step (140 minutes) forward prediction gain is shown from the DQO forward prediction model.

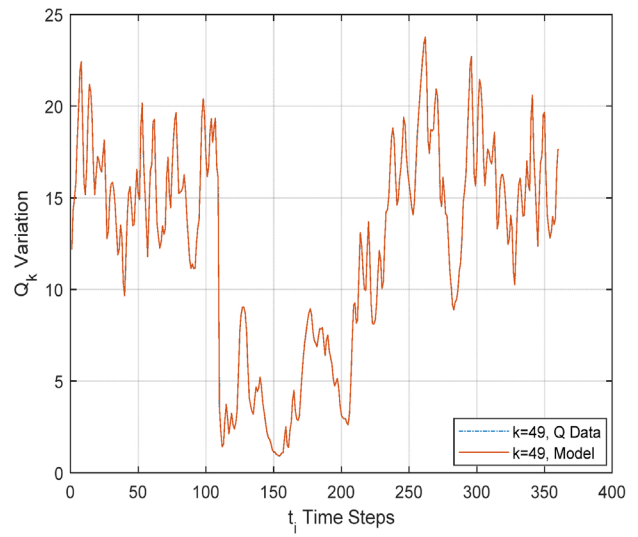
The components of the  $\mathbf{Q}_m^i$  and  $\mathbf{Q}^i$  vectors with time are shown in (a)-(g) for  $k \in [44, 50]$  (with each individual pair and  $k$  marked); and in (h) for  $k \in [31, 43]$ . As shown in **Figures 14(a)-(h)**, the match between the DQO model's output results,  $\mathbf{Q}_m^i$  and the input data,  $\mathbf{Q}^i$ , is excellent for all frequency components over all  $k$  values.

The forward-predicting capability of the DQO model is tested by evaluating forecasted outputs,  $\mathbf{Q}_m^{i+z}$  from previous known values,  $\mathbf{Q}^i$ . Using Equation (25), the model's output,  $\mathbf{Q}_m^{i+z}$ , is calculated at each of future time by  $z = 20$  time steps outside the training time window, while using a 1-step-old DQO matrix,  $\phi^{i,1}$ , on the power index of  $z = 20$ . The forecasted results,  $\mathbf{Q}_m^{i+z}$  are compared with the known future values,  $\mathbf{Q}^{i+z}$ , not used in the DQO model training. **Figure 15(a)** and **Figure 15(b)** show the variation of selected  $\mathbf{Q}_m^{i+z}$  and  $\mathbf{Q}^{i+z}$  vectors over the entire last training window, forward predicted by  $z$  time steps, compared with input data series from measurement. The components of the  $\mathbf{Q}^{i+z}$  and  $\mathbf{Q}_m^{i+z}$  vectors with time are shown in **Figure 15(a)** for  $k = 50$  (with marked pairs of  $\mathbf{Q}^{i+z}$  and  $\mathbf{Q}_m^{i+z}$ ); and in **Figure 15(b)** for  $k \in [31, 44]$ . As shown in **Figure 15(a)** and **Figure 15(b)**, the match between the DQO model's output results and the input data is far lower than the match for model training.

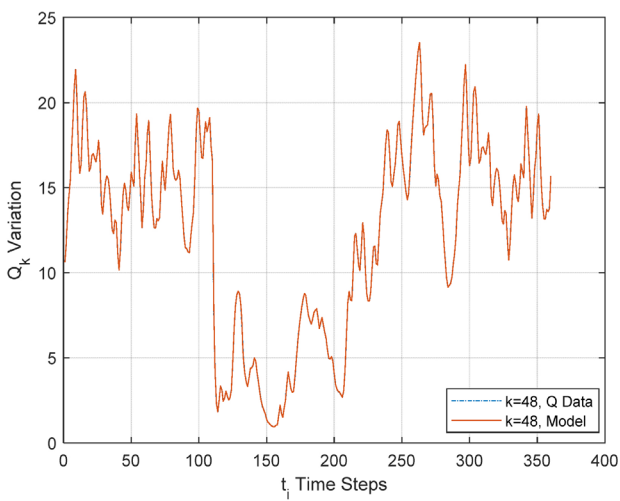
The absolute error of the model fit,  $E(i)$ , for each of the  $327 \times 288$  time step, normalized according to Equation (22), is shown in **Figure 16(a)**. The histogram of the variation is depicted in **Figure 16(b)**. The absolute error of the model fit at forward-predicted instances by  $z = 20$  time steps,  $E_z(i)$ , normalized according to Equation (23), is shown in **Figure 17(a)** of which the histogram is given in **Figure 17(b)**. A steady DQO model performance is seen in forward prediction by 20 forward-step forecast, similar or better in quality than obtained in example E6.



