

1975
Информатика
как эмпирическое исследование:
СИМВОЛЫ И ПОИСК

Аллен Ньюэлл, Херберт Саймон

Тьюринговская премия 1975 г. была присуждена Аллену Ньюэллу и Херберту Саймону на ежегодной конференции ACM, состоявшейся 20 октября в Миннеаполисе. Представляя лауреатов, Бернард А. Галлер, председатель Комитета по присуждению премии им. Тьюринга, заявил:

Я рад, что могу сегодня наградить Тьюринговской премией двух моих давних друзей, профессоров Аллена Ньюэлла и Херberта Саймона, из Университета Карнеги — Меллона.

В результате совместной научной работы, продолжающейся более двадцати лет, начатой вместе с Дж. Шоу из Рэнд Корпорейшн и продолженной в сотрудничестве со многими студентами и сотрудниками Университета Карнеги — Меллона, они внесли основополагающий вклад в исследования по искусственному интеллекту, психологию познания и в обработку списков.

В области искусственного интеллекта они способствовали становлению этой дисциплины как арены интенсивной научной работы, развитию эвристического программирования вообще и эвристического поиска, анализа средств и целей, а также методов индукции в особенности, продемонстрировав достаточность этих механизмов для решения интересных задач.

В психологии они были главными пропагандистами идеи о том, что человеческое сознание может быть описано как символьная система, и они разработали подробные теории решения задач человеком, верbalного обучения и индуктивного поведения в ряде предметных областей, используя компьютерные программы для воплощения этих теорий и моделирования поведения человека.

Они несомненно являются изобретателями обработки списков, внесшими важный вклад как в технологию разработки программного обеспечения, так и в создание концепции компьютера как системы для манипулирования символьными структурами, а не просто процессора для обработки числовых данных.

Это награда почетна для профессоров Ньюэлла и Саймона, но и для нашей Ассоциации большая часть, что мы можем включить их имена в список наших лауреатов, поскольку это повышает престиж и значимость Тьюринговской премии ACM.

* * *

Информатика — это изучение явлений, связанных с вычислительными машинами. Основатели нашего общества отлично понимали это, назвав его Ассоциацией вычислительной техники (Association for Computing Machinery). Машина (причем не только аппаратура, но живая, запрограммированная машина) — вот тот организм, который мы изучаем.

Данная Тьюринговская лекция — десятая по счету. Девять человек, которые были нашими предшественниками на этой трибуне, изложили девять различных точек зрения, поскольку на изучаемый нами организм, вычислительную машину, можно смотреть с разных сторон, и исследовать ее на разных уровнях. Мы глубоко признательны за возможность присутствовать здесь сегодня и изложить еще одну точку зрения, ту, которая послужила основой научной работы, за которую мы удостоены этой награды. Мы будем говорить об информатике как об эмпирическом исследовании.

Наш подход — лишь один из многих возможных; предыдущие лекции вполне прояснили это. Однако все лекторы-лауреаты, вместе взятые, все же не могут очертить весь круг задач нашей науки. Многие фундаментальные ее аспекты остались за пределами этих десяти лекций. И если когда-нибудь и наступит время — а это наверняка будет нескоро — когда на карте принадлежащих информатике земель не останется белых пятен и она будет обсуждена со всех возможных сторон, — тогда-то и придет пора начать все сначала. Ведь такому зайцу, как лектор, приходится каждый год предпринимать спринтерский бросок, чтобы догнать черепаху научного и технического прогресса и преодолеть то расстояние, которое она успела за год проползти своим неторопливым, размеренным шагом по стезе малых, постепенных улучшений. Каждый год порождает новое отставание и требует нового спринтерского броска, так как в науке последнего слова не бывает.

Информатика — это эмпирическая дисциплина. Ее можно было бы назвать экспериментальной наукой, если бы, как и в случае с астрономией, экономикой и геологией, некоторые из ее уникальных форм наблюдения и практики не исключали их подгонку под узкий стереотип экспериментального метода. Тем не менее, это все же эксперименты. Каждая новая построенная машина — это эксперимент. В самом деле, конструирование машины задает природе некий вопрос; мы слушаем ответ, наблюдая машину в работе, и анализируем его, применяя все имеющиеся в нашем распоряжении средства измерения и анализа. Ни машины, ни программы не являются черными ящиками; они создаются и проектируются людьми, мы можем вскрыть их и посмотреть, что там внутри. Мы можем сопоставить их устройство с их поведением и многому научиться

на одном-единственном эксперименте. Нам не нужно создавать 100 экземпляров, скажем, доказывателя теорем, чтобы статистически продемонстрировать, что он не может избежать комбинаторного взрыва при поиске решения тем способом, на который мы рассчитывали. Изучение программы после нескольких прогонов позволяет отыскать ошибку и перейти к новой попытке.

Мы делаем компьютеры и пишем программы по многим причинам. Мы строим их, чтобы они служили обществу, решали его экономические задачи. Но как представители фундаментальной науки мы строим компьютеры и пишем программы для того, чтобы открывать новые явления и исследовать те явления, которые нам уже известны. Общество часто оказывается неспособным это понять; оно считает, что компьютеры и программы следует разрабатывать лишь ради экономической выгоды, которую они могут принести (или же в качестве промежуточных этапов процесса разработки, ведущего к такому использованию). Необходимо понимать, что явления, связанные с компьютерами, глубоки и загадочны, и прояснение их природы требует большой экспериментальной работы. Нужно осознать, что, как и во всякой науке, преимущества, приобретаемые в результате экспериментирования и понимания, реализуются в постоянном обретении новых методов, и именно эти методы позволяют создавать практически полезные инструменты, помогающие обществу достигать своих целей.

Однако смысл нашего выступления не в том, чтобы искать понимания со стороны общественности, далекой от наших проблем. Мы хотим проанализировать один аспект нашей науки, а именно развитие нового фундаментального знания путем эмпирического исследования. Лучше всего сделать это на конкретных примерах. Нам простят, если, учитывая повод, по которому мы приглашены здесь выступать, мы выберем примеры из области наших собственных исследований. Как станет ясно, эти примеры охватывают всю историю разработки искусственного интеллекта, особенно ее ранние этапы. Они основаны на гораздо более обширном материале, чем наши личные достижения. Даже там, где речь будет идти о наших собственных разработках, нужно помнить, что мы работали не одни. Из наших сотрудников следует особенно отметить Клиффа Шоу, вместе с которым мы образовали группу из трех человек и проработали совместно в течение всего волнующего периода конца 50-х годов. Но мы также работали вместе с очень многими студентами и сотрудниками Университета Карнеги — Меллона.

Недостаток времени позволяет выбрать лишь два примера. Первый из них — разработка понятия символной системы.

Второй — разработка понятия эвристического поиска. Оба этих понятия очень важны для углубления нашего понимания того, как обрабатывается информация и как достигается интеллектуальность. Однако они не могут даже приблизительно охватить все направления работ по искусенному интеллекту, хотя они представляются нам полезными для демонстрации характера фундаментального знания в этой области информатики.

1. СИМВОЛЫ И ФИЗИЧЕСКИЕ СИМВОЛЬНЫЕ СИСТЕМЫ

Одним из важнейших вкладов в понимание информатики стало объяснение, причем на весьма фундаментальном уровне, что представляют собой символы. Это объяснение является научным утверждением о том, что такое Природа. Оно выведено опытным путем в ходе длительного и постепенного развития.

Символы лежат в основе разумных действий, которые, конечно, являются главным предметом изучения искусственного интеллекта. По этой причине это центральный вопрос всей информатики. Любая информация обрабатывается компьютерами для достижения некоторых целей, и мы оцениваем интеллектуальность системы по ее способности достигать заявленных целей в условиях изменчивости, трудностей и сложностей, порождаемых характером поставленных задач. Этот общий вклад информатики в достижение интеллектуальности затемняется, если поставленные задачи ограничены чрезмерно упрощенной постановкой, потому что тогда все разнообразие и изменчивость данных могут быть точно предвидены заранее. Это становится более очевидным, когда мы используем компьютеры при решении более широких, сложных и требующих обширных знаний проблем, когда мы пытаемся сделать их своими служителями, способными самостоятельно справляться со всеми неожиданностями реального мира.

Понимание того, каким требованиям должна удовлетворять система, чтобы ее действия были разумными, возникает не сразу. Оно многосоставно, поскольку никакое элементарное свойство не может быть причиной интеллектуальности во всех ее проявлениях. Не существует «принципа интеллектуальности», также как не существует «принципа жизненности», несущего в себе по самой своей природе сущность жизни. Но отсутствие какого-то простого «бога из машины» не означает, что не существует никаких структурных особенностей, необходимых для интеллектуального поведения. Одно из таких необходимых свойств — способность хранить символы и манипулировать

ими. Чтобы поставить этот вопрос научно, можно перефразировать название известной статьи Уоррена Маккалоха [3]: что есть символ, что интеллект может использовать его, и что есть интеллект, что он может использовать символы?

КАЧЕСТВЕННЫЕ СТРУКТУРНЫЕ ПРИНЦИПЫ

Любая наука описывает сущность систем, которые она изучает. Эти описания по своей природе неизменно оказываются качественными, поскольку они устанавливают рамки, в которых можно развивать более детализированное знание. Сущность этих описаний зачастую можно выразить в очень кратких и очень общих утверждениях. Из-за ограниченной конкретности этих утверждений их можно было бы счесть мало-что прибавляющими к общей сумме научных знаний, если бы история не свидетельствовала о величайшей важности таких результатов.

Клеточная теория в биологии. Хороший пример качественного структурного принципа — клеточная теория, утверждающая, что основным структурным элементом всех живых организмов является клетка. Клетки бывают очень разными, но все они содержат ядро, окруженное цитоплазмой, а снаружи все это заключено в клеточную мембрану. Но эта внутренняя структура не является частью описания, которое мы называем клеточной теорией в ее первоначальном виде; это последующее уточнение, появившееся в результате интенсивных исследований. Клеточная теория почти целиком содержитя в приведенном выше утверждении и некоторых достаточно приблизительных представлениях о том, какого размера могут быть клетки. Однако воздействие этого принципа на биологию было огромным, и замешательство среди биологов, предшествующее его постепенному признанию, было велико.

Тектоника плит в геологии. Геология дает интересный пример качественного структурного принципа; интересен он тем, что завоевал признание за последнее десятилетие, так что повышение его статуса еще свежо в памяти. Теория тектоники плит утверждает, что поверхность земного шара представляет собой набор огромных плит, общим числом несколько десятков, которые движутся с геологическими скоростями, сталкиваются, надвигаются или ныряют друг под друга, устремляясь к центру Земли и расплавляясь по мере их погружения. Движения плит объясняют форму и относительное расположение материков и океанов, районов вулканической и сейсмической активности, срединно-океанических хребтов и так далее. После уточнения дополнительных деталей, касающихся размеров и скоростей движения плит, теория в основном была построена.

