ХАРАКТЕРИСТИКИ ПЕРСПЕКТИВНЫХ ПРИНЦИПИАЛЬНО НОВЫХ КОМПЬЮТЕРНЫХ УСТРОЙСТВ И СИСТЕМ

Граничин Олег Николаевич

доктор физ.-мат. наук, профессор

Санкт-Петербургский государственный университет (СПбГУ)

Обсуждается возможные характеристики перспективных принципиально новых компьютерных устройств и систем. Новый компьютер представлен как устройство, состоящее из набора асинхронных динамических моделей (функциональных элементов). Основные черты: стохастичность, гибридность, асинхронность, отсутствие жесткой централизации, динамическая кластеризация на классы связанных моделей. Особое внимание уделено “гипотетической” модели искусственного интеллекта и реализации “за такт” оценки вектора-градиента многопараметрической функции на вычислительном устройстве типа квантового компьютера.

1. Введение

Взглянем на историю развития средств вычислительной техники. За шесть десятилетий пройден путь от ламп, через транзисторы, интегральные микросхемы к сверхбольшим интегральным микросхемам. Что будет дальше? Основной вопрос – как обрабатывать данные? Для людей старшего поколения знакомство с компьютерами начиналось с фантастических романов. Еще школьниками, не видя никогда настоящих компьютеров, многие из них были уверены, что скоро появится «Искусственный Разум», освободив нас от многих рутинных забот. Наверное поэтому люди старшего поколения, занятые в развитии информационных технологий (ИТ), острее чувствуют тенденции кардинальных преобразований. Конечно, кто-то возразит, что многие не дождавшись за 60 лет «искусственного разума», разочаровались в перспективах, тем более, что по мере развития ИТ кроме

такой фантастической цели появилось огромное количество простых и сложных конкретных задач.

Новые потребности, глобализация задач, экспоненциальное возрастание сложности вычислительных систем и наметившаяся в последнее время тенденция по преодолению отставания отечественной ИТ отрасли в развитии суперкомпьютерных вычислений (Т-Платформы, СКИФ-Аврора и др. проекты) заставляют уже в практическом плане задуматься о перспективах и возможной смене парадигмы: «Что такое процесс вычислений?» Объективные сегодняшние тенденции – миниатюризация и повышение производительности процессоров, как это и было предсказано законом Мура – приводят технологии к порогу развития традиционных вычислительных устройств. От приоритетов бесконечного наращивания тактовой частоты и мощности одного процессора производители переходят к многоядерности, параллелизму и т. п.

Сейчас несколько ядер в процессоре переносного компьютера уже норма, в процессорах суперкомпьютеров ядер уже намного больше. «Джин уже выпущен из бутылки», пройдет совсем немного времени и ядер станет несколько десятков, а потом и тысяч. Появятся совершенно другие архитектуры, ядра будут объединяться в сложные блоки, к данным можно будет получать параллельный одновременный доступ разным вычислительным блокам, “общение” вычислительных блоков между собой будет происходить через общую память. В действительности, изменятся многие аспекты парадигмы: что такое вычислительное устройство и что такое вычислительный процесс. Изменятся традиционные представления о том, как устроен компьютер, что такое вычислительная система. Эти процессы принесут изменения и в стиль программирования, и в то, как будут использоваться вычислительные устройства.

Статья организована следующим образом. В разделе 2 описываются возможные характеристики новой парадигмы вычислений. В разделе 3 представлена абстрактная схема обобщенной Машины Тьюринга. В разделе 4

описана “гипотетическая” модель искусственного интеллекта. В разделе 5 детально разобран новый рандомизированный алгоритм стохастической оптимизации, позволяющий наиболее эффективно решать задачи минимизации многопараметрических функционалов типа среднего риска при условии их выпуклости, а также описана реализация “за такт” его итерации на вычислительном устройстве типа квантового компьютера. В заключении подведены итоги.

2. Новая парадигма вычислений

На прошедшей в сентябре 2010 года в Абрау-Дюрсо всеросийской научной конференции «Научный сервис в сети Интернет: суперкомпьютерные центры и задачи» (20-25 сентября 2010 г., г. Новороссийск) во многих докладах ставился вопрос: «Что будет при переходе от сегодняшних производительностей суперкомпьютеров в «TeraFlops» к следующему масштабу «ExaFlops»? Вл. В. Воеводин (2010) говорил: «Переход к «ExaScale», естественно, должен будет затронуть такие важнейшие аспекты вычислительных процессов, как: *модели программирования, степень и уровни параллельности, неоднородность программных и аппаратных систем, сложность иерархии памяти и трудности одновременного доступа к ней в распределенных вычислениях, стек системного и прикладного ПО, надежность, энергопотребление, сверхпараллельный ввод/вывод … »*. Все это неизбежно приведёт к смене парадигмы высокопроизводительных вычислений.

Переход к новой парадигме вычислений приведет, наверное, к тому, что архитектура вычислительных устройств «сдвинется» в сторону «*набора одновременно работающих асинхронных моделей взаимодействующих динамических систем (функциональных элементов)*». Среди новых характерных черт будущей парадигмы все более отчетливо проступают следующие: стохастичность, гибридность, асинхронность, кластерность

(отсутствие жесткой централизации и динамическая кластеризация на классы связанных моделей).

Стохастичность. С одной стороны, хорошо известно, что компьютеры становятся все миниатюрнее и миниатюрнее, размер элементарного вычислительного элемента (вентиля) приближается к размеру молекулы или даже атома. На таком уровне законы классической физики перестают работать и начинают действовать квантовые законы, которые в силу принципа неопределенности Гейзенберга не дают точных ответов о состоянии. С другой стороны, стохастичность – это известное свойство сложных динамических систем, состоящих из огромного числа компонент.

