ПОСТРОЕНИЕ МОДЕЛИ АДАПТИВНОСТИ КИБЕРФИЗИЧЕСКИХ СИСТЕМ: ФУНКЦИОНИРОВАНИЕ И ДЕТЕКТИРОВАНИЕ

Фатин А. Д.[1]

DOI: 10.21681/2311-3456-2024-2-36-43

Цель исследования *– создание сегментной математической модели адаптивности киберфизических систем, позволяющей системе интеллектуально реагировать на аномальные события и реконфигурировать свою топологию для сохранения первичной функциональности, а также математическая формализация первичных задач, служащих базисом математической модели.*

Методы исследования: *метод декомпозиции производственных требований к модели адаптивности киберфизических с точки зрения информационной безопасности на конечные составляющие, математическая формализация и отображение конечных составляющих на множество решений.*

Результат: *в исследовании показано, что задача построения модели адаптивности киберфизических систем в должной мере может быть декомпозирована на пять взаимосвязанных задач, каждая из которых в качестве входных данных использует результаты решения предыдущей: описание функционирования системы, детектирование аномалий, кластеризация элементов системы, изоляция аномальных узлов и реконфигурация топологии. Используя многомерные временные ряды, адаптивный фильтр Калмана, графовые структуры и нейрогенетические сети в контексте первых двух задач, модель обеспечивает комплексный подход к мониторингу и управлению киберфизическими системами, способными адаптироваться к изменениям и поддерживать работоспособность конечных систем. Дополнительно приводятся возможные альтернативы функций расстояния в пространстве решений для оптимизаций затрачиваемых вычислительных ресурсов при построении конечного решения и рекомендации по построению структур данных – для учета гетерогенности узлов, включаемых в конечные системы.*

Научная новизна *заключается в использовании новых средств оптимизации построения конечных нейронных сетей предсказания состояния систем за счет применения эволюционных свойств генетических алгоритмов на топологию первичного субстрата нейроэволюционных сетей решений, а также декомпозиции и полной формализации прикладной задачи построения модели средствами статистических, графовых и временных механизмов с их полной интеграцией.*

Ключевые слова: *киберфизические системы, многомерные временные ряды, аномалии, фильтр Калмана, адаптивность, графовые структуры, нейронные сети.*

BUILDING A MODEL OF ADAPTABILITY OF CYBERPHYSICAL SYSTEMS: OPERATION AND DETECTION

Fatin A. D.[2]

The purpose of the study *is to create a segmented mathematical model of the adaptability of cyber-physical systems, allowing the system to intelligently respond to anomalous events and reconfigure its topology to preserve primary functionality, as well as the mathematical formalization of primary tasks that serve as the basis of the mathematical model.*

Research methods: *method of decomposition of production requirements for the cyber-physical adaptability model from the point of view of information security into final components, mathematical formalization and mapping of final components to a set of solutions.*

Result: *the study shows that the task of constructing a model of adaptability of cyber-physical systems can be properly decomposed into five interrelated tasks, each of which uses the results of solving the previous*

1 Фатин Александр Денисович, аспирант Института компьютерных наук и кибербезопасности, Санкт-Петербургский политехнический университет Петра Великого (СПбПУ), Санкт-Петербург, Россия. ORCID: 0000-0001-6225-264X. E-mail: sasha-fatin@mail.ru

2 Alexander D. Fatin, postgraduate at the Institute of Computer Science and Cybersecurity, Peter the Great St. Petersburg Polytechnic University (SPbPU), Saint-Petersburg, Russia. E-mail: sasha-fatin@mail.ru

one as input data: describing the functioning of the system, detecting anomalies, clustering system elements, isolating anomalous nodes and topology reconfiguration. Using multivariate time series, adaptive Kalman filter, graph structures and neurogenetic networks in the context of the first two tasks, the model provides an integrated approach to monitoring and managing cyber-physical systems that can adapt to changes and maintain the performance of end systems. Additionally, possible alternatives to distance functions in the solution space are provided to optimize the computational resources spent when constructing the final solution and recommendations for constructing data structures to take into account the heterogeneity of nodes included in the final systems.

