3. Valiant L. G. A bridging model for parallel computation // Communications of the ASM. 1990. № 33 (8). P. 103–111.

4. Buurlage J.-W., Bannink T., Wits A. Bulk-synchronous pseudo-streaming algorithms for many-core accelerators. 2016. URL: http://arXiv.org/abs/1608.07200 (дата обращения 24.03.2021).

5. McColl W. F., Tiskin A. Memory-efficient matrix multiplication in the BSP model // Algorithmica. 1999. № 24 (3–4). P. 287–297.

6. Масальских А. В. Параллельный алгоритм одного метода восстановления табличных данных // Изв. Тульского гос. ун-та. Естественные науки. 2014. № 3. С. 167–177.

7. Сударева О. Ю. Встречная оптимизация класса задач трехмерного моделирования для архитектур многоядерных процессоров. Дис. … канд. физ.-мат. наук: 05.13.11. М.: Ин-т систем. программирования им. В. П. Иванникова РАН, 2018.

A. R. Liss
Saint Petersburg Electrotechnical University

G. Yu. Puerov, E. I. Sergeeva
JSC «Concern «Oceanpribor» (Saint Petersburg)

EFFICIENT PARALLEL ALGORITHMS FOR DATA RECOVERY ON UNIFORM GRID

The article is devoted to the computational aspects of real time data recovery problem. One of the known approaches to solving the problem is described. This approach has low computational complexity and good potential in data parallelism. Parallel algorithms for data recovery from known values at the nodes on uniform grid are considered. Such algorithms are widely used in various applied fields of information processing. Approaches to data-parallel computations in bulk-synchronous notation are studied and bulk-synchronous parallel (BSP) model is extended for real-time streaming data processing. A model of bulk-synchronous-pipeline parallelism (BSPP) is built for streaming processing in real time, which is based on the well-known general-purpose parallel computing model BSP. Parallel algorithms for data recovery are described in terms of this extended model. Computational and communicational complexity of the proposed algorithms is assessed. Dependences between the dimensions of the problem and the parameters of the extended model are derived. The software that implements proposed algorithms is described.

Data recovery, multiprocessor system, bulk-synchronous parallelism (BSP), real-time system, signal processing, parallel prosesing, software desing

УДК 004.56

Е. О. Кузнецова, С. А. Петренко
Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В. И. Ульянова (Ленина)

Подход к обнаружению кибератак на основе вычислительного когнитивизма

Исследована комплексная научная задача обнаружения кибератак на отечественные ведомственные и корпоративные информационные системы. Предложен и обоснован подход к созданию систем обнаружения кибератак на основе так называемого вычислительного когнитивизма – сравнительно нового научного направления исследований, в котором познание и когнитивные процессы – это разновидность символьного вычисления. Показано, что когнитивный подход позволяет создавать системы, принципиально отличающиеся от известных систем обнаружения, предупреждения и ликвидаций последствий компьютерных атак уникальной способностью к самостоятельному ассоциированию и синтезу новых знаний о качественных характеристиках и количественных закономерностях информационного противодействия в киберпространстве. Предложен системный облик когнитивной системы обнаружения кибератак на основе вычислительного когнитивизма.

Информационная безопасность, обнаружение кибератак, система обнаружения кибератак, вычислительный когнитивизм, количественные закономерности противодействия в киберпространстве, конвергентные NBIC-технологии, технологии Big Data

В Российской Федерации уже создан ряд государственных и корпоративных центров обнаружения, предупреждения и ликвидации последствий компьютерных атак (СОПКА) или центров реаги-

рования на инциденты компьютерной безопасности, которые по своей функциональности аналогичны зарубежным CERT/CSIRT/SOC. В отечественной практике они известны как СОПКА или СПОКА, например: GOV-CERT.RU (ФСБ России), СПОКА Минобороны, FinCERT (Банк России), CERT Ростехнологий, SOC Газпрома и пр.

Указом Президента РФ от 15.01.2013 № 31 с «О создании государственной системы обнаружения, предупреждения и ликвидации последствий компьютерных атак на информационные ресурсы Российской Федерации» установлено, что ФСБ России разрабатывает методические рекомендации по организации защиты критической информационной инфраструктуры Российской Федерации и организует работы по созданию государственной системы обнаружения, предупреждения и ликвидации последствий компьютерных атак. Концепцией государственной системы обнаружения, предупреждения и ликвидации компьютерных атак на информационные ресурсы Российской Федерации, утвержденной Президентом РФ 12 декабря 2014 г. № К 1274, определена архитектура государственной СОПКА. Основу СОПКА составляют специальные центры обнаружения, предупреждения и ликвидации последствий компьютерных атак, которые подразделяются на центры: ФСБ России, предназначенные для защиты информационных ресурсов органов государственной власти; государственных и коммерческих организаций, предназначенные для защиты собственных информационных ресурсов. При этом упомянутые центры координируются Национальным координационным центром по компьютерным преступлениям при ФСБ России.

В основе корпоративных сегментов СОПКА лежат системные решения отечественных производителей средств защиты информации – как правило, на основе открытого программного кода IDS/IPS Snort, Suricata, Zeek и др., например системы обнаружения кибератак (СОА) АПКШ «Континент», ViPNet IDS 2, ViPNet IDS HS и т. п. Отметим, что СОА Snort и Suricata обладают сопоставимым функционалом, а также схожими наборами правил обнаружения и базами сигнатур кибератак [1], [2]. Так, научно-исследовательский проект Snort развивается с 1998 г. и в настоящее время стал де-факто техническим стандартом для большинства известных решений СОА (рис. 1).

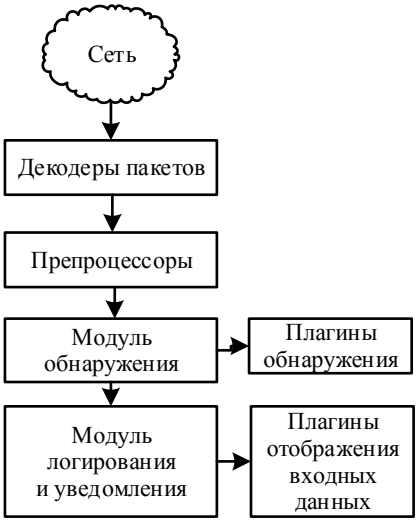

Рис. 1

СОА Suricata развивается с 2010 г. фондом «Open Information Security Foundation» (OISF) и построен на основе многопоточной архитектуры (рис. 2), т. е. этот научно-исследовательский проект можно считать улучшенным вариантом СОА Snort [1].

Отметим, что СОА Snort и Suricata построены на основе клиент-серверной архитектуры, т. е. клиенты упомянутых СОА могут быть установлены на хостах корпоративной системы. Это достаточно важно для обеспечения требуемой производительности до 10 Гбит/с.

