

Марценюк Євген Вікторович – старший викладач, Національний аерокосмічний університет ім. М. С. Жуковського «Харківський авіаційний інститут», м. Харків; тел.: (066) 272-12-44; e-mail: ev_martsen@mail.ru

Марценюк Евгений Викторович – старший преподаватель, Национальный аэрокосмический университет им. Н. Е. Жуковского «Харьковский авиационный институт», г. Харьков; тел.: (066) 272-12-44; e-mail: ev_martsen@mail.ru

Martsenyuk Evgen Victorovich – Associate Professor, National Aerospace University «Kharkiv Aviation Institute», Kharkov; tel.: (066) 272-12-44; e-mail: ev_martsen@mail.ru

Зелений Юрій Олексійович – провідний конструктор відділу турбін ДП «Івченко-Прогрес», м. Запоріжжя; тел.: (0612) 65-62-95; e-mail: yura-green@mail.ru.

Зеленый Юрий Алексеевич – ведущий конструктор отдела турбин ГП «Ивченко-Прогресс», г. Запорожье; тел.: (0612) 65-62-95; e-mail: yura-green@mail.ru.

Zelenyj Yuri Alekseevich – Leading Project Designer of Turbine Department, SE Ivchenko-Progress, Zaporizhia; tel.: (0612) 65-62-95; e-mail: yura-green@mail.ru.

Резник Сергій Борисович – начальник бригади, заступник начальника відділу турбін ДП «Івченко-Прогрес», м. Запоріжжя; тел.: (0612) 65-62-95; e-mail: s.b.reznik@gmail.com.

Резник Сергей Борисович – начальник бригады, заместитель начальника отдела турбин ГП «Ивченко-Прогресс», г. Запорожье; тел.: (0612) 65-62-95; e-mail: s.b.reznik@gmail.com.

Reznik Sergey Borisovich – Group Manager, Deputy Head of Turbine Department, SE Ivchenko-Progress, Zaporizhia; tel.: (0612) 65-62-95; e-mail: s.b.reznik@gmail.com.

Клирик Ростислав Ростиславович – інженер-конструктор 1-ї категорії відділу турбін ДП «Івченко-Прогрес», м. Запоріжжя; тел.: (0612) 656295; e-mail: 03504@ivchenko-progress.com.

Клирик Ростислав Ростиславович – инженер-конструктор 1-й категории отдела турбин ГП «Ивченко-Прогресс», г. Запорожье; тел.: (0612) 656295; e-mail: 03504@ivchenko-progress.com.

Klimik Rostislav Rostislavovich – Senior Design Engineer of Turbine Department, SE Ivchenko-Progress, Zaporizhia; tel.: (0612) 656295; e-mail: 03504@ivchenko-progress.com.

Кулик Тамара Василівна – старший науковий співробітник, доцент, ПНДЛ діагностики авіаційних двигунів Національного аерокосмічного університету ім. М. С. Жуковського «Харківський авіаційний інститут», м. Харків; тел.: (057) 788-45-25; e-mail: aedlab@gmail.com.

Кулик Тамара Васильевна – старший научный сотрудник, ПНИЛ диагностики авиационных двигателей Национального аэрокосмического университета им. Н. Е. Жуковского «Харьковский авиационный институт», г. Харьков; тел.: (057) 788-45-25; e-mail: aedlab@gmail.com.

Kulik Tamara Vasilevna – Senior Researcher at the Laboratory of Aircraft Engines Diagnostics, National Aerospace University «Kharkiv Aviation Institute», Kharkov; tel.: (057) 788-45-25; e-mail: aedlab@gmail.com.

УДК 519.64:004.032.24

В. О. МИЩЕНКО, Б. В. ПАТОЧКИН

ОРГАНИЗАЦИЯ И ОПТИМИЗАЦИЯ КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ ДИФРАКЦИОННЫХ ПРОЦЕССОВ МЕТОДАМИ ДИСКРЕТНЫХ ОСОБЕННОСТЕЙ

При всестороннем исследовании антенн, ретрансляторов и других подобных устройств существует потребность в расчетах параметров дифракции электромагнитных волн на металлических экранах для многих вариантов формы и волновых чисел. Время компьютерного моделирования критично. Перспективными в такой ситуации становятся вычислительные методы дискретных особенностей, в частности, метод дискретных токов (МДТ). В статье разработан метод организации компьютерного моделирования по МДТ с использованием особенностей архитектуры персональных компьютеров и даны результаты проверки эффективности этого метода.

Ключевые слова: электромагнитная волна, дифракция, волновое число, математическая модель, МДО, архитектура компьютера, тайлинг, векторные регистры, OpenMP, скорость вычислений.

Введение. При конструировании и исследовании антенн, ретрансляторов и волноведущих систем существует потребность в расчетах параметров дифракции электромагнитных волн на металлических экранах для многих вариантов формы и значений волнового числа. Для успешного проведения таких расчётов в требуемом объёме с необходимой точностью важнейшим показателем является скорость компьютерного счёта. В данной статье предполагается использование вычислительных *методов дискретных особенностей* (МДО), которые относятся к наиболее эффективным методам численного решения граничных интегральных уравнений задач математической физики, в том числе дифракционных [1] (причём используются уравнения с *сингулярными ядрами* или другие близкие к ним по свойствам *псевдодифференциальные уравнения*). При этом проведение расчётов для широкополосных антенн или *фазированных антенных решеток* (ФАР) с проверками точности за счёт удвоения числа дискретных особенностей сталкивается с большими затратами времени вычислений на персональном

компьютере (ПК) даже, если выбрана 2D постановка задач. Общий подход к ускорению счёта состоит в том, что программная реализация хорошо изученного последовательного алгоритма заменяется реализацией параллельной версии модифицированного для этой цели алгоритма в предположении доступности физически параллельных вычислений. Этот подход вступает в противоречие с весьма ограниченным числом процессорных ядер ПК.

Однако для достижения данной цели на ПК имеются резервы, связанные с особенностями их архитектуры, которые нам удалось задействовать для ускорения вычислений МДО. Наряду с традиционным распараллеливанием вычислений [2, 3], которое использует наличие нескольких параллельно работающих процессоров, удалось задействовать ускоренный обмен между регистрами и кэшем памяти [4 – 6], а также специальные регистры процессора, допускающие параллельное выполнение операций по *принципу векторных процессоров* [7].

