УДК 621.391
© Ю. Добуш[1], І. Демидов[1], М. Климаш[1], 2012

ДОСЛІДЖЕННЯ ВІДКРИТИХ СЕРВІСНИХ ІНТЕРФЕЙСІВ PARLAY В КОНЦЕПЦІЇ «ЕЛЕКТРОННОГО УРЯДУВАННЯ»

У роботі проаналізовано та досліджено моделі систем реалізації інфокомунікаційних послуг в рамках концепції «електронного урядування» на базі відкритих сервісних програмно-прикладних інтерфейсів, представлено результати дослідження процесу надання інфокомунікаційних послуг та аналізу випадкових затримок при їх наданні вузлами мережі NGN.

Запропоновані підходи та методи дозволяють віднайти оптимальні параметри і характеристики, які визначають якість надання послуг системами на основі відкритої сервісної архітектури.

In the work the models of info-communicational services realization based on systems with open software and application interfaces were analyzed and studied within of e-Government concept implementation, the results of the process of info-communicational services provision by NGN network nodes study and random delays analysis in their provision are presented.

The approaches and methods were proposed to find optimal parameters and characteristics that define the quality of service provision of a systems based on open service architecture.

1. ВСТУП

В епоху конвергенції мереж зв'язку все більше значення приділяється розвиткові систем надання послуг. Надання масових послуг, зокрема в процесі реалізації доступу населення до застосувань-сервісів в концепції «електронного урядування», вимагає спеціальної продуктивної та надійної сервісної інфокомунікаційної архітектури. Підвищення вимог користувачів телекомунікаційних мереж до якості послуг та процеси розвитку інфокомунікацій привели до появи якісно нових методів і засобів надання послуг, що ґрунтуються на інтеграції можливостей класичних мереж зв’язку та новітніх засобів мережі Інтернет. Важливою особливістю даних методів стала наявність відкритого програмного інтерфейсу управління послугами, що дозволяє стороннім операторам послуг в короткі терміни розробляти і впроваджувати нові інфокомунікаційні послуги, в тому числі із дотриманням необхідних аутентифікаційних та законодавчих процедур «електронного урядування». Одним з найбільш перспективних напрямків досліджень є створення вузлів надання послуг на базі Parlay API. Ці принципово

[1] Національний університет «Львівська політехніка»

нові підходи до надання сучасних інформаційно-комунікаційних послуг, орієнтованих на мережі зв'язку наступного покоління (NGN), роблять актуальними дослідження моделей і методів надання нових послуг зв'язку, проведені в даній роботі.

Дослідженню систем з використанням прикладних програмних інтерфейсів присвячено ряд робіт [1-7], проте у даних роботах головний акцент робиться на алгоритмічну логіку, пошук тупикових гілок. Для досліджень подібного роду використовуються різні мови опису алгоритмів, такі як SDL, LOTOS, Estelle та мережі Петрі.

Слід зазначити, що в ряді робіт проводяться дослідження систем надання інфокомунікаційних послуг з використанням моделей і методів, які можна застосувати для вивчення систем на базі відкритих інтерфейсів [8-12]. Проте, в даних роботах акцент робиться виключно на дослідження механізмів управління перевантаженням [13].

Метою даного дослідження є розроблення моделей і огляд методів організації інформаційно-комунікаційних послуг в мережах NGN, а також дослідження імовірносно-часових характеристик (ІЧХ) серверів послуг на базі відкритих інтерфейсів.

Оскільки в даній роботі було виконано дослідження імовірносно-часових характеристик систем реалізації послуг у NGN з використанням відкритих інтерфейсів, на основі його результатів стає можливе визначення раціонального варіанту побудови вузла послуг за критерієм якості реалізації послуг і при заданих вихідних параметричних даних.

Інсталяція систем надання додаткових послуг в рамках реалізації концепції «електронного урядування» може здійснюватися як при побудові інфокомунікаційних мереж, так і при модернізації існуючих мереж. Залежно від використовуваних технологій суттєво розрізняються характеристики як мереж, так і сервісних платформ. Таким чином, дослідницьку роботу можна розділити на кілька рівнозначно важливих завдань: дослідження характеристик мережевої інфраструктури, мереж доступу та систем надання послуг зв'язку. Дана робота присвячена вивченню принципів побудови і ймовірносно-часових характеристик серверів комплексних медіа-послуг мережі наступного покоління.

Загалом революційні зміни в технології створення нових інфокомунікаційних послуг для мультисервісних мереж наступного покоління, зокрема для надання послуг в концепції «електронного урядування» вимагають принципово нових підходів до їх аналізу та синтезу, що робить дане дослідження безумовно актуальним.

У створенні технологій надання послуг в мережі наступного покоління бере участь ряд стандартизуючих організацій і консорціумів, що об'єднують мережних операторів та виробників телекомунікаційного

обладнання всього світу. Серед найбільш значимих груп, такі як Parlay, JAIN, OSA, TINA, PINT / SPIRITS, Eurescom Project P909, і SPAN.

