

Автоматизации потока вычислительных работ для моделирования живых систем на суперкомпьютерах*

К.С. Ушенин^{1,2}, Е.Ю.Куклин^{1,2}, Д.А.Бёрдов², А.В. Созыкин^{1,2}

Институт математики и механики им. Н.Н. Красовского УрО РАН¹, Уральский федеральный университет²

В статье рассматривается подход к созданию дружественной пользователю системы автоматизации потока вычислительных работ для моделирования живых систем LiFlow, которая позволяет удобно проводить большие серии вычислительных экспериментов на суперкомпьютерах. При исследовании живых систем часто требуется проводить большое количество вычислительных экспериментов с одной и той же моделью, но разными параметрами. Система LiFlow предоставляет возможность в удобном виде подготавливать исходные данные для большого количества экспериментов, запускать эксперименты на выполнение на суперкомпьютере, хранить и каталогизировать результаты вычислений на системе хранения. Важной особенностью системы является простота и удобство в использовании: система ориентирована на исследователей в области математической биологии и биофизики, для ее применения не нужно иметь глубоких знаний в области параллельных вычислений. Приводится пример использования разработанной системы для проведения исследований с использованием трехмерных моделей левого желудочка сердца.

Ключевые слова: параллельные вычисления, суперкомпьютеры, моделирование живых систем, вычислительный поток работ

1. Введение

Исследования живых систем требуют больших вычислительных ресурсов, их невозможно выполнить за разумное время без параллельных вычислительных систем и суперкомпьютеров. Однако использование параллельных вычислительных систем требует высокой квалификации в области компьютерных наук, которыми многие исследователи, занимающиеся моделированием живых систем, не обладают. Для устранения этой проблемы разрабатываются системы научных вычислительных потоков работ (scientific computational workflow systems) [1-4], которые позволяют объединять распределенные вычислительные ресурсы, приложения, наборы данных и инструментальные средства для проведения вычислительных экспериментов. Такие системы скрывают от пользователей сложность распределенных вычислительных систем и позволяют описать поток работ в графическом виде. Существующие системы научных вычислительных потоков работ способны использовать вычислительными ресурсами разных типов (GRID, суперкомпьютеры, распределенные системы), хранилищами данных (локальное, сетевое, облачное), инструментальными средствами (визуализация, статистическая обработка и т.п.). В результате такие системы очень сложно устроены, тяжелы в установке, сопровождении и практическом применении.

Другим возможным вариантом решения проблемы является использования сред, обеспечивающих интеграцию пакетов прикладных программ с суперкомпьютерами. Примером подобной системы является DiVTB [5], которая предоставляет удобный графический интерфейс пользователя, в котором можно задать параметры вычислительного эксперимента и запустить его на выполнение на суперкомпьютере. Однако такие системы не обеспечивают автоматизацию часто встречающейся в моделировании живых систем задачи запуска серии вычислительных экспериментов (от нескольких десятков до нескольких тысяч) с одной моделью, но с разными параметрами.

* Работа поддержана грантом Президиума РАН программа П.4П "Фундаментальные программы математического моделирования", проект номер 22, и проектом Комплексной программы УрО РАН № 15-7-1-26

Актуальной является разработка легковесной системы управления потоком вычислительных работ для моделирования живых систем, предназначенной для запуска серии вычислительных экспериментов. В такой системе поток вычислительных работ будет достаточно простым. Пользователь тратит много времени на подготовку исходных данных и параметров для проведения экспериментов, а также на обработку результатов. Кроме того, результаты экспериментов нужно хранить длительное время, а также иметь возможность быстро находить нужные результаты среди серии, что сложно при большом количестве экспериментов. Именно на автоматизацию этих процессов и рассчитана система LiFlow, представленная в данной статье.

В работе были использованы суперкомпьютер «УРАН» Института математики и механики УрО РАН и вычислительный кластер Уральского федерального университета.

