

Методы ускорения процесса сборки изделий из тканых материалов

И.Е. Ландовская, В.Д. Фроловский, В.В. Ландовский
ФГБОУ ВПО НГТУ (Новосибирск, Россия)

Аннотация: Статья посвящена применению методов параллельной обработки данных в задаче моделирования процесса сборки изделий из тканых материалов на поверхности твердотельного многогранного объекта с целью ускорения процесса моделирования. Представлены алгоритмы для распараллеливания вычислений, как на центральном процессоре, так и на графическом процессоре видеоадаптера.

Ключевые слова: методы параллельной обработки данных, моделирование тканых материалов, метод частиц, схема с перешагиванием.

ВВЕДЕНИЕ

Моделирование трехмерных поверхностей и изучение их поведения используется не только в науке, медицине, телевидении и дизайне, а также и при моделировании ткани. Мощности современных компьютеров и программные средства для них позволяют человеку создавать и моделировать практически любые интересные его процессы.

Задача моделирования поведения тканых материалов является одной из наиболее увлекательных и сложных задач компьютерной графики. Ответы на вопросы о том, как будет выглядеть ткань с определенными свойствами на определенном объекте, как с изменением свойств ткани изменяется ее драпировка, и другие вопросы в наглядной форме должно давать компьютерное моделирование ткани. Важным в этой задаче является не только достижение наибольшей визуальной реалистичности, но, возможно в большей степени, обеспечение соответствия модели физическим характеристикам ткани, соответствия моделируемых деформаций реальным.

Успешно развивающимся на сегодняшний день подходом является моделирование ткани с использованием метода частиц [1]. Модели частиц применяются для моделирования самых разных явлений таких, как водопады, взрывы, вихревые поля и т. д. Термин «модели частиц» является общим для класса вычислительных моделей, в которых дискретное описание физических явлений включает использование взаимодействующих частиц. Каждая частица имеет набор атрибутов, таких как масса, положение, скорость. Состояние физической системы определяется атрибутами конечного ансамбля частиц, а эволюция системы

определяется законами взаимодействия этих частиц.

Очевидно, чем большим количеством частиц представлена модель, тем точнее будут переданы ее деформационные свойства. В то же время, при увеличении количества частиц в модели, происходит увеличение времени моделирования, которое возрастает пропорционально увеличению количества частиц, которыми представлена сетка тканого материала. Сокращение времени расчетов можно добиться, воспользовавшись методами параллельной обработки данных, которые могут повысить скорость получения результатов за счет использования вычислительных ресурсов нескольких ядер или процессоров одновременно.

1. ДИСКРЕТНАЯ МОДЕЛЬ ТКАНОГО МАТЕРИАЛА

В полотне тканого материала, тонкие волокна скручены в нити, а эти нити более или менее жестко сплетены во взаимосвязанную сеть. Компоненты сети удерживаются вместе трением, а поведение ткани зависит от типа волокна (хлопчатобумажное, шелк, шерсть и т. д.), веса волокна, плотности сплетения, типа сплетения и т. д. Порядок расположения нитей относительно друг друга остается неизменным даже при существенных деформациях, а сама деформация ткани представляет собой только изменение формы и размеров ячеек [2]. Отсюда следует предположение рассматривать ткань как систему частиц, которые размещены в точках пересечения продольных и поперечных нитей, как представлено на *Рис. 1*.

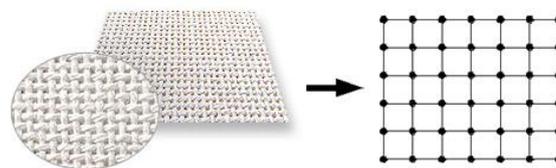


Рис. 1. Дискретная модель ткани

Основные взаимодействия, которые происходят на уровне нити, это: растяжение-сжатие; изгиб и сдвиг [3]. Учитывая, что ткань достаточно легкая и масса ткани в удаленных узлах оказывает пренебрежимо малое влияние на каждую рассматриваемую частицу, предположим, что на каждую внутреннюю частицу влияют 12 соседних частиц. На *Рис. 2* взаимодействия растяжения-сжатия, сдвига и изгиба обозначены связями 1, 2 и 3 соответственно.

