УДК 004.42; 004.052

[2]Б. Мандзій, [1]М. Сенів, [1]О. Лобуцька
Національний університет “Львівська політехніка”,
[1]кафедра програмного забезпечення,
[2]кафедра теоретичної радіотехніки та радіовимірювань

ПРОГРАМНА РЕАЛІЗАЦІЯ МОДЕЛІ НАДІЙНОСТІ ТЕХНІЧНОЇ НЕРЕЗЕРВОВАНОЇ СИСТЕМИ З ОБМЕЖЕНОЮ КІЛЬКІСТЮ ВІДНОВЛЕНЬ

Описано алгоритм та здійснено програмну реалізацію моделі надійності технічної нерезервованої системи з обмеженою кількістю відновлень. Розроблене програмне забезпечення (ПЗ) дає змогу аналізувати показники надійності різних варіантів побудови нерезервованої відновлюваної технічної системи з обмеженою кількістю відновлень для різної кількості елементів та кількості відновлень. У ПЗ передбачено можливість візуалізації отриманих результатів обчислень.

Ключові слова: програмне забезпечення, надійність, технічна нерезервована система.

This paper describes an algorithm and software implementation of reliability model of technical nonreserved system with a limited number of restores. The developed software gives the opportunity to analyze the reliability of different options for constructing nonreserved restorable technical system with a limited number of restores for different number of elements and for the number of restores. The software provides the ability to visualize the calculation results.

Key words: software, reliability, technical nonreserved system.

Вступ. В основу розвитку методів та способів дослідження надійності складних технічних систем покладено математичне моделювання. Під математичною моделлю надійності (надійнісною моделлю) технічної системи розуміють такий аналітично чи статистично представлений об’єкт, який відображає властивості системи з погляду надійності, а його дослідження дає повну інформацію про надійнісні характеристики та параметри системи. Важлива роль математичного моделювання полягає у тому, що воно не потребує значних матеріально-технічних витрат і уможливлює ефективне проведення багатоваріантного аналізу, тоді як проведення натурних експериментів з реальними складними системами вимагає величезних матеріальних витрат, наявності дорогої вимірювальної апаратури та значних витрат часу, тому нерідко неприйнятне. На практиці здебільшого моделювання проводять на основі відомих аналітичних методів з використанням експериментальних даних про відмови та відновлення елементів систем. Також застосовують методи імітаційного та статистичного моделювання, які є універсальними, проте у випадках складних і високонадійних систем використання цих методів пов’язане з величезними затратами машинного часу, які роблять їх практично непридатними. Складність сучасних технічних систем зумовлює великі розмірності їх математичних моделей (десятки–сотні тисяч рівнянь), що практично унеможливлює їх формування та аналіз ручними способами. Надійнісні моделі повинні мати високий рівень достовірності та необхідний ступінь формалізації, який уможливлює автоматизацію їх побудови та проведення аналізу надійності з використанням сучасних комп’ютерних засобів. Поєднання аналітичних методів дослідження надійності з обчислювальними можливостями сучасних комп’ютерів є перспективним напрямом подальшого розвитку прикладних

методів теорії надійності й вимагає розроблення відповідного програмного забезпечення, орієнтованого на розв'язання задач моделювання надійності технічних систем. Серед аналітичних методів [1–6] найбільшого поширення набули методи, основані на теорії марковських випадкових процесів [1–4], які уможливлюють опис надійнісної поведінки технічної системи як потоку випадкових подій (відмов та відновлень), що слідують одна за одною у певній послідовності. Це означає, що потоки відмов та відновлень елементів системи розподілені за законом Пуассона, тобто вони мають такі властивості: ординарності (імовірність одночасної появи двох і більше подій дорівнює нулеві), стаціонарності (середня кількість подій за одиницю часу незмінна), відсутності післядії (для двох часових інтервалів, які не перекриваються, кількість подій, що відбулись на одному із них, не залежить від того, скільки подій відбулось на другому). Ці методи дають змогу визначити показники надійності різного типу технічних систем: відновлюваних і невідновлюваних, нерезервованих і резервованих при різних видах резервування, при різних пріоритетах ремонту тощо. Моделювання надійності технічної системи за допомогою марковського випадкового процесу полягає у тому, що дискретним значенням випадкового процесу ставляться у відповідність стани системи (які характеризуються відповідними векторами станів елементів системи). Отже, надійнісну поведінку системи в часі можна інтерпретувати як випадковий процес зі скінченною множиною значень і неперервною зміною аргументу (часу), тобто як дискретно-неперервний стохастичний процес. Часові залежності безумовних ймовірностей значень випадкового процесу та умовних ймовірностей переходів описується системою диференціальних рівнянь Колмогорова–Чепмена виду:

$$\frac{dP_i(t)}{dt} = -\sum_{i=1}^{N} \lambda_{ij}(t) P_i(t) + \sum_{j=1}^{N} \lambda_{ji}(t) P_j(t); \; i, j = 1,2,\cdots N, \qquad (1)$$

