**Планирование поведения интеллектуального агента со знаковой картиной мира**

Планирование поведения является важной функцией любой интеллектуальной системы управления сложным техническим объектом. В настоящее время в символьной парадигме искусственного интеллекта предложено большое количество различных алгоритмов планирования, в том числе и использующих прецедентную информацию, т.е. опирающихся на приобретенные знания. Однако проблема символизации (обоснования символов) в существующих способах представления знаний не позволяет эффективно использовать разработанные алгоритмы совместно с механизмами обучения при решении широкого спектра прикладных задач реальными интеллектуальными агентами (робототехническими системами). В настоящей работе представлен оригинальный алгоритм планирования (MAP-планировщик), который в качестве основы для приобретения и сохранения знаний, используемых в дальнейшем при планировании поведения, использует знаковую картину мира. В знаковой постановке задачи планирование является когнитивной функцией реализуемой картиной мира субъекта деятельности. Знаковая модель процесса планирования кроме решения проблемы символизации и обеспечения психологической и биологической правдоподобности, позволяет интеллектуальному агенту взаимодействовать с другими участниками при решении коалиционной задачи. В статье представлены описание используемого представления знаний, MAP-алгоритм планирования и модельный эксперимент в «мире кубиков».

*Ключевые слова*: планирование поведения, знаковая картина мира, образ, значение, личностный смысл, каузальная матрица, семиотическая сеть, MAP-алгоритм, планирование на основе прецедентов.

# Введение

Вопрос разработки методов планирования поведения сложного технического или виртуального объекта имеет большую историю и в основном связывается с успехами отдельной области искусственного интеллекта - автоматического планирования. Здесь достигнуты существенные успехи - предложен ряд символьных методов планирования как в классической постановке задачи, когда действия детерминированы (такие алгоритмы планирования, как FF [[20],](#_bookmark25) FD [[19],](#_bookmark24) LAMA [[29]),](#_bookmark34) так и в недетерминированной постановке с учетом ненулевых вероятностей невыполнения действий и вероятностной реакции среды (алгоритмы на основе марковских процессов и динамического программирования [[2;](#_bookmark7) [3]).](#_bookmark8) Однако создание эффективных и быстрых алгоритмов планирования действий основывается на заранее заданных эвристических принципах поиска на графе и на предположении, что набор действий заранее известен, что делает невозможным автоматическую адаптацию системы планирования к новой задаче с новым списком действий. Это означает, что переноса опыта планирования с выделением абстрактных действий, которые могут иметь различную реализацию в разных ситуациях, в классических подходах не происходит. Существенные трудности возникают, когда существующие алгоритмы адаптируются для многоагентного случая, где предполагается, что агенты обладают как различными наборами действий, так и обладают различными знаниями о внешней среде [[5].](#_bookmark10) В случае коалиционного взаимодействия также необходимо обязательное включение элементов обучения для пополнения базы данных одного агента по поступающей от других участников коалиции информации.

В последнее время исследователи в области управления и планирования уделяют повышенное внимание психологически и биологически правдоподобным моделям и архитектурам управления агентами [[22;](#_bookmark27) [33].](#_bookmark38) Использование различных типов памяти (эпизодической, процедурной и др.) в когнитивных архитектурах нацеленно именно на задачу повторить биологические и психологические пути обмена и организации информации для

решения таких задач как управление и планирование поведения. Это связано в первую очередь с тем, что повышающийся уровень сложности тех задач, в которых действуют робототехнические системы (агенты) требует от них большего уровня автономности, универсальности и гибкости, которую не могут обеспечить существующие методы и алгоритмы. Исследователи в области искусственного интеллекта снова обращаются к естественным примерам решения таких задач - к исследованию поведения человека и животных [[28;](#_bookmark33) [47].](#_bookmark52) Психологически правдоподобные модели когнитивных функций, в том числе планирования, нацелены не только на то, чтобы повторить поведение человека в сложных, в том числе коалиционных, условиях, но и по возможности более полно удовлетворить существующим психологическим представлениям о функционировании психики человека. С одной стороны это может привести к повышению ресурсоемкости предлагаемых алгоритмов, но, с другой стороны, позволит реализовать новые возможности, которые раньше оставались вне круга решаемых проблем специалистами по планированию, например, возможности к целеполаганию, распределению ролей в коллективе. Идеи когнитивной психологии и раньше использовались в классическом

планировании, но, в основном, в бихевиориостском ключе. Так идея в разделении множества действий на автоматические, быстро совершаемые, специфические и произвольные, обобщенные, подсказанное психологической теорией [[21],](#_bookmark26) нашла свое воплощение в иерархическом планировании, а идея сохранения опыта планирования - в прецедентном [[4;](#_bookmark9) [9;](#_bookmark14) [16].](#_bookmark21)

В когнитивной психологии имеется ряд направлении, изучающих феномен планирования, среди которых необходимо выделить три основных [[49]:](#_bookmark54) планирование как часть когнитивной схемы [[25],](#_bookmark30) планирование как метапроцесс [[13;](#_bookmark18) [27]](#_bookmark32) и планирование как часть деятельности [[36].](#_bookmark41) В первом направлении для описания поведения человека используются когнитивные схемы. Например, перцептивная схема – это план сбора информации об объектах и событиях, получения новой информации, обеспечивающий ее непротиворечивую интерпретацию. Схема одновременно включает в себя и сам план, и исполнение плана, это структура действия, равно как и структура для действия. Во втором направлении предусматривается наличие метакогнитивных процессов, позволяющих человеку управлять своими когнитивными процессами и знаниями. С точки зрения Стернберга, можно говорить о глобальном (стратегическом) и локальном (тактическом) планировании. Глобальное планирование требует больших затрат времени, но это компенсируется уменьшением времени на локальное, тактическое планирование. Наконец, в третьем подходе, являющимся одним из наиболее общих, рассматривается иерархическая теория деятельности, которая используется в данной работе и описывается в следующем разделе.

Стоит отметить также, что психологически и биологически правдоподобные модели управления и планирования позволяют по-другому взглянуть на проблему символизации (symbol grounding problem) [[1;](#_bookmark6) [7;](#_bookmark12) [17].](#_bookmark22) Нейрофизиологические модели функционирования сенсорных отделов коры головного мозга в купе с психологической теорией категоризации и восприятия служат основой для построения новых непротиворечивых моделей привязки символов к сенсорным данным. Успехи в данном направлении позволили реализовать некоторые модели в робототехнических системах [[18].](#_bookmark23)

В настоящей работе будет представлен новый психологически и биологически правдоподобный метод планирования поведения, основанный на знаковой теории деятельности и моделях строения кортикально-таломических отделов коры головного мозга. Помимо своей ценности с точки зрения моделирования когнитивных функций человека знаковый подход может быть использован при решении ряда коллективных робототехнических задач (например, для задача интеллектуального перемещения [[45;](#_bookmark50) [47]),](#_bookmark52) не решаемой классическими и другими психолого ориентированным методами (такими как BDI [[30]).](#_bookmark35)

Далее статья организована следующим образом. В главе [1](#_bookmark0) вводятся основные понятия, используемые в статье: даны определения картине мира, знаку и его компонентам с психологическими и нейрофизиологическими обоснованиями. В параграфе [1.1](#_bookmark1) вводится понятие каузальной матрицы как математические структуры для описания компонент знака, рассматриваются ее основные характеристики. В параграфе [1.2](#_bookmark2) исследуются сети, формируемые на основе множеств каузальных матриц и моделирующих отношения на компонентах знака. В главе [2](#_bookmark3) вводится понятие семиотический сети как модели картины

мира, рассматриваются основные типа процессов распространения активности по семиотической сети. В главе [3](#_bookmark4) дано описание MAP-алгоритма планирования поведения в знаковой картине мира (на семиотической сети). В заключение, в главе [4](#_bookmark5) приведен модельный пример работы представленного MAP-планировщика.

# Знаковая картина мира

В качестве способа представления знаний в настоящей работе используется модель знаковой картины мира [[38;](#_bookmark43) [39;](#_bookmark44) [41],](#_bookmark46) которая не только хранит знания об объектах, процессах и отношениях внешней среды, но также представляет внутренние параметры интеллектуального агента, определяющие его мотивационную составляющую и опыт действования. Картина мира также включает в себя процедуры оперирования со знаниями: их приобретение и использование в различных процессах, таких как восприятие, рассуждения,

целеполагание, планирование пове[дения[42].](#_bookmark47) Модель картины мира строится на основе психологических представлений о функционировании психики человека, в частности на представлениях культурно-исторического подхода [[34],](#_bookmark39) теории деятельности [[36;](#_bookmark41) [50]](#_bookmark55) и теории дуальных систем [[12;](#_bookmark17) [32].](#_bookmark37) В соответствии с психологическими воззрениями элементом картины мира является четырехкомпонентная структура - знак, которая представляет для субъекта (в нашем случае интеллектуального агента) все сущности внешней среды и внутреннего пространства: объекты, их свойства, процессы, отношения между объектами и процессами. Следует отметить, что знак является продуктом взаимодействия нескольких субъектов деятельности, образующих некоторую группу (культурную среду), т.е. понятие знака изначально предполагает работу картины мира индивида в кооперации с картинами мира других индивидов.

