

А.В.Никитин

Трансформация понимания логики.

Классическое формулирование понятия логики, как искусства спора, искусства доказательства, искусства нахождения истины вполне верно отражает основное направление логики как науки. Так и формировалось определение логики, постоянно изменяясь во времени, и в зависимости от понимания. Сегодня можно сказать и так:

Логика, это способ обоснования собственных решений.

Как мы понимаем, это не наука. Наукой стало объяснение логики для человека и его разума. Все это верно, но это только результат, а не сама логика.

Хочется понять, не *что* мы делаем, а *как* мы это делаем, и, может быть - *зачем*...

Но, стоит нам посмотреть на путь пройденный логикой от логики «слова» до математической, как вопрос – *зачем*, отпадает. Цель видна и понятна.

Это средство формализации понимания, приведение к стандартному доказательству, действующему в тот или иной момент времени.

Формализация логики уходит от ограничений разумного понимания умозаключения в этических рамках, к абстрагированному и формальному математическому пониманию алгебраической формулы логической операции.

Изменение понимания логики, показанное в работах [9, 10], происходило почти 2500 лет. Общее направление этого изменения стали видны только сейчас, когда, наконец-то, построен образец того, к чему так стремилась мысль ученых. Машина. Компьютер.

Он показал то, что было недоступно ранее. Работу машинной логики.

Эту цель нашего понимания логики мы увидели только на этапе создания электронной вычислительной машины. Мы, наконец, осознали, что машина не понимает наших моральных и этических норм, ей безразлично, что мы думаем о проводимых ею действиях. Она действует строго в рамках составленной для неё программы действий.

Но многовековой стереотип человеческого понимания продолжает срабатывать и в отношении машины. Мы все время пытаемся «очеловечить» машину. Найти в её действиях элементы разумности, пусть и на основе наших же программных компонентов «искусственного интеллекта». Вот же, она уже почти понимает нас...

Но органическое «вживание» в программу машины вкладываемых программистом элементов интеллекта, когда они органично вписываются в контекст и становятся неотъемлемой частью результата действия, происходит очень редко.

Чаще происходит постепенное накапливание результатов действия этих «штучек» до абсурдного, что начинает вызывать только отвращение при их очередном проявлении.

Пример? Новый Office, конечно же, удобнее старого. Но...

Открываем Word. Вроде все привычно.

Но, вот мы находим в своем компьютере какой-нибудь старый документ и копируем из него какой-то отрывок с целью его перенесения в новый, только сейчас формируемый...

Пытаемся вставить этот скопированный отрывок в новый документ, и что?

Почему-то изменился шрифт, куда-то делись гиперссылки, ... отрывок перестал быть похожим на оригинал. Я этого совсем не хотел. У меня была задача просто скопировать, а не трансформировать текст по правилам новой программы. Кто и почему пытается сделать это вместо меня?

Это сделал программист. Он посчитал, что его понимание моих желаний более обоснованно, чем мое собственное. И делать так, как делает его программа, это более правильно, чем думаю я.

При этом мне не говорят, что надо сначала сформировать свой стиль, узаконить его в программе, а потом уж она будет свято его придерживаться. В том числе и при копировании из разных источников, с разными использованными шрифтами. Она будет все сводить к одному, выбранному пользователем, стилю формирования документа...

Да, мне скажут, что прежде чем приступить к работе, необходимо познакомиться с инструкциями, там все написано...

Но, раньше мы же что-то уже изучали, зачем нам опять это изучать, когда мы просто хотим продолжить то, что у нас уже было сформировано, в том числе и стиль. Нет, нам предлагают снова да ладом, так, как за нас захотел программист.

Всё так. И не так. Я вижу привычное название и жду от программы тех же привычных действий, о которых я давно знаю из чтения прошлых инструкций. И вдруг не получаю этих привычных действий, а название – то же, что и раньше.

Так это же меня обманули, подставив мне другой товар вместо ожидаемого, оставив прежнее название и внешний вид.

А я это начинаю понимать, только уже споткнувшись на выполнении первых действий, привычных для меня, но не для этой программы.

Вот только теперь я начинаю искать причину этого. И может быть читать руководство. Но при этом уж точно не забуду сказать всё, что я думаю, и о программе, и о программисте, и о компьютере, и о...

Потому, что у меня своя логика, и она тоже правильная, пусть и только для меня.

Вот здесь остановимся. И задумаемся.

Почему у нас, у каждого своя логика? Зачем и почему нас надо учить «правильному» мышлению? Почему у каждого свои понятия о логичности...?

И, тем не менее, мы все в чем-то сходимся, часто думаем одинаково и все считаем те или иные выводы справедливыми и логичными, даже если в ответах на другие вопросы у нас иное мнение. Есть, значит, в наших логических действиях какая-то всеобщая объективность и понимаемость. И одинаковость действий, которая и приводит нас к одним и тем же общим выводам и результатам, которые мы все считаем объективными и обоснованными.

Значит, есть внутри нас какая-то автоматическая логика, которая делает всю эту работу по логическому обоснованию наших действий, даже мимолетных.

Наше понимание логики стремится к формальным автоматным операциям, лежащим в основе наших мыслей, действий, поступков...

Мы стараемся найти разумное объяснение нашим собственным действиям и поступкам через их понимание в образах сознания. Фиксируем результаты наших поисков в виде правил, законов, формул...

Век назад человек наконец-то сформулировал автоматичность своих логических действий через самый формальный на тот момент аппарат доказательства, через математику. Именно эту, математическую логику человек и вложил в создаваемую им вычислительную машину.

Почему в вычислительную?

Потому, что логическую машину человек так и не создал. Не смог. И сделал то, что мог на том этапе развития понимания логики.

Тем не менее, наконец-то заработал именно этот полностью формализованный вариант логики, способный работать самостоятельно без вмешательства человека хотя бы на одном этапе – решении задачи. Математическим методом, но хоть так.

Это уже само по себе – достижение, к которому человек, может быть и неосознанно, но, стремился все эти тысячи лет.

Цель достигнута. Не так, как хотелось, как мечталось, но ... достигнута.

Машина действует сама, какое-то время. Выполняет поставленные задачи, устраняет возникающие проблемы, управляет...

Вот, пока на этой мажорной ноте мы остановимся в сегодняшней оценке современного понимания логики. И немного вспомним, как это было. Как шло развитие логики...

Логика слова.

Древние ученые искали способ доказательства своей правоты в споре за истину. Этот способ был найден в геометрии и потом был назван логикой. Это был сравнительный способ доказательства с формированием обоснования каждого шага. Приближение к истине каждого последующего состояния обоснования в сравнении с предыдущим стало основой этого способа доказательства. И если в геометрии в процессе доказательства сравнивались зримые графические построения, то в логике сравнивались словесные суждения, так или иначе объясняющие предлагаемые сравнения.

Сегодня мы бы сказали, что сравнивались определения понятий и действия. В виде формальных предложений выстроенных в определенном порядке. Этот порядок следования [7], как предложений, так и понятий в предложениях, постепенно сам стал доказательством истины. И даже только одна возможность выстраивания понятий в такую схему уже автоматически доказывала истинность всего построения.

Конечно же, мы говорим о силлогизме. Силлогизм в разных странах и эпохах включал в себя разное количество суждений. В категорическом силлогизме осталось два суждения, достаточных для вывода умозаключения. Той самой истины, ради которой и строился этот силлогизм.

