Автоматическая генерация компьютерной программы, моделирующей нормативно-правовой акт

В. Б. Кудрявцев, Э. Э. Гасанов, Е. М. Перпер

В работе рассматривается задача автоматического распознавания текстов законов, касающихся заполнения форм отчётности. Эта задача состоит в следующем: создать компьютерную программу, которая принимала бы на вход тексты законов, а на выходе выдавала бы другую программу, осуществляющую заполнение формы в соответствии с этими законами. Предложен метод решения этой задачи, основанный на технологии компьютерного моделирования логических процессов.

Ключевые слова: семантический анализ текстов, нормативно-правовые акты, компьютерное моделирование логических процессов.

1. Введение

Одной из ключевых задач теории интеллектуальных систем является задача понимания смысла текстов на естественном языке. Если предположить, что у нас есть некоторая процедура выделения смысла текста, то встаёт проблема: как проверить, что смысл выделен правильно? В принципе, для этой цели можно привлечь эксперта, и положиться на его субъективное мнение, но во многих случаях разные эксперты по разному будут понимать смысл текста. Чтобы этого избежать, предлагается взять для анализа такие тексты, которые разными экспертами должны пониматься одинаково. В качестве такого класса текстов предлагается использовать тексты законов, поскольку априори они одинаково должны пониматься всеми людьми. В на-

шей статье мы ограничиваемся рассмотрением нормативного акта о бухгалтерском учёте [1]. Этот нормативный акт удобен тем, что он касается заполнения некоторых форм бухгалтерской отчётности. Поэтому можно считать, что смыслом данного нормативного акта является некоторая процедура правильного заполнения соответствующих форм отчётности. Следовательно, задачу понимания смысла такого рода текстов можно переформулировать следующим образом: по тексту нормативного акта автоматически сгенерировать компьютерную программу, которая бы заполняла формы отчётности в точном соответствии с требованиями данного нормативного акта. Такого рода программы по заполнению форм отчётности повсеместно используются бухгалтерскими подразделениями организаций, и они пишутся вручную фирмами, разрабатывающими ERP-системы (Enterprise Resource Planning — Управление ресурсами предприятия) [2, 3, 4, 5]. А поскольку законы в России изменяются довольно часто (так, рассматриваемый нами нормативный акт в течение 10 лет претерпел 6 редакций), то задача автоматической генерации по тексту закона компьютерной программы, исполняющей закон, является очень актуальной.

Кафедра МаТИС имеет некоторый опыт в области анализа текстов на русском языке. Так, в решателе математических задач [6], разработанном профессором кафедры МаТИС А. С. Подколзиным, есть раздел, посвященный решению текстовых задач. В частности, в этом разделе решателем осуществляются синтаксический и семантический анализ текста задачи. Пример решателя показывает, что для семантического анализа текстов на естественном языке может успешно использоваться технология, разработанная профессором А. С. Подколзиным, — технология компьютерного моделирования логических процессов. В основе этой технологии лежит понятие «приёма». Каждый прием описывает некоторое преобразование решаемой задачи и состоит из двух частей: условий применимости преобразования и собственно самого преобразования. Тем самым каждый прием сам решает стоит ему применяться или нет. Существует также система приоритета приемов, которая регулирует очередность их срабатывания.

Задачу построения по тексту некоторого закона о бухгалтерском учёте программы, заполняющей соответствующие формы отчётности, предлагается решать в несколько этапов. Первым этапом явля-

ется создание для каждого предложения рассматриваемого текста его семантического графа — графа связей между словами в предложении (подробнее о семантических графах см., например, [7]). На втором этапе каждому семантическому графу сопоставляется логическая формула, которая определяет условия, накладываемые на рассматриваемые в тексте сущности. На третьем этапе каждой переменной логической формулы сопоставляется некоторая сущность или свойство сущности. Наконец, на последнем по всем формулам вместе строится программа, которая в соответствии с накладываемыми на каждую сущность условиями заполняет соответствующие этой сущности поля в форме отчётности. Технология компьютерного моделирования логических процессов используется нами на всех четырех этапах решения данной задачи.

2. Основные понятия и формулировка результатов

Задача, о которой идёт речь в статье, формулируется так: построить программу, на вход которой поступают текст нормативного акта, определяющего правила заполнения некоторой формы отчётности, а результат работы этой программы — программа, спрашивающая у пользователя данные, необходимые и достаточные для заполнения формы, а затем заполняющая эту форму. В качестве программы, которую нужно получить, будем рассматривать объект, называемый *моделью закона*. Этот объект мы определим следующим образом.

Рассмотрим ориентированный граф, каждой вершине которого сопоставлена некоторая процедура одного из следующих видов:

1) получение значения из конкретного поля в памяти. Считается, что это значение передаётся по всем рёбрам, выходящим из такой вершины;

2)запись значения в конкретное поле в памяти. В такую вершину должно вести единственное ребро. Считается, что записываемое значение поступает в данную вершину по этому ребру;

3) вычисление некоторой арифметической функции одного или двух аргументов, логической функции одного или двух аргументов либо унарного или бинарного отношения. В такую вершину должно входить столько рёбер, сколько аргументов у функции или отноше-

ния. По каждому из этих рёбер в вершину поступает значение соответствующего аргумента функции (отношения). Значение функции (отношения) передаётся по всем рёбрам, выходящим из этой вершины.

4) выбор одного значения из нескольких. В эту вершину для некоторого натурального числа $m \geqslant 2$ должно вести $m + 1$ ребро. Эти рёбра должны быть пронумерованы числами от 0 до m. По ребру с номером m в вершину поступает число $i \in \mathbb{Z}$, $0 \leqslant i \leqslant m - 1$. По каждому ребру, выходящему из вершины, передаётся значение, поступившее в вершину по ребру с номером i.

Будем относить вершину к i-му виду вершин, $i \in \{1, 2, 3, 4\}$, если ей сопоставлена процедура i-го вида.

Из принадлежащих графу вершин 2-го вида выделим одну. Графу сопоставляется процедура его вычисления: для выделенной вершины вычисляется значение, поступающее в неё по единственному входящему ребру. Это значение вычисляется с помощью процедуры, сопоставленной вершине, из которой выходит соответствующее ребро.

Для вершины 1-го вида значение, передаваемое по ребру, выходящему из этой вершины, либо извлекается из базы данных предприятия, либо вводится бухгалтером с консоли.

Для вершины 3-го вида сначала вычисляется значение, поступающее в вершину по одному из входящих в неё рёбер. Если этого недостаточно для вычисления значения функции (отношения), то вычисляется значение, поступающее в вершину по другому ребру. В любом случае, дальше вычисляется значение функции (отношения) на полученных значениях аргументов, это значение и передаётся по выходящим из вершины рёбрам.

Для вершины 4-го вида сначала вычисляется число i, поступающее в вершину по входящему в него ребру с максимальным номером, затем вычисляется значение, поступающее в вершину по ребру номер i. Это значение и передаётся по выходящему из вершины ребру.