Были оценены заявленные характеристики производительности СОА Snort и Suricata. Для этого на специально созданном стенде на основе Oracle Virtual Box (рис. 3) были проведены измерения загрузки процессоров, CPU (central processing unit) (в процентах), объема памяти и ресурсоемкости сети в целом, а также общие количества пропущенных сетевых пакетов. Для генерации требуемого сетевого трафика использовался генератор с открытым исходным кодом Ostinato [3], который способен генерировать сетевой трафик до 20 Гбит/с. Для наблюдения и записи измерений CPU, памяти, использования сети и ско-

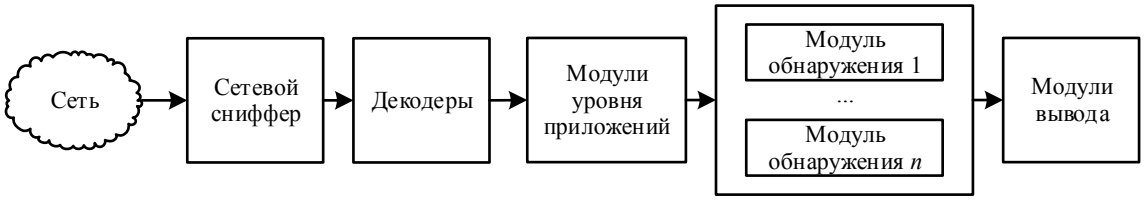

Рис. 2

рости отбрасывания пакетов использовались Collectl, top, dstat, журналы Snort, журналы Suricata tcpdump, IPTRAF, nmapetc.

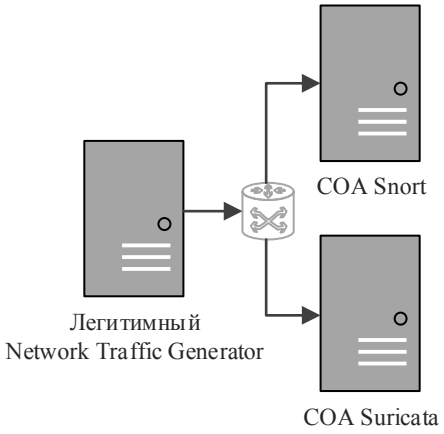

Рис. 3

Были созданы четыре виртуальные машины (ВМ) с характеристиками, представленными в табл. 1. Для агрегации виртуальных машин был задействован виртуальный коммутатор с каналами связи Ethernet 10 Гбит/с. В ходе экспериментов были использованы стандартные правила обнаружения кибератак Snort и Suricata.

Результаты экспериментов показали следующее.

Использование CPU для Snort и Suricata при различных скоростях сбора и обработки трафика. В этом эксперименте на каждую COA были введены 1 000 000 пакетов по 1470 байт для TCP, UDP и ICMP. Скорость подачи пакетов составляла 500 пакетов/с, скорость подачи фонового трафика – 1…10 Гбит/с. Каждая COA была отдельно установлена на идентичных виртуальных машинах с параметрами производительности и набором правил по умолчанию. Эксперимент длился 12 ч, по три четырехчасовых интервала. Данные, собранные в ходе первого эксперимента, показали, что загрузка CPU Suricata была выше, чем у Snort, при обработке тех же 10 Гбит/с сетевого трафика. На рис. 4 представлены результаты сравнения использования CPU для Snort и Suricata для разных скоростей трафика (среднее).

Использование памяти для Snort и Suricata для разных скоростей трафика. В ходе эксперимента также одинаковое количество (1 000 000) пакетов UDP, TCP и ICMP подавалось на обе COA в течение 12 ч (т. е. три четырехчасовых интервала). Исходя из результатов использования памяти, для Snort и Suricata для разных скоростей трафика, представленных на рис. 5, Snort показал значительное преимущество над Suricata. Среднее использование памяти Snort было сравнительно меньше, начиная с 2.7 Гбайт при 1 Гбит/с, и продолжало работать с уменьшенным использованием

Таблица 1

ВМ/Тип ПО	Спецификация	Используемые утилиты
Astra Linux Special Edition релиз Смоленск 1.6	Virtual Machine; 2.5 ГГц, 4 ядра CPU; 4 Гбайт оператив. памяти; 10 Гбит/с Ethernet	Snort 2.9.6.1 COA; Collectl, top, dstat, Snort logs, tcpdump, IPTRAF
Astra Linux Special Edition релиз Смоленск 1.6	Virtual Machine; 2.5 ГГц, 4 ядра CPU; 4 Гбайт оператив. памяти; 10 Гбит/с Ethernet	Suricata 4.1.9 COA;Collectl, top, dstat, Suricata logs, tcpdump, IPTRAF
Легитимный Network Traffic Generator	Virtual Machine; 2 ядра CPU; 2 Гбайт оператив. памяти; 10 Гбит/с Ethernet	Ostinato

