

УДК 004.62.

**Б. І. Мокін<sup>1</sup>**  
**О. В. Мазурук<sup>1</sup>**  
**Н. В. Собчук<sup>1</sup>**  
**Д. О. Шалагай<sup>2</sup>**

## **АВТОРЕГРЕСІЙНІ МОДЕЛІ ВІДНОВЛЕННЯ ТА РОЗВИТКУ ЗРУЙНОВАНОЇ ВІЙНОЮ ЕЛЕКТРОЕНЕРГЕТИКИ УКРАЇНИ**

<sup>1</sup>Вінницький національний технічний університет;

<sup>2</sup>ТОВ «Промавтоматика Вінниця»

*Запропоновано спосіб корекції плану відбудови та розвитку електроенергетики України, розробленого в роботі «Національна рада з відновлення України. Проект Плану відновлення України. Розділ «Енергетична безпека», 2022», до умов, що виникли після бомбардувань об'єктів української електроенергетичної структури російськими агресорами. Синтезовано первинні детерміновані та скореговані математичні моделі, які враховують стан електроенергетики України, що мав місце в дні написання цієї роботи. Оскільки ідея використовувати замість скорегованих математичних моделей авторегресійні моделі була лише озвучена в роботі «Синтез математичних моделей процесу відновлення та розвитку електроенергетики України, наближених до реалій воєнного сьогодення» [1] і записана лише в загальній постановці, то через рік після написання цієї роботи, що пройшов у вирі війни України з російськими окупантами, українська електроенергетична структура зазнала додаткових руйнувань, тож виникла необхідність в синтезі авторегресійних моделей в завершеній формі, готовій для використання інженерами-електриками в проєктній роботі. Тож в цій роботі синтезовано авторегресійні моделі процесів відновлення та розвитку електроенергетики України, адаптовані до реалій воєнного сьогодення, і здійснено обґрунтування вибору класу авторегресій, їхнього порядку та діапазонів генерації імпульсів «білого шуму», якими деформуються детерміновані базові математичні моделі.*

**Ключові слова:** електроенергетика України, план відновлення, воєнне сьогодення, детерміновані моделі процесів, внесення стохастичності, білий шум, авторегресійні моделі.

### **Постановка задачі та вихідні передумови**

В нашій роботі [1] запропоновано спосіб корекції плану відбудови та розвитку електроенергетики України, розробленого в роботі [2], до умов, що виникли після бомбардувань об'єктів української електроенергетичної структури російськими агресорами [3], і синтезовані первинні детерміновані та скореговані математичні моделі, які враховують стан електроенергетики України, що мав місце в дні написання цієї роботи. Але ідея використовувати замість скорегованих математичних моделей авторегресійні моделі була лише озвучена в роботі [1] і записана лише в загальній постановці.

Через рік після написання роботи [1], що пройшов у вирі війни України з російськими окупантами, українська електроенергетична структура зазнала додаткових руйнувань [4], тож виникла необхідність в синтезі авторегресійних моделей в завершеній формі, готовій для використання інженерами-електриками в проєктній роботі. Реалізація цього задуму і здійснюється у цій роботі.

А тому як вихідні передумови для написання цієї роботи використано, по-перше, детерміновані математичні моделі, що синтезовані нами в роботі [1] на основі даних, поданих в роботі [2], по-друге, спосіб корекції детермінованих математичних моделей для врахування наслідків руйнувань української електроенергетичної структури, що викладений нами в роботі [1], а по-третє, Python-

програму синтезу авторегресійних моделей, подану нами в одному з навчальних посібників.

### Розв'язання поставленої задачі

Розпочнемо розв'язання поставленої задачі з того, що перенесемо з роботи [1] детерміновані математичні моделі процесів відновлення та розвитку електроенергетики України, які синтезовані авторами за даними роботи [2], та їхні графічні зображення.

Отже маємо:

$$y_0(t) = 0,000362103174603175 \cdot t^2 + 0,0470039682539683 \cdot t + 23,7; \quad (1)$$

$$y_1(t) = -0,000160383597883598 \cdot t^2 + 0,208062169312169 \cdot t + 23,7; \quad (2)$$

$$y_2(t) = 0,000558862433862434 \cdot t^2 + 0,241494708994709 \cdot t + 24,5; \quad (3)$$

$$w_0(t) = -0,00138888888888889 \cdot t^2 + 0,0470039682539683 \cdot t + 23,7; \quad (4)$$

$$w_1(t) = -0,003125 \cdot t^2 + 1,8375 \cdot t + 143,0; \quad (5)$$

$$w_2(t) = 0,00886243386243386 \cdot t^2 + 1,644708994709 \cdot t + 146,0. \quad (6)$$

Графічна інтерпретація цих розв'язків показана на рис. 1.