Образная компонента знака хранит характерные признаки представляемой сущности и одновременно является функцией построения представления этой сущности на основе потока данных, поступающих как с наружных, так и внутренних сенсоров, и в котором выделяются ключевые признаки. Образная компонента индивидуальна для каждого носителя картины мира и образуется в результате процесса наблюдения и обобщения [[31;](#_bookmark36) [44;](#_bookmark49) [46].](#_bookmark51)

Компонента значения знака представляет обобщенные, концептуальные знания субъекта о сущностях внешней среды, а также внутреннего пространства как себя, так и других участников группы. Эти знания являются согласованными, т.е. одинаковы у всех представителей группы. Коммуникативные процессы, которые протекают в группе субъектов (интеллектуальных агентов), в своей основе используют сообщения, которые строятся по общим для всех значениям знаков, задающих, таким образом, синтаксис протокола коммуникации.

Компонента личностного смысла знака содержит индивидуальный личный опыт взаимодействия субъекта с внешней средой с учетом отношения к этому опыту - послужил ли он достижению некоторой цели (удовлетворению некоторой потребности), или же наоборот, оказался неудачным. Личностный смысл знака является динамической его характеристикой, которая постоянно формируется и обновляется в результате

протекания тех или иных когнитивных процессов (планирования, целеполагания). Именно компонента личностного смысла определяется внутренними характеристиками субъекта и его потребностно-мотивационной сферой.

Наконец, четвертая компонента знак - имя - служит для его идентификации как в коммуникативных процессах, так и в произвольных процессах планирования и рассуждений. Имя знака, как и его значение, является утвержденной, слабо меняющейся в группе субъектов компонентой знака.

Знаки представляет в картине мира субъекта как статические объекты и свойства внешней среды, так и динамические ее составляющие: процессы, ситуации - а также внутренние характеристики агента: действия, объекты и свойства «внутренней среды». Пусть у нас есть объект внешней среды - *лимон*. В картине мира некоторого субъекта он может представляться знаком с именем «лимон», образ которого включает такие признаки как *желтый цвет*, *овальную форму* и *кислый вкус*. Эти признаки также могут быть представлены в картине мира знаками, либо являться информацией, поступающей напрямую от сенсоров. Значением знака «лимон» будут являться те обобщенные действия и процессы, в которых по общему согласованному мнению некоторого коллектива, которому принадлежит субъект, участвует *лимон*: лимон *принято употреблять в пищу*, *использовать как соус для рыбных блюд* или *использовать для профилактики болезней*. Личностный смысл лимона для субъекта - это те конкретные персональные действия и процессы, в которых субъект имел опыт использования лимона, решая некоторую задачу: когда-то я *кидал лимон в соседа по парте в школе* или *сжевал целый лимон и не скривил лицо*. Все действия и процессы также могут быть представлены некоторыми знаками, либо не выводиться на знаковый произвольный уровень и являться некоторыми не обозначаемыми операциями.

Кроме психологических оснований четырхекомпонентной структуры знака имеются нейрофизиологические свидетельства в пользу существования такой структуры хранения и активации элементов индивидуального опыта [11; [35].](#_bookmark40) Кроме того, нейрофизиологические данные служат основной для построения моделей компонент знака и некоторых функций, таких как восприятие и распознавание [[14;](#_bookmark19) [44].](#_bookmark49) Нейрофизиологические свидетельства в пользу высокой однородности строения различных отделов коры больших полушарий мозга, а также участия таламуса в формировании и запоминании временных последовательностей [[6;](#_bookmark11) [8],](#_bookmark13) приводят к используемой в данной статье математической структуре каузальной матрицы [[46]](#_bookmark51) для описания строения компонент знака.

Знаковый подход к представлению знаний и описание процессов, протекающих в знаковой картине мира позволяют решить ряд трудных задач в области ситуационного управления [[40;](#_bookmark45) [43]](#_bookmark48) и управления сложными техническими объектами [[48].](#_bookmark53) Использование знаковой картины мира для реализации функций стратегического уровня управления робототехническими системами [[37]](#_bookmark42) демонстрирует применимость используемого подхода не только для представления знаний, но и для решения задач коалиционного планирования и распределения ролей.

## Компоненты знака

Рассмотрим структуру

компонент знака на примере образной компоненты, которая участвует в распознавании (актуализации) знака, выделении представления об опосредуемом объекте или процессе на основе поступающей из внешней среды сенсорной информации и регистрируемой внутренними сенсорами моторной информации. До именования знак будем называть протознаком или признаком.

Предположим, что во входном потоке данных выделена последовательность (*x*1*, x*2*, . . . , xh*) длины *h* векторов действительных чисел от 0 до 1, которые будем называться *событиями*. Каждое событие *xt* длины *q* представляет собой запись выходов от *q* сенсоров, а каждый элемент события означает уверенность в срабатывании данного сенсора. Например, событие (0*.*1*,* 0*.*9*,* 0*.*9) поступает с трех сенсоров - датчиков красного, синего и зеленого света - и означает, что уверенность в срабатывании датчика красного света составляет 10%, а синего и зеленого - по 90%.

Образная компонента знака должна по входной последовательности данных определить, присутствует ли (закодирован ли) опосредуемый объект или процесс в этой последовательности. Для этого мы будем кодировать характерные признаки объекта или процесса в специальной структуре - каузальной матрице *z* = (*e*1*, e*2*, . . . , eh*) размерности *q* на *h*, где *q* - размерность входных событий, а *h* - длина последовательности входных событий. При этом каждый столбец *et* каузальной матрицы является битовым вектором длины *q* и кодирует те признаки (которым соответствуют 1), которые необходимо должны присутствовать во входном событии в момент времени *t*, чтобы опосредуемый объект или процесс мог быть распознан во входном потоке данных, т.е. задают множество одновременных характерных признаков. Например, образу знака *s*, представляющему объект «квадрат», может соответствовать каузальная матрица z где первая строчка является характеристическим вектором информации с датчика углов на изображении, вторая - с датчика положения визуального сенсора (верхнее положение), третья - нижнее положение сенсора, четвертая - левое положение сенсора, пятая - правое положение (см. [рис.4).](#_bookmark56)

Образу каждого знака может соответствовать несколько каузальных матриц, которые задают различные проявления представляемого объекта или процесса. Весь кортеж каузальных матриц образа знака *s* будем обозначать как *Zp*(*s*).

Случай, когда характерными признаками образа данного знака выступают данные с сенсоров, является частным. В более общей постановке, признаками для образа знака служат другие знаки, которые представляют эти характерные признаки. Таким образом, мы можем сопоставить образу знака *s* множество *Sp*(*s*) мощности *q*, каждому элементу которого соответствует номер строчки каузальной матрицы *z* размера *q* на *h*, т.е. каждому признаку *si* ∈ *Sp*(*s*) соответствует характеристический битовый вектор, задающий на

местах 1 те моменты времени, когда данный признак должен присутствовать во входных данных, чтобы успешно актуализировать знак (распознать образ знака) *s*.

Для уточнения определения множества *Sp*(*s*) введем семейство бинарных отношений

{c*p,* c1*,* c2*, . . .* }, определённых на декартовом произведении *S* × *S*. Будем считать, что

знак

*p p*

*si* является *элементом образа* знака

*s*, (*si, s*) ∈c*p*

или *si* c*p*

*s*, в том случае,

если *si* ∈ *Sp*(*s*). Если известно, что знаку *si* соответствует 1 в *t*-м столбце некоторой каузальной матрицы *z* ∈ *Zp*(*s*) знака *s*, то мы будем использовать вложенное отношение

c*t*

*p*⊂c*p*.

## Каузальная сеть

Введем специальную процедуру Λ*p* : 2*Z* → 2N × 2N, которая каждому кортежу каузальных матриц *Zp*(*s*) ⊂ *Z* образа знака *s* ставит в соответствие два не пересекающихся подмножества индексов собственных столбцов *Ic* ⊂ N*,* ∀*i* ∈ *Ic i* ≤ *h* (индексы столбцов условий) и *Ie* ⊂ N*,* ∀*i* ∈ *Ie i* ≤ *h* (индексы столбцов эффектов): Λ*p*(*Zp*(*s*)) = (*Ic, Ie*)*, Ic* ∩ *Ie* = ∅. Например, если для множества матриц *Z* =

{((1*,* 0)*,* (0*,* 1))} процедура Λ*p* выдает два множества {1} и {2}, то это означает, что

появление признака, соответствующего первой строчке матрицы, вызывает появление

признака, соответствующего второй строчке. Процедура Λ*p* по сути является функцией установления причинно-следственного отношения на множестве входных событий и может реализовываться различными способами, в т.ч. на основе алгоритмов Норриса, FCbO, AddIntent [([23;](#_bookmark28) [24;](#_bookmark29) [26])](#_bookmark31)

В том случае, когда для матриц *Zp*(*s*) образа знака *s* множество столбцов эффектов пусто *Ie* = ∅, т.е. когда по данному множеству каузальных матриц не возможно однозначно определить, какие события всегда предшествуют другим, мы будем считать, что причинно-следственная связь не установлена и знак опосредует некоторый объект или ситуацию. В противном случае будем считать, что знак опосредует некоторое действие или процесс, результат которого кодируется в столбцах эффектов, а условие

* в столбцах условий.

