

Метод формирования обобщенных понятий с использованием темпоральных деревьев решений¹

Аннотация. В работе рассматриваются подходы к построению темпоральных деревьев решений для задачи диагностики технического объекта, предлагаются алгоритмы построения темпоральных деревьев решений. Моделируется процесс диагностики на основе известных данных о возможных неисправностях. Для диагностики в динамике используется многоагентный подход.

Ключевые слова: обобщение, диагностика, темпоральное дерево решений.

Введение

Обнаружение знаний в базах данных является актуальной проблемой современных интеллектуальных систем. Это связано с большим объемом современных баз данных, которые так велики, что практически невозможно вручную проанализировать их для извлечения ценной информации, помогающей принимать важные решения. Отсюда следует, что люди нуждаются в помощи интеллектуальных систем для повышения своих аналитических возможностей. Системы автоматического обнаружения знаний могут анализировать "сырые" данные и предоставлять извлеченную информацию скорее и успешнее, чем аналитик мог бы найти ее самостоятельно [1].

Во многих случаях для описания поведения сложных систем приходится использовать сотни независимых атрибутов, которые необходимо анализировать, чтобы наиболее точно смоделировать поведение системы. В такой ситуации крайне важно решать задачу обобщения для получения компактных описаний классов ситуаций на объекте управления. Одним из эффективных способов получения таких описаний являются деревья решений [2]. Однако следует учесть, что важным параметром, отражающим динамику поведения сложной системы, является время.

Обобщение информации, отражающей изменение ситуаций во времени, требует использования специальных методов, таких как, например, темпоральные деревья решений.

1. Проблема диагностики в искусственном интеллекте

Диагностирование как раздел искусственного интеллекта занимается разработкой методов и алгоритмов, способных определить корректность работы изучаемого объекта (системы). Если система работает некорректно, нужно как можно более точно определить, в какой части системы произошел отказ, и какая ошибка произошла. Определение ошибки происходит на основе наблюдений, которые дают информацию о поведении системы. Термин «диагностирование» также относится к определению неисправности.

В данной работе предлагается применить методы диагностики на основе использования модели объекта [3, 4]. Для имитации поведения объекта и выявления неисправностей может быть использована модель специального вида, которая описывает структуру и поведение сложного технического объекта. Такая модель представляет собой тройку $\langle S, M, R \rangle$, где:

¹ Работа выполнена при финансовой поддержке РФФИ (проект № 09-01-00076)

S – множество переменных, описывающих состояние системы (в технической диагностике – это чаще всего измерения, получаемые от датчиков, установленных в системе, или результаты вычислений, выполненных над полученными измерениями);

M – набор режимов работы, включающих в себя состояния «норма» (корректное поведение) и «неисправность» (некорректное поведение);

R – набор отношений, связывающих множество переменных S , описывающих состояние системы, и набор режимов работы M .

В рассматриваемой задаче информация о функционировании объекта управления представлена множеством результатов измерений, поступающих с датчиков. Таким образом, состояние сложного объекта и его компонентов описывается набором признаков (количественных и качественных), при этом одним из признаков является время. Целью работы является получение обобщенных описаний классов ситуаций, возникающих на объекте, причем с каждым классом ситуаций связываются определенные действия по управлению объектом, называемые далее восстановительными действиями.

Рассмотрим постановку задачи обобщения [1]: дано O – множество объектов, представленных в некоторой интеллектуальной системе. Пусть $V \subset O$ множество объектов системы, относящихся к определенному классу (множество положительных объектов).

$O = V \cup W$; $V \cap W = \emptyset$, W образует множество отрицательных объектов относительно данного класса.

Пусть имеется обучающая выборка $K = K^+ \cup K^-$, такая, что $K^+ \subset V$, $K^- \subset W$,

$$K^+ \cap K^- = \emptyset.$$

Необходимо на основе анализа обучающей выборки построить понятие, разделяющее положительные и отрицательные примеры. Понятие Q считается сформированным, если удалось построить решающее правило, которое для любого примера из обучающей выборки указывает, принадлежит ли этот пример понятию, или нет. Сформированное понятие может быть представлено в виде логической функции, дерева решений, набора продукционных правил вида «ЕСЛИ условие ТО искомое понятие».

Для выбора способа описания классов ситуаций на сложном техническом объекте можно применить различные методы, такие как

продукционные модели [5], нечеткие множества [6], приближенные множества [7], логические функции [8], деревья решений [2]. В данной работе предлагается использовать метод деревьев решений, который широко используется в задачах классификации объектов, представимых признаковыми описаниями. Таким образом, формирование обобщенного понятия Q предлагается выполнить путем построения дерева решений.

Решение задачи обобщения на основе анализа набора признаков предполагает первоначально построение дерева решений, а затем – использование его для классификации ситуаций на объекте. Рассмотрим возможные ситуации на объекте. Введем понятия: C_n – множество ситуаций, возникающих на сложном техническом объекте, которые диагностируются как нормальные; C_f – множество ситуаций, в которых наблюдаются неисправности на объекте. Для использования методов искусственного интеллекта в диагностике предлагается сформировать описание понятий C_n и C_f в рамках введенной модели. На основе полученных обобщенных описаний классов C_n и C_f необходимо выработать рекомендации по выбору восстанавливающего действия на сложном техническом объекте; такое действие должно переводить систему из состояния «неисправность» в состояние «норма». Введем основные понятия и определения, относящиеся к деревьям решений.

2. Понятие темпорального дерева решений

В настоящее время известен целый ряд алгоритмов для построения деревьев решений. Такие алгоритмы – CLS [9], ДРЕВ [1], алгоритм, основанный на метрике Хэмминга [1], ID3 [2] – и различные модификации этих алгоритмов – C4.5 [10], ID5R [11] и др. – обрели широкое распространение и зарекомендовали себя в широком спектре приложений.