Под *гибридностью* будущих процессов вычислений понимается необходимость рассмотрения комбинации непрерывных и дискретных процессов, т. е. учет непрерывной эволюции протекания физических процессов при работе той или иной модели и скачкообразное переключение с одной модели на другую. Увеличение быстродействия вычислительных устройств и уменьшение их размеров с неизбежностью приводит к необходимости операций с “переходными” процессами, серьёзным ограничением классической модели вычислений является разбиение памяти на изолированные биты, потому как, во-первых, сокращение длины такта и расстояний между битами с определенного уровня делает невозможным рассматривать их изолированно в силу законов квантовой механики. Вместо примитивных операций с классическими битами в будущем было бы естественно перейти операциям, задаваемыми теми или иными динамическими моделями микромира, оперирующими с наборами взаимосвязанных “битов”. При этом могут остаться классические операции с битами, для которых простейшими “моделями” являются процессы перехода физической системы (триггера) из состояния “1” в “0” и наоборот. Обоснованием целесообразности рассмотрения более широкого класса моделей являются успехи в разработках для традиционных сложных многомерных задач новых алгоритмов, работающих “за такт”, в которых

результат получается как итог физического адиабатического процесса. Например, П. Шор (1997) предложил алгоритм квантового преобразования Фурье, которое может выполняться за время пропорциональное $(\log N)^2$, а не за $N \log N$ как классическое быстрое преобразование Фурье. В работе Д. Тиена (2003) обсуждается опирающийся на квантовую адиабатическую теорему гипотетически возможный «физический» способ решения за конечное время 10-ой проблемы Гильберта, в работе С. Сысоева и др. (2006) предложен эффективный квантовый алгоритм вычисления “за такт” оценки вектора-градиента многопараметрической функции, задаваемой с большой степенью неопределенностей (см. раздел 5). Типичные для математических алгоритмов операции типа “свертки” функций вполне могут обнаружиться “в природе”. Последние исследования похожих моделей показывают, что их выполнение за счет присущей природе способности к самоорганизации не обязательно “раскладывается” на более простые кирпичики, т. е. не всегда может быть записано в виде классического алгоритма. Один из возможных примеров “аналоговой” реализации свертки функций на большом регулярном массиве квантовых точек с характерными размерами до 2 нм представлен автором совместно с С. Молодцовым (2006).

Асинхронность. Отказ от унифицированных простых вычислительных элементов неизбежно приводит к отказу от синхронизации работы различных компонент, имеющих существенно отличающиеся физические характеристики и свои длительности “тактов”. В рамках классической теории множеств противоречивый смысл понятия единого “такта” выражается в рамках неразрешимости проблемы континуума в рамках аксиоматики Френкеля-Цермело.

Кластерность. Одним из неожиданных результатов многочисленных попыток в разработках (создании, адекватном описании поведения и управлении) сложных стохастических систем оказалась перспективность модели мультиагентных систем, в которой топология связей агентов между собой меняется со временем. При этом понятию агента может

соответствовать как некоторая динамическая модель (компонент системы), так и определенный набор моделей. При отсутствии жесткой централизации такие системы способны эффективно решать достаточно сложные задачи, разбивая их на части и автономно перераспределяя ресурсы на “нижнем” уровне, эффективность часто повышается за счет самоорганизации агентов и динамической кластеризация на классы связанных моделей.

В некотором смысле переход к разработке и созданию моделей сложных вычислений – закономерный этап развития микропрограммирования. Понимание возможных преимуществ работы устройств с эффективным микрокодом отмечалось еще в 50-е гг. прошлого века. Во втором ряду серии ЕС ЭВМ в конструкцию была заложена возможность динамического микропрограммирования. Хорошо известен целый ряд ЭВМ от “Мир-1”' до “Эльбрус”, использующих языки высокого уровня (HLL) как машинные. В настоящее время *технологии микропрограммирования естественным образом сдвигаются в сторону физических процессов*.

3. Обобщение классической схемы Машины Тьюринга

Автором (Граничин, Молодцов, 2006, Васильев, Граничин, 2010, Granichin, Vasil'ev, 2010) предложена и обоснована новая абстракция вычислительного устройства, обобщающая схему классической машины Тьюринга. В рамках новой модели переосмысливаются ставшие традиционными понятия “такта”, “памяти”, “ленты”, “программы” и “состояния”. Если в классическом подходе ячейки памяти используются для хранения дискретной информации и их изменение возможно только в тех случаях, когда указатель ленты показывает на нее, то в новой концепции “ячейка памяти” представляет собой постоянно функционирующую модель какой-то динамической системы (возможно и достаточно сложной), а “лента” — подмножество в общем пространстве состояний, при достижении которого заканчивается очередной “такт” и происходит включение “программы” — скачкообразное переключение с одних моделей на другие.

Указатель “считывающей/записывающей головки” — бинарная строка, задающая список “включенных” моделей, вместе с матрицей “видимости”. Естественно, что классическая ячейка памяти для хранения “бита” является частным случаем такого обобщения.

В любых современных вычислительных устройствах хранение информации основано на тех или иных физических принципах, только эволюция состояния ячейки во время хранения более простая — сохраняется постоянной какая-то физическая характеристика. Другими словами, традиционную информатику можно сравнить с арифметикой, она оперирует цифрами — значениями ячеек памяти (или, более точно — арифметикой конечных двоичных дробей). Предлагаемая новая модель вычислений позволит перейти в информатике от “арифметики” к “функциональному анализу”, исследующему процессы эволюции информации внутри новых “ячеек” (операции с функциями).

Опишем математическую модель, соответствующую новой парадигме вычислительного устройства, рассматривая процесс вычисления как совокупность работающих параллельно динамических систем с пересекающимися частями фазовых пространств (общей памятью). Будем считать, что вычислительная система состоит из счётного набора аналоговых компонент $\{M_i\}_{i=1}^{\infty}$, таких что каждая компонента M_i определяет динамику некоторых компонент x_i общей памяти x (имеющей некоторую гетерогенную природу) по правилу $\dot{x}_i(t) = H_i(x_i, t)$, где $x_i(t) \in \mathrm{X}_i, \mathrm{X}_i \subset \mathrm{X}$.