Мы не будем повторять истории развития формальной логики от Древней Греции и Индии до наших дней. Мы это уже сделали раньше [8]. Здесь мы попробуем оценить объективные причины того, что логика так долго была только логикой слова или высказываний.

Почему же так получилось?

Прежде всего, потому, что человек старался все свои умозаключения свести к сознательной деятельности в том виде, как тогда это понималось. К своему «внутреннему голосу», который и ассоциировался с сознанием, мышлением, пониманием...

Но, «внутренний голос» никак не хотел выводить логические решения, он их только озвучивал, а как они выводились, это всегда оставалось загадкой нашего мышления. Решения, выводимые сознанием, записывались и многократно проверялись на точность и правильность формулировок. Эти записанные и многократно проверенные наборы суждений, выстроенные в определенном порядке, ведущие к определенному, и часто, единственно возможному результату в виде умозаключения, принимались за логические. А процесс их выведения стали называть логикой.

На этом этапе произошла первая крупная подмена понятия. Основой логики стал не способ построения логического доказательства, а сразу его стандартная форма – силлогизм. Произошло это по причине искусственного занижения статуса этой новой науки.

Логика очень долго оставалась частью философии, хотя, по сути, была её основой. Потому, что логические построения и выводы стали основой обоснования философских концепций и теорий, но философия всегда была «царицей» наук и до сих пор не желает отдавать это лидерство. До сих пор.

Хотя, вполне возможно, что это так и есть, что действительно, философия – главная наука. Не знаю. Но логика уж точно не часть философии. Это изначально самостоятельная и очень важная наука.

История же распорядилась иначе, и логика очень долго оставалась частью философии. Причем, искусственно принижаемого статуса. И потому, логикой до сих пор называют любое последовательное движение мысли в заданном направлении. Тем более в философии...

Философия держится на слове. Из слов складываются все философские построения. Слово отражает все движения мысли. А так же, как считалось всегда, и все движения в сознании. Это мы отметим отдельно.

Вот здесь мы попадаем в сложное противоречие между словом и мышлением. Это противоречие наиболее ярко проявляется в понимании роли сознания и подсознания.

Что тут главное?

Подсознательное нахождение правильного решения сложной логической задачи никогда не принималось за «правильное».

[Интуиция](#) была сразу поставлена [противоположностью](#) логике [10]. Хотя сегодня можно точно сказать, что это одно и то же. Логика и работает в подсознании, а в сознании она только отражается в виде отдельных мыслей, часто не связанных друг с другом.

Но такое положение вещей никак не устраивало древних и средневековых философов. И потому, интуиция всегда принималась как непостижимое Начало, но в состав логики не включалась, и более того, всегда противопоставлялась логике.

На основе такого двойственного отношения к интуиции основой силлогизма было принято *умозаключение*:

[Умозаключение](#), умственное действие, связывающее в ряд "посылок" и "следствий" мысли различного содержания; У. реализует в плане "внутренней речи" присущие индивидуальному (или общественному) сознанию нормы и типы такой связи, которые и являются в каждом отдельном случае психологической основой У. Если эти нормы и типы совпадают с правилами и законами логики...

И [там же](#) читаем:

Кроме того, нормы, определяющие "законность" У., не обязательно должны быть логическими. Например, [неполная индукция](#) – это именно У., а не логический вывод, поскольку связь посылок и заключений в индукции имеет фактическую и психологическую основу (в виде известных норм генерализации), но не имеет логической основы – формальных правил, определяющих ход мысли от частного к общему. У. отлично и от рассуждения: последнее – всегда сознательное и произвольное действие мышления, а У., по крайней мере, в его основе, может быть и подсознательным, и произвольным актом.

Вот так устанавливается «законность» умозаключения.

Именно так: не всегда сознательное умозаключение, а не [рассуждение](#), приводит к выводным результатам силлогизма. Таким образом, интуицию в логику не пускают, но и из логики не убирают. Оставляя на крайний случай, как последнюю соломинку для спасения.

Оставляя в логике путь к интуиции, теоретики логики фактически принимали положение, что логические решения принимаются не только сознательно...

Но как соединить слова и бессловесную интуицию?

Для древних и средневековых логиков это невозможно. Разум должен быть главным в логике. И слово, как символ разумности человека. Слово замкнуло на себя все направления развития логики как науки на долгие века.

Логика символа.

Этот период развития логики наступил после разработки основ алгебры [Ф.Виетом](#).

Конечно же, это продолжение внедрения в логику математических методов доказательства, начало которому положено геометрией еще в древние века.

Наконец-то пришло время, когда слово перестало однозначно определять направление развития логики. Эстафета была передана более абстрактному [символу](#). [10]

Это автоматически определило дальнейшую формализацию подхода к логическим высказываниям и суждениям. Высказывания приняли вид математических выражений, как например: $A+B=C$.

Абстрактная математика вывела логику на общие решения в символьном выражении, понимаемые одинаково всеми, и не зависящие от конкретного языка общения людей.

Через пару веков в логике понятий была сформирована теория [кванторов](#), которая смыкается с понятием [предиката](#).

Одновременно с развитием логики понятий развивается и [семиотика](#), исследующая свойства [знаков](#) и [знаковых систем](#). Любых. Естественных и искусственных. Это привело к созданию формальных языковых систем описания алгоритмов решения логических задач. Или иначе, это [языки программирования](#).

Математическая логика.

Тем временем применение математики в формальной логике позволило создать уже математическую [логику высказываний](#). На её основе возникла [логика первого порядка](#). Чуть позже возникала [логика второго порядка](#), и другие логики высших порядков.

На этом этапе развития, теперь уже [математической логики](#), пришло осознание того, что *количество возможных логических ответов соответствует основанию счетной системы*, применяемой для реализации той или иной математической логики в автоматических вычислениях.

Но если постепенное применение [двоичной системы](#) в логических вычислениях человек воспринял почти естественно еще со времен [Аристотеля](#), а [Дж.Буль](#) это только формально закрепил, то остальные счетные системы математической логики уже требовали специального обоснования для своего применения.

Двоичная система счисления позволяет реализовать двоичную логику с системой логических ответов, состоящей из ДА и НЕТ. Троичная позволяет реализовать логику с допустимыми ответами ДА, НЕТ и НЕ ЗНАЮ. Это мы понимаем.

А как будет выглядеть система ответа пятеричной логики?

[Ян Лукасевич](#) постарался найти ответы. Он работал в области логических проблем [индукции](#) и [причинности](#), и логических оснований [теории вероятностей](#). Лукасевич разработал первую систему [многозначной логики](#) — [трёхзначную логику высказываний](#) (1920). В качестве третьего логического значения высказывания было введено значение, выражаемое словами «вероятно», «нейтрально». О каждом высказывании в системе Лукасевича можно сказать: оно либо истинно (1), либо ложно (0), либо нейтрально (1/2). Это стало возможным благодаря тому, что Лукасевич одним из первых, независимо от русского логика [Н. А. Васильева](#), выдвинул тезис о возможности построения логических исчислений, в которых не действует принцип непротиворечивости. На основании трёхзначной логики Лукасевич построил [систему модальной логики](#), в которой наряду с исследованием логических операций над [ассерторическими](#) и [аподиктическими суждениями](#) (утверждениями и отрицаниями) исследуются так называемые [модальные высказывания](#) (сильные и слабые утверждения и отрицания).