Технології надання послуг, що розроблюються даними групами тісно пов'язані з такими поняттями, як прикладні програмні інтерфейси (API) і сторонні оператори послуг. Прикладний програмний інтерфейс – це опис способу, що дозволяє будь-якому фрагменту програмного забезпечення звертатися до іншої програми за отриманням сервісу. Цим сервісом може бути надання доступу до даних або виконання конкретної функції. Історія API інтерфейсів розпочалася зі створення інтерфейсів TAPI і TSAPI, які були випущені в 1993-1994 роках фірмами Microsoft і Novell [14]. Основними завданнями даних інтерфейсів є організація взаємодії між різними програмними компонентами для надання ряду телефонних послуг (в основному обмежених базовими функціями управління викликами та організації конференц-зв'язку). Але в той же час, на базі цих технологій можливе також створення і більш складних систем, таких як операторські центри та системи інтерактивної взаємодії з абонентами. Саме ці інтерфейси визначили розвиток подібних технологій. А також – поява нового учасника телекомунікаційного ринку – стороннього провайдера послуг. Даний провайдер послуг, що може бути державною або комерційною організацією повністю звільняється від однієї з найістотніших частин в організації будь-яких послуг зв'язку – реальної взаємодії з елементами мереж. За рахунок відкритого інтерфейсу, що йому надається, він може керувати мережевими елементами, навіть не замислюючись про реальні технології надання послуги. Від стороннього оператора приховані протоколи взаємодії з мережевими елементами, що особливо важливо, коли в організацію послуги залучені різні мережі, і якщо б розробкою послуги повністю займалися розробники стороннього оператора послуг, їм довелося б для початку вивчити досить складні мережеві протоколи, а вже потім реалізовувати функціональність послуги.

Таким чином, системи з використанням API інтерфейсів дозволяють використовувати ресурси різних мереж і відокремити область розроблення послуг від області застосування, що не тільки збільшує асортимент якісних послуг, але і значно скорочує час їх впровадження в життя. Особливість такого підходу полягає не в наданні будь-яких конкретних нових послуг, а у створенні архітектури, за допомогою якої існуючі послуги та інноваційні послуги стануть доступними стороннім операторам послуг [15]. Загальна архітектура систем на основі відкритих сервісних інтерфейсів Parlay API, що надають користувачам взаємодію та послуги в рамках реалізації «електронного урядування» подана на рис. 1.

Розглянемо систему на базі Parlay API з точки зору моделі обслуговування викликів, функціональної площини послуг і фізичної архітектури.

Для мережі ТМЗК, мобільних мереж та IP-мереж модель обслуговування викликів або запитів з використанням систем Parlay API може бути представлена як базовий процес оброблення виклику мережею і процес обробки логікою послуг Parlay (рис. 2).

З точки зору функціональної площини надання послуг, система Parlay API дуже схожа на інтелектуальну мережу. Для системи Parlay також можна визначити наступні об'єкти (рис. 3):

Базовий процес оброблення викликів-запитів;

Функціональні компоненти послуг (SCF);

Точки ініціалізації послуг і точки завершення послуг.

Функціональні компоненти послуг реалізуються на основі серверів послуг Parlay API і можуть бути задіяні при обслуговуванні викликів-запитів в будь-якій послідовності, визначеній логікою послуги (застосуванням Parlay API).

Архітектура Parlay API дозволяє надавати мережеві послуги сторонніх застосувань за допомогою прикладного програмного інтерфейсу. Основним модулем шлюзу Parlay API є сервер базових послуг (framework), функції якого включають авторизацію, доступ та облік роботи застосувань з серверами послуг.

Однак, ніякі технічні рішення не бувають абсолютними, універсальність API-інтерфейсів може призводити (і зазвичай призводить) до погіршення ймовірносно-часових характеристик (ІЧХ) у порівнянні з індивідуально розроблюваними для конкретного провайдера і для конкретної програмно-апаратної платформи послугами. Більш прогресивні технічні принципи вимагають великих комунікаційних ресурсів, а значить важливо досліджувати ІЧХ платформ надання масових електронних послуг зв'язку із застосуванням API-інтерфейсів, що і становить основне завдання даної роботи.

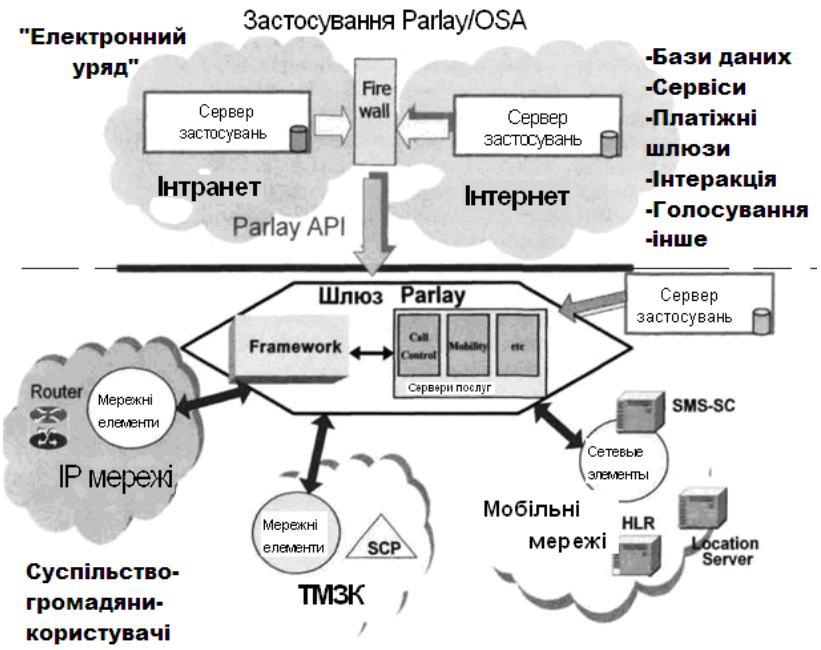


Рис. 1. Загальна архітектура системи надання послуг в концепції «електронного урядування» з використанням Parlay API

