Київський національний університет імені Тараса Шевченка

На правах рукопису

ГРИЦЕНКО Дмитро Володимирович

МЕТОДИ І ЗАСОБИ ПАРАЛЕЛЬНО-РЕКУРСИВНОГО ПРОГРАМУВАННЯ ДЛЯ ТРАНСП'ЮТЕРНИХ КОМПЛЕКСІВ

01.05.03 — математичне та програмне забезпечення обчислювальних машин та систем

Автореферат дисертації на здобуття наукового ступеня кандидата фізико-математичних наук

Київ — 1996

Дисертацією є рукопис.

Роботу виконано в Інституті кібернетики імені В. М. Глушкова НАН України.

Науковий керівник: доктор фізико-математичних наук, професор **Анісімов Анатолій Васильович.**

Офіційні опоненти: член-кореспондент НАН України, доктор фіз.-мат. наук **Ющенко Катерина Логвинівна,**

кандидат фіз.-мат. наук **Нікітченко Микола Степанович.**

Провідна установа: Національний технічний університет України «Київський політехнічний інститут».

Захист відбудеться «——» ———— 199 р. о ——— год. на засіданні спеціалізованої вченої ради Д 01.01.23 при Київському національному університеті ім. Тараса Шевченка за адресою:

252127 Київ 127, проспект Академіка Глушкова, 2, корп. 6, факультет кібернетики, ауд. 40.
(Тел. (044) 266 12 58, факс 266 12 48,
E-mail: vpsh α icchq.univ.kiev.ua).

З дисертацією можна ознайомитись у науковій бібліотеці Київського університету ім. Тараса Шевченка (Київ, вул. Володимирська, 58).

Автореферат розісланий «——» ———— 199 р.

Учений секретар
спеціалізованої вченої ради **В. П. Шевченко**

Загальна характеристика роботи.

Актуальність теми. На сучасному етапі висока продуктивність обчислювальних машин досягається новими архітектурними рішеннями, удосконаленням їх елементної бази, розробкою високоефективного системного програмного забезпечення, методів і алгоритмів обробки інформації. При цьому приоритетним напрямком у цій галузі є створення паралельних ЕОМ, методів та засобів паралельних обчислень.

Протягом всієї історії розвитку обчислювальної техники використання паралельних архітектур розглядається як єдина можливість зростання продуктивності обчислювальних систем при збереженні звичайної, фон-нейманівської, архітектури компонентів. На даний момент теорія паралельних архітектур досягла великої різноманітності. Однією з найновіших розробок у цьому напрямку є багатопроцесорні розподілені системи на базі трансп'ютерів.

Одночасно з розвитком теорії апаратного забезпечення паралельних обчислювальних систем відбувається еволюція засобів конструювання прикладного програмного забезпечення для паралельних обчислювальних хомплексів. Одним із домінуючих та натуральних підходів є розширення існуючих послідовних мов програмування новими операторами паралельного програмування. При цьому виділяються підходи, що базуються на динамічний та статичній моделях паралелизму.

Статичний підхід характеризується тим, що ще до виконання програми відома структура паралельних процесів та взаємовідносина між ними. При динамічному підходу паралельні процеси можуть створюватися і знищуватися протягом виконання програми, зв'язки між процесами можуть перебудовуватися.

Трансп'ютерна технологія програмування, що базується на мові Оккам, є розвитком засобів паралельного програмування, що засновані на статичній моделі паралелізму. Враховуючи факт існування широкого класу алгоритмів, які потребують використання динамічного і рекурсивного паралелізму, існуючих засобів оккамівської технології програмування може виявитись недостатнім для їх реалізації.

Звідси випливає актуальність розробки таких методів і засобів паралельного програмування, яки дають змогу доповнити оккамівську модель паралелізму засобами опису, управління і синхронізації динамічними, паралельно-рекурсивними процесами. Вони повинні суттєво розширити клас алгоритмів, що піддаються розпаралелюванню на трансп'ютерних комплексах, розширивши свободу у виборі підходів до реалізації паралельного програмного забезпечення.

