

МЕТОД ПОШУКУ ОПИСУ ОБ'ЄКТІВ ДІАГНОСТУВАННЯ В ІНФОРМАЦІЙНІЙ ДІАГНОСТИЧНІЙ СИСТЕМІ

В статті розглянуто стратегію пошуку та процес генерації навігаційних маршрутів в інформаційній системі при тестовому діагностуванні. Розроблено алгоритми пошуку опису об'єктів діагностування в середовищі інформаційного забезпечення процесу діагностування.

In the article strategy of search and process of generation of navigation routes is considered in the informative system at a test in diagnosticating. The algorithms of search of description of objects of diagnosticating in the environment of the informative providing of process of diagnosticating are developed.

Ключові слова: діагностування, електронно-обчислювальні пристрої.

Вступ

Технологічний процес діагностування мікропроцесорних пристроїв являє собою послідовність технологічних операцій. Він може бути представлений лінійно, циклічно або комбіновано. Специфікою процесу діагностування є суттєві відмінності в діагностуванні МПП на різних етапах життєвого циклу. В роботі [1] запропоновано моделі процесу діагностування на етапах проектування, виробництва та експлуатації МПП. Особливості процесу діагностування накладають певні вимоги на інформаційне забезпечення.

Для визначення конкретної проблемної області необхідно задати склад об'єктів, значення їх характеристик і відношень між об'єктами та характеристиками.

До складу знань, що описують проблемну область, входять безпосередньо знання про предметну область та знання про методи опрацювання цих знань [2].

Склад знань про процес діагностування як предметну область є наступним:

- перелік компонентів ОД;
- особливості, що пов'язані з дефектами партій компонентів ОД;
- множина відомих несправностей;
- множина ознак прояву несправностей;
- множина дій для підтвердження ознаки прояву несправності;
- множина ступені впевненості в наявності зв'язку між деякими параметрами;
- множина суб'єктивних оцінок;
- при наявності неавтоматизованих процесів виробництва необхідно окремо виділити знання про множину дефектів, які вносяться в ОД в процесі ручного монтажу.

Постановка задачі

Дослідження відомих підходів до інформаційного забезпечення процесу діагностування МПП дозволяє зробити висновок, що до теперішнього часу відсутні ефективні, розраховані на широке коло користувачів методи інформаційного забезпечення [1]. Це в значній мірі знижує ефективність процесу діагностування та його достовірність. Тому одним з питань, що потребують вирішення, є розробка комплексу математичної та алгоритмічної підтримки інформаційного забезпечення процесу діагностування МПП [2].

Для забезпечення ефективного доступу до інформації в середовищі інформаційного забезпечення необхідно оптимізувати її пошук. Для розв'язання цієї задачі в статті запропоновано модель стратегії пошуку та генерації навігаційних маршрутів.

Модель стратегії пошуку та генерації навігаційних маршрутів

Для забезпечення ефективного доступу до інформації в середовищі інформаційного забезпечення процесу діагностування необхідно оптимізувати її пошук. Для розв'язання даної задачі розробимо модель стратегії пошуку та генерації навігаційних маршрутів. Формально модель може бути представлена як

$$MSP = \left\langle LD, T, Z, \Psi, I, \delta, \vartheta, \theta, P_{O\Psi}, T_{O\Psi}, P'_{O\Psi}, T'_{O\Psi} \right\rangle,$$

де LD – вхідна мова, яка складається з непустих множин $T, Z, LD = \langle T, Z \rangle$ (словник).

$T = \{t_1, t_2, \dots, t_k\}$ – множина текстів (повідомлень) в діагностичній БД.

$Z = \{z_1, z_2, \dots, z_l\}$ – множина запитів.

$\Psi = \{\psi_1, \psi_2, \dots, \psi_3\}$ – множина пошукових функцій.

I – інформаційна потреба.

θ – функція відображення.

$P_{O\Psi}$ – повнота пошуку по запиту Z . $T_{O\Psi}$ – точність пошуку по запиту Z .

$P'_{O\psi}$ – повнота обслуговування по запиту Z .

$T'_{O\psi}$ – точність обслуговування по запиту Z .

δ – функція відображення релевантного тексту запиту.

\mathcal{G} – функція відображення пертинентного запиту.

Метод пошуку опису об'єктів діагностування

Один крок пошуку тексту можна промодельовати у вигляді часткового відображення множини запитів в множину текстів. Тоді будь-яке часткове відображення $\Psi: T \rightarrow Z$ є пошуковою функцією. Під стратегією пошуку будемо розуміти послідовність кроків, кожен з яких відрізняється від попереднього запитом та/чи пошуковою функцією. При цьому запит та функцію можна коригувати як в режимі діалогу, так і автоматично з використанням даних про об'єм вибірки знайдених текстів.

