



**МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ
ФЕДЕРАЦИИ**
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
**«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ –
МСХА имени К.А. ТИМИРЯЗЕВА»**
(ФГБОУ ВО РГАУ - МСХА имени К.А. Тимирязева)

Красовская Людмила Владимировна
Пчелинцева Светлана Вячеславовна
Кукарцев Владислав Викторович

Алгоритмы машинного обучения для интеллектуальных систем на основе
нечетких семантических сетей в условиях неопределенности

Монография

Под редакцией к.т.н., доцента
Л. В. Красовской

Москва 2024

УДК 004.8
ББК 16.6
К 78

Рецензенты:

ФГБОУ ВО «Саратовский государственный технический университет имени Гагарина Ю.А.», кандидат технических наук, доцент кафедры механики и мехатроники **И.В. Егоров**
ФГБОУ РИАМА, кандидат технических наук, доцент, проректор по учебной и научной работе **Ю.Н. Егоров**

Алгоритмы машинного обучения для интеллектуальных систем на основе нечетких семантических сетей в условиях неопределенности: монография / Л. В. Красовская, С.В. Пчелинцева, В.В. Кукарцев / под ред. Л.В. Красовской / ФГБОУ ВО «Российский государственный аграрный университет – МСХА имени К. А. Тимирязева». – М. : 2024. – 86 с.

ISBN 978-5-9675-2052-5

В монографическом исследовании рассматриваются проблемы создания и моделирования целесообразного поведения интеллектуальных систем, способных целенаправленно функционировать в недоопределенных проблемных средах. Коллективная научная работа объединила результаты исследований, проведенных профессорско-преподавательским составом кафедры прикладной информатики РГАУ-МСХА им. К.А. Тимирязева.

Издание предназначено для преподавателей, научных сотрудников, аспирантов, руководителей предприятий, предпринимателей и специалистов в области машинного обучения и искусственного интеллекта, также может быть полезно студентам вузов технических направлений.

© Красовская Л.В., 2024

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	4
ГЛАВА 1. СОЗДАНИЕ МОДЕЛИ ОПИСАНИЯ ОБРАЗОВ И СИТУАЦИЙ ПРОБЛЕМНОЙ СРЕДЫ В ПРОЦЕССЕ МОДЕЛИРОВАНИЯ ЦЕЛЕСООБРАЗНОГО ПОВЕДЕНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ. ...	6
1.1. ПОСТАНОВКА ЗАДАЧИ	6
1.2. СИНТЕЗ И ФОРМАЛЬНОЕ ОПИСАНИЕ МОДЕЛИ ОБРАЗОВ ПРОБЛЕМНОЙ СРЕДЫ	10
1.3. РАЗРАБОТКА МОДЕЛИ ОПИСАНИЯ СИТУАЦИИ ПРОБЛЕМНОЙ СРЕДЫ	22
ГЛАВА 2. МОДЕЛИРОВАНИЕ ПРОЦЕССА САМООБУЧЕНИЯ ИС С ИСПОЛЬЗОВАНИЕМ АЛГОРИТМОВ САМООБУЧЕНИЯ НА НЕЧЕТКИХ СЕМАНТИЧЕСКИХ СЕТЯХ В СРЕДАХ ПЕРВОГО УРОВНЯ СЛОЖНОСТИ	34
2.1. ФОРМИРОВАНИЕ ПРОЦЕССА САМООБУЧЕНИЯ НА НЕЧЕТКИХ СЕМАНТИЧЕСКИХ СЕТЯХ В СРЕДАХ ПЕРВОГО УРОВНЯ СЛОЖНОСТИ	34
2.2. ФОРМИРОВАНИЕ ПРОЦЕССА САМООБУЧЕНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ С АКТИВНОЙ ЛОГИКОЙ ПОВЕДЕНИЯ НА НЕЧЕТКИХ СЕМАНТИЧЕСКИХ СЕТЯХ В СРЕДАХ ПЕРВОГО УРОВНЯ СЛОЖНОСТИ	42
2.3. ФОРМИРОВАНИЕ ПРОЦЕССА САМООБУЧЕНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ С АКТИВНО-ПАССИВНОЙ ЛОГИКОЙ ПОВЕДЕНИЯ НА НЕЧЕТКИХ СЕМАНТИЧЕСКИХ СЕТЯХ В СРЕДАХ ПЕРВОГО УРОВНЯ СЛОЖНОСТИ	45
ГЛАВА 3. МОДЕЛИРОВАНИЕ ПОВЕДЕНИЯ ИС С ИСПОЛЬЗОВАНИЕМ АЛГОРИТМОВ САМООБУЧЕНИЯ НА НЕЧЕТКИХ СЕМАНТИЧЕСКИХ СЕТЯХ В СРЕДАХ ВТОРОГО УРОВНЯ СЛОЖНОСТИ	50
3.1. АЛГОРИТМЫ САМООБУЧЕНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ ПРИ НАЛИЧИИ В СРЕДЕ ПРИЧИННО-СЛЕДСТВЕННЫХ СВЯЗЕЙ: ОПИСАНИЕ И АНАЛИЗ	50
3.2. АЛГОРИТМЫ САМООБУЧЕНИЯ ПРИ ПОЯВЛЕНИИ В СРЕДЕ ТОРМОЗНЫХ СИГНАЛОВ: ОПИСАНИЕ И АНАЛИЗ	57
3.3. АЛГОРИТМЫ САМООБУЧЕНИЯ ПРИ НАЛИЧИИ В СРЕДЕ СУММАРНЫХ СИГНАЛОВ: ОПИСАНИЕ И АНАЛИЗ	64
3.4. РАЗРАБОТКА ПРОЦЕДУР ВЫБОРА АЛГОРИТМОВ ЭФФЕКТИВНОГО ПОВЕДЕНИЯ ИНТЕЛЛЕКТУАЛЬНОЙ СИСТЕМЫ	72
СПИСОК ПРИНЯТЫХ СОКРАЩЕНИЙ	77
СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ	78

ВВЕДЕНИЕ

Современные тенденции развития науки и техники диктуют необходимость поиска новых подходов к управлению сложными системами и созданию высокоэффективных систем поддержки принятия решений. Одной из ключевых областей, привлекающих внимание исследователей, становится искусственный интеллект (ИИ), который предоставляет уникальные возможности для автоматизации процессов управления и оптимизации производственных операций. Разработка и внедрение интеллектуальных систем (ИС) на базе новейших достижений ИИ позволяют существенно ускорить научно-технический прогресс и обеспечить качественно новый уровень эффективности производства.

Актуальность данной проблематики обусловлена стремительным ростом числа исследований в сфере ИИ, а также активным внедрением результатов этих исследований в разнообразные сферы человеческой деятельности. Значительный вклад в развитие ИС внесли такие ученые, как Аверкин А.Н., Берштейн Л.С., Вагин В.Н., Ефимов Е.И., Журавлев Ю.Н., Зарипов А.Х., Клыков Ю.И., Литвицева Л.В., Левин Д.Д., Любарский Ю.А., Мелихов А.Н., Нариньяни А.С., Поспелов Д.А., Попов Э.В., Тимофеев А.В., Фин В.К., Цаленко М.Ш., Чернухин Ю.В., Яхно Т.М. и многие другие.

Несмотря на значительные достижения в области разработки ИС, проблема создания эффективных интеллектуальных систем, способных адаптироваться к изменяющимся условиям и принимать обоснованные решения в неопределённых проблемных средах (ПС), остаётся актуальной. Особенно остро стоит вопрос о создании систем, которые могут самостоятельно функционировать в априори неописанных ПС, где полное и точное описание всех возможных ситуаций невозможно. Это связано с тем, что реальные условия функционирования таких систем зачастую настолько сложны и многообразны, что описать все возможные сценарии взаимодействия с окружающей средой заранее крайне затруднительно.

Эта работа направлена на разработку теоретических основ и практических методов построения интеллектуальных систем, способных эффективно функционировать в условиях высокой неопределённости и сложности проблемных сред, что имеет важное значение для дальнейшего развития автоматизированных систем управления и принятия решений.

ГЛАВА 1. СОЗДАНИЕ МОДЕЛИ ОПИСАНИЯ ОБРАЗОВ И СИТУАЦИЙ ПРОБЛЕМНОЙ СРЕДЫ В ПРОЦЕССЕ МОДЕЛИРОВАНИЯ ЦЕЛЕСООБРАЗНОГО ПОВЕДЕНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ.

1.1. ПОСТАНОВКА ЗАДАЧИ

В самом общем виде задачу моделирования целенаправленного поведения интеллектуальной системы (ИС) в различных условиях проблемной среды (ПС) можно сформулировать следующим образом. Допустим, интеллектуальная система, ориентированная на решение определенного класса задач, имеет развитую информационно-измерительную систему и способна обрабатывать множество действий $B = B_i$, $i = \overline{1, m}$ (например, взять заготовку, повернуть влево, перейти в заданную точку и т.д.).

Проблемную среду можно охарактеризовать множеством $Q = \{Q_{i1}\}, i1 = \overline{1, n1}$ находящихся в ней объектов, взаимодействие которых между собой и ИС описывается при помощи бинарных отношений $R = \{r_{i2}\}, i2 = \overline{1, n2}$. Стоящую перед ИС цель функционирования можно определить ситуацией $S_{\text{итек}} \in S$, $S = \{S_{i3}\}, i3 = \overline{1, n3}, S_{\text{рез}} \in S$, где S - множество ситуаций, допустимых целей и подцелей поведения системы.

В случае, когда ситуации $S_{\text{итек}}$ и $S_{\text{рез}}$ неравны между собой, ИС для достижения цели $S_{\text{рез}}$ необходимо устранить все имеющиеся между ними различия, выполнив преобразование первой из них $S_{\text{итек}}$ во вторую $S_{\text{рез}}$ путем имитации отработки спланированной последовательности действий $b_i \in B$.

При этом между ситуациями $S_{\text{итек}}$ и $S_{\text{рез}}$ могут наблюдаться различия следующих видов:

- по значениям одноименных отношений, описывающих взаимодействия одинаковых объектов;
- по характеру отношений между одноименными объектами;

- по состояниям объектов, образующих эти ситуации.

В соответствии с уровнем осведомленности ИС о закономерностях преобразования ситуаций ПС, происходящих в результате обрабатываемых ею действий, можно выделить три уровня определенности условий функционирования.

1. Условия с полностью известными закономерностями преобразования ситуаций среды из одной в другую, когда ИС задана подробная информационная модель конкретной предметной области или в форме пространства состояний, или в форме пространства подзадач, и известен порядок устранения различий между ситуациями $S_{\text{тек}}$ и $S_{\text{рез}}$. К одной из первых планирующих систем, позволяющих реализовать отмеченную задачу, относится решатель GPS[38]. Однако, следует отметить, что система GPS и аналогичные ей планировщики, например, STRIPS [6], осуществляют поиск решения только в пространстве состояний и для них характерна громоздкость конструкций описания формальной модели закономерностей ПС. Дальнейшим перспективным направлением развития подобных интеллектуальных планирующих систем является использование обобщенной стратегии поиска решения, сущность которой заключается в комплексной стратегии поиска, как в пространстве состояний, так и пространстве подзадач[40]. К эффективным разработкам, реализующим обобщенную стратегию поиска решений, можно отнести, например, системы СПРИНТ[41] и СФИНКС[7].

2. Условия с лингвистической недоопределенностью закономерностей преобразования ситуаций среды. В этом случае ИС может быть задана обобщенная модель различных ситуаций ПС безотносительно к конкретной предметной области, и она планирует поведение на основе поступающей из среды информации и ранее накопленного опыта функционирования. Сюда можно отнести, например, случай когда текущая ситуация среды $S_{\text{тек}}$ с определенной степенью близости идентична одной из типовых, часто встречающихся ситуаций, заданных ИС при проектировании или

сформированных ею в процессе функционирования. К одной из эффективных планирующих систем, позволяющих осуществлять планирование поведения в недоопределенных условиях ПС, следует отнести систему адаптивного логического вывода. Однако, узким местом отмеченной системы является ее привязка к конкретной предметной области [94].

3. Условия с априорной неопределенностью преобразования ситуаций среды, когда ИС не может использовать ранее накопленный опыт функционирования, ввиду его отсутствия, и ей заранее не задана информационная модель закономерностей среды, например, если ситуация $S_{\text{итек}}$ отличается от ситуации $S_{\text{рез}}$ по состояниям одноименных объектов и ИС априори неизвестен порядок преобразований этих объектов от одного состояния к другому, происходящих в результате имитации отрабатываемых системой действий.

В первом рассмотренном случае организация планирования поведения сводится к разработке процедур принятия решений и структуры представления знаний, на основе которых осуществляется вывод решения задач.

Во втором случае для решения различных задач ИС должна обладать следующими умениями: автоматически выявлять имеющиеся между ситуациями $S_{\text{итек}}$ и $S_{\text{рез}}$ различия, осуществлять выбор средств и действий позволяющих устранить эти различия; ранжировать порядок отработки выбранных действий в целях преобразования ситуации $S_{\text{итек}}$ в ситуацию $S_{\text{рез}}$. Следовательно, здесь организация планирующей системы ИС сводится к разработке структуры представления знаний в общем виде, безотносительно к конкретной предметной области и процедур планирования, позволяющих реализовать отмеченные выше возможности и задачи.

В третьем случае ИС должна автоматически изучить закономерности преобразования ситуаций среды, на этой основе построить ее

информационную модель и использовать эту модель для дальнейшего планирования поведения. Иными словами, интеллектуальная система должна обладать способностью к самообучению, и задача организации ее планирующей системы сводится к разработке эффективных алгоритмов самообучения, позволяющих адаптироваться к изменяющимся и заранее неопределенным условиям ПС.

Важную и основополагающую роль в организации ИС играет структура представления знаний о закономерностях внешнего мира. Анализ известных принципов представления знаний в интеллектуальных системах принятия решений показывает, что в основном они базируются на подробном описании закономерностей конкретной предметной области и процедурах вывода решений на основе таких знаний [21]. Иногда эти знания хотя и носят достаточно обобщенный характер, но являются, "пассивными", например, в диалоговых ИС, т.е. определяют отношения, возникающие между различными объектами без участия активного "лица" каким является ИС в процессе целесообразного функционирования [41]. Отмеченные особенности приводят к ограничению функциональных возможностей интеллектуальных систем, организованных на основе известных методов представления различного рода знаний.

Одним из возможных путей преодоления вышеуказанных трудностей является представление знаний ИС в виде двух дополняющих друг друга систем отношений, возникающих между объектами ПС. В первой системе отношений (образные или объектные знания) объекты выступают как известные качественные определенности, характеризующиеся признаками, выражающими их собственную природу. Во второй системе (процедурные знания) даются отношения, определяющие характер взаимодействия объектов среды, а также описываются особые отношения между объектами, возникающие в сфере инструментальной деятельности ИС и без нее не существующие. Таким образом, при организации планирующей системы ИС знания, описывающие объекты и закономерности среды, следует обобщить

и систематизировать, представив в виде двух отмеченных систем отношений, возникающих между объектами среды и ИС в процессе ее целенаправленного функционирования.

1.2. СИНТЕЗ И ФОРМАЛЬНОЕ ОПИСАНИЕ МОДЕЛИ ОБРАЗОВ ПРОБЛЕМНОЙ СРЕДЫ

Для систематизированного представления предметных знаний все образы ПС по семантическому признаку можно разделить на пять основных списков [50]:

$Q = \{Q_{i1}\}, i_1 = \overline{1, n_1}$ - множество объектов;

$P = \{P_{i4}\}, i_4 = \overline{1, n_4}$ - множество составных частей объектов;

$J = \{J_{i5}\}, i_5 = \overline{1, n_5}$ - множество свойств или возможностей объектов;

$L = \{L_{i6}\}, i_6 = \overline{1, n_6}$ - множество признаков объектов и их составных частей;

$H = \{H_{i7}\}, i_7 = \overline{1, n_7}$ - множество имен объектов и их составных частей.

Определение 1.1. Объектами называются образы и понятия ПС, имеющие законченный, самостоятельный семантический смысл и содержание (например, автомобиль, станок, дерево решений).

Определение 1.2. Частями объектов называются образы ПС, которые являются составными элементами, определяющими структуру объекта (например, двигатель автомобиля).

В самом общем случае одно и то же понятие может определять как категорию объекта, так и категорию части объекта, например, понятие "ручка" определяет категорию объекта, если речь идет о пишущей ручке, а не о ручке чемодана и т.д. Иными словами, просматривается многозначность в толковании понятий. Обойти указанную трудность можно, если при представлении объектных знаний ИС многозначные понятия, фактически определяющие составные части сложных объектов, согласно их смысловому содержанию представлять в виде составных понятий (например, дверная ручка).

Определение 1.3. Под свойствами и умениями объектов и их составных частей следует понимать соответственно их функциональное назначение и действия, которые они оказывают друг на друга.

Функциональное назначение объектов определяется типом инструмента, в качестве которого они могут быть использованы в процессе поведения и каким образом осуществляется применение этого инструмента.

Определение 1.4. Признаками частей объектов, а, следовательно, и признаками самих объектов называются образы C (характеристики), которые описывают их форму, химический состав, цвет, геометрические размеры и т.п.

Определение 1.5. Именами и названиями объектов и их составных частей называются абстрактные биективно соответствующие им понятия, выраженные на формальном языке ИС (возможно естественном).

Задаваясь на элементах множества объектов Q родовидовыми отношениями, его можно представить как объединение трех подмножеств $Q = Q_1 \cup Q_2 \cup Q_3$, где $Q_1 = \{q_{i8}\}, i_8 = \overline{1, n_8}$ - множество объектов, определяемых понятием "вид" (например, токарный станок № 3);

$Q_2 = \{q_{i9}\}, i_9 = \overline{1, n_9}$ - множество объектов, определяемых понятием "род" (например, птицы);

$Q_3 = \{q_{i10}\}, i_{10} = \overline{1, n_{10}}$ - множество объектов, определяемых понятием "класс" (например, животные).

При этом понятие "вид" определяется рядом конкретных индивидуумов, рассматриваемых как идентичные объекты или объекты, различающиеся между собой по несущественным характеристикам.

Согласно определениям 1.1-1.5 каждый объект $q_{i1} \in Q$ описывается при помощи соответствующего множества характеристик X_{i1} , состоящего из трех подмножеств $P_{i1}, J_{i1} \cup L_{i1}$, т.е. $X_{i1} = P_{i1} \cup J_{i1} \cup L_{i1}$.

Следовательно, объект q_{i1} по характерным только для него элементам этих подмножеств определяет понятие одного вида.

Некоторое множество видов по характерной только для него совокупности описывающих его элементы подмножеств P_{i1}, J_{i1}, L_{i1} объединяется в список и определяет понятие одного рода. Некоторое множество родов по характерному только для него пересечению описывающих его элементы подмножеств P_{i1}, J_{i1}, L_{i1} объединяется в список и определяет понятие одного класса.

Множество составных частей объектов представляет собой объединение следующих трех подмножеств:

$P_1 = \{P_{i11}\}, i_{11} = \overline{1, n_{11}}, P_1$ - множество обобщенных частей объектов или частей объектов, состоящих в свою очередь из нескольких сложных элементов;

$P_2 = \{P_{i12}\}, i_{12} = \overline{1, n_{12}}, P_2$ - множество частей обобщенных частей объектов;

$P_3 = \{P_{i13}\}, i_{13} = \overline{1, n_{13}}, P_3$ - множество неделимых частей объектов.

Свойства объектов можно разделить также на три подмножества:

$J_1 = \{j_{i14}\}, i_{14} = \overline{1, n_{14}}, J_1$ - прямые свойства, определяющие умение объектов и непосредственные формы их взаимодействия без активного участия ИС;

$J_2 = \{j_2\}, i_2 = \overline{1, m_2}, J_2$ - вспомогательные свойства, характеризующие предварительные формы взаимодействия объектов (например, сближение одного объекта с другим и т.п.).

$J_3 = \{j_3\}, i_3 = \overline{1, m_3}, J_3$ - инструментальные свойства, определяющие в качестве какого инструмента, могут быть использованы объекты, и каким образом их следует использовать.

Множество признаков объекта L разбивается на следующие непересекающиеся подмножества:

$L_1 = \{l_{i15}\}, i_{15} = \overline{1, n_{15}}, L_1$ - множество признаков, определяющих различные цвета объектов и их частей;

$L_2 = \{l_{i16}\}, i_{16} = \overline{1, n_{16}}, L_2$ L - различные геометрические формы объектов и их частей;

$L_3 = \{l_{i17}\}, i_{17} = \overline{1, n_{17}}, L_3$ L - признаки, задающие размеры различных объектов;

$L_4 = \{l_{i18}\}, i_{18} = \overline{1, n_{18}}, L_4$ L - элементы химического состава объектов;

$L_5 = \{l_{i19}\}, i_{19} = \overline{1, n_{19}}, L_5$ L - прочие признаки объектов.

Каждому объекту q_{i1} Q можно присвоить абстрактное имя или понятие на языке ИС. Следует различать обобщенные и собственные имена. Обобщенные имена присваиваются некоторому подмножеству объектов (объектов, относящихся к одному виду), а собственные к конкретному индивидууму.

Помимо сказанного, каждый объект проблемной среды q_{i1} Q можно охарактеризовать множеством допустимых состояний $F = f_{i20}, i_{20} = \overline{1, n_{20}}$, которые они могут принимать и изменять в результате оказываемых на него внешних и внутренних воздействий.