The novelty of the research *consists in the use of new means of optimizing the construction of finite neural networks for predicting the state of systems through the use of evolutionary properties of genetic algorithms on the topology of the primary substrate of neuroevolutionary decision networks, as well as decomposition and complete formalization of the applied problem of constructing a model using statistical, graph and temporal mechanisms with their full integration.*

Keywords: *cyberphysical systems, multidimensional time series, anomalies, Kalman filter, adaptability, graph structures, neural networks.*

Введение

Данная работа посвящена теоретическому базису составления и реализации математической модели адаптивности киберфизических систем. Под адаптивностью киберфизических систем в данном случае понимается возможность киберфизической системы интеллектуально реагировать на внештатное (аномальное) поведение и производить реконфигурацию своей топологии с сохранением полного или практически полного набора своих функциональных возможностей.

Задачу построения и реализации описанной выше модели можно разделить на 5 связанных задач, причем исходные данные каждой последующей задачи берут начало из решения предыдущей задачи:

1. Описательная составляющая:

Необходимо каким-либо образом описывать функционирование киберфизической системы в реальном времени, т.е. должна существовать описательная модель работы системы на физическом уровне с помощью данных, циркулирующих в системе.

2. Детектирующая составляющая:

Система должна иметь возможность самостоятельно детектировать аномалии внутри себя (атаки, сильные отклонения от нормального состояния работы и т.д.) [1].

3. Кластерная составляющая:

В случае обнаружения аномалий система должна иметь возможность произвести кластеризацию своих элементов, т.е. выявить, на какие сегменты можно разбить себя (систему) так, чтобы с минимальными потерями можно было изолировать атакованные, зараженные или выведенные из строя узлы системы.

4. Изолирующая составляющая:

Система должна иметь возможность выполнить непосредственную изоляцию аномальных узлов и/или узлов, функционирующих в не нормальном режиме.

5. Реконфигурирующая составляющая:

Система должна иметь возможность выполнить реконфигурацию своей топологии и/или произвести перенастройку своей функциональности с учетом изолированных узлов так, чтобы максимально сохранить свое целевое назначение.

Задачу 1, т.е. описание функционирования киберфизической системы в реальном времени, обычно решают с помощью следующих методов:

1. Многомерные временные ряды [2].
2. Адаптивный алгоритм фильтра Калмана [3].
3. Дискретное вейвлет-преобразование [4].
4. Фрактальное представление топологии системы [5].
5. Графовые структуры разных видов (Классические графы; Динамические графы; Событийные графы; Сигнальные графы, Графы классификации данных и т.д.) [6–8].

Задачу 2, т.е. детектирование аномалий в системе, обычно решают с помощью следующих методов:

1. Оценка критериев самоподобия системы.
2. Предсказание состояния системы на основе статистических инструментов и фрактального анализа [9]:
   - 2.1. Поиск точек разладки на основе Байесовского онлайн алгоритма или наивного Байесовского классификатора [10].
   - 2.2. Использованием коэффициента множественной корреляции.
3. Предсказание состояния системы на основе машинного обучения [11–13].
4. Предсказание состояния системы с помощью нейрогенетических алгоритмов [14].

Задачу 3, т.е. выявление, на какие сегменты можно разбить систему так, чтобы с минимальными потерями можно было изолировать атакованные,

зараженные или выведенные из строя узлы, обычно решают с помощью:

1. ANCA: Алгоритм кластеризации распределенной сети.
2. Алгоритм кластеризации сети, основанный на быстром обнаружении центрального узла [15].
3. Кластеризация сети для обнаружения латентных состояний и точек изменения [16].
4. Вариационное обучение с совместным встраиванием для кластеризации распределенных сетей.
5. Тензорное разложение для кластеризации многослойных сетей.
6. Кластеризация сетевых структур на основе алгоритма пчелиной колонии (ABC).

Задача 4, т.е. изоляция узлов, обычно не представляет теоретической сложности и на практике реализуется отключением узлов, реконфигурацией потоков данных или иными тривиальными методами. Основные сложности могут возникать со следующими моментами:

1. Определение порядка изоляции узлов на основе их важности для функционирования системы и степени риска.
2. Минимизация воздействия изоляции на работу оставшейся части системы.