Однако деревья решений в их классическом виде имеют некоторые ограничения — в частности, невозможность работы с поведением объекта или системы во времени, невозможность принятия решений с течением времени. Без учета фактора времени не удастся проследить динамику изменения состояния системы. Предлагается расширить признаковое описание объектов: введем понятие «время» как один из

атрибутов, используемых при построении дерева решений. Будем использовать дискретное время $t = 0, 1, 2, \dots$.

Вернемся к описанию модели системы. Пусть для слежения за состоянием сложного технического объекта или системы используется некоторое количество датчиков. Обозначим эти датчики $s_1, s_2, \dots, s_n$. Пусть показания с этих датчиков снимаются через некоторый временной интервал t_0 - в каждый момент времени.

Каждую ситуацию из C_n и C_f предлагается рассматривать на некотором временном интервале t^* , включающем определенное количество дискретных значений t . При этом ситуацией будем называть последовательность из t^* значений датчиков с соответствующими им временными метками.

Неформально темпоральное дерево решений T_{temp} - это дерево, в котором вершины-листья помечены видом неисправности и предлагаемым в данной ситуации восстановительным действием, а промежуточные вершины, называемые далее внутренними вершинами, помечены именами атрибутов, в совокупности с видом проверки и временной меткой. Дуги темпорального дерева определяют переходы по результатам проверок значений атрибутов во внутренних вершинах.

Дадим формальное определение.

Темпоральное дерево решений - это взвешенный ациклический ориентированный граф $T_{temp} = (V_{temp}, E_{temp})$. Во множестве вершин V_{temp} выделим вершину $v_0 \in V_{temp}$ - корень дерева. Все вершины разделим на два класса: $V_i \subseteq V_{temp}$ - множество внутренних вершин (узлов) дерева; V_l включает в себя такие вершины, из которых выходят дуги; $V_i \subseteq V_{temp}$ - множество внешних, конечных вершин дерева (листьев); V_l включает в себя такие вершины, из которых дуги не выходят; V_i и V_l образуют разбиение множества вершин V дерева решений $T: V_i \cap V_l = \emptyset, V_i \cup V_l = V_{temp}$.

Внутренние вершины V_i дерева взвешены (помечены) парой $\langle a, t_c \rangle$, где a - имя атрибута, t_c - временная метка.

Вершины-листья V_l взвешены (помечены) названием или номером ситуации из $C_n \cup C_f$. Если ситуация отнесена к классу C_n вершина - лист помечается названием или номером предлагаемого восстановительного действия.

Каждая дуга e дерева решений взвешена условием «атрибут[t_c] = значение_атрибута» (для качественных значений атрибутов) либо «атрибут[t_c] σ значение_атрибута» (для количественных значений атрибутов, $\sigma = \{ \geq, >, =, \leq, < \}$), где «атрибут» - имя атрибута в вершине, из которой исходит дуга e , «значение_атрибута» - одно из возможных значений (количественное или качественное) признака «атрибут»; t_c - момент времени, в который необходимо проводить эту проверку, $0 \leq t_c < t^*$.

Таким образом, основным отличием темпоральных деревьев решений от обычных деревьев решений является наличие метки времени в каждом внутреннем узле дерева. Проверка значения атрибута во внутреннем узле дерева производится только в том случае, если момент времени, которым помечен набор значений датчиков, совпадает с временной меткой в этом узле.

Из приведенных определений видно, что темпоральные деревья решений - это расширение классического понятия деревьев решений [2], позволяющее учитывать фактор времени. Каждый узел такого дерева имеет метку времени, благодаря чему можно учитывать порядок производимых наблюдений и задержку между ними, что в результате приводит к увеличению различающей способности дерева и самым непосредственным образом влияет на процесс принятия решений.

Темпоральные деревья предлагается использовать для решения задач диагностики сложного технического объекта; примером такой задачи может служить задача бортовой диагностики автомобиля. Данные, поступающие с различных датчиков, описывают состояние всей системы в конкретный момент времени, а использование темпоральных деревьев решений позволяет проследить изменение состояния системы за некоторый интервал времени и выявить неисправности или неблагоприятные тенденции, которые могут по истечении некоторого времени повлечь неисправности.

3. Исходные данные для построения темпоральных деревьев решений

Исходными данными как для первоначального обобщения, так и для диагностики, является таблица наблюдений. Также может использоваться некоторая модель для восстановительных

действий: например, может быть введен частичный порядок над действиями - тогда при наличии неразличимых ситуаций и сравнимых действий достаточно будет выполнить наиболее сильное из действий. Рассмотрим пример множества наблюдений, представленный в Табл. 1, где каждой строке соответствует некоторая ситуация s_i , которая определяется: показаниями датчиков $Датчик_j$ в моменты времени $t=0, 1, \dots, 7$, (здесь z, n, l, v, h соответствуют качественным показаниям датчиков: n - норма, l и h соответственно низкое и высокое, v - очень низкое, z - нуль); восстановительным действием $Д_i$ для каждой ситуации; крайним сроком K_i выполнения соответствующего действия.

Такое представление позволяет не вводить каких-либо ограничений на способ представления времени в системе.

На основании этих данных необходимо построить дерево, осуществляющее диагностику сложного технического объекта. Такое дерево решений будем называть темпоральным деревом решений $\tilde{T}$. Внутренним узлам дерева сопоставляются проверки значений датчиков в определенные моменты времени, дуги помечены показаниями датчиков, листья дерева соответствуют выбранным управляющим действиям. Следует отметить, что каждая ситуация, представленная в таблице, определена на временном интервале $[0, 7]$.

Восстановительные действия, которые переводят сложный технический объект или систему из состояния «неисправность» в состояние «норма» часто могут негативно сказываться на функциональности системы. Например, отключение какой-то части системы из-за перегрева может привести к тому, что эта система станет полностью неработоспособной (процессор в компьютере). Менее критичный случай - это простое ограничение функциональности (отключение системы климат-контроля в автомобиле).

Кроме того, некоторые восстановительные действия могут быть связаны между собой: операции, выполняемые восстановительным действием d_1 в ситуации f_1 , могут включать в себя некоторые (или даже все) операции, которые выполняются восстановительным действием d_2 в ситуации f_2 .