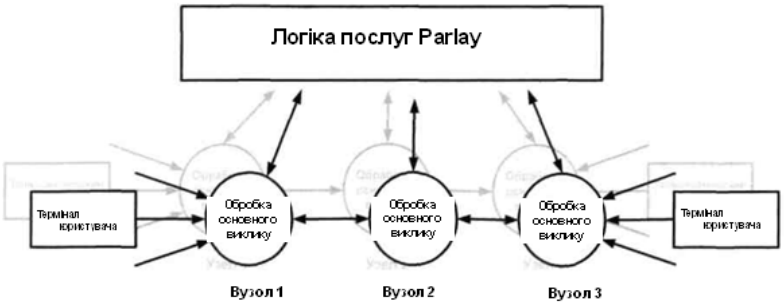


Рис. 2. Модель обслуговування викликів Parlay API

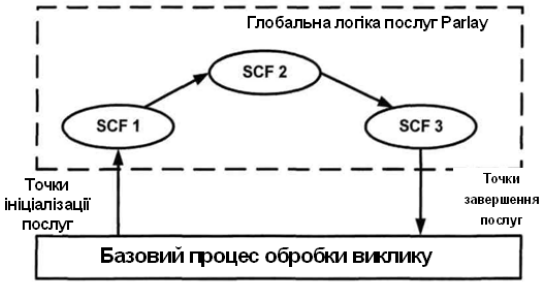


Рис. 3. Функціональна площина реалізації послуг Parlay API.

2. ФУНКЦІОНАЛЬНА МОДЕЛЬ СИСТЕМ НАДАННЯ ІНФОКОМУНІКАЦІЙНИХ ПОСЛУГ З ВИКОРИСТАННЯМ ІНТЕРФЕЙСУ PARLAY API

Організація Eurescom визначає наступні характеристики продуктивності систем Parlay / OSA: продуктивність, затримка повідомлень і час відповіді. Продуктивність системи визначається кількістю вимог, оброблених вузлами системи. Затримка повідомлень – це середній час передачі вхідних повідомлень шлюзом Parlay. А час відповіді визначає швидкість реагування системи на запити [16].

Визначимо функціональні моделі систем Parlay API для централізованої і розподіленої архітектур. На їх основі визначаються математичні моделі систем і обчислюються ймовірносно-часові характеристики, такі як середня кількість вимог в системі, середній час затримки вимоги в системі, і їх залежності від інтенсивності вхідного навантаження, кількості серверів послуг та програм, що взаємодіють з сервером Parlay.

Централізований варіант побудови вузла послуг (рис. 4, а) передбачає створення виділених серверів послуг SCF, які взаємодіють з мережевими елементами з використанням існуючих протоколів. На відміну від цього варіанту, розподілена архітектура використовує функціональні можливості елементів мережі з використанням інтерфейсів Parlay API. Даний варіант передбачає, що елементи мережі підтримують дані інтерфейси (рис. 4, б).

При розподіленому варіанті побудови (рис. 4, б), завдання системи Parlay API зводяться до функцій сервера базових послуг, тобто аутентифікації та авторизації, забезпечення можливості пошуку і використання серверів послуг та інших функцій, пов'язаних з урахуванням використання ресурсів системи. Такий варіант значно простіший з точки зору побудови, але в той же час передбачає значні серйозні вимоги

до мережевих елементів. При побудові мережі з нуля такий варіант може бути кращим, ніж централізований. Проте, очевидно, що у випадку впровадження системи Parlay на існуючій мережі більш економічним є централізований варіант.

Об'єктами функціональних моделей, представлених на рис. 4, а і рис. 4, б, є:

• Джерела вимог, ініційованих користувачами комунікаційних мереж;

• Інфокомунікаційні мережі (ТМЗК, мобільні та IP мережі, зокрема мережа Інтернет);

• Шлюз Parlay послуг, що включає сервер базових послуг та сервери послуг, включно з послугами систем «електронного урядування»;

• Відповідні програми-застосування Parlay;

• Мережі передачі даних, що об'єднують основні об'єкти системи Parlay API.

Основні відмінності між централізованим і розподіленим варіантами побудови систем Parlay API викликані розташуванням функціональних можливостей серверів послуг. Одним із завдань нашого дослідження є порівняння характеристик двох описаних варіантів систем Parlay API і надання рекомендацій щодо їх застосування.

