УДК 621.391

Обзор графо-аналитических подходов к мониторингу информационно-телекоммуникационных сетей и их применение для выявления аномальных состояний

Будко Н. П., Васильев Н. В.

Постановка задачи: *современные подходы к мониторингу телекоммуникационных сетей ориентированы на измерение состояния отдельных устройств и сервисов, оставляя за кадром состояние сети «в целом», как единого объекта мониторинга. Как следствие, для достаточно больших телекоммуникационных сетей, особенно в случае, когда сетевой элемент присутствует в измерении ничтожное по сравнению с периодом существования всей сети время, проанализировать результаты мониторинга становится невозможным. Несмотря на все большее распространение в системах мониторинга методов многомерного анализа данных, задача наглядного представления «здоровья сети» является более чем актуальной. Тем более, что в отличие от многомерных кубов данных, интерпретация и анализ которых сродни искусству, методы на основе редакционного расстояния графов позволяют наглядно визуализировать динамику не только всей сети, но и ее отдельного фрагмента, что требует меньшей квалификации персонала, эксплуатирующего телекоммуникационные системы. Отсутствие методов идентификации состояния в зависимости от степени изменения топологии долгое время затрудняло практическое применение методов на основе расстояния редактирования графов. Однако развитие теоретического базиса, в частности доказательство результатов в области средних графов последовательности и методов кластеризации позволяет снять требование предварительной нормировки меры изменения топологии, заменив его понятием «кластера состояния».* ***Целью работы*** *является разработка на основе научно-методического аппарата оценки последовательностей графов методики выявления аномальных состояний телекоммуникационной сети посредством анализа степени изменения топологии объекта мониторинга.* ***Используемые методы:*** *методы вычислительной теории графов; методы анализа данных (кластерного анализа); линейная алгебра и спектральная теория графов; методы поведенческой аналитики; технологии сетевого мониторинга, как набор инженерных практик, поддерживающих надежную и безотказную работу приложений в настоящем и будущем; Operation Support Systems, как технология поддержки операций; методы системного анализа, структурного синтеза, теории прогноза, теории диагностики, теории классификации.* ***Новизна работы****: новизна исследования определяется полнотой проведенного анализа существующих графо-аналитических подходов и использованием теоретических результатов по усреднению последовательностей графов в рамках алгоритма k-средних для формирования адаптивной методики классификации оценки состояния телекоммуникационной сети.* ***Результат:*** *в работе на основе предварительно проведенного анализа результатов в области мониторинга графов телекоммуникационных сетей, а также методов кластеризации графов предложен подход к выявлению аномальных состояний телекоммуникационной сети. Предложенный подход апробирован на реальных данных ботнет-атаки на телекоммуникационную сеть и показывает довольно четкую идентификацию периодов нахождения сети в различных состояниях, таких как: атака, нормальный и переходный режимы.*

Ключевые слова: *расстояние редактирования графа, средний граф, спектр графа, кластеризация графов, информационно-телекоммуникационная сеть, подсистема сетевого мониторинга.*

Библиографическая ссылка на статью:
Будко Н. П., Васильев Н. В. Обзор графо-аналитических подходов к мониторингу информационно-телекоммуникационных сетей и их применение для выявления аномальных состояний // Системы управления, связи и безопасности. 2021. № 6. С. 53-75. DOI: 10.24412/2410-9916-2021-6-53-75

Reference for citation:
Budko N. P., Vasiliev N. V. Review of graph-analytical approaches to monitoring of information and telecommunication networks and their application to identify abnormal states. *Systems of Control, Communication and Security*, 2021, no. 6, pp. 53-75 (in Russian). DOI: 10.24412/2410-9916-2021-6-53-75

Введение

Анализ современных подсистем функционального контроля информационно-телекоммуникационных сетей (ИТКС) общего пользования (ОП) показывает доминирование реализации парадигмы систем мониторинга уровня сетевых элементов (в концепции Telecommunication Management Notation (TMN)). Реализация подобного подхода можно увидеть в системах мониторинга Zabbix, Nagios, HP OpenView и др. В данных программных комплексах каждый цикл опроса сетевого элемента сводится к выполнению проверки его доступности, опросу структурных (маршрутные таблицы) и динамических (уровни загрузки ресурсов, частота ошибок) характеристик. В этой концепции сетевое управление рассматривает поведение сетевых элементов во взаимосвязи как часть функции мониторинга отказов (fault management) в задачах корреляции и фильтрации неисправностей (event correlation). Как известно, это предполагает использование априорно заданных моделей, описывающих взаимное влияние устройств.

Наиболее часто для решения данной задачи используются модели на основе правил (rule-based evet correlation), описывающих взаимное влияние в виде «Если вышло из строя устройство A, то B и C будут недоступны». В случае выхода из строя A, в журнале проверок элементов будет наблюдаться три события: A, B, C. Применив данное правило, мы сможем определить, что первопричиной отказа является именно A. Составление подобного множества правил корреляции требует от обслуживающего персонала досконального знания сети, что в случае большого ее размера представляет собой достаточно сложную задачу. К тому же любые структурные изменения в сети будут требовать изменения множества правил.

Очевидной формой представления ИТКС ОП является граф. Узлы сети (которыми могут быть группы пользователей или отдельные клиенты и серверы) представляются вершинами графа (множество V), а дуги графа (множество E) представляют логические связи (например, направления связи или маршруты передачи данных) между узлами.

В этом случае состояние ИТКС в каждый момент времени будет описываться некоторым графом, а динамика ее изменения будет сводиться к оценке изменения множеств вершин и ребер (граф-измерения).

Цель статьи: провести обзор моделей и методов мониторинга, ориентированных на граф-измерения. Некоторые из описанных методов проиллюстрированы исследованиями, направленными на выявление аномальных состояний ИТКС ОП, которые были проведены авторами на наборе данных реальной ботнет-атаки на телекоммуникационную сеть.

Основные обозначения и терминологический аппарат

При проведении обзора граф-аналитических подходов к мониторингу ИТКС ОП и их применения для выявления аномальных состояний сетевых инфраструктур в работе вводятся следующие условных обозначения, показанные в таблице 1.

Таблица 1 – Основные обозначения

Обозначение	Физический смысл обозначения
$g = (V, E, \alpha, \beta)$	– Граф, описывающий сеть и представляемый: множеством вершин (узлов) V, множеством дуг E, функцией разметки узлов α: $V \rightarrow L_V$, где L_V – множество меток узлов, функцией разметки ребер β: $E \rightarrow L_E$, где L_E – множество меток рёбер
$d(g, g')$	– Расстояние между графами
$G=\{g_1, g_2, \ldots, g_n\}$	– Медианный граф с минимальным суммарным расстоянием от центра масс до других графов
$c(e)$	– Стоимость операции e редактирования графа (под операцией понимаем замену метки узла, замену метки дуги, вставку узла, вставку дуги, удаление узла, удаление дуги)
$[t, t+1]$	– Временной интервал наблюдения за графом (за временным рядом метрики)
$g_1^{\max} = \{\alpha, \beta\}$	– Максимально общий подграф графа g и g_2 (*maximal common subgraph – MCS*)
$\rho(g) = (L, C, \lambda)$	– Представление графа в метках, где $L = \{\alpha(x) \mid x \in V\}$, $C = \{\alpha(x), \alpha(y) \mid (x, y) \in E\}$, $\lambda(\alpha(x), \alpha(y)) = \beta(x, y)$ для всех дуг $(x, y) \in E$
$\sigma(g)$	– Спектр графа (последовательность собственных чисел матрицы $\boldsymbol{A_g}$ смежности вершин
$P^g = \bigcup_{k \geq 2} P_k^g$	– Совокупность всех путей k длины большей 2 соединяющих вершины графа g
$C = [C_{uv}]$	– Матрица изменений, элементы которой соответствуют удаленным из графа g_1 или добавленным в граф g_2 элементам (узлам, дугам)
$g_1 \Delta g_2$	– Симметричная разница графов
$\varphi_1, \varphi_2, \ldots, \varphi_m$	– Набор пороговых значений
GED	– Расстояние редактирования графа (*graph edit distance*)
msa	– Процедуры сравнения среднего графа с последующим одиночным
mma	– Процедуры сравнения среднего графа с последующим средним
msd	– Процедуры сравнения среднего графа с удаленным одиночным
mmd	– Процедуры сравнения среднего графа с удаленным средним
$c(e)$	– Стоимость операции e, переводящей один граф в другой
$c(s)$	– Суммарная стоимость операций, переводящих один граф в другой
$\gamma(u_i)$	– Число повторений вершины u_i в последовательности графов
MCS	– Максимально общий подграф (*maximal common subgraph*)
$\hat{g}$	– Средний граф последовательности $G = \{g_{n-L+1}, g_{n-L+2}, \ldots, g_n\}$
L	– Ширина скользящего окна наблюдения за графом
l	– Эвристически выбираемый интервал времени для наблюдения за графом сети
L_g	– Лапласиан графа $L_g = D_g - A_g$
D_g	– Матрица степеней
$A_g \{\lambda_1, \lambda_2, \ldots, \lambda_n\}$	– Матрица смежности вершин графа g
k	– Эмпирически выбранный предел суммирования на спектрах графов
$P_k^g(u, v)$	– Множество путей длины k, соединяющих вершины u и v графа
$C = [C_{uv}]$	– Матрица изменений
S	– Виды состояния сети

Размеченные графы

Задание графа предполагает прежде всего выделение множества вершин V и ребер E графа. В качестве вершин могут выступать как устройства, так и отдельные интерфейсы этих устройств (различного уровня семиуровневой модели взаимодействия открытых систем). В то же время под ребром может пониматься как взаимодействие двух устройств, так и взаимодействие двух интерфейсов. В качестве веса ребра между двумя взаимодействующими устройствами можно использовать объем переданных данных. Действительно, эта характеристика показывает, как интенсивно использовался тот или иной маршрут или канал за заданный период времени. Иные характеристики (такие как ширина полосы пропускания, джиттер, задержка и пр.), в итоге, влияют на объем переданных данных и де-факто показывают конкурентное преимущество того или иного маршрута из доступных.