В связи с этим каждому восстановительному действию можно поставить в соответствие некоторую величину, называемую «стоимость восстановительного действия». Эта величина может играть существенную роль при построении темпорального дерева решений, особенно если темпоральное дерево решений будет использоваться для процедуры бортовой диагностики автомобиля, когда время принятия решения и время/сложность выполнения восстановительного действия критичны.

Однако возникает вопрос - как посчитать стоимость восстановительных действий, если ситуации неразличимы на рассматриваемом временном интервале t^* и необходимо выполнить несколько действий?

Предлагается считать стоимость восстановительных действий следующим образом. В случае если все действия независимы, суммарная стоимость нескольких действий определяется просто как сумма их стоимостей. Но если действия зависимы (некоторые из действий включают в себя другие, например действие «остановить автомобиль» включает в себя действие «снизить скорость»), то предлагается ввести частичный порядок на действиях и рассматривать стоимость нескольких действий как стоимость комплексного действия [4]. Пусть $d1, d2, d3$ - действия, стоимости которых заданы, причем $d1 < d2$ (отношение $<$ означает, что действие $d2$ включает в себя все действия, входящие в $d1$), а действия $d1$ и $d3$ не сравнимы. Введем функцию Act , которая определяет стоимость комплексного действия следующим образом:

Табл. 1. Пример исходных данных для построения темпорального дерева

	Датчик 1 (s1)							Датчик 2 (s2)							Датчик 3 (s3)							Д	К			
t	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7		
Сит1	n	n	n	n	h	h			l	v	v	v	v	v			n	n	n	l	l	l			a	5
Сит2	h	h	h						h	n	n						n	n	n						b	2
Сит3	n	n	n	n	h	h	h	h	l	l	l	l	v	v	v	v	n	n	n	l	l	l	v	v	b	7
Сит4	n	n	n	h	h	h	h		l	l	l	l	l	v			n	n	h	h	h	h	h		c	6
Сит5	h	h	h	h					h	n	n	n					n	n	n	l					c	3
Сит6	n	n	n	h	h	h			l	v	v	z	z	z			n	n	n	l	l	v			d	5
Сит7	h	h	h	h	h	h			l	l	n	n	l	v			n	n	n	l	l	v			b	5
Сит8	h	h	h	h	h	h			h	h	n	n	l	l			n	n	n	l	v	z			c	5

$Act(\{d1, d2\}) = Act(d2)$ – если действия сравнимы, то стоимость определяется действием с большей стоимостью;

$Act(\{d1, d3\}) = Act(d1) + Act(d3)$ – если действия несравнимы, то стоимость определяется как сумма стоимостей отдельных действий.

Для рассматриваемого в Табл.1 примера восстановительные действия – это a, b, c, d . Модель восстановительных действий можно задать в виде схемы упорядочения, представленной на Рис. 1. Пусть стоимости действий следующие: $Act(a) = 100$, $Act(b) = 20$, $Act(c) = 50$, $Act(d) = 10$. В данном примере $\{b, c\}$ представляет собой комплексное действие.

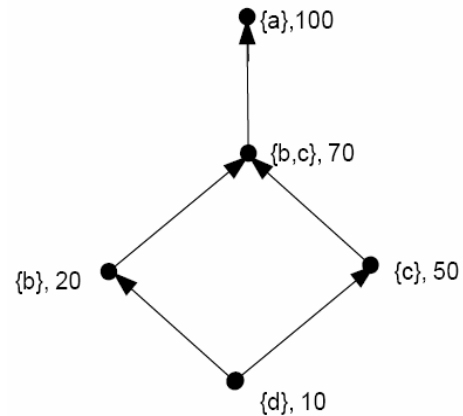


Рис.1. Модель восстановительных действий для примера из Табл.1

4. Алгоритмы построения темпоральных деревьев решений

Появление временной переменной и необходимость учитывать крайние сроки выполнения восстановительных действий приводят к тому, что алгоритм построения дерева решений – теперь уже темпорального – будет несколько модифицирован. Общая схема алгоритма построения темпорального дерева решений приведена на Рис. 2. На вход алгоритма подаются:

- таблица с ситуациями;

- наблюдения в виде множества пар <датчик, временная метка>;

- модель восстановительных действий.

На выходе получаем темпоральное дерево решений. Функция Лист(S, M) на основе ситуаций и модели восстановительных действий строит лист дерева, помечая его множеством восстановительных действий, полученным согласно модели.

В связи с тем, что, во-первых, время — процесс необратимый, а во-вторых, не все значения

Алгоритм Построение_темпорального_дерева

(S : Таблица с ситуациями,

O : Наблюдения,

M : Модель восстановительных действий)

Результат: Темпоральное дерево решений $\tilde{T}$

начало

Если для всех ситуаций из S восстановительные действия совпадают, то вернуть Лист(S, M)

Пусть D — минимальный крайний срок для ситуаций из S .

Если ситуации из S неразличимы на основе показаний датчиков с меткой времени $t \leq D$, то вернуть Лист(S, M).

Выбрать наблюдение $\langle s^*, t' \rangle$, которое будет проверяться в данном узле дерева.

Пусть $s^*_1, s^*_2, \dots, s^*_n$ — различающиеся показания датчика s^* в момент времени t' , а $S^*_j, j=1, 2, \dots, n$ — подмножества ситуаций из S , состоящие из ситуаций с показанием s^*_j датчика s^* в момент времени t' .

Вернуть дерево с корнем, помеченным выбранным наблюдением $\langle s^*, t' \rangle$, и дугами, помеченными $s^*_1, s^*_2, \dots, s^*_n$, соединяющими корень соответственно с деревьями

Построение_темпорального_дерева($S^*_1, O \setminus \langle s^*, t' \rangle, M$)

Построение_темпорального_дерева($S^*_2, O \setminus \langle s^*, t' \rangle, M$)

...