Разумеется, она не была признана до тех пор, пока ей не удалось объяснить ряд фактов, связанных друг с другом и необъяснимых прежде, например соответствие флоры, фауны и стратиграфических особенностей Западной Африки и северо-восточного побережья Южной Америки. Глобальная тектоника плит — это в высокой степени качественная теория. Теперь, когда она стала общепризнанной, кажется, что вся Земля предоставляет свидетельства в ее пользу повсеместно: ведь мы наконец описываем и видим Землю в соответствии с ее собственной структурой.

Бактериальная теория инфекционных заболеваний. Прошло немногим более ста лет с тех пор, когда Пастер предложил бактериальную теорию инфекций — качественный структурный принцип, вызвавший революцию в медицине. Эта теория постулирует, что большинство болезней возникает из-за присутствия и размножения в теле больного маленьких одноклеточных живых организмов, а заражение представляет собой перенос этих организмов от одного хозяина к другому. Значительная часть развития этой теории состоит в определении организмов, связанных с конкретными болезнями, их описании и выяснении их жизненного цикла. Тот факт, что из этого принципа есть много исключений (многие болезни не вызываются микробами), не умаляет его важности. Принцип требует, чтобы мы искали причину определенного типа; он не настаивает, что мы всегда найдем ее.

Атомистическое учение. Это учение составляет любопытный контраст тем только что описанным качественным структурным принципам. В том виде, в котором он возник из исследований Дальтона и его опытов, доказавших, что химические вещества соединяются в постоянных пропорциях, этот принцип дает типичный пример качественного описания структуры: элементы состоят из маленьких однородных частиц, одинаковых для одного элемента, но разных для разных элементов. Но поскольку лежащие в основе этого описания сорта атомов очень просты, а их разнообразие ограничено, вскоре были построены количественные теории, вовравшие в себя всю общую структуру исходной качественной гипотезы. В отношении же клеток, тектонических плит и бактерий соответствующие структуры столь богаты и многообразны, что исходные качественные принципы остаются несводимыми к количественным теориям, и их вклад в общую теорию ясно различим.

Заключение: Качественные структурные принципы встречаются в науке повсеместно. К их числу относятся многие из наших величайших научных открытий. Как показывают приведенные выше примеры, они часто задают рамки, в которых функционирует целая наука.

ФИЗИЧЕСКИЕ СИМВОЛЬНЫЕ СИСТЕМЫ

Вернемся теперь к символам и определим *физическую символическую систему*. Прилагательное «физическая» означает две важные характеристики. Во-первых, такие системы явным образом подчиняются законам физики — они могут быть реализованы инженерными системами, сконструированными из технически реализуемых элементов. Во-вторых, хотя примененный нами термин «символ» предвосхищает ту его интерпретацию, которую мы имеем в виду, но он относится не только к символальным системам, используемым людьми.

Физическая символьная система состоит из набора элементов, называемых символами, которые представляют собой физические конфигурации, входящие в качестве компонент в элементы другого типа, называемые выражениями (или символьными структурами). Так, символьная структура состоит из некоторого числа позиций, занятых символами и связанных между собой некоторыми физическими соотношениями (например, одна позиция находится рядом с предыдущей). В любой момент времени система содержит некоторый набор таких символьных структур. Кроме этих структур, система также содержит набор процессов, действующих на выражения и порождающих новые выражения: процессы создания, модификации, воспроизведения и уничтожения. Физическая символьная система — это машина, порождающая развивающийся во времени набор символьных структур. Эта система существует в мире объектов, более широком, чем сами эти символьные выражения.

Два понятия являются важнейшими для этой структуры, состоящей из выражений, символов и объектов: обозначение и интерпретация.

Обозначение. Выражение обозначает объект, если система, располагая этим выражением, может либо воздействовать на указанный объект, либо ее поведение зависит от этого объекта. В обоих случаях доступ к объекту достигается через соответствующее выражение, что и составляет суть обозначения.

Интерпретация. Система может интерпретировать выражение, если оно обозначает процесс и если, располагая этим выражением, система может осуществить указанный процесс.

Интерпретация предполагает особую форму зависимого действия: по данному выражению система может выполнить указанный процесс, другими словами, вызвать и выполнить свои собственные процессы исходя из выражений, которые эти процессы обозначают.

Система, способная к обозначению и интерпретации в вышеуказанном смысле, должна также удовлетворять ряду дополнительных требований полноты и замкнутости. Мы можем здесь лишь кратко их упомянуть; все они важны и имеют далеко идущие последствия.

1) Символ может использоваться для обозначения любого выражения. Тем самым, если у нас есть символ, заранее неизвестно, какие выражения он может обозначать. Эта произвольность относится только к символам; позиции для символов и их взаимоотношения определяют, какой объект обозначается сложным выражением. 2) Для каждого процесса, который может быть осуществлен машиной, существует по крайней мере одно выражение, обозначающее этот процесс. 3) Существуют процессы для создания любого выражения и для модификации любого выражения произвольным образом. 4) Выражения устойчивы: будучи однажды созданы, они продолжают существовать, пока не будут явным образом модифицированы или уничтожены. 5) Число выражений, которые система может хранить и использовать, по существу неограниченно.

Тот тип системы, который мы только что определили, нельзя считать незнакомым специалистам по информатике. Он сильно напоминает все универсальные компьютеры. Если для определения машины применить язык манипулирования символами, например Лисп, то это родство станет столь тесным, что нашу гипотетическую машину придется признать родной сестрой универсального компьютера. Мы описали подобную систему не для того, чтобы предложить что-то новое. Как раз наоборот: мы хотим показать, что сейчас известно или предполагается относительно систем и удовлетворяет такому описанию.

Теперь мы можем сформулировать научную гипотезу общего характера — качественный структурный принцип для символьных систем:

Гипотеза о физической символной системе. Физическая символьная система обладает необходимыми и достаточными средствами для интеллектуального поведения общего характера.

Необходимость означает, что любая система, обладающая признаками универсального интеллекта, при ее анализе окажется физической символьной системой. Достаточность означает, что любая физическая символьная система достаточного размера может быть организована таким образом, чтобы проявлять интеллектуальное поведение общего характера. Формулировкой «интеллектуальное поведение общего характера» мы хотим указать на тот же диапазон применимости интеллекта,

который мы наблюдаем в поведении человека: что в любой реальной ситуации поведение соответствует целям системы и адаптивно к требованиям, предъявляемым окружающим миром, в пределах некоторых ограничений, относящихся к скорости и сложности.

Ясно, что гипотеза о физической символной системе является качественным структурным принципом. Она характеризует общий класс систем, к которому относятся системы, способные к интеллектуальному поведению.

Этот принцип является эмпирической гипотезой. Мы определили некоторый класс систем; мы спрашиваем, объясняет ли этот класс ряд явлений, обнаруживаемых в реальном мире. Интеллектуальное поведение наблюдается повсюду вокруг нас в живой природе, в основном в поведении человека. Это та форма поведения, которую мы можем распознать по ее результатам независимо от того, осуществляется ли она людьми или же какими-то другими агентами. Эта гипотеза может на самом деле оказаться ошибочной. Интеллектуальное поведение не так просто осуществить, чтобы всякая система демонстрировала еговольно или невольно. В самом деле, некоторые люди в результате анализа приходят к выводу (по философским или научным соображениям), что эта гипотеза действительно ошибочна. С научной точки зрения, защищать или опровергать ее можно только одним способом: приводя эмпирические данные о реальном мире.

Теперь необходимо проследить развитие этой гипотезы и поискать опытные данные, могущие помочь ее проверить.

РАЗВИТИЕ ГИПОТЕЗЫ О СИМВОЛЬНОЙ СИСТЕМЕ

Физическая символная система является примером универсальной машины. Поэтому гипотеза о символной системе предполагает, что интеллект должен быть реализован с помощью универсального компьютера. Однако эта гипотеза утверждает нечто существенно большее, чем рассуждение, часто выводимое из соображений физического детерминизма и состоящее в том, что любое реализуемое вычисление может быть реализовано универсальной машиной, если это вычисление задано. Эта гипотеза более конкретна: она утверждает, что интеллектуальная машина является символной системой, и тем самым делает конкретное допущение об архитектуре и природе интеллектуальных систем. Важно понимать, откуда возникло это конкретное допущение.

Формальная логика. Корни этой гипотезы восходят к идеям Фреже, а также Рассела и Уайтхеда, которые стремились формализовать логику: реализовать фундаментальные математиче-

ские понятия в логике и тем самым поставить понятия доказательства и дедуктивного вывода на твердое основание. Эти усилия достигли кульминации в создании математической логики — знакомых нам пропозициональной логики, логики первого и высших порядков. Был выработан своеобразный подход, который часто называют «игрой в символы». Логика, в которую оказалась включенной вся математика, стала рассматриваться как игра ничего не обозначающими символами, ведущаяся по определенным, чисто синтаксическим правилам. Вся семантика была полностью изгнана из этой игры. Получилась механическая, хотя и довольно свободная (сейчас мы бы назвали ее недетерминистской) система, о которой можно было доказывать различные утверждения. Мы можем назвать это этапом формальных символьных манипуляций.

Этот общий подход хорошо отражен в развитии теории информации. Снова и снова повторялось, что Шеннон определил систему, полезную лишь для коммуникации и отбора и не имеющую никакого отношения к семантике. Выражались сожаления, что столь общее название присвоено такой узкой области, и были предприняты попытки переименовать ее в «теорию избирательной информации» — безуспешные, разумеется.

Машины Тьюринга и цифровой компьютер. Разработку первых цифровых компьютеров и теории автоматов, начиная с собственных исследований Тьюринга 30-х годов, можно рассмотреть одновременно. Они согласуются в подходе к тому, что считать существенным. Возьмем модель, разработанную самим Тьюрингом, поскольку на ней хорошо видны характерные черты этого подхода.

Машина Тьюринга содержит две памяти: неограниченную ленту и управляющий механизм с конечным числом состояний. Лента содержит данные, т. е. знаменитые нули и единицы. Набор операций, которые машина может совершать над лентой, очень мал: чтение, запись и просмотр. Операция чтения не меняет данных, но обеспечивает условные переходы управления в зависимости от данных, находящихся под читающей головкой. Как нам всем известно, эта модель содержит существенные свойства любого компьютера в смысле того, что компьютеры вообще могут делать, хотя разные компьютеры с разными регистрами памяти и разными наборами операций способны выполнять одни и те же вычисления, расходуя разные объемы памяти и требуя разного времени на их выполнение. В частности, модель, известная как машина Тьюринга, содержит в себе как понятие вычислимости (т. е. что может, а что не может быть вычислено), так и понятие универсальной машины, способной сделать все, что в принципе можно сделать на какой угодно машине.

