

# РАЗРАБОТКА МЕТОДОВ И ЯЗЫКОВЫХ СРЕДСТВ СБОРОЧНОГО ПРОГРАММИРОВАНИЯ ДЛЯ РАСПРЕДЕЛЕННЫХ ВЫЧИСЛИТЕЛЬНЫХ СРЕД

*И.А. Сидоров, Е.И. Поздняк*

*Институт динамики систем и теории управления СО РАН, Иркутск*

Рост потребности в высокопроизводительных вычислительных системах для организации ресурсоемких научно-исследовательских экспериментов повысил качественные требования к системному и прикладному программному обеспечению таких систем. Однако обилие технологий для организации высокопроизводительных вычислений не уменьшает числа проблем при использовании различных программно-аппаратных архитектур и требует от предметного специалиста все больших навыков технического характера для их эффективного использования. Кроме того различные подходы к организации вычислений, используемые методологии параллельного программирования, а также методы оптимизации программ порой требуют существенной модификации реализующих алгоритмов, что зачастую является труднорешаемой задачей.

Однако существует широкий класс ресурсоемких задач, для решения которых не требуется существенной модификации реализующих алгоритмов и их адаптации к применению на высокопроизводительных вычислительных системах. Такие задачи могут характеризоваться необходимостью проведения многовариантных расчетов над полем независимых между собой входных данных различными программами для их обработки. Одним из подходов к организации такого рода вычислений является построение программ на основе парадигмы сборочного программирования [1].

При таком подходе к созданию программ для параллельных и распределенных вычислительных сред основные усилия будут направлены не на непосредственное кодирование, а на подбор, настройку и организацию совместного существования совокупности позаимствованных (возможно, из различных источников) многократно используемых компонентов. При этом эффективность от использования высокопроизводительных ресурсов может достигаться за счет применения концепции крупноблочного параллелизма [2].

Для создания программ такого рода и их применения в распределенных вычислительных средах авторами разрабатывается программный комплекс DISCOMP [3]. Вычислительные модули, составляющие функциональное наполнение, представляют собой исполняемые программы, которые могут быть реализованы на различных языках программирования (например, С, Fortran, Pascal и др.) и быть платформо-зависимыми. Удаленный запуск модулей, обмен данными между модулями через файлы и мониторинг узлов ВС реализуются средствами системной части ИК DISCOMP. Языковые средства предоставляют возможности для описания концептуальной модели предметной области, ее объектов и основных свойств, а также непосредственно каркаса распределенной программы. В общем случае, благодаря гибкости и универсальности языковых средств, становится возможным расширить класс допустимых задач, более эффективно использовать ресурсы РВС и повысить надежность вычислений.

В данной работе описываются методы и языковые средства сборочного программирования для распределенных вычислительных сред, в основе информационного ядра

которых используется простая вычислительная модель, а в качестве управляющей структуры - сеть Петри. Приводятся примеры условных и циклических конструкций, конструкций fork/join, методов wait и continue, многометодных технологий и др.

### **Литература**

1. Горбунов-Посадов М. Расширяемые программы. М.: Полиптих, 1999. 336 с.
2. Малышкин В.Э. Мощности вычислительных систем прирастают с параллелизмом // Наука в Сибири. 2001. № 32-33. 3 с.
3. Сидоров И.А., Опарин Г.А., Феоктистов А.Г. Технология организации распределенных вычислений в инструментальном комплексе DISCOMP // Современные технологии. Системный анализ. Моделирование. – 2009. – № 2. – С. 175-180.

# ИССЛЕДОВАНИЕ ЭФФЕКТИВНОСТИ ПАРАЛЛЕЛЬНОГО ДВУХУРОВНЕВОГО ИНДЕКСНОГО АЛГОРИТМА ГЛОБАЛЬНОЙ ОПТИМИЗАЦИИ

С.В. Сидоров

Нижегородский госуниверситет им. Н.И. Лобачевского

Данная работа продолжает развитие параллельных алгоритмов минимизации многоэкстремальных функций при невыпуклых ограничениях, построенных на базе индексного метода. Редукция многомерных задач к одномерным основана на приближенном фрактальном отображении единичного отрезка вещественной оси на гиперкуб (развертки типа кривых Пеано). В одном из параллельных алгоритмов используется схема построения множества кривых Пеано («вращаемые развертки»), которую можно эффективно применять при решении задачи на кластере с десятками и сотнями процессоров. Другой подход к распараллеливанию ориентирован на вычислительные узлы с общей памятью. Двухуровневый параллельный индексный алгоритм глобального поиска объединяет эти подходы, учитывая иерархическую структуру современных высокопроизводительных кластерных систем.