Указанная неоднозначность формулировки сетевого графа разрешается за счет механизма абстракции, который позволяет упростить/детализировать представление графа за счет объединения/раскрытия некоторых вершин или ребер.

Формально, размеченный граф $g=(V,E,\alpha,\beta)$ с V – множеством вершин и E – множеством ребер, g – это граф, на котором заданы функции, назначающие метки (уникальные идентификаторы) вершинам и ребрам:

- α: $V \rightarrow L_V$, где L_V – множество меток узлов;
- β: $E \rightarrow L_E$, где L_E – множество меток ребер.

Как было отмечено перед процедурой анализа графа можно произвести его упрощение (абстракцию). В случае если устройство будет маршрутизатором, оно будет иметь несколько IP-адресов, которым будет соответствовать одна метка – имя устройства.

Пусть дан граф $g=(V,E,\alpha,\beta)$. Представление графа g в метках – это задание такой функции $\rho(g) = (L, C, \lambda)$, где:

1) $L = \{\alpha(x) | x \in E\}$;
2) $C = \{\alpha(x), \alpha(y) | (x, y) \in E\}$;
3) $\lambda(\alpha(x), \alpha(y)) = \beta(x, y)$ для всех дуг (x, y).

Данное представление графа $\rho(g)$ позволяет полностью абстрагироваться от исходного графа и оперировать метками вершин и ребер.

Будем далее считать, что в заданные моменты времени $t_1, t_2, \ldots, t_n$ будут формироваться преставления графов в метках $\rho_1(g), \rho_2(g), \ldots, \rho_n(g)$, на основании которых и будет производится анализ состояния сети.

Расстояние между графами

Вопрос оценки динамики ИТКС ОП как единого целого был впервые рассмотрен в работах [1, 2]. Этот подход дает обнадеживающие результаты при анализе сцен и распознавании изменяемых образов (отпечатков ладони при различных ее положениях в пространстве). Для диагностики аномального поведения сети предложен базовый перечень граф-метрик (расстояний между графами) $d(g, g')$ как меры изменчивости [3]. В простейшем варианте в качестве метода принятия решений может быть сравнение с пороговым значением. Од-

нако в отличие от других показателей, таких как загрузка процессора и памяти телекоммуникационных устройств, пороговое значение расстояние редактирования графов не нормированы и требуют предварительного исследования для каждой конкретной сети.

Согласно [1] базовым понятием граф-метрик является расстояние редактирования графа GED (graph edit distance), которое расширяет понятие расстояния Левенштейна.

Рассмотрим 6 видов операций, которые можно произвести над произвольным графом:

- замена метки узла;
- замена метки дуги;
- вставка узла;
- вставка дуги;
- удаление узла;
- удаление дуги.

Поставим в соответствие каждой операции e её стоимость $c(e)$. Пусть с течением времени t граф $g=(V, E, \alpha, \beta)$ переходит в граф $g_1=(V', E', \alpha', \beta')$.

Определение. *Метрика изменения графа* GED (graph edit distance) $d(g, g')$ будет равна минимуму суммарной стоимости операций, переводящих один граф в другой [1]:

$$d\left(g, g'\right)=\min\left(c\left(s\right)\right)=\min\left\{\sum_{i=1}^{N} c\left(e_i\right)\right\}. \quad (1)$$

Употребление формулы (1) на практике затруднено по причине экспоненциальной сложности, поэтому используются приблизительные методы на основе операций алгебры множеств вершин и ребер графов.

Для графов $g=(V, E, \alpha, \beta)$ и $g_1=(V', E', \alpha', \beta')$ можно использовать следующую формулу:

$$d\left(g, g'\right)=\left|V\right|+\left|V'\right|-2\left|V \cap V'\right|+\left|E\right|+\left|E'\right|-2\left|E_0\right|+\left|E'\right|. \quad (2)$$

Как следует, при равенстве двух графов, расстояние будет минимальным и равным 0. В случае если графы не пересекаются $g \cap g' = \varnothing$, расстояние будет максимальным. Для размеченного графа используется следующая **лемма** [1]: Пусть даны графы $g=(V,E,\alpha,\beta)$ с представлением $\rho(g) = (L, C, \lambda)$, $g_1=(V_1,E_1,\alpha_1,\beta_1)$ с представлением $\rho(g_1) = (L_1,C_1,\lambda_1)$, и $g_2=(V_2,E_2,\alpha_2,\beta_2)$ с представлением $\rho(g_2) = (L_2,C_2,\lambda_2)$. Пусть $L = L_1 \cap L_2$, $C = \{(i,j)|(i,j) \in C_1 \cap C_2\}$. Пусть $C_0 = \{(i,j)|(i,j) \in C_1 \cap C_2$ и $\lambda_1(i,j) = \lambda_2(i,j)\}$, $C'_0 = \{(i,j)|(i,j) \in C_1 \cap C_2$ и $\lambda_1(i,j) \neq \lambda_2(i,j)\}$. Тогда:

$$d(g_1, g_2) = |L_1| + |L_2| - 2|L_1 \cap L_2| + |C_1| + |C_2| - 2|C_0| + |C'_0| \ . \quad (3)$$

Найденное по указанной лемме значение будем далее называть GED (graph edit distance, расстоянием редактирования графа).

Для оценки взвешенного графа также может быть использовано следующее выражение [4]:

$$d(g,g') = \frac{\left|\beta(u,v) - \beta'(u,v)\right|}{\max\left\{\beta(u,v), \beta'(u,v)\right\}} \ . \quad (4)$$

Разделив полученное выражение на общее число ребер, т. е. на $|E \cup E'|$, можно оценить вариацию вектора характеристик ребер для графа в целом. В указанном выражении (4), в случае отсутствия того или иного ребра, вес последнего считается равным 0.

Максимальный общий подграф

Наиболее простой способ оценки «разницы» графов состоит в вычислении пересечения множеств вершин и ребер. Эта идея лежит в концепции максимального общего подграфа MCS (maximal common subgraph) [5].

Пусть даны $g=(V,E,\alpha,\beta)$ и $g_1=(V',E',\alpha',\beta')$, где g_1 – подграф g если $V_1 \subseteq V$, $E_1 \subseteq E$, $\alpha(x)=\alpha_1(x)$, $\beta(x,y)=\beta_1(x,y)$ для любых x и y. Обозначим это как $g_1 \subseteq g$. В случае если $g_1 \subseteq g$ и $g_1 \subseteq g_2$, то g_1 – это *общий подграф* g и g_2. Если $g_1 \subseteq g$ и $g_1 \subseteq g_2$ и не существует другого общего графа $g'=(V',E',\ldots)$, такого что $V_1 \subset V'$ и $E_1 \subset E'$, то g_1 – *максимальный общий подграф* MCS (maximal common subgraph) g и g_2.

Нахождение максимального общего подграфа сводится к нахождению пересечений множеств вершин и ребер. По их мощности можно оценить степень изменений сети.

Более формально для размеченных графов процедура вычисления регламентируется следующей **леммой** [1]: Пусть даны графы $g=(V,E,\alpha,\beta)$ с представлением $\rho(g)=(L,C,\lambda)$, $g_1=(V_1,E_1,\alpha_1,\beta_1)$ с представлением $\rho(g_1)=(L_1,C_1,\lambda_1)$, и $g_2=(V_2,E_2,\alpha_2,\beta_2)$ с представлением $\rho(g_2)=(L_2,C_2,\lambda_2)$. Пусть $L=L_1 \cap L_2$, $C=\{(i,j)|(i,j) \in C_1 \cap C_2\}$. Тогда граф g – *максимальный общий граф графов* g_1 и g_2, или $MCS(g_1,g_2)$.

В качестве меры изменения структуры может быть использовано следующее выражение [1]:

$$d(g,g') = 1 - \frac{\left|MCS(g,g')\right|}{MAX\left\{|g|,|g'|\right\}} \ , \quad (5)$$

где $MCS(g,g')$ – максимальный общий граф g_1 и g_2, $|g|$ – число вершин (или ребер) в графе.

В качестве более сложных метрик можно использовать и другие [6].

Методы на основе средних графов

Усреднение временного ряда характеристики по некоторому интервалу при анализе состояния часто служит простейшим методом борьбы со случайными флуктуациями и выбросами [1, 7-9]. Эта же идея лежит в концепции усреднения граф-измерений, которое основывается на понятии среднего (медианного) графа.

Медианой множества графов $G=\{g_1, g_2,\ldots, g_n\}$ или *средним графом* называется граф g' такой, что:

$$\sum_{i=1}^{N} d\left(g', g_i\right) = \min\left\{\sum_{i=1}^{N} d\left(g', g_i\right) \middle| g \in G\right\}. \quad (6)$$

Согласно определению, медианным граф $\overline{g}$ называется такой граф, суммарное GED которого до каждого члена последовательности минимально.