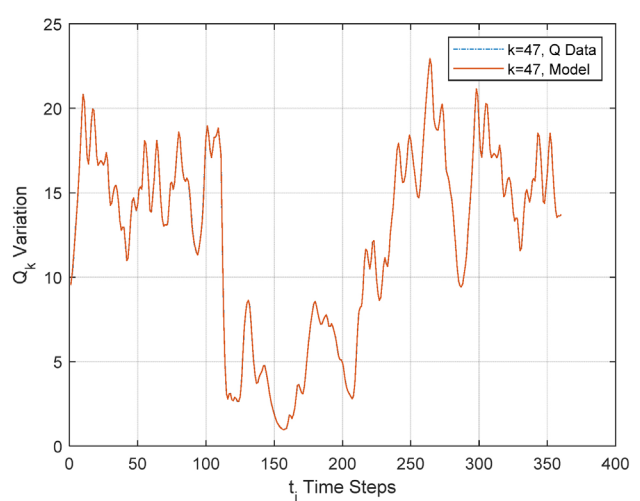
(a)



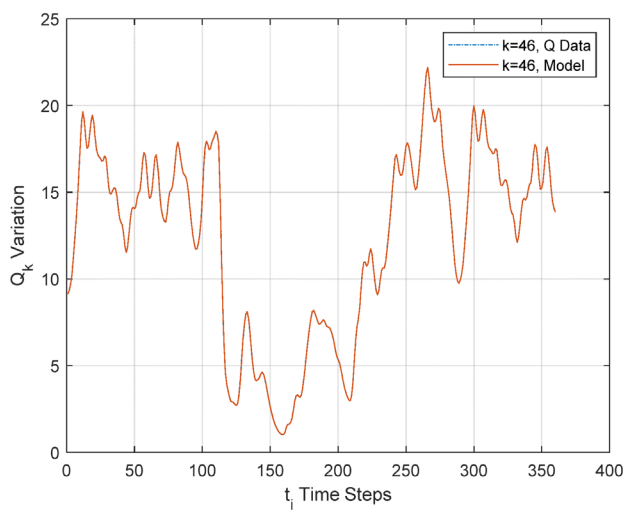
(b)



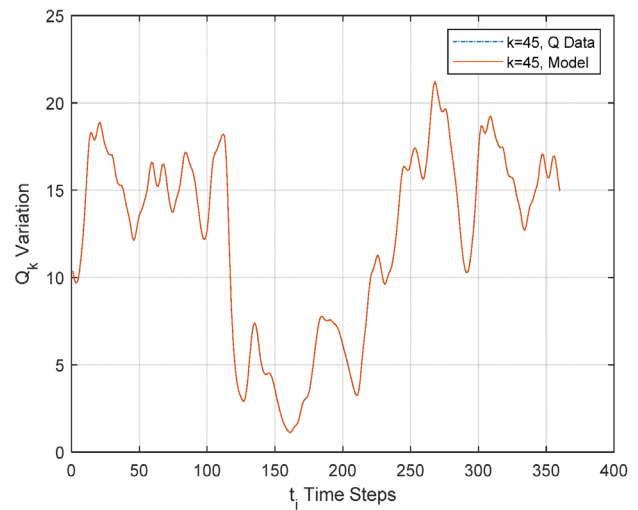
(c)



