STUDIA INFORMATICA
Nr 1 (28) Systems and information technology 2023

Jerzy TCHÓRZEWSKI[1],
ORCID: 0000-0003-2198-7185

Wojciech NABIAŁEK[2],
ORCID: 0000-0002-8361-2816

Artur KSIĘŻOPOLSKI[3]
ORCID: 0009-0006-6686-9093

[1,2] Siedlce University of Natural Sciences and Humanities
Faculty of Exact and Natural Sciences
Institute of Computer Science
ul. 3 Maja 54, 08-110 Siedlce, Poland

[3] Student at University of Natural Sciences and Humanities
Faculty of Exact and Natural Sciences
Institute of Computer Science
ul. 3 Maja 54, 08-110 Siedlce, Poland

Analysis of a web application supporting gym management in terms of control and systems theory

DOI: 10.34739/si.2023.28.03

Abstract. An analysis of the web application supporting the management of the gym was carried out, which was designed in terms of the theory of control and systems. The obtained results of measurements of input and output quantities of selected subsystems were used to carry out the identification, as a result of which models of web application subsystems were obtained. On the basis of the obtained subsystem models, appropriate models were then designed in Simulink for the purposes of simulation and comparative studies of the behavior of subsystem models in relation to real systems that are web applications, including an appropriate analysis of the web application system model was carried out under the name of the gym on its selected subsystems, including in particular the subsystem called Generating a diet. In addition, testing and simulation studies were carried out to check the correct functioning of the web application management system model and the introduction of appropriate changes to the model was proposed, and the obtained research results were discussed, indicating the high accuracy of the obtained web application models.

Keywords. Gym management, Identity theory, MATLAB & Simulink Environment, Modeling, Simulations, Web systems.

1. Introduction

With the increase in the number of web applications, the need to automate their operation and the need to analyze them as systems increases. Such an analysis is possible e.g. on the basis of control and systems theory, in particular in the field of examining the degree of its internal organization and the level of control (management). The presented research results are part of this modern research direction, which includes the analysis of a web application supporting gym management using control theory and systems engineering [1, 4, 14, 24-25]. It was assumed that the analyzed system is the gym management system, which is a robotic control system cooperating with a changing environment [18-19, 21], which as a computer program is a system as an executing process during the operation of a web application [24-25]. In order to capture such an application in terms of control and systems theory, it is important to identify its functioning in order to create a system model [8-10, 21-22, 28-29]. Therefore, in the first place, the places where the data entered and the results obtained should be measured were adopted in the web application.

Then, detailed measurements were made and the subsystem functioning in the web application system was identified. Diet Generation, which in turn consists of five successive subsystems: Breakfast Generation, Lunch Generation, Lunch Generation, Tea Generation, and Dinner Generation[1]. Identification of the above of individual subsystems was carried out using the ARX parametric method in MATLAB and Simulink and System Identification Toolbox [12]. Then, after obtaining all subsystem models, a simulation model of the Diet Generation

[1] The following variables were also used in the software: Breakfast, Second Breakfast, Dinner, Afternoon Snack, Super.

system was designed in Simulink, followed by verification, simulation and comparative tests of the functioning of the implemented web application system as a real system.

As a result of comparative studies of the obtained model in relation to the web application system, appropriate absolute and relative errors were determined. The obtained results of research on absolute errors and relative errors show that such solutions can be used in the automation or robotization of web processes, which in this respect sets a new direction for research. It is worth noting here that the obtained system model, after receiving the input data, behaves in a similar way as the real system of the web application in terms of the permissible absolute and relative error.

2. System modeling for the analysis of web applications

2.1. The essence of systems modeling

Modeling is an experimental method of researching and learning about complex systems, systems, objects, processes and phenomena, leading to obtaining their models. The modeling process begins with a thorough knowledge and understanding of the essence of the tested system, object, process or phenomenon, and ends with the verification of the obtained model. Models, in terms of functionality, can be very similar to the real system, but they never fully correspond to the system functioning in the form of an implemented program in a real development environment.

One of the main disadvantages of such models is that they are inseparable from certain features of the system that may be missed during modeling. Modeling methods are divided into at least three basic groups, i.e. analytical methods, identification methods and neural methods [1, 2, 3, 11, 13, 19-20, 23]. In this study, the identification method by name was used parametric method ARX in MATLAB and Simulink environment with the use of System Identification Toolbox (SIT) [3, 7, 11-12, 26, 29].

2.2. Identification methods

Identification methods are used to obtain a system model where it is not possible to design an analytical model as well as artificial intelligence models, including neural models [17, 21, 28]. When talking about identification methods, we mean finding a system model for which the output values are close to the real system for the same specific values of input variables for the adopted identification method. The identification process is an iterative process carried out in various programming environments [2, 6-8, 12-13, 15-16, 23, 29], which allows to check the compatibility of the behavior of the system model in relation to the system for the same input signals in a specific time interval. In simple terms, the identification process can be reduced to the following basic steps [3, 8-10, 21, 27-29]:

1) preparation of an identification experiment, including measurement of input and output values,
2) conducting an identification experiment, including the selection of an identification method, etc.,
3) initial processing of measurement data, including e.g. elimination of coarse errors, missing data, etc.,
4) selection of the model class in relation to the obtained measurement data (e.g. continuous or discrete model, deterministic or stochastic or probabilistic model, linear or non-linear model, etc.,
5) selection of the model type from the selected class based on the initial assessment, such as model type: AR, ARX, MAX, ARMAX, ARIX, GARCH, etc.,
6) selection of the model structure, i.e. appropriate model parameters,
7) estimating the parameters of the selected model,
8) testing and verification of the model,
9) determination of the model error in relation to the system,
10) checking the quality of the model, etc.

In these identification studies, a parametric model was used, the so-called AutoRegressive with eXogenousinput (ARX), which concerns obtaining a model in the form of a polynomial binding the current output y(t) with a finite number of outputs y(t-k) and inputs u(t-k), which is written in the form [5-6]:

$$y(t) + a_1 y(t-1) + \ldots + a_{na} y(t-na) = b_1 u(t-nk) + \ldots + b_{nb} u(t-nk-nb+1) \tag{1}$$

where:

na – degree of polynomial A(z),

nb – degree of B(z) polynomials,

nk- number of time delay units.

For Multi Input Multi Output (MIMO) systems, the parameters na, nb and nk are vectors.

2.3. A web application supporting gym management

In times of modern workaholism and unhealthy lifestyle, it is important to systematically take care of physical health. Regularly practicing non-competitive sports, including running, cycling, swimming or going to the gym should be the priority of every self-respecting person. Due to the fact that the demand for this type of physical activity is growing, various types of gyms with services for the population are being created, but running them, regardless of the scale of business, is not easy, because customer service consumes a lot of time [5]. To meet these new challenges, web applications are being built to provide administrative support for gym participants and to support gym management.

Thanks to this type of automation, the process of buying passes or signing up for classes is completely electronic, and the user can control the time of their pass or monitor information about their classes and other opportunities offered by the gym without the intervention of an employee. The system that was selected for the study consists of several subsystems, which as a whole definitely improve the work in gym management. The basic functions of the system regarding power plant management, such as: function name, function argument, returned values, sample input data, sample output data, verbal description, etc. are described in [5].

3. Measurements of data on the functioning of the web application

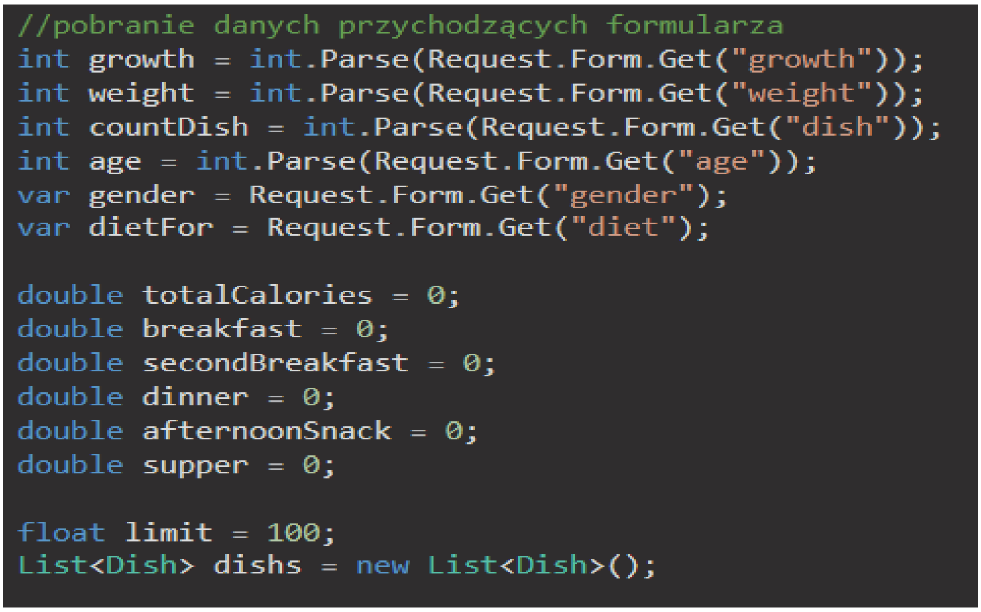
Obtaining an identification model in a parametric form requires measurements of data values in input-output pairs. For these purposes, control measurement points were introduced in the gym management web application in the appropriate places of its individual subprograms. A total of nine basic subroutines in the examined web application were distinguished, which were adopted as subsystems.