Справедливы следующие утверждения относительно свойств процедуры Λ*p*:

* + *Ic* ∩ *Ie* = ∅ — столбец матрицы предсказания не может быть одновременно и условием и эффектом,
  + |*Ic* ∪ *Ie*| = *h* — столбец матрицы предсказания является либо условием либо эффектом,
  + *Ic* /= ∅ — среди столбцов матрицы предсказания должен быть хотя бы один столбец условий, в то время как эффектов может и не быть (в случае объектных признаков),
  + ∀*i* ∈ *Ie, j* ∈ *Ic i > j* — все условия предшествуют эффектам по времени. Схема каузальной матрицы, с учетом выше сказанного, приведена на рис. [2.](#_bookmark58)

Теперь введем понятие каузальной сети, которая будет определять гетерархию на множестве образов. Каузальная сеть *Wp* = (*Vp, Ep*) - является помеченным ориентированным графом, в котором

* + каждому узлу *v* ∈ *Vp* ставится в соответствие кортеж казуальных матриц *Zp*(*s*)

образа некоторого знака *s*, что будем обозначать как *v* → *Zp*(*s*);

* + ребро *e* = (*v*1*, v*2) принадлежит множеству ребер графа *E*, если *v*1 → *Zp*(*s*1)*, v*2 →

*Zp*(*s*2) и *s*1 ∈ *Sp*(*s*2), т.е. если знак *s*1 является элементом образа знаком *s*2;

* + каждому ребру графа *e* = (*v*1*, v*2)*, v*1 → *Zp*(*s*1)*, v*2 → *Zp*(*s*2) ставится в соответствие метка *E* = (*E*1*, E*2*, E*3) - кортеж трех натуральных чисел:
    - *E*1 - индекс исходной матрицы в кортеже *Zp*(*s*1), может принимать специальное значение 0, если исходными могут служить любые матрицы из кортежа;
    - *E*2 - индекс целевой матрицы в кортеже *Zp*(*s*2), строка которой ставится в соответствие признаку *s*1;
    - *E*2 - индекс столбца в целевой матрице, в которой в соответствующей признаку *s*1 строке стоит 1, может принимать положительные значения (столбцы условий) и отрицательные (столбцы эффектов).

Каузальная сеть представляет собой некоторое множество пересекающихся иерархий знаков. Каждый знак представлен множеством каузальных матриц, задающих образ этого знака, а иерархия представляет иерархические связи между образами. Такую связь можно читать как «знак *x* участвует в формировании образа знака *y*». При этом мы специфицируем для какой именно матрицы знака *y* и какого именно столбца этой матрицы нужен знак *x* (метки *E*2 и *E*3 соответственно). В некоторых случаях мы также можем указать и участвующую в процессе формирования образа матрицу знака *x* (метка *E*1). Пример такой сети изображен на рис. [3.](#_bookmark59)

Аналогичным образом определяются каузальные сети для остальных компонент знака

* для значения и личностного смысла. Для каждого знака *s* задаются множества *Sm*(*s*)

и *Sa*(*s*), т.е. определяются семейства отношений {c*m,* c1 *,* c2 *, . . .* } и {c*a,* c1 *,* c2 *, . . .* }.

*m m a a*

Множество

*Sm*(

*s*) интерпретируется как ролевой состав знака *s*, например, элементы

подкласса или роль действия. Множество *Sa*(*s*) интерпретируется как мгновенный

компонентный состав некоторой ситуации, наблюдаемой и переживаемой субъектом, носителем картины мира, в настоящее время. Аналогично определяются множества *Zm*(*s*), *Za*(*s*), процедуры Λ*m* и Λ*a*.

# Семиотическая сеть

Знаком *s* будем называть четверку (*n, p, m, a*), где *n* - имя знака, *p* - образ знака,

кортеж каузальных матриц (*zp*(*s*)*, zp*(*s*)*, . . .* ), соответствующий узлу *wp*(*s*) каузальной

1 2

сети на образах; *m* - значение знака, кортеж каузальных матриц (*zm*(*s*)*, zm*(*s*)*, . . .* ),

1 2

соответствующий узлу *wm*(*s*) каузальной сети на значениях, *a* - личностный смысл

знака, кортеж каузальных матриц (*za*(*s*)*, za*(*s*)*, . . .* ), соответствующий узлу *wa*(*s*)

1 2

каузальной сети на смыслах.

Будем называть *семиотической сетью* пятерку Ω = (*Wp, Wm, Wa, Rn,* Θ), где

* *Wp, Wm, Wa* - соответственно каузальные сети на множестве образов, значений и личностных смыслах,
* *Rn* - семейство отношений на множестве знаков, сгенерированных на основе трех каузальных сетей, т.е. *Rn* = {*Rp, Rm, Ra*},
* Θ - семейство операций на множестве знаков, которые генерируются на основе структуры фрагментов трех типов каузальных сетей, к которым принадлежат соответствующие компоненты знаков (подробнее см. [[41]).](#_bookmark46)

Еще раз отметим, что знак представляет не только объекты внешнего мира, но также процессы, протекающие в нем, выполнимые действия, а также ситуации, наблюдаемые во внешней среде.

Три типа каузальных сетей, составляющих семиотические сеть, не независимы друг от друга. Между узлами каждой сети установлено взаимно-однозначное соответствие: для каждого узла *wx* сети *Wx* найдутся единственные узлы *wy* и *wz* в сетях *Wy, Wz*

(*x, y, z* ∈ {*p, m, a*}), такие, что все три узла соответствуют одному и тому же знаку

*s* = (*wp, wm, wa*). Имя знака служит меткой узлов в каждой сети: в каузальной сети может быть только один узел с данной меткой, а узлы всех сетей с одинаковыми метками

образуют компоненты знака. Для связи каузальных матриц различных типов узлов в рамках одного знака служат специальные функции связывания: Ψ*m,* Ψ*a ,* Ψ*p* и обратные

*p m a*

им Ψ*p ,* Ψ*m,* Ψ*a* [[41].](#_bookmark46) Каждая функция связывания ставит каузальной матрице одного типа

*m a p*

каузальную матрицу другого типа либо генерирует эту матрицу в том случае, если она

отсутствует в соответствующем узле сети.

Введем понятия активности в семиотической сети и процесса его распространения. Введем некоторую метку активности для каузальных матриц сети *Wx* (*x* ∈ {*p, m, a*}) и будем называть активным множество *Z*∗ матриц, обладающих этой меткой. Процесс

*x*

распространения активности представляет собой изменение состава множества *Z*∗ с

*x*

течением времени (каждый дискретный момент) и описывается для каждого типа каузальной сети своей функцией: *ϕa, ϕm, ϕp*. Процесс распространения активности является итерационным, т.е. на каждом шаге новый состав множества активных матриц порождается на основе предыдущего состава и зависит от матриц, туда входящих. В простейшем случае мы будем рассматривать такой процесс, в котором каждая матрица не влияет на ход распространения активности от другой матрицы и поэтому будем считать, что функции *ϕx* принимают на вход одну активную матрицу и выдают новое подмножество активных матриц.

В связи с тем, что ребра каузальных сетей имеют направлением, будем различать распространение активности вверх по сети, когда используются исходящие от узла ребра (функции *ϕ*↑ ), и распространение активности вниз по сети, когда используются входящие в узел ребра (функции *ϕ*↓ ). В дальнейшем, при описании алгоритма

*x*

*x*

планирования, нам понадобятся только функции на сети значения и личностных

смыслов. Каждую функцию *ϕ*↑ *, ϕ*↓

будем параметризовать глубиной распространения

*x x*

активности *dx*, которая указывает на какую глубину просматриваются ребра в данном

направлении (вверх или вниз).

В дальнейшем при описании алгоритма планирования будет использоваться понятие фрагмента казуальной сети. Под фрагментом *F* мы будем подразумевать некоторое множество узлов *V* сети *Wx* = (*Vx, Ex*) вместе со всеми ребрами *E* их связывающими:

*F* = (*V, E*) : *V* ⊆ *Vx* и ∀*e* = (*v*1*, v*2) ∈ *Ev*1 ∈ *V, v*2 ∈ *V* .

# Планирование в знаковой картине мира

Процесс планирования в знаковой картине мира реализуется с помощью MAP- алгоритма и идет в обратном направлении: от конечной ситуации к начальной. Кратко опишем основные этапы его работы. На вход алгоритма поступает описание задачи

*T* = (*NT , S, Sitstart, Sitgoal*)*,*

где *NT* - идентификатор задачи, *S* - множество знаков семиотической сети, *Sitstart* =

(∅*,* ∅*, astart*) - начальная ситуация со смыслом *astart* = {*za*

*start*

}, *Sitgoal* = (∅*,* ∅*, agoal*)

- целевая ситуация со смыслом *agoal* = {*za*

*goal*

}. В общем случае задача *T* является

результатом процедуры «означивания» - формирования картины мира по исходным

описаниям домена планирования *D*, задающему списки возможных действий и типов объектов, и задачи планирования *P* , включающей в себя определение стартовых условий и конечной цели (шаг [1).](#_bookmark4)

Результатом MAP-алгоритма

является план *Plan* = {(*za , za* )*,* (*za , za* )*, . . . ,* (*za , za* )} - последовательность длины *n*

*za a*

*s*1 *p*1

*a*

*s*2 *p*2

*sn pn*

пар (

*si, zpi*), где *zsi* - каузальная матрица некоторого узла сети на личностных смыслах,

представляющая *i*-ую ситуацию планирования, а *za*

*pi*

- каузальная матрица некоторого

личностного смысла, представляющая применяемое в ситуации *za*

*si*

действие. При этом

ситуация *za*

*si*+1

является результатом выполнения действия *za* , в том смысле, который

раскрывается далее при обсуждении алгоритма, *za*

*pi*

*s*1

:= *z*

*a start*

* каузальная матрица,

соответствующая смыслу начальной ситуации, *za*

*sn*

*a goal*

) - каузальная матрица,

соответствующая смыслу целевой ситуации.

