

УДК 004.02+004.67

Метод автоматического построения алгоритмов распознавания участков фазовых траекторий

Коваленко Д. С.

МГУ им. М.В. Ломоносова

e-mail: kovalenkods@gmail.com

получена 1 ноября 2009

Ключевые слова: задача обучения по прецедентам, алгоритм распознавания, обучающая выборка, алгебраический подход к задаче построения алгоритмов выделения трендов.

Рассматривается задача автоматического построения алгоритмов распознавания нештатного поведения динамических систем. Информация о поведении системы доступна в виде фазовой траектории в пространстве показаний датчиков. Особенностью рассматриваемой задачи распознавания является тот факт, что траектории могут быть существенно искажены по амплитуде и растянуты или сжаты по времени в различных условиях работы системы. В работе предложен метод автоматического построения алгоритмов распознавания нештатного поведения динамических систем по заданной обучающей выборке. Метод позволяет строить алгоритмы распознавания, которые имеют более высокую устойчивость к искажениям фазовых траекторий по сравнению с другими известными методами построения алгоритмов распознавания.

1. Задача распознавания нештатного поведения динамических систем

Рассмотрим динамическую систему, информация о поведении которой доступна в виде фазовой траектории в пространстве показаний датчиков $X = (x_1, x_2, \dots, x_n)$, где $x_i = f(t_0 + i \cdot \tau)$ — это вектор действительных чисел, который состоит из показаний датчиков, отражающих состояние системы, в момент времени $t_0 + i \cdot \tau$; τ — интервал времени между последовательными измерениями показаний датчиков; $\frac{1}{\tau}$ — частота опроса датчиков. Система может демонстрировать два типа поведения:

1. Штатное поведение.
2. Нештатное поведение. Возможно несколько классов неисправностей, вызывающих данное поведение.

Все множество траекторий, которые могут быть получены с датчиков системы, назовем допустимыми траекториями и обозначим $V = \{X\}$.

Каждому классу неисправности соответствует некоторая характерная траектория Y_{Anom} , такие траектории будем называть эталонными. Если число классов нештатного поведения системы равно L , то обозначим $Z = \{l\}_1^L \cup \{0\}$ — множество ответов, где 0 — соответствует штатному поведению системы. Множество всех отображений из V в Z обозначим: $A = \{Al : V \rightarrow Z\}$.

Участки траекторий, соответствующие различным классам нештатного поведения, могут входить в анализируемую траекторию X в искаженном относительно эталонных траекторий виде. Искажения могут быть по амплитуде и времени. Под искажением траектории по амплитуде будем понимать изменение абсолютных значений точек траектории, без изменения числа отсчетов. Под искажением траектории по времени будем понимать изменение числа отсчетов, на которых определена траектория, то есть добавление в траекторию новых отсчетов или удаление из нее уже существующих.

Будем предполагать, что искаженные траектории, соответствующие различным классам нештатного поведения, различимы между собой и не содержат друг друга в качестве части.

При этом будем говорить, что две точки траектории равны, если равны значения показаний датчиков в этих точках.

Определение 1. Будем говорить, что одна траектория входит в другую, если одна траектория содержит последовательность точек, равную всем точкам другой траектории, причем порядок следования совпадает. Результатом пересечения будем считать общую часть двух траекторий и будем обозначать: $X^i \cap X^j = X^j$, если X^i содержит в качестве части X^j . В противном случае будем говорить, что траектории не содержат друг друга в качестве части и будем обозначать: $X^i \cap X^j = \emptyset$.

Под ошибкой распознавания первого рода будем понимать такую ошибку распознавания, когда алгоритм распознал вхождение траектории нештатного поведения в наблюдаемую траекторию на тех отсчетах, на которых не наблюдается нештатного поведения. Под ошибкой второго рода будем понимать такую ошибку, когда траектория нештатного поведения реально входит в наблюдаемую траекторию, но не распознается алгоритмом распознавания. Неверная классификация траектории нештатного поведения также относится к ошибкам второго рода.

Задача распознавания нештатного поведения состоит в следующем:

Дано:

- Наблюдаемая многомерная траектория X .
- Набор из L классов нештатного поведения системы, для каждого из которых задана эталонная траектория Y_{Anom}^l . Причем траектории разных классов нештатного поведения не содержат друг друга в качестве части.
- Ограничения на полноту и точность распознавания:

$$e_1 \leq const_1 \text{ и } e_2 \leq const_2, \quad (1)$$

где e_1 – число ошибок распознавания первого рода; e_2 – число ошибок распознавания второго рода; $const_1, const_2$ – заданные числовые ограничения.

Требуется с учетом ограничений на полноту и точность провести распознавание нештатного поведения в работе системы.