К одному из эффективных способов структурированного представления объектных знаний можно отнести сетевые модели, которые позволяют достаточно полно в общем, виде описывать объекты ПС и осуществлять ассоциативные принципы поиска и выборки необходимых данных в процессе принятия решений. В этом случае каждый объект q_{i1} Q описывается соответствующей ему семантической макросетью $M_{i1} = \{m_{i21}^{i1}\}, i_{21} = \overline{1, n_{21}^{i1}}$, где каждая микросеть m_{i21}^{i1} M_{i1} представляет собой взаимосвязанные между собой элементы множеств О,Р,Ж,Л,Ф и Н. При этом каждая макросеть M_{i1} может состоять из следующих пяти типов автономных сетей:

1. $E_1^* = \{e_{i22}\}, i_{22} = \overline{1, n_{22}}$ -сетью возможных состояний объектов, для которых микросеть e_{i22} E_1^* определяется высказыванием вида $e_{i22} = f_{i20}r_{i2}f_{i20}$ или высказыванием $e_{i22} = q_{i1}r_{i2}f_{i20}$, где роль отношения r_{i2} выполняет элемент из

множества свойств J , а отношение r_{i2} задается высказыванием "быть состоянием объекта".

Рассмотрим случай, когда в роли отношения r_{i2} R выступает действие ИС $b_i \in B$, преобразующее состояние $f_{i20} \in F$ в состояние $f_{i20} \in F$. При этом высказывание $e_{i22} = f_{i20} r_{i2} f_{i20}$ заменяется элементарным звеном поведения $f_{i20} \& b_i^1, b_1^2, \dots, b_1^k \rightarrow f_{i20}$. Приведенная запись означает, что если объект находится в состоянии f_{i20} , то действия $b_i^1, b_1^2, \dots, b_1^k$ при их выполнении в заданной последовательности переводят объект в состояние f_{i20} . Выполняя операцию композиции над элементами множества $E_1^* \in M_{i1}$ можно сформировать цепочку действий следующего вида $f_{i20}^0 \& b_i^1 \rightarrow f_{i20}^1 \& b_i^2 \rightarrow \dots \rightarrow f_{i20}^*$, позволяющую перевести объект из состояния f_{i20}^0 , в требуемое состояние f_{i20}^* .

2. $E_2^* = \{e_{i23}\}, i_{23} = \overline{1, n_{23}}$ - сеть составных частей объекта. В ней каждая микросеть $e_{i23} \in E_2^*$ определяется высказыванием следующего содержания: $p_{i12} r_{i2} p_{i13}, q_{i1} r_{i2} p_{i11}, p_{i11} r_{i2} p_{i12}$, где r_{i2} - отношение "быть составной частью".

3. $E_3^* = \{e_{i24}\}, i_{24} = \overline{1, n_{24}}$ - сеть свойств объекта. В этой сети каждая микросеть $e_{i24} \in E_3^*$ есть тройка вида $q_{i1} r_{i2} j_{i6}$, а роль отношения r_{i2} могут выполнять высказывания следующего содержания: "обладать способностью", "обладать умением" и "выступать в качестве инструмента для".

4. $E_4^* = \{e_{i25}\}, i_{25} = \overline{1, n_{25}}$ - сеть описания признаков объекта, где $e_{i25} \in E_4^*$ - высказывание "иметь признак l_{i6} ", например, "иметь синий цвет" и т.д.

В общем случае структуру каждой макросети M_{i1} описания объекта $q_{i1} \in Q$ при помощи нотаций Бекуса можно представить следующим образом:

$\langle \text{объект } q_{i1} \rangle ::= \langle \text{макросеть } M_{i1} \rangle;$

$\langle \text{макросеть } M_{i1} \rangle ::= \langle \text{вход} \rangle \langle \text{тело} \rangle \langle \text{выход} \rangle;$

$\langle \text{вход} \rangle ::= \langle \text{описание внешнего вида объекта} \rangle / \langle \text{имя объекта} \rangle;$

<описание внешнего вида> ::= <контурное изображение
объекта>/<внешние характеристики объекта >;

<имя объекта> ::= <собственное имя>/<обобщенное
имя>/<собственное имя><обобщенное имя>;

<собственное имя> ::= <идентификатор>;

<обобщенное имя> ::= <идентификатор>;

<выход> ::= <вид>/<род>/<класс>/<род>/<класс><вид><род>;

<вид> ::= <вход>/<входы><вход>;

<род> ::= <вход>/<входы><вход>;

<класс> ::= <вход>/<входы><вход>;

<тело> ::= <сеть E_1^* ><сеть E_2^* ><сеть E_3^* ><сеть E_4^* >.

Приведем структуру сетей $E_1^* - E_4^*$, образующих <тело> M_{i1} макросети,
описывающей объект $q_{i1}^* \in Q$.

Описание сети состояний объекта E_1^* :

<сеть E_1^* > ::= <список>/<граф>;

<список> ::= <структуры>;

<структуры> ::= <структура>/<структуры>;

<структура> ::= <микросеть $m_{i21}^{i1} \in M_{i1}$ >;

<микросеть> ::= <<состояние f_{i20} >< отношение r_{i2} > <состояние
 f_{i20} >>>/<<объект q_{i1} ><отношение r_{i2} ><состояние f_{i20} >/ <элементарное звено

поведения $f_{i20} \& b_i^1, b_1^2, \dots, b_1^k \rightarrow f_{i20}$ >;

<состояние f_{i20} > ::= <воспринимаемая фотография >/<описание
внешнего вида>;

<отношение> ::= <иметь состояние>/<перейти из одного состояния в
другое>;

<объект q_{i1} > ::= <вход><тело><выход>;

<граф> ::= <вершина>/<вершины><ветви>;

<вершина> ::= <описание состояния>/<имя объекта>;

$\langle \text{ветви} \rangle ::= \langle \text{ветвь} \rangle / \langle \text{ветви} \rangle$; $\langle \text{ветвь} \rangle ::= \langle \text{отношение} \rangle / \langle \text{действие } b_{i1} \rangle$.

Описание сети E_2^* составных частей объекта:

$\langle \text{сеть } E_2^* \rangle ::= \langle \text{список} \rangle / \langle \text{граф} \rangle$;

$\langle \text{список} \rangle ::= \langle \text{структуры} \rangle$;

$\langle \text{структуры} \rangle ::= \langle \text{структура} \rangle / \langle \text{структуры} \rangle$;

$\langle \text{структура} \rangle ::= \langle \text{микросеть } m_{i21}^{i1} \ M_{i1} \rangle$;

$\langle \text{микросеть} \rangle ::= \langle \text{объект } q_{i1} \rangle \langle \text{отношение } r_{i2} \rangle \langle \text{часть } p_{i4} \rangle$;

$\langle \text{часть } p_{i4} \rangle ::= \langle \text{макросеть } M_{i1} \ (\text{часть } p_{i4}) \rangle$;

$\langle \text{макросеть } M_{i1} \ (\text{часть } p_{i4}) \rangle ::= \langle \text{обобщенная часть } p_{i11} \rangle / \langle \text{часть обобщенной части } p_{i12} \rangle / \langle \text{неделимая часть } p_{i13} \rangle$;

$\langle \text{обобщенная часть} \rangle ::= \langle \text{вход} \rangle$;

$\langle \text{часть обобщенной части} \rangle ::= \langle \text{вход} \rangle$;

$\langle \text{неделимая часть} \rangle ::= \langle \text{вход} \rangle$;

$\langle \text{граф} \rangle ::= \langle \text{вершина} \rangle / \langle \text{вершины} \rangle \langle \text{ветви} \rangle$;

$\langle \text{вершина} \rangle ::= \langle \text{имя части} \rangle / \langle \text{имя объекта} \rangle$;

$\langle \text{ветви} \rangle ::= \langle \text{ветвь} \rangle / \langle \text{ветви} \rangle$;

$\langle \text{ветвь} \rangle ::= \langle \text{отношение} \rangle$;

$\langle \text{отношение} \rangle ::= \langle \text{отношение быть частью объекта } q_{i1} \rangle / \langle \text{быть частью}$

одной из частей объекта q_{i1} \rangle ;

Описание сети E_3^* свойств объекта:

$\langle \text{сеть } E_3^* \rangle ::= \langle \text{список} \rangle / \langle \text{граф} \rangle$;

$\langle \text{список} \rangle ::= \langle \text{структуры} \rangle$;

$\langle \text{структуры} \rangle ::= \langle \text{структура} \rangle / \langle \text{структуры} \rangle$;

$\langle \text{структура} \rangle ::= \langle \text{микросеть } m_{i21}^{i1} \ M_{i1} \rangle$;

$\langle \text{микросеть} \rangle ::= \langle \text{объект } q_{i1} \rangle \langle \text{отношение } r_{i2} \rangle \langle \text{свойство } J_{i5} \rangle$;

$\langle \text{свойство} \rangle ::= \langle \text{действие} \rangle / \langle \text{результат действия} \rangle / \langle \text{применение в качестве инструмента} \rangle$;

$\langle \text{действие} \rangle ::= \langle \text{умение} \rangle / \langle \text{имя действия} \rangle$;

<результат действия>::=<изменение состояния объекта
 q_{i1} >/<изменение ситуации ПС $S_{iтек}$ >;
 <применение в качестве инструмента>::=<где применять><как
 применять>;
 <где применять>::=<ситуация ПС $S_{iтек}$, описывающая условия,
 позволяющие использовать объект в качестве определенного инструмента>;
 <как применять>::=<программа поведения ИС, связанная с
 применением объекта в качестве инструмента>;
 <граф>::=<вершина>/<вершины><ветви>;
 <вершина>::=<имя объекта>/<имя действия>/<имя умения>/<имя и
 назначение инструмента>;
 <ветви>::=<ветвь>/<ветви>;
 <ветвь>::=<отношение>; <отношение>::=<выполнить действие
 b_{i1} >/<обладать умением>/<использовать в качестве инструмента>;
 <имя объекта>::=<вход M_{i1} макросети>/<идентификатор>;
 <имя действия>::=<идентификатор>; <имя умения>::
 =<идентификатор>;

Описание сети E_4^* признаков объекта:

<сеть E_4^* >::=<список>/<граф>;
 <список>::=<структуры>;
 <структуры>::=<структура>/<структуры>;
 <структура>::=<микросеть m_{i21}^{i1} M_{i1} >;
 <микросеть>::=<объект q_{i1} ><отношение r_{i2} ><признак l_{i6} >;
 <объект q_{i1} >::=<имя объекта>/<вход M_{i1} макросети объекта>;
 <имя объекта>::=<идентификатор>;
 <отношение>::=<обладать признаком l_{i6} >;
 <признак>::=<цвет>/<геометрическая форма>/<размеры>/<химический
 состав>/<прочие признаки>;
 <граф>::=<вершина>/<вершины><ветви>;

$\langle \text{вершина} \rangle ::= \langle \text{имя объекта} \rangle / \langle \text{имя признака} \rangle;$
 $\langle \text{имя объекта} \rangle ::= \langle \text{идентификатор} \rangle;$
 $\langle \text{имя признака} \rangle ::= \langle \text{качественное описание признака} \rangle / \langle \text{количественное описание признака} \rangle;$
 $\langle \text{ветви} \rangle ::= \langle \text{ветвь} \rangle / \langle \text{ветви} \rangle;$
 $\langle \text{ветвь} \rangle ::= \langle \text{отношение} \rangle;$
 $\langle \text{отношение} \rangle ::= \langle \text{обладать признаком } l_{i7} \rangle.$

Теоретико-множественные операции над макросетями описания объектов проблемной среды. Рассмотрим теоретико-множественные операции, выполняемые над макросетями M_{i1} на уровне микросетей m_{i21}^{i1} M_{i1} , как неделимых атомов. К ним относятся операции объединения, пересечения, дополнения, а также операции полной и частичной идентификации макросетей. Первые три операции над макросетями выполняются для упаковки информации, хранимой в базе знаний, по мере ее пополнения и приобретения ИС. Две другие операции применяются для идентификации объектов при предварительном их выборе в процессе инструментальной деятельности [45].

Операция объединения выполняется над макросетями с одинаковыми входами и служит для расширения макросетей по мере накопления знаний. Пусть заданы две макросети $M_{i1}^1 = \langle \text{вход}_1 \rangle \langle E_{11}^1, E_{21}^1, E_{31}^1, E_{41}^1 \rangle \langle \text{выход}_1 \rangle$ и $M_{i1}^2 = \langle \text{вход}_2 \rangle \langle E_{12}^2, E_{22}^2, E_{32}^2, E_{42}^2 \rangle \langle \text{выход}_2 \rangle$, для которых $\langle \text{вход}_1 \rangle = \langle \text{вход}_2 \rangle$, а следовательно $\langle \text{выход}_1 \rangle = \langle \text{выход}_2 \rangle$. Тогда, в результате объединения макросетей M_{i1}^1 и M_{i1}^2 получаем макросеть $M_{i1}^3 = M_{i1}^1 \cup M_{i1}^2 = \langle \text{вход}_3 \rangle \langle E_{13}^3, E_{23}^3, E_{33}^3, E_{43}^3 \rangle \langle \text{выход}_3 \rangle$, где $\langle \text{вход}_3 \rangle = \langle \text{вход}_1 \rangle = \langle \text{вход}_2 \rangle$; $\langle \text{выход}_3 \rangle = \langle \text{выход}_1 \rangle = \langle \text{выход}_2 \rangle$;

$$E_{13}^3 = E_{11}^1 \cup E_{12}^2; E_{23}^3 = E_{21}^1 \cup E_{22}^2; E_{33}^3 = E_{31}^1 \cup E_{32}^2; E_{43}^3 = E_{41}^1 \cup E_{42}^2.$$

Операция пересечения выполняется над макросетями, описывающими объекты из одного рода или класса. Эта операция позволяет исключить дублирование хранимой в базе знаний информации, характерной для многих объектов одного рода или класса. Пусть заданы макросети

$M^1_{i1} = \langle \text{вход}_1 \rangle \langle E^1_{1*}, E^1_{2*}, E^1_{3*}, E^1_{4*} \rangle \langle \text{выход}_1 \rangle$ и $M^2_{i1} = \langle \text{вход}_2 \rangle \langle E^2_{1*}, E^2_{2*}, E^2_{3*}, E^2_{4*} \rangle \langle \text{выход}_2 \rangle$, для которых $\langle \text{вход}_1 \rangle = \langle \text{вход}_2 \rangle$, а $\langle \text{выход}_1 \rangle = \langle \text{выход}_2 \rangle$. Тогда, в результате пересечения макросетей M^1_{i1} и M^2_{i1} получаем макросеть $M^3_{i1} = M^1_{i1} \cap M^2_{i1}$; $M^3_{i1} = \langle \text{вход}_3 \rangle \langle E^3_{1*}, E^3_{2*}, E^3_{3*}, E^3_{4*} \rangle \langle \text{выход}_3 \rangle$, где $\langle \text{вход}_3 \rangle = \langle \text{выход}_1 \rangle = \langle \text{выход}_2 \rangle$; $\langle \text{выход}_3 \rangle$ - задается понятием "род объектов" или "класс объектов", если $\langle \text{выход}_1 \rangle = \langle \text{выход}_2 \rangle$ определяется понятием "вид объектов" или "род объектов" соответственно.

Операция дополнения выполняется над произвольными макросетями и позволяет выявлять микросети m^i_{i1} M_{i1} , характерные только для конкретного объекта $q_{i1} \in Q$. Пусть заданы две произвольные макросети $M^1_{i1} = \langle \text{вход}_1 \rangle \langle E^1_{1*}, E^1_{2*}, E^1_{3*}, E^1_{4*} \rangle \langle \text{выход}_1 \rangle$ и $M^2_{i1} = \langle \text{вход}_2 \rangle \langle E^2_{1*}, E^2_{2*}, E^2_{3*}, E^2_{4*} \rangle \langle \text{выход}_2 \rangle$. Тогда, в результате выполнения операции получаем макросеть $M^3_{i1} = M^1_{i1} \setminus M^2_{i1}$ или макросеть $M^3_{i1} = M^2_{i1} \setminus M^1_{i1}$, $M^3_{i1} - M^3_{i1}$, где, например, макросеть $M^3_{i1} = \langle k^* \rangle \langle \text{вход}_3 \rangle \langle E^3_{1*}, E^3_{2*}, E^3_{3*}, E^3_{4*} \rangle \langle \text{выход}_3 \rangle$; $\langle \text{вход}_3 \rangle = \langle \text{вход}_1 \rangle$ и $\langle \text{выход}_3 \rangle = \langle \text{выход}_1 \rangle$; $E^3_{1*} = E^1_{1*} \setminus E^2_{1*}$; $E^3_{2*} = E^1_{2*} \setminus E^2_{2*}$; $E^3_{3*} = E^1_{3*} \setminus E^2_{3*}$; $E^3_{4*} = E^1_{4*} \setminus E^2_{4*}$; k^* -метка, по которой выбирается макросеть M^3_{i1} , описывающая $q_{i1} \in Q$ как индивидуум в дополнение к макросети $M^3_{i1} = M^1_{i1} \setminus M^2_{i1}$.

Операция идентификации объектов может иметь следующее содержание:

1) объект q_{i1} идентичен объекту q_{i1} по своему содержанию, если каждой микросети m^i_{i1} M^1_{i1} , описывающей объект q_{i1} макросети M^1_{i1} , биективно соответствует одна из микросетей m^i_{i1} M^2_{i1} макросети M^2_{i1} , описывающей объект q_{i1} , а $M^1_{i1} = M^2_{i1}$;

2) объект q_{i1} идентичен объекту q_{i1} по своей структуре, если каждой микросети e^1_{i23} E^1_{2*} , описывающей структуру этого объекта сети E^1_{2*} , биективно соответствует микросеть e^2_{i23} E^2_{2*} сети, описывающей структуру объекта q_{i1} , а $E^1_{2*} = E^2_{2*}$;

3) объект q_{i1} идентичен объекту q_{i1} по функциональным возможностям, если каждой микросети $e_{i24}^1 E_3^{1*}$, описывающей свойства объекта q_{i1} сети E_3^{1*} , биективно соответствует $e_{i24}^2 E_3^{2*}$ сети E_3^{2*} , описывающей свойства объекта q_{i1} , а $E_3^{1*} = E_3^{2*}$;

4) объект q_{i1} идентичен объекту q_{i1} по свойствам и признакам, если каждому элементу микросети $e_{i25}^1 E_4^{1*}$, описывающей признаки объекта q_{i1} сети E_4^{1*} , биективно соответствует микросеть $e_{i25}^2 E_4^{2*}$, описывающая признаки объекта q_{i1} , а $E_4^{1*} = E_4^{2*}$.

Операции частичной идентификации могут быть определены следующими способами:

1) объект q_{i1} аналогичен объекту q_{i1} по заданной характеристике m_{i21}^1 , если $(m_{i21}^1 M_{i1}^1) \& (m_{i21}^2 M_{i1}^2)$, где M_{i1}^1 и M_{i1}^2 макросети, соответственно описывающие объекты q_{i1} и q_{i1} ;

2) объект q_{i1} аналогичен объекту q_{i1} по ряду характеристик, если $M_{i1}^1 M_{i1}^2$;

3) объект q_{i1} является составной частью объекта q_{i1} , если $E_2^{1*} \subset E_2^{2*}$, где E_2^{1*} и E_2^{2*} -соответственно сети, описывающие структуру объектов q_{i1} и q_{i1} .

Введем параметр, определяющий степень $r(q_{i1}, q_{i1})$ аналогичности объектов q_{i1} и q_{i1} . Тогда, если $r(q_{i1}, q_{i1})$ принимает значение равное не менее заданному порогу h (обычно $h > 0.5$), то считается, что объекты q_{i1} и q_{i1} аналогичны между собой. Степень аналогичности объектов будем определять эвристическим путем исходя из условия, что чем больше мощность макросети $M = M_{i1}^1 M_{i1}^2$, содержащей общие для сравниваемых объектов микросети, тем аналогичнее эти объекты между собой. При этом необходимо учитывать относительное количество одинаковых микросетей m_{i21} к общему их числу, используемому для полного описания сравниваемых объектов. Рассмотренным условиям удовлетворяет степень аналогичности объектов q_{i1} и q_{i1} вычисляемая следующим образом $r(q_{i1}$ и

$q_{i1}) = [|M| / |M^1_{i1}| + |M| / |M^2_{i2}|] / 2$, где $|M^1_{i1}|$ - мощность макросети, описывающей объект q_{i1} , т.е. количество микросетей, определяющих ее <тело>; $|M^2_{i2}|$ - мощность макросети, описывающей объект o'_{i1} ; $|M|$ - мощность макросети $M = M^1_{i1} \cup M^2_{i2}$. Здесь деление на двойку производится для того, чтобы выполнялось условие $r(q_{i1}, q_{i1}) = 1$ при сравнении абсолютно одинаковых объектов.

В неопределенных условиях функционирования зачастую невозможно дать подробное четкое описание произвольных объектов ПС, а также количественно оценить параметры, определяющих их характеристик. Обойти отмеченные трудности можно путем представления множества характеристик ПС субъективным образом в виде нечеткого множества и качественного их описания в содержательной форме.

В этом случае каждый объект ПС $q_{i1} \in Q$ будет определяться нечетким множеством характеристик $X_i = \langle m_j, x^i_{j4} \rangle$, $j_4 = 1, m_4$ и обозначаться $q_{i1}(x_{i1})$ где $m^i_{j_4} \in [0,1]$ - степень присущности характеристики $x^i_{j_4}$ к нечеткому множеству X_{i1} или к соответствующему этому множеству объекту. В случае, когда для всех $x^i_{j_4} \in X_{i1}$ выполняется условие " $m^i_{j_4} = 1$ ", т.е. все степени принадлежности равны единице, имеем четкое множество характеристик описания объекта.

