В.В. Гетманский, А.С. Горобцов

Адаптация алгоритмов моделирования систем тел со связями для высокопроизводительного кластера

АННОТАЦИЯ. Рассмотрены параллельные алгоритмы для решения задач методом моделирования динамики систем тел со связями. Приведены результаты по ускорению и эффективности расчетов при решении задач моделирования транспортных средств. Проведено сравнение многопоточных и кластерных реализаций алгоритма, а также реализаций с использованием векторизации вычислений. Показаны ограничения по ускорению алгоритмов решения связанных задач.

Kлючевые cлова u фразы: динамика систем тел, связанные задачи, вычислительный кластер, векторизация, многопоточный алгоритм, масштабируемость.

1. Метод моделирования

Для метода моделирования динамики систем тел со связями характерна сильная зависимость по данным, поэтому попытки распараллеливания алгоритмов моделирования наталкиваются на ограничения масштабируемости [1]. Как правило, одну и ту же задачу можно распараллелить различными способами. В настоящей работе предложен анализ комбинаций методов распараллеливания и оптимизации алгоритмов для их выполнения на вычислительном кластере.

В разрабатываемом программном комплексе ФРУНД [2,5] имеется возможность решения связанных задач. При этом имеют место основная модель динамики машиностроительного объекта и вспомогательные модели для расчета теплопередачи и напряженно-деформированного состояния в телах. Моделирование динамики систем тел проводится с помощью численного интегрирования системы дифференциально-алгебраических уравнений в расширенной постановке Лагранжа первого рода [3]:

- © В.В. ГЕТМАНСКИЙ, А.С. ГОРОБЦОВ, 2015
- © ВОЛГОГРАДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ, 2015
- (С) ПРОГРАММНЫЕ СИСТЕМЫ: ТЕОРИЯ И ПРИЛОЖЕНИЯ, 2015

$$\begin{cases} \mathbf{M}\ddot{\mathbf{x}} - \mathbf{D}_{x}^{\mathsf{T}} \alpha (2\mu \mathbf{\Omega}^{2} \dot{\mathbf{\Phi}} + \mathbf{\Omega}^{2} \mathbf{\Phi}) + \mathbf{D}_{x}^{\mathsf{T}} \mathbf{p} = \mathbf{q}(\dot{\mathbf{x}}, \mathbf{x}, t), \\ \mathbf{D}_{x} \ddot{\mathbf{x}} = \mathbf{h}(\dot{\mathbf{x}}, \mathbf{x}). \end{cases}$$
(1)

где ${\bf x}$ -координаты тел, t-время, ${\bf M}$ - матрица инерции, ${\bf D}_{{\bf x}}$ - матрица частных производных уравнений связей, ${\bf \alpha}$ -диагональная матрица коэффициентов, ${\bf \mu}$ - коэффициенты демпфирования, ${\bf \Omega}$ -диагональная матрица собственных частот, ${\bf q}$ и ${\bf h}$ - правые части.

Расчетные области для вспомогательных моделей отдельных деталей конструкции строятся на основе твердотельной геометрии. При этом происходит ее дискретизация и построение модели из дискретных элементов [3]. Понижение порядка системы приводит к описанию всей совокупности моделей системой дифференциально-алгебраических уравнений (2).

$$\begin{cases} \mathbf{M}\dot{\mathbf{v}} + \mathbf{D}_{x}^{\mathrm{T}}\boldsymbol{\lambda} = \mathbf{f}(\mathbf{v}, \mathbf{x}, t), \\ \mathbf{D}_{x}\dot{\mathbf{v}} = \mathbf{h}(\mathbf{v}, \mathbf{x}), \\ \mathbf{M}^{*}\dot{\mathbf{r}} = \mathbf{q}(\mathbf{r}, \mathbf{y}, \mathbf{t}) + \mathbf{f}^{*}(\mathbf{t}), \\ \dot{\mathbf{u}} = \gamma\Delta(\mathbf{u}) + \mathbf{q}'(\mathbf{x}, t), \\ \dot{\mathbf{x}} = \mathbf{v}, \\ \dot{\mathbf{y}} = \mathbf{r} \end{cases}$$
(2)

Поскольку матрицы инерции ${\bf M}$ и ${\bf M}^*$ диагональные, система разрешима относительно первых производных обращением матрицы ${\bf D}_{\rm x}$.