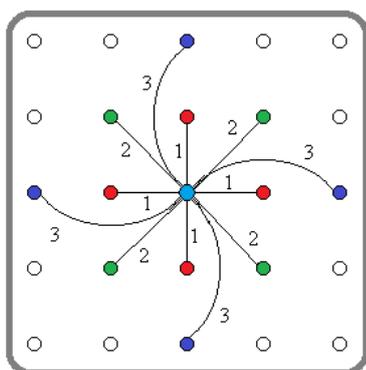


Рис. 2. Взаимодействия между частицами ткани

2. МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ТКАНИ

Движение системы можно описать перемещениями в трехмерном пространстве:

$$r_{ij}(t) = \{x_{ij}(t), y_{ij}(t), z_{ij}(t)\},$$

где $x_{ij}(t), y_{ij}(t), z_{ij}(t)$ – координаты частицы в трехмерном пространстве, t – время.

На каждом временном слое ищут положения узлов в пространстве. При этом каждая из частиц обладает некоторой массой, находится в гравитационном поле, взаимодействует с окружающей средой и соседними частицами [4]. Тогда уравнение движения частицы P_{ij} имеет следующий вид:

$$m_{ij}r_{ij}'' = m_{ij}g - m_{ij}c_{ij}r_{ij}' + \sum_{kl \in R_{ijkl}} F_{im}(r_{ij}, r_{kl}),$$

где m_{ij} – масса частицы; c_{ij} – константа демпфирования; составляющая $-m_{ij}c_{ij}r_{ij}'$ – представляет собой потери энергии, связанные с взаимодействием частицы с окружающей средой; g – ускорение свободного падения. Последняя составляющая уравнения движения, представляет собой результирующую силу взаимодействий между частицами; R_{ijkl} – множество индексов узлов связанных с узлом P_{ij} .

Для решения системы следует воспользоваться схемой с перешагиванием (leapfrog scheme) [1]. Это обусловлено тем, что для воспроизведения на компьютере реального поведения материала, число частиц в моделируемом полотне должно быть велико, и, к тому же, каждое дополнительное вычисление силы требует больших временных затрат.

Схема с перешагиванием является методом второго порядка точности, и, в сравнении с классическим явным методом Эйлера, обладает большей устойчивостью, что позволяет на порядок увеличить шаг интегрирования:

$$\begin{aligned} V_{n+1} &= V_n + hM^{-1}F(r_n, V_n), \\ r_{n+1} &= r_n + hV_{n+1} \end{aligned}$$

где r_n и V_n – векторы положений и скоростей частиц на n -ом шаге интегрирования, h – шаг интегрирования, $F(r, V)$ – вектор-функция, описывающая действие внутренних и внешних

сил на ткань, M – матрица инерции – диагональная матрица, описывающая распределение масс частиц ткани.

3. АЛГОРИТМ ПАРАЛЛЕЛЬНОЙ ОБРАБОТКИ ДАННЫХ НА ЦЕНТРАЛЬНОМ ПРОЦЕССОРЕ

При моделировании твердотельный многогранный объект, на поверхности которого происходит сборка, является неподвижным, а изменение координат происходит только у частиц ткани за счет действия на них различных сил, имитирующих воздействие окружающей среды, таких как стягивающая сила, сила тяжести, сопротивление воздуха, сила трения об объект и прочие.

Как было описано ранее, ткань рассматривается как система частиц, которые размещены в точках пересечения продольных и поперечных нитей. Следовательно, в программе систему частиц, соответствующую детали, удобнее всего представить в виде двумерного динамического массива. Число строк массива соответствует максимальному количеству частиц по длине детали, а число столбцов – максимальному количеству частиц детали по ее ширине. Если элементу массива не соответствует частица детали, то он имеет значение *NULL*. При этом определить взаимодействующие частицы можно исходя из позиции элемента в массиве. Схематично двумерный массив для хранения данных о частицах детали представлен на Рис. 3.

