

РОЗРОБКА ЕВРИСТИЧНИХ АЛГОРИТМІВ КЕРУВАННЯ РОЗПОДІЛОМ ПОТОКУ ЗАЯВОК В СЕРВЕРНИХ СИСТЕМАХ

В статті сформульована концепція роботи розподіленої серверної мережі. Описані механізми взаємодії між об'єктами мережі, ієрархія керуючих об'єктів, а також як результат: розроблено евристичний алгоритм обміну даними клієнта з розподіленою серверною мережею. Описана ієрархічна модель розподіленої серверної мережі. Сформульовані загальні принципи взаємодії між об'єктами мережі, які дають узагальнену уяву про методи обміну даними в системі.

In this article the concept of a distributed server network. The mechanisms of interaction between objects of the network hierarchy of control objects and the following results: A heuristic algorithm for exchanging data with a distributed client server network. We describe a hierarchical model of a distributed server network. The general principles of interaction between network objects that give a generalized picture of the methods of data interchange.

Ключові слова: серверна система, розподіл навантаження, "Демон" серверів.

Вступ. Одним з головних факторів, що забезпечують успішне функціонування великої сучасної компанії, є впровадження в господарську діяльність досягнень в області інформаційних технологій. На сьогоднішній день особливо слід зазначити широке застосування електронних каналів передачі інформації (локальні мережі або глобальна мережа Інтернет). Проте, створення великих інформаційних систем сполучене з рядом проблем: можливість аналізу мережевого трафіка й очікуваного завантаження сервера; вибір архітектури розподіленої системи; маршрутизація пакетів усередині системи; функціональності розподільників навантаження; методи вибору сервера; розподіл збереженої інформації.

Існує декілька засобів підвищення швидкодії Інтернет ресурсу: можна збільшити смугу пропускання, установити високопродуктивне мережне устаткування, розробити ефективні мережеві додатки, оптимізувати й модернізувати програмні й апаратні компоненти сервера, а також взяти на озброєння технологію кешування.

Ще один спосіб підвищення продуктивності вузла полягає в тому, щоб збільшити кількість серверів і розміщувати на них "дзеркальні" копії матеріалів. Таким чином, можна розподілити загальне навантаження по всіх компонентах системи й скоротити час повернення інформації при виконанні сервером внутрішніх процедур обробки клієнтських запитів. При цьому зберігаються й існуючі сервери, оскільки виводити їх з експлуатації й замінити новими не доведеться.

Розподіл, або вирівнювання навантажень, що доводяться на кілька серверів, дозволяє уникнути такої ситуації, коли передані по мережі пакети перевантажують один сервер, у той час як інші простоюють. Для розподілу навантаження між серверами звичайно використовується функція DNS, іменована циклічною вибіркою (round-robin), яка передбачає можливість кругової передачі Ір-Адреси будь-якого сервера, що містить сайт, будь-якому клієнтові; як результат, навантаження рівномірно розподіляється між усіма серверами. Однак цей механізм недостатньо ефективний у тих випадках, де можливості апаратних і програмних компонентів окремих серверів нерівнозначні. З погляду процедури циклічної вибірки служби DNS між цими системами немає ніякої різниці. Крім того, дана функція не має уяви про доступність того або іншого сервера.

Постановка задачі. Задача даної роботи полягає в розробці засобу керування розподілом навантаження в неоднорідних інформаційних системах для підвищення продуктивності територіально розподілених серверних систем.

Огляд основних методів розподілу навантаження

Існує декілька способів розподілу навантаження:

- DNS із циклічним обслуговуванням (у більшості систем уже не використовується);
- кругове балансування навантаження, метод найменшого числа підключень;
- метод "зважених характеристик";
- алгоритм балансування навантаження, що одержав назва " найшвидший";
- метод найбільшого числа підключень;
- моніторинг за допомогою серверного "Демон"а.

Найбільш ефективний і гнучкий, на наш погляд, є метод розподілу навантаження з використанням серверного "Демон"а. Використання цього методу розподілу навантаження обумовлене можливістю, одержувати достовірну інформацію про стан інформаційних серверів і на підставі отриманих даних оперативно впливати на розподіл потоку даних.

