

УДК 681.3

Я.С. Парамуд, М.М. БасюкНаціональний університет „Львівська політехніка”,
кафедра “Електронні обчислювальні машини”

ОСОБЛИВОСТІ ЗАСТОСУВАННЯ ТЕХНОЛОГІЇ КЛІЄНТ/СЕРВЕР В МЕДИЧНИХ ДІАГНОСТИЧНИХ СИСТЕМАХ

© Парамуд Я.С., Басюк М.М., 2003

Запропоновано застосувати технологію клієнт/сервер та байєсівський метод прийняття рішень при розробці медичних діагностичних систем. Розглянуто структуру системи та узагальнений алгоритм її функціонування.

In article is offered to apply technology a client/server and baesovsky method of decision-making by development of medical diagnostic systems. The structure of system is considered and is generalized algorithm of her functioning.

ПОСТАНОВКА ЗАДАЧІ

Ознакою часу, в якому ми живемо, є лавинне нагромадження інформації і бурхливий розвиток мікроелектронної техніки та програмних засобів для їх взаємодії. Відбувається перехід до інформаційних технологій типу клієнт/сервер, тобто до широкого застосування комп'ютерів і програмного забезпечення. Найважливішою складовою частиною прискорення науково-технічного прогресу є широка інформатизація технологічних процесів як на виробництві, так і в побуті.

Медицина не є винятком щодо застосування комп'ютерної техніки, а відтак і новітніх комп'ютерних технологій як в технічному, так і в програмному забезпеченні.

АНАЛІЗ ОСНОВНИХ ДОСЛІДЖЕНЬ І ПУБЛІКАЦІЙ

Сьогодні створено численні діагностичні системи для різних областей науки та техніки, таких як медична діагностика й обстеження пацієнтів, генні та молекулярні дослідження, розробка конфігурації обчислювальних машин, пошук несправностей у пристроях та системах тощо [1]. Однак, всі ці системи скеровані на використання лише на окремому робочому місці, що не дозволяє користуватись системою великій кількості користувачів, що значно ускладнює роботу з ними. Ці системи є комерційні (закриті) і в більшості випадків створюються/модифікуються для використання в певних медичних закладах і є їх внутрішніми розробками, тобто не є доступними для широкого загалу [1, 2].

Сьогодні на ринку програмного забезпечення є декілька широковідомих програмних медичних систем діагностування. Кожна з них орієнтована на вирішення певного роду задач.

CASNET. Призначена для діагностування і вибору стратегії лікування глаукоми ока. Ця система використовує причинно - асоціативну мережу, яка може інтерпретуватись для подання часової прогресії хвороби. Висновки, тобто діагноз даної системи представляється не як єдиний діагноз, а як комбінація тверджень про стан пацієнта. Вони містять такі фактори, як: тип і серйозність хвороби, оцінку поточної терапії і рекомендації для подальшого лікування. Для об'єктивної оцінки роботи системи потрібно враховувати кожний з цих компонентів. Було проведено дослідження для оцінки цієї системи і зроблено висновок про правильність її тверджень у 75 % важких випадків глаукоми [4].

MYCIN. Найвідоміша медична система, що призначена для діагностики кишкових захворювань, бактеріальних хвороб крові та спостереження за станом хворого при менінгіті та бактеріальних інфекціях. Однак вона має деякі обмеження. А саме, її онтологія містить перелік бактерій, ознак і способи обробки і не містить особливості хворих, лікарів, лікарні і події, що зустрічаються час від часу. MYCIN розроблена з метою дослідження діагнозів експертів на підставі приблизних (але важливих) припущень, заснованих на частковій інформації. MYCIN подає знання у вигляді набору правил з коефіцієнтами вірогідності. Правила записані розмовною англійською мовою [3].

MOLGEN. Система призначена для дослідження структури ДНК на основі експериментів з ферментами. MOLGEN є професійною системою для здійснення експериментів в молекулярній генетиці. Вона застосовує багатогранний підхід до проблем стратегії виведення. MOLGEN є ефективним інструментом для молекулярного розуміння структури ДНК [7, 8].

