

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

## **АВТОРЕГРЕСІЙНІ МОДЕЛІ ПРОЦЕСУ ПОВОЄННОГО ВІДНОВЛЕННЯ ТА ПІСЛЯВОЄННОГО РОЗВИТКУ ВДЕ УКРАЇНИ**

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

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

В роботі [1] опубліковано «Проект Плану відновлення України. Енергетика», розроблений у 2022 році Національною радою з відновлення України. В нашій роботі [2] з метою наближення до реалій воєнного сьогоднішнього дня «Проекту Плану...» здійснено синтез математичних моделей процесу відновлення та розвитку електроенергетики України в цілому з урахуванням стохастичного характеру цього процесу. А в роботі [3] з метою наближення до реалій воєнного сьогоднішнього дня «Проекту Плану...» в частині, присвяченій електроенергетиці, авторами здійснено синтез математичних моделей процесу відновлення та розвитку джерел відновлювальної електроенергетики (ВДЕ) України. Цю задачу розв'язано з використанням даних, таблиць та графіків, що характеризують ВДЕ, взятих з роботи [1]. Але як ВДЕ розглядалися не лише сонячні (СЕС), вітрові (ВЕС) та біогазові (БіоЕС) електростанції, як це зазначено в роботі [1], але і гідровідновлювальні (ГЕС) та гідроаккумуляційні (ГАЕС) електростанції, які в роботі [1] до складу ВДЕ не включені. В роботі [3] ми вважали, що війна України з агресивним сусідом триватиме і у 2024 році і у 2025 році, що вноситиме стохастичну складову у плани відновлення і того, що було до війни, і того, що у ці воєнні роки уже відновлено. Вважали також, що і в подальшому процес відновлення і розвитку ВДЕ довгий час міститиме стохастичну складову, для математичного моделювання якої доцільно використовувати авторегресійні моделі 2-го порядку у формі АРПКС(2,0,1), починаючи з нульової відмітки часу у 2023 році. Але ідея використовувати авторегресійні моделі була лише зазначена в нашій роботі [3] і записана лише в загальній постановці. Через рік після написання роботи [3], що пройшов у вирі війни України з російськими окупантами, українська електроенергетична структура в цілому та особливо ВДЕ зазнали додаткових руйнувань [4], тож виникла необхідність в синтезі авторегресійних моделей в завершеній формі, готовій для використання інженерами-електриками в проектній роботі. Такий задум і реалізовано у цій публікації у вигляді Python-програми, вихідні передумови створення якої авторами уже озвучено у нашій доповіді на Міжнародній науковій конференції, що проходила в Києві в листопаді 2024 року, розширені тези якої опубліковані в роботі [5].

**Ключові слова:** електроенергетика України, відновлювальні джерела енергетики, відновлення та розвиток, стохастичний процес, повоєнний та післявоєнний період, авторегресійні моделі.

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

В роботі [1] опубліковано «Проект Плану відновлення України. Енергетика», розроблений у 2022 році Національною радою з відновлення України.

В нашій роботі [2] з метою наближення до реалій воєнного сьогоднішнього дня «Проекту Плану відновлення України. Енергетика» в частині, що стосується електроенергетики, розробленою робочою групою з «Енергетичної безпеки», здійснено синтез математичних моделей процесу відновлення та розвитку електроенергетики України в цілому з урахуванням стохастичного характеру цього процесу.

А в роботі [3], знову ж таки з метою наближення до реалій воєнного сьогодення «Проекту Плану відновлення України. Енергетика», в частині, що стосується електроенергетики, авторами здійснено синтез математичних моделей процесу відновлення та розвитку джерел відновлювальної електроенергетики (ВДЕ) України, які ще до початку воєнної інтервенції з боку сусідньої агресивної держави, розпочатої у лютому 2022 року, вносили уже помітний вклад в електроенергетику України.

Цю задачу розв'язано за використання даних, таблиць та графіків, які характеризують ВДЕ, взятих з роботи [1]. Але як ВДЕ автори розглядали не лише сонячні електростанції (СЕС), вітрові електростанції (ВЕС) та біогазові електростанції (БіоЕС), як це визначено в роботі [1], але і гідралічні електростанції (ГЕС) та гідроакумулювальні електростанції (ГАЕС), які в роботі [1] до складу ВДЕ не включені, а розглядаються окремо від ВДЕ, хоча за своєю природою є такими.