В свою очередь, каждая характеристика объекта ПС может быть задана тройкой $\langle x_i, \mu_i(x_i), T_j \rangle$, где x_i - количественное значение параметров характеристики x_i ; T_j - качественное значение параметров характеристики, определяемое термом лингвистической переменной $T_{i26} \in T$, описывающей эту характеристику (на практике обычно $j=1,5$). $T = \{T_{i26}\}, i_{26} = \overline{1, n_{26}}$ - множество известных лингвистических переменных, биективно соответствующих семантическому определению известных ИС понятий.

В общем случае, каждая лингвистическая переменная может быть представлена набором $R_i = (F_i, T_i, U_i, Z_i, N_i)$, где F_i - наименование лингвистической переменной, например, "расстояние между объектами ПС".

$T_i = R_i^j$ - терм-множество или множество всех названий лингвистических значений переменной x_i , определяющихся нечеткой переменной со значениями из универсального множества;

U_i -универсальное множество значений с базовой переменной x_i ; Z_i -синтаксические правила, порождающие название термов лингвистической переменной; N_i - семантические правила, которые определяют смысловое содержание $N_i(x_i)$ нечеткой переменной, относящейся к x_i .

При этом следует иметь в виду, что множество лингвистических переменных R можно разбить на два подмножества $R_1, R_2 \subset R$ таких, что в R_1 входят только те переменные, которые могут оцениваться как качественно, так и количественно, а в множество R_2 входят только те переменные, которые могут быть оценены лишь качественно, например, лингвистическая переменная "длина" может быть оценена как качественно, так и количественно, а переменная "красота" может быть оценена только качественно. Таким образом, сравнение качественно оцениваемых переменных может быть выполнено только на содержательном нечетком уровне описания. Следовательно, нечетко заданная модель представления объектов ПС является более адекватной и более удобной для распознавания и сравнения образов ПС.

Следует указать, что на нечеткую теоретико-множественную модель объекта ПС естественным образом распространяются все основные положения вышерассмотренной сетевой модели представления объектных значений ИС.

1.3. РАЗРАБОТКА МОДЕЛИ ОПИСАНИЯ СИТУАЦИИ ПРОБЛЕМНОЙ СРЕДЫ

Характерной особенностью ИС способных функционировать в условиях неопределенности является то, что знания таких систем должны быть не только структурированы, но и представлены безотносительно к конкретным условиям функционирования. Для описания ситуаций ПС

безотносительно к конкретной области можно использовать нечеткие семантические сети (НСС) активного и пассивного типа [50].

Формально нечеткая семантическая сеть является ориентированным нечетким мультиграфом $G_1=(V_1,E_1)$, где $V_1=\{v_{i1}\} \quad i_1=1,n_1$ и $E_1=\{e_{i2}\} \quad i_2=1,n_2$ соответственно множество вершин и ребер. Вершины $v_i \in V$ биективно соответствуют объектам ПС, ребра – отношениям, складывающимся в среде между этими объектами. Вершины $v_i \in V$ могут быть двух типов: свободные v_i^* и занятые v_i^0 . Каждая свободная (активная) вершина $v_i^* \in V_1$ определяется множеством характеристик X_i , которым должны обладать конкретные объекты $q_{i1} \in Q$, чтобы была разрешена пометка этой вершины их именами в конкретной (текущей) ситуации ПС. После выполнения такой пометки активная вершина v_i^* становится пассивной v_i^0 и определяется множеством характеристик X_i конкретного объекта, которым она помечена. Иными словами, активная вершина $v_i^* \in V_1$ помечается объектом $q_{i1}(X_i) \in Q$, если выполняется условие $X_i \subseteq X_{i1}$, где запись $q_{i1}(X_{i1})$ означает, что объект q_{i1} описывается множеством характеристик X_{i1} .

Ребра $e_i \in E$ или отношения между объектами ПС задаются тройками $\langle x_i, \mu_i(x), T_j \rangle$, где T_j – нечеткое значение (терм) лингвистической переменной $x_i \in T^*$; $\mu(x_i) \in [0,1]$ – степень принадлежности количественного значения лингвистической переменной $x_i \in T^*$ к интервалу численных значений термина T_j ; $T^* \subseteq R$, где x – множество лингвистических переменных, биективно соответствующих семантическому определению различных отношений.

В рассмотренном случае, при описании НСС ограничения, определяемые элементами T_j терм – множества $T_j(x_i)$ лингвистической переменной R_i и накладываемые на базовые переменные $x_i \in U_i$ задаются четко и вычисляются, исходя из функционального назначения и возможностей ИС. Иначе говоря, множество U_i разбивается на $j=1,R$

непересекающихся открытых справа интервалов согласно заданному на его элементах отношению эквивалентности «находиться внутри j интервала» [45].

Для моделирования процесса самообучения ИС используется тестовый пример следующего содержания. На рабочем участке станка с ЧПУ расположена подставка и стеллаж, на котором лежит заготовка. Роботу необходимо перенести заготовку и вставить ее в рабочий орган станка с ЧПУ. Стеллаж находится на расстоянии $a_1 = 1$ м от ИР и $a_2 = 5$ м от станка. Подставка расположена на расстоянии $a_3 = 0,2$ м от станка и $a_4 = 5$ м от робота. Рабочая зона манипулятора робота задается полусферой радиусом $r_1 < 1$ м. Зона видимости технического зрения робота равна $r_2 < 10$ м.

Пусть оценивается расстояние от робота до подставки с помощью термов лингвистической переменной «расстояние» <совсем рядом>, <рядом>, <близко>, <недалеко>, <далеко>. Для термов: T_1 – «совсем рядом» - верхнее граничное значение $r_1 = 1.5$ м выбирается, исходя из размеров рабочей зоны манипулятора робота; T_2 – «рядом» - граница этого терма $r_1 < r_2 < 10$ м; T_3 – «близко» - верхняя граница $r_3 = 10$ м этого терма берется равной расстоянию, на котором система технического зрения робота способна воспринимать и распознавать известные ему объекты, т.е. численное значение r_2 равно верхнему пределу зоны восприятия робота; T_4 – «недалеко» - граница этого терма $r_3 < r_4 < 30$; T_5 – «далеко» - граница этого терма $r_4 < r_5 < 50$ м.

Для перехода от количественных значений отношений между объектами ПС, измеренных при помощи информационно-измерительной системы ИС и определяемых базовыми значениями x_i лингвистических переменных, к качественным их значениям, т.е. к одному из термов T_j можно использовать преобразования следующего вида:

$$F: x_i \rightarrow \begin{cases} T_j, & \text{если } 0 < x_i < x_i^* ; \\ T_j, & \text{если } x_i^* < x_i^{**} < x_{i+1}^* ; \end{cases} \quad (1.1)$$

$$\left\{ T_j, \text{ если } x_{ik-1}^* < x_{i+1}^* \quad x_{ik}^* \right\}$$

где x_i^* и x_{i+1}^* , $i = \overline{1, k}$ соответственно нижняя и верхняя граница числовых значений x_i^* терма T_j .

Степень принадлежности $\mu(x_i)$ значений базовой переменной x_i^* к множеству числовых значений терма T_j может вычисляться согласно следующему характерному числовому выражению

$$\mu(x_i) = \begin{cases} \frac{x_i - x_i^*}{x_i^{**} - x_i^*}, \text{ если } x_i \in [x_i^*, x_i^{**}]; \\ \frac{x_i - x_i^{**}}{x_{i+1}^* - x_i^{**}}, \text{ если } x_i \in [x_i^{**}, x_{i+1}^*]; \end{cases} \quad (1.2)$$

где $[x_i^*, x_{i+1}^*]$ - интервал численных значений терма T_j ; $[x_i^{**}]$ - середина интервала числовых значений этого терма.

Рассмотренный подход определения компонентов тройки $\langle x_i, \mu(x_i), T_j \rangle$ при помощи выражений (1.1) и (1.2) позволяет легко переходить от количественного значения отношения, определяемого базовой переменной x_i , к качественному значению отношения, выраженному парой $\langle \mu(x_i), T_j \rangle$, а также осуществлять обратный переход от качественного значения к количественному при функционировании в конкретных условиях ПС [90].

Для сравнения двух значений отношений между собой, заданных тройками $\langle x_i, \mu(x_i), T_j \rangle$ и $\langle x_i, \mu(x_i), T_j \rangle$ введем характеристику степени равенства (близости) $\rho(x_i, x_i)$, которая может вычисляться следующим образом:

$$\rho(x_i, x_i) = \begin{cases} 1, \text{ если } (|x_i - x_i| < \varepsilon_0) \& (T_j = T_j); \\ \mu(x_i) \leftrightarrow \mu(x_i), \text{ если } |x_i - x_i| > \varepsilon_0 \& (T_j = T_j) \& ((x_i, x_i) \in [x_i^*, x_i^{**}] \& (x_i, x_i) \in [x_i^{**}, x_{i+1}^*]); \\ 1 - (\mu(x_i) \leftrightarrow (1 - \mu(x_i))), \text{ если } |x_i - x_i| > E \& (T_j = T_j) \& \\ \& ((x_i, x_i) \in [x_i^*, x_i^{**}] \& (x_i, x_i) \in [x_i^{**}, x_{i+1}^*]); \\ 0, \text{ если } T_j \neq T_j, \end{cases} \quad (1.3)$$

где ε_0 - параметр, задающий приведенное значение точности сравнения величины отношения; операция нечеткой эквивалентности, определяемая по

формуле $\min(\max(\mu(x_i), (1 - \mu(x_i))), \max(\mu(x_i), (1 - \mu(x_i))))$; & - конъюнкция, оказывающая одновременность выполняющихся условий.

Приведенное выражение (1.3) можно обосновать следующим образом. Два количественных значения отношения $x_i \rightarrow \langle \mu(x_i), T_j \rangle$ и $x_i \rightarrow \langle \mu(x_i), T_j \rangle$ равны между собой, если они попадают в интервал численных значений терма T_j в окрестность одной и той же точки, определяемую значением параметра ε_0 ; два количественных значения x_i и x_i нечетко равны между собой, если они принадлежат интервалу численных значений одного и того же терма T_j , и, наконец, значения x_i и x_i не равны между собой, если они попадают в интервалы численных значений различных термов лингвистической переменной T_j . Следовательно, два значения одного и того же отношения равны при $\rho(x_i, x_i) = 1$. Эти значения отношения являются нечетко равными, если $\rho(x_i, x_i) > 0$ и они не равны в случае, когда $\rho(x_i, x_i) = 0$.

Определение 1.6. Если НСС $G_1=(V_1, E_1)$ содержит только пассивные вершины $v_i^* \in V_1$,то такая сеть называется пассивной и определяет конкретную ситуацию ПС.

Определение 1.7. Нечеткая семантическая сеть, включающая только одни активные вершины $v_i^* \in V_1, i = \overline{1, n}$ называется нормально активной, т.е. она задана безотносительно к конкретной предметной области.

Таким образом, при пометке свободных вершин $v_i^* \in V_1$ в активной сети G_1 конкретными объектами $q(x_{i1}) \in Q$ ПС, а ребер конкретными значениями отношений между этими объектами, сеть G_1 становится нормально пассивной и определяет текущую ситуацию среды $S_{\text{итек}} \in S$.

Очень часто у ИС в процессе принятия решений возникает необходимость сравнения между собой различных нечетких семантических сетей. Для этой цели вводятся операции определения нечеткого изоморфизма и равенства НСС [50].

Определение 1.8. Две произвольные НСС $G^1_1=(V^1_1,E^1_1)$ и $G^2_1=(V^2_1,E^2_1)$ называются структурно эквивалентными, если они равны без пометок, или соответствующие им графы изоморфны.

Определение 1.9. Две произвольные НСС $G^1_1=(V^1_1,E^1_1)$ и $G^2_1=(V^2_1,E^2_1)$ называются нечетко изоморфными, если для них выполняются условия:

- а) они структурно эквивалентны;
- б) $(\forall v^1_i \in V_1) (\exists v^2_i \in V^2_1)[(X_1=X_2) \cup (X^*_1=X^*_2) \cup (X^*_2 \subset X^0_1) \cup (X^*_1 \subset X^0_2)]$;
- в) $(\forall e^1_i \in E^1_1) (\exists e^2_i \in E^2_1) [T_j, T_j \in T_j (R_i)]$,

где x_i и x_i - множества характеристик, которыми помечены вершины $v_i \in V_1$, если они соответственно являются пассивными или активными, $j=1,2$; T_j и T_j - термы лингвистической переменной R_i , которыми помечены ребра $e^1_i \in E^1_1$ и $e^2_i \in E^2_1$, занимающие одинаковые позиции в структуре сетей G^1_1 и G^2_1 ;
 \cup - логическая дизъюнкция.

Определение 1.10 Две произвольные НСС являются нечетко равными, если для них выполняются условия пп. "а" и "б" определения 1.9 и $(\forall e^1_i \in E^1_1) (\exists e^2_i \in E^2_1) [\rho(e^1_i, e^2_i)=1]$, где $\rho(e^1_i, e^2_i)$ - степень нечеткого равенства пометок ребер $e^1_i \in E^1_1$ и $e^2_i \in E^2_1$, занимающих одинаковые позиции в структуре сетей G^1_1 и G^2_1 , вычисляемая по (1.3).

Следовательно, если $\rho(G^1_1, G^2_1)=1$, то сети G^1_1 и G^2_1 являются нечетко равными, в противном случае при $r_1(G^1_1, G^2_1)=0$ сети G^1_1 и G^2_1 не являются нечетко изоморфными, и между ними имеется, как минимум одно различие, по значениям пометок структурно эквивалентных дуг. При $\rho_1(G^1_1, G^2_1) > 0$ сети G^1_1 и G^2_1 являются также нечетко изоморфными, но при этом в них отсутствуют различия между пометками ребер, занимающих одинаковые позиции в их структуре. Однако следует указать, что приведенная оценка для определения степени нечеткого изоморфизма двух произвольных НСС не позволяет решить, например, следующую задачу: насколько сеть G^1_1 по своему содержанию ближе к сети G^2_1 , чем сеть G^3_1 при условии, что при

попарном сравнении ребер этих сетей между ними имеются различия. Иными словами, независимо от количества различий между этими сетями будет выполняться условие $\rho_1(G^1, G^2) = \rho_1(G^2, G^3) = 0$.

Определение 1.11. Если НСС $G^1=(V^1, E^1)$ и $G^2=(V^2, E^2)$ удовлетворяют пп "б" и "в" определения 1.9 и $(V^1 \subset V^2) \& (E^1 \subset E^2)$, то сеть G^1 является вложено - нечетко изоморфной в сеть G^2 и обозначается $G^1 \simeq G^2$.

Определение 1.12 Если НСС $G^1=(V^1, E^1)$ и $G^2=(V^2, E^2)$ удовлетворяют п "б" определения 1.9, и определения 1.10 и $(V^1 \subset V^2) \& (E^1 \subset E^2)$, то G^1 является вложено-нечетко равной сети G^2 и обозначается как $G^1 \subset G^2$.

Под покрытием активной сети $G^*_1=(V^*_1, E^*_1)$ будем понимать пассивную сеть $G^0_1=(V^0_1, E^0_1)$, для которой однозначное отображение $d: G^*_1 \rightarrow G^0_1$ осуществляется путем выполнения подстановок $a_v: v^*_i \rightarrow v^0_i$, допустимых, если $X^*_i \subset X^0_i$, и $a_e: e^*_i \rightarrow e^0_i$ допустимых, если эти дуги помечены одинаковыми терминами одной и той же лингвистической переменной, где $v^*_i \in V^*_1, i = \overline{1, n}; v^0_i \in V^0_1; X^*_i$ и X^0_i - множество характеристик, которыми соответственно помечены вершины v^*_i и v^0_i ; $e^*_i \in E^*_1; e^0_i \in E^0_1$.

Тогда будем утверждать, что пассивная сеть G^0_1 поглощается активной сетью G^*_1 , если сеть G^0_1 покрывает сеть G^*_1 . Другими словами, активная сеть $G^*_1=(V^*_1, E^*_1)$, поглощающая различные пассивные сети $G^0_1=(V^0_1, E^0_1)$, является их обобщением, учитывающим ограничения, накладываемые действиями ИС на объекты среды. Выполнение таких ограничений в ПС позволяет ИС обрабатывать соответствующие им действия над объектами в целях необходимых изменений значений отношений, складывающихся в ПС между данными объектами.

Предложение 1.1. Активная сеть $G^*_1=(V^*_1, E^*_1)$ поглощает пассивную сеть $G^0_1=(V^0_1, E^0_1)$, если они являются нечетко изоморфными.

Доказательство. Поскольку согласно принятому условию сети G^*_1 и G^0_1 являются нечетко изоморфными, то из пп. "б" и "в" определения 1.9 следует, что для этих сетей выполняются условия

$$а)(\forall v^*_i \in V^*_1) (\exists v^0_i \in V^0_1)[X^*_i \subset X^0_i];$$

б)($\forall e^*_i \in E^*_1$) ($\exists e^0_i \in E^0_1$). Из первого условия вытекает допустимость подстановок $a_v: v^*_i \rightarrow v^0_i, i=1, n$, т.к. выполняются требования пометки активной вершины v^*_i сети G^*_1 объектом, которым помечена соответствующая вершина v^0_i сети G^0_1 .

Из второго условия вытекает допустимость подстановок $a_e: e^*_i \rightarrow e^0_i, i=1, n$, т.к. обе дуги имеют нечетко равные пометки $R^{*j}_i=R^{0j}_i$. Следовательно, отображение $d: G^*_1 \rightarrow G^0_1$ выполнимо, или активная сеть G^*_1 поглощает нечетко изоморфную ей пассивную сеть G^0_1 .

Предложение доказано.

Алгоритм для определения нечеткого изоморфизма между НСС, представленными матрицами смежности опишется следующим образом.

Алгоритм 1.1.

1. Проверить условие: $n^1 = n^2$? Если да, то перейти к п.3, если нет, то перейти к п.2.

2. Сети G^1_1 и G^2_1 не являются структурно эквивалентными, а следовательно, и нечетко изоморфными, конец.

3. Проверить условие: каждый j столбец матрицы N^*_1 имеет одинаково помеченный с ним j' столбец матрицы N^*_2 или $(\forall v^1_i \in V^1_1) (\exists v^2_i \in V^2_1) [(X^0_1 = X^0_2) \cup (X^*_1 = X^*_2) \cup (X^*_2 \subset X^0_1) \cup (X^*_1 \subset X^0_2)]$? Если да, то перейти к п.4; если нет, то перейти к п.2.

4. Проверить условие: порядковые номера j и j' для одноименных столбцов матриц N^*_1 и N^*_2 совпадают? Если да, то перейти к п.6; если нет, перейти к п.5.

5. Выполнить одновременную перестановку одних и тех же столбцов и строк $i'=j'$ матрицы N^*_2 в соответствии с порядковыми номерами $i=j$ одноименных с ними столбцов и строк матрицы N^*_1 . Иными словами,

сформировать матрицу N'_2 , изоморфную матрице N^*_2 , у которой номера столбцов и строк соответственно совпадают с номерами одноименных столбцов и строк матрицы N_1 .

6. Сравнивая элементы $a^{1_{ij}}$ и $a^{2_{ij}}$, для случаев $i=i'$ и $j=j'$, проверить условие: $T_j = T'_j$? Если да, то перейти к п.7; если нет, то перейти к п.2.

7. НСС G^1_1 и G^2_1 являются нечетко изоморфными; Конец.

Приведенный алгоритм может быть также использован для определения нечеткого равенства НСС G^1_1 и G^2_1 , только в этом случае в п.6 выполняется проверка условия « $\rho(e^1_i, e^2_i) = 1; i=1, n$ ».

Следует отметить, что алгоритм 1.1 для определения нечеткого изоморфизма между НСС имеет сложность порядка $O[l(n^2)]$, т.к. здесь сравниваются помеченные графы, где l - коэффициент пропорциональности.

Для упаковки информации в памяти ИС и для ее переработки в процессе принятия решений над НСС могут выполняться следующие нечеткие теоретико-множественные операции: объединение \cup , пересечение \cap , разности \setminus и симметрической разности Q .

Пусть $G^1_1 = (V^1_1, E^1_1)$ и $G^2_1 = (V^2_1, E^2_1)$ - произвольные НСС, определенные на одних и тех же множествах вершин и ребер.

Объединением НСС G^1_1 и G^2_1 называется сеть $G^3_1 = (V^3_1, E^3_1)$ и обозначается $G^3_1 = G^1_1 \cup G^2_1$, если $V^3_1 = V^1_1 \cup V^2_1, E^3_1 = E^1_1 \cup E^2_1$. При этом каждая вершина $v^{3*}_i \in V^3_1$ помечается согласно следующему правилу:

$v^1_i \in V^1_1$, если $(v^3_i \in V^1_1) \& (v^3_i \notin V^2_1)$;

$v^2_i \in V^2_1$, если $(v^3_i \in V^2_1) \& (v^3_i \notin V^1_1)$;

v^0_i , если $(v^3_i \in V^1_1) \& (v^3_i \in V^2_1)$ и обе 2 биективно соответствующие $v^3_i \in V^3_1$ вершины $v^3_i \in V^1_1$ и $v^2_i \in V^2_1$ являются пассивными;

v^*_i , если $(v^3_i \in V^1_1) \& (v^3_i \in V^2_1)$ и хотя бы одна из биективно соответствующих вершине $v^3_i \in V^3_1$ вершин v^1_i или v^2_i является активной,

где $\&$ - логическая конъюнкция ;

вершины $v^1_i \in V^1_1$ и $v^2_i \in V^2_1$ являются одноименными, если для них выполняется условие $[(X^0_1 = X^0_2) \cup (X^*_1 = X^*_2) \cup (X^*_2 \subset X^0_1) \cup (X^*_1 \subset X^0_2)]$;

\cup операция логической дизъюнкции.