Поразительно, что два самых глубоких открытия в обработке информации были сделаны в 30-х годах, прежде чем появились современные компьютеры. Этим мы обязаны гению Алана Тьюринга. Это также результат успехов математической логики того времени и свидетельство того, сколь многим информатика ей обязана. Одновременно с работами Тьюринга появились статьи логиков Эмиля Поста и (независимо) Алонсо Чёрча. Исходя из независимых понятий логистических систем (правил вывода Поста и рекурсивных функций соответственно), они пришли к аналогичным результатам о неразрешимости и универсальности — результатам, из которых, как вскоре было показано, следует, что все три упомянутые выше системы эквивалентны. В самом деле, тот факт, что все эти попытки определить наиболее широкий класс систем обработки информации привели независимо к сходным результатам, укрепляет нашу убежденность, что суть обработки информации адекватно выражена в этих моделях.

Ни в одной из этих систем на поверхности не видно понятия символа как элемента, который что-то обозначает. Данные рассматриваются просто как конечные последовательности из нулей и единиц; действительно, нейтральность данных существенна для сведения вычислений к физическим процессам. Система управления с конечным числом состояний всегда рассматривалась как небольшой по размерам автомат, и смысл логических игр заключался в том, чтобы увидеть, насколько малой может быть система состояний без разрушения универсальности машины. Насколько нам известно, никто никогда не играл в игры с динамическим добавлением новых состояний к управляющему устройству, т. е. не думал об управляющей памяти как о хранилище совокупности знаний системы. То, что было достигнуто на этом этапе, представляло собой половину принципа интерпретации: было показано, что машина может действовать исходя из описания. Этот этап можно назвать автоматическими формальными символыми манипуляциями.

Понятие хранимой программы. С разработкой второго поколения электронных машин в середине 40-х годов (после Эниака) появилось понятие хранимой программы. Это было справедливо названо поворотным пунктом в развитии информатики, как в концептуальном, так и в практическом отношении. Программы теперь стали данными, и с ними можно стало оперировать как с данными. Эта возможность, разумеется, уже присутствовала в неявном виде в модели Тьюринга: описания находятся на той же самой ленте, что и данные. Однако эта идея была реализована лишь тогда, когда машины обрели достаточно большую память, чтобы стало практичным размещать настоящие программы в каком-то внутреннем запоминающем

устройстве. В самом деле, у Эниака было всего 20 внутренних регистров памяти.

Концепция хранимой программы воплощает вторую половину принципа интерпретации, а именно ту его часть, в которой утверждается возможность интерпретации собственных данных системы. Но она еще не содержит понятия обозначения, т. е. физического соотношения, лежащего в основе семантики символов.

Обработка списков. Следующий шаг, сделанный в 1956 г., это обработка списков. Содержимым структур данных теперь были символы в том смысле, в каком они понимаются в нашей физической символьной системе: конфигурации, которые что-то обозначали, у которых были денотаты. Списки содержали адреса, через которые был возможен доступ к другим спискам; отсюда понятие списковых структур. То, что этот подход был новым, было неоднократно продемонстрировано нам в те первые дни обработки списков: коллеги спрашивали у нас, где же находятся данные, т. е. какой список в конце концов содержит тот набор бит, который является содержимым системы. Им казалось странным, что такого набора нет, а есть лишь символы, обозначающие другие символьные структуры.

Обработка списков означала сразу три момента в развитии информатики. Во-первых, это создание подлинно динамических структур памяти в машине, которая ранее считалась обладающей лишь фиксированной структурой. Это добавило к нашему набору операций такие, которые строили и модифицировали структуру, а не только заменяли и изменяли содержимое. Во-вторых, это стало первой демонстрацией фундаментальной концепции: компьютер состоит из набора типов данных и набора операций, подходящих для этих типов данных, так что вычислительная система должна использовать любые типы данных, удобные для решаемых ею задач, независимо от машины, на которой эти вычисления реализуются. В-третьих, обработка списков создала модель процедуры обозначения и тем самым определила символьную манипуляцию в том смысле, в котором мы употребляем это понятие в современной информатике.

Как это часто бывает, практика того времени уже предвосхищала все элементы обработки списков: адреса, само собой, использовались для обеспечения доступа к данным, в машинах с магнитными барабанами применялись записанные последовательно, одна за другой, программы (последовательная адресация) и так далее. Но понятие обработки списков в качестве абстракции создал новый мир, в котором обозначение и динамические символьные структуры стали определяющими характеристиками. Погружение первых систем обработки списков в языки (вроде Лиспа) часто описывается как препятствие

широкому внедрению методов обработки списков в практику программирования; на самом деле это было средством собрать воедино все элементы, составляющие новую абстракцию.

Лисп. Еще один шаг, заслуживающий упоминания, это создание в 1956—1960 гг. языка Лисп [2]. Оно завершило обобщение, освободив списковые структуры из плена, из погружения в конкретные машины, и создало новую формальную систему с S-выражениями, которая, как можно показать, эквивалентна другим универсальным схемам вычисления.

Заключение. То, что понятие значащего символа и символьной манипуляции не возникло вплоть до середины 50-х годов, вовсе не означает, что предшествующие этому шаги были несущественными или менее важными. В своей целокупности это понятие объединяет вычислимость, физическую реализуемость (посредством множества технологических приемов), универсальность, символьное представление процессов (т. е. интерпретируемость) и, наконец, символьные структуры и обозначение. Каждый из этих шагов представляет существенную часть целиного.

Первый шаг в этой цепочке, осуществленный Тьюрингом, был мотивирован теоретически, но все остальные имеют глубокие эмпирические корни. Нас вело вперед развитие самих компьютеров. Принцип хранимой программы возник из опыта работы с Эниаком. Обработка списков выросла из попыток создать интеллектуальные программы. Она была подсказана появлением запоминающих устройств с произвольным доступом, обеспечившим четкую физическую реализацию обозначающего символа в виде адреса. Лисп стал итогом накопления практического опыта обработки списков.

ЭМПИРИЧЕСКИЕ ДОВОДЫ

Перейдем теперь к эмпирическим доводам в поддержку гипотезы о том, что физические символьные системы способны к интеллектуальному поведению и что интеллектуальное поведение общего характера требует физической символьной системы. Эта гипотеза является эмпирическим обобщением, а не теоремой. Нам не известно никакого способа продемонстрировать связь между символьными системами и интеллектом чисто логическим путем. При отсутствии такой возможности остается обратиться к фактам. Наша главная цель, однако, не подробный обзор фактов, а использование доступного нам примера для иллюстрации утверждения о том, что информатика является эмпирической наукой. Поэтому мы лишь укажем, какого рода фактами мы располагаем и какова общая природа процесса опытной проверки.

Понятие физической символьной системы приняло в основном современную форму в середине 50-х годов, и с этого времени можно проследить рост исследований в области искусственного интеллекта как единого направления в информатике. Двадцать лет последующей работы привели к накоплению фактических результатов, относящихся к двум основным темам. Первая группа результатов касается *достаточности* физических символьных систем для порождения интеллекта; это попытки строить и проверять конкретные системы, обладающие такими возможностями. Вторая группа фактов обращается к *необходимости* иметь физическую символьную систему во всех случаях, когда имеются проявления интеллекта. Эти исследования начались с Человека, самой известной интеллектуальной системы, и попыток выяснить, можно ли объяснить его интеллектуальное поведение как работу физической символьной системы. Есть и другие виды опытных данных, которые будут вкратце обсуждены ниже, но эти две группы являются важнейшими. Мы рассмотрим их по очереди. Первая имеет общее название искусственного интеллекта, вторая — исследований по когнитивной психологии.

Конструирование интеллектуальных систем. Основная парадигма первоначальной проверки бактериальной теории заболевания была следующей: выявите болезнь; затем ищите микроб. Аналогичная парадигма инспирировала значительную часть исследований в области искусственного интеллекта: выявите класс задач, решение которых требует интеллектуальных способностей; затем разработайте программу для цифрового компьютера, способную решать такие задачи. Простые и хорошо структурированные области первыми привлекли внимание: игры и головоломки, задачи исследования операций типа составления расписаний и распределения ресурсов, простые задачи на индукцию. Дюжины, если не сотни программ такого рода написаны к настоящему времени, каждая из которых в той или иной степени способна к интеллектуальному поведению в соответствующей области. Конечно, интеллектуальность — это не такая вещь, которая либо присутствует в полном объеме, либо отсутствует целиком, и постоянно происходило развитие к более высоким уровням мастерства в конкретных предметных областях и к расширению самих этих областей. Первые шахматные программы считались успешными, если они могли соблюдать правила игры и проявляли некие признаки целесообразности поведения; чуть позже они достигли уровня начинающих игроков-людей; еще через 10—15 лет они стали выигрывать у серьезных любителей. Прогресс был медленным (а общие трудовые затраты программистов — незначительными), но постоянным, и парадигма «построить и испытать» про-

ходила свой обычный цикл: в целом исследовательская активность воспроизводила на макроскопическом уровне базисный цикл «породить и проверить» многих программ искусственного интеллекта.

Существует постоянно расширяющаяся область, в которой удается достичь интеллектуального поведения. От исходных задач исследователи перешли к более общим и трудным: к системам для чтения, понимания и генерации текстов на естественном языке самыми разными способами, к системам интерпретации визуальных сцен, к системам координации движений руки и глаза, системам проектирования, системам, пишущим компьютерные программы, системам распознавания устной речи — этот список, если не бесконечен, то очень длинен по крайней мере. И если существуют какие-то ограничения, за пределами которых наша гипотеза не работает, то пока что таких границ не видно. Вплоть до сегодняшнего дня скорость прогресса в основном определялась сравнительно скромными ресурсами, отводимыми на научные исследования в этой области, и неизбежностью затраты серьезных усилий и средств на каждый крупный новый проект по созданию системы.

Происходило, конечно, и много другого, чем просто накопление примеров интеллектуальных систем, приспособленных к решению задач из конкретных предметных областей. Было бы удивительным и непривлекательным, если бы оказалось, что программы искусственного интеллекта, выполняющие эти разнообразные функции, не имеют между собой ничего общего, кроме того, что это все примеры физических символьных систем. Поэтому существовал большой интерес к поискам механизмов, обладающих общностью, и общих компонент среди программ, выполняющих разные задачи. Этот поиск выводит теорию за рамки первоначальной гипотезы о символьных системах к более полному описанию особого рода символьных систем, эффективных в области искусственного интеллекта. Во втором разделе нашей работы мы обсудим один пример гипотезы этого второго уровня специфичности: гипотезу эвристического поиска.