В роботі [3], як і в роботі [2], ми вважали, що війна України з агресивним сусідом триватиме і у 2024 році і у 2025 році, що вноситиме стохастичну складову у плани відновлення і того, що було в електроенергетичній структурі України до війни, і того, що у ці воєнні роки уже після руйнації відновлено. Вважали також, що і в подальшому процес відновлення і розвитку ВДЕ міститиме стохастичну складову, обумовлену і процесами розмінування об'єктів і під'їздів до них, і процесами невчасного надходження потрібного обладнання з-за кордону, і проблемами в забезпеченні відновлювальних робіт необхідними коштами та високопрофесійними кадрами.

Отже, за ідеологією робіт [2] і [3] ми розглядали процес відновлення і розвитку ВДЕ, починаючи з нульової відмітки часу у 2023 році, в стохастичному варіанті. Але, якщо в роботі [2], синтезуючи моделі процесу відновлення та розвитку електроенергетики України взагалі, кінцеву точку цього процесу автори визначали 2030 роком, то в роботі [3], синтезуючи моделі процесу відновлення і розвитку ВДЕ, ми з урахуванням прогностичних особливостей, визначених в роботі [1], як кінцеву точку процесу визначили 2033 рік. А це, у зв'язку з вимірюванням часу в кварталах, дозволило нам в процесі синтезу математичних моделей у період з 2023 по 2033 роки використовувати 41 точку на осі часу.

Синтезуючи в роботі [3] математичні моделі процесу відновлення та розвитку ГЕС, ГАЕС, СЕС, ВЕС та БіоЕС, виходили з того, що в 2033 році сумарні потужності цих ВДЕ досягнуть рівня, закладеного в роботі [1].

А щодо сумарних потужностей ГЕС, ГАЕС, СЕС, ВЕС та БіоЕС у початковому 2023 році, то з урахуванням інформації, поданої в роботі [1], про рівні і втрати цих потужностей, викликаних з початку з лютого 2022 року бомбардуваннями та окупацією агресивною сусідньою державою, та коригуванням цих рівнів згідно з інформацією про руйнування ВДЕ, що надійшла додатково, ми їх визначили такими:

$$\begin{cases} P_{\text{ГЕС}(n)} = 4,44 \text{ ГВт} = y_{1n}, \\ P_{\text{ГАЕС}(n)} = 1,86 \text{ ГВт} = y_{2n}, \\ P_{\text{СЕС}(n)} = 4,3 \text{ ГВт} = y_{3n}, \\ P_{\text{ВЕС}(n)} = 0,1 \text{ ГВт} = y_{4n}, \\ P_{\text{БіоЕС}(n)} = 0,18 \text{ ГВт} = y_{5n}. \end{cases} \quad (1)$$

Це дало нам змогу в роботі [3] дійти висновку, що наближено в детермінованому варіанті на відрізку часу з 2023 по 2033 роки, вираженому у кварталах, природи потужностей ГЕС, ГАЕС, СЕС, ВЕС та БіоЕС можна відобразити моделями

$$\begin{cases} y_1(t) = 4,44 + 0,016t, \\ y_2(t) = 1,86 + 0,011t, \\ y_3(t) = 4,3 + 0,1t, \\ y_4(t) = 0,1 + 0,11t, \\ y_5(t) = 0,18 + 0,01t. \end{cases} \quad (2)$$

А для визначення діапазонів  $\Delta_i$ ,  $i = 1, 2, \dots, 5$  та нижньої і верхньої меж  $m_{i1}$ ,  $m_{i2}$  формування імпульсів білого шуму, за допомогою щоквартального підмішування яких у вигляді дискретних стохастичних функцій  $x_i(t_n)$ ,  $i = 1, 2, \dots, 5$ ;  $n = 0, 1, 2, \dots, 40$  з нульовим середнім до моделей (2), заданих у дискретному часі, згідно з ідеологією методу синтезу, запропонованого нами в нашій роботі [2],

ми скористались оцінками додаткових втрат Ольги Буславець, один з варіантів яких подано в її роботі [4] і задали ці додаткові втрати в межах 10 % від тієї потужності ГЕС, ГАЕС, яку ці ВДЕ мали у 2023 році, та в межах 30 % від тієї потужності СЕС, ВЕС та БіоЕС, яку ці ВДЕ мали у 2023 році, та які визначені нами у виразах (1). Тож ці, тепер уже стохастичні моделі

$$w_i(n) = y_i(n) - x_i(n), \quad i = 1, 2, \dots, 5; \quad n = 0, 1, 2, \dots, 40, \quad (3)$$

реалізацію яких здійснено в роботі [3] за допомогою комп'ютерної програми, створеної мовою Python, отримали вигляд, показаний на рис. 1.