CADUCEUS. Медична система, створена в університеті Пітсбурга, призначена для роботи як досвідчений помічник для медичного персоналу. Система складає список хвороб, які має пацієнт, із симптомами, які існують для кожної з них. Система містить 500 хвороб з 3550 різними їх проявами. Це охоплює близько 70 – 75 % медичної діяльності [10].

CENTAURS. Система CENTAURS допомагає фахівцям аналізувати легеневі захворювання. Ця система об'єднує правила і підходи в одній структурі зі застосуванням усіх відомих діагностичних стратегій пошуку захворювання для ефективного порівняння та аналізу захворювань у цьому напрямку. CENTAURS має можливість використовувати знання, що містяться в самій системі, тобто вкладені в неї при розробці, а також має змогу накопичувати їх під час проведення діагностування, а вже згодом використовувати отримані знання для проведення інших діагностувань. Вона є експериментальною і сьогодні знаходиться на стадії розробки [10].

HEADMED. Психіатрична діагностична система HEADMED призначена для терапевтичних досліджень при психозах і нервових зривах. Ця система передбачає можливість надання консультацій відповідно до терапевтичних рекомендацій, здійснюючи вибір лікування, дозування, способу, за допомогою якого ним можна керувати, і в разі необхідності попереджати можливі побічні ефекти [9]. HEADMED – експериментальна система, скерована на перевірку психіатричного діагнозу залежно від результатів клінічної практики.

PUFF. Лабораторна діагностична система, призначена для інтерпретації функції легеневих захворювань, розроблена в Тихоокеанському пресвітерському медичному центрі Стенфордського університету, активно використовується в медичних закладах в усьому світі [8]. Ця діагностична система містить такі незалежні частини: управління набутими знаннями та пояснення щодо інтерпретації результату. Таке використання забезпечило механізм для того, щоб подати предметно - орієнтовані знання у формі правил. Знання подаються шляхом визначення 75 клінічних параметрів, які наводять результати досліджень легеневих функцій типу “вмістимості легень” і “легеневого об'єму”. Ця система містить приблизно 400 правил, за допомогою яких відбувається діагностування; якщо ж система не може зробити аргументований висновок, то вона здійснює діалог з користувачем [6].

MODIS. Система призначена для діагностування різних форм гіпертонії. Крім того, здійснює ряд тестів, за допомогою яких робить висновки про необхідні аналізи, які потрібно зробити пацієнту. На початку діагностування система задає 30 запитань з можливими відповідями, перевіряючи точність відповідей. За відповідями на поставлені

запитання система здійснює попередній аналіз захворювання. Наступні запитання йдуть у напрямку доведення до визначеного діагнозу. Точність системи винятково висока і знаходиться у межах 85 % [9].

Жодна із згаданих медичних діагностичних систем не підтримує технології клієнт/сервер, що є значним недоліком. Застосування технології клієнт/сервер має цілу низку переваг [11]:

- допускає відомчий доступ до даних, що дозволяє певним підрозділам оброблювати тільки ту частину даних, за які вони відповідають;
- забезпечує доступ до даних способом, запропонованим розробником бази даних;
- забезпечує „розподілення праці” між клієнтом та сервером (кожний виконує задачі, для вирішення яких більш пристосований);
- встановлює правила цілісності даних;
- може використовувати покращені можливості цілісності даних, які забезпечуються більшістю серверів баз даних;
- зменшує мережне навантаження, оскільки клієнту повертаються підмножини даних, а не всі таблиці.

Основне завдання даного дослідження – застосувати технологію клієнт/сервер та байєсівський метод прийняття рішення при побудові медичних діагностичних систем.

Проаналізувавши існуючі структури медичних діагностичних систем, було розроблено нову структуру із застосуванням технології клієнт/сервер (рис. 1).