2. Описание алгебраического подхода

Идея использования алгебраического подхода для выделения трендов была предложена в работе [1], в работе [2] было предложено использование этого подхода для обнаружения нештатных режимов работы динамических систем. Основой алгебраического подхода является разметка анализируемой траектории аксиомами и анализ траектории по полученному ряду разметки.

Аксиома $Ax = Ax(X^*, t)$ – это бинарная функция, определенная в точке t и некоторой ее окрестности X^* на траектории X . Аксиома представляет собой булеву формулу над множеством элементарных условий.

Элементарное условие $Ec = Ec(X^*, t, P)$ – это бинарная функция, определенная на отсчете t и некоторой его окрестности X^* на траектории X , зависящая от набора параметров P . Пример элементарного условия:

$$Ec(X^*, t, P) = \begin{cases} true, & \text{если } \forall i \in [t - k, t + j], i \in \mathbf{N} : \\ & x_{i-1} < x_i < x_{i+1}, f(X^*, i) - x_i > w \geq 0 \\ false, & \text{иначе,} \end{cases} \quad (2)$$

где

$$f(X^*, i) = \frac{1}{2} \cdot (x_{i-1} + x_{i+1}). \quad (3)$$

Данное элементарное условие $Ec(X^*, t, P)$ описывает 'выпуклое' возрастание значений траектории X в окрестности $X^* = (x_{t-k}, \dots, x_{t+j})$ точки t . Параметрами условия являются: $P = \{j, k, w\}$.

Другими примерами элементарных условий являются: убывание значений траектории на некотором интервале, ограничения на среднее значение элементов траектории на заданном интервале и др.

Определение 2. Точка траектории размечается аксиомой, если в данной точке условия аксиомы выполняются.

Определение 3. Траектория размечается набором аксиом, если каждая точка траектории размечается некоторой аксиомой из данного набора.

Определение 4. Набор аксиом будем называть системой аксиом, если он удовлетворяет следующим ограничениям:

- Условие полноты. Это условие означает, что для любой точки допустимой траектории найдется аксиома из набора аксиом, её размечающая.

- *Условие однозначности. Это условие заключается в том, что любая точка допустимой траектории может быть размечена лишь одной аксиомой из набора.*

Разметкой траектории $X = (x_1, x_2, \dots, x_n)$, полученной при помощи системы аксиом As , является последовательность $J = (j_1, j_2, \dots, j_n)$, где j_i — номер аксиомы из набора As , которая размечает отсчет x_i .

В рамках алгебраического подхода распознавание нештатного поведения в работе наблюдаемой системы происходит следующим образом:

1. Размечаются эталонные траектории $\{Y_{Anom}\}$, соответствующие различным классам нештатного поведения.
2. Размечается наблюдаемая траектория X и формируется ряд разметки J .
3. В ряду разметки J ищутся подпоследовательности номеров аксиом, соответствующие разметкам эталонных траекторий.

Таким образом, определение нештатного поведения в работе наблюдаемой системы ведется не путем поиска эталонных траекторий $\{Y_{Anom}\}$ в наблюдаемой траектории X , а путем поиска разметок эталонных траекторий в ряду разметки J .

3. Задача построения алгоритма распознавания по обучающей выборке

Пусть задана выборка TS в виде экземпляров траекторий X , полученных в различных условиях работы системы, с различными искажениями и шумами. Траектории обучающей выборки размечены.

Определение 5. Будем называть траекторию размеченной, если:

- Для каждого вхождения в нее участка нештатного поведения указаны отсчеты начала и окончания данного участка;
- Каждому участку нештатного поведения приписан тип нештатной ситуации из множества $Z = \{l\}_1^L \cup \{0\}$.

Обучающую выборку, все траектории которой размечены, будем называть заданной полно.

Возможен другой вариант задания обучающей выборки, при котором для каждого участка нештатного поведения не заданы точки начала и окончания этого участка. Известной считается лишь точка наступления аварии. Под аварией будем понимать некоторый процесс в системе, который приводит к невозможности системы выполнять все или часть своих функций. Считаем, что каждой аварии предшествует ровно один тип нештатного поведения.

Определение 6. Под траекторией, размеченной неполно, будем понимать такую траекторию, для которой известна лишь точка наступления аварии и не известны точки начала и окончания соответствующего участка нештатного поведения. При этом для точки аварии указан тип нештатного поведения из множества $Z = \{l\}_1^L \cup \{0\}$.

Обучающую выборку, все траектории которой размечены неполно, будем называть выборкой, заданной неполно.

Всю заданную выборку можно разделить на обучающую \widetilde{TS} и контрольную \widehat{TS} , то есть $TS = \widetilde{TS} \cup \widehat{TS}$, $\widetilde{TS} \cap \widehat{TS} = \emptyset$.