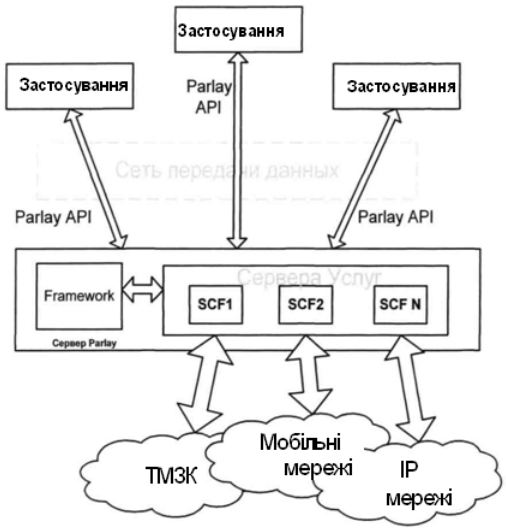
3. МАТЕМАТИЧНА МОДЕЛЬ СИСТЕМИ PARLAY API

При побудові математичної моделі системи Parlay API (рис. 5-6) в даному дослідженні враховувалося наступне:

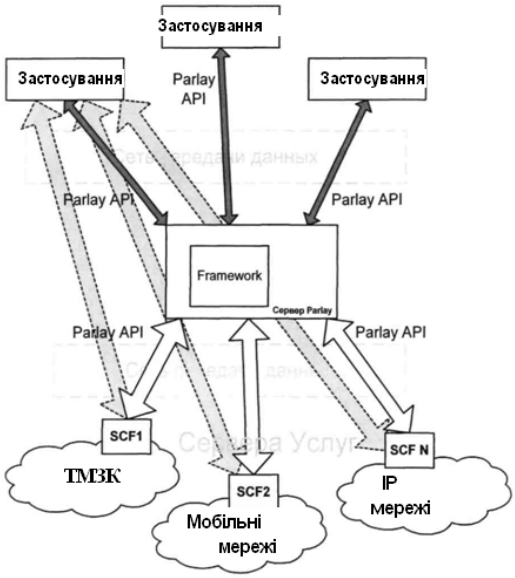
• Система Parlay API складається з ряду незалежних об'єктів: базового сервера, серверів послуг, програм і мереж передачі даних;

• Під вимогами будемо розуміти потоки викликів, які приходять від великої кількості джерел і мають показниковий розподіл інтервалів часу надходження;

• Для взаємодії між об'єктами моделі використовується мережа передачі даних Ethernet, як найбільш поширена і така, що відповідає вимогам щодо швидкості передавання даних;



а)



б)

Рис. 4. Функціональні моделі централізованого (а) та розподіленого (б) варіантів побудови системи Parlay API.

• Вимоги в систему можуть надходити безпосередньо на застосування (наприклад, вимоги, ініційовані самим застосуванням або ті, які надходять через мережу Інтернет) або / і на сервери послуг;

• При взаємодії програми та сервера базових послуг можлива повторна обробка вимоги, тобто, наприклад, вимога про встановлення з'єднання двох абонентів, що не зареєстровані на сервері, надходить на застосування, а процедура їх авторизації може включати в себе кілька етапів.

На даних моделях (рис. 5-6) представлені системи Parlay API з централізованими та розподіленими серверами послуг, що включають k серверів послуг. У загальному випадку сервер Parlay надає послуги різних застосувань, на які надходять вимоги з інтенсивністю $\gamma_1, \gamma_2, ..., \gamma_n$. Інтенсивності оброблення вимог застосуванням, мережею передачі даних, сервером Parlay і серверами послуг відповідно позначені μ_{ap}, μ_{eth}, μ_{fw} *і* μ_{scs}. Інтенсивності надходження вимог до системи через сервери послуг позначені $r_1, r_2, ... , r_n$.

Імовірності того, що вимоги надійдуть на певні сервери послуг, позначені $p_{scs1}, p_{scs2}, ..., p_{scsk}$, а імовірності того, що після обслуговування на сервері базових послуг вимоги знову надійдуть на застосування – $p_{Ap1}, p_{Ap2}, ..., p_{Apn}$.

Відповідна ймовірність в загальному випадку визначається типом послуг, застосуваннями, що використовуються і алгоритмами аутентифікації.

Модель рис. 6 відрізняється тим, що необхідно окремо враховувати наявність мережі передавання даних між сервером базових послуг і серверами послуг, що реалізовані безпосередньо в мережних елементах і, отже, не можуть взаємодіяти з сервером через внутрішні високошвидкісні інтерфейси.

Оскільки одним з дослідницьких завдань є аналіз зміни ІЧХ вузла послуг при використанні прикладного програмного інтерфейсу, то необхідно розглянути математичну модель вузла послуг, у якого область реалізації сервісів не виділена в окрему площину, а реалізована на серверній частині. Така архітектура на сьогодні є найбільш дешевою, але і найбільш вразливою, про що свідчить нежиттєздатність мережних порталів більшості державних структур України в умовах перевантаження запитами. Для того, щоб проводити порівняння близьких за використовуваними технологіями систем згідно моделей, представлених на рис. 5-6, розглянемо математичну модель вузла послуг (рис. 7).