Если результат процедуры вычисления графа при любых исходных значениях полей совпадает с результатом заполнения соответствующего поля в форме отчётности в соответствии с текстом закона, назовём этот граф моделью вычисления поля. Граф, содержащий в себе в качестве подграфа модель вычисления любого поля из формы отчётности, назовём моделью закона.

Построение модели закона по тексту закона предлагается разбить на 4 этапа.

На первом этапе проводится семантический анализ текста, в результате которого для каждого предложения текста строится семантический граф этого предложения. Данный этап в работе подробно не рассматривается, так как программы, осуществляющие семантический анализ текста на русском языке, уже существуют, например, созданная на проекте «Диалинг» (см. [7]).

На втором этапе по семантическому графу каждого предложения закона строится логическая формула, соответствующая этому предложению. Например, из предложения «в случаях улучшения (повышения) первоначально принятых нормативных показателей функционирования объекта основных средств в результате проведенной реконструкции или модернизации организацией пересматривается срок полезного использования по этому объекту» (см. [1]) может получиться формула $((B \vee C)\&(D \vee E)) \to A$, где каждая переменная соответствует некоторой сущности (в данном случае — булевой сущности): A соответствует сущности «организацией пересматривается срок полезного использования объекта основных средств», B — «(произошло) улучшение первоначально принятых нормативных показателей функционирования объекта основных средств», C — «(произошло) повышение первоначально принятых нормативных показателей функционирования объекта основных средств», D — «проведена реконструкция (объекта основных средств)», E — «проведена модернизация (объекта основных средств)».

На третьем этапе каждой переменной логической формулы сопоставляется некоторая сущность или свойство сущности. Эти сущности извлекаются вообще говоря из текста всего закона, а не только данного предложения.

На четвертом этапе из всех формул, полученных из предложений закона, создаётся модель закона. При построении модели учитывается, что каждая из рассматриваемых формул должна быть истинна при любых допустимых значениях переменных формулы (так как каждое предложение закона мы считаем истинным), откуда вытекает зависимость переменных формулы (а значит, некоторых сущностей) от других переменных формулы (других сущностей).

В работе для каждого этапа решения задачи приведены приёмы, с помощью которых этот этап осуществляется. Приведённых приё-

мов достаточно, чтобы построить фрагмент модели положения ПБУ 6/01 (см. [1]) по пунктам этого закона, касающимся годовой суммы амортизационных отчислений по объектам основных средств, если семантический анализ текста этих пунктов уже проведён. Дальнейшие исследования будут касаться накопления большего числа приёмов, что позволит строить модели различных нормативно-правовых актов.

3. Построение упрощённого семантического графа

Рассмотрим следующее предложение, являющееся незначительно изменённой частью одного из предложений пункта 19 положения ПБУ 6/01 (см. [1]): «при способе уменьшаемого остатка годовая сумма амортизационных отчислений определяется исходя из остаточной стоимости объекта основных средств на начало отчетного года и нормы амортизации, исчисленной исходя из срока полезного использования этого объекта и коэффициента не выше трёх, установленного организацией».

Предположим мы уже получили семантический граф, соответствующий этому предложению, например, с помощью программы [7], и семантический граф этого предложения изображён на рисунке 1.

Аббревиатурой «МНА» на рисунке 1 в согласии с [7] обозначен множественный актант.

Следующим шагом мы из семантического графа получим упрощённый семантический граф с помощью приёмов, объединяющих несколько вершин графа в одну, соответствующую, как правило, некоторому полю базы данных.

На применение нескольких приёмов наложены некоторые ограничения. Эти приёмы не применяются, если:

а) вершине α или вершине β сопоставлено слово «больше», «меньше», «превышает», «выше» и т. д., обозначающее некоторый предикат;

б) среди служебных слов и сочетаний слов, сопоставленных вершине β, есть хотя бы одно из следующих: «по», «в течение», «в случаях», «при», «в результате», «на», «в», «в составе», «с».

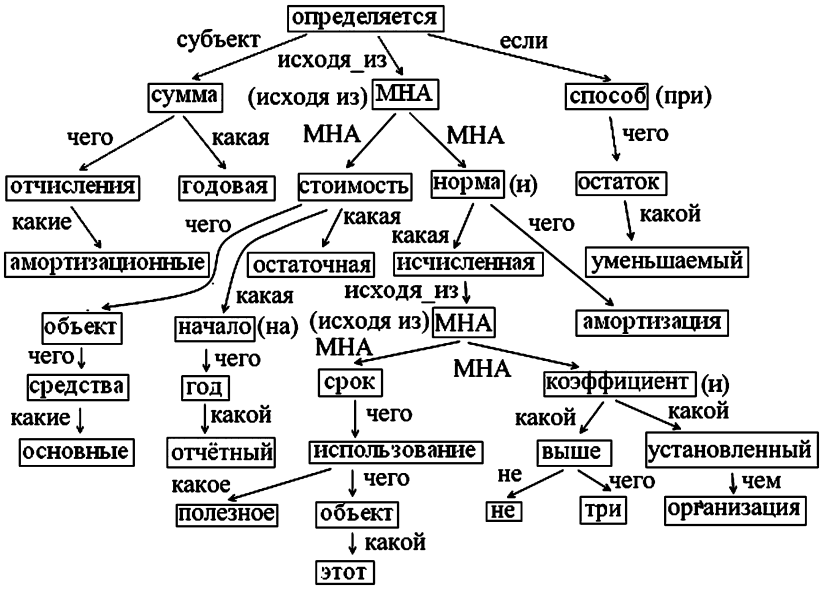

Рис. 1. Семантический граф.

Вот некоторые из приёмов построения упрощенного семантического графа.

1) Пусть из некоторой вершины α цепи выходит только одно ребро, ведущее в некоторую вершину β. Если не выполнено ни одно из условий «а» и «б», то вершины α и β объединяются в одну вершину, которой сопоставляется объединение фрагментов текста, соответствующих вершинам α и β.

2) Пусть из некоторой вершины α в некоторую вершину β ведёт ребро, соответствующее семантическому отношению «чего», «объект» или «субъект». Если из β не выходит ни одно ребро и не выполнено ни одно из условий «а» и «б», то вершины α и β объединяются в одну вершину, которой сопоставляется объединение фрагментов текста, соответствующих вершинам α и β.

3) Пусть «способ» — одно из слов, соответствующих некоторой вершине α, и пусть из α в некоторую вершину β ведёт ребро, соответствующее семантическому отношению «чего». Тогда все вершины, в которые можно перейти из β, вместе с вершинами α и β объединяются в одну вершину. Этой вершине сопоставляется объединение

фрагментов текста, соответствующих каждой из объединяемых вершин.

