



В. С. Яковина, М. М. Сенів, І. І. Симець

Національний університет "Львівська політехніка", м. Львів, Україна

ЗАСОБИ АВТОМАТИЗОВАНОГО ФОРМУЛЮВАННЯ УМОВ ПРАЦЕЗДАТНОСТІ СКЛАДНИХ ТЕХНІЧНИХ СИСТЕМ

Під надійністю техніки розуміють здатність системи або її складових виконувати свої функції за заданих умов та у визначений період часу. Надійність є комплексною характеристикою технічної системи і, відповідно до її призначення, може включати: безвідмовність, довговічність, ремонтпридатність й інші атрибути або їх комбінації. Надійнісний аналіз проводиться з використанням певної кількості засобів графічного моделювання для забезпечення наочності етапів процесу, таких як структурні схеми надійності, дерева відмов та ін. Структурні схеми надійності (англ. *Reliability Block Diagram*) є простим, але наочним методом демонстрації зв'язків між елементами системи за допомогою блоків і з'єднувальних ліній. Моделювання структури технічної системи і визначення характеристик її складових з використанням структурної схеми надійності є потужним та наочним засобом, але складність його зростає відповідно до кількості елементів та вузлів системи. Методи аналізу структурних схем мають експоненціальну складність, яка залежить від кількості елементів та способу їх розміщення. Для вирішення цієї проблеми розроблено засоби автоматизованого формулювання умов працездатності складних технічних систем, які складаються з: алгоритму обходу структурної схеми надійності, алгоритму виявлення послідовного з'єднання та алгоритму виявлення паралельного з'єднання, які виявляють сегменти, котрі з'єднані між собою послідовно та паралельно відповідно. Також подано програмне забезпечення візуалізації структурних схем надійності та автоматизованого формулювання умов працездатності технічних систем, яке дає змогу звільнити користувача від багаторазового виконання рутинних операцій та автоматизувати процес формування вхідних даних для подальшого надійнісного аналізу.

Ключові слова: програмне забезпечення; надійність; структурна схема надійності (ССН).

Вступ. Неперервне ускладнення та підвищення відповідальності функцій, що їх виконують сучасні технічні системи, призводять до ускладнення їхньої структури, внаслідок чого виникає протиріччя між складністю та надійністю системи. Розв'язання цієї суперечності здійснюють способом багатоваріантного математичного моделювання з використанням моделей надійності, під якими розуміють аналітично чи статистично представлені об'єкти, що відображають властивості реальних систем з погляду надійності так, що їх дослідження дає повну інформацію про надійнісні характеристики та параметри цих систем. Складність сучасних технічних систем є причиною великих розмірностей їхніх математичних моделей надійності, яка сягає сотень тисяч рівнянь, що вимагає використання електронно-обчислювальних засобів для їх формування та аналізу. Формування моделі надійності технічної системи починається з побудови структурної схеми надійності у вигляді сукупності певним способом сполучених у сенсі надійності складових частин (DSTU-2861-94, 1994). На

підставі структурної схеми надійності розробник визначає можливі стани системи залежно від станів її складових частин (елементів) і формує умову працездатності чи непрацездатності системи загалом. Можливі різні варіанти формалізованого опису структурної схеми надійності. Більш зручним для є створення графічного зображення структурної схеми надійності, де елементи системи зображені за допомогою геометричної фігури (найчастіше прямокутника) і з'єднують елементи між собою лініями таким способом, щоб структурна схема відображала умову працездатності системи (Polovko & Hurov, 2006; Mandziy et al., 2013) (рис. 1).

Об'єктом дослідження є процес аналізу показників надійності складних технічних систем. *Предмет дослідження* – методи і засоби автоматизованого формулювання умов працездатності складних технічних систем.

Мета роботи полягає в розробленні алгоритмів та програмного забезпечення для автоматизованого формулювання умов працездатності складних технічних систем.

Інформація про авторів:

Яковина Віталій Степанович, д-р техн. наук, професор, завідувач кафедри програмного забезпечення.