Пусть определена целевая функция $\psi(e_1, e_2)$, где e_1 и e_2 — число ошибок распознавания первого и второго рода, которая отвечает следующим требованиям:

- Функция $\psi(e_1, e_2)$ определена для любых неотрицательных целых значений аргументов e_1 и e_2 .
- Функция $\psi(e_1, e_2)$ монотонно возрастает по каждому из аргументов.

Число ошибок первого рода e_1 (второго рода e_2) является функцией от алгоритма распознавания Al и выборки TS , на которой считается число ошибок распознавания:

$$e_1 = e_1(Al, TS) \quad (e_2 = e_2(Al, TS)). \quad (4)$$

Задача построения алгоритма распознавания нештатных ситуаций в работе динамической системы состоит в следующем:

Дано:

- Обучающая выборка \widetilde{TS} и контрольная выборка \widehat{TS} ,
- Целевая функция $\psi(e_1, e_2)$.

Требуется построить алгоритм распознавания Al , который будет удовлетворять следующему набору ограничений:

1. Алгоритм Al должен выдавать ограниченное число ошибок на обучающей выборке \widetilde{TS} :

$$\psi(e_1(Al, \widetilde{TS}), e_2(Al, \widetilde{TS})) \leq P_{wrong}, \quad (5)$$

где P_{wrong} — заданный числовой порог.

2. Алгоритм Al должен обладать способностью к обобщению, то есть распознавать траектории нештатного поведения не только на обучающей выборке \widetilde{TS} , но и на всем множестве допустимых траекторий V .
3. Вычислительная сложность работы алгоритма распознавания $\Theta_{Al}(m)$ на произвольной траектории, длины не больше m , должна быть ограничена наперед заданной функцией $\theta(m)$, которая определяется характеристиками используемого вычислителя и скоростью развития процессов в анализируемой системе:

$$\Theta_{Al}(m) \leq \theta(m). \quad (6)$$

Данная задача построения алгоритма распознавания соответствует классической постановке задачи обучения по прецедентам [3].

Рассмотрим второе требование к решению. Поскольку проверить качество распознавания траекторий нештатного поведения на всем множестве V невозможно, так как оно содержит бесконечное число траекторий, то используется один из общепринятых подходов [3]: использовать контрольную выборку \widehat{TS} для оценки обобщающей способности алгоритма Al .

4. Метод построения алгоритмов распознавания и условия обучаемости

Решением рассматриваемой задачи распознавания в рамках алгебраического подхода является алгоритм Al , который определяется следующими составляющими:

- алгоритм предобработки исходных данных,
- алгоритм разметки и система аксиом,
- алгоритм поиска разметок эталонных траекторий в разметке наблюдаемой траектории.

Все семейство алгоритмов, которые могут быть получены в результате решения рассматриваемой задачи в рамках алгебраического подхода, обозначим S . Данное семейство является достаточно широким, так как число различных алгоритмов предобработки и алгоритмов поиска разметок достаточно велико, число всевозможных аксиом и систем аксиом в общем случае бесконечно. Для решения поставленной задачи предложено разбить S на подсемейства.

Под шаблоном будем понимать следующую комбинацию:

- заданный алгоритм предобработки и диапазон допустимых изменений значений его параметров;
- алгоритм разметки и фиксированное множество элементарных условий, из которых формируются аксиомы;
- заданный алгоритм поиска разметок и диапазон допустимых изменений значений его параметров.

Шаблон определяет подсемейство в семействе решений S . Выделив возможные типы алгоритмов предобработки и алгоритмов поиска разметок, все семейство решений S можно разбить на шаблоны, причем данное разбиение удовлетворяет следующим требованиям:

- Множество шаблонов покрывает все семейство решений S .
- Множества решений, определяемых различными шаблонами, не пересекаются.

Число шаблонов, на которое разбивается семейство S , можно увеличить следующим образом: для каждого выделенного шаблона построить такой же и разграничить области изменения параметров алгоритмов предобработки и поиска разметок.

Разбиение всего семейства решений S на подсемейства позволяет снизить сложность обучения в рамках одного шаблона, так как:

- Изменение типа алгоритма предобработки может существенно изменить входные данные для алгоритма построения системы аксиом. В рамках одного шаблона тип алгоритма предобработки фиксирован.
- Изменение типа алгоритма поиска разметок может существенно изменить рельеф целевой функции для алгоритма распознавания в целом. В рамках одного шаблона тип алгоритма поиска разметок фиксирован.
- Область поиска удается ограничить, отсекая некоторые подсемейства. Например, на основе экспертных данных о системе.