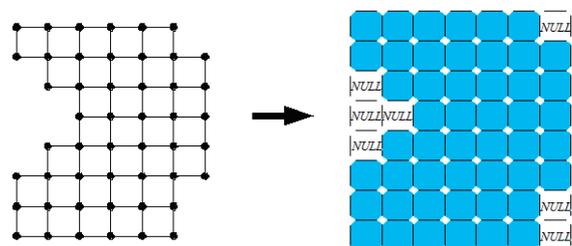


Рис. 3. Структура для хранения данных о частицах детали

На сегодняшний день наиболее целесообразным решением является разработка алгоритма параллельной обработки данных для осуществления расчетов на центральном процессоре, который подходит как для систем с общей памятью, так и систем с распределенной.

Как описано выше, в процессе моделирования для решения уравнений движения частиц ткани применяется явный метод, а именно схема с перешагиванием. Следовательно, на каждом шаге интегрирования вначале вычисляется новое значение скорости, которое затем используется для вычисления значений координат на этом шаге. Этой особенностью можно воспользоваться, при создании алгоритма параллельной обработки данных для систем, как с общей, так и с распределенной памятью.

В то же время, как сказано в разделе 1, для

вычисления скорости частицы на n -ом шаге интегрирования, нужно учитывать значения координат ее соседних частиц с $(n-1)$ -ого шага интегрирования. То есть, в алгоритме параллельной обработки данных для систем с распределенной памятью, нужно предусмотреть этап своевременного обмена данными между процессами.

При работе в системах с распределенной памятью передача данных между процессами всего массива координат частиц деталей, число которых составляет несколько тысяч, занимает больший процент от общего времени, чем сами вычисления и, следовательно, аннулирует все преимущества параллелизма.

Для уменьшения потерь времени на передачу данных между процессами с учетом особенностей задачи моделирования разработан усовершенствованный алгоритм, который позволяет обмениваться процессам только четырьмя строками массива, что значительно повышает его эффективность. Этот алгоритм учитывает, что для вычисления значений координат точки ткани используются значения координат только двенадцати ее соседних частиц.

При моделировании двумерный массив, в котором хранятся данные о частицах детали, разделяется на количество частей, равное количеству запущенных процессов в программе, как представлено на *Рис. 4* [5].

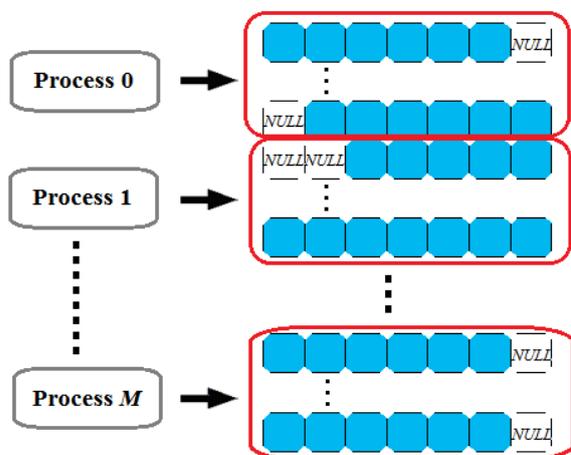


Рис. 4. Разделение данных между процессами

Каждый процесс обрабатывает только свою часть массива, а затем происходит обмен данными, при котором процессы обмениваются между собой значениями координат двух крайних строк своей части массива, *Рис. 5*.

Кроме того, в памяти каждого процесса создается отдельная копия данных массива частиц многогранного объекта, так что при моделировании процесс обрабатывает пересечения только своей части массива частиц детали с гранями объекта, что позволяет избежать лишнего обмена данными между процессами и приводит к резкому сокращению времени вычислений.

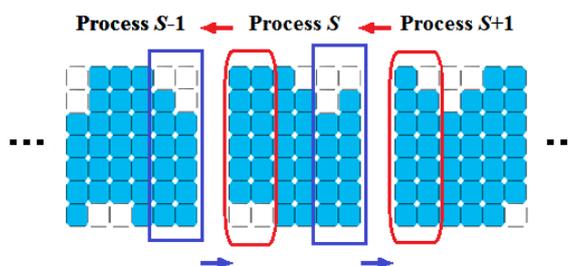


Рис. 5. Обмен данными между процессами

Параллелизм программы достигается как на уровне потоков, за счет применения библиотеки *OpenMP*, так и на уровне процессов – для этого применяется библиотека *MPICH2*. В каждом процессе порождается заданное количество потоков, что полностью соответствует *SPMD*-модели, представленной на *Рис. 6*. Данный алгоритм может быть использован в системах как с общей, так с распределенной памятью, что делает его более универсальным.