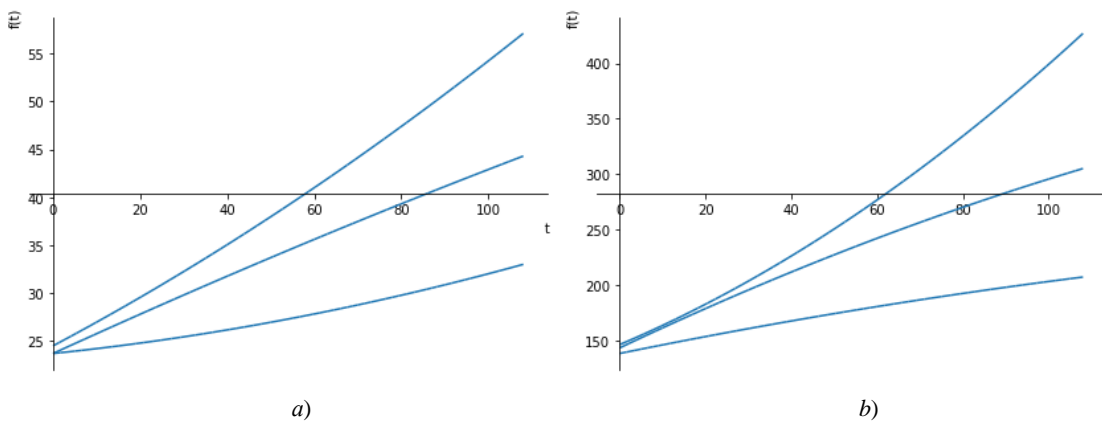


Рис.1. Графіки детермінованих математичних моделей прогнозного зростання: *a* — максимуму навантаження споживачів або, що те саме, зростання сумарного максимуму потужності генерації електроенергії її джерелами (в ГВт); *b* — попиту на електричну енергію в Україні ( в мільярдах кВт-годин) за сценаріями: максимуму економічної ефективності (СМЕЕ), референтного сценарію (РС) та сценарію кліматично нейтральної економіки (СКНЕ) на 2023, 2030 та 2050 роки. На графіках відлік часу, вираженого в річних кварталах, починається з 2023 року

А далі звернемо увагу на твердження, з нашої роботи [1], з якого випливає, що в планах відновлення і розвитку енергетики України на період до 2050 року, розроблених в роботі [2], враховано те, що (цитата): «станом на 2023 рік 39 % потужностей генерації електроенергії в Україні електроенергію не генерували. Але, стартувавши від 61 % потужностей генерації електроенергії в Україні, які станом на 2023 рік генерували електроенергію, в подальшому автори планів процесу відновлення та розвитку енергетики України заклали в ці плани детермінований характер цього процесу, який не відповідає в дійсності його стохастичному характеру, обумовленому тим, що руйнування енергетичної інфраструктури в умовах воєнного сьогодення продовжуватимуться і після 2023 року, як це показано в роботах Ольги Буславець, одна з яких, а саме робота [3], подана в списку використаної літератури. Більше того, стохастичний і водночас нестационарний характер процесу відновлення енергетики України матиме місце і після закінчення воєнних дій протягом досить тривалого часу, оскільки ті об'єкти, що частково зруйновані і під час війни підтримувались в режимі, принаймні, хоча б часткової генерації і передачі електроенергії, необхідно буде вивести з експлуатації і замість них увести побудовані заново, що в умовах дефіциту коштів і необхідності імпортувати потрібне обладнання з інших країн зриватиме будь-які детерміновані терміни введення в експлуатацію цих об'єктів. А тому необхідно цю нестационарну стохастичність врахувати в моделях процесу відновлення та розвитку енергетики України, що, на наш погляд, можна реалізувати у вигляді концепції переходу від детермінованих моделей (1)—(6) до авторегресійних моде-

лей, викладенню основних етапів якої ми і розглянемо далі», — кінець цитати [1].

А далі в нашій роботі [1] запропоновано деформувати — цитуємо: «на відрізьку часу  $T_{sp}$  детерміновані графіки, показані на рис. 1, уведенням до них стохастичних складових у вигляді стаціонарного «білого шуму» з нульовим середнім, імпульси якого генеруватимуться комп'ютерною програмою, створеною мовою Python, в межах заданої нами полоси значень, додаткової до детермінанти у 39 %, тобто в межах від  $-1,185$  до  $+1,185$  для графіка на рис. 1a і в межах від  $-6,9$  до  $+6,9$  для графіка на рис. 1b», — кінець цитати. Нагадаємо, що, виконуючи розрахунки цих меж, ми врахували інформацію з робіт Ольги Буславець [3], згідно з якою втрати і потужності, і споживаної електроенергії в бік їхнього збільшення становлять у 2023 році додатково до детермінанти у 39 %, закладеної в моделі, що запропоновані авторами роботи [1], ще приблизно 10 %.

Як ми уже зазначали в роботі [1], до цього ж етапу реалізації нашого способу синтезу моделей реальних втрат в умовах воєнного сьогодення слід віднести і визначення часового відрізьку  $T_{sp}$ , вимірюного в кварталах, на якому буде здійснено деформування стаціонарним «білим шумом» детермінованих моделей. Цей відрізок, як і в роботі [1], визначимо, зважаючи на такі припущення: вважатимемо, що війна з окупантами триватиме ще увесь 2024 та 2025 роки, а потім ще 5 років, тобто до 2030 року включно, потрібно буде в «зашумлених» умовах відновлювати зруйновану енергетику України. Тому моделювати процес після 2025 року варто теж у визначених на першому етапі умовах. Тож приймемо, що

$$T_{sp} = 30.$$

Отже деформуємо подані на рис. 1 графіки детермінованих моделей стаціонарним «білим шумом» на часовому відрізьку  $T_{sp}$  у визначених нами межах  $[y_{in}, y_{iv}], [w_{in}, w_{iv}]$  комп'ютерною програмою, створеною нами з використанням мови Python і наведеною в роботі [1]. Спотворені «білим шумом» графічні зображення матимуть вигляд:

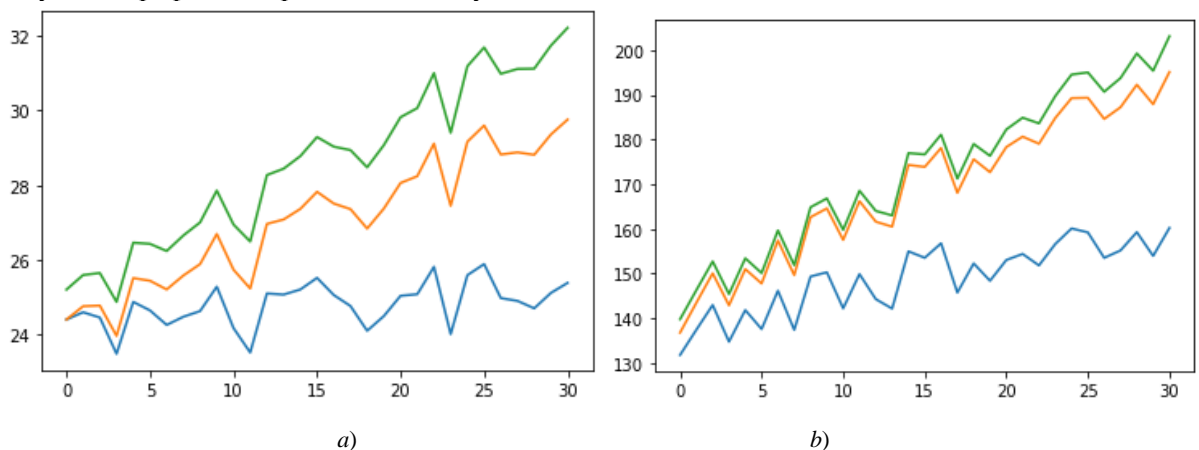


Рис. 2. Графіки математичних моделей: *a* — процесу зростання прогнозного максимуму навантаження споживачів; *b* — процесу зростання прогнозного попиту на електричну енергію в Україні за сценаріями СМЕЕ, РС та СКНЕ на період з 2023 по 2030 роки, адаптовані в роботі [1] до реалій воєнного сьогодення

А на другому етапі реалізації нашого способу синтезу моделей пропонуємо замість деформованих білим шумом детермінованих моделей використовувати для моделювання процесу відновлення та розвитку енергетики України авторегресійні моделі 3-го порядку, а оскільки, як видно з рис. 2, цей процес є не лише стохастичним, але і нестационарним, то його авторегресійні моделі синтезуватимемо у формі АРПКС (3,0,1), тобто у вигляді

$$v_i[k] = y_i[k] - y_i[k-1]; \quad w_i[k] = v_i[k] - y_i[k-1];$$

$$v_i[k] = v_i[0] + \sum_{j=1}^k w_i[j]; \quad y_i[k] = y_i[0] + \sum_{j=1}^k v_i[j], \quad k = 1, 2, \dots, N_{sp}; \quad j = 1, 2, \dots, k; \quad i = 1, 2, 3, 4, 5, 6; \quad (7)$$

$$w_i[k] = g_{i1}w_i[k-1] + g_{i2}w_i[k-2] + g_{i3}w_i[k-3] + a_i,$$

використання яких для моделювання після їхньої ідентифікації з нашої точки зору точніше наблизитимуть процес прогнозування відновлення та розвитку енергетики України до реалій воєнного сьогодення та післявоєнної доби, аніж детерміновані моделі цього процесу, що запропоновані в

роботі [2] та трансформовані нами вище у відповідні детерміновані математичні конструкції з аналітичною структурою.

Стосовно авторегресійних моделей (7) зауважимо, що вагові коефіцієнти  $g_{i1}, g_{i3}, g_{i3}, i=1, 2, 3, 4, 5, 6$  в них потрібно визначати з використанням методики Юла–Уокера, теорія якої викладена у роботі [4], а приклад застосування — у роботі [5].

Що ж до імпульсів «білого шуму»  $a_i$ , то їх потрібно генерувати у діапазоні значень, що задаються новими межами  $[y_{ina}, y_{iva}]$ , які потрібно визначати теж за методикою Юла–Уокера. А вибір нами 3-го порядку авторегресій (7) обумовлений тим, що за кварталного відліку часу відновлення об'єктів енергетичної інфраструктури в кварталі, що аналізується, залежатиме і від того, в якому стані ці об'єкти були в трьох попередніх кварталах, адже для відновлення суттєво зруйнованих джерел теплової генерації ремонтно-відновлювальні роботи можуть тривати упродовж часу в межах до трьох кварталів.

Для ідентифікації регресійних моделей (7) використаємо розширену частину комп'ютерної програми, створену мовою Python у вигляді програми 32 у роботі [5] одним з авторів цієї статті — професором Борисом Мокінім. У зв'язку з тим, що ця програма видасть нам сукупність результатів, що і чисельно і множинно відрізнятимуться від того, що наведено в роботі [5], ми позначимо цю програму у нашому списку Python-програм, використаних і в роботі [1], як програму 4.