Each of them was first subjected to measurements, and then using them to identify the parametric ARX in order to obtain nine basic models of the web application system. These models can be easily combined in Simulink into a simulation model of the web application system [6]. A detailed identification model of the subroutine called Diet Generation was developed, which consists of five outputs, and consequently of five MISO (Multi Input Single Output) subsystems. Each of these subsystems had the same number of inputs of six. So, finally, in this case, the Diet Generation subsystem model consists of five subsystems with the same number of six inputs.

The selection of an appropriate identification model is very difficult, and its appropriate selection is very important for further simulation research. To choose the best model, it is worth using the tools available in the MATLAB environment library entitled System Identification Toolbox [7-10, 12, 20-23, 29]. In order to generate the ARX model, the appropriate input data in the form of a matrix, here [u], and the output data in the form of a matrix, here: [y1], [y2], [y3], [y4] and [y5], had to be imported into MATLAB's Workspace. For each output (yi), the appropriate models had to be generated using SIT.

3.1. Description of the Generate Diet function called GenerateDiet

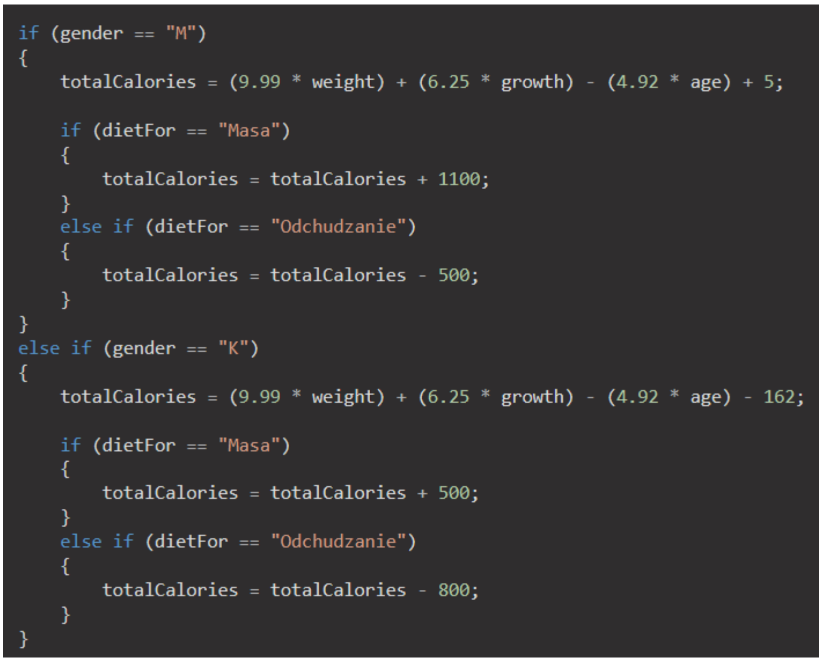
GenerateDiet(int growht, int weight, int age, int ile dan, string dietaNa, string gender) is a function used to generate a diet for the user. At the very beginning, data is collected from the form that must be completed by the user (Figure 1).



```
//pobranie danych przychodzących formularza
int growth = int.Parse(Request.Form.Get("growth"));
int weight = int.Parse(Request.Form.Get("weight"));
int countDish = int.Parse(Request.Form.Get("dish"));
int age = int.Parse(Request.Form.Get("age"));
var gender = Request.Form.Get("gender");
var dietFor = Request.Form.Get("diet");

double totalCalories = 0;
double breakfast = 0;
double secondBreakfast = 0;
double dinner = 0;
double afternoonSnack = 0;
double supper = 0;

float limit = 100;
List<Dish> dishs = new List<Dish>();
```

Figure 1. Listing of data regarding variables collected from the user from the form. Designations: growth – We1, weight – We2, countDish – We3, age – We4, gender – We5, dietaFor – We6. Source: Own study [5-6].

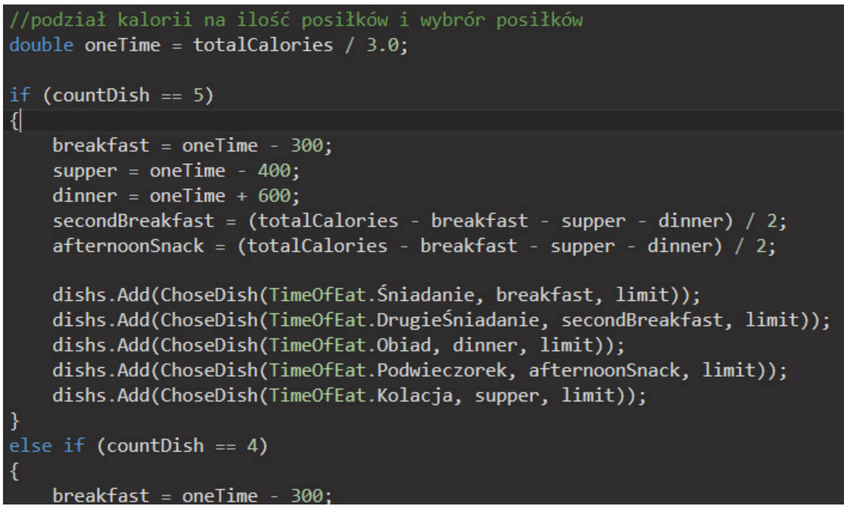
The data that the user must enter are: weight, height, age, number of dishes, type of diet (e.g. whether the diet is to be for slimming or mass, and for what gender, etc.). This function then uses the Mifflin-StJeor method to calculate the appropriate caloric needs for a given user (Figure 2).



```
if (gender == "M")
{
    totalCalories = (9.99 * weight) + (6.25 * growth) - (4.92 * age) + 5;

    if (dietFor == "Masa")
    {
        totalCalories = totalCalories + 1100;
    }
    else if (dietFor == "Odchudzanie")
    {
        totalCalories = totalCalories - 500;
    }
}
else if (gender == "K")
{
    totalCalories = (9.99 * weight) + (6.25 * growth) - (4.92 * age) - 162;

    if (dietFor == "Masa")
    {
        totalCalories = totalCalories + 500;
    }
    else if (dietFor == "Odchudzanie")
    {
        totalCalories = totalCalories - 800;
    }
}
```

Figure 2. Listing of calculations of the appropriate caloric demand based on the data received from the user. Source: Own study [5-6].

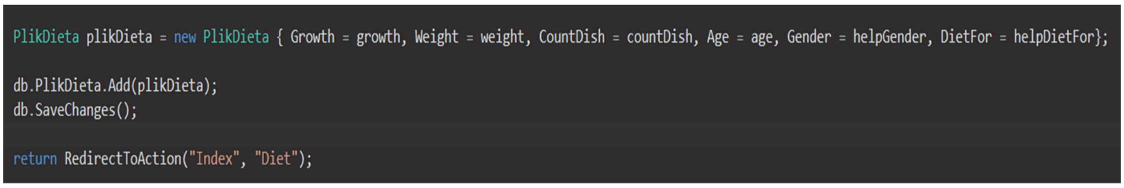
On this basis, taking into account the number of dishes during the day, dishes are selected that best match the user's needs in terms of calories, as shown in Figure 3.



```
//podział kalorii na ilość posiłków i wybrór posiłków
double oneTime = totalCalories / 3.0;

if (countDish == 5)
{
    breakfast = oneTime - 300;
    supper = oneTime - 400;
    dinner = oneTime + 600;
    secondBreakfast = (totalCalories - breakfast - supper - dinner) / 2;
    afternoonSnack = (totalCalories - breakfast - supper - dinner) / 2;

    dishs.Add(ChoseDish(TimeOfEat.Śniadanie, breakfast, limit));
    dishs.Add(ChoseDish(TimeOfEat.DrugieŚniadanie, secondBreakfast, limit));
    dishs.Add(ChoseDish(TimeOfEat.Obiad, dinner, limit));
    dishs.Add(ChoseDish(TimeOfEat.Podwieczorek, afternoonSnack, limit));
    dishs.Add(ChoseDish(TimeOfEat.Kolacja, supper, limit));
}
else if (countDish == 4)
{
    breakfast = oneTime - 300;
```

Figure 3. Listing of food selection based on caloric requirements in a gym management system. Source: Own study [5-6].

