

ИНФОРМАТИКА,
ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА
И УПРАВЛЕНИЕ

INFORMATICS,
COMPUTER ENGINEERING
AND CONTROL

УДК 004.942

DOI: 10.17212/1814-1196-2020-4-159-176

Детекция событий разного типа в гибридных динамических системах*

Е.А. ПОПОВ^а, Ю.В. ШОРНИКОВ^б

630073, РФ, г. Новосибирск, пр. К. Маркса, 20, Новосибирский государственный технический университет

^а e.popov@corp.nstu.ru ^б shornikov@inbox.ru

Гибридные динамические или просто гибридные системы (ГС) являются современным аппаратом для моделирования дискретно-непрерывных процессов в различных прикладных областях: электроэнергетике, космонавтике, промышленности, экономике, диспетчеризации транспорта и т. д. Ключевым отличием ГС от классических динамических является наличие событий переключения непрерывных режимов. Времена событий определяются нулями непрерывных событийных функций. При невозможности аналитического расчета момента возникновения события применяют численные алгоритмы обнаружения, работающие совместно с алгоритмами интегрирования дифференциальных уравнений.

События в ГС принято разделять на события состояния и события времени. В качестве событий времени в литературе обычно рассматриваются только явные события времени с событийными функциями в виде линейных полиномов от времени. В работе рассматривается класс неявных событий времени, а также перечисляются их возможные источники. Более того, в традиционное разбиение событий на односторонние, двусторонние и критичные к точности обнаружения добавляется класс труднообнаруживаемых событий. Такие события характеризуются множественными пересечениями событийной функции нуля в пределах одного шага интегрирования. Не все алгоритмы могут гарантированно обнаруживать события этого типа.

Гетерогенные ГС, состоящие из процессов различной физической природы, в общем случае характеризуются значительной жесткостью и высокой размерностью режимов, обычно представленных в виде дифференциально-алгебраических систем уравнений с событиями разных типов.

Последняя особенность ограничивает применимость классических алгоритмов обнаружения, ориентированных на какой-то один тип событий. Поэтому в работе предлагается методология комплексного обнаружения событий, когда для каждого типа событий выбирается свой алгоритм обнаружения. Совместная работа нескольких алгоритмов может обеспечить корректное обнаружение событий разных типов и, возможно, повысить скорость расчетов.

Для конкретной ГС построен комплексный алгоритм обнаружения событий, гарантированно обнаруживающий все события быстрее в среднем на 17 %.

* Статья получена 02 сентября 2020 г.

Ключевые слова: гетерогенные динамические системы, гибридные динамические системы, дискретно-непрерывные процессы, событийные функции, типы событий, обнаружение событий, вычислительный эксперимент, инструментальное моделирование

ВВЕДЕНИЕ

Во многих областях науки и техники (например, в электроэнергетике, робототехнике, химической промышленности) возникает необходимость в моделировании дискретно-непрерывных процессов. Такие динамические системы с разрывами первого рода в производных фазовых переменных могут быть естественным образом представлены в виде гибридных динамических или просто гибридных систем (ГС) [1–3]. ГС является совокупностью непрерывных режимов, например, в классе задач Коши для явных систем обыкновенных дифференциальных уравнений. В каждый непрерывный момент времени ГС может находиться только в одном режиме. Режимы сменяют друг друга при возникновении определенных событий, которые и приводят к разрывам.

Наличие событий является ключевым отличием ГС от классических динамических. В общем случае мы не можем аналитически решить дифференциальное уравнение из-за невозможности выражения многих первообразных через элементарные функции. В таком случае моменты переключений режимов в ГС также не могут быть определены аналитическими методами. Это определяет необходимость использования численных алгоритмов обнаружения событий.

За все время развития теории дискретно-непрерывных и гибридных систем, начиная с 1970-х годов, были предложены множественные алгоритмы обнаружения событий.

В работе [4] рассматриваются вопросы обнаружения разрывов в производных фазовых переменных и определения их численных характеристик по поведению оценки локальной ошибки численной схемы.

В [5] предлагается альфа-метод третьего порядка точности. В [6] для локализации событий используется метод хорд. В [7] на этой стадии применяется комбинация обратной эрмитовой интерполяции и метода хорд. В работе [8], вдохновленной [9], для интерполяции событийных функций также использована обратная эрмитова интерполяция.

В [10] рассмотрен ограниченный класс событийных функций и предложен алгоритм обнаружения событий на основе рядов Штурма. Представленный алгоритм гарантирует обнаружение событий, событийные функции которых несколько раз пересекают ноль в пределах одного шага интегрирования.

Работа [11] посвящена алгоритму обнаружения событий, ориентированному на использование с алгоритмами интегрирования на основе формулы дифференцирования назад. Представленный алгоритм гарантированно обнаруживает события, как и предложенный в [10]. Стадия обнаружения основана на применении интервальной арифметики и исключающих тестов. На стадии локализации предлагается использовать метод Ньютона–Рафсона, который сходится за малое количество итераций в связи с достаточной близостью начального приближения, полученного на стадии обнаружения. Предложен-

ный алгоритм обнаружения событий гарантированно находит требуемый нуль событийной функции в случае, если его кратность равна единице.

В [12] предложен алгоритм, подобный представленному в [11].

В [13] и [14] авторы расширяют фазовое пространство системы производными событийных функций и на стадии обнаружения проверяют смену знака как событийными функциями, так и их производными.

Во всех вышеперечисленных алгоритмах присутствует стадия обнаружения события, на которой происходит расчет состояния системы в следующий момент времени. При численном моделировании ГС с односторонними событиями в малой окрестности области неопределенности модели это неизбежно приведет к аварийному завершению вычислительного эксперимента при попытке рассчитать состояние системы в области фазового пространства, где не определен текущий режим.

В [15] был предложен иной класс алгоритмов обнаружения, основанных на экстраполяции событийной функции. Этот подход позволяет управлять размером шага интегрирования в соответствии с динамикой событийной функции таким образом, что система будет асимптотически приближаться к границе режима, не пересекая его.

В [16] представлен алгоритм выбора размера шага интегрирования с учетом точности и устойчивости численной схемы и поведения событийной функции.

Применение всех вышеописанных алгоритмов обнаружения оказывается ограниченным при численном моделировании ГС, в которых могут происходить события разных типов. Поэтому актуальным является разработка алгоритмов обнаружения, способных корректно и эффективно обрабатывать события с учётом их типов.