В 1954 году Лукасевич разработал четырёхзначную систему логики, а затем — бесконечнозначные (n-значные) логические системы, в которых множество истинностных значений счётно-бесконечно или имеет [мощность континуума](#) (множества). В качестве истинностных значений выступают [рациональные числа](#) из отрезка (0, 1). Моделями бесконечнозначных логик Лукасевича являются им же разработанные алгебры.

Кстати говоря, [алгебра логики](#) помогла выйти на [формальную систему](#) и [рекурсивное определение](#) ... где все же, оказывается, ... что любые многозначные системы логических ответов фактически можно свести к *четной системе* двоичной логики и к *нечетной системе* ... унарной и троичной системы ответов. Это было понятно и раньше, но с доказательством все стало еще понятнее...

Что такое унарная логика? Это такая логика, которая имеет только один возможный ответ. Какой? С этим придется разбираться отдельно...[3]

Машинная логика.

Как уже не раз говорилось, *любая математическая логика имеет практический смысл только при наличии техники, которая в состоянии реализовать эту логику*.

Как мы выяснили ранее [9, 10], за весь период развития вычислительной техники все основные разработки в области математической логики были сделаны в привязке к двоичной логике, и только это понимание логики стало «единственно правильным». Двоичная математическая логика стала [классической логикой](#).

Сегодня также ничего не изменилось. Все разработки новых архитектур и принципов построения электронных вычислительных машин, в том числе и матричных логических состояний, идут в направлении развития двоичной логики.

И еще стало понятно, что никакие дробные показатели многозначной логики для определения того или иного логического состояния электронных элементов в реальной машине работать не могут.

Если говорить об электронной вычислительной машине, то в современной импульсной технике импульс не может иметь дробного значения. Он или есть, или – нет. Чтобы выходить на определение других параметров импульса, нужны соответствующие [эталон](#)ы ...

И потому, троичная логика для идентификации третьего логического состояния требует или введения второй полярности импульсов, или четкого пространственного разделения точек появления противоположностей логического состояния.

Понятно, что это уже совсем иной способ построения и электронных логических схем, и программных средств для управления ими. [Троичные вычислительные машины](#) разработки [Н.П.Брусенцова](#) это и показали. Его [троичная логика в электронном исполнении](#) имеет логические состояния: (-1), 0, (+1), что не соответствует значениям Я.Лукаевича (0, 1/2, 1). Но в машинном варианте троичной логики пока только такой способ её реализации себя и оправдал.

Вспомним и то, что возможность применения [n - ичных триггеров](#) на несколько устойчивых состояний, а не только на два, как в двоичной логике, была заявлена и в работах [А.П.Стахова](#). Но его разработки в области счетных систем на основе числа Ф для компьютерных вычислений носят чисто математический и технический характер. К развитию математической логики они отношения не имеют. Это подтверждают и результаты их применения. Хотя, если учесть время появления этих разработок, то их техническая новизна несомненна.

Программирование.

С другой стороны, очень интенсивно развивается другая сторона логики символа - [программирование](#). В целом, о программировании мы уже рассказывали в [7]. Сейчас нас более интересуют машинные аспекты.

Здесь мы видим явный прогресс в развитии [программируемых логических интегральных схем](#) (ПЛИС). Некоторые производители ПЛИС предлагают [программные процессоры](#) для своих ПЛИС, которые могут быть модифицированы под конкретную задачу, а затем встроены в ПЛИС. Более подробно о ПЛИС в [обзоре архитектур программируемых логических интегральных схем](#).

Альтернативой ПЛИС являются: [программируемые логические контроллеры](#) (ПЛК), [базовые матричные кристаллы](#) (БМК), требующие заводского производственного процесса для программирования; [ASIC](#) — специализированные заказные [большие интегральные схемы](#) (БИС), которые при мелкосерийном и единичном производстве существенно дороже; специализированные [компьютеры](#), [процессоры](#) (например, [цифровой сигнальный процессор](#)) или [микроконтроллеры](#), которые из-за программного способа реализации алгоритмов в работе медленнее ПЛИС.

И наконец, давно реализуемое направление [Scada](#) (от англ. *supervisory control and data acquisition*, *диспетчерское управление и сбор данных*) — [программный пакет](#), предназначенный для разработки или обеспечения работы в реальном времени систем сбора, обработки, отображения и архивирования информации об объекте мониторинга или управления:

Термин «SCADA» имеет двоякое толкование. Наиболее широко распространено понимание SCADA как приложения^[2], то есть программного комплекса, обеспечивающего выполнение указанных функций, а также инструментальных средств для разработки этого программного обеспечения. Однако, часто под SCADA-системой подразумевают программно-аппаратный комплекс. Подобное понимание термина SCADA более характерно для раздела [телеметрия](#).

Значение термина SCADA претерпело изменения вместе с развитием технологий автоматизации и управления технологическими процессами. В 80-е годы под SCADA-системами чаще понимали программно-аппаратные комплексы сбора данных реального времени. С 90-х годов термин SCADA больше используется для обозначения только программной части [человеко-машинного интерфейса](#) АСУ ТП.

Как мы видим, общее направление идет в сторону развития систем автоматического управления. Правда, под контролем человека. Чисто автоматические системы имеют ограниченное применение. По разным причинам.

И это часто уже не связывается с надежностью работы электронной техники, тут прогресс налицо. Чаще существенное влияние оказывает отсутствие адекватной самостоятельности машины в принятии управляющего решения.

Нам осталось посмотреть последние страницы этой истории.

Практическую реализацию логики символа.

Смотрим – [игра «Жизнь»](#):

Джон Конвей заинтересовался проблемой, предложенной в 1940-х годах известным математиком [Джоном фон Нейманом](#), который пытался создать гипотетическую машину, которая может воспроизводить сама себя. Джону фон Нейману удалось создать математическую модель такой машины с очень сложными правилами. Конвей попытался упростить идеи, предложенные Нейманом, и в конце концов ему удалось создать правила, которые стали правилами игры «Жизнь». Впервые описание этой игры было опубликовано в [октябрьском \(1970 год\) выпуске журнала *Scientific American*](#), в рубрике «Математические игры» [Мартина Гарднера \(Martin Gardner\)](#).

... И хотя игра состоит всего из двух простейших правил, она уже почти сорок лет привлекает пристальное внимание учёных. Игра «Жизнь» оказала определённое влияние на развитие многих разделов математики, информатики и физики^[1]:

- [Теория автоматов](#)
- [Теория алгоритмов](#)
- [Теория игр](#)
- [Алгебра и теория чисел](#)
- [Теория вероятностей](#)
- [Комбинаторика и теория графов](#)
- [Фрактальная геометрия](#)
- [Вычислительная математика](#)

Здесь вроде бы надо восхищаться, а мне грустно. Но, продолжим.