Оба этих момента обычно учитываются в решении задачи 3.

Задача 5, т.е. реконфигурация системы с максимальным возможным сохранением её функционала, обычно под собой подразумевает:

1. Определение новой сетевой топологии без изолирования узлов.
2. Разработку алгоритма перераспределения задач и потоков данных.
3. Адаптацию самой системы к изменениям условия работы для обеспечения её устойчивости и функциональности.

Формализация общей задачи с помощью математической модели

Таким образом, для каждой из задач можно определить следующую математическую формализацию:

1. Функционирование системы:

$$X(t) = F(t, X(t-1), U(t), \varepsilon(t)), \quad (1)$$

где $X(t)$ – состояние системы в момент времени t, $U(t)$ – управляющее воздействие, $\varepsilon(t)$ – случайные внешние факторы, F – функция, описывающая динамику системы.

2. Детектирование аномалий:

$$A(t) = G(X(t), H(t)), \quad (2)$$

где $A(t)$ – индикатор аномалии, $H(t)$ – исторические данные, G – функция детектирования аномалий.

3. Кластеризация:

$$C = K(X, \theta), \quad (3)$$

где C – результат кластеризации, X – данные о состоянии системы, θ – параметры алгоритма кластеризации, K – функция кластеризации системы.

4. Изоляция узлов:

$$I = L(C, \phi), \quad (4)$$

где I – план изоляции узлов, ϕ – параметры, влияющие на выбор узлов для изоляции, C – результат кластеризации, L – функция составления плана изоляции узлов системы.

5. Реконфигурация:

$$R = M(X', \psi), \quad (5)$$

где R – новая конфигурация системы, X' – состояние системы после изоляции узлов, ψ – параметры алгоритма реконфигурации, M – функция реконфигурации топологии системы.

Данная формализация представляет собой высокоуровневое описание решения общей задачи. Для каждой конкретной системы и задачи потребуется детализация и адаптация моделей и алгоритмов.

Решение задачи 1: функционирование системы

Рассмотрим решение задачи 1. В качестве методов решения используем многомерные временные ряды, адаптивный алгоритм фильтра Калмана и графовые структуры для создания комплексной модели (пункты 1, 2, 5 из соответственной задачи 1 во Введении).

Многомерные временные ряды

Пусть система характеризуется набором измеряемых параметров, каждый из которых можно представить в виде временного ряда. Обозначим вектор состояния системы в момент времени t как

$$x(t) \in Rn, \quad (6)$$

где n – количество измеряемых параметров.

Модель временного ряда для каждого параметра может быть представлена как:

$$x(t) = F(t)x(t-1) + B(t)u(t) + w(t), \quad (7)$$

где: $F(t)$ – матрица перехода состояния, которая описывает динамику системы. $B(t)$ – матрица управления, которая связывает управляющий вектор $u(t)$ с состоянием системы. $w(t)$ – вектор шума процесса, предполагается, что он имеет нормальное распределение с нулевым средним и ковариационной матрицей $Q(t)$.

Адаптивный алгоритм фильтра Калмана

Фильтр Калмана используется для оценки состояния системы в реальном времени, минимизируя влияние шума измерений. Он состоит из двух этапов: прогнозирование и коррекция.

1. Прогнозирование:

$$\hat{x}(t|t-1) = F(t)\,\hat{x}(t-1|t-1) + B(t)u(t) \quad (8)$$

$$P(t|t-1) = F(t)P(t-1|t-1)\,F(t)^{\tau} + Q(t) \quad (9)$$

где $\hat{x}(t|t-1)$ – априорная оценка состояния системы в момент времени t, а $P(t|t-1)$ – априорная ковариационная матрица ошибки оценки.

2. Коррекция:

$$K(t) = P(t|t-1)\,H(t)^{\tau}\,[H(t)P(t|t-1)\,H(t)^{\tau} + R(t)]^{-1} \quad (10)$$

$$\hat{x}(t|t) = \hat{x}(t|t-1) + K(t)[z(t) - H(t)\,\hat{x}(t|t-1)] \quad (11)$$

$$P(t|t-1) = [I - K(t)H(t)]P(t|t-1)] \quad (12)$$

где: $K(t)$ – матрица усиления Калмана. $H(t)$ – матрица измерения, которая связывает истинное состояние системы с измерениями. $z(t)$ – вектор измерений в момент времени t. $R(t)$ – ковариационная матрица шума измерений. I – единичная матрица соответствующего размера.