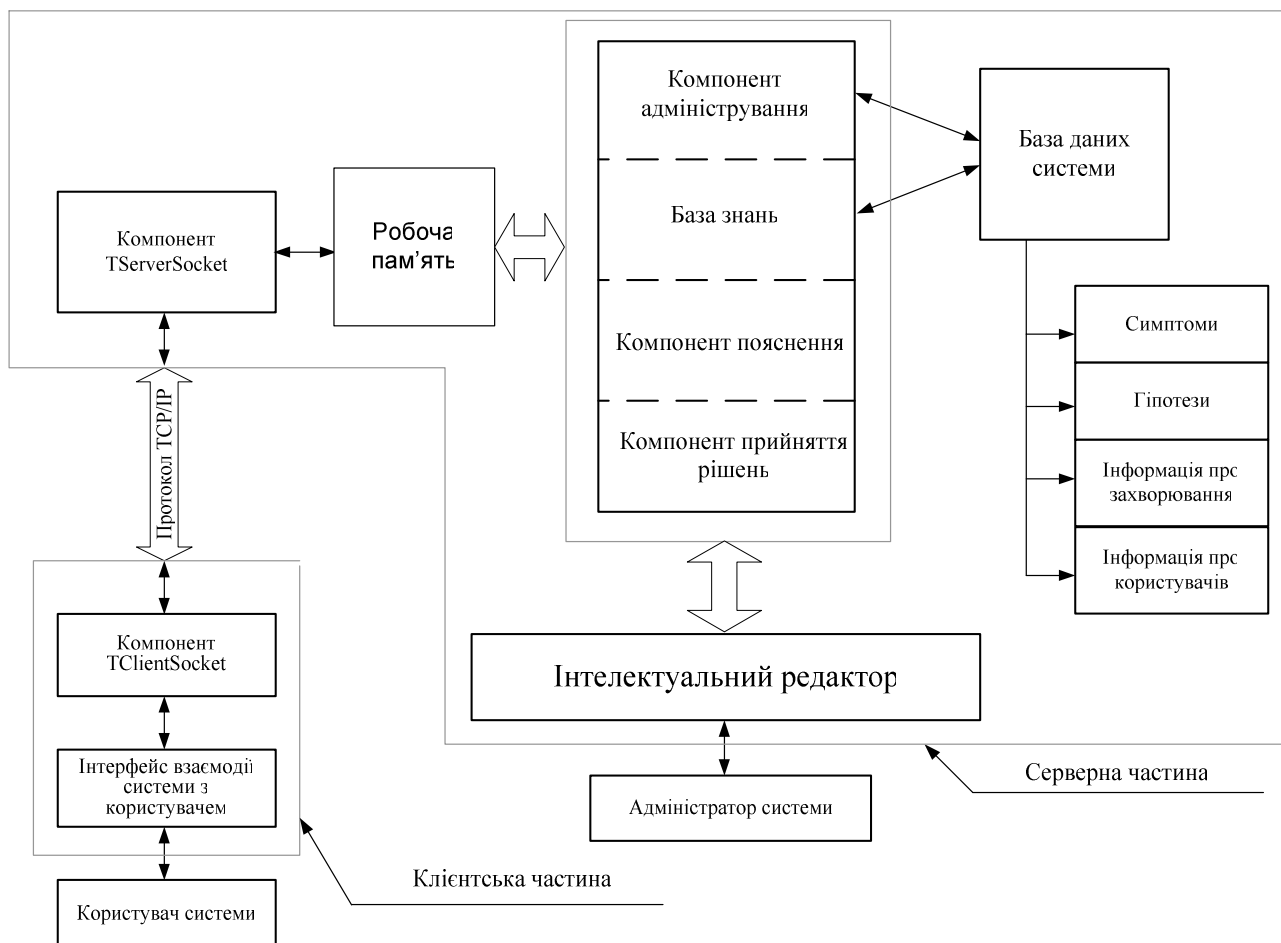


Рис. 1. Структура медичної діагностичної системи

Основними елементами, які забезпечують її функціонування, є:

- ✓ Компонент прийняття рішень – компонент, який моделює хід міркувань фахівця на основі даних, які є в базі знань діагностичної системи;

- ✓ База знань – сукупність знань предметної області, які використовуються при вирішенні задач діагностики. Вона містить посилання на такі елементи: симптоми, гіпотези та інформацію про захворювання, які записані на машинний носій у вигляді бази даних системи;

- ✓ Компонент пояснення – дозволяє користувачу отримувати відповіді на запитання: „Як було отримане таке рішення ?” і „Чому система прийняла саме це рішення ?” Відповідь на перше запитання „Як ?” – це трасування всього процесу отримання рішення із вказанням використовуваних фрагментів бази знань, тобто всіх кроків ланки умовиводу. Відповідь на запитання „Чому ?” – посилання на умовивід, який передував безпосередньо отриманню певного рішення, тобто відхід на один крок назад;

- ✓ Робоча пам'ять – призначена для тимчасового зберігання інформації (фактів, гіпотез та ін.), що є проміжними вирішеннями або результатом взаємодії системи з користувачем, яким є людина, що веде діалог з цією діагностичною системою;

- ✓ Інтелектуальний редактор – забезпечує взаємодію в діалоговому режимі з компонентами системами. Взаємодія здійснюється з компонентами адміністрування, пояснення, прийняття рішення та базою знань. Містить систему меню, шаблонів мови подання знань, підказок („Help”-режим) і ті сервісні засоби, що полегшують роботу з системою;

- ✓ База даних системи – деяка уніфікована сукупність даних, яка сумісно використовується певною групою. Задача її полягає в зберіганні необхідної для певного кола користувачів інформації. В розробленій системі база даних містить: симптоми, гіпотези, інформацію про захворювання та користувачів;

- ✓ Компонент адміністрування – призначений для керування інформацією, що стосується користувачів системи та виконання необхідних службових операцій (створення, знищення, модифікація, надання певних прав та пошуку необхідної інформації);

- ✓ Компонент TServerSocket – серверне сполучення, створює сервер TCP/IP. За допомогою цієї компоненти система налагоджує зв'язок з TCP/IP клієнтом;

- ✓ Компонент TClientSocket – клієнтське сполучення, створює TCP/IP клієнта. За допомогою цього елемента програма сполучається з TCP/IP сервером;

- ✓ Інтерфейс взаємодії системи з користувачем – компонент, який реалізує діалог користувача з діагностичною системою як на етапі введення інформації, так і при отриманні результатів діагностування;

- ✓ Протокол TCP/IP – Transport Control Protocol/Internet Protocol – транспортний мережний протокол, аз допомогою якого здійснюється обмін даними між клієнтською та серверною частиною.

У запропонованій медичній діагностичній системі компонент, що здійснює прийняття рішень, оснований на методі Байєса [13].

Байєсівський метод має строге математичне обґрунтування і на його основі можлива реалізація механізму прийняття рішень, що дозволяють вирішувати задачі діагностики. Ці задачі характеризуються необхідністю обліку невизначеності відповідей на поставлені запитання. Байєсівський метод дозволяє врахувати цю невизначеність. Він заснований на розумінні ймовірності деякої події як якоїсь оцінки, що приписується йому людиною і може змінюватися при одержанні яких-небудь додаткових даних.

Нехай H – подія, яка полягає в тому, що ця гіпотеза правильна, а E – подія, яка полягає в тому, що визначено доказ, який може або не може підтвердити правильність вказаної гіпотези.