Построение_темпорального_дерева($S^*_n, O \setminus \langle s^*, t' \rangle, M$)

конец

Рис. 2. Общая схема алгоритмов построения темпорального дерева решений

датчиков доступны сразу, при построении темпоральных деревьев решений будут рассмотрены два случая:

- при движении от корня дерева к листьям временные метки в узлах дерева не убывают, благодаря этому нет необходимости хранить предшествующие значения датчиков;

- нет ограничений на временные метки в узлах дерева. Необходимо хранить предшествующие значения датчиков, так как они могут потребоваться при дальнейших проверках в узлах дерева.

В работе был исследован алгоритм для построения темпоральных деревьев решений, изложенный в [4] (назовем его CPD как сокращение от фамилий авторов - Console-Picardi-Dupre).

Этот алгоритм предлагается использовать для бортовой диагностики, так как исходным ограничением на темпоральное дерево решений было неумение временных меток при движении от корня дерева к листьям. В связи с этим отпадала необходимость хранить предшествующие значения датчиков для их возможного дальнейшего использования (вычислительные ресурсы, используемые при бортовой диагностике, сильно ограничены). Такое требование, наложенное на темпоральное дерево решений, нашло свое отражение и в алгоритме его построения. Более того, определение некорректной работы и неисправностей при бортовой диагностике часто включает в себя выбор некоторых управляющих действий, которые должны по возможности вернуть объект или систему в корректное состояние. В общем случае, как было показано выше, восстановительное действие имеет некоторую стоимость, которая выражается в том, что функциональность объекта или системы уменьшается, например, уменьшение скорости или полная остановка автомобиля, отключение каких-либо модулей

системы и т. п. Поэтому хотелось бы минимизировать негативный эффект от восстановительных действий.

В связи с этим предлагается рассмотреть функцию ожидаемой стоимости темпорального дерева решений [4] - это ожидаемая стоимость восстановительного действия, выбранного при помощи темпорального дерева решений, относительно распределения вероятностей неисправностей. Ожидаемая стоимость χ темпорального дерева решений определяется формулами:

$$\begin{aligned} \chi(node) &= \text{стоимость действий в } node, \\ &\text{если } node - \text{лист дерева;} \\ \chi(node) &= \sum_{c \in V_{next}} P(\text{перейти из } node \text{ в } c) \cdot \chi(c), \\ &\text{если } node - \text{внутренний узел.} \end{aligned} \quad (1)$$

Здесь V_{next} - множество узлов - потомков узла $node$, P - вероятность перехода по каждой дуге из $node$ в $c \in V_{next}$ (определяется частотой появления различных значений атрибута для ситуаций, представленных в обучающем множестве).

Основные этапы алгоритма CPD представлены на Рис.3. Построение каждого узла темпорального дерева решений состоит из двух этапов — на первом шаге принимается во внимание стоимость, на втором выбирается наиболее информативное наблюдение. Первоначально выбираются наблюдения — будем называть их, как и в [4] «безопасными наблюдениями» - которые смогут обеспечить минимальную ожидаемую стоимость дерева.

Реализация функции *Выбрать_безопасные_наблюдения* приведена на Рис. 4.

В данном алгоритме используется функция «Произвести_разбиение», которая выполняет разбиение множества ситуаций относительно

Алгоритм Выбор_наблюдения_для_разбиения_CPD

(S: Таблица с ситуациями,

O: Наблюдения,

M: Модель восстановительных действий)

Результат: o^* - наиболее информативное наблюдение
начало

O1 = Выбрать_безопасные_наблюдения(S,O,M)

Вернуть Выбор_наблюдения_для_разбиения(S, O1)

Конец

Рис.3. Алгоритм «выбор наблюдения для разбиения — CPD»

Алгоритм Выбрать_безопасные_наблюдения

(S: Таблица с ситуациями,

O: Наблюдения,

M: Модель восстановительных действий)

Результат: O1- безопасные наблюдения

начало

Пусть cost – сумма стоимостей всех действий из S

Обозначим t_{up} - минимальный крайний срок для ситуаций из S;

t_{low} – минимальная временная метка для наблюдений O

$t_{max} = t_{up}$

Обозначим также разбиение part как пару $\langle S, t_{up} \rangle$

Для каждой временной метки tx от t_{up} до t_{low}

part = Произвести_разбиение(part, O, tx)

Посчитать стоимость newcost для получившегося разбиения part

Если newcost < cost, то {cost=newcost; $t_{max}=tx$ }

O1 = наблюдения из O с временными метками, не превосходящими t_{max}

Вернуть O1

конец

Рис.4. Алгоритм выбора безопасных наблюдений

Алгоритм Выбор_наблюдения_для_разбиения

(S: Таблица с ситуациями,

O: Наблюдения)

Результат: o* - наиболее информативное наблюдение

начало

Пусть наблюдение current — любое наблюдение из O

Пусть наименьшая энтропия $\min_entropy = -\log_2|O|$

Для всех $o \in O$

Сделать разбиение множества S на основе возможных значений o

Пусть entropy – энтропия разбиения относительно значений o

Если entropy < $\min_entropy$, то current=o, $\min_entropy=entropy$

Вернуть current

конец

Рис.5. Алгоритм «выбор наблюдения для разбиения»

выбранных наблюдений и указанной метки времени. При получении новых подмножеств (блоков) ситуаций крайний срок для восстановительных действий может увеличиться — тогда необходимо произвести повторное разбиение этого блока.

Затем выполняется шаг 2 алгоритма CPD. Процедура, реализующая этот этап, представлена на Рис. 5. С помощью данной процедуры выбирается наиболее информативное наблюдение. В качестве меры информативности используется теоретико-множественное понятие энтропии [2, 10].

Темпоральное дерево решений, построенное с использованием алгоритма CPD на основе ситуаций Табл.1 и модели восстановительных действий Рис. 1 выглядит как на Рис. 6.