1. ГЕТЕРОГЕННЫЕ ДИНАМИЧЕСКИЕ СИСТЕМЫ

Важным подклассом ГС являются гетерогенные ГС (ГГС), состоящие из компонентов различной физической природы. Примерами таких систем являются: электроэнергетические системы с электромеханическими машинами, цифровыми устройствами автоматики, трансформаторами, линиями электропередачи и нагрузками; биоэлектрические протезы конечностей, считывающие электрическую активность нейронов; электротранспорт и т. д.

Постоянные времени разнородных компонентов обычно значительно отличаются. Поэтому ГГС часто описываются *жесткими* системами дифференциальных уравнений. Например, длительности электромеханических и электромагнитных переходных процессов в электроэнергетических системах могут отличаться на восемь и более порядков [17].

В общем случае ГГС характеризуются *высокой размерностью*, особенно для непрерывных процессов, описываемых системой дифференциальных уравнений в частных производных. Например, циркуляция теплоносителя в ядерном реакторе.

Дифференциально-алгебраические системы уравнений (ДАУ) являются естественной формой описания процессов самой различной природы. Сложные ограничения на фазовые переменные, законы сохранения массы, энергии и момента уменьшают количество степеней свободы совокупной модели по

сравнению с суммой степеней свободы составляющих ее элементов и приводят к системам ДАУ [18]. Естественное представление в форме ДАУ повышает читабельность модели для некомпетентного в вычислительной математике предметного специалиста, который сразу же распознает характерные для его предметной области законы сохранения и другие ограничения. Это позволяет повысить качество моделей за счет раннего обнаружения логических ошибок в процессе моделирования.

С другой стороны, ДАУ могут не задаваться пользователем, а возникать в результате преобразований модели на компонентном или объектно-ориентированном языке моделирования. При составлении совокупной системы уравнений сложной системы из моделей элементов и уравнений связей между ними в общем случае неизбежно возникает неявная система ДАУ [19].

Таким образом, режимы ГГС в общем случае имеют вид задачи Коши для системы ДАУ с ограничениями в виде

$$\mathbf{0} = \mathbf{F}(t, \mathbf{y}(t), \mathbf{y}'(t)),$$

$$\mathbf{y}(t_0) = \mathbf{y}_0, \quad \mathbf{y}'(t_0) = \mathbf{y}'_0,$$

$$pr(t, \mathbf{y}(t)) : \mathbf{g}(t, \mathbf{y}(t)) < \mathbf{0}, \quad t \in [t_0, t^*),$$

где $\mathbf{F} : \mathbb{R} \times \mathbb{R}^N \times \mathbb{R}^N \rightarrow \mathbb{R}^N$ – вектор-функция, удовлетворяющая условиям существования и единственности решения на $[t_0, t^*]$ [20]; t – независимая переменная (время); $\mathbf{y}(t) \in \mathbb{R}^N$ – вектор состояния системы; $\mathbf{y}_0, \mathbf{y}'_0 \in \mathbb{R}^N$ – такие начальные условия, что $\mathbf{0} = \mathbf{F}(t_0, \mathbf{y}_0, \mathbf{y}'_0)$; $pr : \mathbb{R} \times \mathbb{R}^N \rightarrow \{\text{false}, \text{true}\}$ – режимный предикат, являющийся истинным, пока система находится в текущем режиме; $\mathbf{g} : \mathbb{R} \times \mathbb{R}^N \rightarrow \mathbb{R}^S$ – непрерывная событийная вектор-функция, нули которой определяют моменты возникновения событий; x_0 – начальное время режима; t^* – время переключения в следующий режим (события).

В ГГС в связи с наличием разнородных компонентов повышается вероятность возникновения *событий разных типов* в рамках одного прогона модели. Рассмотрим возможные типы событий.

2. ТИПЫ СОБЫТИЙ

Из-за невозможности точного определения момента переключения в связи с ограниченностью разрядной сетки компьютера выделяют три типа событий: односторонние, двусторонние и критичные к точности [3, 15]. В первом случае событийная функция не должна пересечь границу режима $\mathbf{g}(t, \mathbf{y}(t)) = \mathbf{0}$, во втором, наоборот, должна. Критичные к точности обнаружения события должны быть обнаружены с максимальной точностью, но не накладывают ограничений на конечное значение событийной функции. В литературе также поднимался вопрос об обнаружении событий в условиях

множественных пересечений событийной функцией границы режима (рис. 1) [7, 11, 15].

Существуют алгоритмы, не гарантирующие обнаружение события в таких случаях. Более того, такие алгоритмы широко используются в современных инструментальных средах моделирования сложных динамических систем, например, OpenModelica, как будет показано ниже. Поэтому известную классификацию событий по типу границы следует расширить *труднообнаруживаемыми* событиями.

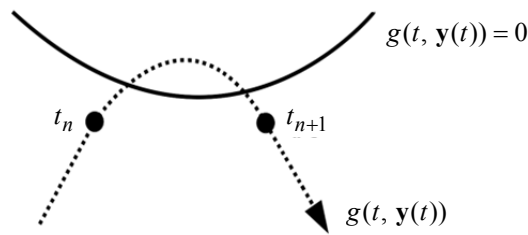


Рис. 1. Труднообнаруживаемое событие

Fig. 1. A difficult-to-detect event

Также события в ГС разделяют на события времени и события состояния по зависимости от фазовых переменных.

Событиями времени называют события, моменты возникновения которых определяются событийными функциями $g(t)$, не зависящими от фазовых переменных системы $y(t)$. Событийные функции *событий состояния* $g(t, y(t))$, наоборот, зависят от фазовых переменных.

В дальнейшем класс событий времени следует разбить на явные и неявные. *Явными событиями времени* будем называть события, событийные функции которых представляют собой линейные полиномы от модельного времени. Например, в момент времени $t = 5$ с должен включиться асинхронный двигатель. Тогда событийная функция в данном случае будет иметь вид $g(t) = t - 5$.

Неявными событиями времени соответственно будем называть все остальные события времени с нелинейными событийными функциями. В литературе обычно рассматривают только явные события времени [14, 18, 19]. Но неявные события времени могут быть частью исходной модели, а также возникнуть в ходе преобразований системы уравнений режима.