Мета та завдання дисертації. Робота присвячена дослідженню та створенню ефективних методів організації паралельного програмування та обчислень в багатопроцесорних комплексах на базі трансп'ютерів.

Метою дисертації є теоретична розробка та реалізація технології динамічного паралельно-рекурсивного програмування для багатопроцесорних трансп'ютерних систем, дослідження можливостей, наданих такою технологією для опису, керування і синхронізації обчислювальних процесів за допомогою динамічних паралельно-рекурсивних просторів, розробка конкретних паралельних алгоритмів, що використовують можливості розроблених інструментальних засобів.

Поставлена мета досягається розв язком таких завдань:

- аналіз можливостей трансп'ютерних архітектур для створення засобів динамічного, паралельно-рекурсивного програмування;
- аналіз існуючих засобів опису статичного і динамічного паралелізму, наданих оккамівською технологією програмування;
- дослідження моделі керуючих просторів в асинхронних паралельних обчисленнях і досвід її використання, з метою застосування цієї моделі для реалізації технології програмування;
- визначення набору операторів паралельного програмування, достатнього для опису складних паралельно-рекурсивних структур, що динамічно змінюються;
- реалізація розширення мови програмування Сі новими конструкціями, які надають відповідні засоби програмного інтерфейсу для розробників;
- використання одержаної технології для опису складних керуючих просторів паралельних обчислювальних процесів;

створення теоретичного обгрунтування та реалізація конкретних паралельних алгоритмів, які використовують нові засоби динамічного і рекурсивного паралелізму, надані розробленою технологією.

Методи досліджень базуються на досягненнях фундаментальних та прикладних досліджень в області теорії паралельних процесів та обчислень, технологій паралельного програмування. Застосовувався теоретико-алгоритмічний підхід к дослідженню рекурсивно-паралельних процесів.

Наукова новизна роботи. Наукова новизна роботи полягає у всесторонньому теоретичному дослідженні можливостей ПАРУС (ПАРУС-Параллельные Асинхронные Рекурсивно Управляемые Пространства) технології програмування, в створенні нового конкретного розширення мови Сі для трансп'ютерних комплексів, що дістав назву ПАРУС-Сі, за допомогою набору засобів опису динамічного і рекурсивного паралелізму. Це розширення суттєво збільшує можливості опису класів алгоритмів, що піддаються розпаралелюванню на трансп'ютерних комплексах, забезпечує створення масштабованих, архітектурно незалежних програм, збільшує швидкість розробки паралельного програмного забезпечення для трансп'ютерних комплексів. Для демонстрації можливостей запропонованої технології розроблений та математично обгрунтований новий паралельно-рекурсивний алгоритм обчислення модулярної експоненти для надвеликих натуральних чисел, який використовується як одна із складових частин систем захисту інформації з загальними ключами.

Практична цінність роботи полягає в дослідженні можливостей практичної реалізації ПАРУС-технології програмування, в створенні конкретного інструментального засобу ПАРУС-Сі для багатопроцесорних трансп'ютерних комплексів. Створений новий паралельний алгоритм модулярного експоненціювання, який може застосовуватися у програмах, що реалізують обчислення надвеликих натуральних чисел, а також

використовуватись в системах захисту інформації з відкритими ключами, що вимагають високої продуктивності у реальному масштабі часу.

Апробація результатів роботи. Результати роботи доповідалися та обговорювалися на 3-й міжнародній конференції з паралельних компьютерних технологій "PACT-95", м.Санкт-Петербург, вересень 1995 р., на семінарах у відділі Інтелектуалізації інформаційних технологій Інституту кібернетики ім. В.М.Глушкова НАН України та кафедри математичної информатики Київського національного університету ім.Т.Г.Шевченка.

