

# ОПЫТ ПРОГРАММИРОВАНИЯ ПАРАЛЛЕЛЬНОЙ АСИНХРОННОЙ МОЛЕКУЛЯРНОЙ ДИНАМИКИ ДЛЯ ПОТОКОВОГО СУПЕРВЫЧИСЛИТЕЛЯ ППВС «БУРАН»

*А.В. Климов, Т.Г. Козлова*

Институт проблем проектирования в микроэлектронике РАН, Москва

Масштабирование задач молекулярной динамики на суперкомпьютерах затрудняется наличием в алгоритмах глобальных барьеров, связанных с организацией пересчета списков соседей. Актуальной является задача написания полностью асинхронного и эффективного алгоритма, опирающегося исключительно на локальные взаимодействия между процессорами, обрабатывающими соседние области. Аналогичные цели ставились другими авторами, при этом использовались традиционные подходы программирования взаимодействий на кластерах: MPI, специальная RDMA библиотека [1], Java RMI [2].

Нами используется потоковая модель вычислений, опирающаяся на односторонние сообщения – токены. Она реализуется потоковым супервычислителем ППВС «БУРАН» [3] и хорошо подходит для написания асинхронных алгоритмов. Распараллеливание производится по принципу пространственной декомпозиции, когда каждый процессор (ядро) отвечает за обработку частиц, находящихся в своей области пространства – кубоиде.

Мы используем метод средней точки (midpoint method), предложенный в работе [1]. Это один из методов «нейтральной территории», в которых расчет взаимодействия производится не обязательно ядром, отвечающим за одну из частиц пары. В данном методе расчет выполняется ядром, в кубоид которого попадает середина отрезка, соединяющего пару частиц. Как показано в [1], в этом варианте объем импортируемой информации о частицах минимален.

Расчет сил проводится с использованием потенциала Леннард-Джонса. Учитываются пары частиц, удаленные друг от друга на расстояние не более радиуса отсечения  $R_c$ . Предполагается, что  $1 < R_c/b < 2$ , где  $b$  – длина стороны кубоида. Уравнения движения интегрируются методом Верле.

В отличие от классического подхода, основанного на формировании и поддержании списков соседей, в методе средней точки в каждом цикле моделирования в кубоид импортируются состояния всех частиц из области импорта, которая простирается не далее чем на 26 соседних кубоидов. Синхронизация опирается только на попарные взаимодействия между соседними ядрами.

Реализация в ППВС имеет уровень зернистости до частиц, то есть состояние каждой частицы передается отдельным токеном с полем данных размером 32 байта. Частица сначала передается двум соседям по Z, затем каждая копия передается соседям по Y, затем по X. Это экономит нагрузку на каналы передачи по сети. Силы суммируются по представителям частиц в каждом ядре, после чего передаются, если требуется, обратно в домашний процессор каждой частицы. Миграция частиц между процессорами объединена с фазой импорта частиц и не требует отдельной коммуникационной фазы.

Поскольку каждая частица передается отдельно, синхронизация обеспечивается счетчиками. Отправитель подсчитывает количество частиц, отправляемых в каждое соседнее ядро, и передает это число туда же, где после их принятия завершается расчет взаимодействий и предварительное суммирование сил. Переданные обратно силы окончательно суммируются, причем завершение суммирования определяется асинхронно по каждой частице, что позволяет начинать расчет уравнения движения и последующий цикл для каждой частицы независимо от других.

Наша работа на данном этапе направлена не на получение значимого прикладного результата, а на определение возможностей и эффективности применения ППВС для задач молекулярной динамики. Предполагается провести серию прогонов как на потактовой модели ППВС, так и на ее эмуляторе на суперкомпьютере. Результаты будут доложены на конференции SMM-2014. Аналогичный вопрос для другой прикладной задачи был рассмотрен в работе [5].

Интересно сравнить возможности и эффективность проектируемой ППВС «Буран» как универсального программируемого супервычислителя, применяемого к задачам молекулярной динамики, с теми же свойствами машины Anton [4], разработанной специально для этих задач компанией D.E. Shaw Research. Будучи построенной из очень специализированных микросхем, эта машина показала 100-кратное превосходство против традиционных кластерных суперЭВМ. Учитывая, что Anton имеет некоторое сходство с ППВС «Буран» (например, потоковая организация вычислений и коммуникаций), мы надеемся, что ППВС «Буран» займет промежуточное положение между традиционными универсальными системами кластерного типа и узко ориентированными машинами специального назначения.

### Литература

1. Bowers, K. J., Chow, E., Xu, X., Dror, R.O., Eastwood, M.P., Gregersen, B.A., Klepeis, J.L., Kolossvary, I., Moraes, M.A., Sacerdoti, F.D., Salmon, J.K., Shan, Y., Shaw, D.E. Scalable Algorithms for Molecular Dynamics Simulations on Commodity Clusters. Proceedings of the ACM/IEEE Conference on Supercomputing (SC06), Article No. 84, New York, NY: IEEE, 2006.
2. Mederski, J., Mikulski, L., Bala, P. Asynchronous Parallel Molecular Dynamics Simulations. Lecture Notes in Computer Science Volume 4967, 2008, pp 439-446.
3. Стемпковский А.Л., Левченко Н.Н., Окунев А.С, Цветков В.В. Параллельная потоковая вычислительная система — дальнейшее развитие архитектуры и структурной организации вычислительной системы с автоматическим распределением ресурсов // "Информационные технологии", 2008, №10, с.2–7.
4. Dror, R.O., Young, C., Shaw, D.E. Anton, a special-purpose molecular simulation machine. In: Encyclopedia of parallel computing, Springer, 2011, pp. 60-71.
5. Климов А.В., Поликарпов М.И. Опыт программирования одной задачи квантовой теории поля для потокового суперпроцессора ППВС «БУРАН». Тезисы докладов XIV Международной Конференции «Супервычисления и математическое моделирование», г. Саров, 1-5 октября 2012 года, РФЯЦ-ВНИИЭФ, Саров, 2012, с. 105-106.