= *z*

**Input:** описание домена планирования *D*, описание задачи планирования *P* , максимальная глубина итераций *imax*

**Output:** план *P lan*

1: T = (*NT , S, Sitstart, Sitgoal*) := GROUND(*P* )

// *NT* - идентификатор задачи, *S* - множество знаков, *Sitstart* = (*idstart*∅*,* ∅*,* {*za*

*start*

}) - начальная

ситуация со смыслом *astart*, *Sitgoal* = (*idgoal*∅*,* ∅*,* {*za*

*goal*

}) - целевая ситуация со смыслом *agoal*

2: *Plan* := MAP\_SEARCH(*T* )

3: **function** MAP\_SEARCH(*T* )

4: *zcur* := *za*

*goal*

5: *zstart* := *za*

*start*

6: *Plans* := MAP\_ITERATION(*zcur* ,*zstart*, ∅, 0)

7: {*Plan*0*, Plan*1*, . . .* } = SORT(*Plans*)

8: **return** *Plan*0

Процесс планирования является иерархическим и состоит из повторения MAP- итерации, включающей в себя четыре этапа (см. рис. [4):](#_bookmark60)

* + *S-этап* – поиск прецедента совершения действий в текущей ситуации,
  + *M-этап* – поиск применимых действий на сети значений,
  + *A-этап* – генерация личностных смыслов, соответствующих найденным значениям,
  + *P-этап* – построение новой текущей ситуации по множеству признаков условий найденных действий,

Кратко, MAP-алгоритм осуществляет итеративную генерацию новых каузальных матриц *znext* личностных смыслов на основе текущей активной матрицы *zcur* до тех пор, пока не будет достигнуто предельное количество шагов *imax* (шаг [10)](#_bookmark4) или не будет целиком активирован начальная матрица *zstart* (шаг [41),](#_bookmark4) соответствующей личностному смыслу *astart* начальной ситуации. В качестве текущей активной каузальной матрицы для первой итерации выступает матрица, соответствующая личностному смыслу

целевой ситуации *za*

*goal*

(шаг [6).](#_bookmark4) После завершения выполнения всех итераций,

найденные планы сортируются по длине (шаг [7)](#_bookmark4) и самый короткий из них является решением задачи планирования в знаковой картине мира (шаг [8)](#_bookmark4).

Первым этапом в MAP-итерации является S-этап. Его суть заключается в том, что в картине мира интеллектуального агента производится поиск прецедентов, т.е. поиск действий, которые совершались в текущих условиях *zcur* . Для этого просматриваются все знаки в картине мира *S* и их личностные смыслы *a*(*s*) (шаги [13–14).](#_bookmark4) Если текущие

условия *zcur* удовлетворяются матрицей *za*, то список прецедентов *A*ˆ*case* пополняется

результатом распространения активности по сети личностных смыслов от знака *s* на расстояние *da* (шаг [16).](#_bookmark4)

Далее в MAP-алгоритме следует M-этап, на котором происходит распространение

активности по сети личностных смыслов на расстояние *da* с целью активации всех

9: **function** MAP\_ITERATION(*zcur* ,*zstart*, *Plancur* , *i*)

10: **if** *i* ≥ *imax* **then**

11: **return** ∅

12:

*A*ˆ*case* := ∅ // Список прецедентов

// *S*-этап

// Поиск прецедентов выполнения действий в текущих условиях

13: **for all** *s* ∈ *S* **do**

14: **for all** *za* ∈ *a*(*s*) **do**

15: **if** *za* ≥ *zcur* **then**

*a*

16:

*A*ˆ*case* = *A*ˆ*case* ∪ *ϕ*↑(*s, da*)

знаков, связанных с текущей ситуацией (шаг [17).](#_bookmark4) Элементы полученного множества каузальных матриц *A*∗ служат отправными точками для распространения активности по сети значений: для каждой матрицы *za* с помощью функции связывания Ψ*m* определяется необходимый узел на каузальной сети значений, от которого активность распространяется на расстояние *dm* (шаг [20).](#_bookmark4) Если активированные матрицы являются каузальными, то они добавляются в множество активных значений *M* ∗ (шаг [22).](#_bookmark4)

*a*

// *M* -этап

// Распространение активности вниз по сети личностных смыслов 17: *A*∗ = *ϕ*↓(*zcur, da*)

*a*

18: *M* ∗ = ∅

19: **for all** *za* ∈ *A*∗ **do**

// Распространение активности вверх по сети значений 20: **for all** *zm* ∈ *ϕ*↑ (*s*(*za*)*, dm*) **do**

*m*

21: **if**

*Ie*(*zm*) /= ∅ **then**

22: *M* ∗ := *M* ∗ ∪ {*zm*}

Затем переходим к A-этапу, на котором происходит генерация каузальных матриц на сети личностных смыслов, которые представляют специфицированные относительно текущих условий *zcur* действия, определяемые активными значениями из множества *M* ∗. Для этой цели служат шаги [25–27,](#_bookmark4) в которых распространение активности на каузальной сети значений на расстояние *dm* приводит к активации множества значений *M* ∗ знаков, связанных с ролевой структурой процедурной матрицы *zm*, а

затем с помощью функции связывания Ψ*a*

*m*

происходит генерация новой каузальной

матрицы на сети личностных смыслов, которая копирует значение *z*∗

*m*

с замещением

абстрактных знаков-ролей объектными знаками, связанными с ролями отношением класс-подкласс. Затем на A-шаге происходит отбор тех каузальных матриц, которые представляют действия, выполнимые в текущих условиях *zcur* (шаги [30–32).](#_bookmark4) Для этого удаляются все каузальные матрицы, эффекты которых не включены в текущую ситуацию (напомним, что планирование осуществляется в обратном направлении). В заключение A-этапа выполняется одна из операций в картине мира *θa*, осуществляющая в данном случае метарегулирование - проверку некоторой эвристики, которая может выражать, например, то правило, что нельзя повторять одинаковые действия, или лучше выполнить

то действие, которое быстрее всего приближает к начальным условиям *zstart* (шаг [33).](#_bookmark4) Любое эвристическое правило также представимо в виде каузальной матрицы личностного смысла знака, представляющего внутреннюю стратегию планирования своего поведения.

23:

// *A*-этап

*A*ˆ*gen* = ∅

24: **for all** *zm* ∈ *M* ∗ **do**

// Распространение активности вниз по сети значений 25: *M* ∗ = *ϕ*↓ (*zm, dm*)

*m*

26: **for all** *z*∗

*m*

∈ *M* ∗ **do**

27: ˆ

:= *A*ˆ

∪ {Ψ*a* (*z*∗ )}

*Agen*

*gen m m*

28:

// Совмещение активности образованных смыслов и текущей ситуации

*A*ˆ = *A*ˆ*gen* ∪ *A*ˆ*case*

29: **for all** *za* ∈ *A*ˆ **do**

30: *zshift* = (*ei*|*i* ∈ *Ie*)

31: **if** *zcur* /≥ *zshift* **then**

32:

33:

*A*ˆ = *A*ˆ \ {*za*}

// Метакогнитивная проверка эвристики

*A*ˆ = {*θa*(*za*)|*za* ∈ *A*ˆ

}

34: **if** *A*ˆ = ∅ **then**

35: **return** ∅

Завершается MAP-алгоритм P-этапом. Здесь для каждой сгенерированной каузальной матрицы *za*, представляющей некоторое действие, формируется новая ситуация *Sitnext*, которая является результатом обратного применения действия в текущих условиях *zcur* . Обратное применение (шаг [39)](#_bookmark4) заключается в формировании каузальной матрицы

*znext*, состоящей из событий, являющихся либо колонками-условиями действия *ei* ∈

{*ek* |*ek* ∈ *za, k* ∈ *Ic*(*za*), либо принадлежащих текущей активной каузальной матрице и не являющихся колонками-эффектами действия *ei* ∈ *zcur* ∧ *ei* /∈ {*ej* |*ej* ∈ *za, j* ∈ *Ie*(*za*)}. В текущий план *Plancur* добавляется пара текущие условия - применимое действие (*zcur, za*). Если новая ситуация не покрывает стартовую ситуацию (шаг [41),](#_bookmark4) то итерации продолжаются с новой текущей ситуацией, пополняя все множество генерируемых

планов *Plansfin*.

Константы *da, dm*,

которые определяют глубину распространения активности в каузальных сетях, являются параметрами алгоритма и задают внутреннюю характеристику носителя картины мира, различаясь от агента к агенту. Обычно в модельных экспериментах эти параметры не превышают 5.

// *P* -этап

36: *Plansfin* := ∅

37: **for all** *za* ∈ *A*ˆ **do**

38: *Plancur* = *Plancur* ∪ {(*zcur, za*)}

// Генерация новой ситуации - применение действия

39: *znext* := (*ei*|(*ei* ∈ *zcur* ∧ *ei* /∈ {*ej* |*ej* ∈ *za, j* ∈ *Ie*(*za*)}) ∨ *ei* ∈ {*ek* |*ek* ∈ *za, k* ∈

*Ic*(*za*)})

40: *Sitnext* = (*idnext,* ∅*,* ∅*,* {*znext*})

41: **if** *znext* ≥ *zstart* **then**

42: *Plansfin* = *Plansfin* ∪ {*Plancur* }

43: **else**

44: *Plansrec* := MAP\_ITERATION(*znext*,*zstart*, *Plancur* , *i* + 1)

45: *Plansfin* = *Plansfin* ∪ *Plansrec*

46: **return** *Plansfin*

# Модельный пример: мир кубиков

Продемонстрируем работу представленного алгоритма планирования поведения с помощью модельного эксперимента, доменом планирования для которого выступает широко известный в области автоматического планирования пример «мир кубиков» [[15].](#_bookmark20) Описание домена на языке PDDL [[10]](#_bookmark15) состоит из определения типа (*blocks*), четырех предикатов (*ontable*, *clear*, *handempty*, *holding*) и четырех действий (*pick-up*, *put-down*, *stack*, *unstack*) (см. табл. [1).](#_bookmark65)

Приведем пример решения с использованием MAP-алгоритма следующей задачи планирования - построение башни из четырех кубиков, лежащих на столе (табл. [1).](#_bookmark65) Фрагмент каузальной сети на личностных смыслах, задающего каузальную матрицу смысла начальной ситуации с именем *start*, приведен на рис. [5.](#_bookmark61) У каждого отдельного кубика (*a*, *b*, *c*, *d*) одна каузальная матрица в узле сети, в то время как для предикатов *clear* и *ontable* имеется по четыре матрице в узле, т.к. они принимают участие в событиях с разными кубиками. Например, матрицы знака *clear* присутствуют в 1, 3, 5 и 7 столбцах матрицы знака *start* одновременно с кубиками *a*, *b*, *c*, *d* соответственно, что означает, что на всех кубиках не лежат другие кубики.

