

РЕАЛІЗАЦІЯ ІНТЕРФЕЙСНОГО АГЕНТА МУЛЬТИАГЕНТНОЇ СИСТЕМИ ДІАГНОСТУВАННЯ ПЕРСОНАЛЬНОГО КОМП'ЮТЕРА

Хмельницький національний університет

Представлено опис способу взаємодії користувача з мультиагентною системою діагностування персонального комп'ютера. Описано структуру запитів, відповідей та інформувань системи діагностування. Наведено продукційні правила, що визначають сценарій опрацювання запитів/відповідей. Представлено приклади діалогу системи діагностування і користувача

Вступ

Однією з основних задач, яка виникає на етапах комплектування і експлуатації персональних комп'ютерів, є забезпечення оптимальних температурних режимів компонентів персональних комп'ютерів (ПК) шляхом підбору необхідної конфігурації за певних умов функціонування. Недотримання оптимальних температурних режимів є одним із чинників, що призводить до передчасного виходу з ладу компонентів ПК та знижує їх надійність [1 - 3].

Впровадження мультиагентної системи діагностування (МАСД) забезпечує підвищення надійності функціонування ПК [4]. Необхідність автономності функціонування системи діагностування та розподілу задач діагностування вимагає використання агентних технологій, які дозволяють децентралізувати збір, опрацювання та аналіз діагностичної інформації.

Постановка задачі

Однією з задач, яка вирішується агентною системою діагностування, є задача організації взаємодії МАСД з користувачем. Відомі реалізації агентних систем узагальнено описують взаємодію агентів з користувачем, не приділяючи уваги методам взаємодії агентної системи з користувачем та реалізації інтерфейсу користувача [5 - 8].

Метою статті є представлення реалізації інтерфейсного агента, основною функцією якого є організація взаємодії користувача і мультиагентної системи діагностування ПК.

Взаємодія користувача і МАСД

Обмін інформацією між системою діагностування і користувачем здійснюється на основі запитів/відповідей користувача, запитів/відповідей системи діагностування та інформувань користувача.

Запити/відповіді МАСД надаються користувачу вербальною мовою, що підвищує їх зрозумілість і знижує рівень вимог до кваліфікації користувача, необхідний для роботи з системою.

Взаємодія інтерфейсного агента з іншими агентами системи діагностування здійснюється за допомогою передачі повідомлень. Структура повідомлень стандартизована згідно вимог *FIPA-ACL* [9]. Схема взаємодії користувача з МАСД за підтримки інтерфейсного агента, представлена на рис.1.

Визначимо наступні типи запитів користувача:

1) запити-прогнози на зміну певних параметрів ПК при зміні умов функціонування або конфігурації системного блоку, що діагностується (якщо збільшити частоту процесора, то наскільки зросте його температура; якщо встановити кулер на жорсткий диск, то наскільки впаде температура жорсткого диску);

2) інформація про значення температур, навантажень, частот компонентів ПК за певний період часу. Результат запиту виводиться у вигляді таблиць, графіків і результатів аналізу параметрів ПК на протязі зазначеного періоду часу;

3) запити на зміну системних параметрів (частот кулерів, робочих частот

компонентів ПК, навантажень компонентів ПК);

4) запити-оптимізатори, які подаються у вигляді вимог користувача по досягненню певних значень параметрів системи. Вони можуть містити обмеження на значення параметрів.

Відповідь системи діагностування приходить у вигляді набору рекомендацій користувачеві й списку способів автоматичного усунення.

При існуванні ряду можливих рішень щодо необхідних значень параметрів,

які задовольняють обмеженням, відбувається діалог МАСД з користувачем із ціллю виявлення його переваг і відсіювання невідповідних рішень (запит: знизити температуру процесора на 5°C ; обмеження: не знижувати ефективність використання процесора нижче 90%; відповідь: автоматичне усунення - збільшення обертів кулера процесора на 20%, рекомендація користувачеві – усунути пил на поверхні радіатора процесора).