Не виключено, що в процесі ремонтно-відновлювальних робіт дисперсія стохастичної складової процесу відновлення та розвитку енергетики України після 2030 року почне зменшуватись, тобто цей процес перейде в розряд гетероскедастичних, що вимагатиме корегування авторегресійних моделей (7), але внесення цих коректив може стати предметом досліджень уже після того, як війна і перший етап відновлення завершаться і такий характер змін почне проявлятися.

Отже ідентифікацію авторегресійних математичних моделей (7) здійснимо за програмою 4, вхідний масив даних для якої ми візьмемо з графіків, поданих на рис. 2, а сама програма має вигляд:

#### Програма 4

```
In [1]: import numpy as np
In [2]: L1=[25.2,25.6,25.7,25.0,26.7,26.6,26.4,26.8,27.2,28.0,27.2,26.8,28.2,28.4,
          28.7,29.2,29.0,28.6,28.4,29.4,30.0,30.2,30.8,29.2,31.3,31.8,31.9,31.0,
          31.1,31.6,32.0]
In [3]: L2=[24.3,24.5,24.8,24.0,25.3,25.2,25.1,25.9,26.0,26.5,26.1,25.3,27.0,27.2,
          27.3,27.8,27.6,27.2,27.0,27.6,27.8,28.1,29.0,28.0,29.2,29.4,29.5,28.8,
          28.8,28.9,30.0]
In [4]: L3=[24.3,24.4,24.4,23.7,25.0,24.3,24.2,24.5,24.8,25.0,24.0,23.5,25.0,25.1,
          25.2,25.3,25.0,24.7,24.3,24.8,25.0,25.0,25.5,24.1,25.2,25.4,25.5,25.0,
          24.8,24.9,25.0]
In [5]: L4=[140.0,144.0,150.0,145.0,152.0,150.0,152.0,160.0,165.0,166.0,160.0,
          165.0,163.0,162.0,171.0,175.0,180.0,172.0,176.0,174.0,180.0,182.0,
          191.0,192.0,195.0,195.0,189.0,190.0,195.0,192.0,200.0]
In [6]: L5=[137.0,143.0,148.0,144.0,150.0,148.0,152.0,159.0,164.0,165.0,159.0,
          164.0,163.0,161.0,170.0,174.0,178.0,171.0,175.0,173.0,178.0,176.0,
          179.0,190.0,193.0,193.0,187.0,189.0,193.0,188.0,190.0]
In [7]: L6=[133.0,138.0,142.0,135.0,140.0,137.0,145.0,138.0,147.0,150.0,141.0,
          145.0,143.0,141.0,151.0,150.0,152.0,145.0,148.0,145.0,147.0,148.0,
          150.0,151.0,153.0,153.0,151.0,152.0,155.0,150.0,155.0]
In [8]: N=31
In [9]: L11=np.diff(L1)
In [10]: L21=np.diff(L2)
In [11]: L31=np.diff(L3)
In [12]: L41=np.diff(L4)
In [13]: L51=np.diff(L5)
In [14]: L61=np.diff(L6)
In [15]: def fun(x):
          return np.sum(x)
In [16]: fun(L11)
In [17]: _(N-1)
Out[17]: 0.22666666666666668
In [18]: fun(L21)
In [20]: fun(L31)
In [19]: _(N-1)
In [21]: _(N-1)
Out[21]: 0.023333333333333331
In [22]: fun(L41)
In [24]: fun(L51)
In [23]: _(N-1)
In [25]: _(N-1)
```

```

Out[23]: 2.0
In [26]: fun(L61)
In [27]: _/(N-1)
Out[27]: 0.7333333333333333
In [31]: L42=np.diff(L41)
In [33]: L62=np.diff(L61)

In [35]: fun (L22)
Out[35]: 0.90000000000000021
In [37]: fun (L42)
Out[37]: 4.0
In [38]: _/(N-2)
Out[38]: 0.13793103448275862
In [41]: fun (L62)
Out[41]: 0.0
In [42]: def func (x):
            return np.dot(x,x)

In [45]: func (L22)
Out[45]: 32.029999999999994
In [46]: q20=_/(N-2);q20
Out[46]: 1.1044827586206893

In [49]: func (L42)
In [50]: q40=_/(N-2);q40
In [51]: func (L52)
In [52]: q50=_/(N-2);q50
In [53]: func (L62)
In [54]: q60=_/(N-2);q60
In [55]: L121=L12[:-1]
In [57]: L221=L22[:-1]
In [59]: L321=L32[:-1]
In [61]: L421=L42[:-1]
In [63]: L521=L52[:-1]
In [65]: L621=L62[:-1]
In [67]: def funct (x,y):
            return np.dot (x,y)

In [68]: funct (L121,L122)

In [70]: funct (L221,L222)
In [71]: q21=_/(N-3);q21
Out[71]: -0.6771428571428573

In [74]: funct (L421,L422)
In [75]: q41=_/(N-3);q41
Out[75]: -35.75

In [78]: funct(L621,L622)
In [79]: q61=_/(N-3);q61
Out[79]: -51.5
In [82]: L2211=L221[:-1]
In [84]: L3211=L321[:-1]
In [86]: L4211=L421[:-1]
In [88]: L5211=L521[:-1]
In [90]: L6211=L621[:-1]
In [92]: funct (L1211,L1222)
In [93]: q12=_/(N-4);q12
Out[93]: 0.009999999999999523

In [96]: funct (L3211,L3222)
In [97]: q32=_/(N-4);q32

Out[25]: 1.7666666666666666
In [28]: L12=np.diff(L11)
In [29]: L22=np.diff(L21)
In [30]: L32=np.diff(L31)
In [32]: L52=np.diff(L51)
In [34]: fun (L12)
Out[34]: -3.552713678800501e-15
In [36]: fun (L32)
Out[36]: 3.552713678800501e-15
In [39]: fun (L52)
Out[39]: -4.0
In [40]: _/(N-2)
Out[40]: -0.13793103448275862

In [43]: func(L12)
Out[43]: 43.620000000000002
In [44]: q10=_/(N-2);q10
Out[44]: 1.5041379310344833

In [47]: func (L32)
Out[47]: 29.259999999999997
In [48]: q30=_/(N-2);q30
Out[48]: 1.0089655172413783
Out[49]: 1614.0
Out[50]: 55.6551724137931
Out[51]: 1436.0
Out[52]: 49.51724137931034
Out[53]: 2022.0
Out[54]: 69.72413793103448
In [56]: L122=L12[1:]
In [58]: L222=L22[1:]
In [60]: L322=L32[1:]
In [62]: L422=L42[1:]
In [64]: L522=L52[1:]
In [66]: L622=L62[1:]

In [69]: q11=_/(N-3);q11
Out[69]: -0.8400000000000004

In [72]: funct (L321,L322)
In [73]: q31=_/(N-3);q31
Out[73]: -0.6021428571428561

In [76]: funct (L521,L522)
In [77]: q51=_/(N-3);q51
Out[77]: -27.571428571428573

In [80]: L1211=L121[:-1]
In [81]: L1222=L122[1:]
In [83]: L2222=L222[1:]
In [85]: L3222=L322[1:]
In [87]: L4222=L422[1:]
In [89]: L5222=L522[1:]
In [91]: L6222=L622[1:]

In [94]: funct (L2211,L2222)
In [95]: q22=_/(N-4);q22
Out[95]: 0.090000000000000065

In [98]: funct (L4211,L4222)