4) Объединение вершин, которым сопоставлены слова, формирующие некоторую сущность (список этих сущностей должен храниться отдельно). Например, объединяется вершина, которой соответствует слово «нормативные», и вершина, которой соответствует слово «показатели», если текстовые фрагменты, сопоставленные этим вершинам, идут в тексте подряд.

Применение приёма 1, 2 или 3 означает проверку для вершины всех условий, указанных в тексте приёма, и произведение в соответствии с приёмом некоторых изменений, если эти условия выполнены. Использование приёма 4 происходит иначе: если в тексте предложения закона встречается любая сущность из списка, то объединяются вершины, которым соответствуют фрагменты текста, формирующие эту сущность.

Алгоритм построения упрощённого семантического графа по исходному семантическому графу состоит в следующем. Сначала применяется приём 4, затем для каждой вершины применяется приём 3, пока все вершины не будут рассмотрены либо граф не изменится. Если в результате применения приёма 3 к какой-либо вершине граф изменился, этот приём применяется к каждой вершине построенного графа, пока все вершины не будут рассмотрены либо граф снова не изменится, и т. д. Далее по той же схеме применяются приёмы 1 и 2 вместе: если после применения к вершине приёма 1 граф не изменился, к этой же вершине применяется приём 2; если и после этого граф не изменился, рассматривается следующая вершина. Если же после применения приёма 1 или приёма 2 граф изменился, то приёмы 1 и 2 применяются по той же схеме к вершинам полученного графа.

В результате применения указанных приемов к семантическому графу, изображенному на рисунке 1, мы получим упрощённый семантический граф, изображенный на рисунке 2.

4. Построение логической формулы

Когда упрощённый семантический граф получен, начинается собственно построение логической формулы. При этом применяются следующие приёмы.

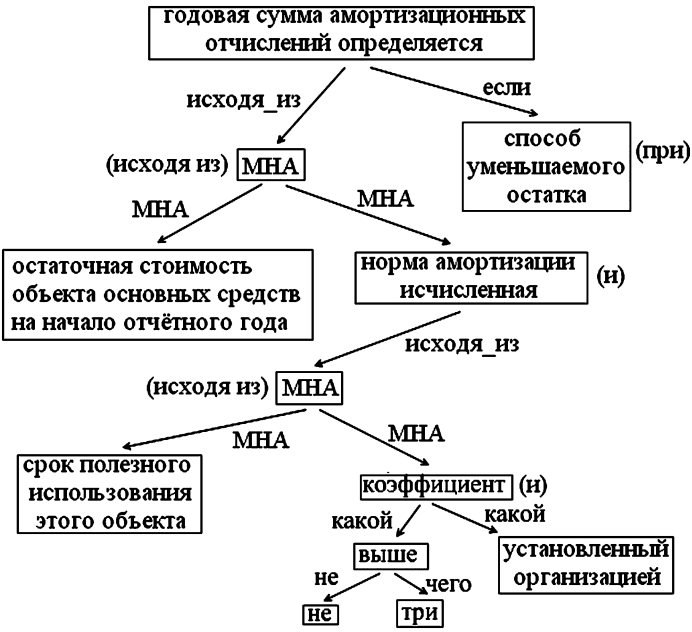

Рис. 2. Упрощённый семантический граф.

1) Если в начале предложения находится одно из следующих слов или сочетаний слов: «по», «в течение», «в случаях», «при», «для», то управляемая этим словом или сочетанием слов часть предложения определяет область применения статьи закона. Пусть это слово или сочетание слов сопоставлено вершине α. Вершину, из которой в α ведёт ребро, обозначим β. Пусть A — формула, соответствующая графу, состоящему из всех вершин, достижимых из α (будем считать, что каждая вершина достижима из себя), и соединяющих их рёбер, B — формула, соответствующая графу, состоящему из всех остальных вершин и соединяющих их рёбер. Итоговая формула будет иметь вид $(A \to B)$.

2) Если в предложении некоторая его часть подчинена действию с помощью слова «кроме», то это означает, что действие выполняется тогда и только тогда, когда не выполняется условие, выраженное частью предложения, управляемой словом «кроме». Пусть это слово сопоставлено вершине α. Вершину, из которой в α ведёт ребро, обозначим β. Пусть A — формула, соответствующая подграфу, состоя-

щему из всех вершин, достижимых из α (будем считать, что каждая вершина достижима из себя), и соединяющих их рёбер, B — формула, соответствующая графу, состоящему из всех остальных вершин и соединяющих их рёбер. Итоговая формула будет иметь вид $(\neg A \sim B)$.

3) Слово «не» означает логическое отрицание. Пусть это слово сопоставлено вершине α, в которую входит ребро, выходящее из вершины β. Если по вершине β построено логическое выражение A, то после применения приёма оно превращается в $(\neg A)$.

4) Слова «либо», «или» при перечислении однородных членов предложения, если речь не идёт о перечислении аргументов некоторой функции, означает логическое «или» либо функцию, истинную, когда истинно значение ровно одного её аргумента. Пусть слово «или» («либо») сопоставлено вершине α_1, в которую входит ребро, выходящее из вершины β, причём это ребро соответствует семантическому отношению «множественный актант» (это значит, что вершине β соответствует несколько однородных членов предложения, один из которых соответствует вершине α_1), а среди служебных слов и сочетаний слов, сопоставленных β, не встречается словосочетание «исходя из». Пусть $\alpha_2, \ldots, \alpha_r$ — остальные вершины, в которые из β ведёт ребро, соответствующее семантическому отношению «множественный актант» (r всегда будет не меньше 2). Возможны 2 случая.

4.1) Среди рассматриваемых вершин $\alpha_j, j \in \{1, \ldots, r\}$, найдётся вершина α_i такая, что из неё выходит ребро, соответствующее семантическому отношению «в случае». Пусть это ребро ведёт в вершину γ. Обозначим через A формулу, построенную по графу, состоящему из всех вершин, достижимых из γ, и соединяющих их рёбер; через B — формулу, построенную из всех остальных вершин, достижимых из α_i, и соединяющих их рёбер. Если осталась лишь одна вершина $\alpha_j, i \neq j$, то обозначим через C формулу, построенную по графу, состоящему из всех вершин, достижимых из α_j, и соединяющих их рёбер. Если же таких вершин α_j, что $j \neq i$, больше одной, то обозначим через C формулу, полученную применением 4.1 или 4.2 ко всем рассматриваемым вершинам $\alpha_s, s \in \{1, \ldots, r\}$, кроме α_i, которую мы исключим из рассмотрения. Тогда результатом применения приёма будет формула $(A\&B \vee \neg A\&C)$.

4.2) Среди рассматриваемых вершин $\alpha_j, j \in \{1, \ldots, r\}$, нет ни одной вершины α_i такой, что из неё выходит ребро, соответствующее семантическому отношению «в случае». Тогда, если обозначить че-

рез $A_j, j \in \{1, \ldots, r\}$, формулу, построенную по графу, состоящему из всех вершин, достижимых из α_j, то в результате применения приёма будет создана формула, являющаяся заключённой в скобки дизъюнкцией всех формул A_j, соответствующих рассматриваемым вершинам $\alpha_j, j \in \{1, \ldots, r\}$.