Первые два уравнения относятся к основной модели, вторые два – к вспомогательным моделям и последние два введены для понижения порядка системы. При разрешении уравнения относительно первых производных получается система обыкновенных ДУ, которая решается явным численным методов интегрирования Рунге-Кутта 4-го порядка, который можно распараллелить, обеспечив синхронизацию зависимых по правым частям уравнений при каждом вычислении правых частей [4], что имеет место 4 раза за одну итерацию.

Вспомогательные модели, имеющие зависимости от основной модели, в рамках одной итерации можно также рассчитывать параллельно, при этом требуется синхронизация на каждой итерации с пересылкой существенно меньшего количества данных, по сравнению с методом распараллеливания системы ОДУ.

Вычислительная сложность расчетного модуля для решения системы (1) зависит от размерности задачи нелинейно, но она существенно ниже, чем вычислительная сложность для вспомогательных расчетных модулей, так как количество неизвестных в системе уравнений (2) на несколько порядков больше, чем в системе (1).

На каждой расчетной итерации параметры между расчетными модулями обмениваются с помощью пересылки сообщений с помошью библиотеки MPI.

Параллельный расчет на кластере выполняется по следующему алгоритму:

- 1.Выполнение итерации для решения системы (1) в момент времени t;
 - 2. Вычисление зависимых параметров;
- 3.Пересылка средствами библиотеки MPI зависимых параметров во вспомогательные расчетные модули;
- 4.Выполнение итерации для решения уравнений системы (2) параллельно;
 - 5. Барьерная синхронизация средствами библиотеки МРІ;
 - 6. Переход к шагу 1.

2. Исследование параллельных алгоритмов

Анализ алгоритмов показал, что распараллеливание в рамках одной итерации на шаге 4, например, распараллеливание циклов с помощью OpenMP, является неэффективным, из-за накладных расходов на создание потоков и синхронизацию, которые занимают существенное время по сравнению с временем расчета одной итерации.

Таким образом, для практического применения, использовано только распараллеливание на уровне структуры модели. Возможны

два варианта: использование МРІ процессов и использование смешанной парадигмы: параллельных процессов и многопоточной реализации алгоритма в рамках одного процесса.

Из-за неэффективности OpenMP для рассматриваемой задачи был реализован многопоточный алгоритм [4] на POSIX-потоках, который в 2 раза быстрее, чем версия с OpenMP (рис. 1).

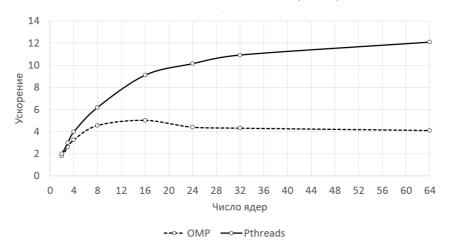


Рис. 1. Сравнение ускорения при использовании OMP и Pthreads

Для тестирования использовалась модель рычага с сеткой большой размерности (3 млн. дискретных элементов). Запуск проводился на системе с NUMA-конфигурацией: четырехсокетной платформа с 16-ядерными процессорами AMD Opteron. Максимальное ускорение, получаемое с помощью алгоритма, составило 9-10 раз на одном узле. Было также исследовано влияние размерности задачи на деградацию производительности вычислений. На задачах размерностью 13 млн. элементов ускорение продолжает расти до 64 параллельных потоков, в то время как для размерности 1 млн. использовать больше 24 потоков не имеет смысла (рис. 2).

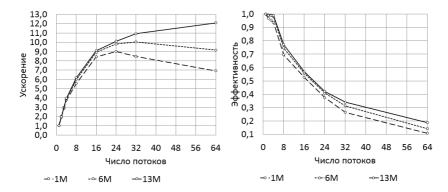


Рис. 2. Сравнение ускорения в многопоточном алгоритме для разной размерности задачи