2. Архитектура системы LiFlow

Система LiFlow рассчитана на упрощенный вариант потока работ, который представлен на рис. 1. На основании созданного пользователем описания серии экспериментов выполняется подготовка данных и программ, которые необходимы для запуска серии на суперкомпьютере. Подготовка включает выбор программы для экспериментов, которая реализует необходимую модель, генерация конфигурационных файлов с необходимыми параметрами, создание файлов с исходными данными для всех экспериментов серии. Затем эксперименты запускаются на суперкомпьютере, результаты экспериментов записываются в хранилище, где может выполняться их обработка: визуализация, статистическая обработка и т.п. От пользователя требуется только сформировать описание эксперимента, все остальное выполняется автоматически. При необходимости пользователь имеет возможность проводить обработку результатов экспериментов в хранилище вручную с использованием дополнительных инструментов.



Рис. 1. Поток работ системы LiFlow

Архитектура системы LiFlow показана на рис. 2. Инструмент подготовки серии экспериментов запускается на компьютере пользователя и предназначен для формирования так называемого вычислительного пакета, который включает в себя все необходимое для запуска серии экспериментов на суперкомпьютере. В состав вычислительного пакета входят: исходный код программы моделирования (загружается из хранилища кода), генератор серии экспериментов, начальные данные и параметры запуска программы моделирования. Вычислительный пакет передается на исполнение на суперкомпьютер с использованием интерфейса вычислительных ресурсов. На суперкомпьютере выполняется компиляция кода программы моделирования и запускается генератор серии экспериментов, формирующий несколько заданий для расчета, которые отличаются друг от друга разными значениями параметров (и, возможно, другими показателями). Задания ставятся в очередь на выполнение на суперкомпьютере. После завершения задания, результаты эксперимента записываются с помощью интерфейса на систему хранения. Дополнительно формируется запись в каталог экспериментов, с описанием эксперимента, который выполнялся в рамках данной задачи, с целью повторного использования результатов.

Каталог экспериментов также служит для обмена исходными данными и результатами моделирования между разными исследователями. При подготовке серии экспериментов, пользователь может просмотреть, какие задачи и с какими параметрами запускали его коллеги.

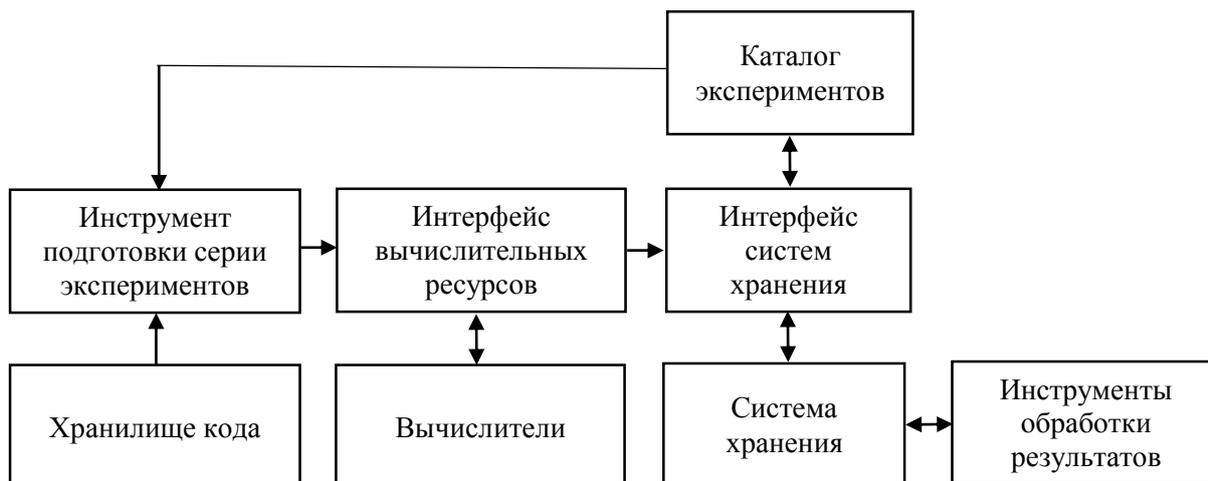


Рис. 2. Архитектура системы LiFlow

3. Практическая реализация

В настоящее время выполнена первая очередь реализации системы LiFlow. В качестве вычислительного пакета используется каталог, в отдельные подкаталоги которого записаны исходный код программы моделирования, генератор экспериментов, исходные данные для генератора, и сценарий запуска экспериментов.