Поиск общности стимулировал создание нескольких программ, разработанных для выделения общих механизмов решения задач, их отделения от требований, предъявляемых конкретными предметными областями. Общий решатель задач (General Problem Solver, GPS) был, вероятно, первым из них; среди его потомков есть такие современные системы, как PLANNER и CONNIVER. Поиск общих компонент привел к общим схемам представления целей и планов, методам построения дискриминирующих сетей, процедурам управления поиском на дереве, механизмам сопоставления с образцом и системам

грамматического разбора. В настоящее время ведется экспериментальная работа с целью найти удобные средства представления временных соотношений предшествования и последовательности действий, причинности, движения и других подобных грамматических и логических категорий. Во все большей степени становится возможным собирать большие интеллектуальные системы из таких базовых компонент как из модулей.

Можно получить некоторое общее представление о том, что происходит, опять обратившись к аналогии с бактериальной теорией болезней. Если первый всплеск исследований, связанных с этой теорией, в основном состоял в поиске микроорганизмов, связанных с каждой болезнью, то последующие усилия были направлены на изучение того, что представляет собой микроб, т. е. на построение нового структурного уровня на фундаменте исходного качественного структурного принципа. В области искусственного интеллекта ранние исследования имели целью построение интеллектуальных программ для широкого многообразия почти случайно выбранных задач, но в дальнейшем исследования сконцентрировались на более избирательно намечаемых целях в стремлении найти и понять общие механизмы для таких систем.

Моделирование использования символов человеком. Гипотеза о символических системах предполагает, что причиной символического поведения человека служит то, что он обладает свойствами физической символической системы. Поэтому результаты попыток смоделировать поведение человека символическими системами становятся важной частью свидетельств в поддержку этой гипотезы, и исследования по искусственноому интеллекту проводятся в тесном сотрудничестве с психологией обработки информации, как обычно называют эту область.

Поиск объяснений интеллектуального поведения человека в рамках символьных систем увенчался крупными достижениями за последние 20 лет, вплоть до того, что теория обработки информации стала ведущим современным подходом в когнитивной психологии (психологии интеллекта). В особенности в таких областях, как решение задач, формирование понятий и долговременная память, модели символьной манипуляции играют ныне решающую роль.

Исследования психологии обработки информации включают два основных вида эмпирического поиска. Первый из них — проведение наблюдений и постановка экспериментов по поведению человека в ходе выполнения заданий, требующих интеллектуальной активности. Второй, очень похожий на параллельные исследования по искусственноому интеллекту, — программирование символьных систем для имитации наблюдаемого поведения человека. Психологические наблюдения и экспери-

менты привели к формулированию гипотез о символных процессах, используемых людьми, и эти гипотезы служат важными источниками идей, пригодных для построения программ. Так, многие идеи основных механизмов GPS были выведены из тщательного изучения протоколов опытов, в которых испытуемые размышляли вслух, выполняя задания по решению задач.

Эмпирический характер информатики теперь стал еще более очевиден, чем показывает эта связь с психологией. Дело не только в том, что для проверки реалистичности объяснений человеческого поведения имитационными моделями необходимы психологические эксперименты, но и в том, что из экспериментов выводятся новые идеи проектирования и построения физических символьных систем.

Другие свидетельства. Основной корпус данных в пользу гипотезы о физических символьных системах, которые мы до сих пор не рассматривали, это отрицательные данные, т. е. отсутствие конкретных альтернативных гипотез о том, как может быть реализовано интеллектуальное поведение — безразлично, человека или машины. Большая часть попыток построить такие гипотезы предпринималась в области психологии. Здесь имеется непрерывный спектр теорий, на одном полюсе которого лежат точки зрения, обычно называемые «бихевиоризмом», а на другом — «гештальт-психологией». Ни одна из этих точек зрения не может быть реальным конкурентом гипотезы о физических символьных системах по двум следующим причинам. Во-первых, ни бихевиористы, ни «гештальтисты» до сих пор не показали и даже не предложили, как можно было бы показать, что постулируемые ими механизмы достаточны для объяснения интеллектуального поведения при выполнении сложных заданий. Во-вторых, ни одна из этих теорий не сформулирована со степенью конкретности, хотя бы отдаленно напоминающей конкретность программ искусственного интеллекта. Фактически альтернативные теории достаточно расплывчаты для того, чтобы их интерпретация в духе теории обработки информации не была чрезмерно трудным делом; это позволило бы включить их в гипотезу о физических символьных системах как частные случаи.

ЗАКЛЮЧЕНИЕ

Мы привели пример гипотезы о физических символьных системах для того, чтобы дать конкретную иллюстрацию основного тезиса. Тезис этот утверждает, что информатика является научной дисциплиной в обычном смысле этого термина: она выдвигает научные гипотезы и затем ищет их подтверждения путем экспериментирования. Есть, однако, и другая причина:

для выбора именно этой гипотезы для иллюстрации нашего тезиса. Гипотеза о физических символьных системах сама по себе является важной научной гипотезой того сорта, который мы ранее назвали «качественными структурными принципами». Она представляет собой крупное открытие в области информатики, которое, если его удастся подтвердить экспериментально, что, по-видимому, и происходит, окажет существенное и длительное воздействие на эту науку.

Теперь мы обратимся ко второму примеру, роли поиска в интеллектуальной деятельности. Эта тема и частная гипотеза о роли поиска, которую мы рассмотрим, также были очень важны для информатики в целом и для искусственного интеллекта в особенности.

2. ЭВРИСТИЧЕСКИЙ ПОИСК

Знание о том, что физические символьные системы обеспечивают возможность интеллектуального поведения, не говорит нам ничего о том, как они этого добиваются. Наш второй пример качественного структурного принципа в информатике отвечает на этот второй вопрос, утверждая, что символьные системы решают задачи, используя процесс эвристического поиска. Это общее положение, как и предыдущее, основано на опытных данных, а не выведено формально из каких-то других посылок. Однако мы скоро увидим, что оно на самом деле имеет некоторую логическую связь с гипотезой о символьных системах, и, вероятно, можно предвидеть формализацию этой связи в будущем. Пока это время не пришло, нам снова придется рассказывать об эмпирическом исследовании. Мы опишем, что известно о самом эвристическом поиске и подытожим опытные факты, показывающие, как он обеспечивает разумность действий. Мы начнем с формулировки этого качественного структурного принципа — гипотезы об эвристическом поиске.

Гипотеза об эвристическом поиске. Решения задач представляются в виде символьных структур. Физическая символьная система проявляет свою интеллектуальность при решении задач посредством поиска — т. е. порождением и последовательной модификацией символьных структур, пока этот процесс не приведет к структуре, отвечающей решению.

Физические символьные системы должны использовать эвристический поиск при решении задач в силу того обстоятельства, что вычислительные возможности таких систем ограничены: за конечное число шагов и в течение ограниченного периода времени они способны осуществить лишь конечное число процессов. Разумеется, это ограничение не очень сильное, так как

оно распространяется на все универсальные машины Тьюринга. Мы, однако, имеем в виду более сильное ограничение — практическую ограниченность. Можно представить себе системы, неограниченные в практическом смысле, а способные, например, вести поиск, параллельно просматривая вершины экспоненциально ветвящегося дерева с постоянной скоростью продвижения в глубину. Нас здесь будут интересовать не такие системы, а те, вычислительные ресурсы которых малы по сравнению со сложностью ситуаций, с которыми они сталкиваются. Такое ограничение не исключает никаких реально существующих символьных систем, идет ли речь о людях или о компьютерах, если реальны сами задачи. Ограничность ресурсов позволяет нам в большинстве случаев рассматривать символьную систему, как если бы она была serialным устройством, способным в каждый момент времени выполнять лишь один процесс. Если она способна выполнить лишь небольшой объем вычислений в любой короткий интервал времени, то мы вполне можем считать, что она производит вычисления последовательно. Поэтому «символьная система с ограниченными ресурсами» и «serialная символьная система» — это практически синонимы. Проблема распределения ограниченных ресурсов по интервалам времени обычно может рассматриваться, если эти интервалы достаточно короткие, как проблема организации работы serialной машины.

РЕШЕНИЕ ЗАДАЧ

Поскольку способность решать задачи обычно считается главным признаком интеллектуальности системы, то естественно, что большая часть истории искусственного интеллекта занимает попытки построить и понять системы решения задач. Вот уже два тысячелетия философы и психологи обсуждают решение задач, и их рассуждения насквозь пронизаны ощущением тайны. Если вы думаете, что нет ничего загадочного и таинственного в символьной системе, решающей задачи, то выдите нашего времени, чьи взгляды сформировались не раньше середины века. Платон (и, согласно его рассказу, Сократ) считали трудным для понимания даже то, как задачи могут быть поставлены, не говоря уже о том, как они могут быть решены. Позвольте вам напомнить, как он формулирует эту загадку в диалоге «Менон»:

Менон: Но каким же образом, Сократ, ты будешь искать вещь, не зная даже, что она такое? Какую из неизвестных тебе вещей изберешь ты предметом исследования? И если ты в лучшем случае натолкнешься на нее, откуда ты узнаешь, что она — именно то, чего ты не знал?

Чтобы справиться с этой головоломкой, Платон изобрел свою знаменитую теорию припоминания: когда вы думаете, что вы открываете или изобретаете что-то, вы на самом деле лишь припоминаете то, что вы уже знали в предшествующем существовании. Если это объяснение кажется вам слишком нелепым, то в наше время есть и другое, значительно более простое и основанное на понимании того, что такое символные системы. Приближенно его можно изложить так:

Поставить задачу — это обозначить (1) *тест* для некоторого класса символьных структур (решений задачи) и (2) *генератор* символьных структур (потенциальных решений). Решить задачу — это породить структуру с помощью (2), которая удовлетворяет тесту (1).

Перед нами стоит задача, если мы знаем, что мы хотим сделать (имеется тест), и при этом мы не знаем немедленно, как это сделать (наш генератор не порождает немедленно символьную структуру, удовлетворяющую тесту). Символьная система способна ставить и решать задачи (иногда), потому что она может порождать и проверять символьные структуры.

Если бы этим исчерпывалось все, что нужно для решения задач, то почему бы просто не породить сразу именно такое выражение, которое удовлетворяет тесту? Фактически именно это мы и делаем, когда желаем и мечтаем. «Если бы желания были лошадьми, то бродяги могли бы ездить верхом» — гласит пословица. Но за пределами мира фантазий это невозможно. Знать, как мы стали бы проверять нечто уже построенное, — это не то же самое, что знать, как это построить, т. е. это не означает, что у нас есть какой-либо генератор, способный порождать решения.

