УДК 575;004.94;579.23+578.81

ВЫСОКОПРОИЗВОДИТЕЛЬНОЕ МОДЕЛИРОВАНИЕ ЭВОЛЮЦИИ ПРОКАРИОТИЧЕСКИХ СООБЩЕСТВ С ИСПОЛЬЗОВАНИЕМ ПРОГРАММНОГО КОМПЛЕКСА «ГАПЛОИДНЫЙ ЭВОЛЮЦИОННЫЙ КОНСТРУКТОР»

© 2012 г. **З.С. Мустафин[1, 2], Ю.Г. Матушкин[1, 2], С.А. Лашин[1, 2]**

[1] Федеральное государственное бюджетное учреждение науки Институт цитологии и генетики Сибирского отделения Российской академии наук, Новосибирск, Россия, e-mail: Zidane-7@yandex.ru;
[2] Новосибирский национальный исследовательский государственный университет, Новосибирск, Россия

Поступила в редакцию 15 июля 2012 г. Принята к публикации 31 августа 2012 г.

В работе приведены результаты разработки высокопроизводительной версии программного комплекса «Гаплоидный эволюционный конструктор» (http://evol-constructor.bionet.nsc.ru), предназначенного для моделирования функционирования и эволюции прокариотических сообществ. Разработана параллельная версия программы, предназначенная для работы на высокопроизводительных кластерах с поддержкой MPI. Оказалось, что общее ускорение параллельной версии программного комплекса почти линейно зависит от числа используемых процессоров, и время расчета сложных моделей сообществ на кластере ЦКП «Биоинформатика» СО РАН уменьшилось с десятков часов до нескольких минут.

Ключевые слова: микробное сообщество, оптимизация, параллельное программирование, моделирование, эволюция.

ВВЕДЕНИЕ

Моделирование эволюции и функционирования прокариотических сообществ является актуальной задачей современной системной биологии как с фундаментальной, так и с практической точек зрения. Прокариоты способны катализировать огромное количество разнообразных биохимических реакций и потому являются участниками большинства природных процессов (Заварзин, 2003). Многие виды прокариот используются человеком в технологических процессах. Математическое и компьютерное моделирование поведения и эволюции прокариотических сообществ в тех или иных условиях можно использовать для нужд современной биологии и медицины (Wang, Post, 2012). Данная работа посвящена развитию методов моделирования эволюции и функционирования сообществ одноклеточных гаплоидных организмов и программного комплекса «Гаплоидный эволюционный конструктор» (ГЭК – доступен по адресу http://evol-constructor.bionet.nsc.ru).

В компьютерной модели ГЭК рассматриваются следующие уровни биологической организации: геномный, метаболический, популяционный и экоценотический (Lashin *et al*., 2012). Также в ГЭК реализована возможность моделирования функционирования генных сетей с учетом популяционных и экоценотических факторов. Впервые применен подход описания «обобщенных геномов популяций» с помощью техники генетических спектров, что позволяет значительно сократить время расчета модели с сохранением точности, сопоставимой с точностью классических индивидуально ориентированных моделей. Кроме того, ГЭК позволяет разрабатывать множество разных подмоделей для одного слоя, фактически создавая библиотеки подмоделей. При этом поскольку интерфейсы взаимодействия между подмоделями

разных слоев (в частности входные и выходные данные) четко специфицированы, при построении общей модели появляется возможность комбинирования различных сочетаний подмоделей разных слоев. Это позволяет исследовать различные аспекты эволюционного процесса в рамках одного программного средства.

ГЭК позволяет моделировать мутации, горизонтальный перенос и потерю генов, фиксацию генетических изменений. Горизонтальный перенос, а также потеря генетического материала изменяют уникальный набор метаболических реакций, характерных для данной популяции клеток («вида»), т. е. фактически структуру клеточного метаболизма, что моделирует появление новых штаммов/«видов» клеток и лежит в основе развития биоразнообразия, моделируемого ГЭК. К числу других возможностей ГЭК относятся возможности моделирования фаговой инфекции (Лашин и др., 2011) и функционирования генных сетей (Lashin, Matushkin, 2012). Интеграция методики моделирования генных сетей открывает широкие методические перспективы для исследования эволюции генных сетей с учетом надгенетических и надорганизменных уровней биологической организации, таких, как популяционный и экоценотический.

В работе приведены результаты разработки и программной реализации высокопроизводительного алгоритма моделирования популяционных процессов в рамках ГЭК: описаны эффективный алгоритм расчета изменения численности прокариотической популяции и параллельная реализация алгоритма с использованием технологии MPI (Корнеев, 2002); проведены тестирование и оценка производительности алгоритма на высокопроизводительном кластере.