Адаптивность фильтра Калмана заключается в динамическом обновлении матриц $F(t)$, $B(t)$, $Q(t)$, $H(t)$ и $R(t)$ на основе входных данных, что позволяет более точно отслеживать изменения в системе.

Графовые структуры

Графовые структуры используются для представления взаимосвязей между различными компонентами системы. Каждый узел графа соответствует компоненту системы, а рёбра отражают связи между ними. Для динамической системы граф может быть представлен как:

$$G(t) = (V(t),E(t)), \quad (13)$$

где $V(t)$ – множество узлов, а $E(t)$ – множество рёбер в момент времени t. Атрибуты узлов и рёбер могут включать информацию о состоянии и динамике соответствующих компонентов и связей.

Интеграция модели

Интеграция всех трёх компонентов в единую модель позволит получить комплексное представление о состоянии и поведении киберфизической системы в реальном времени. Матрицы фильтра Калмана могут быть адаптированы на основе структуры графа, что позволит отображать изменения в топологии системы. Временные ряды будут обновляться с использованием алгоритма Калмана для предоставления текущей оценки состояния системы.

Данная интегрированная модель может быть использована как основа для дальнейшего обнаружения аномалий, кластеризации системы, изоляции узлов и реконфигурации системы.

Стоит отметить тот факт, что интеграция многомерных временных рядов, адаптивного алгоритма фильтра Калмана и графовых структур в единую математическую модель требует синхронизации данных и алгоритмов. Их взаимодействие представляется следующим образом:

1. Определение состояния системы через временные ряды и графы

Данный шаг начинается с создания временного ряда для каждого измеряемого параметра системы и представляет систему в виде графа $G(t) = (V(t),V(t))$, где $V(t)$ – узлы, соответствующие компонентам системы, и $V(t)$ – рёбра, отражающие связи между компонентами.

2. Связь временных рядов и графовой структуры

Каждому узлу графа $vi \in V(t)$ ассоциируем вектор состояния $xi(t)$, который представляет собой многомерный временной ряд. Таким образом, состояние всей системы в момент времени t может быть представлено как совокупность векторов состояний всех узлов:

$$X(t) = x1(t),x2(t),...,xn(t) \quad (14)$$

где n – количество узлов в графе.

3. Применение фильтра Калмана

Фильтр Калмана применяется к каждому временному ряду индивидуально, но с учетом структуры графа. Для узла vi и его вектора состояния $xi(t)$, фильтр Калмана обновляется следующим образом:

3.1. Прогноз:

$$\hat{x}i(t|t-1) = Fi(t)\,\hat{x}i(t|t-1) + Bi(t)ui(t) \quad (15)$$

$$Pi(t|t-1) = Fi(t)\,Pi(t-1|t-1)\,Fi(t)^{\tau} + Qi(t) \quad (16)$$

3.2. Коррекция:

$$Ki(t) = Pi(t|t-1)Hi(t)^{\tau}[Hi(t)Pi(t|t-1)Hi(t)^{\tau} + Ri(t)]^{-1} \quad (17)$$

$$\hat{x}i(t|t) = \hat{x}i(t|t-1) + Ki(t)[zi(t) - Hi(t)\,\hat{x}i(t|t-1)] \quad (18)$$

$$Pi(t|t-1) = [I - Ki(t)Hi(t)]Pi(t|t-1)] \quad (19)$$

4. Адаптация и обновление

Адаптация матриц $Fi(t)$, $Bi(t)$, $Qi(t)$, $Hi(t)$ и $Ri(t)$ происходит на основе динамических изменений в графе и данных временных рядов. Взаимодействие между компонентами системы (узлами графа) может приводить к изменению матриц перехода и управления для отдельных узлов.

5. Финальный результат

Финальный результат модели — это набор оценок состояний всех узлов в системе $\hat{X}i(t|t)$, который обновляется в реальном времени. Данная информация может быть использована для мониторинга системы, а также для дальнейшего обнаружения аномалий, кластеризации системы, изоляции узлов и реконфигурации системы.