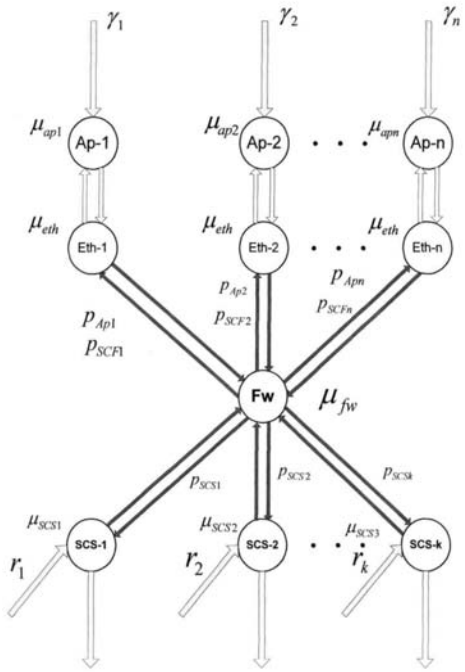


Рис. 5. Математична модель системи Parlay з централізованим сервером послуг

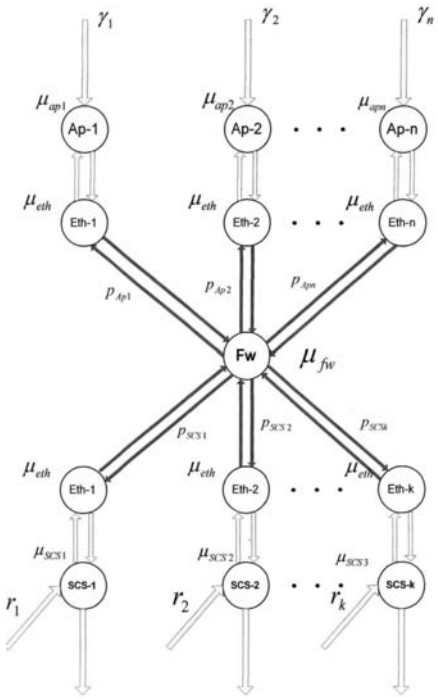


Рис. 6. Математична модель системи Parlay з розподіленими серверами послуг

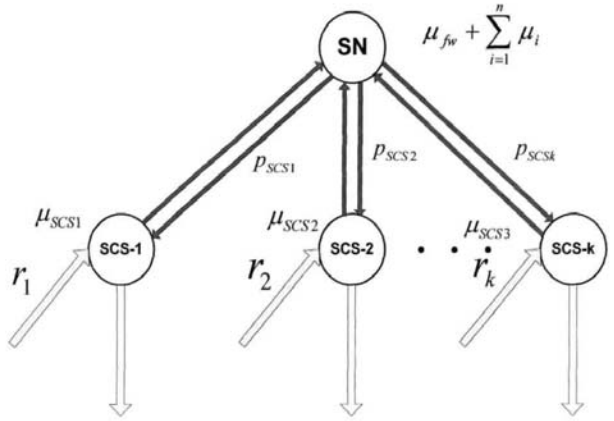


Рис. 7. Математична модель вузла послуг без API інтерфейсу

У моделі, представленій на рис. 7, частина взаємодії з мережними елементами аналогічна серверам послуг моделі рис. 5, а логіка послуг виконується в центральному вузлі SN, так само, як і системні функції, що відповідають за взаємодію всіх об'єктів системи.

Підсумовуючи, можна виділити, що основними характеристиками досліджуваної системи з відкритими сервісними інтерфейсами є середня кількість вимог в системі та середній час обслуговування вимог. Далі розглянемо залежність цих характеристик від параметрів такої системи, до яких слід віднести:

- кількість застосувань;
- інтенсивність надходження вимог на програми ззовні;
- кількість серверів послуг та їх функціональних компонент;
- інтенсивність надходження вимог на сервери ззовні;
- ймовірність повторного обслуговування вимоги застосуванням після обслуговування на сервері базових послуг та інше.

4. ПОРІВНЯННЯ ІЧХ СИСТЕМИ PARLAY API ТА ВУЗЛА ПОСЛУГ З ОБ'ЄДНАНИМИ ФУНКЦІЯМИ УПРАВЛІННЯ І КОМУТАЦІЇ ПОТОКІВ ЗАПИТІВ

Для того, щоб порівняти характеристики двох підходів до побудови сервісних вузлів послуг, класичного вузла послуг інтелектуальної мережі та системи Parlay API, використовувалися математичні моделі, описані вище (рис. 5-7). Їм було надано аналітичної форми та були виконані відповідні розрахунки, зокрема використано теорію відкритих мереж Джексона. При цьому, з точки зору сервісної частини систем, тобто серверів послуг, обидва підходи абсолютно ідентичні (заді-

яно однакову кількість серверів послуг з однаковою кількістю сервісних компонент).

Основна відмінність даних підходів полягає в тому, що в класичному підході логіка послуг реалізується в єдиному сервері, а в Parlay API – на застосуваннях. По цьому середній час обробки для вимоги, що надходить на сервер базових послуг Parlay API, значно менше, ніж середній час обробки вимоги сервером в класичному вузлі послуг. Середній час обслуговування вимоги може бути визначено статично.