## 1. Постановка задачи глобальной оптимизации

Рассмотрим задачу глобальной оптимизации вида

$$\begin{aligned} \varphi^* = \varphi(y^*) = \min \{ \varphi(y) : y \in D, g_j(y) \leq 0, 1 \leq j \leq m \}, \\ D = \{ y \in \mathbb{R}^N : a_i \leq y_i \leq b_i, 1 \leq i \leq N \}, \end{aligned} \quad (1)$$

где целевая функция  $\varphi(y)$  (в дальнейшем обозначаемая также  $g_{m+1}(y)$ ) и левые части ограничений  $g_j(y), 1 \leq j \leq m$ , удовлетворяют условию Липшица с соответствующими константами  $L_j, 1 \leq j \leq m+1$ , а именно

$$|g_j(y_1) - g_j(y_2)| \leq L_j |y_1 - y_2|, 1 \leq j \leq m+1, y_1, y_2 \in D.$$

Используя кривые типа развертки Пеано  $y(x)$ , однозначно отображающие отрезок  $[0, 1]$  на  $N$ -мерный гиперкуб  $P$

$$P = \{ y \in \mathbb{R}^N : -2^{-1} \leq y_i \leq 2^{-1}, 1 \leq i \leq N \} = \{ y(x) : 0 \leq x \leq 1 \},$$

исходную задачу можно редуцировать к следующей одномерной задаче:

$$\varphi(y(x^*)) = \min \{ \varphi(y(x)) : x \in [0, 1], g_j(y(x)) \leq 0, 1 \leq j \leq m \}. \quad (2)$$

Рассматриваемая схема редукции размерности сопоставляет многомерной задаче с липшицевой минимизируемой функцией и липшицевыми ограничениями одномерную задачу, в которой соответствующие функции удовлетворяют равномерному условию Гельдера (см. [2]), т.е.

$$\begin{aligned} |g_j(y(x')) - g_j(y(x''))| \leq K_j |x' - x''|^{1/N}, x', x'' \in [0, 1], \\ 1 \leq j \leq m+1, \end{aligned}$$

где  $N$  есть размерность исходной многомерной задачи, а коэффициенты  $K_j$  связаны с константами Липшица  $L_j$  исходной задачи соотношениями  $K_j \leq 4L_j \sqrt{N}$ .

Различные варианты индексного алгоритма для решения одномерных задач и соответствующая теория сходимости представлены в работах [1, 4, 7].

## 2. Вращаемые развертки и параллельный метод для кластера

Редукция многомерных задач к одномерным с помощью разверток имеет такие важные свойства, как непрерывность и сохранение равномерной ограниченности разностей функций при ограниченности вариации аргумента. Однако при этом происходит потеря части информации о близости точек в многомерном пространстве, так как точка  $x \in [0, 1]$  имеет лишь левых и правых соседей, а соответствующая ей точка  $y(x) \in \mathbb{R}^N$  имеет соседей по  $2^N$  направлениям. А при использовании отображений типа кривой Пеано близким в  $N$ -мерном пространстве образам  $y', y''$  могут соответствовать достаточно далекие прообразы  $x', x''$  на отрезке  $[0, 1]$ . Как результат, единственной точке глобального минимума в многомерной задаче соответствует несколько (не более  $2^N$ ) локальных экстремумов в одномерной задаче, что, естественно, ухудшает свойства одномерной задачи.

Сохранить часть информации о близости точек позволяет использование множества отображений

$$Y_L(x) = \{y^1(x), \dots, y^L(x)\} \quad (3)$$

вместо применения единственной кривой Пеано  $y(x)$  (см. [3, 5]). Каждая кривая Пеано  $y^i(x)$  из  $Y_L(x)$  может быть получена в результате поворота исходной кривой. При этом найдется отображение  $y^i(x)$ , которое точкам многомерного пространства  $y', y''$ , которым при исходном отображении соответствовали достаточно далекие прообразы на отрезке  $[0, 1]$ , будет сопоставлять более близкие прообразы  $x', x''$ .