Первичная математическая модель

Первичная модель интегрирует временные ряды и графовую структуру с фильтром Калмана следующим образом:

Система представлена графом $G(t) = (V(t),E(t))$ с динамически изменяющимися узлами и связями.

Каждый узел vi имеет связанный с ним многомерный временной ряд $xi(t)$, который представляет его состояние.

Фильтр Калмана применяется к каждому узлу для обновления его состояния с учетом взаимодействия с другими узлами.

Адаптивность фильтра обеспечивается путем обновления его параметров на основе изменений в графовой структуре и данных временных рядов.

В результате получаем непрерывно обновляемую оценку состояния системы, которая может быть использована для последующего анализа и принятия решений.

Вторичная математическая модель

Топология сети подразумевает, что узлы могут иметь разные данные, меняющиеся в реальном времени, а также разное количество соединений друг с другом. Таким образом, для отображения топологии и данных разных узлов в многомерные временные ряды, можно использовать следующий подход:

1. *Представление топологии и данных узлов:*

1.1. *Вектор состояния узла:* для каждого узла vi создайте вектор $xi(t)$, который включает все данные узла, изменяющиеся во времени. Это могут быть различные параметры, такие как загрузка CPU, использование памяти, пропускная способность сети и т.д.

1.2. *Вектор связности узла:* для каждого узла vi создайте вектор $ci(t)$, который представляет связи узла с другими узлами в определенный момент времени t. Элементы вектора могут быть бинарными (1, если есть связь, и 0, если связи нет) или взвешенными (например, оценка пропускной способности или задержки связи).

1.3. *Многомерный временной ряд:* объединение векторов состояния и векторов связности всех узлов в один большой вектор $X(t)$ для всей системы в момент времени t. Таким образом получим многомерный временной ряд, который отражает состояние системы и её топологию.

2. О*бновление временных рядов:*

2.1. *Динамическое обновление:* необходимо в реальном времени обновлять векторы состояния $xi(t)$ и векторы связности $ci(t)$ для каждого узла в соответствии с изменениями в данных и топологии.

3. *Пример структуры данных:* для системы из N узлов, многомерный временной ряд $X(t)$ в момент времени t может выглядеть следующим образом:

$$X(t) = [x1(t)\ c1(t) \vdots xN(t)\ cN(t)] \quad (20)$$

где $xi(t)$ и $ci(t)$ представляют состояние и связи i-го узла соответственно.

Итоговая математическая модель

Для упрощения работы с векторами и сохранения модульности изменим формулу выше следующим образом:

Для каждого узла vi в системе создаём объединённый вектор $zi(t)$, который включает в себя как вектор состояния узла $xi(t)$, так и вектор связности $ci(t)$:

$$zi(t) = [xi(t)\ ci(t)] \quad (21)$$

Затем, для формирования многомерного временного ряда $X(t)$ для всей системы, объединяем все индивидуальные объединённые векторы $zi(t)$:

$$X(t) = [z1(t) \vdots zN(t)] \quad (22)$$

где каждый $zi(t)$ представляет собой полную информацию о состоянии и связности i-го узла в момент времени t, а N — это общее количество узлов в системе.

Таким образом, многомерный временной ряд $X(t)$ отражает полную информацию о состояниях всех узлов и их связях в системе в каждый момент времени.

Решение задачи 2: детектирование аномалий

Для решения задачи 2 будет использоваться нейрогенетический алгоритм (пункт 4 из соответственной задачи 2 во Введении), который сочетает в себе нейронные сети и генетические алгоритмы для прогнозирования состояния системы и обнаружения аномалий. Нейронная сеть будет обучаться прогнозировать будущее состояние системы на основе текущих и прошлых данных, в то время как генетический алгоритм будет использоваться для оптимизации параметров и структуры нейронной сети.

Взаимосвязь решения задачи 1 с задачей 2

1. *Сбор данных*

Используем данные о состоянии системы, полученные в задаче 1. Эти данные включают в себя временные ряды каждого параметра системы и их оценки, полученные с помощью фильтра Калмана.