The selected dishes are returned to the user's view. In order to measure data on the functioning of the web application, a new functionality has been designed for the system, responsible for saving input data and output data. For the purpose of recording such data in the database, an additional table was created in which these data are saved. Figure 4 shows that the function writes the input data to the cache (RAM) first. They are saved to a table in the database, and then they are further processed by the function, according to the operation of the system.

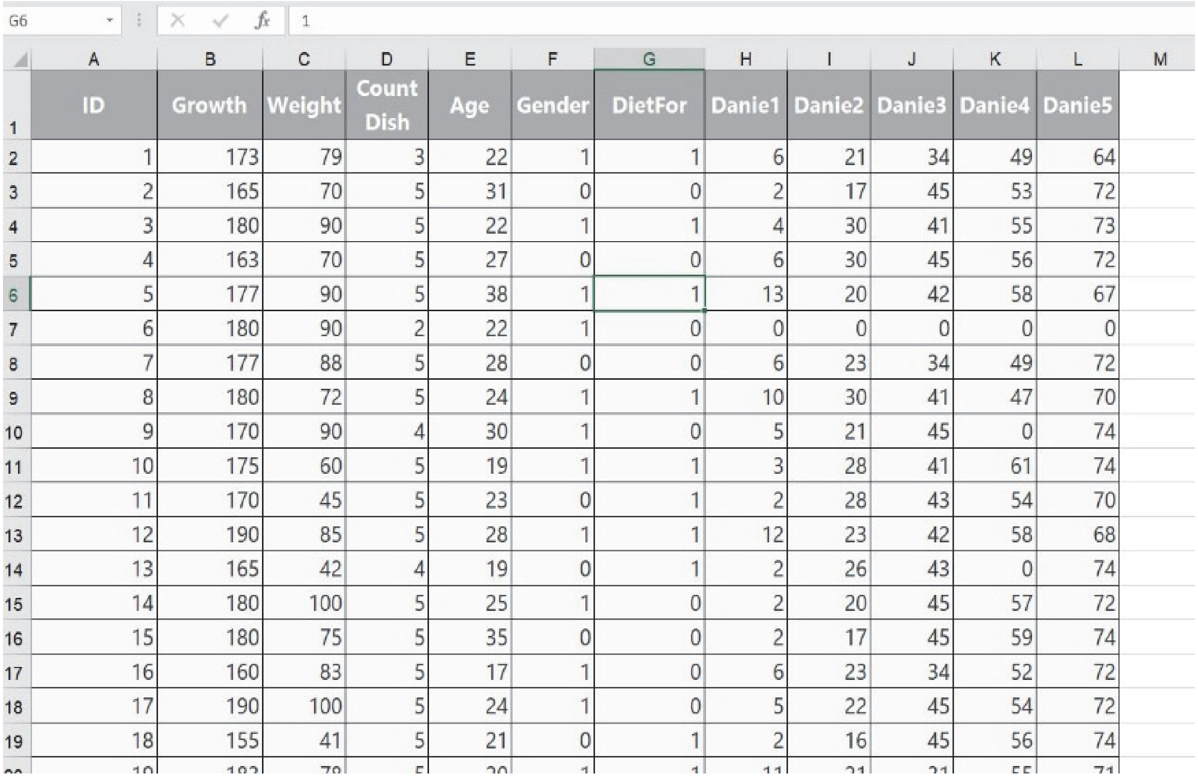


```
PlikDieta plikDieta = new PlikDieta { Growth = growth, Weight = weight, CountDish = countDish, Age = age, Gender = helpGender, DietFor = helpDietFor};

db.PlikDieta.Add(plikDieta);
db.SaveChanges();

return RedirectToAction("Index", "Diet");
```

Figure 4. Listing of saving the input and output of a function to a table in the database. Source: Own study [6, 12].

The next step after the data processing was completed was to save the processed data (output data) to a table in the database. Before the function ends, the system runs the save method and all output data goes to the appropriate database tables. The last step of preparing the data for identification was to download them from the database and export them to an Excel file in such a way that identification could be carried out on them.

Due to the fact that at the beginning of the design of the table for storing data, it was already designed, at the moment it only needed to be copied. Thus, the appropriate "Select" was made on the database and then all data was exported to an Excel file as shown in Figure 5.



ID	Growth	Weight	Count Dish	Age	Gender	DietFor	Danie1	Danie2	Danie3	Danie4	Danie5
1	173	79	3	22	1	1	6	21	34	49	64
2	165	70	5	31	0	0	2	17	45	53	72
3	180	90	5	22	1	1	4	30	41	55	73
4	163	70	5	27	0	0	6	30	45	56	72
5	177	90	5	38	1	1	13	20	42	58	67
6	180	90	2	22	1	0	0	0	0	0	0
7	177	88	5	28	0	0	6	23	34	49	72
8	180	72	5	24	1	1	10	30	41	47	70
9	170	90	4	30	1	0	5	21	45	0	74
10	175	60	5	19	1	1	3	28	41	61	74
11	170	45	5	23	0	1	2	28	43	54	70
12	190	85	5	28	1	1	12	23	42	58	68
13	165	42	4	19	0	1	2	26	43	0	74
14	180	100	5	25	1	0	2	20	45	57	72
15	180	75	5	35	0	0	2	17	45	59	74
16	160	83	5	17	1	0	6	23	34	52	72
17	190	100	5	24	1	0	5	22	45	54	72
18	155	41	5	21	0	1	2	16	45	56	74

Figure 5. PrintScreen of input and output data transferred from the database to an Excel file. Designations: Daniei (i=1-5) - i – dish, Source: Own elaboration [5-6].

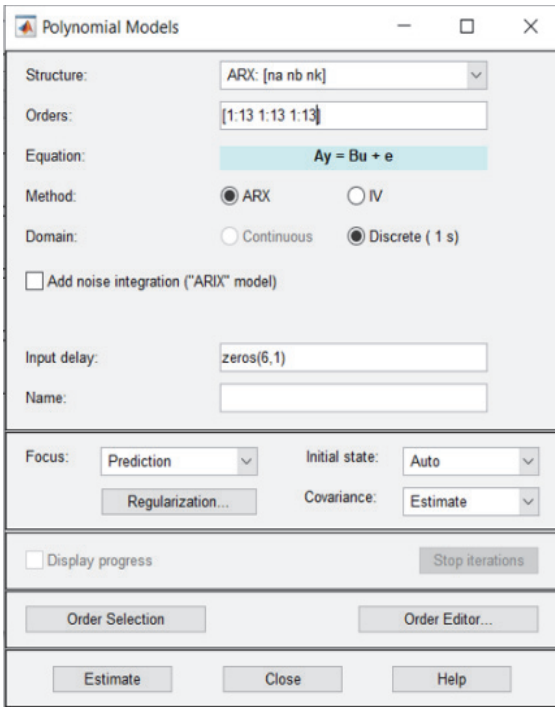
A total of six input quantities were distinguished in the data, i.e.: Ex1 - height, Ex2 - weight, Ex3 - number of dishes, Ex4 - age, Ex5 - gender, diet and five output quantities, i.e.: Y1 - breakfast, Y2 - lunch, Y3 - lunch, Y4 - afternoon snack, Y5 - dinner.

4. Identification of the system called Generating breakfast

The first model obtained as a result of parametric identification using the ARX method was the breakfast generation model (U-Y1). It was a Multi Input Single Output (MISO) model containing six inputs marked: Ex1 - height, Ex2 - weight, Ex3 - number of dishes, Ex4 - age, Ex5 - gender, Ex6 - diet and one breakfast menu output (Y1).

So, the first step after entering data into Workspace was to generate the best ARX model in such a way that its compliance with real data was as high as possible. The "Polynomial models" function (Figure 6) was used in the identification, i.e. the function of generating parametric models of systems [5-6].

As a result of the identification, the ARX model of the breakfast generation subsystem was obtained under the name U-Y1 with an accuracy of 94.42% (Figure 7).



Polynomial Models

Structure: ARX: [na nb nk]

Orders: [1:13 1:13 1:13]

Equation: Ay = Bu + e

Method: ● ARX ○ IV

Domain: ○ Continuous ● Discrete (1 s)

□ Add noise integration ("ARIX" model)

Input delay: zeros(6,1)

Name:

Focus: Prediction

Initial state: Auto

Covariance: Estimate

□ Display progress

Figure 6. View of using Polynomial Models for model Y1. Source: Own study using System Identification Toolbox in MATLAB and Simulink [6, 12].