Проект игра «Жизнь» входит в направление программирования с общим названием «Искусственная жизнь»:

Искусственная жизнь (англ. *a-life*, от *artificial life*) — изучение [жизни](#), живых систем и их [эволюции](#) при помощи созданных человеком [моделей](#) и устройств. Данная область науки изучает механизм процессов, присущих всем живым системам, невзирая на их природу. Хотя этот термин чаще всего применяется к компьютерному моделированию жизненных процессов, он также подходит и к жизни в пробирке (англ. *wet alife*), изучению искусственно созданных [белков](#) и других молекул. Для простоты эта статья описывает компьютерную жизнь.

Технологии

- [Клеточные автоматы](#) — часто используются для моделирования жизни, особенно из-за лёгкости масштабирования и параллелизации. Клеточные автоматы и искусственная жизнь исторически близко связаны.
- [Нейронные сети](#) — иногда используются для моделирования интеллекта агентов. Хотя это традиционно технология, более близкая к созданию [искусственного интеллекта](#), нейронные сети могут быть полезны для моделирования динамики популяций или высокоразвитых самообучающихся организмов. Симбиоз между обучением и эволюцией центральная задача теорий о развитии инстинктов высших организмов, как например в [эффекте Болдуина](#).

Критика [искусственной жизни](#):

В истории искусственной жизни было довольно много споров и противоречий. [Джон Мейнард Смит](#) в 1995 году критиковал некоторые работы по искусственной жизни, называя их «наукой без фактов» (англ. *fact-free science*). Однако недавние публикации^[1] по искусственной жизни в крупных научных журналах, таких как [Science](#) и [Nature](#) свидетельствуют о том, что технологии, используемые для моделирования искусственной жизни, признаются научным сообществом, по крайней мере, для изучения [эволюции](#).

Процесс самовоспроизведения, о возможности реализации которого мечтали фон Нейман и Норберт Винер [10], был ими же переведен в разряд чисто математической задачи. Задача быстро была доведена до модели.

И вот как, оказывается, это было:

История

Станислав Улам, работая в Лос-Аламосской национальной лаборатории в 1940-е годы, изучал рост кристаллов, используя простую решёточную модель. В это же время Джон фон Нейман, коллега Улама, работал над проблемой самовоспроизводящихся систем. Первоначальная концепция фон Неймана основывалась на идее робота, собирающего другого робота. Такая модель известна как кинематическая. Разработав эту модель, фон Нейман осознал сложность создания самовоспроизводящегося робота и, в частности, обеспечения необходимого "запаса частей", из которого должен строиться робот. Улам предложил фон Нейману использовать более абстрактную математическую модель, подобную той, что Улам использовал для изучения роста кристаллов. Таким образом возникла первая клеточно-автоматная система. Подобно решётке Улама, клеточный автомат фон Неймана двухмерный, а самовоспроизводящийся робот описан алгоритмически. Результатом явился универсальный конструктор, работающий "внутри" клеточного автомата с окрестностью, включающей непосредственно прилегающие ячейки, и имеющего 29 состояний. Фон Нейман доказал, что для такой модели существует паттерн, который будет бесконечно копировать самого себя.

Также в 1940-е годы, Норберт Винер и Артуро Розенблют разработали клеточно-автоматную модель возбудимой среды. Целью было математическое описание распространения импульса в сердечных нервных узлах. Их оригинальная работа продолжает цитироваться в современных исследованиях по аритмии и возбудимым средам.

В 1960-е годы клеточные автоматы изучались как частный тип динамических систем, и впервые была установлена их связь с областью символической динамики. В 1969 году Г.А.Хедланд провёл обзор результатов, полученных в этом направлении. Наиболее значимым результатом явилось описание набора правил клеточного автомата как множества непрерывных эндоморфизмов в сдвиговом пространстве.

В 1970-е получила известность двухмерная клеточно-автоматная модель с двумя состояниями, известная как игра "Жизнь". Изобретенная Джоном Конвеем популяризованная Мартинем Гарднером, она использует следующие правила: если клетка имеет двух "живых" соседей, она остаётся в прежнем состоянии. Если клетка имеет трёх "живых" соседей, она переходит в "живое" состояние. В остальных случаях клетка "умирает". Несмотря на свою простоту, система проявляет огромное разнообразие поведения, колеблясь между очевидным хаосом и порядком. Одним из феноменов игры "Жизнь" являются глайдеры - сочетания клеток, движущиеся по сетке как единое целое. Возможно построить автомат, в котором глайдеры будут выполнять некоторые вычисления, и впоследствии было показано, что игра "Жизнь" может эмулировать универсальную машину Тьюринга.

В 1969 году немецкий инженер Конрад Цузе опубликовал книгу "Вычислимый космос", где выдвинул предположение, что физические законы дискретны по своей природе, и что вся Вселенная является гигантским клеточным автоматом. Это была первая книга из области, называемой сейчас цифровой физикой.

В 1983 Стивен Вольфрам опубликовал первую из серии статей, исследующих очень простой, но до сих пор неизученный класс клеточных автоматов, называемых элементарными клеточными автоматами. Неожиданная сложность поведения этих простых автоматов привела Вольфрама к предположению, что сложность естественных систем обусловлена сходным механизмом. Кроме того, в течение этого периода Вольфрам формулирует концепцию истинной случайности и вычислительной неприводимости, и выдвигает предположение, что Правило 110 (англ.)русск может быть универсальным - факт, доказанный в 1990 году ассистентом Вольфрама Мэтью Куком.

В 2002 году Вольфрам публикует 1280-страничный текст "Новый тип науки" (A New Kind of Science), где широко аргументирует, что достижения в области клеточных автоматов не являются изолированными, но весьма устойчивы и имеют большое значение для всех областей науки.

11-го ноября 2002 года Пауль Чепмен (Paul Charman) построил образец Жизни, который является РММ (Регистровой Машиной Минского). Фактически РММ эквивалентна машине Тьюринга. Первая версия образца была большой (268,096 живых ячеек на площади 4,558 x 21,469 клеток) и медленной (20 поколений/сек при использовании Life32 Иогана Бонтеса (Johan Bontes) на 400 MHz AMD K6-II). Таким образом, в игре Жизнь можно выполнить любой алгоритм, который можно реализовать на современном компьютере.

Мы опять загнали в тупик хорошую идею. Сложнейшую логическую задачу «Жизнь» с функцией самовоспроизведения мы постепенно свели к простейшей математической модели, которая и стала основой дальнейшего развития ... математики. Получается, что математики изменили понимание понятия «искусственная жизнь», закрепив его за программой и её математическим обеспечением.

Точно так же была математизирована и эволюция. Постепенно она была доведена до простейших задач математики и программирования. Например, как «генетический алгоритм»:

Генетический алгоритм (англ. *genetic algorithm*) — это эвристический алгоритм поиска, используемый для решения задач оптимизации и моделирования путём случайного подбора, комбинирования и вариации искоемых параметров с использованием механизмов, напоминающих биологическую эволюцию. Является разновидностью эволюционных вычислений, с помощью которых решаются оптимизационные задачи с использованием методов естественной эволюции, таких как наследование, мутации, отбор и кроссинговер. Отличительной особенностью генетического алгоритма является акцент на использование оператора «скрещивания», который производит операцию рекомбинации решений-кандидатов, роль которой аналогична роли скрещивания в живой природе.

О генетических алгоритмах можно посмотреть [здесь](#).