Математично цей метод подається виразом (1) [14]:

$$P(H:E) = P(E:H) / (P(E:H)P(H) + P(E:\text{не}H)P(\text{не}H)) \quad (1),$$

де $P(H)$ – ймовірність того, що подія H справедлива (тобто, апіорна ймовірність – це ймовірність настання події H без врахування факту існування E);

$P(E)$ – ймовірність того, що подія E відбулась;

$P(H:E)$ – умовна ймовірність настання події H , яка визначається з врахуванням того, що подія E вже настала (тобто, апостеріорна ймовірність – це ймовірність настання H при умові, що нам відомий факт існування E);

$P(E:H) = P(E_1:H)P(E_2:H)\dots P(E_n:H)$ – умовна ймовірність настання події E , яка визначається з врахуванням того, що подія H вже настала;

$P(\text{не}H) = 1 - P(H)$ – ймовірність того, що подія H помилкова;

$P(E:\text{не}H)$ – ймовірність настання події E при умові, що подія H помилкова.

У цій медичній діагностичній системі вважаємо, що для деякої події H існує велика кількість окремих відомостей, які її підтверджують або заперечують і послідовно виявляються в процесі функціонування. Назвемо їх відповідно E_1, E_2, \dots, E_n . Якщо вони всі виявляються одночасно і не залежать одна від однієї, то можна підрахувати $P(E:H)$ як добуток окремих ймовірностей $P(E_i:H)$ та обчислити $P(H:E)$, де E – подія, яка полягає в тому, що “здійснилися всі E_i ”. Однак цей процес розбитий на етапи, які полягають у підсумовуванні окремих відомостей та визначення їх впливу на умовну ймовірність у міру надходження окремих E_i . Це реалізовано завдяки використанню апіорних та апостеріорних ймовірностей:

1. $P(H)$ – апіорна ймовірність події H ;

2. Запис для даної відомості E_i $P(E_i:H)$ і $P(E_i:\text{не}H)$;

3. З врахуванням методу Байєса підраховуємо $P(H:E_i)$ або $P(H:\text{не}E_i) = P(\text{не}E:H)P(H) / (P(\text{не}E:H)P(H) + P(\text{не}E:\text{не}H)P(\text{не}H))$ залежно від виходу E_i , тобто обчислимо апостеріорну ймовірність події H .

4. Тепер можна не звертати увагу на всі настали що E_i і визначити поточну апостеріорну ймовірність події H як нову апіорну ймовірність H . Так, нехай $P(H) = P(H:E_i)$ або $P(H:\text{не}E_i)$ залежно від значення E_i ;

5. Нарешті вибираємо нову відомість E_i для розгляду і переходимо до п. 1.

Узагальнений алгоритм функціонування медичної діагностичної системи на базі технології клієнт/сервер наведено на рис. 2.

Типова архітектура клієнт/сервер складається із програми-клієнта кінцевого користувача, яка має доступ до даних на віддаленому комп'ютері - сервері [12]. Не введене стандартне визначення, що таке клієнт і чим займається сервер. Однак можна вважати, що сервер забезпечує сервіс, а клієнт вимагає його у сервера. До одного і того ж сервера може звертатись багато клієнтів з вимогою забезпечити їм який-небудь сервіс, і саме сервер вирішує, як опрацювати такі запити. Фактично сервер виконує основну частину роботи системи. Він деякою мірою керує клієнтом: як той має отримати доступ і керувати даними. Реально клієнтські додатки призначені тільки для надання необхідних даних користувачу або для їх доставки від кінцевого користувача. Клієнтські додатки забезпечують інтерфейс користувача для управління даними на сервері. Необхідний сервіс запитується клієнтом в сервера. У цьому випадку клієнт просто виконує запит і забезпечує необхідні для цього дані. Сервер же несе відповідальність за обробку запиту. Однак це не означає, що клієнт не може виконувати яку-небудь логічну дію самостійно; цілком можливо, що клієнт відіграє велику роль в певному додатку, тобто може виконувати в ньому основну частину роботи. Сервер забезпечує сервіс клієнту. Він, власне кажучи, очікує, поки клієнт зробить запит, а потім обробляє цей запит. Сервер повинен мати здатність обробляти декілька запитів від декількох клієнтів, а також розміщувати ці запити за пріоритетами. Найчастіше сервер запущений постійно для того, щоб не припинявся доступ до його сервісів. Клієнт і сервер не обов'язково повинні розміщуватись на різних комп'ютерах [11].

