ФАНТАЗИЯ
или
НАУКА

на пути
к искусственному интеллекту

ДА ПОСПЕЛОВ

ФАНТАЗИЯ
ИЛИ
НАУКА

на пути
к искусственному
интеллекту

МОСКВА "НАУКА"
ГЛАВНАЯ РЕДАКЦИЯ
ФИЗИКО-МАТЕМАТИЧЕСКОЙ ЛИТЕРАТУРЫ
1982

32.81
П 62
УДК 62-50

Поспелов Д. А.

П. 62. Фантазия или наука: на пути к искусственному интеллекту—М.: Наука. Главная редакция физико-математической литературы, 1982.— 224 с, 15 илл.— 35 коп.

В книге рассматривается становление идей, приведших к возникновению систем искусственного интеллекта — устройств, способны* осуществлять некоторые виды творческой деятельности Обсуждаются фундаментальные проблемы, составляющие ядре этой новой науки, и дается прогноз ее ближайшего развития. В простой и доступной форме излагаются основные результаты теории интеллектуальных систем и научных проблем в области интегральных роботов. Книга предназначена для широкого круга читателей, интересующихся современным развитием кибернетики: школьников старших классов, преподавателей, студентов, лекторов, а также специалистов в тех областях, где внедряются современные ЭВМ.

1502000000 — 114 **ББК 32,81**
П----------------------168-82
053 **(02)**-82 **6Ф0.1**

п**1502000000-Ш**
053(02)-82

НЕСКОЛЬКО ВВОДНЫХ СЛОВ

— *Кролик — он умный!*— сказал *Пух в раздумье.*
— *Да,*— сказал *Пятачок,— Кролик — он хитрый!*
— *У него настоящие Мозги.*
— Да,— *сказал Пятачок,— у Кролика настоящие Мозги.*
Наступило долгое молчание.
— *Наверно, поэтому,*— сказал, *наконец. Пух,— наверно, поэтому-то он* никогда ничего *не понимает!*

А. А. МИЛН
Винни-Пух и все-все-все

Казалось бы, книгу об искусственном интеллекте и интеллектуальных системах нужно было начинать с определения этих понятий, с точного очерчивания области, отведенной для этой новейшей науки. Но именно это невозможно пока сделать. Специалисты до сих пор не могут предложить подобных определений. И зыбки границы тех исследований в области интеллектуальных систем, которые отделяют их от собственно кибернетических исследований, исследований в области программирования, вычислительных машин, математической психологии и структурной лингвистики. Где-то на стыке многих научных дисциплин происходит в наши дни вычленение нового научного направления со своим предметом исследования, своими специфическими методами и мировоззрением.

Поэтому автор вынужден вместо точных и ясных определений избрать менее эффективный, но длительный путь раскрытия содержания основных понятий с помощью примеров и умозрительных рассуждений. Отсюда и само построение книги. От начальной эпохи развития инженерной мысли и формальной логики — к четкому пониманию революционного воздействия кибернетики и вычислительных машин на проблему автоматизации интеллектуальных процессов, к современным робототехническим устройствам. Только так, в исторической перспективе, возможно постепенное описание основных концепций исследований в области искусственного интеллекта. Отдельные ручейки наших знаний об имитации человеческих возможностей, слабые и петляющие вначале, постепенно сливаются и превращаются на наших глазах в широкую полноводную реку, конечная цель которой все яснее виднеется на горизонте.

За время, прошедшее после первой международной конференции по проблемам искусственного интеллекта, состоявшейся в США в 1969 году, сделан огромный шаг вперед в слиянии разнородных исследований в этой области в единое научное направление. Одновременно специалисты, занимающиеся интеллектуальными системами, сумели исключить ряд направлений из сферы своих интересов.

Поскольку основная задача автора состояла в показе процесса формирования основных идей и методов искусственного интеллекта, то в книге мало внимания уделяется описанию тех или иных конкретных программ и- устройств, имитирующих интеллектуальную деятельность человека. Это задача других книг.

Исходя из характера книги в тексте не даются ссылки на использованные источники, что сделало бы изложение слишком громоздким и наукообразным.

Первая глава книги читается легче второй, а вторая — легче третьей. И это не случайно, ибо для уяснения сути научных проблем искусственного интеллекта необходим хотя бы минимальный набор знаний, традиционно не входящих в общеобразовательные курсы. Поэтому в книге в самых малых дозах эти знания вводятся. Иначе получилась бы не научно-популярная книга, а просто популярная, писать которую автор не берется. Тем же, кто «все это знает» и интересуется лишь современными проблемами интеллектуальных систем, можно с легким сердцем посоветовать пропустить первые две главы книги и начать чтение прямо с третьей.

Вот, пожалуй, и все, что хотел сказать автор перед тем, как пригласить читателя пройти с ним от истоков исследований в области интеллектуальных систем к их бурному расцвету в настоящее время и к попытке обрисовать будущее этой новой науки.

ГЛАВА ПЕРВАЯ
ПРЕДЫСТОРИЯ

— Вы *знаете отличительные особенности* животного, *называемого раком?*
— *Да, конечно. Рак — это рыба красного цвета, которая ходит боком.*
— *Ну, что ж. Я вижу, что вы кое-что знаете о раке...*

Студенческий фольклор

1.1. «Искусственные люди»

Подвиг Фомы. Начнем с двух легенд, дошедших до нас из смутных далей средневековья. Но, подобно сказкам, преданиям и мифам, легенды никогда не возникают на пустом месте. В основе их лежат какие-то реалии, и лишь фантастическое преломление этих реалий, может быть, просто не понятых современниками, придает легендам оттенок необычности и сказочности.

Кёльн середины XIII столетия. Время богословских споров, окончательного оформления канонов. Увлечение алхимией и астрологией, в недрах которых начинает постепенно зарождаться научное мировоззрение будущего. Центр по изучению теологии в Кёльне — прообраз будущих научных учреждений. Здесь есть и теоретики-схоласты, спорящие о самых незначительных — на. наш взгляд, практиков,— вещах. Среди них совсем еще молодой монах Фома Аквинский (Аквинат), который вскоре- прославится своими богословскими трактатами и станет основателем целой философской системы, лежащей в основе современного мировоззрения католической церкви. Здесь же и широко образованный Альберт фон Больштедт, прозванный еще при жизни Великим, ибо велика была ученость этого монаха-доминиканца. Был он не только теоретиком, но и прекрасным практиком и изобретателем. Став учителем Фомы Аквинского еще в Парижском университете, Альберт фон Больштедт взял его с собой в Кёльн в качестве сотрудника и помощника.

Не было, казалось, более противоположных по внешнему и внутреннему складу людей. Подвижный, сухощавый и низкорослый Альберт и медлительный, чрезмерно полный и рослый молчаливый Фома. Недаром в Париже его прозвали Немым Быком. Но по закону о сходстве противоположностей между ними возникла и укреплялась дружба.

Наступил момент, когда Альберт пригласил Фому в свою тайную мастерскую, в которой, уединившись от всех, создавал он хитроумные устройства и механизмы. Вечерело, когда Фома постучал в дверь мастерской, Альберт впустил его. При тусклом свете, падавшем из узкого окна, Фома увидел вдруг женщину, тело которой светилось каким-то неземным светом. Она подняла руку, повернула к нему голову и вежливо поздоровалась. Но было в ней что-то страшное и неживое. Воспитанный в благочестии и почитании бога, Фома сразу же понял, что это дьявольское наваждение. И недаром его прозвали Быком. Не был трусом Фома Аквинский. Тростью, с которой обычно ходил по улицам Кёльна, в мгновение ока разбил он «порождение дьявола», шепча молитву и ограждая себя крестным знамением.

Альберт потерял те несколько мгновений, которые понадобились Фоме для уничтожения лучшего инженерного творения XIII века — человекоподобной куклы, которая умела ходить, двигать руками и говорить. С болью и гневом закричал Альберт: «Фома, Фома, что ты наделал? Ты уничтожил мою тридцатилетнюю работу!».

Мы не знаем, как была устроена кукла Альберта Великого. История не сохранила нам ее описания. Но вера в то, что Альберту Великому удалось с помощью дьявола создать искусственного человека, оказалась живучей и послужила созданию многих иных легенд об этом действительно выдающемся ученом и инженере средневековья.

Так кончается первая из легенд. Отметим только, что Фома, увидев куклу, принял ее за живого человека. И лишь в следующее мгновенье, обнаружив в ее движениях и голосе нечеловеческие элементы, понял, что перед ним — «порождение дьявола». В каком-то

смысле Альберт Великий должен был быть доволен: его искусственное творение было принято за живое.

Робот из глины. Часы в центре Старого города Праги знамениты на весь мир. Вот уже много столетий Смерть дергает колокол и -пересыпает песок в песочных часах, музыканты играют, чередой проходят апостолы и кричит петух, возвещая о конце маленькой мистерии. Толпы туристов всегда с нетерпением ждут этого зрелища. Но. наша цель — не эти часы. Нас интересует нечто другое.

Если от Староместской площади Праги пройти через узкий проход мимо храма Святого Николая, то вскоре можно выйти к зданию Новой Ратуши. Построено это здание в начале нашего века в модном тогда стиле Сецессион. Известный чешский скульптор того времени Л. Шалоун украсил . здание скульптурами. Присмотримся к одной из них, стоящей в неглубокой нише на углу здания. Она изображает глубокого старика с длинными волосами, седыми прядями ниспадающими ему на грудь. Он одет в средневековую одежду раввина. Мудро и безучастно смотрит он на прильнувшую к нему молодую обнаженную женщину. И правая рука раввина готовится оттолкнуть ее. А к ногам старца жмется шелудивый и голодный пес.

А недалеко от этой скульптуры на древнейшем из сохранившихся в Европе кладбищ можно найти могильный камень с надписью, говорящей о том, что под ним похоронен раввин Иегуда Лев бен Безалел. Именно он и изображен Л. Шалоуном. Имя раввина окружено множеством легенд. Но для нас интересна одна из них.

У раввина была очень сварливая Жена. Она часто упрекала его, что занятия отнимают все его время. А дома так много дел, что его помощь просто необходима. И лучше бы он помогал своей замученной жене, а не тратил бы время на пустые выдумки. По-видимому, даже мудрости, приписываемой Иегуде Льву бен Безалелу, не хватало, чтобы потушить семейный пожар. И тогда он принял решение. Из глины он создал человекоподобного великана, получившего имя Голем. Стоило только вложить ему в рот специальный шарик с магическими формулами — шем, как Голем оживал и выполнял все приказания хозяина и тех лиц, которых его создатель ему указывал.

Теперь бен Безалел был спокоен. Утром, оживив Голема и передав его жене, он мог спокойно заниматься своими делами. А Голем таскал воду и хворост, мыл посуду и выполнял еще сотню дел, так необходимых в домашнем хозяйстве. Для того чтобы «выключить» Голема, достаточно было вынуть шем. Как только это происходило, Голем прекращал свою деятельность и застывал неподвижно. Голем трудился исправно. Лишь одно ограничение было для него. Правоверный раввин не мог позволить Голему работать в субботу. Поэтому вечером в пятницу Иегуда Лев бен Безалел всегда вынимал шем. Но однажды он забыл сделать это. И в субботу Голем стал неистовствовать. Он полностью вышел из повиновения и стал разрушать все, что попадалось на пути. С большим трудом удалось бен Безалелу вынуть у него шем, после чего Голем, недвижимый, рухнул на пол. Его отнесли на чердак старой синагоги, где он постепенно рассыпался в прах. До сих пор эта старая синагога, называемая Староновой, стоит недалеко от тех мест, где когда-то разгуливал глиняный великан Голем.

В этой легенде, очень популярной среди чехов и еврейского населения Праги и существующей в большом количестве версий, неоднократно подчеркивается, что Голем являл собой копию человека, отличаясь от него лишь ростом. У него, как и у человека, есть имя. Его называли Йозеф. В некоторых версиях легенды Голем не только домашний работник, но и охранитель населения от притеснителей.

Гомункулус*). Идея о возможности создания подобия человека, поведение которого ничем бы не отличалось от человеческого, в условиях средневековья представлялась весьма еретической. Вся религиозная догматика, все мировоззрение христианской церкви в корне Противоречили возможности искусственного повторения того, что когда-то было сотворено богом. И тем не менее эта идея все время «висела в воздухе». Теология и естествознание, магия и экспериментальные исследования причудливо переплетались в работах средневековых ученых. В трудах выдающегося врача

*) По представлениям средневековых алхимиков, гомункулус — это существо,-подобное человеку, которое якобы можно получить искусственно (в пробирке),

и естествоиспытателя Парацельса (настоящее его имя Филипп Ауреол Теофраст Бомбаст фон Гогенгейм), жившего в первой половине XVI века, содержится рецептура изготовления полного подобия человека с помощью серии химических реакций, производимых над спермой человека. Кстати, Парацельс, как и упоминавшийся нами Альберт фон Больштедт, был мона-. хом и совмещал свои научные изыскания с теологией. Идея гомункулуса долгие годы занимала мысли алхимиков и ученых. От Парацельса до современного итальянского физиолога Петруччио — таков путь этой идеи. Для нас Же интересно широкое распространение веры в то, что вполне возможно создание существа, подобного человеку, путем, отличным от того, который избрала природа.

Андроиды. Эта идея в XVIII веке нашла свое воплощение в механических людях — андроидах. К этому времени достигла своего расцвета технология изготовления механических часовых механизмов. Именно часовщики, овладевшие тайнами своего ремесла, стали изготовлять андроидов. Таких попыток зафиксировано множество. Наиболее известными являются творения французского механика Жака де Вокансона и швейцарских часовщиков Пьера-Жака Дро и его сына Анри Дро.

Среди моделей Вокансона наиболее интересен для нас флейтист (игрушка с часовым механизмом). Внешне он выглядел как обычный человек. Когда он брал настоящую флейту, подносил ее к губам и, перебирая пальцами, играл одну из одиннадцати мелодий, то можно было подумать, что перед нами не механизм, а живой человек.

В 1774 году на выставке в Париже демонстрировались три андроида Пьера-Жака и Анри Дро: писец, рисовальщик и музыкантша. Писец, внешне походивший на шестилетнего ребенка, старательно макал гусиное перо в чернила, писал на бумаге отдельные слова и целые фразы и аккуратно стряхивал избыток чернил, цопавших на перо. Ходили даже слухи, что пишет вовсе не механическая кукла, а дочь Анри Дро, хотя, среди фраз, написанных писцом, была и такая: «Мы —андроиды». «Человечность» писца достигалась тем, что в. процессе письма кукла двигала головой и туловищем, имитируя человеческие движения во время подобной деятельности. Кроме писца в семье андроидов были, как уже сказано, рисовальщик, под рукой которого возникали различные рисунки пером, и музыкантша, исполнявшая определенный набор музыкальных произведений на фисгармонии. Как и писец, эти куклы не просто выполняли свою работу, но и сопровождали ее движениями глаз и мышц тела, создавая полную иллюзию, что перед зрителем находится не механическое устройство, а живой человек. Отметим, что немало простодушных зрителей XVIII века было уверено, что дело тут «нечисто». Слухи об андроидах постепенно приобретали среди людей, их не видевших, характер сенсационности и таинственности.

В «Песочном человеке» известного немецкого писателя Э. Т. А. Гофмана описывается андроид — женщина по имени Олимпия, которую никак нельзя было отличить от живого человека. И в оживающем Щелкунчике того же автора явно видны следы влияния создания андроидов на идею о возможности «оживить» неодушевленную материю.

Известно около двух десятков андроидов XVIII и начала XIX века. Одни из них прославились далеко за пределами родины их создателей, другие сыграли более скромную роль, как, например, «писец» придворного австрийского механика Фридриха Кнаусса или «трубач» немца Михаила Кауфмана. Для нас интересно проследить судьбу еще одного андроида, хотя многие детали, связанные с его историей, до сих пор остаются невыясненными. В том же XVIII- веке венгерский механик Фаркаш Кемпелен создал искусственного шахматиста. Эта кукла, размеры которой соответствовали взрослому мужчине, одетая в костюм турка, сидела за шахматным столиком-шкафчиком. Желающие могли сыграть с ней партию. Андроид играл настолько хорошо, что редко кому из его противников удавалось добиться победы. Много десятилетий возил по всему миру Кемпелен своего андроида. Тайна его казалась непостижимой. В отличие от всех ранее известных механических людей, «турок» Кемпелена выполнял действия, которые не могли быть заранее жестко заложены в его конструкции, ибо партии, разыгрываемые на шахматной доске, зависели не только от него, но и от его противников. И все-таки это была

мистификация. В тайниках шкафчика-подставки прятался шахматист. Кемпелен действительно был выдающимся механиком. Он создал хитроумнейшие приспособления, благодаря которым человек-шахматист мог фиксировать текущее расположение фигур на шахматной доске и управлять движением рук и тела «турка». Но как это происходило в точности, мы не знаем и, по-видимому, никогда уже не узнаем. «Турок» Кемпелена на многие годы пережил своего создателя. Он переходил из рук в руки, принося своим хозяевам значительный доход. Но однажды он погиб при большом пожаре.

Уже после Второй мировой войны случайно был обнаружен более ранний вариант андроида-шахматиста. Его отремонтировали. Теперь желающие могут осмотреть это хитроумное устройство и даже сыграть с ним партию, наслаждаясь искусством человека, «тайно» управляющего андроидом. Но остается недоказанной идентичность найденного варианта шахматиста тому знаменитому «турку», который долгие годы удивлял людей.

Чтобы закончить историю андроидов, мы должны назвать еще одно имя. Более шестидесяти лет тому назад чешский писатель Карел Чапек в драме «R. U. R» сделал главными героями механических людей. Они ничем не отличались внешне от человека, но превосходили его физически и интеллектуально. Чапек назвал их роботами. Это слово вошло во все языки мира, вытеснив название «андроид» и сохранив его лишь за механическими куклами прошлого.

Первые роботы XX века, выполненные в металле, фактически оставались все теми же андроидами. Но кое-что в них появилось и новое. Робот «Телевокс» американского инженера Дж. Венсли также имел вполне антропоморфный вид. Он мог двигать руками, вставать на ноги, садиться. И все это он выполнял не по жесткой программе, а по указанию своего изобретателя. Команды вводились в робот с помощью свистка. В роботе «Эрик», созданном в конце двадцатых годов в Англии, зрителей поражало то, что он выполнял команды, подаваемые обычным голосом. Еще более ошеломлял Зрителей огромный, двухтонный робот «Альфа», способный по приказанию голосом вставать, садиться, двигать руками и пальцами, стрелять из пистолета, говорить, свистеть и петь. Создателем этого чуда начала тридцатых: годов был английский физик Гарри Мей. Правда, как утверждают многие, роботы типа «Эрик» или «Альфа» действовали в паре с живым помощником. Только теперь не было необходимости, подобно Кемпелену, прятать помощника в самом роботе. Беспроволочная связь XX века позволила помощнику находиться вне робота.

Пожалуй, апофеозом развития роботов, непосредственно опирающегося на опыт, накопленный при. построении андроидов, стали роботы тридцатых годов XX века. Один из них — механический лектор, читавший лекцию о пищеварении на выставке «Столетие прогресса» в Чикаго в 1933 году. Во время лекции робот раздвигал одежду и высвечивал свои внутренности, полностью имитирующие устройство человеческого пищеварительного тракта. По ходу лекций робот показывал руками соответствующие отделы этого тракта, по которым двигалась «пища». В 1934 году были продемонстрированы публике робот, исполнявший на цитре по заказу зрителей любую из 3000 мелодий, и робот «Вилли» — усовершенствованный вариант робота «Альфа». В этих роботах механические движения настолько напоминали человеческие, что многие зрители были уверены, что их просто обманывают и вместо роботов показывают загримированных людей.

О дальнейшей истории развития роботов мы расскажем позже. А сейчас подведем некоторые итоги развития внешнего моделирования человеческого поведения и человеческих возможностей.

Тупиковая ветвь. Вся история андроидов и первых роботов происходила под знаком жесткой запрограммированности их поведения. Хитроумнейшие механические и электромеханические (а позже и электронные) приспособления, все эти шестерни, кулачки, реле и пружины, обеспечивали безупречное выполнение андроидами одних и тех же неизменных программ действий, протекавших в зафиксированных ситуациях. «Писец» всегда писал фразы из зафиксированного заранее набора фраз. Если в его чернильнице не оказывалось чернил, то он все равно продолжал макать в нее свое перо и водить пером по бумаге. «Флейтист» наигрывал свои мелодии, пока в руках у него была флейта. Но если вместо флейты в его руках оказывалась такая же по размерам и форме круглая палка, то пальцы его по-прежнему нажимали на места, где должны были быть клапаны и отверстия.

Таким образом, основным недостатком андроидов было отсутствие обратной связи между внешней средой и их действиями. Если в среде что-то менялось, если действие становилось бессмысленным, андроид все равно продолжал выполнять программу, заложенную в него конструктором. Этим он и выдавал свою неодушевленность, механистичность.

Успех андроида Кемпелена во многом определялся тем, что он мог демонстрировать возможность управляющего им человека приспосабливаться к изменяющейся ситуации. Когда Наполеон играл партию в шахматы со знаменитым «турком», он нарочно сделал ошибочный ход. «Турок» поправил ошибку и сделал ответный ход. Тогда Наполеон снова сделал ход, недопустимый с точки зрения правил движения фигур. «Турок» снова исправил его ошибку. Когда же Наполеон в третий раз нарушил правила игры, «турок» просто смахнул фигуры с доски и «выключился». Император был доволен. Целесообразность поведения «турка» в этой непредвиденной ситуации оказалась выше всяких похвал.

Отсутствие обратной связи со средой и способности адаптации к меняющимся условиям деятельности не позволили андроидам стать родоначальниками роботов Чапека. Эволюция андроидов оказалась тупиковой и не могла преодолеть возникших на ее пути препятствий.

Роботы предвоенного времени выполняли присущие им действия в различных Последовательностях. Но формирование последовательности определялось не контактом робота с внешним миром, а навязывалось ему извне с помощью командных сигналов.

И все-таки усилия, потраченные на создание андроидов и роботов первой половины нашего века, были не напрасны. Если даже не говорить о различных побочных эффектах, возникших в результате занятий роботами (радиоуправляемые гражданские и военные объекты, простейшие манипуляторы и многое другое), то надо отметить, что создатели андроидов и роботов первого поколения положительно решили вопрос о возможности механической имитации антропоморфных движений. Результаты эти сыграли большую роль в развитии биомеханики (теории человеческих движений), что позволило в наше время создавать протезы для людей, лишившихся возможности полноценно пользоваться своим телом. Руки андроидов были первыми манипуляторами, созданными человеком. И в современных манипуляторах, как бы совершенны они ни были, легко проглядывается генетическое родство с созданиями прекрасных механиков и инженеров прошлых веков;

Мы проследили этапы внешнего моделирования человеческих возможностей в области движения и действий. Но параллельно этому все время шли поиск и развитие методов, позволяющих имитировать внутренние процессы, протекающие в человеке. И в первую очередь процессы, связанные со святая святых — мышлением.

1.2. Логика рассуждений

Сократические беседы. Греция V века до н. э. представляла собой зрелище одновременно печальное и прекрасное. Наступал закат эллинистического периода. Рабовладельческая демократия изжила себя. Время было смутное, цели общества и пути их достижения неясными. Это ощущение неустроенности, беспокойства о завтрашнем дне, надвигающихся катаклизмов послужило источником для расцвета диспутов, споров и философских систем. Это было время софистов, создавших из споров целую науку, способных доказать все что угодно, опровергнуть любой тезис. Казалось, вся Греция занята, поиском истин, обоснованием и опровержением кардинальных основ государственности, этики, эстетики и морали.

Среди множества софистов, основной целью которых было показать иллюзорность истинности, ее зависимость от исходных посылок и способа рассуждения, выделяется фигура Сократа, непримиримого их врага, сделавшего основной целью своего искусства спора — поиск и утверждение истинности. С его именем связаны своеобразные соревнования спорящих, получившие название сократических бесед. По-видимому, такая форма диспута возникла еще до Сократа. Но его бле-

стящая Манера их проведения затмила предшественников, оставив его имя навечно связанным с этим единственным в своем роде зрелищем-соревнованием.

Сократические беседы протекали следующим образом. Предлагалась некоторая тема для обсуждения, некоторое положение, истинность которого априорно не предполагалась. Например, предлагалось обсудить положение о том, что отдельные недостатки людей могут оказаться полезными для укрепления государства, или обсудить тезис: «Знание есть не что иное, как чувственное восприятие». Далее диспут происходил следующим образом. Один из спорящих задавал вопросы, предполагавшие возможность однозначного утвердительного или отрицательного ответа. Отвечающий должен был давать эти альтернативные ответы, при необходимости поясняя их примерами из жизни, аналогиями и общими положениями. Цель спрашивающего — запутать отвечающего, показать, что его ответы приводят к логическому противоречию, отрицают друг друга. В таких диспутах оттачиваются формулировки, вырабатываются приемы построения цепочек вопросов на основании ответов, которые дает другой участник соревнования.

Великим учеником Сократа был Платон, донесший в своих сочинениях тексты и дух этих бесед. Примерно через сто лет после этих споров, в которых греческие философы достигли совершенства, появился человек, с чьим именем навсегда оказались связанными фундаментальные результаты в логике, теологии, математике и искусственном интеллекте.

Аристотель. Этим человеком был Аристотель — ученик престарелого, но все еще прекрасного в своем красноречии и логике рассуждений Платона. Любовь Платона к Сократу, которую тот пронес через всю свою жизнь, воспринял и его молодой ученик. Став зрелым мыслителем, естествоиспытателем и философом, Аристотель создает свое любимое детище, прославившее его имя в веках,— *теорию силлогизмов,* с помощью которой можно безупречно вести сократические беседы. Появление этого творения- древнего гения поражает. Впервые за всю историю человечества один человек создал целую науку, завершенную во всех ее основных частях. Такого случая в истории больше не наблюдалось. Многие столетия логика Аристотеля оставалась неизменной. Средневековым схоластам она представлялась божественным откровением, ученым XIX века — неизъяснимым совершенством. И лишь с конца XIX века началось расширение и развитие логических идей, оставившие далеко позади учение о силлогизмах великого греческого мыслителя.