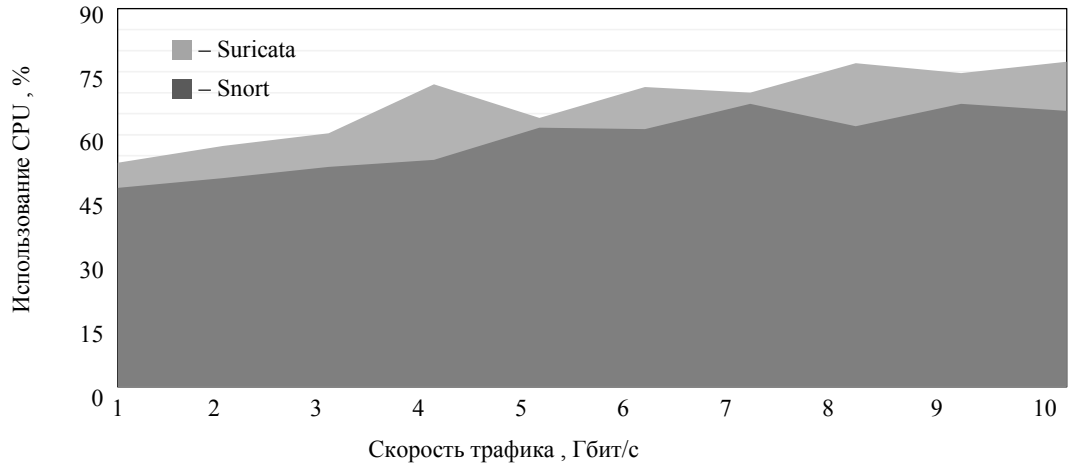

Рис. 4

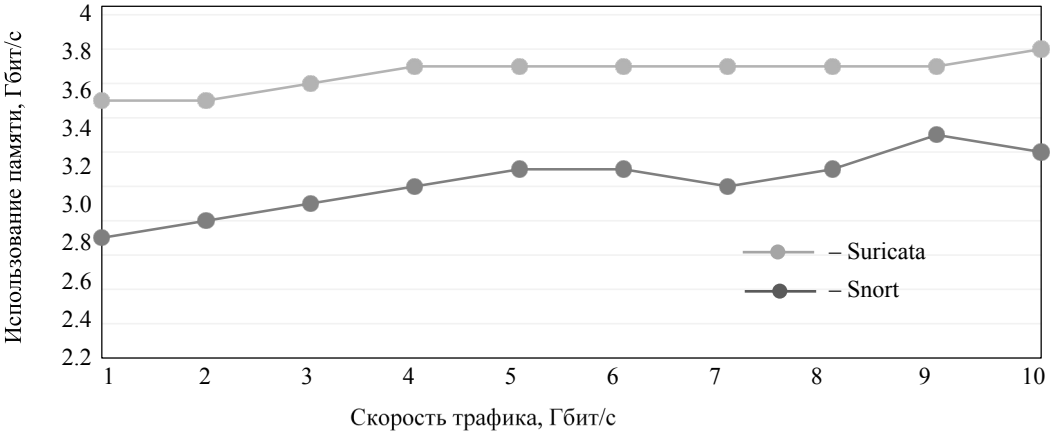

Рис. 5

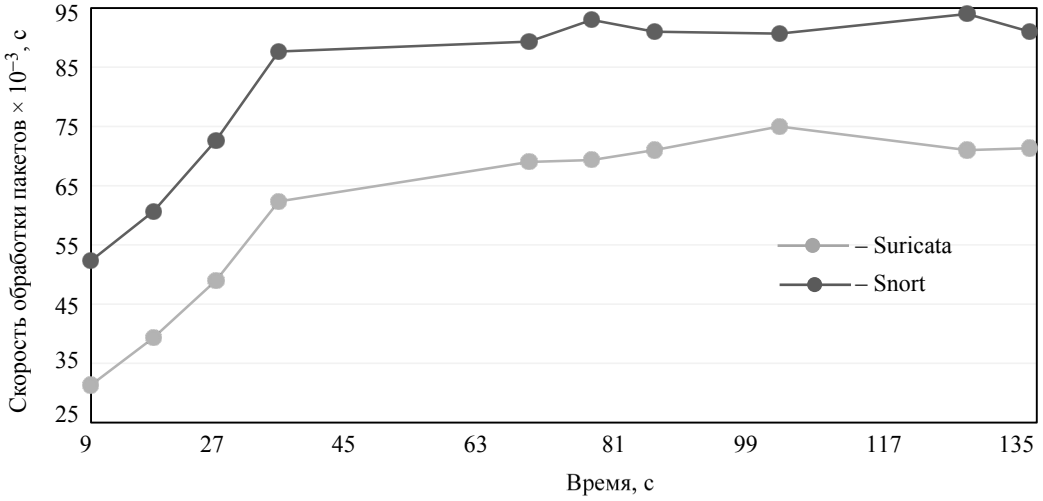

Рис. 6

памяти на всех скоростях сети, достигая высшей точки в использовании памяти 3.2 Гбайт при обработке на скорости сети 10 Гбит/с. Вероятно, такое использование памяти Suricata больше связано с многопоточной архитектурой.

Скорость обработки пакетов Snort и Suricata. Механизм обработки пакетов Snort был медленнее, чем механизм обработки Suricata. Одинаковое количество (1 000 000) пакетов UDP, TCP и ICMP было подано в обе СОА, как и в предыдущих экспериментах, в течение 12 ч. Suricata показала превосходную производительность по сравнению с Snort при обработке большого количества пакетов. Средняя скорость обработки составила 82 223 пак./с в течение 12 ч, тогда как средняя скорость обработки Snort составила 60 866 пак./с за тот же период. На рис. 6 показывает преимущества Suricata в скорости обработки пакетов над Snort.

Среднее количество пакетов, пропускаемых Snort и Suricata при разной скорости сети. В табл. 2 несложно отследить, что, как и в предыдущем эксперименте, Snort оказался менее устойчивым к высокой скорости сети, нежели Suricata, и при скорости сети 10 Гбит/с пропускает больше пакетов, чем Suricata. Эксперимент проводился на том же количестве пакетов в трех блоках по 5 ч, и среднее значение было взято из трех значений для UDP, TCP и ICMP для обеих СОА.

Таблица 2

Скорость сети, Гбит/с	Пропущенные пакеты, %					
	UDP		TCP		ICMP	
	Snort	Suricata	Snort	Suricata	Snort	Suricata
1	3.2	1.5	1.5	0	7.0	3.5
2	4.1	2.0	3.1	1.0	8.2	4.1
3	4.8	2.5	4.9	2.2	9.8	4.9
4	6.3	2.9	7.2	3.6	10.5	5.5
5	7.5	3.7	9.5	4.1	11.5	6.2
6	9.6	5.0	12.1	5.3	13.0	6.9
7	10.8	5.9	14.8	6.2	14.2	8.0
8	11.7	6.6	16.4	7.4	15.6	8.6
9	12.8	7.2	18.2	8.0	16.1	9.1
10	14.0	8.0	20.0	9.0	17.0	10.0

В результате было показано, что СОА Snort использует меньше вычислительных ресурсов для обработки сетевого трафика со скоростью 10 Гбит/с, а СОА Suricata требуется больше памяти и ресурсов CPU. При этом СОА Suricata отличается меньшим количеством пропущенных пакетов при скорости сбора и обработки в 10 Гбит/с. Общий и существенный недостаток упомянутых СОА – это неспособность обнаруживать ранее неизвестные кибератаки (примерно 55 из 100 % всех возможных кибератак). СОА Snort и Suricata оказались неспособными к адаптивности и самоорганизации в условиях роста угроз безопасности, поэтому была поставлена научная задача разработки новых моделей, методов и средств СОА, способных к *самостоятельному познанию и поведению,* в том числе для решения следующих актуальных задач:

– распознавание образов (паттернов и кластеров) как известных, так и неизвестных ранее кибератак;

– обучение и выработка типовых сценариев предупреждения, обнаружения и противодействия в киберпространстве;

– порождение, накопление и обработка новых знаний о количественных закономерностях информационного противодействия в киберпространстве;

– представление «глубокой» семантики обнаружения и предупреждения кибератак;

– подготовка и реализация ответных сценариев предупреждения кибератак и пр.

Научный задел для решения задачи. Термин «когнитивный» происходит от *лат*. «cognitio» – познание. Впервые когнитивный подход упоминается в монографии Ульрика Найссера «Когнитивная психология» [4], в которой процессы мозговой активности рассмотрены как процессы обработки информации. Дальнейшее развитие математических моделей мыслительных процессов способствовало развитию когнитивного подхода в технической сфере. Появились первые «искусственные когнитивные системы», представляющие собой «интеллектуальные» программно-аппаратные комплексы на основе традиционной архитектуры Дж. фон Неймана [5], [6].

В основе современного когнитивного подхода находятся методы познания, восприятия и накопления информации, а также методы мышления или использования этой информации для «рассудительного» решения задач. Считается, что искусственные когнитивные системы способны «повторить» сложные поведенческие функции нервной системы и даже мыслительные процессы человека. Этим, собственно, и объясняется интерес к когнитивному подходу для решения задачи обнаружения и предупреждения кибератак на критически важные информационные ресурсы Российской Федерации.