Общую схему метода обучения алгоритмов распознавания нештатного поведения с использованием шаблонов можно представить следующим образом:

1. Разбиение всего семейства решений на шаблоны $S = \{S_1, S_2, \dots, S_h\}$.
2. Для каждого шаблона S_i происходит обучение алгоритма распознавания:
 - (a) Построение системы аксиом.
 - (b) Определение значений параметров алгоритма поиска разметок.
 - (c) Определение значений параметров алгоритма предобработки исходных данных.
 - (d) Проверка критерия останова и переход на шаг 3, в случае его выполнения, иначе переход в подпункт (a).
3. Выбор лучшего алгоритма распознавания из алгоритмов, полученных при обучении в рамках каждого из построенных шаблонов.

Выбор лучшего алгоритма распознавания в п.3 происходит на основе значений целевой функции $\psi(e_1, e_2)$ на контрольной выборке \widehat{TS} .

Алгоритм построения системы аксиом по обучающей выборке, заданной полно, рассмотрен в работе [4]. Алгоритм построения системы аксиом по обучающей выборке, заданной неполно, детально рассмотрен в разделе 5.

Для поиска настроек алгоритмов предобработки и поиска разметок используются алгоритмы локальной оптимизации, в частности алгоритм градиентного спуска.

Для предобработки использовались следующие алгоритмы: сглаживания, сжатия и интерполяции траектории на произвольный коэффициент, быстрое преобразование Фурье. Для поиска разметок использовались следующие алгоритмы: алгоритмы на основе метрики Минковского и DTW (Dynamic Time Warping) [5], алгоритмы на основе нейросетей, кроме того, был предложен алгоритм на основе расширенных разметок [6].

Критерием останова рассматриваемого итерационного алгоритма обучения в рамках шаблона является следующий составной критерий:

- Выполнены условия задачи обучения алгоритма распознавания, перечисленные выше. В этом случае алгоритм считается завершившимся удачно.
- Общее число итераций алгоритма превысило наперед заданное значение или число итераций без улучшения решения превысило заданный параметр. В этом случае алгоритм считается завершившимся неудачно.

Необходимым условием обучаемости семейства алгоритмов распознавания S является различимость разметок эталонных траекторий $\{Y_{Anom}\}$ из обучающей выборки, заданной полно, которое может быть сформулировано в виде следующего ограничения на используемый набор элементарных условий:

$$\forall Y_{Anom}^i, Y_{Anom}^j \in TS, i \neq j : \exists Ec_k \in \{Ec\} \text{ и } \exists t \in Y_{Anom}^i, Y_{Anom}^j : \\ Ec_k(Y_{Anom}^i, t, P) \neq Ec_k(Y_{Anom}^j, t, P), \quad (7)$$

где $\{Ec\}$ — это множество всех заданных элементарных условий.

Это условие означает, что в наборе заданных элементарных условий должны содержаться такие, что для любых двух траекторий из заданной выборки, соответствующих различным классам нештатного поведения, найдется такое элементарное условие и такая точка в каждой из траекторий, на которой данное условие выполнено в одной траектории и не выполнено в другой. Сложность проверки этого условия квадратично зависит от числа эталонных траекторий и линейно от числа элементарных условий.

5. Общая схема алгоритма построения системы аксиом

Алгоритм построения системы аксиом по обучающей выборке, заданной неполно, состоит из следующих основных этапов:

1. Выбор элементарных условий и настройка их параметров по обучающей выборке. На этом этапе из множества всех описанных элементарных условий выбираются те, которые чаще выполняются на траекториях, содержащих участки нештатного поведения, и реже на траекториях нормального поведения из обучающей выборки. Рассматривается весь доступный набор элементарных условий и для каждого условия выбираются наиболее подходящие параметры. В результате, выходом данного этапа является набор из элементарных условий и их параметров.
2. Формирование аксиом из выбранных элементарных условий. На данном этапе происходит объединение элементарных условий в аксиомы. Лучшие аксиомы, которые чаще выполняются на траекториях, содержащих участки нештатного поведения, и реже на траекториях нормального поведения, сохраняются и поступают на вход следующему этапу алгоритма.

3. Выбор эталонной траектории для каждого класса нештатного поведения и формирование системы аксиом. На данном этапе происходит построение системы аксиом, формирование разметок эталонных траекторий и выбор тех из них, которые наилучшим образом описывают участки нештатного поведения. Выбор лучших производится на основании их оценки по контрольной выборке.

Каждый из этапов алгоритма построения системы аксиом рассмотрен подробно далее.

Алгоритм выбора элементарных условий

Пусть всего описано n элементарных условий. Алгоритм выбора элементарных условий и их параметров состоит из следующих этапов:

1. Создание набора из $2 \cdot n$ элементарных условий, в который включаются все заданные условия вместе с их отрицаниями:

$$Set_{EC} = \{Ec_1, \overline{Ec_1}, Ec_2, \overline{Ec_2}, \dots, Ec_n, \overline{Ec_n}\}. \quad (8)$$

2. Для каждого параметра каждого элементарного условия из набора Set_{EC} определяются минимальные и максимальные границы изменения этого параметра, а также шаг его изменения. Значения определяются путем прогона каждого условия по траекториям обучающей выборки, причем для каждого элементарного условия используется свой способ определения границ. Шаг изменения определяется по формуле:

$$\Delta_{EC}^{param} = \frac{\max_{EC}^{param} - \min_{EC}^{param}}{St}, \quad (9)$$

где \max_{EC}^{param} , \min_{EC}^{param} — это максимальные и минимальные значения для некоторого параметра $param$ элементарного условия Ec .