Email: vitaliy.s.yakovyna@lpnu.ua; <https://orcid.org/0000-0003-0133-8591>

Сенів Максим Михайлович, канд. техн. наук, доцент, кафедра програмного забезпечення. Email: max1sudden@gmail.com

Симець Іван Ігорович, аспірант, кафедра програмного забезпечення. Email: cemivan1995@gmail.com

Цитування за DSTU: Яковина В. С., Сенів М. М., Симець І. І. Засоби автоматизованого формулювання умов працездатності складних технічних систем. Науковий вісник НЛТУ України. 2019, т. 29, № 9. С. 136–141.

Citation APA: Yakovyna, V. S., Seniv, M. M., & Symets, I. I. (2019). Techniques of automated formulation of the condition of complex technical systems operability. *Scientific Bulletin of UNFU*, 29(9), 136–141. <https://doi.org/10.36930/40290924>

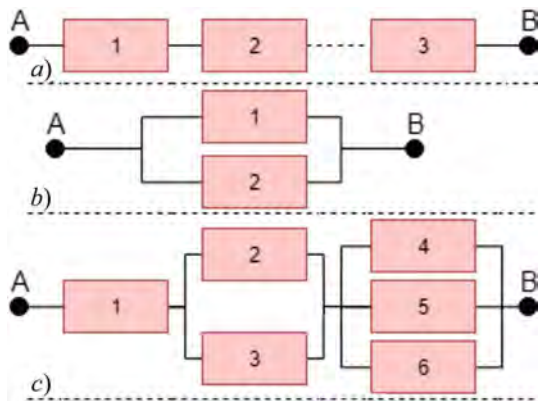


Рис. 1. Структурні схеми надійності простих технічних систем

Основне завдання роботи – розробити алгоритм обходу структурної схеми надійності, алгоритм виявлення послідовного з'єднання та алгоритм виявлення паралельного з'єднання; здійснити програмну реалізацію засобу візуалізації структурних схем надійності та автоматизованого формулювання умов працездатності технічних систем.

Науковою новизною дослідження є алгоритми та методи аналізу структурних схем надійності складних технічних систем.

Практична значущість отриманих результатів: розроблені алгоритми реалізовано у вигляді програмного засобу, який дає можливість автоматизованого формулювання умов працездатності технічних систем. Використання розробленого програмного забезпечення, порівняно з аналогами, дає вигоду за часом близько 23 % при системі, яка складається зі 100 модулів.

Матеріал і методи дослідження. Найзручнішим у практичному використанні є спосіб одночасного визначення умов працездатності системи під час створення графічного зображення її структурної схеми надійності. Умови працездатності системи можна описувати за допомогою функцій алгебри логіки під час створення графічного зображення її структурної схеми надійності. Суть цього способу полягає у такому. Під час аналізу топології схеми спочатку виявляються сегменти схеми, які є послідовним з'єднанням елементів, а відтак розглядається спосіб з'єднання виявлених сегментів між собою. Під час послідовного з'єднання елементи об'єднуються логічною операцією AND, а в разі паралельного з'єднання сегменти об'єднуються логічною операцією OR (Mandziy et al., 2013).

У разі складної структурної схеми це завдання істотно ускладнюється, адже реальні складні технічні системи можуть мати десятки або сотні тисяч елементів. Тому актуальною залишається задача розроблення алгоритмів, які дадуть змогу автоматизувати опрацювання структурних схем надійності та побудувати зображення структурної схеми надійності технічної системи та автоматизованого формулювання умови її працездатності.

Аналіз літературних джерел. У роботі (Mandziy et al., 2013; Bobalo et al., 2014) запропоновано метод створення умов працездатності і відмови за допомогою графічного представлення структурної схеми надійності, але цей метод має низку недоліків. Центральним об'єктом запропонованого підходу є матриця надійності системи. Ця матриця заповнюється на етапі створення структурної схеми надійності, але після завершення створення зміни в матриці стають важкою процедурою, додавання нового елемента або зміна зв'язків у струк-

турній схемі вимагають повної перебудови матриці. Ця операція є часомісткою для систем з великою кількістю елементів.

Для уникнення проблем з використанням матриці надійності в роботі (Mandziy et al., 2015) запропоновано рекурсивний алгоритм обходу структурної схеми надійності для визначення умов відмови і працездатності. Він базується на визначенні послідовних і паралельних з'єднань із чітко визначеними границями розгалужених підсистем (усі вихідні лінії із вузла повинні входити в один кінцевий вузол).