Существенный вклад в становление и развитие современного когнитивного подхода внесли труды ведущих отечественных и зарубежных ученых:

– в области теоретической информатики и искусственного интеллекта (А. Н. Колмогоров, А. И. Мальцев, Г. С. Поспелов, Д. А. Поспелов, Э. В. Попов, Г. С. Осипов, О. П. Кузнецов, В. К. Финн, В. Н. Вагин, А. П. Еремеев, В. М. Курейчик, В. Л. Стефанюк, И. Б. Фоминых, Л. Заде, Н. Нильсон, Г. Саймон, Дж. Люгер, С. Рассел и П. Норвиг);

– по специальным вопросам информатики и искусственного интеллекта – онтологии, псевдофизические логики, модели пространственных рассуждений (Ст. Лесьневский, А. Тарский, Т. Грубер, Н. Гуарино, Р. Мизогучи, А. Варзи, Л. Вье, В. Куйперс, Б. Смит, М. Айелло, Д. А. Поспелов, Г. С. Плесневич, Т. А. Гаврилова, А. С. Клещев, И. Л. Артемьева, Л. С. Массель, Т. В. Ворожцова и др);

– по моделям и методам диалоговых систем (Т. Виноград, Д. А. Поспелов, Э. В. Попов, А. П. Ершов, Л. Т. Кузин, А. Б. Преображенский, Г. В. Рыбина, А. С. Нариньяни, П. И. Соснин и др.);

– в области теории нечетких множеств, лингвистических переменных и грануляции информации (Л. Заде, Д. Дюбуа, А. Прад, В. Педрич, Б. Турксен, В. Новак, Ю. Яо, А. Н. Аверкин, Р. А. Алиев, И. З. Батыршин, Н. Г. Ярушкина, С. М. Ковалев, А. Б. Петровский и др.);

– по теории агентов и многоагентных систем (К. Хьюитт, М. Вулдридж, Н. Дженнингс, М. Дженесерет, Ж. Фербе, В. Субраманьян, И. Шоэм, В. И. Городецкий, Р. М. Юсупов, И. В. Котенко, П. О. Скобелев, Л. А. Станкевич, В. Ф. Хорошевский, В. Б. Тарасов и др.);

– в области интеллектуальных роботов и систем управления роботами (Е. П. Попов,

Д. Е. Охоцимский, И. М. Макаров, И. А. Каляев, Ф. Куафе, В. М. Лохин, Э. Накано, В. Е. Павловский, А. К. Платонов, А. В. Тимофеев, Е. И. Юревич, А. С. Ющенко, Д. А. Добрынин, В. Э. Карпов и др.);

– по специальным вопросам кибербезопасности и обеспечения устойчивости функционирования критически важных информационных систем национальной инфраструктуры (А. Г. Ломако, М. А. Еремеев, С. А. Петренко, Д. Н. Бирюков, В. М. Зима, В. А. Новиков, И. Е. Горбачев, С. В. Пилькевич, В. А. Овчаров, А. А. Платонов и др.).

Предпосылками современного когнитивного подхода послужили фундаментальные результаты:

– математической логики (от Аристотеля до А. Н. Колмогорова);

– математической теории вычислимости (от А. Тьюринга до А. И. Мальцева);

– теории вычислительных машин архитектуры Дж. фон Неймана;

– теории порождающих грамматик Н. Хомского;

– теории вычислительной нейрофизики Дж. Марра и пр.

При этом основа так называемого вычислительного когнитивизма была сформирована следующими выдающимися учеными.

Аристотель (384–322 гг. до н. э.) ввел основные термины и определения логики, сформулировал ряд умозаключений и доказательств, ввел понятие «логическая операция», определил основные законы мышления, в том числе законы противоречия и исключения третьего [7], тем самым предприняв попытку свести размышление, или, точнее, умозаключение, к вычислению на основании исходных положений. В результате вплотную приблизился к одному из разделов будущей математической логики – теории доказательств.

Готфрид Вильгельм Лейбниц (1646–1716 гг.) предложил путь для перевода логики «из словесного царства, полного неопределенностей, в царство математики, где отношения между объектами или высказываниями определяются совершенно точно» [8]. Другими словами, предложил ученым перейти от бесплодных споров к вычислениям для доказательства своей правоты.

Джордж Буль (1815–1864 гг.) разработал систему обозначений и правил, применимую ко всевозможным объектам, от чисел и букв до предложений, которая позволила представить высказывания в виде утверждений, истинность или ложность которых требуется доказать [9], [10]. В результате были заложены основы современной математической логики, главным образом ее основные разделы, алгебра логики (булева алгебра) и алгебра высказываний.

Алан Мэтисон Тьюринг (1912–1954 гг.) ввел понятие алгоритма, которое позволило описывать любую вычислимую (или «эффективно вычислимую») процедуру, сводя ее к заданному набору элементарных «механических» составляющих шагов и операций. Одновременно с А. Тьюрингом эквивалентные схемы были предложены А. Черчем и Э. Постом [9], [10]. По мнению А. Тьюринга должным образом запрограммированная универсальная цифровая вычислительная машина в принципе способна демонстрировать интеллектуальное поведение, в том числе проигрывать «имитацию» (*англ.* «imitation game») – тест Тьюринга, по результатам которого машина, отвечающая на вопросы человека, может быть принята за разумное существо.

Дальнейшее развитие классической пропозициональной логики (Дж. Буль, Г. Фреге и Б. Рассел) и математической теории вычислимости (А. Тьюринг, А. Черч и Э. Пост) позволило формализовать идеи Т. Гоббса и Г. Лейбница о том, что мыслительные процессы могут быть представлены как вычислительные и алгоритмические.

Аллен Ньюэлл (1927–1992 гг.) и *Герберт Александер Саймон* (1916–2001 гг.) ввели понятие физической символьной системы, которая представляет собой реализацию универсальной машины, «производящей с ходом времени изменяющийся набор символьных структур» [9]. Была выдвинута гипотеза, что подобные физические символьные системы обладают всеми необходимыми и достаточными условиями для общего интеллектуального действия.

Аврам Ноам Хомский (род. 1928) предложил теорию порождающих грамматик [11], в которой проводились фундаментальные различия между языковой компетенцией (знанием языка идеальным говорящим-слушающим) и реальным употреблением языка, глубинным и поверхностным уровнями синтаксического представления, постулировалось наличие врожденных грамматических структур.

Дэвид Кортни Марр (1945–1980 гг.) предложил основы «вычислительной нейронауки» (*англ.* computational neuroscience) и существенно развил вычислительные основы зрения и мозга [12]. Со-

гласно Д. Марру для понимания процесса обработки информации необходимо учитывать информационную теорию вычислительных устройств. В частности, представление вычислительного устройства, алгоритм машинных вычислений, а также физическую (техническую) реализацию выбранного представления и алгоритма. В результате была предложена универсальная схема анализа процессов обработки информации естественными и искусственными техническими системами.

Андрей Николаевич Колмогоров (1903–1987 гг.) существенно развил такие понятия как алгоритм, автомат, случайность, энтропия, информация, сложность.

А. Н. Колмогоровым был выдвинут важнейший методологический тезис искусственной жизни: «если свойство той или иной материальной системы «быть живой» или обладать способностью «мыслить» будет определено чисто функциональным образом (например, любая система, с которой можно обсуждать проблемы современной науки и литературы будет признаваться мыслящей), то придется признать в принципе вполне осуществимым искусственное создание живых и мыслящих существ» [10].

По мнению А. Н. Колмогорова переработка информации и процессы управления в живых организмах построены на сложном переплетении:

а) дискретных (цифровых) и непрерывных механизмов;

б) детерминистического и вероятностного принципов действия.

При этом дискретные механизмы являются ведущими: «Не существует состоятельных аргументов в пользу принципиальной ограниченности возможностей дискретных механизмов по сравнению с непрерывными. При анализе явлений жизни существенна не диалектика бесконечного, а диалектика большого (чисто арифметическая комбинация большого числа элементов создает и непрерывность, и новые качества)» [10].