Указанные резервы использованы путём создания соответствующих модификаций вычислительного метода дискретных токов (МДТ из семейства МДО) и разработки программных модулей на языках программирования Ада [4, 5] и С++ [6]. В языке Ада для организации параллельных вычислений имеются удобные средства высокого уровня (собственно, поэтому мы его использовали), но эффекты ускорения несколько снижаются при его реализации на базе Run Time System. Язык С/С++ позволяет наиболее полно воспользоваться возможностями системного программирования. В этом случае, однако, требуется понимание и знание как языка С++, так и отдельных деталей работы с памятью и процессором, что мешает использованию библиотеки модулей ускорения МДТ специалистами по моделированию, которые не являются одновременно и специалистами по программированию.

Целью данной работы является построение для нужд исследователей дифракционных процессов в технических системах практичной информационной технологии компьютерного моделирования МДТ, которая сохранит в существенном достигнутую скорость вычислений.

Постановка задачи. Для достижения поставленной цели решались две задачи.

Первая – разработка проекта библиотеки компьютерного моделирования дифракции электромагнитных волн на идеально проводящих экранах (в 2D постановке задач), которая позволяет без специальных знаний или долгого своего изучения организовывать и проводить вычисления, существенно ускоренные за счёт многопоточности процесса, мультиядерности процессора, оптимизации работы с кэшем памяти и векторных регистров процессора.

Вторая – верификация и оценка степени эффективности тех приложений для вычислительных экспериментов по моделированию дифракционных процессов, которые можно собрать из модулей данной библиотеки.

Важнейшие модули этой библиотеки, которую мы назовём SDCM (Speed Discrete Currents Modeling), были фактически уже разработаны на языке программирования С++ в процессе разработки модификаций МДТ для использования эффектов *тайлинга* и параллельных вычислений [7]. Однако они для своего понимания и использования в качестве компонент требуют от пользователя, который организует вычисления, свободного владения низкоуровневыми средствами управления параллельными процессами. Такой квалификации у специалистов по моделированию дифракции ожидать нельзя.

Поэтому в данной работе архитектура и основные системные решения разработаны на базе открытого стандарта для распараллеливания программ OpenMP. Выбор пал на это высокоуровневое средство распараллеливания потому, что при его использовании не изменяется однопоточный код, и он хорошо зарекомендовал себя у прикладников понятностью команд и простотой использования при удовлетворительной эффективности [8].

Верификация модулей библиотеки SDCM основана на сравнении получаемых результатов моделирования с результатами работы традиционных реализаций МДТ.

Оценка эффективности по скорости работы проводилась на основе серий вычислительных экспериментов с разными реализациями в разных режимах. При этом оценивалась предполагаемая потеря эффективности как плата за удобство настройки готовых компонент при создании приложений компьютерного моделирования.

Используемые математические модели дифракционных процессов. Пусть в плоской задаче дифракции электромагнитных волн на идеально проводящих контурах зависимость от времени задается множителем $e^{i\omega t}$. Амплитуды наблюдаемого полного поля

$$\vec{E}^{tot} = \vec{E} + \vec{E}_0, \quad \vec{H}^{tot} = \vec{H} + \vec{H}_0, \quad (1)$$

где \vec{E}_0, \vec{H}_0 – падающее поле; E, H – рассеянное поле, причем, как известно, достаточно рассмотреть следующие два случая.

Случай E поляризации:

$$E_0 = (0, 0, U_0), \quad H_0 = \left(\frac{1}{i\omega\mu} \frac{\partial U_0}{\partial y}, \frac{1}{i\omega\mu} \frac{\partial U_0}{\partial x}, 0 \right) \quad (2)$$

(и тогда аналогичную структуру имеют векторы полей $\vec{E}^{tot}, \vec{H}^{tot}$ и E, H).

Случай H поляризации:

$$H_0 = (0, 0, U_0), \quad E_0 = \left(-\frac{1}{i\omega\epsilon} \frac{\partial U_0}{\partial y}, \frac{1}{i\omega\epsilon} \frac{\partial U_0}{\partial x}, 0 \right), \quad (3)$$

причём аналогичную структуру имеют векторы полей \vec{E}^{tot} , \vec{H}^{tot} и E , H например,

$$H = (0, 0, U), E = \left(-\frac{1}{i\omega\epsilon} \frac{\partial U}{\partial y}, \frac{1}{i\omega\epsilon} \frac{\partial U}{\partial x}, 0 \right). \quad (3)$$

Алгоритмы численного решения таких задач, основанные на МДТ [1], зависят от типа поляризации падающего поля (электрическая или магнитная) и вида контуров, представляющих сечения экранов (замкнутые или разомкнутые). Учитывая, что речь идёт о применении известного метода, ограничимся здесь для примера формулами, которые описывают точные исходные и дискретные (аппроксимирующие граничные интегральные уравнения) модели дифракционного процесса, в случае H поляризации для замкнутых контуров.

Единственная ненулевая компонента магнитной составляющей рассеянного поля U удовлетворяет уравнению Гельмгольца вне множества $C = \bigcup_{n=0}^{N-1} C_n$, где C_n – n -контур, а N – количество контуров:

$$\Delta U(X) + k^2 U(X) = 0, \quad x \in R^2 / \bar{C}, \quad X = (x, y). \quad (5)$$

На контуре C выполняется условие Неймана

$$\left. \frac{\partial U(X)}{\partial n} \right|_c = - \left. \frac{\partial U_0(X)}{\partial n} \right|_c \quad (6)$$

на бесконечности – условия излучения Зоммерфельда

$$U(X) \Big|_{|X| \rightarrow \infty} = O\left(\frac{1}{\sqrt{r}}\right), \quad r = |X|, \quad (7)$$

$$\frac{\partial U(X)}{\partial r} - ikU(X) \Big|_{x \rightarrow \infty} = O\left(\frac{1}{r^{3/2}}\right). \quad (8)$$

Согласно МДТ эти модели, имеющие дифференциальную форму, преобразуются к форме с граничными псевдодифференциальными операторами. Вычислительный метод, подлежащий ускорению, предполагает приведение каждого из таких уравнений к каноническому виду:

$$\frac{i}{2\pi} \int_0^{2\pi} \frac{j(t)}{\sin^2\left(\frac{\tau-t}{2}\right)} dt + b(\tau) \cdot \int_0^{2\pi} \ln \left[\sin\left(\frac{\tau-t}{2}\right) \right] j(t) dt + \int_0^{2\pi} K(\tau, t) j(t) dt = 4h_\tau(\tau) E_\tau^0(\tau); \quad (9)$$

(в случае замкнутого контура), где,

$$b(\tau) = -k^2 \left((x')^2 + (y')^2 \right) \cdot \frac{i}{\pi}; \quad (10)$$

$$K(\tau, t) = \frac{\partial}{\partial \tau} \frac{\partial}{\partial t} H_0^{(2)}(\kappa L) - k^2 h(\tau, t) H_0^{(2)}(\kappa L_0) u(t) - a(\tau) \frac{1}{\left[\sin\left(\frac{\tau-t}{2}\right) \right]^2} + b(\tau) \ln \left[\sin\left(\frac{\tau-t}{2}\right) \right], \quad (11)$$

где $j(t)$ – искомая непрерывная функция на контуре (определяющая плотность «наведенного» тока в металле экрана).