Ребра $e^3_i \in E^3_1$ помечаются исходя из следующих условий:

а) если вершины инцидентные ребрам e^1_i и e^2_i в НСС являются разноименными, то

e^1_i , если $(e^3_i \in E^1_1) \& (e^3_i \in E^2_1)$;

e^2_i , если $(e^3_i \in E^1_1) \& (e^3_i \in E^2_1)$;

в) если вершины, инцидентные ребрам e^1_i и e^2_i являются одноименными, то $\langle \mu(x_i)^3, T_j \rangle$, $\mu(x_i)^3 = \max(\mu(x_i)^1, \mu(x_i)^2)$,

$T_j^3 = T_j^1$, если $T_j^1 = T_j^2$

$\langle \mu(x_i)^1, T_j^1 \rangle \& \langle \mu(x_i)^2, T_j^2 \rangle$, если $T_j^1 \neq T_j^2$,

где конъюнкция $\&$ обозначает одновременную пометку ребра $e^3_i \in E^3_1$ парами $\langle \mu(x_i)^1, T_j^1 \rangle$ и $\langle \mu(x_i)^2, T_j^2 \rangle$.

Пересечением НСС $G^1_1 = (V^1_1, E^1_1)$ и $G^2_1 = (V^2_1, E^2_1)$ называется сеть $G^3_1 = (V^3_1, E^3_1)$ и обозначается $G^3_1 = G^1_1 \cup G^2_1$, если $V^3_1 = V^1_1 \cap V^2_1, E^3_1 = E^1_1 \cap E^2_1$.

При этом каждая v^3_i вершина сети G^3_1 помечается согласно следующему правилу:

v^0_i , если $(v^3_i \in V^1_1) \& (v^3_i \in V^2_1)$ и обе одноименные вершины v^1_i и

v^2_i , биективно соответствующие v^3_i являются пассивными; $v^3_i \rightarrow v^*_i$,

если $(v^3_i \in V^1_1) \& (v^3_i \in V^2_1)$ и хотя бы одна из одноименных вершин v^1_i

и v^2_i биективно соответствующих v^3_i , является активной;

Ребра $e^3_i \in E^3_1$ помечаются следующим образом: $e^3_i = \langle \mu(x_i)^3, T_j^3 \rangle$

где $\mu(x_i)^3 = \min(\mu(x_i)^1, \mu(x_i)^2)$, когда $(e^3_i \in E^1_1) \& (e^3_i \in E^2_1)$, т.е. $T_j^1 = T_j^2 = T_j^3$;

$e^3_i = 0$ в противном случае.

Разностью нечетких семантических сетей G^1_1 и G^2_1 называется сеть $G^3_1 = (V^3_1, E^3_1)$ и обозначается $G^3_1 = G^1_1 \setminus G^2_1$, если $V^3_1 = V^1_1 \setminus V^2_1, E^3_1 = E^1_1 \setminus E^2_1$, где

\setminus - теоретико-множественная операция разности; $V^3_1 = v^1_i (v^1_i \in V^1_1) \& (v^1_i \notin V^2_1)$,

вершина $v_i^1 \in V^2_1$ тогда и только тогда, когда не будет существовать ни одной вершины такой, чтобы для v_i^1 и v_i^2 выполнялось условие:

$$[(X^0_1 = X^0_2) \cup (X^*_1 = X^*_2) \cup (X^*_2 \subset X^0_1) \cup (X^*_1 \subset X^0_2)];$$

$$(\exists e^1_i \in E^1_1) \& (\exists e^1_i \in E^2_1), \text{ дуга } e^1_i \in E^2_1$$

тогда и только тогда, когда для всех дуг $e^2_i \in E^2_1$ выполняется условие $\rho(e^1_i, e^2_i) = 0$. Следует иметь в виду, что данная операция не является симметричной, т.е. $(E^1_1 \setminus E^2_1) \neq (E^2_1 \setminus E^1_1)$.

Симметрической разностью НСС G^1_1 и G^2_1 называется сеть $G^3_1 = (V^3_1, E^3_1)$ и обозначается $G^3_1 = G^1_1 \ominus G^2_1$, если $V^3_1 = (V^1_1 \setminus V^2_1) \cup (V^2_1 \setminus V^1_1)$, а $E^3_1 = (E^1_1 \setminus E^2_1) \cup (E^2_1 \setminus E^1_1)$.

Конкретизацией активной сети G^*_1 называется пассивная сеть G^0_1 , полученная из первой путем пометки ее вершин конкретными объектами ПС согласно условию допустимости такой пометки.

Склеиванием двух сетей G^1_1 и G^2_1 называется и обозначается сеть $G^3_1 = G^1_1 * G^2_1$, $G^3_1 = (V^3_1, E^3_1)$, которая получается в результате введения между одноименными вершинами в сетях G^1_1 и G^2_1 неориентированных ребер, помеченных нулями.

Разрезанием сети G^3_1 на компоненты связности $G^i_1 \in G^3_1$ называется удаление из ее структуры неориентированных ребер, помеченных нулями.

С учетом выражений (1.1) и (1.2) мультиграф, биективно соответствующий описанной ситуации ПС, представлен на рис.1.

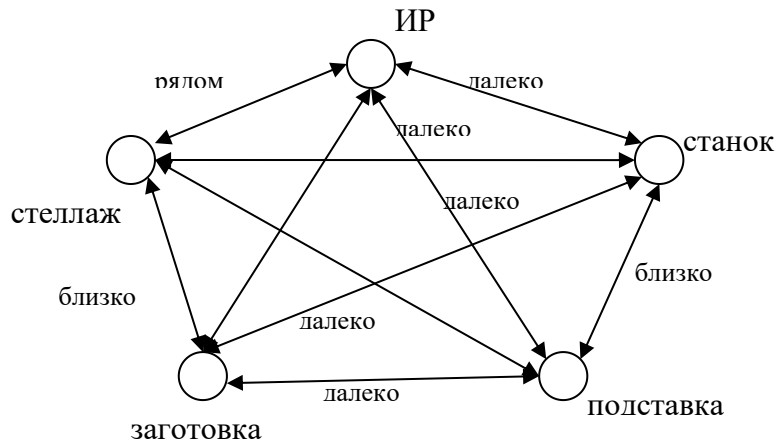


Рис.1

Аналитически этот граф можно описать следующим образом:

{ИР (((<<0.2/рядом>заготовка>) & <0.2/рядом>стеллаж>) &<1/далеко>станок>) &<1/далеко>подставка>)} & {станок ((<<0,2/рядом>подставка> & <<1/далеко>стеллаж>))} & {подставка ((<<0.2/рядом>станок>) & <<1/далеко>заготовка>)}, где &- конъюнкция, обозначающая одновременность происходящего события; "рядом", "близко", "далеко" - термы лингвистической переменной "расстояние". При аналитическом описании НСС выражение, заключенное в фигурные скобки называется отдельным фрагментом сети, а имена объектов, с которых начинаются фрагменты, определяются как ключевые понятия фрагментов. Фрагменты НСС, необходимые для описания ПС достаточного для принятия решений, можно определить эвристическим путем. Например, первый базовый фрагмент строится относительно ключевой вершины сети, помеченной понятием "ИР". Следующие необходимые фрагменты формируются относительно объектов, входящих в структуру заданного на текущий момент времени целевого условия, но над которыми ИС непосредственных действий не выполняет. Третий вид фрагментов строится в процессе функционирования относительно объектов, используемых ИС в качестве вспомогательных инструментов.

ГЛАВА 2. МОДЕЛИРОВАНИЕ ПРОЦЕССА САМООБУЧЕНИЯ ИС С ИСПОЛЬЗОВАНИЕМ АЛГОРИТМОВ САМООБУЧЕНИЯ НА НЕЧЕТКИХ СЕМАНТИЧЕСКИХ СЕТЯХ В СРЕДАХ ПЕРВОГО УРОВНЯ СЛОЖНОСТИ

2.1. ФОРМИРОВАНИЕ ПРОЦЕССА САМООБУЧЕНИЯ НА НЕЧЕТКИХ СЕМАНТИЧЕСКИХ СЕТЯХ В СРЕДАХ ПЕРВОГО УРОВНЯ СЛОЖНОСТИ

В средах первого уровня сложности ИС формируются программы целесообразного поведения (ПЦП) в виде простой цепи без различного рода разветвлений. Для этой цели могут быть использованы различные алгоритмы самообучения, выбор которых зависит от динамики протекающих в ПС процессов. К таким алгоритмам можно отнести процедуры с активно-пассивной логикой поведения, с активной логикой поведения; процедуры, связанные с организацией поисково-исследовательской деятельности, т.е. с закреплением формируемых программ поведения по новизне воспринимаемых сигналов, или по степени близости получаемых результатов процессе самообучения к достижению цели.

Рассмотрим ПС как множество взаимосвязанных между собой объектов и происходящих независимо от ИР событий $Q = \{q_i\}, i = \overline{1, n}$. В каждый конкретный момент времени t среду можно охарактеризовать текущей ситуацией $S_{\text{итек}} S$, определяемой текущими состояниями находящихся в ней объектов и характером отношений между объектами. Часть ситуаций S - множество $S = \{S_i\}, i = \overline{1, n}$ будем называть стандартными (безусловными). Эти ситуации определяют различные цели и подцели поведения ИС и вызывают у нее при восприятии соответствующие стандартные реакции (СР), связанные с достижением заданной цели $S_{\text{итек}}$. Под априорно неописанной будем понимать ПС, если:

- Заранее не известны закономерности преобразования ситуаций среды, происходящие в результате обрабатываемых роботом действий;
- Априорно не задано, каким образом влияют события $q_i \in Q$ на изменение ситуаций ПС.

Рассмотрим ИР, обладающего зрением и способного изменять ситуации ПС в результате обработки различных действий $b_{j1} \in B$. В самом общем случае, в условиях полной неопределенности, функционирование ИР может базироваться на алгоритмах самообучения (АС) или адаптации, в основе которых заложены метод проб и ошибок и эвристические процедуры, позволяющие сокращать количество обрабатываемых действий в процессе адаптации. Таким образом, задача ИР, способного целесообразно функционировать в условиях неопределенности, сводится к разработке АС, позволяющих формировать программы целесообразного поведения (ПЦП) на основе изучения различных закономерностей преобразования ситуации ПС.

При этом модель поведения ИР формируется в процедурной форме представления в виде множества ПЦП следующего содержания:

$S_{i_{\text{мек}}}^1 \ \& \ b_{j1}^1 \rightarrow S_{i+1_{\text{мек}}}^2 \ \& \ b_{j1}^2 \rightarrow S_{i+2_{\text{мек}}}^k \ \& \ b_{j1}^k \rightarrow S_{\text{цел}}$, где каждый одношаговый акт поведения, например, $S_{i_{\text{мек}}}^1 \ \& \ b_{j1}^1 \rightarrow S_{i+1_{\text{мек}}}^2$, означает, что при восприятии ИР ситуации $S_{i_{\text{мек}}}^1$ отработанное им действие b_{j1}^1 преобразует ее в ситуацию $S_{i+1_{\text{тек}}}$.

Таким образом, в самом общем случае, алгоритмы целесообразного поведения ИР могут быть сгруппированы в следующую иерархическую систему, приведенную на рис. 2.1.

Первый уровень этой системы (нижний) определяет различные безусловные программы поведения, обрабатываемые в ответ на восприятие ИР безусловных сигналов $c_i \in C_2$, второй – ПЦП, выработанные при помощи АС в априорно неописанных условиях ПС, третий – АС, предназначенные для изучения закономерностей преобразования ситуаций ПС и формирования на этой основе различных ПЦП, четвертый – процедуры выбора формы и

алгоритма функционирования ИР, адекватного сложившимся в ПС условиям и приводящего к достижению заданной цели.



Рис 2.1. Иерархическая система алгоритмов целесообразного поведения ИР в условиях полной неопределенности (стрелки показывают возможные переходы от одной формы поведения к другой)

Определение 2.1. Под безусловными программами поведения следует понимать типовые действия (операции), заложенные в память при проектировании и обрабатываемые в ответ на восприятие в ПС соответствующего им безусловного сигнала c_i S .

Стандартные сигналы и реакции задаются в виде троек $J_i = \langle \langle S_{тек} \rangle, СР, \langle \text{цель достигнута} \rangle \rangle, i = \overline{1, n}$. Например, пусть для ИР целевое условие функционирования определяется ситуацией, когда объект находится в рабочей зоне манипулятора. Для другого случая стандартной является ситуация, когда этот объект попадает в пределы рабочей зоны манипулятора робота. В ответ на восприятие этой ситуации следует СР «выполнить захват». После имитации отработки СР заданная цель считается достигнутой.

Определение 2.2. Под ПЦП робота следует понимать последовательность взаимосвязанных правил поведения, выявленную в конкретных определенных условиях ПС и приводящую в этих условиях к достижению соответствующего сигнала (стандартной ситуации).

Другими словами, ПЦП представляют собой последовательность одношаговых актов поведения, позволяющих преобразовать каждую

ситуацию $S_{\text{итек}} \in S$, закрепленную в этой программе, в определенную стандартную ситуацию $S_{\text{итек}} \in S_{\text{цел}}$.

Определение 2.3. Алгоритмы функционирования ИР, позволяющие автоматически выявлять различные по своей структуре одношаговые акты поведения, связанные с достижением стоящей цели и на этой основе формировать ПЦП, будем называть алгоритмами самообучения.

Выбор конкретного алгоритма функционирования, адекватного сложившимся в ПС условиям, определяется динамикой протекающих в среде процессов, структурой взаимодействия ее объектов и различных событий $q_i \in Q$. При этом ПС могут быть статические, динамические, ациклические, циклические, первого и второго уровня сложности [48].

Определение 2.4. Статической (квазистатической) называется ПС, в которой преобразование ситуаций $S_{\text{итек}} \in S$, из одной в другую на заданном промежутке времени t_0 происходит только в результате обрабатываемых ИР действий $b_j \in B$ и не зависит от событий $q_i \in Q$. В противном случае среда называется динамической, т.е. она может изменять свои состояния в результате не зависящих от работы внешних возмущений $q_i \in Q$.

Определение 2.5. Ациклической называется ПС, в которой в процессе формирования отдельной ПЦП отсутствует повторяемость ее ситуаций $S_{\text{итек}} \in S$ и событий $q_i \in Q$. В противном случае, среда называется циклической, т.е. в ней робот в процессе формирования отдельной ПЦП может неоднократно воспринимать одни и те же ситуации $S_{\text{итек}} \in S$.

Определение 2.6. Среда называется ПС первого уровня сложности, если формируемые в ней ПЦП являются только простыми цепями элементарных актов поведения. В противном случае ПС называется средой второго уровня сложности.

В одной и той же ситуации $S_{\text{итек}} \in S$, описывающей взаимодействие объектов ПС в пространстве и времени, ИР может формировать различные по назначению и содержанию ПЦП. Следовательно, текущая ситуация ПС

S_{itek} S сама по себе не однозначно определять характер действий робота в процессе целесообразного функционирования. При этом при аналогии с биологическими системами [50] важную роль в организации поведения ИР может приобретать не только ситуация, сколько знаки и их значения, т.е. так называемые пусковые и условные сигналы, которые как бы из истории поведения в данной ситуации локализуют только то, что непосредственно связано с достижением цели или стандартного сигнала. Ситуация же создает некоторый общий фон готовности ИР к определенной целенаправленной деятельности, для которой некоторые фрагменты ситуаций становятся знаками или условными сигналами, если они имеют индивидуальную особенность, позволяющую ИР отличать от других сигналов.

Таким образом, ПЦП могут определять следующей структурой

$$L(s_{i1}^*) = S_{imek}^1 \& b_{j1}^1 \rightarrow s_{i+1mek}^2 \& b_{j1}^2 \rightarrow s_{i+2mek}^k \& b_{j1}^k \rightarrow s_{цел},$$

где s_{i1}^* - описание фоновой ситуации, при восприятии которой ПЦП $L(s_{i1}^*)$

гарантировано, приводит к достижению стандартного сигнала $S_{цел}$;

S_i $S, i_2 = \overline{i, m_2}$ - множество условных сигналов, ассоциирующихся ИР с достижением сигнала $S_{цел}$ и определяемых в процессе самообучения по мере преобразования исходной ситуации ПС S_{itek} S .

Следует иметь в виду, что фоновая ситуация s_{i1}^* должна быть носителем только минимума необходимых и достаточных условий для достижения сигнала $S_{цел}$ в различных ситуациях ПС s_{imek}^t S . В общем случае $s_{i1}^* \rightarrow S_{цел}$.

Таким образом, для организации знаковосигнального принципа управления целесообразным поведением ИР в условиях полной неопределенности необходимо построить процесс самообучения робота так, чтобы для формируемых ПЦП были справедливы следующие правила вывода:

$$\frac{(S_{i+2mek}^k \& b_{j1}^k \rightarrow S_{i+2mek}^{k+1}) \& (S_{i+2mek}^k \rightarrow S_{цел})}{S_{imek \ j1}^k \rightarrow S_{i+2mek}^{k+1}}; (2.1)$$

$$\frac{(s_{i2}^* \rightarrow s_{i_{мек}}^t) \& L(s_{i1}^*) \rightarrow S_{цел}}{s_{i_{мек}}^t \& L(s_{i1}^*) \rightarrow S_{цел}}, (2.2)$$

где знак \rightarrow обозначает операцию условного вложения фоновой ситуации s_{i1}^* в текущую ситуацию ПС $s_{i_{мек}}^t$.

Здесь правило (2.1) означает, что если после восприятия сигнала $S_{i+2_{мек}}^k$ отработка действий b_j^k связана с появлением сигнала $S_{i+2_{мек}}^{k+1}$ и сигнал $S_{i+2_{мек}}^k$ содержится в ситуации $s_{i_{мек}}^t$, то отработка действий b_j^k в этой ситуации приводит к появлению в ней сигнала $S_{i_{мек}}^{k+1}$. Правило (2.2) показывает, что если фоновая ситуация s_i^* условно входит в структуру ситуации $s_{i_{мек}}^t$, то отработка в этой ситуации ПЦП $L(s_{i1}^*)$ приводит к появлению в ПС сигнала $S_{цел}$.

Для описания ситуаций проблемной среды (ПС) в процессе самообучения интеллектуальной системы (ИС) используются статические нечеткие семантические сети (НСС)[48].

Формально НСС представляет собой ориентированный нечеткий мультиграф $G_1=(V_1, E_1)$, где $V_1=\{v_i\}$, $i=1, n$, $E_1=\{e_i\}$, $i=1, n$ - соответственно множество вершин и ребер. Вершины V_1 биективно соответствуют объектам, находящимся и функционирующим в статической ПС, а ребра-отношениям, складывающимся в среде между этими объектами. При этом в статической ПС преобразование ситуаций $s_{i_{тек}} \in S$ из одной в другую на заданном промежутке t_0 происходит только в результате обрабатываемых ИС действий b_j B и не зависит от различных событий, протекающих в среде независимо от интеллектуальной системы. В противном случае среда называется динамической и будет описываться динамическими НСС, т.е. она может изменять свои состояния независимо от действий ИС.

Вершины $v_i \in V_1$ могут быть двух видов: свободные /слоты/ и занятые. Каждая свободная вершина $v_i^c \in V_1$ определяется множеством X_i допустимых характеристик, которыми должны обладать различные объекты $q_{i2} \in Q$, чтобы

была разрешена пометка этой вершины их именами в конкретной ситуации ПС. После выполнения указанной пометки вершина v_i^c становится занятой и задается множеством характеристик конкретного объекта o_{i2} . Другими словами, вершина v_i^c может быть помечена именем объекта $q_{i2} \in Q$, если $X_i \subseteq X_{i2}$.

Ребра $e_i \in E_1$, или отношения между объектами ПС характеризуются тройками $\langle x_i, \mu_i(x_i), T_j \rangle$, где T_j - нечеткое значение лингвистической переменной (ЛП) $T_j \in T$; $\mu_i(x_i) \in [0,1]$ - степень принадлежности значения базовой переменной ЛП x_i к интервалу численных значений терма T_j ; $T = \{x_i\}$, $i=1, n$ - множество ЛП, биективно соответствующих качественным значениям различных отношений.

Ограничения, определяемые элементами терм - множества ЛП $T(x_i)$ и накладываемые на значения базовых переменных x_i задаются четко. Они берутся исходя из предельных эксплуатационных характеристик ИР. Иначе говоря, множество U_i числовых значений каждой базовой переменной x_i разбиваются на $j=1, k_i$ непересекающихся открытых справа интервала согласно заданному на его элементах отношению эквивалентности "находиться внутри j интервала".

Таким образом, при пометке свободных вершин в нормально активной сети именами конкретных объектов, находящихся в ПС, а ребер конкретными значениями отношений, сеть становится пассивной и определяет текущую ситуацию среды. Две произвольные сети и называются структурно эквивалентными, если они равны без пометок.

Применение НСС для описания ситуаций ПС позволяет представить действия, обрабатываемые ИР в следующем формате: $\langle \text{имя действия} \rangle \langle \text{условия, выполнение которых в ПС требуется для успешной отработки действия} \rangle \langle \text{результат отработки действия} \rangle$.