Робота виконувалась у рамках програм з держбюджетної та госпдоговірної тематики, що велись на кафедрі Математичної інформатики Київського національного університету ім.Т.Г.Шевченка та віддилу № 465 Інтелектуалізації інформаційних технологій Інституту кібернетики ім.В.М.Глушкова НАН України:

тема №2 ВИ-КУ-10-УО, розділ 08.03, програма 08;

тема №4 "Макроме", розділ 09.06, програма 09;

що виконувались відповідно до рішення Кабінету Міністрів України від 26 червня 1995 р. №452-007;

тема ИП 465.03 "Розробка принципів створення систем інтелектуалізації інформаційних та робототехнічних комплексів";

тема ГКНТ КН 465.02 Шифр проекту 05.02.03./011-95 "Системи штучного інтелекту базовані на асоціативно-пошукових алгоритмах", комплексного проекту 05.02.03/001К-95.

Публікації. По темі дисертації опубликовано 4 друковані роботи.

Структура роботи. Дисертаційна робота складається з вступу, трьох глав, висновків,списку літературі та двох додатків.

Загальний обсяг роботи - 112 сторінок, у тому числі 10 малюнків. Бібліографія - 72 найменувань.

Зміст роботи.

У вступі обгрунтовується важливість та актуальність теми дисертації, викладені мета і методика досліджень. Сформульовані основні результати, досягнені в процесі виконання работи, їх наукова новизна. Коротко викладається зміст розділів дисертаційної роботи.

Перша глава присвячена вивченню сучасних тенденцій у розвитку паралельних архітектур і методів створення паралельного програмного забезпечення.

У розділі 1.1 розглядаються основні тенденції в галузі створення апаратних засобів реалізації паралельних обчислень, а також викладені основні шляхи розвитку систем конструювання прикладного програмного забезпечення для багатопроцесорних обчислювальних комплексів.

Наводяться класифікація паралельних архітектур, що базується на співвідношенні числа потоків команд і потоків даних, а також способи застосування різних архітектур при організації паралельних обчислень. Основна увага приділяється паралельним ЕОМ архітектури з множинним потоком команд і множинним потоком даних (МКМД), як найбільш універсальному і гнучкому виду. Однією з сучасних розробок архітектур типу МКМД є трансп'ютерні системи.

Вивчаються сучасні підходи до створення паралельного програмного забезпечення. Серед них розширення компіляторів новими операторами паралельного програмування, додавання нового мовного рівня для опису паралелізму, розширення компіляторів засобами виявлення паралельних операцій, визначення нової мови і системи компіляції.

Аналізується метод макроконвейєрної обробки та сімейство мов МАЯК, як один из можливих підходів до паралельного програмування для архітектур МКМД.

Розглядається стан сучасної технології паралельного програмування для трансп'ютерних систем, обмеження які вона накладає на розробку паралельних алгоритмів.

Наводяться деякі результати досліджень, які спрямовані на розширення діапазону алгоритмів, що розпаралелюються на трансп'ютерних комплексах, та є засобами високого рівня для створення паралельних програм, таких, що скорочують час розробки програмного забезпечення. Серед них - комплекс розпаралелювання Сі-программ для трансп'ютерних систем, розширення мови Сі Сінапс/3, система Quickplay.

Обґрунтовується необхідність створення технології динамічного паралельно-рекурсивного програмування для багатопроцесорних трансп'ютерних комплексів, які мають вигляд розширення послідовної мови засобами створення, керування та синхронізації параллельних процесів, які утворюють динамічні керуючі простори.

У розділі 1.2 досліджуються архітектурні рішення підтримки паралельних обчислень, що реалізовані у мікропроцесорах на базі трансп'ютерів.

Нове сімейство мікропроцесорів, що одержали назву "трансп'ютери", фірма INMOS (Великобританія) розробила спеціально для використання в паралельних розподілених системах. Порівнюючи з іншими мікропроцесорами, трансп'ютер має таку специфіку: в нього вмонтовані послідовні лінії зв'язку для взаємодії з іншими трансп'ютерами, та реалізована апаратна підтримка розподілу часу.