Пусть в каждый момент времени t лишь небольшая часть моделей (компонент вычислительного устройства) оказывает влияние на изменение состояния памяти всей системы. Другими словами, в каждый момент времени t можно определить бесконечную строку битов $S(t) = \{s_i(t)\}_{i=1}^{\infty}$, состоящую из конечного числа единиц: $s_i(t) \in \{0,1\}$, для $i \le i_{\max}(t)$ и $s_i(t) = 0$, для $i > i_{\max}(t)$. Тогда общая динамика системы описывается как $\dot{x}(t) = \sum_{i=1}^{\infty} s_i(t) H_i(x_i, t)$, где $x(t) \in \mathrm{X}$ и $x_i(t) \in \mathrm{X}_i$.

Множество всевозможных строк $S(t)$ обозначим Σ.

Не умаляя общности, можно считать, что $S(t)$ — кусочно-постоянная функция времени, т. е. на определенных интервалах те или иные модели (компоненты ВУ) «включены». Т. о. получаем гибридную динамическую систему в традиционном понимании этого термина. Общая динамика системы на k-м интервале постоянства $S(t)$ от t_k до t_{k+1} (со значением S_k) описывается как:

$$\dot{x}_{S_k}(t) = \sum_{i=1}^{i_{\max}(t_k)} s_i(t_k) H_i(x_i, t) = \sum_{i \in \sup p(S_k)} H_i(x_i, t), t \in [t_k, t_{k+1}),$$

где $x_{S_k}(t) \in \overline{\mathrm{X}}_{S_k} = \bigcup_{i \in \sup p(S_k)} \mathrm{X}_i \subseteq \mathrm{X}$.

Для полного описания динамики всей системы необходимо определить правило изменения $S(t)$. Для описания процессов решения (или моделирования) большого количества практических задач достаточно ограничиться заданием направленного графа переходов, связывающего возможные переходы $S_k \xrightarrow{J_{k,j}} S_j; S_k, S_j \subset \Sigma$ с указанием условий $J_{k,j} \in J$, при выполнении которых происходит переключение изменения динамики всей системы с набора моделей S_k на S_j. Условием останова системы является $S(t) = 0$, которое эквивалентно тому, что ни один из компонентов памяти не является активным.

Таким образом, можно определить, что в новом смысле программа P — это правила изменения $S(t)$. Если правила хранить в памяти ВУ и дать возможность их корректировки со временем, то с точки зрения программирования это в некотором смысле будет соответствовать архитектуре фон Неймана, когда и данные и программа хранятся в одной памяти и программа меняется в процессе работы системы.

Более общая схема получается при задании стохастических правил переходов. При этом детерминированный случай хорошо обобщается, задавая достаточно высокие вероятности переходов для определенных в нем переходов и оставляя равновероятный, почти нулевой, уровень вероятности

для всех остальных. Но даже такая схема может оказаться более эффективной (по аналогии с известным методом отжига для поиска глобального минимума функции). Случайно установленные “новые связи” могут оказаться полезными для системы и можно допустить корректировку вероятностей переходов в определенных ситуациях с целью повышения их значимости. Такой механизм будет приводить к элементам самоорганизации моделей в процессе работы.

Вероятностное задание отображений эволюции и программ позволит реализовывать с помощью новой модели динамические системы, стохастические гибридные системы, вероятностные автоматы, системы со стохастическим управлением и т. п., не описываемые детерминированными законами. Для организации работы такой системы, наверное, целесообразно будет использовать рандомизированные алгоритмы, в которых один или несколько шагов или несколько шагов базируются на случайном выборе оного из многих детермированных правил. Типичными трудностями при обработке информации в распределенных сетях и реальном времени являются ограничения ресурсов и недостаточная вариативность данных наблюдения. Рандомизация как раз и может быть использована для обогащения данных наблюдений, а рандомизированные сценарные подходы позволяют решать задачи “эффективно с высокой вероятностью” для почти всех ограничений, которые часто может быть очень много. Детальное описание истории развития и перспектив рандомизированных алгоритмов сделано автором совместно с Б.Т. Поляком в монографии «Рандомизированные алгоритмы оценивания и оптимизации при почти произвольных помехах» (2003), в которой подчеркиваются их лучшие и интересные черты:

– значительное сокращение числа операций (минимаксная оптимальность среди других алгоритмов);

– устранение влияния на работу системы систематических погрешностей, которые практически неизбежны при изменяющейся со временем модели динамической системы;

– независимость сложности алгоритма от размерности данных.

4. Искусственный интеллект

Вернемся к первому абзацу статьи. За более чем пятидесятилетнюю историю практического использования компьютеров одним из основных разочарований для человечества стала невозможность создания искусственного интеллекта. Несмотря на внушительный прогресс в сфере создания мощных вычислительных устройств, пока еще нельзя говорить о детальном понимании этой задачи или о направлениях исследований, которые могли бы гарантированно дать результат. Большая часть работ по искусственному интеллекту, известных автору, посвящена решению конкретных задач, или реализациям каких-либо эвристических поведенческих алгоритмов, эффективно справляющихся с тестовыми заданиями и терпящих неудачу на задачах реального мира. Многие исследователи согласны с тем, что пока еще не разработана методология создания алгоритмов и устройств для решения задач искусственного интеллекта. Разработка подобной методологии по всей видимости связана с анализом различных задач искусственного интеллекта, выявлением аналогичных частей и созданием новой формальной логики, адаптированной под эти части.