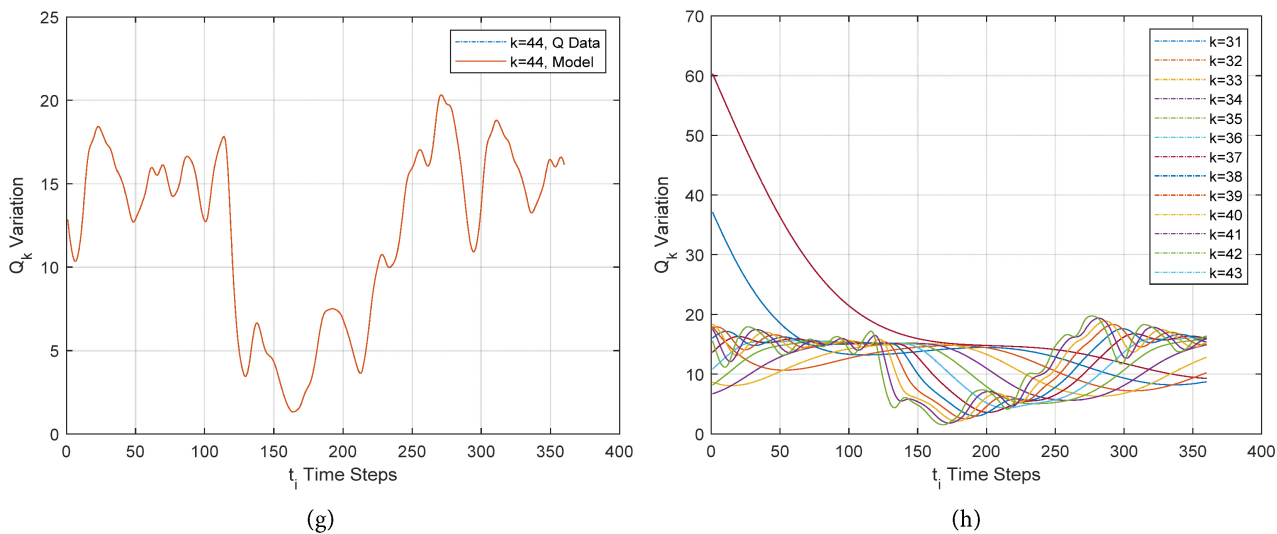
(d)



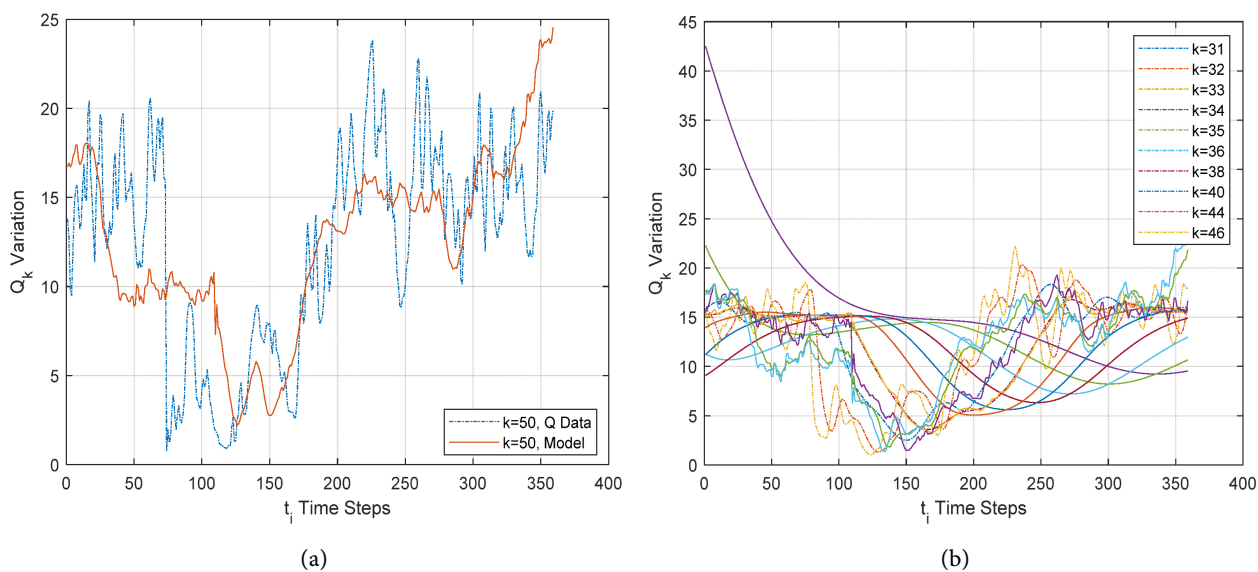
(e)



(f)

