

# Системы искусственного интеллекта

02.03.03 - Математическое обеспечение и администрирование информационных систем, направленность (профиль)  
- разработка и администрирование информационных систем

09.03.03 - Прикладная информатика, направленность (профиль) - прикладная информатика в экономике

<http://vikchas.ru>

## Тема 1. Введение в искусственный интеллект и машинное обучение Лекция 3 «Интеллектуальные агенты - продолжение»

**Часовских Виктор Петрович**

д-р техн. наук, профессор кафедры ШИиКМ

ФГБОУ ВО «Уральский государственный экономический  
университет»

Екатеринбург 2023

## Рефлексные, основанные на модели

Одним из наиболее эффективных способов организации работы в условиях частичной наблюдаемости, является отслеживаемость агентом той части мира, которая воспринимается им в данный момент времени. Это означает, что агент должен поддерживать внутреннее состояние, зависящее от истории актов восприятия и отражать некоторые из ненаблюдаемых аспектов текущего состояния. Для обновления внутренней информации о состоянии в программе агента используются знания двух видов. Во-первых, информация о том, как изменяется мир независимо от агента, а во-вторых, информация о том, как действия агента влияют на окружающий мир. На основе первого вида информации строится модель мира.



## Программа рефлексорного агента с поведением, основанным на модели

**function** MODEL-BASED-REFLEX-AGENT(*percept*) **returns** действие *action*

**persistent:** *state*, представление агента о текущем состоянии мира

*transition\_model*, описание, как следующее состояние мира зависит от его текущего состояния и действия

*sensor\_model*, описание, как текущее состояние мира отражается в восприятии агента

*rules*, множество правил "условие-действие"

*action*, последнее по времени действие; исходно не определено

**state** ← UPDATE-STATE(*state*, *action*, *percept*, *transition\_model*, *sensor\_model*)

**rule** ← RULE-MATCH(*state*, *rules*)

Рефлексорный агент с поведением, основанным на модели. Он отслеживает текущее состояние мира, используя внутреннюю модель, а затем выбирает действие таким же образом, как и простой рефлексорный агент.

## Действующие на основе цели

В реальной жизни для принятия решения не всегда достаточно информации из окружающей среды.

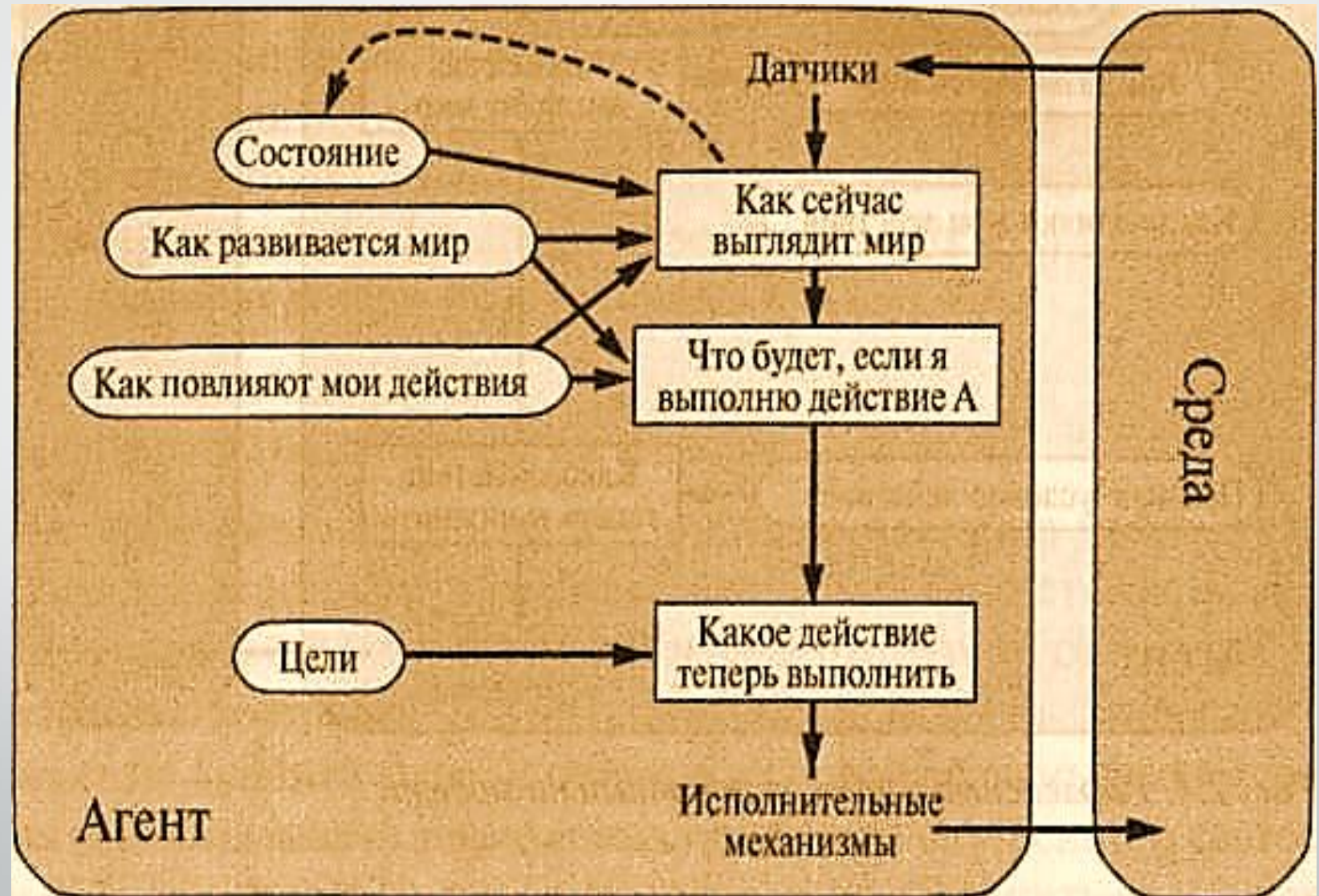
Так, например, человек подходит к перекрестку, у него есть на выбор три направления движения, выбор направления зависит от цели.