Человечество накопило большой массив задач, которые компьютеры умеют эффективно решать. Можно послать ракету в определенную точку, можно быстро вычислить преобразование Фурье, можно быстро найти решение какого-нибудь важного уравнения и много-много других (см. Рис. 1).

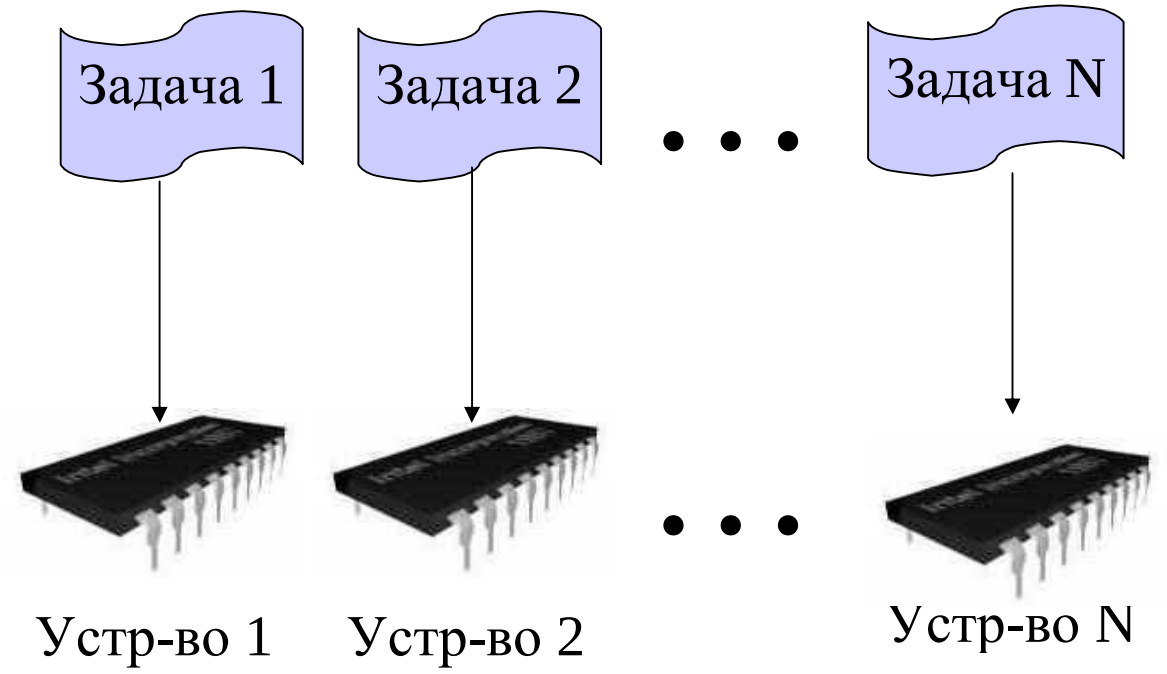
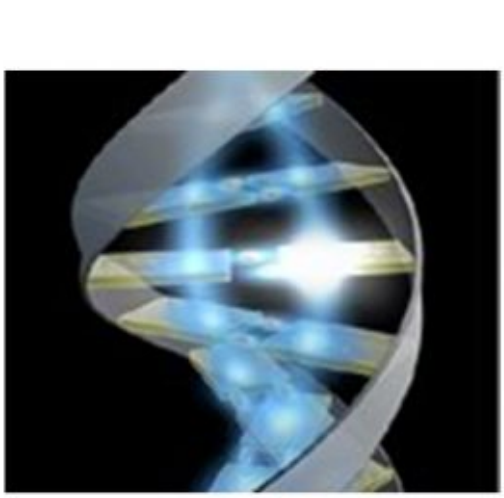


Рис. 1. Набор задач и решений.

Но как нам поступить, если мы хотим собрать универсальный вычислитель, который способен один выполнить их все? Предположим, что мы делаем не вычислитель, быстро решающий уравнения, а наша цель сделать что-то похожее на нас, на людей (искусственное мыслящее существо), которое в условиях неопределенностей способно распознать реальную ситуацию, выбрать адекватную ей задачу и решить ее. Например, среди всего реализованного набора задач принимает решение о выборе блока, ответственного за решение определенной задачи, который «говорит: Да, эту ситуацию я контролирую, это моя цель, пора стрелять или вижу угрозу, пора убегать и т. п.»

Такие системы плохо «вписываются» в традиционную концепцию архитектуры компьютера, в которой операции обычно выполняются последовательно, данные загружаются последовательно, для выполнении того или иного действия надо последовательно пройти некоторые шаги А, Б, В и т. д., как-то их перебрать. Но пока мы их перебираем, зачастую решаемая задача перестает быть актуальной.

Как будет в перспективе? Простое решение собрать все вычислительные блоки вместе в сегодняшних условиях наткнется на проблемы отвода тепла, одновременной доставки информации, выбора ведущего блока и многими другими. Наверное, когда-то мы сможем собрать блоки (микросхемы), решающие выбранные нами задачи (желательно все) в «клубок» закрученной спирали, как в молекулах ДНК у нас в клетках (см. Рис. 2), которые решают огромное количество задач (функций).



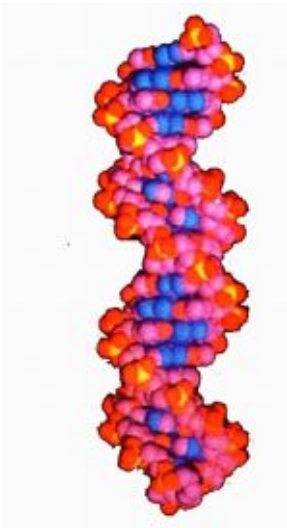


Рис. 2.. Информационный резонанс (слева). Фрагмент молекулы ДНК (справа).

