

УДК 681.513

И.А. ОСОТОВ, О.Ф. МИХАЛЬ

Харьковский национальный университет радиоэлектроники, Украина

ИССЛЕДОВАНИЕ ЭФФЕКТИВНОСТИ ЛОКАЛЬНО-ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ НА МНОГОЯДЕРНЫХ ПРОЦЕССОРАХ

Локально-параллельная обработка информации, при которой обеспечивается эффективная реализация, в частности, нечётко-логических вычислений на процессорах общего назначения, приобретает специфические черты при реализации на однокристалльных многоядерных процессорах. Анализ тенденций развития процессорной техники и вероятных структурных изменений многоядерных процессоров позволяет прогнозировать перспективность расширенного использования локально-параллельных методов обработки. В машинном эксперименте реализован выигрыш в эффективности порядка 15% для локально-параллельного алгоритма с потоком обработки данных внутри распараллеленной части.

Ключевые слова: локально-параллельная обработка информации, многоядерные процессоры.

Введение

Гарантоспособность, как концепцию разработок надёжных и безопасных систем, собранных из ненадёжных компонентов, целесообразно развивать исходя из реального прототипа: *живых системах* (ЖС). Актуальность такого подхода определяется апробированностью ЖС.

В [1] рассмотрены иерархические параллельно-организованные ЖС, с ретроспективой на тенденции развития *вычислительных устройств* (ВУ).

Показано, что перспективно использовать ВУ общего назначения с программной реализацией на них *локально-параллельных* (ЛП) [2, 3] методов обработки информации.

Новизна применения принципа ЛП связана с многоядерностью процессоров ВУ. На настоящий момент ЛП обработка на многоядерных процессорах не достаточно изучена. В частности для известных [2, 3] ЛП алгоритмов не разработаны версии их выполнения на многоядерных процессорах; ввиду чего уже разработанные [4, 5] методы исследования для многоядерных процессоров не апробированы на известных ЛП алгоритмах. Этим несколько тормозится развитие многоядерной ЛП алгоритмической базы.

Между тем, ожидаемый прирост эффективности составляет десятки процентов, чем определяется целесообразность проведения разработок в данном направлении.

Целью настоящей работы является рассмотрение метода исследования и представление результатов по эффективности применения ЛП на ВУ с многоядерными процессорами на примере конкретного алгоритмов обработки нечёткой информации [2, 3].

1. Иерархичность и параллеленность

В плане реализации гарантоспособности, неоспоримым образцом являются ЖС. Базовые биологические (органические высокомолекулярные) материалы, из которых собраны ЖС, имеют малую устойчивость и критичны к условиям внешней среды. Тем не менее, вследствие иерархической структурной организации, ЖС демонстрируют долговременное устойчивое функционирование и приспособляемость к изменениям условий внешней среды – сохранение *эксплуатационным характеристикам* (ЭХ) в широких диапазонах по нескольким параметрам ($T_1, T_2, \dots, T_j, \dots$). В этом они значительно превосходят менее сложно организованные, и казалось бы более надёжные, технические системы, разработанные человеком. Важнейшими особенностями обеспечения высоких ЭХ в ЖС, являются многоуровневость (иерархичность) и параллельная организация в каждом из уровней (внутриуровневый локальный параллелизм).

Иерархичность и распараллеленность ЖС в упрощённом описании выглядят следующим образом. Имеются объекты ЖС и внешняя среда в которой находятся ЖС. Каждый из объектов ЖС является многоуровневым по сложности. На i -м уровне сложности ЖС состоит из функционально однотипных объектов, каждый из которых состоит из объектов $(i-1)$ -го уровня. Понятие «внешняя среда» – относительное: объект $(i+1)$ -го уровня является внешней средой для объектов $(i-1)$ -го уровня. Поэтому группы объектов $(i-1)$ -го уровня объединяются в объекты i -го уровня сложности с целью противостоять воздействиям внешней среды. Для этого на i -м уровне однотипные элементы $(i-1)$ -го уровня взаи-