Рассмотрим следующий простой пример. Тело падает вертикально вниз с ускорением свободного падения g . Ось Y направим вертикально вверх. В начальный момент времени $t_0 = 0$ объект находится в точке y_0 , а его скорость будет v_0 . Силой сопротивления воздуха пренебрегаем. Необходимо определить время падения до поверхности Земли, моделируемой прямой $y = 0$. ГС включает два режима: «падение» и «покой». В режиме «покой» вычислительный эксперимент завершается. Тогда режим «падение» имеет вид

$$y'(t) = v(t), \quad y(0) = y_0,$$

$$v'(t) = -q, \quad v(0) = v_0,$$

$$pr(y(t)): g(y(t)) = -y(t) < 0.$$

Уравнение движения и закон изменения скорости могут быть получены аналитически, например, в модуле компьютерной алгебры вычислительного ядра инструментальной среды моделирования. Тогда режим «падение» принимает вид

$$y(t) = -\frac{gt^2}{2} + v_0t + y_0,$$

$$v(t) = -gt + v_0,$$

$$pr_2(t): \tilde{g}(t) = \frac{gt^2}{2} - v_0t - y_0 < 0,$$

т. е. событие состояния стало неявным событием времени.

Обнаружение (планирование) события времени сводится к нахождению минимального корня уравнения $g(t) = 0$, не предшествующего моменту переключения в режим, t_0 .

Если гарантированно найти требуемый корень не удастся или на это ушло бы слишком много времени, с точки зрения модуля планирования событий времени такое неявное событие времени является *необнаруживаемым*, т. е. фактически переходит в класс событий состояния, работа с которыми осуществляется с применением алгоритма обнаружения событий состояния.

События времени, момент возникновения которых может быть определен до начала вычислительного эксперимента, называют *экзогенными*, а события времени, момент возникновения которых может быть определен только в процессе вычислительного эксперимента с моделью, — *эндогенными* [18, 19].

На рис. 2 представлена расширенная классификация событий в ГС.

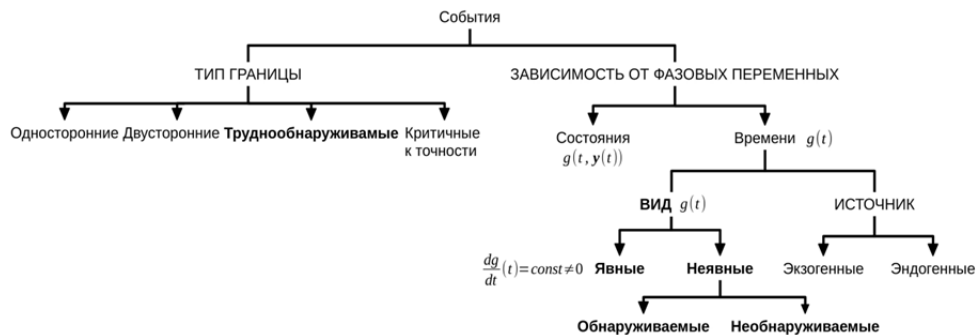


Рис. 2. Расширенная классификация событий в ГС

Fig. 2. An extended classification of HS events

В некоторых работах, например в [11, 21], события времени и состояния называют явными и неявными соответственно.

3. КОМПЛЕКСНОЕ ОБНАРУЖЕНИЕ СОБЫТИЙ

Практически во всех инструментальных средах моделирования сложных динамических систем используется какой-то один алгоритм обнаружения событий, который, по мнению разработчика, является наилучшим при решении большинства задач в его системе. Обычно это алгоритм без гарантии обнаружения труднообнаруживаемых событий, фактически настроенный на работу с критичными к точности обнаружения или двусторонними событиями.

Вместо поиска наилучшего алгоритма обнаружения в настоящей работе предлагается использовать методологию *комплексного обнаружения*, заключающуюся в следующем. Обнаруживаемые события времени планируются наперед. Для односторонних, двусторонних, труднообнаруживаемых и критичных к точности обнаружения событий используются свои отдельные алгоритмы обнаружения. Причем возможно использование одного и того же алгоритма, только по-разному настроенного, для обнаружения событий нескольких типов. Тогда в предельном случае при использовании одного алгоритма обнаружения для всех типов событий состояния комплексный алгоритм вырождается в традиционный, разделяющий события только на события состояния и времени.

Комплексный алгоритм обнаружения событий в общем случае позволяет воспользоваться преимуществами и сгладить слабые стороны традиционных алгоритмов обнаружения. Более того, только комплексные алгоритмы обнаружения позволяют корректно рассчитывать глобальное решение ГС с несколькими типами событий (например, многие ГГС).

Рассмотрим предложенную методологию на примере.

В плоском мире имеется комната в форме окружности радиуса R_O , в центре которой находится препятствие в форме окружности радиусом $R_I < R_O$ (рис. 3). Поместим начало декартовой системы координат в центр окружностей. По комнате непрерывно движется агент, положение которого описывается радиус-вектором $\mathbf{r}(t) = (x(t), y(t))$. При столкновении с внешней или внутренней стеной происходит абсолютно упругий отскок – изменяется направление вектора скорости агента $\mathbf{v}(t) = (v_x(t), v_y(t))$ без изменения его модуля. В начальный момент времени агент находится в точке $\mathbf{r}_0 = (x_0, y_0)$, $R_I < |\mathbf{r}_0| < R_O$, а его скорость равняется $\mathbf{v}_0 = (v_{x_0}, v_{y_0})$. Тогда движение агента может быть описано следующей задачей Коши с ограничениями:

$$\mathbf{r}'(t) = \mathbf{v}(t), \quad \mathbf{r}(t_0) = \mathbf{r}_0, \quad \mathbf{v}'(t) = \mathbf{0}, \quad \mathbf{v}(t_0) = \mathbf{v}_0,$$

$$pr(\mathbf{r}(t), \mathbf{v}(t)) : (g_1(\mathbf{r}(t)) < 0 \vee g_2(\mathbf{r}(t), \mathbf{v}(t)) < 0) \wedge$$

$$(g_3(\mathbf{r}(t)) < 0 \vee g_4(\mathbf{r}(t), \mathbf{v}(t)) < 0),$$

где событийные функции $g_1(\mathbf{r}(t)) = |\mathbf{r}(t)| - R_O$ и $g_3(\mathbf{r}(t)) = R_I - |\mathbf{r}(t)|$ описывают события столкновения с внешней и внутренней стеной соответ-

венно, а $g_2(\mathbf{r}(t), \mathbf{v}(t)) = -g_4(\mathbf{r}(t), \mathbf{v}(t)) = \mathbf{r}(t) \cdot \mathbf{v}(t)$ обеспечивают однократный отскок.