St — это наперед заданная константа, определяющая размер сетки разбиения множества значений параметра $param$ элементарного условия Ec .

3. Для каждого элементарного условия из Set_{EC} и каждого набора значений его параметров из выбранной сетки вычисляется значение целевой функции F_{EC} на обучающей выборке \widetilde{TS} . Максимальные по значению целевой функции элементарные условия с их параметрами сохраняются и выдаются как результат работы алгоритма на первом шаге.

Целевая функция $F_{EC}(\widetilde{TS})$ выбирается таким образом, чтобы удовлетворять следующему набору требований:

- Если хотя бы на одном отсчете траектории $X \in \widetilde{TS}$, содержащей участки нештатного поведения, выполняется элементарное условие Ec , то целевая функция $F_{EC}(\widetilde{TS})$ должна быть больше, чем если бы Ec не выполнялось ни на одном отсчете X .

- Если условие E_c выполняется хотя бы на одном отсчете траектории нормального поведения $X \in \widetilde{TS}$, то целевая функция $F_{EC}(\widetilde{TS})$ должна быть меньше, чем если бы E_c не выполнялось ни на одном отсчете X .

Требования к целевой функции составлены таким образом, чтобы при работе алгоритма выбора элементарных условий были найдены такие элементарные условия, которые описывают траектории нештатного поведения, то есть выполняются именно на таких траекториях. Для того, чтобы учесть, что некоторое элементарное условие E_{c_i} может выполняться на траекториях нормального поведения и не выполняться на траекториях нештатного поведения, в набор Set_{EC} на первом этапе алгоритма включаются все элементарные условия вместе с их отрицаниями. Тогда $\overline{E_{c_i}}$ с тем же набором параметров будет выполняться на траекториях нештатного поведения и не выполняться на траекториях нормального поведения.

Определим целевую функцию для элементарного условия на обучающей выборке следующим образом:

$$F_{EC} = I_{EC}^A \cdot C_1 \cdot \frac{A_{EC}^{num}}{A^{num}} + (I_{EC}^A - 1) \cdot C_2 + (1 - I_{EC}^N) \cdot C_3 - I_{EC}^N \cdot C_4 \cdot \frac{N_{EC}^{inc}}{N^{size}}, \quad (10)$$

где

$$I_{EC}^A = \begin{cases} 1, & \text{если } A_{EC}^{num} > 0 \\ 0, & \text{иначе} \end{cases} \quad \text{— это индикатор, который равен 1, если элементарное}$$

условие выполнено хотя бы на одном отсчете траектории, содержащей участки нештатного поведения;

C_1, \dots, C_4 — положительные константы, которые являются параметрами и задаются до начала выполнения алгоритма;

A_{EC}^{num} — число траекторий, содержащих участки нештатного поведения, на которых выполнено данное элементарное условие хотя бы на одном отсчете;

A^{num} — общее число траекторий, содержащих участки нештатного поведения;

$$I_{EC}^N = \begin{cases} 1, & \text{если } N_{EC}^{inc} > 0 \\ 0, & \text{иначе} \end{cases} \quad \text{— индикатор, который равен 1, если элементарное}$$

условие выполнено хотя бы на одном отсчете траектории нормального поведения;

N_{EC}^{inc} — число отсчетов, на которых выполнено данное элементарное условие, в траекториях нормального поведения;

N^{size} — общее число отсчетов траекторий нормального поведения.

Алгоритм формирования аксиом

На данном этапе происходит формирование аксиом из выбранных ранее элементарных условий. Основные шаги данного алгоритма:

1. Создание начального набора аксиом:

$$Set_{AX} = \{Ax_1, Ax_2, \dots, Ax_m\}, \quad (11)$$

где $Ax_i = E_{c_i}, \forall i \in [1, m]$.

Для каждого элементарного условия, выбранного ранее, создается аксиома, состоящая только из данного условия, и включается в набор Set_{AX} .

2. Итеративное изменение набора аксиом Set_{AX} :

- (a) Для каждых двух различных аксиом
- Ax_i
- и
- Ax_j
- из набора
- Set_{AX}
- строится две новые аксиомы:

$$Ax_k = Ax_i \wedge Ax_j \text{ и } Ax_l = Ax_i \vee Ax_j. \quad (12)$$

Для новых аксиом Ax_k , Ax_l вычисляется значение целевой функции F_{AX} на обучающей выборке. Если значение целевой функции одной из созданных аксиом больше значения целевой функции для Ax_i , Ax_j , то лучшая из двух новых аксиом сохраняется в наборе Set_{AX}^{new} .