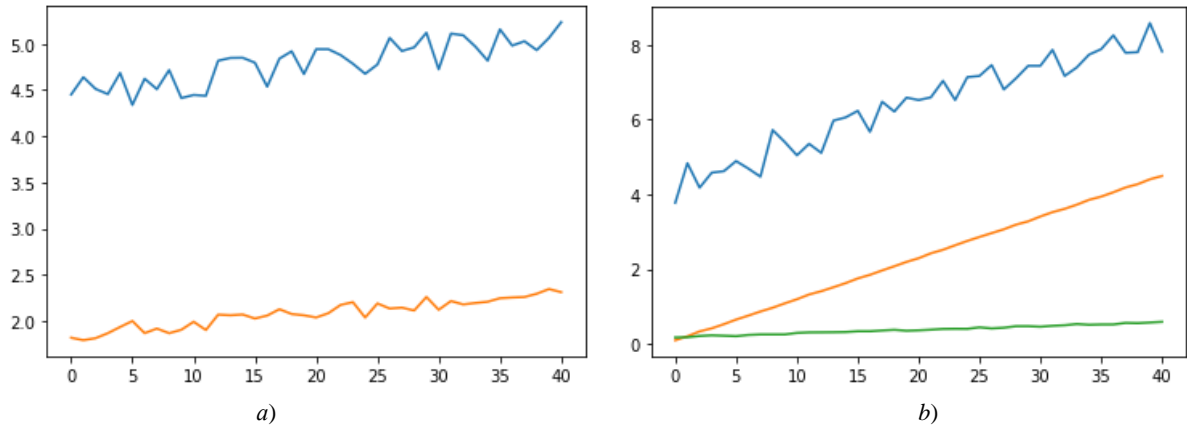


Рис. 1. Графіки процесів песимістичного характеру щодо відновлення та розвитку потужностей: *a* — ГЕС, ГАЕС; *б* — СЕС, ВЕС, БіоЕС в ГВт в період з 2023 по 2033 роки з урахуванням воєнного сьогодення

Аналізуючи графіки рис. 1, бачимо, що для прогнозування процесів відновлення та розвитку потужностей ВЕС і БіоЕС у період з 2023 по 2033 роки (в разі вимірювання цих потужностей у ГВт, а час у кварталах) можна використовувати детерміновані математичні моделі  $y_4(t)$  та  $y_5(t)$ , задані відповідними функціями у множині (2), оскільки вони є трендами із зовсім незначними стохастичними складовими.

Але з цих же графіків рис. 1, видно, що для прогнозування процесів відновлення та розвитку потужностей ГЕС, ГАЕС та СЕС у цей же період та в тих самих умовах необхідно замість детермінованих моделей  $y_1(t)$ ,  $y_2(t)$ ,  $y_3(t)$  використовувати авторегресійні моделі 2-го порядку. А оскільки ці процеси є не лише стохастичними, але і нестационарними, то ці авторегресійні моделі синтезуватимемо у формі АРПКС(2,0,1), тобто у такому вигляді:

$$\begin{aligned} w_i^*(k) &= w_i(k) - w_i(k-1); \quad y_i(k) = y_i(0) + \sum_{j=1}^k y_i^*[j], \\ k &= 1, 2, \dots, N; \quad j = 0, 1, 2, \dots, k; \quad i = 1, 2, 3, \\ y_i^*[k] &= p_{1i} w_i^*[k-1] + p_{2i} w_i^*[k-2] + u_i. \end{aligned} \quad (4)$$

Але ідея використовувати замість скоригованих детермінованих математичних моделей авторегресійні моделі була лише означена в нашій роботі [3] і записана лише в загальній постановці.

