

УДК 004.3:004.891.3

Ю.П. КЛЬОЦ, А.С. КОЛІСНИК, Я.М. РУДИЙ

Хмельницький національний університет, Україна

ОЦІНКА ТРУДОМІСТКОСТІ АЛГОРИТМІВ БЕЗСЛОВНИКОВОГО ДІАГНОСТУВАННЯ ЦИФРОВИХ СИСТЕМ

В статті представлено два алгоритми безсловникового діагностування цифрових систем та проведено оцінку їх трудомісткості в порівнянні з класичними методами діагностування. Отримано результати, що свідчать про значно меншу трудомісткість процесу діагностування навіть за умови, коли алгоритм визначення міри підозри виконано і в результаті його виконання отримано однакові ймовірності станів Li. За оптимальних початкових умов діагностування реалізація запропонованого методу, на відміну від класичних, потребує значно меншої кількості циклів моделювання, що в свою чергу дозволяє використати його для діагностування складних цифрових та мікропроцесорних систем.

Ключові слова: тестування, діагностування цифрових систем, безсловникові алгоритми, безсловникове діагностування.

Вступ

При діагностуванні технічного стану цифрової апаратури виникають значні труднощі, що пов'язані з великою розмаїтістю можливих несправностей. Кількість можливих дефектів вже однієї інтегральної схеми (ІС) теоретично визначається комбінаторними числами, порядок яких перевищує реальні можливості сучасних комп'ютерів навіть для їхнього простого перерахування і запам'ятовування. Так, наприклад, якщо комбінаційна схема має n входів, то існує 2^n різних булевих функцій від n змінних і теоретично кожна з цих функцій може відповідати одному з можливих технічних станів ІС. Саме тому при побудові алгоритмів діагностування обмежуються спрощеними і найбільш примітивними моделями несправностей, причому обмежують і їх кратність (у більшості випадків традиційні алгоритми орієнтовані на однократні константні або логічні несправності). [1, 2]

1. Постановка задачі

Дослідження особливостей методів пошуку несправностей цифрових пристроїв показує, що найбільше ускладнень на етапі підготовки вихідних даних для діагностування виникає в ході складання словників несправностей, в яких містяться перелік можливих несправних станів об'єкта діагностування та характерні ознаки прояву несправностей, що дозволяють розрізнити стани. На відміну від класичних методів пошуку несправностей [1, 2] в безсловниковому методі [3, 4] запропоновано відмовитись від словників несправностей. Для порівняння ефек-

тивності безсловникового і класичних методів необхідно оцінити трудомісткість цих методів.

2. Основна частина

Сучасні цифрові системи теоретично можуть мати мільйони і десятки мільйонів несправностей, тобто несправних станів. Справний же стан має бути один. Тому постає питання – чи не краще мати опис (тест-вектор) одного справного стану при виконанні заданих функцій, тобто правильних контрольованих сигналів (відповідних реакцій) на стимульовані сигнали – сигнали, які подають на вхід об'єкта з метою отримання інформації про його технічний стан (тестові впливи).

У такому разі має зберігатися один опис для цифрової системи з “жорсткою” логікою чи набір описів, що відповідає кількості виконуваних функцій програмно-керованою системою.

Головна суть зазначеного підходу полягає у тому, що відпадає необхідність у складанні тест-векторів, котрі моделюють несправності, і тест-векторів, що задають трансляцію несправностей на вихід схеми. А тому і відпадає необхідність у словниках несправностей. Такий метод виявлення несправностей цифрових пристроїв за описом справних станів, які заздалегідь відомі, у кожному мінімальному проміжку часу (такті) не передбачає наявності словників описів несправностей.

Для визначення конкретного компонента, що вийшов з ладу, розроблена велика кількість різноманітних методів [1]. Але майже всі вони орієнтовані на побудову в тому чи іншому вигляді словника несправностей, тому найдоцільніше скористатися запропонованими нижче алгоритмами.