В чем же суть придуманного Аристотелем? Главной своей задачей он ставил выяснение причин, лежащих в основе получения верного или неверного результата в процессе сократической беседы. А затем — нахождение такого способа спора, при котором из верных посылок — всегда следовали бы только верные заключения.

Первое, что сумел сделать гениальный греческий мыслитель,— это ввести понятие *силлогизма,* до сих пор играющего одну из центральных ролей в современной формальной логике. Что же такое силлогизм?

Прежде чем мы ответим на этот вопрос, рассмотрим несколько примеров. Каждый пример будет состоять из трех предложений. Будем предполагать, что третье предложение вытекает из двух предшествующих. Примеры мы обозначим буквами, а предложения, входящие в них, цифрами.

А. 1. Все люди смертны.
2. Гай Юлий Цезарь — человек.
3. Гай Юлий Цезарь смертен.

Б. 1. Нервные стрессы истощают силы.
2. Современная жизнь полна нервных стрессов.
3. Современная жизнь истощает силы.

В. 1. Все врут календари.
2. В календаре написано, что Лев Толстой умер в 1910 году.
3. Неверно, что Лев Толстой умер в 1910 году.

Г. 1. Лишь тот, кто храбр, достоин славы.
2. Некоторые хвастуны на деле оказываются трусами.
3. Некоторые хвастуны недостойны славы.

Д. 1. Все мизантропы считают, что человек зол от природы.
2. Герцог Ларошфуко — мизантроп.
3. Герцог Ларошфуко считает, что человек зол от природы.

В этих примерах первое предложение сообщает нам некоторый общий факт, а второе относится к какому-

то частному случаю, связанному с общим фактом из первого предложения. Поэтому в логике принято называть первое предложение *большой посылкой,,* а второе — *малой посылкой.* Третье предложение есть некоторое заключение, которое мы как бы выводим из большой и малой посылок. При этом мы как бы переносим некоторое свойство с общего случая на частный. В примере А этим свойством является то, что люди смертны, в примере Б — истощение сил и т. д. Процесс движения от общего к частному называется *дедукцией.* И те умозаключения, которые иллюстрируются нашими примерами, можно назвать *дедуктивными умозаключениями.* В повседневной жизни мы используем их постоянно, почти никогда не отдавая себе отчета в этом. Вы зашли к знакомым в гости. Хозяева предлагают вам сесть. Вы мгновенно проводите дедуктивное умозаключение, когда садитесь на стул, кресло или диван. Ибо все эти предметы принадлежат к общему классу предметов, предназначенных для сидения — на них можно садиться. Это позволяет вам провести рассуждение, которое в нашей записи выглядит следующим образом:

1. На мебель, предназначенную для сидения, можно садиться.
2. Этот стул есть мебель, предназначенная для сиденья.
3. На этот стул можно садиться.

Но вряд ли, занимая место в гостях, вы будете в явной форме проводить подобное рассуждение.

Дедуктивные рассуждения в жизни человека — обычное явление. Но верны ли они? По-видимому, не —всегда. Вернемся к нашим примерам. Относительно первого из них у читателя вряд ли возникнут сомнения. Вывод о смертности Гая Юлия Цезаря ко всему прочему подтвержден историей. Второе рассуждение для некоторых из читателей может показаться менее бесспорным. Например, некоторые могут считать, что нервные стрессы не истощают силы, а мобилизуют их. Другим само выражение «истощают силы» может показаться не слишком понятным. Относительно третьего примера почти все согласятся с тем, что фраза В.1 сомнительна. Большая посылка, скорее всего, ложна. Сказано это в некотором метафорическом смысле, а не в виде точного утверждения, отражающего повседневный опыт использования календарей. А значит, неверен и вывод, содержащийся в предложении В.З. Могут возникнуть сомнения и относительно полной справедливости большой посылки в примере Г, хотя подобная фраза вполне уместна, например, в поэтическом тексте. Но вывод и малая посылка в этом рассуждении вполне приемлемы, по-видимому, для всех читателей. Рассуждение Д интересно своей малой посылкой. От истинности или ложности ее зависит истинность или ложность вывода Д.З. Сам Франсуа Ларошфуко никогда не утверждал, что он мизантроп. Но итоги его наблюдений над нравами аристократического общества, собранные в его «Максимах», заставили многих современников Ларошфуко считать его мизантропом. Длительное время утверждение Д.З было почти непреложной истиной для всех, кого интересовала личность автора «Максим». И лишь совсем недавно советским исследователям Ю. С. Мартемьянову и Г. В. Дорофееву удалось неопровержимо доказать ошибочность этого мнения. Знаменательно то, что само это доказательство было получено с помощью дедуктивных умозаключений, опиравшихся на высказывания Ларошфуко, зафиксированные в «Максимах». К этому факту мы вернемся чуть позже.

Итак, не всякие дедуктивные умозаключения могут быть истинными. И одна из основных заслуг Аристотеля состоит в том, что он сумел четко сформулировать условия, при которых истинность вывода в дедуктивном рассуждении всегда обеспечивается, если большая и малая посылки верны. Силлогизм есть такая форма записи дедуктивного умозаключения, приведение к которой однозначно дает ответ об истинности или ложности вывода из двух истинных посылок. Аристотель оставался на уровне содержательных умозаключений. Для него важен был не формализм силлогистических фигур, а смысл, заключенный в высказываниях. Но его ученики и последователи пошли значительно дальше. Теофраст и Эвдем, принадлежавшие к перипатетической школе, отделили логику от метафизики и теории познания. Логические схемы стали самоцелью. У стоиков (особенно у Хрисиппа из Сицилии) этот отрыв логических форм от содержательного мышления произошел окончательно. И, наконец, полная формализация силлогистики Аристотеля, создание той дедуктивной системы, которая оставалась неизменной многие века, было завершено греческим философом-идеалистом Порфирием, написавшим «Введение в категории Аристотеля»—главный источник знакомства с логикой его великого предшественника в средние века.

После этого логика Аристотеля стала распространяться на Запад и Восток, впитывая в себя особенности мышления различных народов. «Последний римлянин» (как он называл себя в своих сочинениях) Боэций перевел Аристотеля на латинский язык и дал к переводу свой обширный комментарий. Восточные ученые, особенно сирийцы и копты, распространили идеи силлогистики по всей Малой Азии, занесли их в Армению и Византию. Интересно, что многие произведения греческих логиков и, в частности, очень важное для развития логики в современном понимании этого слова «Введение» Порфирия сохранились лишь в пересказах ученых Армении. А имя армянского ученого Давида Анахта Непобедимого оказалось навсегда связанным с критикой таких силлогизмов, как, например:

1. Писистрат получил армию и стал тираном.
2. Дионисий получает армию.
3. Следовательно, Дионисий станет тираном.

Логика Аристотеля означала громадный шаг вперед. Он создал теорию, позволяющую судить об истинности или ложности выводов из истинных посылок. Силлогистика Аристотеля выдержала испытания временем. И во времена Средневековья, и в Новое время, да и в наши дни силлогистика играет определенную роль в том круге наук, которые занимаются теорией логических рассуждений и выводов. Правда, за те долгие столетия, которые отделяют нас от Аристотеля, силлогистика не раз подвергалась жестокой критике, в нее привносилось нечто новое, делались попытки ее радикального изменения, но ядро этой теории сохранилось. Однако для нас, для той цели, ради которой написана эта книга, небесполезно указать на некоторые слабые стороны силлогистики Аристотеля.

Герменевтика. В наше время немногим известна наука с таким названием. Осталась она лишь в программах теологических факультетов зарубежных университетов и других учебных заведений, где готовятся кадры богословов и служителей церкви. Расцвет герменевтики в прошлом. В средневековых университетах она играла важную роль, и нельзя представить себе средневекового ученого, который не владел бы методами герменевтики.

Слово «герменевтика» греческое. По смыслу оно ближе всего к неуклюжему слову «толмачество». Как явствует из словарей русского языка, толмач — это человек, в обязанности которого входит передача понятным языком непонятного текста. Толмач выступает как интерпретатор этого «темного текста».

В античной Греции таким «темным текстом» были поэмы Гомера. Для тогдашней широкой публики их надо было разъяснять, вскрывать смысл того, что хотел сказать их автор. Греки античности были в таком же положении, как современный читатель, который попытается понять по первоисточнику «Слово о полку Игореве». Без толмача — специалиста по древнерусской культуре и языку ему явно не обойтись.

Но герменевтика это не только наука о толковании неясных текстов. Это еще и наука о выявлении смысла, который содержится в этом тексте в неявной форме, а также наука о получении новых утверждений из тех, которые содержатся в данном тексте. Эта ее последняя задача для нас наиболее интересна.

За тридцать лет до нашей эры среди древнееврейских теологов выделялся ученостью некто Гиллель. Это была эпоха жарких споров среди последователей иудаизма о толковании Пятикнижия (первых пяти книг Библии). Многие положения этих книг, «написанных Моисеем непосредственно под диктовку бога», были не слишком ясными, указания были неполны и могли истолковываться неоднозначно. Талмудисты, специализировавшиеся но толкованию «богоданной» книги, спорили между собой, противоречили друг другу. Число толкований росло, и современный Талмуд сохранил следы этого богословского спора до наших дней. В этих условиях Гиллель попытался навести хотя бы некоторый порядок в способах толкований и вывода новых положений из текста, каждое предложение которого, как полагали, было истиной. Им были предложены семь приемов такого вывода. По-древнееврейски эти приемы называются «миддот». Приведем в качестве образца приемов Гиллеля два первых, названных им "Хекеш" и «Кал-ва-Хомер».

Рассмотрим следующий пример*):

1. Трамвай — общественный транспорт.
2. Автобус — общественный транспорт.
3. Трамвай перевозит пассажиров.
4. Следовательно, автобус перевозит пассажиров.

Вывод, полученный нами, основан на том, что трамвай и автобус схожи между собой, между ними есть некоторая аналогия. «Хекеш» есть заключение по аналогии. Из того, что трамвай и автобус являются общественным транспортом (и этим похожи между собой), вытекает аналогичность их и в отношении перевозки пассажиров.

Всегда ли верны рассуждения, основанные на приеме «Хекеш»? Конечно, нет. Следующий пример иллюстрирует это:

1. Эдгар По писатель.
2. Теодор Драйзер писатель.
3. Эдгар По писал детективные рассказы.
4. Следовательно, 'Теодор Драйзер писал детективные рассказы.

Значит, рассуждение по аналогии не всегда верно. Это было замечено уже первыми критиками приемов Гиллеля. Среди ученых-талмудистов существовал даже специальный термин для обозначения возражений против предлагаемых исследователями - герменевтиками приемов вывода новых положений. По русски этот термин можно было бы перевести словом «ломка». На основании анализа «Хекеша» значительно позже было показано, что он справедлив лишь в том случае, когда «Хекеш» можно свести к силлогизмам Аристотеля. При этом первый из приведенных нами примеров рассуждения примет вид:

1. Любой общественный транспорт служит для перевозки пассажиров.
2. Автобус — общественный транспорт.
3. Следовательно, автобус перевозит пассажиров.

А для второго примера построить соответствующий ему силлогизм Аристотеля оказывается невозможным. Это требует истинности утверждения о том, что все писатели пишут детективные рассказы, что, конечно, неверно.

*) Разумеется, Гиллель приводил примеры, связанные с богословскими проблемами.

Однако в реальной жизни метод аналогий люди используют весьма часто, далеко не всегда заботясь о его логической обоснованности. Опасный своей нелогичностью, необоснованностью, метод аналогий привлекает людей возможностью получать новые факты на основании имеющихся частных знаний.

Этой же особенностью обладает второй прием Гиллеля «Кал-ва-Хомер», который позже, в средневековой Европе получил название *conclusio e minore ad majus* (заключение от меньшего к большему).

Перевод древнееврейского названия этого приема звучит примерно так: «Из легкого с отягощением». Прием этот столь же свойствен повседневным рассуждениям людей, как и «Хекеш». Суть его поясняет следующий пример:

1. За проезд без билета полагается штраф.
2. Воровство хуже, чем проезд без билета.
3. За воровство полагается штраф больший, чем штраф за безбилетный проезд.

Другими словами, множество некоторых поступков, за которые полагается наказание, линейно упорядочивается на некоторой шкале. И если за поступок, более легкий с точки зрения этой шкалы, полагается некоторое наказание, то за более тяжелый поступок наказание не может быть меньшим. Заметим, что уголовный кодекс любой страны всегда придерживается справедливости вывода с помощью приема «Кал-ва-Хомер». Этот прием в юриспруденции играет весьма важную роль при толковании законов.

Парным приемом, называемым *conculsio e majore ad minus* (заключение от большего к меньшему), служит рассуждение, иллюстрируемое следующим примером:

1. За комод красного дерева заплачено 400 рублей.
2. Стул красного дерева дешевле комода.
3. Его цена не может быть больше 400 рублей.

Конечно, прием «Кал-ва-Хомер» может применяться и необоснованно, но люди часто идут на это. Соответствующие примеры читатель, вероятно, легко найдет в своем личном опыте.

Остальные приемы Гиллеля обладают той же «человечностью», что и два рассмотренных нами. Его дело продолжили другие талмудисты. Рабби Исмаил довел число приемов до 13, а живший в конце II века рабби Элиезер использовал для своих богословских

толкований уже 52 приема, подавляющее большинство которых не сводилось к силлогизмам Аристотеля. Христианские богословы использовали для своих толкований силлогизмы Аристотеля, ряд приемов талмудистов и сами разработали герменевтические приемы, отражавшие специфику человеческих умозаключений.

Но средневековье сменило основную идею Аристотеля. Если для Аристотеля главной целью силлогистики было согласование мнений двух людей, убеждение одного из них в том, что утверждал другой, то для средневековых богословов силлогистика выступала как орудие согласования высказываний с теологическими догмами. Упоминавшийся уже нами Фома Аквинский использовал учение Аристотеля для логического обоснования догматов христианского учения. Такая «смена вех» потребовала от средневековых герменевтиков разработки новых видов силлогизмов, отсутствовавших у Аристотеля.

Так возникла и развилась герменевтика — искусство истолкования текстов. Современным его образцом может служить толкование «Максим» Ларошфуко. В чем суть этой работы? Сначала формируется совокупность утверждений, истинность которых либо принимается большинством людей данного круга, либо зафиксирована в текстах «Максим». Примером утверждений первого типа могут служить: «Человек стремится получать удовольствие», «Недостижение удовольствия огорчает человека». Примерами утверждений второго типа являются: «Человек хочет быть лучше других» или «Слабый человек не может быть искренним». Эти утверждения образуют исходный базис для выводов новых утверждений. Выводы опираются на специальные герменевтические приемы. Примером их могут служить: «Человек, имеющий цель *X*, ищет средства, чтобы достичь *X*» или «Из двух зол человек выбирает меньшее». Применяя к исходным утверждениям эти приемы, мы получаем новые утверждения.

Авторы работы о Ларошфуко построили такую систему исходных утверждений, которая соответствует Ларошфуко. Они показали, что для вывода всех утверждений Ларошфуко, содержащихся в «Максимах», достаточно только введенных ими начальных утверждений. А среди этих утверждений нет утверждения «Люди злы от природы», которое соответствовало бы тому, что Ларошфуко был мизантропом. Интересно, что построенная Ю. С. Мартемьяновым и Г. В. Дорофеевым система вывода такова, что она может порождать новые высказывания «в духе Ларошфуко». Вот одно из порожденных таким образом высказываний: «Именно потому, что человек хочет быть слишком хорошим, он не может быть им».

Но вернемся опять назад, в глубь веков, к истокам логики человеческих рассуждений, к Аристотелю.

Индукция. Роль Аристотеля не сводится только к изобретению силлогистики и построению теории дедуктивных умозаключений. В сократических беседах нужны были и другие средства убеждения противника в споре. Дедуктивные рассуждения Основывались на том, что спорящего сначала надо было убедить в правильности большой посылки, а затем, заставив принять малую посылку, подвести его к истинности заключения. Но можно было поступить и по-другому. Можно было бы сначала убедить противника в истинности ряда малых посылок, а потом из этих примеров перейти к тому, что справедливо и некоторое общее утверждение. Такое рассуждение, идущее от частного к общему, принято называть *индуктивным.*

Собственно индуктивны, как правило, сами большие посылки в силлогистике Аристотеля. Например, большая посылка в силлогизме «Все люди смертны» основана, вообще говоря, на обобщении единичных реальных фактов смертности конкретных людей, а не представляет собой абсолютную истину, данную свыше.

Однако во времена Аристотеля и особенно во времена средневековья индуктивные методы не были в чести. Герменевтики исходили из «боговдохновенных истин священных книг», которые и использовались в качестве общих посылок в силлогизмах. Время индукции наступило тогда, когда человек стал ощущать себя частицей окружающей природы, стал пытаться анализировать фактические данные, добываемые не путем рассуждений, а на основе эмпирического опыта. Значение индукции для приобретения новых знаний о мире было уже осознано Леонардо да Винчи, Коперником, Телезио, Чезальпино и другими выдающимися умами второй половины XV — начала XVI вв.

Честь четкого пояснения сути индуктивных рассуждений принадлежит английскому философу Фрэнсису Бэкону. Свой трактат он назвал «Новый органон», подчеркнув тем самым, что этот труд представляет собой новый этап развития логики, зафиксированной в логическом своде Аристотеля «Органон».

И опять произошла смена целей использования логических рассуждений. Вместо согласования высказываний с теологическими догмами основной задачей стало согласование высказываний с реальными наблюдаемыми фактами, с результатами опытов, накапливаемых эмпирическим знанием. Фрэнсис Бэкон своим огромным научным и политическим авторитетом как бы закрепил в сознании общества критическое отношение к наследию Аристотеля. Слова его далекого предшественника и однофамильца английского философа и естествоиспытателя монаха-францисканца Роджера Бэкона «Было бы лучше сжечь сочинения Аристотеля и начать все с начала, нежели принимать его заключения без проверки» стали знаменем науки наступающей эпохи.

В жизненной практике людей индуктивные рассуждения встречаются куда чаще дедуктивных. Всю свою жизнь мы накапливаем отдельные факты и наблюдения, формируем на этой основе свое представление о мире, его свойствах и закономерностях. Дедуктивная система выступает у нас лишь как результат целой серии наблюдений и индуктивных умозаключений.

Однако связь между индукцией и дедукцией не односторонняя. Дедуктивные выводы сами могут служить основанием для постановки тех или иных экспериментов, быть побудительными мотивами для поиска новых конкретных фактов и возникновения некоторого индуктивного процесса.

Фрэнсис Бэкон был очень занятым человеком. Он был судьей и крупным государственным деятелем. Наукой он занимался «в свободное от работы время», между делом. И, может быть, поэтому в его сочинении, блестяще написанном и полном тонких наблюдений и выводов, нет самого метода индуктивных рассуждений. Здравый смысл властвует в «Новом органоне», но в этой книге нельзя найти руководства к действию, чего-либо даже отдаленно похожего на стройные фигуры силлогизмов Аристотеля. Вряд ли стоит особенно порицать за это «отца индуктивных методов познания мира». Индуктивная логика даже и сейчас, когда со времен ее декларации прошло столько времени, все еще не обрела той формальной строгости, которая присуща дедуктивной системе Аристотеля. Ибо для индуктивной логики нужно знание и теории вероятностей, и математической статистики, и ряда других наук, которых во времена Фрэнсиса Бэкона еще не было.

О том, чего достигла современная индуктивная логика, мы поговорим в свое время. Сейчас же для нас важно одно: в отличие от дедукции, опирающейся на ранее зафиксированные, как бы вложенные в мозг человека знания, индукция требует от него выводов из наблюдений фактов, существующих вне его, в окружающем его мире. А это само по себе требует специальных методов описания фактов и методов их систематизации. Нужна целая наука, посвященная решению этих проблем. Но прежде чем перейти к рассмотрению предыстории этой науки, проследим, чем закончилась эволюция дедуктивной логики к началу эры вычислительных машин.

От логики к математике. Наступил XIX век. В этом веке господствовали рационализм и несокрушимая вера в мощь научного мышления. Наука и техника казались всемогущими, а научные теории — объективными и опирающимися на прочный фундамент. Стремление к научной точности и ясности иногда даже отодвигало на второй план содержательные аспекты. Особенно это касалось математики и смежных о ней дисциплин. Логика, все еще опирающаяся на авторитет древнего греческого философа и его последователей, стала объектом суровой научной критики.

Оксфорд конца двадцатых годов XIX века. Студенты в традиционных костюмах учат традиционную схоластическую логику по популярному тогда учебнику Ольдрича. Учат ее почти механически, запоминая многочисленные виды силлогизмов, модусы и фигуры. Наука, называемая логикой, по существу, мертва. И, как подтверждение этому, звучит рефрен стихов, сочиненных еще в XIII веке, в которых перечислены все модусы фигур силлогизмов Аристотеля: «Барбара, Целарент, Дарли...». Такая наука не могла не вызвать протест у рационалистов XIX века. Этот протест нашел свое выражение в работах молодых тогда английских логиков, философов и естествоиспытателей, собиравшихся чуть ли не ежедневно в небольшой комнате одного из преподавателей Оксфорда. Здесь были Копльстон, Уэтли, Ньюмен и Джон Милль. Все они внесли свой вклад в обновление традиционной науки. Их критика основных положений схоластической логики, попытка синтеза дедуктивных и индуктивных выводов, требование эмпирических посылок возбудили новый интерес к логике, подготавливали большой переворот в развитии этой науки.

Обычно творцом этого переворота считают английского математика и логика Джорджа Буля. Именно он впервые описал дедуктивную систему как некоторый формальный математический объект и ввел в обиход понятие *исчисления.* Буль рассматривал три вида исчислений: исчисление классов, исчисление высказываний и исчисление отношений. В современной математической логике (а именно так она стала называться со времен Дж. Буля, одного из ее основоположников) осталось лишь исчисление высказываний Буля, а два других его исчисления превратились в исчисление предикатов.

В чем же заслуга Буля? Ответ на этот вопрос несколько парадоксален. Буль сумел исключить из логики семантику. Во всяком случае, свести ее к точной и однозначно понимаемой математической семантике. Вместо силлогизмов такого вида, как мы рассматривали выше, каждое утверждение которых переполнено семантикой, присущей человеческому языку, Буль стал записывать их в виде некоторых формальных правил вывода. Примером его может служить правило, называемое в логике *modus ponens:* «Если *А* является истинным фактом и верно то, что из *А* всегда следует *В,* то *В* является истинным фактом». Это правило было, конечно, известно задолго до Буля. Но он первый смог придать ему однозначный математический смысл.

Дальнейшее развитие идеи Буля привело к тому, что к первой трети XX века была окончательно сформирована идея *формальной системы.* Это понятие будет для нас в дальнейшем чрезвычайно важным с точки зрения интеллектуальных искусственных систем. Поэтому читателю придется приложить некоторые усилия для пониманий того, что будет сказано ниже.

Пусть мы имеем множество объектов произвольной природы. Обозначим это множество буквой *А*. Относительно элементов, входящих в него, мы ничего не знаем. Однако мы предполагаем, что для нас реально выполнение двух операций. Если нам предъявляют некоторый объект, то мы однозначно можем сказать принадлежит ли он множеству *А* или не принадлежит. А для любых двух элементов из множества *А* мы можем сказать, совпадают ли они между собой пли нет. Другими словами, каждый элемент из множеств *А* мы можем назвать некоторым именем, присущим только ему. Примерами множества *А* с подобными свойствами могут служить множество прописных букв русского алфавита, множество деталей детского конструктора, помещающихся в стандартной фабричной упаковке, множество сотрудников некоторого фиксированного учреждения в данный момент времени. Буквы могут иметь различное написание, но либо идентифицируются некоторым именем (а, бе, ве и т.д. либо, если опознать их невозможно, не относятся к буквам русского алфавита. Детали конструктора входят в набор согласно прилагаемой описи, но в наборе могут встречаться различные совпадающие по названию детали. В множестве сотрудников совпадений не бывает (конечно, вполне возможно, что в данном учреждении работают два Ивана Ивановича Иванова, но тогда в качестве их отличительного признака выступает, например, возраст или цвет волос).

Так устроено множество *А,* которое можно назвать *множеством базовых элементов.* Это множество является первым элементом формальной системы.

Введем теперь конечный набор правил, с помощью которых из этих элементов можно строить правильные совокупности элементов. Сами правила такого типа мы будем называть *синтаксическими правилами* и обозначать буквой *Р.* Они образуют второй элемент.

Вернемся к нашим примерам множества *А.* Для первого из них правила Р описывают рождение слов из букв русского алфавита. Как всем нам хорошо известно, слова образуются последовательным приписыванием букв друг к другу. Каждое слово отделяется от других специальными концевыми маркерами — пробелами. При обычных условиях их никак специально не обозначают. Просто между словами оставляют больший

просвет, ЧЕМ просвет между буквами одного слова. Но бывают случаи, когда эти маркеры надо специально указывать. Например, при вводе текста на русском языке в ЭВМ вместо фразы «Федя учит японский язык» могла бы возникнуть потребность в написании фразы: «§Федя§учит§японский§язык§». Здесь значок §, не входящий в русский алфавит, играет Роль Разделителя слов — пробела.

Существуют ли такие правила *Р*, которые выделяли бы среди всех совокупностей из букв русского алфавита только те, которые мы признали бы за слова. Ответ на этот вопрос довольно сложен. Ясно. например, что такие совокупности, как

ааа ааа р
ааа или роооо
ш

невозможно считать словами в обычном понимании. Но придумал же поэт Андрей Вознесенский специальвый вид «изобразительной поэзии», произведения ко-

Алуна канулА

Рис. 1.