5) Союз «и» или отсутствие союзов при перечислении однородных членов предложения, если речь не идёт о перечислении аргументов некоторой функции, означает логическое «и» либо «или». Пусть из вершины β выходит $r \geqslant 2$ рёбер, которым соответствует семантическое отношение «множественный актант», и ведут эти рёбра в вершины $\alpha_1, \ldots, \alpha_r$, причём среди служебных слов и сочетаний слов, соответствующих этих вершинам, нет «или», «либо» и «а также». Пусть $A_j, j \in \{1, \ldots, r\}$ — формула, построенная по графу, состоящему из всех вершин, достижимых из α_j. Если для вершины β выполнены все те же условия, что и для вершины β из пункта 1, то результатом применения приёма будет формула $(A_1 \vee A_2 \vee \ldots \vee A_r)$. В противном случае в результате применения приёма будет создана формула $(A_1 \& A_2 \& \ldots \& A_r)$.

6) Слова «исходя из» означают некоторую функцию. Пусть эти слова сопоставлены вершине γ, а β — вершина, из которой в β идёт ребро. Пусть B — переменная, обозначающая объект, определённый текстовым фрагментом, соответствующим вершине β. Возможно несколько случаев:

6.1) Вершине γ не сопоставлен множественный актант. Пусть C — переменная, обозначающая объект, определённый текстовым фрагментом, соответствующим вершине γ. Тогда в результате применения приёма графу, состоящему из вершин β и γ и соединяющего их ребра, будет поставлена в соответствие формула $((B = f_\beta(C)) \& G)$, где f_β — некоторая функция, в тексте закона не определённая явно, G — формула, полученная путём применения приёма 6 или приёма 8 к вершине γ, если она удовлетворяет условиям применения хотя бы одного из этих приёмов (не существует вершин, к которым одновременно применимы приёмы 6 и 8), и тождественная истина иначе.

6.2) Вершине γ сопоставлен множественный актант. Пусть из вершины γ выходит $r \geqslant 2$ рёбер, которым соответствует семантическое отношение «множественный актант», и ведут эти рёбра в вершины $\alpha_1, \ldots, \alpha_r$, причём среди служебных слов и сочетаний слов, соответствующих этих вершинам, нет «или», «либо» и «а также».

Пусть $A_j, j \in \{1, \ldots, r\}$ — переменная, обозначающая объект, определённый текстовым фрагментом, соответствующим вершине α_j, либо поставленная в соответствие вершине α_j вспомогательная переменная, если α_j соответствует множественный актант. Тогда в результате применения приёма графу, состоящему из вершин $\beta, \gamma, \alpha_1, \ldots, \alpha_r$ и соединяющих их рёбер, будет поставлена в соответствие формула $((B = f_\beta(A_1, A_2, \ldots, A_r))\&D_1\& \ldots \&D_r\&G_1\& \ldots \&G_r)$, где f_β — некоторая функция, в тексте закона не определённая явно. Если вершине α_j не сопоставлен множественный актант, то D_j — тождественная истина, иначе это соответствующая множественному актанту формула, строящаяся с помощью приёма из пункта 7, применённого к вершине α_j. G_j — формула, полученная путём применения приёма 6 или приёма 8 к вершине α_j, если она удовлетворяет условиям применения хотя бы одного из этих приёмов, и тождественная истина иначе.

6.3) Вершине γ сопоставлен множественный актант. Пусть из вершины γ выходит $r \geqslant 2$ рёбер, которым соответствует семантическое отношение «множественный актант», и ведут эти рёбра в вершины $\alpha_1, \ldots, \alpha_r$, причём среди служебных слов и сочетаний слов, соответствующих этих вершинам, есть «или» или «либо». Тогда в результате применения приёма графу, состоящему из вершин $\beta, \gamma, \alpha_1, \ldots, \alpha_r$ и соединяющих их рёбер, будет поставлена в соответствие формула $((B = f_\beta(C))\&D\&G)$, где f_β — некоторая функция, в тексте закона не определённая явно, C — поставленная в соответствие вершине γ вспомогательная переменная, D — формула, строящаяся с помощью приёма из пункта 7, применённого к вершине γ. G — формула, полученная путём применения приёма 6 или приёма 8 к вершине γ, если она удовлетворяет условиям применения хотя бы одного из этих приёмов, и тождественная истина иначе.

7) Пусть вершине β сопоставлен множественный актант, и либо среди служебных слов, соответствующих β, есть «исходя из», либо из вершины, которой соответствуют эти слова, можно попасть в β, переходя по рёбрам (в соответствии с их направлением), соответствующим семантическому отношению «множественный актант». Пусть также из вершины β выходит $r \geqslant 2$ рёбер, которым соответствует семантическое отношение «множественный актант», и ведут эти рёбра в вершины $\alpha_1, \ldots, \alpha_r$, причём среди служебных слов и сочетаний слов, соответствующих этих вершинам, есть «или» или «либо». Пусть B — поставленная в соответствие вершине β вспомогательная

переменная, $A_j, j \in \{1, \ldots, r\}$ — переменная, обозначающая объект, определённый текстовым фрагментом, соответствующим вершине α_j, либо поставленная в соответствие вершине α_j вспомогательная переменная. Пусть $D_j, j \in \{1, \ldots, r\}$ — тождественная истина, если вершине α_j не сопоставлен множественный актант; в противном случае это соответствующая множественному актанту формула, строящаяся с помощью приёма из пункта 7, применённого к вершине α_j. Пусть также G_j — формула, полученная путём применения приёма 6 или приёма 8 к вершине α_j, если она удовлетворяет условиям применения хотя бы одного из этих приёмов, и тождественная истина иначе. Возможны 2 случая:

7.1) Среди рассматриваемых вершин $\alpha_j, j \in \{1, \ldots, r\}$, найдётся вершина α_i такая, что из неё выходит ребро, соответствующее семантическому отношению «в случае». Пусть это ребро ведёт в вершину η. Обозначим через E формулу, построенную по графу, состоящему из всех вершин, достижимых из η, и соединяющих их рёбер. Если осталась лишь одна вершина $\alpha_j, i \neq j$, то результатом применения приёма будет формула $(E\&(B = A_i)\&D_i\&G_i \vee \neg E\&(B = A_j)\&D_j\&G_j)$. Если же таких вершин α_j, что $j \neq i$, больше одной, то обозначим через F формулу, полученную применением 7.1 или 7.2 ко всем рассматриваемым вершинам $\alpha_s, s \in \{1, \ldots, r\}$, кроме α_i. Тогда результатом применения приёма будет формула $(E\&(B = A_i)\&D_i\&G_i \vee \neg E\&F)$.