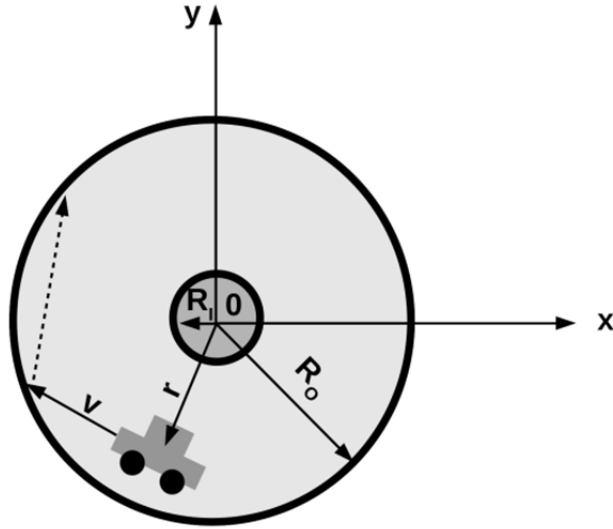


Рис. 3. Агент в круглой комнате с препятствием

Fig. 3. An agent in a circle-shaped room with an obstacle

Диаграмма состояний системы представлена на рис. 4. ГС имеет один режим и два перехода. Один происходит при столкновении с внешней стеной, когда $g_1(\mathbf{r}(t)) \geq 0$ и $g_2(\mathbf{r}(t), \mathbf{v}(t)) \geq 0$, а второй – при столкновении с внутренней стеной, когда $g_3(\mathbf{r}(t)) \geq 0$ и $g_4(\mathbf{r}(t), \mathbf{v}(t)) \geq 0$. Оба перехода осуществляют мгновенное изменение значения \mathbf{v} , что и моделирует абсолютно упругий отскок. При столкновении со стеной нормальная стене составляющая вектора скорости изменяется на противоположную:

$$\mathbf{v}(t^*) = \mathbf{v}(t^* - 0) - 2 \text{proj}_{\mathbf{N}(\mathbf{r}(t^* - 0))} \mathbf{v}(t^* - 0),$$

где $\mathbf{v}(t^* - 0)$ и $\mathbf{r}(t^* - 0)$ означают конечные значения переменных в предыдущем режиме; $\mathbf{N}(\mathbf{r}(t))$ – нормаль к стене, которая в месте столкновения совпадает с радиус-вектором агента.

События, описываемые $g_1(\mathbf{r}(t)) \geq 0 \wedge g_2(\mathbf{r}(t), \mathbf{v}(t)) \geq 0$, относятся к категории критичных к точности обнаружения.

При движении по траектории, близкой к касательной к внутренней стене, за один шаг интегрирования может произойти два пересечения нуля событийными функциями $g_3(\mathbf{r}(t)) \geq 0 \wedge g_4(\mathbf{r}(t), \mathbf{v}(t)) \geq 0$, поэтому эти столкновения представляют собой труднообнаруживаемые события.

В скалярном виде система уравнений с ограничениями принимает вид

$$x'(t) = v_x(t), \quad x(t_0) = x_0, \quad y'(t) = v_y(t), \quad y(t_0) = y_0,$$

$$v_x'(t) = 0, \quad v_x(t_0) = v_{x_0}, \quad v_y'(t) = 0, \quad v_y(t_0) = v_{y_0},$$

$$pr_2(x(t), y(t), v_x(t), v_y(t)) : (\tilde{g}_1(x(t), y(t)) < 0 \vee \tilde{g}_2(x(t), y(t), v_x(t), v_y(t)) < 0) \wedge$$

$$(\tilde{g}_3(x(t), y(t)) < 0 \vee \tilde{g}_4(x(t), y(t), v_x(t), v_y(t)) < 0),$$

$$\text{где } \tilde{g}_2(x(t), y(t), v_x(t), v_y(t)) = -\tilde{g}_4(x(t), y(t), v_x(t), v_y(t)) = x(t)v_x(t) + y(t)v_y(t),$$

$$\tilde{g}_1(x(t), y(t)) = \sqrt{x^2(t) + y^2(t)} - R_O, \quad \tilde{g}_3(x(t), y(t)) = R_I - \sqrt{x^2(t) + y^2(t)}.$$

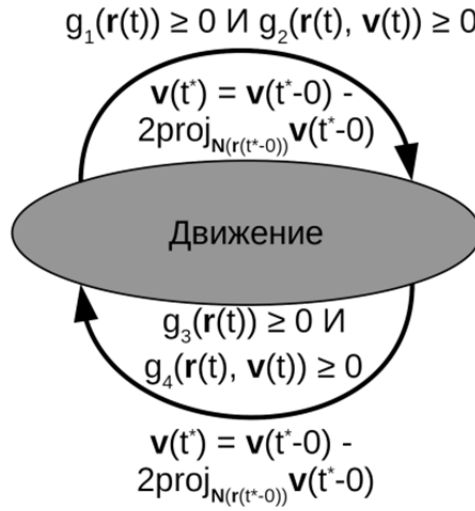


Рис. 4. Диаграмма состояний агента

Fig. 4. An agent's state chart

Выполним расчеты модели со следующими значениями параметров: $R_O = 5$, $R_I = 1$, $\mathbf{r}_0 = (-1, 1; -1, 1)$, $\mathbf{v}_0 = (1, 5; 2, 5)$, минимальный размер шага интегрирования $h_{\min} = 10^{-12}$, максимальный размер шага интегрирования $h_{\max} = 1$, начальный размер шага интегрирования $h_0 = 0.05$, точность интегрирования $\varepsilon_i = 10^{-6}$, точность обнаружения событий $\varepsilon_e = 10^{-6}$, $t_0 = 0$, длина интервала моделирования $T_s = 100$, параметр алгоритма обнаружения событий на основе экстраполяции $\gamma = 0,6$ [16]. Производные событийных функций вычислялись аналитически. Интегрирование будем проводить с помощью численного алгоритма Рунге–Кутты–Мерсона четвертого порядка точности [3].

Программная модель системы, построенная в инструментальной среде ИСМА [22], представлена на рис. 5.