Сімейство трансп'ютерів являє собою набір системних компонентів, які можуть бути використані для створення високопродуктивних паралельних систем. Завдяки такому поєднанню забезпечується можливість побудови таких систем із різних представників цього сімейства. Всі трансп'ютери мають апаратну підтримку паралельного виконання і забезпечують високу продуктивність при виконанні диспетчеризації процесів, операцій міжпроцесного і міжтрансп'ютерного зв'язку. З метою створення мультитрансп'ютерних комплексів може бути з'єднана в мережу будь-яка кількість процесорів, що підключаються. Продуктивність таких комплексів збільшується лінійно із збільшенням кількості процессорів, що використовуються.

У розділі 1.3 досліджуються сучасні системи опису паралелізму для трансп'ютерних комплексів, що базуються на концепції взаємодіючих послідовних процесів Хоара. Розглядаються засоби, надані оккамівською технологією програмування, а також базовані на ній засоби більш високого рівня.

Основною універсальною мовою для програмування трансп'ютерів є Оккам. Він є мовою високого рівня, що дає змогу описувати паралельні взаємодіючі процеси. Трансп'ютер має апаратну підтримку основних конструкцій Оккама, що дає змогу виконувати програми, написані на ньому з ефективністю асемблера.

Мова програмування Оккам дозволяє описувати прикладну задачу, як набір паралельно працюючих процесів, які взаємодіють по каналах. У такій нотації кожний Оккам-процес описує поведінку одного компонента прикладної задачі, а кожний канал описує взаємодію між компонентами. На основі Оккама були розроблені інші мови паралельного програмування на трансп'ютерах: Паралельний Сі, Паралельний Паскаль, Паралельний Фортран та інші.

У них були реалізовані деякі засоби вищого рівня, ніж ті, які представляє мова Оккам. Ці засоби дещо вирішують проблему динамічного паралелізму, хоча існування в них деяких обмежень значно звужує клас алгоритмів, що розв'язуються за допомогою цих методів. Система ПАРУС-Сі, що розроблюється, є новим засобом високого рівня для створення паралельного програмного забезпечення для трансп'ютерних комплексів. Воно базується на поняттях керуючого простору і інформаційному каналі. Засади ПАРУС-технології з'явились раніше, ніж перший опис мови Оккам.

Друга глава присвячена питанням створення системи динамічного паралельно-рекурсивного програмування для трансп'ютерних комплексів, зокрема, мови програмування ПАРУС-Сі.

Розділ 2.1 описує основні положення теорії асинхронних керуючих просторів у паралельних обчисленнях, яка є в основі ПАРУС-технології програмування для трансп'ютерних комплексів. ПАРУС-технологія

програмування грунтується на концепції керуючого простору (КП). КП - це граф, що динамічно змінюється, за допомогою якого описується логічна і комунікаційна структура досліджуваної задачі (системи), та відбуваються динамічні зміни в ній. Іншими словами КП - це система керуючих точок і каналів, які задають структурну організацію паралельного процесу. Керуюча точка - це система послідовних алгоритмічних модулів, що визначають один процес. Канал - це інформаційний зв'язок між точками.

Система програмування ПАРУС надає програмні засоби, що забезпечують побудову і модифікацію КП, збереження та передачу інформації шляхом КП, навантаження КП алгоритмічними модулями. Завдяки керуючим просторам, ПАРУС-технологія забезпечує підтримку:

- структурного паралельного програмування на основі як статичного, так і динамічного паралелізмів;
- опис рекурсії за даними та рекурсії за керуванням. Забезпечує шляхом взаємодії не тільки синхронізацію, але й видозміну керування (на основі засобів базової мови);
- опис на логічному рівні в явному вигляді розподілу ресурсів і схему перекомутації в якості результату паралельної програми;
- розробки систем віртуальних ПАРУС-машин, які забезпечують можливість паралельної обробки інформації на різних рівнях деталізації;

Теорія керуючих просторів, що лежить в основі ПАРУС-технології програмування, пропонувалась як узагальнення моделі взаємодіючих послідовних процесів, що полягає в явному впровадженні поняття керуючого простору обчислювального процесу.