де $P_i(t)$, $P_j(t)$ — імовірності значень випадкового процесу (імовірності перебування системи) в момент часу t відповідно у станах x_i та x_j

- $\lambda_{ij}(t)$ — інтенсивність переходу зі стану x_i у стан x_j;
- $\lambda_{ji}(t)$ — інтенсивність переходу зі стану x_j у стан x_i.
- N — кількість можливих станів системи.

Розв'язок рівняння (1) дає змогу визначити потрібні показники надійності (імовірність безвідмовної роботи, функцію та коефіцієнт готовності, середній час безвідмовної роботи, середнє напрацювання на відмову, середній час відновлення тощо), а також побудувати граф станів і переходів системи, який наочно представляє зв'язки між окремими станами. Вимога підвищення ступеня адекватності моделей пов'язана зі збільшенням кількості станів, в яких може перебувати об'єкт проектування, що, своєю чергою, призводить до збільшення розмірності моделей і ускладнює процес аналізу. Розв'язання цього протиріччя можливе шляхом автоматизації процесу формування моделей та їх аналізу. У цій статті розглянуто варіант реалізації програмного забезпечення , призначеного для автоматизованого формування моделі надійності нерезервованої технічної системи з обмеженою кількістю відновлень. Зазначимо, що в літературі [1, 2] описані моделі надійності аналогічних систем, але з необмеженою кількістю відновлень, що практично реалізувати неможливо.

Опис структури та правил функціонування аналізованої системи. Структурна схема надійності технічної системи являє собою послідовне з'єднання N елементів (рис.1), що означає, що система є працездатною лише тоді, коли працездатні усі елементи. При відмові будь-якого елемента система перестає функціонувати, ремонтний орган відновлює елемент, який відмовив, після його відновлення система стає працездатною і продовжує функціонувати. Кожний елемент може бути відновлений лише KR разів. При більшій кількості відмов будь-якого елемента настає катастрофічна відмова системи. Із сказаного випливає, що будь-який елемент системи може перебувати в одному з таких станів:

- F- стан нормального функціонування;
- Rк- стан к-го відновлення (ремонту);
- PF- стан припинення функціонування (простою), спричиненого відмовою іншого елемента;

Рис. 1 Структурна схема надійності системи

Для прикладу розглянемо систему з двох різних елементів для випадку, коли задана допустима кількість відновлень кожного елемента дорівнює 1. При другій відмові будь-якого з елементів настає катастрофічна відмова системи. Граф станів для такої системи зображено на рис. 2.

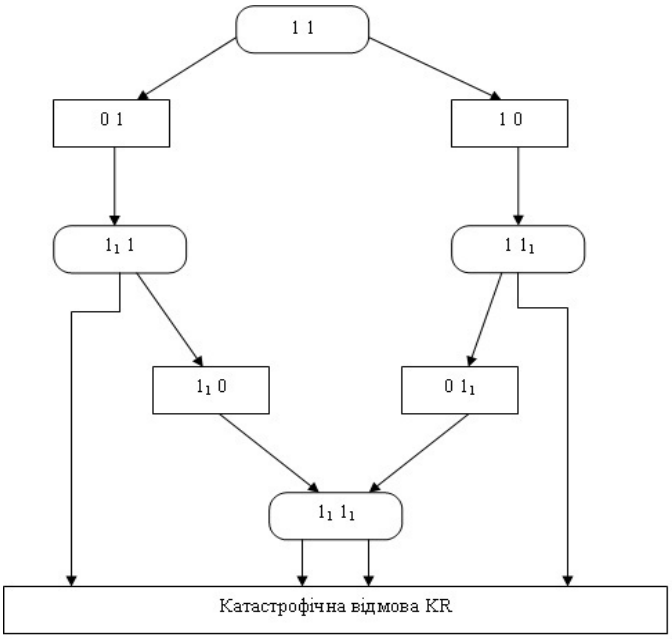

Рис. 2. Граф станів системи із двох елементів

На графі показано такі стани:

(1,1) - початковий стан, обидва елементи працездатні і нормально функціонують (стани елементів відповідно F , F);

(0,1) - перший елемент відновлюється перший раз, другий елемент працездатний в стані простою (стани елементів відповідно R1, PF);

(1,0) - перший елемент працездатний в стані простою, другий елемент відновлюється перший раз (стани елементів відповідно PF, R1);

(1_1,1) – обидва елементи нормально функціонують, перший елемент відновлений один раз (стани елементів відповідно F , F);

(1,1_1) – обидва елементи нормально функціонують, другий елемент відновлений один раз (стани елементів відповідно F , F);

(1_1,0) – перший елемент працездатний після першого відновлення і перебуває в стані простою, другий елемент відновлюється перший раз (стани елементів відповідно PF, R1);

(0,1_1) – перший елемент відновлюється перший раз, другий –працездатний після першого відновлення і перебуває в стані простою (стани елементів відповідно R1, F);

(1_1,1_1) – обидва елементи нормально функціонують, кожен з них відновлений один раз (стани елементів відповідно F , F).

Отже, як бачимо, система має 4 стани працездатності, в яких обидва елементи перебувають в стані нормального функціонування F (за умови, що кількість відновлень кожного з них не перевищує задане значення KR), 4 стани простою, в яких один із елементів в стані відновлення R (за умови, що кількість його відновлень не перевищує задане значення KR), та один стан катастрофічної відмови, в який система попадає, якщо будь-який із елементів відмовив більше ніж задане значення KR.

Програмна реалізація та дослідження тестової моделі надійності системи.

Для програмної реалізації тестової моделі надійності технічної нерезервованої системи було розроблено алгоритм формування графу системи та переходів станів [7]. Вихідними даними для алгоритму є:

- кількість послідовно з'єднаних елементів системи;
- визначена кількість відновлень для кожного елемента;
- правила функціонування системи:
  - ✓ відновлення елемента починається відразу ж після його відмови, при тому система не функціонує, усі інші елементи перебувають у стані простою і тому не можуть відмовляти;
  - ✓ катастрофічна відмова системи настає тоді, коли кількість відновлень будь-якого з елементів вичерпана і саме цей елемент знову відмовив (тобто немає можливості відновити його ще раз);
  - ✓ система продовжує функціонувати після відновлення елемента, який відмовив, якщо кількість його відновлень не перевищує заданого значення .

Виконані дослідження показали, що стани системи "працездатний" та "відновлення" відповідають алгоритмічній рекурсії в два етапи. Оскільки ми на початку припускаємо, що система перебуває в працездатному стані, то алгоритм запускає для неї спершу функцію генерування відмов елементів, після чого для кожної комбінації запускає функцію відновлення, далі знову по черзі функцію відмов і далі функцію відновлення доти, доки є можливість відновлення. На прикладі з трьох елементів це виглядатиме так:

1. Генерування відмови 1-го елемента (інші залишаються справними).
2. Запуск відновлення першого елемента.
3. Генерування відмови 1-го елемента (інші залишаються справними).
4. Переходимо в пункт 3, якщо є ще доступні відновлення, інакше переходимо до генерування відмови 2-го елемента (коли інші залишаються справними).
5. Запуск відновлення 2-го елемента.
6. І так доти, доки є можливість відновлення одного з елементів.

В простому варіанті ми маємо граф станів, у вершині якого є працездатний вихідний стан, далі отримаємо нащадків, кожен з яких має лише один несправний елемент, але у всіх різний. Далі кожен нащадок має лише по одному нащадку, який є справним станом системи, лише з тією відмінністю від першого, що він зберігає інформацію про отримані відмови та хто саме відмовив (який елемент). Для кожного з таких нащадків діємо як на початку, вважаючи їх справними станами і формуючи для кожного набір своїх нащадків з одним несправним елементом, у кожного різним. Для кожного такого несправного нащадка отримуємо справного, з інформацією про те, хто і скільки разів відмовив в цій вітці графа за аналогією з попереднім описом. І так робимо доти, доки не вичерпається можлива кількість відновлень. Очевидно, що чим ближче до листків – отримуватимемо все більше однакових нащадків (однакові нащадки – це ті, в яких показники кількості відмов за кожними окремими елементами однакові), в такому випадку програмно здійснюється об'єднання таких нащадків в один спільний зі збереженням зв'язків із батьківськими об'єктами – отримуємо граф з однією вершиною на початку і однією в кінці.