Якщо $Z_1 \in Z$ – деякі запити, Ψ_i – пошукові функції ($i = 1, l, l \geq 1$), тоді під записом $\{\psi_1(z_1), \psi_2(z_2), \dots, \psi_l(z_l)\}$ – будемо розуміти стратегію пошуку.

Реалізуємо один крок пошуку повідомлень і текстів з використанням наступної вхідної і вихідної інформації. Вхідна інформація – запит $Z \in ZLD$, множина текстів TLV і множина $\omega(TLV)$ їх пошукових образів. Вихідна інформація – множина $\pi(z)$ текстів, знайдених по запиту Z . Сформулюємо запит Z , на основі запиту Z отримаємо повідомлення $a = a_1, a_2, \dots, a_l = Z$, ($l \geq 1$).

Побудуємо синтаксичний граф G повідомлення a . На першому етапі синтаксичного аналізу виявимо в повідомленні всі синтагми. Синтагми є членами деяких синтагматичних структур, які будемо виявляти в аналізованому ланцюжку.

На другому етапі аналізу з всіх синтагм, які ввійшли в трюх і чотириохсловні сигматичні структури, виключимо базові (визначаючі) слова, так як вони не будуть в подальшому впливати на побудову синтаксичного дерева. Ознакою завершення процедури аналізу є одержання визначаючого члена єдиної синтагми.

Реалізуємо процедуру виключення із повідомлення слів, вхідна інформація – повідомлення a і графік D , який включає всі вершини вхідного повідомлення. Вихідна інформація – повідомлення, одержане з початкового видалення деяких слів, і графік створений з вхідного графіка D додаванням до нього деяких дуг. Даний алгоритм включає наступні кроки.

Алгоритм 1

1. Змінній l присвоїти значення, рівне числу слів в повідомленні a і пронумерувати їх в порядку зростання.

2. Якщо $l = 1$, то синтаксичне дерево побудоване, інакше перейти до кроку 3.

3. Якщо в графіку D є дуга (a_2, a_1) , видалити з повідомлення слово a_1 і перейти до кроку 4.

4. Змінній $c := a_1$, $d := a_2$. Змінній $k = 12$, якщо c – визначаючий її член, а d – визначаємий з'єднати вершини a_1 і a_2 дугою (a_1, a_2) і перейти до наступного кроку. Змінній $k = 12$, якщо d – визначаючий, а c – визначаємий член ланцюжка cd , з'єднати вершини a_1 і a_2 дугою (a_2, a_1) , видалити з повідомлення слово a_1 і повернутись до кроку 1. Якщо ланцюжок cd не є синтагмою змінній $k = \phi$ і перейти до кроку 5.

5. Якщо в графіку D є дуга (a_{l-1}, a_l) , то видалити з повідомлення слово a_l і повернутись до кроку 1, інакше – до кроку 6.

6. Змінній $c := a_{l-1}$, $d := a_l$. Змінній $k = 12$, якщо c – визначаючий член послідовності cd , з'єднати вершини a_{l-1} і a_l дугою (a_{l-1}, a_l) , видалити з повідомлення слово a_l і повернутись до кроку 1. Змінній $k = 21$, якщо d – визначаючий член послідовності cd , а c – визначаємий, з'єднати вершини a_{l-1} і a_l дугою (a_l, a_{l-1}) і перейти до кроку 7. Змінній $k = \phi$, якщо послідовність cd не є синтагмою і перейти до кроку 7.

7. Кінець алгоритму 1.

На наступному кроці побудови синтаксичного графіка реалізуємо процес пошуку всіх синтагм в повідомленні a і графіку D , вихідна інформація – повідомлення a і графік D , доповнений деякими дугами. Алгоритм, реалізуючий дану процедуру, наступний.

Алгоритм 2

1. Змінній l присвоїти значення, рівне числу слів в повідомленні a і пронумерувати їх в порядку зростання числами $1, 2, \dots, l$.

2. Якщо $l = 1$, то синтаксичне дерево побудоване, інакше перейти до кроку 3.

3. $i := \phi$.

4. $i := i + 1$.