У розділі 2.2 вводяться оператори ПАРУС-ядра для мови Сі, що реалізують створення, знищення, керування, передачу інформації та синхронізацію паралельних процесів у керуючому просторі алгоритмів.

Наводиться опис угод, яких повинен дотримуватися розробник ПАРУС-программ. Описується набір функцій ПАРУС-Сі, що реалізують динамічне, паралельно-рекурсивне керування процесами.

Керуючою точкою в ПАРУС-Сі служить паралельний процес, який виконується на одному із вузлів комплексу, що базується на описаній

спеціальним способом процедурі мови Сі, і зв'язаний програмними каналами з іншими керуючими точками.

Сукупність взаємодіючих керуючих точок визначає керуючий простір даної системи. Завжди існує коренева керуюча точка, яка є початком для побудови керуючого простору.

Множину функцій, що реалізують ядро ПАРУС для мови Сі можно поділити на три категорії: функції керування точками керуючого простору, функції обміну інформацією - синхронізації, функції сервісу та налагодженню. Дані функції не виключають можливості використання всієї різноманітності бібліотечних функцій, які входять до стандартної поставки мови Сі.

Реалізований варіант ПАРУС-Сі включає в себе такі фунції для задання паралельних взаємодіючих процесів. Функції *root_sp* і *task_sp* служать для реєстрації процедур, що описують керуючі точки системи і надання їм логічних номерів (дескрипторів). Функція *setp* динамічно створює паралельний процес на основі зареєстрованої процедури. Забезпечується можливість рекурсивного створення паралельних процесів на основі однієї і тієї самої процедури. Функція endp служить для видалення керуючої точки з керуючого простору системи.

Функції обміну інформацією служать для пересилання повідом-лень різної структури і синхронізації між керуючими точками. У параметрах цих функцій звичайно вказується дескриптор точки, на яку спрямоване повідомлення, номер каналу і саме повідомлення. При цьому розробнику не обов'язково враховувати архітектурні особливості трансп'ютерної системи. Всі проблеми, пов'язані з маршрутизацією повідомлення по трансп'ютерній мережі бере на себе ядро ПАРУС. Ці функції реалізують прийом та передачу наборів даних різних типів: рядків (*sendp, getp*), цілочисельних змінних (*sendp_i, getp_i*), змінних з плаваючою комою (*sendp_f, getp_f*), а також функції, що реалізують прийом та передачу масивів цілочисельних змінних і змінних з плаваючою комою (*sendp_ia, getp_ia, sendp_if, getp_if*).

Третя група включає функції *reportp(), printp()*. reportp здійснює виведення інформації про стан паралельної системи. До неї входить інформація про різноманітні параметри виконання ПАРУС-програм, яка може використовуватися з метою налагодження. За допомогою функції *printp* може здійснюватися виведення інформації з керуючої точки на дисплей.

У розділі 2.3 описується внутрішня структура і принципи організації системи програмування ПАРУС-Сі. Розглядаються переваги кільцевої топології для релізації систем розподілення обчислень.

Система паралельного програмування, що реалізується на багатопроцесорному комплексі, повинна забезпечувати:

- використання існуючого програмного та математичного забезпечення, базових алгоритмічних мов;
- паралельну і розподілену обробку інформації;
- ефективне завантаження і реконфігурацію системи;
- стійкість до збоїв і відмов устаткування
- обробку інформації у реальному масштабі часу.

Найбільш придатною основою для розв'язування поставлених завдань є використання трансп'ютерних мереж.