моорганізуються так, що в цілому на i -м рівні забезпечуються зворотні зв'язки, підтримуючі об'єкт i -го рівня в рамках параметрів $(t_{\min,j,i}; t_{\max,j,i})$, де в індексах: j – номер параметра зовнішньої середовища T_j ; i – номер рівня ієрархічної організації. При цьому для ЖС в цілому по j -му параметру зовнішньої середовища T_j забезпечується вложеність інтервалів значень параметрів:

$$\begin{aligned} t_{\min,j,1} \geq t_{\min,j,2} \geq t_{\min,j,3} \geq \dots \geq t_{\min,j,(i-1)} \geq t_{\min,j,i} \geq \dots; \\ t_{\max,j,1} \leq t_{\max,j,2} \leq t_{\max,j,3} \leq \dots \leq t_{\max,j,(i-1)} \leq t_{\max,j,i} \leq \dots \end{aligned} \quad (1)$$

В результаті оказується, що об'єкт ієрархії ЖС, знаходячись на верхньому рівні, ефективно функціонує в середі, руйнівній для вихідних матеріалів (базовий рівень).

Чоловік являється частинним випадком (одним із видів) ЖС, тому приведені описання (системний генезис) плюс продовження на соціальний аспект розвитку, стосуються також і до нього. Діяльність людини в системному плані направлена на розширення своїх можливостей і вдосконалення виконуваних функцій. Тому розробляється навколишнє середовище людини (від примітивних інструментів праці до ВУ і засобів телекомунікації) являється його антропоморфним продовженням: посиленням можливостей, покращенням системних ЕХ. Виходячи з цього, доцільно розробляти з дотриманням ієрархічності і паралельності в кожному з рівнів. Цим, зокрема, забезпечуються надійність, взаємозамінюваність, можливість реконфігурації при частковій відмові, а також додаткова функціональність за рахунок паралельного (фоновий) використання резервних можливостей.

Така загальна картина, часто залишається поза уваги за технічними подробицями.

Далі ми розглянемо деякий частинний випадок «введення» нового (програмного) розпаралелення всередині існуючого апаратного рівня. Особливістю являється те, що ВУ багатоядерне і розглядається внутрішньорівневою програмною розпаралеленням на рівні, який сам уже являється апаратно розпаралеленим.

2. Локальна паралельність

Принцип локально-паралельної (ЛП) обробки інформації розглянуто в [1-3]. Ефективність ЛП обробки ілюструється наступним алгоритмічним прикладом. Нехай існує n виразень $c_i = a_i + b_i$; $i = 1, 2, \dots, n$. Значення a_i і b_i задані. Потрібно знайти значення c_i . На однопоточному ВУ ця задача розв'язується за $4n$ кроків: по кожному з виразень завантажити a_i , завантажити b_i , виконати сумування, вивести результат c_i . Згідно ЛП принципу, значення a_i і b_i конкатенуються

$$\begin{aligned} A &= a_1 \oplus 0 \oplus a_2 \oplus 0 \oplus \dots \oplus 0 \oplus a_i \oplus 0 \oplus \dots \oplus 0 \oplus a_n; \\ B &= b_1 \oplus 0 \oplus b_2 \oplus 0 \oplus \dots \oplus 0 \oplus b_i \oplus 0 \oplus \dots \oplus 0 \oplus b_n; \end{aligned}$$

а сукупність всіх n результатів c_i отримується як конкатенація

$$C = c_1 \oplus 0 \oplus c_2 \oplus 0 \oplus \dots \oplus 0 \oplus c_i \oplus 0 \oplus \dots \oplus 0 \oplus c_n$$

в ході виконання 4-крокової операції $C = A+B$.

В даному прикладі роздільні нулі вставляються між конкатенантами a_i і b_i реалізації переноса розряду при сумуванні. Числа a_i і b_i повинні бути додатними і мати обмежену розрядність, так щоб конкатенації A , B і C по довжині в бітах не перевищали розрядність процесора.