Покроковий опис алгоритму.

1. Визначення початкової послідовності станів.
2. Для кожного елемента стану виконати "відмову".
3. Виконати відновлення для елемента, що відмовив, якщо можливо, якщо ні – закінчити гілку.
4. Чи вже існує такий самий "справний" стан? Якщо існує, об'єднати з ним. Перейти до кроку 2.
5. Завершення алгоритму після завершення усіх гілок.

Цей граф є набором об'єктів, зв'язаних між собою за посиланнями. Кожен об'єкт такого графу описує певний конкретний стан системи, для якого здійснювалися формування графа та

застосовуються відповідні обчислення. Зв'язки між об'єктами - це представлення реально можливих переходів системи із одного стану в інший. Відсутність зв'язку вказує на те, що згідно з визначеними для цього алгоритму умовами такий перехід є неможливим.

Отриманий граф дає змогу зберігати в собі усю інформацію про конкретний стан системи. Маючи доступ до такої структури даних, можна дізнатися, з якого в який стан може перейти система. Для кращого розуміння процесу формування графу станів, зауважимо, що він формується в декілька етапів. Перший етап – це визначення початкового стану. Після визначення початкового стану він передається у програмну функцію генерації відмови, яка формує каркас графу станів з переходами між станами відповідно до послідовності можливих відмов. В кожен момент часу може відмовити лише один елемент. Коли граф зі зв'язками відмов сформований, для нього викликається функція генерування відновлення.

Ця функція проходить по усіх станах та зв'язує їх з тими, в які чи з яких можливий перехід внаслідок відновлення. Очевидно, що такі переходи є можливими лише до сусідніх рівнів. Останній рівень завжди є рівнем катастрофічної відмови, оскільки включає усі можливі стани непрацездатності системи.

В результаті розробки та досліджень роботи алгоритму виявлено, що з урахуванням всіх станів системи матриця станів та переходів є досить розрідженою. Блок-схему роботи алгоритму подано на рис. 3.

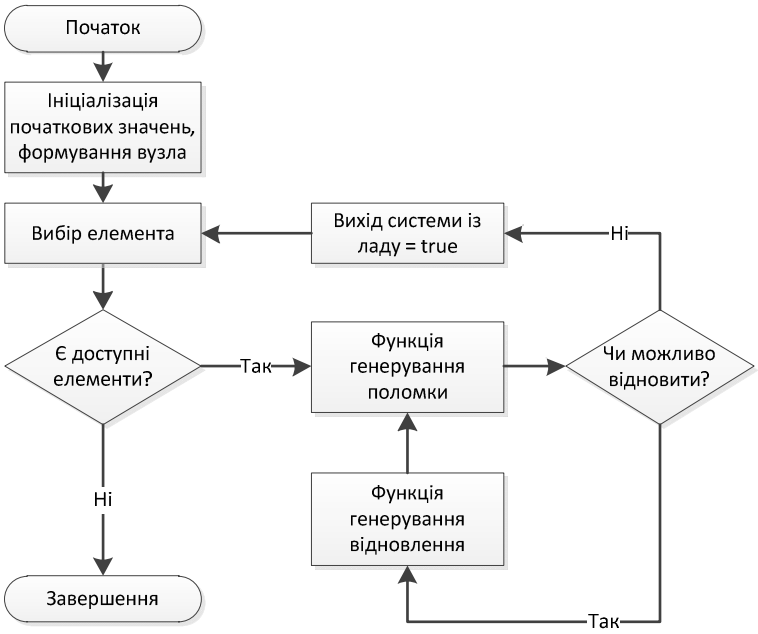

Рис. 3. Блок-схема алгоритму

Для аналізованої системи досліджено, скільки часу витрачено на генерацію та побудову таблиці станів, залежно від кількості елементів у системі та кількості можливих відновлень. Результати досліджень представлено в табл. 1.

Таблиця 1

Часові затрати на аналіз системи

Кількість елементів	Кількість відновлень	Час виконання, с
1	20	0,05
2	7	1
3	3	2
4	2	15
5	1	5

На підставі інформації про стани, в яких може перебувати система, можна програмно сформувати систему диференціальних рівнянь Колмогорова–Чепмена (рис. 4.)