Через свою рекурсивну природу цей алгоритм накладає певні обмеження у побудові структурної схеми надійності і не придатний для визначення умов працездатності для схем певного типу, оскільки не володіє інформацією про повну топологію схеми, а лише будує умову працездатності в процесі обходу схеми. Окрім цього, цей алгоритм коректно працює, якщо у схемі чітко встановлені вузли початку і кінця для паралельних підсистем. Наприклад, у структурній схемі надійності із трьома паралельними підсистемами всі гілки, які виходять з одного вузла, повинні входити в один і той самий вузол, в іншому разі цей алгоритм буде видавати некоректну умову працездатності системи або зациклюватися.

Результати дослідження. Зважаючи на обмеження наведених вище алгоритмів виникла потреба розробити алгоритми, які забезпечували б початковий аналіз топології структурної схеми надійності перед початком формулювання умови її працездатності.

Було розроблено три алгоритми, які за комплексного застосування здійснюють аналіз структурної схеми надійності технічної системи й автоматизують формулювання умови її працездатності. Це такі алгоритми:

1. Алгоритм обходу структурної схеми надійності, який аналізує топологію схеми і виявляє масив сегментів, кожен з яких є послідовним з'єднанням елементів (модулів) або одним елементом;
2. Алгоритм послідовного з'єднання, який виявляє сегменти, котрі з'єднані між собою послідовно;
3. Алгоритм паралельного з'єднання, який виявляє сегменти, котрі з'єднані між собою паралельно.

Нижче розглянуто практичне застосування розроблених алгоритмів для аналізу структурних схем надійності технічних систем. Спочатку необхідно провести аналіз топології і визначення масиву сегментів схеми.

Структурну схему надійності складної технічної системи можна побудувати за допомогою трьох основних графічних компонентів (рис. 2):

- модуль – це графічна одиниця, яка представляє одну із робочих частин системи, яка виконує певні технічні функції;
- лінія – це засіб, який дає змогу встановлювати зв'язок між модулями або вузлами;
- вузол – це проміжний елемент схеми, який потрібен для виконання розгалужень, зображення паралельних підсистем схеми.

Після завершення створення структурної схеми надійності для визначення умови працездатності потрібно отримати інформацію про топологію схеми, її сегменти і модулі та вузли, які вони містять. Це виконується за допомогою алгоритму обходу структурної схеми надійності (блок – схема алгоритм у роботі (Bobalo et al., 2014).

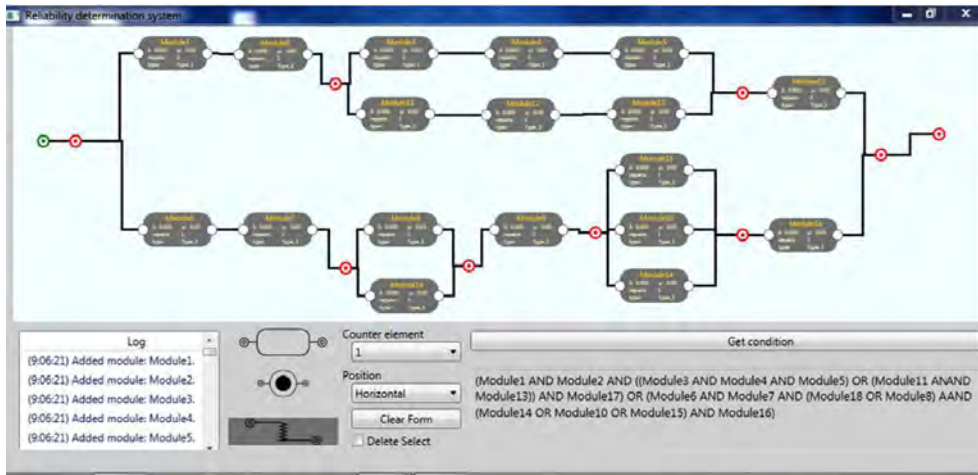


Рис. 2. Графічні компоненти для побудови структурних схем надійності

Цей алгоритм виконує поступовий обхід схеми і внаслідок повинен повернути масив сегментів системи. Сегмент – це складова одиниця структурної схеми надійності, яка описує зв'язок двох сусідніх вузлів схеми і часткову умову працездатності елементів для цього сегмента (сегмент може містити від одного до n-елементів, які з'єднані послідовно). У сегменті міститься інформація про його початковий і кінцевий вузли, модулі і часткову умову працездатності (рис. 3).