Механізм розподілу навантаження між серверами мережі

Серверна мережа являє собою ряд розподілених серверів зв'язаних між собою каналами зв'язку. У якості каналів зв'язки використовується Internet і Ethernet з'єднання, VPN мережі або інші способи серверної комунікації. Якість з'єднання й пропускна здатність каналів зв'язки для кожного сервера індивідуальна, як і апаратне забезпечення, через що сервери мають різну продуктивність. Крім забезпечення

роботи мережі, сервери вирішують свої локальні завдання, що також впливає на їхню продуктивність.

Сервери мережі розділяються на два типи: інформаційні сервери й "Демон" серверів. Інформаційні сервери обслуговують заявки клієнтів, на кожному з них виконується завдання, яке ми будемо йменувати "агентом" інформаційного сервера. "Демон" серверів виконує збір даних про систему й ухвалює рішення щодо адміністрування розподіленої серверної мережі. Схематично робота серверної мережі представлено на рис. 1.

Агенти інформаційних серверів – це самостійні одиниці розподіленої серверної мережі, кожен з них має власне уявлення про навколишнє його середовище, побудоване виходячи із власних даних і даних отриманих від інших агентів інформаційних серверів. На даному етапі розвитку розподілених систем немає єдиного визначення агента. Ми під інтелектуальним агентом будемо розуміти систему, що володіє рядом характеристик:

- автономність; дії агента визначаються тільки його внутрішнім станом, жодні зовнішні подразники не можуть директивно впливати на поведінку агента, якщо це не було передбачено його структурою;

- реактивність; агенти існують у рамках деякого оточення, з яким взаємодіють, тобто здатні сприймати зміни середовища й реагувати на них;

- проактивність; проявляють цілеспрямовану поведінку для розв'язку свого завдання, тобто агент намагається розв'язати поставлене перед ним завдання в умовах мінливого середовища, для чого планує свої дії самостійно;

- соціальні здібності; здатність взаємодіяти й кооперуватися один з одним для розв'язку поставлених завдань. [1].

Завданням агента є збір інформації про навантаження на свій інформаційний сервер і формування звітних даних у якій-небудь формі. Координатором роботи серверної системи виступає виділений сервер – "Демон" серверів. Даний сервер збирає інформацію від агентів інформаційних серверів, аналізує її й ухвалює рішення щодо розподілу потоку заявок клієнтів між інформаційними серверами мережі.

Клієнтами мережі виступають вилучені термінали. Зв'язок між клієнтами й серверами організовано засобами Інтернет з'єднання, можливі варіанти локальної або віртуально-локальної мережі з'єднання.

Наприклад, керуючий проектом може запустити або призупинити виконання завдань, але він не управляє агентами в системі, відмінними від себе самого як агента.

Крім менеджера проекту, передбачається відсутність централізованого керування завданнями. Підзадачі створюються, коли необхідно задовольнити деяких запит. У системі відсутнє глобальне планування, як централізоване, так і розподілене. Хоча, звичайно, існують заздалегідь певні послідовності виконання деяких підзадач. Якщо відомо, що деяку підзадачу найкраще виконувати саме в цьому порядку, то порядок буде застосований.