МДТ позволяет ставить в соответствие таким моделям дискретные модели, обеспечивающие приближенное нахождение плотностей токов на экранах (т.н. *дискретные токи*) со сколь угодно высокой точностью за счёт выбора достаточно большого числа узлов дискретизации (параметр N). Воспроизведём конечные формулы:

$$\frac{2i}{2n+1} \sum_{k=0}^{2n} j_n(\phi_k^n) A(\phi_0, \phi_k^n) - b(\phi_0) \sum_{k=0}^{2n} j_n(\phi_k^n) B(\phi_0, \phi_k^n) + \sum_{k=0}^{2n} K(\tau, \phi_k^n) \frac{2\pi}{2n+1} = 4h_\tau(\tau) E_\tau^0(\tau), \quad (12)$$

где $N = 2n + 1$, искомый вектор является дискретным приближением к искомой функции и

$$A(\phi_0, \phi_k^n) = \left[\frac{\sin^2 \frac{n}{2} (\phi_0 - \phi_k^n)}{\sin^2 \frac{1}{2} (\phi_0 - \phi_k^n)} - \frac{n \sin\left(n + \frac{1}{2}\right) (\phi_0 - \phi_k^n)}{\sin \frac{1}{2} (\phi_0 - \phi_k^n)} \right], \quad n = (N-1)/2, \quad k = 0 \dots N; \quad (13)$$

$$B(\phi_0, \phi_k^n) = \frac{2}{2n+1} \left[\ln 2 + \sum_{p=1}^n \frac{\cos p(\phi_0 - \phi_k^n)}{p} \right]. \quad (14)$$

Очевидно, что (12) – это формально совместная система линейных алгебраических уравнений. Доказательство невырожденности таких систем и сходимость их решений к точному решению в определённых пространствах см. в [1].

Реализация быстрого варианта МДТ в системе параллельного программирования OpenMP. Технику распараллеливание с помощью OpenMP, которую мы применили, лучше пояснить на предельно упрощенном примере. Пусть это будет умножение матриц.

При организации данных в программе используются двумерные динамические массивы:

```
double** A;
double** B;
double** C;
```

Исходный алгоритм, представленный на языке программирования C/C++ имеет вид:

```
for (int i = 0; i < size; i++) {
    for (int j = 0; j < size; j++) {
        for (int k = 0; k < size; k++) {
            C[i][j] += A[i][k] * B[k][j];
        }
    }
}
```

(15)

В силу его простоты достаточно добавить одну строку:

```
#pragma omp parallel for shared(A, B, C)
```

Эта строка сообщает компилятору, что следующий далее цикл можно распараллелить, считая массивы в памяти A, B и C *общими ресурсами*. Под общим ресурсом здесь подразумевается, что каждая операция внутри цикла, связанная с этими массивами переменных, должна выполняться над ними, а не над их локальными копиями в процессах. При отсутствии декларации shared(A, B, C) в каждом потоке создавалась бы копия этих переменных. Хотя это и не критично в данном примере, но в приложении МДТ при использовании данных со стека привело бы к получению неправильных результатов. В итоге реализация матричного умножения с параллельной обработкой получает вид:

```
#pragma omp parallel for shared(A, B, C)
for (int i = 0; i < size; i++) {
    for (int j = 0; j < size; j++) {
        for (int k = 0; k < size; k++) {
            C[i][j] += A[i][k] * B[k][j];
        }
    }
}
```

(16)

Посмотрим, каким образом это сказывается на ускорении вычислений (табл. 1).

Таблица 1 – Результаты тестирования параллельной на базе OpenMP (15) и исходной (14) реализации демо примера

Размерность матриц	1000	2000	4000	10000
Время выполнения фрагмента (15), с	2,3	21,46	274	6947
Время выполнения фрагмента (16), с	8,1	84	1091	1744
Коэффициент ускорения при 4-х ядрах	3,52	3,91	3,98	3,98

Таким образом, ускорение выполнения алгоритмов, в которых можно выделить 4 слабо взаимодействующих процесса, теоретически может на 4-ядерном процессоре практически достигать значения 4. Использование OpenMP позволяет с ростом размерностей перемножаемых матриц достигать максимального теоретического ускорения с незначительным недобором (0.5%). Однако мы видим, что при средних (с точки зрения современных компьютерных вычислений) размерностях матриц «накладные расходы» по времени могут составить около 14%, что заставляет ставить задачу об обязательном исследовании этого вопроса для параллельных реализаций более сложных алгоритмов, когда взаимодействие процессов носит систематический характер.

Архитектура и схемы распараллеливания вычислений библиотеки SDCM. На диаграмме компонентов рис. 1 показана агрегация модулей библиотеки SDCM. В случае минимального объема результата, который хотят получить организаторы вычислительного эксперимента МДТ, интересуются обычно диаграммой направленности и функцией плотности тока, который соответствует рассеянному полю (часто в преобразованном для удобства виде). Эта функция позволяет сразу или в дальнейшем строить картину рассеянного или полного поля. При минимальном запрашиваемом результате, а также при необходимости численного построения поля в заданных точках внешней области пользователю пакета SDCM достаточно написать свою главную программу, которая активизирует компоненты, показанные на рис. 1. При других постановках задач (например, спектральных) придется выполнять настройку компонент. Труднее всего это сделать для Synchronization. Эта трудность отпадает для версии библиотеки, основанной на OpenMP.