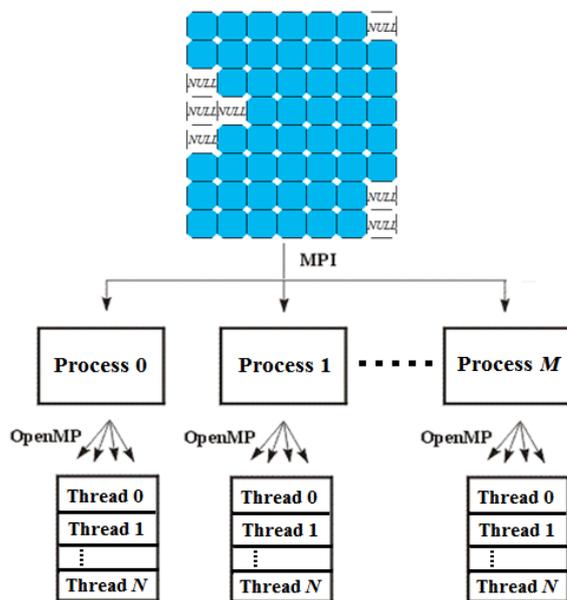


Рис. 6. *SPMD*-модель работы программы

Представленный алгоритм протестирован в восьмиядерной многопроцессорной среде, на кафедре вычислительной техники Новосибирского государственного технического университета с использованием вычислительного кластера с процессорами *Intel Xeon E5310 Clovertown* с частотой 1600 МГц и общей памятью 32 Гб . Тестирование проводилось с использованием различных контуров деталей и различных типов объектов. Результаты эксперимента по вычислению зависимости времени выполнения 10000 итераций моделирования движения ткани из 987 частиц при взаимодействии с объектом из 300 граней от количества запущенных процессов и потоков представлены в *Таблице 1*.

Зависимость времени моделирования от количества запущенных процессов и потоков

Количество процессов	1	1	1	1	2	2	4	4	8
Количество потоков	1	2	4	8	1	4	1	2	1
Время моделирования, с	71,8	42,5	25,7	16,1	43,4	16,9	28,0	18,2	20,8

В качестве критерия оптимальности выступает время получения результатов моделирования. Этот критерий выбран, так как он является одним из основных для данной задачи.

Из Таблицы 1 видно, что наименьшее время моделирования параллельный алгоритм показывает при работе в системах с общей памятью, так как отсутствует расход времени на обмен данными между процессами. Следовательно, для достижения максимальной эффективности от применения параллельного алгоритма необходимо, по-возможности, создавать максимальное количество потоков, а количество процессов сводить к минимуму.

4. АЛГОРИТМ ПАРАЛЛЕЛЬНОЙ ОБРАБОТКИ ДАННЫХ НА ГРАФИЧЕСКОМ ПРОЦЕССОРЕ ВИДЕОКАРТЫ

Структура графического процессора такова, что позволяет запускать на выполнение практически одновременно большое количество потоков (нитей). Как отмечено выше, при моделировании поведения ткани движение частиц материала не является абсолютно независимым. В то же время, так как для вычисления скорости частицы на n -ом шаге интегрирования, требуется учитывать значения координат ее соседних частиц с $(n-1)$ -ого шага интегрирования, такая задача подходит для распараллеливания данных на графическом процессоре. Для повышения производительности следует представить данные, передаваемые в память графического процессора не массивами структур, а классами, содержащими несколько массивов:

```
class Particles_fabric {
public:
    float *coord_x;
    float *coord_y;
    float *coord_z;
    ...
    float *V_x;
    float *V_y;
    float *V_z;
    ...
    float *F_x;
    float *F_y;
    float *F_z;
};
```

В программе используются двумерные массивы, которые хранятся в памяти в виде одной связанной области. Адрес элемента массива с

индексами i, j равен $i \cdot N_y + j$, где N_y – количество столбцов в массиве. Это означает, что элементы, имеющие близкие индексы по оси y , расположены в соседних ячейках памяти [6]. Таким образом, если различные нити обрабатывают различные ячейки по оси y , они будут запрашивать данные из соседних элементов массивов.