Траектория агента показана на рис. 6.

```

1  const R_O = 5;
2  const R_I = 1;
3
4  x' = vx; x(t0) = -1.1;
5  y' = vy; y(t0) = -1.1;
6  vx' = 0; vx(t0) = 1.5;
7  vy' = 0; vy(t0) = 2.5;
8
9  r = sqrt(x * x + y * y);
10 dotProduct = x * vx + y * vy;
11
12 state motion((r >= R_O && dotProduct >= 0) || (r <= R_I && -dotProduct >= 0))
13 {
14   set vx = vx - 2 * dotProduct / r / r * x;
15   set vy = vy - 2 * dotProduct / r / r * y;
16 }
17 from init, motion;

```

Рис. 5. Программная модель в ISMA

Fig. 5. A program model in ISMA

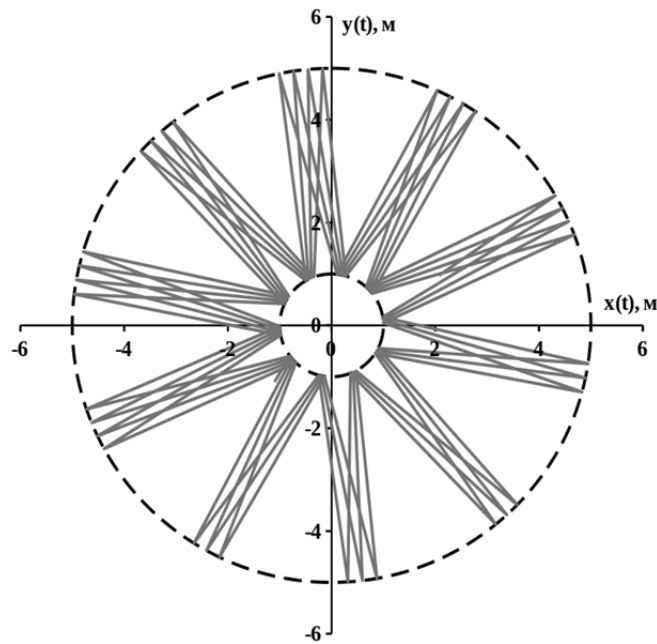


Рис. 6. Решение с применением алгоритма на основе экстраполяции

Fig. 6. Solution obtained using the extrapolation-based algorithm

Построим модель системы в OpenModelica [18] на языке Modelica (рис. 7). Особенности блока when снимают необходимость в применении событийных функций $g_2(\mathbf{r}(t), \mathbf{v}(t))$ и $g_4(\mathbf{r}(t), \mathbf{v}(t))$, обеспечивающих единственность отскока от границы в исходной математической модели.

```

1 model TestCircle
2   Modelica.SIunits.Position x(start = -1.1, fixed = true);
3   Modelica.SIunits.Position y(start = -1.1, fixed = true);
4   Modelica.SIunits.Velocity vx(start = 1.5, fixed = true);
5   Modelica.SIunits.Velocity vy(start = 2.5, fixed = true);
6   parameter Modelica.SIunits.Radius R_0 = 5.0;
7   parameter Modelica.SIunits.Radius R_I = 1.0;
8   Modelica.SIunits.Radius r;
9   equation
10    der(x) = vx;
11    der(y) = vy;
12    der(vx) = 0.0;
13    der(vy) = 0.0;
14    r = sqrt(x * x + y * y);
15
16    when r >= R_0 or r <= R_I then
17      reinit(vx, pre(vx) - 2.0 * (pre(x) * pre(vx) + pre(y) * pre(vy)) / pre(r) / pre(r) * pre(x));
18      reinit(vy, pre(vy) - 2.0 * (pre(x) * pre(vx) + pre(y) * pre(vy)) / pre(r) / pre(r) * pre(y));
19    end when;
20
21 end TestCircle;

```

Рис. 7. Программная модель в OpenModelica

Fig. 7. A program model in OpenModelica

Выполним расчеты модели алгоритмом DASSL с $h_{\max} = 1$, $\varepsilon_i = 10^{-6}$, максимальным порядком 4 и отключенной плотной выдачей. Результаты вычислительного эксперимента представлены на рис. 8.

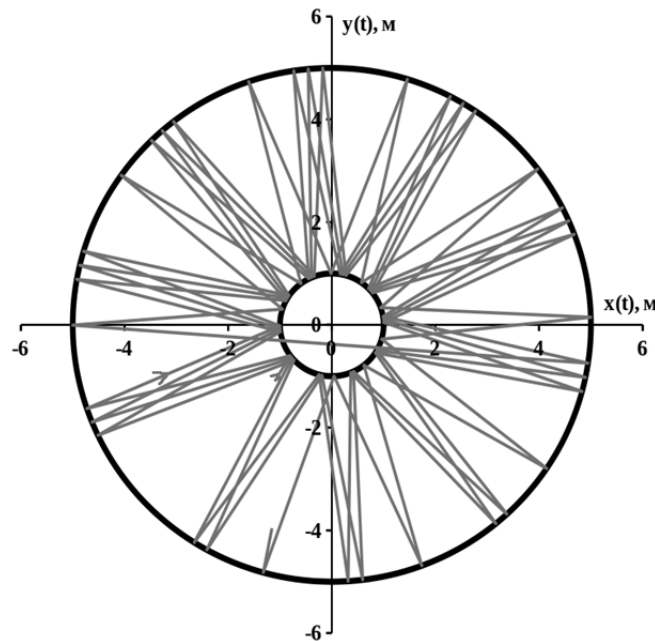


Рис. 8. Результаты расчета в OpenModelica

Fig. 8. Simulation results in OpenModelica

Как можно видеть, агент один раз пересек внутреннее препятствие насквозь, т. е. одно событие было пропущено. В итоге было получено некорректное глобальное решение, не соответствующее математической модели системы. Вероятность пропуска столкновения с внутренней преградой может быть повышена путем выбора меньшего значения внутреннего радиуса R_I . Более того, чем ближе траектория движения агента к касательной к окружно-

сти в ближайшей ее точке, тем выше шанс пропуска события при использовании алгоритмов на основе интерполяции, аналогичных использованному выше.

Воспользуемся комплексным алгоритмом обнаружения со следующей структурой. Для труднообнаруживаемых событий будем использовать алгоритм обнаружения на основе экстраполяции [16], а для обнаружения критичных к точности событий – алгоритм на основе итерационного поиска момента переключения методом деления шага интегрирования пополам [19]. Во втором случае нет никакого смысла использовать требовательный к ресурсам алгоритм с гарантией обнаружения, так как пропуск события физически невозможен: событийная функция пересекает границу режима ровно один раз.

Выполним расчеты модели с теми же параметрами, но с использованием комплексного алгоритма обнаружения и его алгоритмов-компонентов по отдельности при разных значениях длины интервала моделирования $T_s = \{100, 500, 1000, 2000, 5000, 10000, 20000, 50000, 100000\}$.