7.2) Среди рассматриваемых вершин $\alpha_j, j \in \{1, \ldots, r\}$, нет ни одной вершины такой, что из неё выходит ребро, соответствующее семантическому отношению «в случае». Пусть рассматриваемые вершины — $\alpha_{k_1}, \ldots, \alpha_{k_s}$, $s \in \mathbb{N}$, $s \geqslant 2$. Тогда в результате применения приёма будет создана формула $((B = A_{k_1})\&D_{k_1}\&G_{k_1} \vee \ldots \vee (B = A_{k_s})\&D_{k_s}\&G_{k_s})$.

8) Словам «больше», «меньше», «превышает», «выше» и т. д. в формуле будет соответствовать предикат «$>$» или «$<$». Пусть β — вершина, которой сопоставлено одно из этих слов, γ — вершина, из которой в β ведёт ребро. Пусть выходящее из β ребро, которому соответствует семантическое отношение «чего» (такое ребро должно найтись, причём ровно одно) ведёт в α. Пусть A — переменная, обозначающая объект, определённый текстовым фрагментом, соответствующим вершине α, C — переменная, обозначающая объект, определённый текстовым фрагментом, соответствующим вершине γ. Тогда, если β сопоставлено слово «меньше», «ниже», то графу, состояще-

му из вершин α, β, γ и соединяющих их рёбер, будет соответствовать формула $(C < A)$, в противном случае — формула $(C > A)$.

Применение приёма означает проверку для вершины всех условий, указанных в тексте приёма, и построение некоторой формулы, если эти условия выполнены. Предполагается, что один и тот же приём не может быть применён к одной и той же вершине (обозначенной во всех приёмах как β) более одного раза, за исключением случаев, когда приём обращается рекурсивно к себе.

Алгоритм построения формулы по графу, состоящему из всех вершин, достижимых из данной, и соединяющих их рёбер, состоит в следующем. Проверяется, является ли данная вершина вершиной β из приёмов 2–6 или 8. Если да, то применяется все приёмы, которые можно применить (в порядке возрастания номеров), притом применяемые приёмы могут вызвать этот же алгоритм для некоторых вершин, к которым из данной идёт ребро. Если ни один из приёмов 2–6, 8 нельзя применить, данной вершине сопоставляется переменная, обозначающая объект, определённый текстовым фрагментом, соответствующим этой вершине (исключением является вершина, которой сопоставлено слово «случай» или «случаи», и из которой выходит ровно одно ребро; в этом случае некоторой переменной сопоставляются и данная вершина, и та вершина, в которую из неё идёт ребро) Производится конъюнкция этой переменной и всех формул, построенных с помощью этого же алгоритма по вершинам, в которые из соответствующих данной переменной вершин идут рёбра.

Итоговый алгоритм построения формулы в соответствии с описанными приёмами следующий.

Если условия приёма 1 выполнены для какой-либо вершины (не должно быть более одной такой вершины), то в результате применения этого приёма исходный граф разбивается на 2 подграфа. В каждом из этих подграфов выбирается корень, и к ним поочерёдно применяется описанный выше алгоритм построения формулы по графу, состоящему из всех вершин, достижимых из данной.

Если условия приёма 1 не выполнены ни для какой вершины, то в графе выбирается корень, и к нему применяется описанный выше алгоритм построения формулы по графу, состоящему из всех вершин, достижимых из данной.

Результат применения алгоритма построения формулы к упрощённому семантическому графу, изображенному на рисунке 2, будет

иметь следующий вид (с точностью до опускания лишних скобок, переобозначения переменных и порядка следования операндов в коммутативных операциях):

$$A \to ((B = f_\alpha(C, D))\&((D = f_\beta(E, F))\&!(F > 3))), \qquad (1)$$

где переменной A соответствуют слова «способ уменьшаемого остатка», B — «годовая сумма амортизационных отчислений определяется», C — «остаточная стоимость объекта основных средств на начало отчётного года», D — «норма амортизации, исчисленная», E — «срок полезного использования этого объекта», F — «коэффициент», f_α и f_β — некоторые функции, не определённые в предложении явно.

5. Извлечение сущностей

Пусть по каждому предложению текста закона уже построена формула. Переменные этих формул соответствуют определённым текстовым фрагментам. Каждой переменной нужно сопоставить соответствующую сущность, причём некоторым переменным из разных формул может быть сопоставлена одна и та же сущность. Часто сущность полностью определяется текстовым фрагментом (предполагается, что если слова в двух фрагментах текста отличаются только своей формой, например, находятся в разных падежах, то эти два фрагмента определяют одну и ту же сущность). В некоторых случаях этого, однако, не происходит, поэтому на данном шаге применяется несколько приёмов, сопоставляющих переменной некоторую сущность. Сущность, являющаяся атрибутом x какой-либо другой сущности y, обозначается как $y.x$.

Вот некоторые из используемых приёмов.

1) Если в формуле переменная определяется как значение некоторой функции, и в соответствующий этой переменной фрагмент текста входит одно из следующих слов или сочетаний слов «определяется», «определение происходит», «исчисленный», то перечисленные слова удаляются из текстового фрагмента, после чего сущность определяется в соответствии с остальными приёмами.

2) Если текстовый фрагмент, соответствующий переменной, представляет собой действие y и его субъект x, то переменной ставится в соответствие сущность $x.y$.

3) Если текстовый фрагмент, соответствующий переменной, выглядит как «a объекта основных средств b» (или просто «a объекта»), где a, b — произвольные фрагменты текста, причём a непуст, то производится попытка применить приём 2, считая, что переменной сопоставлен текстовый фрагмент $z =''\ a\ b''$. Если приём 2 удаётся использовать, и в результате переменной сопоставляется сущность $x.y$, то переменной сопоставляется сущность «объект основных средств».$x.y$. Если приём 2 использовать не удаётся, то переменной сопоставляется сущность «объект основных средств».z.

4) Если ко всем вершинам графа, соответствующим данной переменной, на этапе построения формулы не был применён ни один из приёмов 2–6, 8, и среди этих вершин нет корня графа, то переменной сопоставляется сущность $x.y$, где y — соответствующий этим вершинам текстовый фрагмент, а x — сущность, соответствующая вершине графа, из которой в одну из рассматриваемых вершин идёт ребро (эта вершина не должна соответствовать рассматриваемой переменной).

5) Если правило 4 неприменимо, переменная сопоставлена ровно одной вершине α, ей соответствуют слова «коэффициент», «норма амортизации», и α — не корень графа, то рассматривается такая вершина β, что из всех вершин, которым не соответствует множественный актант и из которых в α ведёт цепь рёбер, β имеет самую короткую цепь рёбер, ведущую в α. Пусть y — сущность, сопоставленная переменной, либо соответствующий текстовый фрагмент, если приёмы 2–4 неприменимы. Тогда, если x — сущность, соответствующая вершине β, то переменной ставится в соответствие сущность $x.y$.