Зрозуміло, n -кратний вииграш досягнуто без урахування витрат часу на операції конкатенації-деконкатенації, охоплюючи обчислювальний блок. Якщо мова йде про коротку обчислювальну процедуру (типу $C = A+B$), то ЛП варіант реалізації швидше за все буде проігнаним по часу внаслідок наявності допоміжних операцій формування A і B і витягання результатів c_i з C . Неважко в реальному випадку обчислювальний блок достатньо складний і включає більше значущий набір різноманітних операцій, виконуваних над наборами даних без проміжних конкатенації-деконкатенації, – вииграш в продуктивності може бути значущим. Вииграш зростає з збільшенням розрядності процесора, а також з «загрубленням» системи – зниженням точності представлення даних при скороченні розмірів конкатенируваних сегментів. Обидва вказані напрями в відповідній інтерпретації – мають конкретні технічні застосування.

Ріст розрядності процесорів являється довготривалою стійкою тенденцією розвитку процесорної техніки. Розрядність регістра процесора це об'єм безпосередньо адресуваної оперативної пам'яті. В кінцевому рахунку, об'єм пам'яті визначається гранично допустима розмірність завдань, які можуть бути розв'язані на даному ВУ за прийнятний час. Розвиток техніки ВУ стимулює виникнення нових обчислювальних завдань, які в свою чергу стимулюють зростання вимог до технічних характеристик знову розробляються ВУ. Ця петля зворотного зв'язку працює вже десятиліттями і немає принципових причин відходу від даної парадигми.

Зниження точності представлення даних до прийнятних меж – допустимо і доцільно, зокрема, в системах, базуючих на використанні експертних знань. Носителями вихідних (первоначальних) експертних знань є люди. В зв'язку з цим повинно звертати увагу на відомий з психології факт: людина може стійко розрізняти лише невелике число об'єктів: 5 ± 2

шт. При работе с большим числом объектов различение идёт с разделением на подуровни: предъявляемые объекты сначала группируются в блоки по 3–7 шт., затем последовательно осуществляется обработка блоков. Таким образом, для кодирования одного из реально различаемых экспертом состояний требуется до 3 бит информации. Использование более высокой разрядности может предполагать округление, сглаживание, То есть, возможно, искажение исходных экспертных знаний.

Рассмотрим ЛП в плане практической реализации.

3. Многоядерные процессоры

В настоящее время разрядность процессоров общего назначения возросла до 64 бит. Имеются так же специализированные графические процессоры, применяемые, в частности, в игровых приставках, имеющие разрядность 128 и 256 бит. При этом тактовая частота ядра процессора достигла гигагерцевого диапазона. Дальнейшая конъюнктура рынка процессоров, по-видимому, складывается так, что конкурентоспособный рост производительности ВУ не обеспечивается ростом разрядности регистров процессора. Для основной массы задач общего назначения разрядность 64 бита является достаточной для представления данных, а аппаратное ЛП-представление данных, по видимому, не обеспечивается существующими инженерными (схемными и конфигурационными) решениями в гигагерцевом диапазоне. В техническом отношении подобная задача, разумеется, является не тривиальной: речь идёт о микроминиатюризации в области СВЧ, где помимо чисто схемных решений существенны так же и конфигурационные факторы. Вместе с тем процессорная техника должна развиваться и характеристики должны наращиваться, поскольку развёрнуто и профинансировано несколько конкурирующих организационных структур, занимающихся разработкой и производством процессоров. В сложившейся ситуации, по-видимому, оптимальным по сложной группе критериев «капиталовложения – сроки разработки – выигрыш в производительности» является курс на создание многоядерных однокристалльных процессоров. Данное направление было впервые (преждевременно) реализовано специалистами фирмы Digital Equipment Corporation ещё в 1992 г. В виде 64-разрядного RISC процессора DEC-Alpha.