И, наконец, искусственный интеллект:

Искусственный интеллект (ИИ, англ. *Artificial intelligence, AI*) — наука и технология создания интеллектуальных машин, особенно интеллектуальных компьютерных программ. ИИ связан со сходной задачей использования компьютеров для понимания человеческого интеллекта, но не обязательно ограничивается биологически правдоподобными методами.^[1]

... Пояняя своё определение, Джон Маккарти указывает: «Проблема состоит в том, что пока мы не можем в целом определить, какие вычислительные процедуры мы хотим называть интеллектуальными. Мы понимаем некоторые механизмы интеллекта и не понимаем остальные. Поэтому под интеллектом в пределах этой науки понимается только вычислительная составляющая способности достигать целей в мире»^[1].

В то же время существует и точка зрения, согласно которой интеллект может быть только биологическим феноменом^[2].

Определить не можем, но точно знаем, что задача ИИ решается вычислительными методами. Мы не знаем, что можно отнести к интеллекту, но уже определяем методы его реализации. При этом мы не знаем, чем искусственный интеллект отличается от биологического, нашего. И отличается ли... Не в философском понимании, а в самом конкретном техническом понимании машинной логики. На уровне простейших машинных операций.

И, тем не менее, часто читаем, что «...в программе применены элементы искусственного интеллекта...». Хотя, объяснить, что это такое – элементы или признаки ИИ, вряд ли кто сможет.

Вот наиболее известное направление ... «с признаками ИИ»:

Агентские программы (программы-агенты, *agent's programs*) — группа прикладных программ, обладающих признаками искусственного интеллекта и способных после получения задания работать самостоятельно без участия пользователя. Агентская программа может задавать пользователю вопрос, чтобы определить свои последующие действия. Существуют агентские программы, способные адаптироваться к изменяющимся условиям работы, существенным образом меняя алгоритм своей работы и характеристики, а также взаимодействовать с другими агентскими программами. Теорию «агентов» начал разрабатывать и продвигать в начале 1990-х годов Алан Кей. Агентские программы используются в сервисах и приложениях преимущественно в Интернете, включая поисковые системы (особенно в бизнесе), почтовых сервисах, системах идентификации пользователей, экранирующих шлюзах, защитных экранах

Алан Кей, это имя мы постараемся запомнить. Но, программные агенты сегодня связываются и с разработками других авторов:

Концепцию агента можно проследить начиная с модели акторов Карла Хьюитта: «автономный, интерактивный и одновременно исполняющий несколько функций объект, обладающий внутренним состоянием и информационным обменом».^[5]

...«Навигатор по знаниям», созданный в 1987 г. Джоном Скалли, представлял собой видеосвязь между конечным пользователем и агентом. Будучи первым в своём роде, он послужил основой для опытной серии неудачных нисходящих реализаций, вместо того, чтобы шаг за шагом подходить к восходящим методам. Диапазон типов агентов в настоящее время (с 1990 г.) довольно широк: WWW, поисковые системы и т. д.

Об программных агентах чуть подробнее:

Термин «агент» описывает программную абстракцию, идею или концепцию, подобно таким терминам ООП как метод, функция или объект. Концепция агента обеспечивает удобный и мощный способ описания сложной программной сущности, которая способна действовать с определённой степенью автономности с целью выполнения задач от имени пользователя. Но в отличие от объектов, которые определяются в терминах *методов* и *атрибутов*, агент определяется посредством описания его поведения.^[31]

Разные авторы предлагают разные определения агента, обычно при этом в него включают такие понятия, как:

- *живучесть* (код не исполняется по требованию, а работает непрерывно и решает сам, когда он должен совершить какие-либо действия);
- *автономность* (агент имеет способность выбора задач, приоритетов, целенаправленного поведения и принятия решений без участия человека);
- *социальное поведение* (агенты имеют возможность привлекать другие компоненты посредством определённых связей и координации действий, они могут совместно работать над выполнением задач);
- *реактивность* (агенты воспринимают контекст, в котором они работают, и реагируют на него должным образом).

Здесь можно закончить обзор реализации машинной логики.

С одной стороны, мы видим большой спектр направлений применения алгоритмов для решения задач в автоматической режиме. Программных и вычислительных.

Это так. А с другой?

Все направления ориентированы на компьютер и программирование. Пока мы пытаемся реализовать программный интеллект.

Конечно, всё что показано, нельзя расценивать как что-то плохое или хорошее. Это уже история понимания логики на разных этапах. Переходы от слова к понятию, затем к символу. К его математическому пониманию. Потом к алгоритмическому. Для машинной реализации логического действия.

Вроде бы всё закономерно и объективно понятно. Так должно было быть и так сделано. Потому, что весь ход исторического понимания вел именно к такому, математическому пониманию машинной логики.

Да, сегодня мы уже ясно видим, что двоичная логика давно не отвечает нашим требованиям адекватности получаемых логических решений. И потому при использовании этой логики никаких принципиально новых качественных прорывов в построении систем искусственного интеллекта не ожидается. Для этого нужна другая логика...

Эту ситуацию могла изменить кибернетика, но когда-то она слишком быстро перешла от самостоятельного пути развития к сопровождению вычислительных методов, в том числе и в системах управления. Что и привело эту науку к её сегодняшнему состоянию стагнации.

Странности понимания.

Смещение понятий, ... такое в науке происходило неоднократно.

Вот, например, как теперь говорят о теории автоматов:

Теория автоматов — раздел дискретной математики, изучающий абстрактные автоматы — вычислительные машины, представленные в виде математических моделей — и задачи, которые они могут решать.

Теория автоматов наиболее тесно связана с теорией алгоритмов: автомат преобразует дискретную информацию по шагам в дискретные моменты времени и формирует результат по шагам заданного алгоритма.

Типовые задачи

- **Построение и минимизация автоматов** — построение абстрактного автомата из заданного класса, решающего заданную задачу (принимающего заданный язык), возможно, с последующей минимизацией по числу состояний или числу переходов.
- **Синтез автоматов** — построение системы из заданных «элементарных автоматов», эквивалентную заданному автомату. Такой автомат называется структурным. Применяется, например, при синтезе цифровых электрических схем на заданной элементной базе.

Попутно читаем о теории алгоритмов:

Теория алгоритмов — наука, изучающая общие свойства и закономерности алгоритмов и разнообразные формальные модели их представления. К задачам теории алгоритмов относятся формальное доказательство алгоритмической неразрешимости задач, асимптотический анализ сложности алгоритмов, классификация алгоритмов в соответствии с классами сложности, разработка критериев сравнительной оценки качества алгоритмов и т. п.

Надо же, теория автоматов – раздел дискретной математики, а теория алгоритмов – отдельная наука. Действительно, удивительное – рядом...

Нет, ну действительно, странно же...

Математическая индукция, и даже трансфинитная индукция есть, а логической индукции на таком же уровне абстракции - нет. Или почти нет...

Читаем:

Индукция (лат. *inductio* — наведение) — процесс логического вывода на основе перехода от частного положения к общему. Индуктивное умозаключение связывает частные предпосылки с заключением не строго через законы логики, а скорее через некоторые фактические, психологические или математические представления.^[1]

Объективным основанием индуктивного умозаключения является всеобщая связь явлений в природе.

Различают полную индукцию — метод доказательства, при котором утверждение доказывается для конечного числа частных случаев, исчерпывающих все возможности, и неполную индукцию — наблюдения за отдельными частными случаями наводят на гипотезу, которая, конечно, нуждается в доказательстве. Также для доказательств используется метод математической индукции.