- (b) Аксиомы из набора
- Set_{AX}^{new}
- добавляются в
- Set_{AX}
- :

$$Set_{AX} = Set_{AX} \cup Set_{AX}^{new}. \quad (13)$$

После чего множество Set_{AX}^{new} полагается пустым.

- (c) Проверка критерия останова. Если выполнено одно из следующих условий, то алгоритм формирования аксиом завершается. Условия останова:
- Не создано ни одной новой аксиомы, которая была добавлена в набор Set_{AX} ;
 - Число итераций изменения набора аксиом превысило наперед заданный параметр.
- (d) Сокращение числа элементов набора Set_{AX} до заданного размера и переход на шаг 2а алгоритма.

Целевая функция $F_{AX}(\widetilde{TS})$ для аксиомы Ax на обучающей выборке \widetilde{TS} должна удовлетворять следующему набору ограничений:

- Если хотя бы на одном отсчете траектории из $X \in \widetilde{TS}$, содержащей участки нештатного поведения, выполняется аксиома Ax , то целевая функция $F_{AX}(\widetilde{TS})$ должна быть больше, чем если бы Ax не выполнялось ни на одном отсчете X .
- Если аксиома Ax выполняется хотя бы на одном отсчете траектории нормального поведения $X \in \widetilde{TS}$, то целевая функция $F_{AX}(\widetilde{TS})$ должна быть меньше, чем если бы Ax не выполнялась ни на одном отсчете X .
- Чем ближе число участков, размеченных аксиомой Ax , к числу участков нештатного поведения одного типа на траекториях выборки \widetilde{TS} , тем больше должно быть значение целевой функции $F_{AX}(\widetilde{TS})$.
- Чем меньше разброс длин участков, размеченных траекторией Ax , тем больше должно быть значение целевой функции $F_{AX}(\widetilde{TS})$.

Требования к целевой функции составлены таким образом, чтобы при работе алгоритма формирования аксиом были построены такие аксиомы, которые описывают траектории нештатного поведения, то есть выполняется именно на таких траекториях. При этом учитывается, что разброс длин участков нештатного поведения ограничен.

Определим целевую функцию для аксиомы Ax на выборке \widetilde{TS} следующим образом:

$$F_{AX} = I_{AX}^A \cdot C_5 \cdot \frac{A_{AX}^{num}}{A^{num}} + (I_{AX}^A - 1) \cdot C_6 + C_7 \cdot \frac{A_{AX}^{area} + A^{num} - |A_{AX}^{area} - A^{num}|}{A_{AX}^{area} + A^{num}} + C_8 \cdot R_{AX}^{\min} + \\ + C_9 \cdot (1 - T_{AX}^{\max})^{C_{10}} + (1 - I_{AX}^N) \cdot C_{11} - I_{AX}^N \cdot C_{12} \cdot \frac{N_{AX}^{inc}}{N^{size}} + C_{13} \cdot \frac{C_{14}}{C_{14} + N_{AX}^{area}}, \quad (14)$$

где

$$I_{AX}^A = \begin{cases} 1, & \text{если } A_{AX}^{num} > 0 \\ 0, & \text{иначе} \end{cases} \quad \text{— это индикатор, который равен 1, если аксиома}$$

выполнена хотя бы на одном отсчете траектории, содержащей участки нештатного поведения;

C_5, \dots, C_{14} — положительные константы, которые являются параметрами и задаются до начала выполнения алгоритма;

A_{AX}^{num} — число траекторий, содержащих участки нештатного поведения, на которых выполнена данная аксиома хотя бы на одном отсчете;

A^{num} — общее число траекторий, содержащих участки нештатного поведения;

A_{AX}^{area} — число областей, состоящих из идущих подряд отсчетов траекторий, содержащих участки нештатного поведения, на которых выполняется аксиома;

$R_{AX}^{\min} = \min \left(\frac{\min(\text{size}(A_{AX}^{area}))}{\max(\text{size}(A_{AX}^{area}))} \right)$ — минимальное значение отношения минимальной длины области, состоящей из подряд идущих отсчетов, на которых выполняется аксиома, т.е. минимальной области из A_{AX}^{area} , к максимальной длине такой области на траектории, содержащей участки нештатного поведения. Минимум берется по всем траекториям, содержащим участки нештатного поведения, относительно которых вычисляется значение функции F_{AX} ;