Максимальное число различных поворотов развертки, отображающей  $N$ -мерный гиперкуб на одномерный отрезок, составляет  $2^N$ . Использование всех из них является избыточным, требуется выбрать лишь часть из всех возможных вариантов. В предложенной схеме преобразование развертки осуществляется в виде поворота на угол  $\pm\pi/2$  в каждой из координатных плоскостей. Число подобных пар поворотов определяется числом координатных плоскостей пространства, которое равно  $C_N^2 = \frac{N(N-1)}{2}$ , а общее

число преобразований будет равно  $N(N-1)$ . Учитывая исходное отображение, приходим к заключению, что данный способ позволяет строить до  $N(N-1)+1$  развертки для отображения  $N$ -мерной области на соответствующие одномерные отрезки. В случае необходимости данный способ построения множества отображений может быть легко «отмасштабирован» для получения большего (вплоть до  $2^N$ ) числа разверток.

Использование множественных отображений (3) позволяет решать задачу (1) путем параллельного решения  $L$  задач вида (2) на наборе отрезков  $[0, 1]$ . Каждая одномерная задача решается на отдельном процессоре с использованием развертки  $y^s$ ,  $1 \leq s \leq L$ . Результаты испытания в точке  $x^k$ , полученные конкретным процессором для решаемой им задачи, интерпретируются как результаты испытаний во всех остальных задачах (в соответствующих точках  $x^{k1}, \dots, x^{kL}$ ) и рассылаются другим процессорам. При таком подходе испытание в точке  $x^k \in [0, 1]$ , осуществляемое в  $s$ -й задаче, состоит в последовательности действий:

1. Определить образ  $y^k = y^s(x^k)$  при соответствии  $y^s(x)$ .
2. Проинформировать остальные процессоры о начале проведения испытания в точке  $y^k$  (блокирование точки  $y^k$ ).
3. Вычислить величины  $g_1(y^k), \dots, g_v(y^k)$ , где значения индекса  $v \leq m$  определяются условиями

$$g_j(y^k) \leq 0, 1 \leq j < v, g_v(y^k) > 0, v \leq m.$$

Выявление первого нарушенного ограничения прерывает испытание в точке  $y^k$ . В случае, когда точка  $y^k$  допустима, испытание включает вычисление значений всех

функционалов задачи, при этом значение индекса принимается равным величине  $v=m+1$ . Тройка

$$y_s(x^k), v=v(x^k), z^k=g_v(y^s(x^k)) \quad (4)$$

является результатом испытания в точке  $x^k$ .

4. Определить прообразы  $x^{kl} \in [0, 1]$ ,  $1 \leq l \leq L$ , точки  $y^k$ , и интерпретировать испытание, проведенное в точке  $y^k \in D$ , как проведение испытаний в  $L$  точках

$$x^{k1}, \dots, x^{kl}, \quad (5)$$

с одинаковыми результатами

$$\begin{aligned} v(x^{k1}) &= \dots = v(x^{kl}) = v(x^k), \\ g_v(y^1(x^{k1})) &= \dots = g_v(y^L(x^{kl})) = z^k. \end{aligned}$$

Проинформировать остальные процессоры о результатах испытания в точке  $y^k$ , разослав им тройки  $(y^k, v, z^k)$ .

Каждый процессор имеет свою копию программных средств, реализующих вычисление функций задачи, и решающее правило алгоритма, а также собственное хранилище поисковой информации. Для организации взаимодействия на каждом процессоре создается очередь, в которую процессоры помещают информацию о выполненных итерациях в виде троек: точка очередной итерации, индекс и значение из (4), причем индекс заблокированной точки полагается равным  $-1$ , а значение функции в ней не определено.

### 3. Параллельный метод для вычислительного узла с общей памятью

Второй подход к распараллеливанию эксплуатирует характеристическую схему индексного алгоритма. В редуцированной одномерной задаче на очередной итерации для каждого интервала  $(x_{i-1}, x_i)$ ,  $1 \leq i \leq k+1$ , вычисляется характеристика  $R(i)$ , где

$$R(i) = \begin{cases} \Delta_i + \frac{(z_i - z_{i-1})^2}{r_v^2 \mu_v^2 \Delta_i} - 2 \frac{(z_i + z_{i-1} - 2z_v^*)}{r_v \mu_v}, & v=v(x_{i-1})=v(x_i) \\ 2\Delta_i - 4 \frac{(z_i - z_v^*)}{r_v \mu_v}, & v=v(x_i) > v(x_{i-1}) \\ 2\Delta_i - 4 \frac{(z_{i-1} - z_v^*)}{r_v \mu_v(l, \tau)}, & v=v(x_{i-1}) > v(x_i) \end{cases}$$

$$\Delta_i = (x_i - x_{i-1})^{1/N}.$$

Величины  $r_v > 1$ ,  $1 \leq v \leq m+1$ , являются параметрами алгоритма.