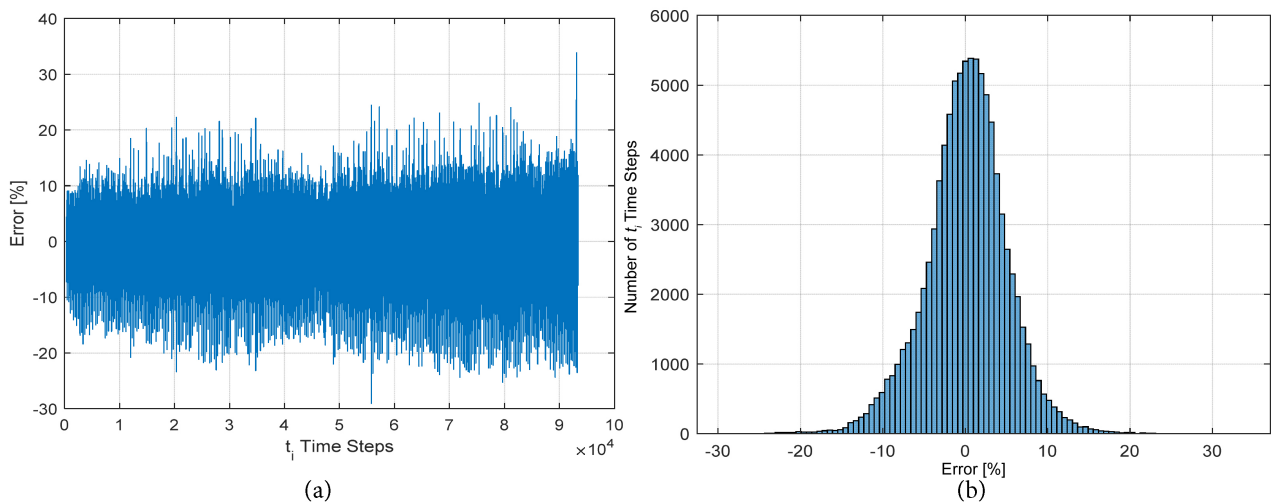


**Figure 14.** (a)-(h). Variation of the  $Q_M^i$  and  $Q_D^i$  vectors with time for CH4 source input data series from measurement; (a)-(g):  $k \in [44, 50]$  (with each individual pair and  $k$  marked, albeit may not be visible); (h):  $k \in [31, 45]$  (with only each  $k$  marked as no difference between  $Q_D^i$  and  $Q_M^i$  can be seen).

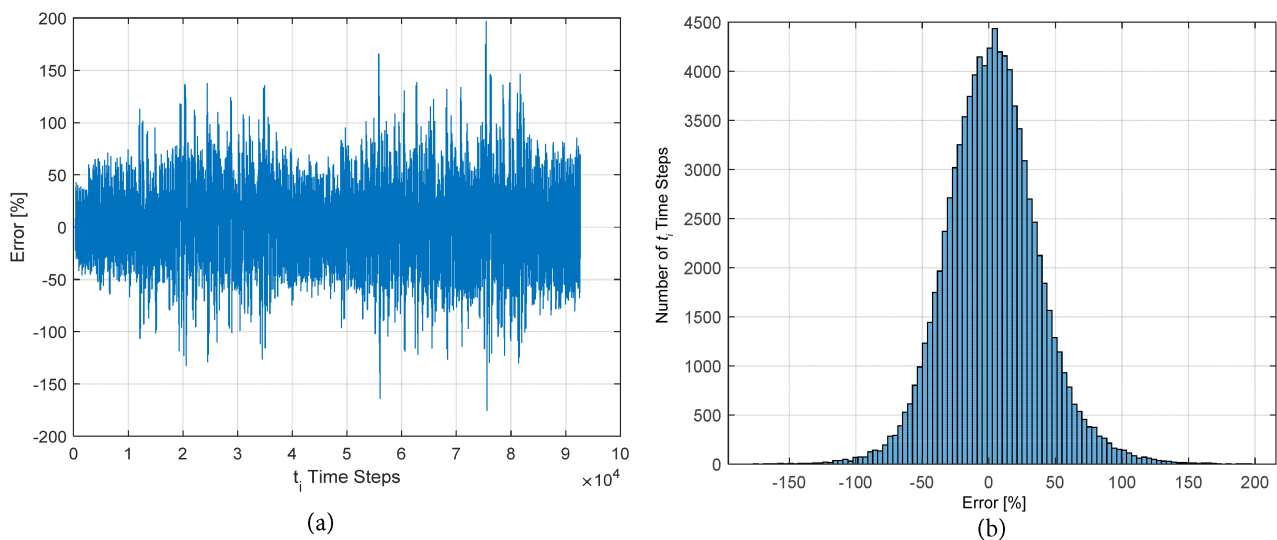


**Figure 15.** (a), (b) Variation of selected  $Q_M^{i+z}$  and  $Q_D^{i+z}$  vectors forward predicted by  $z = 20$  time steps, compared with input data series from CH4 source measurement; (a):  $k = 50$  (with marked pairs of  $Q_D^{i+z}$  and  $Q_M^{i+z}$ ); (b):  $k \in [31, 46]$  (with only each  $k$  marked as no difference between model and data can be seen).