6) Если приёмы 2–5 неприменимы, то переменной сопоставляется сущность x, где x — соответствующий этой переменной фрагмент текста.

7) Если переменная, как следует из формулы, имеет булев тип, то уже соответствующая этой переменной сущность x меняется на x.Is_valid, которая принимает значение 1, если x имеет место, и 0 иначе.

Если применить указанные приемы к упрощенному семантическому графу, изображенному на рисунке 2, и переменным формулы (1), то получим следующие сущности.

A — "Годовая сумма амортизационных отчислений"."Способ уменьшаемого остатка".Is_valid.

B — "Годовая сумма амортизационных отчислений".

C — "Объект основных средств"."Остаточная стоимость на начало отчётного года".

D — "Годовая сумма амортизационных отчислений"."Норма амортизации".

E — "Объект основных средств"."Срок полезного использования".

F — "Годовая сумма амортизационных отчислений"."Норма амортизации"."Коэффициент".

6. Построение модели закона

Когда каждой переменной сопоставлена некоторая сущность, можно перейти непосредственно к построению модели. Как и на всех предыдущих этапах, это можно сделать с помощью применения определённых приёмов к имеющимся формулам. Основные из этих приёмов следующие.

1) Пусть формула имеет вид $A \to B$, где в B имеется выражение вида $C = D$, но нет выражения вида $E = C$. Найдём в уже построенном фрагменте графа модели закона вершину второго вида, в которой происходит запись значения в поле памяти, соответствующее C из рассматриваемой формулы. Возможны 2 варианта.

1.1) Такая вершина есть. Обозначим её α. Рассмотрим вершину, из которой в α ведёт ребро. Обозначим её β. Далее пройдём из β по цепочке рёбер с номером 0 против их направления, пока эта цепочка не закончится. Обозначим вершину, в которую мы попадём, как γ. Далее выполним действия, указанные в пункте 1.3.

1.2) Такой вершины нет. Тогда выберем какую-либо не рассмотренную ранее вершину и будем считать её вершиной второго вида, соответствующей C из рассматриваемой формулы. Обозначим эту вершину α. Выберем ещё одну не рассмотренную ранее вершину, обозначим её γ. Выпустим из неё ребро в вершину α. Далее выполним действия, указанные в пункте 1.3.

1.3) Будем считать γ вершиной четвёртого вида. Рассмотрим какие-либо три новые вершины, выпустим из них в γ по одному ребру, пронумеруем эти рёбра числами 0, 1 и 2. Тогда вершине, из которой в γ ведёт ребро номер 2, будет соответствовать вычисление логического выражения A. К вершине, из которой в γ ведёт ребро номер

1, следует применить приём 3 относительно переменной C. Вершина, из которой в γ ведёт ребро номер 0, будет, возможно, рассмотрена в результате применения 1.1 к одной из оставшихся формул. Если этого не произойдёт, значит, процедура вычисления модели никогда не затронет эту вершину.

2) Пусть в формуле имеется выражение вида $A = B$, нет выражения вида $C = A$, и приём 1 к формуле неприменим. Тогда выберем какую-либо не рассмотренную ранее вершину и будем считать её вершиной второго вида, соответствующей A из рассматриваемой формулы. Обозначим эту вершину α. Выберем ещё одну не рассмотренную ранее вершину, обозначим её γ. Выпустим из неё ребро в вершину α. Вершине γ будет соответствовать вычисление функции B (при помощи приёма 3).

3) По фрагменту формулы, соответствующему вычислению значения переменной B, строится один из следующих фрагментов модели (приём 3.2 используется, если 3.1 неприменим).

3.1) Пусть в формуле есть подформула вида $A\&(B = C) \vee \neg A\&D$, причём в D входит выражение вида $B = E$. Выберем из всех таких подформул данной формулы (для фиксированного B) ту, что не содержится в другой такой подформуле. Пусть α — вершина, соответствующая вычислению значения B. Будем считать, что это — вершина четвёртого вида. Выберем какие-либо три новые вершины, выпустим из них в α по одному ребру, пронумеруем эти рёбра числами 0, 1 и 2. Тогда вершине, из которой в α ведёт ребро номер 2, будет соответствовать вычисление логического выражения A. Вершине, из которой в α ведёт ребро номер 1, будет соответствовать вычисление функции C (смотрите приём 3.2). Если D имеет вид $(B = E)$, вершине, из которой в α ведёт ребро номер 0, будет соответствовать вычисление функции E (также с помощью приёма 3.2); в противном случае следует использовать приём 3.1, рассматривая эту вершину в качестве α, а D в качестве выбранной подформулы.

3.2) Если в формуле есть подформула $B = A$, где A — некоторая переменная, и нет выражений вида $A = C$, где C — некоторая переменная или функция, то будем считать, что рассматриваемая вершина — первого вида, и в ней происходит получение значения A из соответствующего поля памяти. В противном случае речь идёт о вычислении некоторой арифметической функции от нескольких аргументов $A_1, \ldots, A_k$, $k \in \mathbb{N}$. Если эта функция в тексте закона не задана

явно, то можно, например, потребовать у пользователя её задания. Если функция уже задана, то ей можно сопоставить несколько вершин третьего типа, моделирующих вычисление этой функции (так, чтобы последний этап вычисления функции проходил в вершине α). Рёбра, передающие значения аргументов $A_1, \ldots, A_k$, будут входить в некоторые из этих вершин. К каждой вершине, из которой эти рёбра выходят, применяется приём 3 относительно соответствующей переменной.

4) Пусть переменная A — аргумент некоторой функции, и на эту переменную наложено некоторое условие (в виде бинарного или унарного отношения, которому она должна удовлетворять). Тогда выполнение этого условия проверяется в той вершине, в которой вычисляется значение A. В случае невыполнения условия программа требует у пользователя изменить значение переменных, с помощью которых вычисляется A (как правило, это собственно переменная A).

5) Пусть формула имеет вид $A \to B_1 \& \ldots \& B_k, \in \mathbb{N}$, где $B_1, \ldots, B_k$ — переменные. Тогда такая формула преобразуется в конъюнкцию формул $A \to B_1, \ldots, A \to B_k$, к которым можно применять приём 6.

6) Пусть формула имеет вид $A \to B$, где B — переменная, A — подформула, не содержащая никаких переменных, кроме булевых. Пусть B' — переменная B (точнее, соответствующая ей сущность), определённая с помощью всех остальных формул ($B' = 0$, если B не определяется из оставшихся формул). Тогда B определяется как $A \vee B'$.

7) Пусть формула имеет вид $A \to (B \sim C)$ либо $A \to (C \sim B)$, где B — переменная, A, C — подформулы, не содержащие никаких переменных, кроме булевых. Если C — тоже переменная, то для того, чтобы отличить B от C, можно использовать семантический граф: в нём соответствующие этим переменным текстовые фрагменты должны быть связаны каким-либо семантическим отношением. За B принимается переменная, чей текстовый фрагмент является главным в этом семантическом отношении. Пусть B' — переменная B (точнее, соответствующая ей сущность), определённая с помощью всех остальных формул ($B' = 0$, если B не определяется из оставшихся формул). Тогда B определяется как $A \& C \vee B'$.