В таблице представлены численные характеристики вычислительных экспериментов с системой: N_e означает количество обнаруженных событий, T_{CPU} – время ЦПУ Intel i7-7700 в миллисекундах, усредненное по 1000 прогонам модели, N_h – количество шагов интегрирования, N_f – количество расчетов правой части, N_i – количество возвратов в процессе итерационного поиска корня событийной функции. Алгоритм на основе экстраполяции обозначен Э, итерационный алгоритм на основе деления отрезка пополам – И, а комплексный алгоритм, включающий оба предыдущих, – И + Э.

Сравнение алгоритмов обнаружения событий

Comparison of the event detection algorithms

T_s , с	Алгоритм	N_e	N_h	N_f	N_i	T_{CPU} , мс
100	И	72	2351	19230	1495	1,219
	Э	72	4084	24504	0	1,755
	И + Э	72	3140	23868	838	1,429
500	И	359	11821	96710	7521	5,966
	Э	359	20372	122232	0	8,917
	И + Э	359	15671	119064	4173	7,318
1000	И	719	23696	193885	15081	12,151
	Э	719	40794	244764	0	18,492
	И + Э	719	31377	238500	8373	14,796
2000	И	1437	47392	387755	30159	24,619
	Э	1437	81544	489264	0	37,092
	И + Э	1437	62672	476424	16732	29,764

Окончание таблицы

5000	И	3591	118476	969345	75393	62,072
	Э	3591	203786	1222716	0	89,150
	И + Э	3591	156653	1191030	41852	75,207
10000	И	7182	236981	1938930	150805	124,405
	Э	7182	407576	2445456	0	177,273
	И + Э	7182	313229	2381580	83701	151,733
20000	И	14364	473981	3878035	301626	251,141
	Э	14364	815150	4890900	0	352,109
	И + Э	14364	626484	4763340	167406	299,564
50000	И	35908	1184940	9694955	754051	618,403
	Э	35908	2037779	12226674	0	872,656
	И + Э	35908	1566088	11907444	418486	738,028
100000	И	71816	2369904	19390115	1508119	1275,320
	Э	71816	4075557	24453342	0	1805,003
	И + Э	71816	3131907	23812518	836846	1464,494

На рис. 9 представлена зависимость времени ЦПУ в логарифмическом масштабе от длины интервала моделирования для всех трех алгоритмов обнаружения.

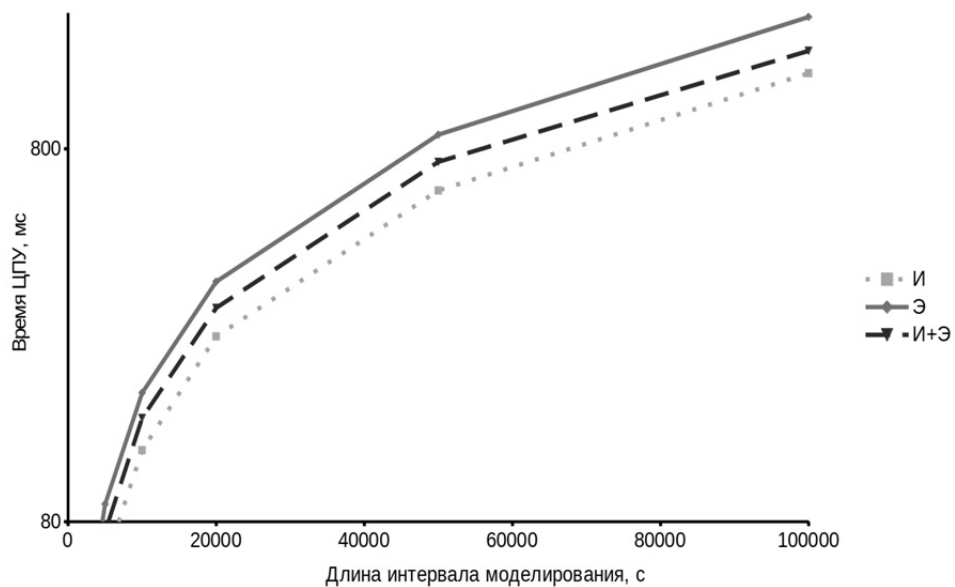


Рис. 9. Время ЦПУ при различных алгоритмах обнаружения событий

Fig. 9. CPU time for different event detection algorithms

По результатам видно, что время расчета практически линейно зависит от длины интервала моделирования. Из-за особенности задачи события распределены по нему почти равномерно. Наглядно видно, что использованный комплексный алгоритм обнаружения занял примерно среднюю позицию по вычислительным затратам между своими алгоритмами-компонентами. Среднее ускорение по сравнению с алгоритмом на основе экстраполяции составило около 17 %. С другой стороны, использованный комплексный алгоритм обнаружения обеспечивает то же корректное обнаружение событий столкновения с внутренней стеной, что и одиночный алгоритм на основе экстраполяции. Таким образом, в данном случае был построен комплексный алгоритм обнаружения, использующий преимущества своих двух алгоритмов-компонентов (высокую скорость работы и гарантированное обнаружение событий) и сглаживающий их недостатки (отсутствие гарантии обнаружения событий и медленную скорость работы).

ЗАКЛЮЧЕНИЕ

Важным практическим классом гибридных систем являются гетерогенные гибридные системы, состоящие из процессов различной физической природы. Они в общем случае характеризуются жесткими режимами высокой размерности, часто в форме дифференциально-алгебраических систем уравнений, и событиями разных типов. Последняя особенность ограничивает применимость традиционных алгоритмов обнаружения событий, ориентированных на какой-то один тип событий.

Принятая классификация событий по типу границы расширена трудно обнаруживаемыми событиями. В большинстве современных инструментальных сред моделирования сложных динамических систем используются алгоритмы обнаружения событий, не дающие гарантий обнаружения таких событий. Кроме этого, в работе были рассмотрены неявные события времени.

При численном моделировании гетерогенных гибридных систем и гибридных систем с событиями разных типов предложено использовать методологию комплексного обнаружения, когда события каждого типа обнаруживаются отдельным алгоритмом. Такие комплексные алгоритмы обнаружения позволяют корректно проводить вычислительный эксперимент, а также в общем случае оказываются эффективнее по временным затратам.

Для конкретной задачи построен комплексный алгоритм обнаружения, обеспечивший среднее ускорение в 17 % по сравнению с алгоритмом-компонентом, выдающим то же корректное глобальное решение.