Но, не будем торопиться с выводами. Там же:

Неполная индукция не является доказательной с точки зрения формальной логики, может привести к ошибочным заключениям. Вместе с тем, неполная индукция является основным способом получения новых знаний. Доказательная сила неполной индукции ограничена, заключение носит вероятностный характер, требует приведения дополнительного доказательства.

Не больше и не меньше...

Не является, но ... является, хоть и требует...

В общем, мы поняли.

За границами классической математической логики почему-то остались такие основные логические понятия, как *цель, условия, действие, движение, границы и переходы*, [6] и т.д.

Посмотрим далее...

Кажется, в основных определениях математической логики мы несколько свободно относимся к понятию *функции*:

Функция (отображение, оператор, преобразование) — математическое понятие, отражающее связь между элементами множеств. Можно сказать, что функция — это «закон», по которому каждому элементу одного множества (называемому областью определения) ставится в соответствие некоторый элемент другого множества (называемого областью значений).

Математическое понятие функции выражает интуитивное представление о том, как одна величина полностью определяет значение другой величины. Так значение переменной x однозначно определяет значение выражения x^2 , а значение месяца однозначно определяет значение следующего за ним месяца, также любому человеку можно сопоставить другого человека — его отца. Аналогично, некоторый задуманный заранее алгоритм по варьируемым входным данным выдаёт определённые выходные данные.

...К концу XIX века понятие функции переросло рамки числовых систем. Первыми это сделали векторные функции, вскоре Фреге ввёл логические функции (1879), а после появления теории множеств Дедекин (1887) и Пеано (1911) сформулировали современное универсальное определение.

В математической логике вместо *операций* мы уже давно перешли к логическим *функциям*, не особенно задумываясь о *результате*. А надо бы:

Результат — заключительное последствие последовательности действий или событий, выраженных качественно или количественно. Возможные результаты включают преимущество, неудобство, выгоду, потерю, ценность и победу.

Цель описывает желаемый, но возможно еще не достигнутый результат.

Некоторые типы результата:

- в математике — окончательное значение вычисления (например, арифметической операции), функции или статистического выражения
- в компьютерных науках — значение, возвращаемое функцией, состояние системы или список записей, соответствующих запросу (например поиск в Интернете). *Тип результата* — тип возвращенных функцией данных.
- в науке — итог эксперимента (смотри Нулевая гипотеза)
- в теории вероятности — элементарное событие

Прочитали? А поняли?

В этом определении ярко выражено домысливание того, что нам хочется получить. Из того, что у нас есть фактически. Но мы забываем, что например, функция $y=x^2$ и выражение $4=2^2$, это *разные* математические выражения.

Функция имеет только одну форму записи: $y=x^2$, и никакого другого результата дать не может. Результат дает вычисление математического выражения, например, $2^2 = \dots$, полученного подстановкой конкретного значения переменной в общую формулу функции.

Логическая функция $a \wedge b$ также не имеет результата. Это алгоритм. Для получения результата нужно установить конкретные значения переменных и только тогда вычислением можно получить результат.

В программировании функцией названа *подпрограмма*, которая требует задания исходных данных для проведения вычисления по формуле функции. Она эти данные получает, вычисляет, и выдает (возвращает) результат.

То, что написано в определении результата для компьютерных наук не соответствует фактическим действиям. Мы опять принимаем желаемое за действительное, домысливаем результат.

В математике и математической логике - *результат, это зафиксированное значение вычисления, полученное при выполнении счетной операции*. Зафиксированное, вот это мы постоянно забываем.

В математике для этого введен специальный знак ... равенства. И потому мы пишем $2+2=4$, понимая, что мы посчитали и получили число 4, что это и есть результат вычисления. Мы его отдельно записали, через знак равенства.

Действие или операция закончена, когда есть результат.

Правда, насколько знак равенства соответствует своему предназначению, фиксировать получение результата, это тоже еще большой вопрос. Мы это уже разбирали [13].

Но пока хоть так...

Любая логическая функция, например, $a \wedge b$ в том виде, как мы записали, в компьютере вообще отсутствует как таковая, её там просто нет. В таком виде логическая функция есть только на бумаге. Больше нигде. В компьютере это электронная схема *мгновенного преобразования* сигналов входов в выходной сигнал. *Без фиксации результата*, между прочим.

Принятая процедура фиксации результата, как в математике через знак равенства, в компьютере есть, но в другом месте. Это тоже схема - триггер или ключ. Там фиксируется состояние результата вычисления конкретных значений функции. Но, для компьютера это отдельная операция, не имеющая отношения к логическим функциям.

А мы, получается, опять домысливаем то, что нам хочется, и упорно не видим то, что есть фактически. Мы забываем, что работаем не с математическими или логическими функциями, а с их *техническими эквивалентами*.

Автоматическая машинная логика не может рассматривать функцию и вычисление по её формуле так, как это рассматриваем мы, с домысливанием последующих вычислительных действий.

Для машины это разные задачи. Одна задача, это определение функции, вторая - получение результата при конкретике вычисления значений функции по данным переменных. В компьютере это так и есть. Только почему мы об этом не помним, непонятно.

Математики недостаточно...

И, к сожалению, можно только снова констатировать, что формирование определения понятий с ощутимым сдвигом в сторону математики привело к изменению понимания сути этих понятий. Например:

- *Прикладной расчет для математического анализа реальных электронных схем стал основным пониманием [теории автоматов](#).*
- *Моделирование перестало быть отдельным способом реализации вариантного решения, это стало частью математической [теории алгоритмов](#).*
- *Математический алгоритм вдруг стал ассоциироваться с реальными процессами [моделирования эволюции](#) и [искусственной жизни](#).*
- *Логические методы поиска решений стали математическими, а проблема нахождения решений получила название «[Искусственный Интеллект](#)».*

Почему эти сложные проблемы нашли только математические решения?

Не знаю.

Кстати сказать, очень правильно оценивает настоящее состояние д.ф.-м.н., профессор [А.А.Жданов](#). Вот некоторые его ответы [20]:

Александр Аркадьевич, как вы оцениваете научные достижения в области моделирования нервных систем сегодня? Как скоро искусственный интеллект сможет стать полноценной заменой человеческому?

– Тут имеет место некий обман в постановке задачи. Моделировать мозг особенно никто не хочет. На самом деле, стараются сделать искусственный интеллект так, чтобы он, боже упаси, не походил на человеческий. Нам ведь надо, чтобы он не на себя работал, а на нас, и лучше нас рассчитывал всякие сложные процессы вроде траектории полёта снарядов, решал разные практические задачи. Эта машина по своим свойствам не должна походить на естественный животный мир: не должна хотеть ни спать, ни есть. Поэтому к задачам моделирования нервных систем и понимания, как работает человеческий мозг, «искусственный интеллект» не имеет никакого отношения. Чего хотят люди, создавая механизмы? Облегчить себе жизнь, создать механизированного раба, который будет за нас выполнять работу: грести на галерах, воевать, выращивать хлопок, рубить тростник. Чего хотят люди, создавая искусственный интеллект? Получить интеллектуального раба, который вместо нас будет билеты продавать, пылесосить, водить электрички и т.д.