Параметри сегмента:
StartNode - поч. вузол
EndNode - кін. вузол
PathString - умова працездатності для цього сегмента
Modulees - список модулів в сегменті
Sequence - порядковий номер сегмента

Рис. 3. Параметри сегмента структурної схеми надійності

Алгоритм складається з таких кроків (при виході отримуємо масив сегментів схеми):

- 1) Починається опрацювання всіх ліній з'єднання схеми.

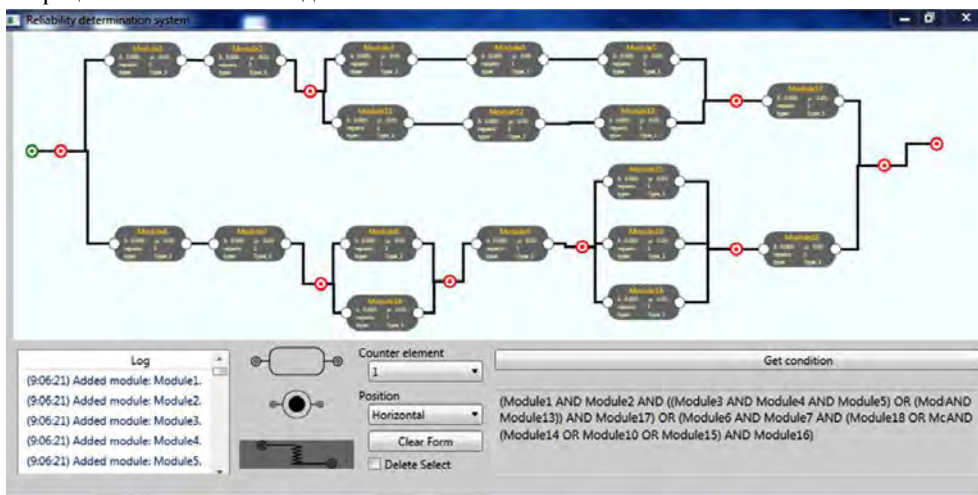


Рис. 4. Частина структурної схеми надійності

Табл. 1. Масив сегментів частини схеми надійності

№	Вузол початку	Вузол кінця	Шлях
1	1	2	Module8
2	1	2	Module18
3	2	3	Module9
4	3	4	Module15
5	3	4	Module10
6	3	4	Module14

- 2) Якщо Element1 і Element2 належать типу Node, додаємо новий сегмент типу Node to Node в масив сегментів, інакше переходимо до наступного кроку.
- 3) Якщо Element1 має тип Node і Element2 має тип Module, додаємо сегмент типу Node to Module із встановленим початковим вузлом сегмента в масив сегментів, інакше переходимо до наступного кроку.
- 4) Якщо Element1 і Element2 належать типу Module, додаємо модуль в сегмент (у список послідовно розташованих модулів цього сегмента) й оновлюємо часткову умову працездатності для цього сегмента, інакше переходимо до наступного кроку.
- 5) Якщо Element1 має тип Module і Element2 має тип Node, додаємо сегмент типу Module to Node із встановленим кінцевим вузлом сегмента в масив сегментів, інакше переходимо до наступного кроку.
- 6) Після завершення побудови сегмента для даної лінії з'єднання схеми – позначаємо її як опрацьовану і переходимо до наступної лінії у схемі.

На рис. 4 зображено частину структурної схеми надійності, а в табл. 1 – масив фрагментів системи.

Саме завдяки застосуванню алгоритму обходу структурної схеми надійності ми й отримуємо декомпозицію схеми (див. рис. 4) у вигляді табл. 1.

Наступними кроками після розбиття топології схеми на сегменти буде послідовне і паралельне з'єднання сегментів та формулювання часткових умов працездатності, поки не залишиться одна єдина умова працездатності (один сегмент масиву).