8) Если функция, определяющая некоторую булеву переменную, не содержит никаких переменных, кроме булевых, то она моделиру-

ется с помощью несколько вершин третьего типа. В некоторые из этих вершин будут входить рёбра, соответствующие переменным.

Алгоритм построения модели закона с помощью этих приёмов следующий. Сначала для каждой формулы, не содержащей никаких переменных, кроме булевых, применяется приём 5; затем ко всем таким формулам применяются приёмы 6 и 7; наконец, к каждой такой формуле применяется восьмой приём. Ко всем остальным формулам применяется приём 1 или 2, а затем — приём 4.

Пример фрагмента модели закона, построенного по формуле (1), изображен на рисунке 3. Здесь предполагается, что $f_\alpha(x, y) = x * y$ и $f_\beta(x, y) = y/x$.

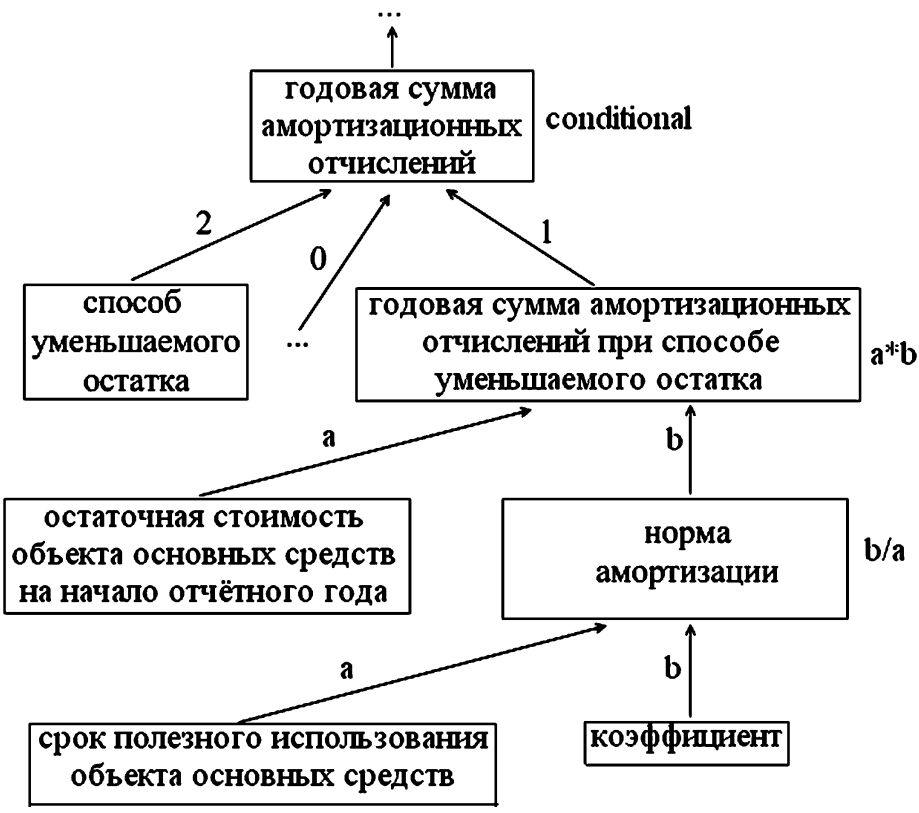

Рис. 3. Фрагмент модели закона. Справа от вершин 3-го вида показаны реализуемые в этих вершинах функции от аргументов a и b. Слово «conditional» сопоставлено вершине 4-го вида.

7. Пример

Рассмотрим ещё один пример текста из [1]: «В течение срока полезного использования объекта основных средств начисление амортизационных отчислений не приостанавливается, кроме случаев перевода его по решению руководителя организации на консервацию на срок более трех месяцев, а также в период восстановления объекта, продолжительность которого превышает 12 месяцев.»

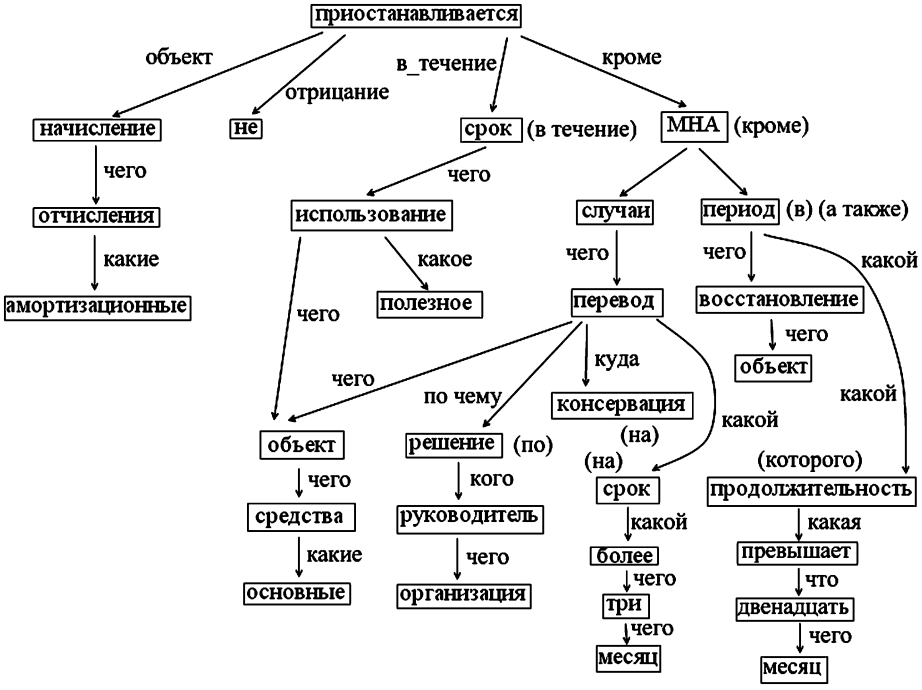

Рис. 4. Семантический граф.

Семантический граф этого предложения приведен на рисунке 4.

Упрощённый семантический граф этого предложения приведен на рисунке 5.