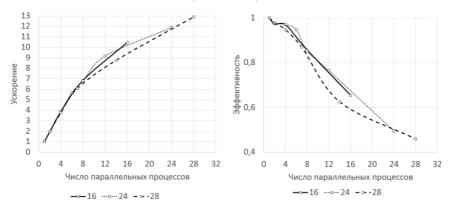
Для запуска моделей на разных узлах был использован метод синхронизации процессов на основе MPI функций коллективного обмена MPI_Alltoall. В методе составляется таблица синхронизации, в которой указываются адреса и объемы пересылаемых данных из памяти одного процесса в память другого процесса. По этой таблице для функции MPI_Alltoall заполняются вектора смещений и длин пересылаемых и принимаемых участков памяти. Используются также функции упаковки и распаковки разреженных данных в непрерывный буфер для пересылки MPI_Pack и MPI_Unpack. Фрагмент кода реализации приведен ниже.

```
Adjacents adjacents = dec.Adjacents(rank);
for (Adjacents adjacent : adjacents)
{
    MPI_Type_indexed(count, adjacent.lengths,
        adjacent.BufferSend, adjacent.BufferSize, &position, comm);
    MPI_Type_commit(&adjacent.type);
}
...
while (SolveIteration())
{
```

```
for (Adjacents adjacent : adjacents)
   MPI Pack(vars, 1, adjacent.Type, adjacent.BufferSend,
     adjacent.BefferSize, &position, comm);
   MPI Alltoall(
     adjacent.BufferSend,
     adjacent.SendCounts.
     adjacent.SendOffsets,
     MPI DOUBLE,
     adjacent.BufferRecv,
     adjacent.RecvCounts,
     adjacent.RecvOffsets,
     MPI DOUBLE,
     comm);
 for (Adjacents adjacent : adjacents)
   MPI Unpack(adjacent.BufferRecv, adjacent.BufferSize,
                     &position, vars, 1, adjacent. Type, comm);
 MPI Barrier(comm);
}
```

Поскольку в общем случае все модели взаимодействуют с несколькими другими моделями, данный метод является наиболее эффективным для рассматриваемой задачи и дает максимальное ускорение в 10-13 раз при запуске на разных узлах (рис. 3), после чего эффективность существенно падает.





7

Рис. 3. Сравнение ускорения при расчете на нескольких узлах кластера

Комбинация методов позволяет получить ускорение, которое в среднем составляет 9-10 раз. Использование параллельных МРІ-процессов, вместо потоков, дает ускорение в 8-9 раз. Исследования, проведенные на различных аппаратных конфигурациях кластера, показали, что узким местом алгоритма является частая синхронизация при малом времени расчета одной итерации, поэтому полученное ускорение является максимально возможным для разработанного параллельного метода, так как, на наш взгляд, параллельный алгоритм является наиболее оптимальным.

Для дальнейшей оптимизации алгоритмов использовалась векторизация на CPU и на GPU. В случае отсутствия движения тел (расчет теплопроводности) были реализованы векторизованные версии вычисления правых частей для AVX-инструкций и технологии CUDA (вычисления на GPU). В первом случае удалось ускорить вычисления в 2.5 раза для двойной точности и в 4 раз для одинарной. В случае с GPU вычисления с двойной точностью ускорились в 7 раз, а с одинарной точностью в 20 раз (рис. 4).

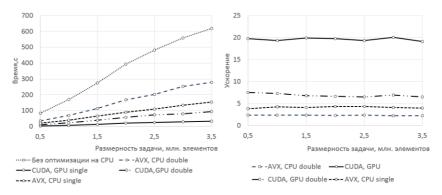


Рис. 4. Сравнение времени расчета и ускорения для векторизации на CPU и GPU

При наличии движения тел (расчет динамического напряженнодеформированного состояния) векторизация была реализована только на СРU, что дало ускорение в 2-2.5 раза при расчете пространственных преобразований (умножение матриц поворота на векторы) и правых частей.

Фрагменты исходного кода для векторизации с использованием AVX приведены ниже.

3. Примеры решенных задач

Разработанная технология распараллеливания использована при моделировании транспортных динамики средств. Проведено решение связанных задач по моделированию элементов подвесок с гидропневматическими рессорами (ГПР). Построены модели теплопередачи в ГПР и элементах редукторов мостов, а также модели напряженно-деформированного состояния в рычагах подвески

Были проведены расчеты для различных моделей транспортных средств, в составе которых были рассчитаны вспомогательные модели теплопередачи в амортизаторах и модели напряженно-деформированного состояния рычагов подвесок. Примеры моделей транспортных средств с кадрами расчета динамики в системе ФРУНД показаны на рис. 5.

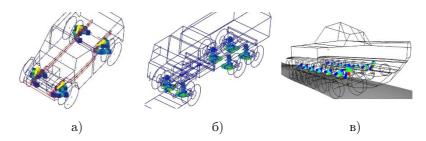


Рис. 5. Модели транспортных средств: а) автомобиль с 4 подвесками и 16 вспомогательными моделями, б) автомобиль с 6 подвесками и 24 вспомогательными моделями, в) платформа с 28 вспомогательными моделями

Для модифицированной модели автомобиля с 4 независимыми подвесками (рис. 4 а) исследовано напряжение в верхних и нижних рычагах подвесок и теплопередача в корпусе и штоке амортизатора. На каждую из 4 подвесок приходилось по 4 решателя, общее число решателей 16.