СПИСОК ЛИТЕРАТУРЫ

1. Goebel R., Sanfelice R.G., Teel A.R. Hybrid dynamical systems: modeling, stability, and robustness. – Princeton: Princeton University Press, 2012. – 232 p.
2. Колесов Ю.Б., Сениченков Ю.Б. Моделирование систем. Динамические и гибридные системы: учебное пособие. – СПб.: БХВ-Петербург, 2012. – 224 с.
3. Новиков Е.А., Шорников Ю.В. Компьютерное моделирование жестких гибридных систем. – Новосибирск: Изд-во НГТУ, 2012. – 451 с.

4. Gear C.W., Østerby O. Solving ordinary differential equations with discontinuities // ACM Transactions on Mathematical Software. – 1984. – Vol. 10, N 1. – P. 23–44. – DOI: 10.1145/356068.356071.
5. O'Regan P.G. Step size adjustment at discontinuities for fourth order Runge-Kutta methods // The Computer Journal. – 1970. – Vol. 13, N 4. – P. 401–404. – DOI: 10.1093/comjnl/13.4.401.
6. Hay J.L., Crosbie R.E., Chaplin R.I. Integration routines for systems with discontinuities // The Computer Journal. – 1974. – Vol. 17, N 3. – P. 275–278.
7. Cellier F.E. Combined continuous/discrete system simulation by use of digital computers: techniques and tools. PhD Thesis. – Zürich, Switzerland: Swiss Federal Institute of Technology Zürich, 1979. – 285 p.
8. Ellison D. Efficient automatic integration of ordinary differential equations with discontinuities // Mathematics and Computers in Simulation (MATCOM). – 1981. – Vol. 23, N 1. – P. 12–20. – DOI: 10.1016/0378-4754(81)90003-3.
9. Shampine L.F. Solving ordinary differential equations for simulation // Mathematics and Computers in Simulation. – 1978. – Vol. 20, N 3. – P. 204–207.
10. Shampine L.F., Gladwell I., Brankin R.W. Reliable solution of special event location problems for ODEs // ACM Transactions on Mathematical Software. – 1991. – Vol. 17, N 1. – P. 11–25.
11. Park T., Barton P.I. State event location in differential-algebraic models // ACM Transactions on Modeling and Computer Simulation. – 1996. – Vol. 6, N 2. – P. 137–165.
12. Bahl V., Linninger A.A. Modeling of continuous-discrete processes // Hybrid Systems: Computation and Control, HSCC 2001. – Berlin; Heidelberg: Springer, 2001. – P. 387–402. – (Lecture Notes in Computer Science; vol. 2034).
13. Carver M.B. Efficient integration over discontinuities in ordinary differential equation simulations // Mathematics and Computers in Simulation. – 1978. – Vol. 20, N 3. – P. 190–196.
14. Mao G., Petzold L.R. Efficient integration over discontinuities for differential-algebraic systems // Computers and Mathematics with Applications. – 2002. – Vol. 43, N 1/2. – P. 65–79.
15. Esposito J.M., Kumar V. A state event detection algorithm for numerically simulating hybrid systems with model singularities // ACM Transactions on Modeling and Computer Simulation. – 2007. – Vol. 17, N 1. – P. 1–es.
16. Шорников Ю.В. Прикладное математическое, алгоритмическое и программное обеспечение компьютерного анализа гибридных систем: дис. ... д-ра техн. наук. – Новосибирск, 2009. – 313 с.
17. Веников В.А. Переходные электромеханические процессы в электрических системах. – М.: Высшая школа, 1985. – 536 с.
18. Urquía M.A., Martín-Villalba C. Modeling and simulation in Engineering using Modelica. – Madrid, Spain: UNED Editorial, 2018. – 298 p.
19. Cellier F.E., Greifeneder J. Continuous system modeling. – New York: Springer-Verlag, 1991. – 756 p.
20. Brenan K.E., Campbell S.L., Petzold L.R. Numerical solution of initial-value problems in differential-algebraic equations. – Philadelphia: Society for Industrial and Applied Mathematics, 1995. – 251 p. – URL: <https://epubs.siam.org/doi/book/10.1137/1.9781611971224> (accessed: 20.12.2020).
21. Pantelides C.C. SPEEDUP—recent advances in process simulation // Computers and Chemical Engineering. – 1988. – Vol. 12, N 7. – P. 745–755.
22. Шорников Ю.В., Бессонов А.В. Компоненты ядра программного комплекса «ИСМА 2015»: свидетельство о гос. регистрации программы для ЭВМ № 2015617235. – М., 2015.

Попов Евгений Александрович, ассистент кафедры автоматизированных систем управления Новосибирского государственного технического университета. Основное направление научных исследований – языки моделирования и алгоритмы численного анализа гибридных динамических систем. Имеет 27 публикаций. E-mail: e.porov@corp.nstu.ru

Шорников Юрий Владимирович, доктор технических наук, профессор, профессор кафедры автоматизированных систем управления Новосибирского государственного технического университета. Основное направление научных исследований – математическое, лингвистическое и программное обеспечение моделирования гибридных динамических систем. Имеет около 200 публикаций. E-mail: shornikov@inbox.ru

Popov Evgeny A., teaching assistant at the Automated Control Systems Department in Novosibirsk State Technical University. The main area of his research is modeling languages and simulation algorithms for hybrid dynamic systems. He has 27 publications. E-mail: e.popov@corp.nstu.ru

Shornikov Yuri V., D.Sc. (Eng.), professor, professor at the Automated Control Systems Department in Novosibirsk State Technical University. The main area of his research is mathematical and linguistic software for modeling and simulation of hybrid dynamic systems. He has about 200 publications. E-mail: shornikov@inbox.ru

DOI: 10.17212/1814-1196-2020-4-159-176

Detection of different type events in hybrid dynamical systems *

E.A. POPOV^a, YU.V. SHORNIKOV^b

Novosibirsk State Technical University, 20 K. Marx Prospekt, Novosibirsk, 630073, Russian Federation

^a *e.popov@corp.nstu.ru* ^b *shornikov@inbox.ru*

Abstract

Hybrid dynamical or simply hybrid systems (HS) are a modern apparatus for modeling discrete-continuous processes in different applications such as power engineering, aeronautics, manufacturing, economics, transport dispatching, etc. The key difference of HSs from classical dynamic systems is the presence of continuous mode switching events. Event times are defined by the zeroes of continuous event functions. If it is impossible to symbolically compute an event time. To do it one uses an event detection algorithm working together with a differential equation integration algorithm.

Events in HSs are traditionally divided into state events and time events. Only explicit time events with event functions in the form of linear polynomials in time are usually considered in the literature. This paper addresses the class of implicit time events and lists their possible sources. Moreover, the traditional classification of events into unilateral, bilateral and accuracy critical events is expanded by adding difficult-to-detect events. These events are characterized by event functions crossing zero several times within one integration step. Not all algorithms can guarantee detecting events of this type.