Через рік після написання роботи [3], 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. Цей задум і реалізовано у цій публікації у вигляді Python-програми, вихідні передумови створення якої нами уже зазначені у тезах нашої доповіді на Міжнародній науковій конференції, опублікованих в роботі [5].

```
In [1]: import numpy
In [2]: from numpy import*
In [3]: N=40
In [4]: m11=-0.222; m21=-0.093;\
        m31=-0.645
In [5]: m12=0.222; m22=0.093;\
        m32=0.645
In [6]: t=linspace(0,N,N+1)
```

```
In [7]: def y1(t):
        return 4.44+0.016*t
In [8]: y1vec=vectorize(y1)
In [9]: y11=y1vec(t)
In [10]: def y2(t):
        return 1.86+0.011*t
In [11]: y2vec=vectorize(y2)
In [12]: y22=y2vec(t)
```

```

In [13]: def y3(t):
         return 4.30+0.10*t
In [14]: y3vec=vectorize(y3)
In [15]: y33=y3vec(t)
In [16]: g1=random.uniform(m11,
                             m12,41)
In [17]: g2=random.uniform(m21,
                             m22,41)
In [18]: g3=random.uniform(m31,
                             m32,41)
In [19]: y111=y11-g1
In [20]: y222=y22-g2
In [21]: y333=y33-g3
In [22]: L1=list (y111)
In [23]: L2=list (y222)
In [24]: L3=list (y333)
In [25]: L11=diff (L1)
In [26]: L21=diff (L2)
In [27]: L31=diff (L3)
In [28]: def fun(x):
         return sum (x)
In [29]: fun(L11)
Out[29]: 0.6985935242327264
In [30]: _/(N-1)
Out[30]: 0.017912654467505805
In [31]: fun(L21)
Out [31]: 0.40094917297013066
In [32]: _/(N-1)
Out[32]: 0.010280748024875145
In [33]: fun(L31)
Out[33]: 4.031708771904489
In [34]: _/(N-1)
Out[34]: 0.103377147997551
In [35]: L12=diff (L11)
In [36]: L22=diff (L21)
In [37]: L32=diff (L31)
In [38]: fun (L12)
Out[38]: 0.037857139702390796
In [39]: fun (L22)
Out[39]: 0.18498991166662693
In [40]: fun (L32)
Out[40]: 1.1121459593545202
In [41]: def func (x):
         return dot(x, x)
In [42]: func (L11)
Out[42]: 1.2802224178610895
In [43]: q10=_/(N-1); q10
Out[43]: 0.03282621584259204
In [44]: func (L21)
Out[44]: 0.2760423995860152
In [45]: q20=_/(N-1); q20
Out[45]: 0.007078010245795261
In [46]: func (L31)
Out[46]: 10.978193302639152
In [47]: q30=_/(N-1); q30
Out[47]: 0.28149213596510647
In [48]: L111=L11[: -1]
In [49]: L112=L11[1:]
In [50]: L211=L21[: -1]
In [51]: L212=L21[1:]
In [52]: L311=L31[: -1]
In [53]: L312=L31[1:]
In [54]: def funct (x, y):
         return dot (x, y)
In [55]: funct (L111, L112)
Out[55]: -0.475961149410624
In [56]: q11=_/(N-2); q11
Out[56]: -0.012525293405542736
In [57]: funct (L211, L212)
Out[57]: -0.11732849921015807
In [58]: q21=_/(N-2); q21
Out[58]: -0.003087592084477844
In [59]: funct (L311, L312)
Out[59]: -3.895105873827287
In [60]: q31=_/(N-2); q31
Out[60]: -0.10250278615334966
In [61]: L1111=L111[: -1]
In [62]: L1122=L112[1:]
In [63]: L2111=L211[: -1]
In [64]: L2122=L212[1:]
In [65]: L3111=L311[: -1]
In [66]: L3122=L312[1:]
In [67]: funct (L1111, L1122)
Out[67]: -0.15991913675542
In [68]: q12=_/(N-3); q12
Out[68]: -0.004322138831227567
In [69]: funct (L2111, L2122)
Out[69]: -0.01287666343443047
In [70]: q22=_/(N-3); q22
Out[70]: -0.000348017930660283
In [71]: funct (L3111, L3122)
Out[71]: -0.3694380463027638
In [72]: q32=_/(N-3); q32
Out[72]: -0.009984812062236859
In [73]: r10=q10/q10
In [74]: r11=q11/q10
In [75]: r12=q12/q10
In [76]: r20=q20/q20
In [77]: r21=q21/q20
In [78]: r22=q22/q20
In [79]: r30=q30/q30
In [80]: r31=q31/q30
In [81]: r32=q32/q30
In [82]: L7= [r10, r11, r12]
In [83]: L8= [r20, r21, r22]
In [84]: L9= [r30, r31, r32]
In [85]: import sympy
In [86]: from sympy import*
In [87]: r1, r10, r11, r12= \
         symbols ('r1 r10 r11 r12')
In [88]: r2, r20, r21, r22= \
         symbols ('r2 r20 r21 r22')
In [89]: r3, r30, r31, r32= \
         symbols ('r3 r30 r31 r32 ')