```

```

Out[97]: 0.058148148148
In [99]: q42=_(N-4);q42
Out[99]: 14.25925925925926

In [100]: funct (L5211,L5222)
In [101]: q52=_(N-4);q52
Out[101]: 4.5555555555555555

In [104]: L12111=L1211[:-1]
In [106]: L22111=L2211[:-1]
In [108]: L32111=L3211[:-1]
In [110]: L42111=L4211[:-1]
In [112]: L52111=L5211[:-1]
In [114]: L62111=L6211[:-1]
In [116]: funct (L12111,L12222)
In [117]: q13=_(N-5);q13
Out[117]: -0.04307692307692235

In [120]: funct (L32111,L32222)
In [121]: q33=_(N-5);q33
Out[121]: 0.00769230769230752

In [124]: funct (L52111,L52222)
In [125]: q53=_(N-5);q53
Out[125]: -6.0

In [128]: r10=q10/q10
In [130]: r12=q12/q10
In [132]: r20=q20/q20
In [134]: r22=q22/q20
In [136]: r30=q30/q30
In [138]: r32=q32/q30
In [140]: r40=q40/q40
In [142]: r42=q42/q40
In [144]: r50=q50/q50
In [146]: r52=q52/q50
In [148]: r60=q60/q60
In [150]: r62=q62/q60
In [152]: L7=[r10,r11,r12,r13]
In [154]: L07=[r40,r41,r42,r43]
In [156]: L9=[r30,r31,r32,r33]
In [158]: import sympy
In [159]: from sympy import*
In [160]: r1,r10,r11,r12,r13=symbols('r1 r10 r11 r12 r13')
In [161]: r2,r20,r21,r22,r23=symbols('r2 r20 r21 r22 r23')
In [162]: r3,r30,r31,r32,r33=symbols('r3 r30 r31 r32 r33')
In [163]: r4,r40,r41,r42,r43=symbols('r4 r40 r41 r42 r43')
In [164]: r5,r50,r51,r52,r53=symbols('r5 r50 r51 r52 r53')
In [165]: r6,r60,r61,r62,r63=symbols('r6 r60 r61 r62 r63')
In [166]: M1,M2,M3,M4,M5,M6=symbols('M1 M2 M3 M4 M5 M6')
In [167]: g1,g11,g12,g13=symbols('g1 g11 g12 g13')
In [168]: g2,g21,g22,g23=symbols('g2 g21 g22 g23')
In [169]: g3,g31,g32,g33=symbols('g3 g31 g32 g33')
In [170]: g4,g41,g42,g43=symbols('g4 g41 g42 g43')
In [171]: g5,g51,g52,g53=symbols('g5 g51 g52 g53')
In [172]: g6,g61,g62,g63=symbols('g6 g61 g62 g63')
In [173]: M1=Matrix([[r10,r11,r12],[r11,r10,r11],[r12,r11,r10]])
In [174]: M2=Matrix([[r20,r21,r22],[r21,r20,r21],[r22,r21,r20]])
In [175]: M3=Matrix([[r30,r31,r32],[r31,r30,r31],[r32,r31,r30]])
In [176]: M4=Matrix([[r40,r41,r42],[r41,r40,r41],[r42,r41,r40]])
In [177]: M5=Matrix([[r50,r51,r52],[r51,r50,r51],[r52,r51,r50]])
In [102]: funct (L6211,L6222)
In [103]: q62=_(N-4);q62
Out[103]: 21.925925925925927

In [105]: L12222=L1222[1:]
In [107]: L22222=L2222[1:]
In [109]: L32222=L3222[1:]
In [111]: L42222=L4222[1:]
In [113]: L52222=L5222[1:]
In [115]: L62222=L6222[1:]

In [118]: funct (L22111,L22222)
In [119]: q23=_(N-5);q23
Out[119]: 0.15615384615384537

In [122]: funct (L42111,L42222)
In [123]: q43=_(N-5);q43
Out[123]: -7.5

In [126]: funct (L62111,L62222)
In [127]: q63=_(N-5);q63
Out[127]: -3.8076923076923075

In [129]: r11=q11/q10
In [131]: r13=q13/q10
In [133]: r21=q21/q20
In [135]: r23=q23/q20
In [137]: r31=q31/q30
In [139]: r33=q33/q30
In [141]: r41=q41/q40
In [143]: r43=q43/q40
In [145]: r51=q51/q50
In [147]: r53=q53/q50
In [149]: r61=q61/q60
In [151]: r63=q63/q60
In [153]: L8=[r20,r21,r22,r23]
In [155]: L08=[r50,r51,r52,r53]
In [157]: L09=[r60,r61,r62,r63]