Алгоритм расчета изменения численности популяции

Анализ однопроцессорной версии ГЭК при помощи профилировщика Intel Parallel Amplifier (http://software.intel.com/ru-ru/articles/intel-parallel-studio-home) показал, что при моделировании сообществ с высоким генетическим разнообразием (10^6–10^8 уникальных аллельных комбинаций) практически все время выполнения уходит на вычисление единственной функции – изменения численности популяции независимо от типа трофической стратегии (рис. 1).

На рис. 1 показано, что с ростом количества аллельных комбинаций время выполнения программы практически полностью концентрируется на функции изменения численности популяции. Опишем эту функцию более формально.

Основным объектом, с которым работает эта функция, является ОГП (обобщенный геном популяции). ОГП в ГЭК – это многомерное распределение частот аллелей для всех генов, присутствующих у особей популяции. Наличие гена в клетках популяции в ГЭК подразумевает наличие в метаболизме этих клеток процесса синтеза или утилизации соответствующего субстрата; аллель как вариант гена определяет конкретное значение константы скорости соответствующего процесса. Заметим, что в рамках ГЭК каждый признак однозначно определяется одним геном, и ген рассматривается как единица наследования.

На рис. 2 показан ОГП, содержащий 4 гена: с тремя, одним, четырьмя, двумя и возможными аллельными вариантами соответственно. Для расчета прироста численности популяции с учетом различной приспособленности особей, несущих разные аллельные комбинации, необходимо рассмотреть все возможные такие комбинации и учесть в каждой из них изменение размера субпопуляции с помощью заданной пользователем функции роста популяции, так называемой трофической стратегии (Лашин и др., 2009), а затем посчитать итоговый размер популяции и концентрации аллельных вариантов.

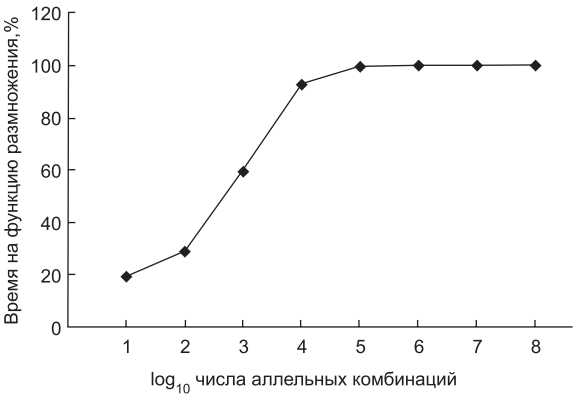

Рис. 1. Время выполнения функции изменения численности популяции относительно времени работы программы.

```
0(0.5)  1(0.2)  2(0.3)          – распределение аллелей для гена 1
3(1)                           – распределение аллелей для гена 2
4(0.1)  5(0.1)  6(0.4)  7(0.4) – распределение аллелей для гена 3
8(0.9)  9(0.1)                 – распределение аллелей для гена 4
```

Рис. 2. Пример представления обобщенного генома популяции в ГЭК.

Идея предлагаемого в статье алгоритма, реализующего изменение численности популяции, заключается в переборе всех возможных аллельных комбинаций (всего их n) в одном цикле длины n. Комбинации записываются с помощью специального массива, состоящего из индексов каждого аллельного варианта комбинации. В качестве примера рассмотрим табл. 1.

Все аллельные варианты всех генов записываются в один массив (полужирным текстом выделены текущие аллельные варианты каждого гена). С помощью массива индексов (в стартовой комбинации он имеет вид (0, 3, 4, 8)) выписывается первая комбинация, и индекс текущего аллельного варианта в последнем гене увеличивается на 1. Если при этом в последнем гене получен последний аллельный вариант, то значение выбранного аллельного варианта в предыдущем гене увеличивается на 1, а в последнем гене выбирается первый аллельный вариант. Таким образом, с помощью массива индексов осуществляется полный перебор всех аллельных комбинаций в популяции.

Достоинство такого подхода состоит в том, что полученный цикл может быть разбит на любое число параллельных процессов (протестировано до 900 процессов). Каждому процессу необходимо «знать» массивы значений и концентраций аллельных вариантов (так называемые «развертки» значений и концентраций), а также стартовую и конечную позиции своего фрагмента цикла. По окончании вычислений в каждом процессе полученные данные суммируются в корневой процесс с помощью функции массового суммирования MPI_Reduce, происходит присваивание посчитанных результатов исходному объекту и алгоритм завершает работу (рис. 3). Все пересылки реализуются с помощью функций MPI_Bcast и MPI_Reduce, все процессы выполняют приблизительно одинаковый объем работы (несущественные отличия возникают, если число комбинаций не кратно числу процессов).