торой *изопы,* состоят из графических фигур, образуемых словами (рис. 1). В пояснение своего изопа «а луна канула», посвященного полету человека на Луну, поэт пишет:

«Мой изоп „а луна канула" читается слева направо и обратно. Мне хотелось, чтобы зритель как бы проследил взглядом по буквам путь капсулы от Земли к Луне и обратно [поэтому сам изоп изображается строчкой согнутой по параболе, напоминающей траекторию полета.-Д. *П.]. Последнее „а" перевернуто, так как для человека на Луне Земля маячит сверху как Луна для землян».*

Изопы — это нечто специальное. Но как узнать в нормально написанных последовательностях букв: слова это или не слова? Конечно, "щьннеуц" не может быть словом русского языка, ибо в нем явно нарушены нормы языковой фонетики. Здесь сомнений нет. И столь же несомненно, что слово «мама» принадлежит русскому языку. Но вот слово «квинтуплекс» или слово «сиволдай»? Что это: слова русского языка или нет?*).

Мы так долго говорили об этом, чтобы подчеркнуть, что факт существования синтаксических правил *Р* означает весьма четкое понимание, что есть правильные совокупности элементов множества *А*.

Для второго примера с помощью правил *Р* порождаются всевозможные сочетания из элементов конструктора, получаемые за счет крепежа, приложенного к конструктору. Однако соединения этих элементов с помощью сварки или путем изгибания деталей конструктора в рамках правил *Р* должны рассматриваться как недопустимые.

Для третьего примера правильные совокупности сотрудников некоторого учреждения могут определяться, например, распоряжением директора, которым устанавливаются структурные подразделения (отделы, лаборатории и т. д.). Правила *Р* указывают принадлежность того или иного сотрудника к той или иной вычлененной директором совокупности.

Третьим элементом формальной системы (правда, не совсем обязательным, но для нас удобным) являются *аксиомы*. Обычно у людей, помнящих курс школьной математики, с этим понятием связаны какие-то соображения очевидности, истинности и верности независимо от обстоятельств того, что определяется этим словом. В формальных системах ничего этого нет. Аксиомы — это просто произвольно выделенные правильные совокупности элементов. Число их никакой роли не играет. В крайнем случае все правильные совокупности элементов *А* можно, считать аксиомами. Чтобы не затягивать изложение, обратимся лишь к третьему из множеств *А*. Для пего аксиомами, например, можно считать директора и всех начальников подразделений, каждый из которых не считается включенным в совокупность сотрудников, образующих

*) Эти слова принадлежат русскому языку. См., например, Орфографический словарь русского языка, изд. 9-е.— М.: Советская энциклопедия, 1969.

сами подразделения. Множество аксиом обозначим буквой *Н*.

Последним элементом формальной системы являются *семантические правила* или *правила вывода*, которые мы будем обозначать буквой *П*. С их помощью из аксиом «выводятся» новые правильные совокупности. Считается, что аксиомы априорно выведены, а совокупности, получаемые в результате применения семантических правил, обладают некоторым смыслом в данной системе. Для второго из наших иллюстративных примеров среди всех правильных совокупностей элементов, образующихся из исходных деталей конструктора, осмысленными можно, например, считать только те, которые в качестве примера приведены в прилагаемом к конструктору руководстве.

Теперь можно задать формальную систему в виде четверки *(А, Р, Н, П)*.

Специалистами было показано, что и силлогистика Аристотеля, и исчисления Буля, и многие другие логические исчисления укладываются в жесткие рамки формальной системы. Формальная система оказалась завершающим шагом на пути развития дедуктивных систем рассуждений. И в этом завершающем шаге исчезла логика и победила математика. Математическая логика как бы поглотила логику предшествующего времени. К каким это результатам привело, мы обсудим позднее. Заметим только, что не все они были положительными с точки зрения проблем искусственного интеллекта.

При изложении истории возникновения и формализации логических рассуждений мы опирались лишь на европейскую традицию. Однако развитие логики шло и в других странах. Упомянем, например, о традиции индийских логических исследований. В Индии было создано несколько логических дедуктивных систем: старая и новая ньяя, буддийская логика и ряд других. Эти логические системы в некоторых отношениях превосходили достижения Аристотеля и ученых средневековой Европы (например, новая ньяя в вопросах анализа отношения проникновения, столь волновавших средневековых схоластов, ушла далеко вперед по сравнению с силлогистикой). Но они были недоступны европейским ученым. Лишь в последние десятилетия проявился интерес к этим достижениям.

Для нас важным является то, что к середине XX века уже существовало сформировавшееся здание теории логических рассуждений.

1.3. Языки описания мира

Вавилонская башня. Легенда о Вавилонской башне, пришедшая из глубины веков, отражает то удивление, которое испытывали люди уже в древности от многообразия и непохожести человеческих языков. Даже ближайшие соседи, жившие в почти одинаковых условиях и занимавшиеся одним и тем же трудом, зачастую говорили на столь непохожих языках, что абсолютно не могли понять друг друга. Казалось бы, что единая Земля, на которой мы живем, тот общечеловеческий мир, который нас окружает, должны бы были породить и единый человеческий язык. И для объяснения, почему этого не произошло, была придумана история о гневе бога, наказавшего человечество за его гордыню. Люди, обладавшие единым языком, были столь сильны, что решили построить башню до самого неба, подняться на нее и стать «равными» самому богу. Но в процессе реализации этого грандиозного замысла бог сумел нарушить планы людей, разделив строителей на группы и дав каждой из групп человечества свой особый и непонятный другим язык. Потеряв способность действовать сообща, люди не смогли завершить свое дерзкое начинание.

Так говорит легенда. И во времена ее младенчества люди верили, что в будущем языковые барьеры исчезнут и снова возникнет на Земле единый, общепонятный язык. Еще в VII в. до н. э. пророк Софоний говорил о светлых будущих временах, когда уста всех людей будут произносить лишь слова единого «очищенного» языка.

Когда Аристотель создал свою силлогистику, а его последователи стали ее развивать, то наука впервые столкнулась с тем, что неточность и многозначность естественных языков могут приводить к логическим ошибкам.

Неточности и различия в человеческих языках неоднократно приводили к серьезным последствиям. Например, из-за того, что в арамейском языке «дух святой» женского рода, он не мог быть отцом Иисуса Христа. И в апокрифическом Евангелии от евреев можно прочитать: «Дух святой — матерь моя». Следствием этого лингвистического различия явилось выделение в христианстве секты, которая вступила в спор с основным христианским течением, утверждавшим догмат Святой Троицы»

Приведем пример из другой временной эпохи. В первой половине XX века Бенджамин Ли Уорф, о гипотезе которого, связывающей язык и мышление, мы еще будем говорить в этой книге, работал в обществе страхования от огня. Разумеется, в этом обществе весьма внимательно относились к анализу причин возникновения пожаров. Уорфу удалось обнаружить среди них одну, весьма необычную. Эта причина крылась в неточности языка, в неправильном соотнесении слов и предметов реального мира. Что такое «свинцовый лом»? Даже если читатель никогда не сталкивался с подобным понятием, то у него наверняка при этих словах возникнет некоторая предметная ассоциация: груда металлических обрезков, кусков неправильной формы, остатков каких-то безнадежно испорченных вещей из свинца. И негорючесть свинцового лома очевидна.

Однако при анализе многих пожаров на производстве оказалось, что именно свинцовый лом был их источником, ибо среди лома находились обломки вышедших из употребления конденсаторов. Эти обломки в большом количестве содержали обрывки парафинированных бумажных прокладок — прекрасного горючего материала. И стоило свинцовому лому достаточно нагреться, как возникал пожар.

Или еще один пример. Когда-то знаменитый французский баснописец Жан де Лафонтен написал басню «Кузнечик и муравей». В ней шла речь о трудолюбивом муравье и беспечном кузнечике, пропрыгавшем все лето, не заботясь о суровой приближающейся зиме. По-французски кузнечик женского рода *(la santerelle)* и его можно награждать эпитетами «попрыгунья», «шалунья» и т. п. Но когда И. А. Крылов стал переводить басню «Кузнечик и муравей» на русский язык, то, желая сохранить женскую сущность персонажа — кузнечика, он, в силу норм русского языка, не смог это сделать. Так возникла стрекоза, существо женского рода. Но что же получилось в результате? Полная несообразица. Ибо у Крылова стрекоза поет и танцует, что этому насекомому никак не свойственно.

Поэтому стремление к устранению лингвистических различий в языках, увеличению их точности при описании реального мира возникло еще на том начальном этапе развития науки, когда человек стал задумываться над сущностью своего языка. Тогда начала свою жизнь мечта о возможности создания идеально строгого языка, термины которого не допускали бы многозначного толкования. Эта мечта казалась скоро осуществимой. Ведь сумел же великий Евклид создать стройную аксиоматическую систему для геометрии. Наряду с силлогистикой Аристотеля геометрия Евклида казалось настолько совершенной, что вплоть до XIX века ее фундамент не вызывал никаких сомнений. . И понадобился гений Лобачевского, чтобы опровергнуть это казавшееся незыблемым мнение.

Эти две цели — создание единого языка общения и создание строгого научного языка описания действительности и человеческих суждений — иногда рассматривались как совместные, а иногда разделялись.

Идея всеобщего, языка оказалась весьма живучей. Известно несколько сот проектов такого языка. Подавляющее большинство их так и осталось проектами, в которые свято верили лишь их авторы. Но некоторые из искусственных языков оставили более заметный след в истории человеческой культуры. Для наших целей интересно обсудить некоторые из идей, использовавшихся авторами подобных языков.

Жесты и вещи. Беда Достопочтенный, обычно называемый отцом английской истории, жил на рубеже VII и VIII вв. Как и большинство просвещенных людеи того времени, был он монахом, жил в графстве Нортамберленд. Время было неспокойное - и смутное, полное кровавых переворотов, битв и интриг. История творилась на глазах. Беда Достопочтенный, подобно древнерусскому летописцу Нестору, болея душой за родную землю, писал страницы ее истории. Но кроме того, Беда Достопочтенный был миссионером по призванию. Он мечтал о просвещении язычников-бриттов, которые погрязли в идолопоклонстве и не очень-то хотели принимать от «просвещенных» англов христианство. И одной из причин этого был языковый барьер.

Язык бриттов резко отличался от языка англов. Будучи истинным ученым, Беда Достопочтенный поставил перед собой задачу в общем виде: можно ли создать такой язык коммуникации между людьми, не понимающими друг друга, с помощью которого удалось бы передать другому народу глубокие истины христианства? Результатом этих размышлений Беды Достопочтенного явился трактат «О числах и знаках», в котором он изложил основы специального языка жестов. По мнению великого историка, этот язык, названный им «языком перстным», мог решить поставленную задачу.

Как уже ясно из самого названия языка, опирался он не на звучащие слона, а на специальные жесты, каждый из которых и комбинации из них передавали определенные понятия окружающего мира. Для того чтобы семантика жестов стала понятной собеседнику, использовались указующие жесты, соотносящие знаки языка с предметами реального мира.

Вспомним, как часто мы. в разговоре используем жесты. «Дайте мне чашку, — говорим мы.— Нет, не ту, а вот эту». Последние слова мы обычно сопровождаем жестом руки.

Приведем другой пример. Найдя Пятницу, Робинзон встал перед трудной проблемой. Пятница не понимал ни слова по-английски. Робинзон стал давать ему первые уроки языка. Сначала Робинзон несколько раз показал пальцем на себя и произнес: «Робинзон». Затем, несколько раз показав пальцем, на своего собеседника, каждый раз произносил: «Пятница». Эти жесты соотнесли слова — имена с собеседниками, и Пятница усвоил два первых английских слова. Далее дело пошло аналогичным образом. Указывая на хижину, Робинзон говорил: «дом», и Пятница получал в свое распоряжение еще одно слово — предмет.

Замена слов предметами реального мира, предъявляемыми собеседникам, иногда позволяет прекратить бесплодные споры. Можно долго спорить о достоинствах собаки, но многие основания для спора отпадают, как только предмет спора появится в поле зрения спорщиков.

В сатирическом романе Джонатана Свифта «Путешествие Гулливера» многие ученые и мудрые люди Лагадо, столицы Лапуты, таскают на плечах большие узлы с вещами, приспособленными ими для выражения своих мыслей и желаний при необходимости вести пространный разговор. Рассказ об этом оригинальном способе коммуникации Свифт заключает следующим выводом:

"...великим преимуществом этого изобретения является то, что им можно пользоваться как всемирным языком, понятным для всех цивилизованных наций, ибо мебель и домашняя утварь всюду одинакова или очень похожа, так что ее употребление легко может быть понято. Таким образом посланники без труда могут говорить с иностранными королями и их министрами, язык которых им совершенно не известен".

Язык жестов оказал значительное влияние на средневековую схоластику и теологию (вспомните, как важен тот или иной жест в иконографии). В романе Франсуа Рабле «Гаргантюа и Пантагрюэль» этот язык послужил темой для блестящей пародии. Английский ученый муж Таумаст, прослышав о беспримерной учености Пантагрюэля, приезжает к нему и вызывает его на научный диспут. Условия диспута он формулирует следующим образом:

«...я *хочу диспутировать только знаками, молча, ибо все эти предметы до того трудны, что слова человеческие не выразят их так, как бы мне хотелось».*

Как известно, Пантагрюэль уклонился от диспута под тем предлогом, что у него есть ученик. И если англичанин победит его в подобном споре, то тогда Пантагрюэль примет вызов. Этим учеником выступает по воле Рабле пройдоха и плут Панург, бродячий философ, которых в средневековой Европе было предостаточно.

Англичанин принял эти условия. Между Таумастом и Панургом состоялся диспут, во время которого они старались не произнести ни слова, а объяснялись лишь жестами — знаками. Недостаток места не позволяет нам воспроизвести главу XIX второй книги романа, в которой изложен ход диспута. Приведем лишь один эпизод:

«Вдруг Панург поднял правую руку, засунул большой ее палец в правую же ноздрю, а остальные четыре пальца сжал и вытянул на уровне кончика носа, левый глаз совершенно закрыл, а правый прищурил, низко опустив и бровь и веко; затем высоко поднял левую руку, плотно сжал и вытянул четыре пальца, а большой палец поднял, после чего левая его рука приняла

такое же точно положение, как и правая, отделяло же их одну от другой расстояние в полтора локтя. Потом он опустил обе руки, а затем поднял до уровня плеч и как бы нацелился в нос англичанину».

Это пародийное описание тем не менее верно отражает систему «языка перстного» Беды Достопочтенного. Заметим, что современный «язык», которым пользуются люди, не обладающие даром речи, построен по этому же принципу*).

Философский язык. Недостатки естественных человеческих языков, связанные с их многозначностью и неточностью, обращали на себя внимание еще в Древней Греции. Софисты приводили многочисленные примеры языковых парадоксов. В диалоге «Евтидем» Платона приведено немало примеров софизмов. Вот один из них. Человека, вошедшего в комнату, спрашивают: «Знаешь ли ты человека, стоящего сейчас за занавесом?». «Нет», — отвечает вошедший. «За занавесом стоит твой отец. Следовательно, ты его не знаешь». Ясно, что парадокс, который отражен в выводе, возник просто из-за многозначности слов языка. Во все времена и, по-видимому, у всех народов такие языковые развлечения всегда имели успех.

Появление силлогистики заставило исследователей весьма осторожно относиться к языковым конструкциям. Почему, например, неверно рассуждение типа: Иван — человек. Петр — не Иван. Следовательно, Петр — не человек? Только потому, что понятие «человек» имеет некоторый объем, который включает в себя не только Ивана. Но в языке это знание никак не отражается. Поэтому обычный человеческий язык не может быть идеалом для точного логического знания и рассуждений о мире.

В XVII веке идея создания точного языка для описания знаний об окружающем мире нашла многочисленных сторонников. К этому времени Франсуа Виет ввел в математику алгебраическую символику, оказавшую огромное воздействие на развитие не только алгебры, но и геометрии. Геометрия стала точной наукой именно благодаря новому языку. Декарт перевел содержательные утверждения Евклида и его последова-

*) В отличие от научно разработанной пальцевой фонетической системы для немых и глухонемых.

телей на язык алгебры, создал основы той науки, которая теперь называется аналитической геометрией. Такие неточные и нестрогие объекты, как «точки», «прямые», «плоскости», связанные с человеческими представлениями, заменились четкими и однозначно понимаемыми всеми математическими понятиями. Этот успех побудил Декарта к поиску средств для построения столь же точного языка для всех наук. В 1621 году он пишет письмо французскому ученому аббату Марену Мерсенну, в котором так формулирует идею подобного языка: «Установить порядок в мыслях так же, как он установлен в числах; тогда, как за один день можно назвать все числа до бесконечности и записать их на незнакомом языке, хотя имен чисел — бесчисленное множество, так же можно поступить со всеми другими словами, необходимыми для выражения любых человеческих мыслей». Декарт считал, что такой язык, классифицирующий все человеческие знания о мире, обладающий четкой различительной силой, однозначно понимаемый всеми людьми, должен основываться на некоторой философско-логической идее всеобщей классификации. Именно поэтому он назвал подобный язык философским.

Декарт не только высказал идею философского языка. Он пытался практически его построить. На базе шести известных ему европейских языков он создал грамматику будущего языка науки. В этом языке все понятия закодированы с помощью цифр, что позволяет проводить арифметические и алгебраические операции с понятиями. Декарт не довел свою работу до логического завершения. Но в последующие годы многие энтузиасты пытались использовать его идеи математизации языковых понятий. Такова, например, пазиграфия Тирояна, созданная в начале XX века. В ней имеется около 4000 слов, закодированных числами, а грамматические формы выражаются с помощью знаков математических операций. Например, «камень» кодируется как 2252, а «дом» — как 1104. Прибавление нуля к коду означает множественное число: 22520—«камни», 11040— «дома». Введение справа еще одного цифрового разряда, отделенного чертой, дает возможность ввести падежи: 2252/2 — «камня», 11040/3 —«домам». Выражение вида $\sqrt{2252}$ означает глагол «окаменеть», а сумма 2252 + 1104— «каменный дом».

Философский язык привлекал и Лейбница, твердо верившего в то, что математика и ее язык обладают универсальностью с точки зрения описания и объяснения всего, что мы наблюдаем в окружающем нас мире. Лейбниц мечтал о том, что когда такой язык будет создан, то на основании некоторой суммы знаний, описанных на нем, станет возможным строго логически получать новые знания, выводя их (не обращаясь к внешнему миру) средствами самого философского языка.

Подобно философскому камню алхимиков, философский язык так и остался недостижимой «Синей птицей». Но усилия, потраченные на его создание, не пропали даром. Классификация Карла Линнея — первая строго научная классификация в естествознании — возникла на основе идей философского языка. Универсальные классификаторы, позволяющие упорядочивать огромные книгохранилища библиотек, также опираются на идею родо-видовой классификации философских языков. Вывод в формальных логических исчислениях реализует идею получения новых знаний из старых, высказанную впервые при попытках построения философского языка. И, как мы увидим дальше, современные исследования в области интеллектуальных систем также активно используют те идеи о строгом языке знаний, которые были сформулированы еще 350 лет тому назад.

Языки без исключений. Что является наиболее трудным, когда мы начинаем изучать незнакомый нам язык? По-видимому, однозначного ответа на этот вопрос не существует. Сколько людей, столько и мнений. Но среди этих мнений часто будут повторяться сетования на зыбкость языковых правил, на необходимость запоминания и заучивания многочисленных исключений из них. Сколько мучений, например, доставляют всем изучающим английский, французский или немецкий язык неправильные глаголы. И как много времени и сил затрачивали школьники, учащиеся дореволюционной России, заучивая слова, в которых полагалось писать букву «ять». Подобных примеров много, ибо нет в мире естественного языка, в котором не было бы исключений из общих правил. А в некоторых языках, по-видимому, единственным правилом является то, что из любого правила есть многочисленные исключения.

Поэтому наряду с мечтой о точном научном языке всегда жила мечта о таком всеобщем языке общения, учить который было бы легко и просто. Этот язык представлялся настолько стройным и логически совершенным, что в нем были бы невозможны неправильности и исключения.

Наиболее известным языком такого типа является эсперанто. Эсперанто, пожалуй, единственный искусственный язык для коммуникации между людьми и хранения знаний в письменной форме, который получил некоторое распространение. На этом языке умеет говорить несколько сот тысяч человек, на эсперанто имеется научная литература, на этот язык переведены основные произведения мировой литературы, на языке эсперанто выходят газеты и журналы.

Эсперанто не знает деления по родам, все существительные одинаково образуют множественное число, все глаголы спрягаются единообразно, все существительные склоняются по единой схеме, от любого существительного по четкому единому правилу можно образовать прилагательное, а от любого глагола — причастие.

Однако эта логичность языков типа эсперанто касается только синтаксиса. Она совершенно не связана с уточнением понятий языка, уточнением их связи по объемам. В этих языках не ставится задача устранения семантической неоднозначности естественных языков. С точки зрения семантики эсперанто ничуть не лучше и не хуже любого европейского языка. И если на эсперанто перевести остроумный пример неправильного вывода, придуманный советским логиком Л. Гокиели: «Если от поезда отцепить последний вагон, то у поезда не будет последнего вагона», то это рассуждение сохранит на эсперанто все свои особенности.

1.4. Кратное резюме

Мы проследили долгий путь некоторых идей, связанных с имитацией внешних форм поведения человека, его способности к рассуждениям и коммуникации. К началу XX века многие из этих идей дали богатые плоды, как, например, идея формализации рассуждении. Другие остались на уровне начальных догадок и мечты. Однако все они оказались нужными и полезными, когда психология, отделившись от философии, стала самостоятельной наукой о человеке и когда появились технические устройства, названные электронными вычислительными машинами, способные имитировать многие специфические особенности мышления и жизнедеятельности человека. Но это уже история, которая непосредственно предшествует появлению науки с интригующим названием «Теория искусственного интеллекта».

ГЛАВА ВТОРАЯ

ИСТОРИЯ

Чистая математика делает то, что можно, и так, как нужно. Прикладная математика делает то, что нужно, и так, как можно.

Профессиональный фольклор

2.1. Формализация процедур

Алгоритм. Развитие науки в первой половине XX века протекало под влиянием физических и математических идей. Физика и математика бурно развивались и казалось, что их триумфальное шествие нельзя замедлить. Ясность и строгость математических моделей служили почти недосягаемым эталоном для моделей в других науках. Математики стремились сделать воздвигаемое ими здание своей науки полностью ясным и точным. Возникла даже специальная научная дисциплина, названная «Основания математики». Она изучала тот фундамент, на котором воздвигнута математика, в частности, стремилась уточнить смысл базовых, исходных понятий, используемых в математике. Одним из таких понятий является *алгоритм.*

Если открыть энциклопедический словарь, то можно прочитать, что под алгоритмом понимается конечный набор правил для выполнения некоторой процедуры, удовлетворяющий трем основным требованиям.

Первое — *массовость.* Предписание должно обеспечивать выполнение не одной конкретной процедуры, а быть пригодным для реализации класса однородных процедур. Пусть, например, вам необходимо позвонить знакомым в другой город. Вы раскрываете телефонную книгу и находите в ней правила вызова абонентов из другого города. Предположим, там написано: «Наберите цифру 8. Услышав непрерывный гудок, наберите код вызываемого города (приложена таблица),

а затем шестизначный номер вызываемого абонента. Если этот номер состоит из меньшего числа цифр, то перед ним наберите столько нулей, сколько цифр в вызываемом номере не хватает до шестизначного».

То, что вы прочитали, есть некоторый алгоритм. Здесь требование массовости состоит в том, что указания позволяют решать задачу вызова интересующего вас абонента при любом изменении его личного номера и для любого из городов, с которыми имеется в настоящее время автоматическая телефонная связь. Это как бы правила вызова абонента «вообще». В отличие от этого, например, поиск приятеля в районе новостроек, осуществляемый по схеме, нарисованной им, требованию массовости не удовлетворяет. Схема позволяет решить только одну фиксированную задачу поиска. И никак не помогает организовать процесс поиска какого-либо другого знакомого.

Второе требование обычно называется *детерминированностью.* Слово это, означает, что указания, образующие алгоритм, должны быть однозначно понимаемыми. В них должны отсутствовать какие-либо неоднозначности. Детерминированность обеспечивает одинаковость результата, получаемого при выполнении алгоритма, если исходные данные сохраняют свое значение. Реализация алгоритма, таким образом, никак не зависит от воли и желаний исполнителя.

Алгоритм вызова абонентов из другого города, конечно, удовлетворяет требованию детерминированности. При неизменных коде города и номере абонента, не совершая ошибочных действий, противоречащих указаниям алгоритма, вы всегда соединитесь с тем, кто вам нужен. Но если перед тем как позвонить, вы выбираете абонента из большого списка (как часто бывает в новогоднюю ночь), а сам выбор осуществляете случайным образом, то положение меняется. Если в качестве первого указания к автоматической связи будет «Выберите случайно фамилию из списка», то такой набор указаний не будет соответствовать требованию детерминированности.

И, наконец, третье требование — *результативность.* Это требование обеспечивает конечность применения указаний. Результат должен быть получен за конечное число шагов либо за конечное число шагов мы должны получить указание на неприменимость данной системы указаний для решения интересующей нас задачи. Результативность вызова иногороднего абонента обеспечивается тем, что либо вы получаете необходимое соединение, либо прерывистые гудки сообщают вам, что в данных условиях решение задачи невозможно. Если же вы уже полчаса слышите, в телефонной трубке вкрадчивый и нежный голос, произносящий с равными интервалами «ждите ответа», то результативность не обеспечивается. Для этого следовало бы, например, после двадцатого повторения фразы об ожидании сказать: «Извините, перезвоните еще раз».

Итак, алгоритм — это любая система указаний, которая обладает свойствами *массовости, детерминированности* и *результативности.*

Уточнение алгоритма. В математике понятие алгоритма используется весьма широко. По существу, все математические процедуры, приводящие к решению тех или иных задач, основаны на алгоритмах. Долгое время казалось, что того интуитивного понимания алгоритма, которое изложено выше, математике достаточно. Если кто-то из математиков предлагал новый способ решения той или иной задачи и в явной форме описывал алгоритм ее решения, то вряд ли кого-нибудь надо было убеждать в его существовании.