5. Якщо в графіку D є дуга (a_i, a_{i+1}) та дуга (a_{i+1}, a_i) , то перейти до кроку 7, інакше до кроку 8.
6. Змінній $c := a_i$, $d := a_{i+1}$. Визначити значення k (див. Алгоритм 1).
7. Якщо $i + 1 = l$, то перейти до кроку 9, інакше – до кроку 4.
8. Якщо $k = 12$, то з'єднати вершини a_i і a_{i+1} графіка D дугою (a_{i+1}, a_i) . Перейти до кроку 7.
9. $i := 1$
10. $i := i + 1$
11. Якщо $i = l - 1$, то перейти до кроку 12. Інакше перевірити присутність в графіку D одночасно дуги (a_{i+1}, a_i) та (a_{i-1}, a_i) . Якщо так, то видалити з графіка D дугу (a_{i+1}, a_i) і перейти на крок 10, інакше перейти до кроку 1.
12. Кінець алгоритму 2.

Наступний крок побудови синтаксичного дерева складається в знаходженні синтагматичних структур типу $\overleftarrow{x(\overleftarrow{xx})}$ і $\overleftarrow{x(\overleftarrow{xx})}$ в повідомлення a . Вхідна інформація – повідомлення a і графік D , вихідна інформація – повідомлення, отримане з a видаленням базових слів в синтагматичних структурах типу $\overleftarrow{x(\overleftarrow{xx})}$ і $\overleftarrow{x(\overleftarrow{xx})}$ і графік, отриманий з D додаванням в нього деяких дуг. Алгоритм, реалізуючий дану процедуру, включає наступні кроки.

Алгоритм 3

1. Змінній l присвоїти значення, рівне числу слів в повідомленні a і пронумерувати їх в порядку зростання числами $1, 2, \dots, l$
2. Якщо $l = 1$, то синтаксичне дерево побудоване, інакше перейти до кроку 3.
3. Якщо $l = 2$, і в графіку D є дуга (a_2, a_1) чи (a_1, a_2) , то синтаксичне дерево побудоване. Якщо $l = 2$, і вершини a_1 і a_2 не з'єднані дугою (в будь-якому напрямку), то вхідне повідомлення не є проєктивним, інакше перейти до кроку 4.
4. $i := \phi$.
5. $i := i + 1$.
6. Якщо $l = 2$, і в графіку D є дуга (a_{i+2}, a_{i+1}) , то перейти на крок 8.
7. Змінній $c := a_i$, $d := a_{i+2}$. Визначити значення k (див. Алгоритм 1). Якщо $k = \phi$, то видалити з повідомлення слово a_{i+1} , то з'єднати вершини a_i і a_{i+2} графіка D дугою (a_i, a_{i+2}) , при $k = 12$, чи дугою (a_{i+2}, a_i) при $k = 21$ і перейти до кроку 1. Якщо $k = \phi$, то перейти до кроку 8.
8. Якщо $i + 2 = l$, то перейти до кроку 9, інакше перейти до кроку 5.
9. Кінець алгоритму 3.

Алгоритм 4

Реалізуємо програму синтагматичних структур з трьох слів типу $\overleftarrow{(xx)x}$ і $\overleftarrow{(xx)x}$ повідомленні a . Вхідна інформація – повідомлення a і графік D . Вхідна інформація – повідомлення a і графік D , вихідна інформація – повідомлення, отримане з a видаленням деяких вершин і графік, отриманий з D додаванням в нього деяких дуг. Алгоритм 4, реалізація даної процедури співпадає з алгоритмом 3, крім кроку 6. В алгоритмі 4 цей крок буде мати наступний вигляд:

“Якщо $l = 2$, і в графіку D є дуга (a_i, a_{i+1}) , то перейти до кроку 7, інакше – до кроку 8.”

На наступному кроці побудови синтаксичного графіка необхідно реалізувати процедуру пошуку синтагм типу $\overrightarrow{xx} | \overleftarrow{xx}$, вхідна інформація – повідомлення a з видаленням деяких слів і графік, D з додаванням в нього деяких дуг. Для пошуку всіх синтагм даного типу і побудови синтаксичного графіка виконаємо наступний алгоритм.

Алгоритм 5

1. Змінній l присвоїти значення, рівне числу слів в повідомленні a .
2. Якщо $l = 1$, то синтаксичне дерево побудоване, інакше перейти до кроку 3.
3. Якщо $l = 2$ в графіку D є дуга (a_1, a_2) чи дуга (a_2, a_1) , то синтаксичне дерево побудоване. Якщо $l = 2$ і вершини a_1 і a_2 не з'єднані дугою (в будь-якому напрямку), то повідомлення a не є проєктивним, інакше перейти до кроку 4.
4. $i := \phi$
5. $i := i + 1$
6. Якщо в графіку D є дуга (a_i, a_{i+1}) , то перейти на крок 7, інакше на крок 9.