В текущем варианте реализации генератор серии экспериментов представляет собой сценарий, создающих серию экспериментов используя конфигурационный файл программы моделирования в качестве шаблона и варьируя в нем параметры. Поддерживается два варианта варьирования параметров на основании конфигурационных файлов, содержащих:

- Диапазон изменения параметра: начальное значение, конечное значение, шаг изменения. По одной записи генерируется несколько экспериментов.
- Конкретное значение параметра, используемое для эксперимента. Необходимо задать значения параметров для всех желаемых экспериментов серии.

Сценарий запуска экспериментов обеспечивает копирование вычислительного пакета на суперкомпьютер, компиляцию исходного кода программы моделирования и запуск генератора серии экспериментов.

Интерфейс с вычислительным ресурсом реализован с использованием системы запуска задач SLURM [6], которая используется на суперкомпьютере «УРАН» Института математики и механики УрО РАН и вычислительный кластер Уральского федерального университета. Сценарий запуска эксперимента ставит сгенерированные задачи для серии экспериментов в очередь задач SLURM. После завершения задачи запускается сценарий, который копирует данные на систему хранения по протоколу SFTP.

Все сценарии реализованы на языке Python. Каталог экспериментов и хранилище кода компьютерной модели реализовано в виде git репозитория, предоставляемого сторонними сервисами.

Недостатком текущей реализации системы является отсутствие механизма обработки отказов. Если при выполнении эксперимента произойдет ошибка, он не будет повторен. Такой подход выбран из-за того, что отказ может произойти не только из-за неисправности оборудования или системного программного обеспечения, но и из-за ошибки в программе моделирования или неверной комбинации параметров, для которых вычисления не могут быть выполнены. Повторный запуск в таких ситуациях не приведет к решению проблемы, а только создаст ненужную нагрузку на суперкомпьютер. Однако сбой при выполнении одного эксперимента не ведет к остановке выполнения всей серии экспериментов.

4. Примеры использования системы LiFlow

Разработанная система LiFlow применялась для проведения экспериментов моделирования левого желудочка (ЛЖ) сердца человека. В работе [7] проводилось исследование влияния хода волокон в анатомической модели ЛЖ на скорость и согласованность его электрофизиологической активации. Проводилась серия из 55 экспериментов, в которой все параметры кроме двух, отвечающих за направление хода волокон в электрофизиологической модели, были зафиксированы.

Та же система может быть применена для повторения результатов исследований, которые ранее проводились вручную. При исследовании скорости возбуждения миокарда левого желудочка сердца человека с использованием анатомической модели, которая позволяла менять форму желудочка и направление хода волокон в нем [8], проводилось две серии экспериментов. В серии экспериментов изменялась область начальной активации, ход направления волокон в анатомической модели и отношение коэффициентов в тензоре диффузии электрофизиологической модели. В общей сложности, работа основана на более чем на 36 экспериментах, значения параметров для которых можно сгенерировать по определенным правилам.

В работе [9] проводилось исследование динамики спиральных волн в модели ЛЖ сердца человека с различной геометрией и направлением хода волокон. В ходе исследования было выполнено несколько серий экспериментов с анатомиями, которые приближают нормальную и патологическую анатомию ЛЖ. В каждой серии экспериментов варьировалась толщина верхушки, значение тензора диффузии и место начального запуска волны. В общей сложности работа основана на более чем на 84 экспериментах.