Первая часть - $\langle \text{имя действия} \rangle$ является идентификатором действия. Вторая часть - $\langle \text{условия, выполнение которых в ПС требуется для успешной}$

отработки действия> - представляет собой активную НСС, формальное описание которой является мультиграфом $G_{1i}=(V_{1i},E_{1i})$, где V_{1i} - множество свободных вершин, каждая из которых помечается списком характеристик X_i , которыми должны обладать объекты, чтобы было допустимым выполнение над ними действия ФД.

Дуги E_{1i} представляются тройками $\langle x_i, \mu_i(x_i), T_j \rangle$, определяющими значения отношений между ИС и объектами, при выполнении которых робот может отработать над этими объектами действие ФД. В случае, когда требуемые условия для отработки действия не выполняются в ПС, то выявляются все различия между целевыми и текущими ситуациями и на этой основе, должен быть сформирован план поведения, позволяющий устранять эти различия. Для формирования такого плана в работе предлагается использовать алгоритм самообучения с имитацией обрабатываемых роботом действий.

Третья часть ФД - <результат отработки действия>- представляет собой НСС получаемую из сети G_{1i} после отработки действия этого фрейма. Например, типовой ФД имеет следующее описание <положить объект ОБ1* на объект ОБ2*> $\langle \{ \langle \text{ИР} // (\langle 0; \text{рядом} \rangle \langle \text{ОБ1}^* \rangle / \langle 0,5; \text{рядом} \rangle \langle \text{ОБ2}^* \rangle / \} \rangle \langle \{ \langle \text{ОБ1}^* \rangle \langle 0; \text{рядом} \rangle \langle \text{ОБ2}^* \rangle / \} \rangle$.

Исходная запись означает, что ОБ1* находится в схвате манипулятора, а ОБ2* - в пределах рабочей зоны робота. Тогда после имитации отработки действия ФД объекты ОБ1* и ОБ2* будут находиться на одном месте и объект ОБ1* будет лежать на ОБ2*. При этом активная вершина $\langle \text{ОБ1}^* \rangle$ задается списком характеристик {иметь размеры m_1^* , иметь вес m_2^* }, а слот $\langle \text{ОБ2}^* \rangle$ - {иметь плоскую поверхность площадью S_0 }, где m_1^* , m_2^* .

2.2. ФОРМИРОВАНИЕ ПРОЦЕССА САМООБУЧЕНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ С АКТИВНОЙ ЛОГИКОЙ ПОВЕДЕНИЯ НА НЕЧЕТКИХ СЕМАНТИЧЕСКИХ СЕТЯХ В СРЕДАХ ПЕРВОГО УРОВНЯ СЛОЖНОСТИ

Основная идея, закладываемая в алгоритм самообучения с имитацией отработки действий заключается в том, что в отличие от ранее описанных алгоритмов, в которых преобразование ситуаций из одной в другую осуществляется путем непосредственной реализации в ПС пробных действий, как уже отмечалось выше, здесь происходит имитация отработки выбранных действий на формальной модели среды [84,93,94].

Имитация отработки осуществляется следующим образом. По содержанию НСС, определяющей текущую ситуацию ПС с помощью второй части модели описания действий, выявляются действия, которые могут быть непосредственно реализованы в среде, а по третьей части модели вносятся соответствующие действию изменения значений отношений между объектами среды. Если в результате имитации отработки действия текущая ситуация приближается к целевой ситуации по своему содержанию то, выявляется элементарный акт поведения следующего вида $S_{i_{тек}} \& b_j \rightarrow S_{тек}$, где приведенная запись означает, что при восприятии текущей ситуации ПС $S_{i_{тек}}$ отработка действия b_j приводит к ее преобразованию в результирующую ситуацию $S_{тек}$.

Структурированное описание алгоритма самообучения с имитацией отработки действий на модели ПС имеет следующее содержание.

Исходные данные: ситуации $S_{цел}$, $S_{исх}$, описание действий B

Входные переменные $S_{i_{тек}} \in S_{тек}, S_{цел}$.

Выходные переменные $S_{i_{тек}} \& B_j^1 \rightarrow S_{i+1_{тек}}^1 \& B_j^2 \rightarrow S_{i+2_{тек}}^2 \& B_j^3 \rightarrow \dots \rightarrow S_{цел}$.

1. Начало.
2. Описать $S_{i_{тек}}$ сформировав соответствующую ей НСС.

3. Выбрать действия $v_j \in V$, которые возможно непосредственно реализовано в среде в текущей ситуации $S_{\text{итек}}$.
4. Выполнить имитацию отработки выбранного действия $v_j \in V$ путем замены в НСС $S_{\text{итек}}$ соответствующих значений отношений на их значения из 3-ей части ФД "результат отработки действия".
5. Определить степень близости $\rho(S_{\text{итек}}, S_{\text{цел}})$ в текущей ситуации $S_{\text{итек}}$ и в ситуации $S_{i+1\text{тек}}$, затем проверить условие $\rho(S_{i+1\text{тек}}, S_{\text{цел}}) > \rho(S_{\text{итек}}, S_{\text{цел}})$ если да, то перейти к п.7, в противном случае перейти к п.6.
6. Восстановить в НСС $S_{\text{итек}}$ значения отношений, которые они имели до имитации отработки действий и переход на п.8
7. Исключить действие v_j из множества V .
8. Проверить условие: «выполнены все действия $v_j \in V$ и не найдено действие которое приблизило $S_{\text{итек}}$ к $S_{\text{цел}}$, то из V выбрать действие v_j такое, что $S_{\text{итек}} \& v_j \rightarrow S_{i+1\text{тек}}$, где $S_{i+1\text{тек}} \neq S_{\text{итек}}$
9. Запомнить цепочки $S_{\text{итек}} \& v_j \rightarrow S_{\text{цел}}$.
10. Если $S_{\text{итек}} = S_{\text{цел}}$, то перейти к п.11, если нет, то дальнейшее самообучение нецелесообразно, перейти к п.4.
11. Конец.

Согласно основному принципу, по которому строится алгоритм самообучения видно, что его эффективное применение возможно только в статической ПС, и т.к. самообучение сопровождается только активными шагами самообучения, а ПЦП формируется на основе этого алгоритма путем закрепления сигналов по степени близости с заданной целью. Такое закрепление позволяет избежать появления одношаговых актов поведения, обусловленных случайным совпадением во времени действий ИС и не зависящих от него преобразований ситуаций ПС.

При этом на первом этапе самообучения строится формальная модель исходной ситуации ПС в виде НСС. Далее осуществляется выбор действий v_j принадлежащих множеству действий V , которые согласно содержанию второй части описания ФД могут быть непосредственно отработаны в ПС.

При этом отработка выбранных действий имитируется, и тем самым в процессе самообучения не происходят реальные изменения среды, которые могут привести к ненужным ее преобразованиям. Затем путем сравнения исходной ситуации и целевой, а так же текущей и целевой ситуации определяется степень полезности результатов, полученных путем имитации преобразований среды. Если в процессе имитации отработанное действие приближает текущую ситуацию к целевой, то запоминается соответствующий акт поведения. В противном случае восстанавливается исходное содержание НСС путем преобразования ее отношений между объектами к исходным значениям, которые они принимали до имитации отработки действий.

Таким образом, в основу построенного алгоритма самообучения закладываются следующие основные механизмы самообучения:

- формирование программ целесообразного поведения (ПЦП) на основе их закрепления достижением цели;
- изучение закономерностей преобразования ситуаций ПС путем имитации отработки пробных действий;
- закрепление сигналов в ПЦП по новизне содержания воспринимаемых сигналов и степени близости получаемых результатов к цели при имитации отработки действий в ПС.

В основе первого принципа лежит механизм позволяющий изучать закономерности преобразования ситуаций ПС, отделяя их от случайных явлений, и тем самым обеспечивать отбор только полезной информации с точки зрения достижения цели. Второй принцип позволяет выявлять только необходимые для достижения цели действия. Третий принцип характеризует механизм, позволяющий ИС последовательно во времени наращивать цепь ПЦП, выявляя закономерности.

2.3. ФОРМИРОВАНИЕ ПРОЦЕССА САМООБУЧЕНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ С АКТИВНО-ПАССИВНОЙ ЛОГИКОЙ ПОВЕДЕНИЯ НА НЕЧЕТКИХ СЕМАНТИЧЕСКИХ СЕТЯХ В СРЕДАХ ПЕРВОГО УРОВНЯ СЛОЖНОСТИ

В алгоритме с активно-пассивной логикой сначала выполняется наблюдение за изменением ситуации. Если произошло самопроизвольное преобразование ситуации, то в этой вновь сформированной ситуации ИС по выше описанному принципу выявляет целесообразное воздействие и формирует отдельное звено целесообразного поведения, запоминая цепочку $S_{\text{итек}} \& b_j \rightarrow S_{\text{цел}}$. Далее принимается $S_{\text{цел}}$ за $S_{\text{итек}}$ и продолжаются наблюдения за изменением среды. Если процесс не привел к достижению цели, то ИС переходит к самообучению на основе активной логики поведения. В работе приведено структурированное описание указанных алгоритмов самообучения, определяющих число пробных действий, выполненных ИС в процессе самообучения в зависимости от числа объектов, характеризующих ПС. Указанная сложность имеет порядок $]n^2[$, n – число объектов ПС.

С учетом ранее рассмотренных механизмов адаптации и принятых обозначений, алгоритм самообучения ИС с активно-пассивной логикой поведения представится следующим образом [99].

Алгоритм 2.2.

Исходные данные: ситуации $S_{\text{исх}}$, $S_{\text{цел}}$, описание действий V ,

Входные переменные: $S_{\text{итек}} \in S$

Выходные переменные: $S_{\text{итек}} \& v_j \rightarrow S_{\text{итек}}^1 \& v_{j+1} \rightarrow \dots \rightarrow S_{\text{итек}}^n \& v_{jn} \rightarrow \dots \rightarrow S_{\text{цел}}$.

1. Начало.
2. Описать текущую ситуацию внешней среды $S_{\text{тек}}$.
3. Выполнить наблюдения за изменением текущей ситуации ПС.
4. Если произошло самопроизвольное изменение преобразованной ситуации, сформировать множество действий $V_i \in V$, которые можно обработать в ПС.

5. Выполнить имитацию отработки выбранного действия b .
6. Проверить условие: «находятся среди них действия, приводящие к достижению цели?» Если да, то перейти к п.8, если нет, перейти к п.7.
7. Исключить проверяемое действие b_j из числа реализуемых на текущем шаге самообучения.
8. Запомнить цепочку $S_{\text{итек}} \& V_j \rightarrow S_{\text{цел}}$. Принять $S_{\text{цел}}$ за $S_{\text{итек}}$, наблюдая за изменением среды.
9. Проверить условие: «цель достигнута?» Если нет, то перейти к п.4, если да, перейти к п.10.
10. Конец.

Из содержания видно, что в нем каждый активный шаг самообучения чередуется с "пассивным" шагом. При этом адаптация ИС в ПС происходит в динамике реального времени по мере изменения ситуаций в среде путем перехода системы от детерминированного функционирования, определяемого отработкой действий элементарных актов поведения, закрепленных в цепи формируемой ПЦП, к самообучению при попадании в новые условия среды.

Для выполнения анализа и количественной оценки сложности АС в различных условиях ПС вводим следующие ограничения:

1) обязательным условием выработки ПЦП является наличие или появление в ПС необходимого стандартного сигнала $S_{\text{цел}}$;

2) суммарная вероятность перехода ПС от состояния $S_{\text{итек}}$ к другому состоянию такому, что на каждом этапе активного самообучения ИС равна единице при условии, что интеллектуальная система выполнит все действия из множества V_j (частным случаем данного условия является то, что для каждого действия из V_j в ПС существует переход от одного состояния к другому);

3) действия ИС из множества V_j равновозможны в условиях априорной неопределенности;

4) структура представления ИС $S_{\text{итек}}$ S и действий b_j B в памяти ИС такова, что для хранения каждого из них достаточно заданного объема памяти m_0 ячеек (в общем случае, количество ячеек памяти m_0 определяется характером решаемых ИС задач, например, если в качестве ИС выступают состояния отдельных объектов, то $m_0=1$);

5) количество различных сигналов в ПС таково, что для хранения информации, связанной с формированием в среде любой по сложности ПЦП, достаточно оперативной памяти ИС;

6) из ограничений пп.4 и 5 вытекает, что для определения функциональной и емкостной сложности АС можно воспользоваться равномерными весовыми критериями, приняв за единицу функциональной сложности алгоритмов самообучения отработку ИС одного пробного действия, а емкостной - единичный объем m_0 её оперативной памяти.

Определение 2.7. Под функциональной сложностью АС будем понимать количество пробных действий, обрабатываемых ИС в процессе самообучения на основе этого алгоритма.

Определим параметр k_0 , характеризующий количество воспринимаемых ИС одних и тех же сигналов ПС на различных этапах адаптации в процессе формирования отдельной ПЦП.

Определение 2.8. Циклическая ПС называется средой n - категории сложности, если характеризующий ее параметр $k_0 = n$. При $k_0 = 1$ ПС называется ПС первой категории сложности и называется ациклической [48].

Следует отметить, что АС 2.2. в основном предназначен для адаптации к динамическим ПС. Однако, он может быть использован и для самообучения в статических средах. При этом, функционирование ИС должно носить только активный характер, по истечению заданного интервала времени, при условии, что в ПС не происходит самопроизвольных преобразований, ИС переходит к активным манипуляциям.

Формально взаимодействие ИС со статической ПС первой категории сложности можно представить в виде ориентированного графа $G_3 =$

(V_3, E_3) , $V_3 = \{v_i\}, i=1, n$ -множество вершин, биективно соответствующих состояниям среды; $E_3 = \{e_i\}, i=1, n$ - множество дуг, определяющих переходы ПС от состояния к состоянию, происходящие в результате обрабатываемых ИС действий. Таким образом, процесс самообучения ИС можно рассматривать как автоматическую генерацию графа G_3 при условии, что ей априорно не известны результаты, к которым могут привести обрабатываемые действия.

ВЫВОДЫ К ГЛАВЕ 2.

Представление процедурных знаний робота в виде фреймовых моделей позволяет ИС целенаправленно функционировать в различных объектах ПС и дает возможность обобщить решение разнообразных задач и сузить процесс поиска целесообразных форм поведения робота.

Представление процедурных знаний робота в виде фреймовых моделей имеет свои недостатки. ИС может решать только те задачи, которые предусмотрены в множестве в виде активных НСС, ситуаций ПС. Для наделения ИС возможностью целенаправленно функционировать, в условиях ПС разработаны алгоритмы самообучения в средах первого уровня сложности. Описанные алгоритмы самообучения обеспечивают определённые критерии достоверности обрабатываемых ИС сигналов. Вполне надёжный критерий достоверности определяется механизмом закрепления элементарных актов поведения на каждом шаге функционирования, новизной воспринимаемых в среде сигналов и совпадением выполнения выработанной программы поведения с достижением цели. Алгоритмы 2.1 и 2.2 увеличивают способности ИС к последующему самообучению.

В динамических ПС эффективными являются алгоритмы самообучения, построенные на основе алгоритма 2.2 с активно - пассивной логикой поведения. В такие алгоритмы самообучения закладывается двойное закрепление достижением цели вначале на активном шаге, а затем на пассивном. В статических ПС предпочтительней использовать алгоритмы

самообучения, построенные на основе алгоритма 2.1 с активной логикой поведения в сравнении с алгоритмом 2.2 и не требующего самопроизвольного преобразования ситуаций среды.

ГЛАВА 3. МОДЕЛИРОВАНИЕ ПОВЕДЕНИЯ ИС С ИСПОЛЬЗОВАНИЕМ АЛГОРИТМОВ САМООБУЧЕНИЯ НА НЕЧЕТКИХ СЕМАНТИЧЕСКИХ СЕТЯХ В СРЕДАХ ВТОРОГО УРОВНЯ СЛОЖНОСТИ

Уточним понятие ПС второго уровня сложности. Среда второго уровня сложности характеризуется наличием разветвленной взаимосвязи условных сигналов, определяющих и сопровождающих различные преобразования ее допустимых ситуаций.

Таким образом, конкретная ПС может быть отнесена к категории среды второго уровня сложности в следующих случаях:

- при наличии в среде причинно - следственных связей между условными сигналами и происходящими событиями;
- при появлении "тормозных сигналов", препятствующих отработке ранее сформированной программы целесообразного поведения, которая до появления этого сигнала в ПС приводила к достижению соответствующего стандартного сигнала;
- при появлении суммарных комплексов условных сигналов, определяющих необходимость одновременного выполнения в среде нескольких событий, связанных с достижением заданной цели.

Рассмотрим алгоритмы самообучения, позволяющие ИС адаптироваться к перечисленным условиям ИС.

3.1. АЛГОРИТМЫ САМООБУЧЕНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ ПРИ НАЛИЧИИ В СРЕДЕ ПРИЧИННО-СЛЕДСТВЕННЫХ СВЯЗЕЙ: ОПИСАНИЕ И АНАЛИЗ

Отметим, что при выполнении в ПС независимых от ИС событий, связанных с появлением в ней соответствующих им сигналов среду следует отнести к классу динамических сред. Иными словами, в такой ПС происходят независимые от интеллектуальной системы преобразования допустимых ситуаций, влияющих на процесс ее адаптации к изменяющимся условиям среды. Следовательно, в процессе самообучения ИС при наличии в

ПС причинно-следственных связей, обусловленных выполнением в ней различных по своему содержанию событий, необходимо исключить возможность закрепления в вырабатываемых ПЦП случайных элементарных актов поведения $S_{\text{итек}} \& b_j \rightarrow S^1_{\text{цел}}$, формирующихся в результате неверного определения причин изменения ситуаций среды.

Для этой цели при разработке алгоритма самообучения необходимо использовать механизмы действия многократного совпадения двух условных сигналов во времени. Приведем один из возможных принципов реализации этого механизма. Введем параметр δ , принимающий значения из интервала $[0,1]$ и определяющий степень уверенности ИС в истинности отдельного элементарного акта поведения. Считаем, что если параметр δ для соответствующего элементарного акта поведения в процессе многократной его реализации принимает значение, равное единице, то данное правило отражает закономерное преобразование ПС. В противном случае, при $\delta = 0$ принимается решение о том, что элементарный акт поведения носит случайный характер. Для программы целесообразного поведения, состоящей из k_2 решающих правил, значение степени уверенности ИС δ^* для эффективного применения этой программы поведения может определяться согласно выражению $\delta^* = \min_{w=1, \overline{k_2}} \delta_w$,

где δ_w – степень уверенности в истинности w элементарного акта поведения, входящего в структуру искомой ПЦП.

Если у ИС имеется несколько альтернативных ПЦП $L_i \in L$, приводящих к достижению определенного стандартного сигнала s^*_i , то предпочтение отдается наращиванию и выполнению программы целесообразного поведения с большим значением параметра δ^* . При наличии двух ПЦП с одинаковыми значениями параметра δ^* используется программа, имеющая меньшую сложность реализации.

Для реализации рассмотренного механизма многократного совпадения двух сигналов во времени, необходимо выбрать количество k_3 повторных совпадений условных сигналов, при котором принимается решение об

истинности формируемого элементарного акта поведения. Значение k_3 можно определить эвристическим путем, опираясь на накопленный опыт поведения или на основе результата опроса экспертов. Зная k_3 можно вычислить шаг приращения $\Delta\delta$ параметра δ_w , как $\Delta\delta = 1/k_3$. Отсюда, удачная отработка действия элементарного акта поведения поощряется увеличением параметра δ на $\Delta\delta$ т.е. $\Delta\delta = \delta_w + \Delta\delta$, в противном случае следует штраф и $\Delta\delta = \delta_w - \Delta\delta$.

Используя описанный механизм, ИС, наблюдая за происходящими в ПС событиями, принимает следующее решение: если после восприятия условного сигнала S_{meki} через интервал времени t появляется сигнал S_{i+1mek} и данное совпадение проявляется многократно, т.е. для него $\delta = 1$, то сигнал $S_{итек}$ рассматривается как предвестник появления условного сигнала, на который система реагирует определенным образом. Далее, если сигнал $S_{итек}$ появляется часто, например, периодически по истечении интервала времени t_2 , то изменений в характере поведения ИС не происходит. Робот ожидает самопроизвольного появления требуемого условного сигнала S_{i+1mek} и реагирует на него определенным образом согласно ПЦП, в которой закреплён этот сигнал.

В противном случае ИС принимает решение о необходимости достижения одного из условных сигналов $S_{итек}$ или S_{i+1mek} при помощи обрабатываемых им действий $b_{j1} \in B$. Цепочки действий, приводящие к появлению условных сигналов $S_{итек}$ и S_{i+1mek} в среде, ИС формирует на основании алгоритма самообучения 2.2 вырабатывая соответствующие ПЦП.

В случае, когда предвестниковый сигнал $S_{итек}$ предсказывает появление вредного или тормозного сигнала S_{i+1mek} для достижения стоящей перед ИС целью $S_{рез}$ то вырабатывается ПЦП, приводящая к выключению сигнала $S_{итек}$ до появления связанного условного с ним сигнала S_{i+1mek} . Если по истечении заданного промежутка времени t_2 ПЦП, приводящая к выключению сигнала $S_{итек}$, полностью еще не сформирована, то по мере

появления условного сигнала $S_{i+1мек}$ ИС вырабатывает ПЦП, связанную с его устранением, т.д., пока не будут выработаны обе ПЦП, приводящие к выключению тормозных сигналов $S_{итек}$ и $S_{i+1мек}$.