То есть, агенту требуется не только информация о мире, внутреннем состоянии, но и информация о цели, которая будет описывать желаемые ситуации.

Программа агента может комбинировать эти виды информации для выбора действий, которые позволят достичь цели.



# Действующие на основе цели



## Действующие на основе полезности

В действительности в большинстве вариантов среды для выработки высококачественного поведения одного лишь учета целей недостаточно. Цели позволяют провести лишь жесткое бинарное различие между состояниями «удовлетворенности» и «неудовлетворенности», тогда как более общие показатели производительности должны обеспечивать сравнение различных состояний мира в точном соответствии с тем, насколько удовлетворенным станет агент, если их удастся достичь.

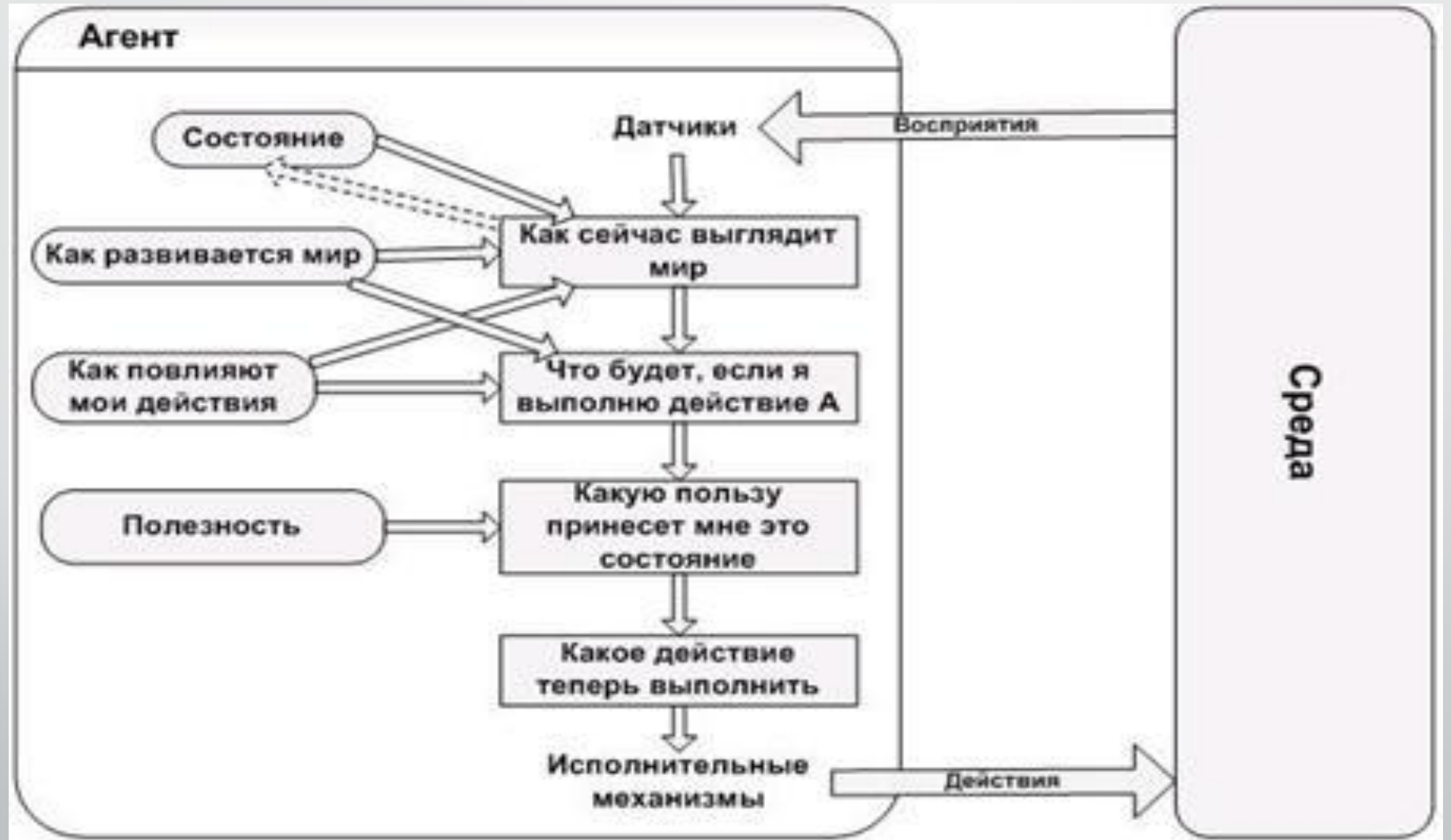
Функция полезности отображает состояние (или последовательность состояний) на вещественное число, которое обозначает соответствующую степень удовлетворенности агента.

Полная спецификация функции полезности обеспечивает возможность принимать рациональные решения в описанных ниже двух случаях, когда этого не позволяют сделать цели:

- если имеются конфликтующие цели, такие, что могут быть достигнуты только некоторые из них (например, или скорость, или безопасность), то функция полезности позволяет найти приемлемый компромисс;
- если имеется несколько целей, к которым может стремиться агент, но ни одна из них не может быть достигнута со всей определенностью, то функция полезности предоставляет удобный способ взвешенной оценки вероятности успеха с учетом важности целей.