На входи ОД подаємо тестову послідовність X_T , спостерігаючи на виходах реакцію об'єкта Y і фіксуємо момент, коли ця реакція не буде відповідати еталонній (правильній). З цього моменту починаємо моделювати можливі несправності кожного по черзі з компонентів пристрою, інвертуючи (знову також по черзі) його вихідні сигнали та слідкуючи за збіжністю вихідних реакцій. Очевидно, якщо така збіжність станеться, то у нас з'являться деякі підстави вважати, що компонент, несправність якого в даний момент моделюється, і є несправним (зазначимо, що це є лише деякою підозрою, а не переконанням). Продовживши аналогічну процедуру для інших незбіжностей вихідних реакцій, можна сформуванати деякий список підозрюваних в несправності компонентів. Ця процедура, на відміну від відомих, суттєво простіша в реалізації завдяки принаймні двом перевагам [4]:

1) кількість компонентів сучасних цифрових систем не перевищує кількох десятків, тому і відповідна кількість програмних лічильників, що формують список підозрюваних, теж суттєво обмежена;

2) процедуру тестування не обов'язково проводити до кінця, (тобто її можна припинити, як тільки виникне достатня впевненість в достовірності результату діагнозу). Саме ця обставина дозволяє суттєво скоротити тривалість діагностичного експерименту (або час оброблення його результатів).

Загалом процес тестування за безсловниковим методом пошуку несправностей складається з двох етапів

Для реалізації першого етапу запропоновано алгоритм першого етапу:

1.1. Визначити тестову послідовність X для реалізації процесу діагностування об'єкта та множини еталонних значень відповідних реакцій Y^{et} .

1.2. Сформуванати одномірний масив L розмірності n (n – кількість компонентів об'єкта діагностування) з нульовими вихідними значеннями елементів.

1.3. Поставити у відповідність кожному компоненту об'єкта діагностування елемент масиву L .

1.4. Подати на входи об'єкта діагностування тестові впливи та зняти з контрольних точок відповідні реакції.

1.5. Якщо значення отриманих сигналів відповідних реакцій об'єкта діагностування співпадають з еталонними, перейти до п. 1.12.

1.6. Виконати моделювання функціонування об'єкта діагностування та визначити компоненти, які приймали участь у формуванні сигналів відповідних реакцій, значення яких не співпадає з еталонними.

1.7. Прийняти значення змінної I рівним 1.

1.8. Якщо I -тий компонент об'єкта діагностування не приймав участі в формуванні сигналів

відповідних реакцій, значення яких не співпадає з еталонними, перейти до п.1.10.

1.9. Збільшити значення I -го елемента масиву L на 1.

1.10. Збільшити значення змінної I на 1.

1.11. Якщо I менше або дорівнює n , перейти до п.1.8.

1.12. Якщо тестова послідовність X вміщує тестові впливи, які ще не подавалися на об'єкт діагностування, перейти до п.1.4.

1.13. Якщо значення всіх елементів масиву L дорівнює 0, перейти до п.1.16.

1.14. Передати накопичену інформацію для реалізації алгоритму 2.

1.15. Перейти до п.1.17.

1.16. Встановити діагноз "об'єкт діагностування працездатний".

1.17. Кінець алгоритму.

Згідно запропонованого алгоритму першого етапу результатом однократної перевірки буде множина P , яка містить в собі усі підозрювані в несправності компоненти.

До числа підозрюваних в несправності компонентів обов'язково попадуть і ті, через які несправність транспортується до виходів пристрою. Після проведення перевірок, для всіх тестових дій, на яких прийняті хибні відповідні реакції отримаємо m множин підозрюваних в несправності компонентів.

Для випадку однократних несправностей отримані множини необхідно перетнути $M = \bigcap m_i$. Отримана множина M містить несправний компонент, та компоненти, через які несправність транспортується до виходів пристрою. При чому $|M| \leq |K|$.

У випадку кратних несправностей необхідно побудувати масив L $|L|=|K|$ і зіставити кожному елементу масиву компоненту пристрою. У випадку попадання компонента K_j до множини підозрюваних збільшити на одиницю значення i -того елемента масиву.

В результаті виконання першого алгоритму над пристроєм, що містить декілька несправностей, буде отримано масив L з ненульовими елементами. Значення елементів є частотами попадань компонентів в множину підозрюваних. Якщо поділити значення кожного елемента масиву на загальну суму усіх елементів масиву, то отриману величину можна вважати ймовірністю несправності компонента.

Враховуючи те, що час підготовки тестової послідовності та обробки результатів діагностування значно перевищує час подачі тестової послідовності і прийому відповідних реакцій, для оцінки ефективності запропонованого методу використаємо такий параметр, як кількість циклів моделювання. Під циклом моделювання M будемо розуміти процес визначення реакцій схеми при подачі на модель ОД

чергового тестового вектора.