Среди знаменитых проблем, выдвинутых Давидом Гильбертом, одним из крупнейших математиков XX столетия, была и такая шуточная проблема: «Построить алгоритм, строящий необходимый алгоритм решения любой точно поставленной задачи». Если бы такой универсальный алгоритм был построен, то деятельность математика стала бы механизированной. Получив некоторую задачу и формализовав ее (т. е. точно описав ее на языке математики), он использовал бы процедуру универсального алгоритма и получал бы механически необходимое решение.

Более или менее ясно, что такой универсальный алгоритм существовать не может. Но как это доказать? Ведь для того чтобы точно доказать, что алгоритм не существует, надо иметь точное математическое понятие алгоритма. Так шутка Гильберта стала толчком для поиска математических уточнений понятия «алгоритм»» Их было придумано немало, около двух десятков. Потом было доказано, что все они эквивалентны между собой, а затем крупнейший советский математик А. Н. Колмогоров построил схему, лежащую в основе любого уточнения этого понятия и с помощью которой можно придумывать много новых уточнений.

Нам будет интересно рассмотреть два таких уточнения. Первое из них связано с именем Алана Матисона Тьюринга, английского математика, оказавшего значительное влияние на развитие кибернетики и теории искусственного интеллекта. Им было предложено уточнение понятия алгоритма, получившее название *машины Тьюринга.* Эта «машина» возникла в конце 30-х годов, почти на два десятилетия опередив появление электронных вычислительных машин, теоретическим прообразом которых она является. Машина Тьюринга состоит из разделенной на клетки бесконечной ленты, управляющего устройства, способного перемещаться вдоль ленты, и специальной таблицы управления, задающей функционирование машины. На ленте можно записывать «слова», составленные из символов некоторого фиксированного алфавита. В каждый такт работы управляющее устройство извлекает символ, стоящий в той клеточке, против которой находится «глаз» управляющего устройства. Прочитав символ, управляющее устройство анализирует то состояние, в котором оно в данный момент находится. Список возможных состояний устройства фиксирован заранее и конечен. Пара — прочитанный символ с ленты и номер состояния — позволяет обратиться к таблице управления. Строки этой таблицы закодированы символами, которые могут встретиться на ленте, а столбцы — номерами состояний управляющего устройства. Таким образом, пара, состоящая из прочитанного машиной символа и текущего состояния устройства, определяет единственную клетку таблицы. В ней записана информация о том, что должна сделать машина. Эта информация состоит из указания символа, который необходимо записать на ленту вместо прочитанного, указания состояния, в которое должно перейти управляющее устройство, и указания на его перемещение вправо или влево вдоль ленты. Выполнение этих указаний завершает один такт работы машины Тьюринга.

Проиллюстрируем работу машины Тьюринга на простом примере. Зададим таблицу управления в виде, показанном на стр. 46 (табл. 1): Символ 0 соответствует пустой клетке ленты. Слова языка, которые «понимает» машина Тьюринга, состоят из последовательностей единиц, записанных в соседних клетках ленты без пропусков, либо из последовательности единиц, в которой одна из клеточек, не являющаяся крайней, заполнена символом +. Примерами слов, понятных машине, могут служить 111 или 11 + 1. Слова же 1 + +1 или + 11 машина не понимает. Как видно из таблицы, управляющее устройство может находиться в трех различных состояниях. Состояние q_0 — особое. Оно соот-

Таблица 1

	q_0	q_1	q_2
0	$0q_0$С	$0q_1$П	$1q_0$С
1	$1q_0$С	$0q_2$П	$1q_2$П
+	$+q_0$С		$1q_0$С .

ветствует выключенной машине. Как видно из первого столбца таблицы, в состоянии q_0 символы на ленте сохраняются неизменными, само состояние не меняется и движения нет (символ С означает «стоп»). Два других состояния — рабочие. Работа машины начинается с того, что управляющее устройство переводится в состояние q_1. При этом слово должно располагаться на ленте справа от местоположения «глаза» управляющего устройства. Пусть, например, имеется следующая ситуация:

0 0 1 1 1 + 1 1 0 0.

q_1

Местоположение управляющего устройства отмечено символом состояния q_1. Обратимся к таблице. В состоянии q_1, считав символ 0, машина должна сохранить его, сохранить, свое состояние и сдвинуться вправо на одну клетку ленты (этот сдвиг обозначен в таблице символом П). Эта информация содержится в средней клетке первой строки таблицы управления. После этого возникнет следующая ситуация:

0 0 1 1 1 + 1 1 0 0.

$q1$

Она аналогична предыдущей, и машина Тьюринга на этом такте выполнит сдвиг вправо на одну клетку

ленты, сохранив состояние q_1. В результате возникнет другая ситуация. Считав символ 1 в состоянии q_1, машина стирает его, заменяет на 0, меняет состояние на q_2 и сдвигается вправо на одну клетку ленты. Новая ситуация имеет вид:

0 0 0 1 1 + 1 1 0 0.

q_2

В состоянии q_2, как это следует из таблицы управления, все символы 1 на ленте сохраняются, состояние q_2 также сохраняется, а управляющее устройство сдвигается вправо. Так будет происходить до тех пор, пока не возникнет ситуация:

0 0 0 1 1 + 1 1 0 0.

q_2

Таблица управления в этой ситуации предписывает заменить + на единицу и прекратить работу. Окончательная ситуация на ленте:

0 0 0 1 1 1 1 1 0 0.

Какую же работу выполнила машина? Если договориться интерпретировать слова ее входного языка как записи натуральных чисел, в которых величина числа определяется числом единиц в этой записи, а знак + трактовать как знак арифметического сложения, то машина, заданная нашей таблицей управления, будет осуществлять операцию сложения натуральных чисел. В примере она нашла сумму 3 и 2, получив ответ 5.

Таблица управления описывает алгоритм сложения натуральных чисел, ибо задаваемая ею процедура пригодна для сложения любых натуральных чисел, детерминирована и результативна. Заметим, что если на ленте в качестве исходного слова написано нечто машине непонятное, например + 1 +, то машина может переработать эту запись в слово 11 +, если начальные условия работы такие же, как и при переработке правильных слов. Но интерпретация подобной работы затруднительна.

Другое уточнение понятия алгоритма связано с именем выдающегося советского специалиста в области математической логики А. А. Маркова. В качестве математического объекта, соответствующего интуитивному понятию алгоритма, он предложил специальную запись, названную им *нормальным алгорифмом*)*. Нормальный алгорифм задается упорядоченным списком следующего вида:

$Q_1 \Rightarrow R_1, Q_2 \Rightarrow R_2, \ldots, Q_n \Rightarrow R_n$

Выражение $Q_1 \Rightarrow R_1$, называется *подстановкой.* Пусть дано некоторое слово N, которое «понятно» нормальному алгорифму. Подстановка $Q_i \Rightarrow R_i$ применяется к слову N следующим образом. Если слово Q_i входит в состав слова N, то его первое вхождение заменяется на R_i. Если слово Q_i, в слово N не входит, то N сохраняется неизменным. Пусть, например, мы пользуемся словами русского языка и применяем подстановку «с $\Rightarrow$ антре» к слову «скот». Так как «слово» «с» входит в слово «скот», то подстановка приведет к изменению слова «скот» на слово «антрекот». Если теперь к полученному слову применить подстановку «кот $\Rightarrow$ пренер», то мы получим из слова «антрекот» слово «антрепренер».

Упорядоченность подстановок в записи нормального алгорифма не прихоть. Существует жесткий порядок выполнения нормального алгорифма. К исходному слову N сначала всегда применяется первая по порядку подстановка. Если она изменила слово, то измененное слово рассматривается как новое и к нему вновь применяется первая по порядку подстановка. Так будет до тех пор, пока ее применение не изменит слова. Только после этого начнет работать вторая подстановка. А третья сможет начать работать, если ни первая, ни вторая подстановка не меняют слова. Если же все подстановки, образующие нормальный алгорифм, не меняют слова, то оно считается окончательным и применение подстановок прекращается.

Такие жесткие правила применения подстановок обеспечивают детерминированность процедуры, задаваемой нормальным алгорифмом. То, что эта процедура обладает свойством массовости, также, по-видимому, не вызывает сомнений. Слова *N,* к которым применяется заданная система подстановок, не фиксированы.

*) При переводах с греческого языка буква 0 заменяется буквамп «т» ют «ф». Мы пишем «театр», а не «феатр», но «арифметика», а не «аритметпка». Жесткого правила замены букв нет. А. А. Марков и его ученики пишут «алгорифм», но большинство придерживается написания «алгоритм». Поэтому мы сохраним букву «ф» только в названии конструкции, предложенной А. А. Марковым.

Менее очевидно выполнение требования результативности. Однако путем введения в правые части подстановок (там, где это необходимо) специальных указателей прекращения выполнения алгорифма, можно добиться выполнения и этого требования.

Тезис об универсальности. Все авторы уточнений интуитивного понятия алгоритма формулируют некоторый тезис, который можно назвать тезисом об универсальности. Для машин Тьюринга и нормальных алгорифмов Маркова эти тезисы звучат следующим образом.

Каков бы ни был алгоритм в интуитивном смысле, для него можно построить такую машину Тьюринга, которая будет выполнять ту же работу, что и алгоритм в интуитивном смысле.

Каков бы ни был алгоритм в интуитивном смысле, для него можно построить такой нормальный алгорифм Маркова, который будет выполнять ту же работу, что и алгоритм в интуитивном смысле.

Верны и обратные утверждения. Все, что реализуют любые машины Тьюринга или нормальные алгорифмы, есть процедуры, совпадающие с тем, что подразумевают люди, когда говорят об алгоритмах.

Важность этих утверждений очевидна. В человеческой деятельности алгоритмические процедуры встречаются весьма часто. И точное описание протекания подобных процедур, полная их формализация позволяют строить теорию алгоритмических процедур и открывает пути к их воспроизведению техническими устройствами.

В теории алгоритмов тезис об универсальности необходимо принимать на веру. Доказать в строгом математическом смысле его, конечно, нельзя. Однако при любом уточнении понятия алгоритма можно, доказать теорему о существовании универсального имитирующего алгоритма. Поясним это на примере машины Тьюринга.

Одна машина Тьюринга отличается от другой числом различных состояний управляющего устройства, числом различных символов, которые можно использовать для представления информации на ленте, и тем, как устроена таблица управления. Размер таблицы управления определяется произведением числа ее строк на число столбцов, т. е., другими словами, произведением числа различных допустимых символов на ленте на число различных допустимых состояний управляющего устройства. Обозначим это произведение буквой S и будем характеризовать этим числом сложность машины Тьюринга. Например, сложность машины Тьюринга, выполняющей сложение двух любых натуральных чисел, которую мы обсуждали, равна 6 (именно 6, а не 9, так как столбец, соответствующий состоянию q_0, можно в таблицу не записывать).

Пусть теперь у нас две машины Тьюринга, которые мы обозначим через T_1 и T_2. Закодируем таблицу управления машины T_1 в символах ленты машины T_2. Затем запишем этот код и некоторое входное слово N машины T_1 в качестве входного слова для T_2. Запустим T_2 в работу. Если процедура будет результативной и через конечное число шагов мы получим на ленте T_2 некоторое конечное слово N', то сравним N' с тем конечным словом, которое должно было бы получиться при обработке слова N машиной T_1 Если результаты обработки совпадают между собой (с точностью до кодирования) для произвольного слова N, то говорят, что машина T_2 имитирует работу машины T_1.

Для имитации необходимо иметь машину T_2 с вполне определенной таблицей управления, зависящей от принятого способа кодирования имитируемой машины. Этот процесс кодирования мы будем называть *программированием.* Таким образом, программирование есть процесс кодирования некоторого алгоритма, отраженного в таблице управления машины T_1 на языке, понятном машине T_2.

Вопрос о существовании имитирующей машины для некоторой машины Тьюринга, конечно, интересен. Но еще более интересен вопрос о возможности существования универсальной имитирующей машины Тьюринга. Такая машина, если она существует, способна с помощью программирования имитировать работу любой конкретной машины.

Одним из важнейших результатов теории алгоритмов является доказательство существования такой универсальной имитирующей машины Тьюринга. Более того, предложено несколько таблиц управления конечной сложности, реализующих эту универсальную машину. Остается открытым вопрос о минимальной сложности такой универсальной имитирующей машины, но и примеры подобных машин, в которых S лежит в пределах

нескольких десятков, вполне убедительны. Одновременно доказано, что программирование таблиц управления имитируемых машин и слов, подаваемых на ленту универсальной имитирующей машины Тьюринга, само по себе не требует процедуры такой же сложности, что и имитация.

Аналогичные доказательства существуют и для других способов уточнения понятия алгоритма. В частности, существует универсальный имитирующий нормальный алгорифм Маркова. С его помощью может быть реализован запрограммированный соответствующим образом любой нормальный алгорифм, предназначенный для решения определенной задачи.

Исчисления и алгоритмы. Алгоритмические процедуры представляют собой образец четкости и ясности. Детерминированность и результативность, свойственные им, регламентируют получение однозначных результатов при одинаковых исходных данных. Если бы человек в своей повседневной жизни мог пользоваться только подобными процедурами решения возникающих перед ним задач, то поведение его было бы полностью предсказуемым, ибо все люди в одной и той же ситуации при наличии одной и той же конечной цели, используя одинаковые алгоритмы, действовали бы с удручающим однообразием.

...Большой и шумный многоэтажный универмаг. На первом этаже очередь к лифтам. Когда кабина наполняется, лифтер нажимает кнопку и лифт стремительно несется к верхнему этажу. Здесь, как правило, кабина освобождается, и лифтер, нажав другую кнопку, спускается за очередной группой покупателей. Процедуры, которые выполняет лифтер, чисто алгоритмические. Поэтому сам он как личность не оказывает на процесс перевозки никакого влияния. Работа эта однообразна и утомительна. Лифтеры часто меняются для отдыха, но реализация процедур от этого не изменяется. Лифтера вообще может не быть. Роль лифтера может с таким же результатом выполнить любой человек, находящийся в кабине лифта.

Во многих социальных антиутопиях алгоритмически регламентированное поведение членов общества подчеркивает бездушие и унизительность подобного существования. Однако алгоритмизация принятия решений и поведения очень часто бывает жизненно необходимой. Она нужна тогда, когда у человека нет времени на рассуждения, обдумывание вариантов и выбор альтернатив. Она нужна и в тех случаях, когда заведомо обеспечивает наилучший путь к цели, достижение которой необходимо для человека. И она нужна, как правило, тогда, когда мы хотим заменить человеческий труд с помощью технических устройств.

Попробуем ослабить жесткие требования алгоритма, что позволит рассмотреть более гибкие процедуры. Отказ от массовости ничего особенно нового не дает. Просто система предписаний станет применимой лишь в одном фиксированном случае. Но то, что мы храним схемы поиска квартир друзей и знакомых, показывает важность и таких систем предписаний.

Более перспективен отказ от детерминированности. Это даст нам более гибкую процедуру. Можно, например, вместо нормального алгорифма рассматривать такую же систему подстановок, но выбор подстановок делать не по той строго однозначной схеме, которая предписывается А. А. Марковым.

Припишем каждой подстановке $Q_i \Rightarrow R_i$ вероятность ее выбора, которую мы будем обозначать как h_i. Реализация процедуры будет организована следующим образом. Пусть имеется исходное слово N. Организуем процесс получения значений Случайной величины v, принимающей значения от 1 до n с вероятностями $h_1, h_2, \ldots, h_n$. Получаемые на каждом шаге значения этой величины указывают на ту подстановку, которую нужно применить к слову. Если, например, система подстановок имеет вид «т$\Rightarrow$к», «о$\Rightarrow$а», «пар$\Rightarrow$бак» и исходное слово есть слово «топор», то нормальный алгорифм последовательно переработает его в слова: «копор», «капор», «капар», «кабак». На этом применение нормального алгорифма закончится. Если же вместо нормального алгорифма мы возьмем вероятностную схему выбора подстановок, то результат, конечно, не будет однозначным. Организуем вычисление случайной величины v, принимающей значения 1, 2 и 3 по числу подстановок, следующим образом. На каждом шаге будем подкидывать две монеты. Условимся, что если обе монеты выпадают гербом кверху, то случайная величина v принимает значение 1. Если обе монеты выпадают гербом книзу, то v принимает значение 3. В тех случаях, когда монеты выпадают разными сторонами

кверху, v принимает значение 2. При этом величины вероятностей h_1 и h_3 равны 0,25, а вероятность h_2 имеет значение 0,5.

Начнем подкидывать монеты. Пусть в первый раз $v = 2$. Тогда слово «топор» преобразуется в слово «тапор». Бросим монеты еще раз. Пусть снова $v = 2$. Преобразуемое слово примет вид «тапар». Еще раз подкидываем монеты. Пусть $v = 3$. Теперь новое полученное слово будет «табак». Процесс можно продолжать и дальше. Но уже сказанного достаточно, чтобы понять отличие нашей процедуры от нормального алгорифма. Будем называть такие процедуры *вероятностными алгоритмами.* В вероятностных алгоритмах, если вероятности h_i объективно заданы процессом получения значений случайной величины v, можно априорно вычислить вероятность того или иного преобразования исходного слова N за заданное число шагов. Поэтому вероятностные алгоритмы как бы объективируются этими априорными оценками.

Но можно еще более ослабить жесткость задания процедуры. Можно, например, считать, что объективных значений h_i не существует. Например, разные люди пользуются разными соображениями при выборе очередных подстановок, т. е. допускается полный произвол в выборе подстановки на каждом шаге процесса. Процедуры подобного типа получили название *исчислений.* В исчислениях обычно фиксируются множество исходных слов и система подстановок. А исследуется то множество результативных слов, которое возникает при использовании этих подстановок в любых комбинациях для преобразования множества исходных слов.

Читатель вправе спросить: «Зачем нужна столь слабо организованная процедура и для чего надо изучать свойства конечного множества результатов?»*). Ответ на этот вопрос, интересный для целей нашей книги, состоит в следующем. Уже упоминалось о формализации силлогистики Аристотеля. Но что представляет собой силлогистика? Она состоит из множества априорно заданных утверждений, истинность которых предполагается (например, «Все люди смертны» или «Гай Юлий Цезарь —человек»), и набора правил вывода заклю-

*) Хотя ясно, что она может быть полезной, например, при игре в «Эрудита» или «Балду».

чений из двух посылок, обеспечивающих истинность результата. Другими словами, силлогистику можно превратить в исчисление. Правда, в строгом смысле это удалось сделать относительно недавно известному польскому логику Я. Лукасевичу. Кроме исчисления силлогизмов, после появления математической логики активно изучались исчисления высказываний и исчисления предикатов. Опишем очень кратко суть этих исчислений, играющих значительную роль в современной теории искусственного интеллекта.

Будем называть *высказыванием* любое утверждение, относительно которого в данный момент можно сказать, что оно является истинным или ложным. Например, утверждение «В книге, которую вы сейчас читаете, 30 страниц» является ложным, а утверждение «Аристотель был греком» является истинным. Заметим, что не все предложения естественного языка могут быть интерпретированы как истинные или ложные. Что мы можем сказать в связи с этим о предложении «Пробежать стометровку за 13 секунд легко и приятно»? Истинность этого утверждения зависит от того, кто его оценивает.

Высказывания будем обозначать малыми латинскими буквами. Договоримся также условно писать, что высказывание равно 0, если оно ложно, и равно 1, если оно истинно. Такая договоренность позволит нам ввести специальную алгебру высказываний. Для этого используются специальные операции над высказываниями: отрицание (черта над буквой), конъюнкция ($\wedge$) дизъюнкция ($\vee$) и импликация ($\rightarrow$). Эти четыре операции задаются табл. 2.

Поясним сначала смысл операции отрицания. Из табл. 2 следует, что если некоторое высказывание a ложно, т. е. $a = 0$, то отрицание этого высказывания, обозначенное через a, является истинным, т. е. $a = 1$.

Таблица 2

a	b	▯	b	$a \wedge b$	$a \vee b$	$a \rightarrow b$
0	0	1	1	0	0	1
0	1	1	0	0	1	1
1	0	0	1	0	1	0
1	1	0	0	1	1	1

В языке этому соответствует отрицательная конструкция. Пусть a есть высказывание «Аристотель был греком». Для него мы можем написать: $a = 1$. Отрицанием этого высказывания является утверждение «Аристотель не был греком». Ясно, что $a = 0$. Операция отрицания относится к одному высказыванию. Остальные три операции оперируют с двумя высказываниями и зависят от истинности и ложности пары высказываний.

Конъюнкция в языке соответствует, как правило, соединительному союзу «и». Обозначим, как и ранее, через a высказывание «Аристотель был греком». Через b обозначим высказывание «Сократ был византийцем». Конъюнкция этих двух высказываний имеет вид: «Аристотель был греком и Сократ был византийцем»*). При этом $a = 1$, $b = 0$ и, как это правильно определяет таблица $ab = 0$ (здесь и далее для компактности записи знак конъюнкции опускается.)

Несколько труднее интерпретировать языковыми примерами операцию дизъюнкции. Наиболее часто в языке она выражается неразделительным значением слова «или», т. е. таким его употреблением, когда нет. альтернативного выбора (или одно, или другое), а допускается возможность одновременного выбора и того, и другого. Рассмотрим утверждение «Сократ был греком или философом». Здесь «или» соединяет два утверждения, которые мы можем написать отдельно: «Сократ был греком» и «Сократ был философом», обозначив их соответственно через a и b. В нашем случае $a = b = 1$ и, согласно таблице определения логических операций, все высказывание $a \vee b$ является истинным.

Операции импликации соответствует в естественном языке конструкция «Если а, то *b»,* хотя в явной форме эта конструкция может быть выражена и другими словами. Вспомним разговор, который происходит между Алисой, Зайцем и Шляпочником во всемирно известной сказке Льюиса Кэрролла «Приключения Алисы в Стране чудес»:

"—Добавьте-ка себе чаю,—с величайшей серьезностью обратился к Алисе Заяц.

*) Конечно, с точки зрения литературной нормы следовало бы сказать «Аристотель был греком, а Сократ был византийцем», но для нас более удобно использовать союз «и» вместо союза «а». **(зачеркнутое читайте как надчеркнутое)**
$a \equiv a$ (пример)- прим. сканирующего

— Мне добавлять не к чему,— обиженно сказала Алиса.— Мне еще никто ничего не наливал.

— От ничего нельзя убавить,— сказал Шляпочник. — А добавлять к нему можно сколько угодно".

Преобразуем последние слова Шляпочника следующим образом. Через a обозначим высказывание «Здесь ничего нет», а через b — высказывание «Сюда можно нечто добавить». В высказывании Шляпочника $a = 1$ и $b = 1$, а следовательно, как показывает наша таблица, и все высказывание Шляпочника $(a \Rightarrow b) = 1$. Но это еще не все, что сказал Шляпочник Алисе. Он указал еще на ошибочность утверждения «Если здесь ничего нет, то сюда нельзя нечто добавить». Другими словами, он указал на то, что при $a = 1$ и $b = 0$ все заключение $a \to b$ ложно. Этот факт также отражен в таблице определения импликации.

Но самым трудным истолкованием импликации является ее истинность при ложности a. Так, например, в ситуации, когда Алиса принимает участие в «Чаепитии со сдвигом», оказываются истинными два высказывания: «Если неверно, что здесь ничего нет, то сюда можно нечто добавить» и «Если неверно, что здесь ничего нет, то сюда нельзя ничего добавить».

Заметим, что с точки зрения приписывания истинности и ложности высказываниям операция импликации не является самостоятельной. Имеет место равенство: $a \to b = a \vee b$. Проверить это можно, построив таблицу интерпретации для операции $a \vee b$ и сравнив полученный столбик значений со столбиком, определяющим импликацию *).

Традиционно в классических исчислениях высказываний сохраняют все четыре операции, допуская некоторую избыточность. Введенные нами операции позволяют строить формулы сложных высказываний.

В рассказе «Мой первый роман» Шолом-Алейхем пишет о том, как его, молодого, веселого и голодного, наняли в учителя к сыну богатого деревенского жителя (как пишет Шолом-Алейхем, *«...думается мне, вы вовсе не обязаны знать, что у меня есть дядя, а у дяди есть тетя, что у тети есть знакомый, а у знакомого есть родственник, что у родственника* — свояк...»). Юный ученик, не желавший набираться «ученой мудрости», сформулировал свои требования к учителю в следующей тираде:

«Вот что: если вы хотите остаться у нас, если вы хотите, чтобы мы стали друзьями, если вы не хотите, чтобы вам пришлось уезжать отсюда, забросьте книги под стол... Будем играть в шашки, в „шестьдесят шесть" или давайте валяться на кроватях и плевать в потолок».

Представим это красочное высказывание будущего богача в виде некоторой логической формулы. Введем обозначения для отдельных высказываний. Пусть a есть высказывание «хотеть остаться в доме богатого свояка», b— высказывание «хотеть стать другом сына свояка», a — «хотеть уезжать из дома свояка» (ясно, что a и a противоположны с точки зрения истинности, истинность одного влечет ложность другого и наоборот в условиях двоичной логики рассуждений), c — «забросить книги под стол», d — «играть в шашки», e — «играть в „шестьдесят шесть"», f — «валяться на кровати», g — «плевать в потолок». Тогда с некоторым допущением потери оттенков тирада ученика выразится следующей формулой:

$$aba \to c(d \vee e \vee fg).$$

Исчисление высказываний строится следующим образом. Из множества возможных формул выбирается конечное подмножество таких формул, которые являются тождественно истинными. Истинность этих формул не меняется, как бы ни менялась истинность входящих в них высказываний. Примером такой формулы может служить $a \vee a$. Таблица определения дизъюнкции показывает, что она ложна только в том случае, когда ложны оба ее аргумента. А для формулы $a \vee a$ этот случай невозможен. Эти выбранные формулы называют *аксиомами.*

Кроме того, добавляются правила вывода, с помощью которых из одних формул могут выводиться другие. Примером традиционного правила вывода для исчисления высказываний может служить правило *то-*

*) На самом деле, можно показать, что $a \vee b = \underline{ab}$ или $ab = \underline{a \vee b}$. И независимыми являются лишь пары операций: отрицание и конъюнкция или отрицание и дизъюнкция. Кроме того, поскольку $a \to 0 = a$, то нуль и импликация также могут быть исходными при построении других операций.

dus ponens, восходящее еще к силлогистике Аристотеля. Это правило имеет следующий вид: «Если формула A и формула $A \to B$ выводимы, то выводима формула B». Большие буквы в этой записи обозначают не отдельные высказывания, а сколь угодно сложные формулы. Правила вывода строятся так, чтобы их применение не нарушало тождественной истинности формул. Другими словами, если исходные формулы, к которым применяется правило вывода, являются тождественно истинными, то и выводимые формулы должны быть такими же. Обычно к системе аксиом и правил вывода предъявляются два требования: полноты и непротиворечивости. Требование полноты означает, что имеющийся набор аксиом и правил вывода обеспечивает выводимость любой возможной тождественно истинной формулы в исчислении высказываний. А требование непротиворечивости говорит о том, что наряду с этими тождественно истинными формулами не могут выводиться тождественно ложные формулы. Правда, требование непротиворечивости автоматически выполняется, если исходные аксиомы тождественно истинны, а правила вывода сохраняют истинность формул.