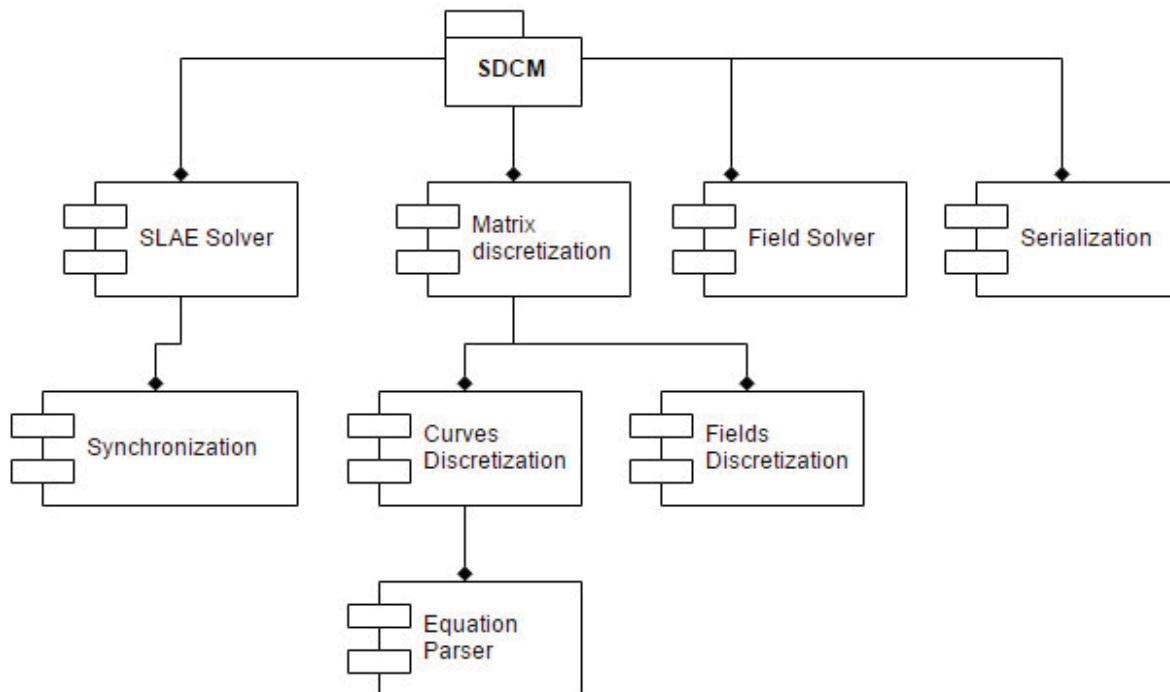


Рис. 1 – Модульная архитектура библиотеки SDCM.

Сведения о назначении и размерах модулей собраны в табл. 2.

Таблица 2 – Общие сведения о реализации компонентов в библиотеке SDCM

Модуль :	Краткое описание :	LOC :
SLAE Solver :	решение СЛАУ вычислительным методом компактной схемы Гаусса в вариантах: <ul style="list-style-type: none"> ● стандартный, ● модифицированный для оптимизации работы с кэш памятью, ● параллельная модификация оптимизации работы с кэш памятью, ● параллельная модификация с использованием векторных регистров при оптимизации работы с кэш памятью 	155
Curves Discretization :	сохранение информации о контурах в виде системы узлов вычислительного метода (МДТ)	140
Fields Discretization :	вычисление правых частей СЛАУ	70
Matrix Discretization :	формирование матрицы МДТ: <ul style="list-style-type: none"> ● естественный алгоритм, ● модифицированный для оптимизации работы с кэш памятью, ● параллельная модификация оптимизации работы с кэш памятью, ● параллельная модификация с использованием векторных регистров при оптимизации работы с кэш памятью 	120
Field Solver :	вычисление диаграммы направленности и рассеянного поля в заданных точках	50
Synchronization :	при использовании OpenMP не используется	103
Equation Parser :	аналитическое задание контуров	260
Serialization :	ввод-вывод по формату JSON	350

Распараллеливание заполнения матрицы и вычисления диаграммы направленности реализуется слабо взаимодействующими процессами, и обеспечивается командой компилятору по обсуждённой выше схеме:

```

#pragma omp parallel for shared(matrix)
for(int i = 0; i < matrix_size; i++)
{
    for(int j = 0; j < matrix_size; j++)
    {
        matrix[i][j] = calculate_element(...);
    }
}
    
```


Распараллеливание блочного варианта *компактной* схемы метода Гаусса, которая используется для решения системы линейных уравнений МДТ как раз с целью облегчения параллельных модификаций [4], осложняется систематическим и нетривиальным взаимодействием вычислительных потоков. Однако нами создан следующий код:

```

for (int i = 0; i < block_number; i++) {
    // middle blocks
    // full U and L part
    for (int k = 0; k < i; k++)
        for (int ii = lBorder(i), rbi = rBorder(i); ii < rbi; ii++)
            for (int ik = lBorder(k), rbk = rBorder(k); ik < rbk; ik++)
                for (int ij = lBorder(i); ij < rbi; ij++)
                    luMatrix[ii][ij] -= luMatrix[ii][ik] * luMatrix[ik][ij];
    // U and L part last block
    for (int ii = lBorder(i), rbi = rBorder(i); ii < rbi; ii++) {
        for (int ik = lBorder(i); ik < ii; ik++) {
            for (int ij = ii; ij < rbi; ij++)
                luMatrix[ii][ij] -= luMatrix[ii][ik] * luMatrix[ik][ij];
            for (int ij = ii + 1; ij < rbi; ij++)
                luMatrix[ij][ii] -= luMatrix[ij][ik] * luMatrix[ik][ii];
        }
    }
    // end of L part
    for (int ij = ii + 1, rbi = rBorder(i); ij < rbi; ij++)
        luMatrix[ij][ii] /= luMatrix[ii][ii];
}

//U blocks
#pragma omp parallel for shared(luMatrix, i)
for (int j = i + 1; j < bn; j++) {
    for (int k = 0; k < i; k++)
    {
        for (int ii = lBorder(i), rbi = rBorder(i); ii < rbi; ii++)
            for (int ik = lBorder(k), rbk = rBorder(k); ik < rbk; ik++)
                for (int ij = lBorder(j), rbj = rBorder(j); ij < rbj; ij++)
                    luMatrix[ii][ij] -= luMatrix[ii][ik] * luMatrix[ik][ij];
    }
    // U part last block
    for (int ii = lBorder(i), rbi = rBorder(i); ii < rbi; ii++)
        for (int ik = lBorder(i); ik < ii; ik++)
            for (int ij = lBorder(j), rbj = rBorder(j); ij < rbj; ij++)
                luMatrix[ii][ij] -= luMatrix[ii][ik] * luMatrix[ik][ij];
}

//L blocks
#pragma omp parallel for shared(luMatrix, i)
for (int j = i + 1; j < bn; j++) {
    for (int k = 0; k < i; k++)
        for (int ij = lBorder(j), rbj = rBorder(j); ij < rbj; ij++)
            for (int ik = lBorder(k), rbk = rBorder(k); ik < rbk; ik++)
                for (int ii = lBorder(i), rbi = rBorder(i); ii < rbi; ii++)
                    luMatrix[ij][ii] -= luMatrix[ij][ik] * luMatrix[ik][ii];
    // L part last block
    for (int ii = lBorder(i), rbi = rBorder(i); ii < rbi; ii++) {
        for (int ik = lBorder(i); ik < ii; ik++)
            for (int ij = lBorder(j), rbj = rBorder(j); ij < rbj; ij++)
                luMatrix[ij][ii] -= luMatrix[ij][ik] * luMatrix[ik][ii];
        for (int ij = lBorder(j), rbj = rBorder(j); ij < rbj; ij++)
            luMatrix[ij][ii] /= luMatrix[ii][ii];
    }
}
}
}