Якщо два або більше сегментів мають однакові початкові й однакові кінцеві вузли, то такі сегменти вважають паралельними, і тому вони з'єднуються за допомогою алгоритму паралельного з'єднання сегментів (блок – схема алгоритму роботи (Bobalo et al., 2014) із використанням операції OR (АБО). На рис. 5 зображено два паралельних сегменти, які після опрацювання цим алгоритмом будуть з'єднані і їхня часткова умова працездатності виглядатиме так:

$$\text{Module8 OR Module18.} \quad (1)$$

Кроки виконання алгоритму:

- 1) Опрацювання всіх сегментів, які містяться в масиві, отриманому із алгоритму обходу структурної схеми надійності.
- 2) Порівняння поточного сегмента із усіма іншими сегментами.
- 3) Якщо початковий і кінцевий вузли поточного сегмента рівні із початковим і кінцевим сегментами іншого сегмента в масиві, то ці сегменти паралельно розташовані

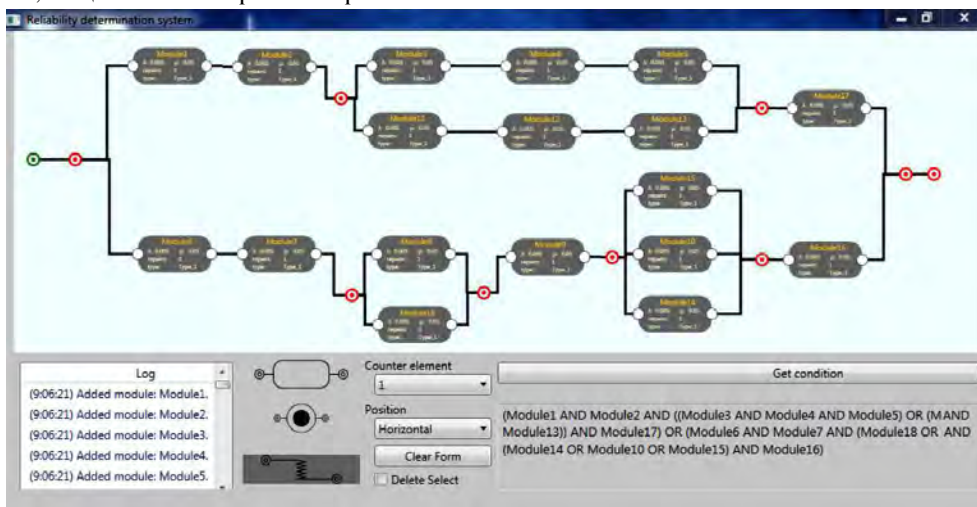


Рис. 5. Паралельні сегменти структурної схеми надійності

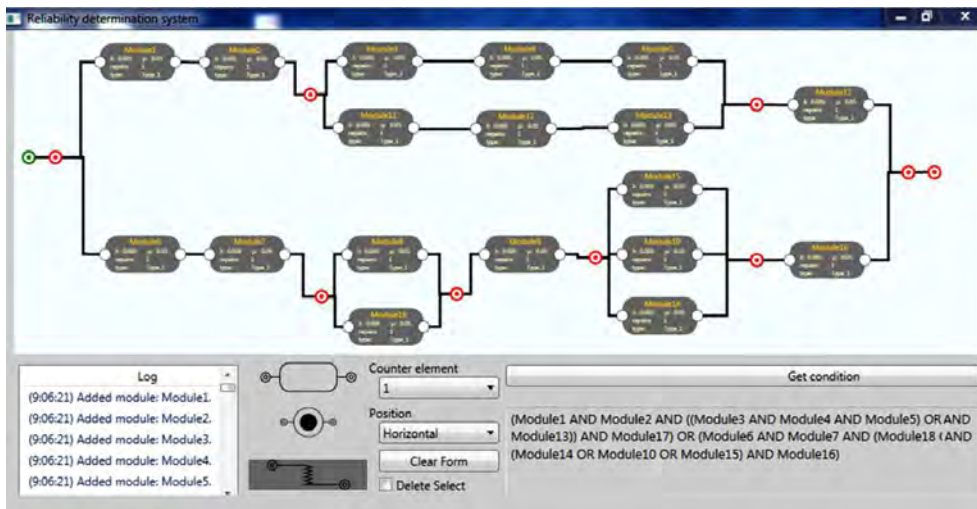


Рис. 6. З'єднання послідовного сегмента структурної схеми надійності

Кроки виконання алгоритму:

- 1) Опрацювання всіх сегментів, які містяться в масиві, отриманому внаслідок роботи алгоритму паралельного з'єднання сегментів.
- 2) Порівняння поточного сегмента із усіма іншими сегментами.
- 3) Якщо кінцевий вузол поточного сегмента є початковим вузлом для іншого сегмента (і цей збіг є тільки одним у всьому масиві сегментів), то ці сегменти з'єднані послі-

і їх часткові умови працездатності можна об'єднати за допомогою операції OR.

- 4) Видалення сегмента, який був паралельно з'єднаний із списку сегментів, оскільки ми вже використали його часткову умову працездатності.
- 5) Продовження пошуку паралельних сегментів.

Після виявлення паралельного з'єднання сегментів системи потрібно використати алгоритм послідовного з'єднання сегментів для того, щоб перевірити, чи в масиві сегментів існує такий сегмент, в якому кінцевий вузол є єдиним кінцевим вузлом у масиві, і сегмент, у якому початковий вузол є єдиним початковим масивом у масиві, тому вони з'єднуються за допомогою алгоритму послідовного з'єднання сегментів (блок – схема алгоритму роботи (Bobalo et al., 2014) із використанням операції AND(I). На рис. 6 зображено продовження схеми, наведеної на рис. 5 з послідовним включенням одного модуля.

довно і їх часткові умови працездатності можна об'єднати за допомогою операції AND.

- 4) Видалення сегмента, який був паралельно з'єднаний із списку сегментів, оскільки ми вже використали його часткову умову працездатності.
- 5) Продовження пошуку послідовних сегментів.

Отже, після виконання паралельного з'єднання фрагментів ми отримали часткову умову працездатності (1) і вузол, що об'єднує паралельну підсистему і Module9 та

задовольняє умову виконання послідовного з'єднання. Одночасно отримуємо часткову умову працездатності: $(Module8 \text{ OR } Module18) \text{ AND } Module9.$ (2)

Якщо в масиві сегментів залишився лише один елемент, то отримано кінцеву умову працездатності для системи, якщо ні, то потрібно повернутись до визначення послідовних і паралельних з'єднань на структурній схемі надійності.

Обговорення та апробація отриманих результатів. Для демонстрації і тестування роботи даних алгоритмів було розроблено модуль візуалізації структурних схем надійності та автоматизованого формулювання умови працездатності технічної системи, який є

складовою частиною програмної системи надійнісного аналізу технічних систем. Цей модуль розроблено із використанням середовища розробки Microsoft Visual Studio, мови програмування C# і технології для створення користувацьких інтерфейсів Windows Presentation Foundation (WPF).

На рис. 7 зображено вікно інтерфейсу засобу для візуалізації структурної схеми надійності і побудови умови працездатності із використанням алгоритмів, описаних вище, в якому міститься графічне зображення структурної схеми надійності технічної системи і внизу справа наведена умова її працездатності у вигляді логічної функції.