The DQO model performance is analyzed for the critical, disturbed day in forecasting of a CH4 surge, surpassing the threshold value of 2% by a synthetically induced bump during day 325. The time gain by using the 26-step forward predicting DQO model against the real-time input data is directly evaluated. The temporal Methane concentration variation,  $c_{TG}$  in quantum vector form is back-calculated from the modeled  $Q_m^{i+20}$  prediction for the 100qm Methane source term. The comparison between the  $Q_m^{i+20}$  (forward modeled) and  $Q^i$  (measured)



**Figure 16.** (a), (b) Variations of normalized absolute DQO model error for CH<sub>4</sub> source (last two days disturbed time period excluded); (a) Model error over the training time window,  $E(i)$ ; and (b) Histogram of the model error.



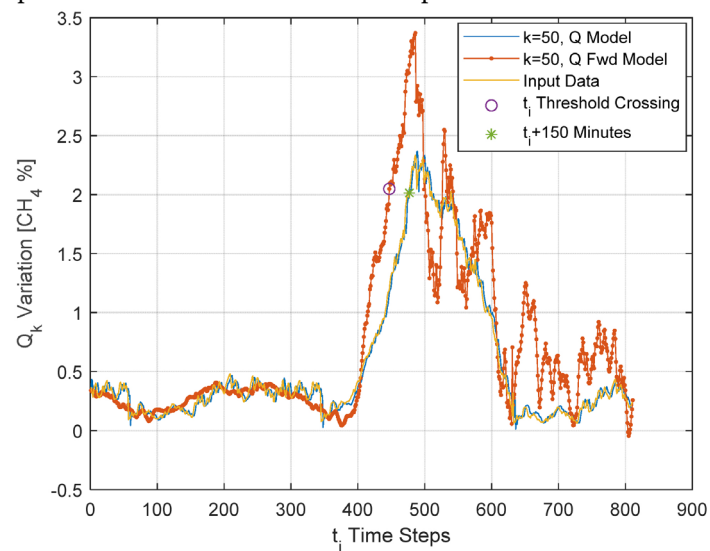
**Figure 17.** (a), (b) Variations of normalized absolute DQO model error for CH<sub>4</sub> source (last two days disturbed time period excluded); (a) Model error in  $z = 20$  time steps (180 minutes) forward prediction,  $E_z(i)$ ; and (b) Histogram of the model error.

concentrations are depicted in **Figure 18**, showing an actual time gain of 150 minutes for forecasting a future threshold crossing event at 2% against the real-time data.

## 5. Brief Discussion of the Results

A dynamic model identification method is described with definition of quantum vectors, representing a time series of data,  $X(t_i)$ . Definitions and examples are given for the time compartmentalization into bins; the data processing and distribution into bins; and transport of the quantum of data between the bins during step-by-step sliding from the most recent to the past time periods. Various

data compression methods are shown for comparison of characteristics including



**Figure 18.** DQO model fit and a 20-step CH<sub>4</sub> concentration forward prediction for a disturbed day; an actual, 150 minutes forward prediction gain is shown from the DQO forward prediction model.

the common, sliding time window averaging and a new property, named the “moving window quantum of data”. The moving window quantum value in each bin is defined from the contained  $X(t_i)$  data for constructing a set of base vectors of the dynamic operator of the system. It is shown that the quantum vector form for retaining past and present data characteristics is most advantageous for time series analysis for short-time and long-time memory effects of the modeled system as the data is efficiently compressed from tens of thousands recorded numbers into only fifty elements without losing pertinent information.

The compressed form of data into quantum vectors is used as the linear space for building a DQO model,  $\phi^{i,z}$ , for the system at every time step for a real-time process. Definition of the  $\phi^{i,z}$  operator and its training data set, as well as the solution for identification from input data are both given in mathematical forms.

Forward prediction is defined, using the  $\phi^{i,z}$  operator as it inherently includes a time step  $z$  for model identification. As shown in three illustrative examples, the quality of model identification of  $\phi^{i,z}$  foretell that of the error in forward prediction, a useful feature in practical applications. A steady DQO model performance within about  $\pm 20\%$  normalized error up to 1 hour forward-step forecast is shown in the E5 outside weather temperature example, making the method appealing, especially in comparison to published results for LSTM NN models with poorer forward prediction performance. In addition to excellent stability, the computational time for model DQO identification and forward prediction at each time step takes 18 milliseconds using a laptop computer.