```

Отметим, что объём необходимых команд системы OpenMP снова невелик, а схема их внедрения в код логична:

```
for (int block = 0; block < block_number; i++)
```

```

{
  // middle block calculation

  //U blocks
  #pragma omp parallel for shared(matrix, i)
  //top blocks calculation

  //L blocks
  #pragma omp parallel for shared(matrix, i)
  //left blocks calculation
}

```

К сожалению, при этом не удаётся избежать того недостатка, что создание потоков привязано к окончанию вычисления каждого из диагональных блоков матрицы системы линейных уравнений. Это влечёт дополнительные накладные расходы на поддержку параллельных процессов (в сравнении с базовым случаем слабо взаимодействующих процессов).

Гипотеза 1. При решении этой системы относительный вклад в общую продолжительность её решения накладных вычислительных расходов на распараллеливание по указанной схеме с ростом числа блоков (M^2) уменьшается.

Действительно, эта общая продолжительность должна была бы быть примерно пропорциональной M^2 , тогда как продолжительность накладных расходов, связанных с созданием потоков, предположительно пропорциональным M .

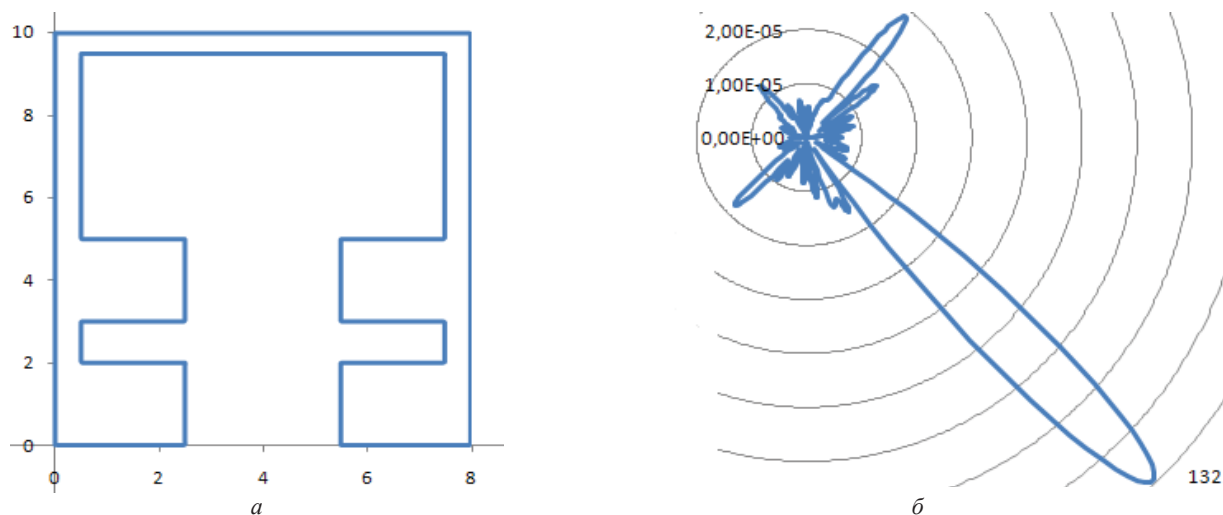


Рис. 2 – Пример, связанный с моделированием дифракции плоской волны на элементе ФАР [9]: a – контур, моделирующий поперечное сечение; b – пример диаграммы направленности ($k = \pi$, $\alpha = \pi/4$).

Вычислительные эксперименты, включая примеры, соответствующие реальным антеннам. Численные эксперименты по продолжительности компьютерного моделирования с использованием разных модификаций МДТ проводились на следующей платформе:

- процессор: Intel core i5-4430,
- ОЗУ: 8GB,
- ОС: Windows 10,
- Компилятор: Mingw 4.9.4 .

Результаты оказались мало чувствительными к конкретным формам контуров, представлявших поперечные сечения цилиндрических идеально проводящих экранов, а также к их взаимному расположению (со сближением не более, чем на длину волны). При нескольких экранах все они в последствии выбирались одинаковыми. В частности использовались контуры, соответствующие форме реальных антенн, волноведущих конструкций или щелевых резонаторов. Пример представлен на рис. 1, 2., а результаты – на рис. 3, 4.

Выводы по результатам тестирования. В табл. 3 представлена типичная серия экспериментов по вычислению плотностей токов и диаграмм направленности для нескольких контуров (проводящих экранов), значение волнового числа $k = \pi$ (от него результаты замеров продолжительности выполнения зависели примерно линейно при не более чем 4-кратном увеличении, тогда как от угла падения не зависели вовсе). Видно, что с усложнением конфигурации дифрагирующей системы (увеличение числа контуров) и уменьшением длины волны (что требовало пропорционального увеличения числа точек дискретизации на каждом контуре) небольшие наклад-

ные расходы, связанные с использованием OpenMP, выросли в 3 раза, достигнув всего 6%, но при этом показатель эффективности – коэффициент ускорения счёта вырос в 7,7 раза. Такие показатели нам представляются с точки зрения вычислителя-прикладника вполне приемлемыми.