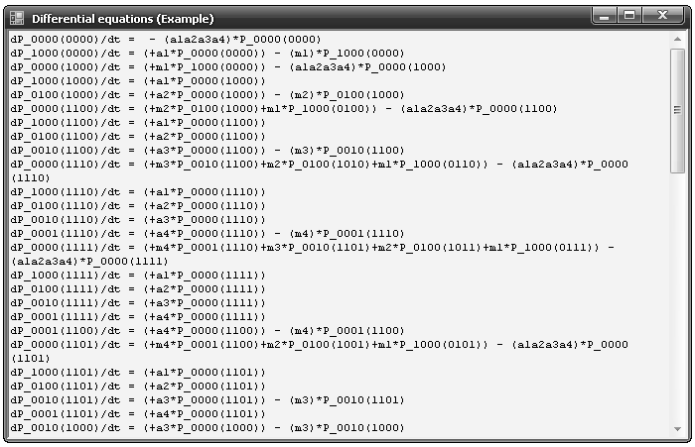

Рис. 4. Вікно автоматизованого формування рівнянь станів системи

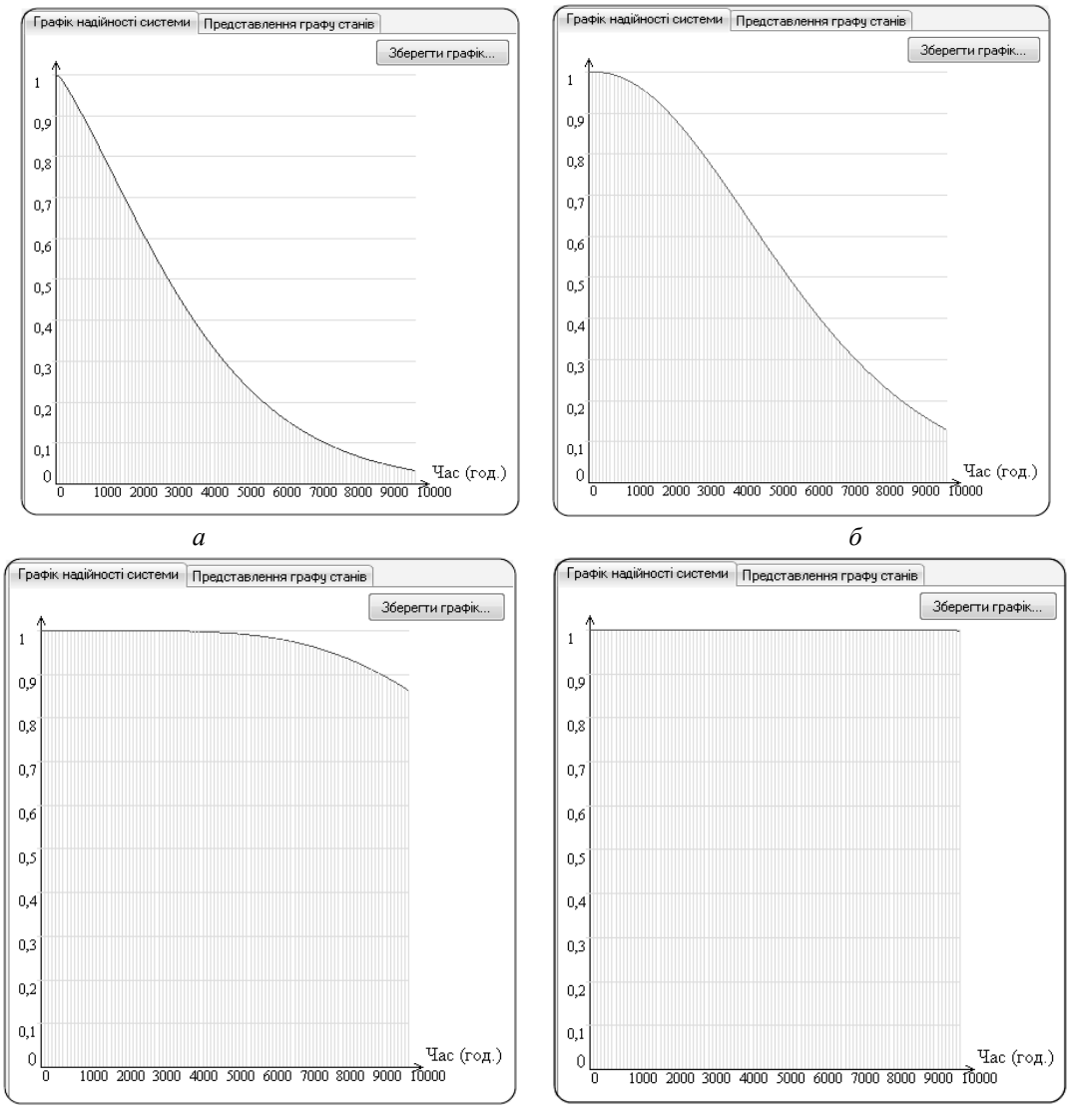

Рис. 5. Часові залежності ймовірності безвідмовної роботи системи для різних варіантів її побудови та відновлення: а – 4 елементи з 1-м відновленням; б – 3 елементи з 2-ма відновленнями; в – 2 елементи з 6-ма відновленнями; г – 2 елементи з 10-ма відновленнями

Для розрахунку часових залежностей ймовірності перебування системи у працездатному стані необхідно розв’язати дану систему рівнянь одним із числових методів. Зрозуміло, що чим більше елементів буде мати система та чим більше відновлень передбачено для кожного елемента системи, тим більше часу буде займати формування системи диференціальних рівнянь та їх розв’язання. Приклади розрахованих часових залежностей імовірності безвідмовної роботи системи подано на рисунках 5(а) –5(г).

Отримані результати розрахунків дають змогу розробникам провести порівняльний аналіз різних варіантів та вибрати кращий за заданим критерієм. Для зручності користувачів, які працюватимуть із програмною системою автоматизованої генерації графа станів та переходів, реалізовано можливість зберігання поточного графіка ймовірності перебування системи в працездатному стані з накладанням на нього наступного. Це дасть змогу наочно порівнювати отримані результати, без зберігання кожного графіка окремо. Приклад такого відображення видно на рис. 6.