```

```

In [178]: M6=Matrix([[r60,r61,r62],[r61,r60,r61],[r62,61,r60]])
In [179]: M1=M1.subs([(r10,L7[0]),(r11,L7[1]),(r12,L7[2])])
In [180]: M2=M2.subs([(r20,L8[0]),(r21,L8[1]),(r22,L8[2])])
In [181]: M3=M3.subs([(r30,L9[0]),(r31,L9[1]),(r32,L9[2])])
In [182]: M4=M4.subs([(r40,L07[0]),(r41,L07[1]),(r42,L07[2])])
In [183]: M5=M5.subs([(r50,L08[0]),(r51,L08[1]),(r52,L08[2])])
In [184]: M6=M6.subs([(r60,L09[0]),(r61,L09[1]),(r62,L09[2])])
In [185]: r1=Matrix([r11,r12,r13])
In [186]: r2=Matrix([r21,r22,r23])
In [187]: r3=Matrix([r31,r32,r33])
In [188]: r4=Matrix([r41,r42,r43])
In [189]: r5=Matrix([r51,r52,r53])
In [190]: r6=Matrix([r61,r62,r63])
In [191]: r1=r1.subs([(r11,L7[1]),(r12,L7[2]),(r13,L7[3])])
In [192]: r2=r2.subs([(r21,L8[1]),(r22,L8[2]),(r23,L8[3])])
In [193]: r3=r3.subs([(r31,L9[1]),(r32,L9[2]),(r33,L9[3])])
In [194]: r4=r4.subs([(r41,L07[1]),(r42,L07[2]),(r43,L07[3])])
In [195]: r5=r5.subs([(r51,L08[1]),(r52,L08[2]),(r53,L08[3])])
In [196]: r6=r6.subs([(r61,L09[1]),(r62,L09[2]),(r63,L09[3])])
In [197]: B1=simplify(M1.inv())
In [198]: B2=simplify(M2.inv())
In [199]: B3=simplify(M3.inv())
In [200]: B4=simplify(M4.inv())
In [201]: B5=simplify(M5.inv())
In [202]: B6=simplify(M6.inv())
In [203]: g1=Matrix([g11,g12,g13])
In [204]: g2=Matrix([g21,g22,g23])
In [205]: g3=Matrix([g31,g32,g33])
In [206]: g4=Matrix([g41,g42,g43])
In [207]: g5=Matrix([g51,g52,g53])
In [208]: g6=Matrix([g61,g62,g63])
In [209]: g1=B1*r1
In [210]: g11=g1[0,0];g11
Out[210]: -1.02103356730580
In [211]: g12=g1[1,0];g12
Out[211]: -0.835101240928020
In [212]: g13=g1[2,0];g13
Out[212]: -0.487478202199174
In [213]: g2=B2*r2
In [214]: g21=g2[0,0];g21
Out[214]: -0.974464275600005

In [215]: g22=g2[1,0];g22
Out[215]: -0.609785460248206

In [216]: g23=g2[2,0];g23
Out[216]: -0.153063718305497

In [217]: g3=B3*r3
In [218]: g31=g3[0,0];g31
Out[218]: -1.07418874103335
In [219]: g32=g3[1,0];g32
Out[219]: -0.841733099052374
In [220]: g33=g3[2,0];g33
Out[220]: -0.432808824491917
In [221]: g4=B4*r4
In [222]: g41=g4[0,0];g41
Out[222]: -0.860908425296780
In [223]: g42=g4[1,0];g42
Out[223]: -0.411438113890511
In [224]: g43=g4[2,0];g43
Out[224]: -0.178473908282002
In [225]: g5=B5*r5
In [226]: g51=g5[0,0];g51
Out[226]: -0.849633039375595
In [227]: g52=g5[1,0];g52
Out[227]: -0.587019676423628
In [228]: g53=g5[2,0];g53
Out[228]: -0.369859468251363
In [229]: g6=B6*r6
In [230]: g61=g6[0,0];g61
Out[230]: -0.883867036348098
In [231]: g62=g6[1,0];g62
Out[231]: -0.00376536216178815