Я бы хотел сделать машину, похожую хотя бы на жука или на дождевого червя. Это нужно для того, чтобы понимать, как работает мозг. Это первое. Второе: было бы полезно создать машины со своей волей, работающие в областях, где они не были бы нам конкурентами – в шахтах, на других планетах. Причём работали бы они не как рабы, а как живые организмы, жили бы в своё удовольствие, но попутно делали что-то для человечества, меняли окружающую среду, добывали полезные ископаемые, выполняли функции водителей, которые бы не уставали, привыкали к маршруту, частично походили бы на живых, но всё-таки решали не свои, а наши задачи. Интересно было бы отправить такие искусственные организмы жить на другие планеты, откуда бы они собирали и поставляли нам информацию о Земле и Космосе.

Лучше и не скажешь. Всё сформулировано точно.

Но, далее еще интереснее [20]:

... В природе ничего этого нет, это всё прекрасно работало тысячу и миллион лет назад без всяких уравнений, но вот людям надо это описать, и они придумывают для этого инструменты. Придумали они математику, некоторое количество людей в неё посвящено и что-то там понимают. Постепенно эти яйцеголовые математику развивают, у них получается описать, как камень падает, как он летит по параболе... А математика – это инструмент, некая система образов, которая позволяет описывать и рассчитывать наперёд эти процессы. Математические знания сильно развились благодаря некой маленькой группе людей, которые стали строить машины, системы управления, запускать ракеты.

О чём это говорит? О том, что этот математический, физический инструмент оказался удобным, эффективным и позволяет какие-то задачи решать. Но аппетиты растут. Захотелось людям построить такие системы управления, которые похожи на те, что есть в природе. И тут оказалось, что математики недостаточно, надо учитывать огромное количество факторов, которое этими методами практически невозможно учесть.

Была обоснованная эйфория по поводу того, что удалось рассчитать полёт спутника в безвоздушном пространстве. Но спутник – это математический объект в безвоздушном пространстве, а вот рассчитать движение консервной банки, которая болтается у вас на гвозде возле сарая, не удастся, потому что там надо учесть миллион факторов. И вот сейчас приходит осознание того, что математика не может описать все существующие в природе процессы. Встаёт жуткий вопрос: а как же какая-то там мышшь величиной с палец умеет, условно говоря, оценить движение банки, а мы со всеми нашими компьютерами, математиками и академиками не можем?

А ответ в том, что эти нервные системы работают совсем по другим принципам. Там нет никаких уравнений, производных, квадратов...

Работает это всё, скорее, методом проб и ошибок. У природы очень много времени, она может себе позволить пробовать и ошибаться очень долго. Конечно, метода «проб и ошибок» в чистом виде не существует – он всегда как-то организован, связан ограничениями, механизмами, которые что-то аппроксимируют, направляя этот процесс и т.д.

Это говорит один из ведущих специалистов в области ИИ. И, как мне кажется, стоит к этому прислушаться. Не надо переоценивать математику, она не всесильна.

Но мы почему-то уже не одну сотню лет продолжаем свято верить в математику, как единственную панацею, ведущую к пониманию всей сути Живого.

Хотя, сегодня похоже, что это уже не совсем так. Математическая логика не смогла продвинуться далеко вперед более по причине ограниченности средств математики, чем от недостаточности понимания целей и задач, требующих решения.

Сегодня мы уже знаем, что математика, это только одна из частей логики, отдельная, ограниченная постулатами, логическая система проведения математических действий. В логике таких систем много. Например, наша речь. Одна из таких логических систем – система обоснования на основе доказательств, применяемая, в том числе и в философии.

Н.П.Брусенцов, оценивая мою счетную логику, заметил в своем письме, что «...логика в принципе не может быть "счетной" - она теория качеств, а не арифметика».

Прошло несколько лет, прежде чем я по достоинству оценил сказанное. И поставил «качество» первой весовой частью в системе единиц логики [5].

Но, логика, как «теория качеств» должна объяснить и понятие «значимости» информации [4]. Вполне конкретно, на уровне простой формулы математической абстракции, а не объемного объяснения. Вот об этом с практической стороны для человеческой памяти:

«Рецепторы воспринимают весьма большой объем информации, поступающей от среды. Если бы вся эта информация поступала в память, последняя очень быстро переполнилась бы и было бы практически невозможно в ней разобраться. Поэтому необходимо выделить наиболее существенную, значимую (или могущую быть значимой для объекта) информацию и только её передавать для фиксации в память.» [19]

Похоже, что логика действительно нуждается в новой основе для формирования автоматической логики. Математики недостаточно...

Дальнейшее развитие логики.

Куда идет развитие логики?

Если продолжить ту линию развития понимания, которую мы видим сейчас, то логика идет к механизмам автоматического формирования системы понятий и аксиом для построения логики саморазвития. Основное направление развития логики идет в направлении понимания глубинных логических действий на уровне клетки. Именно туда ведет нас путь познания. Там заложены основы нашей логики. Не только человека, а всего Живого.

Наверное, уже не надо углубляться в клеточные процессы. Мы и делали это уже не раз [11-17], и вряд ли это сейчас необходимо. Уже ни у кого не вызывает сомнений то, что основа информационных технологий клетки, ДНК и РНК, при высоком уровне точной повторяемости информации в то же время постоянно видоизменяется.

По этой причине мы имеем то, что называется [эволюцией](#).

Этим занимается эволюционная биология:

Эволюционная биология — раздел биологии, изучающий происхождение видов от общих предков, наследственность и изменчивость их признаков, размножение и разнообразие форм в ходе эволюционного развития. Развитие отдельных видов обычно рассматривается в контексте глобальных преобразований флор и фаун, как компонентов биосферы. Эволюционная биология начала оформляться в качестве раздела биологии с широким признанием идей об изменчивости видов во второй половине XIX века.

С другой стороны, мы уже знаем, что далеко не все изменения ДНК и РНК ведут к изменениям клетки немедленно. Множество мутаций почему-то не происходит, хотя изменения ДНК уже произошли. Эти измененные информационные массивы как-то сохраняются, они и не участвуют в клеточных процессах, но и не уничтожаются, как ошибочное копирование. Ошибки сохраняются и используются. Отдельными порциями, разделенными во времени. Клетка как бы проверяет их на работоспособность, отдельно от всего остального производственного процесса. Немного об этом, например, в [18].

Этот момент хорошо согласуется с нейтральной теорией молекулярной эволюции:

Нейтральная теория молекулярной эволюции — теория, утверждающая, что подавляющее число мутаций на молекулярном уровне носит нейтральный по отношению к естественному отбору характер. Как следствие, значительная часть внутривидовой изменчивости (особенно в малых популяциях) объясняется не действием отбора, а случайным дрейфом мутантных аллелей, которые нейтральны или почти нейтральны.

Теория была разработана Мото Кимурой в конце 1960-х годов. Теория нейтральной эволюции хорошо согласуется с фактом постоянной скорости закрепления мутаций на молекулярном уровне, что позволяет, к примеру, оценивать время расхождения видов.

Теория нейтральной эволюции не оспаривает решающей роли естественного отбора в развитии жизни на Земле. Дискуссия ведётся касательно доли мутаций, имеющих приспособительное значение. Большинство биологов признаёт ряд результатов теории нейтральной эволюции, хотя и не разделяет некоторые сильные утверждения, первоначально высказанные М. Кимурой.