Современные исследования когнитивных систем проводятся на основе нейрофизиологических принципов построения нервной системы и когнитивных методов познавательной и мыслительной деятельности человека. Например, в работах Л. А. Станкевича [13] обосновано применение искусственных когнитивных систем с гибридными архитектурами в робототехнике. При этом когнитивная система определена как система, которая способна познавать свое окружение и адаптироваться к нему или изменять его за счет накопленных в процессе функционирования знаний и приобретенных навыков. Явно выделены два основных типа искусственных когнитивных систем: собственно когнитивные и эмерджентные (см. рис. 7).

К собственно когнитивным системам отнесены:

– традиционные символьные системы (Ньюэл и Саймон);

– системы на основе теории познания, которые используют обучение и приобретение символьных знаний (Андерсон);

– системы на основе теории практического вывода и высокоуровневых психологических концепций убеждения, желания, намерения (Братман).

Здесь первые способны порождать некоторые символьные структуры или выражения. При этом под символом понимается физический паттерн, представляющий некоторый компонент выражения (или символьной структуры). Вторые основаны на системе продукций и обобщенной модели мышления и познания человека, содержащей память, знания, принятие решений, обучение и пр. При этом обучение содержит декларативный и процедурный этапы в зависимости от знаний обучаемого. Третьи реализуют процесс принятия решения, аналогичный традиционному практическому выводу.

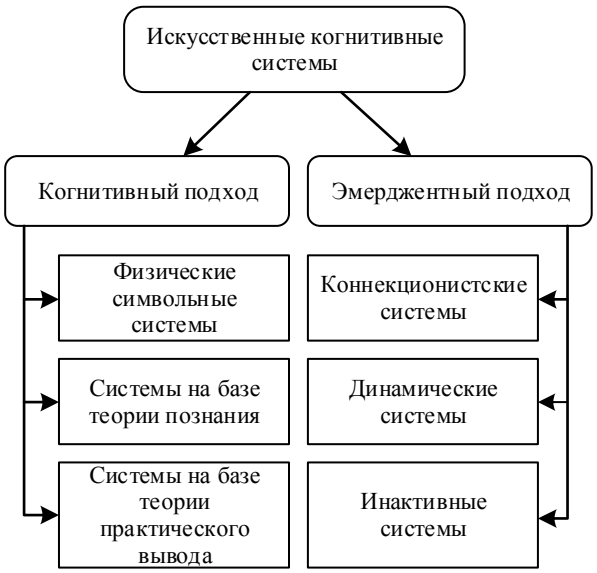

Рис. 7

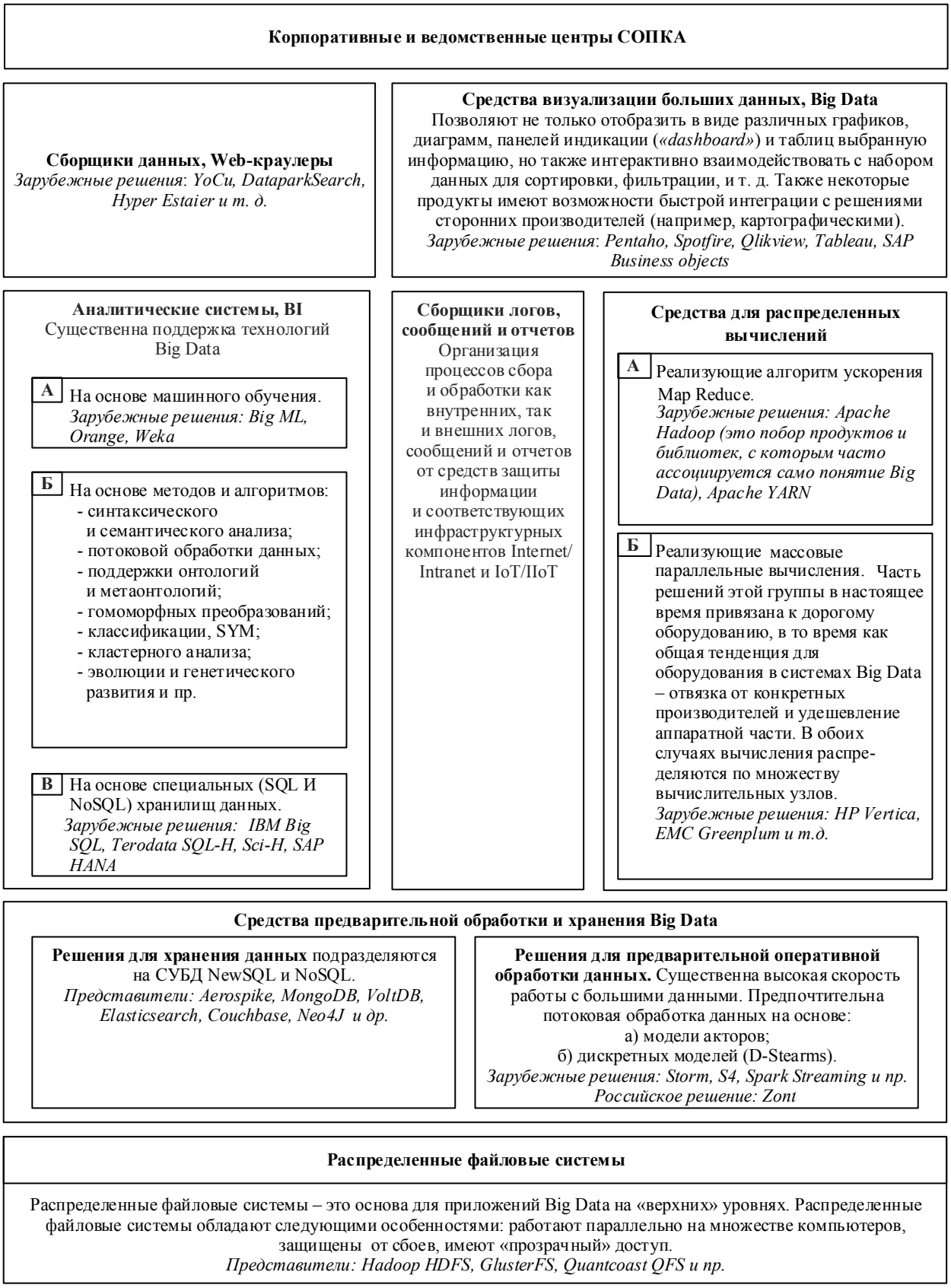
Корпоративные и ведомственные центры СОПКА
Сборщики данных, Web-краулеры
Зарубежные решения: YoCu, DataparkSearch, Hyper Estaier и т. д.
Средства визуализации больших данных, Big Data
Позволяют не только отобразить в виде различных графиков, диаграмм, панелей индикации («dashboard») и таблиц выбранную информацию, но также интерактивно взаимодействовать с набором данных для сортировки, фильтрации, и т. д. Также некоторые продукты имеют возможности быстрой интеграции с решениями сторонних производителей (например, картографическими).
Зарубежные решения: Pentaho, Spotfire, Qlikview, Tableau, SAP Business objects
Аналитические системы, BI
Существенна поддержка технологий Big Data
А
На основе машинного обучения.
Зарубежные решения: Big ML, Orange, Weka
Б
На основе методов и алгоритмов:
- синтаксического и семантического анализа;
- потоковой обработки данных;
- поддержки онтологий и метаонтологий;
- гомоморфных преобразований;
- классификации, SYM;
- кластерного анализа;
- эволюции и генетического развития и пр.
В
На основе специальных (SQL И NoSQL) хранилищ данных.
Зарубежные решения: IBM Big SQL, Terodata SQL-H, Sci-H, SAP HANA
Сборщики логов, сообщений и отчетов
Организация процессов сбора и обработки как внутренних, так и внешних логов, сообщений и отчетов от средств защиты информации и соответствующих инфраструктурных компонентов Internet/ Intranet и IoT/IIoT
Средства для распределенных вычислений
А
Реализующие алгоритм ускорения Map Reduce.
Зарубежные решения: Apache Hadoop (это побор продуктов и библиотек, с которым часто ассоциируется само понятие Big Data), Apache YARN
Б
Реализующие массовые параллельные вычисления. Часть решений этой группы в настоящее время привязана к дорогому оборудованию, в то время как общая тенденция для оборудования в системах Big Data – отвязка от конкретных производителей и удешевление аппаратной части. В обоих случаях вычисления распределяются по множеству вычислительных узлов.
Зарубежные решения: HP Vertica, EMC Greenplum и т.д.
Средства предварительной обработки и хранения Big Data
Решения для хранения данных подразделяются на СУБД NewSQL и NoSQL.
Представители: Aerospike, MongoDB, VoltDB, Elasticsearch, Couchbase, Neo4J и др.
Решения для предварительной оперативной обработки данных. Существенна высокая скорость работы с большими данными. Предпочтительна потоковая обработка данных на основе:
а) модели акторов;
б) дискретных моделей (D-Stearms).
Зарубежные решения: Storm, S4, Spark Streaming и пр.
Российское решение: Zont
Распределенные файловые системы
Распределенные файловые системы – это основа для приложений Big Data на «верхних» уровнях. Распределенные файловые системы обладают следующими особенностями: работают параллельно на множестве компьютеров, защищены от сбоев, имеют «прозрачный» доступ.
Представители: Hadoop HDFS, GlusterFS, Quantcoast QFS и пр.