На рис. [6](#_bookmark62) представлена целевая ситуация, в которой все четыре кубика составлены в башню: кубик *d* находится на столе, кубик *c* - на *d*, *b* - на *c*, и, наконец, на самом верху - кубик *a*. Предикат *on*, который задает отношение «находится на» может быть представлен в виде процедурный каузальной матрицы, чтобы явно продемонстрировать несимметричность этого отношения, хотя использование объектной матрицы никак не влияет на результат. Здесь также от каждого кубика в ситуации участвует одна каузальная матрица, а предикат *on* представлен в виде узла с тремя каузальными матрицами, т.к. участвует в матрице знака целевой ситуации *goal* в различных столбцах с тремя различными кубиками.

На рис. [7](#_bookmark63) представлен фрагмент каузальной сети на значениях, представляющий собой элементы процедурной каузальной матрицы знака *stack* и отношения «класс-подкласс»

объектов-кубиков, класса *block* и ролей в действии *stack*: *block?x* (аналог семантической роли «объект») *block?y* (аналог семантической роли «директив»). Здесь необходимо отметить, что метка *E*1 ребра *v* (индекс исходной матрицы узла, из которого исходит

ребро *v*) в случае отношения «класс-подкласс» (*a*→*block*, *block*→*block?x*) принимает

специальное нулевое значение, что означает, что исходной может быть любая казуальная

матрица данного узла. Иными словами, роль *block?x* может играть любой из кубиков *a*, *b*, *c* или *d*.

Рассмотрим этапы MAP-алгоритма: S, M, A и P-этапы - на первой итерации алгоритма. Рассмотрим простейший случай, когда наш интеллектуальный агент не накопил опыта действования в условиях данной задачи. В следствие этого на S-

этапе множество прецедентов

*A*ˆ*case* будет пусто. С учетом того, что планирование

осуществляется в обратном направлении, на первом M-этапе мы рассматриваем целевую ситуацию как текущую активную матрицу предсказания *zcur* и распространение от нее активности вниз по сети личностных смыслов будет активировать множество *A*∗, совпадающее с фрагментом, изображенным на [рис.5.](#_bookmark61) В множество значений *M* ∗ попадут значения знаков, представляющих кубики *a*,*b*,*c*,*d* - это все связанные с ними

по сети значений процедурные знаки *stack*, *unstack*, *pick-up*, *put-down*. На рис. [8](#_bookmark64) слева представлен фрагмент каузальный сети на значениях, включающий процедурную матрицу знака *unstack*. Для активации матрицы знака *unstack* от матрицы знака *a* достаточно использовать в качестве константы *dm* значение в три ребра.

На A-этапе происходит генерация новых каузальных матриц *A*ˆ*gen* в сети личностных

смыслов путем распространения активности вниз по сети значений. Пример такого распространения, в результате которого образуется новая каузальная матрица знака *unstack*, представлен на рис. [8](#_bookmark64) справа. Новая каузальная матрица на сети личностных смыслов является копией соответствующей матрица на сети значений с заменой ссылок, указывающих на знаки-роли, на ссылки, указывающие на объектные не абстрактные знаки, представляющие кубики. В нашем примере будет сгенерирована матрица, соответствующая действию *unstack*(*a*, *b*) - снять кубик *a* с кубика *b*. На данном этапе будет сформировано по четыре матрицы для одноместных действий и двенадцать - для двухместных.

В завершение A-этапа эффекты построенных процедурных матриц проверяются на применимость в условиях текущей ситуации и среди применимых действий отбираются те, которые удовлетворяют некоторому метакогнитивному правилу (эвристике) *θa*. В нашем примере, единственным применимым действием из всех сформированных вариантов будет действие *unstack*(*a*,*b*). В качестве эвристики может быть использовано жадное правило: выбираем те действия, которые максимально быстро приближают к успеху (новая ситуация имеет больше общих признаков с целевой).

В конце итерации, на P-этапе, генерируется новая каузальная матрица *znext* в сети личностных смыслов знака, представляющего следующую ситуацию планирования. В нашем примере новая текущая ситуация будет совпадать с предыдущей за исключение того, что кубик *a* теперь находится в манипуляторе, а на кубике *b* теперь ничего не находится. В текущий план *Plancur* добавляется пара (*zcur, za*) каузальных матриц текущей ситуации и выбранного действия. Т.к. новая ситуация не включает в себя стартовую ситуации начинаем новую итерацию.

В результате работы MAP-алгоритма в нашем примере будет получен план из 6 действий: *pick-up*(*c*), *stack*(*c*,*d*), *pick-up*(*b*), *stack*(*b*,*c*), *pick-up*(*a*), *stack*(*a*,*b*). В завершение работы агента над этой задачей он сохраняет прецедент планирования в своей картине мира: он сохраняет начальную и конечную ситуацию в виде новых знаков и образует новый процедурный знак, который можно назвать как «построить башню». Единственным признаком в столбце условий данного знака будет начальная ситуация, единственным признаком в столбце эффектов - целевая ситуация. После этого интеллектуальный агент сможет решить ту же задачу, найдя на S-этапе необходимое действие, которое сразу приведет к цели. Такая же ситуация может возникнуть и в другой задаче по ходу ее решения, что приведет к сокращению пространства поиска подходящих действий.

# Заключение

В классической символьной постановке задачи планирования в искусственном интеллекте возникает проблема совмещения символьных алгоритмов планирования с методами обучения, сохраняющими как опыт планирования, так и обеспечивающими адаптацию действий к новым условиям. Данная проблема смыкается с проблемой символизации - привязки используемых в классическом способе представления знаний символов к реальным объектам, процессам и свойствам внешней среды. Особенно остро данные проблемы проявляется при реализации обучаемых робототехнических систем, для которых важно сопоставлять символы, используемые при концептуальном планировании с данными, поступающими от сенсоров. При этом, когда перед сложной технической системой ставится задача планирования в довольно широком спектре условий, в том числе и коалиционных, подходы с заранее сформированной, хоть и пополняемой, базой знаний показывают свою неэффективность. Способ представления знаний, на котором базируются функции управления интеллектуальным агентом, должен изначально поддерживать возможность привязки символов к данным сенсоров и поддерживать как представление внутренней информации, так и обобщенной, согласованной с другими участниками коалиции информации. В настоящей работе эти задачи решаются с использованием знаковой картины мира. Представлен оригинальный метод планирования (MAP-алгоритм), который использует и сохраняет прецедентную информацию в процессе синтезе плана. Используемый четырехкомпонетный элемент картины мира (знак) позволяет кодировать не только информацию о внешней среде, но и внутренние характеристики и мотивационно-потребностные свойства, а так же общие коллективные знания. Представленный алгоритм также может быть использован и для составления коалиционных планов. Для демонстрации работы MAP-планирощика приведен модельный пример составления плана для одной из задач «мира кубиков». Программная реализация и модельные эксперименты представлены в репозитории [https://github.com/cog-isa/map-planner.](https://github.com/cog-isa/map-planner)

# Список литературы

1. *Barsalou L. W.* Perceptual symbol systems // The Behavioral and brain sciences. — 1999. — Т. 22, № 4. — 577—609, discussion 610—660.
2. *Barto A. G.*, *Bradtke S. J.*, *Singh S. P.* Learning to act using real-time dynamic programming // Artificial Intelligence. — 1995. — Т. 72, № 1—2. — С. 81—138.
3. *Bonet B.*, *Geffner H.* Solving POMDPs: RTDP-Bel vs. point-based algorithms // IJCAI International Joint Conference on Artificial Intelligence. — 2009. — С. 1641—1646.
4. *Borrajo D.*, *Roubíčková A.*, *Serina I.* Progress in Case-Based Planning // ACM Computing Surveys. — 2015. — Янв. — Т. 47, № 2. — С. 1—39.
5. *Brafman R. I.* A Privacy Preserving Algorithm for Multi-Agent Planning and Search // Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015). — 2015. — С. 1530—1536.
6. *Buxhoeveden D. P.*, *Casanova M.* The minicolumn hypothesis in neuroscience // Brain. — 2002. — Т. 125, № 5. — С. 935—951.
7. *Chella A.*, *Frixione M.*, *Gaglio S.* Anchoring symbols to conceptual spaces: The case of dynamic scenarios // Robotics and Autonomous Systems. — 2003. — Т. 43, № 2—3. — С. 175—188.
8. *Constantinople C. M.*, *Bruno R. M.* Deep cortical layers are activated directly by thalamus // Science. — 2013. — Т. 1591, June. — С. 1591—1594.
9. *De La Rosa T.*, *Garcia-Olaya A.*, *Borrajo D.* A case-based approach to heuristic planning // Applied Intelligence. — 2013. — Т. 39, № 1. — С. 184—201.
10. Deterministic planning in the fifth international planning competition: PDDL3 and experimental evaluation of the planners / A. E. Gerevini [и др.] // Artificial Intelligence. — 2009. — Т. 173, № 5—6. — С. 619—668.
11. *Edelman G. M.* Neural Darwinism: The Theory Of Neuronal Group Selection. — New York : Basic Books, 1987. — С. 400.
12. *Evans J.*, *Stanovich K. E.* Dual-process theories of higher cognition: Advancing the debate // Perspectives on Psychological Science. — 2013. — Т. 8, № 3. — С. 223—241.
13. *Flavell J. H.* Metacognition and cognitive monitoring: A new area of cognitive–developmental inquiry // American Psychologist. — 1979. — Т. 34,

№ 10. — С. 906—911.