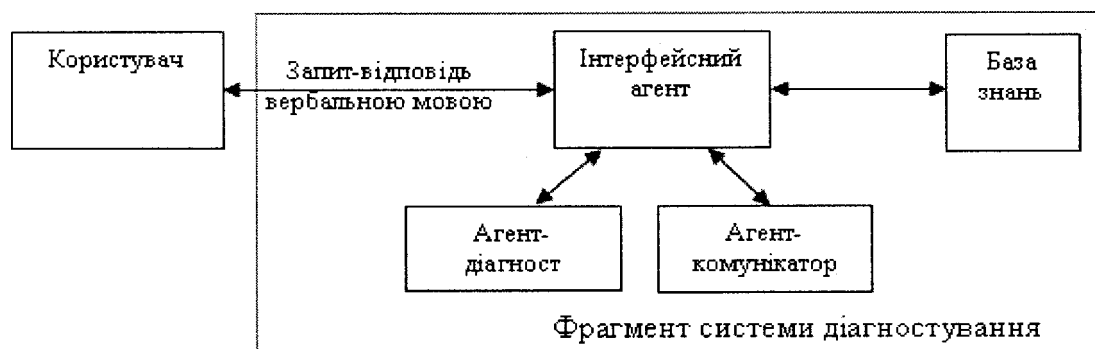


Рис.1. Схема взаємодії користувача з МАСД

Запити системи діагностування:

1) уточнення конфігурації ПК. Проводиться у випадку відсутності діагностичної інформації щодо певних компонентів ПК, або при суперечливості даних;

2) запит на вибір вимог користувача (ціль: знизити температуру процесора не менше, ніж на 2°C ; спосіб 1: знизити частоту процесора – зміна температури складе $\Delta T = -2^{\circ}\text{C}$, спосіб 2: знизити навантаження процесора на 20% – зміна температури складе $\Delta T = -3^{\circ}\text{C}$);

3) підтвердження дій МАСД чи користувача (підтвердження запиту, підтвердження зміни умов функціонування, підтвердження скасування дії).

Інформування користувача:

1) настання певної події (температура процесора становить 65°C);

2) системна інформація (запит у процесі опрацювання діагностичної інформації, недостатність діагностичної інформації для реалізації процесу діагностування).

Структура-запит складається з:

- ідентифікатора запиту;

- поля "одержувач", що вказує на одержувача запиту;

- поля "адресат", що вказує на відправника запиту;

- тіла запиту, що включає інформацію запиту;

- статусу запиту, що вказує на стан запиту.

Запит може приймати один з наступних станів: не опрацьований, у процесі опрацювання, відкладений, опрацьований.

Структура-відповідь містить:

- ідентифікатор запиту, на який було дано відповідь;

- поле "одержувач";

- поле "адресат";

- статус відповіді, що приймає одне із двох значень: "відповідь прочитана", "відповідь не прочитана".

Інформування користувача містить поле тексту з інформацією й поле статусу: "візуалізовано", "не візуалізовано".

Запити, відповіді й інформування містять також дату й час надходження і опрацювання. Для ідентифікації запиту отримують особистий ідентифікаційний

номер, відповіді на запит містять ідентифікаційний номер запиту, на який вони відповідають.

Структури запиту, відповіді та інформування користувача представлено на рис. 2.

Ідент.	Одержувач	Адресат	Поле запиту	Статус	Дата/час отримання	Дата/час опрацюв.
--------	-----------	---------	-------------	--------	--------------------	-------------------

Структура запиту

Ідент. запиту	Одержувач	Адресат	Поле відповіді	Статус	Дата/час отримання	Дата/час опрацюв.
---------------	-----------	---------	----------------	--------	--------------------	-------------------

Структура відповіді

Текст інформування	Статус
--------------------	--------

Структура інформування користувача

Рис. 2. Структури запиту, відповіді та інформування користувача

Функціонування інтерфейсного агента

При опрацюванні інтерфейсним агентом запитів, відповідей та інформувань користувача найвищий пріоритет мають запити користувача, середній пріоритет – інформування користувача. Запити МАСД мають найнижчий пріоритет.