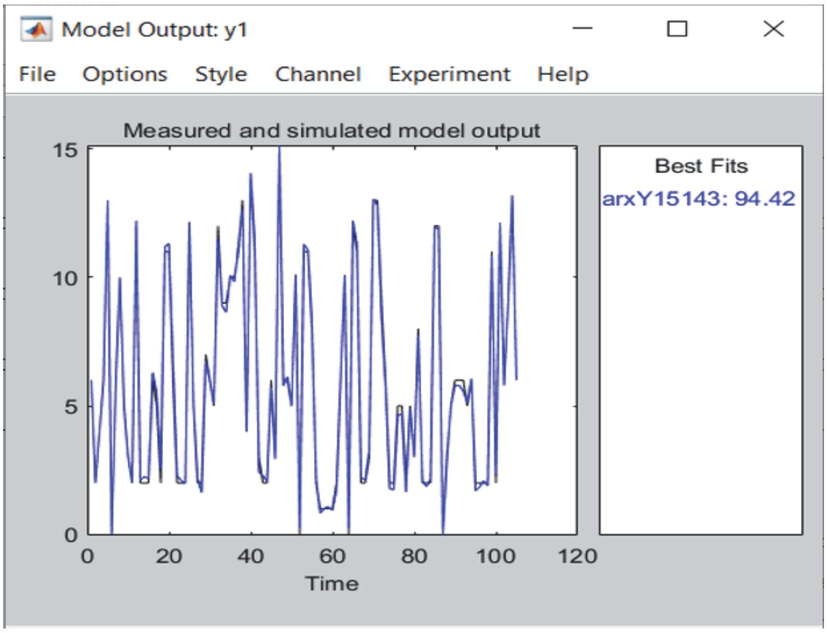


Figure 7. The course of output from the U-Y1 model and subsystem. Designations: X axis – time (consecutive numbers of data pairs used in identification) Y1 axis – dish identifiers. Source: Own elaboration using System Identification Toolbox [6, 12].

It can be seen that the course of the output from the model in relation to the output from the system is matched to the real system to the extent of 94.42%. In a similar way, the remaining subsystems in the Diet generation system were identified, i.e. they are marked as: U-Y2 - lunch generation model, U-Y3 - lunch generation model, U-Y4 - afternoon tea generation model, U-Y5 - supper generation model.

In the same way as in the case of the identification of the Breakfast Generation subsystem, the identification of the other subsystems in the Diet Generation system was carried out using the "Polynomial Models" function in the identification. As a result of identification, ARX models of subsystems marked as: U-Y2, U-Y3, U-Y4, U-Y5 were obtained at the accuracy level of: 73.58%, 83.55%, 73.56%, 73.46%.

Assessment of the quality of the model in relation to the RDN system most often requires determining the absolute and relative errors, MSE error, MAPE, the R^2 determination index as well as the effectiveness of the RDN system models, the efficiency and robustness of the system and the RDN system model, as well as examining the sensitivity of the RDN system model.

When assessing the quality of the model to the system obtained as a result of identification, various ranges of fit of the model to the system are most often established, and the lower the error, e.g. MSE, the higher the fit, with the obtained fit above 70.00% plus, and even 80.00% plus or 90.00% plus as in the case under consideration is quite high. If there is a need to increase the matching, which was unnecessary in the case under consideration, for example, an evolutionary algorithm can be used for this purpose.

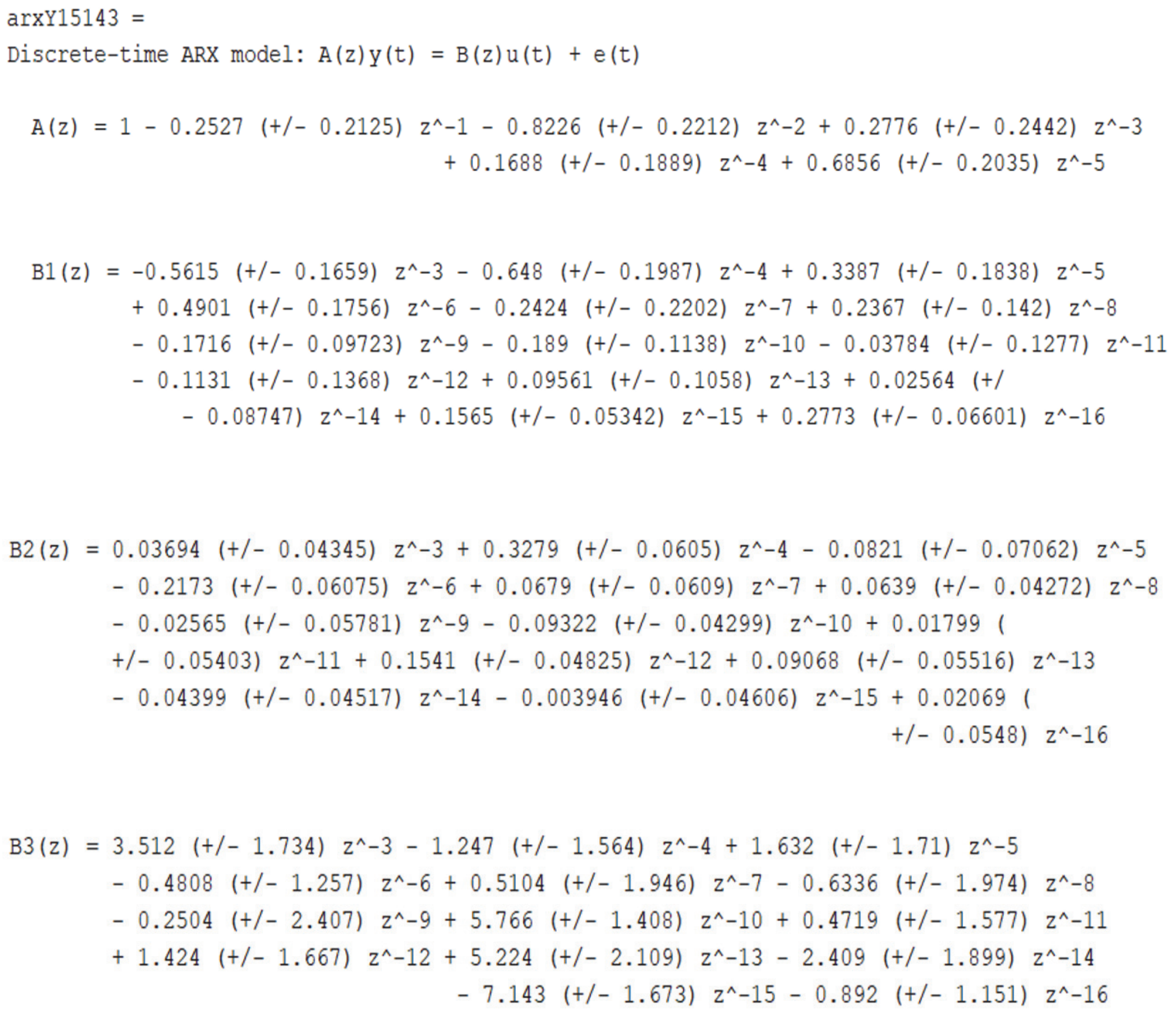
5. Parametric models of the web application management system

As a result of parametric identification of the ARX using the System Identification Toolbox in the MATLAB and Simulink environment, appropriate models of the system's subsystems were obtained, which is the web application for power plant management. The obtained mathematical models of these subsystems can be further used in simulation modeling of the web application. The simulation modeling of the power plant management system was carried out using the Simulink tool, which enables graphical design of system models in the form of block diagrams for the purposes of simulation, comparative and sensitivity testing.

The principle of designing a simulation model is to use the ready-made functions available in the Simulink library, which can be implemented in the form of appropriate blocks, to which appropriate parameter values should be assigned, or in the absence of them - building own blocks with assigned functions [6]. The construction of the simulation system model of the web application was based on the mathematical model of individual subsystems obtained as a result of parametric identification using the ARX method. A total of about 300 subsystems were obtained, and each of the subsystems was composed of subsequent subsystems. And so the following subsystems were obtained in the web application system: registration and logging in, classes, my classes, passes, basket, my orders, diet. Detailed simulation modeling carried out for the diet generation subsystem, which as a system consists of the following subsystems: breakfast (U-Y1), lunch (U-Y2), lunch (U-Y3), afternoon snack (U-Y4), dinner (U-Y5). The simulation model for the Breakfast Generation subsystem will be discussed in detail here.