Рассмотрим еще одну аппроксимацию расстояния между графами, облегчающую процедуру вычисления среднего графа. Будем считать, что операция замены метки ребра с весом $\beta_1(e)$ на метку с весом $\beta_2(e)$ будет иметь стоимость $|\beta_1(e) - \beta_1(e)|$. В случае добавления или же удаления ребра из графа, стоимость операции будет равна весу ребра, т. е. $|\beta_1(e) - 0|$.

$$\begin{aligned} d_2\left(g_1, g_2\right) &= c\cdot\left[\left|V_1\right| + \left|V_2\right| - 2\left|V_1 \cap V_2\right|\right] + \\ &+ \sum_{e\in E_1\cap E_2}\left|\beta_1(e) - \beta_2(e)\right| + \sum_{e\in E_1\setminus(E_1\cap E_2)}\beta_1(e) + \sum_{e\in E_2\setminus(E_1\cap E_2)}\beta_2(e). \end{aligned} \quad (7)$$

Константа c позволяет учитывать величину влияния операций вставки/удаления узлов по отношению к операции над ребрами графа.

Рассмотрим объединенный граф $g=(V, E, \alpha, \beta)$ последовательности $G = \{g_1, g_2,\ldots,g_n\}$, где $V=\bigcup_{i=1}^{n} V_i$, $E=\bigcup_{i=1}^{n} E_i$ и обозначим через $\gamma(u_i)$ число повторений вершины u_i в последовательности графов.

Определим граф $\hat{g}=(\hat{V}, \hat{E}, \alpha, \hat{\beta})$ следующим образом:

$$\begin{aligned} &\hat{V} = \left\{u \mid u \in V \text{ и } \gamma(u) > n/2\right\}, \\ &\hat{E} = \left\{(u,v) \mid u, v \in \hat{V}\right\}, \\ &\hat{\beta}(u,v) = \operatorname{med}\left(\beta_i(u,v) \mid i = 1\ldots n\right). \end{aligned} \quad (8)$$

Согласно теореме, доказанной в [1], данный граф является медианным (средним) по GED, вычисляемому по формуле (7). Он не является уникальным, т. к. операция вставки-замены узла позволяет получить семейство средних графов.

Вычисление среднего графа лежит в основе целого ряда методов выявления аномальных состояний сети. Рассмотрим их более подробно.

Сравнение среднего графа с последующим одиночным (*msa*).

В данном процессе производится вычисление среднего графа по «скользящему окну» длиной L.

Пусть $\hat{g}$ – средний граф последовательности $G = \{g_{n-L+1}, g_{n-L+2}, \ldots, g_n\}$. Тогда расстояние, $d_2\left(\hat{g}, g_{n+1}\right)$, в сравнении с предопределенным порогом может быть использовано для выявления *аномальных* (*скачкообразных*) изменений в поведении сети.

В качестве порога можно использовать среднее изменение GED сети по скользящему окну:

$$\varphi = \frac{1}{L}\sum_{i=n-L}^{n} d_2\left(\hat{g}_n, g_i\right).$$

Событие «аномальное поведение сети» генерируется при условии:

$$d_2\left(\hat{g}_n, g_{n+1}\right) \geq \alpha \cdot \varphi .$$

Как было указано, средний граф не является уникальным, в связи с чем, если было получено семейство средних графов $\left(\hat{g}_{n_1}, \hat{g}_{n_2} \ldots \hat{g}_{n_m}\right)$, можно вычислить набор пороговых значений (φ_1, φ_2,…, φ_m). Решение о состоянии сети можно определить из следующего условия:

$$d_2\left(\hat{g}_{n_1}, g_{n+1}\right) \geq \alpha \cdot \varphi_1 \wedge d_2\left(\hat{g}_{n_2}, g_{n+1}\right) \geq \alpha \cdot \varphi_2 \wedge \ldots \wedge d_2\left(\hat{g}_{n_m}, g_{n+1}\right) \geq \alpha \cdot \varphi_m . \quad (9)$$

Для иллюстрации данного метода приведем сравнительные графики последовательных графов-измерений ботнет-атаки на телекоммуникационную сеть из набора данных CTU-13 [10] (набор номер 13). На рис. 1 красным цветом показан график расстояния редактирования, синим – расстояния редактирования между средним графом, полученным усреднением по окну длинной 15 графов и последующим согласно алгоритму *msa*. Указанный график иллюстрирует возможность выявить периоды существенных изменений сети и эффект фильтрации локальных выбросов.

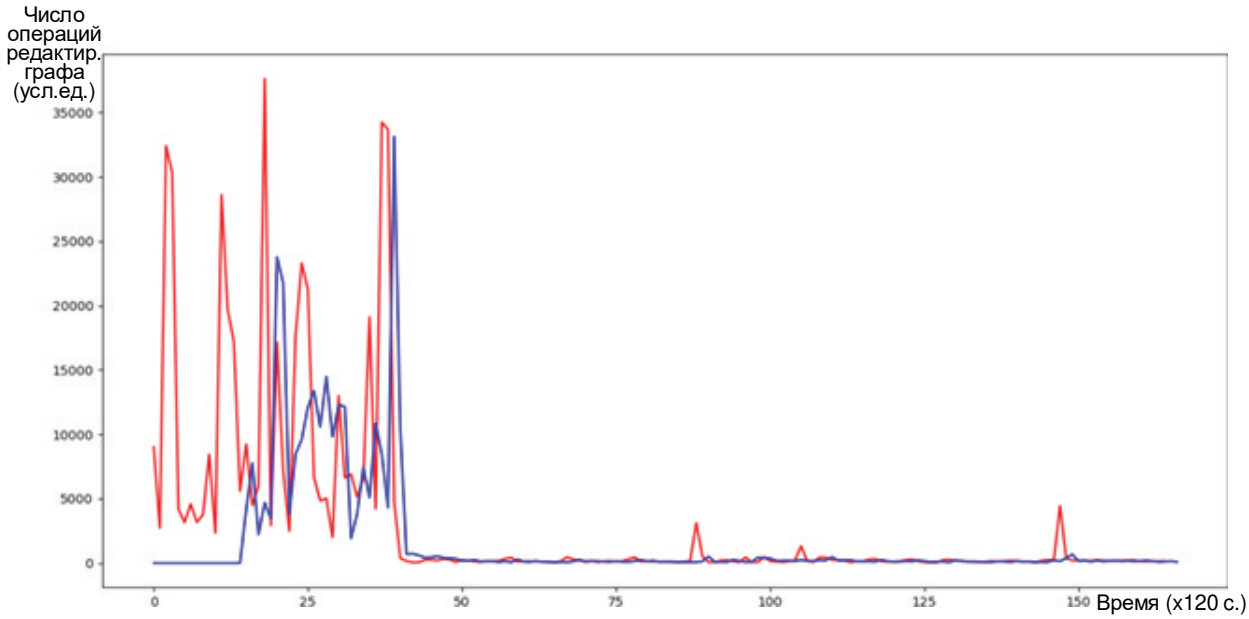

Рис. 1. Эффект фильтрации при усреднении последовательности графов

Сравнение среднего графа с последующим средним (*mma*).

В данной схеме в последовательных скользящих окнах L_1 $G_1 = \left\{g_{n-L+1}, \ldots, g_n\right\}$ и L_2 $G_2 = \left\{g_{n-L+1}, \ldots, g_{n+L+2}\right\}$ вычисляются средние графы $\hat{g}_n$ и $\hat{g}_{n+1}$.

В качестве правила принятия решения о состоянии сети используется следующее уравнение:

$$d_2\left(\hat{g}_n, \hat{g}_{n+1}\right) \geq \alpha \cdot \left[\frac{L_1\varphi_1 + L_2\varphi_2}{L_1 + L_2}\right]. \quad (10)$$

Сравнение среднего графа с удаленным одиночным (*msd*).

В случае, если имеет место *постепенное* изменение состояния сети, целесообразно сравнивать средний граф $\hat{g}_n$ не с последующим g_{n+1}, а с отстоящим на l измерений, где l выбирается эвристически.

Сравнение среднего графа с удаленным средним (*mmd*).

Данный метод представляет собой комбинацию предыдущего случая и сравнения последовательных средних. Как описано выше, рассмотрим средний граф $\hat{g}_1$ по множеству $G_1 = \{g_{n-L1+1}, \ldots, g_n\}$ и $\hat{g}_2$ по множеству $G_2 = \{g_{n+l+1}, \ldots, g_{n+l+L}\}$. Сравнение удаленных друг от друга граф-измерений позволяет оценить абсолютную величину «постепенного» изменения состояния сети.

Методы анализа на основе спектра графов сети

Пусть задан граф $g = (V, E, \alpha, \beta)$ с матрицей смежности вершин A_g. Спектром графа $\sigma(g)$ назовем последовательность собственных чисел матрицы A_g $\{\lambda_1, \lambda_2, \ldots, \lambda_n\}$.

В настоящее время известна также другая методика исследования свойств (неориентированного) графа на основе собственных чисел матрицы Кирхгофа (Лапласиан графа) [1, 11]: $L_g = D_g - A_g$, где D_g – матрица степеней определяется следующим образом:

$$D_g = \operatorname{diag}\left\{\sum_{v \in V} \beta(u, v) \,|\, u \in V_g\right\}.$$

В случае невзвешенного графа, элементами матрицы D_g будут степени вершин. В случае ориентированного графа матрица Кирхгофа определяется следующим выражением: $L_g = D_g - \left(A_g + A_g^T\right)$.