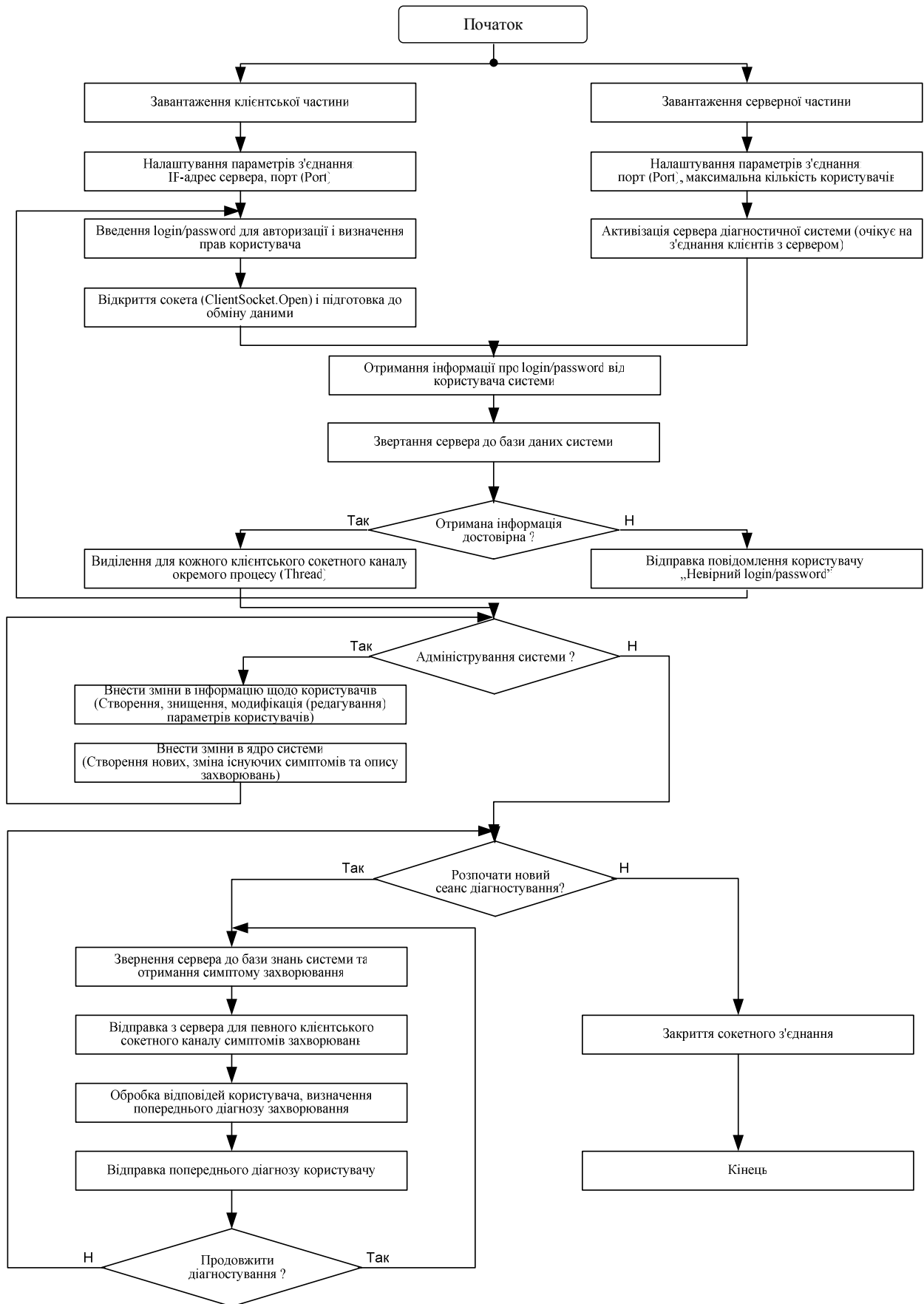


Рис. 2. Узагальнений алгоритм функціонування медичної діагностичної системи

Робота діагностичної системи розпочинається зі завантаження серверної частини, проведення налаштувань параметрів з'єднання (порт, на якому працюватиме система, максимальна кількість користувачів, що зможуть працювати з нею). Пізніше проводиться її активізація, а саме, відбувається прослуховування мережі на наявність під'єднаних клієнтів.