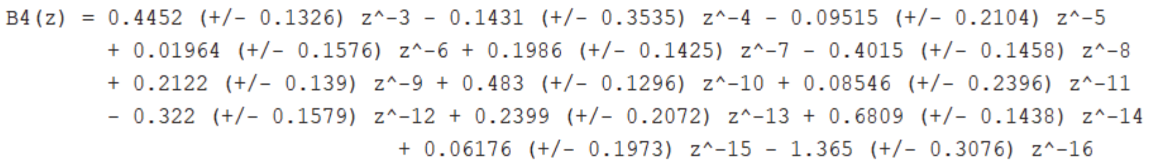
5.1. Simulation model of the Breakfast Generating subsystem

As a result of the identification of the subsystem Generating breakfast under the name U-Y1, the ARX model was obtained in the form:

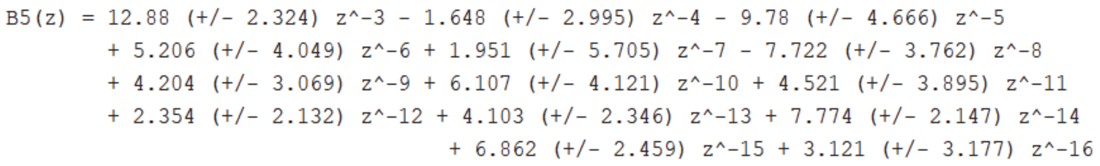


```
arxY15143 =
Discrete-time ARX model: A(z)y(t) = B(z)u(t) + e(t)

  A(z) = 1 - 0.2527 (+/- 0.2125) z^-1 - 0.8226 (+/- 0.2212) z^-2 + 0.2776 (+/- 0.2442) z^-3
                                 + 0.1688 (+/- 0.1889) z^-4 + 0.6856 (+/- 0.2035) z^-5


  B1(z) = -0.5615 (+/- 0.1659) z^-3 - 0.648 (+/- 0.1987) z^-4 + 0.3387 (+/- 0.1838) z^-5
          + 0.4901 (+/- 0.1756) z^-6 - 0.2424 (+/- 0.2202) z^-7 + 0.2367 (+/- 0.142) z^-8
          - 0.1716 (+/- 0.09723) z^-9 - 0.189 (+/- 0.1138) z^-10 - 0.03784 (+/- 0.1277) z^-11
          - 0.1131 (+/- 0.1368) z^-12 + 0.09561 (+/- 0.1058) z^-13 + 0.02564 (+/
              - 0.08747) z^-14 + 0.1565 (+/- 0.05342) z^-15 + 0.2773 (+/- 0.06601) z^-16



B2(z) = 0.03694 (+/- 0.04345) z^-3 + 0.3279 (+/- 0.0605) z^-4 - 0.0821 (+/- 0.07062) z^-5
        - 0.2173 (+/- 0.06075) z^-6 + 0.0679 (+/- 0.0609) z^-7 + 0.0639 (+/- 0.04272) z^-8
        - 0.02565 (+/- 0.05781) z^-9 - 0.09322 (+/- 0.04299) z^-10 + 0.01799 (
        +/- 0.05403) z^-11 + 0.1541 (+/- 0.04825) z^-12 + 0.09068 (+/- 0.05516) z^-13
        - 0.04399 (+/- 0.04517) z^-14 - 0.003946 (+/- 0.04606) z^-15 + 0.02069 (
                                                                  +/- 0.0548) z^-16


B3(z) = 3.512 (+/- 1.734) z^-3 - 1.247 (+/- 1.564) z^-4 + 1.632 (+/- 1.71) z^-5
        - 0.4808 (+/- 1.257) z^-6 + 0.5104 (+/- 1.946) z^-7 - 0.6336 (+/- 1.974) z^-8
        - 0.2504 (+/- 2.407) z^-9 + 5.766 (+/- 1.408) z^-10 + 0.4719 (+/- 1.577) z^-11
        + 1.424 (+/- 1.667) z^-12 + 5.224 (+/- 2.109) z^-13 - 2.409 (+/- 1.899) z^-14
                                    - 7.143 (+/- 1.673) z^-15 - 0.892 (+/- 1.151) z^-16
```

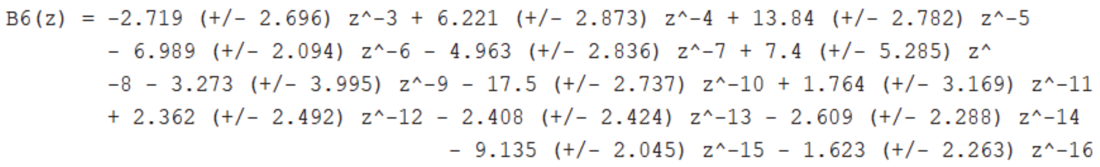
(2)



```
B4(z) = 0.4452 (+/- 0.1326) z^-3 - 0.1431 (+/- 0.3535) z^-4 - 0.09515 (+/- 0.2104) z^-5
        + 0.01964 (+/- 0.1576) z^-6 + 0.1986 (+/- 0.1425) z^-7 - 0.4015 (+/- 0.1458) z^-8
        + 0.2122 (+/- 0.139) z^-9 + 0.483 (+/- 0.1296) z^-10 + 0.08546 (+/- 0.2396) z^-11
        - 0.322 (+/- 0.1579) z^-12 + 0.2399 (+/- 0.2072) z^-13 + 0.6809 (+/- 0.1438) z^-14
                              + 0.06176 (+/- 0.1973) z^-15 - 1.365 (+/- 0.3076) z^-16
```



```
B5(z) = 12.88 (+/- 2.324) z^-3 - 1.648 (+/- 2.995) z^-4 - 9.78 (+/- 4.666) z^-5
        + 5.206 (+/- 4.049) z^-6 + 1.951 (+/- 5.705) z^-7 - 7.722 (+/- 3.762) z^-8
        + 4.204 (+/- 3.069) z^-9 + 6.107 (+/- 4.121) z^-10 + 4.521 (+/- 3.895) z^-11
        + 2.354 (+/- 2.132) z^-12 + 4.103 (+/- 2.346) z^-13 + 7.774 (+/- 2.147) z^-14
                                  + 6.862 (+/- 2.459) z^-15 + 3.121 (+/- 3.177) z^-16
```



```
B6(z) = -2.719 (+/- 2.696) z^-3 + 6.221 (+/- 2.873) z^-4 + 13.84 (+/- 2.782) z^-5
        - 6.989 (+/- 2.094) z^-6 - 4.963 (+/- 2.836) z^-7 + 7.4 (+/- 5.285) z^
        -8 - 3.273 (+/- 3.995) z^-9 - 17.5 (+/- 2.737) z^-10 + 1.764 (+/- 3.169) z^-11
        + 2.362 (+/- 2.492) z^-12 - 2.408 (+/- 2.424) z^-13 - 2.609 (+/- 2.288) z^-14
                                  - 9.135 (+/- 2.045) z^-15 - 1.623 (+/- 2.263) z^-16
```

where:

A(z) – polynomial associated with the output y1(t),

B1(z), B2(z), …, B6(z) – polynomials related to inputs u1(t), u2(t), …, u6(t),

z – time shift operator (as numbers of pairs used in identification),

t – number of input-output pair used in identification.

Thus, the parametric model arxY15143 consists of 83 terms relating the parameters na, nb, nk to the appropriate input and output quantities, i.e. six B(z) polynomials and one A(z) polynomial. In order to obtain a simulation model in Simulink using the parametric model of Generating Breakfast, it was first necessary to enter the appropriate matrices containing the values of individual inputs and outputs into Workspace, and then select appropriate blocks representing the appropriate model functions for the model (2). These were blocks of the type: From Workspace, Gain, Transport Delay, Sum, To Workspace, Scope, Demux, etc.

The next step was to design a simulation model suitable for use in simulation, comparative and sensitivity studies. First, the subsystem was launched, which was to simulate the polynomial A(z) (Figure 8), with the block diagrams of individual subsystems marked as: A1(z-1) describing Subsystem Y1 connected with z-1, A1(z-2) describing Subsystem Y1 connected with z-2, A1(z-3) describing Subsystem Y1 connected with z-3 , A1(z-4) describing Subsystem

Y1 connected with z-4, A1(z-5) describing Subsystem Y1 connected with z-5. One of them as an example of a simulation is shown in Figure 9.

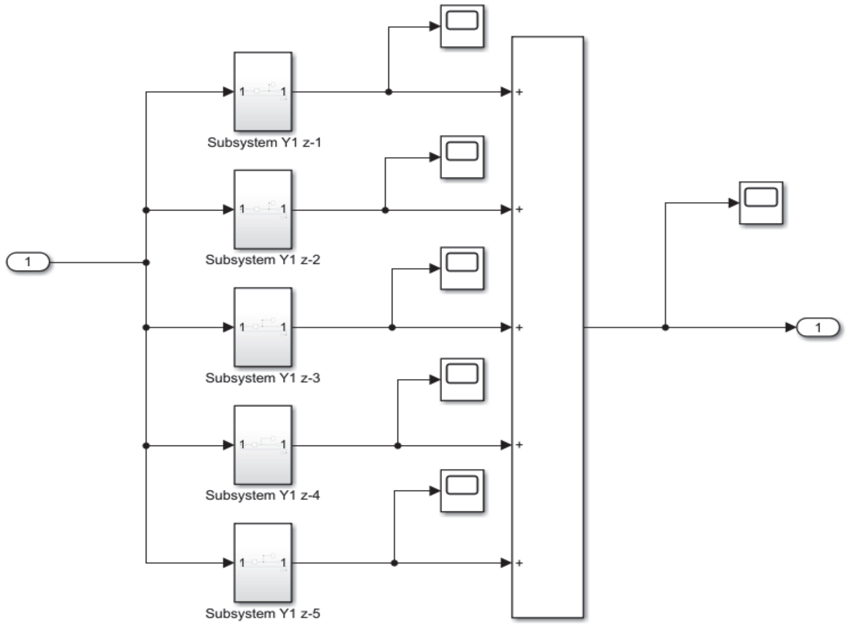


Figure 8. A subsystem for simulating the operation of a real system model under the name U-Y1. Source: Own study using Simulink [6, 12].