Таблица 1

Возможные аллельные комбинации

Аллельные варианты всех генов	Полученные комбинации
(**0**, 1, 2, **3**, **4**, 5, 6, 7, **8**, 9)	(0, 3, 4, 8)
(**0**, 1, 2, **3**, **4**, 5, 6, 7, 8, **9**)	(0, 3, 4, 9)
(**0**, 1, 2, **3**, 4, **5**, 6, 7, **8**, 9)	(0, 3, 5, 8)
…	…
(0, 1, **2**, **3**, 4, 5, 6, 7, 8, **9**)	(2, 3, 7, 9)

Тестирование алгоритма

В рамках работы было проведено тестирование алгоритма на 6-ядерных процессорах X5670 2.93 GHz (Westmere) кластера НКС 30-Т (http://bioinformatics.bionet.nsc.ru/). Алгоритм был верифицирован на тестовом наборе сценариев ГЭК (Лашин и др., 2009; Lashin *et al*., 2012). Затем были составлены специальные нагрузочные тесты (10^6–10^8 аллельных комбинаций). Результаты тестирования алгоритма приведены в табл. 2.

В ряде вычислительных экспериментов было показано, что эффективность распараллеливания иногда превышает значение 1. Это объясняется тем, что на различных узлах кластера могут быть получены различные значения времени выполнения программы. Расчет параллельной версии на узлах, отличных от узлов, на которых рассчитывалась последовательная версия, может дать увеличение или уменьшение эффективности. Среднеквадратичное отклонение для каждой выборки результатов показано в таблице в столбце «дисперсия», с округлением вверх до секунд. Столбец «ускорение» показывает эффективность распараллеливания для различного числа процессов. Получено практически линейное ускорение на протяжении всего тестирования.

Таким образом, время выполнения программы на современных процессорах сократилось с 8 ч до 2 мин, при этом требуется всего 12 уз-

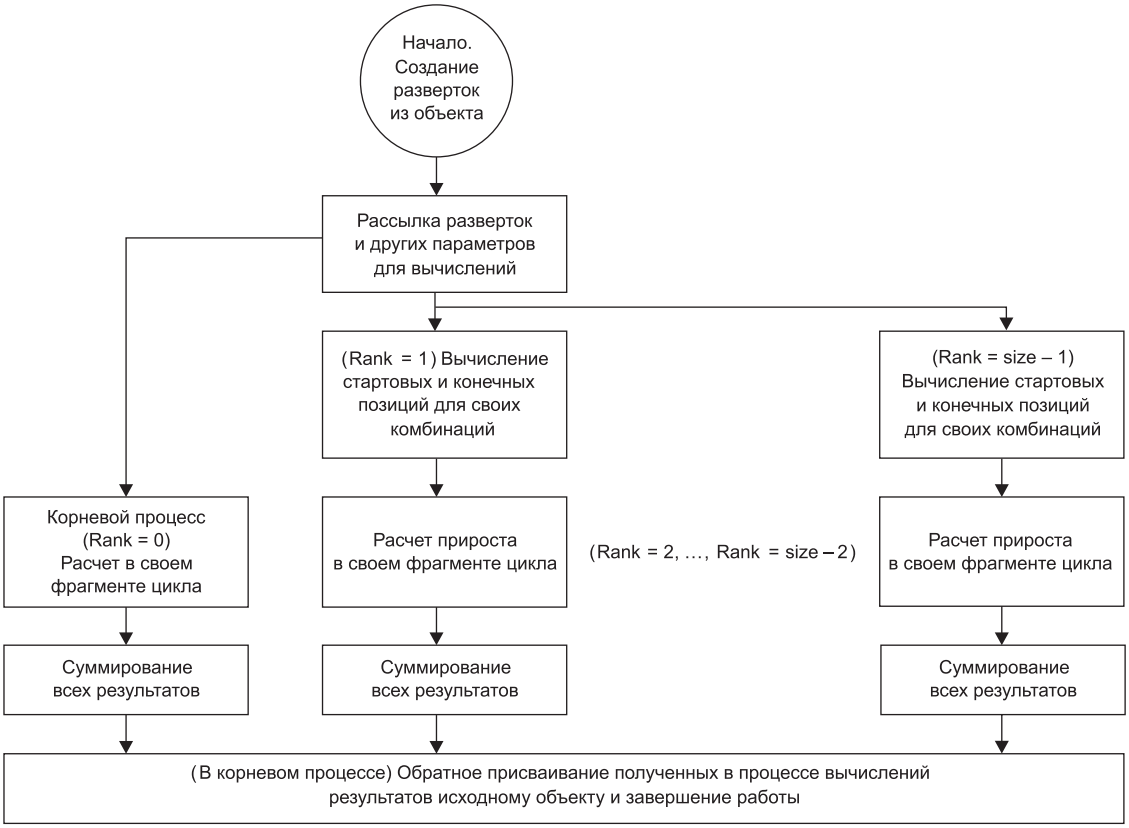

Рис. 3. Схема распараллеливания алгоритма.