Как могут решаться важные для универсального вычислителя задачи с доступом к памяти, с параллелизмом, данными? Гипотетически можно представить, что химическое или электромагнитное воздействие (например, луч света) на такой «клубок» может одновременно воздействовать сразу на все блоки и каждый из них одновременно с другими «примеряет» поступающую информацию «на себя». Традиционная альтернатива параллелизму – перебирать по очереди и смотреть кому лучше подходит. В том блоке, который распознал адекватность текущей информации его задаче, которому информация подошла лучше остальных, можно представить себе возникновение состояния некоторого *информационного резонанса* (см. Рис. 2а).

Рассмотрим пример интеллектуальной системы, работающей на перечисленных выше принципах. Представим себе робота,

взаимодействующего с реальным миром через стандартные устройства --- датчики, камеры, микрофоны, радиолокаторы, гусеницы, щупальца, и т. д. Центральный компьютер робота состоит из тысяч устройств (блоков из моделей), обрабатывающих одни и те же данные, поступающие из внешнего мира, и создающих внутреннее представление мира в “мозгу” робота. Это внутреннее представление по очевидным причинам будет создаваться с помехами и постоянно корректироваться.

Центральный компьютер использует созданное представление мира для расчета оптимальной траектории, позволяющей роботу избежать опасностей и выполнить поставленное задание. Функционирование робота в подобных условиях можно представить как одновременное выполнение нескольких параллельных задач, контролируемых разными устройствами в центральном компьютере.

Следует отметить, что если резонансные устройства не конкурируют за общий ресурс (камеру, руку и т. д.), то нет необходимости выбирать из них главное.

Работа всех устройств робота (в том числе и вычислительных) регулируется вектором параметров, размерность которого может быть очень большой. Поиск оптимальных значений этих параметров возможен только после постановки задачи, включающей в себя определение функционала качества. В данном случае функционал качества будет оцениваться центральным компьютером. Основные возможные критерии --- соответствие внутренней картины мира реальным условиям (число коррекций), число и качество информационных резонансов и т. д. Настройку параметра с высокой размерностью удобнее всего производить рандомизированным алгоритмом стохастической оптимизации, описанном в следующем разделе, который хорошо ложится на логику квантовых вычислительных устройств и по многим параметрам предпочтительнее других подобных алгоритмов для оптимизации систем в реальном времени.

Более конкретный пример. Предположим, что у нас есть набор устройств, способных распознавать определенные виды изображений (каждое из устройств настроено на какой-либо конкретный объект). Все эти устройства получают в реальном времени изображение с некоторой цифровой камеры. В каждый момент времени только одно из устройств, объявившее себя главным, может управлять камерой. Существует также одно устройство, управляющее камерой в случаях, когда никто не объявил себя главным. Оно может действовать по какому-нибудь простому алгоритму – например, осматриваться или сканировать некоторый важный участок, в котором ожидается появление объектов. Картинка с камеры постоянно подается на входы всех распознавателей, и те в свою очередь пытаются распознать в ней свой объект. Результатом их работы является степень совпадения в процентах (например, объект является кошкой с вероятностью 75%). Под информационным резонансом естественно понимать превышение “степени достоверности” у какого-то из устройств заранее установленного порога (например, 90%), в результате это устройство объявляет себя главным и берет на себя управление камерой для более детального изучения объекта или слежения за ним.

5. Вычисление “за такт” градиента многопараметрической функции

Рассмотрим задачу о выборе оптимального значения многомерного параметра сложной системы в условиях реального времени и оценке ее работы при неконтролируемых возмущениях.

Математически при достаточно общих предположениях эту проблему можно сформулировать как задачу о минимизации функции

$$f(x) = \mathrm{E}_w\{F(w,x)\}$$

(типа функционала среднего риска), зависящей от контролируемого векторного q-мерного аргумента x и неконтролируемого случайного вектора w с, вообще-то говоря, неизвестным распределением. Здесь $\mathrm{E}_w\{\cdot\}$ – операция усреднения по распределению w.

Если $f(\cdot)$ – непрерывно дифференцируемая функция, то необходимым условием того, что θ – точка минимума функции $f(\cdot)$, является равенство нулю в этой точке ее вектор-градиента $\nabla f(\theta) = 0$. Предположив, что известны значения вектор-градиента функции $f(\cdot)$ и матицы ее вторых производных (гессиана), для нахождения точки минимума можно воспользоваться классической схемой вычислений по методу Ньютона:

$$\hat{\theta}_k = \hat{\theta}_{k-1} - [\nabla^2 f(\hat{\theta}_{k-1})]^{-1} \nabla f(\hat{\theta}_{k-1}),$$

где $n = 1,2,\ldots$. Если матрица-гессиан $\nabla^2 f(\hat{\theta}_{k-1})$ в некоторой окрестности точки θ задает положительный ограниченный оператор и начальное значение $\hat{\theta}_0$ выбрано достаточно близко к точке локального минимума θ, то последовательность оценок $\{\hat{\theta}_k\}$ сходится к θ. Недостатком этого алгоритма является необходимость обращать матрицу-гессиан на каждом шаге, что может представлять собой определенную трудность при большой размерности. В некоторых случаях удается выбрать рекуррентный способ для пересчета матриц, обратных к гессиану. Для упрощения алгоритма их иногда обоснованно заменяют на положительные числа α_k, получая в результате алгоритм типа стохастической аппроксимации.

Если значения градиента функции $f(\cdot)$ неизвестны, то стандартным подходом к решению задачи является использование конечных разностей для аппроксимации градиента. Обозначим через e_i стандартный единичный вектор в направлении i-й координаты. Пусть $\{\beta_k\}$ – некоторая последовательность положительных чисел. В качестве аппроксимации i-й компоненты вектор-градиента можно использовать фурмулы

$$\nabla f(\hat{\theta}_{k-1})^{(i)} \approx \frac{f(\hat{\theta}_{k-1} + \beta_k e_i) - f(\hat{\theta}_{k-1} - \beta_k e_i)}{2\beta_k}.$$

Отметим, что этот стандартный подход к аппроксимации вектор-градиента требует на каждом шаге алгоритма оценивания произвести $2q$ измерений значений минимизируемой функции при размерности искомого минимизирующего вектора, равной q.