2. *Подготовка данных*

Данные из задачи 1 представляют собой временные ряды, которые могут быть использованы непосредственно как входные данные для нейронной сети. Сформируем обучающий набор данных, где входы – это последовательности измерений до момента времени t, а выход – это состояние системы в момент $t+1$.

3. *Нейронная сеть*

Разработаем нейронную сеть, которая будет принимать вектора состояний системы в качестве входных данных и выдавать прогноз состояния на следующий шаг времени. Нейронная сеть может быть

рекуррентной (например, LSTM или GRU), что может позволить ей эффективнее обрабатывать временные зависимости в данных.

4. *Генетический алгоритм*

Генетический алгоритм будет использоваться для оптимизации структуры и весов нейронной сети. Популяция кандидатов (нейронных сетей) будет оцениваться на основе их способности точно прогнозировать будущее состояние системы. Операции генетического алгоритма, включая отбор, кроссовер и мутацию, будут применяться для создания новых поколений кандидатов.

5. *Обучение и валидация*

Нейронная сеть будет обучаться на исторических данных, а генетический алгоритм будет использоваться для улучшения её параметров и структуры. После обучения модель будет проверяться на валидационном наборе данных для оценки её способности прогнозировать будущее состояние системы.

6. *Детектирование аномалий*

После обучения модель будет применяться в реальном времени для прогнозирования следующего состояния системы. Реальное состояние системы, полученное из задачи 1, будет сравниваться с прогнозируемым. Если расхождение между прогнозируемым и реальным состоянием превышает заранее определённый порог, система регистрирует аномалию.

7. *Реакция на аномалии*

При обнаружении аномалии система может предпринять ряд действий, включая уведомление операторов, автоматическую изоляцию подозрительных узлов и инициацию процедур восстановления.

Математическая модель

Пусть $X(t)$ – вектор состояния системы в момент времени t, полученный в задаче 1. Нейрогенетическая модель M прогнозирует состояние $\hat{X}(t+1)$ на следующем шаге:

$$\hat{X}(t+1) = M(X(t), X(t-1), \ldots, X(t-n), \Theta), \quad (23)$$

где: M – нейрогенетическая модель. Θ – параметры модели, оптимизированные генетическим алгоритмом. n – размер окна исторических данных.

Аномалия детектируется, если:

$$d(X(t+1), \hat{X}(t+1)) > \tau \quad (24)$$

где: d – функция расстояния (например, Евклидово расстояние). τ – пороговое значение для детектирования аномалии.

Этот подход позволяет не только обнаруживать аномалии в системе, но и обеспечивает основу для адаптивного управления и оптимизации параметров системы в реальном времени.

Альтернатива функции расстояния

Для сравнения предсказанных значений с реальными можно использовать также несколько иных подходов:

1. *Среднеквадратичная ошибка (MSE):*

Показывает среднюю квадратичную разницу между предсказанными и реальными значениями.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Yi,\text{предсказанное} - Yi,\text{реальное})^2 \quad (25)$$

2. *Средняя абсолютная ошибка (MAE):*

Показывает среднюю абсолютную разницу между предсказанными и реальными значениями.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|Yi,\text{предсказанное} - Yi,\text{реальное}|^2 \quad (26)$$

3. Коэффициент детерминации (R^2):

Показывает, какая доля вариативности реальных данных объясняется моделью.

Определение порогового значения

Определение порогового значения τ для детектирования аномалий является ключевым шагом в процессе мониторинга состояния киберфизических систем. Порог должен быть установлен таким образом, чтобы минимизировать количество ложных срабатываний (ложноположительные результаты) и пропусков реальных аномалий (ложноотрицательные результаты). Обычно используют один из следующих методов определения порога:

1. *Статистический подход:*

1.1. *Стандартное отклонение:* если предполагается, что нормальное поведение системы распределено нормально, порог можно установить на уровне $\mu \pm k\sigma$, где μ – среднее значение прогнозируемых ошибок, σ – стандартное отклонение, а k – количество стандартных отклонений, которое обычно находится в диапазоне от 2 до 3 для большинства случаев.