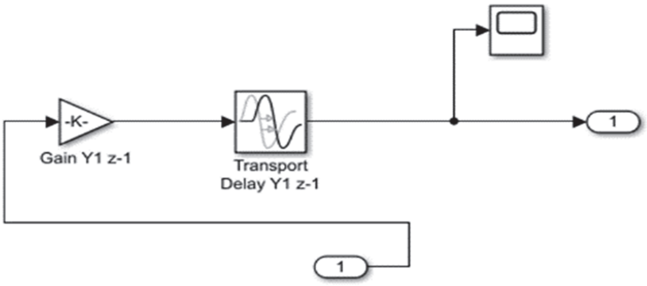


Figure 9. U-Y1 simulation support subsystem. Source: Own elaboration using Simulink [6, 12].

The parameters of the model (2) in terms of the A(z) polynomial were used to design the subsystem implementing the A(z) polynomial in the Breakfast Generation system. In this way, five subsystems composed of proportional terms (coefficients of the polynomial A(z) designed with Gain blocks) and delay terms (time shift elements appearing in the polynomial A(z) designed with Transport Delay blocks) were created, which were connected to the adder block, thus giving the total effect of the output Y2(t) on the input to this particular subsystem (feedback). In order to design the subsystems implementing polynomials B1(z), B2(z), B3(z), B4(z), B5(z), B6(z) in the Breakfast Generation system, the relevant model parameters (2) were used in a similar way.

In this way, six subsystems of type B(z) were obtained, each such subsystem B(z) being further composed of proportional terms (coefficients of the appropriate polynomial B(z) designed using Gain blocks) and delay terms (time shift elements appearing in the appropriate polynomial B(z) designed using Transport Delay blocks), which were connected to the adder, thus giving a total impact on the output Y1(t), while the input to this subsystem was taken from the block From Workspace after separating the appropriate entrances on the Demux block.

An example of a block for the U1 input implementing the B1(z) polynomial is shown in Figure 10, with one of its individual subsystems on the block marked respectively in Figure 11. On the other hand, Figure 12 shows the entire subsystem simulating the real system in terms of generating the diet of the U-Y1 system, and Figure 13 shows the result of the system error simulation (behavior of the output from the system model in relation to the real system), whose course ranged from (0 ÷ 15)%.

6. Analysis of the web application supporting gym management

Control theory deals with analysis and modeling of arrangements, systems, objects and processes of various nature (physical, biological, genetic, economic, social, etc.) [1, 4, 21, 24-25, 28]. The models of the Diet Generation system obtained as a result of parametric identification are most often modeled with the use of Simulink in order to obtain a simulation model. In terms of control and systems theory, such a model is an object with specific properties [8, 21, 29]. The resulting model was the ARX discrete model. The discrete model can be further transformed into a continuous model, and then, for example, into a continuous model in the state space and, based on this model, the state of the system's internal organization (state and changes of matrix **A**) and the level of system control (state and changes of matrix **B**) can be examined on the basis of this model. Models designed in this way make it possible to control a given object or process in such a way that it behaves in the desired way [1, 4, 18, 20-21].

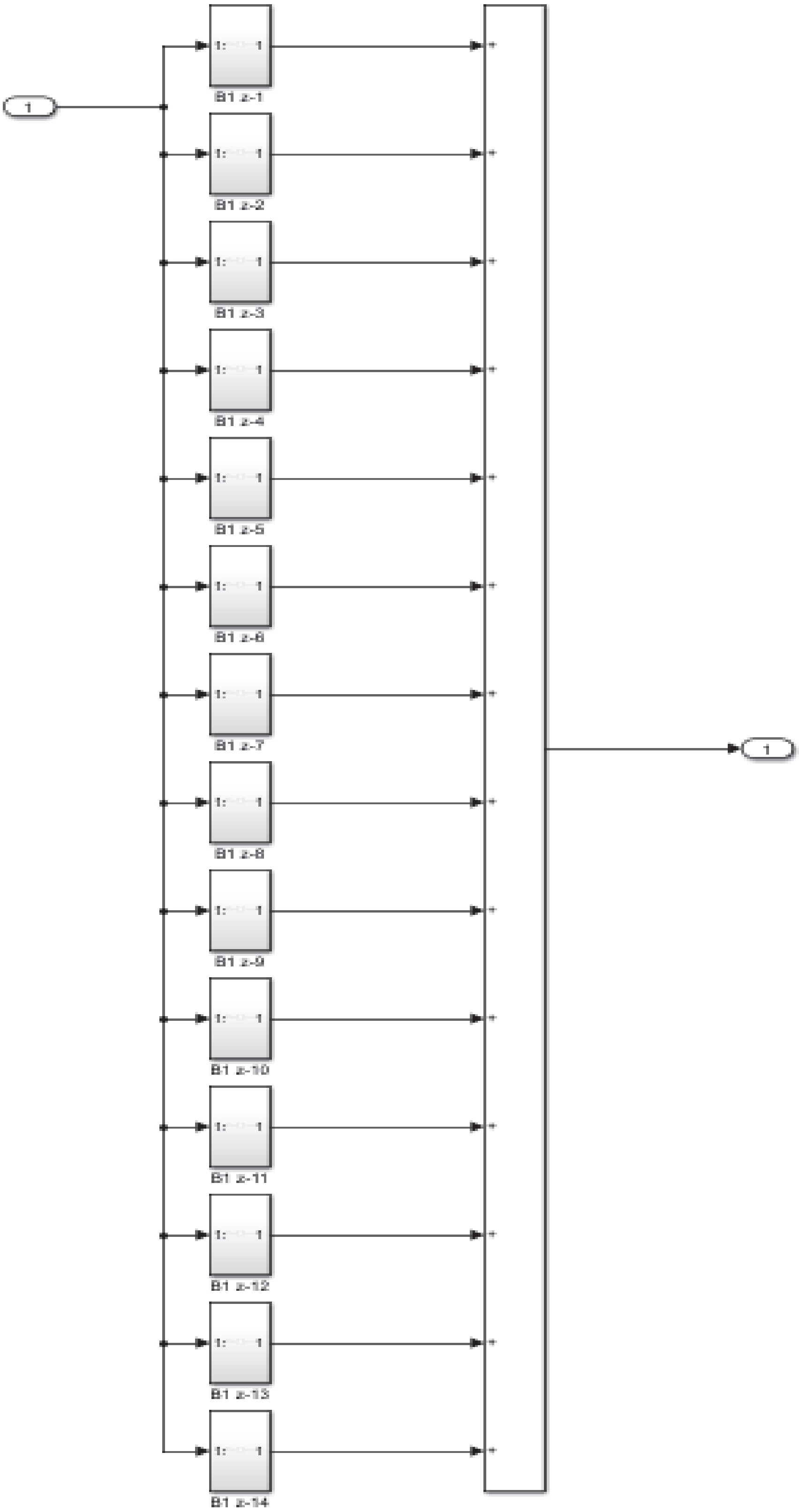


Figure 10. A subsystem realizing the B1(z) polynomial for the U-Y1 system. Source: Own study using Simulink [6, 12].

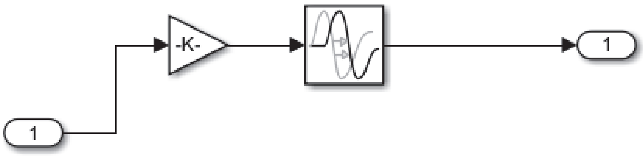


Figure 11. The subsystem realizing the term of the B1(z) polynomial appearing as a coefficient at the z-1 delay for the U-Y1 system. Source: Own study using Simulink [6, 12].