Как и в алгоритме для выполнения на центральном процессоре, между блоками происходит обмен через глобальную память не всеми данными, а только той частью массива, которая содержит информацию о координатах частиц, являющихся соседними для частиц других блоков.

Зависимость времени выполнения 10000 итераций моделирования движения ткани из 4056 частиц от размера блока отражены в Таблице 2.

Тестирование представленного алгоритма проводилось на видеокарте *NVIDIA GeForce GTX 760*. Следует заметить, что при хронометраже не учитывалось время копирования данных между центральным и графическим процессорами, а максимально возможный размер блока для данного графического процессора составляет 32, так как $32 \times 32 = 1024$ – предельное значение для произведения измерений блока.

Таблица 2

Зависимость времени моделирования от размера блока

Размер блока	Время выполнения, с
не блочный	16,51
4×4	91,1
8×8	16,8
16×16	11,2
32×32	10,3

Стоит отметить, что время работы ядерной функции не должно превышать двух секунд из-за механизма *Time-Out Detection and Recovery (TDR)*, используемого в *Microsoft Windows*.

На Рис. 7 приведены графики для сравнения производительности центрального процессора *Intel Core i7-4770K Haswell* (запуск задачи в 8 потоков) и графического процессора видеокарты *NVIDIA GeForce GTX 760* (размер блока 32×32) в задаче моделирования поведения ткани в зависимости от количества частиц модели при вычислении 10000 итераций.

В процессе моделирования учитываются коэффициенты среды, такие как сила тяжести, сопротивление воздуха, сила трения об объект и

прочие, что придает процессу большую реалистичность. Итоговый результат моделирования падения хлопчатобумажной лоскутной ткани на кресло представлен на *Рис. 8*.

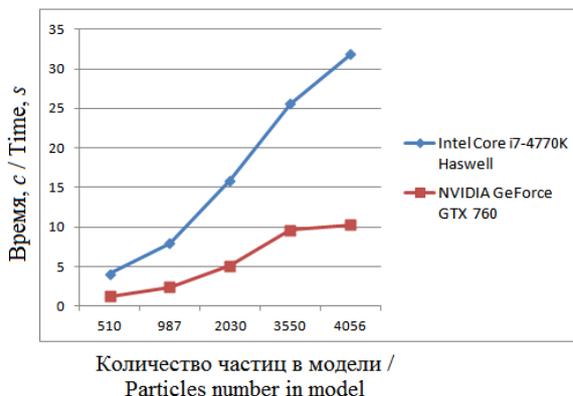


Рис. 7. Графики производительности центрального и графического процессоров в задаче моделирования ткани

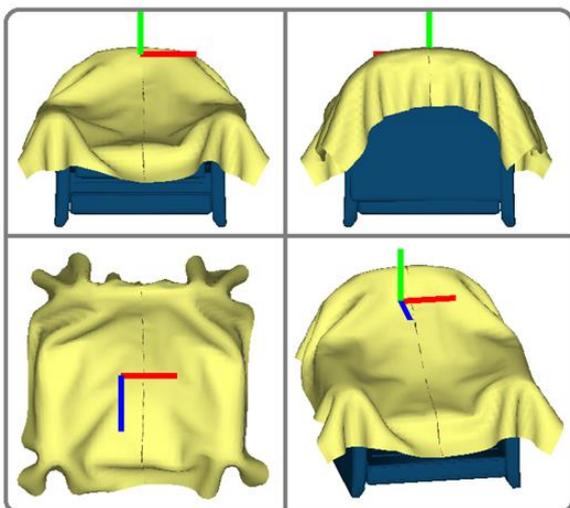


Рис. 8. Результаты моделирования

ЗАКЛЮЧЕНИЕ

В статье описаны методы параллельной обработки данных с использованием центрального и графического процессоров. Данные алгоритмы были протестированы на примерах разных видов деталей и форм объектов.

Для тестирования был разработан комплекс программных средств на языке C++ в среде *Microsoft Visual Studio 2012*. Для реализации параллельной обработки данных на центральном процессоре используются интерфейсы параллельного программирования *OpenMP* и *MPI*, а для обработки данных с использованием ресурсов графического процессора видеоадаптера применяется технология *CUDA*.