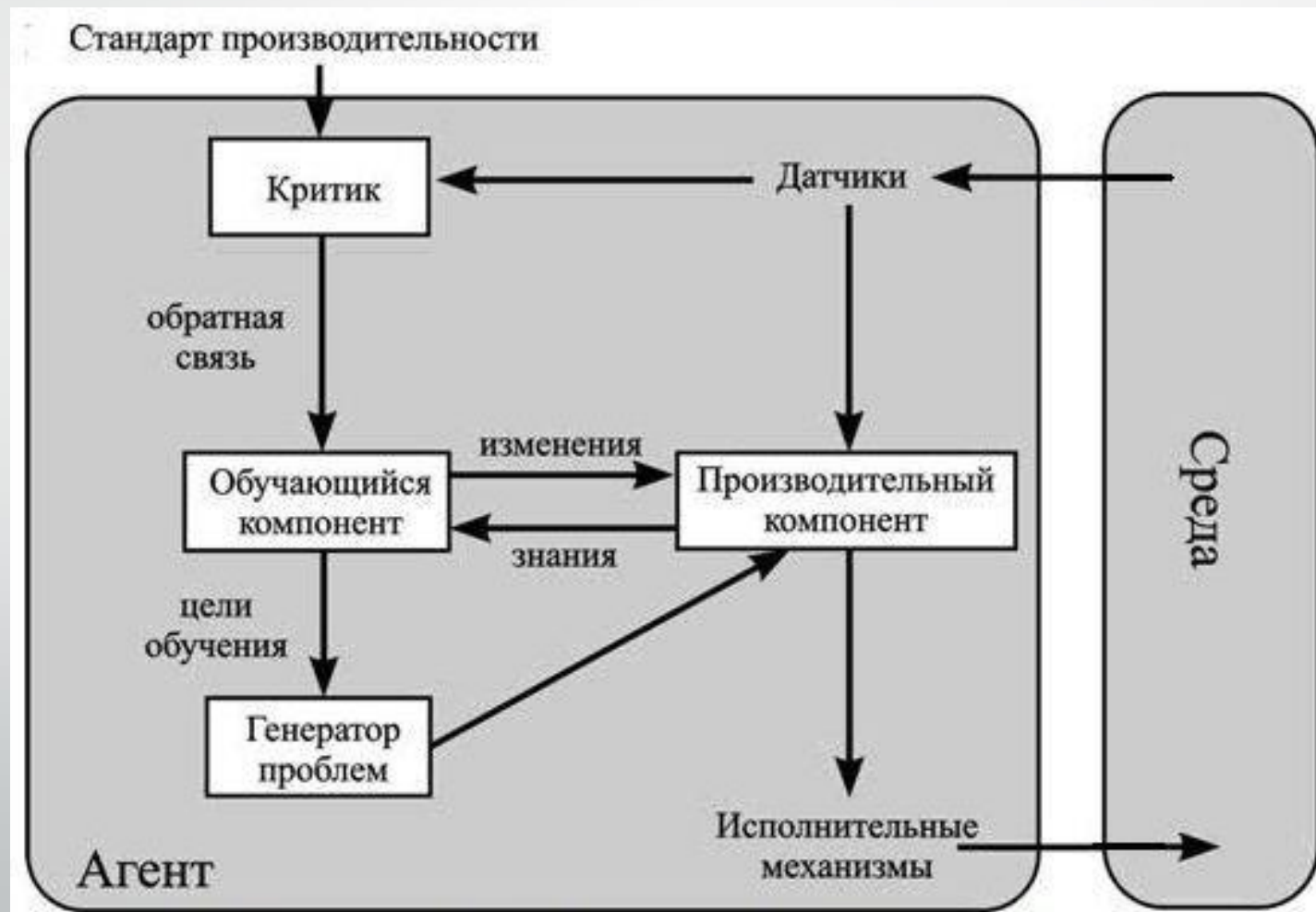


# Агенты, основанные на полезности



## Обучающиеся агенты

Рассмотренные виды агентов имеют один общий недостаток. Этим недостатком является то, что агенты не могут обучаться, что является серьезным изъяном в интеллектуальных системах.



Обучающийся агент имеет четыре компонента. **Критик** - выполняет функции оценщика действий агента с учетом постоянного стандарта производительности. Критик необходим в данной структуре, поскольку сам агент не понимает успешны ли его действия или нет. Стандарт производительности должен быть постоянным.

**Обучающий компонент** отвечает за внесение усовершенствований, а **производительный компонент** за выбор внешних действий. Обучающий компонент использует информацию от критика с оценкой того, как действует агент, и определяет его дальнейшие действия. Обучающий компонент полностью зависит от производительного компонента. В моделировании такого агента прежде всего нужно получить ответ на вопрос: «Какой производительный компонент потребуется моему агенту, после того как он обучится выполнять свои функции?».

**Генератор проблем** – служит для выбора действий, которые должны привести к получению совершенно нового информационного опыта. Поскольку производительный компонент выбирает только наилучшие действия, то возможно в один момент, производственный компонент будет использовать одни и те же действия все время, полагая что они являются наилучшими, а генератор проблем служит для выбора менее оптимальных действий в начале, но возможно наилучших в конечном результате. То есть генератор проблем предназначен для того, чтобы система смогла экспериментировать, находя наилучшие решения.

Но если агент готов к тому, чтобы немного поэкспериментировать и в кратковременной перспективе выполнять действия, которые, возможно, окажутся не совсем оптимальными, он получает возможность обнаружить гораздо лучшие действия с точки зрения долговременной перспективы.

**Генератор проблем** предназначен именно для того, чтобы предлагать такие исследовательские действия. Именно этим занимаются ученые, проводя эксперименты.