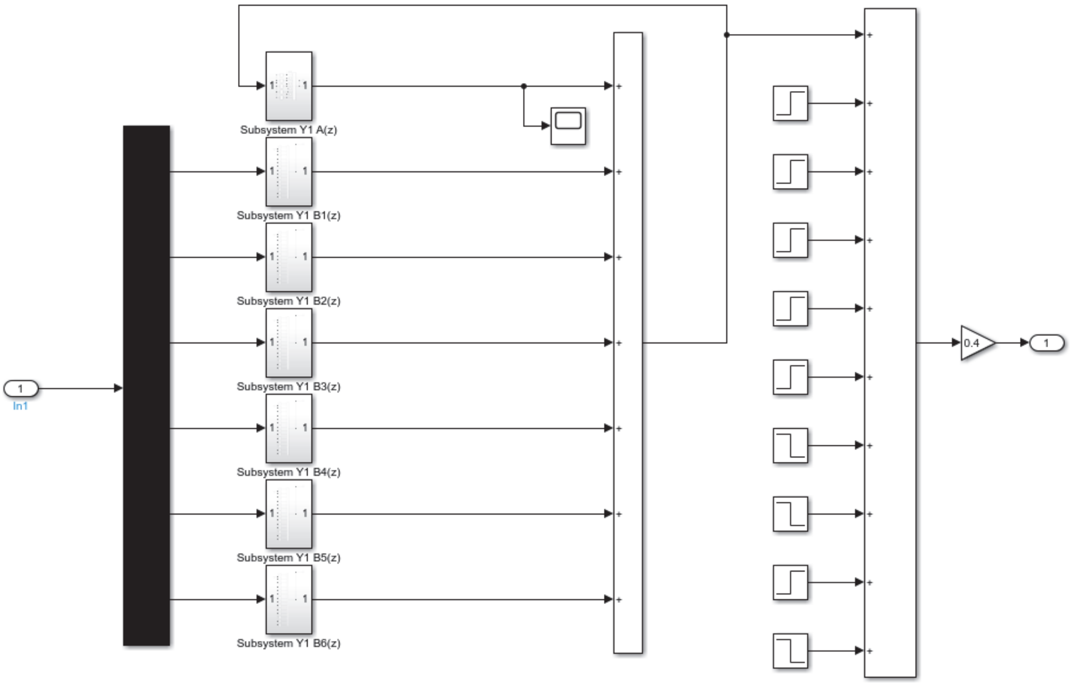


Figure 12. A subsystem simulating a real system called Generating a diet for the U-Y1 system. Source: Own study using Simulink [6, 12].

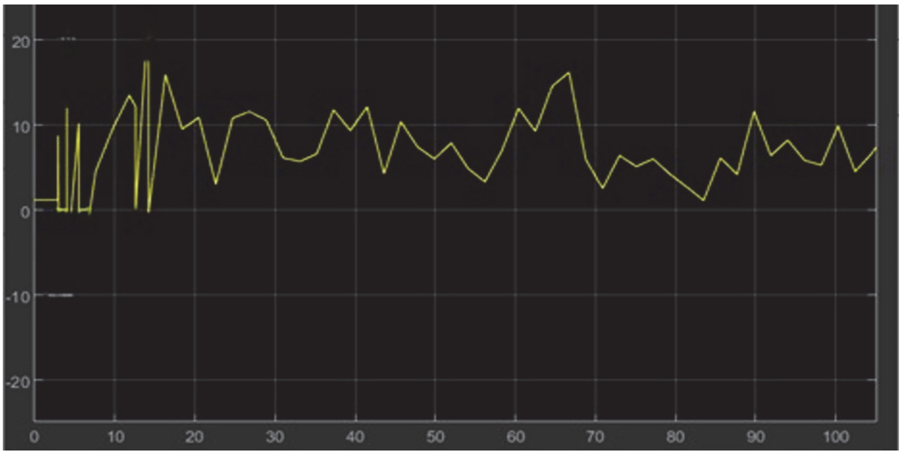


Figure 13. The results of the simulation for U-Y1. Designations: X axis - time (consecutive numbers of data pairs used in the simulation), Y axis - model error values in relation to real data. Source: Own study using Simulink [6, 12].

6.1. System analysis Generating a diet

For simulation, comparative and sensitivity testing purposes, based on the obtained simulation models of the subsystems of the Diet Generation system, a comprehensive simulation model was built as shown in Figure 14. This model consists of 5 subsystems, each of which processes the input data U to the appropriate outputs Y1 - Y5.

Each of the subsystems is a complex simulation model based on a mathematical model described by equations (2). With such a model at our disposal, we can fully simulate the operation of the pn system. Generating a diet which is one of the subsystems of the gym management system. The examples of the obtained test results in the form of the course of the relevant errors between the model and system signals are presented in Table 2. As a result of the analysis of absolute and relative errors, it is possible to indicate possible directions for further improvement of the quality of the simulation model. The determined absolute errors fall within the appropriate limits for the subsystem: Breakfast generation: (0÷2.5)%, Second breakfast generation: (0÷2.0)%, Lunch generation: (3.0÷3.5)%, Tea generation: (4.0÷4.5)%, Dinner generation: (6.0÷8.0)% and relative errors: Breakfast generation: (-5.0)÷0.0)%, Lunch generation: (-1.5÷1.0)%, Lunch generation: (-0.5÷0.5)%, Tea generation: (-5.0÷3.0)%, Dinner generation: (-0.5÷3.0)%.

7. Conclusions and directions for further research

The article presents the results of the analysis of a web application supporting gym management using control theory and systems on the example of a gym management system. The conducted research experiment consisted of i.a. on measuring selected input and output quantities for individual functions, and then, on their basis, identifying individual subsystems in the web application system as an executing program. The identification was carried out in the MATLAB and Simulink environment using the System Identification Toolbox, and the model verification as well as simulation and comparative tests were carried out using Simulink.

It was also shown that it is possible to identify the system of the web application supporting the management of the power plant in terms of the theory of systems control and the possibility of its implementation in the MATLAB and Simulink environments, which required, among others: designing a research experiment enabling the measurement of input and output values of individual subsystems operating within the web application during its operation, selection of the appropriate structure of the identification model and using it, the identification of the subsystem validation of the obtained model using Simulink for the purposes of verification, simulation and comparative research, analysis of the Generating breakfast system model as a subsystem of the web application called Gym, verification, simulation and comparative research to examine the quality of functioning of the system model called Generating a diet as a subsystem of the web application management system, etc.

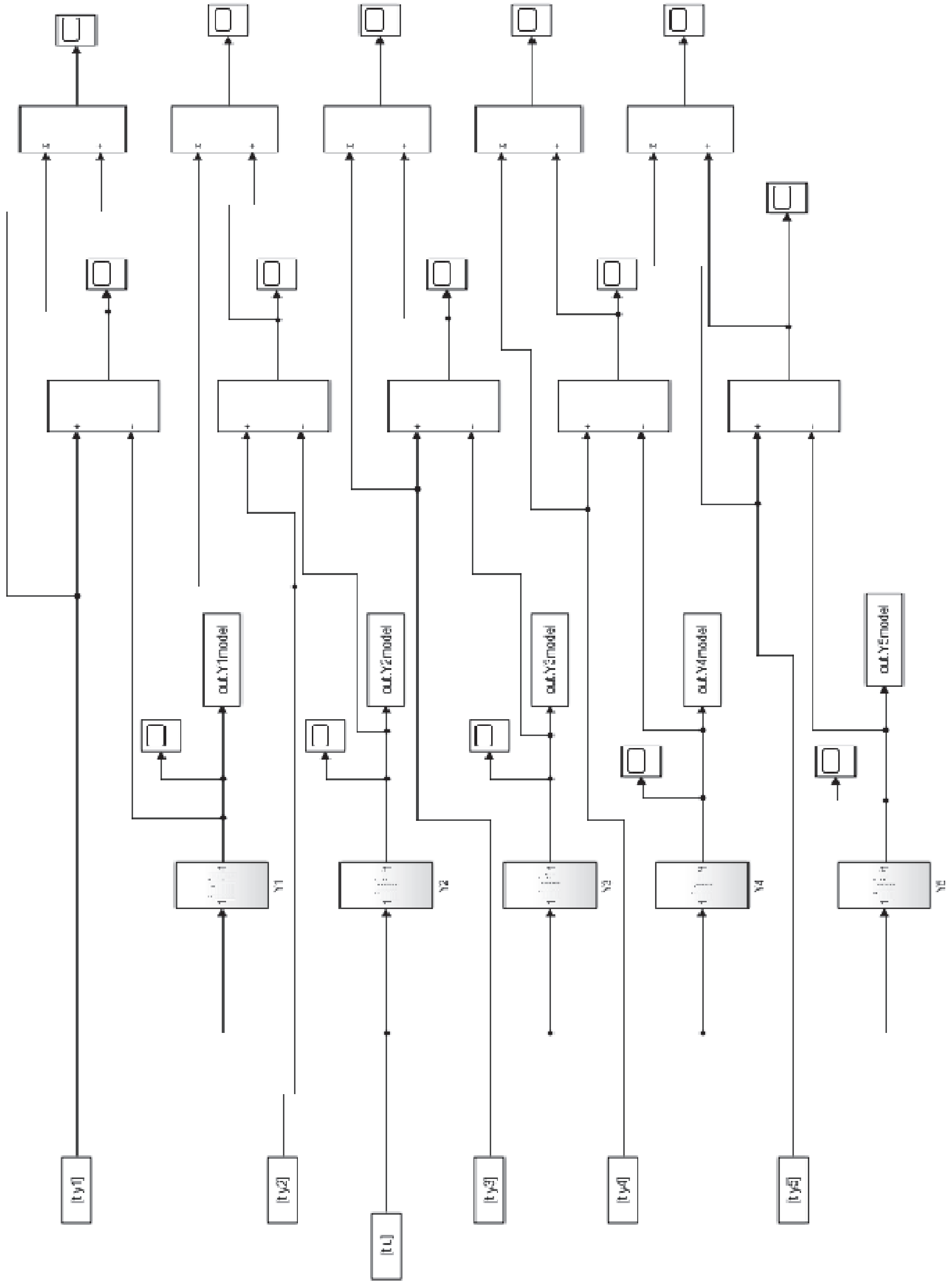
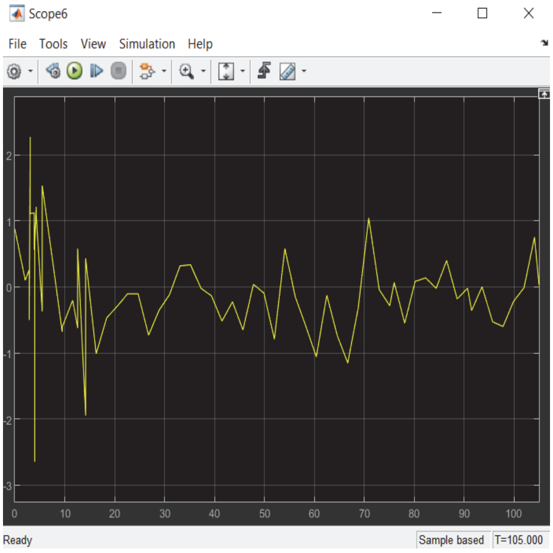
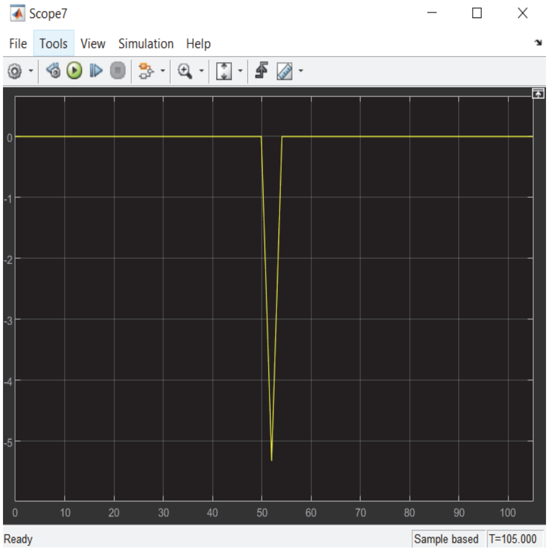