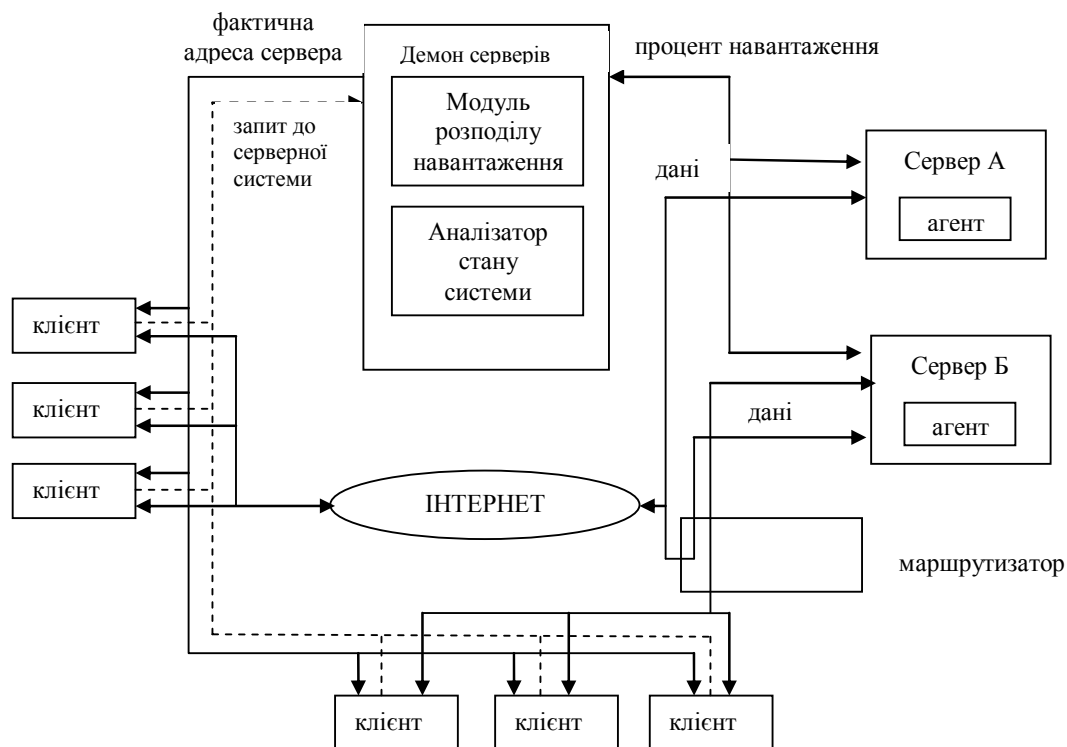


Рис. 1. Схема роботи системи розподілу навантаження

Важливим є питання глобальної цілісності мережі. У системі підтримується тільки мінімальний рівень цілісності. Передбачається, що кожен агент має локальну копію даних моделі, з якої він працює. У

деяких випадках, наприклад при проектуванні механізмів, є можливість організувати одночасну роботу декількох агентів над однією моделлю. Але навіть у цьому випадку різні версії проекту, створювані різними групами зберігаються окремо в окремих базах даних з різними версіями. Об'єднання версій проекту, що надходять від окремих груп, відбувається з деякою періодичністю на нарадах. За цей процес відповідальний менеджер продукту, який за допомогою відповідного об'єднуючого інструмента створює загальну версію проекту, з якої робота буде продовжена в рамках груп.

Клієнти є джерелом формування заявок. Потрапляючи в серверну систему, заявки обробляються "Демоном" серверів.

Наприклад, керуючий проектом може запустити або призупинити виконання завдань, але він не управляє агентами в системі, відмінними від себе самого як агента.

Крім менеджера проекту, передбачається відсутність централізованого керування завданнями. Підзадачі створюються, коли необхідно задовольнити деякий запит. У системі відсутнє глобальне планування як централізоване, так і розподілене. Хоча, звичайно, існують заздалегідь певні послідовності виконання деяких підзадач. Якщо відомо, що деяку підзадачу найкраще виконувати саме в цьому порядку, то порядок буде застосований.

Важливим є питання глобальної цілісності мережі. У системі підтримується тільки мінімальний рівень цілісності. Передбачається, що кожний агент має локальну копію даних моделі, з якої він працює. У деяких випадках, наприклад при проектуванні механізмів, є можливість організувати одночасну роботу декількох агентів над однією моделлю. Але навіть у цьому випадку різні версії проекту, створювані різними групами, зберігаються окремо в окремих базах даних з різними версіями. Об'єднання версій проекту, що надходять від окремих груп, відбувається з деякою періодичністю на нарадах. За цей процес відповідальний менеджер продукту, який за допомогою відповідного об'єднуючого інструмента, створює загальну версію проекту, з якої робота буде продовжена в рамках груп.