$T_{AX}^{\max} = \max \left(\frac{A_{AX}^{inc}}{A_{AX}^{size}} \right)$ — максимальное значение отношения числа отсчетов, на которых выполняется аксиома, к общему числу отсчетов траектории, содержащей участки нештатного поведения. Максимум берется по всем траекториям, на которых вычисляется значение данной функции F_{AX} для данной аксиомы;

$$I_{AX}^N = \begin{cases} 1, & \text{если } N_{AX}^{inc} > 0 \\ 0, & \text{иначе} \end{cases} \quad \text{— индикатор, который равен 1, если аксиома выпол-}$$

няется хотя бы на одном отсчете траектории нормального поведения;

N_{AX}^{inc} — число отсчетов, на которых выполняется данная аксиома, в траекториях нормального поведения;

N^{size} — общее число отсчетов траекторий нормального поведения;

N_{AX}^{area} — число областей, состоящих из идущих подряд отсчетов траекторий нормального поведения, на которых выполняется данная аксиома.

Алгоритм построения системы аксиом и разметок траекторий нештатного поведения

Построение системы аксиом происходит путем объединения в набор части аксиом, построенных на этапе 2 основного алгоритма. Чтобы полученный набор аксиом удо-

влетворял требованиям однозначности и полноты, заданным в определении системы аксиом, он преобразуется следующим образом:

- Для удовлетворения требования однозначности вводится приоритет аксиом в наборе. При разметке траекторий аксиомы выбираются из системы в соответствии с приоритетом. Если некоторая аксиома выполняется, то аксиомы с меньшим приоритетом не проверяются.
- Для удовлетворения требования полноты в набор аксиом добавляется специальная аксиома, которую будем называть 'нулевой' и обозначать Ax_0 . Нулевая аксиома выполняется на любой точке траектории и всегда имеет наименьший приоритет в наборе аксиом.

Алгоритм формирования системы аксиом включает следующие этапы:

1. Для каждой аксиомы, построенной на предыдущем этапе, создается система аксиом, состоящая только из данной аксиомы и формируется набор систем аксиом Set_{AS} :

$$Set_{AS} = \{As_1, As_2, \dots, As_r\}, \quad (15)$$

где $As_i = \{Ax_i, Ax_0\}, \forall i \in [1, r]$.

2. Для каждой системы аксиом As_i из множества Set_{AS} запускается итеративный процесс ее дополнения аксиомами из набора Set_{AX} , сформированного на этапе 2 основного алгоритма:

- (a) Для каждой аксиомы Ax_j из набора аксиом Set_{Ax}/As_i , т.е. такой, которая не входит в As_i , формируется новая система аксиом $As_k = As_i \cup Ax_j$, полученная путем добавления аксиомы Ax_j в систему As_i . Причем аксиома Ax_j включается в систему аксиом с наименьшим приоритетом, за исключением нулевой аксиомы, приоритет которой всегда считается наименьшим в системе аксиом.

- (b) Для каждой построенной системы аксиом As_k формируются разметки эталонных траекторий: траектории обучающей выборки \widetilde{TS} , содержащие участки нештатного поведения, размечаются системой аксиом As_k . Затем разметки таких траекторий по одному типу нештатного поведения объединяются. Полученные разметки для различных типов нештатного поведения назовем обобщенными разметками.

Объединение разметок в обобщенные происходит путем выделения наибольшей совпадающей части в заданном наборе разметок. Если хотя бы одна аксиома из системы выполняется хотя бы на одном отсчете всех траекторий, содержащих участки нештатного поведения, то обобщенная разметка будет содержать минимум один отсчет, размеченный данной аксиомой. Обобщенная разметка так же может оказаться пустой.

- (c) Для каждой построенной системы аксиом As_k производится разметка траекторий контрольной выборки \widehat{TS} и вычисление значения целевой функции $F_{AS}(\widehat{TS})$.

- (d) Если минимальное полученное значение F_{AS} меньше значения целевой функции для исходной системы аксиом As_i , то As_i заменяется той системой аксиом, для которой получено минимальное значение F_{AS} .
- (e) Если одно из следующих условий выполнено, то происходит переход на шаг 3 данного алгоритма, иначе переход на шаг 2:
- Систему аксиом As_i не удалось улучшить, т.е. построить на ее основе новую систему с меньшим значением целевой функции;
 - Достигнуто значение целевой функции для системы аксиом As_i на контрольной выборке \widehat{TS} , не превышающее наперед заданное значение.
3. Выбор лучшей системы аксиом из всех построенных на предыдущем этапе по значению целевой функции $F_{AS}(\widehat{TS})$ на контрольной выборке \widehat{TS} .

Целевой функцией для системы аксиом F_{AS} на контрольной выборке \widehat{TS} является функция $\psi(e_1(Al, \widehat{TS}), e_2(Al, \widehat{TS}))$, заданная в постановке задачи обучения алгоритма распознавания.