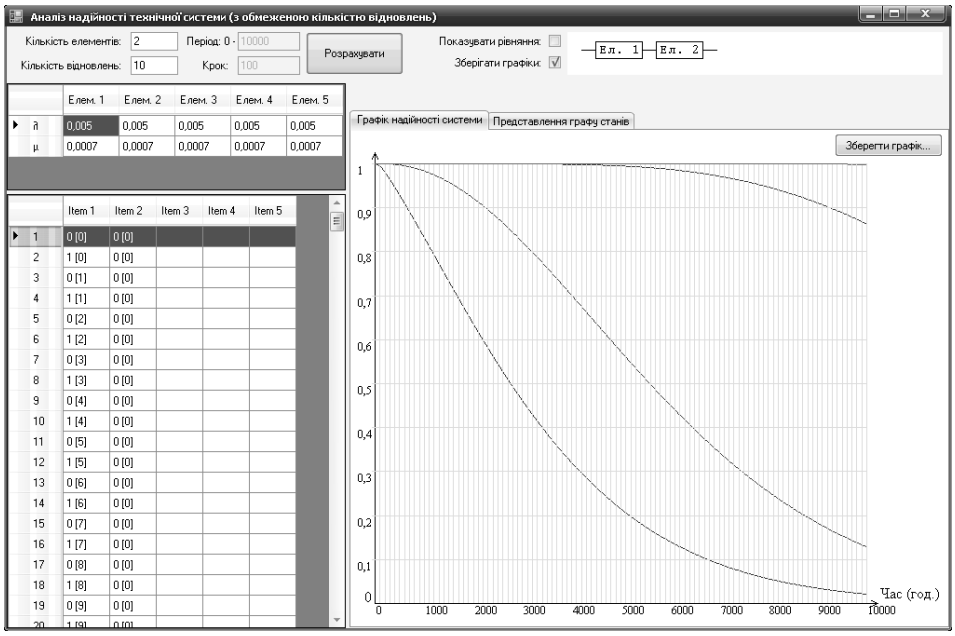

Рис. 6. Відображення поточних та попередніх результатів розрахунків

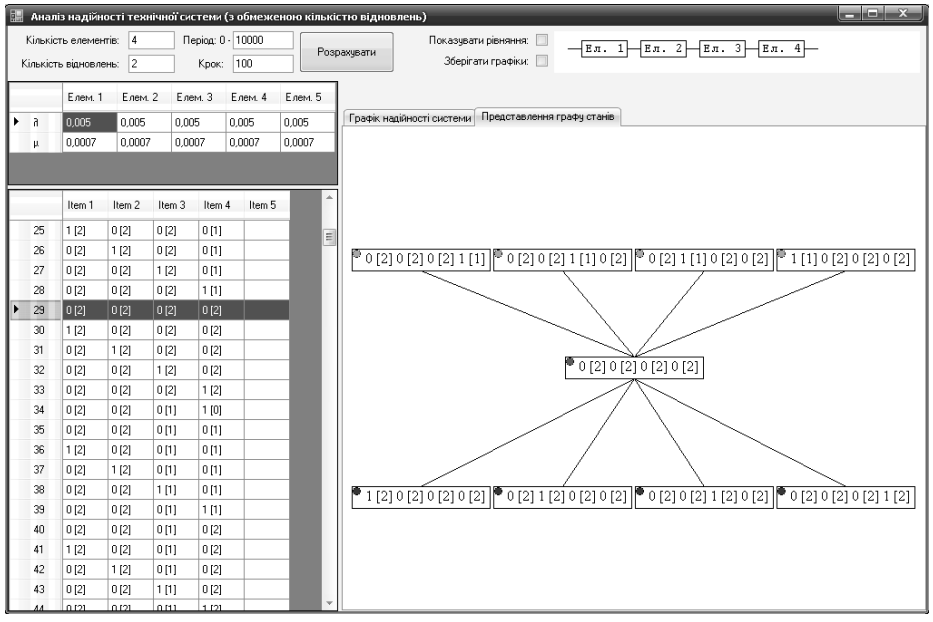

Рис. 7. Візуалізація фрагмента графа станів та переходів

Будь-який з отриманих графіків можна зберегти в форматі *.png, *.jpg або *.bmp, що надає змогу користувачу після проведення тестів зафіксувати отриманий результат, а не проходити його знову.

В системі також передбачено можливість графічного відображення окремих фрагментів графа станів. Оскільки граф складної системи буде дуже загромадженим і незрозумілим із-за великої кількості станів та переходів, передбачено можливість наочно відображати взаємозв'язки між вибраними конкретними станами, користуючись автоматично сформованою таблицею станів. Граф формується для кожного з представлених в таблиці станів. Один з таких випадків представлено на рис. 7.

Висновки

В межах даної роботи програмно реалізовано модель надійності технічної нерезервованої системи з обмеженою кількістю відновлень, що дало змогу автоматизувати генерацію графа станів та переходів моделі та значно зменшити часові затрати на цей процес.

Розроблена програма дає змогу аналізувати показники надійності різних варіантів побудови нерезервованої відновлюваної технічної системи з обмеженою кількістю відновлень для 5 елементів та кількості відновлень - 20.