При розгляді характеристик систем враховувалося, що середній час обслуговування вимоги на сервері Parlay дорівнює 5 мс, на застосуванні – 50 мс. Якщо вважати, що послуги, які надаються системами однакові, тоді середній час обслуговування вимоги в класичному вузлі послуг приблизно дорівнює часу обслуговування вимоги на застосуванні системи Parlay API. Припустимо, що за рахунок збільшення продуктивності сервера цей час був знижений до 40 мс. Суцільною лінією на рис. 8 представлена залежність кількості вимог в системі від кількості застосувань, при ймовірності необхідності повторного обслуговування вимоги застосуванням p_{API} = 0,1 (Parlay 1). Точкова залежність побудована для p_{API} = 0,5 (Parlay 2). Третя залежність – Parlay 3 – для класичного вузла надання послуг (без застосування архітектури Parlay).

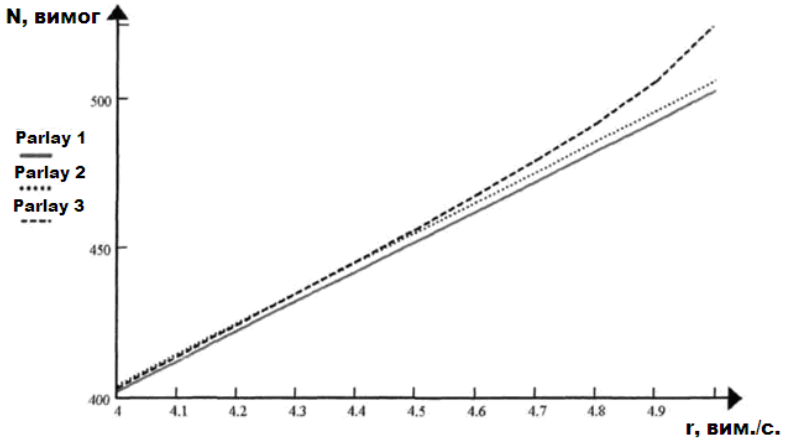


Рис. 8. Залежність середнього числа вимог в системі від інтенсивності надходження вимог на сервери послуг

На рис 8. представлена залежність середнього числа вимог, які знаходяться в сервісній системі від інтенсивності надходження вимог. Залежності носять лінійний характер, поки інтенсивність надходження вимог до системи не досягає 4,7 – 4,8 вимог/с. При таких значеннях

для класичного сервера послуг різко зростає середнє число викликів в системі, що пов'язано з наближенням коефіцієнта використання сервера до 1, що, відповідно, і впливає на збільшення середнього часу перебування вимоги в системі (рис. 9).

Як видно з рис. 9, при невеликих навантаженнях середній час перебування вимоги в класичному вузлі послуг менше, ніж для Parlay-систем з різними параметрами (ймовірностями необхідності повторного обслуговування вимоги застосуванням). Але при збільшенні навантаження на систему в певних точках відбувається перетин характеристик, і системи Parlay стають більш ефективними. Це пов'язано з тим, що реалізацією послуг в даних системах займається не єдиний сервер, а ряд незалежних застосувань, оптимізованих під роботу з певними типами послуг.

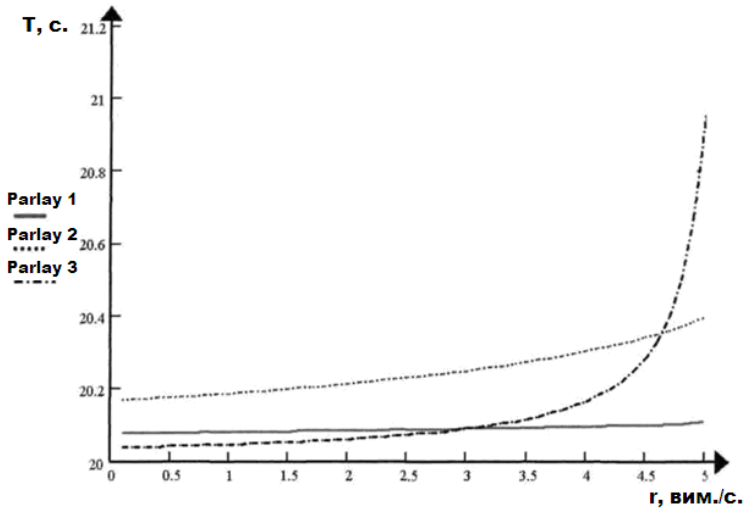


Рис. 9. Залежність середнього часу перебування вимоги в системі від інтенсивності надходження вимог на сервери послуг

На наступному рисунку представлені залежності середнього часу перебування вимоги в системі від кількості серверів послуг. Дані залежності показують наскільки та чи інша сервісна архітектура масштабована для роботи в різних мережах і з різними серверами послуг. З рис. 10 бачимо, що при невеликій кількості серверів послуг характеристики класичного вузла послуг є кращими, але при збільшенні числа серверів послуг середній час перебування вимоги в системі починає різко збільшуватися і при 5-8 серверах послуг характеристики є вже гіршим, ніж для системи Parlay API.