1.2. *Межквартильный размах (IQR)*: для непараметрического подхода порог может быть установлен с использованием IQR, где аномалиями считаются точки за пределами $Q1 - k \times IQR$ и $Q3 + k \times IQR$, где $Q1$ и $Q3$ – первый и третий квартили, соответственно, а k обычно равно 1.5.

2. *Машинное обучение:*

2.1. *Кросс-валидация:* кросс-валидация используется на обучающем наборе данных для определения порога, который минимизирует ошибки первого и второго рода.

2.2. *ROC-кривая и AUC:* необходимо произвести оценку модели на валидационном наборе данных и построение ROC-кривой, после чего выбрать порог, который максимизирует площадь под ROC-кривой (AUC) в разумных пределах.

3. *Оценка рисков:*

3.1. *Оценка воздействия:* необходимо установить порог, исходя из приемлемого уровня риска и потенциального воздействия аномалии на систему. Например, если последствия аномалии могут быть критическими, порог следует установить ниже для более раннего обнаружения.

3.2. *Стоимость ошибок:* если стоимость ложноположительных и ложноотрицательных результатов различна, можно использовать методы оптимизации для минимизации ожидаемых затрат.

4. *Практические методы:*

4.1. *Экспертная оценка:* в некоторых случаях порог может быть установлен на основе знаний экспертов, которые знакомы с конкретной системой и понимают её поведение.

4.2. *Итеративная настройка:* следует начать с некоторого предполагаемого порога, затем итеративно изменяя его, наблюдая за системой в реальном времени и корректируя порог в соответствии с полученными результатами.

Определение порога для детектирования аномалий часто требует комбинации вышеупомянутых подходов и итеративного процесса тестирования и валидации. Это включает в себя непрерывный мониторинг и обновление порога в соответствии с изменениями в поведении системы и её эксплуатационной среде.

Вывод

В данной работе была разработана и теоретически обоснована комплексная математическая модель адаптивности киберфизических систем (решены первые 2 из 5 задач), предназначенная для интеллектуального реагирования на аномальные события и эффективной реконфигурации системной топологии с целью поддержания её функциональности. Модель включает в себя пять взаимосвязанных этапов, охватывающих описание функционирования системы, детектирование аномалий, кластеризацию, изоляцию узлов и реконфигурацию топологии, каждый из которых базируется на результатах предыдущего этапа.

Использование многомерных временных рядов, адаптивного фильтра Калмана и графовых структур позволило создать динамичную модель, способную анализировать и прогнозировать состояние системы в реальном времени. Данный подход обеспечивает возможность оперативного обнаружения и реагирования на потенциальные угрозы и нарушения, минимизируя риски и последствия аномальных событий.

Внедрение предложенной модели в киберфизические системы обещает повышение их устойчивости и адаптивности, что является критически важным для обеспечения их безопасной и надежной работы в условиях постоянно меняющейся внешней среды и внутренних параметров. На основе полученных данных и результатов исследования были сформулированы рекомендации по реализации модели, в том числе предложен подход к определению оптимальной архитектуры нейронных сетей, что позволяет сохранить связность данных и обеспечить комплексный анализ состояния системы.

Перспективы дальнейших исследований включают детализацию и адаптацию модели под конкретные типы киберфизических систем, экспериментальную проверку и валидацию предложенных методов на практике, а также разработку программных и аппаратных средств для внедрения модели в реальные системы. Следующий этап работы предполагает более глубокое изучение и интеграцию третьей, четвертой и пятой задач, что позволит разработать полноценную систему адаптивного управления киберфизическими системами.

Научный руководитель: *Павленко Евгений Юрьевич, кандидат технических наук, доцент Высшей школы кибербезопасности Института комьютерных наук и кибербезопасности, Санкт-Петербургский политехнический университет Петра Великого (СПбПУ), Санкт-Петербург, Россия. ORCID: 0000-0003-1345-1874. E-mail: pavlenko_eyu@spbstu.ru*