1. *George D.*, *Hawkins J.* Towards a mathematical theory of cortical micro-circuits // PLoS computational biology. — 2009. — Т. 5, № 10. — e1000532.
2. *Gupta N.*, *Nau D. S.* On the complexity of Blocks-World planning // Artificial Intelligence. — 1992. — Т. 56, № 2—3. — С. 223—254.
3. *Hammond K. J.* Case-based planning: A framework for planning from experience // Cognitive Science. — 1990. — Т. 14, № 3. — С. 385—443.
4. *Harnad S.* Symbol Grounding Problem // Physica. — 1990. — Т. 42. — С. 335—346. — arXiv: [\FontspecSetCheckBoolFalse9906002 [arXiv:cs.AI].](http://arxiv.org/abs/9906002)
5. *Heintz F.*, *Kvarnstrom J.*, *Doherty P.* Bridging the sense-reasoning gap: DyKnow

- Stream-based middleware for knowledge processing // Advanced Engineering Informatics. — 2010. — Т. 24, № 1. — С. 14—26.

1. *Helmert M.* The fast downward planning system // Journal of Artificial Intelligence Research. — 2006. — Т. 26. — С. 191—246. — arXiv: \FontspecSetCheckBoolFalsearXiv:1109.6051v1.
2. *Hoffmann J.*, *Nebel B.* The FF Planning System: Fast Plan Generation Through Heuristic Search // Journal of Artificial Intelligence Research. — 2001. — Т. 14. — С. 253—302.
3. *Kahneman D.* Thinking Fast and Slow. — New York : Penguin, 2011. — С. 443.
4. *Kelley T. D.* Developing a psychologically inspired cognitive architecture for robotic control: The Symbolic and Subsymbolic Robotic Intelligence Control System (SS- RICS) // International Journal of Advanced Robotic Systems. — 2006. — Т. 3, № 3. — С. 219—222.
5. *Krajca P.*, *Outrata J.*, *Vychodil V.* Advances in algorithms based on CbO // Proceedings of the 7th International Conference on Concept Lattices and Their Applications / под ред.

M. Kryszkiewicz, S. Obiedkov. — CEUR, 2010. — С. 325—337.

1. *Merwe D. V. D.*, *Obiedkov S.*, *Kourie D.* AddIntent: A new incremental algorithm for constructing concept lattices // Concept Lattices / под ред. P. Eklund. — Springer Berlin Heidelberg, 2004. — С. 372—385. — (Lecture Notes in Computer Science).
2. *Neisser U.* Cognition and Reality: Principles and Implications of Cognitive Psychology. — W. H. Freeman, Company, 1976. — С. 230.
3. *Norris E. M.* An Algorithm for Computing the Maximal Rectangles in a Binary Relation // Revue Roumaine de Mathématiques Pures et Appliquées. — 1978. — Т. 23, № 2. — С. 243—250.
4. Practical intelligence in everyday life / R. J. Sternberg [и др.]. — Cambridge University Press, 2000. — С. 304.
5. *Red’ko V.*, *Burtsev M.* Modeling of Mechanism of Plan Formation by New Caledonian Crows // Procedia Computer Science. — 2016. — Т. 88.
6. *Richter S.*, *Westphal M.* The LAMA planner: Guiding cost-based anytime planning with landmarks // Journal of Artificial Intelligence Research. — 2010. — Т. 39. — С. 127—177. — arXiv: [\FontspecSetCheckBoolFalse1401.3839.](http://arxiv.org/abs/1401.3839)
7. *Sardina S.*, *Silva L. D.*, *Padgham L.* Hierarchical Planning in BDI Agent Programming Languages: A Formal Approach // Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems (AAMAS’06). — 2006. — С. 1001—1008.
8. *Skrynnik A.*, *Petrov A.*, *Panov A. I.* Hierarchical Temporal Memory Implementation with Explicit States Extraction // Biologically Inspired Cognitive Architectures (BICA) for Young Scientists / под ред. A. V. Samsonovich, V. V. Klimov, G. V. Rybina. — Springer International Publishing, 2016. — С. 219—225. — (Advances in Intelligent Systems and Computing).
9. *Stanovich K. E.* Distinguishing the reflective, algorithmic, and autonomous minds: Is it time for a tri-process theory? // In two minds: Dual processes and beyond / под ред. J. Evans, K. Frankish. — Oxford University Press, 2009. — С. 55—88.
10. *Sun R.*, *Hélie S.* Psychologically realistic cognitive agents: taking human cognition seriously // Journal of Experimental & Theoretical Artificial Intelligence. — 2012. — Т. 25, № 1. — С. 65—92.
11. *Выготский Л. С.* Мышление и речь. — Изд. 5-е, — М. : Издательство ”Лабиринт”, 1999. — С. 352.
12. *Иваницкий А. М.* Мозговая основа субъективных переживаний: гипотеза информационного синтеза // Журнал высшей нервной деятельности. — 1996. — Т. 46, № 2. — С. 241—282.
13. *Леонтьев А. Н.* Деятельность. Сознание. Личность. — Изд. 2-е. — М. : Политиздат, 1977. — С. 304.
14. *Макаров Д. А.*, *Панов А. И.*, *Яковлев К. С.* Архитектура многоуровневой интеллектуальной системы управления беспилотными летательными аппаратами // Искусственный интеллект и принятие решений. — 2015. — № 3. — С. 18—33.
15. *Осипов Г. С.* Знаковые модели как альтернатива символьным // Гибридные и синергетические интеллектуальные системы: Материалы III Всероссийской Поспеловской конференции с международным участием / под ред. А. В. Колесников. — Издательство БФУ им. Иммануила Канта, 2016. — С. 56—69.
16. *Осипов Г. С.* Когнитивное метамоделирование // Информационные технологии и системы: Труды Четвертой международной научной конференции / под ред. Ю. С. Попков, А. В. Мельников. — Челябинск : Изд-во Челяб. гос. ун–та, 2015. — С. 94—100.
17. *Осипов Г. С.* От ситуационного управления к прикладной семиотике // Новости искусственного интеллекта. — 2002. — № 6. — С. 3—7.
18. *Осипов Г. С.*, *Панов А. И.*, *Чудова Н. В.* Управление поведением как функция сознания. I. Картина мира и целеполагание // Известия Российской академии наук. Теория и системы управления. — 2014. — № 4. — С. 49—62.
19. *Осипов Г. С.*, *Панов А. И.*, *Чудова Н. В.* Управление поведением как функция сознания. II. Синтез плана поведения // Известия Российский академии наук. Теория и системы управления. — 2015. — № 6. — С. 47—61.
20. *Осипов Г. С.*, *Поспелов Д. А.* Прикладная семиотика // Новости искусственного интеллекта. — 1999. — № 1. — С. 9—35.
21. *Панов А. И.* Алгебраические свойства операторов распознавания в моделях зрительного восприятия // Машинное обучение и анализ данных. — 2014. — Т. 1,

№ 7. — С. 863—874.

1. *Панов А. И.* Представление знаний автономных агентов, планирующих согласованные перемещения // Робототехника и техническая кибернетика. — 2015. — 4(9). — С. 34—40.
2. *Панов А. И.*, *Петров А. В.* Иерархическая временная память как модель восприятия и её автоматное представление // Шестая Международная конференция

«Системный анализ и информационные технологии» САИТ-2015 (15-20 июня 2015 г., г. Светлогорск, Россия): Труды конференции. В 2-х т. Т. 1. — М. : ИСА РАН, 2015. — С. 198—202.

1. *Панов А. И.*, *Яковлев К. С.* Взаимодействие стратегического и тактического планирования поведения коалиций агентов в динамической среде // Искусственный интеллект и принятие решений. — 2016. — № 4. — (В печати).
2. Принципы построения многоуровневых архитектур систем управления беспилотными летательными аппаратами / Д. В. Зубарев [и др.] // Авиакосмическое приборостроение. — 2013. — № 4. — С. 10—28.
3. *Чувгунова О. А.* Планирование как предмет психологического исследования // Психологические исследования. — 2015. — Т. 8, № 43. — С. 11.
4. *Чудова Н. В.* Концептуальная модель картины мира для задачи моделирования поведения, основанного на сознании // Искусственный интеллект и принятие решений. — 2012. — № 2. — С. 51—62.