```

```

In [90]: M1, M2, M3= \
symbols ('M1 M2 M3')
In [91]: p1, p11, p12= \
symbols ('p1 p11 p12')
In [92]: p2, p21, p22= \
symbols ('p2 p21 p22')
In [93]: p3, p31, p32= \
symbols ('p3 p31 p32')
In [94]: M1=Matrix([[r10, r11],
[r11,r10]])
In [95]: M2=Matrix([[r20, r21],
[r21,r20]])
In [96]: M3=Matrix([[r30, r31],
[r31,r30]])
In [97]: M1=M1.subs([(r10,
L7[0]), (r11, L7[1])])
In [98]: M2=M2.subs([(r20,
L8[0]), (r21, L8[1])])
In [99]: M3=M3.subs([(r30,
L9[0]), (r31, L9[1])])
In [100]: r1=Matrix([r11, r12])
In [101]: r2=Matrix([r21, r22])
In [102]: r3=Matrix([r31, r32])
In [103]: r1=r1.subs([(r11, L7[1]), (r12, L7[2])])

In [104]: r2=r2.subs([(r21,
L8[1]), (r22, L8[2])])
In [105]: r3=r3.subs([(r31,
L9[1]), (r32, L9[2])])
In [106]: B1=simplify(M1.inv())
In [107]: B2=simplify(M2.inv())
In [108]: B3=simplify(M3.inv())
In [109]: p1=Matrix([p11, p12])
In [110]: p2=Matrix([p21, p22])
In [111]: p3=Matrix([p31, p32])
In [112]: p1=B1*r1
In [113]: p11=p1[0,0]; p11
Out[113]:
-0.505382238786753
In [114]: p12=p1[1,0]; p12
Out[114]:
-0.324502821313947
In [115]: p2=B2*r2
In [116]: p21=p2[0,0]; p21
Out[116]:
-0.565229716891099
In [117]: p22=p2[1,0]; p22
Out[117]: -0.295735193614389
In [118]: p3=B3*r3
In [119]: p31=p3[0,0]; p31
Out[119]:
-0.434697569450163
In [120]: p32=p3[1,0]; p32
Out[120]:
-0.193762159209055
In [121]: p1=p1.evalf(3); p1
Out[121]:
Matrix([
[-0.505],
[-0.325]])
In [122]: p2=p2.evalf(3); p2
Out [122]:
Matrix([
[-0.565],
[-0.296]])
In [123]: p3=p3.evalf(3); p3
Out [123]:
Matrix([
[-0.435],
[-0.194]])
In [124]: u1, u2, u3= \
symbols ('u1 u2 u3')
In [125]: sk1, sk2, sk3= \
symbols('sk1 sk2 sk3')
In [126]: kv1, kv2, kv3=symbols('kv1 kv2 kv3')
In [127]: a11, a12, a21, a22, \
a31, a32= symbols ('a11
a12 a21 a22 a31 a32')
In [128]: sk1=q10-p11*q11-\
-p12*q12
In [129]: kv1=sk1**(0.5); kv1
Out[129]:
0.158409623366114
In [130]: a11=-2*kv1; a11
Out[130]:
-0.316819246732227
In [131]: a12=2*kv1; a12
Out[131]:
0.316819246732227
In [132]: sk2=q20-p21*q21-\
-p22*q22
In [133]: kv2=sk2**(0.5); kv2
Out[133]: 0.0723179804468125
In [134]: a21=-2*kv2; a21
Out[134]:
-0.144635960893625
In [135]: a22=2*kv2; a22
Out[135]:
0.144635960893625
In [136]: sk3=q30-p31*q31-\
-p32*q32
In [137]: kv3=sk3**(0.5); kv3
Out[137]:
0.484767722953890
In [138]: a31=-2*kv3; a31
Out[138]:
-0.969535445907781
In [139]: a32=2*kv3; a32
Out[139]:
0.969535445907781

```

**Кінець програми**

## Аналіз програми

Почнемо цей аналіз з констатації того, що командами In [4], In [5], In [16], In [17], In [18] в пакеті `numpy` мовою Python ми задаємо границі руйнувань, а командами In [7]—In [15], In [19]—In [21] формуємо самі процеси руйнувань.