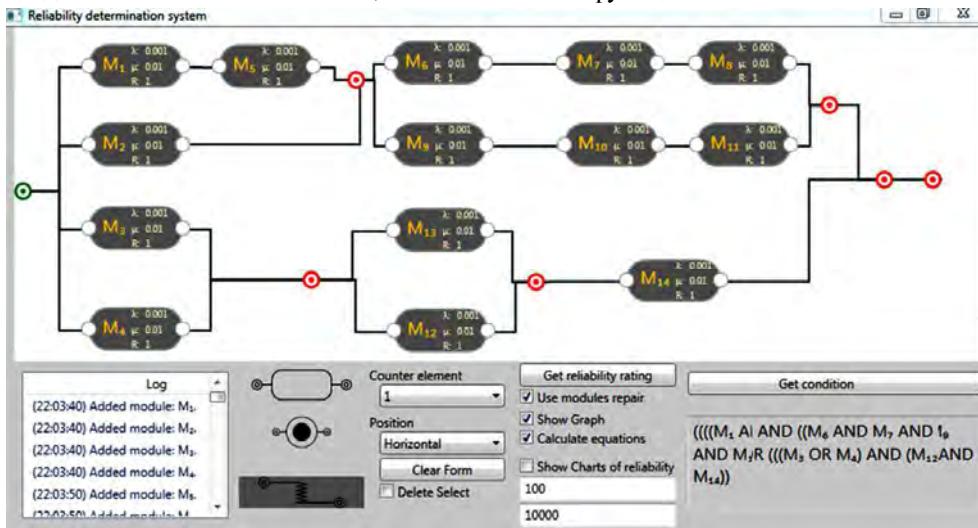


Рис. 7. Вікно інтерфейсу

Дослідили швидкість роботи засобу при паралельному і послідовному сполученнях із різною кількістю модулів у схемі. У табл. 2 наведено агрегацію отриманих даних для об'єктивної оцінки швидкості методу (наведено максимальний, середній і мінімальний час роботи методу при двох типах з'єднання).

Табл. 2. Агреговані значення часу побудови умови працездатності

№	Кількість елементів	Час, мс		
		Max	Min	Ave
1	1	0,042	0,036	0,032
2	2	0,047	0,039	0,035
3	5	0,054	0,047	0,044
4	10	0,083	0,079	0,068
5	20	0,140	0,121	0,098
6	50	0,330	0,283	0,250
7	100	0,650	0,603	0,550

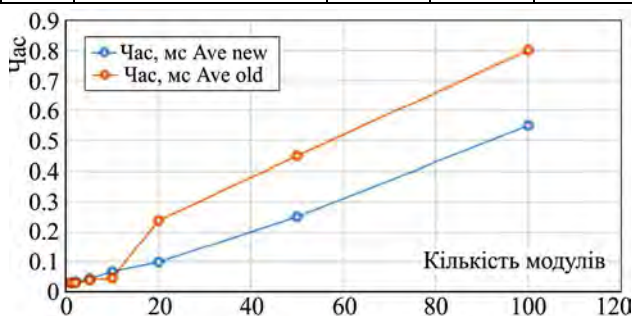


Рис. 8. Порівняння швидкості роботи отриманого алгоритму з рекурсивним алгоритмом

Швидкість розроблених алгоритмів порівняли із швидкістю рекурсивного алгоритму, а результати наведено на рис. 8. Із графіка видно, що за невеликої кількості

модулів рекурсивний алгоритм виконує побудову умови працездатності швидше, це зумовлено тим, що алгоритм, який ми отримали, базується на аналізі всієї схеми і після цього починає будувати умову, а рекурсивний алгоритм це робить у міру обходу схеми. Проте зі збільшенням кількості модулів метод, який базується на аналізі схеми надійності з розбиттям її на сегменти, дає значний вигоду у швидкості порівняно з рекурсивним алгоритмом (Seniv, Mykuliak & Senechko, 2016).

Висновки. У цій роботі представлено алгоритми для автоматизованого формулювання умови працездатності технічних систем:

- алгоритму обходу структурної схеми надійності, який аналізує топологію схеми і виявляє масив сегментів, кожен з яких є послідовним з'єднанням елементів (модулів) або одним елементом;
- алгоритму послідовного з'єднання, який виявляє сегменти, котрі з'єднані між собою послідовно;
- алгоритму паралельного з'єднання, який виявляє сегменти, котрі з'єднані між собою паралельно.

Пропоновані алгоритми дають змогу розробляти програмні засоби для автоматизованого формулювання умови працездатності технічних систем будь-якого рівня складності та опрацьовувати структурні схеми надійності різноманітної конфігурації.