В процесі тестового діагностування несправними можуть бути лише ті компоненти, які приймають участь в генерації помилкових відповідних реакцій, тобто для яких справедливо твердження:

$$\bar{Y}_i^{er} = f(K_j), \text{ при } \bar{X}_i = \text{const}, \bar{Z}_i = \text{const}, \bar{Y}_i^p \neq \bar{Y}_i^{er}, \quad (1)$$

де X – множина тестових дій $X = \{x_1, x_2, \dots, x_i, \dots, x_n\}$, n – кількість тестів

Y^{er} – множина еталонних відповідних реакцій

$$Y^{er} = \{y_i^{er}\}, i=1, n$$

Y^p – множина реальних відповідних реакцій $Y^p = \{y_i^p\}, i=1, n$

Z – множина внутрішніх змінних $Z = \{z_i\}, i=1, n$

K_j – j -й компонент пристрою $j=1, m$, m – кількість компонентів пристрою.

Для визначення підозрюваних компонентів необхідно провести перевірку умови (1) для всіх компонентів на всіх помилкових тестових реакціях.

Звідси отримаємо:

$$M_{\Pi} = n^x \times m,$$

де n^x – кількість тестових впливів, на яких отримано помилкові відповідні реакції; m – кількість компонентів.

Загальна кількість циклів моделювання для реалізації першого алгоритму складе:

$$M_I = n + M_{\Pi} = n + n^x \times m \quad (2)$$

Для реалізації двоетапного методу запропоновано алгоритм другого етапу:

2.1. Прийняти значення змінної I рівним номеру елемента масиву L , який має найбільше значення і не розглядався раніше в процесі моделювання несправностей.

2.2. Для I -того компонента задати множину моделей можливих несправних станів компонента M . $|M|=k$.

2.3. Прийняти значення змінної J рівним 1.

2.4. В моделі об'єкта діагностування замінити модель I -го елемента J -ою моделлю його несправної роботи.

2.5. На отриману модель подати множину тестових впливів X .

2.6. Якщо отримані відповідні реакції співпадають з Y^p – реальними відповідними реакціями об'єкта діагностування, перейти до п. 2.12.

2.7. Збільшити значення змінної J на 1.

2.8. Якщо J менше або дорівнює k , перейти до п. 2.4.

2.9. Якщо в масиві L є ненульові елементи, які не розглядалися раніше в процесі моделювання несправностей, перейти до п. 2.1.

2.10. Передати накопичену інформацію експерту для встановлення діагнозу або визначення подальшого ходу тестових випробувань об'єкта діагностування.

гностування.

2.11. Перейти до п.2.13.

2.12. Встановити діагноз « I -й компонент має несправність, яка визначається j -ою моделлю».

2.13. Кінець алгоритму.

При виконанні другого алгоритму для забезпечення якнайшвидшого знаходження несправності необхідно розпочинати з компонентів, які найбільш віддалені від виходів пристрою.

Проведемо оцінку ефективності алгоритму другого етапу. Вихідними даними для алгоритму ідентифікації несправності є:

T – тестова послідовність $T: \{t_1, t_2, \dots, t_n\}$, де $n=|T|$, що може бути згенерована як детермінованими так і випадковими методами;

R_p – послідовність реальних відповідних реакцій $R: \{r_1, r_2, \dots, r_m\}$, де $n=|T|=|R|$;

$L: \{L_1, L_2, \dots, L_i, \dots, L_k\}$ – множина станів ОД, $i=1, 2, \dots, k$, k – кількість можливих станів ОД, $k=|L|$, при чому L_1 – справний стан ОД, L_2-L_k – несправні стани ОД;

$P(L_i)$ – ймовірність перебування ОД в i -му стані $i=1, 2, \dots, k$.

Для забезпечення встановлення стану ОД впорядкуємо стани L_i згідно умови

$$P(L_{i-1}) \geq P(L_i) \geq P(L_{i+1}), i = \overline{2, k-1}.$$

Кожному стану L_i поставимо у відповідність гіпотезу S_i , згідно якої ОД знаходиться саме в стані L_i . Процес встановлення стану ОД полягає в послідовній перевірці гіпотез S_i і продовжується до моменту підтвердження однієї з цих гіпотез. Цей процес зображено на рис. 1.