Рис. 8

К эмерджентным системам отнесены:

– коннекционистские системы;

– динамические системы;

– инактивные системы.

Первые реализуют параллельную обработку распределенных паттернов активации, используя статистические свойства, а не логические правила. Вторые изучают различные самоорганизующиеся моторные системы и системы восприятия человека, разбирая соответствующие метастабильные паттерны поведения. В-третьих, определение когнитивной сущности, т. е. целенаправленного поведения системы, происходит, когда она взаимодействует со средой.

Таким образом, предлагается общая методология разработки гибридных когнитивных технических систем [13]:

1. Формализованные когнитивные концепции и методы создания эффективных самообучающихся и самомодифицирующихся систем.

2. Методы синтеза оригинальных когнитивных компонентов (модулей и сетей модулей), способных к накоплению знаний с помощью обучения и самообучения. При этом компоненты строятся на основе комбинации нейрологических, иммунологических и триангуляционных адаптивных элементов, наиболее эффективных для многомерной функциональной аппроксимации, а также соответствующих поведенческих сетей.

Методы реализации когнитивных компонентов и систем на основе специально разработанных программных средств. Программная реализация когнитивных компонентов базируется на оригинальных моделях обработки информации и обучения, а когнитивных систем – на основе многоагентной технологии. Здесь когнитивная многоагентность позволяет создать распределенные когнитивные системы с высоким уровнем сложности поведения.

В работах Тарасова В. Б. [6] представлена система онтологий для когнитивных агентов на примере мобильных роботов. Построена метаон-