7. $i = j + 2$. Якщо $j = l$, то виключити з повідомлення слово a_{i+1} і перейти на крок 1, інакше на крок 8.

8. $i = j + 1$, $c := a_i + 1$, $d := a_j$. Визначити значення k (див. Алгоритм 1). Якщо $k = \phi$, то перейти до кроку 7, інакше до кроку 5.

9. Кінець алгоритму 5.

На наступному кроці побудови синтаксичного графіка необхідно реалізувати процедуру пошуку базових чотирьохслівних синтагматичних структур типу $(\overline{xx})(\overline{xx})$ і $(\overline{xx})(\overline{xx})$. Вхідна і вихідна інформація – повідомлення a і графік D . Алгоритм, реалізуючий дану процедуру, складається з наступних кроків.

Алгоритм 6

1. Змінній l присвоїти значення, рівне числу слів в повідомленні a і пронумерувати їх в порядку зростання числами $1, 2, \dots, l$

2. Якщо $l = 1$, то синтаксичне дерево побудоване, інакше перейти до кроку 3.

3. Якщо $l = 2$ в графіку D є дуга (a_1, a_2) чи дуга (a_2, a_1) , то синтаксичне дерево побудоване.

Якщо $l = 2$ і вершини a_1 і a_2 не з'єднані дугою (в будь-якому напрямку), то повідомлення a не є проєктивним, інакше перейти до кроку 4.

4. Якщо $l = 3$, то перейти до кроку 10, інакше – до кроку 5.

5. $i := \phi$

6. $i := i + 1$

7. Якщо в графіку D є дуги (a_i, a_{i+1}) , (a_{i+3}, a_{i+2}) , і немає дуг (a_{i+1}, a_{i+2}) , (a_{i+2}, a_{i+1}) , перейти на крок 9.

8. Змінній $c := a_i$, $d := a_{i+3}$. Визначити значення k (див. Алгоритм 1). Якщо $k = \phi$, то перейти до кроку 9, інакше виключити з повідомлення a слова a_{i+1} і a_{i+2} , доповнити графік D дугою (a_i, a_{i+3}) при $k = 12$, чи дугою (a_{i+3}, a_i) при $k = 21$ і перейти до кроку 1.

9. Якщо $i + 3 = l$, то перейти до кроку 10, інакше до кроку 6.

10. Кінець.

Даний підхід не тільки реалізує побудову синтаксичного графіка проєктованих повідомлень, але і здійснює контроль вхідних повідомлень на проєктивність.

Після побудови синтаксичного графіка виконуємо наступний алгоритм.

Алгоритм 7

1. Змінній k присвоїти значення, рівне числу всіх орланцюжків $\varepsilon(z) = \{b_1, b_2, \dots, b_n\}$ довжини l в дереві D і пронумерувати їх числами від 1 до k , тобто представити графік D у вигляді множини $C = \{C_1, C_2, \dots, C_k\}$.

2. Сформувати пошукове розпорядження помістивши в множину $b_j (j = 1, k)$ орланцюжок c_j і всі орланцюжки довжиною 1, яким відповідають повідомлення внутрішньої мови представлені орланцюжком c_j

3. В множині $\omega(TLV)$ знайти сукупність всіх пошукових образів текстів, задовольняючих співвідношення (3), а в множині TLV – сукупність текстів $\Pi(z)$.

4. Кінець.

Якщо необхідно корегувати запит, то виконати алгоритм 8. При виконанні алгоритму корегується запит користувача. Вхідна інформація – множина $\Pi(z)$ текстів, знайдених по запиту $Z \in ZLV$. Вихідна інформація – відкоригований запит. Алгоритм включає наступні кроки.

Алгоритм 8

1. Якщо $\Pi(z) = \phi$, то перейти на крок 5, інакше – на крок 2.

2. $TLV : TLV \setminus \Pi(z)$

3. Видати користувачу текст $t \in \Pi(z)$, для якого значення функції відповідності мінімальне. Корегувати запит. Якщо корегування закінчено, відновити множину TLV і перейти до кроку 7, інакше перейти до кроку 4.

4. $\Pi(z) := \Pi(z) \setminus \{t\}$ Якщо $\Pi(z) \neq \phi$, то перейти на крок 3, інакше – на крок 5.