Клієнти є джерелом формування заявок. Потрапляючи в серверну систему, заявки обробляються "Демоном" серверів [2].

Даний сервер виконує ряд дій з аналізу стану системи й передає запит клієнта на один з інформаційних серверів мережі. Інформаційний сервер, у свою чергу, одержуючи переадресований запит клієнта, обробляє його, формує пакет звітних даних по запиту й відправляє його клієнтові. Надалі, якщо зв'язок клієнта з інформаційним сервером не переривалася, то всі запити, сформовані клієнтом, обробляються даним інформаційним сервером. На рис. 2 представлений алгоритм обміну даними клієнта із серверною системою.

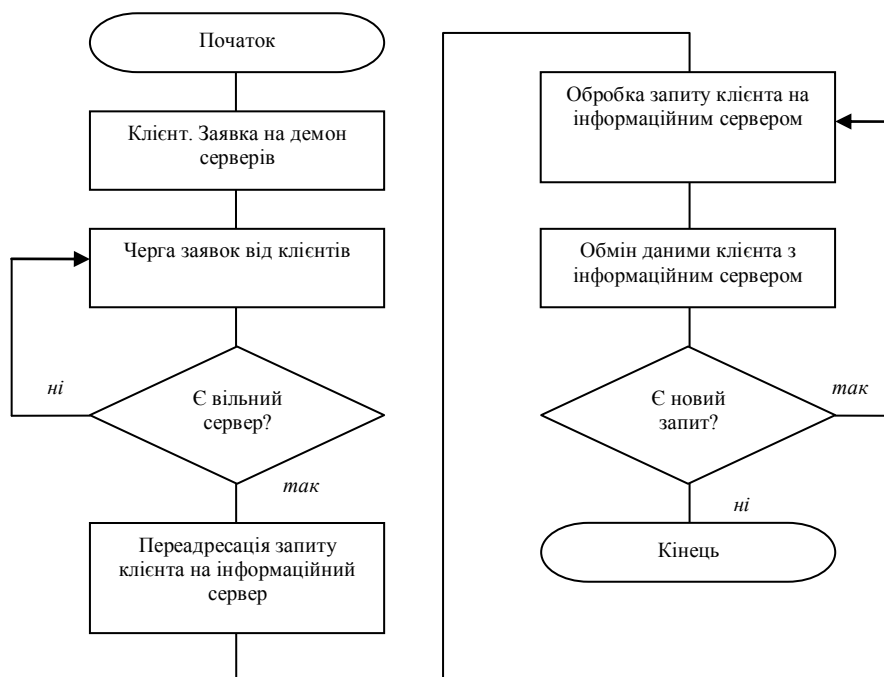


Рис. 2. Алгоритм обміну даними клієнта з інформаційною системою

Розглянутий евристичний алгоритм обміну даними клієнта із серверною системою ілюструє узагальнений механізм роботи серверної мережі (рис. 1). Далі більш докладно розглянемо механізми взаємодії окремих частин мережі.

Механізми роботи "Демона" серверів і інформаційного сервера

Розглянемо взаємодію як окремих частин серверної мережі, так і внутрішній пристрій об'єктів керування, таких як логічні модулі отримання розв'язків, об'єкти оцінки стану системи й інші об'єкти взаємодії серверної мережі.

Першим розглянемо об'єкт керування серверної мережі – "Демон" серверів за своєю суттю є

об'єктом розподілу навантаження всієї серверної системи. На нього покладений ряд функцій: розподіл потоку заявок, аналіз стану інформаційних серверів, формування таблиці пріоритетів, короткострокове прогнозування стану мережі. "Демон" серверів складається з логічних блоків: модуля розподілу навантаження й аналізатора стану системи (рис. 3).

Логічний модуль розподілу навантаження складається з декількох частин – блок вибору інформаційного сервера з таблиці пріоритетів, блок переадресації запиту клієнта інформаційному серверу й блок прогнозування стану системи.