Введення пріоритетів необхідно у зв'язку з існуванням можливості виникнення ситуацій одночасного надходження двох запитів різного типу. Запит користувача має найвищий пріоритет, оскільки при прийнятті рішення користувачеві можуть знадобитися додаткові дані про стан компонентів ПК, умови їх функціонування й температурні режими. Також, користувач враховує інформацію, що надходить від системи діагностування у вигляді інформувань. Типовою є ситуація, коли виникає запит від системи діагностування до користувача, користувач одержує інформаційні повідомлення від системи діагностування, посилає запити системі діагностування, одержує відповіді й на основі отриманих даних відсилає відповідь системі діагностування. Система пріоритетів дозволяє користувачеві одержувати інформацію від системи діагностування під час запиту, що забезпечує прийняття більш зважених рішень.

Функціонування інтерфейсного агента описується множиною правил-продукцій, ліва частина яких містить умови, а права – виконувану дію.

Приклад правил інтерфейсного агента, що описують сценарій опрацювання запитів/відповідей:

```

if вхідний_запит_відповідь
then відкласти_опрацювання
and поставити_вхідний_запит_в_чергу
and x = вибрати_запит_відповідь_найвищого_пріоритету
and поставити_на_опрацювання(x);
if статус_вільний
then x = вибрати_запит_відповідь_найвищого_пріоритету
and поставити_на_опрацювання(x);
if пріоритет(запит_на_виконанні) < пріоритет(вибрати_запит_відповідь_найвищого_пріоритету)
then відкласти_обробку
and x = вибрати_запит_відповідь_найвищого_пріоритету
and поставити_на_опрацювання(x).

```

Приклади взаємодії користувача з МАСД

Типовий приклад взаємодії користувача з системою діагностування ілюструє рис.3. Діагностування здійснюється для наступної конфігурації: процесор – *Sempron 2500+*, системна плата – *Abit KV8 Pro*, оперативна пам'ять – 512 Мб. При побудові екранних форм використано правило “золотого перетину” (5:3). При побудові інтерфейсу використано підхід до проектування інтерфейсів, описаний у роботах [10, 11].

Під час роботи ПК система діагностування визначила, що температура процесора підвищилась до 65°C і не є опти-

мальною. У зв'язку з цим користувач отримав інформування, що ймовірною причиною підвищення температури є значне навантаження процесора.

Для визначення динаміки зміни температури процесора і залежності температури процесора від його навантаження користувач сформував запит про навантаження процесора на протязі останніх трьох годин і зміну температури процесора.

Не бажаючи знижувати продуктивності системи, користувач сформував запит можливих варіантів по зниженню температури процесора, ввівши обмеження

на зниження навантаження (навантаження процесора не нижче 100%).

На основі знань про різні формфактори системного блоку, у яких залежності температур від умов функціонування відрізняються, система надала користувачу зустрічний запит про тип системного блоку. Отримавши цю інформацію, система діагностування надала ряд рекомендацій по зниженню температури процесора.

Під час діалогу користувача з МАСД температура процесора знизилась до 62°C, про що користувач був повідомлений інформуванням.