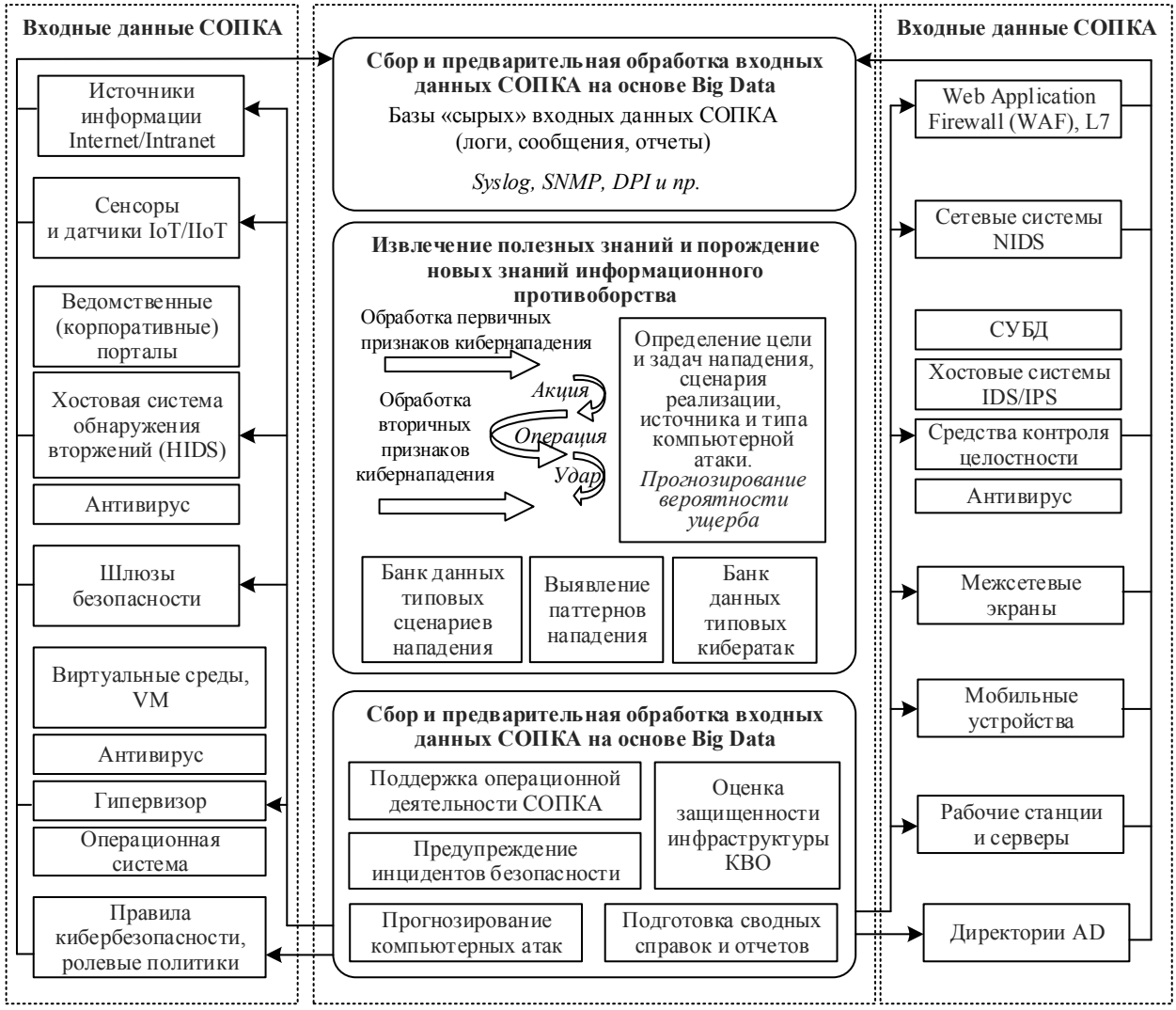

Рис. 9

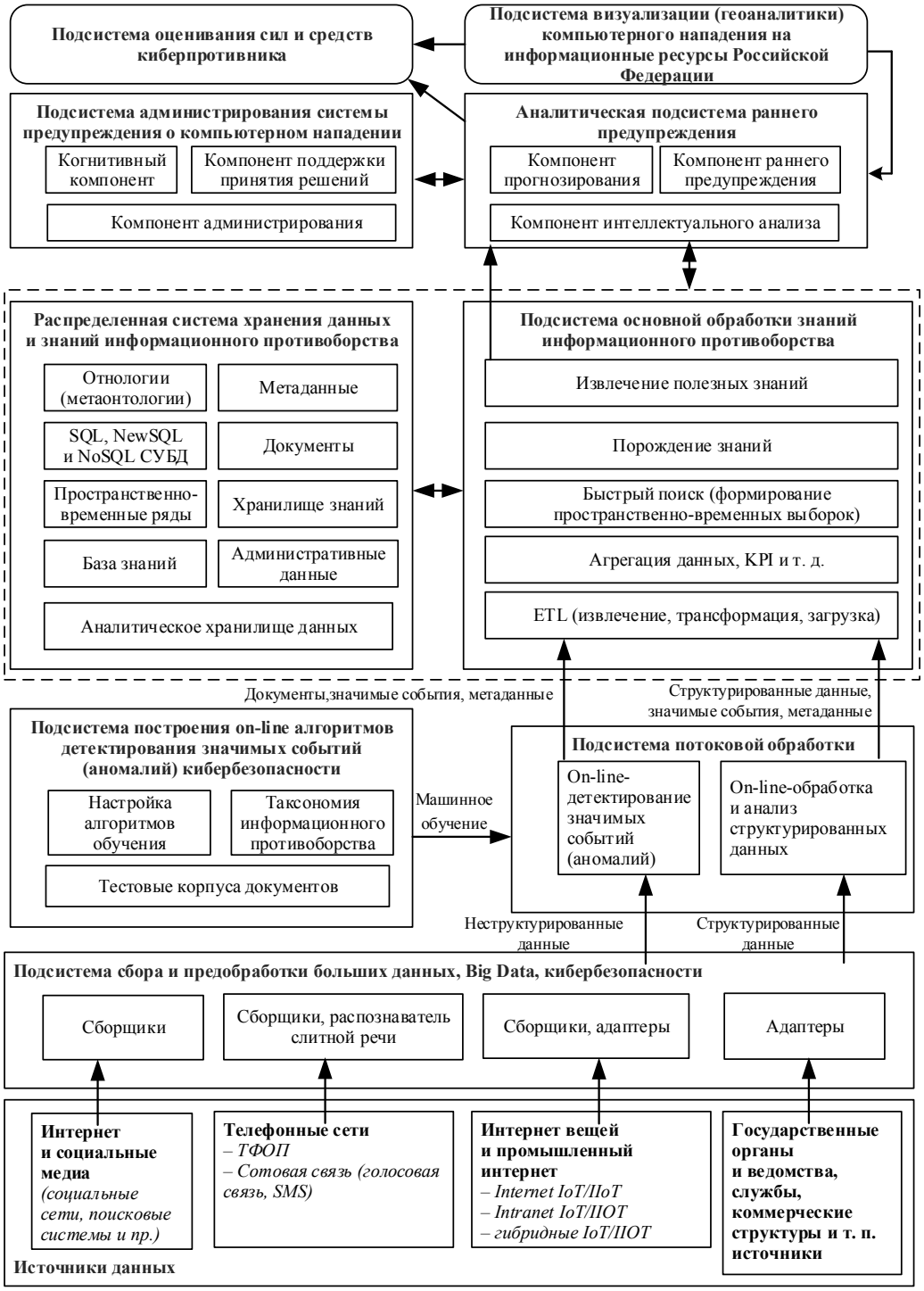

Рис. 10

тология грануляции информации. Дано представление онтологий на базе нечетких алгебраических систем. Исследованы пространственные отношения для организации диалогового управления когнитивными мобильными роботами. Описан мереотопологический подход к анализу пространственных онтологий. Рассмотрены четкие и нечеткие мереотопологические пространственные отношения. Предложена гранулярная онтология пространства.

Технологический задел для решения задачи. В качестве технологической основы для решения поставленной задачи предлагается рассмотреть современные программно-аппаратные комплексы анализа и обработки событий информационной безопасности (см. рис. 8).

Вместе с тем, на практике создание когнитивной системы обнаружения и предупреждения кибератак на информационные ресурсы Российской

Когнитивная подсистема раннего предупреждения о компьютерном нападении на информационные ресурсы Российской Федерации

1. Развитие традиционной компоненты СОПКА на основе технологий Big Data

2. Создание аналитической компоненты СОПКА на основе «вычислительного когнитивизма»

3. Мониторинг состояния защищенности и устойчивости функционирования критической инфраструктуры в киберпространстве:
- введение паспорта КВО информатизации (инвентаризация, категорирование, классификация, определение требований и пр.);
- разработка планов первоочередных действий и пр.;

сертификация средств обеспечения информационной безопасности (аттестация объекта по требованиям безопасности);
- контроль критериев и показателей безопасности и устойчивости функционирования упомянутых объектов;
- ведение базы данных инцидентов кибербезопасности и пр.

4. Выявление первичных признаков кибернападения на информационные ресурсы Российской Федерации:
- распознавание структурных, инвариантных и корреляционных признаков кибернападения;
- пополнение базы данных первичных признаков кибернападения;
- уточнение сценариев кибернападения;
- выработка адекватных мер сдерживания, компенсации и пр.

5. Выявление вторичных признаков кибернападения на информационные ресурсы Российской Федерации:
- выявление корреляционных связей и зависимостей между признаками;
- пополнение базы данных вторичных признаков кибернападения;
- уточнение сценариев кибернападения;
- выработка адекватных мер сдерживания, компенсации и пр.

6. От обнаружения к предупреждению:
- раннее предупреждение о компьютерном нападении на информационные ресурсы РФ;
- прогнозирование компьютерного нападения киберврага;
- оценка вероятного ущерба в случае кибернападения;
- подготовка ответных сценариев сдерживания и принуждения к кибермиру

7. Извлечение полезных знаний и порождение новых знаний в области информационного противоборства на основе:
- новых NBIC-моделей:
  – нейроморфные, подобные организации живой нервной системы;
  – кортикоморфные, подобные организации коры головного мозга;
  – геноморфные, подобные генетическим и эпигенетическим механизмам размножения и развития живых организмов.
- моделей и методов математической логики и искусственного интеллекта:
  – когнитивные агенты;
  – искусственные нейронные сети прямого распространения, обучаемые по методу Левенберга–Марквардта;
  – обучаемые иерархически упорядоченные нейросети и бинарные нейросети;
  – разнообразные представления динамических порогов и классификаторов сетевых пакетов на основе метрики Евклида–Махаланобиса и метода опорных векторов;
  – статистические (корреляционные) и инвариантные профилировщики;
  – комплексные полимодельные представления и пр.