Перші два блоки, як видно з їхніх назв, виконують функції безпосереднього розподілу заявок клієнтів між інформаційними серверами мережі. Блок прогнозування стану системи виконує наступні функції оцінка ємності вступників заявок, зіставлення заявки з можливостями обраного інформаційного сервера, формування короткострокового прогнозу щодо відсотку завантаження інформаційного сервера в момент потрапляння на нього заявки клієнта.



Рис. 3. Механізм роботи блоку прогнозування стану мережі

Механізм роботи блоку прогнозування стану серверної мережі досить простий. На вхід серверної мережі надходить потік заявок, "Демон" серверів пропускає через себе потік заявок клієнтів, формує статистичні дані, зібрані по параметрах клієнтських запитів і зіставляє їх з даними, що надходять від агентів інформаційних серверів. У результаті створюється оновлювана, статистична база даних утримуюча інформації про якусь частину вступників заявок клієнтів.

На підставі сформованої бази даних відбувається формування короткострокового прогнозу про стан серверної мережі.

На підставі виводів блоку прогнозування стану системи можна робити припущення щодо завантаження інформаційних серверів і вносити відповідні коректування в таблицю пріоритетів.

"Демон" одержує інформацію про завантаження інформаційних серверів від агентів розташованих на них:

- відсоток завантаження центрального процесора, якщо на інформаційному сервері розташовано кілька процесорів, то передається середнє завантаження;
- відсоток використання каналу зв'язки;
- відсоток завантаження жорсткого диска;
- відсоток використання оперативної пам'яті.

Аналізуючи стан серверів системи (блок аналізу стану системи), "Демон" динамічно формує таблицю пріоритетів відповідно до закономірності, по якій відбувається розподіл навантаження, обумовленого функцією розподілу, яка використовується в цей момент.

На підставі динамічно сформованої таблиці пріоритетів "Демон" вибирає фактичну адресу сервера й перенаправляє запит клієнта на нього, після цього "Демон" звертається до запиту наступного клієнта.

Формування таблиці пріоритетів відбувається щораз, коли "Демон" одержує звіти від агентів про стан інформаційних серверів. Одержання "Демоном" серверів запитів від клієнтів не впливає на формування таблиці пріоритетів.

Увесь зміст розподілу навантаження між серверами зводиться до рівномірного завантаження серверів [3].

Таким чином, порядок побудови інформаційних серверів у таблиці пріоритетів строго залежить від функції розподілу вхідного потоку заявок, яка використовується в модулі розподілу навантаження.

Важливим аспектом при побудові розподіленої серверної системи є питання синхронізації об'єктів серверної системи, для цього є кілька причин:

- одержання "Демоном" повної інформації про стан інформаційних серверів, про всі одразу або про декілька окремо, залежно від поточного завдання;
- виявлення не робочих ділянок мережі, або визначення ділянок, що знову підключилися до серверної мережі, після збою або з інших причин, що не відповідали на запити "Демон"а;

- визначення перевантажених ділянок, частин серверної мережі;
- Розглянуті механізми взаємодії об'єктів серверної мережі дають узагальнене поняття про методи обміну даними в системі у зв'язуваннях: клієнт – сервер, "Демон" – інформаційний сервер. Запропоновані механізми керування потоками в розподіленій серверній мережі, такі як:
- можливість використання кожної, найбільш підходящої функції розподілу вхідного потоку заявок;
 - формування зведеної таблиці пріоритетів серверної системи, яка надає можливість аналізу продуктивності як системи в цілому, так і окремих її частин;
 - передбачений блок прогнозування стану мережі.

Агенти системи й методи взаємодії між ними

Одним з важливих аспектів у роботі розподіленої серверної мережі є робота агента інформаційного сервера. Агенти інформаційних серверів крім участі в діалозі з "Демоном" серверів повинні взаємодіяти між собою. Це обумовлене необхідністю підтримки цілісності інформації, її актуальності. Для цього розглянемо методи взаємодії між агентами, протоколи обміну даними.

Перед початком розгляду механізмів взаємодії між агентами інформаційних серверів, звернемося до структури самого агента. У першу чергу представимо структуру агента інформаційного сервера.

Як видно зі схеми, представленій на Рисунку 4, агент інформаційного сервера складається з декількох функціональних частин:

- інтерфейсу;
- планувальника завдань;
- бази знань.