В техническом отношении перспективность однокристалльной многоядерности состоит в том, что каждое из ядер может быть высокочастотным, а взаимодействие между ядрами реализуется на более низких частотах через кеш-память 2-го, 3-го и т.д.

уровней. Таким образом, иерархические уровни образуются здесь по вложенности, объёмам памяти и убыванию рабочих частот. Вложенность уровней кеш-памяти соответствует (аналогична) изложенной выше (п. 1) концепции об иерархичности ЖС, а последовательность частот, убывающая по уровням кеш-памяти, – вложенности диапазонов значений параметров (1).

Могут так же реализовываться различные иерархии дисциплин по автономному или совместно-му использованию различных уровней кеш-памяти. Представляется достаточно вероятным, что прогресс по числу ядер будет сопровождаться прогрессом по организации многоуровневой кэш-памяти, направленным на повышение рабочих частот и усиление обобществления памяти для всё более эффективного взаимодействия между отдельными ядрами. В конечном счёте путь развития определяется накоплением инженерного опыта при разработке процессорных систем с освоением СВЧ-диапазона.

В рамках описанного (возможно, ограниченно) понимания перспектив развития процессорной техники, сегодняшние многоядерные однокристалльные процессоры предстают как промежуточный этап перед полностью СВЧ-частотными процессорами с большой разрядностью регистров и реконфигурируемостью по числу ядер. В самом деле: n -ядерный однокристалльный процессор с разрядностью каждого ядра m может быть эквивалентен m -разрядному одноядерному процессору, если обеспечена связь (передача данных) на тактовой частоте ядер между старшим разрядом i -го и младшим разрядом $(i+1)$ -го ядер. В этом случае регистры отдельных ядер процессора конкатенированы в гиперрегистры виртуального единого гиперядра.

В связи со сказанным, перспективна реализация ЛП алгоритмов на многоядерных структурах. Технически и программно – ЛП и многоядерность не противоречат друг другу, а в идейном отношении – они направлены на реализацию единой цели: внутриуровневой распараллеленности системы.

Рассмотрим комбинированное ЛП многоядерное алгоритмическое решение на примере ЛП алгоритма операции нахождения нечёткого теоретико-множественного пересечения двух наборов значений функций принадлежности [2, 3].

4. Алгоритм

В табл. 1 приведено пошаговое описание ЛП алгоритма операции пересечения. Операндами A и B являются конкатенации сегментов, в каждом из которых находится масштабированное значение функции принадлежности. Количество и длина сегментов (на примере 10-сегментного 3-битового варианта,

удобного для реализации на 32-разрядных ВУ) определяются константами

$$E_1=119304647_{10}=000111000111000111000111000111_2;$$

$$E_2=954437176_{10}=111000111000111000111000111000_2;$$

с использованием которых выделяются множества нечётных и чётных (соответственно) сегментов. Числа 10 и 2 в индексах при значениях E_1 и E_2 обозначают системы счисления. Константы L_1 и L_2 предназначены для помещения «сигнальных» единиц в младшие разряды сегментов. В 10-сегментном 3-битовом варианте:

$$L_1=1090785344_{10};$$

$$L_2=653667127_{10}.$$

Смысл остальных обозначений в таблице 1 следующий: $E_0=E_1+E_2$; $L_0=L_1+L_2$; and, or и xor – битовые логические операции; $(C \gg n)$ – битовый сдвиг регистра процессора, содержащего C , на n бит в сторону младших разрядов с потерей содержимого; при 3-битовых сегментах $n = 3$.

Далее после склеивания (Шаг 4) формируются прямая и инверсная маски, с использованием которых собирается результат (Шаг 7).