Например, все знают, что значит «решить» задачу, если наша цель — одержать победу в шахматной партии. Существует простой тест для определения выигрышных позиций, а именно: король противника не может избежать шаха, т. е. ему поставлен мат. В мире фантазий мы просто порождаем стратегию, которая ведет к мату при любых контратаках противника. Увы, неизвестно никакого генератора подобной стратегии для существующих символьных систем (людей или машин). Вместо этого хорошие ходы в шахматной игре ищутся путем порождения различных вариантов и последующей мучительной процедуры их оценивания с использованием приближенных и часто ошибочных мер, которые предположительно указывают вероятность того, что та или иная последовательность ходов лежит на маршруте, приводящем к выигрышной позиции. Генераторы ходов существуют, не существует генераторов выигрышных ходов.

Прежде чем для данной задачи можно будет создать генератор ходов, должно существовать проблемное пространство, т. е. пространство символьных структур, в котором можно представить проблемные ситуации, включая исходную и целевую. Генераторы ходов — это процессы модификации одной ситуации из проблемного пространства в другую ситуацию того же пространства. Основные характеристики физических символьных систем гарантируют, что они могут представлять проблемные пространства и обладают генераторами ходов. Каким образом в каждой конкретной ситуации они синтезируют проблемное пространство и генераторы ходов, подходящие для этой ситуации, — это вопрос, все еще в очень большой степени находящийся на переднем крае исследований по искусственному интеллекту.

Когда символьная система располагает постановкой задачи и проблемным пространством, она должна выполнить следующее задание: использовать свои ограниченные вычислительные ресурсы для порождения возможных решений, одного за другим, пока она не найдет такое, которое удовлетворяет тесту, определяющему постановку задачи. Если бы система могла как-то управлять порядком, в котором порождаются потенциальные решения, то было бы желательно, чтобы настоящие решения обладали высокой вероятностью быть порожденными на ранних этапах этого процесса. Символьная система проявляет интеллектуальность в той степени, в которой она преуспела в этом. Для системы с ограниченными вычислительными ресурсами интеллектуальность состоит в том, чтобы на каждом этапе мудро выбирать следующий шаг.

ПОИСК В РЕШЕНИИ ЗАДАЧ

В течение первых десяти лет исследований по искусственному интеллекту изучение решения задач означало почти то же самое, что изучение процессов поиска. Из наших описаний сущности задач и решения задач легко видеть, почему это было так. На самом деле можно было бы спросить, могло ли быть по-другому. Но прождя член отвечать на этот вопрос, мы должны глубже изучить природу процессов поиска, насколько она выявила за эти десять лет исследований.

Извлечение информации из проблемного пространства. Рассмотрим набор символьных структур, небольшое подмножество которого составляют решения данной задачи. Предположим кроме того, что решения случайным образом распределены по всему пространству. Это означает, что нет никакой информации, которая позволила бы любому генератору поиска работать лучше, чем случайный поиск. Тогда никакая символь-

ная система не смогла бы проявить большую (или меньшую) интеллектуальность, чем другая, при решении задачи, хотя одной могло бы повезти больше, чем другой.

Поэтому необходимое условие появления интеллекта состоит в том, что распределение решений не является полностью случайным, т. е. пространство символьных структур обладает по крайней мере некоторой степенью организованности и упорядоченности. Второе условие — то, что этот порядок в пространстве символьных структур является более или менее распознаваемым. Третье условие интеллектуальности — то, что генератор потенциальных решений способен вести себя по-разному в зависимости от того, какой именно порядок обнаружен. Проблемное пространство должно содержать информацию, а символьная система должна быть способна извлекать и использовать ее. Для начала рассмотрим очень простой пример, в котором интеллектуальность очень просто обеспечить.

Рассмотрим задачу разрешения простого алгебраического уравнения

$$AX+B=CX+D.$$

Тест определяет решение как любое выражение вида $X=E$, такое что $AE+B=CE+D$. Теперь можно использовать в качестве генератора любой процесс, порождающий числа, которые можно было проверить подстановкой в последнее уравнение. Мы не можем называть такой генератор интеллектуальным.

Другой способ — использовать генераторы, опирающиеся на тот факт, что исходное уравнение можно модифицировать добавлением или вычитанием равных величин из обеих частей уравнения, а также умножением или делением их на равные величины, не изменяя его решений. Но конечно, мы можем получить еще больше информации для управления генератором, сравнив исходное выражение с видом решения и сделав именно те изменения в уравнении, которые оставляют его решение неизмененным и одновременно приводят его к желаемому виду. Такой генератор может заметить, что в правой части исходного уравнения имеется нежелательный член CX , вычесть его из обеих частей и снова сгруппировать однородные члены. Затем он может заметить нежелательный член B в левой части и вычесть его. Наконец, он может избавиться от нежелательного коэффициента ($A-C$) в левой части посредством деления.

Так в ходе этой процедуры, которая теперь проявляет явные черты интеллектуальности, генератор порождает последовательно сменяющие друг друга символные структуры, каждая из которых получается модификацией предыдущей; эти модификации нацелены на уменьшение различий между исходной структурой и формой выражения, являющегося тестом; при

этом другие условия, которым должно отвечать решение, сохраняются.

Этот простой пример уже иллюстрирует многие основные механизмы, применяемые символыми системами при интеллектуальном решении задач. Во-первых, каждое последующее выражение не порождается независимо, но порождается модификацией порожденного ранее. Во-вторых, эти модификации делаются не наугад, а зависят от информации двух разных типов. Они зависят от информации, постоянной для всего этого класса алгебраических задач, и поэтому встроенной в саму структуру генератора: все модификации выражений должны оставлять решения уравнения неизменными. Модификации также зависят от информации, изменяющейся на каждом шаге: обнаруженных различий в форме, которые еще остаются, между текущим выражением и желаемым выражением. В результате генератор включает в себя некоторые тесты, которым должно удовлетворять решение: выражения, не удовлетворяющие этим тестам, никогда не будут порождены. Использование информации первого типа гарантирует, что действительно порождается лишь небольшое подмножество всех возможных выражений, но это подмножество содержит решение. Использование информации второго типа позволяет достичь решения посредством последовательных приближений, применяя простую форму анализа соответствия средств и целей для направления поиска.

Нет никакой тайны в том, откуда берется информация, направляющая поиск. Нам не нужно следовать Платону, наделяя символические системы предыдущим существованием, в котором они уже знали решения. Умеренно сложная система порождения и проверки решает задачу без всякой нужды в теории переселения душ.

Деревья поиска. Простая алгебраическая задача может показаться необычным, даже патологическим примером поиска. Это несомненно не метод проб и ошибок, так как хотя в нем было несколько проб, но не было ни одной ошибки. Мы больше привыкли думать о поиске при решении задач как о порождении обильно ветвящихся деревьев возможных частичных решений, число которых может доходить до тысяч, даже миллионов ветвей, прежде чем одна из них приведет к решению. Так, если из каждого порожденного генератором выражения он затем порождает B новых ветвей, то дерево будет расти как B^D , где D — глубина. Дерево, выросшее для алгебраической задачи, обладает тем странным свойством, что его коэффициент ветвлений B равен единице.

Шахматные программы обычно порождают широкие деревья поиска, в некоторых случаях достигающие миллиона ветвей и

более. (Хотя этот пример помогает проиллюстрировать наши утверждения о деревьях поиска, необходимо заметить, что цель поиска в шахматах — не порождение возможных решений, а их оценивание (проверка).) Одна из тем исследований игровых программ в основном была связана с улучшением представления шахматной доски и процессов ходов на этой доске с целью ускорить поиск и сделать возможным поиск на деревьях больших размеров. Мотивация этого направления, конечно, та, что чем глубже динамический поиск, тем точнее будут оценки, которые его завершают. С другой стороны, есть надежные эмпирические данные о том, что сильнейшие шахматисты-люди, т. е. гроссмейстеры, редко изучают деревья, содержащие более ста вариантов. Эта экономия достигается не столько за счет меньшей по сравнению с шахматными программами глубины поиска, сколько за счет очень редкого и избирательного ветвления в каждой вершине. Без ухудшения качества оценок это возможно лишь при встраивании большей избирательности в сам генератор, чтобы он мог выбирать для порождения лишь те ветви, которые с очень высокой вероятностью дадут важную и нужную информацию о качестве позиции.

До некоторой степени парадоксально звучащий вывод, к которому привело это обсуждение, состоит в том, что поиск, т. е. последовательное порождение структур, потенциально способных представлять решение, является основополагающим аспектом проявления символными системами интеллектуальности при решении задач, но объем поиска не является мерой интеллектуальности, при этом проявленной. Задачу делает трудной не большой объем поиска, необходимого для отыскания ее решения, а большой объем поиска, который оказался бы необходимым, если бы не был применен требуемый уровень интеллектуальности. Когда символная система, пытающаяся решить задачу, знает достаточно о том, что ей делать, она прямо идет к своей цели; но всякий раз, когда ее знание становится недостаточным, когда она вступает на неведомую землю, она сталкивается с угрозой пройти через обширный поиск, прежде чем снова найти свой путь.

Возможность экспоненциального взрывного роста дерева поиска, присущая любой схеме порождения решений задачи, предупреждает нас, что не стоит полагаться на грубую силу компьютеров — даже самых больших и быстрых — в качестве компенсации невежества и неразборчивости генераторов решений. Время от времени у некоторых людей возникает надежда, что будет найден достаточно быстродействующий компьютер, который можно будет так хитроумно запрограммировать, чтобы он смог хорошо играть в шахматы методом слепого поиска. О шахматной игре неизвестно ничего такого, что исключало бы

такую возможность теоретически. Эмпирические исследования управления поиском по деревьям больших размеров, приведшие к довольно скромным результатам, сделали это направление намного менее перспективным, чем оно было тогда, когда шахматы были впервые выбраны в качестве подходящего объекта исследований по искусственному интеллекту. Мы должны считать этот факт одним из важных эмпирических результатов изучения шахматных программ.

Формы интеллектуальности. Назначение интеллекта тем самым состоит в том, чтобы предотвратить всегда возможный экспоненциальный рост дерева поиска (комбинаторный взрыв). Как этого можно добиться? Первый способ, уже проиллюстрированный примером решения алгебраического уравнения и шахматными программами, генерирующими только «хорошие» ходы для последующего анализа, это встроить избирательность в генератор: порождать только такие структуры, которые имеют шанс оказаться решениями или лежать на пути, ведущем к решению. Обычно это приводит к уменьшению ветвления, но не устраняет его полностью. Взрывной экспоненциальный рост в конечном итоге не предотвращается (за исключением особенно хорошо структурированных ситуаций, как в примере из алгебры), а лишь отодвигается на некоторый срок. Поэтому интеллектуальная система должна дополнить избирательность своего генератора решений какими-то другими методами направления поиска, использующими информацию о структуре пространства решений.