# Список иллюстраций

* 1. [Визуальная интерпретация каузальной матрицы](#_bookmark57) 22
  2. [Схема каузальной матрицы](#_bookmark58) 23
  3. [Схема каузальной сети. Здесь каузальные матрицы изображены в виде](#_bookmark59) [квадратов, столбцы условий - левая белая часть квадрата, столбцы](#_bookmark59) [эффектов - черная правая часть квадратов. Метка *E*1 отображается в](#_bookmark59) [начале каждой стрелки, метка *E*2 определяется как номер квадрата, к](#_bookmark59) [которому идет стрелка, а метка *E*3 отображается в конце каждой стрелки.](#_bookmark59) 24
  4. [Схема процесса планирования поведения](#_bookmark60) 25
  5. [Начальная ситуация: все четыре кубика лежат на столе.](#_bookmark61) 26
  6. [Целевая ситуация: башня из четырех кубиков.](#_bookmark62) 27
  7. [Фрагмент каузальной сети на значениях: представление действия *stack*.](#_bookmark63) . 28
  8. [Распространение активности по сети значений, генерация личностного](#_bookmark64) [смысла знака *unstack*](#_bookmark64) 29

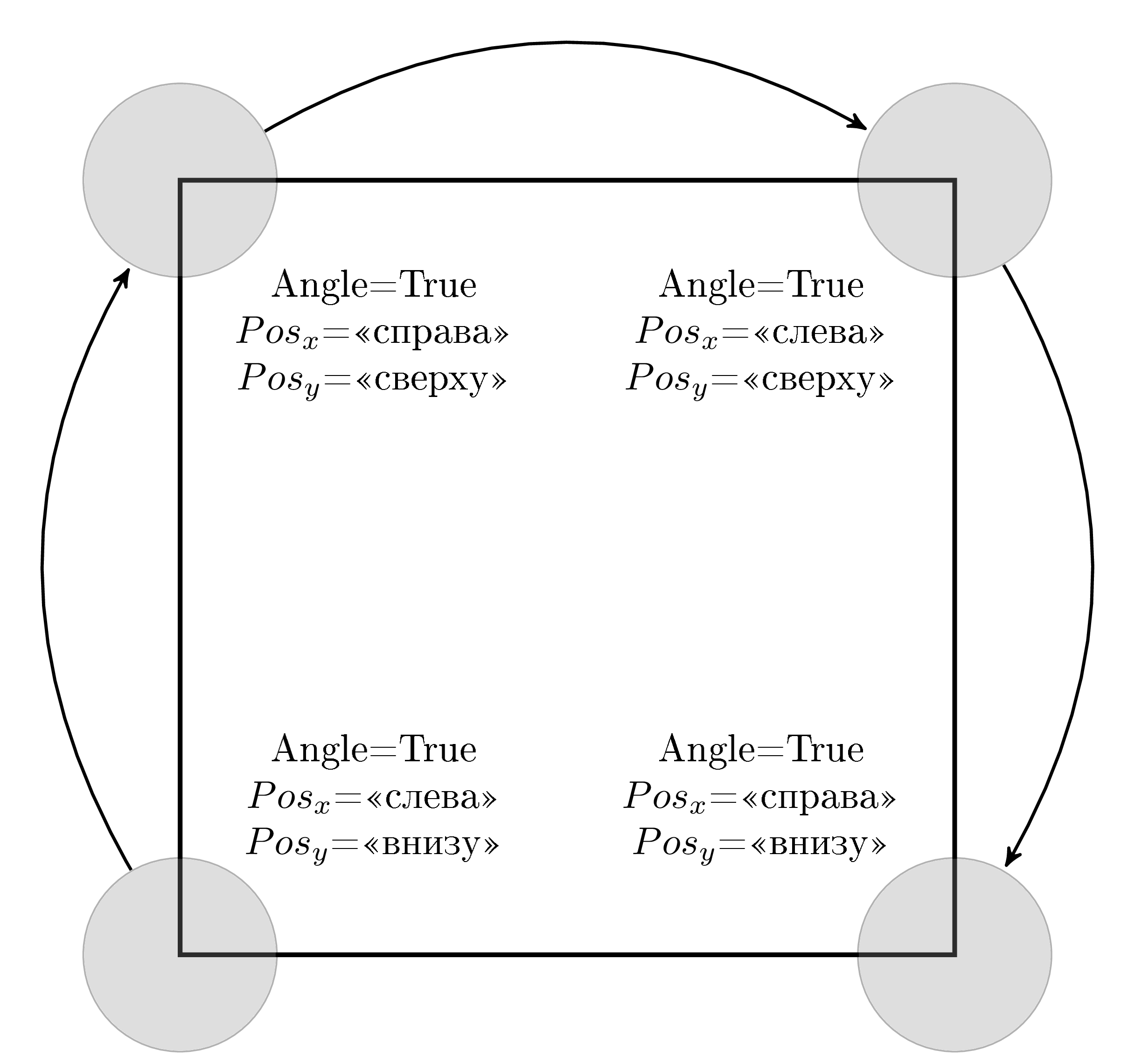


Рис. 1: Визуальная интерпретация каузальной матрицы

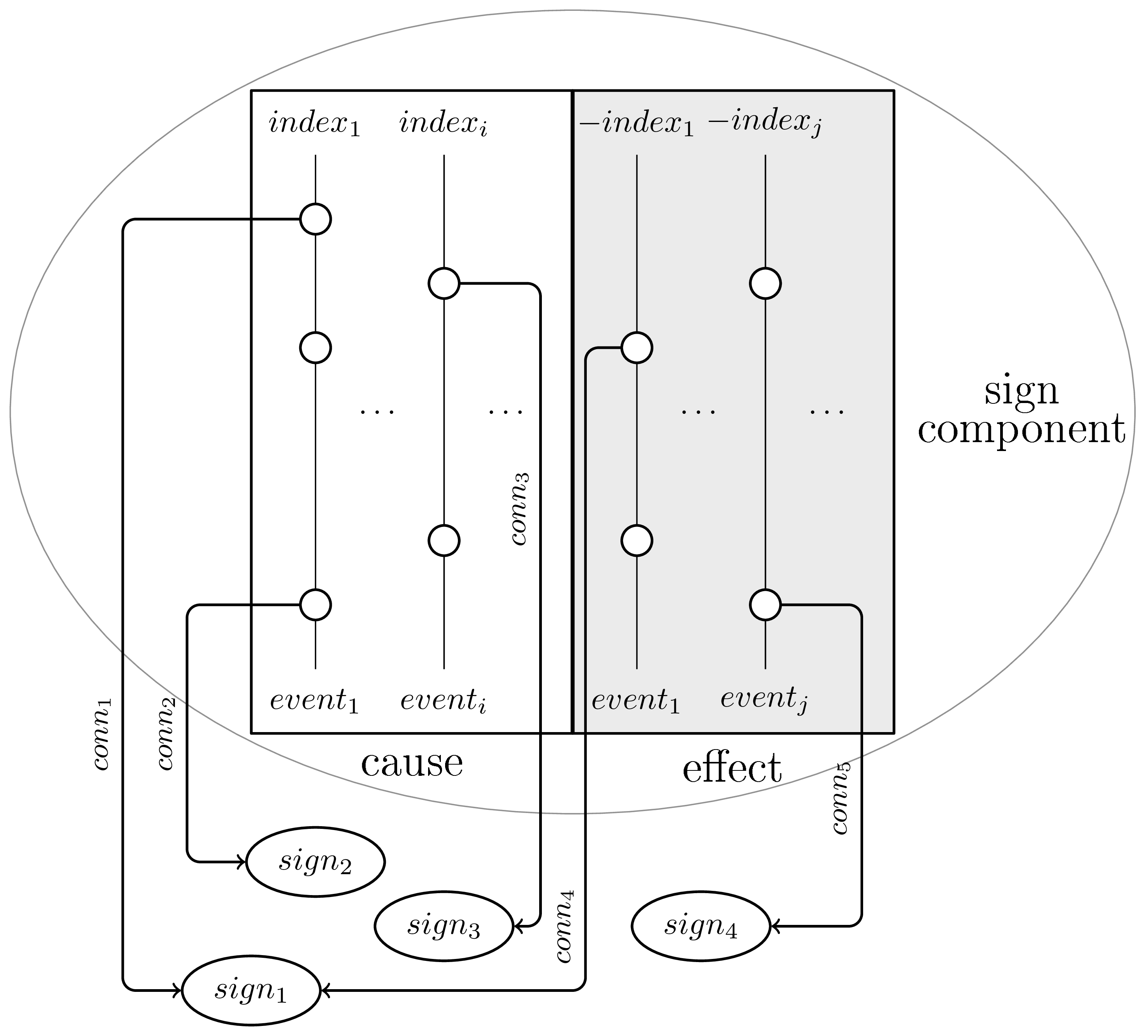


Рис. 2: Схема каузальной матрицы

✏ ✑

✶

✶

✏ ✑

✶

✷

✹

✸

✶

✶

✶

✶

✏

✑

✏

✑

✏ ✑

✏

✑

Рис. 3: Схема каузальной сети. Здесь каузальные матрицы изображены в виде квадратов, столбцы условий - левая белая часть квадрата, столбцы эффектов - черная правая часть квадратов. Метка *E*1 отображается в начале каждой стрелки, метка *E*2 определяется как номер квадрата, к которому идет стрелка, а метка *E*3 отображается в конце каждой стрелки.