Аналогичное исследование проведено для модели автомобиля с 6 независимыми подвесками и общим числом решателей 24 (рис. 4 б).

Исследование по нагреву ГПР проведено с использованием модели гусеничной платформы с 28 вспомогательными решателями (рис. 4 в). Результаты по ускорению расчетов с разным числом параллельных процессов, соответствующие этим моделям показаны на рис. 3.

Заключение

В связи с ограничением масштабируемости параллельного метода моделирования динамики систем тел со связями, дальнейшая адаптация алгоритмов для высокопроизводительных кластеров возможна за счет векторизации под новые архитектуры, например, поддерживающие AXV-512, и переноса кода пространственных преобразований для определения положения тел на графические вычислители и сопроцессоры.

Решение задачи ускорения вычислений в моделировании динамики является актуальной, в связи с усложнением моделей, увеличением размерности задач и необходимостью проводить многократные вычислительно сложные расчеты за конечное время.

Список литературы

- [1] Getmanskiy V.V., Gorobtsov A.S., Sergeev E.S., Ismailov T.D., Shapovalov O.V. Concurrent simulation of multibody systems coupled with stress-strain and heat transfer solvers// Journal of Computational Science, 3(6), p.492-497, 2012.
- [2] Сергеев, Е.С., Гетманский В.В., Горобцов А.С. *Перенос системы многотельной динамики на вычислительный кластер* // Научно-технические ведомости Санкт-Петербургского гос. политехн. ун-та, 2010, Вып. 101, с. 93-99.
- [3] Bayo E., Garcia de Jalon J., Serna M.A. A modified Lagrangian formulation for the dynamic analysis of constrained mechanical systems// Comput. Methods Appl. Mech. Eng., 71(2), p. 183-195, 1988.
- [4] Гетманский В.В., Горобцов А.С., Измайлов Т.Д. Распараллеливание расчёта напряжённо-деформированного состояния тела

в многотельной модели методом декомпозиции расчётной области // Известия ВолгГТУ. Серия "Актуальные проблемы управления, вычислительной техники и информатики в технических системах". Вып. 16, ВолгГТУ, № 8 (111), 2013, с. 5-10.

[5] Горобцов А.С., Гетманский В.В., Андреев А.Е., Doan D.T. Simulation and Visualization Software for Vehicle Dynamics Analysis Using Multibody System Approach // Creativity in Intelligent Technologies and Data Science. CIT&DS 2015: Proceedings / ed. by A. Kravets et. al., Springer International Publishing, Switzerland, 2015, p. 379-391.

Об авторах:

Гетманский Виктор Викторович



Младший научный сотрудник кафедры «Высшая математика» Волг Γ ТУ, к.т.н..

Область деятельности: разработка параллельных алгоритмов для программного комплекса Φ РУНД решения задач динамики систем тел со связями.

e-mail: victor.getmanskiy@gmail.com



Горобцов Александр Сергеевич

Заведующий кафедрой Высшая математика ВолгГТУ, д.т.н., старший научный сотрудник.

Область деятельности: Разработка методов формирования и численного интегрирования уравнений систем тел, позволяющих решать задачи для систем произвольной структуры, выделять и преобразовывать избыточные связи. Задачи синтеза управляемого движения многомерных систем.

e-mail: vm@vstu.ru

Образец ссылки на публикацию:

В.В. Гетманский, А.С. Горобцов. Адаптация алгоритмов моделирования систем тел со связями для высокопроизводительного кластера // Программные системы: теория и приложения: электрон. научн. журн. 2015. Т. ?, N ?(??), с. ??—??.

URL: http://psta.psiras.ru/read/???

V.V. Getmanskiy, A.S.Gorobtsov Multibody system dynamics simulation algorithms adaptation for high-performance computing cluster.

ABSTRACT. Parallel algorithms for multibody system dynamics method are considered. Speedup and efficiency of the algorithms for simulation of vehicles are obtained and presented. Comparative study of multi-threaded and cluster versions of algorithms with vectorization techniques is performed. The limitations of scalability for coupled problems algorithms are defined and presented.

Key Words and Phrases: multibody system dynamics, coupled problems, computing cluster, vectorization, multi-threaded algorithm, scaling.