Существует великое множество конкретных систем аксиом и правил вывода, обеспечивающих требования полноты и непротиворечивости.

Вспомним основную задачу Аристотеля: найти формальные приемы сохранения истинности выводов при истинности исходных посылок. В исчислении высказываний эта идея реализована полностью. Однако силлогистика Аристотеля шире исчисления высказываний. Она объемлет его. И, как уже говорилось, лишь совсем недавно Я. Лукасевич построил формальное расширение исчисления высказываний, позволившее окончательно решить задачу, возникшую на заре человеческой научной мысли.

Это расширение связано с введением в исчисление высказываний кроме символов, обозначающих отдельные высказывания, еще двух элементов, характерных для силлогизмов Аристотеля. Один из них имеет вид Aab и соответствует утверждению «все a суть b», а второй имеет вид Iab и соответствует утверждению «некоторые a суть b». К системе аксиом исчисления высказываний, обладающей свойствами полноты и непротиворечивости, добавляются еще четыре аксиомы:

1. Все a суть a: $A(a, a)$.
2, Некоторые a суть a: $I(a, a)$.
3. Если все a суть b и все b суть c, то все a суть c: $(A(a, b)A(b, c)) \to A(a, c)$.
4. Если все a суть b и некоторые b суть c, то некоторые a суть c: $(A(a, b)I(b, c)) \to I(a, c)$.

Отрицания А интерпретируются соответственно как «Ни одно a не есть b» и «Некоторые a не есть b».

Заметим, что третья аксиома Я. Лукасевича представляет собой формализованную запись силлогизма следующего типа:

1. Все люди смертны.
2. Греки — люди.
3. Греки смертны.

Четвертая аксиома, предложенная в исчислении Я. Лукасевича, также представляет собой силлогизм Аристотеля, но несколько иного вида. Примером его может служить силлогизм:

1. Все сладкоежки любят конфеты.
2. Некоторые люди сладкоежки.
3. Некоторые люди любят конфеты.

В исчислении Я. Лукасевича выводятся все силлогизмы, которые в силлогистике Аристотеля признаются правильными. А неправильные силлогизмы не выводятся.

Но, как уже говорилось, развитие исчисления высказываний шло не по пути приближения его к силлогистике Аристотеля, а по пути, использующему общие для математики фундаментальные идеи, связанные с понятиями множества и отношений между элементами, входящими в различные множества, т. е. с понятиями функции и функциональной зависимости. Именно на этом пути возникло и развилось исчисление предикатов.

Исчисление предикатов как бы расширяет исчисление высказываний. В нем наряду с высказываниями, истинность и ложность которых фиксирована, вводятся еще переменные высказывания, которые могут быть истинными или ложными в зависимости от того, какое значение принимает переменная. Рассмотрим, например, утверждение «x был греком». Что можно сказать об его истинности? Конечно, ничего, ибо кто такой x. пока остается неясным. Чтобы ответ на поставленный вопрос приобрел бы некоторый смысл, необходимо задать область определения x. Пусть, например, x — это множество, состоящее из четырех имен: Аристотель, Платон, Сократ, Фома Аквинский. Подставляя вместо x любое из этих имен, мы получим высказывания, три из которых будут истинными, а одно, относящееся к Фоме Аквинскому,— ложным.

Утверждение вида *«x был греком»* называется *предикатом.* Обозначается оно $P(x)$, где P есть символ предиката «был греком». Предикат может зависеть не от одной переменной, а от нескольких. Например, предикат *«x старше y»* зависит от двух переменных, а предикат *«x* находится между y и z» зависит от трех переменых. После одного из возможных означиваний переменных в этом последнем предикате мы можем получить, например, высказывание «Москва находится между Дели и Токио», которое является ложным.

В исчислении предикатов используются два квантора, которые носят название *квантора общности* и *квантора существования.* Их традиционное обозначение соответственно $\forall$ и $\exists$. Первый из них в формуле $\forall x P(x)$ читается «для всех x имеет место $P(x)$», а второй квантор в формуле $\exists x P(x)$ читается «существует такое x, что $P(x)$...».

Исчисление предикатов включает в себя все формулы обычного исчисления высказываний, а также формулы, содержащие кроме символов высказываний еще предикатные символы или предикатные символы с кванторами.

Для иллюстрации формул в исчислении предикатов рассмотрим запись в этом виде известного афоризма Козьмы Пруткова:

«Нет столь великой вещи, которую не превзошла бы величиною еще большая; нет вещи столь малой, в которую не вместилась бы еще меньшая».

Если через x, y и z обозначить классы соответствующих вещей, то афоризм Пруткова приобретет следующий формальный вид:

$$(\forall x \exists y P(y,x))(\forall x \exists z Q(z,x)).$$

Здесь предикат $P(y, x)$ есть *«y* больше *x»*, а предикат $Q(z, x)$ — «z меньше *x»*.

Предикат естественно считать тождественно истинным, если он сохраняет истинность при любом означивании входящих в него переменных. Взяв в качестве аксиом аксиомы исчисления высказываний, обладающие свойствами полноты и непротиворечивости, и добавив к ним несколько новых аксиом, связанных с предикатами и кванторами, можно получить исчисление предикатов. В этом исчислении будут выводиться все тождественно истинные предикаты и только они.

Отметим одну важную для нас особенность исчисления предикатов. Казалось бы, что силлогистика Аристотеля должна быть частью этого исчисления, так как символы A и I, интерпретируемые как «все a суть b» и «некоторые a суть b» можно записать в виде предикатных формул следующего вида:

$$\forall x(P(x) \to Q(x)) \quad \text{и} \quad \exists x\,(P(x) \to Q(x)).$$

При этом аксиомы, добавленные Я. Лукасевичем к исчислению высказываний, станут предикатными формулами. Исчисление Я. Лукасевича можно было бы заменить исчислением предикатов. Но этот переход оказывается неверным. Подобная интерпретация символов A и I сужает их применимость, так как предикаты не допускают определения на пустых множествах, а символы A и I допускают такое использование.

Квазиалгоритмы. Приставка «квази» означает «как бы». Таким образом, *квазиалгоритмы* — это процедуры, похожие на алгоритмы, по не являющиеся ими. Мы уже столкнулись с квазиалгоритмами, когда рассматривали вероятностные алгоритмы, ибо для этих процедур не выполняется требование детерминированности. Квазиалгоритмами являются и исчисления, в которых порядок применения правил вывода не фиксирован. Кроме того, в исчислениях, как правило, нет никакой возможности узнать заранее, выводима ли данная формула в нем пли нет. Именно поэтому основное внимание исследователей привлекали исчисления, в которых множество выводимых формул обладало бы определенной особенностью, которая допускала бы априорную проверку. Это имеет место в рассмотренных нами исчислениях, выводящих лишь тождественно истинные формулы и обладающих свойством полноты.

Пусть нам дана формула F исчисления высказываний или исчисления предикатов. Если мы можем проверить ее на тождественную истинность, то этого достаточно для ответа на вопрос, выводится ли F в данном исчислении или нет. Другое дело, что подобная проверка сама может оказаться весьма громоздкой процедурой.

Однако существуют и другие квазиалгоритмы, представляющие большой интерес для описания процедур, присущих человеку и имитирующим его искусственным системам. Прежде всего это так называемые *размытые алгоритмы* *).

Чаще всего с размытыми алгоритмами сталкивается домашняя хозяйка, когда она к приходу гостей собирается приготовить что-нибудь, руководствуясь тем или иным пособием по кулинарии. Предположим, что некоторое время тому назад она вместе с друзьями совершила незабываемое путешествие по Армении. До сих пор вспоминают они удивительную синеву неба Армянского нагорья, сахарный конус Арарата, вечность и одухотворенность каменных плит — хачкаров. Стройные силуэты армянских церквей и шумные улицы Еревана сливаются в их памяти воедино. И еще — армянская кухня с ее удивительными и неожиданными вкусовыми сочетаниями.

Конечно, друзьям будет приятно, если их в Москве угостят каким-нибудь армянским блюдом, которое напомнит им прекрасное путешествие. Наша гипотетическая хозяйка принимает решение приготовить что-либо из армянской кухни, например знаменитый суп под названием «спас», который так им всем понравился у друзей в Ереване. Итак, спас. Она открывает книгу кулинарных рецептов и читает следующее: «Пять столовых ложек муки развести с яйцом, тщательно взбить. Один литр мацуна развести пополам с водой. Соединить обе смеси, постепенно вводя вторую в первую. Поставить смесь на огонь и, непрерывно помешивая, довести до кипения. Отварить полстакана риса до мягкости, но так, чтобы рис не разварился. Поджарить на сливочном масле три мелко нарезанные луковицы до желтого цвета. Приготовить мелко нарубленную мяту (полстакана) и кинзу (стакан). Добавить в смесь лук, зелень и снова варить, доведя один раз до кипения».

*) Термин «размытые» происходит от английского слова *fuzzy*. В настоящее время существует теория размытых множеств, размытая логика, размытая топология и т. п. Основателем этой специфической математики был известный американский ученый Лотфи Заде.

По форме эта система указаний задает некоторую процедуру с определенной последовательностью шагов. Одни из этих шагов (например, «Один литр мацуна развести пополам с водой») однозначны и не вызывают трудностей. Но другие могут пониматься не единственным образом. Что, например, означает «тщательно взбить»? Где указания на момент прекращения процесса взбивания? И что значит отварить рис «до мягкости, но так, чтобы рис не разварился»? Какой момент в жарке лука соответствует тому, что он приобрел желтый цвет? Ответы на все эти вопросы расплывчаты, неточны. Обычно говорят, что выполнение этих этапов процедуры требует интуиции, чего-то трудно выразимого в точных предписаниях. И недаром при выполнении подобных кулинарных рецептов отсутствует детерминированности. Одно и то же блюдо, приготовленное на основании одного и того же рецепта двумя людьми, будет в чем-то неуловимо (а иногда и ощутимо) различаться. Все кулинарное искусство держится на грани каких-то «чуть-чуть», никак не формализуемых и недетерминированных.

Квазиалгоритмы расплывчатого типа встречаются не только в кулинарии. Указания типа «Пройди несколько шагов», «Нажми на кнопку сильнее», «Делай эти упражнения изредка» весьма «человечны». Но все они размыты, ибо неясно, что такое «несколько», непонятно в строгом смысле этого слова, что значит «сильнее» или «изредка».

С другой стороны, ясно, что в этих указаниях, как бы расплывчаты и неопределенны они ни были, содержится позитивная информация, которую человек использует в своей практической деятельности. И этим они отличаются от указаний типа «Пойди туда, сам не знаю, куда. Принеси то, сам не знаю, что». В них чувствуются некоторые глубинные правила, не меняющиеся произвольно в процессе исполнения.

Алгоритмические языки. Формализация описания процедур происходит в рамках некоторого языка. Например, для описания исчислений мы использовали язык математической логики, а для описания алгоритмов — язык таблиц управления машин Тьюринга или язык подстановок нормальных алгорифмов. Такие языки являются узконаправленными на описание процедур того или иного типа. В отличие от них алгоритмические языки, бурное развитие которых связано с появлением и совершенствованием ЭВМ, претендуют на определенную широту области своего применения.

Основная идея любого алгоритмического языка базируется на разделении данных и операторов или того, «над чем совершаются действия», и того, «что именно делается». Каждый оператор имеет множество входных данных я множество выходных данных и преобразует по присущему ему правилу входные данные в выходные. Это преобразование может быть сколь угодно сложным, но фиксированным в описании оператора.

Операторы бывают двух типов. Одни из них производят преобразование информации, другие определяют порядок выполнения операторов первого типа. Это разделение содержательно, так как и те и другие операторы вырабатывают выходные данные. Но для операторов первого типа эти выходные данные суть некоторые промежуточные результаты процедуры, которые могут использоваться в качестве входных в других операторах. А для операторов второго типа выходными данными являются номера операторов, которые должны будут далее выполняться. Будем обозначать операторы первого типа через A_i, операторы второго типа— через P_i В скобках при операторах первого типа будем писать множества их исходных и выходных аргументов, разделяя эти два множества точкой с запятой. Для операторов второго типа также будем записывать два множества, но второе множество всегда будет состоять из одного элемента — номера некоторого оператора. Номер оператора — это его порядковый номер в записи на алгоритмическом языке. И пока в этой записи не встретится оператор второго типа, выполнение операторов идет согласно порядку их номеров. Оператор второго типа может нарушить этот порядок. Происходит это следующим образом. Пусть оператор второго типа $P(x_1, x_2, \dots, x_n; i)$ проверяет выполнение некоторого условия, зависящего от значений $x_1, x_2, \dots, x_n$. Если это условие выполнено, то данный оператор никакого влияния на процедуру не оказывает. Просто начинает выполняться оператор со следующим порядковым номером в записи процедуры. Если же проверяемое условие не выполняется, то следующим будет выполняться оператор с номером i.

Введем вспомогательный оператор $B(u; v)$, который носит название *оператора присваивания.* Его работа состоит в замене символа u на символ v, т. е. как бы в выполнении операции отождествления этих двух различных символов.

Проиллюстрируем записи трех процедур на алгоритмическом языке предложенного нами типа. В качестве первого примера возьмем процедуру вычисления значения $y = x^2 + \sin x + e^x$ при некотором фиксированном значении $x = a$. Пусть в нашем распоряжении имеются четыре оператора первого типа:

$$A_1(g;\ h),\ \text{для которого}\ h = g^2,$$

$$A_2(k;\ l),\ \text{для которого}\ l = \sin k,$$

$$A_3(m;\ n),\ \text{для которого}\ n = e^m,$$

$$A_4(p,\ q,\ r;\ t),\ \text{для которого}\ t = p + q + r.$$

Тогда процедуру вычисления y при $x = a$ можно представить в виде следующей записи:

$$B^1 (a; g)\, A_1{}^2 (g; h)\, B^3 (a; k)\, A_2{}^4 (k; l)\, B^5 (a; m)\, A_3{}^6 (m; n)$$
$$B^7 (h; p)\, B^8 (l; q)\, B^9 (n; r)\, A_4{}^{10} (p, q, r; t)\, B^{11}(t; y).$$

Как видно из этой записи, в ней используются лишь операторы первого типа и операторы присваивания (порядковые номера операторов в процедуре отмечены верхними индексами).

В качестве второго примера опишем процедуру, имитирующую процесс возникновения и угасания условного рефлекса. Сначала опишем модель этого процесса, известного каждому еще со школьных времен. Имеются два раздражителя ρ_1 и ρ_2, которые соответственно носят название *безусловного* и *условного.* Безусловный раздражитель ρ_1 всегда вызывает у организма реакцию R. Например, в классических опытах И. П. Павлова в качестве раздражителя ρ_1 выступала пища, а реакция R характеризовалась выделением желудочного сока у собаки при виде пищи. Условный раздражитель, вообще говоря, реакцию R не вызывает. Примером такого раздражителя может служить звонок. Но если в течение длительного времени оба раздражителя появляются одновременно (звонок всегда сопровождает появление пищи), то в некоторый момент возникает условный рефлекс и на условный раздражитель ρ_2, и появление только этого раздражителя также вызывает реакцию *R*.

Если теперь длительное время подавать только условный раздражитель ρ_2, то наступит момент угасания условного рефлекса, т. е. наступит такой момент, когда появление ρ_2 в первый раз не вызовет реакции *R*.

Попробуем построить модель описанного процесса. Введем переменную величину q_i индекс которой меняется от опыта к опыту. Если в данном опыте действует только раздражитель ρ_1, то q_i не меняет своего значения. Если в опыте одновременно действуют оба раздражителя ρ_1 и ρ_2, то величина q_1 увеличивается на некоторое фиксированное число *b*. Если в данном опыте действует только раздражитель ρ_2 и рефлекс еще не угас, то величина q_i уменьшается на некоторую величину *d*. Если же рефлекс угас, то q_i не меняет своего значения. Начальное значение q_i есть q_0 — оно характеризует априорную расположенность данного подопытного животного или человека к установлению условных рефлексов.

Условием возникновения рефлекса является выполнение неравенства $q_i > q$, где константа q — характеристика реакции *R* и раздражителей ρ_1 и ρ_2, используемых в опытах.

Введем, наконец, случайную величину *v*, принимающую положительные и отрицательные значения из некоторого симметричного относительно нуля отрезка. Закон распределения *v* произволен. Единственным ограничением является то, что математическое ожидание *v* предполагается равным нулю. Значение *v* на каждом шаге эксперимента будет прибавляться к q_i. Величина *v* характеризует недетерминированность процесса установления и угасания условного рефлекса. Она вносит определенные искажения в процесс увеличения и уменьшения значений q_i. Требование нулевого математического ожидания для *v* содержательно означает, что для достаточно больших интервалов эксперимента ее влияние на исход опытов будет близким к. нулю.

Переходя к алгоритмической записи процесса установления и угасания условного рефлекса, договоримся считать, что наличие раздражителя ρ_i в данном опыте

соответствует равенству ρ_i единице, а отсутствие соответствующего раздражителя эквивалентно тому, что ρ_i принимается равным нулю.

Введем оператор второго типа $P_1(x;\ i)$. Смысл этого оператора заключается в следующем: он проверяет выполнение равенства $x = 1$. Если это равенство не выполняется, то следующий выполняемый в записи оператор имеет номер *i*. Начнем строить описание процедуры. Для простоты мы не будем в явном виде выписывать операторы присваивания. Любознательные читатели могут с легкостью сделать это сами.

, На каждом шаге эксперимента возможны четыре комбинации подаваемых раздражителей: $\rho_1 = \rho_2 = 0$; $\rho_1 = \rho_2 = 1$; $\rho_1 = 1$, $\rho_2 = 0$; $\rho_1 = 0$, $\rho_2 = 1$. Первый случай соответствует отсутствию каких-либо раздражителей. Ясно, что в этой ситуации ничего не должно происходить, величина q_i должна сохраняться, а реакция *R* не возникать. Во втором случае мы имеем процесс установления и закрепления условного рефлекса. Реакция *R* должна обязательно вызываться, поскольку присутствует безусловный раздражитель ρ_1, а величина q_i должна увеличиваться. В третьем случае есть только безусловный раздражитель. Это значит, что реакция *R* должна появиться, а значение q_i меняться не должно. Наконец, Последний случай соответствует процессу угасания условного рефлекса. Величина q_i должна быть уменьшена, и если после этого выполняется неравенство $q_{i+1} > q$, то должна появиться реакция *R*. Кроме того, во втором и четвертом случаях величина q_i должна претерпеть случайные изменения под влиянием величины *v*.

Рассмотрим последовательность из двух операторов второго типа P_1^1 (ρ_1;) P_1^2 (ρ_2;). Выходные значения этих операторов пока не указаны. Если $\rho_1 = 1$, то первый из этих операторов не оказывает на процесс никакого влияния и происходит проверка выполнимости условия, характерного для второго оператора. Если и $\rho_2 = 1$, то после рассматриваемой пары операторов будет выполняться оператор с номером 3. Если же $\rho_2 =$ $= 0$, то будет выполняться оператор, номер которого указан в качестве выходного в операторе P_1^2 (ρ_2;). Наконец, если $\rho_1 = 0$, то дальнейшее выполнение процедуры будет определяться тем оператором, номер которого указан в качестве выходного в операторе P_1^1 $(\rho_1;\ )$. Последовательность двух операторов второго типа, рассмотренная нами, позволяет разделить процесс на три ветви: $\rho_1 = 0$, ρ_2 — любое; $\rho_1 = 1$, $\rho_2 = 1$; $\rho_1 = 1$, $\rho_2 = 0$.

Введем операторы первого типа: $A_1(q_i;, q_{i+1})$ — прибавление b к текущему значению q_i; $A_2(q_{i+1};\ q_{i+1})$ — прибавление к q_{i+1} значения случайной величины ν; $A_3(q_i;q_{i+1})$ — вычитание из q_i величины d, если $q_i - d \geq$ $\geq q$, в противном случае сохранение равенства $q_i = q$. Введем еще два специальных оператора: A_R и S. Оператор A_R сообщает о том, что реакция R имеет место. Оператор S прекращает выполнение процедуры. С учетом введенных операторов запишем процедуру имитации установления и угасания условного рефлекса следующим образом:

$$P_1^1(\rho_1;7)\,P_1^2(\rho_2;5)\,A_1^3(q_i;q_{i+1})\,A_2^4(q_{i+1};q_{i+1})\,A_R^5S^6P_1^7(\rho_2,;6)$$
$$A_3^8(q_i;q_{i+1})\,A_2^9(q_{i+1};q_{i+1})\,P_2^{10}(q_{i+1};6)\,A_R^{11}S^{12}.$$

Поскольку уже полностью определился порядок следования операторов в записи, то стало возможным указать в операторах второго типа номера операторов, к которым осуществляется переход.

Все операторы кроме $P_2(q_{i+1};\ )$ уже объяснены. Оператор второго типа $P_2(q_{i+1},\ )$ проверяет выполнение условия $q_{i+1}>q$. Если оно выполняется, то реакция R вызывается. В противном случае ее быть не должно.

В качестве третьего примера записи на алгоритмическом языке рассмотрим процедуру приготовления спаса, описанную в разделе о квазиалгоритмах. Введем оператор первого типа $A_1(m,\ l;\ s)$ — отмерить l столовых ложек муки. В качестве входного аргумента выступает мука, а в качестве выходного — заданное ее количество (с помощью входного параметра l). Если исходное количество муки меньше l столовых ложек, то оператор неприменим. Поэтому перед ним надо использовать оператор второго типа $P_1(m,\ l;\ )$, проверяющий соответствующее количественное неравенство. Оператор $A_2(s,\ j;\ c)$ соответствует разведению муки и яйца. В качестве его выходного результата образуется смесь. Оператор $A_3(c;\ c^*)$ есть оператор взбивания смеси (c^* означает сбитую смесь), а оператор второго рода $P_2(c^*;\ )$ определяет качество взбитой смеси. Условие это выполняется, если смесь взбита «тщательно». Заметим, что при алгоритмической записи нас не интересует, насколько конструктивно может быть реализован данный оператор. Оператор $A_4(\nu,\ 1;\ w)$ отмеряет один литр мацуна, выбирая его из наличного количества. Для его применимости необходима проверка, которую предварительно осуществляет оператор $P_3(\nu,\ 1;\ )$. $A_5(c^*,\ w.\ u)$ — оператор соединения смесей путем постепенного введения второй смеси в первую. $A_6(u;\ u^*)$ — оператор кипячения с непрерывным помешиванием (u^* — вскипяченная смесь). $A_7(r,\ ½;\ t)$ — оператор, отмеряющий полстакана риса. $P_4(r,\ ½;\ )$ — оператор, проверяющий наличие полстакана риса в исходном количестве риса. $A_8(t;\ t^*)$ — размытый оператор, отваривающий рис «до мягкости, но так, чтобы рис не разварился» (t^* — отваренный рис). $A_9(p,\ 3;\ q)$ — оператор отбора из лука трех луковиц. $P_5(p,\ 3;\ )$ — оператор проверки возможности выбора трех луковиц. $A_{10}(q;\ q^*)$ — снова размытый оператор, осуществляющий поджаривание лука на сливочном масле «до желтого цвета» (q^* — поджаренный лук). $A_{11}(h,\ ½;\ k)$ и $A_{12}(g,\ 1;\ n)$ — операторы выделения полстакана мяты и стакана кинзы, а операторы $P_6(h,\ ½;\ )$ и $P_7(g,\ 1;\ )$ соответственно — операторы возможности выполнения этого. $A_{13}(u^*,\ q^*,\ k,\ n;\ f)$ — оператор соединения смеси, лука и зелени двух видов.

Теперь мы можем записать процедуру приготовления спаса на алгоритмическом языке:

$$P_1^1(m,5;23)\,A_1^2(m,5;s)A_2^3(s,j;c)\,A_3^4(c;c^*)\,P_2^5(c^*;4)$$
$$P_3^6(\nu,1;23)\,A_4^7(\nu,l;w)A_5^8(c^*,w,u)\,A_6^9(u;u^*)$$
$$P_4^{10}(r,\ ½;\ 23)\ A_7^{11}\ (r,\ ½;\ t)A_8^{12}(t;t^*)P_5^{13}(p,\ 3;23)$$
$$A_9^{14}(p,3;q)\,A_{10}^{15}(q,q^*)\,P_6^{16}(h,½;23)\,A_{11}^{17}(h,½;k)$$
$$P_7^{18}(g,1;23)A_{12}^{19}(g,1;n)\,A_{13}^{20}(u^*,q^*,k,n;f)\,A_6^{21}(f;f^*)\,A_Q^{22}S^{23}.$$

Оператор A_Q^{22} сообщает кулинару, что спас готов. Если каких-нибудь продуктов нет, то происходит прекращение процесса приготовления спаса. Оператор $P_2(c^*;\ 4)$ возвращает нас к выполнению оператора $A_3(c;\ c^*)$ до тех пор, пока смесь не будет взбита надлежащим образом. В этом месте образуется как бы цикл. Подобные же циклы имеют место и при выполнении некоторых других операторов, например оператора $A_{10}(q;\ q^*)$, в процессе реализации которого надо проверять «желтизну» получаемого в данный момент лука (в явном виде эти проверки для простоты записи не выписаны).