Таблица 2

Результаты тестирования на X5670

Количество процессов	Время работы ч:м:с	Эффективность распараллеливания	Ускорение	Дисперсия, с
1	8:02:26	1	1	249
2	4:10:24	0,9633	1,9266	65
4	2:05:20	0,9623	3,8492	188
8	1:02:03	0,9718	7,7744	4
16	0:31:06	0,9695	15,512	3
24	0:20:48	0,9664	23,1936	3
36	0:13:46	0,9734	35,0424	1
64	0:07:52	0,9582	61,3248	2
96	0:05:13	0,9633	92,4768	1
144	0:03:32	0,9481	136,5264	1
264	0:02:03	0,8914	235,3296	2

лов вычислительного комплекса, и на каждый параллельный процесс необходимо всего 2 Мб оперативной памяти.

ЗАКЛЮЧЕНИЕ

Нами была проведена оптимизация алгоритма расчета изменения численности прокариотической популяции в программном комплексе ГЭК, которая позволила проводить расчеты на параллельных высокопроизводительных вычислительных кластерах. Чем сложнее структура моделируемого сообщества, чем больше в этом сообществе генетическое разнообразие, тем большую долю в выполнении программы занимает выполнение функции расчета изменения численности популяции, соответственно, на сложных моделях выигрыш от оптимизации является максимальным. Показано, что ускорение в зависимости от числа использованных процессоров близко к линейному и достигает максимума при моделировании сообществ с большим генетическим разнообразием. Мы считаем, что именно такие сообщества представляют наибольший интерес для исследования, и надеемся, что высокопроизводительная версия ГЭК, представленная в данной статье, позволит пользователю увеличить сложность и разнообразие моделируемых биологических ситуаций, что будет способствовать развитию эволюционной биологии.

БЛАГОДАРНОСТИ

Работа была поддержана следующими грантами: РФФИ 12-07-00671, Междисциплинарные интеграционные проекты СО РАН №№ 47, 87, Научная школа-5278.2012.4; Программа РАН № 28.

ЛИТЕРАТУРА

Заварзин Г.А. Лекции по природоведческой микробиологии. М.: Наука, 2003. С. 348.

Корнеев В.Д. Параллельное программирование в MPI. 2-е изд. испр. Новосибирск: ИВМиМГ СО РАН, 2002. 215 с.

Лашин С.А., Суслов В.В., Матушкин Ю.Г. Моделирование эволюции трофически замкнутых сообществ с компенсаторным и некомпенсаторным метаболизмом // Информ. вестник ВОГиС. 2009. Т. 13. № 1. С. 150–158.

Лашин С.А., Матушкин Ю.Г., Суслов В.В., Колчанов Н.А. Эволюционные тренды в системах «Прокариотическое сообщество» и «Прокариотическое сообщество–фаг» // Генетика. 2011. Т. 47. № 12. С. 1676–1685.

Lashin S.A., Matushkin Yu.G. Haploid evolutionary constructor: new features and further challenges // In Silico. Biol. 2012. V. 11. No. 3. P. 125–135.

Lashin S.A., Matushkin Yu.G., Suslov V.V., Kolchanov N.A. Computer modeling of genome complexity variation trends in prokaryotic communities under varying habitat conditions // Ecol. Modelling. 2012. V. 224. No. 1. P. 124–129.

Wang G., Post W.M. A theoretical reassessment of microbial maintenance and implications for microbial ecology modeling // FEMS Microbiol. Ecol. 2012. Sep;81(3):610-7. doi: 10.1111/j.1574-6941.2012.01389.x. Epub 2012 Apr 30. (ссылка в пабмед: http://www.ncbi.nlm.nih.gov/pubmed/22500928).

HIGH-THROUGHPUT SIMULATIONS OF PROKARYOTIC COMMUNITY EVOLUTION WITH HAPLOID EVOLUTIONARY CONSTRUCTOR

Z.S. Mustafin[1, 2], Yu. G. Matushkin[1, 2], S.A. Lashin[1, 2]

[1] Institute of Cytology and Genetics SB RAS, Novosibirsk, Russia, e-mail: Zidane-7@yandex.ru;
[2] Novosibirsk National Research State University, Novosibirsk, Russia

Summary

The results of the development of a high-throughput version of the software package Haploid Evolutionary Constructor (HEC), available at http://evol-constructor.bionet.nsc.ru, are presented. The software is used to simulate the functioning and evolution of prokaryotic communities. A parallel version of the software package was created using the MPI technology. The test was performed on a cluster of the Bioinformatics shared access center. The acceleration obtained was almost linear. The simulation time of complex bacterial communities was reduced from dozens of hours to several minutes.

Key words: microbial communities, optimization, parallel computing, modeling, evolution.