На основе полученных собственных значений матрицы смежности вершин графа или матрицы Кирхгофа вычисляется расстояние между графами (GED) [1]:

$$d\left(g, g'\right) = \sqrt{\frac{\sum_{i=1}^{k} \left(\lambda_i - \mu_i\right)^2}{\min\left\{\sum_{i=1}^{k} \lambda_i^{\,2}, \sum_{i=1}^{k} \mu_i^{\,2}\right\}}}. \quad (11)$$

Для спектров графов $\sigma\left(A_g\right) = \left\{\lambda_1, \lambda_2, ..., \lambda_n\right\}$, $\sigma\left(A_{g'}\right) = \left\{\mu_1, \mu_2, ..., \mu_n\right\}$ k – эмпирически выбранный предел суммирования. В приложениях распознавания образов и обработки изображений экспериментально установлено оптимальное значение $k = 20$.

Аналогичное выражение может быть приведено и для матрицы Кирхгофа.

Экспериментальные исследования [3] показывают сходное поведение данной метрики с приведенными ранее в данной работе.

Сетевые измерения на основе структуры графа

Передача данных в сети осуществляется посредством маршрутов, поэтому удаление вершины приводит к отказу маршрутов, содержащих данную вершину в качестве промежуточной. Исходя из этих соображений в качестве чувствительной метрики состояния сети можно использовать рассмотренное ранее расстояние редактирования GED, основанное на числе путей, содержащих заданную вершину(ы) [12].

Для вершин u, $v \in V_g$ рассмотрим следующее множества путей графа:

$P_k^g(u,v)$ – путей длины k, соединяющих вершины u и v;

P_k^g – совокупность путей длины k в графе;

$P_k^g(u,v) = \bigcup_{k>2} P_k^g(u,v)$ – множество путей длины большей 2 соединяющих вершины u и v;

$P^g = \bigcup_{k\geq 2} P_k^g$ – совокупность всех путей.

Для выделенного (непустого) подмножества ребер $\hat{E} \subseteq E$, сформируем новый граф $g(\hat{E}) = (V', E', \alpha, \beta)$, таким образом, что в исходном графе остаются только те дуги, которые содержатся в маршрутах, содержащих дуги из $\hat{E} \subseteq E$. Более формально, граф $g(\hat{E})$ формируется следующим образом:

1) $V' = V$.

2) Ребро $e \in E'$ тогда и только тогда, когда $\exists p \in P_k^g : e \in p$ и $\exists e_1 \in \hat{E}, e_1 \in p$.

3) $\alpha' = \alpha$.

4) Веса ребер в G_g равны числу маршрутов в P^g, содержащих данное ребро в качестве компонента (и, по крайней мере, одно ребро из предопределенного набора $\hat{E} \subseteq E$).

5) Атрибуты весов в созданном описанным способом графе $g(\hat{E})$ отражают степень важности ребер в процессах передачи данных через сеть, и поэтому определяют степень влияния на связность узлов.

Сравнение двух созданных на основе $g_1 = (V_1, E_1, \alpha_1, \beta_1)$ и $g_2 = (V_2, E_2, \alpha_2, \beta_2)$ графов, может быть осуществлено по формуле (6). В качестве $\hat{E}$ выбирается E_1 и E_2. Также можно использовать множество ребер $mcs(g_1, g_2)$. Общей рекомендаций при создании $\hat{E}$ является включение наиболее значимых связей сети.

Вариантом описанной методики является исследование 2-компонентной связности графа. В результате смежные вершины в полученном графе соответствуют вершинам исходного графа, соединенных через общего соседа. Как результат, полученная структура более чувствительна к изменениям в топологии (включение/исключением вершин, ребер), нежели исходная структура. Однако

в данном случае, граф остается невзвешенным. Сравнение полученных структур может быть произведено по формулам (3) и (4).

Идентификация областей изменения

Симметричная разность графов.

При анализе динамики сети важным является не только установление факта изменения, приводящего к ошибкам, но и выявление компонент графа сети, приводящих к возникновению событий [1, 12].

Расстояние между двумя графами $g_1 = (V_1, E_1, \alpha_1, \beta_1)$ и $g_2 = (V_2, E_2, \alpha_2, \beta_2)$ может быть охарактеризовано при помощи матрицы изменений $C = [C_{uv}]$, элементы которой соответствуют удаленным из g_1 или добавленным в g_2. Строки и столбцы матрицы C соответствуют множеству $V_1 \cup V_2$. В случае удаления или добавления ребра (u, v) соответствующий элемент матрицы будет равен 1, в случае если соответствующее ребро присутствует в обоих графах, соответствующий элемент будет равен 0. Данная матрица описывает граф, называемый *симметричной разницей графов*, и обозначается $g_1 \Delta g_2$.

Сумма элементов по строкам (или столбцам) матрицы C дает вектор изменений относительно вершин объединенного множества $V_1 \cup V_2$. Ранжирование с последующим выделением n максимальных компонент позволяет локализовать области изменений.

Указанный подход может быть распространен на взвешенные графы, при этом компоненты симметричной разности вычисляются по формуле:

$$C_{u,v} = \frac{\left|\beta(u,v) - \beta'(u,v)\right|}{\max\left\{\beta(u,v), \beta'(u,v)\right\}}, \tag{12}$$

где $(u,v) \in E_1 \bigcup E_2$.

Аналогично, для анализа динамики может быть использована группа симметричных разностей высшего порядка:

$$g_i \Delta^2 g_{i+2} = \left(g_i \Delta g_{i+1}\right) \Delta \left(g_{i+1} \Delta g_{i+2}\right),$$

$$g_i \Delta^3 g_{i+4} = \left(g_i \Delta^2 g_{i+2}\right) \Delta \left(g_{i+2} \Delta^2 g_{i+4}\right),$$

...

Анализ на основе графа соседей вершин.

Альтернативой симметричной разности является подход на основе измерения расстояния между соответствующими (последовательными во времени) графами соседей вершины. Данная техника позволяет получить вектор расстояний между графами из g_1 и g_2. Каждая координата вектора соответствует расстоянию между графами соседей с «точки зрения» отдельной вершины и смежных с нею вершин, что и позволяет *выявить области изменений.*

Последовательные измерения по времени состояния сети по описывающим их графам могут быть сравнены, используя описанный выше подход, где в качестве измерения расстояния между графами применить формулы (2) – (5). Граф соседей вершины, присутствующей только в одном графе, сравнивается с

пустым графом. Результатом операции является вектор расстояний графов соседей вершин: $d = \left[d\left(g_1'(u), g_2'(u) \right) \right]$.

Соседний подграф вершин описывает связи с вершинами, связанными 1 дугой. Для целей анализа целесообразно также рассмотреть 2-соседний граф, описывающий 1- и 2-компонентную связность, т. е. включающий 1 и 2 достижимые вершины, вместе с связывающими их ребрами.

Алгоритм выявления аномального состояния сетевой инфраструктуры на основе модифицированной процедуры кластеризации *k*-средних

Как отмечено выше, в современных сетевых инфраструктурах процедура мониторинга должна осуществляться в режиме реального времени. Причем если в момент времени t состояние наблюдаемой сети принять за исходное состояние (первое множество вершин и ребер на сетевом графе g), то в промежуток времени $t + 1$ в силу внутренних (изменение режимов работы, величины обрабатываемого трафика и пр.), а также внешних (ошибки персонала, дестабилизирующие фактры (ДФ) и пр.) воздействий на динамической структуре будет наблюдаться совершенно другое состояние (второе множество вершин и ребер сетевого графа g_1), в момент времени $t + 2$ может наблюдаться третье состояние, описываемое сетевым графом g_2, или сеть может вернуться в исходное состояние, описываемое сетевым графом g и т. д. Каждое из этих состояний характеризуется расстоянием между графами $d(g, g_1)$, $d(g, g_2)$ и пр. Если исходное состояние сети, описываемое графом g принять за эталонное и определить порог на изменение расстояния между ним и новыми графами, образующимися в моменты времени $t + 1$, $t + 2$ и т. д., как и порог на суммарное расстояние от него до каждого образованного графа, то в случае превышения величины порога будем считать, что сеть перешла в иное состояние. При этом нормальное состояние ИТКС характеризуется допустимыми изменениями топологии сети, описываемыми некоторым множеством графов, также, как и другие виды состояний ИТКС определяются некоторым множеством графов. Эти множества образуют кластеры, где средний граф является её центром [13-15].

Функционирование алгоритмов кластеризации и их эффективность обычно оценивают с помощью таких параметров как временная (быстродействие) и их пространственная (объем обрабатываемой информации) сложность. Например, в [16] представлена оценка процедур кластеризации, используемых при обработке числовых значений: *FarthestFirst*, *k*-средних, метод ближайшего соседа, *EM* и *EM*-модифицированный. Для получения их сравнительной оценки сгенерированы тестовые последовательности данных (исходные кластеры) в виде точек двумерного евклидова пространства. При тестировании алгоритмов в ходе решения задачи кластеризации были измерены объемы оперативной памяти и время исполнения алгоритмов, таблица 2. В [16] приводится алгоритм оценки качества кластеризации, по которому получены следующие индексы качества: *FarthestFirst* – 0,48; метод ближайшего соседа – 0,32; *EM*-алгоритм – 0,9; *EM* (модифицированный) – 0,6; *k*-средних – 0,85.