У разі успішного здійснення першої фази роботи проводиться завантаження клієнтської частини діагностичної системи, в якій здійснюється налаштування параметрів з'єднання (IP-адрес сервера та порт). Далі відбувається процес заповнення полів login/password, що використовуються для авторизації і визначення прав користувача. На основі введеної інформації здійснюється налагодження сеансу зв'язку з сервером, під час якого відбувається перевірка параметрів з'єднання (відповідності login/password, максимальної кількості користувачів). Якщо параметри визначені правильно, то для кожного клієнтського сокетного каналу виділяється окремий процес, у межах якого відбувається обмін даними між клієнтською та серверною частинами.

Потім розпочинається сам сеанс діагностування. Він складається з таких етапів, як: звернення сервера до бази знань системи та отримання симптому захворювання; відправка з сервера для певного клієнтського каналу симптомів захворювань; обробка відповідей користувача, визначення попереднього/остаточного діагнозу захворювання.

Для здійснення нового сеансу діагностування отримані операції проводяться в описаному порядку.

ВИСНОВКИ

Наведений аналіз свідчить, що відомі діагностичні системи важко адаптувати до технології клієнт/сервер, оскільки вони розроблені із застосуванням інших технологій. Запропоновані структурні та алгоритмічні рішення, байєсівський метод прийняття рішень забезпечують побудову ефективних медичних діагностичних систем. Застосування технології клієнт/сервер при розробці медичних діагностичних систем дасть змогу використовувати переваги цієї архітектури (оперативне використання діагностичної системи, що збереже час користувачів, а можливість оперативного внесення даних дасть змогу ідентифікувати найновіші захворювання в найкоротший час) та встановлювати діагноз захворювань з мінімальними часовими затратами.

1. Вавилова А.Б., Базилев К.Р. *Основи построения диагностических систем.* – М.: Медицина, 2002. – 1204 с.: ил. 2. Гаврилов В.О., Коновалов Д.И. *Медицинская диагностика.* – М.: Медицина, 1998. – 569 с. 3. Джексон П. *Введение в экспертные системы.* – СПб: Вильямс, 2001. – 604 с. 4. Жевина Л.О., Заревич Г.В. *Диагностика и программирование.* – М.: Медицина, 1999. – 878 с. 5. Мешалин В.Л. *Применение экспертных системы в диагностике.* – М.: Медицина, 1998. – 698 с. 6. Нейлор К. *Как построить свою экспертную систему.* – М.: Энергоатомиздат, 1991. – 286 с. 7. Орловский С.А. *Проблемы принятия решений при нечеткой информации.* – М.: Наука, 1992. – 306 с. 8. Уотермен Д. *Руководство по экспертным системам.* – М: Мир, 1996. – 388 с. 9. Элли Кумбс М. *Экспертные системы: концепции и примеры.* – М.: Финансы и статистика, 1998. – 591 с. 10. Chen L., Chan C.W. *Ontology construction from knowledge acquisition. Pacific Knowledge Acquisition Workshop (PKAW 2000), Sydney, Australia, December, 2000.* 11. Баас. *Роб и др. Delphi 5. Полное руководство.* – К.: BHV, 2000. – 797 с. 12. Кэнтю М. *Delphi 5 для профессионалов.* – М.: Питер, 2001, – 944с. 13. Аверкина И.З. Батулин А.Н. *Нечеткие множества в моделях управления и искусственного интеллекта.* – М.:Наука, 1998. – 612 с. 14. Мешалин В.Л. *Применение экспертных систем в диагностике.* – М.: Медицина, 1998. – 698 с.