Поясним работу алгоритма CPD по построению темпорального дерева решений на примере

данных из Табл.1. В корне дерева необходимо разместить проверку: пару вида $\langle si, t' \rangle$, где si – датчик, t' - момент времени. Определяем t_{up} – максимальное время наблюдения, до наступления которого можно не принимать никаких решений. Из Табл. 1 находим $t_{up}=2$. Анализируем поведение объекта в наблюдаемых ситуациях в моменты времени $t=0, 1, 2$. Для каждого t выполняем разбиение ситуаций на классы по принципу совпадения значений атрибутов (значениями атрибутов являются показания датчиков h, l, n, v, z). По формулам (1) для полученных разбиений считаются оценки стоимости выбора управляющего действия; выбирается разбиение с наименьшей оценкой стоимости. В примере было выбрано разбиение $\{\{Cит1\}, \{Cит6\}, \{Cит2, Cит5\}, \{Cит7\}, \{Cит8\}, \{Cит3\}, \{Cит4\}\}$, полученное в момент времени $t=1$. На основе данного разбиения

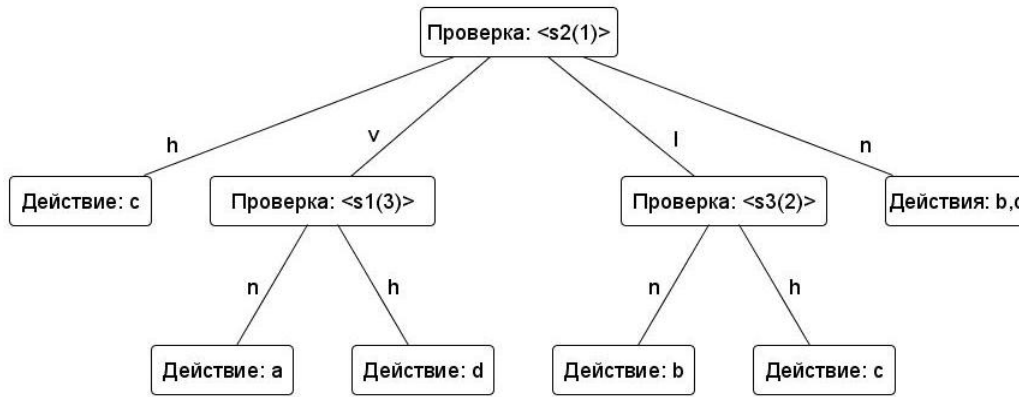


Рис 6. Темпоральное дерево решений (алгоритм CPD)

выполняется вычисление информативности наблюдений [2], и выбирается наблюдение с наилучшей оценкой. В нашем случае это s2 – наблюдение, основанное на показаниях Датчика 2 в момент времени t=1. Дальнейшие шаги алгоритма позволяют достроить и уточнить дерево решений. Так, вершина, к которой ведет дуга с меткой h, является листом: ей сопоставлена единственная ситуация Сит8 с единственным возможным управляющим действием c. Вершина, к которой ведет дуга с меткой v, не является конечной; две ситуации, связанные с данной вершиной (Сит1, Сит6), требуют различных управляющих действий (a, d), поэтому в данном узле вводится дополнительная проверка. Верши-

на, к которой ведет дуга с меткой n, связана с ситуациями Сит2 и Сит5. В данном случае эта вершина конечная, в ней возможны два управляющих действия: b, c; проблема в том, что показания датчиков в ситуациях 2 и 5 полностью совпадают в моменты времени t = 0, 1, 2, а при t = 2 решение по выбору действия обязательно должно быть принято.

В работе предлагается оригинальный алгоритм (назовем его Temporal ID3), который является расширением алгоритма ID3 [2], учитывающим фактор времени. Данный алгоритм представлен на Рис.7. По сравнению с алгоритмом CPD, на темпоральное дерево решений не накладывается никаких ограничений по временным меткам в узлах.

Алгоритм Temporal_ID3

(S: таблица с ситуациями,

O: наблюдения,

M: модель восстановительных действий)

Результат: Темпоральное дерево решений $\tilde{T}$

начало

Если для всех ситуаций из S восстановительные действия совпадают,
то вернуть Лист(S, M)

Если множество наблюдений O пусто, то вернуть Лист(S, M)

Пусть D – минимальный крайний срок для ситуаций из S

Если ситуации из S неразличимы на основе показаний датчиков
с меткой времени t ≤ D, то вернуть Лист(S, M)

<s*, t*> = Выбор_наблюдения_для_разбиения_temporal_ID3(S, O)

Пусть s*₁, s*₂, ..., s*_n – различающиеся показания датчика s* в момент
времени t', а S*_j, j=1,2,...,n – подмножества ситуаций из S, состоящие
из ситуаций с показанием s*_j датчика s* в момент времени t'.

Вернуть дерево с корнем, помеченным выбранным наблюдением
<s*, t*>, и дугами, помеченными s*₁, s*₂, ..., s*_n, соединяющими

корень соответственно с деревьями

Temporal_ID3(S*₁, O \ {<s*, t*>}, M)

Temporal_ID3(S*₂, O \ {<s*, t*>}, M)

...

Temporal_ID3(S*_n, O \ {<s*, t*>}, M)

Конец

Рис.7. Алгоритм Temporal ID3

Однако не учитывается стоимость восстановительных действий при выборе наблюдения на каждом шаге. Кроме того, снимается ограничение на неуменшение временных меток при движении от корня дерева к листьям, что приводит к необходимости сохранять некоторые значения датчиков для дальнейшего использования при проведении диагностики.

При выборе наблюдения для разбиения используется критерий «прирост информативности» Куинлана [2]. Процедура выбора представлена на Рис.8. Так как ограничение на неуменшение временных меток при движении от корня дерева к листьям снято, при построении темпорального дерева решений показания датчиков будем рассматривать как обычные атрибуты [12] — например, показание датчика s_1 в момент времени $t=0$ (Табл.1) будет атрибутом $\langle s_1, 0 \rangle$, в момент времени $t=1$ — атрибутом $\langle s_1, 1 \rangle$ и т. д. Алгоритм строит такое дерево, в котором с каждым узлом ассоциирован атрибут, являющийся наиболее информативным среди всех атрибутов, еще не рассмотренных на пути от корня дерева.