Для реалізації ПАРУС-системи для трансп'ютерного кільця був обраний варіант організації кільця з маркерним доступом. Основний алгоритм, який описує роботу вузлів кільцевої мережі з передачею маркера, має такий вигляд.

```
повторювати
    повторювати
        передавати одержані дані ;
    доки не буде отримано вільного маркеру
    якщо є пакет даних для передачі то
        послати зайнятий маркер ;
        послати пакет даних ;
        чекати доки не буде отримано зайнятий маркер ;
        послати вільний маркер ;
        отримати (без передачі) пакет даних ;
    інакше
        відіслати вільний маркер ;
    кінець якщо
доки система працює
```

Маркер являє собою інформаційний пакет, що має специфічну структуру і в загалі має такий вигляд:

НВ - номер вузла

КП - код повідомлення

П1...П6 - параметри повідомлення

Дані - вміст повідомлення (якщо є).

Довжина поля даних міститься в одному з параметрів. ПАРУС-система являє собою сукупність вихідних модулів, у які шляхом макропідстановки вбудовуються тексти процедур, написаних у процесі розробки ПАРУС-додатка. Таким чином нема необхідності динамічно вантажувати об'єктні модулі процедур в процесі виконання програми. В результаті такої реалізації маємо такі переваги в створенні паралельного програмного забезпечення для мультитрансп'ютерних систем: масштабованість, конфігураційна незалежність, автоматичне розпаралелювання, динамично-рекурсивний виклик процедур, координаційність.

Глава 3 присвячена аспектам практичного використання мови ПАРУС-Сі для реалізації конкретних алгоритмів паралельних обчислень.

У розділі 3.1 досліджуються можливі підходи у використанні системи ПАРУС-Сі для побудови складних керуючих просторів паралельних процесів.

Наводяться прийоми використання операторів системи для реалізації паралельно-рекурсивних, динамічних структур деревоподібного типу, N-ступеневого конвейера, куба. На конкретних прикладах описується використання різних конструкцій системи для здійснення керування і синхронізації керуючих просторів.

Розділ 3.2 присвячений опису паралельного алгоритму обчислення модулярної експоненти, що має велике значення для реалізації систем захисту інформації з відкритими ключами. Наводяться основні теореми, що лежать в основі даного алгоритму, Описуються системи шифрування, які можуть використовувати даний алгоритм.

. Велика обчислювальна ємкість алгоритму модулярного експоненціювання над великими числами є основним гальмуючим фактором у системах захисту інформіції з відкритими ключами, що базую.ься на схемі логарифмічного обміну Діффі і Хеллмана або методі шифрування RSA.

Зокрема, кодування за методом RSA забезпечується обчисленням односторонньою функцією RSA з потаємним ходом $y = x^e \pmod n$, де x — додатне ціле, що не перевищує $n = pq$, p і q – великі нерівні числа, такі, що $\varphi(n)=(p-1)(q-1)$, e – додатне ціле, що не перевищує φ(n), для якого НОД $(e,\varphi(n))=1$. Іншими словами x –повідомлення, що шифрується, а y – зашифроване повідомлення. Відкритим ключем є числа n и e.

Для дешифрування використовується обернена функція, що має вигляд: $x = y^d \pmod n$, де d – єдине ціле, менше за n, і задовольняє умови $de = 1 \pmod{\phi(n)}$. Показник d, необхідний для дешифрування знаходимо за допомогою алгоритму Евклида, який обчислює НОД $(e,\varphi(n))$. $\varphi(n)$, в свою чергу, можна знайти, знаючи прості p і q, $n=pq$. При зикористанні в ключах надвеликих натуральних чисел знаходження потрібних простих множників з метою злому захисту є практично неможливим.

Схема логарифмічного обміну також базується на модулярному піднесенні до ступеня натуральних чисел великої розмірності. Отже, реалізація паралельного виконання алгоритму модулярного експоненціювання має велике значення для побудови високонадійних і високопродуктивних систем захисту інформації.