В случае неудачной попытки сформировать ПЦП, связанную с выключением (или включением) предвестникового сигнала $S_{i+1мек}$, ИС уточняет условный сигнал путем выявления независящего от него события $q_i \in Q$, приводящего к включению предвестника при наличии в ПС необходимых для этого условий, определяющихся условным сигналом $S_{i+2мек}$. После выработки элементарного акта поведения $S_{итек} \& q_i \rightarrow S_{i+1мек}$ путем наблюдения за происходящими в среде событиями формируется комбинированный условный сигнал $q_i \rightarrow S_{итек}$. Затем ИС выявляет действие $b_j \in B$, приводящее к появлению в среде необходимого события $q_i \in Q$ и формирует элементарный акт поведения $S_{итек} \& b_j \rightarrow q_i$. После этого формируется ПЦП, связанная с выполнением в ПС события $q_i \in Q$ с учетом различных допустимых ситуаций среды.

Для построения алгоритма самообучения при наличии в среде причинно-следственной взаимосвязи условных сигналов $S_{итек} \in S$ и событий $q_i \in Q$ используем ранее принятые обозначения при описании алгоритмов 2.1 и 2.2, а также следующие операторы:

t - отрезок времени, отведенный ИС под ожидание самопроизвольного появления в ПС необходимого события $q_i \in Q$;

t_m - отрезок времени, в течение которого ИС ожидает самопроизвольного появления в ПС события $q_i \in Q$;

\subset - обозначение операции нечеткого изоморфного вложения НСС, определяющей произвольный условный сигнал, в нечеткую семантическую сеть, описывающую текущую ситуацию ПС, например, $S_{итек} \subset S$;

$s^{1t} = s^{2t}$ - равенство ситуаций $s^{1t}, s^{2t} \in S$ между собой. ИС принимает решение, что $s^{1t} = s^{2t}$, если описывающие эти ситуации НСС G^{1t} и G^{2t} являются нечетко равными;

$S_{meki} \& S_{i+1mek}$ - обозначение одновременного присутствия в ПС сигналов S_{i+1mek} и S_{meki} .

С учетом принятых обозначений алгоритм самообучения ИС при наличии в ПС причинно-следственных связей представится следующим образом [92,93].

Алгоритм 3.1.

Исходные условия: ситуации $S_{исх}, S_{цел}$

Входные переменные: $S_{итек} \in S, q_i \in Q$.

Выходные переменные: элементарные акты поведения $S_{итек} \& b_{j1} \rightarrow S_{i+1mek}$;
 $(S_{i+2mek} \& S_{i+1mek}) \& b_{j1} \rightarrow S_{итек}$.

1.Начало

2.Описать текущую ситуацию внешней среды $S_{итек}$ и соответствующую ей НСС G_1 .

3.Запустить таймер отсчета времени t , наблюдая за изменением состояний ПС. С включением каждого нового события $q_i \in Q$ сформировать текущую ситуацию S_{i+1mek} и соответствующую ей НСС G_2 .

4.Проверить условие: $S_{итек} = S_{i+1mek}$? Если да, то перейти к п.5;если нет, то перейти к п.6.

6.Сформировать сигнал $a_m = G_2 \setminus G_1$ и $a'_j = G_1 \setminus G_2$; $\Delta\delta = \delta_w + 1/k_3$. Записать в память элементарный акт поведения

$$a'_i \& q_i \xrightarrow{T_m} a_m, \text{ где } T_m = t_1.$$

7.Принять за подцель функционирования в процессе самообучения условный сигнал $q_i \xrightarrow{T_m} a_m$.

8.Описать текущую ситуацию ПС $S_{теки+2}$ и сформировать соответствующую ей НСС G_3 .

9.Принять за подцель функционирования в процессе самообучения ПС появление события q_i .

10. Проверить условие: событие q_i включено? Если да, то перейти к п.16;если нет, то перейти к п.11.

11.Сформировать множество действий $V_j \in V$.

12.Выбрать действие b_j согласно равномерному закону распределения вероятности их выполнения.

13.Выполнить имитацию отработки действия b_j^* .

14.Проверить условие: событие q_i включено? Если да, то перейти к п.11; если нет, то перейти к п.16 .

15.Исключить действие b_j^* из множества V на текущем шаге самообучения.

16. После включения события q_i сформировать текущую ситуацию ПС $S_{i+3тек}$ и соответствующую ей НСС G_4 .

17.Проверить условие: $S_{цел} \subset G_4$? Если да, то выполняются стандартные реакции и перейти к п.18;если нет, то перейти к п.27.

18.Проверить условие: событие q_i включено в результате имитации отработки действия b_j^* ? Если да, то перейти к п.19 ;если нет, то перейти к п.20.

19.Сформировать сигнал $a'_i = G_3 \setminus G_4$; Занести в память элементарный акт поведения $a'_i \& b_j^* \rightarrow q_i$, $\delta_j = \delta_j + \Delta\delta$; сформировать элементарный акт поведения $(a_i \& a_i) \& b_j^* \rightarrow S_{цел}$; $\delta_j = \delta_j + 1/k_3$.

20.Проверить необходимость появления заданного сигнала $S_{цел}$. Если такая необходимость есть, то перейти к п.21; в противном случае закончить самообучение.

21.Сформировать текущую ситуацию ПС $S_{i+3тек}$ и соответствующую ей НСС G_5 .

22. Проверить условие: $a_i \in G_5$? Если да, то перейти к п. 24, если нет, то перейти к п. 23.

23. Используя алгоритм 2.2., сформировать ПЦП, приводящую к включению в ПС сигнала a_i . Перейти к п. 16.

24. Проверить условие: $a_i \leq G_s$? Если да, то перейти к п.25, если нет, то перейти к п. 27.

25. Провести имитацию отработки действия b_j^* . Проверить условие: в среде произошло включение события q_i ? Если да, то перейти к п.25; если нет, то $\delta_j = \delta_j - \Delta\delta$, перейти к п.26.

26. Проверить условие: $\delta > 0$? Если да, то перейти к п.20; если нет, то перейти к п. 3.

27. Используя алгоритм самообучения 2.2, сформировать ПЦП, приводящую к включению в ПС события q_i ; перейти к п.16.

25. Используя алгоритм самообучения 2.2, сформировать ПЦП, приводящую к включению в ПС сигнала $S_{\text{цел}}$.

26. Конец.

3.2. АЛГОРИТМЫ САМООБУЧЕНИЯ ПРИ ПОЯВЛЕНИИ В СРЕДЕ ТОРМОЗНЫХ СИГНАЛОВ: ОПИСАНИЕ И АНАЛИЗ

Алгоритмы выработки условных рефлексов высокоорганизованных живых существ при появлении в среде "тормозных сигналов" впервые были изложены в [21]. Здесь процесс формирования у живых существ сложных условных рефлексов (рефлексов второго и большего порядка [50]) рассматривается как одна из разновидностей полифазного поведения и сводится к выявлению "тормозных сигналов" на основе наблюдения за их самопроизвольным включением и выключением на фоне ранее закрепленных в условном рефлексе условных сигналов.

Для интеллектуального робота такая организация процесса адаптации к различным ситуациям ПС является неприемлемой, т.к. в реальных условиях функционирования промежутки времени между самопроизвольными включениями и выключениями "тормозных сигналов" могут иметь значительную протяженность. В этой связи, опираясь на описание ситуаций проблемной среды в виде нечетких семантических сетей возможна организация алгоритма самообучения ИС, позволяющего вырабатывать программы целесообразного поведения, связанные с выявлением и выключением "тормозных сигналов" в процессе функционирования в квазистационарных средах. После выключения "тормозного сигнала" ИС приступает к отработке ранее сформированной программы целесообразного поведения, связанной с достижением заданного стандартного сигнала s^* . Иными словами, можно разработать алгоритм самообучения, позволяющий устанавливать взаимосвязь между появлением в проблемной среде новых условных сигналов и связанной с ними неудачной отработкой ранее сформированной в этой среде программы целесообразного поведения [61].

В общем случае при появлении в проблемной среде тормозных сигналов процесс самообучения ИС складывается из трех взаимосвязанных этапов:

- первый этап сводится к выработке основной цепи программы целесообразного поведения, приводящей к достижению определенного стандартного (целевого) сигнала $S_{цел}$. При этом формируется программа целесообразного поведения следующего содержания

$$S^1_{i_{тек}} \& b^1_j \rightarrow S^2_{i+1_{тек}} \& b^2_j \rightarrow \dots \rightarrow S^k_{i+n_{тек}} \& b^k_j \rightarrow S_{цел};$$

- на втором этапе на фоне одного из условных сигналов $S^1_{i_{тек}} - S^k_{i+n_{тек}}$ основной цепи программы поведения, выявляется тормозной сигнал (ТС), например,

$$S^1_{i_{тек}} \& b^1_j \rightarrow S^2_{i+1_{тек}} \& b^2_j \rightarrow \dots \rightarrow (S^k_{i+n_{тек}} \& TC) \& b^k_j \rightarrow S_{цел},$$

$S^1_{i_{тек}} \& b^1_j \rightarrow S^2_{i+1_{тек}} \& b^2_j \rightarrow \dots \rightarrow (S^k_{i+n_{тек}} \& TC) \& b^k_j \rightarrow S_{цел}$, где элементарный акт поведения $(S^k_{i+n_{тек}} \& TC) \& b^k_j \rightarrow S_{цел}$ означает, что при появлении в проблемной среде сигнала ТС на фоне условного сигнала $S^k_{i+n_{тек}}$ действие b^k_j перестает приводить в ней к достижению целевого сигнала $S_{цел}$, а элементарный акт поведения $(S^k_{i+n_{тек}} \& TC) \& b^k_j \rightarrow S_{цел}$ показывает, что при отсутствии в ПС "тормозного сигнала" ТС на фоне условного сигнала $S^k_{i+n_{тек}}$ действие b^k_j вновь приводит к появлению стандартного сигнала $S_{цел}$;

- третий этап самообучения интеллектуальных систем сводится к выработке цепи программы целесообразного поведения, приводящей к выключению "тормозного сигнала" ТС при его появлении в проблемной среде. При этом строится ветвь программы целесообразного поведения, имеющая следующее содержание

$$(S^k_{i+n_{тек}} \& TC) \& b^1_j \rightarrow (S^k_{i+n_{тек}} \& TC^1) \& b^2_j \rightarrow \dots \rightarrow (S^k_{i+n_{тек}} \& TC^k) \& b^k_j \rightarrow (S^k_{i+n_{тек}} \& TC).$$

Рассмотрим допустимые изменения проблемной среды, которые могут повлиять на достижение цели $S_{цел}$ ИС путем имитации отработки основной цепи программы целесообразного поведения [93]:

1. При изменении значений и характера отношений, определяющих взаимное влияние объектов среды друг на друга.

2. При изменении состояний объектов в сравнении с их состояниями, в которых они находились, когда в проблемной среде вырабатывалась основная цепь программы целесообразного поведения.

3. При появлении в проблемной среде новых объектов, препятствующих выполнению действий, закрепленных в основной цепи программы целесообразного поведения.

Если в проблемной среде произошли изменения количественных значений пространственных отношений, то задача включения "тормозного сигнала" сводится к определению различий между ситуациями соответственно характеризующими текущую $S_{\text{тек}}$ и исходную $S_{\text{исх}}$ ситуацию среды, в которой формировалась цепь программы целесообразного поведения. При этом, различия между сигналами $S_{\text{тек}}$ и $(S_{\text{тек}} \& \text{ТС})$ могут устраняться на основе применения одного из алгоритмов планирования поведения на нечетких семантических сетях, выбираемого исходя из заданных условий функционирования [62].

Если в среде произошли качественные изменения значений отношений между объектами, то для преобразования ситуации $S_{\text{тек}}$ в ситуацию $S_{\text{исх}}$ можно воспользоваться следующими эвристическими правилами. На первом шаге планирования поведения, позволяющего выявлять различия, имеющиеся в нечетких семантических сетях, определяющих ситуации $S_{\text{тек}}$ и $(S_{\text{тек}} \& \text{ТС})$, выделить пары различных по содержанию отношений r' и r'' , определяющих взаимодействие одинаковых объектов в ситуациях $S_{\text{тек}}$ и $S_{\text{исх}}$ объектов. Затем, на основе хранящихся знаний из памяти извлекается элемент, имеющий следующую структуру

$$\langle r \{b_j^1, b_j^2, \dots, b_j^k\} r \rangle,$$

где $\{b_j^1, b_j^2, \dots, b_j^k\}$ – множество действий, которые в условиях неопределенности с равной вероятностью позволяют выполнить преобразования отношения r' в отношение r'' . Таким образом, после выборки типового элемента представления знаний по методу проб и ошибок выявляется необходимое действие из множества $\{b_j^1, b_j^2, \dots, b_j^k\}$. После

устранения всех различий, наблюдающихся между $S_{\text{итек}}$ и $(S_{\text{итек}} \& \text{ТС})$ ИС для достижения целевой ситуации $S_{\text{цел}}$ имитирует отработку действия основной цепи программы условного поведения.

Во втором случае задача устранения различий между ситуациями $S_{\text{итек}}$ и $(S_{\text{итек}} \& \text{ТС})$ решается путем манипулирования отдельными объектами среды до их преобразования к необходимым состояниям. В условиях со значительным уровнем неопределенности, когда ИС заранее неизвестны закономерности преобразования объектов от состояния к состоянию в результате обрабатываемых ею действий, задача может быть решена на основе алгоритма самообучения с активно-пассивной логикой поведения в динамических ПС и на основе алгоритма самообучения с активной логикой поведения в статических средах.

В третьем случае, выявление тормозного сигнала ТС осуществляется после неудачной отработки одного из действий, закрепленных в основной цепи программы условного поведения. Формально этот процесс сводится к вычислению нечеткого изоморфного вложения ситуации проблемной среды $S_{\text{итек}}$ в нечеткую семантическую сеть G . При этом "тормозной сигнал" ТС определяется на основе операции нечеткого дополнения над нечеткими семантическими сетями G^t_i и G^*_i т.е. $\text{ТС} = G^t_i \setminus G^*_i$, где G^*_i - нечеткая семантическая сеть, биективно соответствующая исходной $S_{\text{исх}}$ ситуации проблемной среды. После выявления "тормозного сигнала" ИС определяет действие, позволяющее выполнить его устранение. После устранения "тормозного сигнала" ИС выполняет действия основной цепи программы целесообразного поведения.

В общем случае, процесс устранения "тормозного сигнала" строится по следующей схеме. Вначале выявляется сам "тормозной сигнал" ТС. Затем определяются эффективные действия $b_j \in B$, преобразующие этот сигнал; для этой цели ИС использует алгоритм самообучения, например, 2.2. с активно-пассивной логикой поведения. При этом, после каждого нового преобразования сигнала ТС у ИС вырабатывается элементарный акт

поведения $TC \& b_j \rightarrow TC'$ и она пробует обработать действие b_j^k , закрепленное в основной цепи целесообразного поведения в элементарном акте $(S_{i\text{тек}} \& TC) \& b_j^k \rightarrow S_{i+1\text{тек}}$. Если действие b_j^k после преобразования "тормозного сигнала" из TC в TC' приводит к включению в проблемной среде $S_{i+1\text{тек}}$, то принимается решение, что "тормозной сигнал" TC является выключенным, ИС функционирует, обрабатывая основную цепь программы целесообразного поведения. В результате же устранения "тормозного сигнала" TC у ИС формируется цепочка условного поведения следующего вида

$$TC \& b^{1j} \rightarrow TC^1 \& \rightarrow b^{2j} \rightarrow \dots \rightarrow TC^k \& b^{kj} \rightarrow TC'$$

Таким образом, типичным проявлением "тормозного сигнала" является появление в проблемной среде объектов, препятствующих обработке действий основной цепи программы целесообразного поведения. Рассмотрим структурированное описание алгоритма самообучения интеллектуальных систем при наличии в проблемной среде "тормозных сигналов" связанных с появлением в ней новых объектов или при изменении отдельными объектами своих состояний.

Для построения этого алгоритма введем следующие обозначения:

- $W(S_{\text{цел}})$ -основная цепь программы целесообразного поведения, связанная с достижением цели $S_{\text{цел}}$;
- $a_j^* \in W(S_{\text{цел}})$ -запись, означающая, что условный сигнал a_j^* закреплен в программе целесообразного поведения $W(S_{\text{цел}})$;
- $b_j \in W(c_i^*)$ -запись, означающая, что действие b_j закреплено в программе целесообразного поведения $W(c_i^*)$;
- $G_i^t \quad G^T$ - нечеткая операция объединения нечетких семантических сетей, биективно соответствующих ситуации $S_{i\text{тек}}$ и тормозному сигналу; в этом случае нечеткая семантическая сеть $G^T = G_i^t \setminus G_i^*$, где G_i^t -нечеткая семантическая сеть, определяемая исходной ситуацией $S_{i\text{исх}}$;
- V -мощность множества V .

С учетом приведенных обозначений алгоритм самообучения интеллектуальных систем при наличии в проблемной среде тормозных сигналов запишется следующим образом.

Алгоритм 3.2

Исходные условия: $S_{\text{цел}}, W(S_{\text{цел}})$

Входные переменные: $S_{i\text{тек}} \in S$,

Выходные переменные: программы целесообразного поведения $W(TC)$.

1.Начало.

2.Выполнить осмотр в проблемной среде; сформировать ситуацию $S_{i\text{тек}}$ и соответствующую ей НСС G_1 .

3.Проверить условия: в структуре программы целесообразного поведения $W(S_{\text{цел}})$ имеются сигналы $S_{i\text{тек}} \subset G_1$? Если да, перейти к п.5; если нет, перейти к п.4.

4.Сформировать основную цепь функционирования на основе алгоритма 2.2. , после восприятия сигналов $S_{\text{цел}}$ выполнить стандартные реакции; конец.

5.Выполнить действие $b_j \in W(S_{\text{цел}})$ элементарного акта поведения $S_{i\text{тек}} \& b_j \rightarrow S_{i+1\text{тек}}$, где $S_{i\text{тек}} \in G_1$.

6.Выполнить осмотр проблемной среды; сформировать текущую ситуацию среды $S_{i+1\text{тек}}$ и соответствующую ей НСС G_2 .

7.Проверить условие: $S_{i\text{тек}} = S_{i+1\text{тек}}$? Если да, перейти к п.8;если нет, перейти к п.9.

8.Проверить условие: $S_{\text{цел}} \in G_2$? Если да, выполнить стандартные реакции, конец; если нет, перейти к п.5.

9.Проверить условие: $S_{i+1\text{тек}} \in G_2$? Если да, перейти к п.10;если нет, перейти к п.4.

10.Сформировать тормозной сигнал $T = G_2 \setminus (S_{i\text{тек}} \cup S_{i+1\text{тек}})$.

11.Выбрать действие $b'_j \in V_1$ согласно равномерному закону распределения вероятностей его выполнения.

12.Проверить условие: b'_j хранится в w_1 ? Если да, исключить действие из множества V_1 , перейти к п.13;если нет, перейти к п.15.

14. Устранить тормозной сигнал невозможно; конец.

15. Выполнить имитацию отработки выбранного действия b'_j , воздействуя на объекты, образующие тормозной сигнал ТС.

16. Выполнить осмотр проблемной среды, сформировать ситуацию $S_{i+2\text{тек}}$ и соответствующую ей нечеткую семантическую сеть G_3 . Определить сигнал $ТС' = G_2 \setminus G_3$.

17. Проверить условие: $ТС' = ТС$? Если да, перейти к п.19; если нет, перейти к п.18.

18. Отнести действие b'_j к множеству действий, занесенных в память w_1 , перейти к п.11.

19. Сформировать и запомнить элементарный акт поведения $ТС \& b'_j \rightarrow ТС'$.

20. Выполнить имитацию отработки действия $b^*_j \in W(s_{\text{цел}})$ согласно элементарному акту поведения $S_{i+2\text{тек}} \& b^*_j \rightarrow S_{i+1\text{тек}}$, где $S_{i+2\text{тек}} = S_{i+1\text{тек}} \cup ТС'$; $S_{i+1\text{тек}} \in W(s_{\text{цел}})$.

21. Сформировать сигнал $S_{i+1\text{тек}}$, характеризующий проблемную среду после отработки действия b^*_j .

22. Проверить условие: " $S_{i+2\text{тек}} = S_{i+1\text{тек}}$? Если да, тормозной сигнал ТС выключен, перейти к п.5; если нет, $w_1 = 0$, сформировать множество действий V_1 , перейти к п.11.

23. Конец.

На основании приведенного алгоритма самообучения ИС способна вырабатывать программы целесообразного поведения следующего вида _

$$S_{i\text{тек}} \& b^1_j \rightarrow \dots \rightarrow (S_{i+n\text{тек}} \& ТС) \& b^k_j \rightarrow S_{\text{цел}}$$

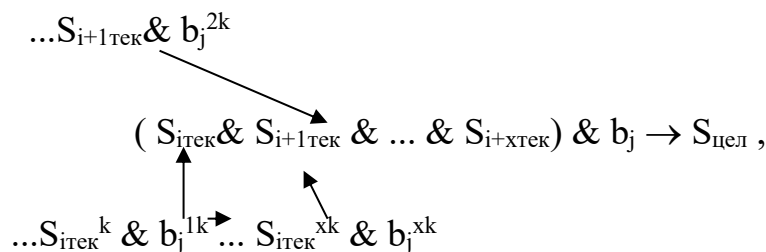
$$S_{i+1\text{тек}} \& b^1_j \rightarrow \dots \rightarrow S_{i+n\text{тек}} \& b^k_j..$$

В общем случае интеллектуальная система может столкнуться с появлением в проблемной среде нескольких взаимосвязанных тормозных сигналов, приводящих к необходимости выработки сложной разветвленной программы целесообразного поведения. Например, когда наличие в проблемной среде «тормозного сигнала» $ТС^2$ не позволяет устранить другой

тормозной сигнал $ТС^1$, влияющий на результат обработки основной цепи программы целесообразного поведения, приводящей после обработки к появлению в среде стандартного сигнала $S_{цел}$. В этом случае взаимосвязь "тормозных сигналов" $ТС^1$ и $ТС^2$ может устанавливаться последовательно во времени в следующем порядке. Вначале вырабатывается основная цепь программы целесообразного поведения. Затем выявляется и устраняется «тормозной сигнал» $ТС^1$ при отсутствии сигнала $ТС^2$. После этого выявляется и устраняется "тормозной сигнал" $ТС^2$ по мере его появления.