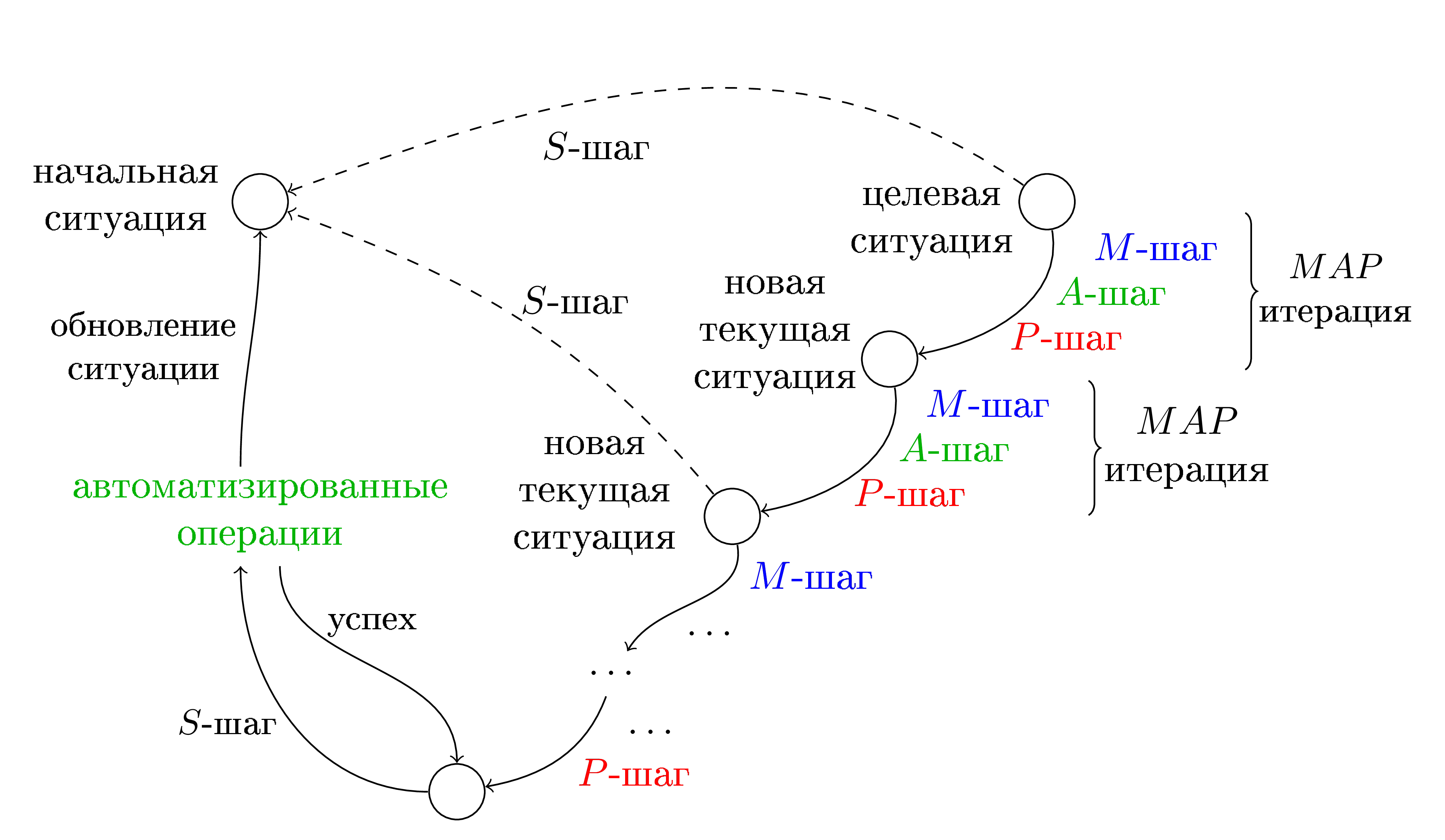


Рис. 4: Схема процесса планирования поведения

“clear”

“ontable”

1 2 3 4

2

4

3

5

2

4 6 8

7

1

3

1

2

1

“start”

3

8

7

1

5 6

4

1

1

1

1 1

1 1

“a” “b” “c” “d”



a

b

c

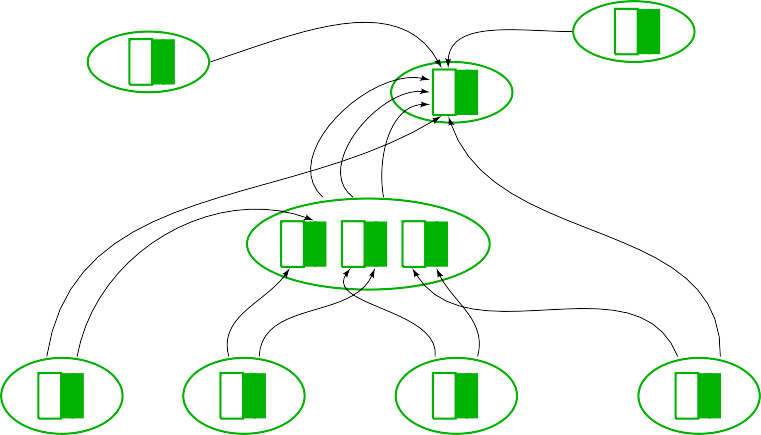
d

Рис. 5: Начальная ситуация: все четыре кубика лежат на столе.

“ontable”

1 5 4 1

3



2 1 “goal”

3 2

1 5

|  |  |  |
| --- | --- | --- |
|  | a |  |
| b |
| c |
| d |

“clear”

4 -1

“on”

1 1 -1

1

1 -1 1 1

1

1 1

“a”

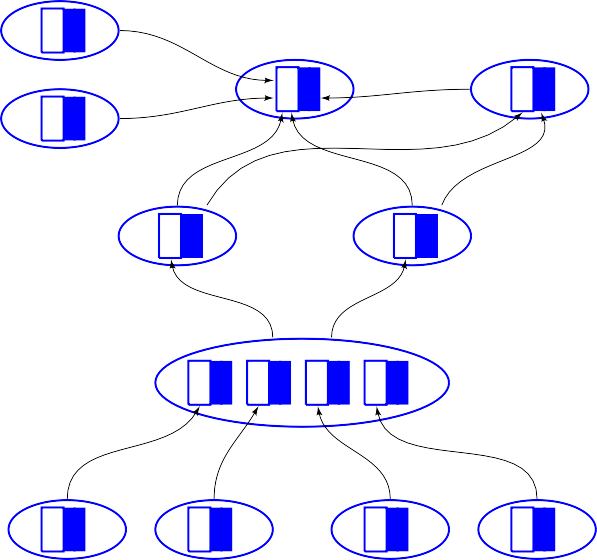
1 1 “b” “c” “d”

Рис. 6: Целевая ситуация: башня из четырех кубиков.

“holding” “clear”

1

2



1

1

1

1 1

“block?x”

“stack”

-1 1

1

2 1 1

“on”

a

-1

1 1

0 0

“block”

1 1 1

1

1 1 1

“block?y”

1



b

c

d

“a” “b” “c” “d”

Рис. 7: Фрагмент каузальной сети на значениях: представление действия *stack*.

“on”

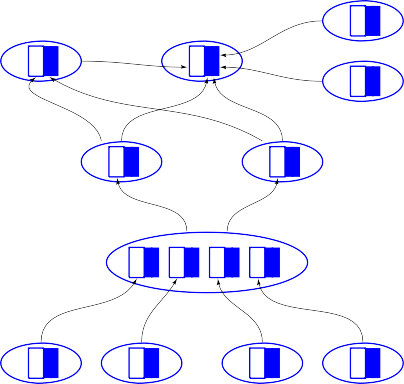
1

1 -1

1

“unstack” 1

-1



1 -2

1

-1 -2

1 1 1

“holding” “clear”

“on”

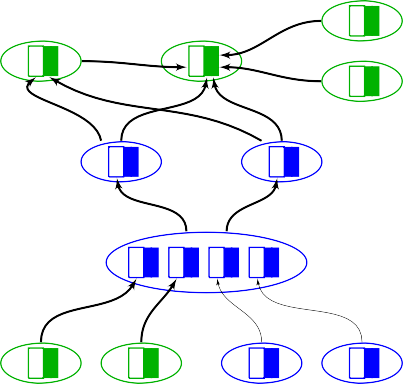
1

1 -1

1

“unstack” 1

-1



1 -2

1

-1 -2

1 1 1

“holding” “clear”

“block?x” “block?y”

“block?x” “block?y”

1 1

0 0

“block”

1 1

0 0

“block”

1 1 1

1 1 1 1 1

1 1 1

1 1 1 1 1

“a” “b”

“c” “d”

“a” “b”

“c” “d”

Рис. 8: Распространение активности по сети значений, генерация личностного смысла знака *unstack*.

# Список таблиц

[1 Описание домена планирование «мир кубиков» и задачи построения башни.](#_bookmark65) 31

|  |  |  |
| --- | --- | --- |
| (define (**domain BLOCKS**)  (:requirements :strips :typing) (:types block)  (:predicates (on ?x - block ?y - block)  (ontable ?x - block) (clear ?x - block) (handempty)  (holding ?x - block) ) | (:**action pick-up**  :parameters (?x - block)  :precondition (and (clear ?x) (ontable ?x) (handempty))  :effect  (and (not (ontable ?x)) (not (clear ?x))  (not (handempty)) (holding ?x))) | (:**action put-down**  :parameters (?x - block)  :precondition (holding ?x)  :effect  (and (not (holding ?x)) (clear ?x)  (handempty) (ontable ?x))) |
| (:**action stack**  :parameters (?x - block ?y - block)  :precondition (and (holding ?x) (clear ?y))  :effect  (and (not (holding ?x)) (not (clear ?y))  (clear ?x) (handempty) (on ?x ?y))) | (:**action unstack**  :parameters (?x - block ?y - block)  :precondition (and (on ?x ?y) (clear ?x) (handempty))  :effect  (and (holding ?x) (clear ?y)  (not (clear ?x)) (not (handempty)) (not (on ?x ?y))))) | (define (**problem BLOCKS-4-0**)  (:domain BLOCKS) (:objects D B A C - block)  (:INIT (CLEAR C) (CLEAR A) (CLEAR B) (CLEAR D) (ONTABLE C) (ONTABLE A) (ONTABLE B) (ONTABLE D) (HANDEMPTY))  (:goal (AND (ON D C) (ON C B)  (ON B A))) ) |

Таблица 1: Описание домена планирование «мир кубиков» и задачи построения башни.