Двадцатилетний опыт управления поиском по дереву в разных проблемных ситуациях дал небольшой набор общих методов, составляющий часть инструментария любого исследователя искусственного интеллекта. Поскольку эти методы были описаны в общих обзорах вроде [4], мы можем очень кратко подытожить их здесь.

В последовательном эвристическом поиске основной вопрос всегда стоит так: что делать дальше? В поиске по дереву этот вопрос в свою очередь разбивается на два вопроса: [1] из какой вершины дерева вести поиск дальше? [2] в каком направлении двигаться из этой вершины? Информация, полезная для ответа на первый из этих вопросов, может быть рассматриваема как мера относительного расстояния различных вершин от цели. Информация, полезная для ответа на второй вопрос (в каком направлении вести поиск), часто может быть получена, как в алгебраическом примере, обнаружением специфических различий между структурой, представленной текущей вершиной, и структурой, обозначающей цель и описываемой тестом, с последующим выбором действий, связанных с уменьшением этих специфических различий. В этом и состоит метод, из-

вестный как анализ средств и целей; он играет центральную роль в структуре GPS (общего решателя проблем).

Важность эмпирических работ в качестве источника общих идей в исследованиях по ИИ можно ясно продемонстрировать, проследив историю создания многочисленных программ для решения задач, в которой центральную роль играли две идеи: поиск с выбором первыми самых лучших кандидатов и анализ средств и целей. Зачатки выбора первыми самых перспективных промежуточных решений уже присутствовали, хотя сам метод тогда еще не имел названия, в программе «Logic Theorist» 1955 г. Программа GPS, воплощающая анализ средств и целей, появилась около 1957 г., но она совмещала этот анализ не с выбором первой лучшей вершины, а с модифицированным методом поиска в глубину. Шахматные программы как правило были связаны (по соображениям экономики памяти) с поиском в глубину, дополненным после 1958 г. мощной процедурой альфа-бета усечения. Похоже, что каждый из этих методов неоднократно изобретался заново, и трудно найти общие, не зависящие от конкретной задачи теоретические обсуждения решения задач в терминах этих концепций до середины или второй половины 60-х годов. Масштабы поддержки, которую они получили со стороны формализованной математической теории, все еще незначительны: некоторые теоремы о сокращении объема поиска, которое может быть обеспечено применением альфа-бета эвристики, пара теорем (рассмотренных в [4]) о поиске по кратчайшему пути, и несколько полученных в самое последнее время теорем о поиске с выбором лучшей вершины с применением вероятностной оценочной функции.

«Слабые» и «сильные» методы. Рассмотренные до сих пор методы ставили своей целью ограничение, а не предотвращение экспоненциального роста. По этой причине они были удачно названы «слабыми методами» — методами, которые следует применять, когда знания символьной системы или степень структурированности проблемного пространства недостаточны для того, чтобы полностью исключить поиск. Поучительно сравнить высокоструктурированную ситуацию, которая может быть выражена, скажем, как задача линейного программирования, с менее структурированными ситуациями комбинаторных задач вроде задачи о коммивояжере или составления расписаний. («Менее структурированная» здесь означает недостаточность или отсутствие теории, описывающей структуру проблемного пространства.) ,

При решении задач линейного программирования может потребоваться большой объем вычислений, но поиск не ветвится. Каждый шаг — это шаг на пути к решению. При решении комбинаторных задач или при доказательстве теорем поиска по

дереву редко удается избежать, и успех зависит от эвристических методов поиска описанного выше типа.

Не все направления решения задач в области ИИ следовали обрисованному выше пути. Пример несколько иного подхода дают работы по системам доказательства теорем. Здесь идеи, заимствованные из математики и логики, сильно повлияли на направление исследований. Например, использование эвристик встречало сопротивление в тех случаях, когда полнота не могла быть доказана (что несколько странно, так как про наиболее интересные математические системы известно, что они неразрешимы). Поскольку полнота редко может быть доказана для эвристических методов поиска с выбором первой лучшей вершины или для многих видов селективных генераторов, то результат этого требования был весьма разочаровывающим. Когда программы доказательства теорем стали постоянно терпеть неудачи из-за комбинаторных взрывов их деревьев поиска, разработчики обратились к селективным эвристикам, которые во многих случаях оказались аналогичными эвристикам, используемым в общих программах решения задач. Например, эвристика множества поддержки — это разновидность обратного поиска, приспособленного к проблемной среде доказательства теорем методом разрешения.

Итоги опыта работы. Мы описали, как действует наш второй качественный структурный принцип, утверждающий, что символьные системы решают задачи посредством эвристического поиска. Кроме того, мы изучили некоторые дополнительные свойства эвристического поиска, в частности, угрозу, что он всегда может столкнуться с экспоненциальным ростом дерева поиска, а также некоторые средства, которые он использует для борьбы с этой угрозой. По поводу того, насколько эвристический поиск был эффективен в качестве механизма решения задач, мнения различны и зависят от того, какие проблемные области рассматривались и какие критерии адекватности принимались. Успех гарантирован, если мы не слишком много хотим, а если хотеть слишком много, то гарантирован провал. Результаты практической работы с такими программами можно подытожить примерно так. Немногие программы решают задачи на профессиональном уровне экспертов. Наиболее известные примеры «экспертных» систем — программа для игры в шашки Сэмюэла и система DENDRAL Фейгенбаума и Ледерберга, но можно указать также многие программы эвристического поиска для таких проблемных областей, как составление расписаний и целочисленное программирование в исследовании операций. В ряде областей программы действуют на уровне компетентных непрофессионалов: шахматы, некоторые проблемные области доказательства теорем, многие

виды игр и головоломок. Уровень компетентности, свойственный людям, пока что недоступен программам, имеющим сложный перцептивный «вход»: распознавателям устной речи, анализаторам визуальных сцен, роботам, передвигающимся в реальном пространстве и времени. Тем не менее и в этих сложных проблемных ситуациях достигнуты впечатляющие успехи и накоплен большой опыт.

Пока что мы не располагаем глубокими теоретическими объяснениями, почему возникает именно такая картина качества работы программы для разных областей. Из эмпирических данных, однако, можно вывести два утверждения. Во-первых, из того, что нам известно о поведении людей-экспертов при выполнении ими заданий вроде игры в шахматы, можно с большой вероятностью предположить, что любая система, способная достичь такого уровня компетентности, должна иметь доступ к очень большим объемам семантической информации, хранимым в ее памяти. Во-вторых, превосходство человека в заданиях, связанных с большими объемами перцептивной информации, отчасти может быть приписано специализированным встроенным структурам параллельной обработки информации, присущим уху и глазу человека.

В любом случае качество работы систем должно обязательно зависеть и от свойств проблемных областей, и от свойств символьных систем, используемых для решения задач из этих областей. Для большинства интересующих нас областей реальной жизни структура пространства решений оказалась недостаточно простой, чтобы дать какие-либо теоремы о сложности или как-то иначе, нежели эмпирически, подсказать, насколько велики задачи реального мира по сравнению с возможностями наших символьных систем их решать. Эта ситуация может измениться, но пока это не произошло, мы должны полагаться на эмпирические исследования, применяя лучшие решатели задач, которые мы можем построить, в качестве основного источника знаний о масштабах и особенностях сложности задач. Даже в высокоструктурированных областях вроде линейного программирования теория была намного полезнее в укреплении эвристических подходов, лежащих в основе наиболее мощных алгоритмов решения таких задач, чем в углублении анализа их сложности.

ИНТЕЛЛЕКТ БЕЗ ОБШИРНОГО ПОИСКА

Наш анализ интеллектуальности приравнял ее к способности извлекать и использовать информацию о структуре проблемного пространства с целью сделать порождение решения задачи настолько быстрым и непосредственным, насколько это возмож-

но. Новые направления улучшения способности символьных систем решать задачи можно отождествить тем самым с новыми способами извлекать и использовать эту информацию. Можно выявить по крайней мере три таких способа.

Нелокальное использование информации. Во-первых, несколько исследователей заметили, что информация, собранная в процессе поиска по дереву, обычно используется лишь локально, чтобы помочь принять решения в конкретной вершине, в которой информация была порождена. Информация о шахматной позиции, полученная динамическим анализом поддерева продолжений, обычно используется для оценивания именно этой позиции, но не для оценивания других позиций, которые могут обладать многими сходными чертами. Поэтому одни и те же факты приходится неоднократно открывать заново в различных вершинах дерева поиска. Нельзя решить этой проблемы, просто взяв информацию из контекста, в котором она возникла, и использовав ее в более широком контексте, поскольку такая информация может оказаться истинной лишь для ограниченного круга ситуаций. За последние годы было предпринято несколько попыток перенести информацию из контекста, в котором она появилась, на другие подходящие контексты. Хотя пока рано судить о мощи этой идеи и о том, как именно ее следует осуществлять, она довольно многообещающа. Важное направление исследований, которое развито в работе [1], — применение анализа причин для определения множества ситуаций, в которых конкретная порция информации остается справедливой. Например, если слабость шахматной позиции может быть прослежена назад к тому ходу, в результате которого она возникла, то та же самая слабость может присутствовать и в других позициях, служащих продолжениями этого хода.

Система распознавания речи HEARSAY использует другой способ сделать информацию доступной глобально. Эта система пытается распознать фрагменты устной речи, осуществляя параллельный поиск на нескольких различных уровнях: фонетическом, лексическом, синтаксическом и семантическом. По мере того, как каждое из этих направлений поиска выдвигает и оценивает гипотезы, оно сообщает эту информацию общей «доске объявлений», которую могут читать все источники информации. Эта общая информация может быть использована, например, для отбрасывания гипотез или целого класса гипотез, которые в противном случае пришлось бы исследовать путем поиска одному из процессов. Итак, увеличение нашей способности использовать полученную поиском по дереву информацию нелокально обещает повысить интеллектуальность систем, решающих задачи.

Семантические распознающие системы. Вторая активно изу-

чаемая возможность повысить интеллектуальность — снабдить символьную систему обширной семантической информацией о проблемной области, с которой она имеет дело. Например, эмпирическое исследование силы игры шахматистов-мастеров показывает, что главным источником силы мастера является хранимая в его памяти информация, позволяющая ему распознать большое количество характерных особенностей позиций на доске и их комбинаций, а также информация, использующая это узнавание, чтобы предложить действия, подходящие для распознанных особенностей позиции. Эта общая идея была, разумеется, включена в шахматные программы почти с самого начала. Новым было осознание числа таких комбинаций и связанной с ними информации, которую, возможно, придется хранить для обеспечения игры на уровне мастеров: это примерно 50 000 комбинаций.