При выборе наблюдения для разбиения следует рассматривать только те наблюдения, для которых временные метки не превосходят крайнего срока для всей таблицы ситуаций S — например, для Табл. 1 это будут наблюдения с временными метками $t \leq 2$.

Пример дерева решений, построенного с использованием алгоритма Temporal ID3 на основе данных из Табл. 1, приведен на Рис. 9.

Рассмотрим подробнее процесс формирования этого дерева решений. Исходные данные, представленные в Табл. 1, будем рассматривать

как массив, в котором имеются 24 информативных атрибута; решающим является атрибут D — выбор управляющего действия. Атрибут K является вспомогательным, на основании значений этого параметра — критического времени принятия решений — будем осуществлять выборку данных из таблицы.

На первом шаге построения темпорального дерева решений надо выбрать ситуации, требующие наиболее быстрой реакции. В примере такие ситуации определяются значением $K=2$. В соответствии с этим выберем из исходной таблицы атрибуты с временными метками $t=0, 1, 2$. Полученная таблица будет содержать 9 атрибутов ($s1(0), s1(1), s1(2), s2(0), s2(1), s2(2), s3(0), s3(1), s3(2)$); поиск среди них наиболее информативного для размещения его в корне дерева решений проводится на основании вычисления оценок прироста информативности [2]. На первом шаге был выбран атрибут $s2(1)$ как наиболее информативный, он размещается в корне дерева. Четыре дуги, взвешенные значениями h, n, v, l , ведут к вершинам следующего уровня. Из этих вершин одна является конечной (взвешена действием c), в остальных случаях требуются дополнительные проверки условий. Для формирования ветвей в вершине, связанной с корнем дугой v , необходимо провести новую выборку из таблицы данных. В случае $s2(1)=v$ для ситуаций $Cит1$ и $Cит6$ требуются разные управляющие действия: a и d ; при этом в обеих ситуациях $K=5$. В соответствии с этим включаем в выборку атрибуты со значениями t от 0 до 5 (за исключением атрибута $s2(1)$), и проводим поиск наиболее информативного атрибута в таблице, содержащей 2 строки ($Cит1, Cит6$) и 17 атрибутов.

Алгоритм Выбор_наблюдения_для_разбиения_temporal_ID3

(S : Таблица с ситуациями,

O : Наблюдения)

Результат: o^* - наиболее информативное наблюдение

начало

Для всех наблюдений $\langle s, t \rangle$ из O вычисляем количество информации, которое получаем благодаря этому наблюдению:

$Gain(\langle s, t \rangle, S) = Info(S) - Info(\langle s, t \rangle, S)$, где

$Info(S)$ — энтропия для ситуаций из S (распределение восстановительных действий)

$Info(\langle s, t \rangle, S)$ — взвешенное среднее информации, необходимой для идентификации класса ситуации в каждом подмножестве, полученном при разбиении множества ситуаций из S на основе значений $\langle s, t \rangle$

Вернуть $\langle s^*, t^* \rangle$ - наблюдение с наибольшим значением $Gain(\langle s, t \rangle, S)$

конец

Рис.8. Процедура выбора наблюдения для разбиения temporal_ID3

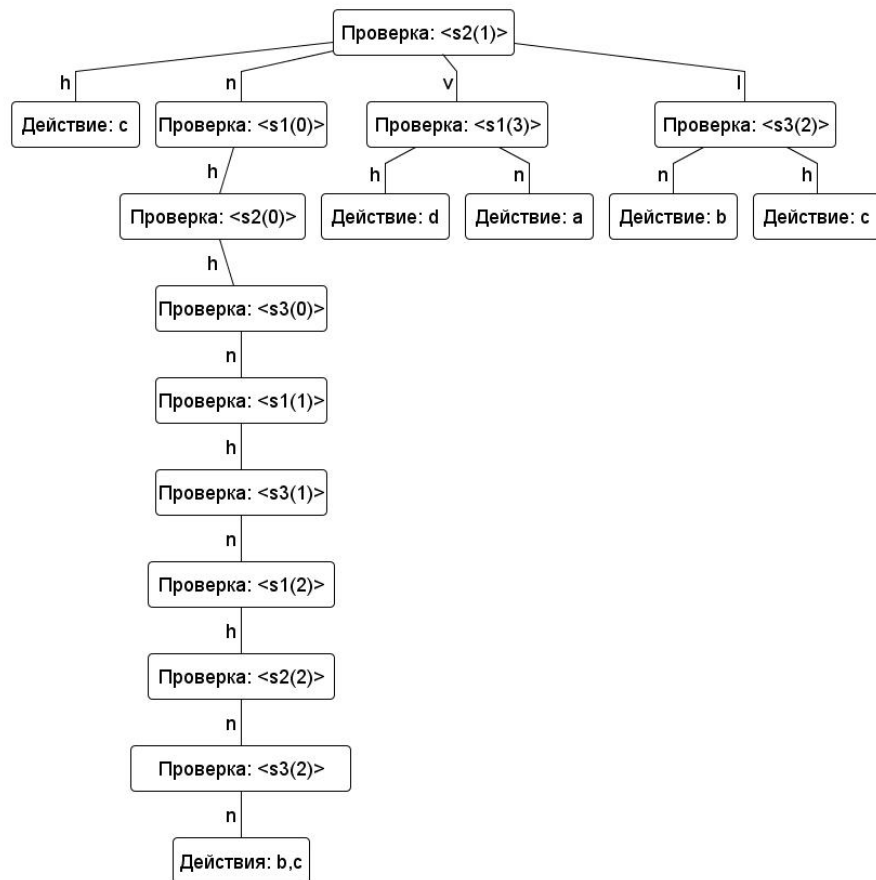


Рис. 9. Темпоральное дерево решений (алгоритм Temporal ID3)

Как показано на Рис. 9, для размещения в узле второго уровня был выбран наиболее информативный атрибут $s1(3)$; ветвление по остальным вершинам выполняется аналогично. Вершина становится листом, если с ней связано единственное восстанавливающее действие, либо рассмотрены все информативные атрибуты.