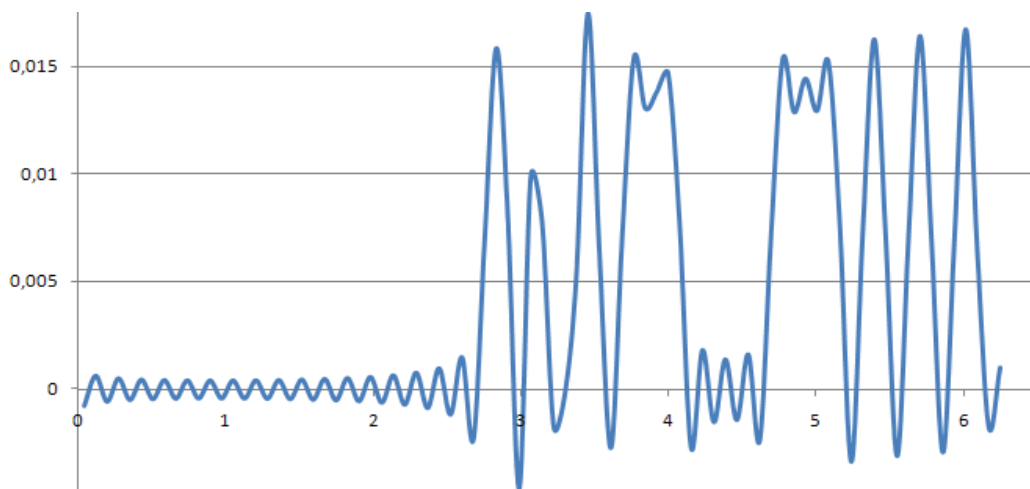


Рис. 3 – Вещественная составляющая плотности тока, порождающего рассеянное поле (вдоль масштаб условный, $k = \pi$, $\alpha = \pi/4$).

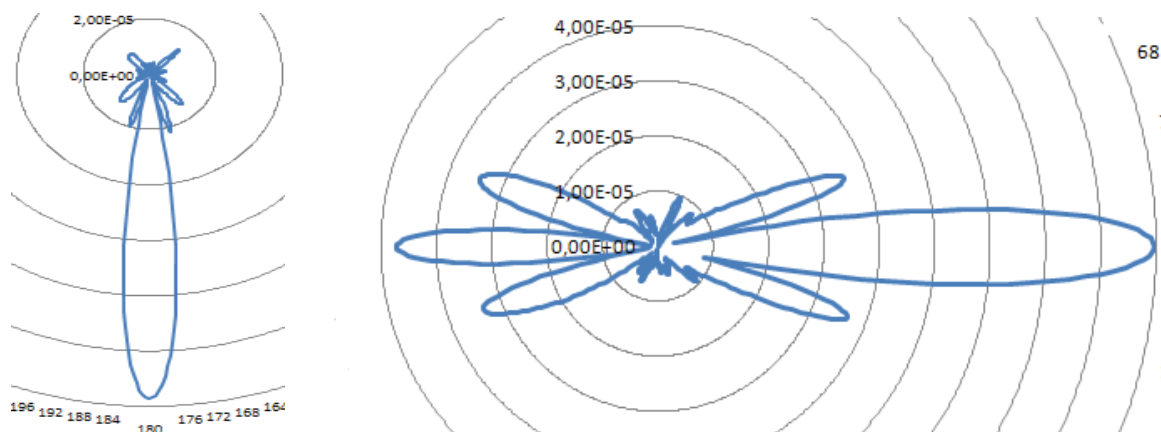


Рис. 4 – Диаграммы направленности в случае других углов падения ($\alpha = \pi/2$, $\alpha = 0$).

Таблица 3 – Время выполнения 3-х реализаций МДТ для 2, 5 и 10 контуров с разным числом точек дискретизации на них

Размерность СЛАУ	Время при исходной реализации	Время, если потоки, тайлинг, векторные регистры	Время: + OpenMP	Общее ускорение	Потеря в скорости за счёт OpenMP
2x100	0.095	0.022	0.022	4.3	2.1%
2x200	0.735	0.156	0.160	4.6	2.5%
5x100	0.647	0.108	0.114	5.7	4.9%
2x500	11.958	1.892	1.667	6.2	2.3%
5x200	6.888	0.625	0.652	10.6	4.4%
10x100	6.008	0.405	0.429	14.0	6.1%
2x1000	96.218	13.470	12.189	7.0	2.1%
10x200	51.898	2.390	0.968	20.506	5.9%
5x500	117.968	7.150	7.441	15.9	4.1%
5x1000	1091.550	53.313	55.494	19.7	4.1%
10x500	1006.100	31.950	33.861	29.7	6.0%
10x1000	8688.100	246.805	261.723	33.2	6.0%

Использование OpenMP вполне удовлетворяет поставленному требованию удобства пользователя, настраивающего программные компоненты библиотеки при её использовании для компьютерного моделирования дифракционных процессов. Потеря в эффективности (которая описывалась коэффициентом ускорения) в среднем около 5% в проведенных экспериментах объясняется неизбежным пересозданием потоков при решении системы линейных уравнений, чего не было при реализации, использовавшей непосредственное написание синхронизирующего кода программистом.

Заключення. Принципиальним результатом проведеного дослідження являється можливість зберегти ефективність комп'ютерного моделювання МДО, яку забезпечують тайлинг і векторні реєстри процесора, відмовляючись від складного (і тем ненадійного при відсутності у користувачів спеціальної кваліфікації) модуля синхронізації за рахунок застосування високоуровневих засобів OpenMP. А саме, в проведених чисельних експериментах у поточної версії бібліотеки реалізації МДТ не спостерігалося втрати продуктивності більш ніж на 6% в порівнянні з методом, який передбачав ручне написання блоку синхронізації паралельних процесів [6,7]. При цьому гіпотеза 1 підтверджується в тому сенсі, що збільшення кількості блоків СЛАУ є наслідком збільшення кількості використовуваних дискретних потоків при збереженні кількості блоків і старих контурів. Може бути висунута і друга гіпотеза.

Гіпотеза 2. При інших рівних умовах накладні витрати на використання OpenMP (по часу виконання) менше при наявності меншої кількості контурів (т. є. металічних екранів).