Галилей не считал, что сбрасывание камней с вершины Пизанской башни является самоцелью. Он не ставил задачи просто разбить эти булыжники вдребезги или вправить ими мозги неудачливым прохожим. Его замысел состоял в том, чтобы изменить взгляды, сложившиеся в его собственной голове, сформулировав лучшую теорию движения объектов.

**Обучающий компонент** может вносить изменения в любой из компонентов "знаний", присутствующих на схемах структуры агентов



В простейших случаях обучение будет осуществляться непосредственно на основании последовательности актов восприятия. Наблюдение за парами последовательных состояний среды может позволить агенту получить информацию о том, "какое влияние оказывают мои действия" и "как изменяется мир" в ответ на эти действия.

Например, после того как водитель автоматизированного такси применит определенное тормозное давление во время езды по мокрой дороге, он сможет узнать, какое снижение скорости при этом фактически было достигнуто и наблюдалось ли скольжение по дороге.

На основании полученных результатов генератор проблем может идентифицировать определенные части модели, нуждающиеся в улучшении, и предложить серию экспериментов в виде испытания действия тормозов на разных дорожных покрытиях и в разных погодных условиях.

Для агента, действующего на основе модели, улучшение ее компонентов таким образом, чтобы они точнее соответствовали реальности, почти всегда является хорошей идеей и не зависит от внешнего стандарта производительности. (Иногда с вычислительной точки зрения лучше иметь простую, слегка неточную модель, чем совершенную, но невероятно сложную.)

Информация из внешнего стандарта потребуется вам лишь при необходимости обучения рефлексного компонента или функции полезности.

Например, предположим, что агент - водитель такси, не получает чаевые от тех пассажиров, которые после поездки чувствовали себя утомленными и полностью разбитыми. Внешний стандарт производительности должен информировать агента, что отсутствие чаевых - это отрицательный вклад в его общую производительность.

В этом случае в результате обучения агент получает возможность усвоить, что грубые маневры, раздражающие и утомляющие пассажиров, не позволяют повысить оценку его собственной функции полезности. В этом смысле стандарт производительности позволяет выделить определенную часть входных результатов восприятия как **вознаграждение** (или **штраф**), непосредственно предоставляющее обратную связь в отношении качества поведения агента.

Именно с этой точки зрения в жизни животных можно рассматривать такие жестко закрепленные стандарты производительности, как боль или голод.

В более общем смысле *выбор человека* также может предоставлять информацию о его предпочтениях.

Например, предположим, что агент - водитель такси не имел сведений о том, что людям обычно не нравятся громкие звуки, и остановился на идее непрерывно нажимать на сигнал (клаксон), чтобы предупреждать пешеходов о своем приближении. Последующее за этим поведение пассажиров – затыкание ушей, использование ненормативной лексики и, возможно, обрыв проводов, идущих к сигналу (клаксон) - предоставит агенту возможность успешно обновить свою функцию полезности.

Подводя итог, отметим, что агенты имеют множество компонентов, а сами эти компоненты могут быть представлены в программе агента многими способами, поэтому создается впечатление, что существует большое разнообразие методов обучения.

Однако у всех этих методов есть один объединяющий аспект. Процесс обучения в интеллектуальных агентах в целом можно охарактеризовать как процесс модификации каждого компонента агента с целью более точного соответствия этих компонентов доступной информации обратной связи и тем самым улучшения общей производительности агента.