5. Моделирование процесса диагностики на основе темпоральных деревьев решений

Темпоральные деревья решений, полученные одним из рассмотренных способов, использовались для решения задачи бортовой диагностики автомобиля, такой объект представляет достаточно сложную систему, снабженную множеством датчиков. Использование темпоральных деревьев решений позволяет на основании показаний датчиков проследить изменение состояния системы за некоторый интервал времени и выявить неисправности или

неблагоприятные тенденции, которые могут по истечении некоторого времени повлечь неисправности. В случае обнаружения неисправности либо отказа с помощью темпорального дерева решений предлагается выбрать возможное восстановительное действие, способное предотвратить аварию. Как было показано выше, действия (восстановительные действия) характеризуются некоторой стоимостью, которая отражает уменьшение функциональности системы. Таким образом, основная цель процедуры диагностики заключается в выборе оптимального действия при динамическом изменении параметров.

Возможны два подхода к процессу диагностики ситуаций при динамическом изменении параметров объекта наблюдения: апостериорная диагностика и диагностика в псевдореальном времени.

Для апостериорной диагностики (т.е. такой диагностики, когда известны значения датчиков на всем рассматриваемом временном интервале) можно использовать само построенное де-

рево решений. Однако при классификации «на лету», когда значения датчиков для последующих временных отсчетов еще неизвестны, одного дерева решений недостаточно.

В нашей задаче процесс диагностики в псевдореальном времени можно рассматривать как обработку последовательно поступающих значений датчиков системой диагностики и выявление некорректного поведения на основе изначально заданных ситуаций. Построенное предварительно темпоральное дерево решений само по себе не может справиться с этой задачей, так как по построению видно, что условие неубывания временных меток при движении от корня дерева к листьям не выполняется. Поэтому потребуется некий агент, который, во-первых, будет следить за временными метками, а во-вторых, будет какое-то время хранить не потребовавшиеся на очередном шаге значения датчиков для их возможного дальнейшего использования.

Кроме того, при построении темпорального дерева решений был введен временной интервал t^* , на котором задавались изначально некорректные ситуации. Потребуется t^* выше-описанных агентов, чтобы определить возможную некорректную ситуацию: первый агент начинает работу в момент времени $t=0$, второй — в момент времени $t=1$, ..., последний, t^* -ый агент начинает работу в момент времени $t=t^*-1$. Для организации совместной работы этих агентов потребуется агент-координатор, который будет получать информацию с датчиков, рассылать ее рабочим агентам и получать от них сведения о некорректной работе объекта или системы (Рис. 10).

Предложенная многоагентная система используется как при диагностике с использованием деревьев решений, построенных алгоритмом CPD, так и при диагностике с использованием деревьев решений, построенных алгоритмом temporal ID3. Отличие только в том, что в первом случае рабочие агенты не хранят предшествующие значения датчиков. В случае, если на очередном шаге нет проверок в узле дерева (так как не совпадают временные метки), агент просто ждет появления последующих значений. Такая система может использоваться и при бортовой диагностике, так как менее требовательна к вычислительным ресурсам.

Моделирование процесса диагностики на основе изложенных алгоритмов состоит из нескольких этапов: первый — построение темпорального дерева решений на основе таблицы с исходными некорректными ситуациями; второй — создание многоагентной системы для диагностики в псевдореальном времени; третий — непосредственно процесс диагностики. Результаты программного моделирования представлены в Табл. 2. Исходные данные для таблицы, на основе которой проводилось построение темпорального дерева решений, генерировались случайным образом по описанию предметной области. Каждая строка таблицы соответствует одному тесту. Пример № 0 (строка в Табл. 2) соответствует тесту, представленному в Табл. 1. Описание предметной области (метаданные) включает в себя:

- описание датчиков и их возможных значений;
- временной интервал t^* , в течение которого может возникнуть некорректная ситуация;

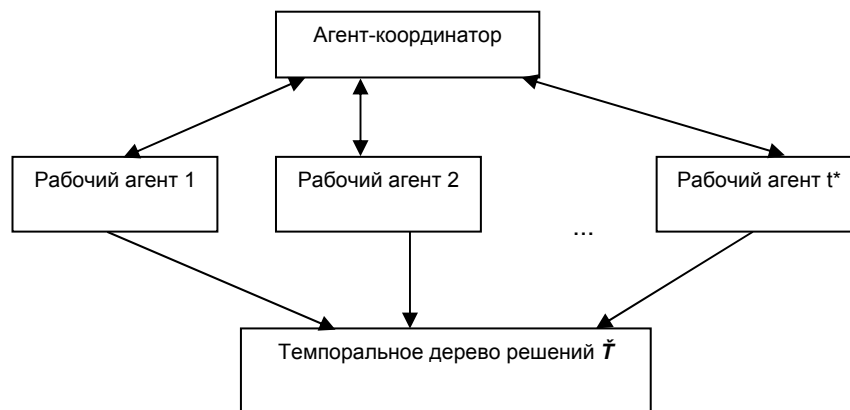


Рис. 10. Взаимодействие агентов при диагностике в псевдореальном времени

- описание восстановительных действий, их стоимостей и порядка над ними.

Для каждого теста выбранным алгоритмом строится темпоральное дерево решений. Полученное дерево решений далее используется многоагентной системой для отнесения ситуации, возникшей на объекте, к одному из классов ситуаций, для которых известны необходимые управляющие воздействия.