Далее определяется  $s$  ( $s$  – как правило, число вычислительных ядер на узле кластера) интервалов  $(x_t, x_{t+1})_k$ ,  $1 \leq k \leq s$ , которым соответствует  $s$  наибольших характеристик

$$R_k(t) = \max\{R(i): 1 \leq i \leq k+1\}, 1 \leq k \leq s. \quad (6)$$

На  $s$  вычислительных ядрах параллельно проводится очередное испытание в средней точке интервала  $(x_t, x_{t+1})_k$ ,  $1 \leq k \leq s$ , если индексы его конечных точек не совпадают, т.е.

$$x^{q+1} = \frac{(x_t + x_{t+1})_k}{2}, \quad v(x_{t-1}) \neq v(x_t), 1 \leq k \leq s.$$

В противном случае провести испытание в точке

$$x^{q+1} = \frac{(x_t + x_{t+1})_k}{2} - \text{sign}(z_t - z_{t-1})_k \frac{1}{2r_v} \left[ \frac{|z_t - z_{t-1}|_k}{\mu_v} \right]^N,$$

$$v(x_{t-1}) = v(x_t), 1 \leq k \leq s.$$

Результаты испытания заносятся в общее хранилище поисковой информации.

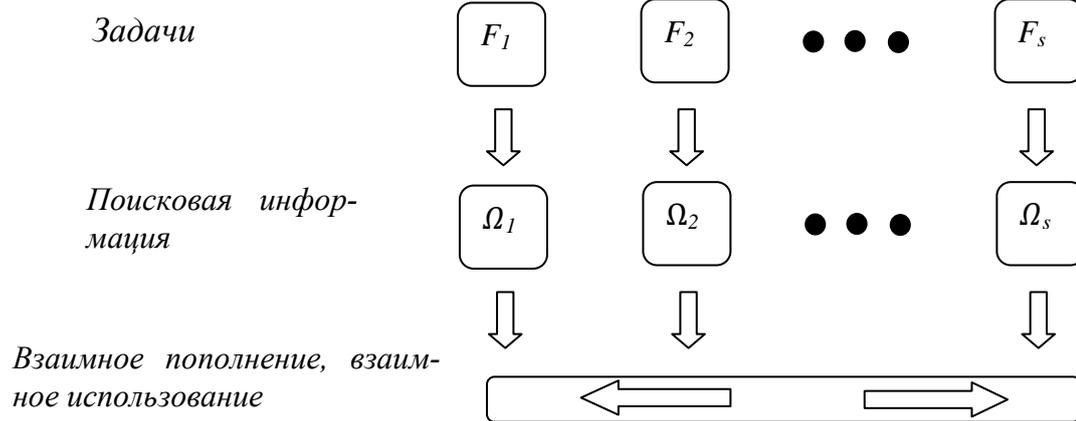


Рис 1. Информационная связанность подзадач оптимизации на узле с общей памятью

#### 4. Двухуровневый параллельный алгоритм

Объединение двух описанных подходов позволяет получить гибридный алгоритм, эффективно использующий распределение поисковой информации по узлам кластера и общую память в рамках одного узла.