Heterogeneous HSs including processes of different physical nature are in general characterized by significantly stiff and high-dimensional modes usually defined in a form of differential-algebraic systems of equations with events of different types.

The last feature limits the application of classical event detection algorithms oriented to a single event type. That is why the paper proposes the methodology of complex event detection consisting in using separate event detection algorithm for each event type. The joint work of several algorithms can ensure correct detection of events of different types and also may improve the efficiency.

A complex event detection algorithm guaranteeing detection of all events is constructed for a particular HS. The complex algorithm demonstrates an average speed up of 17%.

* Received 02 September 2020.

Keywords: heterogeneous dynamical systems, hybrid dynamical systems, discrete-continuous processes, event functions, event types, event detection, simulation, computer-aided modeling and simulation

REFERENCES

1. Goebel R., Sanfelice R.G., Teel A.R. *Hybrid dynamical systems: modeling, stability, and robustness*. Princeton, Princeton University Press, 2012. 232 p.
2. Kolesov Yu.B., Senichenkov Yu.B. *Modelirovanie sistem. Dinamicheskie i gibridnye sistemy* [System modeling and simulation. Dynamical and hybrid systems]. St. Petersburg, BHV-Peterburg Publ., 2012. 224 p.
3. Novikov E.A., Shornikov Yu.V. *Komp'yuternoe modelirovanie zhestkikh gibridnykh sistem* [Computer modeling and simulation of stiff hybrid systems]. Novosibirsk, NSTU Publ., 2012. 451 p.
4. Gear C.W., Østerby O. Solving ordinary differential equations with discontinuities. *ACM Transactions on Mathematical Software*, 1984, vol. 10, no. 1, pp. 23–44. DOI: 10.1145/356068.356071.
5. O'Regan P.G. Step size adjustment at discontinuities for fourth order Runge-Kutta methods. *The Computer Journal*, 1970, vol. 13, no. 4, pp. 401–404. DOI: 10.1093/comjnl/13.4.401.
6. Hay J.L., Crosbie R.E., Chaplin R.I. Integration routines for systems with discontinuities. *The Computer Journal*, 1974, vol. 17, no. 3, pp. 275–278.
7. Cellier F.E. *Combined continuous/discrete system simulation by use of digital computers: techniques and tools. PhD Thesis*. Zürich, Switzerland, Swiss Federal Institute of Technology Zürich, 1979. 285 p.
8. Ellison D. Efficient automatic integration of ordinary differential equations with discontinuities. *Mathematics and Computers in Simulation (MATCOM)*, 1981, vol. 23, no. 1, pp. 12–20. DOI: 10.1016/0378-4754(81)90003-3.
9. Shampine L.F. Solving ordinary differential equations for simulation. *Mathematics and Computers in Simulation*, 1978, vol. 20, no. 3, pp. 204–207.
10. Shampine L.F., Gladwell I., Brankin R.W. Reliable solution of special event location problems for ODEs. *ACM Transactions on Mathematical Software*, 1991, vol. 17, no. 1, pp. 11–25.
11. Park T., Barton P.I. State event location in differential-algebraic models. *ACM Transactions on Modeling and Computer Simulation*, 1996, vol. 6, no. 2, pp. 137–165.
12. Bahl V., Linninger A.A. Modeling of continuous-discrete processes. *Hybrid Systems: Computation and Control, HSCC 2001*. Berlin, Heidelberg, Springer, 2001, pp. 387–402.
13. Carver M.B. Efficient integration over discontinuities in ordinary differential equation simulations. *Mathematics and Computers in Simulation*, 1978, vol. 20, no. 3, pp. 190–196.
14. Mao G., Petzold L.R. Efficient integration over discontinuities for differential-algebraic systems. *Computers and Mathematics with Applications*, 2002, vol. 43, no. 1/2, pp. 65–79.
15. Esposito J.M., Kumar V. A state event detection algorithm for numerically simulating hybrid systems with model singularities. *ACM Transactions on Modeling and Computer Simulation*, 2007, vol. 17, no. 1, pp. 1–es.
16. Shornikov Yu.V. *Prikladnoe matematicheskoe, algoritmicheskoe i programmnoe obespechenie kom-p'yuternogo analiza gibridnykh sistem*. Diss. dokt. tekhn. nauk [Applied mathematical software and algorithms for computer analysis of hybrid systems. Dr. eng. sci. diss.]. Novosibirsk, 2009. 313 p.
17. Venikov V.A. *Perekhodnye elektromekhanicheskie protsessy v elektricheskikh sistemakh* [Electromechanical transients in electric power systems]. Moscow, Vysshaya shkola Publ., 1985. 536 p.
18. Urquía M.A., Martín-Villalba C. *Modeling and simulation in Engineering using Modelica*. Madrid, Spain, UNED Editorial, 2018. 298 p.
19. Cellier F.E., Greifeneder J. *Continuous system modeling*. New York, Springer-Verlag, 1991. 756 p.

20. Brenan K.E., Campbell S.L., Petzold L.R. *Numerical solution of initial-value problems in differential-algebraic equations*. Philadelphia, Society for Industrial and Applied Mathematics, 1995. 251 p. Available at: <https://epubs.siam.org/doi/book/10.1137/1.9781611971224> (accessed 20.12.2020).

21. Pantelides C.C. SPEEDUP—recent advances in process simulation. *Computers and Chemical Engineering*, 1988, vol. 12, no. 7, pp. 745–755.

22. Shornikov Yu.V., Bessonov A.V. *Komponenty yadra programmnoy kompleksa "ISMA 2015"* [Core components of the software suite "ISMA 2015"]. The Certificate on official registration of the computer program. No.2015617235, 2015.

Для цитирования:

Попов Е.А., Шорников Ю.В. Детекция событий разного типа в гибридных динамических системах // Научный вестник НГТУ. – 2020. – № 4 (80). – С. 159–176. – DOI: 10.17212/1814-1196-2020-4-159-176.

For citation:

Popov E.A., Shornikov Yu.V. *Detektsiya sobytii raznogo tipa v gibridnykh dinamicheskikh sistemakh* [Detection of different type events in hybrid dynamical systems]. *Nauchnyi vestnik Novosibirskogo gosudarstvennogo tekhnicheskogo universiteta = Science bulletin of the Novosibirsk state technical university*, 2020, no. 4 (80), pp. 159–176. DOI: 10.17212/1814-1196-2020-4-159-176.