Второй этап моделирования заключается в генерации ситуаций, которые могут происходить на объекте, с последующей классификацией сгенерированных ситуаций. Представленные тесты содержат в среднем 40% некорректных ситуаций, остальные ситуации относятся к корректным. В Табл. 2 показано, какое количество тестовых ситуаций отнесено каждым из алгоритмов к некорректным. Были рассмотрены два варианта диагностирования. В первом случае выполняется классификация ситуации целиком – это возможно, если показания датчиков сняты на временном интервале, не превышающем временной интервал, указанный в исходной

таблице. При этом для классификации требуется единственное дерево. Другой вариант - классификация на большом временном интервале: на вход последовательно поступают показания датчиков. В этом случае для классификации потребуется несколько деревьев решений (точнее, столько, какова величина временного интервала, использованного при построении деревьев). Помимо этого потребуются агенты, работающие с каждым из этих деревьев (выполняют функции «получение данных», «контроль времени») и агент-координатор, который получает данные и рассылает их рабочим агентам. Такая многоагентная система позволяет проводить бортовую диагностику в псевдореальном времени.

При проведении эксперимента было выявлено, что темпоральные деревья решений, построенные с использованием алгоритмов CPD и Temporal ID3, правильно определяют значительное количество некорректных ситуаций и выбирают нужные восстановительные действия. При этом следует отметить, что одним из

Табл. 2. Результаты моделирования процесса диагностики

№	Ситуации для построения темпорального дерева		t*	Число ситуаций для апостериорной диагностики	Число временных тактов для диагностики в псевдореальном времени	Число некорректных ситуаций, выявленных алгоритмом			
	количество	Мин. крайний срок				Апостериорная диагностика		Диагностика в псевдореальном времени	
						T ID3	CPD	T ID3	CPD
(0)	8	8	8	20	20	3	7	6	12
(1)	8	3	8	200	200	68	99	69	111
(2)	8	3	8	200	200	64	105	79	113
(3)	8	3	8	200	200	64	112	73	116
(4)	20	3	8	200	200	139	143	147	150
(5)	20	5	8	200	200	126	128	112	142
(6)	20	6	16	200	200	178	147	173	143
(7)	20	8	16	200	200	180	134	181	122
(8)	20	10	16	200	200	161	115	157	126
(9)	20	9	24	200	200	133	162	115	165
(10)	20	12	24	200	200	168	132	164	138
(11)	20	15	24	200	200	171	152	149	147
(12)	20	3	8	200	200	75	159	69	156
(13)	20	3	8	200	200	152	154	161	151
(14)	20	5	8	200	200	130	111	120	112
(15)	20	6	16	200	200	130	162	130	146
(16)	20	8	16	200	200	132	134	99	126
(17)	20	10	16	200	200	150	129	151	128
(18)	20	9	24	200	200	146	134	157	111
(19)	20	12	24	200	200	115	123	107	120
(20)	20	15	24	200	200	141	157	128	156
(21)	20	3	8	200	200	117	132	108	127
(22)	20	4	8	200	200	134	175	121	163
(23)	20	5	8	200	200	124	94	124	105
(24)	20	6	16	200	200	125	155	130	156
(25)	20	9	24	200	200	128	123	125	116

недостатков алгоритма CPD является большое число ложных срабатываний — так называемых ошибок первого рода. Поэтому минимизация размера дерева решений не должна быть единственным критерием при его построении.

С этой точки зрения алгоритм Temporal ID3 является более предпочтительным. Большее по размерам дерево решений, полученное с помощью данного алгоритма, может обладать большей различающей способностью, однако при этом вырастают требования к вычислительным ресурсам. В связи с этим необходимо попытаться найти некий разумный компромисс.

Заключение

В работе был предложен подход к диагностике состояний динамического объекта с использованием деревьев решений. Для использования деревьев решений в целях диагностики ситуаций был введен фактор времени как один из важных параметров.

Рассмотрены структуры исходных данных для задачи диагностики с учетом временного фактора. Дано формальное описание темпоральных деревьев решений.

Рассмотрен алгоритм CPD для построения темпоральных деревьев решений. Предложен алгоритм Temporal ID3, использующий иные критерии для построения дерева решений. Проведено программное моделирование работы данных алгоритмов и дано сравнение их воз-

можностей при апостериорной диагностике и диагностике в псевдореальном времени.

Литература

1. Вагин В.Н., Головина Е.Ю., Загорянская А.А., Фомина М.В. Достоверный и правдоподобный вывод в интеллектуальных системах. Второе издание. - М.; Физматлит, 2008.
2. Quinlan J.R. Induction of decision trees // Machine learning. 1986. №1.
3. Brusoni V., Console L., Terenziani P., Dupré D. A spectrum of definitions for temporal model-based diagnosis // Artificial Intelligence. 1998. №102.
4. Console L., Picardi C., Dupré D. Temporal decision trees: model-based diagnosis of dynamic systems on-board // Journal of artificial intelligence research. 2003. №19.
5. Klahr D., Langley P., Neches R. Production System Models of Learning and Development, Cambridge, Mass.: The MIT Press, 1987.
6. Zadeh L. Fuzzy sets, Information and Control, №8, 1965
7. Pawlak Z. Rough sets: theoretical aspects of reasoning about data, Dordrecht; Boston: Kluwer Academic Publishers, 1991.
8. Поспелов Д.А. Искусственный интеллект. Справочник в трех томах, М.: Радио и связь, 1990.
9. Хант Э., Мартин Д., Стоун Ф. Моделирование процесса формирования понятий на вычислительной машине, М.: Мир, 1970
10. Quinlan J.R. Improved use of continuous attributes in C4.5, Journal of artificial intelligence research, №4, 1996
11. Utgoff P.E. Incremental induction of decision trees, Machine learning, №4, 1989
12. Karimi K., Hamilton H.J. Temporal Rules and Temporal Decision Trees: A C4.5 Approach, Tech. Report CS-2001-02, 2001.

Антипов Сергей Геннадьевич. Аспирант. МЭИ (ТУ). E-mail: antisergey@gmail.com

Фомина Марина Владимировна. Кандидат технических наук, доцент МЭИ (ТУ). E-mail: m_fomina2000@mail.ru