# A PROGRAMMING EXPERIENCE WITH PARALLEL ASYNCHRONOUS MOLECULAR DYNAMICS IN DATAFLOW SUPERPROCESSOR PDCS "BURAN"

*A.V. Klimov, T.G. Kozlova*

Institute of Design Problems in Microelectronics,  
Russian Academy of Sciences, Moscow

Scaling molecular dynamics on supercomputers is largely complicated by the presence of global barriers which are used to maintain lists of neighbor particles. There is a challenge of creating a fully asynchronous and efficient algorithm relying solely on local interactions between the cores responsible for nearby areas. Similar goals have been pursued by other authors while using traditional approaches to interactions on clusters: MPI, special RDMA library [1], Java RMI [2].

We use the dataflow computation model based on one-way messages called tokens. It is implemented by the dataflow superprocessor PDCS "Buran" [3] and is well suited for writing asynchronous algorithms. Parallelization is based on spatial decomposition, where each processor is responsible for particles in its own region of space called cuboid.

We use the midpoint method proposed in [1]. This is a kind of a "neutral territory" method, which sometimes computes an interaction between two particles in a core on which neither particle resides. In the midpoint method it is a core whose cuboid contains the midpoint of the segment between the two particles. As shown in [1], in this embodiment, the volume of transmitted particle information is minimal.

Forces are calculated using Lennard-Jones potential taking into account pairs of particles separated by a distance less than the cut-off radius  $R_c$ . It is assumed that  $1 < R_c/b < 2$ , where  $b$  is a cuboid side length. The motion equations are integrated by Verlet method.

In contrast to the classical approach based on the creation and maintenance of neighbor lists, in the midpoint method in each model cycle each home cuboid imports states of all the particles of the import region that extends to 26 neighbor cuboids. Synchronization relies only on pairwise interactions between neighbor cores.

The implementation in PDCS has a level of granularity to particles, i.e., the state of each particle is passed as a separate token with the data field of some 32 bytes. A particle is first passed to the two neighbors over Z, each copy is then passed to neighbors over Y and finally over X. This way, the network communication load is saved. The forces are summed up over interacting particles inside each core and then passed back, if needed, to the home core of each particle. Particle migration between cores is unified with the particle import phase and does not require a separate communication phase.

Since particles are passed individually, the synchronization is ensured by counters. The sender calculates the amount of particles sent to each neighboring core and passes the number to each that core, which, after the receipt, completes the calculation of interactions and the preliminary summation of forces. The returned forces are added finally, and the completion is determined asynchronously on each particle, which allows us to start the calculation of motion equations and the subsequent cycle for each particle independently.

Our work at this stage is aimed not to obtain meaningful applied results, but to identify opportunities and effectiveness of using the PDCS for molecular dynamics problems. We are going to hold a series of runs in a cycle-accurate model of the PDCS as well as in its emulator in a cluster supercomputer. The results will be presented at the conference SMM-2014. A similar question for other application tasks was considered in [5].

It is interesting to compare the capabilities and effectiveness of the developed PDCS "Buran" as a universal programmable superprocessor applied to molecular dynamics with those of the dedicated machine Anton [4] constructed by D.E. Shaw Research. Built of highly specialized chips, this machine showed a 100-fold speed-up against the traditional cluster supercomputer. Given that Anton has some similarities with the PDCS "Buran" (e.g. dataflow organization of computation and communication), we hope that the PDCS "Buran" will occupy an intermediate position between the traditional universal systems of the cluster type and narrowly focused special purpose supercomputers.

### References

6. Bowers, K. J., Chow, E., Xu, X., Dror, R.O., Eastwood, M.P., Gregersen, B.A., Klepeis, J.L., Kolossvary, I., Moraes, M.A., Sacerdoti, F.D., Salmon, J.K., Shan, Y., Shaw, D.E. Scalable Algorithms for Molecular Dynamics Simulations on Commodity Clusters. Proceedings of the ACM/IEEE Conference on Supercomputing (SC06), Article No. 84, New York, NY: IEEE, 2006.
7. Mederski, J., Mikulski, L., Bala, P. Asynchronous Parallel Molecular Dynamics Simulations. Lecture Notes in Computer Science Volume 4967, 2008, pp 439-446.
8. Stempkovsky A.L., Levchenko N.N., Okunev A.S, Tsvetkov V.V. Parallel dataflow computing system – the further development of architecture and the structural organization of the computing system with automatic distribution of resources // "Informatsionnyje tekhnologii", 2008, No. 10, pp.2-7.
9. Dror, R.O., Young, C., Shaw, D.E. Anton, a special-purpose molecular simulation machine. In: Encyclopedia of parallel computing, Springer, 2011, pp. 60-71.
10. Klimov A.V., Polikarpov M.I. A programming experience of a Quantum Field Theory problem for dataflow superprocessor PDCS "Buran". Preliminary proceedings of XIV International Conference «Supercomputing and Mathematical Modeling», RFNC-VNIIEF, Sarov, 2012, pp. 105-106.