6. Заключение

Представлена система автоматизации потока вычислительных работ LiFlow, предназначенная для автоматизации запуска серии из большого числа вычислительных экспериментов для моделирования живых систем на суперкомпьютерах. Отличительной особенностью системы LiFlow является автоматическая генерация исходных данных и параметров для проведения серии экспериментов, а также автоматическое сохранение и каталогизация результатов экспериментов в долговременном хранилище. Система была применена для моделирования ЛЖ сердца человека. Использование LiFlow позволило значительно сократить время подготовки серии экспериментов, а также сделало обработку результатов более удобной.

В качестве направлений дальнейших работ можно выделить:

- Разработка графического интерфейса инструмента подготовки серии экспериментов, позволяющего пользователю нажатием одной кнопки запустить большое количество экспериментов на суперкомпьютере.
- Реализация интерфейса вычислительных ресурсов для систем запуска задач, отличных от SLURM, а также для облачных систем.
- Реализация интерфейса систем хранения для облачных хранилищ данных.
- Создание более сложных и универсальных генераторов серий экспериментов.

Литература

1. Hull D., Wolstencroft K., Stevens R., Goble C., Pocock M. R., Li P., Oinn T. Taverna: A tool for building and running workflows of services // *Nucleic Acids Res.* 2006 Jul 1; 34(Web Server issue): W729–W732.
2. Ghanem M., Curcin V., Wendel P. Guo, Y. Building and Using Analytical Workflows in Discovery Net // *Data Mining Techniques in Grid Computing Environments* (ed W. Dubitzky), John Wiley & Sons, Ltd, Chichester, UK. 2008. doi: 10.1002/9780470699904.ch8.
3. Taylor I., Shields M., Wang I., Harrison A. Visual Grid Workflow in Triana // *Journal of Grid Computing.* 2005, Vol. 3, No. 3, P. 153-169.

4. Ludäscher B., Altintas I., Berkley Ch., Higgins D., Jaeger E., Jones M., Lee E.A., Tao J., Zhao Y. Scientific workflow management and the Kepler system: Research Articles // *Concurrency and Computation: Practice & Experience - Workflow in Grid Systems*. 2006. Vol. 18, No. 10. P. 1039-1065.
5. Савченко Д.И., Радченко Г.И. DiVTB Server: среда выполнения виртуальных экспериментов // *Параллельные вычислительные технологии (ПаВТ'2013): труды международной научной конференции (1–5 апреля 2013 г., г. Челябинск)*. Челябинск: Издательский центр ЮУрГУ, 2013. С. 532-539.
6. Jette M.A., Yoo A.B., Grondona M. SLURM: Simple Linux Utility for Resource Management // *Lecture Notes in Computer Science: Proceedings of Job Scheduling Strategies for Parallel Processing (JSSPP) 2003*. Vol 2862. P.44-60.
7. Ushenin K, Byordov D. An HPC-Based Approach to Study Living System Computational Model Parameter Dependency // *Proceedings of the 1st Ural Workshop on Parallel, Distributed, and Cloud Computing for Young Scientists Yekaterinburg, Russia*. CEUR Workshop Proceedings Vol. 1513, P. 67–74.
8. Pravdin S.F., Dierckx H., Katsnelson L.B., Solovyova O., Markhasin V.S., Panfilov A.V. Electrical Wave Propagation in an Anisotropic Model of the Left Ventricle Based on Analytical Description of Cardiac Architecture. *PLoS ONE* 9(5): e93617. 2014.
9. Pravdin S., Dierckx H, Markhasin V.S., and Panfilov A.V. Drift of Scroll Wave Filaments in an Anisotropic Model of the Left Ventricle of the Human Heart. *BioMed Research International*, vol. 2015, Article ID 389830, 13 pages, 2015. doi:10.1155/2015/389830.