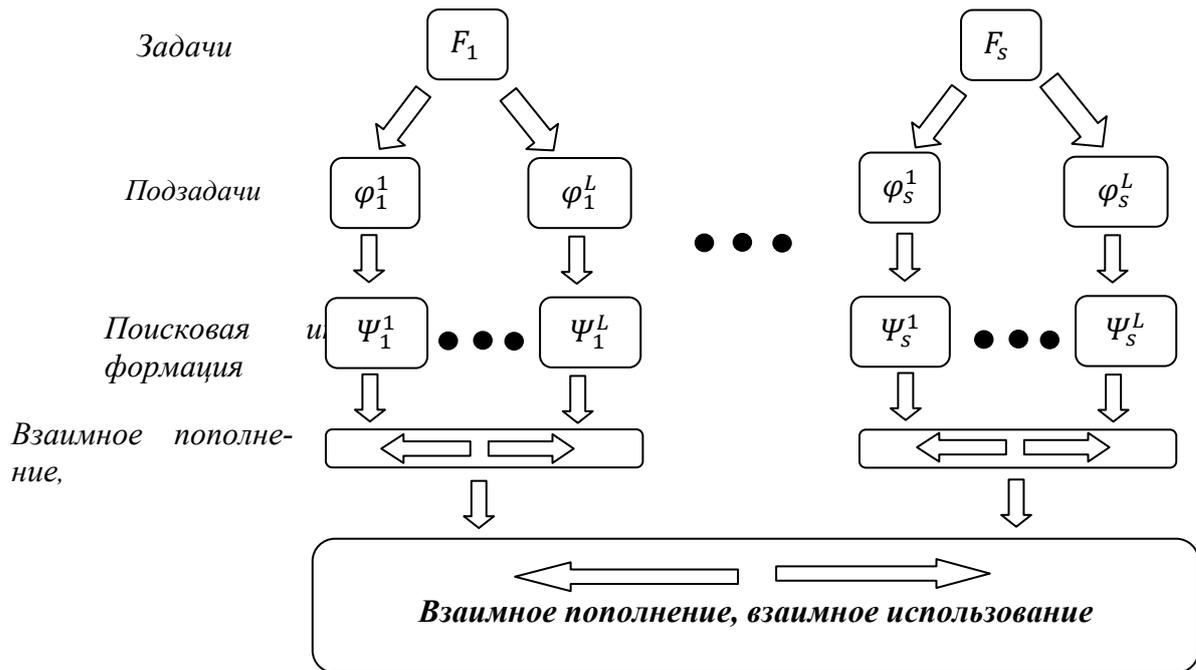


Рис 2. 2-х уровневая модель информационная связанности подзадач

#### Заключение

Данная работа исследует двухуровневую схему параллельных вычислений, позволяющую исключить передачу данных на вычислительном узле с общей памятью. Вся вычислительная система асинхронно решает информационно связанные подзадачи для каждой разверки (по одной на узел кластера). Проведено исследование эффективности данного алгоритма. Результаты вычислений приводятся в докладе.

Работа выполнена при поддержке Минобрнауки России (соглашение 14.В37.21.0878).

## Литература

1. Стронгин Р.Г. Численные методы в многоэкстремальных задачах. (Информационно-статистические алгоритмы). М.: Наука, 1978.
2. Стронгин Р.Г. Поиск глобального оптимума. М.: Знание, 1990.
3. Стронгин Р.Г. Параллельная многоэкстремальная оптимизация с использованием множества разверток // Ж. вычисл. матем. и матем. физ. 1991, Т.31, №8. С. 1173–1185.
4. Стронгин Р.Г., Баркалов К.А. О сходимости индексного алгоритма в задачах условной оптимизации с  $\varepsilon$ -резервированными решениями // Математические вопросы кибернетики. М.: Наука, 1999. С. 273 – 288.
5. Strongin R.G., Sergeyev Ya.D. Global optimization with non-convex constraints. Sequential and parallel algorithms. Kluwer Academic Publishers, Dordrecht, 2000.
6. Баркалов К.А., Стронгин Р.Г. Метод глобальной оптимизации с адаптивным порядком проверки ограничений // Ж. вычисл. матем. и матем. физ. 2002, Т.42, №9. С. 1338–1350.
7. Баркалов К.А. Ускорение сходимости в задачах условной глобальной оптимизации. Нижний Новгород: Изд-во Нижегородского гос. ун-та, 2005.
8. Баркалов К.А., Рябов В.В., Сидоров С.В. Использование кривых Пеано в параллельной глобальной оптимизации // Материалы Девятой международной конференции-семинара «Высокопроизводительные параллельные вычисления на кластерных системах», Владимир, 2009. С. 44–47.
9. Стронгин Р.Г., Гергель В.П., Баркалов К.А. Параллельные методы решения задач глобальной оптимизации // Известия вузов. Приборостроение. – 2009, Т. 52, № 10. С. 25–32.
10. Баркалов К.А., Рябов В.В., Сидоров С.В. Масштабируемые параллельные алгоритмы глобальной оптимизации со смешанной локально-глобальной стратегией // Материалы международной научной конференции «Параллельные вычислительные технологии 2010», Уфа, 2010. С. 402–409.
11. Стронгин Р.Г., Маркин Д.Л. Минимизация многоэкстремальных функций при невыпуклых ограничениях // Кибернетика. 1986. №4.
12. Törn A. and Zilinskas A. Global Optimization. Lecture Notes in Computer Science, No 350, Springer-Verlag, Berlin, 1989.