Паралельний алгоритм обчислення модулярної експоненти базується на використанні деяких властивостей лінійних форм Фібоначчі максимального рангу, зокрема на побудові лінійного дерева Фібоначчі для показника ступеня. Тобто для обчислення виразу $x^y \bmod z$ показник повинен бути представлений у вигляді лінійної форми Фібоначчі максимального рангу t: $y = aF_{t-1} + bF_t$. Числа

a і b в свою чергу також можуть розкладатися в лінейні форми Фібоначчі максимального рангу. Таким чином отримаємо лінейне дерево Фібоначчі. Глибина дерева Фібоначчі дорівнює O(loglog y) (Теорема 1).

Таким чином, вираз $x^y \bmod z$ перетворюється на вираз $(x^{F_{k-1}})^a * (x^{F_k})^b \bmod z$. Числа $x^{F_{k-1}}$ і x^{F_k} можуть обчислюватися паралельно.

Рекурсивно повторюючи розклад у лінійну форму Фібоначчі коефіцієнтів a і b, ми зведемо обчислення $x^y \bmod z$ до операцій розкладу у лінійну форму і рекурсивного піднесення до ступеня чисел Фібоначчі. Загальний час паралельного алгоритму потребує O(Tloglog y) паралельних операцій. Де T - час, потрібний до піднесення до ступеня, яке є числом Фібоначчі (Теорема 2). Операція піднесення до ступеня типу X^{F_k}, де F_k - це k-е число Фібоначчі, потребує k операцій множення.

У розділі 3.3 наводяться описи структури ПАРУС-програми, яка реалізує паралельний алгоритм обчислення модулярної експоненти для натуральних чисел великої розмірності.

Джерело паралелізму алгоритму, що розглядується, полягає в тому, що процес розкладу показника у лінійну форму Фібоначчі подається у вигляді бінарного дерева. Вузли одного рівня у цьому дереві мають незалежні набори даних і можуть обчислюватись паралельно.

Візуально процес поділяється на дві фази: проходження вниз по дереву Фібоначчі для показника ступеня і підіймання вгору з обчисленням $X^{F_k} = X^{F_{k-1}} * X^{F_{k-1}}$. Рекурсивна функція загалом має такий вигляд:

```
int Exponenta(X,Y,Z) //X^Y modZ
{
 if (Y==0) return (1);
 if (Y==1) return (X);
 /*Розкласти показник у лінійну форму Фібоначчі*/
 Y = A * F_{k-1} + B * F_k
 R1=Exponenta( X^{F_{k-1}}, A, Z);
 R1=Exponenta( X^{F_k}, A, Z);
```

return(R1*R2 mod Z);

}

Алгоритм розкладу у лінійну форму базується на двох перетвореннях:

1. Якщо $b > a$, то враховуючи, що $F_{k-1} = F_{k+1} - F_k$, підвищуємо ранг розкладу: $aF_{k-1} + bF_k = (b -)F_k + aF_{k+1}$.

2. (Підготовка a і b до перетворення 1). Якщо $b < a$, то подавши $aF_{k-1} + bF_k$ як $(a - sF_k)F_{k-1} + (b + sF_{k-1})F_k$ знаходимо якесь s, що відповідає умові $\frac{a}{F_k} < s < \frac{a-b}{F_{k+1}}$. Якщо s знайдено, то перевизначимо a і b : $a = a - sF_k, b = b + sF_{k+1}$. Якщо ні, то розклад максимального рангу досягнуто.

Написання ПАРУС-програми складається з двох етапів: реалізація обчислювальної та координаційної частин алгоритму. Для опису обчислювальної частини використовуються бібліотечні функції мови Сі, а також послідовні функції бібліотеки арифметичних операцій над натуральними числами великої розмірності. Реалізація координаційної моделі алгоритму здійснювалась за допомогою засобів високого рівня системи ПАРУС-Сі по організації та синхронізації динамічними паралельно-рекурсивними керуючими просторами.