Також даною роботою завершується основане на аналізі результатів чисельних експериментів дослідження ефективності паралельних реалізацій МДТ на ПК, оскільки для цих цілей використані результати випробування таких реалізацій методами [7] для випадку H поляризації на тестах, які відповідають реальним антенним пристроям зі складною геометрією. Раніше подібні експерименти були проведені і узагальнені для випадку E поляризації для систем розімкнутих і замкнутих контурів [6]. В цьому випадку в МДТ використовуються граничні інтегральні рівняння з ядрами меншої сингулярності, ніж при H поляризації. Коефіцієнт прискорення досягав 40. Даною серією нових експериментів комп'ютерного моделювання підтверджується, як ми бачили, досягнення, як правило, від 10 до 30 кратного прискорення рахунку на 4 ядерних процесорах при використанні не менше 1000 точок дискретизації. Оскільки довільне падаюче поле в 2D постановках можна представити суперпозицією полів з заданими поляризаціями і розсіяне поле буде суперпозицією рішень для цих поляризацій, то і в цьому випадку можна утверджувати багаторазове (до 30 раз) прискорення обчислень для складних дифракуючих систем, які представляють практичний інтерес з точки зору комп'ютерного моделювання дифракційних процесів.

Практичним результатом є побудова базової версії програмної бібліотеки комп'ютерного моделювання SDCM, яка, на відміну від близької по призначенню бібліотеки ЕДЕМ [10], робить застосування МДТ в ОКР можливим не за рахунок евристичного спрощення складних завдань дифракції, а за рахунок форсування внутрішніх резервів обчислювальної потужності сучасних ПК.

Список літератури: 1. Гандель Ю.В., Еременко С.В., Полянська Т.С. Математичні питання методу дискретних потоків. Обґрунтування чисельного методу дискретних особливостей рішення двовимірних завдань дифракції електромагнітних хвиль // Учебне посібник. Ч. 2 – Харків: ХГУ, 1992. – 145 с. 2. Мищенко В.О. К моделюванню електромагнітних явлень на базі використання методів дискретних особливостей для рішення гіперсингулярних інтегральних рівнянь // Труды міжнародної конференції по обчислювальній математиці МКВМ – 2004. Новосибірськ: ІВМіМГ РАН, 2004, – Ч. II. – С. 555 – 560. 3. Gahov A.V., Mishchenko V.O. Parallelism for diffraction processes modeling on the base of discrete singularities methods // Труды Научно-технічної конф. з міжнародним участям «Комп'ютерне моделювання в наукоємких технологіях» (КМНТ – 2010). – Х.: ХНУ, 2010. – Ч. 2. – С. 50 – 53. 4. Мищенко В.О., Паточкін Б.В. Оптимізація компактної схеми Гаусса для багатоядерних процесорів // Вестник ХНУ. – № 981. – Вип. 18. – С. 70 – 81. 5. Parallel Gaussian compact scheme [Електронний ресурс]. – Режим доступу: http://www.mediascan.by/index.files/parallel_gaussian_compscheme_win32.zip. 6. Паточкін Б.В. Мінімізація часу комп'ютерного моделювання задачі дифракції на екранах методом оптимізації локальності пам'яті // Вісник Кременчуцького національного університету імені Михайла Остроградського. – 2014. – Вип. 6 (89). – С. 58 – 64. 7. Мищенко В.О., Паточкін Б.В. Модифікація методу дискретних потоків для використання векторних реєстрів процесору ПК при чисельному моделюванні дифракції на екранах // Матеріали конференції «Сучасні проблеми моделювання та обчислювальних методів». – Рівне, 2015. – С. 130. 8. Воеводин В.В., Воеводин Вл.В. Паралельні обчислення. – СПб.: БХВ-Петербург, 2002. – 608 с. 9. Фазированная антенная решётка [Електронний ресурс]. – Режим доступу: <http://dic.academic.ru/dic.nsf/bse/143796> Фазированная. 10. Davydov A., Zakharov E. Opportunities of program EDEM for development of devices of antenna techniques // Antennas. – 2006. - № 10. - P. 52 – 57.

Bibliography (transliterated): 1. Gandel', Yu. V., S. V. Eremenko and T. S. Poljanskaja. *Matematicheskie voprosy metoda diskretnykh tokov. Obosnovanie chislennogo metoda diskretnykh osobennostej reshenija dvumernykh zadach difrakcii elektromagnitnykh voln. Uchebnoe posobie. Vol. 2.* – Kharkov: KhGU, 1992. Print. 2. Mishchenko, V. O. "K modelirovaniyu elektromagnitnykh javlenij na baze ispol'zovanija metodov diskretnykh osobennostej dlja reshenija gipersinguljarnykh integral'nykh uravnenij." *Trudy mezhdunarodnoj konferencii po vychislitel'noj matematike MKVM-2004.* Novosibirsk: IVMiMG RAN, 2004, No. 2. 555–560. Print. 3. Gahov, A. V., and V. O. Mishchenko. "Parallelism for diffraction processes modeling on the base of discrete singularities methods." *Trudy Nauchno-technicheskoiy konf. s mezhdunarodnym uchastiem "Kompjuterne modelirovanie v naukojomykh tehnologijah."* Vol. 2. Kharkov: KhNU, 2010. 50–53. Print. 4. Mishchenko, V. O., and B. V. Patochkin. "Optimizacija kompaktnoj shemy Gaussa dlja mnogojadernykh processorov." *Vestnik KhNU.* No. 981. Vol. 18. 70–81. Print. 5. "Parallel Gaussian compact scheme." *Mediascan.* Web. 14 October 2015. <http://www.mediascan.by/index.files/parallel_gaussian_compscheme_win32.zip>. 6. Patochkin, B. V. "Minimizacija vremeni kompjuterного modelirovanija zadachi difrakcii na ekranah metodom optimizacii lokal'nosti pamjati." *Visnyk Kremenčuc'kogo nacional'nogo univertsytetu imeni Myhajla Ostrograds'kogo.* No. 6 (89). 2014. 58–64. Print. 7. Mishchenko, V. O., and B. V. Patochkin. "Modyfikacija metodu dyskretnykh tokiv dlja vykorystannja vektornykh registriv processoru PK pry chyslovomu modeljuvanni dyfrakcii na ekranah." *Materialy konferencii "Suchasni problemy modeljuvanija ta obchysljuval'nyh metodiv"*. Rivne, 2015. 130. Print. 8. Voevodin, V. V., and Vl. V. Voevodin. *Parallel'nye vychislenija.* Sankt-Petersburg: BHV–Peterburg, 2002. Print. 9. *Bol'shaja sovetskaja jenciklopedija.* "Fazirovannaja antennaja reshjotka." *Akademik.* Web. 14 September 2015. <<http://dic.academic.ru/dic.nsf/bse/143796> Fazirovannaja>. 10. Davydov, A., and E. Zakharov. "Opportunities of program EDEM for development of devices of antenna techniques." *Antennas.* No. 10. 2006. 52–57. Print.