Результаты, представленные в работе, имеют значение для развития информационных технологий в области геометрического моделирования и проектирования. Научной

новизной в статье являются предложенные алгоритмы параллельной обработки данных в процессе моделирования для описанной дискретной модели ткани.

Результаты данной работы могут быть использованы для создания систем автоматизации геометрического моделирования и проектирования, систем виртуальной реальности и компьютерной помощи для специалистов швейной промышленности, а также в работе дизайнеров одежды и мебельного производства.

ЛИТЕРАТУРА

- [1] Hockney W., Eastwood J., Computer Simulation Using Particles. McGraw-Hill, New York, 1981.
- [2] Breen D., House D. A physically based model of woven cloth. The Visual Computer, 8(5-6):264–277, June 1992.
- [3] Kawabata S. The standardisation and analysis of hand evaluation. Technical report. The textile machinery society of Japan, Osaka, July 1980.
- [4] Ландовский В.В., Фроловский В.Д. Исследование методов интегрирования дифференциальных уравнений в задаче моделирования поведения ткани на основе метода частиц. Сибирский журнал вычислительной математики. Том 9, 2006. С. 287–298.
- [5] Хьюз К., Хьюз Т. Параллельное и распределенное программирование на C++: Пер. с англ. – М.: Издательский дом «Вильямс», 2004. – 672 с.
- [6] Сандерс Дж., Кэндрот Э. Технология CUDA в примерах: введение в программирование графических процессоров: Пер. с англ. – М.: ДМК Пресс, 2011. – 232 с.



Ландовская Ирина Евгеньевна, аспирант кафедры автоматизированных систем управления НГТУ. Область научных интересов: компьютерное моделирование поведения тканых материалов.
E-mail: nairy@rambler.ru



Фроловский Владимир Дмитриевич, профессор кафедры автоматизированных систем управления НГТУ, профессор, доктор технических наук. Область научных интересов: моделирование и автоматизация процессов геометрического проектирования.
E-mail: frolovskij@corp.nstu.ru



Ландовский Владимир Владимирович, доцент кафедры автоматизированных систем управления НГТУ, кандидат технических наук. Область научных интересов: моделирование процессов сборки трехмерных изделий из плоских заготовок.
E-mail: landovskij@corp.nstu.ru

Accelerating methods of the fabric products assembly

I.E. LANDOVSKAYA, V.D. FROLOVSKY,
V.V. LANDOVSKY

Abstract: Using of the parallel processing methods during the simulation of the fabric material products on the solid plane–bounded object’s surface to accelerate the simulation process is considered in the paper. The algorithms for paralleling the calculations on the central processing unit and on the graphics processor of the video display adapter are presented.

Keywords: the parallel processing methods, simulation of the fabric materials, particle method, leapfrog scheme.

REFERENCES

- [1] Hockney W., Eastwood J., Computer Simulation Using Particles. McGraw-Hill, New York, 1981.
- [2] Breen D., House D. A physically based model of woven cloth. The Visual Computer, 8(5-6):264–277, June 1992.
- [3] Kawabata S. The standardisation and analysis of hand evaluation. Technical report, The textile machinery society of Japan, Osaka, July 1980.
- [4] Landovsky V.V., Frolovsky V.D. Issledovanie metodov integrirovaniya differencial'nyh uravnenij v zadache modelirovaniya povedeniya tkani na osnove metoda chastic [Investigation of the techniques of the difference equation integration in the problem of the simulation of the fabric materials behavior by particle method]. Sibirskij zhurnal vychislitel'noj matematiki. - Siberian Journal of Numerical Mathematics. 2006. vol. 9. pp. 287-298.
- [5] Hughes C., Hughes T. Parallelnoe i raspredelennoe programmirovaniye na C++: Per. s angl. [Parallel and distributed programming using C++] – M.: Izdatel'skij dom «Vilyams», 2004. – 672 p.
- [6] Sanders J., Kandrot E. Tekhnologiya CUDA v primerakh: vvedeniye v programmirovaniye graficheskikh protsessorov: Per. s angl. [CUDA by example. An introduction to general-purpose GPU programming] – M.: DMK Press, 2011. – 232 p.