Приведенные нами примеры позволяют сделать весьма важный для дальнейшего вывод. Алгоритмические языки могут описывать процессы различной природы: истинные алгоритмы, недетерминированные вероятностные процессы, квазиалгоритмы. Они способны описывать и процедуры, характерные для исчислений. Другими словами, алгоритмические языки обладают универсальностью с точки зрения описания процедур. А если сами операторы, входящие в описания, могут быть конструктивно или каким-либо иным способом реализованы, то могут быть реализованы и все процедуры, описанные в виде последовательностей этих операторов с учетом возможных переходов в этих последовательностях при невыполнении тех или иных условий.

2.2. Реализация процедур

Точная механика. Вспомним снова андроидов, совершавших довольно сложные процедуры. Что помогало реализовать их? «Внутренность» любого андроида первого периода — это скопления огромного количества зубчатых колес, пружин, храповиков и маятников. Недаром первыми конструкторами механических людей были, как правило, часовщики, и в основу реализации процедур, выполнявшихся андроидами, была положена идея часового механизма. Мы уже отмечали недостаток подобного подхода: конструкция жестко определяла реализацию процедуры. Другими словами, память о том, что необходимо делать, реализовывалась в виде механически соединенных элементов, связи между которыми не менялись. Чтобы изменить программу поведения, изменить содержимое памяти устройства, нужно было переделать всю его конструкцию.

А можно ли сделать «сменную» память? Можно ли менять только тот узел конструкции, в котором записана процедура, оставляя все части, ответственные за

ее выполнение, неизмененными? Конечно, можно. И ряд приемов такого рода был найден механиками весьма давно.

Многие еще помуят негромкие звуки шарманки. Шарманка и шарманщики — это целая эпоха. Шарманка — предшественник граммофона, патефона да, пожалуй, и современного проигрывателя. Это нехитрое устройство состояло из набора металлических язычков различной длины и вращающегося барабана, на поверхности которого в определенном порядке были расположены колки. Задевая за язычки, заставляя их колебаться с определенной частотой, колки формировали мелодию. А шарманщик лишь крутил ручку, вращая барабан с колками. В музыкальных шкатулках шарманщика заменяла пружина, которая заводилась от ключа или от самой крышки, когда ее открывали. Память шарманки или музыкальной шкатулки — барабан с колками. Более точно: то или иное расположение колков на поверхности барабана. Если сменить это расположение, то сменится и мелодия.

Позднее вместо барабана появятся медные диски с выступами-колками, потом патефонные пластинки с нанесенными на их поверхности бороздками, затем магнитофонная лента. Но принцип останется: сменная память и неизменное устройство реализации процедуры, записанной тем или иным способом в этой памяти.

Заметим, что у устройств типа шарманки был один недостаток. Медленные вальсы и душещипательные романсы исполнялись ими с той же скоростью, что и веселые польки или марши. Опытные шарманщики устраняли этот недостаток за счет неравномерного вращения ручки инструмента. Они то ускоряли, то замедляли вращение барабана в соответствии со своим эстетическим пониманием исполняемой мелодии. Но музыкальные шкатулки этого делать не умели, ибо в них отсутствовала обратная связь между реализацией заложенной в них процедуры и внешними факторами. Поэтому и андроид-писец усердно писал, когда в чернильнице было пусто, а механическая утка, упав на неровной поверхности, продолжала имитировать движение и крякать.

Однако принцип обратной связи в механических автоматах также был известен весьма давно. В 1588 году итальянский изобретатель Рамелли издал книгу, в которой впервые в истории инженерной науки собрал и описал большое количество различных машин и механизмов. Книга эта называется «Различные замысловатые машины». И среди других «замысловатых машин» в книге описана мельница, функционирование которой поражало современников. Известно, что скорость работы жерновов определяется твердостью зерна. Чем тверже зерно, тем с меньшей скоростью надо вращать жернова и тем меньше должна быть порция зерна, находящаяся между жерновами. В удивительной мельнице, описанной в книге Рамелли, на ось жерновов была насажена граненая муфта, которая при вращении жерновов периодически встряхивала желоб, подающий зерно к жерновам. Если вращение шло быстро, то количество поступающего зерна увеличивалось, если же вращение из-за твердости зерна замедлялось, то замедлялась и подача зерна для помола.

В последующие годы идея использования обратной связи при реализации управляющих процедур стала основной в самых различных областях человеческой деятельности.

Вернемся, однако, к сменной памяти в процессе реализации процедур. Уже известный нам Жак де Вокансон и не менее известный изобретатель Жозеф Мари Жаккар придумали сменную память, идея которой сохранилась до наших дней. С помощью определенным образом расположенных отверстий на сменных картонных картах изобретатели научились управлять ткацкими станками, на которых вырабатывались ткани различного рисунка. Каждая карта Вокансона — Жаккара соответствовала определенному рисунку, образующемуся в процессе работы ткацкого станка. Современные перфоленты — далекие правнуки этих сменных картонных карт.

Таким образом, в области механики был накоплен достаточно богатый арсенал средств для реализации тех или иных заданных процедур. В теории машин и механизмов существуют толстенные справочники, в которых описаны механизмы, реализующие почти все мыслимые механически воспроизводимые процедуры.

Явным недостатком этих механических исполнителей была их узкая специализация, жесткая ориентированность на выполнение программ только определенного типа.

Неистовый МАГ. В XX веке далеко не каждому удается стать открывателем нового направления. Михаилу Александровичу Гаврилову в этом отношении повезло. Он стал родоначальником новой области — теории логического синтеза дискретных устройств. В тридцатых годах нашего века сначала советский ученый В. И. Шестаков, а затем независимо от него известный в последующие годы как «отец теории информации» американец Клод Элвуд Шеннон обнаружили явную аналогию между формулами исчисления высказываний и поведением релейных электрических схем. Однако, обнаружив этот парадоксальный факт, оба исследователя не придали этому открытию должного значения. Их публикации так бы и остались забавным научным кунштюком, если бы не догадка М. А. Гаврилова о том, что открытая ими аналогия куда более глубока, чем это кажется на первый взгляд. М. А. Гаврилов предположил и блестяще обосновал вывод о том, что между исчислениями высказываний и определенными типами электрических схем, состоящих из замыкающих и размыкающих контактов реле и связей между ними, нет принципиальной разницы. Любая формула исчисления высказываний имеет своего представителя в классе параллельно-последовательных релейно-контактных схем и любая схема, замыкание и размыкание цепей в которой происходит с помощью контактов реле, может быть описана в виде некоторой формулы исчисления высказываний.

Необычность этого факта ошеломительна. Кажется невероятным, что обнаруживается столь тесная связь между порождением человеческого духа, восходящим к великому Аристотелю, и вполне земными и привычными для любого инженера схемами, выполненными на реле. Чтобы читатель смог поверить в столь необычное сходство, покажем, в чем состоит суть открытия М. А. Гаврилова.

Возьмем реле, содержащее питающую обмотку и замыкающий контакт. Работа этого реле весьма проста. Если в обмотку подать ток, то контакт замкнется. Соединим теперь один вывод контакта с источником постоянного напряжения, например с батарейкой от карманного фонаря. Другую клемму этой батарейки и другой вывод контакта реле подведем к лампочке карманного фонаря. Будем рассматривать два высказывания: «Обмотка возбуждена» и «Лампочка горит», обозначив их соответственно через x и y. Что. можно сказать об их истинности? Легко видеть, что имеет место равенство $y = x$. Другими словами, высказывания x и y будут истинными или ложными одновременно. Если же заменить замыкающий контакт реле на размыкающий, который при прохождении тока в обмотке разрывает цепь, а при отсутствии его является нормально замкнутым, то соотношение истинности наших высказываний может быть описано соотношением $y = x$ (размыкающий контакт реле имитирует операцию отрицания).

Соединим теперь два реле так, чтобы их замыкающие контакты были бы включены последовательно друг с другом. Через x_1 и x_2 обозначим соответственно высказывания «Обмотка первого реле возбуждена» и «Обмотка второго реле возбуждена». Предположим, что питание обмоток реле можно осуществлять независимо. Истинность высказывания «Лампочка горит» зависит теперь от истинности высказываний x_1 и x_2, Эта зависимость выражается соотношением $y = x_1x_2$. Таким образом, последовательное соединение замыкающих контактов двух реле моделирует операцию конъюнкции. Можно легко убедиться, что параллельное соединение замыкающих контактов двух реле даст схему, для которой $y = x_1, \vee x_2,$ а параллельное соединение размыкающего контакта одного реле и замыкающего контакта другого реле даст имитацию операции импликации, ибо в этом случае $y = x_1 \vee x_2 = x_1 \rightarrow x_2$.

В классических исчислениях вполне достаточно этих четырех операций для описания системы аксиом (как говорилось в разделе об исчислениях, на самом деле этот список операций даже избыточен). Каждая аксиома может быть представлена в виде некоторой релейно-контактной схемы, обладающей тем свойством, что при любом возбуждении или невозбуждении обмоток реле, входящих в эту схему, лампочка никогда не гаснет. Этот физический феномен соответствует тождественной истинности формул, являющихся аксиомами *).

*) Пожалуй, простейшей схемой такого типа является схема с одним реле, имеющим замыкающий и размыкающий контакты, которые в схеме соединены параллельно. При этом $y = x \vee x$.

Правила вывода также могут интерпретироваться как некоторые вполне определенные преобразования тождественно-истинных схем. Множеством всех выводимых схем будут при этом схемы, для которых соединение контактов обеспечивает постоянное включение лампочки.

М. А. Гаврилов был не логиком, а инженером, специалистом по управлению сложными техническими системами. Поэтому мысль его шла не в направлении моделирования исчислений. В открытой аналогии он сумел увидеть аппарат, с помощью которого можно формализовать синтез структур систем управления. Так родилась идея *логического синтеза.* Если условия работы устройства управления можно представить в виде некоторой логической формулы, то новая теория давала возможность проверять полноту и непротиворечивость исходного задания, минимизировать расход элементов при построении устройства и удовлетворять целому ряду дополнительных ограничений, связанных с его техническим воплощением.

Рождение и утверждение нового взгляда на технические дискретные системы управления проходило нелегко. В чем только не обвиняли М. А. Гаврилова: и в идеализме, и в стремлении дегуманизировать логику, и во многих других вещах. Защита его докторской диссертации напоминала сражение, и гамлетовская формула «быть или не быть» новой науке звучала на ней вполне уместно. Страсть и вера ученого победили. И сейчас, когда с тех памятных дней прошло каких-нибудь тридцать лет и на основе теории релейно-контактных схем выросли и структурная теория конечных автоматов, и теория проектирования дискретных систем управления, и ряд других дочерних наук, даже не верится, что начало их пробивалось с таким трудом. С тех пор М. А. Гаврилова его ученики за глаза называли МАГом, а иногда и Неистовым МАГом.

Автоматные процедуры. Пусть имеются базовые элементы, реализующие основные логические операции: отрицание, конъюнкцию и дизъюнкцию. В теории конечных автоматов построены методы синтеза из этих элементов устройств, реализующих процедуры определенного типа, получившие название *автоматных.* В настоящее время они находят широкое применение в ряде подсистем управления роботами.

Конечный автомат — это устройство, работающее в дискретные моменты времени, называемые *рабочими тактами* или просто *тактами.* На вход конечного автомата могут поступать сигналы из конечного, заранее заданного списка, а на выходе его появляются выходные сигналы, также принадлежащие конечному и априорно заданному списку. Значение выходного сигнала в некоторый такт работы определяется входным сигналом, поступившим на автомат в этот такт, и тем внутренним состоянием, в котором автомат находится. Множество внутренних состояний, так же как и множества входных и выходных сигналов, конечно и раз и навсегда для данного автомата задано. Таким образом, конечный автомат в любой такт своей работы осуществляет отображение $(X_t, Y_t) \Rightarrow Z_t$, где X_t, Y_t, и Z_t — соответственно входной сигнал, внутреннее состояние автомата и выходный сигнал в такт работы с номером t. Одновременно с формированием выходного сигнала автомат формирует и свое новое состояние, т. е. реализует отображение $(X_t, Y_t) \Rightarrow Y_{t+1}$. Эта пара отображений может быть задана в виде таблицы, строки которой соответствуют всем допустимым, сигналам на входе автомата, а столбцы — его внутренним состояниям. На пересечении стоят пары Z_t и Y_{t+1} определяемые соответствующим входным сигналом и состоянием автомата. Такая таблица называется *автоматной.*

Рассмотрим следующий пример. Робот-упаковщик должен укладывать в подарочные коробки три предмета: флакон духов, флакон одеколона и коробочку пудры. Робот обладает «зрением» и снабжен тремя манипуляторами. Один манипулятор имеет захват для взятия стеклянных флаконов, второй — захват для взятия картонных коробочек пудры (о третьем манипуляторе будет сказано ниже). Элементы, составляющие набор, поступают на рабочий стол робота неравномерно и без определенного порядка. Цикл работы робота-упаковщика таков, что, начав собирать один из наборов, он должен полностью завершить сборку прежде, чем перейдет к укладке другого набора. Порядок заполнения подарочной коробки безразличен. Необходимо создать систему управления действиями такого робота-упаковщика.

Входные сигналы удобно кодировать трехэлементными двоичными наборами. Единица на соответствующем месте означает наличие перед роботом на столе определенного предмета: духов, одеколона или пудры. Нуль на соответствующем месте означает отсутствие предметов данной группы. Например, набор 011 означает, что на рабочем столе робота есть флаконы с одеколоном и коробочки пудры, но нет флаконов с духами. Информация о виде текущего набора поступает роботу от его «глаза» (например, от телевизионной передающей камеры, дополненной устройством анализа зрительной сцены и выделения в ней интересующих робота предметов — флаконов и коробочек пудры).

Выходные сигналы бывают пяти типов:

$$Z^1,\ Z^2,\ Z^3,\ Z^4 \text{ и } Z^5$$

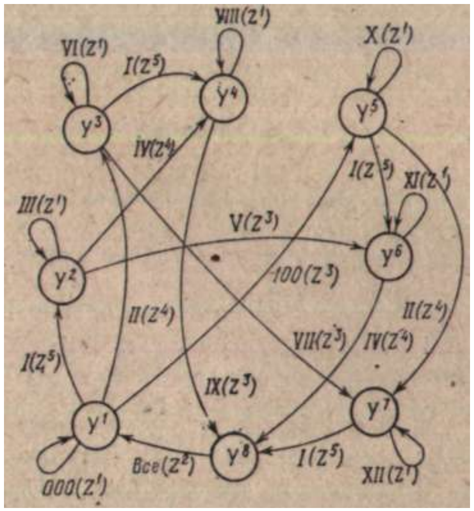
Сигнал Z^1 означает, что надо ждать; сигнал Z^2 — закрыть заполненную коробку, сдвинуть ее на конвейер готовой продукции, взять новую коробку для набора; Z^3 — взять флакон духов и положить в коробку; Z^4 — взять флакон одеколона и положить в коробку; Z^5 — взять коробочку пудры и положить в коробку. При этом сигнал Z^1 «никуда не поступает», т. е. никакие исполнительные механизмы робота не инициируются этим сигналом. Сигнал Z^2 поступает на третий манипулятор, который по жесткой программе осуществляет предписываемые этим сигналом действия. Сигналы Z^3 и Z^4 выдаются на манипулятор, снабженный захватом для флаконов, а сигнал Z^5 — на манипулятор, приспособленный для взятия коробочек пудры.

Составим теперь автоматную таблицу (табл. 3).

Таблица 3

Робот видит			В коробке есть (состояние робота):							
духи	одеколон	пудру	начальное $\overline{Y^1}$	пудра Y^2	одеколон Y^3	пудра и одеколон Y^4	духи Y^5	духи и пудра Y^6	духи и одеколон Y^7	пудра, одеколон и духи Y^8
0	0	0	1,1	2,1	3,1	4,1	5,1	6,1	7,1	1,2.
0	0	1	2,5	2,1	4,5	4,1	6,5	6,1	8,5	1,2
0	1	0	3,4	4,4	3,1	4,1	7,4	8,4	7,1	1,2
0	1	1	2,5	4,4	4,5	4,1	6,5	8,4	8,5	1,2
1	0	0	5,3	6,3	7,3	8,3	5,1	6,1	7,1	1,2
1	0	1	2,5	6,3	4,5	8,3	6,5	6,1	8,5	1,2
1	1	0	3,4	4,4	7,3	8,3	7,4	8,4	7,1	1,2
1	1	1	2,5	4,4	4,5	8,3	6,5	8,4	8,5	1,2

В этой таблице для простоты на пересечении строк и столбцов поставлены только числовые индексы новых внутренних состояний автомата и его выходных сигналов. Цикл работы робота начинается в состоянии Y^1 и заканчивается переходом из состояния Y^8 в состояние Y^1. В этот момент предшествующая коробка наполнена, приготовлена новая коробка и робот готов ее заполнять.



I	II	III	IV	V	VI	VI	VIII
001	010	000	010	100	000	100	000
011	110	001	011	101	010	110	001
101			110				010
111			111				011

IX	X	XI	XII
100	000	000	000
101	100	001	010
110		100	100
111		101	110

Рис. 2

Часто вместо представления в виде автоматной таблицы используют эквивалентное, но более наглядное представление автомата в виде *автоматного графа.* Для нашего примера такой граф изображен на рис. 2. Кружки — вершины графа соответствуют внутренним состояниям автомата, стрелки определяют смену этих

состояний под влиянием входных сигналов из множеств надписанных около стрелок и обозначенных римскими цифрами. Формирующийся при этом выходной сигнал написан рядом в скобках. Состав входных множеств указан в таблице.

Легко усмотреть определенную аналогию между автоматной таблицей и введенной нами ранее таблицей управления для машины Тьюринга. Эта аналогия имеет глубокий смысл. Конечный автомат есть специальный класс машин Тьюринга, для которых множество допустимых на ленте слов как бы фиксировано и работа с ними происходит по законам, более специализированным, чем в машине Тьюринга в общем случае. Это позволяет утверждать, что автоматные процедуры, реализуемые конечными автоматами, представляют собой подмножество алгоритмических процедур.

Это суждение не столь уж неоправданно. Можно отметить, что подавляющее большинство существующих в настоящее время систем управления, техническими объектами, если управление осуществляется дискретно, формально может быть описано на языке конечных автоматов.

Модель конечного автомата может быть обобщена на вероятностный случай (так называемые *вероятностные автоматы*), и тогда можно описать подкласс вероятностных алгоритмов и устройств, их реализующих. В частности, можно описать функционирование различных технических устройств, используемых при порождении распределений случайных величин необходимого вида. В настоящее время делаются попытки создать модель квазиавтомата или размытого конечного автомата, пригодного для описания соответствующего подмножества квазиалгоритмических процедур.

Для нас это не столь уж важно. Для нас интересно другое. Какой смысл можно придать работе конечного автомата с точки зрения поведения живых существ?

Будем рассматривать сигналы X_t, как стимулы, а сигналы Z_t, как реакции конечного автомата на X_t. Тогда, если бы автомат имел только одно фиксированное внутреннее состояние, то на любой стимул X^i он всегда бы давал однозначный ответ-реакцию Z^j. При наличии множества внутренних состояний автомат способен разнообразить свое поведение. В частности, он способен «помнить» некоторую конечную предысторию и в зависимости от этой памяти вести себя при одном и том же стимуле различным образом. Если, например, наш робот-упаковщик видит духи, но они уже есть в коробке, то он не берет их со стола.

Для вероятностных автоматов можно организовать вероятностный выбор реакции на стимул, задавая необходимым образом закон распределения выходных сигналов на заданные входные сигналы. В квазиавтоматах эти реакции могут иметь произвольный недетерминированный характер. Таким образом, семейство конечных автоматов различного типа способно имитировать стимульно-реактивную модель поведения, например реакцию амебы на электрическое раздражение.

С помощью вероятностного автомата можно, в частности, реализовать процессы возникновения и угасания условного рефлекса, которые мы описали в разделе об алгоритмических языках.

Но как конечные, так и вероятностные автоматы не могут реализовать алгоритмы произвольной природы. Им «подвластно» лишь некоторое подмножество алгоритмических процедур. Для конструкции универсального алгоритмического устройства надо выйти за ограничения автоматов.

Универсальные вычислители. Идея о конструктивном воплощении машины Тьюринга витала в воздухе. В нескольких странах исследовательские группы занимались этой проблемой. Приближалась пора электронных вычислительных машин, появление которых обычно приравнивается к появлению первых паровых машин и двигателей внутреннего сгорания. В Пенсильванском университете в США такую исследовательскую группу возглавляли талантливые инженеры Моучли и Эккерт. Эта группа значительно опередила остальных исследователей. В 1946 году первая в мире электронная вычислительная машина, названная ее создателями ENIAC, начала производить вычисления. Эта первая машина во многом отличалась от привычных нам современных ЭВМ. В ней имелся ряд специализированных устройств, предназначенных для деления, извлечения квадратного корня, умножения и т. п. Эти устройства могли работать независимо и параллельно. Общее управление ими осуществлялось центральным программным блоком. Но самой программы в теперешнем смысле еще не было. Программирование осуществлялось путем коммутации отдельных блоков машины с помощью специальных кабелей, штекеров и механических переключателей. Числа в этой первой машине представлялись не двоичными кодами, а в привычной человеку десятичной системе счисления*). .

Успех группы Пенсильванского университета побудил других исследователей увеличить темпы. Этому способствовали также недостатки машины ENIAC, выявившие основные проблемы, которые необходимо было решить. В той же Пенсильванской группе уже начал разрабатываться проект новой ЭВМ, названной EDVAC, в которой просматривались базовые принципы, положенные в основу современных ЭВМ: разделение машины на арифметическое устройство, способное выполнять весь набор необходимых операций, оперативное запоминающее устройство, способное хранить большой массив данных и производить запись и считывание их за короткое время, централизованное устройство управления, осуществляющее взаимодействие арифметического и запоминающего устройств при реализации программы, и, наконец, входное и выходное устройства, предназначенные для общения машины с пользователем. Уже было ясно, что двоичное представление чисел сулит большие перспективы и что программа работы должна задаваться не коммутацией блоков человеком, а записываться в память машины, считываться оттуда и управлять работой остальных блоков машины в темпе этого считывания.

Именно поэтому первой «настоящей» ЭВМ следует считать не машину ENIAC и не устройства, подобные ей, а введенную в эксплуатацию в 1949 году в Великобритании ЭВМ, разработанную М. Уилксом при участии А. Тьюринга, получившего возможность вопло-

*) Вопрос о приоритете той или иной исследовательской группы в создании ЭВМ до сих пор является спорным. В 1973 г. в США происходил судебный процесс, целью которого являлось решение вопроса о приоритете Пенсильванской группы в этих исследованиях. Дело, занимавшее 1250 страниц, позволило суду придти к выводу, что еще в 1939—1941 гг. в колледже штата Айова под руководством Атанасова была разработана и построена в виде лабораторного макета ЭВМ, которая во многом превосходила ENIAC. А в Германии в 1937—1942 гг. инженер К. Цузе также разработал проект ЭВМ. не реализованный из-за отсутствия ассигнований.

тить идеи своей абстрактной машины на техническом уровне.

Создание первой советской ЭВМ связано с именем Сергея Алексеевича Лебедева. Небольшому коллективу ее создателей в Киеве понадобилось всего два года, чтобы от «нуля» придти к работающей ЭВМ. В 1950 году машина МЭСМ начала работать. А еще через два года под руководством С. А. Лебедева, но уже в Москве, была создана машина БЭСМ-1, прародительница целого поколения БЭСМ.

К концу пятидесятых годов было построено много ЭВМ большой мощности, накопился опыт работы с ними, найдены основные конструктивные решения и приемы программирования.

Но самым важным является то, что при любом конструктивном решении, при любом способе организации работы с машиной ее прототипом всегда остается машина Тьюринга. Вспомним еще раз его универсальную машину, способную выполнять работу любой конкретной машины, если на ленту будет записана таблица управления этой конкретной машиной. Современная ЭВМ действует полностью аналогично. Роль кодирования таблицы управления играет программирование. Любой процесс, который может быть запрограммирован в операторах, понятных устройству управления ЭВМ, может быть введен в память ЭВМ и реализован ею. Система этих операторов в ЭВМ обладает свойством полноты в том смысле, что она достаточна для реализации в машине любого алгоритма в точном смысле этого слова. Память ЭВМ обладает достаточным объемом для записи реально интересных программ и хранения необходимых данных. Именно поэтому ЭВМ представляет собой воплощение идеи универсального вычислителя.

На самом деле возможности ЭВМ несколько шире. Мы рассматривали ранее не только алгоритмы, но и вероятностные алгоритмы и квазиалгоритмы. В современных ЭВМ имеются средства для реализации процесса получения значений дискретных случайных величин. А это значит, что ЭВМ могут реализовать и вероятностные алгоритмы. Возможности такого типа используются в настоящее время весьма интенсивно. Далее мы вернемся к этой мысли и проиллюстрируем наше утверждение. Если ЭВМ будет оснащена средствами для имитации операторов, встречающихся в квазиалгоритмах, то ЭВМ сможет в силу своей конструкции реализовать и такие процедуры.

Очень важно обратить внимание на одно обстоятельство, связанное с работой ЭВМ. Давайте подойдем к машине, в которую введена программа, и будем наблюдать, что она делает. Опыт этот, конечно, гипотетический. Скорости работы современных ЭВМ столь велики, что пошаговый анализ ее действий возможен только тогда, когда ЭВМ просто напечатает на выходной ленте все, что она делала, а мы будем анализировать уже эту запись. Итак, мы наблюдаем за работой машины. На каждом шаге она оперирует кодовыми словами, хранящимися в ее памяти. Происходит переписывание кодовых слов, стирание их, производятся различные арифметические и логические операции над ними, выделяются отдельные разряды и т. д. Но что же делает ЭВМ? Каков смысл решаемой задачи? Вычисляет ли она корни квадратного уравнения или сочиняет музыкальное произведение? А может быть она решает дифференциальное уравнение? Или играет в шашки?