Как поступить, если нельзя использовать в алгоритме не только градиент функции $f(\cdot)$, но и ее точные значения? Такая проблема возникает, если вид функций $f(\cdot)$ и $F(\cdot,\cdot)$ известен не полностью либо если на вычисление соответствующих значений затрачивается чрезмерное количество усилий при дороговизне экспериментов или большой размерности вектора неизвестных параметров. Более того, в задачах оптимизации достаточно часто можно воспользоваться только зашумленной информацией о значениях функции $F(w,x)$ в выбираемых точках x с неконтролируемыми при этом значениями случайной величины w.

Существенно упростить процедуру оценки градиента и улучшить характеристики ее оценок позволяет включение одновременно в канал наблюдения, через выбираемый параметр, и в направление вектора изменения очередной оценки так называемого *пробного одновременного возмущения*. В отличие от классических конечно-разностных процедур при выборе очередной точки измерения функции случайному возмущению подвергаются одновременно все координаты.

Пусть $\{\Delta_k\}$ – последовательность наблюдаемых, независимых бернуллиевских случайных векторов (координаты вектора Δ_k не зависят друг от друга и принимают с равной вероятностью значения плюс/минус единица). Оказывается, что при зашумленных наблюдениях без существенных потерь в скорости сходимости для построения состоятельной последовательности оценок можно воспользоваться алгоритмом, использующим всего два зашумленных измерения функции $F(\cdot,\cdot)$ на каждой итерации:

$$\hat{\theta}_k = \hat{\theta}_{k-1} - \alpha_k \Delta_k \frac{y_k^+ - y_k^-}{2\beta_k}, \qquad y_k^{\pm} = F(w_k^{\pm}, \hat{\theta}_{k-1} \pm \beta_k \Delta_k) + v_k^{\pm}.$$

Более того, аналогичными свойствами обладает алгоритм с одним зашумленным наблюдением на каждой итерации:

$$\hat{\theta}_k = \hat{\theta}_{k-1} - \frac{\alpha_k}{\beta_k} \Delta_k y_k, \qquad y_k = F(w_k, \hat{\theta}_{k-1} + \beta_k \Delta_k) + v_k.$$

В дальнейшем удобнее будет использовать другой вид алгоритма с одним измерением:

$$\hat{\theta}_k = \hat{\theta}_{k-1} - \gamma_k (x_k - \hat{\theta}_{k-1}) y_k, \quad x_k = \hat{\theta}_{k-1} + \beta_k \Delta_k, \; y_k = F(w_k, x_k) + v_k, \; \gamma_k = \frac{\alpha_k}{\beta_k^2}. \qquad (1)$$

Эти рекуррентные процедуры называются *рандомизированными алгоритмами стохастической аппроксимации*, так как в их структуру неотъемлемой частью входит случайное пробное одновременное по всем координатам возмущение, которое также одновременно используется и в задании направления очередного изменения оценки и при выборе новой точки измерения. Иногда встречаются названия *стохастическая аппроксимация со случайными направлениями*, *поисковый алгоритм стохастической аппроксимации* или *стохастическая аппроксимация с возмущением на входе*. В англоязычной литературе широко используется название *одновременно возмущаемая стохастическая аппроксимация* (“simultaneous perturbation stochastic approximation”, SPSA).

Преимущества SPSA:

- асимптотически оптимальная скорость сходимости (Поляк, Цыбаков, 1990);
- минимальность числа измерений на итерации (Spall, 1992);
- состоятельность при почти произвольных помехах (Граничин, 1989);
- работоспособность в нестационарных задачах (Вахитов, Граничин, Гуревич, 2009);
- легко может быть реализован на квантовом компьютере (Вахитов, Граничин, Сысоев, 2006).

В (Граничин, Поляк, 2003) приведены точные условия, обеспечивающие состоятельность оценок рандомизированных алгоритмов стохастической аппроксимации, из которых наиболее существенным является условие о слабой коррелированности пробного возмущения $\{\Delta_k\}$ и последовательностей неопределенностей $\{w_k\}$ и $\{v_k\}$. Естественно, что среднеквадратичная скорость сходимости первого рандомизированного

алгоритма с двумя измерениями обычно выше, чем у второго. Но стоит заметить, что в целом ряде практических задач оптимизации систем реального времени, обнаружения сигналов и адаптивного управления важно иметь возможность использовать алгоритм только с одним наблюдением на каждом шаге, так как в этих задачах трудно сделать не только $2q$ наблюдений, как в классических конечно-разностных процедурах, но недоступны даже два наблюдения с независимыми от Δ_k помехами. В отличие от оценивания по классическим конечно-разностным процедурам применение рандомизированных алгоритмов эффективно и при почти произвольных аддитивных помехах в наблюдении $\{v_k\}$. В подтверждение этого факта в случае неизвестной, но ограниченной детерминированной последовательности помех $\{v_k\}$ остановимся здесь только на неформальном объяснении, близком к доказательству в (Граничин, 1989).

Пусть дважды непрерывно дифференцируемая вещественная функция $f(X)$ вещественного аргумента X имеет в R^q единственный минимум в некоторой точке θ:

$$(X-\theta)^T \nabla f(X) \geq \mu \| X-\theta \|^2, \forall X \in \mathrm{R}^q$$

с некоторой постоянной $\mu > 0$, и для градиента функции выполнено условие Липшица:

$$\| \nabla f(X) - \nabla f(\theta) \| \leq \| X-\theta \|, \forall X, \theta \in \mathrm{R}^q$$

с некоторой постоянной A. Выберем пробное одновременное возмущение Δ_k принимающим с равной вероятностью значения плюс/минус единица независимо от v_k.