Програма тестувалася на трансп'ютерному комплексі, який складається з процесорів T800 і T425 із загальною продуктивністю 270 MIPS і 12 MFLOPS. Хост-машиною є персональний комп'ютер на базі процесору Intel Pentium.

У **висновку** наводяться основні результати виконаної роботи.

Підсумки

Основні результати роботи.

- Проведен аналіз сучасного стану технології паралельної обробки інформації на основі багатопроцесорних трансп'ютерних комплексів.
- Досліджена модель паралельних асинхронних процесів, що задаються за допомогою керуючиї просторів. Ця модель є основою для реалізації новий технології паралельно-рекурсивного програмування на трансп'ютерному комплексі.
- Визначен та обгрунтован набір операторів паралельного програмування, достатнього для опису динамічних паралельно-рекурсивних структур.
- Розроблено і реалізовано паралельне розширення мови програмування Сі, що забезпечуєть нові єфективні засоби створення паралельного програмного забезпечення для трансп'ютерних комплексів. Основними характеристиками ціх засобов є масштабованість, конфігураційна незалежність, автоматичне розпаралелювання, динамично-рекурсивний виклик процедур.
- На основі розроблених методів і засобів досліджені процедури розробки паралельних рекурсивних структур за допомогою керуючих просторів.
- Запропонован, теоретично обгрунтован і реалізован новий паралельно-рекурсивний алгоритм модулярного возведення у ступень для чисел великої розмірності. Цей алгоритм є демонстрацією можливостей запропонованної технології для вирішування задач захисту інформації.

Список публікацій.

За темою дисертації опубліковані такі роботи:

1. Анісімова О.А., Гриценко Д.В. Використання трансп'ютерних мереж в задачах візуалізації складних динамічних систем ітеративного типу.//Вісник київського університету.-№2 -1993

2. Анісімова О.А., Гриценко Д.В. Порівнення ПАРУС та трансп'ютерної технології програмування.//Там же.-№ 1 .-1994

3. Гриценко Д.В. Основы параллельного программирования на языке ПАРУС-Си.-Киев, 1995.-31 с.-(Препр. / НАН Украины. Ин-т кибернетики им. В.М.Глушкова; 95-22).

4. Гриценко Д.В. Параллельно-рекурсивный алгоритм вычисления модульной экспоненты.//Доповіді національної академії наук України. №3 , 1996.

Гриценко Д. В. Методы и средства параллельно-рекурсивного программирования для транспьютерных комплексов. Рукопись. Диссертация на соискание ученой степени кандидата физико-математических наук по специальности 01.05.03 — математическое и программное обеспечение вычислительных машин и систем. Киевский национальный университет им. Тараса Шевченко. Киев, 1996.

Защищается диссертация, в которой изучается модель параллельных вычислений на основе управляющих пространств. Разработано и реализовано параллельное расширение языка Си, обеспечивающее новые эффективные средства создания параллельного программного обеспечения для транспьютерных комплексов. Предложен, теоретически обоснован и реализован при помощи созданных программных средств новый параллельно-рекурсивный алгоритм модулярного возведения в степень для натуральных чисел большой размерности.

A model of parallel computations, which is based on the theory of controlling spaces is elaborated in this thesis. A new concurrent extension of the programming language C is suggested. This gives effective tools for creating transputer software. A new parallel algorithm for modular exponentiation for large integers is suggested and studied in detalis. This algorithm is implemented by means of developed programming technology.

Ключові слова: паралельні обчислення, трансп'ютери, паралельне програмування, керуючі простори, мова програмування Сі, модулярне піднесення до ступеня.

Підписано до друку 08.04.96. Формат 60×84/16. Папір для розмнож. апар. Офс. друк. Ум. друк. арк. 0,93. Ум. фарбо-відб. 1,05. Обл.-вид. арк. 1,0. Зам. 221. Тираж 100 прим.

Редакційно-видавничий відділ з поліграфічною дільницею
Інституту кібернетики імені В. М. Глушкова НАН України
252022 Київ 22, проспект Академіка Глушкова, 40