На рисунке 6 показан упрощённый семантический граф вместе с переменными, сопоставленными его вершинам или множествам вершин. Формула, построенная по рассматриваемому предложению, имеет вид

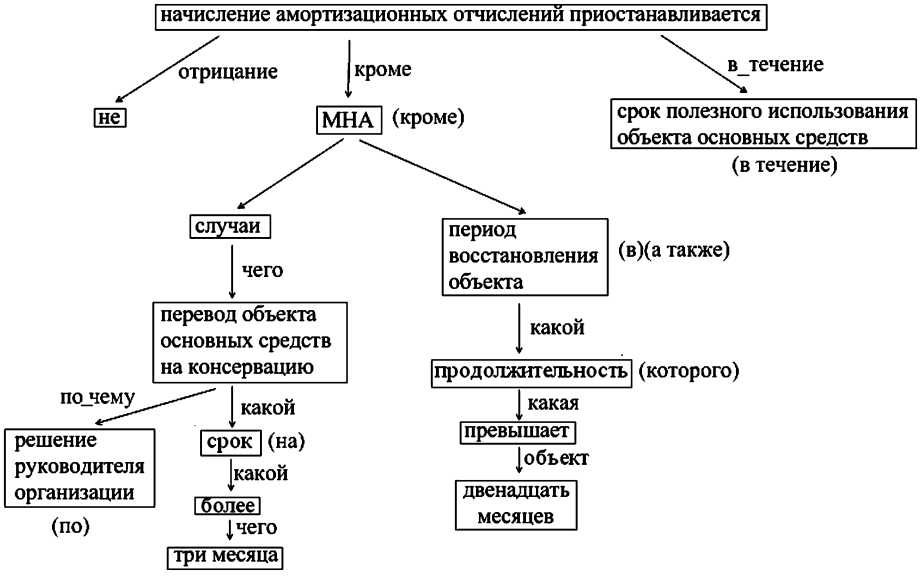

Рис. 5. Упрощённый семантический граф.

$$A \to (C \sim (H\&L\&(P > O) \vee M\&(R > Q))),$$

причём переменным формулы сопоставлены следующие сущности:

A — "Объект основных средств"."Срок полезного использования".Is_valid;

C — "Начисление амортизационных отчислений"."Приостанавливается".Is_valid;

H — "Объект основных средств"."Случаи перевода на консервацию".Is_valid;

L — "Объект основных средств"."Случаи перевода на консервацию"."Решение руководителя организации".Is_valid;

P — "Объект основных средств"."Случаи перевода на консервацию"."Срок";

O — "три месяца";

M — "Объект основных средств" . "Период восстановления".Is_valid;

R — "Объект основных средств"."Период восстановления"."Продолжительность";

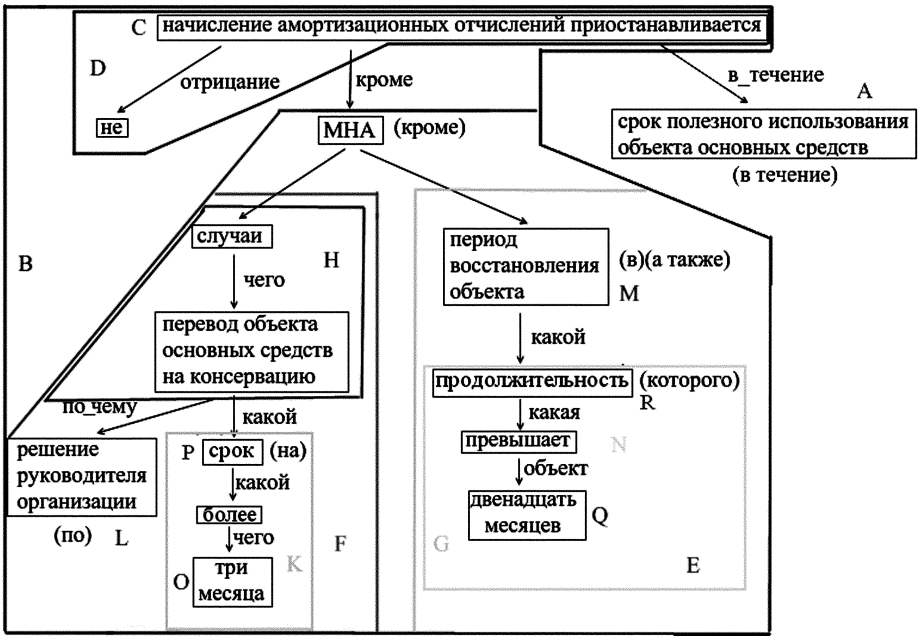

Рис. 6. Упрощённый семантический граф, по которому уже построена формула $A \to (C \sim (H\&L\&(P > O) \vee M\&(R > Q)))$.

Q — "двенадцать месяцев".

И, наконец, на рисунке 7 приведен фрагмент модели закона, соответствующего нашему предложению.

Список литературы

[1] Приказ Министерства финансов Российской Федерации от 30 марта 2001 г. № 26н Об утверждении положения по бухгалтерскому учету «Учет основных средств» ПБУ 6/01. [http://base.consultant.ru/cons/cgi/online.cgi?req = doc; base=LAW; n=111056]

[2] SAP R/3. [http://www.sap.com/cis]

[3] Oracle E-Business Suite. [http://www.oracle.com/ru]

[4] Microsoft Dynamics AX — Microsoft Axapta. [http://www.microsoft.com/en-us/dynamics/erp.aspx]

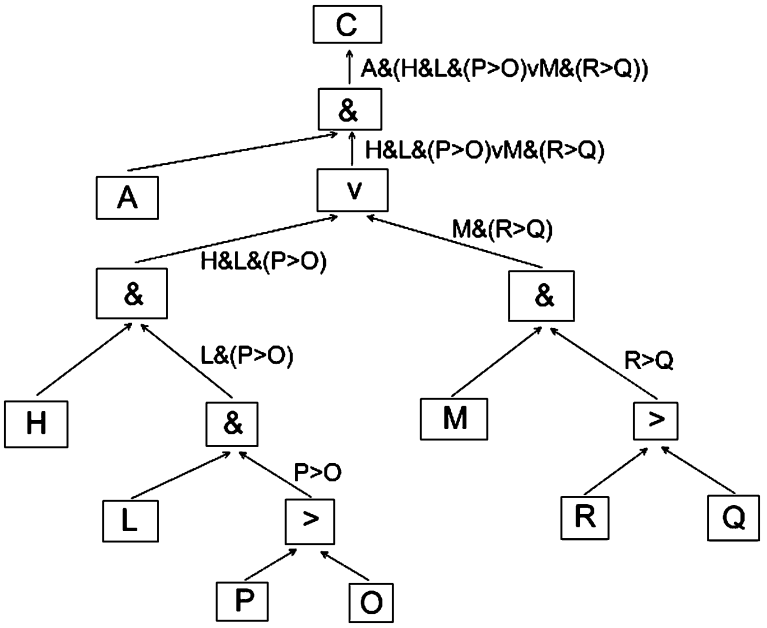

Рис. 7. Фрагмент модели закона.

[5] 1С: Предприятие 8. [http://v8.1c.ru]

[6] Подколзин А.С. Компьютерное моделирование логических процессов. Т. 1. — М.: Физматлит, 2008. [http://intsys.msu.ru/staff/podkolzin/KMLP1.htm]

[7] Сокирко А. Семантические словари в автоматической обработке текста (по материалам системы ДИАЛИНГ). [http://www.aot.ru/docs/sokirko]