Литература

1. *Подсистема предупреждения компьютерных атак на объекты критической информационной инфраструктуры: анализ функционирования и реализации / Котенко И. В. и др. // Вопросы кибербезопасности. – 2023. – № 1 (53). – С. 13–27. DOI: 10.21681/2311-3456-2023-1-13-27*
2. *Семенов В. В. Метод мониторинга состояния элементов киберфизических систем на основе анализа временных рядов / В. В. Семенов // Научно-технический вестник информационных технологий, механики и оптики. – 2022. – №6. – С. 1150–1158.*
3. *Лаврова Д. С. Прогнозирование состояния компонентов интеллектуальных сетей энергоснабжения smart grid для раннего обнаружения кибератак / Д. С. Лаврова // Проблемы информационной безопасности. Компьютерные системы. – 2019. – № 4. – С. 101–104*
4. *Коршунов Г. И. Моделирование физических сред для оптимизации цифрового управления в киберфизических системах / Г. И. Коршунов // НиКСС. – 2023. – №1 (41). – С. 23–27.*
5. *Бурый А. С., Ловцов Д. А. Информационные структуры умного города на основе киберфизических систем / А. С. Бурый, Д. А. Ловцов // Правовая информатика. – 2022. – №4. – С. 15–26. DOI: 10.21681/1994-104-2022-4-15-26*
6. *Фатин А. Д., Павленко Е. Ю. Анализ моделей представления киберфизических систем в задачах обеспечения информационной безопасности / А. Д. Фатин, Е. Ю. Павленко // Проблемы информационной безопасности. Компьютерные системы. 2020. – № 2. – С. 109–121.*

7. *Лаврова Д. С. Моделирование сетевой инфраструктуры сложных объектов для решения задачи противодействия кибератакам / Д. С. Лаврова, Д. П. Зегжда, Е. А Зайцева // Вопросы кибербезопасности. – 2019. – № 2. – С. 13–20. DOI: 10.21681/2311-3456-2019-2-13-20*
8. *Оценивание защищенности информационных систем на основе графовой модели эксплойтов / Федорченко Е. В. и др. // Вопросы кибербезопасности. – 2023. – № 3 (55). – С. 23-36. DOI: 10.21681/2311-3456-2023-3-23-26*
9. *Метод раннего обнаружения кибератак на основе интеграции фрактального анализа и статистических методов / Котенко И. В. и др. // Первая миля. – 2021. – № 6 (98). – С. 64–71. DOI: 10.22184/2070-8963.2021.98.6.64.70.*
10. *Югай П. Э., Жуковский Е. В., Семенов П. О. Особенности обнаружения вредоносных установочных файлов с использованием алгоритмов машинного обучения / П. Э. Югай, Е. В. Жуковский, П. О. Семенов // Проблемы информационной безопасности. Компьютерные системы. – 2023. – №2 (54). – С. 37–46.*
11. *Сергадеева А. И., Лаврова Д. С. Применение модульной нейронной сети для обнаружения DDoS-атак / А. И. Сергадеева, Д. С. Лаврова // Проблемы информационной безопасности. Компьютерные системы. – 2023. – №1 (53). – С. 111–118.*
12. *Выявление вредоносных исполняемых файлов на основе статико-динамического анализа с использованием машинного обучения / Огнев Р. А. и др. // Проблемы информационной безопасности. Компьютерные системы. – 2021. – №4. – С. 9–25.*
13. *Павлычев А. В., Стародубов М. И., Галимов А. Д. Использование алгоритма машинного обучения RANDOM FOREST для выявления сложных компьютерных инцидентов / А. В Павлычев., М. И. Стародубов, А. Д. Галимов // Вопросы кибербезопасности. 2022. – № 5 (51). – С. 74–81.*
14. *Neuroevolutionary Approach to Ensuring the Security of Cyber-Physical Systems / Fatin A., Pavlenko E., Zegzhda P. // Cyber-Physical Systems and Control II. Lecture Notes in Networks and Systems. – Vol 460. – pp. 441–450. DOI: 10.1007/978-3-031-20875-1_40.*
15. *Ziruo J., Fuqiang Q. Network Clustering Algorithm Based on Fast Detection of Central Node / J. Ziruo, Q. Fuqiang // Scientific Programming. – 2022. – pp 1–5. DOI: 10.1155/2022/4905190.*
16. *Network Clustering for Latent State and Changepoint Detection / Madeline Navarro et al. // arXiv – CS – Social and Information Networks, 2021. DOI: arxiv-2111.01273*