В інтерфейс входять служби забезпечення агента, які відповідають за його взаємодію з оточенням. Далі ми докладно опишемо роботу служб відповідальних за роботу агента, а поки зупинимось на загальному згадуванні.

База знань агента служить для зберігання всіх без винятку знань, отриманих у процесі життя агента. Сюди входять база моделей агентів, база знань про розв'язуване завдання й база накопичених знань.

У базі моделей агентів зберігаються знання про пристрій і інтерфейсах викликів інших агентів. Споконвічно, у базі є деяка інформація про пристрій інших агентів, яка потрібна для початку роботи. Знання містяться в цю базу в міру взаємодії з іншими агентами. Одержання й зберігання таких знань дуже важливо в агентній системі, оскільки загальна конфігурація системи (кількість, функції й состав агентів) може мінятися із часом без зупинки функціонування.

База знань про розв'язуване завдання містить умови завдання, а також знання, одержувані в процесі розв'язки. Вона зберігає проміжні результати розв'язку підзадач. Також, у базі даних зберігаються знання про способи розв'язку завдань і методи вибору цих способів.

Агент повинен мати ряд характеристик, які його описують як самостійну одиницю в системі. Для виконання роботи, агент повинен мати які-небудь можливості, які в нього закладає розроблювач. Але крім них кожний агент має ряд сервісів – служб агента, які забезпечують його роботу в системі, виконують ідентифікацію агента, забезпечують його зв'язок з "зовнішнім" світом.

Розгляд механізму взаємодії між агентами почнемо з того, що визначимо, яким образом агенти інформаційних серверів одержують інформацію про інших агентів, що входять до складу розподіленої серверної мережі.

Агенти інформаційних серверів можуть працювати в декількох режимах зв'язку:

- Синхронний. Являє собою такий порядок роботи, коли вторинний об'єкт передає відповіді на запити, що надходять від первинного об'єкта. Зміст даного режиму в тому, що викликуваний об'єкт після передачі відповіді на запит переходить у режим очікування й перш ніж зможе знову віддавати, повинен дочекатися вступу наступного запиту передачі;
- Асинхронний. У даному режимі роботи вторинний об'єкт самостійно ухвалює рішення щодо передачі даних і не повинен чекати для цього одержання запиту передачі від первинного об'єкта;
- Загальний (broadcast). Даний вид зв'язку передбачає можливість широкого вішання якогось агента. Тобто, надає йому можливість відправлення одного пакета всім об'єктам мережі.
- Груповий (multicast). Агент використовує такий режим для відправлення пакета даних обраній групі об'єктів мережі.

У розподіленій серверній мережі агент самостійно, без допомоги, зв'язується з потрібними йому об'єктами мережі. Фундаментальний аспект у зв'язках між об'єктами – це структура повідомлень, якими вони обмінюються, представлення цих повідомлень і їх транспортування [3].

Описані методи роботи агентів у складі розподіленої серверної мережі дають визначення структури взаємодії між агентами:

- специфікації, що визначають архітектуру елементів взаємодії (агенти);
- сформовані рекомендації зі специфікацій агентів розподіленої серверної мережі й комунікаційних технологій;

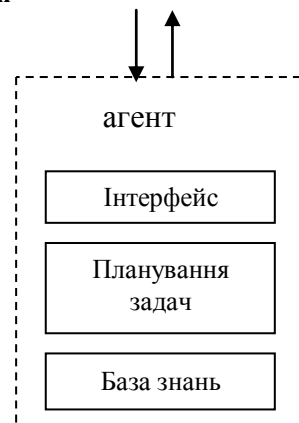


Рис. 4. Модель агента інформаційного сервера

- специфікації, що описують можливість взаємодії об'єктів у розподіленій серверній мережі й методи обміну інформацією в рамках розподіленої серверної мережі.