Таблица 2 – Результаты оценки эффективности алгоритмов кластеризации по объему обрабатываемой информации и быстродействию [16]

Сравниваемые алгоритмы кластеризации	Временная сложность, секунды							Пространственная сложность (объем памяти), Мбит							
	750	1000	1500	2000	3000	10000	50000	750	1000	1500	2000	2500	3000	10000	50000
FarthestFirst	0,01	0,06	0,01	0,01	0,04	0,07	0,2	0,23	0,36	0,29	0,37	0,50	0,58	1,67	3,62
k-средних	0,01	0,07	0,11	0,26	0,41	1,95	9,23	0,90	3,08	1,93	3,56	1,91	0,74	2,69	16,59
EM	0,47	1,2	2,31	2,74	4,66	20,22	124,1	1,09	1,45	3,13	4,68	1,60	5,24	9,03	35,37
EM модифицированный	1,45	2,24	5,5	11,4	23,3	298,9	9049	1,57	2,98	7,16	5,52	3,36	5,69	11,64	45,05
Метод ближайшего сосед	18	75	124	219	477	900	34122	24,57	54,1	97,1	206,9	294,9	386,2	1103,33	2837,4

Оценка эффективности алгоритмов кластеризации получена по результатам тестовой проверки на основе установления контрольного значения в объектах кластеризации и расчета индекса качества кластеризации. Анализ результатов показал, что лучшими показателями индекса качества кластеризации обладают алгоритм *k*-средних и *EM* алгоритм со значениями 0,85 и 0,9 соответственно. При этом по сравнению с *EM* алгоритм *k*-средних имеет в среднем в 10 раз выше быстродействие, что дает основание выбрать его для обработки больших объемов данных (измерительной информации) и обучения классификатора оценки состояния всей распределенной ИТКС ОП в целом, рис. 2, в то время, как для обработки данных временных рядов метрик сетевых устройств, как правило, используют *EM* алгоритм [3].

Блок-схема алгоритма выявления аномального состояния ИТКС фактически представляет из себя этап структурного анализа сети и состоит из подэтапа обучения классификатора (построенного по модифицированной процедуре алгоритма *k*-средних) и подэтапа непосредственно идентификации состояния сети:

На шаге 1 осуществляют ввод исходных данных о подконтрольной ИТКС.

На шаге 2 получаем последовательность графов состояния сети $G(g_0, g_1, g_2,\ldots, g_i,..., g_n)$ в моменты времени: $t_0, t_1, t_2,..., t_i,\ldots, t_n$, где g_0 начальный (проектный) граф в момент времени t_0 с эталонным состоянием, когда техническое состояние (ТС) всех элементов сети и их параметров находятся в номинале.

Обычная процедура обучения классификатора предполагает, что в качестве исходных данных для идентификации нормального и аномальных состояний сети используются облака данных как неупорядоченные наборы данных, не привязанные к какой-либо из шкал измерений. Однако в отличие от процедуры TDA (Topology Date Analysis) [3, 17], применяемой для анализа временных рядов метрик элементов сети, в данном алгоритме облако данных представляют множеством точек в заданном топологическом пространстве метрик графов, описывающих состояния ИТКС во времени. А поскольку в нем исходные данные представлены сетевыми графами, то последовательность графов сети *G* преобразуется в облако точек, где каждому графу, описываемому графовым расстоянием от графа g_0, ставится в соответствие точка в облаке данных, рис. 3 *а*).

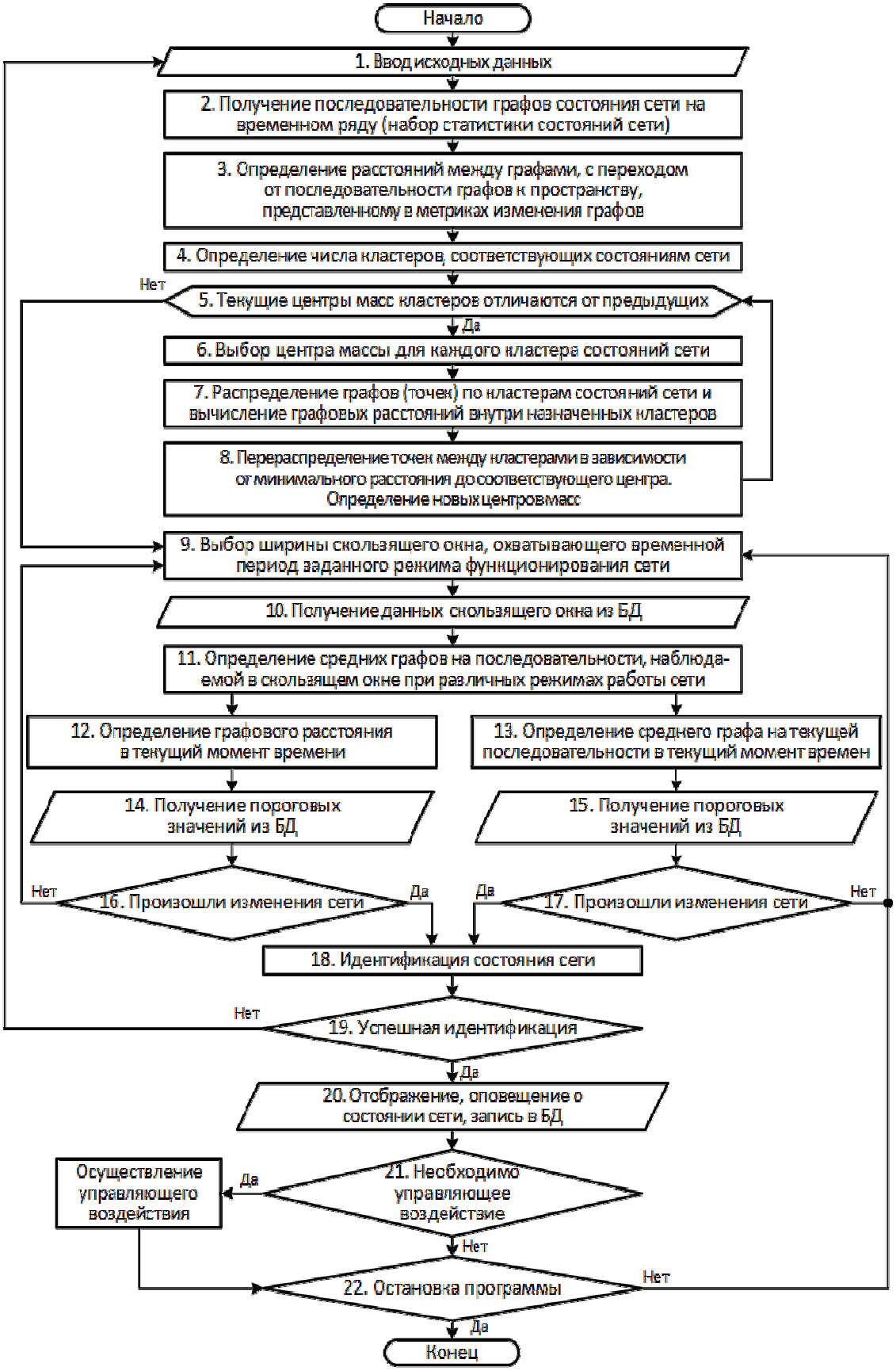

Рис. 2. Блок-схема алгоритма выявления аномального состояния ИТКС ОП на основе модифицированной процедуры кластеризации *k*-средних

DOI: 10.24412/2410-9916-2021-6-53-75
URL: https://sccs.intelgr.com/archive/2021-06/04-Budko.pdf

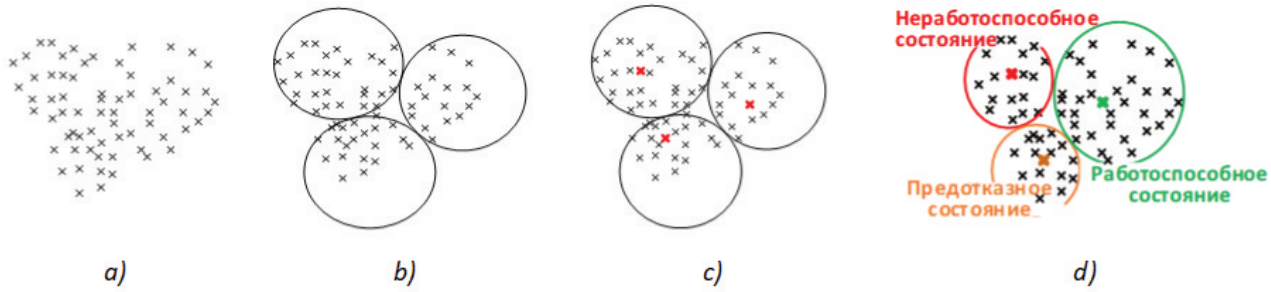

Рис. 3. Процедура модифицированного алгоритма *k*-средних при определении состояния сети по их графам

На шаге 3 определяем расстояния между графами g_0 и g_i, $i=\overline{1,n}$ с переходом от последовательности графов к пространству в метриках изменения, редактирования графа (graph edit distance, GED):

– замена метки дуги → изменение состояния канала связи;
– вставка узла → восстановление (наращивание) узлов сети;
– вставка дуги → восстановление (добавление) канала связи;
– удаление узла → отказ узла (деградация сети);
– удаление дуги → отказ канала (нарушение связности – деградация сети).