Також представлено програмне забезпечення візуалізації структурних схем надійності та автоматизованого формулювання умов працездатності технічних систем, яке дає змогу звільнити користувача від багаторазового виконання цієї операції та автоматизувати процес формування вхідних даних для подальшого надійнісного аналізу, одночасно виключаючи можливість внесення помилок. Роботу представлених засобів про-

ілюстровано на прикладі структурної схеми надійності технічної системи, яка складається з 14 модулів, та 5 вузлів, що відповідає параметрам реальних технічних систем. Використання цих засобів, порівняно з використанням запрограмованого рекурсивного алгоритму обходу структурної схеми надійності, дає часовий виграш близько 23 % при системі, яка складається зі 100 модулів.

References

- Bobalo, Yu., Seniv, M., Yakovyna, V., & Symets, I. (2014). Method of Reliability Block Diagram Visualization and Automated Construction of Technical System Operability Condition. *Advances in Intelligent Systems and Computing III*, 871, 599–610. Cham: Springer.
- DSTU-2861-94. (1994). Derzhavnyi standart Ukrainy. Nadiinist tekhniki. Analiz nadiinosti. Osnovni polozhennia. [In Ukrainian].
- Mandzii, B., Seniv, M., Yakovyna, V., & Mosondz, N. (2015). Prohramna realizatsiia udoskonalenoii modeli nadiinosti tekhnichnoi rezervovanoi systemy z obmezhenoiu kilkistiu vidnovlen. (Ser. Computer science and information technology). *Bulletin of Lviv Polytechnic National University*, 826, 43–51. [In Ukrainian].
- Mandziy, B., Seniv, M., Mosondz, N., & Sambir, A. (2013). Programming Visualization System of Block Diagram Reliability for Program Complex ASNA-4. *Proceedings of the 13th International Conference The Experience of Designing and Application of CAD Systems in Microelectronics CADSM 2015*, Lviv – Polyana, Ukraine, (pp. 258–262).
- Polovko, A. M., & Hurov, S. V. (2006). *Osnovy teoryi nadezhnosti*. St. Petersburg: BKhV-Peterburh, 704 p. [In Russian].
- Seniv, M., Mykuliak, A., & Senechko, A. (2016). Recursive algorithm of traversing reliability block diagram for creation reliability and refuse logical expressions. *12th International Conference on Perspective Technologies and Methods in MEMS Design*, Lviv, April 20–24, 2016, (pp. 124–129).

V. S. Yakovyna, M. M. Seniv, I. I. Symets

Lviv Polytechnic National University, Lviv, Ukraina

TECHNIQUES OF AUTOMATED FORMULATION OF THE CONDITION OF COMPLEX TECHNICAL SYSTEMS OPERABILITY

What is understood by the reliability of equipment is the ability of a system or its components to perform their functions under specified conditions and for a specified period of time. Reliability is a complex characteristic of a technical system and, accordingly to its intended use, may include as follows: failure-free performance, durability, overhaul and other attributes or combinations thereof. The analysis of reliability is performed with the usage of a number of graphical modeling tools to provide clarity of the process steps such as reliability block diagrams, failure trees, etc. Reliability block diagrams are a simple but clear method of demonstrating links between elements of the system using blocks and connecting lines. Modeling the structure of the technical system and determining the characteristics of its components using reliability block diagram is a powerful and visual tool, but its complexity increases according to the number of elements and components of the system. Block diagram analysis methods have an exponential complexity, which depends on the number of elements and their placement. To solve this problem, tools have been developed for the automated formulation of the operational conditions of complex technical systems, consisting of: algorithm of bypassing the reliability block diagram, algorithm of detection of serial connection and algorithm of detection of parallel connection, which detect segments that are connected in series and in parallel, respectively. The developed algorithms allow for the automatic formulation of the working conditions of technical systems of any level of complexity and elaboration of reliability block diagrams of various configurations. Software for visualization of reliability block diagrams and automated formulation of operating conditions of technical systems is also presented, which allows exempting the user from multiple routine operations and to automate the process of input generation for further reliable analysis, while eliminating the possibility of human mistake factor. The work of the presented tools is illustrated by the example of a reliability block diagram of a technical system consisting of 14 modules and 5 nodes, which corresponds to the parameters of real technical systems. The usage of these tools in comparison with the usage of the programmed recursive algorithm of bypassing the reliability block diagram gives a time gain of about 23 % for the number of modules of the system equal to 100.

Keywords: software; reliability; reliability block diagram (RBD).