Возможность замены поиска распознаванием связана с тем, что некоторая, и особенно редкая, конфигурация может содержать огромный объем информации, если она тесно связана со структурой проблемного пространства. Когда эта структура «нерегулярна» и не допускает простого математического описания, тогда знание большого числа связанных с ней конфигураций может оказаться ключом к интеллектуальному поведению. Так это или нет для некой конкретной проблемной области — вопрос, который проще решить эмпирическим исследованием, чем теоретически. Наш опыт работы с символыми системами, щедро наделенными семантической информацией и способностями к распознаванию образов для ее оценивания, все еще чрезвычайно ограничен.

Выше обсуждалась семантическая информация, связанная именно с системами распознавания образов. Конечно, существует также обширная область исследований по искусственно-му интеллекту, относящихся к обработке семантической информации и организации семантической информации посредством семантических регистров памяти; эти исследования выходят за рамки обсуждаемых здесь тем.

Выбор подходящих представлений. Третье направление исследований связано с возможностью сократить объем поиска или вовсе избежать его, подходящим образом выбрав проблемное пространство. Стандартный пример, ярко иллюстрирующий эту возможность, это задача об изуродованной шахматной доске. Обычная 64-клеточная шахматная доска может быть замощена в точности 32 прямоугольниками, каждый из которых закрывает ровно две клетки. Предположим теперь, что мы вырезали две клетки, находящиеся на противоположных концах большой диагонали. Можно ли эту изуродованную доску замостить в точности 31 прямоугольником? При буквально ангель-

ском терпении невозможность это сделать можно было бы доказать, перепробовав все возможные замощения. Другой способ, пригодный для менее терпеливых и более умных, состоит в том, чтобы заметить: противоположные (по диагонали) углы шахматной доски имеют одинаковый цвет. Следовательно, у изуродованной доски клеток одного цвета на две меньше, чем клеток другого цвета. Но каждый прямоугольник покрывает одну клетку одного цвета и одну клетку другого, а любой набор прямоугольников должен покрывать одинаковое число клеток каждого цвета. Поэтому решение невозможно. Как может символьная система открыть это простое индуктивное соображение вместо того, чтобы безнадежно пытаться решить задачу исчерпывающим поиском среди всех возможных замощений? Следовало бы удостоить систему, сумевшую найти это решение, высокой отметки за сообразительность.

Возможно, однако, что постановка этой задачи не освобождает нас от процессов поиска. Мы просто перенесли поиск из пространства возможных решений задачи в пространство возможных представлений. В любом случае весь процесс перехода от одного представления к другому и отыскания и оценивания представлений в значительной степени является неисследованной областью для решения задач посредством поиска. Качественные структурные принципы, управляющие представлениями, еще предстоит открыть. Поиск таких принципов почти наверняка привлечет большое внимание в предстоящее десятилетие.

ЗАКЛЮЧЕНИЕ

Подведем итоги нашего обзора связи между символыми системами и интеллектом. Путь от «Менона» Платона к современным взглядам был долгим, но, вероятно, можно считать обнадеживающим, что большая часть успехов на этом пути была достигнута в двадцатом веке, а значительная часть этой части — начиная с его середины. Мышление оставалось совершенно непонятным и необъяснимым до тех пор, пока современная формальная логика не предложила интерпретировать его как манипуляцию формальными символами. Казалось, что оно по-прежнему обитает в основном на небе отвлеченных платоновых идей или в столь же таинственных мирах человеческого сознания, пока компьютеры не научили нас, как машины могут обрабатывать символы. А. М. Тьюринг, памяти которого посвящено наше сегодняшнее заседание, сделал свой огромный вклад в эту проблематику как раз в середине века, на перекрестке линий развития, ведущих от формальной логики к компьютеру.

Физические символные системы. Изучение логики и компьютеров открыло нам, что интеллект основан на физических символных системах. Это наиболее фундаментальный качественный структурный принцип в информатике.

Символьные системы являются совокупностями конфигураций и процессов, причем последние могут создавать, разрушать и модифицировать первые. Наиболее важные свойства конфигураций — то, что они могут обозначать объекты, процессы или другие конфигурации, а также то, что когда они обозначают процессы, их можно интерпретировать. Интерпретация означает выполнение обозначаемых процессов. Два самых важных класса символьных систем, которые нам известны, это люди и компьютеры.

Наше современное понимание символьных систем прошло, как отмечалось выше, несколько последовательных этапов в своем развитии. Формальная логика приучила нас к символам, рассматриваемым синтаксически, как сырой материал для мышления, а также к идеи манипулирования ими согласно тщательно определенным формальным процессам. Машина Тьюринга сделала синтаксическую обработку символов по-настоящему механической и подтвердила потенциальную универсальность строго определенных символьных систем. Понятие хранимой программы для компьютеров подтвердило интерпретируемость символов, уже неявно присутствующую в идее машины Тьюринга. Обработка списков выдвинула на первый план денотационные возможности символов и определила обработку символов такими способами, которые допускали независимость от фиксированной структуры физического устройства (машины). К 1956 г. все эти концепции были уже доступны, как и оборудование для их реализации. Могло начаться изучение интеллектуальности символьных систем — предмет науки об искусственном интеллекте.

Эвристический поиск. Второй качественный структурный принцип в искусственном интеллекте — то, что символьные системы решают задачи, порождая возможные решения и проверяя их, т. е. посредством поиска. Обычно решения отыскиваются путем создания символьных выражений и их последовательной модификации до тех пор, пока они не будут удовлетворять условиям, отвечающим решению. Итак, символьные системы решают задачи с помощью поиска. Поскольку их ресурсы ограничены, поиск не может быть выполнен мгновенно, а должен быть последовательным. Это понятие охватывает и одиночный путь от исходной точки до цели, и, если необходимы уточнения и возвраты, целое дерево таких путей.

Символьные системы не могут выглядеть интеллектуальными, когда их окружает сплошной хаос. Они проявляют интел-

лектуальность, извлекая информацию из проблемной области и используя эту информацию для направления поиска, избегая ложных поворотов и блуждания по кругу. Проблемная область должна содержать информацию, т. е. обладать некоторой упорядоченностью и структурированностью, чтобы этот метод мог работать. Парадокс из платоновского диалога «Менон» разрешается тем наблюдением, что эта информация может запоминаться, но новая информация также может извлекаться из области, обозначаемой символами. В обоих случаях первоисточник этой информации — предметная область, из которой берутся задачи.

Эмпирические основания. Исследования по искусственному интеллекту стремятся выяснить, как должны быть устроены символические системы, чтобы их действия были интеллектуальными. За двадцать лет работы в этой области накопился большой объем знаний; которых хватило бы на несколько книг (и такие книги уже написаны); большей частью это весьма конкретные эмпирические данные о поведении разных типов символьных систем при решении задач из разных предметных областей. На основании этого опыта появилось, однако, и несколько обобщений, не ограниченных предметной областью или типом символьной системы, но относящихся к общим свойствам интеллекта и способам его реализации.

В настоящей лекции мы попытались сформулировать некоторые из этих обобщений. В основном они качественные, а не математические. Они больше похожи на теоретические обобщения, известные для геологии или биологии развития, чем на те, которыми оперирует теоретическая физика. Они достаточно сильны, чтобы позволить нам сегодня проектировать и строить умеренно интеллектуальные системы для широкого спектра проблемных областей, а также добиться весьма глубокого понимания того, как человеческий разум работает во многих ситуациях.

Что дальше? В нашем докладе мы упомянули известные нерешенные вопросы, а также поставили новые; и тех, и других довольно много. Мы не наблюдаем уменьшения энтузиазма исследований, который был характерен для этой области в течение последних 25 лет. Два ресурсных ограничения будут определять скорость прогресса в ней в течение такого же периода в будущем. Одно из них — доступные вычислительные мощности. Второе, и, вероятно, более важное, — число талантливых молодых специалистов по информатике, избирающих эту область как наиболее привлекательную и трудную из всех, им открытых.

А. М. Тьюринг закончил свою знаменитую работу «Вычислительные машины и интеллект» [5] словами:

«Мы можем заглянуть вперед лишь недалеко, но мы видим там многое, что должно быть сделано».

Многое из того, что Тьюринг в 1950 г. считал требующим осуществления, уже сделано, но в нынешней повестке дня вопросов, требующих ответа, не меньше, чем всегда. Возможно, мы вкладываем в простое утверждение, процитированное выше, более глубокий смысл, чем его автор, но нам хотелось бы думать, что в этом высказывании Тьюринг признал фундаментальную истину, которую все специалисты по информатике интуитивно знают: для всех физических символьных систем, которые, подобно всем нам, обречены на последовательный поиск в проблемной ситуации, решающий вопрос всегда один и тот же — что делать дальше?

БЛАГОДАРНОСТЬ

Исследования авторов на протяжении многих лет поддерживались отчасти Агентством перспективных исследований Министерства обороны (руководимым Отделом научных исследований ВВС), а отчасти Национальным институтом психиатрии.

ЛИТЕРАТУРА

1. Berliner H. Chess as problem: the development of a tactics analyzer. Ph. D. Th., Computer Sci. Dep., Carnegie-Mellon U. (1975).
2. McCarthy J. Recursive functions of symbolic expressions and their computation by machine. Comm. ACM 3, 4 (April 1960), 184—195.
3. McCulloch W. S. What is a number, that a man may know it, and a man, that he may know a number. General Semantics Bulletin, Nos. 26 and 27 (1961), 7—18.
4. Nilsson N. J. Problem Solving Methods in Artificial Intelligence. McGraw-Hill, New York, 1971.
5. Turing A. M. Computing machinery and intelligence. Mind 59 (Oct. 1950), 433—460.

ПОСТСКРИПТУМ

РАЗМЫШЛЕНИЯ О ДЕСЯТОЙ ТЬЮРИНГОВСКОЙ ЛЕКЦИИ:
ИНФОРМАТИКА КАК ЭМПИРИЧЕСКОЕ ИССЛЕДОВАНИЕ —
СИМВОЛЫ И ПОИСК

Аллен Ньюэлл и Херберт А. Саймон
 Факультет информатики
 Отделение психологии
 Университет Карнеги — Меллона
 Питтсбург, Пенсильвания, 15213

Наша Тьюринговская лекция была прочитана в 1975 г., через двадцать лет после начала исследований по искусственноому интеллекту в середине 50-х годов. Сейчас прошло еще десять лет. В этой лекции мы в основном

воздерживались от пророчеств и постановки задач на будущее, предпочтя вместо этого обосновать некую истину: информатика есть опытная наука. Мы сделали это, проследив историческое развитие двух общих принципов, лежащих в основе интеллектуального поведения: необходимости физических символьных систем и поиска. Можно теперь поинтересоваться, укрепило или ослабило прошедшее десятилетие позицию и оценки, выдвинутые тогда.