```

```

In [232]: g63=g6[2,0];g63
Out[232]: 0.453023103209171
In [233]: g1=g1.evalf(3)
In [234]: g2=g2.evalf(3)
In [235]: g3=g3.evalf(3)
In [236]: g4=g4.evalf(3)
In [237]: g5=g5.evalf(3)
In [238]: g6=g6.evalf(3)
In [239]: a1,a2,a3,a4,a5,a6=symbols('a1 a2 a3 a4 a5 a6')
In [240]: sk1,sk2,sk3,sk4,sk5,sk6=symbols('sk1 sk2 sk3 sk4 sk5 sk6')
In [241]: kv1,kv2,kv3,kv4,kv5,kv6=symbols('kv1 kv2 kv3 kv4 kv5 kv6')
In [242]: a11,a12,a21,a22,a31,a32,a41,a42,a51,a52,a61,a62=\
symbols('a11 a12 a21 a22 a31 a32 a41 a42 a51 a52 a61 a62')
In [243]: sk1=q10-g11*q11-g12*q12-g13*q13
In [244]: kv1=sk1**(0.5);kv1
Out[244]: 0.796129189195499
In [245]: a11=-2*kv1;a11
Out[245]: -1.59225837839100
In [246]: a12=2*kv1;a12
Out[246]: 1.59225837839100
In [247]: sk2=q20-g21*q21-g22*q22-g23*q23
In [248]: kv2=sk2**(0.5);kv2
Out[248]: 0.723473160939374
In [249]: a21=-2*kv2;a21
Out[249]: -1.44694632187875
In [250]: a22=2*kv2;a22
Out[250]: 1.44694632187875
In [251]: sk3=q30-g31*q31-g32*q32-g33*q33
In [252]: kv3=sk3**(0.5);kv3
Out[252]: 0.643758463400445
In [253]: a31=-2*kv3;a31
Out[253]: -1.28751692680089
In [254]: a32=2*kv3;a32
Out[254]: 1.28751692680089
In [255]: sk4=q40-g41*q41-g42*q42-g43*q43
In [256]: kv4=sk4**(0.5);kv4
Out[256]: 5.42272483465864
In [257]: a41=-2*kv4;a41
Out[257]: -10.8454496693173
In [258]: a42=2*kv4;a42
Out[258]: 10.8454496693173
In [259]: sk5=q50-g51*q51-g52*q52-g53*q53
In [260]: kv5=sk5**(0.5);kv5
Out[260]: 5.15234787848067
In [261]: a51=-2*kv5;a51
Out[261]: -10.3046957569613
In [262]: a52=2*kv5;a52
Out[262]: 10.3046957569613
In [263]: sk6=q60-g61*q61-g62*q62-g63*q63
In [264]: kv6=sk6**(0.5);kv6
Out[264]: 5.10024677797531
In [265]: a61=-2*kv6;a61
Out[265]: -10.2004935559506
In [266]: a62=2*kv6;a62
Out[266]: 10.2004935559506

```

Кінець програми 4.

Підставляючи чисельні значення коефіцієнтів  $g_{i1}, g_{i2}, g_{i3}, i = 1, 2, 3, 4, 5, 6$ , визначених програмою 4, у вирази (7), матимемо авторегресійні моделі процесів відновлення та розвитку енергетики України у вигляді спільної складової

$$v_i[k] = y_i[k] - y_i[k-1]; \quad w_i[k] = v_i[k] - v_i[k-1];$$

$$v_i[k] = v_i[0] + \sum_{j=1}^k w_i[j]; \quad y_i[k] = y_i[0] + \sum_{j=1}^k v_i[j], \quad k = 1, 2, \dots, N_{sp}; \quad j = 1, 2, \dots, k; \quad i = 1, 2, 3, 4, 5, 6; \quad (8)$$

та авторегресій



$$w_1[k] = -1,02w_1[k-1] - 0,835w_1[k-2] - 0,487w_1[k-3] + a_1; \quad (9)$$

$$w_2[k] = -0,974w_2[k-1] - 0,61w_2[k-2] - 0,153w_2[k-3] + a_2; \quad (10)$$

$$w_3[k] = -1,07w_3[k-1] - 0,842w_3[k-2] - 0,433w_3[k-3] + a_3; \quad (11)$$

$$w_4[k] = -0,861w_4[k-1] - 0,411w_4[k-2] - 0,178w_4[k-3] + a_4; \quad (12)$$

$$w_5[k] = -0,85w_5[k-1] - 0,587w_5[k-2] - 0,378w_5[k-3] + a_5; \quad (13)$$

$$w_6[k] = -0,884w_6[k-1] - 0,004w_6[k-2] - 0,453w_6[k-3] + a_6. \quad (14)$$

Зауважимо, що повний текст програми 4 викладено в статті заради того, щоб в разі, якщо хтось із дослідників вирішить синтезувати авторегресії з іншим порядком та з іншими параметрами генерації імпульсів «білого шуму», то щоб він мав змогу не витратити час на створення свого варіанту програми, а міг використати уже готовий варіант цієї програми, створений нами.

### Висновки

В доповнення до результатів, викладених в нашій роботі [1], синтезовано авторегресійні моделі процесів відновлення та розвитку електроенергетики України, адаптовані до реалій воєнного сьогодення.