Таблица 1

Пошаговое описание ЛП операции

Шаг	Метод реализации $\min(A,B)$
1	Прореживание: $A_1=A$ and E_1 ; $A_2=A$ and E_2 ; $B_1=B$ and E_1 ; $B_2=B$ and E_2 .
2	Установка меток и арифметическое вычитание: $C_1=(A_1 \text{ or } L_2)-B_1$; $C_2=(A_2 \text{ or } L_1)-B_2$.
3	Сопряженное прореживание: $C_1=C_1$ and L_2 ; $C_2=C_2$ and L_1 ; или $C_1=C_1$ and E_2 ; $C_2=C_2$ and E_1 .
4	Склеивание: $C=C_1$ "or" C_2 .
5	Формирование маски: $M=C-(C \gg n)$.
6	Формирование инверсной маски: $\bar{M} = M \text{ xor } E_0$.
7	Результат: $(A \text{ and } \bar{M}) \text{ or } (B \text{ and } M)$;

Легко видеть, что Шаги 1-3 содержат по две группы действий: производимые с чётными и нечётными сегментами. Эти группы являются алгоритмически распараллеливаемыми. Затем в Шагах 4-7 результаты объединяются и вычисления идут последовательно.

Таким образом, на одноядерных процессорах весь алгоритм может выполняться последовательно, на многоядерных – Шаги 1-3 параллельно, затем оставшая часть – последовательно. В результате, на

многоядерных процессорах потенциально достижим дополнительный прирост эффективности – сокращение времени обработки.

Разработанная ранее [4, 5] методика машинного эксперимента по анализу эффективности была адаптирована с учётом структурной схемы ЛП алгоритма (табл. 1). Согласно плану эксперимента, замеры затрат времени на выполнение проводились по фиксации моментов времени для одно- и двух-ядерного вариантов реализации алгоритма при одинаковом числе итераций во внутреннем цикле.

Коэффициент эффективности определён как отношение времён выполнения одноядерного варианта (аппаратно-последовательного) к двух-ядерному (с внутрипроцессорным аппаратным распараллеливанием). То есть, если коэффициент превышает единицу – имеется выигрыш по эффективности.

Программная реализация механизма внутрипроцессорного распараллеливания алгоритма таблица 1 рассмотрена в [4, 5]. Применены стандартные языковые конструкции инструментария WinAPI: функции `SetThreadAffinityMask` и `SetThreadIdealProcessor`.

В ходе *предварительных* экспериментов установлено, что распараллеливание одинарной вычислительной операции не приводит к выигрышу в производительности. Одноядерная реализация алгоритма выполняется даже несколько быстрее многоядерной. Это обусловлено тем, что выигрыш времени от распараллеливания теряется на фоне затрат на системные процедуры по выделению конкретных ядер под конкретные операции. Выигрыш появляется если используются массивы входных данных. В этом случае распараллеливание вычисления происходит на потоках, привязанных к конкретным ядрам вычислительного узла.

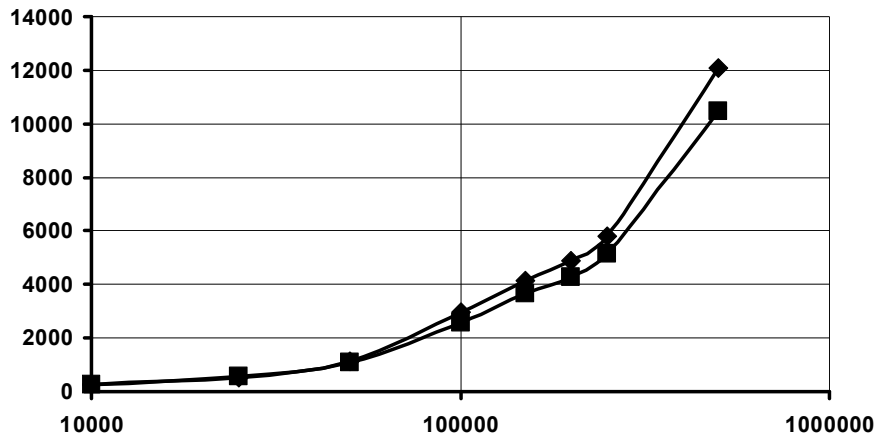
С учётом обнаруженной особенности, *окончательный* план эксперимента был скорректирован: в двух-ядерном варианте итерации по Шагам 1-3 и 4-7 были выделены в две отдельные группы, с сохранением одинакового числа итераций в каждой группе. В исправленном варианте структура распределения ядер процессора задаётся однократно для всей группы данных, а не переназначается всякий раз заново. Таким образом, исчезают потери времени на переназначение, которые в действительности непосредственно не относятся в работе алгоритма таблица 1.