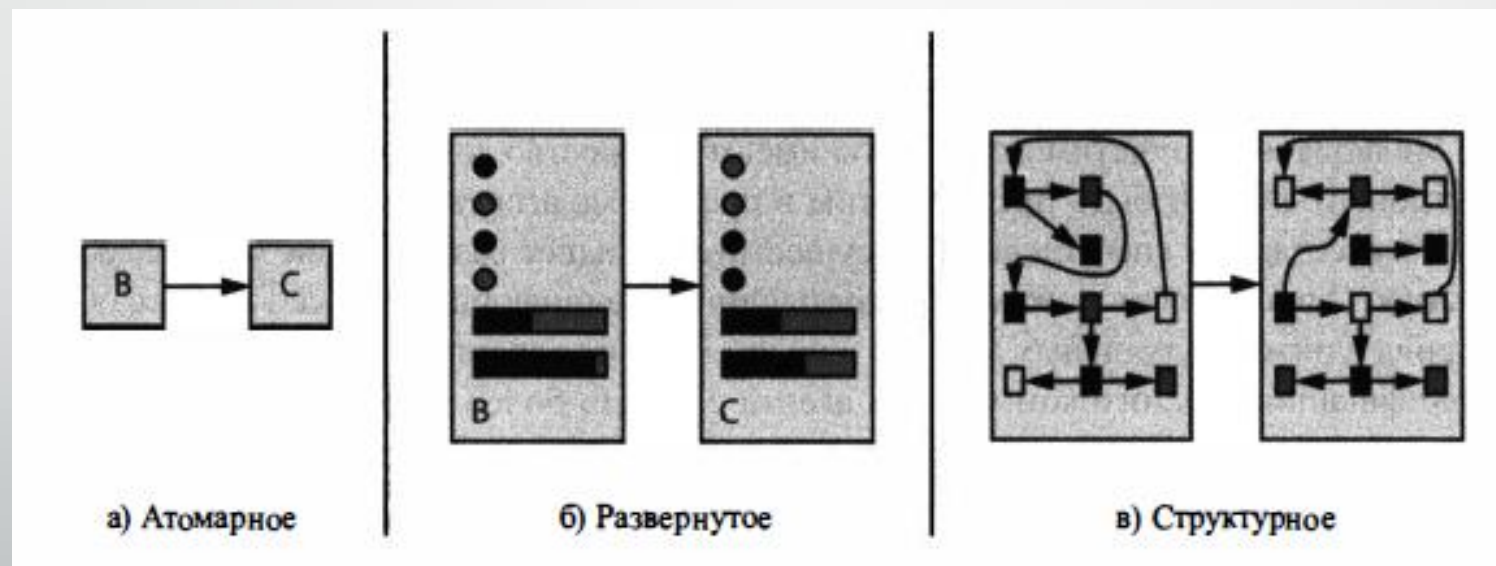
## Как работают компоненты агентских программ

Рассмотренные программы агентов были описаны (в терминах самого высокого уровня) как состоящие из различных компонентов, функции которых заключаются в предоставлении ответов на определенные вопросы: "Как сейчас выглядит мир?", "Какое действие я должен сейчас предпринять?", "Что произойдет в результате моих действий?"

Следующий вопрос, неизбежно возникающий у читателя, - "А как вообще эти компоненты работают?" Чтобы дать хоть сколько-нибудь правильный ответ на такой вопрос, потребуются тысячи страниц. Поэтому я хочу лишь обратить внимание читателя на некоторые основные различия в способах, которыми эти компоненты могут представлять окружающую среду, в которой находится агент.

Можно разместить представления вдоль оси в порядке увеличения сложности и степени выразительности - **атомарное, развернутое и структурное.**

Чтобы проиллюстрировать эту идею, полезно будет рассмотреть определенный компонент агента, например тот, который дает ответ на вопрос "Что произойдет в результате моих действий?" Этот компонент описывает изменения, которые могут произойти в среде в результате выполнения действия. Покажем, как эти изменения могут быть представлены:



*Атомарное* представление:

любое состояние (например, *B* или *C*) является черным ящиком, не имеющим какой-либо внутренней структуры.

*Развернутое* представление:

состояние представлено как вектор значений атрибутов, которые могут быть логическими, действительными числами или некоторым символом из фиксированного набора.

*Структурное* представление: состояние включает объекты, каждый из которых может иметь собственные атрибуты, а также связи с другими объектами

В **развернутом представлении** каждое состояние раскладывается в фиксированный набор **переменных** или **атрибутов**, каждый из которых может иметь определенное **значение**.

Рассмотрим описание задачи определения маршрута автоматизированного такси, но выполненное уже с большей точностью.

Нас будет интересовать не только неделимое местонахождение в том или ином городе, но и количество бензина в баке, текущие ГЛОНАСС - координаты, светится ли индикатор уровня машинного масла, сколько денег потрачено на дорожные сборы, какая радиостанция сейчас в пределах досягаемости и т.д.