Computational workflow system for simulation of living systems on supercomputers

K.S.Ushenin^{1,2}, E.Y. Kuklin^{1,2}, D.A. Byordov², A.V. Sozykin^{1,2}

N.N. Krasovskii Institute of Mathematics and Mechanic UB RAS¹, Ural Federal University²

The paper describes the approach to create user-friendly computational workflow system LiFlow for living system simulation. The LiFlow system allows conducting large series of computational experiments on supercomputers. Living system simulation often requires carrying out large amount of experiments with the same model but different parameter values. The LiFlow system provides the convenient interface that allows preparing input data for large number of experiments, executing the experiments on the supercomputer, and storing and cataloguing results of experiments on the storage system. The distinctive feature of the system is the simplicity and ease of use: the system is intended to be used by researchers in mathematical biology and biophysics, who do not have deep knowledge in parallel computing. The examples of the LiFlow system utilization for left ventricle of human heart simulation are described.

Keywords: parallel computing, supercomputing, living system modeling, computational workflow

References

1. Hull D., Wolstencroft K., Stevens R., Goble C., Pocock M. R., Li P., Oinn T. Taverna: A tool for building and running workflows of services // *Nucleic Acids Res.* 2006 Jul 1; 34(Web Server issue): W729–W732.
2. Ghanem M., Curcin V., Wendel P. Guo, Y. Building and Using Analytical Workflows in Discovery Net // *Data Mining Techniques in Grid Computing Environments* (ed W. Dubitzky), John Wiley & Sons, Ltd, Chichester, UK. 2008. doi: 10.1002/9780470699904.ch8.
3. Taylor I., Shields M., Wang I., Harrison A. Visual Grid Workflow in Triana // *Journal of Grid Computing.* 2005, Vol. 3, No. 3, P. 153-169.
4. Ludäscher B., Altintas I., Berkley Ch., Higgins D., Jaeger E., Jones M., Lee E.A., Tao J., Zhao Y. Scientific workflow management and the Kepler system: Research Articles // *Concurrency and Computation: Practice & Experience - Workflow in Grid Systems.* 2006. Vol. 18, No. 10. P. 1039-1065.
5. Savchenko D.I., Radchenko G.I. DiVTB Server: sreda vypolneniya virtual'nykh eksperimentov [DiVTB Server: an environment for virtual experiments executions] // *Parallelnye vychislitel'nye tekhnologii (PaVT'2013): trudy mezhduna-rodnoy nauchnoy konferentsii (1-5 April 2013, Chelyabinsk) [Parallel Computational Technologies (PCT'2010): Proceedings of the International Scientific Conference (1-5 April 2013, Chelyabinsk, Russia)]* Chelyabinsk, Publishing of the South Ural State University, 2013. P. 532-539.
6. Jette M.A., Yoo A.B., Grondona M. SLURM: Simple Linux Utility for Resource Management // *Lecture Notes in Computer Science: Proceedings of Job Scheduling Strategies for Parallel Processing (JSSPP) 2003.* Vol 2862. P.44-60.
7. Ushenin K, Byordov D. An HPC-Based Approach to Study Living System Computational Model Parameter Dependency // *Proceedings of the 1st Ural Workshop on Parallel, Distributed, and Cloud Computing for Young Scientists Yekaterinburg, Russia.* CEUR Workshop Proceedings Vol. 1513, P. 67–74.

8. Pravdin S.F., Dierckx H., Katsnelson L.B., Solovyova O., Markhasin V.S., Panfilov A.V. Electrical Wave Propagation in an Anisotropic Model of the Left Ventricle Based on Analytical Description of Cardiac Architecture. PLoS ONE 9(5): e93617. 2014.
9. Pravdin S., Dierckx H, Markhasin V.S., and Panfilov A.V. Drift of Scroll Wave Filaments in an Anisotropic Model of the Left Ventricle of the Human Heart. BioMed Research International, vol. 2015, Article ID 389830, 13 pages, 2015. doi:10.1155/2015/389830.