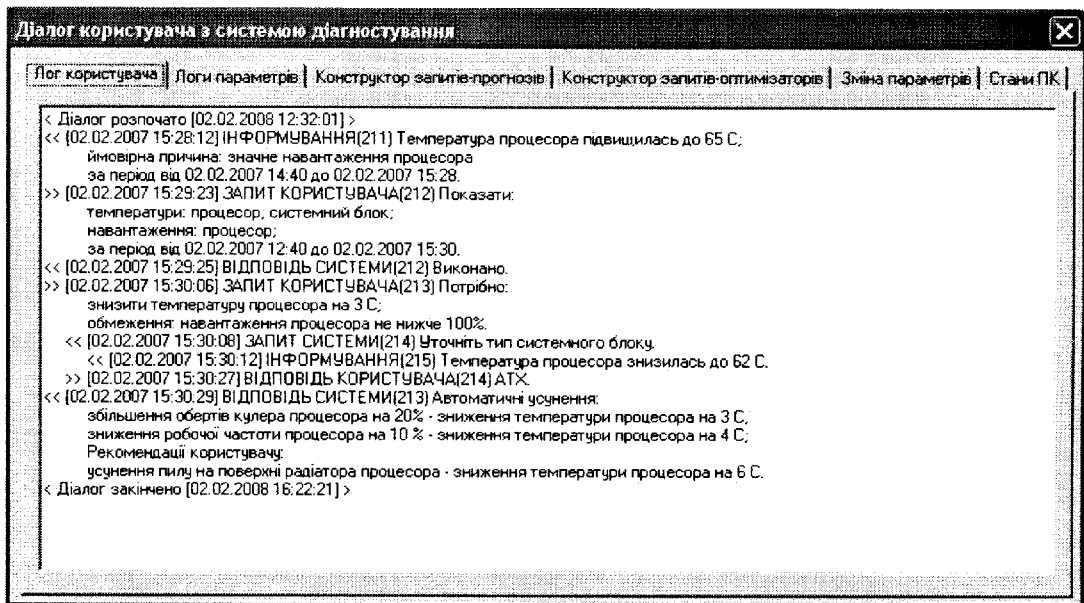


Рис. 3. Приклад діалогу користувача з системою діагностування

З метою уникнення помилок при формуванні запитів користувачем у системі діагностування передбачено автоматизовані засоби – конструктори запитів-прогнозів, запитів-оптимізаторів та запитів на зміну параметрів. Формування запитів зводиться до вибору користувачем необхідних параметрів з списків, а стрічка запиту формується безпосередньо системою діагностування.

Приклад конструювання простого запиту на прогнозування температури процесора при зміні його частоти на 100 MHz (конфігурація, умови функціонування – по замовчуванню) наведено на рис. 4.

Інтерфейсний агент здобуває дані і знання з бази знань. У разі, якщо йому необхідно отримати діагностичну інформацію від агента-діагноста або агента-

комунікатора, він веде діалог з ними за допомогою повідомлень. Приклад внутрішньої структури типового ACL-сумісного повідомлення при взаємодії інтерфейсного агента і агента-діагноста:

```
(request
  :sender interface_agent
  :receiver agent_communicator
  :content (get (cpu_t, system_t,
    cpu_usage) where (cpu = Sempron 2500+, motherboard = Abit KV8 Pro,
    ram = 512 Mb Kingston,
    cpu_usage >= 80%))
  :conversation-id 202
),
```

де request - вказує, що повідомлення є запитом;

sender – інтерфейсний агент, який відсилає запит;