## **СИГНАЛЬНЫЕ ПРОЦЕССЫ В НЕЙРОННЫХ СЕТЯХ МОЗГА. ВИРТУАЛЬНОЕ МОДЕЛИРОВАНИЕ ДЛЯ ЗАДАЧ ДИАГНОСТИКИ ПАТОЛОГИЙ И ТЕСТИРОВАНИЯ ЛЕКАРСТВ**

***А.Ю. Симонов<sup>1</sup>, В.И. Миронов<sup>1</sup>, И.С. Прокин<sup>1</sup>, С.Ю. Гордлеева<sup>2</sup>,  
И.В. Мухина<sup>3</sup>, В.Б. Казанцев<sup>2</sup>***

*<sup>1</sup>Нижегородский госуниверситет им. Н.И. Лобачевского*

*<sup>2</sup>Институт прикладной физики РАН, Нижний Новгород*

*<sup>3</sup>Нижегородская государственная медицинская академия*

Моделирование процессов генерации и распространения электрических и химических сигналов в нейрональных сетях мозга является в настоящее время одним из актуальных приложений современных суперкомпьютерных технологий. В работе мы рассматриваем различные аспекты такого моделирования. В частности, процессы биоэлектрической активности диссоциированных культур нейронов гиппокампа могут быть эффективно исследованы в виртуальных моделях спайковых нейронных сетей, связанных между собой посредством синаптических контактов. Объединяемые в большие модельные сети, виртуальные клетки способны демонстрировать эффекты сетевой сигнализации, наблюдаемые экспериментально. Здесь принципиальными моментами являются исследование механизмов генерации и распространения характерных паттернов нейросетевой активности, параметрических условий существования ключевых режимов генерации, механизмов переключения между такими режимами и соответствующих им структурно-функциональных перестроек.

Другая задача связана с компьютерной реконструкцией роста нейрональной сети согласно уравнениям диффузионного транспорта строительных белков и навигации конуса роста дендритных отростков и аксона согласно распределению факторов роста. Ветвление конуса роста описывается в рамках стохастической феноменологической модели. Процесс формирования отростка является зависимым от многих факторов, имеющих как внутриклеточную природу, так и поступающих из межклеточного пространства, которые и являются причиной столь широкого разнообразия типов нейрональных клеток. Процесс удлинения отростка зависит от величины концентрации активных веществ (белков роста), продуцируемых в соме клетки. Пространственное ориентирование отростка осуществляется посредством специальных сигнальных молекул (факторов роста), продуцируемых нейронами, которые, диффундируя сквозь межклеточное пространство, формируют управляющее поле, воздействующее на конус роста нейрита. Применяя данную модель можно конструировать виртуальные сети нейронов с реалистичной морфологией и тестировать различные условия развития нейронных систем в онтогенезе.

Другой тип клеток в мозге, астроциты, а также внеклеточный матрикс мозга, составляющий перинеурональные сети тоже могут принимать активное участие в передаче сигналов между нейронами и влиять на процессы обработки информации и формирования высших когнитивных функций. Например, экспериментально установлено, что кальциевые сигналы в астроцитах связаны с выбросом во внеклеточное пространство особых химических веществ, которые могут воздействовать на пре- и постсинаптические рецепторы, изменяя их функции. Данное направление также является перспектив-

ным в части математического моделирования и создания виртуальных моделей астроцитарных и астроцит-нейронных сетей.

Важной задачей здесь, в том числе и с точки зрения приложений параллельных вычислений, является обработка больших объёмов экспериментальных данных. Примером таких данных являются многоканальные записи электрической активности диссоциированных культур нейронов гиппокампа, выращиваемых на мультиэлектродных зондах. Сигнал с такой культуры регистрируется множеством планарных электродов, расположенных на дне пробирки, в которой выращиваются нейроны. С увеличением пространственного разрешения такой технологии возрастает потребность в вычислительных мощностях, необходимых для обработки и анализа данных. Помимо этого существует ряд задач, в которых требуется обработка потока многоканальных данных в режиме реального времени, например, при формировании моторных команд для управления внешним электронно-механическим устройством (роботом). Такие культуры являются одной из наиболее перспективных экспериментальных моделей для исследования патологических процессов, лежащих в основе многих нейродегенеративных заболеваний. Кроме того данный подход может быть использован для задач тестирования воздействия лекарственных препаратов на функциональное состояние нервной ткани.

В докладе будут изложены основные современные представления в перечисленных областях науки, представлены недавние результаты группы соавторов и сформулированы наиболее интересные и перспективные задачи в этих областях для привлечения высокопроизводительных ресурсов современных суперкомпьютерных технологий.