Парадоксально, но ответить на эти вопросы нельзя. ЭВМ в определенном смысле все делает одинаково. Она всегда что-то складывает, умножает, переписывает. Но внешний смысл этих действий скрыт от ЭВМ. Что она делает, знает только человек, который написал программу и ввел ее в память ЭВМ. Поэтому, когда говорят, что ЭВМ сочиняет музыку или стихи, играет в шашки или шахматы, решает дифференциальное уравнение или расшифровывает неизвестный язык, всегда надо помнить, что это не более, чем метафора. Все это делает не ЭВМ, а ЭВМ вместе с человеком, написавшим программу решения соответствующей задачи.

2.3. Моделирование и имитация

Адаптация. Когда мы говорим о живых организмах, то в нашем представлении они наделяются рядом свойств, присущих им всем. Одно из этих свойств — умение адаптироваться к изменяющимся окружающим условиям. среда, в которой обитают живые организмы.

не является абсолютно неизменной. Какие-то ее параметры, важные с точки зрения выживаемости организма, могут менять свои значения и организм должен как-то приспосабливать свое функционирование к этим изменениям. Возникает вопрос: может ли техническое устройство имитировать с достаточной полнотой это свойство живых организмов?

Впервые точная постановка этой проблемы и ее положительное решение применительно к автоматам были реализованы в работах выдающегося советского исследователя Михаила Львовича Цетлина. Подобно ряду других ученых, сыгравших решающую роль в развитии кибернетики, М. Л. Цетлин был разносторонним специалистом: прекрасный инженер и физик-экспериментатор, математик и конструктор в области медицинского приборостроения. М. Л. Цетлин профессионально разбирался в топологии и теории управления, теории конечных автоматов и абстрактной алгебре, в радиофизике и технологии изготовления электронных приборов. Но столь же профессионально он мог говорить о физиологии и литературе, тонкостях немецкого языка и юриспруденции. Личные душевные качества М. Л. Цетлина — умение сопереживать другим людям и неудержимое желание им помочь, полное бескорыстие в научных исследованиях и в обычной жизни, стремление приобщить к интересующим его проблемам как можно большее число специалистов, способность показать и доказать важность и нужность тех или иных исследований — сделали его признанным главой (правда, в отношении М. Л. Цетлина это слово лучше было бы заменить словом «душой») большой школы, объединяющей математиков, биологов, медиков и инженеров. На ежегодных симпозиумах в этом коллективе рождались необычные постановки задач, стоящих на стыке столь различных на первый взгляд наук. И одной из этих задач стала задача о «поведении маленького животного в большом мире» *).

Рассмотрим сначала эту задачу в некоторой частной постановке. Пусть животное через свои рецепторы может получать от окружающей его среды сигналы двух типов: поощрения и наказания. Эти сигналы связаны с действиями животного, реализуемыми в данной среде. Связь между действиями и сигналами от среды носит вероятностный характер. Если животное совершает некоторое действие d_i, то с вероятностью p_i оно получает сигнал наказания, а с вероятностью $(1 - p_i)$ — сигнал поощрения. Вся беда состоит в том, что животное не знает теории вероятностей и значения p_i для него абсолютно неизвестны. Возникает задача: сможет ли животное адаптироваться к такой стационарной вероятностной среде? Может ли оно как-то минимизировать свои неудачи, уменьшить число сигналов наказания?

*) - Эта формулировка принадлежит М. Л. Цетлину, всегда предпочитавшему наглядность наукообразию,

В начале нашего века зоопсихолог Эдуард Торндайк провел следующий эксперимент с животными. Имелся Т-образный лабиринт с тремя площадками. На площадку, находившуюся в основании буквы Т, помещалось подопытное животное, а на две другие площадки, находившиеся у концов горизонтальной перекладины буквы Т, помещалась приманка. Животное могло делать альтернативный выбор: добежав до развилки, оно могло повернуть к левой площадке (действие d_1 или к правой площадке (действие d_2). Но по пути к приманке его ожидала неприятность. В стенки коридора были вмонтированы электроды. С некоторой фиксированной вероятностью на них подавалось напряжение и тогда пробегавшее мимо них животное получало болевое раздражение — среда выдавала сигнал наказания. Сигналом же поощрения среды была та пища, которая ожидала животное на конечной площадке. Если в эксперименте вероятность раздражения в одном из коридоров (например, в левом) намного превосходила вероятность такого раздражения в другом коридоре (в правом), то естественно было бы считать, что животное адаптируется к условиям среды, если после серии пробежек оно будет предпочитать поворачивать в правый коридор, а не в левый.

Больше всего Торндайк экспериментировал с крысами. Оказалось, что крысы очень быстро оценивают более безопасный путь и уверенно выбирают его даже при не слишком большой разнице вероятностей наказаний. Другие подопытные животные делали это с разной степенью адаптируемости, но способность эта оказалась присущей всем видам животных, участвовавших в опытах, Проблема, поставленная М. Л. Цетлиным, заключалась в поиске такого технического устройства, которое могло бы имитировать процесс подобной адаптации. И самым удивительным оказалось не то, что такое устройство было найдено, а то, что технически оно явилось весьма простым вероятностным автоматом. Поскольку эта конструкция будет использоваться нами в дальнейших разделах, рассмотрим ее подробнее.

Итак, пусть у нас имеется автомат, способный воспринимать два входных сигнала: $x^1 = 1$ и $x^2 = 0$. Первый сигнал мы будем называть *наказанием,* а второй — *поощрением.* Автомат может иметь на выходе n выходных сигналов $z^1, z^2, \ldots, z^n$, которые мы будем называть *действиями.* Среда задается вектором, имеющим n компонентов: $C = (p_1, p_2, \ldots, p_n)$. При этом p_i есть вероятность того, что за действие z^i автомат получит от среды сигнал наказания (за это же действие автомат может получить сигнал поощрения с вероятностью $1 - p_i$). Еще раз подчеркнем, что хотя значения p_i объективно существуют, автомату они априорно неизвестны. Если бы это было не так, то задача адаптации решалась бы тривиально. Если бы, например, крыса «знала», что среда, в которой ей приходится бегать за пищей, определяется как $C = (0{,}9;\ 0{,}001)$, то она всегда сворачивала бы только направо, ибо в левом коридоре ей бы пришлось испытывать болевое раздражение «почти всегда».

Рассмотрим сначала следующий способ выбора очередных действий автомата. На каждом такте своей работы автомат равновероятно выбирает одно из своих n действий. Если бы их было два, как у животного в Т-образном лабиринте, то перед поворотом он мог бы «бросать монетку» и в зависимости от ее выпадения (герб или решетка) выбирать левый или правый коридор. Ясно, что такая стратегия не учитывает предыстории (обратной связи от среды). Здесь не происходит накопления опыта и не может быть адаптации. Если эксперимент длится весьма долго (теоретически бесконечно долго), то из теории вероятностей следует, что автомат может накопить за это время достаточно хорошие оценки неизвестных ему значений вероятностей наказания и поощрения. Для этого он каким-то образом должен подсчитывать относительные частоты наказаний за действия. С увеличением отрезка времени наблюдения эти частоты будут все лучше и лучше аппроксимировать значения соответствующих вероятностей.

Пусть $\max p_i$ и $\min p_i$ — соответственно наибольшее и наименьшее значения p_i в описании среды. Например, для среды $C = (0{,}7;\ 0{,}6;\ 0{,}003;\ 0{,}75)$ эти значения равны соответственно 0,75 и 0,003. Ясно, что как бы автомат ни выбирал свои действия, он не может добиться меньшего числа наказаний по сравнению с постоянным выбором только третьего действия и не может накопить этих наказаний больше по сравнению со случаем постоянного выбора только четвертого действия. Можно сказать, что автомат действует оптимальным образом, если после некоторого периода адаптации он станет выбирать лишь третье действие. И его поведение будет все еще целесообразным, если стратегия выбора действий будет обеспечивать меньший суммарный проигрыш, чем при случайном равновероятном выборе действий. Если же стратегия автомата такова, что этот суммарный проигрыш будет больше, чем при случайном равновероятном выборе (это происходит, например, когда автомат в данной конкретной среде всегда выбирает только четвертое или только первое действие либо, например, равновероятно чередует выбор только этой пары действий), то поведение автомата следует признать нецелесообразным.

Покажем теперь, что существуют конструкции автоматов, которые обеспечивают целесообразное поведение в стационарной среде. Зафиксируем число действий автомата $n = 4$. Это ограничение сделано исключительно для наглядности нашего изложения. Как будет видно далее, по сути конкретное значение n нигде не используется. Возьмем конечный автомат, у которого число внутренних состояний есть $4m$, где m — некоторый параметр. Вместо автоматной таблицы воспользуемся заданием работы автомата в виде автоматного графа (рис. 3). Для определенности на этом рисунке изображен автомат с $m = 4$. Входному сигналу $x^1 = 1$ (наказание) соответствуют переходы, показанные штриховыми стрелками, а входному сигналу $x^2 = 0$ (поощрение) — сплошные стрелки. Выходные сигналы автомата определяются только тем состоянием, в котором он находится в данном такте. Состояния разбиты на четыре группы по четыре состояния в каждой группе (при произвольных n и m будет n групп состояний по m состояний в каждой). Каждой группе состояний соответствует определенное действие. Начальное состояние автомата может быть любым. Предположим, что в начальный момент автомат находится в состоянии, которое на рисунке заштриховано, а среда, с которой взаимодействует автомат, это та среда, которую мы только что рассматривали в качестве конкретного примера.

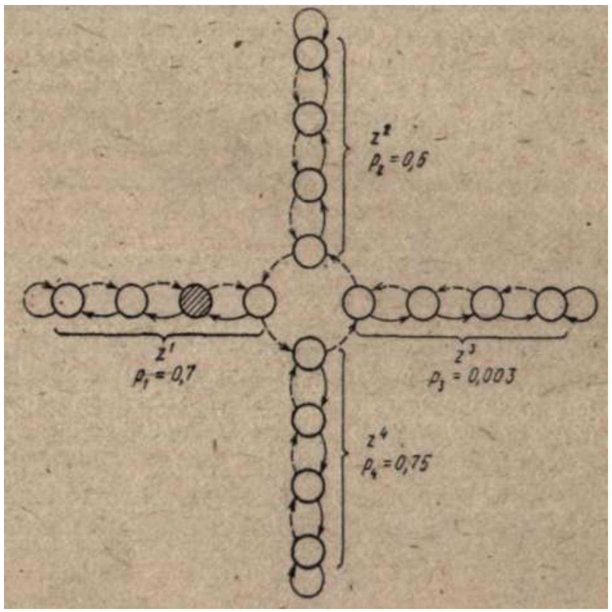


Рис. 3.

Проследим функционирование этого автомата. Заштрихованное состояние определяет выходной сигнал z^1. На следующем такте работы автомат с вероятностью 0,7 получит наказание и перейдет в состояние, расположенное справа от заштрихованного. С вероятностью 0,3 он получит сигнал поощрения и перейдет в состояние, расположенное слева от заштрихованного. При этом он остается в той группе состояний, которым соответствует выдача в среду сигнала z^1. Поэтому на следующем такте своего взаимодействия со средой автомат получит наказание с вероятностью 0,7 и поощре-

нив с вероятностью 0,3. Поскольку сигналы эти среда формирует независимо от предыстории, то вероятность двукратного получения сигнала наказания равна $0,7 \cdot 0,7 = 0,49$, вероятность получения двух сигналов поощрения подряд равна $0,3 \cdot 0,3 = 0,09$, а вероятность получения двух разнозначных сигналов равна $1 - (0,49 + 0,09) = 0,42$. Значит, после двух тактов работы автомат с вероятностью 0,49 выйдет из группы состояний, в которой он совершает действие z^1, с вероятностью 0,42 останется в начальном состоянии и с вероятностью 0,09 переместится вглубь группы состояний, характеризующихся действием z^1.

Эти рассуждения можно продолжить. Но уже из сказанного становится ясным, что автомат имеет немного шансов на то, чтобы сохранять свои состояния, принадлежащие к группе, где реализуется действие z^1. Попав в соседнюю группу, он также со временем покинет ее, ибо вероятность показания за действия z^2 все-таки больше вероятности поощрения. Но, попав в третью группу, автомат имеет большие шансы остаться в ней навсегда. Особенно, если ему «повезет» и он не получит сразу же, как только перейдет в первое состояние этой группы, сигнал наказания, последствием чего будет переход автомата в начальное состояние четвертой группы состояний.

Для тех читателей, которые знакомы с теорией вероятностей и дискретными цепями Маркова, наши качественные рассуждения могут быть подкреплены вычислениями значений финальных вероятностей пребывания автомата в той или иной группе состояний.

Теперь становится понятным значение параметра m.. Он характеризует глубину памяти автомата, его способность к инерции сохранения действия при получении сигналов наказания. Если m мало, то случайные маловероятные последовательности сигналов наказания на входе автомата могут вывести его даже из самых хороших групп состояний, обеспечивающих оптимальное функционирование в данной среде. В предельном случае при $m = 1$ поведение автомата может быть нецелесообразным.

Вспомним знаменитую притчу об Иванушке-дурачке. Встретив похороны, он стал радостно смеяться и приплясывать. И получил заслуженное наказание. Его здорово поколотили. Тогда он сменил действие. Встретив свадебный поезд, он стал рыдать и оплакивать новобрачных и их горькую судьбу. И опять был бит. В описанной ситуации Иванушка-дурачок имеет выбор из двух действий: выказывать радость и выказывать печаль. Предположим, что в той сказочной стране, где он живет, число свадеб примерно в десять раз превышает число похороп. Тогда за действие z^1 наказание будет формироваться с вероятностью 0,1, а за второе действие — с вероятностью 0,9. Цифры эти, конечно, Иванушке-дурачку априорно неизвестны. Если он действует как автомат, описанный нами, то чередование радости и печали при $m = 1$ будет происходить довольно часто. И довольно частое битье неизбежно. Но при увеличении значения m Иванушка-дурачок приобретает некоторую инертность. А оптимальное его поведение будет обеспечивать стратегия, при которой любая процессия будет встречаться им с радостью.

Класс автоматов, который мы описали, был назван М. Л. Цетлиным *автоматами с линейной тактикой.* Для этих автоматов был получен следующий фундаментальный результат: если стационарная среда такова, что имеется хотя бы одно действие автомата с линейной тактикой, за которое вероятность поощрения больше вероятности наказания, то при неограниченном увеличении параметра m можно всегда добиться оптимального поведения автомата. Если указанное ограничение не выполняется, то автомат будет функционировать не оптимально, но целесообразно, т. е. лучше механизма с равновероятным выбором действий.

Можно несколько изменить конструкцию автомата с линейной тактикой для того, чтобы снять ограничение на значения вероятностей штрафа за действия. Если оставить неизменным поведение автомата при получении сигналов наказания, а при получении сигнала поощрения переводить автомат из любого состояния данной группы в ее самое глубокое состояние, то мы получим класс автоматов, названных специалистами *доверчивыми.* Такие автоматы обеспечивают оптимальную адаптацию в любой заранее неизвестной стационарной вероятностной среде.

Этот результат удивил многих биологов. Оказалось, что такое сложное свойство, как приспособление к неизвестному стационарному миру, можно смоделировать чрезвычайно простыми техническими средствами. Ибо

технически автомат с линейной тактикой — это простой набор из сдвиговых регистров и несложной схемы их взаимодействия, а доверчивый автомат ненамного сложнее.

Динамическая адаптация. Правда, в руках скептиков был весомый козырь. Реальный мир, в котором живут организмы, вовсе не описывается в виде стационарной среды. Это мир динамический, постоянно меняющий свои свойства. То, что полезно для животного летом, зимой может оказаться не только бесполезным, но и вредным. То, чему я сегодня научился в результате адаптации, завтра может оказаться смертельно опасным и потребуется срочное переучивание. А что будет происходить с «маленькими животными в большом мире», если этот мир станет изменчивым, динамическим? Сохранят ли они в нем хотя бы целесообразность своего поведения?

Введем понятие *переключающейся динамической среды.* Рассмотрим l стационарных сред и зададим l^2 вероятностей вида q_{ij}, где i и j принимают все значения от 1 до l. Если в некоторый такт работы автомат функционирует в стационарной среде C_i, то в следующем такте работы эта среда сохранится лишь с вероятностью q_{ii}. С вероятностью q_{i1} она сменится средой C_1, с вероятностью q_{i2} — средой C_2 и т. д. Естественным требованием при этом является выполнение равенства $q_{i1} + q_{i2} + \ldots + q_{il} = 1$ для любого значений i.

Если в такую динамическую среду поместить автомат с линейной тактикой, то целесообразность его поведения будет определяться соотношением величин q_{ij} между собой и величиной m. Качественно ясно, что слишком малое значение m при медленно меняющихся средах (т. е. при тех средах, для которых вероятности q_{ii} сохранения среды неизменной достаточно велики) будет приводить к плохому функционированию автомата в стационарной случайной среде. Однако слишком большая инерция перестройки автомата, определяемая большим значением m, при часто меняющихся средах также приводит к плохому функционированию автомата, который не успевает приспособиться к среде до ее замены на другую.

В работах учеников М. Л. Цетлина было показано, что для фиксированных переключающихся сред существуют оптимальные значения m, которые обеспечивают наилучшее возможное функционирование автоматов этого класса. Так, известным советским кибернетиком В. И. Варшавским была предложена конструкция специального вероятностного автомата с переменной структурой, который при определенных условиях обеспечивает целесообразное поведение в любой переключающейся среде, а в ряде частных случаев ведет себя оптимально. Конструкция таких автоматов с переменной структурой оказывается столь же несложной, что и автоматов с линейной тактикой.

Считая, что любая динамическая среда может быть с достаточной степенью точности аппроксимирована переключающейся средой, можно считать, что в классе конечных и вероятностных автоматов найдены конструкции, позволяющие положительно ответить на вопрос о возможности воспроизведения в технических устройствах процесса адаптации в среде, для которой априорно не известны закономерности ее реакций на действие автомата. Конечно, описание явления адаптации в живых организмах значительно сложнее той модели, на примере которой мы продемонстрировали это явление. Но у специалистов имеется глубокая уверенность, что в автоматах, предложенных М. Л. Цетлиным и его последователями, найдено принципиальное решение, которое может быть положено в основу построения систем адаптации в перспективных интеллектуальных системах, которые будут функционировать в средах с априорно неполностью известными свойствами.

«Кормушки» и «Общая касса». Следующим шагом в исследованиях М. Л. Цетлина и его учеников был переход к моделированию целесообразного поведения коллектива автоматов. При этом на модели проверялась верность глобальной гипотезы, выдвинутой М. Л. Цетлиным совместно с известным математиком И. М Гельфандом. Гипотеза эта получила название "Принципа наименьшего взаимодействия". Суть ее состояла в том, что внешне сложные формы поведения живых систем определяются совокупностью большого числа элементов, каждый из которых осуществляет лишь простейшие процедуры, и взаимодействие между элементами происходит при минимально возможных потоках информации. В биологических системах мы часто наблюдаем этот суммарный эффект от совокупности простейших действий. Например, несмотря на большое количество клеток синусного узла сердца, работающих с различной частотой, происходит синхронизация работы узла за счет клетки, функционирующей с наибольшей частотой. Этот факт, теоретически установленный И. М. Гельфандом, С. А. Ковалевым и Л. М. Чайлахяном, был одним из стимулов формирования «Принципа наименьшего взаимодействия». Аналогичные процессы характеризуют работу пула мотонейронов, управляющих движением мышц, распространение возбуждений в нервных тканях, объясняют явление залповой активности дыхательного центра и многие другие физиологические процессы.

Этот же принцип, по мнению М. Л. Цетлина, мог бы быть положен в основу анализа сложных поведенческих актов, характерных для живых организмов. Особенно ясно это наблюдается в таком «организме», как муравейник или пчелиный рой. Отдельные особи в них могут выполнять жесткие программы действий, не слишком часто взаимодействуя между собой. Но в результате этих действий целостный «организм» может функционировать весьма гибко и решать такие задачи, которые были бы не под силу обставляющим его особям.

Именно поэтому исследование целесообразности поведения коллектива, составленного из довольно простых автоматов, обменивающихся между собой минимумом информации, оказывается весьма интересным. Ибо создание интеллектуальных систем по типу муравейника или пчелиного роя может привести к неожиданным результатам.

В фантастической повести Станислава Лема «Непобедимый» описана такая ситуация. Миллионы простейших элементов кристаллической природы, обладающие лишь тривиальным поведением, могут при необходимости соединяться в сверхорганизм (в повести он назван «тучей»). Объединенная память тучи хранит огромный объем знаний и навыков по борьбе с пришельцами из других миров. И туча оказывается непобедимой. После решения стоящей перед сверхорганизмом проблемы он снова «рассыпается» на простейшие элементы, ведущие свою незатейливую жизнь.

Было исследовано много моделей коллективного поведения, опирающихся на «Принцип наименьшего взаимодействия». Остановимся на одной из них.

Эта модель получила название «Кормушки» по той первичной интерпретации, которую предложили ее создатели. Пусть имеется n кормушек, где могут кормиться N простейших организмов, в качестве которых в модели могут быть использованы какие-либо автоматы, оптимально ведущие себя в стационарных средах (например, доверчивый автомат или автомат с линейной тактикой с учетом необходимого ограничения на среду). Каждая кормушка с номером i имеет запас пищи a_i. Действия автоматов сводятся к выбору той или иной кормушки. Элементарный акт поведения коллектива автоматов состоит в том, что в определенный момент все автоматы делают какое-либо действие, т. е. выбирают себе кормушку. После этого для каждой кормушки происходит подсчет числа автоматов, выбравших ее. Если для кормушки с номером i это число равно m, то каждый из автоматов, выбравший эту кормушку, получает возможность "съесть"a_1:m пищи. Локальная цель автомата - "съесть" как можно больше пищи. А глобальная цель всего коллектива - "съесть" максимальное суммарное количество пищи.

Такая постановка задачи может быть интерпретирована, например, как задача пастуха, распределяющего стадо из N овец на n пастбищ, каждое из которых имеет различный запас кормов, характеризуемый величинами a_i. "Цели" овец и цель пастуха в этой задаче находятся в том же соотношении, что и в модели с кормушками. После того как автомат получал свой выигрыш, среда производила сравнение выигрыша данного автомата со средним показателем по коллективу и выдавала сигнал наказания, если личный выигрыш данного автомата был меньше этого среднего выигрыша. В противном случае на вход автомата подавался сигнал поощрения. Идея такого формирования входного сигнала исходила из того, что "в коллективе надо подтягивать отстающих". По этому сигналу происходило то или иное изменение состояния автомата, предопределяющее его очередное действие - выбор той же кормушки или смену ее в соответствии с переходом автомата в другую группу внутренних состояний из-за сигнала наказания. Для наглядности рассмотрим конкретную ситуацию. Пусть имеются четыре кормушки с уровнями пищи

100, 70, 40 и 15 и имеются пять автоматов, которые должны быть распределены по кормушкам. Если бы уровни пищи в кормушках автоматам были бы заранее известны, то с точки зрения их личных целей наилучшим распределением было бы следующее: два автомата в первой кормушке (каждый получает по 50 единиц пищи), два автомата во второй кормушке (каждый получает по 35 единиц пищи) и один автомат в третьей кормушке; четвертая кормушка — пуста. Однако с точки зрения глобальной цели коллектива более продуктивным было бы любое распределение автоматов, при котором все кормушки оказались бы занятыми.

Как же ведут себя автоматы с линейной тактикой и другие типы автоматов, обладающих оптимальным поведением в стационарных случайных средах? Оказалось, что это зависит все от того же параметра m, который характеризует инерционность смены действий автоматами. Чем больше инерционность, тем устойчивее описанное выше распределение автоматов по кормушкам, тем менее достижима глобальная цель коллектива. Коллектив ведет себя вполне разумно с точки зрения входящих в него индивидов, но не наилучшим образом с точки зрения "пастуха".

Можно, однако, несколько изменить управление автоматами, осуществляемое средой. Будем суммировать все индивидуальные выигрыши автоматов и делить их поровну на всех членов коллектива. Сигнал наказания среда будет выдавать на все автоматы коллектива одновременно, если выигрыши автоматов меньше среднего ожидаемого выигрыша при оптимальном с точки зрения коллектива распределении автоматов. В противном случае на входы автоматов коллектива поступает сигнал поощрения. Подобная процедура называется принципом «общей кассы». Она позволяет организовать поведение коллектива таким образом, что при достаточно больших значениях m достигается цель «пастуха» и автоматы занимают все четыре кормушки. Заметим, что при малых значениях m принцип «общей кассы» может привести к тому, что глобальная цель коллектива не достигается, а личные выигрыши автоматов становятся в среднем меньше, чем при отсутствии «общей кассы». М. Л. Цетлин со свойственной ему любовью к ассоциативным образам, называл это явление «вредом уравниловки при низком уровне сознательности».

В нашей модели автоматы не получают никакой непосредственной информации о действиях других участников коллектива и не общаются между собой. Можно, однако, ввести такое общение. Простейшей его формой является парное взаимодействие автоматов, при котором они обмениваются информацией о своем текущем состоянии или о предстоящем выборе кормушки (о ближайшем своем действии). Уже такое взаимодействие автоматов, если пары обменивающихся информацией автоматов выбираются случайным равновероятным образом, приводит к тому же результату, что и «общая касса». Но возможны и более глубокие формы взаимодействия, которые позволяют моделировать весьма сложные формы поведения коллектива. Примером может служить так называемое *рефлексивное поведение,* при котором автомат, прежде чем принять решение о своих действиях, прогнозирует, какие действия совершают другие автоматы в коллективе. В таких коллективах можно ввести различие между автоматами, характеризуемое различными рангами рефлексии. Под *рангом рефлексии* понимается уровень отражения в автомате глубины рассуждений других членов коллектива.