Figure 14. System model for generating a diet on a gym management portal. Source: Own study using Simulink [6, 12].

Table 1. Courses of errors in the subsystems of the Diet Generation system. Source: Own study [6, 12].

name	Absolute error cource	Relative error course
U-Y1	 Scope6 File Tools View Simulation Help 2 1 0 -1 -2 -3 0 10 20 30 40 50 60 70 80 90 100 Ready Sample based T=105.000	 Scope7 File Tools View Simulation Help 0 -1 -2 -3 -4 -5 0 10 20 30 40 50 60 70 80 90 100 Ready Sample based T=105.000

The error curves shown in Table 1 indicate that they were relatively low, around ∓3% for absolute error course and approximately -5% for relative error course.

References

1. Bubnicki Z.: Teoria i algorytmy sterowania (in Polish), Wydawnictwo Naukowe PWN, Warsaw, 2005.
2. Drwal G., Grzymkowski R., Kapusta A., Słota D.: Mathematica. Programowanie i zastosowanie (in Polish), Wydawnictwo Pracowni Komputerowej Jacka Skalmierskiego, pages 197, 1995.
3. Hayati Z., Shafieirad M., [i inni]: Parameter estimation of MIMO two-dimensional ARMAX model based on IGLS method, Control and Cybernetics, Vol. 50, No 3, pp. 303-322, 2021.
4. Kaczorek T.: Teoria sterowania i systemów (in Polish), WN PWN, Warsaw, 1993.
5. Księżopolski A., Projekt i implementacja portalu wspomagającego zarządzanie siłownią (in Polish), Praca inżynierska pod kierunkiem dr inż. Mirosława Barańskiego napisana w Instytucie Informatyki na Wydziale Nauk Ścisłych UPH w Siedlcach, Siedlce, pages 53, 2020.

6. Księżopolski A., Analiza aplikacji webowej wspomagającej zarządzanie siłownią w ujęciu teorii sterowania i systemów (in Polish), praca magisterska pod kierunkiem dr hab. inż. Jerzego Tchórzewskiego napisana w Instytucie Informatyki na Wydziale Nauk Ścisłych UPH w Siedlcach, Siedlce, pages 98, 2021.

7. Ljung L., Singh R., Chen T.: Regularization Features in the System Identification Toolbox, IFAC-Papers OnLine, Vol. 48, Issue 28, pp. 745–750, 2015.

8. Marlęga R.: Identyfikacja i metaidentyfikacja systemu zarządzania Towarową Giełdą Energii Elektrycznej (in Polish), rozprawa doktorska pod kierunkiem dr hab. inż. Jerzego Tchórzewskiego oraz dr hab. Arkadiusza Jurczuka, Wydział Inżynierii Zarządzania, Politechnika Białostocka, Białystok, pages 238, 2022.

9. Marlęga R., Tchórzewski J.: Hourly identification and simulation of the TGE S.A. Day-Ahead Market system, Control and Cybernetics, Vol. 51, No. 4, pp. 523-555, 2022.

10. Marlęga R.: A methodology of identification and metaidentification research on the example of Day Ahead Market System, Studia Informatica. Systems and Information Technology, Vol. 2(27), pp. 109-137, 2022.

11. Mrozek B., Mrozek Z., Matlab i Simulink. Podręcznik użytkownika (in Polish), Helion, Warszawa, pages 280, 2017.

12. MATLAB & Simulink, Simulink Getting Started Guide, MathWorks, USA, 2002-2022.

13. Missbach M., Anderson G.: SAP w 24 godziny (in Polish). Wydanie V, Helion, Warszawa, pages 432, 2016.

14. Mynarski S.: Analiza danych. Makromechanizmy (in Polish), Wydawnictwo AE, Kraków, pages 90, 2000.

15. Rivera D. E., Guzmán J.L., Berenguel M., S. Dormido: Understanding closed-loop identification with ITCLI, IFAC - PapersOnLine, Vol. 48, Issue 28, pp. 739-744, 2015.

16. Rutkowski L., Metody i techniki sztucznej inteligencji (in Polish), Wydawnictwo Naukowe PWN, Warsaw, pages 452, 2023.

17. Sradomski W., MATLAB. Praktyczny podręcznik modelowania (in Polish), Helion, Warsaw, pages 280, 2015.

18. Tadeusiewicz R.: Badanie właściwości układów współpracujących ze stochastycznie zmiennym środowiskiem (in Polish), Postępy Cybernetyki, Nr. 4 (2), pp. 21-29, 1979.

19. Tchórzewski J.: Metody sztucznej inteligencji i informatyki kwantowej (in Polish), WN UPH w Siedlcach, Siedlce, pages 343, 2021.

20. Tchórzewski J.: The Security of Polish Electrical Power System Development, [in:] Different Faces of Security from Knowledge to Management, [ed.] Raczkowski K., [i in.], Institute for Security and Development Policy, Stockholm-Nacka, Sweden, pp. 237-263, 2020.
21. Tchórzewski J.: Rozwój systemu elektroenergetycznego w ujęciu teorii sterowania i systemów (in Polish), OW Politechniki Wrocławskiej, Wrocław, pages 190, 2013.
22. Tchórzewski J.: Identification of Electric Power System from Point of View of Development, Polish Journal of Environmental Studies, Vol. 18, No. 4B, pp.191-194, 2009.
23. Tchórzewski J.: The development process of electrical power system from the point of view of efficiency. Poznan University of Technology Academic Journals, No 60, Electrical Engineering, pp. 159-171, 2009.
24. Tchórzewski J.: Cybernetyka życia i rozwoju systemów (in Polish), Monografie nr 22, WSRP w Siedlcach, Siedlce, pages 408, 1992.
25. Tchórzewski J.: Inżynieria rozwoju systemów (in Polish), Monografie nr 18, WSRP w Siedlcach, Siedlce, pages 279, 1990.
26. Trusz M., Tserakh U.: GARCH(1,1) models with stable residuals, Studia Informatica. Systems and Information Technology, Vol. 22, No. 1-2, pp. 47-57, 2018.
27. Wesołowski Z.: Estimating Reliability Characteristics for Homogeneous and Heterogeneous Systems, Studia Informatica. Systems and Information Technology, Vol. 1-2(18), pp. 47-59, 2014.
28. Ziętkiewicz J.: Identyfikacja obiektów sterowania (in Polish), Wydawnictwo Politechniki Poznańskiej, Poznań, pages 64, 2018,
29. Zimmer A., Englot A.: Identyfikacja obiektów i sygnałów. Teoria i praktyka dla użytkowników MATLABA (in Polish). Politechnika Krakowska. Kraków, pages 239, 2005.