На шаге 4 на множестве *G* определяем кластеры *k* по видам состояния сети $S_1,S_2,...,S_k, S_i \bigcap S_j = \emptyset, i \neq j$. Одной из особенностей алгоритма *k*-средних является заранее определяемое число кластеров. Для мониторинга сетевых инфраструктур, как правило, в конечной интерпретации оператора подсистемы мониторинга таких состояний всего два «работоспособное» состояние («норма») – сеть выполняет свои функции и «неработоспособное» состояние («авария», или блокировка сети). Для недопущения внезапного перехода сети в аварию, особый интерес представляет «предотказное» состояние, введенное ГОСТ 27.002–2015 и характеризуемое повышенным риском отказа объекта контроля, возникающего в результате внешних ДФ и внутренних воздействий в процессе ее функционирования. В соответствии с международной классификацией (Рек. ITU-T M.3703) «предотказное состояние» соотносится с «критическим». В связи с изложенным, на шаге 13 произвольно определяют на облаке точек три кластера, которые в последующем, после завершения схождения алгоритма *k*-средних, будут соответствовать основным состояниям ИТКС, рис. 3 *b*).

На шаге 5 проверяем условие прекращения процедуры обучения классификатора $g_k^{(t)} = g_k^{(t-1)}$, когда центры масс кластеров (медианные графы) на текущем временном отсчете (*t*) не отличаются от предыдущего (*t*-1).

На шаге 6 на заданных при реализации шага 4 кластерах графов выбираем *k* средних графов g_i, $i=\overline{1,k}$, являющихся начальными центрами кластеров (центрами масс – центроидами) путем процедуры минимизации суммы квадратов расстояний от каждой точки заданного кластера до его центра, рис. 3 *c*).

На шаге 7 осуществляем перераспределение всех точек графов по кластерам, соответствующим тому или иному состоянию сети путем определения расстояния до центров масс (полученных на шаге 6) от каждого из наблюдаемых графов,

рис. 3 *d*). Если окажется, что рассматриваемый граф ближе к медианному графу (тяготеет к нему), описывающему первое (нормальное) состояние сети – «1», следовательно состояние данного графа имеет такое же состояние, как и граф с центром «1». Если рассматриваемый граф ближе к медианному графу, описывающего состояние «2», то состояние этого графа имеет такое же состояние, как и граф с центром «2», и т. д.

На шаге 8 по выражению (8) пересчитываем центры вновь сформированных кластеров по правилу среднего медианного графа по всем графам кластера, после чего итерационную процедуру повторяем по шагу 5. Поскольку в динамической системе, к которой относят ИТКС, состояние сети постоянно изменяется, то выход из строя узла сети (вершины графа) или канала связи (ребра графа) влечет за собой перемаршрутизацию, направленную на восстановление функционального состояния сети. Поэтому с течением времени граф изменится, в связи с чем на каждом интервале времени мониторинга сети необходима итерация:

- по определению новых кластеров ее состояния в следующий момент времени $t + 1$;
- назначению центров масс (медианных графов), соответствующих видам ТС сети;
- определению расстояния наблюдаемого графа до центров масс медианных графов;
- сравнение вычисленных расстояний и по их минимуму – идентификация вида состояния сети.

При этом итерационная процедура повторяется до момента времени, когда рассматриваемый граф не окажется ближе к центру кластера «2» графа, имеющему «предотказное» ТС, или к центрам кластеров «3» или «1» графа, имеющих состояние «авария» или «норма» соответственно.

Таким образом, при выполнении условия шага 5, когда центроиды кластеров не перемещаются в графовом пространстве $g_k^{(t)} = g_k^{(t-1)}$, итерационные процедуры кластеризации прекращают.

Подэтап обучения классификатора состояния ИТКС проводится заблаговременно (off-line), как правило в период опытной эксплуатации, но накопление статистики функционального состояния ИТКС (дообучение) может осуществляться на всех этапах эксплуатации (ЖЦ) при разных режимах работы сети.

На шаге 9 проводим выбор ширины скользящих окон, охватывающих периоды режимов работы сети: ненагруженный L_1, нагруженный L_2, перегруженный L_3.

На шаге 10 получаем из базы данных (БД) заданные значения параметров скользящего окна.

На шаге 11 определяем медианные графы на последовательностях $G(g_1, g_2,..., g_n)$, наблюдаемых в скользящих окнах шириной L_1, L_2, L_3 при разны режимах работы сети, что важно для дальнейшего уточнения ее состояния путем сравнения с текущим графом g_i.

На шагах 12, 14, 16 для выявления скачкообразных (аномальных) изменений на сети осуществляем сравнение среднего графа, рассчитанного в скользящих

окнах шириной L_1, L_2, L_3 с последующим одиночным либо (с последующим средним) по процедурам, описанным выше (*msa*, *mma*).

На шагах 13, 15, 17 для выявления постепенных изменений сети осуществляем сравнение среднего графа, рассчитанного в скользящих окнах шириной L_1, L_2, L_3 с удаленным одиночным (удаленным средним) по процедурам, описанным выше (*msd*, *mmd*).

На шаге 18 осуществляем идентификацию состояния сети и репликацию БД.

На шаге 19 при успешной идентификации переход на новый цикл мониторинга.

На шаге 20 компонент интеллектуальной обработки транслирует на компонент отображения измерительную информацию о виде состояния ИТКС в интересах системы поддержки принятия решения (СППР) с записью в БД.

На шаге 21 в СППР транслируем команду по результатам цикла мониторинга.

Таким образом, предложенный алгоритм оценки состояния сети на основе измерения расстояния графов и шагов алгоритма *k*-средних является невероятностной версией *EM*-алгоритма, рассмотренного в [3]. Однако, поскольку предлагаемый алгоритм в основе своей использует методологию кластеризации, используемую в процедуре алгоритма *k*-средних, то данную версию можно назвать его модификацией.

В *отличие* от классического алгоритма *k*-среднего, где работа ведется над точками евклидова пространства, здесь оперируем терминами пространства сетевых графов с метриками в виде графовых расстояний, при этом в качестве исходных данных для классификации в алгоритме *k*-средних используют облака данных как неупорядоченные наборы данных, не привязанные к какой-либо из шкал измерений, а в предложенном алгоритме данные представлены множеством графов в топологическом пространстве их метрик, описывающих состояния сети во времени.

Экспериментальные исследования

Некоторые из приведенных методов были опробованы на наборе данных CTU-13 [10] (5,8 и 13 наборы соответственно), описывающего трафик внешней ботнет-атаки на корпоративную сеть, собранный в Чешском техническом университете (г. Прага). Набор данных представляет собой запись трафика, проходящего через граничный маршрутизатор, и содержит последовательность записей вида «Время – IP отправителя – Порт отправителя – IP получателя – Порт получателя – Объем переданных данных». Время наблюдения за сетью составляет 4 часа. Указанный временной интервал был разделен на интервалы по 120 секунд. В каждом интервале производилось построение графа взаимодействия. Метками вершин являлись IP-адреса, а весами ребер – суммарный объем переданных данных между соответствующей парой адресов.

На рис. 4-6 представлены графики (красный цвет) парных расстояний редактирования (GED) между последовательными графами, вычисленными по формуле (7). Синим цветом показано относительное количество вершин в графе

№6. 2021
ISSN 2410-9916

в указанный момент времени. Ось абсцисс – ось времени (x120 с.) ось ординат – число операций редактирования графа (например, удаление узла – 500 усл. ед.).

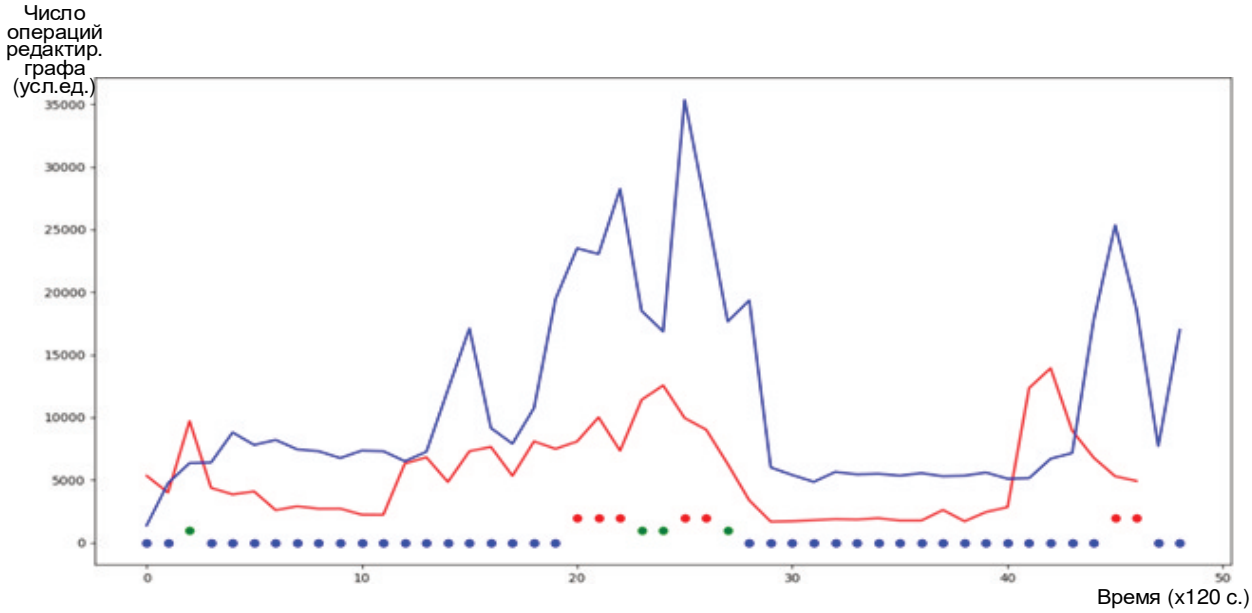

Рис. 4. Кластеризация состояния сети под ботнет-атакой (5 набор из CTU-13)

В наборе данных, визуализированном на рис. 4 и 5 сеть находится под атакой ботнет-сети Virut. Из рисунков следует, что метод кластеризации позволяет отделить существенные для целей анализа значительные изменения структуры взаимодействия от несущественных.