Різке збільшення часу перебування вимоги в системі пов'язане з перевантаженням основного сервера. Для того, щоб класичний вузол послуг зміг працювати з великою кількістю серверів послуг, необхідно збільшувати його продуктивність, однак це не є найбільш ефективним рішенням, тому що, незважаючи на те, що продуктивність сучасних обчислювальних систем величезна, вона все ж обмежена, а заміна таких систем – дорога.

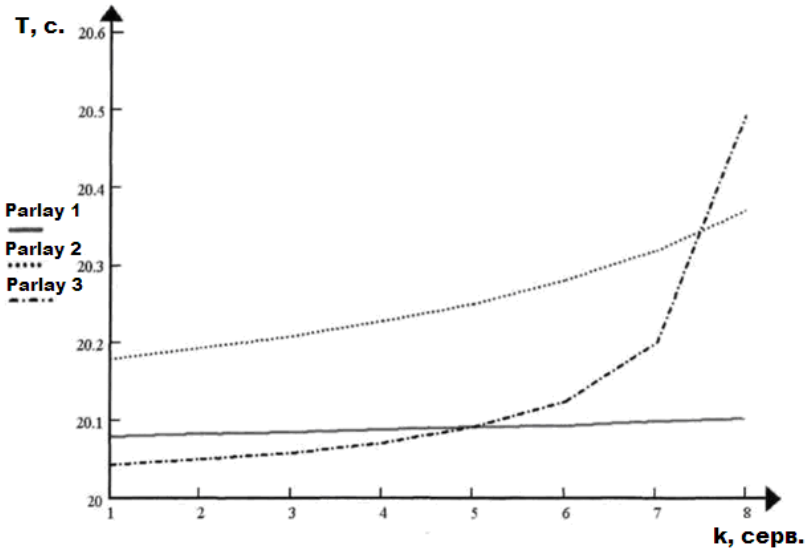


Рис. 10. Залежність середнього часу перебування вимоги в системі від кількості серверів послуг

5. ПОРІВНЯННЯ ІЧХ СИСТЕМ PARLAY API РОЗПОДІЛЕНОЇ ТА ЦЕНТРАЛІЗОВАНОЇ АРХІТЕКТУРИ

Як вже було описано вище, основна відмінність даних сервісних архітектур побудови систем Parlay API полягає в тому, що при розподіленому підході до побудови серверів послуг з'являються додаткові ділянки службових мереж передачі даних, параметри яких повинні враховуватися. Математична модель розподіленої архітектури представлена на рис. 6.

На рис. 11-12 представлені залежності середнього часу перебування вимоги в системі від інтенсивності надходження вимог на сервери послуг. Перша залежність наведена для розподіленої архітектури при коефіцієнті використання мережі передачі даних між сервером базових послуг і серверами послуг менше 0,25.

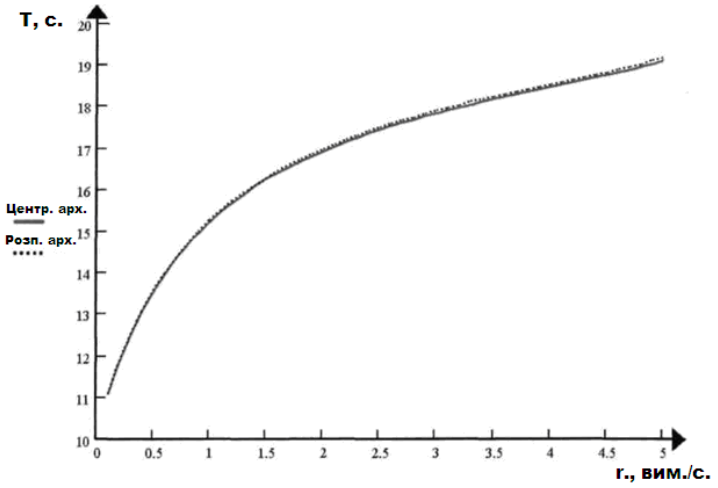


Рис. 11. Залежність середнього часу перебування вимоги в системі від інтенсивності надходження (низький коефіцієнт використання службової мережі передачі даних при розподіленій архітектурі)

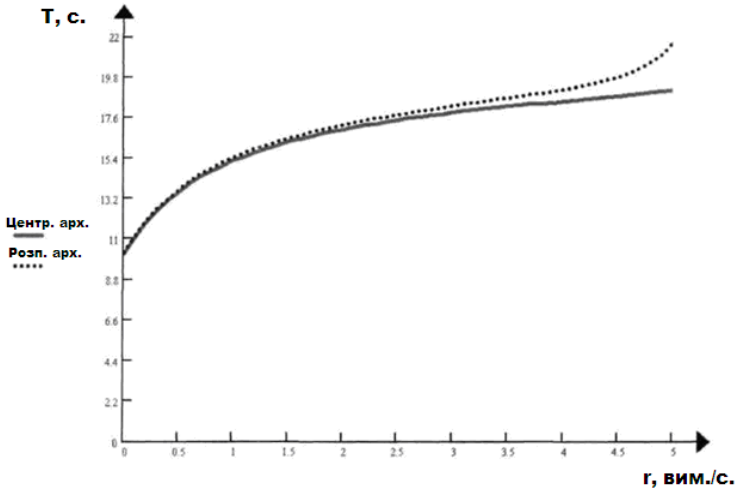


Рис. 12. Залежність середнього часу перебування вимоги в системі від інтенсивності надходження (високий коефіцієнт використання службової мережі передачі даних при розподіленій архітектурі)