Многое случилось за эти десять лет. И информатика, и искусственный интеллект продолжали развиваться в научном, техническом и экономическом отношениях. Явно не высказанное, но подразумеваемое положение этой лекции состояло в том, что искусственный интеллект является частью информатики, причем обе эти дисциплины выросли на одной и той же идейной основе. В этом плане общественное развитие продолжало следовать логике (оно не всегда это делает), и искусственный интеллект продолжает оставаться частью информатики. Если и можно говорить о каких-то изменениях, то их отношения становятся все теснее по мере того, как приложения интеллектуальных систем к разработке программного обеспечения под видом экспертных систем становятся все более привлекательными.

Основное утверждение об эмпирическом исследовании отразилось во всех разделах информатики. Многое произошло повсюду, но сосредоточимся на искусственном интеллекте. Взрыв новых работ по экспертным системам, развитие исследований по обучающимся системам и разработки в области интеллектуального обучения служат важными примерами направлений, принесших крупные результаты с тех пор, как лекция была прочитана (и никак не отраженных в ней). Все они в сильнейшей степени руководствовались эмпирическими исследованиями. Даже появление работ по логическому программированию, которое в наибольшей степени связано с формальными процедурами и доказательством теорем, обязано значительной частью своей жизнеспособности превращению в программистскую деятельность — а в ней решающую роль играет практический опыт.

В развитии теории были, разумеется значительные достижения. Особенно связанными с содержанием нашей лекции были работы по анализу сложности эвристического поиска, например недавно опубликованная книга Пирла [10]. Но это также иллюстрирует обычный заколдованный круг науки, в котором теория наконец начинает расти после того, как накопится достаточно проанализированного опыта. Нам все еще чего-то не хватает для того, чтобы замкнуть этот круг доведением теории до такого состояния, когда она обеспечивает рутинный подход к планированию дальнейших опытов, и с этого момента опытные данные и теория могут развиваться рука об руку. Это время несомненно придет, хотя пока мы до него немного не дожили.

Мы выбрали эту аудиторию, чтобы высказать два фундаментальных принципа (о символах и поиске), которые представляются нам общеизвестными в практике и понимании искусственного интеллекта, но их истинная роль еще недостаточно осознана. На самом деле это подлинный фундамент всякого интеллектуального действия. История развития этих двух принципов за последнее десятилетие была несколько различной, но оба, как нам кажется, изменились в тех направлениях, которые подтверждают их справедливость по существу.

Выдвижение на первый план гипотезы о физических символьных системах доказало свою полезность в области искусственного интеллекта, хотя мы впоследствии действительно признали разумным сформулировать эту гипотезу более подробно [7]. Эта гипотеза была принята в довольно общем смысле для выражения того взгляда на сознание, который возник в связи с появлением компьютера. Однако это не означает, что мы считаем такой подход единственным возможным. Существуют интеллектуальные позиции, отвергающие любые вычислительные подходы к сознанию и считающие упомянутую гипотезу несомненно ложной [3, 11]. Для нас важнее две другие позиции. Одна из них встречается среди философов, многие из которых верят, что

главная проблема семантики или интенциональности — как символы обозначают объекты внешнего мира — не имеет отношения к физическим символным системам. Другая позиция встречается среди коннекционистов из специалистов по искусственному интеллекту и психологии познания, которые верят, что существуют формы организации обработки данных (реализуемые в виде нейронных сетей, перцепtronов или реальных нервных систем), которые могут делать все, на что способны символные системы, но в них невозможно обнаружить никаких сущностей, отождествимых с символами. В обоих случаях необходимы, несомненно, дополнительные исследования, и они, конечно, будут проведены. Как бы то ни было, потенциальные возможности символов по-прежнему представляются нам очень широкими, так что мы делаем ставку на гипотезу о символных системах.

Заслуживает упоминания то развитие, которое получила гипотеза о физических символных системах. В практике информатики и искусственного интеллекта принято описывать системы в терминах знания, которым они обладают, понимая под этим, что существуют механизмы обработки данных, вынуждающие систему вести себя так, как если бы она могла использовать это знание для достижения целей, которым система должна служить. Эта практика распространяется на проектирование, в котором задание знания, которым система должна обладать, означает задание механизмов, которые должны быть построены. Мы воспользовались возможностью, аналогичной той, которую предоставила нам Тьюринговская лекция, а именно обращение президента Американской ассоциации искусственного интеллекта (AAAI), чтобы также выразить эту практику в точных терминах [8]. Мы определили другой уровень компьютерной системы, расположенный над уровнем символов и названный нами уровнем знания. Это весьма точно соответствует тому, что в философии Дан Деннетт назвал интенциональной установкой [2]. Корни такого подхода лежат, конечно, в той замечательной особенности адаптивных систем, что их поведение полностью определяется проблемной средой, а это маскирует природу их внутренних механизмов [9, 12]. И снова мотивы, побудившие нас определить структурный уровень организации системы, который мы назвали уровнем знания, были теми же, что и в Тьюринговской лекции — выразить то, что знает каждый хороший специалист-практик в области информатики, но в такой форме, которая допускает дальнейший технический прогресс. Появились первые признаки того, что такой прогресс для уровня знания уже начинается [6].

Возвращаясь ко второй гипотезе (об эвристическом поиске), напомним, что признание важности такого поиска было явным и повсеместным в первые годы исследований по искусственному интеллекту. Нашей целью в Тьюринговской лекции было стремление подчеркнуть, что этот поиск является неотъемлемой и существенной компонентой всякого интеллектуального поведения, а не просто одним интересным механизмом среди многих других. Случилось так, что 1975 г., когда эта лекция была прочитана, как раз предшествовал расцвету той точки зрения, что знание имеет решающее значение для интеллектуального поведения. Эта тенденция нарастала с начала 70-х годов. Симптомом этого нового подхода стало возникновение новой области экспертных систем, а также новая роль инженера знаний [4]. В своих крайних проявлениях это новое движение дошло до утверждения, будто в искусственном интеллекте произошла смена парадигмы, приведшая к тому, что поиск в конце концов был отброшен, и новым ведущим принципом ИИ一下子 становится знание [5].

Другая интерпретация этих перемен (та, которой придерживаемся мы) состоит в том, что никакой революции не произошло, а имел место обычный цикл аккомодации, ассимиляции и достижения равновесия, который Пиаже описывает как нормальный процесс развития понимания (хотя он говорил о развитии ребенка, а не о становлении науки). Наука развивается, расширяя каждую новую грань понимания по мере ее возникновения —

она приспособливается (аккомодация) к новому пониманию посредством напряженного стремления усвоить (ассимиляция) это понимание. Конец 70-х и начало 80-х годов были посвящены изучению того, что означает для системы обладание достаточным знанием о ее проблемной области — достаточным для того, чтобы обойтись без обширного поиска в проблемном пространстве, но все же при этом решать задачи, требующие интеллекта, а не просто реализации небольших (по объему вычислительной работы) алгоритмов. (Когда объем знаний возрастает, эти системы, конечно, начинают требовать поиска правил в базе знаний.) Соответственно задачи, решаемые этими системами, хотя они и были взяты из реального мира, отличались также небольшой интеллектуальной (т. е. требующей логического вывода) сложностью. Роль поиска в трудных интеллектуально задачах оставалась очевидной для тех, кто продолжал работать над программами с целью их завершения; этого трудно избежать, когда угроза комбинаторного взрыва маячит за каждым углом. Усвоив теперь некоторые из механизмов, позволяющих системам обладать значительными по объему базами знаний, специалисты по искусственному интеллекту, похоже, пришли к пониманию, что и поиск, и знания играют существенную роль.

Последнее наше замечание касается шахматной игры, которая неоднократно упоминалась на протяжении всей лекции, давая (как она всегда это делает) прекрасные иллюстрации многих ее положений. Достигнутый за десять лет прогресс очевиден: шахматный компьютер Hitech [1] сравнялся по силе игры с лучшими мастерами (его рейтинг 2340, тогда как у мастеров этот показатель варьирует от 2200 до 2400). Он продолжает расти, хотя никто не знает, как долго еще он будет подниматься. Система Hitech сама по себе иллюстрирует много интересных утверждений. Во-первых, она снова подтвердила роль эвристического поиска. Во-вторых, она построена на обширном поиске (200 000 позиций в секунду), так что она показывает движение именно в том направлении, которое в той лекции мы сочли ошибочным. Любопытно оказаться неправым, когда речь идет о новом научном знании. Но третий, самый важный теоретический урок, который можно извлечь из этой системы, это как раз то, что подчеркивалось в лекции, а именно: интеллектуальное поведение включает в себя сложное взаимодействие знания, полученного посредством поиска, и знания, полученного из хранимой в памяти структуры распознавания. Последние 200 очков повышения рейтинга системы Hitech — и победы, сделавшие ее знаменитой, — целиком обусловлены добавлением знания к машине с фиксированными, хотя и обширными возможностями поиска. Четвертое и последнее: поразительное мастерство системы Hitech и новые явления, которые она породила, являются еще одним свидетельством, если такие еще требуются, что прогресс в информатике и искусственном интеллекте происходит благодаря эмпирическим исследованиям.

ЛИТЕРАТУРА

1. Berliner H., Ebeling C. The SUPREM architecture: A new intelligent paradigm.
2. Dennet D. C. Brainstorms. Bradford/MIT Press, Cambridge, Mass., 1978.
3. Dreyfus H. L. What Computers Can't Do: A Critique of Artificial Reason, 2nd ed. Harper and Row, New York, 1979.
4. Feigenbaum E. A. The art of artificial intelligence: Themes and case studies in knowledge engineering. In: Proceedings of the 5th International Joint Conference on Artificial Intelligence. Computer Science Dept., Carnegie-Mellon Univ., Pittsburgh, Pa., 1977.
5. Goldstein I., Papert S. Artificial intelligence, language and the study of knowledge. Cognitive Sci., 1 (1977), 84—124.

6. Levesque H. J. Foundations of a functional approach to knowledge representation. *Artif. Intell.* 23 (1984), 155—212.
7. Newell A. Physical symbol systems. *Cognitive Sci.* 4 (1980), 135—183.
8. Newell A. The knowledge level. *Artif. Intell.* 18 (1982), 87—127.
9. Newell A., Simon H. A. *Human Problem Solving*. Prentice-Hall, Englewood Cliffs, N. J., 1972.
10. Pearl J. *Heuristics: Intelligent Search Strategies for Computer Problem Solving*. Addison-Wesley, Reading, Mass., 1984.
11. Searle J. Minds, brains and programs. *Behav. Brain Sci.* 3 (1980), 417—457.
12. Simon H. A. *The Sciences of the Artificial*. MIT Press, Cambridge, Mass., 1969.