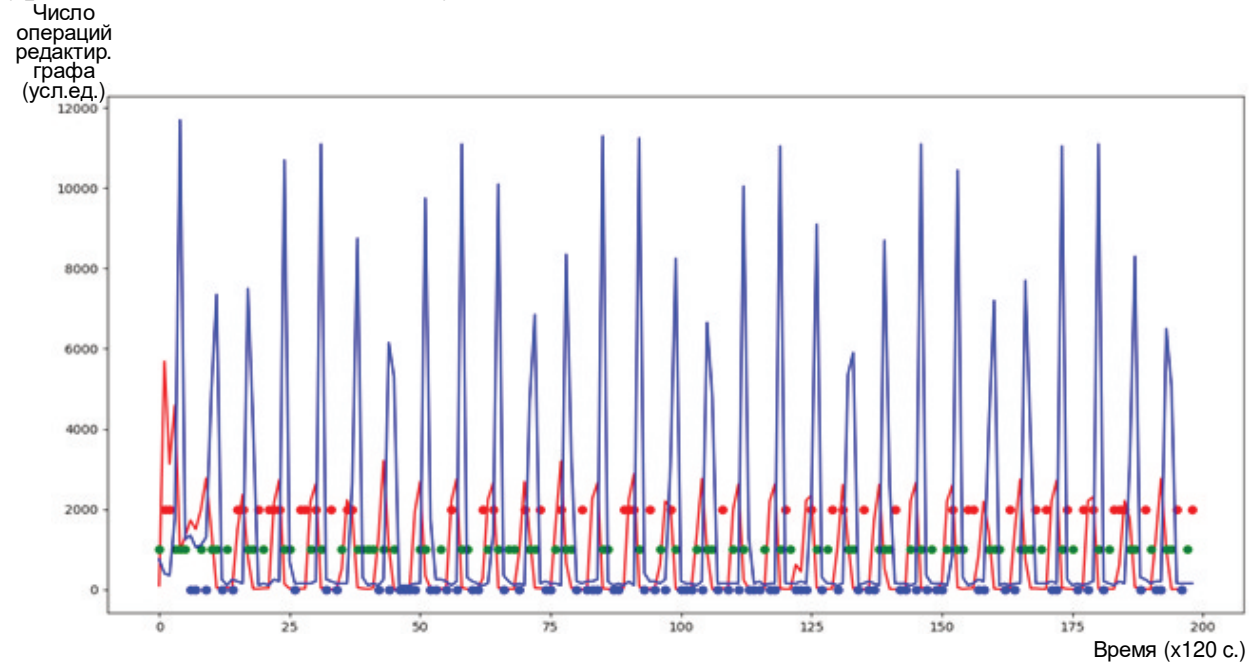

Рис. 5. Кластеризация состояния сети под ботнет-атакой (8 набор из CTU-13)

DOI: 10.24412/2410-9916-2021-6-53-75
URL: https://sccs.intelgr.com/archive/2021-06/04-Budko.pdf

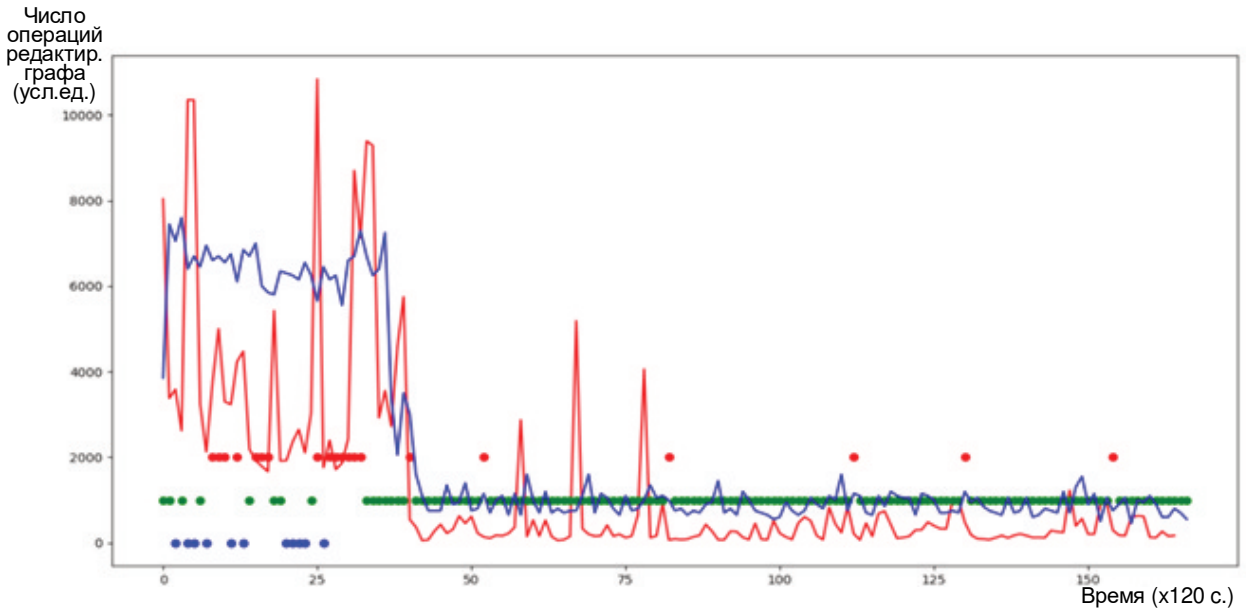

Рис. 6. Кластеризация состояния сети под ботнет-атакой (13 набор из CTU-13)

Более нагляден набор данных, визуализированный на рис. 6. Рисунок демонстрирует трафик распределенной DDoS атака ботнет-сети Virut. Как следует из рисунка, форма графика GED малоинформативна. На протяжении всего периода наблюдения видны колебания структуры. Более четко выделить периоды атаки на сеть позволяет описанная кластеризация на основе метода *k*-средних (показана в виде точечных графиков). Полученный набор графов был разбит на 3 кластера, каждому из которых соответствует своя «дорожка» (с представлением зелеными, синими и красными точками различные классы состояний сети). На графике четко прослеживается «подготовительный» этап атаки с 0 до 100 временных отсчета и собственно атаку – с 100 по 160 временные отсчеты.

Заключение

В данной работе произведен обзор методов сетевых измерений на основе структуры графов сети с использованием таких понятий как средние графы и спектры графов сети, которые ориентированы на проведение мониторинга реального времени на базе граф-измерений. Определены основные области применения указанных методов. На примере журналов трафика атаки на реальную сеть продемонстрирована практическая возможность использования данной группы методов, в частности графовой версии алгоритма *k*-средних для выявления возможных состояний телекоммуникационной сети и последующей диагностики текущего состояния. Продемонстрирована возможность использования процедуры усреднения последовательности графов для фильтрации возможных выбросов. Основным результатом и научной *новизной* работы является разработка на основе теоретических результатов о средних графах и алгоритма *k*-средних модифицированной процедуры кластеризации *k*-средних, отличающейся от классической, где работа ведется над точками евклидова пространства, тем, что оперирует терминами пространства сетевых графов с метриками

в виде графовых расстояний, при этом в качестве исходных данных для классификации в алгоритме *k*-средних используют облака данных как неупорядоченные наборы данных, не привязанные к какой-либо из шкал измерений, а в предложенном алгоритме данные представлены множеством графов в топологическом пространстве их метрик, описывающих состояния сети во времени.

Литература

1. Bunke H., Dickinson P. J., Kraetzl M., Wallis W. D. A Graph-Theoretic Approach to Enterprise Network Dynamics. – Basel: Birkhauser, 2007. – 226 p.

2. Fischer A., Suen C. Y., Frinken F., Riesen K., Bunke H. A Fast Matching Algorithm for Graph-Based Handwriting Recognition // Graph-Based Representations in Pattern Recognition. 2013. Vol. 7877. P. 194-203.

3. Аллакин В. В., Будко Н. П., Васильев Н. В. Общий подход к построению перспективных систем мониторинга распределенных информационно-телекоммуникационных сетей // Системы управления, связи и безопасности. 2021. №4. С. 125-227. DOI: 10.24412/2410-9916-2021-4-125-227.

4. Umeyama S. An eigendecomposition approach to weighted graph matching problems // IEEE Transactions on Pattern Recognition and Machine Intelligence. 1988. Vol. 10. No. 5. P. 695-703.

5. Bunke H., Günter S. Weighted mean of a pair of graphs // Computing. 2001. Vol. 67. No. 3. P. 209-224.

6. Wallis W. D., Shoubridge P. J., Kraetzl M., Ray D. Graph distances using graph union // Pattern Recognition Letters. 2001. Vol. 22. P. 701-704.

7. Bunke H. On a relation between graph edit distance and maximal common subgraph // Pattern Recognition Letters. 1997. Vol. 18. P. 689-694.