Рассмотрим последовательность оценок $\{\hat{\theta}_k\}$, формируемых по рандомизированному алгоритму стохастической аппроксимации с одним измерением. Обозначим $D_k = \| \hat{\theta}_k - \theta \|^2$ и, учитывая центрированности Δ_k и независимость Δ_k от v_k, оценим условное математическое ожидание:

$$\mathrm{E}\{D_k \mid \hat{\theta}_i, i<k\} \le D_{k-1} - \frac{\alpha_k}{\beta_k}(\hat{\theta}_{k-1}-\theta)^T \mathrm{E}\{\Delta_k y_k \mid \hat{\theta}_i, i<k\} + \frac{\alpha_k^2}{\beta_k^2}\mathrm{E}\{\| \Delta_k \|^2 \; y_k^2 \mid \hat{\theta}_i, i<k\} = D_{k-1} -$$

$$-\frac{\alpha_k}{\beta_k}(\hat{\theta}_{k-1}-\theta)^T \mathrm{E}\{\Delta_k f(\theta_{k-1}+\beta_k\Delta_k) \mid \hat{\theta}_i, i<k\} + \frac{\alpha_k^2}{q\beta_k^2}\mathrm{E}\{y_k^2 \mid \hat{\theta}_i, i<k\}.$$

Разложив значение функции $f(\hat{\theta}_{k-1}+\beta_k\Delta_k)$ по формуле Тейлора, получим

$$f(\hat{\theta}_{k-1}+\beta_k\Delta_k) = f(\hat{\theta}_{k-1}) + \beta_k\Delta_k^T \nabla f(\hat{\theta}_{k-1}) + \beta_k^2\zeta_k,$$

где ζ_k – некоторое число между $\hat{\theta}_{k-1}$ и $\hat{\theta}_{k-1}+\beta_k\Delta_k$ (вообще говоря, ζ_k – случайная величина). С учетом последней формулы, принимая во внимание центрированность Δ_k, выводим:

$$\mathrm{E}\{D_k \mid \hat{\theta}_i, i<k\} \le (1+\frac{1}{2}\alpha_k\beta_k)D_{k-1} - \alpha_k(\hat{\theta}_{k-1}-\theta)^T \nabla f(\hat{\theta}_{k-1}) + \xi_k,$$

где $\xi_k = \mathrm{E}\{\alpha_k\beta_k\zeta_k^2/2 + \alpha_k^2\beta_k^{-2}y_k^2 \mid \hat{\theta}_i, i<k\}$. Отсюда, в силу сильной выпуклости функции $f(\cdot)$, видно, что последовательность $\{D_k\}$ – почти супермартингал:

$$\mathrm{E}\{D_k \mid \hat{\theta}_i, i<k\} \le (1-\gamma_k)D_{k-1} + \xi_k,$$

где $\gamma_k = \mu\alpha_k - \alpha_k\beta_k/2$. Если предположить, что для числовых последовательностей $\{\alpha_k\}$ и $\{\beta_k\}$ выполняются условия

$$\sum_k \alpha_k = \infty, \quad \beta_k \to 0, \quad \sum_k \frac{\alpha_k^2}{\beta_k^2} < \infty, \quad \sum_k \alpha_k\beta_k < \infty,$$

то выполнены все условия леммы Роббинса-Сигмунда (1971) о сходимости почти супермартингалов. Следовательно, последовательность оценок $\{\hat{\theta}_n\}$ сходится к θ с вероятностью единица. При этом дисперсия ошибки оценивания прямо пропорциональна α_n^2/β_n^2.

Рассмотрим проблему выбора вычислительного устройства, наилучшим образом подходящего для выполнения рандомизированного алгоритма стохастической оптимизации с одним измерением функции штрафа на итерации. Предположим, что разрядность используемого квантового вычислителя равна n. Пусть на самом деле задан квантовый “черный ящик”', который вычисляет $f(x)$. Более точно, пусть определен унитарный оператор, реализующий на квантовом компьютере функцию $f(x)$,

который задан на всех определяющих аргумент функции классических двоичных цепочках x длины qn и “управляющей” произвольной двоичной цепочке z длины n, так:

$$U_f :|x\rangle|z\rangle \to |x\rangle|z \oplus f(x)\rangle,$$

где $\oplus$ – побитовоя операция “логического или”'. Это способ задания оператора на базисных векторах. На все остальные вектора оператор продолжается линейно. Легко видеть, что построенный оператор обратим и действует в комплексном пространстве размерности 2^{qn+n}.

Рассмотрим квантовую цепь, изображенную на Рис. 3, которая позволяет получить следующую оценку точки минимума функции по рандомизированному алгоритму стохастической аппроксимации с одним измерением (1) за “один такт”.

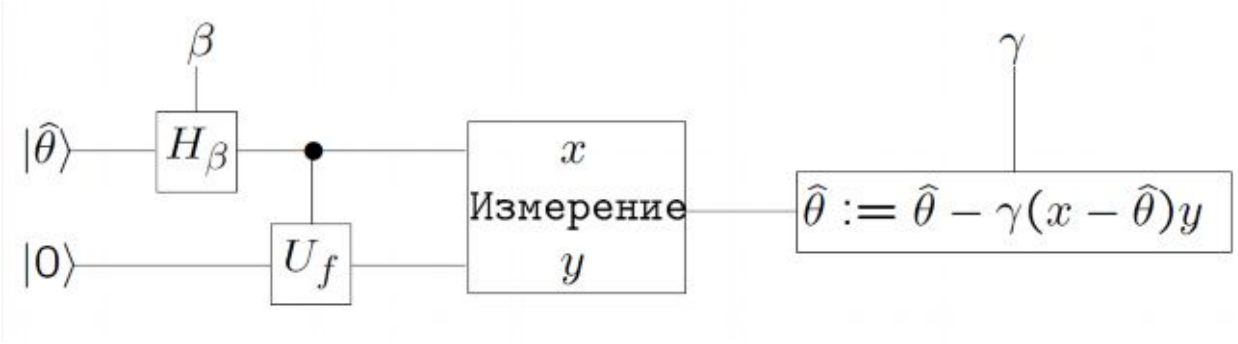


Рис. 3. Вычисление “за такт” градиента многопараметрической функции.