Зачем нам все это, мы же говорим о логике?

Это и есть логика. Правила работы с информацией.

Это, то самое «правильное мышление», которое диктует нам логическая машина клетки. На основе этих правил работает как логика клетки, так и человека.

В данном случае мы говорим о системе аксиом автоматической логики клетки, сформированных на основе опыта, зафиксированного в изменениях ДНК и РНК.

Изменения ДНК и РНК клетки имеют как целенаправленный, так и случайный характер. Но клетка не в состоянии отличить одни изменения от других. По этой причине все изменения проходят один и тот же сложный путь поэтапного закрепления их в структуре ДНК, этой долговременной системе памяти клетки.

Всё это говорит об одновременном участии в выработке логического решения, как случайных факторов, так и направленных действий [4]. Эту особенность работы нашей логики мы совершенно забываем при разработках любых математических логик.

В разрабатываемых логиках мы требуем непротиворечивости.

Хотя, мы уже давно знаем:

Гёдель был логиком и философом науки. Наиболее известное достижение Гёделя — это сформулированные и доказанные им теоремы о неполноте, опубликованные в 1931 году.

Одна из них гласит, что *любая эффективно аксиоматизируемая теория, в достаточно богатом языке, достаточном для определения натуральных чисел и операций сложения и умножения является неполной либо противоречивой.*

Это надо понимать так, что *любая логическая система всегда будет обладать какой-то неполнотой и противоречивостью.* Любая сложная логическая система. Без исключений.

Требование соблюдения внутренней непротиворечивости логической системы хоть и очень желательно, но практически невыполнимо. Видимо *вопрос не в противоречиях, а их согласовании при локальном использовании в доказательствах.*

Как мы понимаем, возможна только локальная непротиворечивость, так же как и локальная полнота обоснования. Из этого и будем исходить...

Для логики:

- **Неполнота** означает наличие высказываний, которые нельзя ни доказать, ни опровергнуть, исходя из аксиом этой теории.
- **Противоречивость** — возможность доказать любое высказывание: как истинное и как ложное.
- **Эффективная аксиоматизируемость** понимается как возможность алгоритмически решить, является ли данное утверждение аксиомой.

И попутно ... об аксиоме:

Аксиомы — это те положения теории, которые принимаются за исходные, причём вопрос об истинности решается либо в рамках других научных теорий, либо посредством интерпретации данной теории.^[1]

Аксиоматизация теории — явное указание конечного или счётного, рекурсивно перечислимого (как, например, в аксиоматике Пеано) набора аксиом и правил вывода. После того как даны названия изучаемым объектам и их основным отношениям, а также аксиомы, которым эти отношения должны подчиняться, всё дальнейшее изложение должно основываться исключительно лишь на этих аксиомах, не опираясь на обычное конкретное значение этих объектов и их отношений. Утверждения на основе аксиом называются теоремами. С формальной точки зрения, сами аксиомы также входят в число теорем.

Нам осталось только приложить данные положения к машинной логике и получить *локально непротиворечивую логическую систему локально полного обоснования целей, и действий для их достижения*. В автоматическом варианте исполнения этого набора действий.

Всё так. Нужна именно такая система автоматической логики, позволяющая решать целевые локальные логические задачи, в том числе и задачи управления, в реальных условиях *неполноты, противоречивости, недостаточности информации и ограниченного набора возможных действий*. И стараться достигать необходимого результата.

Ничего сверхъестественного. Нормальная цель развития логики.

Для машины в самый раз.

Екатеринбург
ноябрь 2012 г.

Литература.

1. Никитин А.В. На пути к Машинному Разуму. Круг третий. (Части 1,2) // «Академия Тринитаризма», М., Эл № 77-6567, публ.12887, 31.01.2006
<http://www.trinitas.ru/rus/doc/0023/001a/00230029.htm>
2. Никитин А.В. Эволюционный путь саморазвития искусственного интеллекта // «Академия Тринитаризма», М., Эл № 77-6567, публ.14738, 19.03.2008
<http://www.trinitas.ru/rus/doc/0016/001c/00161450.pdf>
3. Никитин А.В. Логическая система Само... http://andrejnikitin.narod.ru/logika_samo.htm
4. Никитин А.В., Логика автономных систем // «Академия Тринитаризма», М., Эл № 77-6567, публ.15858, 28.03.2010 <http://www.trinitas.ru/rus/doc/0016/001c/00161628.htm>
5. Никитин А.В. Логика автономных систем – 2. Машинная логика.
http://andrejnikitin.narod.ru/meshin_logic.htm
6. Никитин А.В., Логика управления клетки // «Академия Тринитаризма», М., Эл № 77-6567, публ.17037, 29.11.2011 <http://www.trinitas.ru/rus/doc/0016/001c/1905-nik.pdf>
7. Никитин А.В. Механистическое понимание логики // «Академия Тринитаризма», М., Эл № 77-6567, публ.17252, 24.01.2012 <http://www.trinitas.ru/rus/doc/0016/001c/1928-nik.pdf>

8. Никитин А.В., О логике и логической машине // «Академия Тринитаризма», М., Эл № 77-6567, публ.17459, 15.05.2012 <http://www.trinitas.ru/rus/doc/0016/001c/1956-nik.pdf>
9. Никитин А.В. Скептический взгляд на логику // «Академия Тринитаризма», М., Эл № 77-6567, публ.17501, 04.06.2012 <http://www.trinitas.ru/rus/doc/0016/001c/1965-nik.pdf>
10. Никитин А.В. Понимание логики. // «Академия Тринитаризма», М., Эл № 77-6567, публ.17625, 20.08.2012 <http://www.trinitas.ru/rus/doc/0016/001c/00161989.htm>
11. Никитин А.В. Работа рибосомы при трансляции белка.
<http://andrejnikitin.narod.ru/ribosoma.htm>
12. Никитин А.В. Проблемы понимания системы кодирования ДНК.
http://andrejnikitin.narod.ru/Problem_DNK.htm
13. Никитин А.В. Тожество и тавтология. <http://andrejnikitin.narod.ru/tavtalogia.pdf>
14. Никитин А.В. Считывание и обработка информации ДНК.
http://andrejnikitin.narod.ru/information_DNK1.htm
15. Никитин А.В. Триплеты в ДНК. <http://andrejnikitin.narod.ru/tripletDNK.htm>
16. Никитин А.В. Информация в ДНК, РНК и белках.
http://andrejnikitin.narod.ru/InformaciaDNK_RNK_belkah.htm
17. Никитин А.В. От «мира РНК» к Началу Жизни...
<http://andrejnikitin.narod.ru/otRNKkNachalu.htm>
18. С. Г. ИНГЕ-ВЕЧТОМОВ [Трансляция как способ существования живых систем, или в чем смысл "бессмысленных" кодонов.](#) Санкт-Петербургский государственный университет.
19. Гаазе-Рапопорт М.Г., Пospelов Д.А. От амебы до робота: модели поведения , М.: Наука, 1987, -286с. (djvu) <http://www.4tivo.com/education/6872-gaaze-rapoport-m.-g.-pospelov-d.-a.-ot.html>
20. Щукина Ника, Интеллектуальные рабы для человека
http://www.strf.ru/material.aspx?CatalogId=222&d_no=48050