5. Якщо $\varepsilon(z) = 0$, то відновити множину TLV і перейти на крок 7, інакше виключити з пошукового розпорядження $\varepsilon(z) = \{b_1, b_2, \dots, b_k\}$ ($k \geq 1$). Спочатку по одній множині $b_j (j = 1, l)$, потім по дві і т.д. і впорядкувати їх у вказаній послідовності, утворивши кортеж розпоряджень $p' = \langle p'_1, p'_2, \dots, p'_n \rangle$ ($n \geq 1$)

6. Якщо $p' = \emptyset$, то $\Pi(z) := TLV$, перейти на крок 1, інакше в множині $\omega(TLV)$ знайти сукупність всіх пошукових образів текстів по пошуковому розпорядженню p_1' . Виключити p_1' з множини p' пронумерувати розпорядження, які залишились числами 1,2,... Якщо хоч би один текст знайдено, позначити їх множину через $\Pi(z)$ і перейти на крок 2, інакше на крок 6.

7. Кінець алгоритму.

Висновки.

Розглянутий метод пошуку призначений для комплексного підвищення ефективності діагностування сучасних електронно-обчислювальних пристроїв і систем. Він використовується в діагностичних інформаційних системах при пошуку опису тексто-графічної інформації про об'єкти діагностування і властиві їм несправності.

Література

1. Муляр І. В., Джулій В. М. Представлення моделі ООБД інформаційного забезпечення тестового діагностування мікропроцесорних пристроїв // Вісник ТУП. – 2004. – № 2. Частина 1, том 1 – С.79– 83.
2. Муляр І.В., Джулій В.М. Інформаційні процеси та моделі їх представлення при тестовому комбінованому діагностуванні // Вимірювальна та обчислювальна техніка в технологічних процесах. – 1999. – № 4.
3. Муляр І.В., Джулій В.М. Гіпертекстова модель представлення інформації в базах діагностичних даних. // Вісник ТУП. – 2001. – № 1. – С.189– 191.
4. Локазюк В. М, Муляр І. В., Джулій В. М. Архітектура автоматизованої системи інтелектуалізації баз даних процесу тестового комбінованого діагностування // Вісник ТУП. – 2002. – № 3. – С.97– 100.

Надійшла 9.9.2009 р.

УДК 004.056

О.А. МЯСІЩЕВ, А.В. ДЖУЛІЙ
Хмельницький національний університет

НАПРЯМКИ ВИРІШЕННЯ ПРОБЛЕМ ЗАХИСТУ ІНФОРМАЦІЇ В МЕРЕЖАХ

У даній статті розглядаються концепції безпеки захисту інформації в мережах, основні етапи забезпечення безпеки. Система інформаційної безпеки підприємства повинна враховувати появу нових технологій і сервісів, а також задовольняти загальним вимогам, поставленим сьогодні до будь-яких елементів корпоративної інформаційної системи. Наявність централізованих засобів керування продуктами безпеки є обов'язковою умовою для можливості їхнього застосування в корпоративному масштабі.

In this article conceptions of safety and defense of information are examined in networks, basic stages of providing of safety. The system of informative safety of enterprise must take into account appearance of new technologies and services, and also satisfy general requirements put today to any elements of the corporate informative system Presence the centralized facilities of management safety products is necessary condition for possibility of their application in a corporate scale.

Ключові слова: захист інформації в мережах.

Вступ. Сучасні системи захисту інформації повинні відповідати запитам сучасного бізнесу в умовах росту числа погроз безпеки інформації, що виходять із самої корпоративної мережі. Усе більше співробітників змушені працювати із чутливими ресурсами інформаційної системи, перебуваючи поза її межами. Неухильно зростає швидкість появи нових корпоративних додатків. Збільшуються масштаби й складність мереж. Із цих причин традиційна концепція побудови систем безпеки корпоративної мережі у вигляді захисту тільки зовнішніх каналів зв'язку офіс-офіс (LAN-LAN) уже недостатня. Сучасні системи безпеки повинні захищати не окремі елементи мережі, а інформацію у вигляді інформаційних ресурсів і потоків незалежно від місця й часу їхнього виникнення [1].

Концепція інформаційної безпеки. Інформаційна безпека є складовою частиною інформаційних технологій – області, що розвивається надзвичайно високими темпами. Розробка сучасної системи інформаційної безпеки вимагає, з одного боку, відстеження швидких змін в інформаційних технологіях і погрозах, що з'являються, а з іншого боку – обліку реальних характеристик апаратного й програмного забезпечення корпоративних мереж і систем. Процедура придбання пристроїв інформаційної безпеки не складна. Істотно більш складним є рішення проблем: як захищати і які засоби безпеки застосовувати? Це рішення охоплює й керування інформаційною безпекою, включаючи планування, розробку політики безпеки й проектування необхідних процедур безпеки [1, 2].