Висновки. Таким чином, в роботі сформульована концепція роботи розподіленої серверної мережі. Описані механізми взаємодії між об'єктами мережі, ієрархія керуючих об'єктів, а також отримані наступні результати: розроблено евристичний алгоритм обміну даними клієнта з розподіленою серверною мережею. Описана ієрархічна модель розподіленої серверної мережі. Сформульовані загальні принципи взаємодії між об'єктами мережі, які дають узагальнену уяву про методи обміну даними в системі у зв'язуваннях: клієнт – сервер, "Демон" – інформаційний сервер. Розглянуті механізми взаємодії об'єктів серверної мережі. Запропоновані механізми керування потоками в розподіленій серверній мережі, такі як можливість використання кожної, найбільш підходящої функції розподілу вхідного потоку заявок; формування зведеної таблиці пріоритетів серверної мережі; передбачений блок прогнозування стану мережі.

Література

1. Барфилд Э. Программирование клиент-сервер в локальных вычислительных сетях / Э. Барфилд, Б. Уолтере. – М.: Филинь, 1997. – 424 С.
2. Бройдо В.Л. Вычислительные системы, сети и телекоммуникации / Бройдо В.Л. – Спб.: Питер, 2002. – 688 С.
3. Шадхін В.Ю. Оперативний розподіл навантаження на мережі передачі даних / В.Ю Шадхін., В.О. Компанієць, М.О. Каразія / Тези доповідей VII Всеукраїнська науково-практична конференція "Інформаційні технології в світі, науці і техніці". – Черкаси, 2010. – С. 8.

Надійшла 12.1.2011 р.

УДК 621.317.77

І.С. ПЯТІН, О.М. АДАБІР
Хмельницький національний університет

РОЗШИРЕННЯ ЧАСТОТНОГО ДІАПАЗОНУ ФАЗОВИХ ДЕТЕКТОРІВ РАДІОЕЛЕКТРОННИХ СИСТЕМ

Розглядаються алгоритми обробки вимірювального і опорного сигналів фазових детекторів без перетворення і з перетворенням частоти. Описано принцип роботи стробоскопічного перетворювача частоти. Проаналізовані дискретні сигнали на виході стробоскопічного ключа.

The algorithms of treatment of measuring and supporting signals of phase detectors are examined without transformation and with transformation of frequency. Described principle of work of stroboscopic transformer of frequency. Discrete signals are analysed on the output of the stroboscopic key.

Ключові слова: фазовий детектор, стробоскопічний перетворювач частоти, дискретна система.

Вступ

Використання інформації, що міститься в фазі сигналу, дозволяє одержати найбільш достовірну інформацію на тлі завад, забезпечити з найбільшою точністю визначення координат і елементів руху об'єктів, створити шумоподібні сигнали, що відрізняються найбільшою завадозахищеністю і прихованістю [1].

Вимір різниці фаз необхідний при дослідженні фазових характеристик електричних кіл, підсилювачів, фільтрів, трансформаторів. Найбільше поширення фазові методи знайшли у радіонавігаційній техніці для виявлення і визначення місця знаходження об'єктів, в системах автоматичного керування і наведення, пристроях виміру відстані, радіолокаторах, системах близького та далекого зв'язку, на телебаченні, а також в інших галузях науки і техніки. Дуже важливими показниками фазового детектора є похибка визначення фазового зсуву і діапазон робочих частот. Зі зростанням робочої частоти фазового детектора збільшуються його похибки. Тому задачею є вибір способів розширення частотного діапазону, що забезпечує менші похибки визначення фазового зсуву.

Аналіз останніх досліджень і публікацій

Можливості фазових методів реалізуються за допомогою засобів вимірювання, що працюють у широкому частотному діапазоні. При цьому використовують такі алгоритми обробки вимірювального і опорного сигналів [2]:

- векторне додавання або віднімання порівнюваних сигналів;
- перемноження порівнюваних сигналів;
- обробка сигналів різними ключовими схемами перетворення фази на часовий інтервал;
- масштабно-часове перетворення з переносом фазових співвідношень на сигнали однієї частоти.

Перша група алгоритмів реалізується за допомогою фазових детекторів, в яких здійснюється векторне додавання з наступним амплітудним детектуванням результуючого сигналу за допомогою