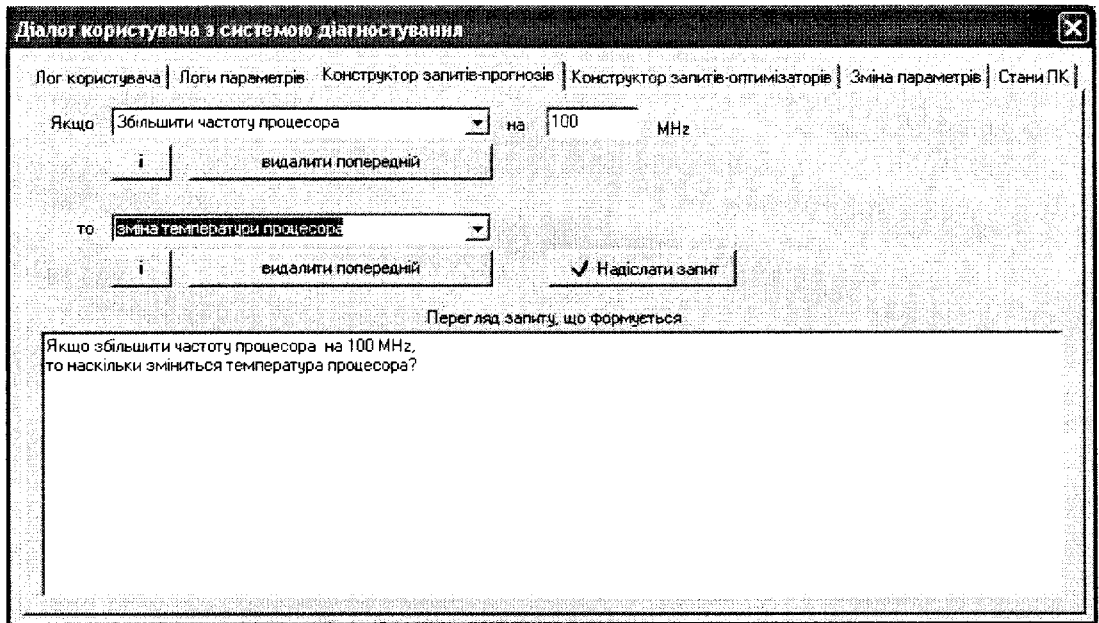


Рис. 4. Конструювання простого запиту-прогнозу

receiver – агент-комунікатор, який отримує запит;

content – запит температури процесора, температури системного блоку, навантаження процесора при навантаженні процесора не нижче 80% для конфігурацій, які є відповідними до вказаної: процесор – *Sempron 2500+*, системна плата – *Abit KV8 Pro*, оперативна пам'ять – 512 МБ *Kingston*;

conversation-id – ідентифікатор запиту.

Висновки

Розроблена реалізація інтерфейсного агента надала ряд переваг при організації процесу взаємодії користувача з МАСД:

- опис можливих актів взаємодії користувача з МАСД та їх типізація на запити, відповіді та інформування дозволило формалізувати їх у вигляді повідомлень, якими обмінюються агенти;

- використання вимог *FIPA-ACL* при побудові повідомлень забезпечило стандартизацію міжагентної взаємодії МАСД;

- застосування системи пріоритетів при організації взаємодії МАСД з користувачем надало користувачу можливість одержувати додаткову інформацію від системи діагностування під час запиту, що забезпечило прийняття більш якісних рішень.

Подальшими кроками у розвитку інтерфейсного агента є:

- накопичення досвіду взаємодії користувача і МАСД з ціллю вдосконалення розуміння та сприйняття інтерфейсу користувачем;

- підвищення рівня автоматизації взаємодії користувача з МАСД для уникнення помилок та зниження вимог до рівня кваліфікації користувача.

Список літератури

1. *Поморова О.В.* Теоретичні основи, методи та засоби інтелектуального діагностування комп'ютерних систем. Монографія. – Хмельницький: ТОВ „Тріада-М”, 2006. – 253 с.

2. *Чайковський Д. Ю.* Дослідження температурних режимів жорстких дисків персональних комп'ютерів // *Радіоелектронні і комп'ютерні системи*. Т.6. – Харків, 2007. – С. 72 – 76.

3. *Поморова О.В., Чайковський Д.Ю.* Аналіз і дослідження температурних режимів компонентів персональних комп'ютерів // *Вісник Хмельницького національного університету* №3, Т.1. – Хмельницький, 2007. – С. 231 – 236.

4. *Поморова О.В., Чайковський Д.Ю.* Розподілена мультиагентна система діагностування комп'ютерних пристроїв // *Оптоелектроніка*. – №2(10) – 2005. – С. 114 – 118.

5. *Рассел С., Норвіг П.* Искусственный интеллект: современный подход, 2-е изд. Пер. с англ. – М.: Издательский дом “Вильямс”, 2006. – 1408 с.

6. AgentBuilder An Integrated Toolkit for Constructing Intelligent Software Agents // Revision 1.3, February 18, 1999. Reticular Systems, Inc.

7. *DeLoach S.A., Wood M.F.* Developing Multiagent Systems with agentTool. In Intelligent Agents VII. Agent Theories Architectures and Languages, 7th International Workshop (ATAL 2000, Boston, MA, USA, July 7-9, 2000).

8. JACK Intelligent Agent - <http://www.agent-software.com.au/shared/products/index.html>.

9. FIPA ACL Message Structure Specification. Foundation for Intelligent Physical Agents, 2000.

10. *Виктор Лаврус.* Золотое сечение <http://n-t.ru/tp/iz/zs.htm>.

11. *Джеф Раскин.* Интерфейс: новые направления в проектировании компьютерных систем. – М.: Изд. дом “Вильямс”, 2004. – 272 с.