В то время как два различных **атомарных** состояния не имеют ничего общего между собой - это просто два разных черных ящика, - два разных **развернутых состояния** могут иметь одинаковые значения некоторых атрибутов (например, одни и те же ГЛОНАСС-координаты), но значения остальных будут различаться (например, бензина в баке много или нет вовсе). Такое представление намного упрощает выработку решения, как перейти из одного состояния в другое. Многие важные области ИИ предполагают использование развернутого представления состояний, включая алгоритмы удовлетворения ограничений, логику высказываний, планирование, байесовские, а также различные алгоритмы машинного обучения.

**Структурные представления** лежат в основе реляционных баз данных и логики первого порядка, вероятностных моделей первого порядка и большей части методов обработки естественного языка. В действительности многое из того, что люди выражают на естественном языке, касается именно объектов и их отношений.

Ось, вдоль которой мы расположили атомарное, развернутое и структурное представления является осью повышения **выразительности представления**.

В общем, более выразительное представление может включать, по крайней мере в сжатой форме, все то, что включает менее выразительное, плюс еще кое-что.

Часто более выразительный язык является **гораздо** более лаконичным. Например, правила игры в шахматы могут быть записаны на одной-двух страницах на языке, использующем структурное представление, таком как логика первого порядка, но потребуются тысячи страниц, если записать их на языке, использующем развернутое представление, таком как логика высказываний, и около  $10^{30}$  страниц, если воспользоваться языком с атомарным представлением, таким как язык конечных автоматов.



С другой стороны, с ростом выразительности представлений выполнение рассуждений и процессы обучения становятся более сложными. Для того чтобы получить преимущества от выразительных представлений и одновременно избежать свойственных им недостатков, интеллектуальным системам в реальном мире может потребоваться работать во всех точках вдоль оси выразительности одновременно.

Другая ось для представлений предполагает отображение концепций на определенное место в физической памяти, будь то память компьютера или мозг. Если между понятиями и ячейками памяти имеет место отображение "один-к-одному", мы называем эту ситуацию **локализованным представлением.**

С другой стороны, если представление концепции распределено по многим ячейкам памяти и каждая ячейка памяти используется как часть представления **нескольких** различных концепций, такая ситуация называется **распределенным представлением**.

Распределенные представления являются более надежными в отношении шума и потери информации. При локализованном представлении отображение концепции на ячейки памяти является произвольным, и если в результате ошибки при передаче информации будет искажено значение всего нескольких битов, понятие **мир** может быть спутано с несвязанным понятием **тир**.

Но в случае распределенного представления каждую концепцию можно представить себе как точку в многомерном пространстве, и если несколько битов исказятся в этом случае, то обращение произойдет к другой, достаточно близкой точке в этом пространстве, которое будет иметь достаточно близкое значение.

## Основные рассмотренные идеи

- **Агентом** является нечто воспринимающее и действующее в определенной среде. **Функция агента** определяет действие, предпринимаемое агентом в ответ в любую последовательность актов восприятия.
- **Показатели производительности** оценивают поведение агента в среде. **Рациональный агент** действует так, чтобы максимизировать ожидаемые значения показателей производительности, с учетом последовательности актов восприятия, полученной агентом к данному моменту.

- Спецификация **проблемной среды** включает определения показателей производительности, внешней среды, исполнительных механизмов и датчиков. Первым этапом проектирования агента всегда должно быть определение проблемной среды с наибольшей возможной полнотой.
- Варианты проблемной среды классифицируются по нескольким важным размерностям. Они могут быть полностью или частично наблюдаемыми, детерминированными или стохастическими, эпизодическими или последовательными, статическими или динамическими, дискретными или непрерывными, а также одноагентными или мультиагентными.

- **Программа агента** реализует функцию агента. Существует целый ряд основных проектов программ агента, соответствующих характеру явно воспринимаемой информации, которая используется в процессе принятия решения. Разные проекты характеризуются различной эффективностью, компактностью и гибкостью. Выбор наиболее подходящего проекта программы агента зависит от характера среды.

- **Простые рефлексные агенты** отвечают непосредственно на акты восприятия, тогда как **рефлексные агенты, основанные на модели**, поддерживают внутреннее состояние, прослеживая те аспекты среды, которые не наблюдаются в текущем акте восприятия. **Агенты, действующие на основе цели**, организуют свои действия так, чтобы достигнуть своих целей, а **агенты, действующие с учетом полезности**, пытаются максимизировать свою собственную ожидаемую “удовлетворенность”.
- Все агенты способны улучшать свою работу благодаря **обучению**.



# Благодарю за внимание!