Поступила (received) 30.09.2015

Мищенко Віктор Олегович – доктор технічних наук, доцент, Харківський національний університет імені В. Н. Каразіна, м. Харків; тел.: (057) 705-42-81; e-mail: mischenko@univer.kharkov.ua.

Мищенко Виктор Олегович – доктор технических наук, доцент, Харьковский национальный университет имени В. Н. Каразина, г. Харьков; тел.: (057) 705-42-81; e-mail: mischenko@univer.kharkov.ua.

Mishchenko Viktor Olegovich – Doctor of Technical Sciences, Associate professor, V. N. Karazin Kharkov National University, Kharkov; tel.: (057) 705-42-81; e-mail: mischenko@univer.kharkov.ua.

Паточкін Борис Вікторович – аспірант, Харківський національний університет імені В. Н. Каразіна, м. Харків; тел.: (057) 705-42-81; тел.: (050) 999-79-65; e-mail: equilibrium2702@gmail.com.

Паточкин Борис Викторович – аспирант, Харьковский национальный университет имени В. Н. Каразина, г. Харьков; тел.: (057) 705-42-81; тел.: (050) 999-79-65; e-mail: equilibrium2702@gmail.com.

Patochkin Boris Viktorovich – postgraduate, V. N. Karazin Kharkov National University, Kharkov; tel.: (057) 705-42-81; tel.: (050) 999-79-65; e-mail: equilibrium2702@gmail.com.

УДК 517.955.8

Е. А. НАБОКА

СИНХРОНИЗАЦИЯ КОЛЕБАНИЙ ДВУХ СВЯЗАННЫХ ПЛАСТИН БЕРГЕРА С НЕЛИНЕЙНЫМ ВНУТРЕННИМ И ГРАНИЧНЫМ ДЕМПФИРОВАНИЕМ. ЧАСТЬ 2

Рассматривается модель Бергера нелинейных колебаний двух одинаковых упруго связанных пластин с частично зашпеленной и частично свободной границей. Предполагается, что нелинейные диссипационные силы действуют во внутренней части пластин и на свободной части их границ. Изучена зависимость структуры глобального аттрактора системы от параметра γ , пропорционального интенсивности взаимодействия пластин. В этой части работы описана структура верхнего предела аттрактора при $\gamma \rightarrow \infty$. Установлено, что он совпадает с множеством $\text{diag}(\tilde{\mathcal{A}} \times \tilde{\mathcal{A}})$ – диагональю прямого произведения двух экземпляров аттрактора системы, описывающей колебания одной пластины. Также при дополнительных условиях на функции демпфирования доказано, что и сам аттрактор системы двух связанных пластин совпадает с множеством $\text{diag}(\tilde{\mathcal{A}} \times \tilde{\mathcal{A}})$ для достаточно больших значений параметра γ .

Ключевые слова: модель Бергера, упруго связанные пластины, асимптотическая синхронизация, нелинейная диссипация, свободная граница.

Обзор результатов части 1. Явления синхронизации различных составных механических, химических, биологических, социальных систем вызывают большой интерес в литературе (смотрите [1 – 13]).

В данной серии работ изучается синхронизация механической системы, состоящей из двух упруго связанных пластин с частично зашпеленной и частично свободной границей и нелинейными диссипационными силами, действующими как во внутренних частях пластин, так и на свободных участках их границ. Соответствующая математическая модель, известная в литературе как *модель Бергера* нелинейных колебаний пластины [14], имеет следующий вид:

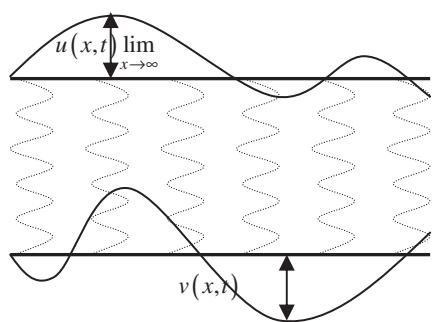


Рис. 1 – Система связанных пластин:
 $u(x,t), v(x,t)$ – вертикальные отклонения пластин относительно состояния покоя.

$$u = u(x,t), v = v(x,t), \quad x \in \Omega \subset \mathbb{R}^2, t \geq 0;$$

$$\begin{cases} u_{tt} + B(u_t) + \Delta^2 u + \left(Q - \beta \int_{\Omega} |\nabla u|^2 d\Omega \right) \Delta u + \gamma(u-v) = p, \\ v_{tt} + B(v_t) + \Delta^2 v + \left(Q - \beta \int_{\Omega} |\nabla v|^2 d\Omega \right) \Delta v + \gamma(v-u) = p, \end{cases} \quad x \in \Omega, t > 0; \quad (1)$$

$$u|_{t=0} = u^0, \quad v|_{t=0} = v^0, \quad u_t|_{t=0} = u^1, \quad v_t|_{t=0} = v^1; \quad (2)$$

$$u = \partial_n u = v = \partial_n v = 0, \quad x \in \Gamma_0; \quad (3)$$

$$\Delta u + (1-\mu)\Sigma_1 u = 0, \quad \Delta v + (1-\mu)\Sigma_1 v = 0, \quad x \in \Gamma_1; \quad (4)$$

$$\begin{aligned} \partial_n \Delta u + (1-\mu)\Sigma_2 u &= d(x)g(u_t) + \sigma u - \left(Q - \beta \int_{\Omega} |\nabla u|^2 d\Omega \right) \partial_n u, \\ \partial_n \Delta v + (1-\mu)\Sigma_2 v &= d(x)g(v_t) + \sigma v - \left(Q - \beta \int_{\Omega} |\nabla v|^2 d\Omega \right) \partial_n v, \end{aligned} \quad x \in \Gamma_1; \quad (5)$$

где Ω – плоская область, занимаемая пластинами в состоянии покоя; граница области $\partial\Omega = \Gamma_0 \cup \Gamma_1$; $\Gamma_0 \cap \Gamma_1 = \emptyset$, Γ_0 – зашпеленная часть границы, Γ_1 – ее свободная часть; функции $u(x,t), v(x,t)$ описывают