Типовые результаты (Intel Pentium Dual-Core E5200; 2.5 GHz; 4 Gb RAM; OS Windows 7 x86) иллюстрируются графиками рис. 1 а и б.

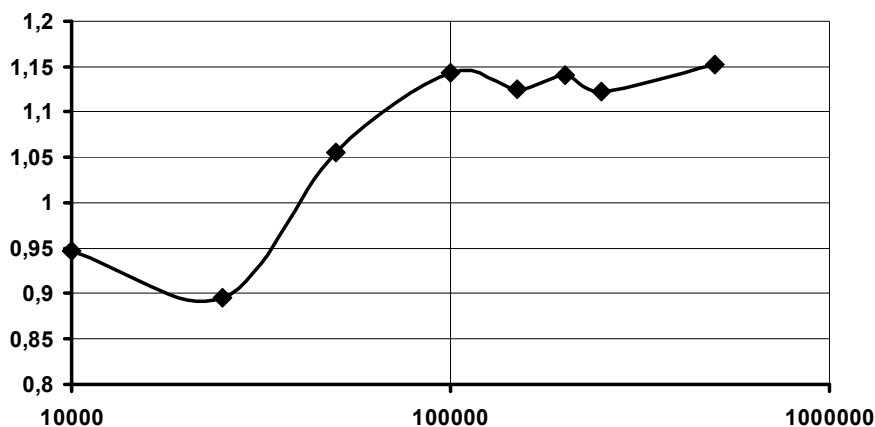
На графиках по оси абсцисс отложено число итераций в логарифмических координатах. На рис. 1, а представлены зависимости времени выполнения циклов итераций в микросекундах. для одно-

ядерной (верхняя кривая) и многоядерной (нижняя кривая) реализаций. Согласно числовым данным, на участке до ~30000 итераций график одноядерной кривой проходит ниже графика многоядерной, что отображено зависимостью коэффициента эффектив-

ности рис. 1, б. Таким образом, представленный участок изменения числа итераций – в пределах двух порядков – отображает переход от проигрышности к выигрышности по эффективности в точке ~30000 итераций.



а



б

Рис. 1. Зависимости времени выполнения (а) и коэффициента эффективности (б) от числа итераций в машинном эксперименте.

Полученные результаты демонстрируют выигрыш эффективности ~15% при поточной организации вычислений внутри распараллеленной части алгоритма, что характерно в частности при обработке массивов однотипных данных.

Как следует из графика рис. 1, б, при объеме итераций свыше 100000 эффективность «выходит на полку», что соответствует выигрышу от распараллеливания по сравнению с потерями на однократное распределение потоков по ядрам процессора. Переход от проигрышности алгоритма к выигрышности может быть охарактеризован как «плавный» или «резкий», зависимости от масштаба итераций, применительно к конкретной решаемой задаче.

График рис. 1, б иллюстрирует изменение числа итераций на 2 порядка и охватывает переход ме-

жду двумя «полками». Таким образом, выигрышность алгоритма критична к динамическому диапазону по числу итераций. Если алгоритм должен применяться к выборкам разного объема в районе «перехода между двумя полками», то по вопросу эффективности требуется дополнительное исследование.

Выводы

На примере алгоритма локально-параллельной операции нечеткого теоретико-множественного пересечения, методом машинного эксперимента показано, что выигрыш в эффективности от использования внутрипроцессорной многоядерности может составлять десятки процентов.