8. Bunke H., Kraetzl M., Shoubridge P. J., Wallis W. D. Measuring abnormal change in large data networks // In Proceedings of the International Conference on Information: Decision and Control. Adelaide. 2002. P. 53-58.

9. Bunke H., Messmer B. Recent advances in graph matching // International Journal of Pattern Recognition and Artificial Intelligence. 1997. Vol. 11. No. 1. P. 169-203.

10. Garcia S., Grill M., Stiborek J., Zunino A. An empirical comparison of botnet detection methods // Computers and Security Journal. 2014. vol. 45. P. 100-123. URL: http://dx.doi.org/10.1016/j.cose.2014.05.011 (дата обращения 15.11.2021).

11. Цветкович Д., Дуб М., Захс Х. Спектры графов: теория и применение. Киев: Наукова думка, 1984. 384 с.

12. Dickinson P. Graph Based Techniques for Measurement of Intranet Dynamics. PhD thesis. Adelaide: Institute for Telecommunications Research, University of South Australia, 2006.

13. Günter S., Bunke H. Self-organizing map for clustering in the graph domain // Pattern Recognition Letters. 2002. Vol. 23. No. 4. P. 405-417.

14. Günter S., Bunke H. Validation indices for graph clustering // Pattern Recognition Letters. 2003. Vol. 24. No. 8. P. 1107-1113.

15. Jiang X., Bunke H. On median graphs: Properties, algorithms, and applications // IEEE Transactions on Pattern Analysis and Machine Intelligence. 2001. Vol. 23. No. 10. P. 1144-1151.

16. Бильгаева Л. П., Самбялов З. Г. Оценка качества алгоритмов кластеризации // Вестник Восточно-Сибирского государственного университета технологий и управления. 2013. № 6. С. 53-60.

17. Нашивочников Н. В., Пустарнаков В. Ф. Топологические методы анализа в системах поведенческой аналитики // Вопросы кибербезопасности. 2021. № 2. С. 26-36. DOI: 10.21681/2311-3456-2021-2-26-36.

References

1. Bunke H., Dickinson P. J., Kraetzl M., Wallis W. D. *A Graph-Theoretic Approach to Enterprise Network Dynamics*. Basel, Birkhauser, 2007. 226 p.

2. Fischer A., Suen C. Y., Frinken F., Riesen K., Bunke H. A Fast Matching Algorithm for Graph-Based Handwriting Recognition. *Graph-Based Representations in Pattern Recognition*, 2013, vol. 7877, pp. 194–203.

3. Allakin V. V., Budko N. P., Vasiliev N. V. A general approach to the construction of advanced monitoring systems for distributed information and telecommunications networks. *Systems of Control, Communication and Security*, 2021, no. 4, pp. 125-227. DOI: 10.24412/2410-9916-2021-4-125-227 (in Russian).

4. Umeyama S. An eigendecomposition approach to weighted graph matching problems. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 1988, vol. 10, no. 5, pp. 695-703.

5. Bunke H., Günter S. Weighted mean of a pair of graphs. *Computing*, 2001, vol. 67, no. 3, pp. 209-224.

6. Wallis W. D., Shoubridge P. J., Kraetzl M., Ray D. Graph distances using graph union. *Pattern Recognition Letters*, 2001, vol. 22, pp. 701-704.

7. Bunke H. On a relation between graph edit distance and maximal common subgraph. *Pattern Recognition Letters*, 1997, vol. 18, pp. 689-694.

8. Bunke H., Kraetzl M., Shoubridge P. J., Wallis W. D. Measuring abnormal change in large data networks. *In Proceedings of the International Conference on Information, Decision and Control.* Adelaide, 2002, pp. 53-58.

9. Bunke H., Messmer B. Recent advances in graph matching. *International Journal of Pattern Recognition and Artificial Intelligence*, 1997, vol. 11, no. 1, pp. 169-203.

10. Garcia S., Grill M., Stiborek J., Zunino A. An empirical comparison of botnet detection methods. *Computers and Security Journal*, 2014, vol. 45, pp. 100-123. Available at: http://dx.doi.org/10.1016/j.cose.2014.05.011 (accessed 15 November 2021).

11. Tsvetkovich D., Dubh M., Sachs H. *Spektry grafov. Teoriya i primenenie* [Spectra of graphs. Theory and application]. Kiev, Naukova dumka Publ., 1984. 384 p. (in Russian).

12. Dickinson P. Graph Based Techniques for Measurement of Intranet Dynamics. PhD thesis. Adelaide, Institute for Telecommunications Research, University of South Australia, 2006.

13. Günter S., Bunke H. Self-organizing map for clustering in the graph domain. *Pattern Recognition Letters*, 2002, vol. 23, no. 4, pp. 405-417.

14. Günter S., Bunke H. Validation indices for graph clustering. *Pattern Recognition Letters*, 2003, vol. 24, no. 8, pp. 1107-1113.

15. Jiang X., Bunke H. On median graphs: Properties, algorithms, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2001, vol. 23, no. 10, pp. 1144-1151.

16. Bilgaeva L. P., Sambyalov Z. G. Ocenka kachestva algoritmov klasterizacii [Evaluation of the quality of clustering algorithms]. *Vestnik Vostochno-Sibirskogo gosudarstvennogo universiteta tekhnologij i upravleniya*, 2013, no. 6, pp. 53-60 (in Russian).

18. Nashivochnikov N. V., Pustarnakov V. F. Topologicheskie metody analiza v sistemah povedencheskoj analitiki [Topological methods of analysis in behavioral analytics systems]. *Voprosy kiberbezopasnosti*, 2021, no. 2, pp. 26-36. DOI: 10.21681/2311-3456-2021-2-26-36 (in Russian).

Статья поступила 05.12.2021 г.

Информация об авторах

Будко Никита Павлович – соискатель ученой степени кандидата технических наук. Независимый специалист. Область научных интересов: мониторинг информационных ресурсов; сбор и обработка информации. E–mail: budko62@mail.ru

Адрес: 194064, г. Санкт-Петербург, ул. Бутлерова, 9, корп. 3, кв. 252.

Васильев Николай Владимирович – кандидат технических наук. Начальник сектора. Публичное акционерное общество «Информационные телекоммуникационные технологии». Область научных интересов: мониторинг информационных ресурсов; сбор и обработка информации. E–mail: gandvik1984@gmail.com

Адрес: 197342, Россия, г. Санкт-Петербург, ул. Кантемировская, д. 8.

Review of graph-analytical approaches to monitoring of information and telecommunication networks and their application to identify abnormal states

N. P. Budko, N. V. Vasiliev

Task statement: *Problem statement: modern approaches to monitoring telecommunications networks are focused on measuring the state of individual devices and services, leaving behind the scenes the state of the network "as a whole" as a single monitoring object. As a consequence, for sufficiently large telecommunication networks, especially in the case when the network element is present in the measurement for*

*an insignificant time compared to the period of existence of the entire network, it becomes impossible to analyze the monitoring results. Despite the increasing prevalence of multidimensional data analysis methods in monitoring systems, the task of visual representation of the "health of the network" is more than relevant. Moreover, unlike multidimensional data cubes, the interpretation and analysis of which is akin to art, methods based on the editorial distance of graphs make it possible to visually visualize the dynamics of not only the entire network, but also its individual fragment, which requires less qualification of personnel operating telecommunication systems. The lack of methods for identifying the state depending on the degree of topology change has long hindered the practical application of methods based on the distance of graph editing. However, the development of the theoretical basis, in particular, the proof of results in the field of average sequence graphs and clustering methods, makes it possible to remove the requirement of preliminary normalization of the measure of topology change, replacing it with the concept of a "cluster of state". **The purpose of the work** is to develop, on the basis of a scientific and methodological apparatus for evaluating graph sequences, a technique for detecting abnormal states of a telecommunications network by analyzing the degree of change in the topology of a monitoring object. **Methods used:** methods of computational graph theory; methods of data analysis (cluster analysis); linear algebra and spectral graph theory; methods of behavioral analytics; network monitoring technologies as a set of engineering practices that support reliable and trouble-free operation of applications in the present and future; Operation Support Systems, as a technology to support operations; methods of system analysis, structural synthesis, prediction theory, diagnostic theory, classification theory. **The novelty of the work:** the novelty of the research is determined by the completeness of the analysis of existing graph-analytical approaches and the use of theoretical results on the averaging of graph sequences within the k-means algorithm to form an adaptive classification methodology for assessing the state of a telecommunications network. **Result:** based on a preliminary analysis of the results in the field of graph monitoring of telecommunication networks, as well as graph clustering methods, an approach to identifying abnormal states of a telecommunications network is proposed. The proposed approach has been tested on real data of a botnet attack on a telecommunications network and shows a fairly clear identification of the periods when the network is in various states, such as: attack, normal and transient modes.*

Keywords: *graph editing distance, average graph, graph spectrum, graph clustering, information and telecommunication network, network monitoring subsystem.*

Information about Authors

Nikita Pavlovich Budko – Doctoral Student. An independent specialist. Field of research: information monitoring; data acquisition. E–mail: budko62@mail.ru

Address: 194064, Russia, St. Petersburg, Butlerova str., build. 9/3, sq. 252.

Nikolay Vladimirovich Vasiliev – Ph.D. of Engineering Sciences. The head of the sector. Public Joint Stock Company "Information Telecommunications Technologies". Field of research: information monitoring; data acquisition. E–mail: gandvik1984@gmail.com

Address: 197342, Russia, St. Petersburg, Kantemirovskaya St. 8.