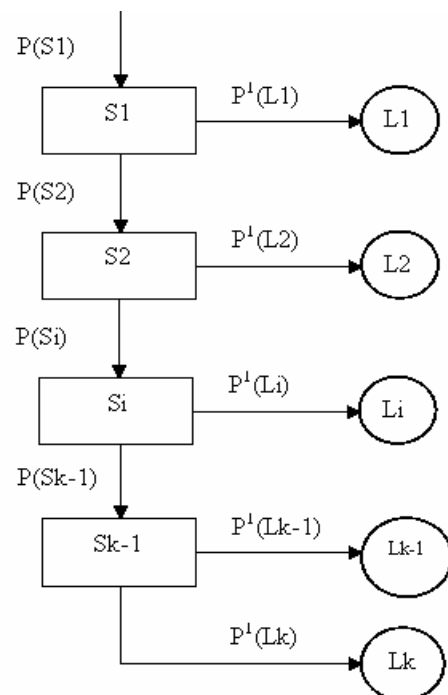


Рис. 1. Процес перевірки гіпотез

На цьому рисунку $P(S_i)$ – ймовірність перевірки i -тої гіпотези, $P(L_i)$ – ймовірність знаходження ОД в L_i -тому стані при умові, що попередньо перевірені гіпотези $S_1 - S_{i-1}$ спростовані.

Враховуючи те, що гіпотези $S_1 - S_{i-1}$ спростовані і події $L_1 - L_{i-1}$ є неможливими маємо:

$$P(L_i) = P(S_i) \frac{P(L_i)}{1 - \sum_{j=1}^{i-1} P(L_j)} \quad (3)$$

Виходячи з того, що події $L_1 - L_{i-1}$ та подія, яка полягає в перевірці гіпотези S_i , є несумісними і складають повну групу подій (сума їх ймовірностей дорівнює 1) маємо:

$$P(S_i) = 1 - \sum_{j=1}^{i-1} P(L_j) \quad (4)$$

Перевірка гіпотези S_i полягає в подачі на модель ОД тестової послідовності T , яка складається з n тестових впливів і порівняння отриманих змодельованих відповідних реакцій R_m з реальними відповідними реакціями R_p . З огляду на те, що події, які полягають в перевірці гіпотез S_i , є випадковими отримаємо середню кількість циклів моделювання, необхідну для реалізації другого алгоритму

$$\bar{M} = \sum_{i=1}^{k-1} n \times P(S_i) = n \times \sum_{i=1}^{k-1} P(S_i)$$

У випадку, коли ймовірності станів L_i невідомі приймаємо $P(L_i) = P(L_j) = 1/k$. В цьому випадку формули (3) та (4) приймуть наступний вигляд:

$$P(L_i) = P(S_i) \frac{P(L_i)}{1 - \sum_{j=1}^{i-1} P(L_j)} =$$

$$= P(S_i) \frac{\frac{1}{k}}{1 - \frac{i-1}{k}} = P(S_i) \frac{1}{k-i+1};$$

$$P(S_i) = 1 - \sum_{j=1}^{i-1} P(S_j) \frac{1}{k-j+1}$$

Звідси маємо:

$$P(S_1) = 1;$$

$$P(S_2) = 1 - 1/k = (k-1)/k;$$

$$P(S_3) = 1 - [(k-1)/k * 1/(k-1) + (k-1)/k * 1/(k-1)] = 1 - 2/k = (k-2)/k;$$

$$P(S_i) = (k-i+1)/k$$

Тому:

$$\bar{M} = n \times \sum_{i=1}^{k-1} P(S_i) = n \times \sum_{i=1}^{k-1} \frac{k-i+1}{k} =$$

$$= \frac{n}{k} \times \sum_{i=1}^{k-1} (k-i+1) = n \times \left(\frac{k-1}{2} - \frac{1}{k} \right)$$

при досить великих k $\bar{M} \rightarrow \frac{n \times k}{2}$

У випадку не рівних ймовірностей станів, які розраховуються при виконанні алгоритму визначення міри підозри (першого алгоритму), \bar{M} ще менше.

Загальна кількість циклів моделювання складе:

$$M = M1 + \bar{M} = n + n^x \times m + \frac{n \times k}{2}$$

Найбільший час виконання першого алгоритму буде за умови $n^x = n$ – всі відповідні реакції хибні. В загальному випадку прийемо $k = m/c$ (кожен компонент має c моделей роботи). Тому загальний час тестування за безсловниковими алгоритмами складе:

$$M = n \left(\frac{2c + k + kc}{2c} \right) \quad (5)$$

За класичними методами для побудови повного словника несправностей необхідно промодельовати роботу ОД у всіх можливих станах на всій послідовності тестових векторів.