В ПЗ реалізовано програмний інструментарій для візуалізації обчислень та передбачено функціональну можливість накладання отриманих графічних результатів для їх порівняння. Щоб виключити повторення вже здійснених операцій, передбачено можливість збереження графіків у одному з трьох найбільш поширених графічних форматів (*.png, *.jpg або *.bmp).

1. Половко А.М., Гуров С.В. Основы теории надежности. – СПб.: БХВ-Петербург, 2006. – 704 с. 2. Надежность технических систем: Справочник / Ю.К.Беляев, В.А.Боатырев, В.В.Болотин и др.; Под ред. И.А. Ущакова. - М.: Радио и связь, 1985. – 608 с. 3. Бусленко Н. П., Калашников В.В., Коваленко И. Н. Лекции по теории сложных систем. Учебное пособие. – М.: Советское радио, 1973. – 441 с. 4. Королюк В. С., Турбин А. Ф. Полумарковские процессы и их приложения. – К.: Наукова думка, 1976. – 182 с. 5. Коваленко И. Н. Исследование по анализу надежности сложных систем. – Киев: Наукова думка, 1975, 209 с. 6. Henlej E. J., Kumamoto H. Reliability engineering and risk assessment. – No. 4: Prentice-Hall Inc., 1981. 7. Analytical Model Reliability of Non-redundant Technical Systems with a Limited Number of Repairs / Bohdan Mandziy, Maksym Seniv, Olha Lobutska // Proc. Of the VIIth International and Technical Conference CSIT 2012, Lviv, Ukraine, 20-24 November 2012. – Lviv, 2012. – P.78-80.