Two additional examples are shown for analyzing and controlling atmospher-

ic conditions for safety and health in working and living. DQO models are identified and used for forecasting methane concentration variations from monitored data. A hypothesis is tested regarding a time advantage that may be gained by DQO model prediction, and saved for preventive interventions to alleviate impending hazard conditions at any monitored, living or working place. The hypothesis is tested quantitatively, using two forward-prediction algorithms to consider in a mine safety and health applications.

## 6. Concluding Remarks

- A new method is presented for AR time series analysis of a real-time, continuous data stream.
- A new type of data compression, using data quantum vectors, is developed, and implemented for practical applications.
- A new type of DQO model-building and identification method is described.
- Three numerical application examples are shown using real-world input data for DQO model identification. Performance metrics of the DQO model are demonstrated in forward prediction applications.
- The hypothesis test about significant time gain is affirmed by forward prediction using the DQO model in the racing for preventive interventions to counter impending hazard events in atmospheric conditions.

## Acknowledgements

A research granted from the Alpha Foundation for Mine Safety and Health is gratefully recognized. The research was thankfully supported by the GINOP-2.3.2-15-2016-00010. “Development of Enhanced Engineering Methods with the Aim at Utilization of Subterranean Energy Resources” project of the Research Institute of Applied Earth Sciences of the University of Miskolc in the framework of the Széchenyi 2020 Plan, funded by the European Union, co-financed by the European Structural and Investment Funds.

## Conflicts of Interest

The author declares no conflicts of interest regarding the publication of this paper.

## References

- [1] Jung, D. and Choi, Y. (2021) Systematic Review of Machine Learning Applications in Mining: Exploration, Exploitation, and Reclamation. *Minerals*, **11**, 148. <https://doi.org/10.3390/min11020148>
- [2] Rojas, R. (1996) *Neural Networks*. Springer, Heidelberg. <https://doi.org/10.1007/978-3-642-61068-4>
- [3] Lin, T., Horne, B.G., Tino, P. and Giles, C.L., (1996) Learning Long-Term Dependencies in NARX Recurrent Neural Networks. *IEEE Transactions on Neural Networks*, **7**, 1329-1338. <https://doi.org/10.1109/72.548162>



- 
- [4] Miao, K., Han, T., Yao, Y., Lu, H., Chen, P., Wang, B. and Zhang, J. (2020) Application of LSTM for Short Term Fog Forecasting Based on Meteorological Elements. *Neurocomputing*, **408**, 285-291. <https://doi.org/10.1016/j.neucom.2019.12.129>
- [5] Dias, T., Belle, B., and Danko, G., (2021) Methane Concentration forward Prediction Using Machine Learning from Measurements in Underground Mines. *AusIMM Conference*, Perth, 6-8 December 2021, 1-21.
- [6] Horvath, L. and Koloszka, P. (2012) *Inference for Functional Data with Applications*. Springer, New York.
- [7] Box, G. and Jenkins, G. (1970) *Time Series Analysis: Forecasting and Control*. Holden-Day, San Francisco.
- [8] Milionis, A. and Galanopoulos, N. (2019) Forecasting Economic Time Series in the Presence of Variance Instability and Outliers. *Theoretical Economics Letters*, **9**, 2940-2964. <https://doi.org/10.4236/tel.2019.98182>
- [9] Pan, W., Lai, D., Song, Y. and Follis, J. (2019) Time Series Analysis of Energy Intensity, Value Added Tax and Corporate Income Tax: A Case Study of the Non-Ferrous Metal Industry, Jiangxi Province, China. *Journal of Data Analysis and Information Processing*, **7**, 108-117. <https://doi.org/10.4236/jdaip.2019.73007>
- [10] Abebe, S. (2018) Application of Time Series Analysis to Annual Rainfall Values in Debre Markos Town, Ethiopia. *Computational Water, Energy, and Environmental Engineering*, **7**, 81-94. <https://doi.org/10.4236/cweee.2018.73005>
- [11] Danko, G. (2006) Functional or Operator Representation of Numerical Heat and Mass Transport Models. *ASME Journal of Heat Transfer*, **128**, 162-175. <https://doi.org/10.1115/1.2136919>
- [12] Shumway, R. and Stoffer, D. (2017) *Time Series Analysis and Its Applications: With R Examples*. 4th Edition, Springer, Cham. <https://doi.org/10.1007/978-3-319-52452-8>
- [13] Park, K.I. (2018) *Fundamentals of Probability and Stochastic Processes with Applications to Communications*. Springer, Cham. <https://doi.org/10.1007/978-3-319-68075-0>