3.3. АЛГОРИТМЫ САМООБУЧЕНИЯ ПРИ НАЛИЧИИ В СРЕДЕ СУММАРНЫХ СИГНАЛОВ: ОПИСАНИЕ И АНАЛИЗ

Процесс самообучения ИС в проблемной среде при наличии суммарных сигналов сводится к выработке программ целесообразного поведения, имеющих следующую структуру [63,91]



где $(S_{iтек} \& S_{i+1тек} \& \dots \& S_{iтек}^{xk}) \& b_j \rightarrow S_{цел}$ - элементарный акт поведения, которое означает, что действие b_j после обработки приведет к появлению в проблемной среде целевой ситуации $S_{цел}$ при одновременном присутствии в ней условных сигналов $S_{iтек}, S_{i+1тек}, \dots, S_{iтек}^{xk}$, образующих суммарный комплекс.

Рассмотрим два возможных случая:

- когда ИС априори задано содержание суммарного сигнала, и она его уточняет и корректирует в конкретных условиях проблемной среды;
- когда ИС априори неизвестно содержание суммарного сигнала, и она его выявляет в процессе адаптации к проблемной среде.

В первом случае необходимость одновременного присутствия в проблемной среде всех компонентов $S_{iтек}, S_{i+1тек}, \dots, S_{iтек}^{xk}$ закрепляемого в

вырабатываемой программе целесообразного поведения суммарного сигнала, может проверяться путем поочередного их выключения с последующей обработкой действия b_j . Например, если после обработки элементарного акта поведения $(S_{i_{тек}} \& S_{i+1_{тек}} \& \dots \& S_{i_{тек}}^{xk}) \& b_j \rightarrow S'_{i_{тек}}$ при отсутствии одного из компонентов суммарного сигнала в проблемной среде появляется условный сигнал $S'_{i_{тек}}$, то отсутствующий компонент исключается из содержания суммарного сигнала в данных условиях проблемной среды и соответствующая ему простая цепь функционирования не формируется. В противном случае отсутствующий компонент приобретает роль обязательной составляющей суммарного сигнала в текущих условиях проблемной среды.

Во втором случае суммарный сигнал может быть выявлен в процессе самообучения ИС путем наблюдения за характером изменений, происходящих в проблемной среде. В данном случае поведение ИС носит пассивный характер и осуществляется следующим образом. Если при наличии в проблемной среде нескольких явно выраженных сигналов (например, несколько объектов в определенных состояниях) выполненное ИС действие приводит к необходимым преобразованиям, а в случае отсутствия любого из них таких изменений не происходит, то суммарный сигнал и выполненное действие запоминаются. Затем ИС проверяет необходимость одновременного присутствия всех компонентов суммарного сигнала для достижения цели путем поочередного их выключения с последующей обработкой необходимых действий, связанных с достижением закрепленных в программе целесообразного поведения сигналов.

Очевидно, что такой подход рационально использовать только в условиях функционирования, когда не накладываются ограничения на длину промежутка времени, по истечению которого ИС должна достигнуть $S_{цел}$, определяющего целевое условие поведения. Это обусловлено тем, что рассмотренный принцип формирования условных программ поведения требует обработки дополнительных действий, приводящих к поочередному выключению каждого из компонентов суммарного сигнала с последующим

их восстановлением. Таким образом, в процессе выработки программ целесообразного поведения при ограниченных ресурсах времени, отведенного на достижение цели, следует сразу же запоминать все составляющие суммарного сигнала и проверять их необходимость для достижения цели по мере последующих отработок сформированной программы целесообразного поведения. При этом если в проблемной среде отсутствует хотя бы один из компонентов суммарного сигнала и ИС после отработки соответствующего действия не воспринимает необходимого условного или стандартного сигнала, то формируется простая цепь условной программы, связанная с появлением в проблемной среде недостающего компонента суммарного сигнала [69].

Рассмотрим структурированное описание алгоритма самообучения 3.3. при наличии в статической проблемной среде суммарных сигналов.

Пусть ИС априори задано элементарный акт поведения $(S_{i_{тек}} \& S_{i+1_{тек}}) \& b_j \rightarrow S_{цел}$, содержащее суммарный сигнал $S_{i_{тек}} \& S_{i+1_{тек}}$, состоящий из двух компонентов. Тогда, используя обозначения, принятые при построении алгоритмов самообучения 2.1-3.2, алгоритм самообучения 3.3 запишется следующим образом.

Алгоритм 3.3.

Исходные условия: $(S_{i_{тек}} \& S_{i+1_{тек}}) \& b_j \rightarrow S_{цел}$

Входные переменные: $S_{i_{тек}} \in S$.

Выходные переменные: элементарные акты поведения программы целесообразного поведения.

1. Начало.
2. Выполнить осмотр проблемной среды, сформировать текущую ситуацию $S_{i_{тек}}$ и биективно соответствующую ей нечеткую семантическую сеть G_1 .
3. Проверить условие: $(S_{i_{тек}} \subseteq G_1) \& (S_{i+1_{тек}} \subseteq G_1)$? Если да, перейти к п.4; если нет, перейти к п.5.

4. Выполнить имитацию обработки действия b_i элементарного акта поведения $(S_{i\text{тек}} \& S_{i+1\text{тек}}) \& b_j \rightarrow S_{\text{цел}}$; после восприятия сигнала $S_{\text{цел}}$ выполнить стандартные реакции, конец.

5. Проверить условие: $(S_{i\text{тек}} \not\subseteq G_3) \& (S_{i\text{тек}} \subseteq G_3)$? Если да, перейти к п.6; если нет, перейти к п.11.

6. Выполнить имитацию обработки действия b_i элементарного акта поведения $(S_{i\text{тек}} \& S_{i+1\text{тек}}) \& b_j \rightarrow S_{\text{цел}}$;

7. Выполнить осмотр проблемной среды; сформировать текущую ситуацию $S_{i+2\text{тек}}$ и соответствующую ей нечеткую семантическую сеть G_2 .

8. Проверить условие: $S_{\text{цел}} \subseteq G_2$? Если да, перейти к п.9; если нет, перейти к п.15.

9. Исключить отсутствующий компонент из состава суммарного сигнала $(S_{i\text{тек}} \& S_{i+1\text{тек}})$ в данной проблемной среде.

10. Обработать стандартные реакции; конец.

11. Проверить условие: $(S_{i\text{тек}} \subseteq G_1) \& (S_{i+1\text{тек}} \not\subseteq G_1)$? Если да, перейти к п.12; если нет, перейти к п.17.

12. Выполнить имитацию обработки действия b_j элементарного акта поведения $(S_{i\text{тек}} \& S_{i+1\text{тек}}) \& b_j \rightarrow S_{\text{цел}}$.

13. Выполнить осмотр проблемной среды, сформировать текущую ситуацию $S_{i+3\text{тек}}$ и биективно соответствующую ей нечеткую семантическую сеть G_3 .

14. Проверить условие: $S_{\text{цел}} \subseteq G_3$? Если да, перейти к п.9; если нет, перейти к п.16.

15. Используя алгоритм 2.1 сформировать цепь программы целесообразного поведения, связанную с достижением условного сигнала $S_{i\text{тек}}$; после восприятия этого сигнала, перейти к п.4.

16. Используя алгоритм 2.1 сформировать цепь программы целесообразного поведения, связанную с достижением условного сигнала $S_{i+1\text{тек}}$; после восприятия этого сигнала, перейти к п.4.

17. Используя алгоритм 2.1 сформировать цепи поведения, приводящие к достижению условных сигналов $S_{i\text{тек}}$ и $S_{i+1\text{тек}}$; после восприятия этих сигналов, перейти к п.4.

18. Конец.

Следует указать, что по аналогичной схеме строится и алгоритм самообучения интеллектуальной системы при наличии в проблемной среде суммарных сигналов, содержащих большее число компонентов.

Рассмотрим алгоритм самообучения ИС при наличии в динамической проблемной среде априорно не заданных суммарных сигналов (обозначим этот алгоритм как алгоритм 3.4.)

Для описания алгоритма 3.4., используем ранее принятые и следующие вновь введенные операторы:

$W_{\text{кон}}$ и $W_{\text{исх.}}$ - соответственно ячейки памяти, отведенные под хранение заданной цели и сигнала, характеризующего текущее состояние проблемной среды.

Алгоритм 3.4.

Входные переменные : $S_{i\text{тек}} \in S, S_{\text{цел.}}$

Выходные переменные: программы целесообразного поведения, содержащие суммарные сигналы.

1. Начало.
2. Выполнить осмотр проблемной среды, сформировать ситуацию среды $S_{\text{исх}}$ и биективно соответствующую ей нечеткую семантическую сеть $G_{\text{исх.}}$
3. Сформировать текущую ситуацию проблемной среды $S_{j\text{тек}}$ и биективно соответствующую ей нечеткую семантическую сеть G_1 .
4. Занести в память заданный стандартный сигнал, определяемый целевой ситуацией $S_{\text{цел.}}$
5. Сформировать условный сигнал проблемной среды, выполнив операцию нечеткого пересечения между семантическими сетями G_1 и $G_{\text{цел.}}$,

т.е. $S_{j\text{тек}} = G_i \cap G_{\text{цел}}$, где $G_{\text{цел}}$ -нечеткая семантическая сеть, биективно соответствующая ситуации $S_{\text{цел}}$.

6. Проверить условие: сигнал $S_{j\text{тек}}$ является суммарным, т.е. $S_{j\text{тек}} = (S_{j+1\text{тек}}, S_{j+2\text{тек}}, \dots, S_{j\text{птек}})$? Если да, перейти к п.7;если нет, перейти к п.8.

7. Занести в память $w_{\text{исх}}$ суммарный сигнал $S_{j\text{тек}} = (S_{j+1\text{тек}}, S_{j+2\text{тек}}, \dots, S_{j\text{птек}})$; перейти к п.9.

8. Занести в память $w_{\text{исх}}$ отдельный сигнал $S_{j\text{тек}}$.

9. Выбрать действие b_j из множества действий B согласно равномерному закону распределения вероятностей их выполнения.

10. Выполнить имитацию отработки действие b_j .

11. Выполнить осмотр проблемной среды, сформировать текущую ситуацию среды $S_{j+1\text{тек}}$ и соответствующую ей нечеткую семантическую сеть G_2 .

12. Определить условный сигнал проблемной среды $S_j^* = G_2 / G_1$.

13. Проверить условие: $S_j^* = 0$, т.е. изменений в среде не произошло? Если да, занести действие b_i в память w_2 , перейти к п.9; если нет, перейти к п.14.

14. Выполнить осмотр проблемной среды, сформировать текущую ситуацию среды $S_{j+2\text{тек}}$ и соответствующую ей нечеткую семантическую сеть G_3 . Проверить условие: $G_3 \subset G_2$? Если да, перейти к п.17;если нет, перейти к п.15.

15. Обнулить ячейки памяти W_2 . Выполнить осмотр проблемной среды, сформировать текущую ситуацию среды $S_{j+2\text{тек}}$ и соответствующую ей нечеткую семантическую сеть G_4 . Проверить условие: воспринят суммарный сигнал S_j ? Если да, перейти к п.7;если нет, перейти к п.8.

16. Проверить условие: воспринятые сигналы закреплены в последнем элементарном акте поведения формируемой программы целесообразного поведения, т.е. этот сигнал определяет локальную подцель текущего шага самообучения? Если да, перейти к п.20; если нет, перейти к п.14.

17. Проверить условие: предыдущее состояние проблемной среды характеризовалось суммарным сигналом S_j ? Если да, перейти к п.18; если нет, перейти к п.19.

18. Занести в память элементарный акт поведения $(S_{j_{\text{тек}}}\& S_{j+1_{\text{тек}}}\&\dots\& S_{j_n})$ & $b_j \rightarrow S_{\text{цел}}$; перейти к п.20.

19. Занести в память элементарное звено поведения $S_{j+2_{\text{тек}}}$ & $b_j \rightarrow S_{\text{цел}}$, где $S_{j+2_{\text{тек}}}$ - условный сигнал, характеризующий состояние проблемной среды, при котором действие b_j привело к появлению в ней стандартного $S_{\text{цел}}$, или сигнала $S_{j_{\text{тек}}}$, закрепленного в последнем элементарном акте поведения формируемой программы целесообразного поведения.

20. Проверить условие: сигнал $S_{j_{\text{тек}}}$, характеризующий предыдущее состояние проблемной среды является суммарным? Если да, перейти к п.21; если нет, перейти к п.22.

21. Занести в память сформированную часть $(S_{j_{\text{тек}}}\& S_{j+1_{\text{тек}}}\&\dots\& S_{j_n})\&b_j \rightarrow S_{\text{цел}}$ элементарного звена поведения $(S_{j_{\text{тек}}}\& S_{j+1_{\text{тек}}}\&\dots\& S_{j_n})\& b_j \rightarrow S_{\text{цел}}^*$, т.к. условный сигнал $S_{\text{цел}}^*$ был занесен в память на предыдущем шаге самообучения; перейти к п.23.

22. Записать в память сформированную часть $S_{j_{\text{тек}}}\&b_j$ элементарного звена поведения $S_{j_{\text{тек}}}$ & $b_j \rightarrow S_{\text{цел}}^*$.

23. Выполнить осмотр проблемной среды, сформировать текущую ситуацию среды $S_{j+3_{\text{тек}}}$ и соответствующую ей нечеткую семантическую сеть G_4 . Сформировать условный сигнал, характеризующий текущее состояние проблемной среды.

24. Проверить условие: воспринятый сигнал закреплен в последнем выработанном элементарном звене поведения формируемой программы целесообразного поведения? Если да, перейти к п.25; если нет, перейти к п.42.

25. Проверить условие: сигнал, закрепленный в последнем элементарном звене поведения, является суммарным? Если да, перейти к п.26; если нет, перейти к п.39.

26. Проверить условие: восприняты все составляющие закрепленного суммарного сигнала $S_{j\text{тек}} = (S_{j+1\text{тек}}, S_{j+2\text{тек}}, \dots, S_{j\text{птек}})$? Если да, перейти к п.27; если нет, перейти к п.36.

27. Выполнить действие b_j последнего элементарного звена поведения, закрепленного в формируемой программе целесообразного поведения.

28. Выполнить осмотр проблемной среды, сформировать текущую ситуацию среды $S_{j+4\text{тек}}$ и соответствующую ей нечеткую семантическую сеть G_5 . Определить условный сигнал a_{j+1} , характеризующий текущее состояние среды.

29. Проверить условие: сигнал a_{j+1} закреплен в предыдущем за последним элементарным звеном поведения формируемой программы целесообразного поведения? Если да, перейти к п.30; если нет, перейти к п.34.

30. Выполнить имитацию отработки действия, закрепленного формируемой программе целесообразного поведения, начиная от условного сигнала a_{j+1} .

31. Проверить условие: последний сигнал, закрепленный в формируемой программе целесообразного поведения, является суммарным? Если да, перейти к п.32; если нет, перейти к п.33.

32. Занести в память $W_{\text{цел}}$ суммарный сигнал $S_{j\text{тек}} = (S_{j+1\text{тек}}, S_{j+2\text{тек}}, \dots, S_{j\text{птек}})$, закрепленный в последнем элементарном звене поведения формируемой программы целесообразного поведения; перейти к п.5.

33. Занести в память $W_{\text{цел}}$ последний сигнал, закрепленный в формируемой программе целесообразного поведения; перейти к п.5.

34. Обнулить ячейки памяти, в которых хранится последнее выработанное элементарный акт поведения формируемой программы целесообразного поведения.

35. Принять за целевой - сигнал, закрепленный в предпоследнем выработанном элементарном акте поведения формируемой программы целесообразного поведения; перейти к п.5.

36. Выполнить действие b_j последнего элементарного звена поведения формируемой программы целесообразного поведения.

37. Выполнить обзор проблемной среды, сформировать текущую ситуацию среды $S_{j+4_{\text{тек}}}$ и соответствующую ей нечеткую семантическую сеть G_5 . Определить условный сигнал a_{j+2} , характеризующий текущее состояние проблемной среды.

38. Проверить условие: условный сигнал $S_{j+4_{\text{тек}}}$ закреплен в предпоследнем элементарном звене поведения формируемой программы целесообразного поведения? Если да, перейти к п.39; если нет, перейти к п.37.

39. Выполнить действие b_j , закрепленное в последнем элементарном акте поведения формируемой условной программы поведения.

40. Выполнить осмотр проблемной среды, сформировать текущую ситуацию среды $S_{j+5_{\text{тек}}}$ и соответствующую ей нечеткую семантическую сеть G_6 . Определить условный сигнал a_{j+3} , характеризующий текущую ситуацию проблемной среды.

41. Проверить условие: сигнал a_{j+3} закреплен в предпоследнем элементарном звене поведения формируемой программы целесообразного поведения? Если да, перейти к п.30; если нет, перейти к п.34.

42. Проверить условие: воспринятые сигналы закреплены в вырабатываемой программе целесообразного поведения? Если да, перейти к п.45; если нет, перейти к п.41.

43. Самообучение на основе алгоритма с активной логикой проводить нецелесообразно.

44. Самообучение выполнить на основе алгоритма с активной логикой.

45. Конец.

3.4. РАЗРАБОТКА ПРОЦЕДУР ВЫБОРА АЛГОРИТМОВ ЭФФЕКТИВНОГО ПОВЕДЕНИЯ ИНТЕЛЛЕКТУАЛЬНОЙ СИСТЕМЫ

Для эффективного взаимодействия ИС с ПС ей необходимо выбрать наиболее подходящий для сложившейся ситуации алгоритм поведения, связанного с преобразованиями ситуации среды для достижения цели [71].

Для этого ИС может использовать следующие правила выбора различных алгоритмов функционирования:

1. ИС обрабатывает стандартные реакции при восприятии в ПС соответствующей ситуации $S_{\text{цел}}$, определяющей целевое условие функционирования.

2. ИС выполняет ПЦП, если среда характеризуется $S_{\text{итек}} \in S$, закрепленным в ПЦП, связанной с достижением цели.

3. Если имитация отработки программы поведения не приводит к цели $S_{\text{цел}}$, то отработка прекращается, и принимает решение о возможном наличии в ПС тормозного сигнала.

4. Система самообучается согласно соответствующему алгоритму самообучения при наличии в нем тормозного сигнала.

5. В динамической ПС самообучение ИС базируется на основе алгоритмов с активно-пассивной логикой поведения. ИС принимает решение, что ПС является динамической путем наблюдения за характером изменений, происходящих в проблемной среде.

6. В квазистационарных ПС самообучение ИС базируется на алгоритмах с активной логикой. Система принимает решение, что конкретная ПС является квазистационарной, так как изменения происходят в ней в результате действий ИС.

7. При наличии суммарных сигналов самообучение ИС осуществляется на основе соответствующих алгоритмов. ИС принимает решение о наличии в среде априорно неизвестных суммарных сигналов, если между исходной и целевой ситуациями среды имеется ряд различий по состояниям находящихся в них объектов.

Таким образом, процесс самообучения ИС разбивается на несколько автономных этапов, каждый из которых определяется достижением целевого сигнала $S_{\text{цел}}$. Если в процессе преобразования ситуации ПС от начального состояния $S_{\text{исх}}$ к начальному состоянию $S_{\text{итек}}$ ИС воспринимает подцель не

текущего, а следующего этапа $S_{i+1\text{тек}}$, то в качестве подцели следующего этапа поведения принимается $S_{i+2\text{тек}}$ и так далее до достижения $S_{\text{цел}}$.

Структура алгоритма выбора целесообразного поведения ИС определяется на основе имеющихся в наличии алгоритмов самообучения. Перечень используемых для этих целей средств: алгоритмы самообучения 2.1, 2.2., 3.1-3.4, предназначенные для построения модели функционирования, отражающих закономерности преобразований ситуаций среды, происходящих в результате имитируемых ИС действий.

Алгоритм целесообразного поведения интеллектуальной системы можно представить в следующем виде.

Алгоритм 3.5.

Исходные условия: алгоритмы 2.1, 2.2, 3.1 - 3.4, $S_{\text{цел}}$.

Входные переменные: $S_{\text{итек}}$ S

Выходные переменные: выбранные алгоритмы функционирования.

1. Начало
2. Выполнить осмотр ПС, сформировав текущую ситуацию среды $S_{\text{итек}}$ и соответствующую ей НСС G_1 .
3. Проверить условие: имеется ли в среде $S_{\text{итек}}$, которая удовлетворяет условию $S_{\text{итек}} \in G_1$?
4. $S_{\text{итек}}$ представляет собой опасность? Если да, перейти к п. 5, если нет, то перейти к п.6.
5. Запомнить действия, соответствующие $S_{\text{итек}}$ и перейти к п.2.
6. Проверить условие: $S_{\text{цел}} \in G_1$? Если да, перейти к п.7, если нет, то перейти к п. 15
7. Выполнить стандартные реакции, соответствующие $S_{\text{цел}}$. Если необходимо повторить $S_{\text{цел}}$, то перейти к п.8, в противном случае, к п.21.
8. Выполнить наблюдения за самопроизвольными изменениями ПС. Если наблюдения есть, перейти к п.7, если нет, перейти к п. 9.
9. Проверить условие: подцель определяет суммарный сигнал? Если да, перейти к п. 10, если нет, то перейти к п. 11.