Для подачи на вход вычислителя первоначально подготавливается суперпозиция 2^q возмущенных значений текущего вектора оценки:

$$x_k = H_\beta |\hat{\theta}_{k-1}\rangle = \frac{1}{2^{\frac{q}{2}}} \sum_{\Delta_i \in \{-1,+1\}^q} |\hat{\theta}_{k-1} + \beta_k \Delta_i\rangle,$$

где ± 1 рассматриваются как n-разрядные числа. После применения унитарного оператора U_f к $|x_k\rangle|0\rangle$ получаем

$$U_f |x_k\rangle|0\rangle = \frac{1}{2^q} \sum_{\Delta_i \in \{-1,+1\}^q} |\hat{\theta}_{k-1} + \beta_k \Delta_i\rangle |f(\hat{\theta}_{k-1} + \beta_k \Delta_i)\rangle.$$

Из общих свойств модели квантовых вычислений следует, что после измерения полученного состояния, с вероятностью $\frac{1}{2^q}$ будет получен один из векторов вида

$$|\hat{\theta}_{k-1} + \beta_k \Delta_i\rangle \,|\, f(\hat{\theta}_{k-1} + \beta_k \Delta_i)\rangle,\ \Delta_i \in \{-1,+1\}^q.$$

Из первых qn разрядов этого вектора легко получить реализацию вектора случайного возмущения Δ_i. По рандомизированному алгоритму оценивания координаты вектора Δ_i надо умножить на соответствующее значение функции потерь в возмущенной точке, которое записано в последних n разрядах полученного после измерения результата

$$|\hat{\theta}_{k-1} + \beta_k \Delta_i\rangle \,|\, f(\hat{\theta}_{k-1} + \beta_k \Delta_i)\rangle,\ \Delta_i \in \{-1,+1\}^q.$$

5. Заключение

Описанная новая модель вычислений позволяет описывать если не все, то подавляющее большинство процессов реального мира, а также работу всевозможных существующих и будущих вычислительных устройств, включая аналоговые и биокомпьютеры, нейрокомпьютеры, квантовые компьютеры и т. д. Особенностью предлагаемого подхода является отказ от редукции сложности в процессе вычисления. Сложность вычислимого объекта должна быть эквивалентна сложности вычисляемого. Другими словами, понятие вычислительной сложности правильнее рассматривать относительно выбранной системы базисных эволюционных примитивов, а не относительно традиционно рассматриваемых битовых преобразований {0; 1}. Квантовые и нейрокомпьютеры обещают сильно изменить представления о вычислительной мощности современных вычислительных устройств. Увеличение вычислительной мощности, возможное за счет использования новых моделей вычислений, основывающихся на физических явлениях, позволяет предположить, что в будущем новые компьютеры смогут решать задачи, невыполнимые для обычных компьютеров.

Список литературы

Вахитов А.Т., Граничин О.Н., Сысоев С.С., (2006) Точность оценивания рандомизированного алгоритма стохастической оптимизации. *Автоматика и телемеханика*, № 4 , с. 86-96.

Вахитов А.Т., Граничин О.Н., Гуревич Л.С. (2009) Алгоритм стохастической аппроксимации с пробным возмущением на входе в нестационарной задаче оптимизации // Автоматика и телемеханика, № 11, с. 70-79.

Воеводин В.В., (2010) *Всерос. научн. конф. «Научный сервис в сети Интернет: суперкомпьютерные центры и задачи»* 20-25 сент. 2010 г., Новороссийск – М.

Граничин О.Н., (1989) Об одной стохастической рекуррентной процедуре при зависимых помехах в наблюдении, использующей на входе пробные возмущения. *Вестник Ленингр. ун-та*, сер. 1, вып. 1, с. 19--21.

Граничин О.Н., Васильев В.И., (2010) Гибридная модель процесса вычислений: обобщение концепции машины Тьюринга. *Нейрокомпьютеры: разработка, применение*, № 6, с. 51 -58.

Граничин О.Н., Молодцов С.Л., (2006) *Создание гибридных сверхбыстрых компьютеров и системное программирование*. СПб, 108 с.

Граничин О.Н., Поляк Б.Т., (2003) *Рандомизированные алгоритмы оценивания и оптимизации при почти произвольных помехах*. М.: Наука, 291с.

Поляк Б.Т., Цыбаков А.Б., (1990) Оптимальные порядки точности поисковых алгоритмов стохастической аппроксимации. *Проблемы передачи информации*, № 2, с. 45-53.

Shor, P., (1997) Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput. 26*, pp. 1484-1509.

Tien, D.K., (2003) Computing the non-computable, *Contemporary Physics, 44*, pp. 51-71.

Granichin O.N., Vasil'ev V.I., (2010) Computational model based on evolutionary primitives. Turing machine generalization. *International Journal of Nanotechnology and Molecular Computation*, v. 2, no. 2, pp. 30-43.

Robbins H., Siegmuud D., (1971) A convergence theorem for nonnegative almost super--martingales and some applications. In: *Optimizing Methods in Statistics*, Rustagi J.S. ed.. New York: Academic Press, pp. 233-257.

Spall J.C., (1992) Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE Transactions on Automatic Control*, vol. 37, pp. 332-341.