На рис. 12 коефіцієнт використання ділянок службової мережі передачі даних значно вищий – 0,5-0,9. З представлених залежностей видно, що при досить невисоких коефіцієнтах використання службової

мережі передачі даних, характеристики систем Parlay різних архітектур практично не відрізняються. Це пов'язано з тим, що додаткові ділянки мереж передачі даних не вносять додаткових затримок в процеси обробки сервісних вимог. Таким чином для масштабів держави при розвитку концепції «електронного урядування» не є суттєвим внутрішній характер архітектури сервісних інтерфейсів, але відповідні інфокомунікаційні системи повинні будуватися на основі сучасних мереж передавання даних, та за зручними і ефективними принципами, що відмінні від класично прийнятих для вузлів обслуговування, наприклад з використанням сервісних архітектур, які розробляються на базі консорціуму Parlay.

6. ВИСНОВКИ

Аналіз еволюції інфокомунікаційних мереж та дослідження вимог до технологій надання послуг зв'язку при переході до мережі наступного покоління в рамках реалізації концепції «електронного урядування» визначили архітектуру Parlay API як одну з найбільш сучасних, що задовольняє вимогам по функціональності, швидкості розроблення та впровадження необхідних сервісів.

На основі отриманих у ході досліджень функціональних та математичних моделей проведений розрахунок характеристик системи з відкритими інтерфейсами. Проведено аналіз імовірнісно-часових характеристик оброблення вимог в системі Parlay API. Отримані результати дозволяють на підставі вихідних даних про продуктивність обчислювальної системи оцінити середній час оброблення сервісних вимог об'єктами системи Parlay API.

Виконано порівняння характеристик класичного вузла послуг та системи Parlay API. Доведено ефективність застосування архітектури Parlay API при великих навантаженнях і великій кількості серверів послуг.

Досліджено централізований та розподілений архітектурні варіанти побудови сервісних систем на базі інтерфейсів Parlay API. Отримані результати показують, що при невеликих значеннях коефіцієнта використання службової мережі передачі даних системи з такими архітектурами володіють схожими характеристиками.

Для масштабів держави Україна при розвитку концепції «електронного урядування» не є суттєвим внутрішній характер архітектури сервісних інтерфейсів, але відповідні інфокомунікаційні системи повинні будуватися на основі сучасних мереж передавання даних, та за зручними і ефективними принципами, що відмінні від класично прийнятих для вузлів обслуговування, наприклад з використанням відкри-

тих сервісних архітектур, які розробляються на базі консорціуму Parlay.

1. Hellenthal J.W., Panken F J.M., Wegdam M. Validation of the Parlay API through prototyping // IEEE. 2001. 2. Harel D. Statecharts: a visual formalism for complex systems // Sci. Computer Program., vol.8,1987. – pp. 213-274. 3. Stefan Leue. Methods and Semantics for Telecommunications System Engineering. – Bern, 1994. 4. Stefan Leue. QoS specification based on SDL/MSC and Temporal Logic. 5. Stefan Leue. Specifying Real-Time Requirements for SDL Specifications - A Temporal Logic - Based Approach. 6. Christian Schwingenschlogl, Stefan Schonauer. Application and service development using UML and SDL // Institute of Communication Networks. 2000. 7. Lin Liu, Jonathan Billington. Modelling and Analysis of the CES Protocol of H.245 // Computer Systems Engineering Center. University of South Australia. 8. Louise Lorentsen, Antti-Pekka Tuoviren, Jianli Xu. Modelling Feature Interaction Patterns in Nokia Mobile Phones using Coloured Petri Nets and Design/CPN. University of Aarhus, Denmark. 9. Melen R., Nyberg C, Tognon S. Performance evaluation of an Parlay gateway // Telecomltalia Lab. 2001. 10. Parekh A., Gallager R. A generalized processor sharing approach to flow control – the single node case. Proc. Infocom'94. Toronto. 1994. 11. Andersson J., Kihl M., Nyberg C. Performance analysis and modeling of an OSA gateway // PWC2003. 2003. 12. Andersson J., Kihl M., Nyberg C. Performance analysis and overload control of an open service access (OSA) architecture. Lund Institute of Technology, Sweden. 2003. 13. Ivan Demydov, Mykhailo Klymash, Natalia Kryvinska, Christine Strauss Enterprise Distributed Service Platform – Network Architecture and Topology Optimization // Int. J. Space-Based and Situated Computing, Vol. 2, No. 1, 2012, P. 23-30. 14. Гольдштейн Б.С, Фрейнкман В.А. Call-центры и компьютерная телефония. – СПб.: БХВ – Санкт-Петербург, 2002. 15. Kocan Kristin F., Montgomery Warren A. Service Creation for Next-Generation Networks // Bell Labs Technical Journal, 2002. – pp. 63-79. 16. Geir Gylterud, Steinar Brede, Michael Walkden, Sergio Tognon. Nonfunctional aspects and requirements related to Parlay/OSA products. White Paper EURESCOM,2002.