10. Осуществить самообучение на основе алгоритма 3.3. После восприятия сигнала $S_{iтек} S$ в процессе самообучения перейти к п.2.

11. Осуществить самообучение на основе алгоритма 2.2. с активно-пассивной логикой поведения. После восприятия сигнала $S_{iтек} S$ в процессе самообучения перейти к п.2.

12. Проверить условие: подцель определяет суммарный сигнал? Если да, перейти к п. 13, если нет, то перейти к п. 14.

13. Осуществить самообучение на основе алгоритма 3.4. После восприятия сигнала $S_{iтек} S$ в процессе самообучения перейти к п.2.

14. Осуществить самообучение на основе алгоритма 2.1. с активной логикой поведения. После восприятия сигнала $S_{iтек} S$ в процессе самообучения перейти к п.2.

15. Выполнить действия b_j^1 элементарных актов поведения $S_{iтек} \& b_j^1 \rightarrow S_{i+1тек}$.

16. Выполнить осмотр ПС, сформировав текущую ситуацию среды $S_{i+1тек}$ и соответствующую ей НСС G_2 . Проверить условие: в среде воспринят сигнал $S_{цел}$? Если да, то перейти к п.7, если нет, то перейти к п.17.

17. Если неоднократно приходило к достижению $S_{цел}$, то принимается решение о наличии тормозного сигнала. Проверить условие: есть ПЦП, позволяющая устранить тормозной сигнал? Если да, то перейти к п.18, если нет, то перейти к п.19.

18. Выявить тормозной сигнал $ТС$ и устранить его при помощи отработки импликативных правил $T \& b_j^1 \rightarrow T_i^1$, перейти к п. 15.

19. Проверить условие: тормозной сигнал устранен? Если да, то перейти к п. 20, если нет, то перейти к п. 21.

20. Осуществить самообучение в ПС на основе алгоритма 3.2. После устранения тормозного сигнала перейти к п.15.

21. Конец.

Приведенный алгоритм 3.5 позволяет ИС реализовать динамику поведения ИС в проблемной среде при попадании в недоопределенные условия [94].

ВЫВОДЫ К ГЛАВЕ 3.

Алгоритмы самообучения 3.1 - 3.4 позволяют ИС адаптироваться к сложным условиям ПС при наличии в ней связанных различным образом между собой событий и условных сигналов.

Приведённые алгоритмы целесообразного поведения носят достаточно универсальный характер и могут быть использованы для организации процесса функционирования адаптивных интеллектуальных систем, имеющих многоцелевое назначение или различную специализацию. Например, для организации процесса самообучения интеллектуальной системы в труднодоступных и агрессивных для человека средах.

Однако следует отметить, что ряд приведённых алгоритмов (например, алгоритм 3.4) имеют сложную и громоздкую структуру, а при неблагоприятных условиях ПС это может привести к отработке ИС в процессе адаптации достаточно большого количества пробных действий. Прежде всего, это обусловлено отсутствием в алгоритмах эффективных механизмов предварительного отбора необходимых для достижения цели действий, приводящих к сокращению их перебора по методу проб и ошибок. Решение вопросов, связанных с организацией таких механизмов, возможно в том случае, если между текущей и целевой ситуациями ПС наблюдаются только различия по значениям одноимённых отношений, которые можно выявить на основе их анализа при соответствующем описании действий $v_{j1} \in V$.

Таким образом, можно заключить, что рассмотренный подход организации целесообразного планирования поведения рационально использовать для организации процесса адаптации ИС в средах, подробную информационную модель которых априорно описать практически невозможно из-за отсутствия необходимой информации и сложно выполнить подробный анализ воспринимаемых в ПС ситуаций.

СПИСОК ПРИНЯТЫХ СОКРАЩЕНИЙ

АС – алгоритмы самообучения

ИС – интеллектуальная система

ИР – интеллектуальный робот

ЛП – лингвистическая переменная

ПС – проблемная среда

НСС – нечеткая семантическая сеть

СР – стандартная реакция

ПЦП – программа целесообразного поведения

ФД – фрейм действий

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Поддьяков Н.Н. Мышление дошкольника.- М.: Педагогика, 1977.-272 с.
2. Усманов А.А., Напалков А.В. Алгоритмы интеллектуальной деятельности.- Ташкент: ФАН,1979.-190 с.
3. Розенблат Ф. Принципы нейродинамики: Перцептроны и теория механизмов мозга.- М.: Мир,1965.-243 с.
4. Амосов Н.М., Касаткин А.М., Касаткина Т.М., Талаев С.А. Автоматы и разумное поведение: Опыт моделирования. - Киев: ИК, АНУССР,1973.-170 с.
5. Каляев А.В., Чернухин Ю.В., Носков В.Н. Каляев И.А.Однородные управляющие структуры адаптивных роботов.- М.: Наука,1990.-152 с.
6. Факс Р.,Харт П., Нильсон П. Новое направление в автоматическом решении задач роботом // В кн.: Интегральные роботы. Вып.2 / Под ред. Поздняка Г.Е.-М.: Мир,1975. -С.406-439 .
7. Ефимов Е.И. Решатели интеллектуальных задач.- М.: Наука, 1982.-316 с.
8. Попов Э.В., Фридман Г.Р. Алгоритмические основы интеллектуальных роботов и искусственного интеллекта.- М.: Наука, 1976.-400 с.
9. Тимофеев А.В. Роботы и искусственный интеллект.- М.: Наука,1978.-170 с.
10. Фин В.К. Индуктивные модели // Представление знаний в человеко-машинных и робототехнических системах.- М.:ВИНИТИ, 1984.- С.58-76.
- 11.Приобретение знаний / Под ред. Осиги С.-М.:Мир,1990.-304 с.
- 12.Мики Д. Формирование и выполнение планов вычислительной машиной // В кн.: Интегральные роботы. Вып.2 /Под ред. Поздняка Г.Е.- М.: Мир, 1975.- С.379-405
- 13.Робототехника и гибкие автоматизированные производства // Кн.: 6: Техническая имитация интеллекта / Под ред. Макарова И.Н.-М.:Высшая школа,1986.-144 с.
- 14.Нариньяни А.С. Система продукции как модульный программный комплекс // Прикладные и экспериментальные лингвистические процессоры. - Новосибирск: СО АН СССР. Препринт, 1982.-С.125-152.

15. Яхно Т.М. Система продукций как стиль программирования задач искусственного интеллекта.- Новосибирск: ВЦ АН СССР,1984. Препринт. Вып.9.-33 с.
16. Искусственный интеллект: Применение в интегрированных производственных системах/ Под ред. Кьюсиака Э.- М.: Машиностроение, 1991.-544 с.
17. Сулла, Р. В. Применение нейронных сетей в сфере искусственного интеллекта / Р. В. Сулла // Студенческая наука для развития информационного общества : сборник материалов VI Всероссийской научно-технической конференции, Ставрополь, 22–26 мая 2017 года. Том Часть 1. – Ставрополь: Северо-Кавказский федеральный университет, 2017. – С. 199-200.
18. Программирование на языке РЕФАЛ. Препринты. - М.: ИПМ. АН СССР. 1972. N41,N43;44,N48,49.
19. Осуга С. Обработка знаний. -М.:Мир, 1989. -293 с.
20. Лорьер Ж.Л. Системы искусственного интеллекта. -М.: Мир,1991. -517 с.
21. Кузин Л.Т. Основы кибернетики. Том 2: Основы кибернетических моделей. -М.:Энергия,1979.-584 с.
22. Левин Д.Е. Инструментальный комплекс программирования на основе языков высокого уровня.-М.:Наука,1987.-200 с.
23. Лозовский В.С. Семантические сети // Представление знаний в человеко-машинных и робототехнических системах.- М.: ВИНТИ,1984.Т.А.-С.84-120.
24. Вагин В.Н. Дедукция и обобщение в системах принятия решений. -М.: Наука,1988.-384 с.
25. Представление и использование знаний/ Под ред. Уэно Х., Исидзука М. -М.:Мир,1990. -280 с.
26. Кузин Е.С., Фоминых И.Б., Хахалин Г.К. Системы принятия решений в интегральных роботах (Обзор). В кн.: Интегральные роботы / Под ред. Поздняка Г.Е.-М.:Мир,1975.- С.364-377.

- 27.31. Минский М. Структура для представления знаний // В кн.: Психология машинного зрения. Под ред. Уинстона П. -М.: Мир, 1978.-С. 249-340.
28. Пospelов Д.А. Логико-лингвистические модели в системах управления.- М.: Энергоиздат, 1981.-232 с.
29. Пospelов Д.А. Большие системы. Ситуационное управление.- М.: Знание, 1985.-62 с.
30. Клыкoв Ю.И. Ситуационное управление большими системами. -М.: Энергия, 1974.-134 с.
31. Аверкин А.Н., Батыршин А.З., Блишун А.Ф. и др. Нечеткие множества в моделях управления и искусственного управления / Под ред. Пospelова Д.А.-М.: Наука, 1976.-312 с.
32. Борисов А.Н., Алексеев А.В., Меркурьева Г.В. и др. Обработка нечеткой информации в системах принятия решений. - М.: Радио и связь, 1989.-304 с.
33. Ежкова И.Г., Пospelов Д.А. Принятие решений при нечетких основаниях:
1. Универсальная школа // Известия АН СССР. Техническая кибернетика. 1977. N6.- С.3-11.
34. Ежкова И.Г., Пospelов Д.А. Принятие решений при нечетких основаниях:
2. Схемы вывода // Известия АН СССР. Техническая кибернетика. 1978. N2.- С.5-11.
35. Мелихов А.М., Берштейн А.С., Коровин С.Я. Ситуационные советующие системы с нечеткой логикой.-М.: Наука, 1990.-272 с.
36. Заде Л. Понятие лингвистической переменной и его применение к принятию приближенных решений .-М.: Мир, 1976.-168 с.
37. Рафаэль Б. Думающий компьютер.-М.: Мир, 1979,-407 с.
38. Аверкин А.Н., Гвида Дж., Пospelов Д.А. Обобщенная стратегия в решениях проблем //Известия АН СССР.Техническая кибернетика.1978.N5.- С.71-83.
39. Ерохин Е.А., Судейкин М.Н. Планирование операций интегрального робота //Известия АН СССР: Техническая кибернетика.1981.N2.-С.80-92.

40. Попов Э.В. Общение с ЭВМ на естественном языке.-М.: Наука,1982.-320 с.
41. Берштейн Л.С., Мелехин В.Б. Структура декларативного представления знаний интегрального робота // Управляющие системы и машины.1990.№6.- С.75-82.
42. Клацки Р. Память человека: Структуры и процессы. -М.:Мир,1980.-238 с.
43. Кохонен Г. Ассоциативная память.- М.:Мир,1980.-238 с.
44. Зеляковский И.Ю., Мелехин В.Б. Вопросы организации семантической памяти интегрального робота. Деп. ВИНТИ.1983.-14 с. N 118-82 Деп.
45. Мелехин В.Б. Классификация семантической информации интегрального робота и способы ее представления в долговременной памяти. Деп. ВИНТИ.1983.-12 с. N7025-83 Деп.
- 46.Берштейн Л.С., Мелехин В.Б. Структура процедурного представления знаний интегрального робота: 1.Расплывчатые семантические сети. Известия АН СССР.Техническая кибернетика.1988.№6.-С.119-124.
- 47.Берштейн Л.С., Мелехин В.Б. Планирование поведения интегрального робота при расплывчатом представлении знаний//Микропроцессорные системы контроля и управления.- Рига:Рижский политехнический институт.1987.-С.117-126.
- 48.Берштейн Л.С. Мелехин В.Б. Планирование поведения интеллектуального робота.-М.:Энергоатомиздат,1994.-240 с.
- 49.Мелехин В.Б. Механизация самообучения интегрального робота. Деп. в ВИНТИ, 1981.-18 с. N117-82 Деп.
50. Ворнин Л.Г. Физиология высшей нервной деятельности. М. :Высшая школа,1979.-340 с.
- 51.Шингаров Г.Х. Условный рефлекс и проблема знака и значения. М.:Наука,1979. 200 с.

52. Берштейн Л.С., Мелехин В.Б. Организация знаковосигнального принципа управления целесообразным поведением интегрального робота // Автоматика и телемеханика 1991. N12. - С.118-127.
53. Литвак Б.Г. Экспертная информация: Методы получения и анализа.- М.: Радио и связь, 1982.-184 с.
54. Мелехин В.Б. Об алгоритме самообучения интегрального робота с активно-пассивной логикой поведения // Кибернетика, 1984. N4.-С.107-211.
55. Берштейн Л.С., Мелехин В.Б. Моделирование оптимальных условных рефлексов поведения робота . В кн.: Методы построения алгоритмических моделей сложных систем. Таганрог: ТРТИ, 1986, Вып.6. -С.69-74.
56. Берштейн Л.С., Мелехин В.Б. Об алгоритме автоматического формирования условных программ функционирования интегрального робота в условиях неопределенности // Кибернетика 1990. N5.- С.79-84.
57. Анохин П.К. Биология и физиология условного рефлекса.-М.: Медицина, 1968.-469 с.
58. Ахо А., Хокпрофт Дж., Ульман Дж. Построение и анализ вычислительных алгоритмов.-М. Мир, 1979.-535 с.
59. Оре О. Теория графов.-М.: Наука, 1980.-336 с.
60. Бирюков Д.А. К вопросу о природе ориентировочной реакции // В кн.: Избранные труды / Под ред. Н.Н.Василевского.-Л.: Медицина, 1973.- С.99-105.
61. Берштейн Л.С., Мелехин В.Б. Процедуры автоматического формирования программ функционирования интегрального робота при появлении в среде тормозных сигналов // Кибернетика и системный анализ. 1992. N1.- С.148-154.
62. Берштейн Л.С., Мелехин В.Б. Процедуры планирования поведения робота при расплывчатом представлении знаний // Микропроцессорные системы контроля и управления.-Рига: Рижский политехнический институт, 1987.- С.117-125.
63. Мелехин В.Б. Алгоритмы самообучения интегрального робота в сложных средах // Кибернетика. 1986. N1.- С.102-108.

64. Берштейн Л.С., Мелехин В.Б., Канаев М.М. Процедуры формирования комбинированных условных программ для интеллектуальных систем управления // Приборостроение. 1995. том 38. N 3,4. - С.7-10.
65. Мелехин В.Б. Процедуры выбора алгоритма целесообразного поведения интегрального робота // Кибернетика 1988. N2. - С.94-101.
66. Растрингин Л.А., Риппа К.К. Автоматная теория случайного поиска.- Рига: Зинатне, 1973.-344 с.
67. Берштейн Л.С., Мелехин В.Б. Декомпозиция нечетких семантических сетей для планирования операций интегрального робота // Известия. АН СССР Техническая кибернетика. 1988. N5. - С.115-123.
68. Берштейн Л.С., Мелехин В.Б. Структура процедурного представления знаний интегрального робота. 2. Фрейм-микропрограммы поведения // Известия АН СССР. Техническая кибернетика. N5. - С.190-194.
69. Берштейн Л.С., Мелехин В.Б. Об алгоритме формирования рациональных условных программ поведения интегрального робота // Известия ВУЗов Северо-Кавказского региона. Технические науки. N 2. 1996. -С.47-61.
70. Хант Э. Искусственный интеллект.- М.: Мир, 1978.-558 с.
71. Берштейн Л.С., Мелехин В.Б. Процедуры принятия решений интегрального робота в условиях неопределенности // Автоматика и телемеханика. 1989. N8. -С. 106-115.
72. Берштейн Л.С., Мелехин В.Б. Процедуры обобщения опыта функционирования интегрального робота // Автоматика и телемеханика. 1989. N 12. - С.102-112.
73. Литвинцева Л.В., Поспелов Д.А. Пополнение знаний. В кн.: Искусственный интеллект.- В 3-х кн. Кн.2. Модели и методы: Справочник / Под ред. Поспелова Д.А.-М.: Радио и связь, 1990. -С.76-81.
74. Поспелов Д.А. О человеческих рассуждениях в интеллектуальных системах. В кн.: Логика рассуждений и ее моделирование // Вопросы кибернетики. -М.: АН СССР. 1982.-С.5-37.

75. Тейз А., Грибон П., Луи Ж. и др. Логический подход к искусственному интеллекту: От классической логики к логическому программированию.- М.: Мир, 1990.-434 с.
76. Берштейн Л.С., Мелехин В.Б., Михайлов Л.Л. Индуктивные и традиционные схемы вывода логики казуально-зависимых рассуждений для принятия решений в немонотонных средах// Информатика и вычислительная техника: Теория и приложения. Махачкала: Даг. госуниверситет. 1994. Вып. 1.- С.99-108.
77. Ефимов А.Н. Логика диалоговых рассуждений// Известия АН СССР. 1988. N 5.- С.129-145.
78. Берштейн Л.С., Ильягуев П.М., Мелехин В.Б. Интеллектуальные системы.- Махачкала: Дагкнигоиздат, 1996.-67 с.
79. Лукасевич Я. Аристотельская силлогистика с позиций современной формальной логики.- М.: Иностранлит., 1959.-237 с.
80. Мелехин В.Б. Процедуры вычисления расстояний между разнесенными препятствиями на местности и определения координат точки встречи робота с подвижной целью// Многопроцессорные вычислительные структуры.- Таганрог : ТРТИ. 1977. Вып.9.- С.74-76.
81. Мелехин В.Б. Алгоритмы автоматического формирования модели окружающей среды транспортного робота // Кибернетика. 1985. N 4.- С.98-103.
82. Берштейн Л.С., Мелехин В.Б., Канаев М.М. Планирование траектории интегральным роботом при преследовании подвижной цели// Актуальные проблемы информатики, управления и радиоэлектроники.- Махачкала: ДГТУ. 1995.- С.69-77.
83. Берштейн Л.С., Мелехин В.Б. Нечеткие сетевые модели для представления сейсмических данных. Тез. докл. Международного симпозиума "Каспий-Балтика 95". -С.-Петербург, 1995.-С.41.
84. Мелехин А.В., Кагиров К.А., Красовская Л.В. Организация самообучения интеллектуальных систем на основе имитации

целесообразного поведения // Технические науки. Вестник ДГТУ. - Махачкала: ДГТУ, 2002. - №5 (0.7/0.4 п.л.)

85. Мелехин В.Б., Красовская Л.В. Информационная модель представления динамических знаний в интеллектуальных системах прогнозирования аварийных и чрезвычайных ситуаций // Современные информационные технологии в управлении. Сб. тез. докл. Всероссийской научно-технической конференции. – Махачкала, 2003 (0.15/0.1 п.л.).

86. Красовская Л.В., Магомедов Х.Д., Мелехин А.В. Динамические модели представления знаний в интеллектуальных системах прогнозирования аварийных и чрезвычайных ситуаций // Информационные и телекоммуникационные системы: сетевые технологии. Материалы III Республиканской научно-практической конференции. – Махачкала: ДНЦ РАН, 2004 (0.6/0.2 п.л.).

87. Мелехин В. Б., Красовская Л. В. Организация самообучения интеллектуального робота на нечетких семантических сетях // Неделя науки-2004. Сборник докладов XXV итоговой научно-технической конференции преподавателей, сотрудников, аспирантов и студентов ДГТУ.– Махачкала: ДГТУ, 2004 (0.15/0.1).

88. Красовская Л. В. Особенности организации самообучения ИС в режиме полифазного поведения // Неделя науки- 2005. Сб. тез. докл. XXVI итоговой научно-технической конференции преподавателей, сотрудников, аспирантов и студентов ДГТУ. – Махачкала: ДГТУ, 2005.

89. Красовская Л. В. Алгоритмы самообучения интеллектуальных систем на нечетких семантических сетях // Технические науки. Вестник ДГТУ.- Махачкала: ДГТУ, 2005. - №7.

90. Красовская Л. В. Несмещенные оценки сравнения отношений в нечетких семантических сетях // Технические науки. Вестник ДГТУ.- Махачкала, 2005.- №7.

91. Красовская Л.В. Алгоритмы самообучения интеллектуальных систем на нечетких семантических сетях в проблемной среде при наличии в ней

суммарных сигналов // Современные информационные технологии в проектировании, управлении и экономике. Часть 1. Сборник научных трудов. – Махачкала: ДГТУ, 2005.

92. Красовская, Л. В. Моделирование процессов самообучения интеллектуальных систем в проблемной среде при наличии в ней причинно-следственных связей / Л. В. Красовская // Актуальные проблемы прикладной математики : Материалы IV Международной научной конференции, Нальчик - Эльбрус, 22–26 мая 2018 года. – Нальчик - Эльбрус: Институт прикладной математики и автоматизации – филиал ФГБНУ «Федеральный научный центр «Кабардино-Балкарский научный центр Российской академии наук», 2018. – С. 136. – EDN CYDEEDG..

93. Красовская Л.В. Алгоритмы самообучения интеллектуальных систем на нечетких семантических сетях с имитацией отработки действий // Радиоэлектроника.- Известия вузов России. 2006. – № 4.

94. Красовская Л. В. Моделирование процессов самообучения интеллектуальных систем на нечетких семантических сетях в недоопределенных средах: автореферат диссертации на соискание ученой степени кандидата технических наук / Красовская Людмила Владимировна. – Махачкала, 2006. – 17 с.