Здійснено обґрунтування вибору класу авторегресій, їхнього порядку та діапазонів генерації імпульсів «білого шуму», якими деформуються детерміновані базові математичні моделі.

### СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

[1] Б. І. Мокін, Д. О. Шалагай, «Синтез математичних моделей процесу відновлення та розвитку електроенергетики України, наближених до реалій воєнного сьогодення.» *Вісник Вінницького політехнічного інституту*, № 6, с. 42-47, 2023. <https://doi.org/10.31649/1997-9266-2023-171-6-6-13>.

[2] Національна рада з відновлення України. Проект Плану відновлення України. Розділ «Енергетична безпека», 2022. 164 с. [Електронний ресурс]. Режим доступу: <https://www.kmu.gov.ua/storage/app/sites/1/recoveryrada/ua/restoration-and-development-of-infrastructure.pdf>.

[3] Ольга Буславець, *Результати споживання електроенергії за 2022 рік*. [Електронний ресурс]. Режим доступу: <https://www.facebook.com/olhabuslavets/posts/pfbid0jCwzcCNaLCCCHx93MMeRv9KLS7Tn8Fny7wJWwffZSgzoYJq7pCbVgNKz9ttZGwWPI>. Дата звернення: 17.02.2024.

[4] Б. І. Мокін, В. Б. Мокін, і О. Б. Мокін, *Функціональний аналіз, адаптований до прикладних задач в галузі інформаційних технологій*. Вінниця, Україна: ВНТУ, 2020, 192 с. ISBN 978-966-641-796-4.

[5] Б. І. Мокін, В. Б. Мокін, і О. Б. Мокін, *Навчальний посібник по опануванню студентами способів розв'язання задач з функціонального аналізу мовою Python. Частина 2*. Вінниця, Україна: ВНТУ, 2023, 144 с. ISBN 978-966-641-926-5.

Рекомендована кафедрою системного аналізу та інформаційних технологій ВНТУ

Стаття надійшла до редакції 23.09.2024

**Мокін Борис Іванович** — академік НАПН України, д-р техн. наук, професор, професор кафедри системного аналізу та інформаційних технологій, e-mail: [borys.mokin@gmail.com](mailto:borys.mokin@gmail.com);

**Мазурук Олег Володимирович** — аспірант кафедри системного аналізу та інформаційних технологій, e-mail: [omazuruk3@gmail.com](mailto:omazuruk3@gmail.com);

**Собчук Наталія Валеріївна** — канд. техн. наук, доцент, доцент кафедри електричних станцій та систем, email: [natashasobchuk37@gmail.com](mailto:natashasobchuk37@gmail.com).

Вінницький національний технічний університет, Вінниця;

**Шалагай Дмитро Олександрович** — інженер-проектувальник, e-mail: [d.shalagai@gmail.com](mailto:d.shalagai@gmail.com).

ТОВ «Промавтоматика Вінниця», Вінниця

**B. I. Mokin**<sup>1</sup>  
**O. V. Mazuruk**<sup>1</sup>  
**N. V. Sobchuk**<sup>1</sup>  
**D. O. Shalagai**<sup>2</sup>

## **Autoregressive Models for the Recovery and Development of War-Damaged Power Industry of Ukraine**

<sup>1</sup>Vinnitsia National Technical University;

<sup>2</sup>Ltd “Promavtomatika Vinnitsia”

*Method for adjusting the reconstruction and development plan of Ukraine's power industry, developed in the work “National Council for the Recovery of Ukraine. Project for the Recovery Plan of Ukraine. Section 'Energy Security,' 2022, 164 p.” has been proposed, adapting it to the conditions that arose after the bombing of Ukrainian energy infrastructure by Russian aggressors. Primary deterministic and adjusted mathematical models have been synthesized, which take into account the state of Ukraine's power industry at the time of writing this work. Since the idea of using autoregressive models instead of the adjusted mathematical models was only mentioned in the work “Synthesis of Mathematical Models for the Process of Recovery and Development of Ukraine's Power Industry, Approximated to the Realities of Wartime. Bulletin of Vinnitsia Polytechnic Institute, 2023” and was presented in a general form, a year after the writing of that work, during the course of Ukraine's war with Russian invaders, Ukraine's power infrastructure suffered additional damage, necessitating the synthesis of autoregressive models in a completed form, ready for use by electrical engineers in project work. Thus, in this work, autoregressive models of the processes of recovery and development of Ukraine's power industry, adapted to the realities of wartime, have been synthesized, and the choice of the autoregression class, their order, and the ranges of white noise impulse generation, which distort the deterministic base mathematical models, has been substantiated.*

**Keywords:** power industry of Ukraine, recovery plan, wartime realities, deterministic process models, stochasticity introduction, white noise, autoregressive models.

**Mokin Borys I.** — Academician of NAPS of Ukraine, Dr. Sc. (Eng.), Professor, Professor of the Chair of System Analysis and Information Technologies, e-mail: borys.mokin@gmail.com ;

**Mazuruk Oleh V.** — Post-Graduate Student of the Chair of System Analysis and Information Technologies, e-mail: omazuruk3@gmail.com ;

**Shalagai Dmytro O.** — Design Engineer, e-mail: d.shalagai@gmail.com ;

**Sobchuk Nataliia V.** — Cand. Sc. (Eng.), Associate Professor of the Chair of Electric Power Stations and Systems, e-mail: natashasobchuk37@gmail.com