Командами In [25]—In [27] формуються перші різниці процесів відновлення з урахуванням руйнувань, а командами In [28]—In [34] здійснюється перевірка реалізацій цих перших різниць на стаціонарність.

Командами In [35]—In [37] формуються другі різниці процесів відновлення з урахуванням руйнувань, а командами In [38]—In [40] здійснюється перевірка реалізацій цих других різниць на стаціонарність.

Командами In [41]—In [47] ми обчислюємо дисперсії реалізацій перших різниць процесів відновлення з урахуванням руйнувань.

Командами In [48]—In [53] здійснюється перший зсув реалізацій перших різниць процесів відновлення з урахуванням руйнувань, а командами In [54]—In [60] ми обчислюються перші коваріації реалізацій перших різниць процесів відновлення з урахуванням руйнувань.

Командами In [61]—In [66] здійснюється другий зсув реалізацій перших різниць процесів відновлення з урахуванням руйнувань, а командами In [67]—In [72] обчислюються другі коваріації реалізацій перших різниць процесів відновлення з урахуванням руйнувань.

Командами In [73]—In [81] обчислюються кореляційні коефіцієнти реалізацій перших різниць процесів відновлення з урахуванням руйнувань, а командами In [82]—In [84] формуються їхні списки. І на цьому етапі ми завершуємо обчислення в пакеті `numpy` і переходимо в пакет `sympy`, в який переносимо усі необхідні нам в подальшому дані командами In [85]—In [105].

Командами In [106]—In [123] обчислюються методом Юла–Уокера коефіцієнти авторегресійних моделей реалізацій перших різниць процесів відновлення з урахуванням руйнувань, а командами In [124]—In [139] формуються діапазони генерації імпульсів білого шуму, які потрібно додавати в авторегресійні моделі за їхнього використання для прогнозування наступних значень перших різниць процесів відновлення.

Підставляючи в рівняння (4) числові значення коефіцієнтів, отримані в результаті виконання команди In [121], отримаємо авторегресійну модель

$$y_1^*[k] = -0,505w_1^*[k-1] - 0,325w_1^*[k-2] + u_1[k], \quad k = 1, 2, \dots, \quad (5)$$

придатну для прогнозування процесу відновлення ГЕС. При цьому рівень імпульсів білого шуму  $u_1[k]$ , які потрібно на кожному кроці підмішувати в модель, потрібно задавати з діапазону значень, визначених командами In [130], In [131].

Підставляючи в рівняння (4) числові значення коефіцієнтів, отримані в результаті виконання команди In [122], отримаємо авторегресійну модель

$$y_2^*[k] = -0,565w_2^*[k-1] - 0,296w_2^*[k-2] + u_2[k], \quad k = 1, 2, \dots, \quad (6)$$

придатну для прогнозування процесу відновлення ГАЕС. При цьому рівень імпульсів білого шуму  $u_2[k]$ , які потрібно на кожному кроці підмішувати в модель, потрібно задавати з діапазону значень, визначених командами In [134], In [135].

Підставляючи в рівняння (4) числові значення коефіцієнтів, отримані в результаті виконання команди In [123], отримаємо авторегресійну модель

$$y_3^*[k] = -0,435w_3^*[k-1] - 0,194w_3^*[k-2] + u_3[k], \quad k = 1, 2, \dots, \quad (7)$$

придатну для прогнозування процесу відновлення СЕС. При цьому рівень імпульсів білого шуму, які потрібно на кожному кроці підмішувати в модель, потрібно задавати з діапазону значень, визначених командами In [138], In [139].

## Висновки

Синтезовані авторегресійні моделі процесів відновлення зруйнованих війною таких сукупностей джерел відновлювальної енергетики, якими є ГЕС, ГАЕС та СЕС.

Як вихідні умови для синтезу авторегресійних моделей використані детерміновані дані, запро-

повані в «Проекті Плану відновлення України. Енергетика» в частині, що стосується електроенергетики, розробленому робочою групою з «Енергетичної безпеки».

Для врахування руйнувань, спричинених війною, детерміновані проєктні умови «зашумлено» білим шумом з діапазонів, визначених рівнем пошкоджень.

Ідентифікація авторегресійних моделей здійснена з використанням метода Юла–Уокера.