Кількість циклів моделювання складе:

$$M_k = c \times n, \quad (6)$$

де n – кількість тестових впливів, необхідна для реалізації одного циклу, котрий відповідатиме одному можливому стану ОД; c – кількість можливих станів всіх компонентів ОД, як справного так і несправних.

Порівнюючи (5) та (6) і враховуючи те, що $c > 1$, $k < c$ та $P(S_i) < 1$, отримаємо, що

$$M < M_k \quad (7)$$

Висновок

З отриманого результату (7) видно, що запропонований метод забезпечує зменшення трудомісткості процесу діагностування навіть за умови, коли алгоритм визначення міри підозри виконано і в результаті його виконання отримано однакові ймовірності станів L_i . За оптимальних початкових умов діагностування реалізація запропонованого методу, на відміну від класичних, потребує значно меншої кількості циклів моделювання, що в свою чергу дозволяє використати його для діагностування складних цифрових та мікропроцесорних систем.

Література

1. Основы технической диагностики / Под ред. П.П. Пархоменко. – М.: Энергия, 1976. – 464 с.
2. Локажук В.М. Контроль і діагностування обчислювальних пристроїв та систем. – Хмельницький: ТУП, 1996. – 175 с.

3. Савченко Ю.Г. Методика та алгоритми діагностування цифрових пристроїв з використанням безсловникових процедур / Ю.Г. Савченко, Ю.П. Кльоц, О.С. Савченко, В. М. Чешиун // Вісник технологічного університету Поділля. – 2001. – № 1. – С. 43-48.

4. Кльоц Ю.П. Моделі несправностей та їх використання в задачах діагностування цифрових пристроїв / Ю.П. Кльоц, В.М. Локазюк, Ю.Г. Савченко // Вісник Хмельницького національного університету. – Хмельницький:ХНУ. – 2005. – № 4, Ч.1, Т.2. – С. 34-39.

Поступила в редакцію 19.02.2010

Рецензент: д.т.н., професор, професор кафедри системного програмування О.В. Поморова, Хмельницький національний університет, Хмельницький.

ОЦЕНКА ТРУДОЕМКОСТИ АЛГОРИТМОВ БЕССЛОВАРНОГО ДИАГНОСТИРОВАНИЯ ЦИФРОВЫХ СИСТЕМ

Ю.П. Кльоц, А.С. Колесник, Я.М. Рудой

В статье представлено два алгоритма бессловарного диагностирования цифровых систем и проведена оценка их трудоемкости в сравнении с классическими методами диагностирования. Получены результаты, которые свидетельствуют о значительно меньшей трудоемкости процесса диагностирования даже при условии, когда алгоритм определения меры подозрения выполнен и в результате его выполнения получены одинаковые вероятности состояний. При оптимальных начальных условиях диагностирования реализация предложенного метода, в отличие от классических, нуждается в значительно меньшем количестве циклов моделирования, что, в свою очередь, позволяет использовать его для диагностирования сложных цифровых и микропроцессорных систем.

Ключевые слова: тестирование, диагностирование цифровых систем, бессловарные алгоритмы, бессловарное диагностирование.

ASSESSMENT LABORIOUSNESS OF THE ALGORITHM NON-DICTIONARY DIAGNOSIS OF DIGITAL SYSTEMS

Y.P. Klyots, A.S. Kolesnik, Y.M. Rudyj

The article presented two algorithms non-dictionary diagnosis of digital systems and an evaluation of their labor in comparison with classical methods of diagnosis. The results, which showed much less complexity of the proposed algorithms. Results which testify to considerably less labour intensiveness of process of diagnosticating even subject to the condition are got, when the algorithm of determination of measure of suspicion is executed and as a result of his implementation identical probabilities of the states are got. At the optimum initial conditions of diagnosticating realization of the offered method, unlike classic, needs considerably less of design cycles, that, in same queue, allows to utilize him for diagnosticating of the difficult digital and microprocessor systems.

Key words: testing, diagnosis of digital systems, non-dictionary algorithms, non-dictionary diagnosis.

Кльоц Юрій Павлович – к.т.н., доцент, доцент кафедр системного програмування, Хмельницький національний університет, Хмельницький, Україна, sprklyots@gmail.com.

Колісник Андрій Сергійович – магістрант кафедри системного програмування, Хмельницький національний університет, Хмельницький, Україна, kism@beta.tup.km.ua,.

Рудий Ярослав Михайлович – магістрант кафедри системного програмування, Хмельницький національний університет, Хмельницький, Україна, kism@beta.tup.km.ua,.