6. Результаты численного исследования

Было проведено численное исследование на модельных данных предложенного метода построения алгоритмов распознавания и ряда известных методов. Были исследованы алгоритмы на основе нейросетей, алгоритмы на основе метрики Минковского и DTW (Dynamic Time Warping) [5], алгоритмы на основе преобразований Фурье и вейвлет-преобразований, алгоритм "Гусеница" (Singular Spectrum Analysis) [7].

Алгоритм распознавания	Число ошибок первого рода	Процент ошибок второго рода
Алгоритм распознавания на основе нейросетей	291	17%
"Гусеница" (SSA)	106	32%
Алгоритм на основе преобразований Фурье	180	30%
Алгоритм на основе вейвлет-преобразований	95	16%
Алгоритм на основе метрики Минковского	320	35%
Dynamic Time Warping	330	25%
Алгебраический подход	18	5%

Таблица 1. Результаты численного исследования алгоритмов распознавания

Выборка TS была задана полно. Общее число траекторий нештатного поведения в модельном ряду — 50, число классов нештатного поведения — 8. Длина траекторий

обучающей \widehat{TS} и контрольной выборки \widehat{TS} — 10000 отсчетов, средняя длина эталонной траектории — 15 отсчетов. Характеристики искажений участков нештатного поведения относительно эталонных траекторий: искажения по времени до 50%, по амплитуде до 10%. Результаты численного исследования представлены в таблице 1.

В условиях существенных искажений траекторий нештатного поведения алгоритм распознавания, построенный с помощью предложенного метода построения алгоритмов распознавания, показал лучшие результаты среди всех исследованных алгоритмов.

При использовании выборки, заданной неполно, ни один из перечисленных методов распознавания не применим, кроме предложенного в данной работе метода для алгебраического подхода. При следующих характеристиках исходных данных: длина траекторий обучающей \widehat{TS} и контрольной выборки \widehat{TS} — 20000 отсчетов, искажения по времени — до 10%, по амплитуде — до 5%, были получены следующие результаты: число ошибок первого рода — 17, процент ошибок второго рода — 10%. Полученные результаты показывают возможность применения и высокую эффективность распознавания нештатного поведения алгоритмом, построенным с помощью предложенного метода построения алгоритмов распознавания.

Список литературы

1. Рудаков К. В., Чехович Ю. В. О проблеме синтеза обучающих алгоритмов выделения трендов (алгебраический подход) // Прикладная математика и информатика. М.: Изд-во факультета ВМиК МГУ, 2001. № 8. С. 97-114.
2. Коваленко Д. С., Костенко В. А., Васин Е. А. Исследование применимости алгебраического подхода к анализу временных рядов // Методы и средства обработки информации. М.: Изд-во факультета ВМиК МГУ, 2005. С. 553-559.
3. Воронцов К. В. Комбинаторный подход к оценке качества обучаемых алгоритмов // Математические вопросы кибернетики. 2004. №13. С.5-36. <http://www.ccas.ru/frc/papers/voron04mpc.pdf>
4. Коваленко Д. С., Костенко В. А. Метод построения алгоритмов распознавания, основанных на идеях аксиоматического подхода // Сборник научных трудов XI Всероссийской научно-технической конференции «Нейроинформатика-2009». М.: Изд-во МИФИ, 2009.
5. Keogh E. J., Michael J. Pazzani Derivative Dynamic Time Warping. // First SIAM International Conference on Data Mining (SDM'2001), Chicago, USA. 2001. <http://www.ics.uci.edu/pazzani/Publications/sdm01.pdf>.
6. Коваленко Д. С. Методы нечеткого сравнения и голосования для построения распознавателей нештатного поведения динамических систем // Труды V Московской международной конференции по исследованию операций «ORM-2007». М.: МАКС Пресс, 2007. С.123-125.

7. Данилов Д. Л., Жиглявский А. А. Главные компоненты временных рядов: метод "Гусеница" // СПб.: Санкт-Петербургский университет, 1997.

Method of automated construction of pattern recognition algorithms on phase paths

Dmitry Kovalenko

Keywords: machine learning, recognition algorithm, training set, problem of constructing an algorithm on a training set, algebraic approach.

The problem of automated construction of recognizers of anomalies in the behavior of complicated dynamical systems is solved by means of analyzing trajectories obtained from sensors surrounding the system. A specific feature of the problem consists in the fact that, depending on the individual properties of the system and conditions of its operation, trajectories that contain anomalies may significantly differ from each other in amplitude and length. Besides, the training set could be incompletely defined. The algorithm described here is based on the idea of applying an algebraic approach to the labeling of trajectories. It allows to construct recognizers of abnormal behavior of complicated dynamical systems. The training of the algorithm could be done on an incompletely defined training set.

Сведения об авторе:

Коваленко Дмитрий Сергеевич,
МГУ им. М.В. Ломоносова, аспирант.