С применением стандартного инструментария WinAPI реализован выигрыш в эффективности ~15%. Подобная структура приемлема, в частности, при обработке крупных массивов данных.

Литература

1. Мохаммад, Али. Перспективы реализации локально-параллельных вычислений на многоядерных процессорах [Текст] / Али Мохаммад, О.Ф. Михаль // Радиоэлектронные и компьютерные системы. – 2008. – № 6 (33). – С. 234 – 237.

2. Михаль, О.Ф. Принципы организации систем нечеткого регулирования на однородных локально-параллельных алгоритмах [Текст] / О.Ф. Михаль, О.Г. Руденко // Управляющие системы и машины. – 2001. – № 3. – С. 3 – 10.

3. Михаль, О.Ф. Моделирование распределенных информационно-управляющих систем средствами локально-параллельных алгоритмов обработки нечеткой информации [Текст] / О.Ф. Михаль // Про-

блемы бионики. Всеукраинский межведомственный научно-технический сборник. – Х.: ХНУРЭ, 2001. – № 54. – С. 28 – 34.

4. Осотов, И.А. Применение локально-параллельных алгоритмов с процессорно-зависимой многопоточностью. [Текст] / И.А. Осотов, О.Ф. Михаль // Информационные технологии в навигации и управлении: состояние и перспективы развития. Материалы первой международной научно-технической конференции. – К.: ДП "ЦНДІ НІУ", 2010. – С. 53.

5. Осотов, И.А. Реализация локально-параллельных алгоритмов с использованием процессорно-зависимой многопоточности [Текст] / И.А. Осотов, О.Ф. Михаль // Информатика, математическое моделирование, экономика: Сборник научных статей по итогам Международной научно-практической конференции, г. Смоленск, 22 апреля 2011 г. В 2-х томах. Т. 1 - Смоленск: Смоленский филиал АНО ВПО ЦС РФ "Российский университет кооперации", 2011. – С. 14 – 20.

Поступила в редакцию 12.03.2012

Рецензент: д-р техн. наук, проф. Ю.П. Кондратенко, Черноморский государственный им. Петра Могилы, Николаев, Украина.

ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ ЛОКАЛЬНО-ПАРАЛЕЛЬНИХ ОБЧИСЛЕНЬ НА БАГАТОЯДЕРНИХ ПРОЦЕСОРАХ

І.О. Осотов, О.П. Міхаль

Локально-паралельна обробка інформації, при якій забезпечується ефективна реалізація, зокрема, нечітко-логічних обчислень на процесорах загального призначення, набуває специфічні риси при реалізації на однокристальних багатоядерних процесорах. Аналіз тенденцій розвитку процесорної техніки і вірогідних структурних змін багатоядерних процесорів дозволяє прогнозувати перспективність розширеного використання локально-паралельних методів обробки. У машинному експерименті реалізований виграш в ефективності порядку 15% для локально-паралельного алгоритму з потоком обробки даних усередині розпаралеленої частини.

Ключові слова: локально-паралельна обробка інформації, багатоядерні процесори.

INVESTIGATION OF EFFICIENCY OF LOCAL-PARALLEL CALCULATIONS BASED ON MULTIKERNEL PROCESSORS

I.A. Osotov, O.Ph. Mikhal

Locally-parallel processing of information in which an efficient implementation is provided, in particular, the fuzzy-logic algorithms on general purpose processors acquires specific features of the implementation on single-chip multi-core processors. Analysis of trends in processor technology and the likely structural changes in multi-core processors allows predicting the prospects of increased use of locally-parallel processing techniques. An efficiency gain of about 15% for locally-parallel algorithm with the flow of data within the parallelized parts is achieved.

Key words: locally-parallel processing of information, multicore processors.

Осотов Иван Александрович – аспирант кафедры ЭВМ Харьковского национального университета радиоэлектроники, Харьков, Украина.

Михаль Олег Филиппович – д-р техн. наук, проф. кафедры ЭВМ Харьковского национального университета радиоэлектроники, Харьков, Украина.