В разі іншого рівня руйнувань ВДЕ у порівнянні з тим, що використаний під час створення цієї програми, запропонована програма синтезу авторегресійних моделей залишається дійсною, але в неї командами In [4], In [5] потрібно внести нові рівні руйнувань.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

[1] 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>.

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

[3] Б. І. Мокін, Д. О. Шалагай, і О. В. Мазурук, «Синтез наближених до воєнного сьогодення математичних моделей процесу відновлення та розвитку джерел відновлювальної енергетики України», *Вісник Вінницького політехнічного інституту*, № 1, с. 17-24, 2024. <https://doi.org/10.31649/1997-9266-2024-172-1-17-24>.

[4] *Результати споживання електроенергії за 2022 рік*. [Електронний ресурс]. Режим доступу: <https://www.facebook.com/olhabuslavets/posts/pfbid0jCwzcCNaLCCSHx93MMeRv9KLS7Tn8Fny7wJWwffZSzgoYJq7pCbVgNKz9ttZGwWPI>.

[5] Б. І. Мокін, О. Б. Мокін, і Д. О. Шалагай, «Авторегресійні моделі процесу відновлення та розвитку зруйнованих війною відновлювальних джерел енергетики України», в матеріалах XXIII Міжнародної науково-практичної конференції: «Математичне моделювання та інформаційно-комунікаційні технології для зміцнення та відновлення», Київ, Україна, 12-13 листопада 2024 р., с. 42-46.

[6] 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.

[7] 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.

Рекомендована кафедрою системного аналізу та інформаційних технологій ВНТУ

Стаття надійшла до редакції 8.01.2024

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

**Мокін Олександр Борисович** — д-р техн. наук, професор, професор кафедри системного аналізу та інформаційних технологій, e-mail: abmokin@gmail.com ;

**Мазурук Олег Володимирович** — аспірант кафедри системного аналізу та інформаційних технологій, e-mail: omazuruk3@gmail.com .

Вінницький національний технічний університет, Вінниця;

**Шалагай Дмитро Олександрович** — інженер-проектувальник, e-mail: d.shalagai@gmail.com .

ТОВ «Промавтоматика Вінниця»

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

## **Autoregressive Models of the Process of Post-War Recovery and Post-War Development of the Renewable Sources of Energy in Ukraine**

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

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

*The paper [1] published the “Draft Recovery Plan for Ukraine: Energy,” developed in 2022 by the National Council for the Restoration of Ukraine. In our work [2], to align the “Draft Plan...” with the realities of the military situation, we synthesized mathematical models to represent the restoration and development processes of Ukraine’s electric power industry as a whole, taking into account the stochastic nature of these processes. In our work [3], we further refined the “Draft Plan...” to address the military realities specific to the electric power industry. We developed mathematical models for the restoration and development of renewable electricity sources (RES) in Ukraine. This task was accomplished using various data, tables, and graphs characterizing RES as presented in [1]. However, we expanded the definition of RES beyond solar (SPP), wind (WPP), and biogas (BPP) power plants, as defined in [1], to include hydroelectric (HPP) and pumped storage (PSH) power plants, which were excluded from the RES category in [1]. In [3], we assumed that Ukraine’s war with its aggressive neighbor would continue into 2024 and 2025, introducing a stochastic element into restoration plans—both for pre-war infrastructure and assets restored during the war. We also presumed that the restoration and development of RES would continue to exhibit stochastic characteristics over an extended period. For this reason, we proposed using second-order autoregressive models in the form of ARPKC(2,0,1), starting from the zero time point in 2023. However, in [3], the idea of employing autoregressive models was only mentioned in a general formulation. A year after completing [3], during the ongoing war between Ukraine and Russia, Ukraine’s overall electricity infrastructure and RES, in particular, suffered further destruction [4]. This necessitated the synthesis of autoregressive models in a fully operational form, suitable for use by electrical engineers in design projects. This idea is implemented in the present publication as a Python program. The foundational prerequisites for this program were initially presented in our report at the International Scientific Conference held in Kyiv in November 2024, with extended abstracts published in [5].*

**Keywords:** power industry of Ukraine, renewable energy sources, recovery and development, stochastic process, post-war period, 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 ;

**Mokin Oleksandr B.** — Dr. Sc. (Eng.), Professor, Professor of the Chair of System Analysis and Information Technologies, e-mail: abmokin@gmail.com ;

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

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