При нулевом ранге рефлексии автомат принимает решение о выборе действий только на основании сигналов, от среды. При первом ранге рефлексии он предполагает, что остальные члены коллектива имеют нулевой ранг рефлексии, проводит рассуждение за них на основании описания общей ситуации, а затем на основании полученного прогноза принимает решение о выборе своего поведения. При втором ранге рефлексии автомат предполагает первый ранг рефлексии у остальных участников коллектива. Проведя предварительное рассуждение об их выборах, он опять принимает наилучшее при выполнении его предположений решение о выборе.

Таким образом, рефлексия связана с процедурой «размышления за другого». Ей соответствуют такие языковые конструкции, как «Я думаю, что он думает...» или «Он думает, что я думаю...». Блестящим примером рефлексии глубокого ранга может служить следующее английское стихотворение в переводе

С. Маршака:

«— *Он целовал вас, кажется?*
— *Боюсь, что это так!*
— *Но как же вы позволили?*
— *Ах, он такой чудак!*
Он думал, что уснула я
И все во сне стерплю,
Иль думал, что я думала,
Что думал он: я сплю!».

Конечно, когда мы предполагаем в другом человеке наличие некоторого ранга рефлексии, мы можем и ошибаться. И девушка — героиня стихотворения также может ошибаться, приписывая молодому человеку слишком глубокий ранг рефлексии. Возможно, что он у него на самом деле был нулевой.

Поэтому и автоматы в коллективе, предполагающие наличие у остальных членов коллектива ранга, на единицу меньшего того, которым они обладают сами, могут жестоко ошибаться. При моделировании на ЭВМ поведения коллективов автоматов, в котором разные автоматы обладали различными рангами рефлексии, выяснилось, что целесообразное поведение отдельных автоматов и целесообразное поведение всего : коллектива зависят от того, как в процентном отношении распределены ранги в коллективе. Например, если большинство автоматов имеет нулевой ранг, а некоторая небольшая группа автоматов имеет первый ранг рефлексии, то при распределении но кормушкам автоматы с нулевым рангом рефлексии имеют больший личный выигрыш, чем автоматы с нулевым рангом. Коллектив же автоматов ведет себя целесообразно, получая общий выигрыш, превышающий общий выигрыш коллектива, в котором рефлексии нет (но ниже, чем при достаточно больших значениях m в условиях общей кассы). Однако если большинство автоматов имеет нулевой ранг, а часть автоматов — второй ранг рефлексии, то эта последняя группа автоматов будет проигрывать из-за «самообмана», так как она неправильно прогнозирует действия большинства автоматов. Коллективный выигрыш в такой ситуации также будет более низким, чем в однородном коллективе с нулевым рангом рефлексии.

Свойство рефлексии может использоваться при организации человеческого поведения для таких процедур, как обман, блеф, навязывание собственного представления и мнения и т. п. Об этом пишет, например, Андрей Платонов в рассказе «Иван Великий».

«Обычно враги обстреливают из пушек свои оставленные рубежи, понимая, что мы можем поселить своих солдат в траншеях, отрытых прежде фашистами. Но мы, понимая немцев, обычно не расселяли свои войска в траншеях, оставленных противником. А когда враги, проведав об этом, переставали обстреливать оставленные траншеи, считая их пустыми, мы начинали иногда пользоваться ими».

Подобный способ поведения вообще весьма характерен для людей и потому возможность моделирования его представляется интересной и поучительной.

Модели коллективов автоматов способны имитировать и другие формы целесообразного поведения и поведения, связанного с теми или иными ошибочными оценками реальности. Чтобы не утомлять читателя множеством таких примеров, укажем еще на одну модель, в которой для нас важен переход от двоичной логики, использовавшейся в предшествующих примерах (наказание — поощрение), к логике более высокого порядка.

Пусть автоматы совершают свои прогнозы о действиях других автоматов на основании лишь частичной информированности о сложившейся ситуации. Тогда группа автоматов, относительно которой формируется прогноз об их будущих действиях, распадается на три подгруппы. Относительно первой из них прогнозирующий автомат точно уверен, что они совершат действие, благоприятное для него. Относительно второй — что эти автоматы совершат действия, неблагоприятные для него. Но относительно прогноза о действиях автоматов третьей подгруппы сказать ничего определенного нельзя. Для этого слишком мало информации. Обозначим через n_1, n_2 и n_3 соответственно число автоматов, попавших в первую, вторую и третью подгруппы.

Различие в поведении автоматов будет связано с тем, что они делают с автоматами из третьей подгруппы. Они могут, например, отнести все n_3 автоматов к первой подгруппе, т. е. предположить, что «мир устроен прекрасно и ждать от него надо только хорошего». Такие автоматы можно было бы назвать крайними оптимистами. Наоборот, автоматы, которые объединяют третью подгруппу со второй, являют собой пример крайних пессимистов. Объективисты делят третью подгруппу пополам (или почти пополам, если n_3 нечетно) и распределяют эти «половины» по оставшимся двум подгруппам. Ясно, что при различных соотношениях параметров n_1, n_2, n_3 и при различных распределениях третьей подгруппы по первым двум возникают различные уровни оптимизма (пессимизма) у автоматов.

Логика работы подобных автоматов опирается на утверждения, принимающие три значения истинности: истинно, ложно и неопределенно. Вместо двоичной логики возникает троичная логика. В принципе, введя различные градации неопределенности (например, неопределенно, по ближе к истине; полностью неопределенно; неопределенно, но ближе ко лжи), мы можем получать логики более высоких порядков (в данном примере логику пятого порядка). Многозначные логики и связанные с ними многозначные исчисления высказываний и предикатов — новая, бурно развивающаяся в настоящее время область логики. Достигнутые здесь результаты позволяют надеяться на построение моделей поведения и человеческих рассуждений, более адекватно описывающих истинное положение дел, чем это делала классическая двоичная логика.

Но о многозначных и других экзотических логиках мы будем говорить позднее. Вернемся к нашим автоматам. Если теперь в задаче о распределении их по кормушкам ввести различие в автоматах, приписав им различные уровни оптимизма (пессимизма), то мы получим неоднородный коллектив автоматов. Моделирование его поведения показывает, что целесообразность поведения отдельных автоматов и всего коллектива зависит, как и в случае различной рефлексии автоматов, от распределения уровней оптимизма (пессимизма) по коллективу. В крайних случаях, когда все члены коллектива являются крайними оптимистами или пессимистами, а значение n_3 велико, автоматы и весь коллектив ведут себя нецелесообразно. Когда коллектив состоит только из объективистов, то его поведение совпадает с обычным поведением автоматов в однородном коллективе. При других распределениях уровней оптимизма (пессимизма) коллектив может приближаться к функционированию при наличии «общей кассы»,когда m велико.

Самовоспроизведение и эволюция. Кажется, что эти свойства живого столь таинственны и глубоки, что их моделирование заранее обречено на неудачу. И тем не менее именно в этой области теории автоматов есть, чем гордиться. В первую очередь этим мы обязаны выдающемуся специалисту в области кибернетики Джону фон Нейману, роль которого сравнима с ролью А. Эйнштейна в физике.

Он родился в 1903 году в Будапеште, занимался математикой, закончил Будапештский университет в 1927 г. и переехал в Берлин, а затем в Гамбург. Времена становились все хуже и хуже. В 1930 году он переезжает в США, где и заканчивает свой жизненный путь в феврале 1957 года. За это время фон Нейман сделал весьма много. Он участвовал в разработке проекта атомной бомбы, занимался созданием водородной бомбы, теорией автоматов, созданием ЭВМ, предоказанием погоды, разработкой межконтинентальной ракетной программы и многими другими вопросами. Влияние его на развитие современной науки еще полностью не оценено.

Следуя А. Тьюрингу, поставившему два глобальных вопроса: 1) можно ли описать процесс, называемый алгоритмическим, строго формально? 2) существует ли описание универсального автомата, способного воспроизвести любой алгоритмический процесс?, на которые Тьюринг получил положительные ответы, фон Нейман также поставил два вопроса: 1) может ли один автомат сконструировать другой автомат по его описанию? 2) существует ли универсальный конструирующий автомат?

Понятие «сконструировать» фон Нейману пришлось уточнить. Он рассмотрел двумерное пространство, как бы разделенное на стандартные клетки одного и того же размера. Каждая клетка этого пространства представляет собой простейшее автоматное устройство, способное находиться в одном из состояний из некоторого фиксированного множества состояний. Взаимодействие клеток осуществляется через четырех ближайших соседей. Каждая клетка стандартным образом соединена с НИМИ и может сообщать соседям о своем состоянии в воспринимать состояние каждого соседа.

Сконструировать автомат в этом пространстве — это значит перевести часть клеток исходного пространства в определенные состояния так, чтобы в прямоугольнике, ограничивающем эти клетки, происходил процесс такого же типа, как и в конечном автомате, а вне этого прямоугольника все клетки находились бы в состоянии покоя. Другими словами, конструирование заключается во внешнем возбуждении части клеток среды, обеспечивающем дальнейшее течение нужного нам процесса.

Прежде чем, излагать дальше идеи фон Неймана, опишем одну игру, которой активно развлекаются программисты наших дней, а после описания ее условий популярных журналах — и многие любители комбинаторных головоломок. Эта игра, изобретенная Джоном Конвеем, имеет следующие правила.

Как и фон Нейман, Конвей рассматривает двумерное клеточное пространство. Каждая клетка этого пространства может находиться в двух состояниях: активном и пассивном. Состояния клеток связаны с соседними клетками. В отличие от фон Неймана, Конвей для каждой клетки рассматривает не четырех, а восемь соседей. В начальный момент в пространство извне вносится некоторая конфигурация из активных клеток. Это зародыш. Далее переход клеток в активное (рождение) и пассивное (отмирание) состояния происходит по следующим правилам, предложенным Конвеем:

1) если данная активная клетка имеет двух или трех активных соседей, то она продолжает оставаться в активном состоянии;

2) если данная пассивная клетка имеет трех активных соседей, то она переходит в активное состояние (рождается);

3) если данная активная клетка имеет число активпых соседей, меньшее "двух или большее трех, то она переходит в пассивное состояние (отмирает).

В зависимости от вида зародыша его дальнейшая судьба может быть различной. Посмотрим на рис. 4. Для случая *А* судьба зародыша трагична. Он погибает на втором шаге своего существования. Для случая *Б* зародыш превращается в устойчивую конфигурацию, названную Конвеем «блок». Блок, полученный из зародыша на втором шаге, остается неизменным на протяжении всех последующих тактов, так как каждая активная клетка имеет трех активных соседей, что поддерживает ее существование. В случае *В* имеет место колебательный процесс. Зародыш как бы периодически поворачивается на девяносто градусов. Эту конфигурацию Конвей назвал «светофор». Остальные зародыши-триплеты, состоящие из трех активных клеток,

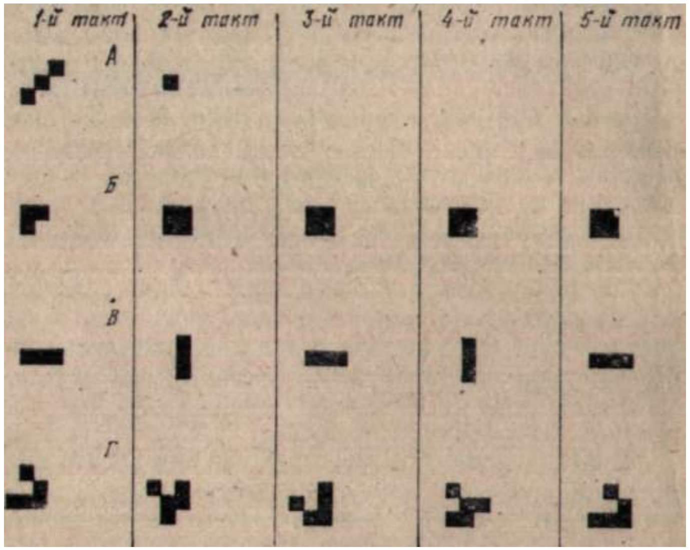


Рис. 4

погибают после первого или второго такта (ясно, что по условиям игры зародыши из одной или двух активных клеток погибают тотчас же).

Насколько неожиданной может быть судьба зародыша, показывает случай *Г*. Эта конфигурация названа Конвеем «глиссер». За два своих такта он смещается, поворачивается на 90° и зеркально отображается, а еще через два такта глиссер восстанавливается в прежнем виде, сместившись от своего начального положения вправо. Так он и будет скользить по клеточному пространству, пока не уйдет в бесконечность или не «разобьется» о его край.

Играя в игру Конвея, нельзя отделаться от ощущения, что конфигурации, причудливо возникающие на клетках пространства, живут какой-то собственной

жизнью. И, наверное, поэтому, Конвей так и назвал придуманную им игру — «Эволюция».

Вернемся теперь к конструированию автомата в среде. Конструирующий автомат сам расположен в этой же среде, занимая в ней определенный прямоугольник. К этому автомату добавлен потенциально неограниченный линейный массив клеток, в котором описывается закон функционирования конструируемого автомата. Описание каждого конкретного автомата конечно, так как полностью задается конечной автоматной таблицей. Но поскольку речь идет об универсальном конструирующем автомате, то необходимо предусмотреть возможность иметь описание любого автомата, не ограничивая априорно размеры его автоматной таблицы.

На основании этого описания конструирующий автомат должен создать в выделенной части свободного клеточного пространства автомат, являющийся целью его работы.

Центральным результатом фон Неймана является решение проблемы создания конструирующего автомата. Он предложил список из 29 состояний определенного типа и функции перехода для клеток среды, зависящие от состояний четырех ее соседей, в одно из этих 29 состояний. И доказал, что при этих условиях существуют положительные ответы на поставленные им два вопроса.

Итак, можно создать конструирующий автомат и построить универсальный автомат подобного типа, т. е. автомат, способный воссоздать в среде любой другой конечный автомат. В частности, этим конструируемым автоматом может быть и сам конструирующий автомат. А это приводит к положительному ответу на вопрос о возможности самовоспроизведения автоматов в клеточной структуре (вспомните об игре «Эволюция», в которой пространство может заполняться копиями первоначального зародыша). И для самовоспроизведения достаточно все тех же 29 состояний, которые использовались при конструировании.

«Что наша жизнь? — Игра!». Именно так определяется в партии Германна в «Пиковой даме» важность игрового поведения людей в различных жизненных ситуациях. Специальная математическая дисциплина, называемая «Теория игр» (у истоков ее, кстати, находился все тот же фон Нейман), находит большое применение в экономике, теории управления, социологии и других науках. Но нас будет интересовать не теория процессов, называемых *играми,* а имитация их с помощью технических средств.

Стокгольм 1976 года. Концертный зал «Биргер ярл». Большое количество зрителей с неослабным вниманием следит за шахматными поединками на нескольких досках. Все как на обычном шахматном турнире, если бы не отсутствие шахматистов. Их здесь нет, ибо идет первый всемирный матч между машинными программами, играющими в шахматы. Матч проходит на таком уровне, что в зале машинные выборы ходов вызывают гул одобрения и аплодисменты, хотя здесь собрались не только энтузиасты шахматных программ, но и скептики. Как известно, первым чемпионом мира среди шахматных программ стала советская программа «Каисса» (средневековый английский поэт Уильям Джонс, большой поклонник этой древней игры, считал, что греки недооценивали специальный вид искусства — шахматы и поэтому при перечислении муз забыли о музе шахмат, которой Джонс и дал имя «Каисса»).

Психологическое значение факта борьбы ЭВМ за звание чемпиона мира по шахматам трудно переоценить. Пожалуй, впервые мы воочию увидели, как ЭВМ властно вторгается в область интеллекта: заменяет интеллект шахматиста неким его машинным подобием. Конечно, нужно все время помнить, что у истоков машинных программ стоят люди, вложившие в них все те сведения об игре в шахматы, которыми они располагали. Можно считать, что за звание чемпиона мира борются не программы, реализованные на ЭВМ, а сами программисты, их создавшие. Но это так и не так. Программисты заложили в память ЭВМ некоторые общие принципы игры. Но как помогут эти принципы ЭВМ для выбора хода в конкретной ситуации, с которой она столкнется в игре, программистам до конца не ясно. Идет борьба идей, которая в каждом конкретном случае может обернуться победой, а возможно, и горьким поражением.

Читатель, наверное, знает, что ЭВМ играют не только в шахматы. Шашки и древняя игра калах, домино и азартная карточная игра «железная дорога» (знаменитая «железка» русских игорных домов), морской бой и пришедшее из далекой Японии го — во всех этих и многих других играх ЭВМ пробуют свои силы с меньшим или большим успехом. Для нас интересно, как это происходит.

В основе всех игровых программ для игр типа шашек или шахмат лежит одна общая идея, связанная с процедурой поиска по дереву игры. Суть этой процедуры мы обсудим на примере очень простенькой игры, которой увлекаются младшие школьники и многие дошкольники. Это игра в «крестики-нолики» на поле размером 3x3 клетки. Будем считать, что нам необходимо научить ЭВМ играть в эту игру, т. е. создать программу игры в «крестики-нолики» на поле 3 x 3. Для

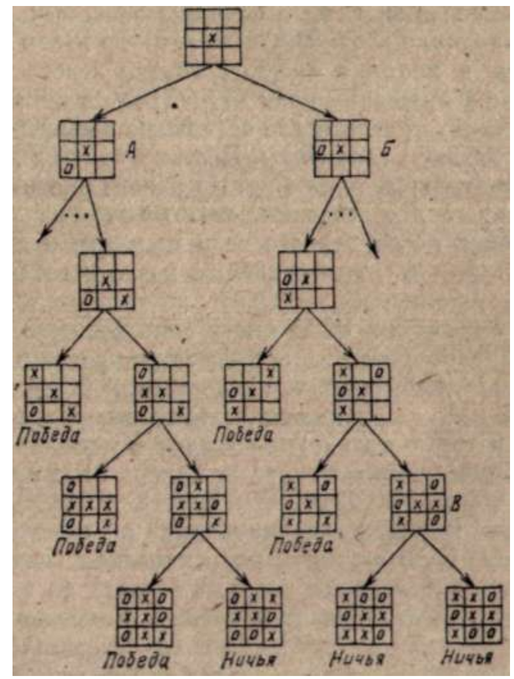


Рис. 5.

облегчения дальнейшего изложения (принципиально это ничего не меняет) рассмотрим только тот вариант игры, при котором машина делает первый ход. Для определенности будем считать, что ЭВМ ставит в клетки поля крестики, а ее противник (человек или другая ЭВМ) — нолики. Правила игры предполагаются известными всем читателям книги. Наши рассуждения мы будем сопровождать соответствующими изображениями складывающихся в процессе игры ситуаций (рис. 5).

Так как ЭВМ делает первый ход, то ее наилучшее решение — поставить крестик в клетку, находящуюся в середине игрового поля. На этот ход противник может ответить восемью различными способами, ибо для написания нолика он имеет восемь свободных клеток. Однако ясно, что эти восемь возможностей, по существу, представляют собой только две возможности: поставить нолик в угловую клетку или поставить его в неугловую клетку. Происходит это потому, что поле игры симметрично. Если в качестве основных двух ответов противника зафиксировать нолик в левом нижнем углу поля и нолик в средней клетке левого столбца поля, то при вращении поля относительно центральной клетки можно привести все остальные выборы противника к этим двум вариантам. Поэтому будем в дальнейшем рассматривать только их. На нашем рисунке этому соответствуют две стрелки, выходящие из позиции, сложившейся после первого хода машины, к двум возможным позициям, возникающим после ответа противника.

В этих позициях ЭВМ имеет семь возможных ответов. Проанализировав их последствия, программист может выбрать, например, те продолжения игры, которые показаны на нашем рисунке. Остальные возможные стрелки, которые на рисунке «ведут в никуда», можно в дальнейшем не рассматривать. Теперь ЭВМ находится в одном шаге от победы, а у противника есть только один ход, который не приводит к немедленному проигрышу. Если этот ход не реализован (как в этом случае пошел противник, роли не играет), то на следующем шаге ЭВМ добивается победы. Если же противник выбрал ход правильно, то из оставшихся пяти возможных ходов программа выбирает тот, который указан на рисунке. Теперь опять возможны два случая. Если противник ошибется (как — это для нас не играет роли), то ЭВМ победит его. Если же этого не произойдет, то машина выбирает ход, указанный на рисунке. В зависимости от очередного хода противника машина либо выигрывает, либо сводит игру в ничью.

Можно рассмотреть и тот случай, когда первым игру по жребию начинает не ЭВМ, а ее противник. Построив подобную же диаграмму, можно придти к выводу, что при любой ошибке противника ЭВМ выиграет партию, а если противник не ошибается, то сведет ее вничью.

Взглянем еще раз на рисунок. Он напоминает *дерево,* перевернутое «вверх ногами». Начальная позиция — как бы комель этого дерева. С развитием партии оно ветвится, и на концах ветвей находятся завершающие позиции игры. Каждая стрелка отвечает одному полуходу, а пара последовательных стрелок — ходу игры, когда оба игрока сделали свои выборы. На нашем дереве шесть полуходов (начальный полуход, принадлежащий ЭВМ, которая ставит на пустом поле первый крестик, на рисунке не показан), и, следовательно, три хода.

Если мы проанализируем это дерево, то увидим, что составляющие его части неравноценны (эти составляющие части обычно называют *поддеревьями*). Например, после первого хода мы имеем два поддерева: левое и правое, которые на рисунке отмечены буквами *А* и *Б.* Если считать, что противник ошибается случайно и равновероятно, то дерево*) *А* лучше дерева *Б,* ибо в дереве *А* три конечные ветви заканчиваются победой ЭВМ и только одна — ничьей, а в дереве *Б* — две конечные ветви заканчиваются ее победой и две — ничьей. Кроме того, заметим, что если ЭВМ получила позицию, из которой начинает ветвиться дерево *В,* то игра может и не продолжаться: независимо от выбора дальнейших ходов игра все равно закончится вничью.

Такие деревья могут быть составлены не только для игры в «крестики-нолики», но и для других игр. Только для более сложных игр деревья эти будут весьма большими. Например, дерево игры для простых шашек с учетом всех симметричных позиций насчитывает около десятка тысяч ветвей, идущих от комля. А дерево для игры в шахматы таково, что для перечисления его ветвей не хватило бы времени работы современной ЭВМ в течение жизни программиста, поставившего перед ней такую задачу.

*) Сами поддеревья являются деревьями для своих продолжений, поэтому о них можно говорить снова, как о деревьях.

Как же действуют тогда игровые программы, играющие в игры типа шашек или шахмат? Так как дерево игры представить в памяти ЭВМ и проанализировать заранее, как мы это сделали для игры в «крестики-нолики», невозможно, то вместо этого вводятся специальные процедуры, позволяющие просматривать не все возможные продолжения к уже построенной части дерева (она отражает то, как протекала партия до этого), а лишь некоторые из них и на фиксированную глубину. Основной проблемой при этом становится выбор этих продолжений и выбор той глубины просмотра, которую нужно осуществить, чтобы получить априорную оценку варианта продолжения игры.

Когда на дереве игры в «крестики-нолики» мы указывали очередной ход, который делает ЭВМ, то мы предварительно анализировали, какой именно ход является в данной позиции наиболее выгодным с точки зрения ЭВМ. Мы отсекали все те ходы, которые неизбежно привели бы к проигрышу партии, среди остальных на первое место ставили продолжения, приводящие к неизбежному выигрышу, а вслед за ними такие, которые в худшем случае обеспечивают нам ничью, а в лучшем — выигрыш. Другими словами, мы как-то ранжировали возможные продолжения игры и выбирали наилучшее с точки зрения этого ранжирования продолжение.

Если дерево игры велико, то такое ранжирование принципиально тоже можно провести. Правда, оно будет неточным, так как просмотр дерева мы сможем осуществить лишь частично. При этом мы рассуждаем за нашего противника (при выборе его ответного полухода) так, как будто бы он уже знает наш полуход и отвечает на него наиболее плохим для нас образом. На самом же деле противник может ответить и по-другому. И даже более сильно, чем мы предполагали из-за неполного анализа дальнейших вариантов продолжения игры.

Именно поэтому матч шахматных программ и вызвал такой интерес. В нем как бы сравнивались различные методы оценки позиций на дереве игры, придуманные различными программистами. Одни программы действовали «вширь»: перебирали много вариантов продолжений, но на небольшое число полуходов. Другие спускались «вглубь» на большое число полуходов,

но анализировали лишь небольшое число вариантов, ибо время на обдумывание машиной своего хода в матче было, как и для людей, фиксированным и просрочка его грозила поражением.

Программа «Каисса» во время первого мирового первенства анализировала дерево игры следующим образом. Если имелся форсированный вариант продолжения, приводящий к проигрышу противника, он просматривался ею достаточно глубоко. При обеспеченном выигрыше этот вариант имел наиболее высокий ранг и реализовывался машиной. Если же форсированных вариантов не было, то программа анализировала дерево игры, возникающее в текущей позиции, на пять полуходов вперед и принимала решение о выборе своего хода, . сравнив с помощью специальных оценочных функции (они учитывают материальное преимущество при выигрыше и отдаче фигур на этом продолжении и позиционное преимущество в результате продвижения фигур) все варианты. За время, прошедшее с тех пор, шахматные программы, в том числе и программа «Каисса», улучшили свои характеристики и, в частности, увеличили глубину перебора в нефорсированных вариантах.

Эвристика и эвристики. Откуда берутся приемы усечения числа перебираемых вариантов в дереве игры? Конечно, из человеческого опыта, из практики людей-шахматистов, решающих во время шахматной партии те же самые проблемы, что и ЭВМ. Недаром среди создателей лучших шахматных программ так много известных шахматистов. И среди них неоднократный чемпион мира М. М. Ботвинник. Создаваемая под его руководством программа «Пионер» вберет в себя весь опыт этого выдающегося шахматиста. И когда состоится очередное первенство мира, возможно, что именно программа «Пионер» продемонстрирует наилучшее качество игры. Но кто может предугадать, к кому будет благосклонна муза шахмат!

Человек, решая разнообразные интеллектуальные задачи, очень часто сталкивается с. невозможностью полного анализа последствии тех или иных своих решений. У него не хватает ни времени, ни сил, чтобы до конца проанализировать все возможные варианты развития событий. В этих условиях он поступает чисто по-человечески. Он отбрасывает те варианты, которые ему