8. Разработка методических указаний по работе с когнитивной СОПКА

9. Организация киберучений для отработки навыков раннего предупреждения о компьютерном нападении на информационные ресурсы Российской Федерации

10. Разработка необходимых нормативных документов

11. Подготовка и переподготовка сотрудников по вопросам раннего предупреждения о компьютерном нападении на информационные ресурсы Российской Федерации

12. Выработка предложений по развитию национальной (и международной) нормативной базы в части раннего предупреждения о компьютерном нападении

Рис. 11

Федерации оказалось далеко не тривиальной задачей. Потребовалось провести соответствующие научные исследования и решить ряд сложных научно-технических задач. Например, такие задачи, как классификация входных данных, выявление первичных и вторичных признаков компью-

терного нападения, раннее обнаружение кибератак, многофакторное прогнозирование компьютерного нападения, моделирование распространения кибератак, обучение, порождение новых знаний о количественных закономерностях информационного противодействия в киберпространстве, не имели готовых стандартных решений. Кроме того, нужно обеспечить сбор, обработку, хранение сверхбольших объемов структурированной и неструктурированной информации от разнообразных источников Inernet/Intranet и IoT/IIoT (тематика Big Data и Big Data Analytics), а также проведение аналитических вычислений на них.

Возможная системная архитектура когнитивной системы обнаружения и предупреждения кибератак на основе вычислительного когнитивизма представлена на рис. 9, а на рис. 10 – технический компонент ПАК «Обнаружения кибератак – 2021».

Полученный положительный опыт разработки новых методов и средств СОА на основе вычислительного когнитивизма свидетельствует о целесообразности поэтапного решения возникших инженерных задач. Алгоритм создания и развития ПАК «Обнаружения кибератак – 2021» представлен на рис. 11.

Этап 1 – развитие технического компонента традиционной СОПКА на основе технологий больших данных, Big Data – создание высокопроизводительного корпоративного (ведомственного) сегмента обнаружения, предупреждения и ликвидации последствий компьютерных атак.

Этап 2 – создание аналитического компонента на основе «вычислительного когнитивизма» – реализация собственно когнитивного компонента системы раннего предупреждения о компьютерном нападении, способного самостоятельно извлекать и порождать полезные знания из больших объемов структурированной и неструктурированной информации для операционной поддержки СОПКА.

При этом упомянутый технический компонент СОПКА на основе технологий Big Data целесообразно наделить функциями:

– сбора больших данных о состоянии информационной безопасности в контролируемых информационных ресурсах;

– обнаружения и ликвидации последствий компьютерных атак на информационные ресурсы;

– поддержки средств программно-технического мониторинга событий информационной безопасности;

– взаимодействия с центрами государственной СОПКА;

– информирования по вопросам обнаружения, предупреждения компьютерных атак и ликвидации их последствий и пр.

Упомянутый аналитический компонент на основе «вычислительного когнитивизма» целесообразно наделить функциями:

– раннего предупреждения о компьютерном нападении на информационные ресурсы;

– выявления и порождения новых полезных знаний о качественных характеристиках и количественных закономерностях информационного противодействия в киберпространстве;

– прогнозирования инцидентов безопасности, вызванных известными и не известными ранее кобератаками;

– подготовки сценариев сдерживания киберпротивника и планирования ответных действий, адекватных компьютерной агрессии;

– подготовки шаблонов методических документов по вопросам раннего предупреждения, пресечения компьютерных атак и ликвидации их последствий, и пр.

Отдельного внимания требует рассмотрение вопроса о практике использования технологий больших данных Big Data для организации потоковой обработки данных кибербезопасности, а также практические вопросы семантического управления так называемыми мастер-данными кибербезопасности (Master Data Management, MDM) для предлагаемого ПАК «Обнаружение кибератак 2021».

СПИСОК ЛИТЕРАТУРЫ

1. Suricata User Guide. URL: https://suricata.readthedocs.io/en/suricata-4.1.9/ (дата обращения 03.02.2021).

2. SNORT Users Manual. URL: http://manual-snort-org.s3-website-us-east-1.amazonaws.com/ (дата обращения 03.02.2021).

3. Ostinato User Guide. URL: https://userguide.ostinato.org/ (дата обращения 03.02.2021).

4. Neisser U. Cognitive Psychology. New York: Appleton-Century-Crofts, 1967.

5. Величковский Б. М. Когнитивные технические системы // Компьютеры, мозг, познание: успехи когнитивных наук / отв. ред. Б. М. Величковский, В. Д. Соловьев. М.: Наука, 2008. С. 273–292.

6. Тарасов В. Б. От многоагентных систем к интеллектуальным организациям. Сер.: Науки об искусственном. М.: Эдиториал УРСС, 2002.

7. Аристотель. Соч.: в 4 т. Сер. Философское наследие. М.: Мысль, 1976–1983.

8. Лейбниц Г. В. Соч.: в 4 т. М.: Мысль, 1982.

9. Аналит. докл. «Подходы к формированию и запуску новых отраслей промышленности в контексте Национальной технологической инициативы, на примере сферы "Технологии и системы цифровой реальности и перспективные "человеко-компьютерные" интерфейсы (в части нейроэлектроники)». М.: Агентство стратегических инициатив – АСИ, 2015.

10. Колмогоров А. Н. Автоматы и жизнь // Кибернетика ожидаемая и кибернетика неожиданная / под ред. А. И. Берга, Э. Кольмана. Москва: Наука, 1968. С. 12–30.

11. Chomsky N. Syntactic Structures. The Hague: Mouton, 1957. (Переиздание: Chomsky N. Syntactic Structures. De Gruyter Mouton, 2002.)

12. Marr D. Vision. An informational investigation into the human representation and processing of visual information. Cambridge, Massachusetts, London, England: The MIT Press, 2010.

13. Станкевич Л. А. Искусственные когнитивные системы // Науч. сессия НИЯУ МИФИ. XII Всерос. науч.-техн. конф. «Нейроинформатика–2010». Лекции. М.: НИЯУ МИФИ, 2010. С. 106–160.

E. O. Kuznetsova, S. A. Petrenko
Saint Petersburg Electrotechnical University

COMPUTATIONAL COGNITIVISM-BASED APPROACH TO INTRUSION DETECTION

Explores the complex scientific problem of detecting intrusion in domestic departmental and corporate information systems. An approach to the creation of intrusion detection system based on the so-called «computational cognitivism» is proposed and substantiated – a relatively new scientific direction of research, in which cognition and cognitive processes are a kind of symbolic computation. It is shown that the cognitive approach makes it possible to create systems that fundamentally differ from the known systems for detecting, preventing and eliminating the consequences of computer attacks by the unique ability to independently associate and synthesize new knowledge about the qualitative characteristics and quantitative patterns of information countermeasures in cyberspace. The systematic appearance of a cognitive system for detecting intrusion based on computational cognitivism is proposed.

Information security, intrusion detection, intrusion detection system, computational cognitivism, quantitative patterns of countermeasures in cyberspace, converging NBIC-technologies, Big Data technologies

УДК 004.067

С. А. Аббас, А. И. Водяхо
Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В. И. Ульянова (Ленина)

Н. А. Жукова
Федеральный исследовательский центр Российской академии наук

Архитектурное проектирование кибер-физических систем, построенных на платформах интернета вещей

Рассматриваются особенности архитектурного этапа проектирования кибер-физических систем, построенных на платформах туманных вычислений. Анализируются основные типы вариабельности, встречающиеся в современных кибер-физических системах, определяются архитектурно-значимые характеристики и приводятся типовые постановки задач архитектурного проектирования для рассматриваемого класса систем. Предлагается последовательность принятия архитектурных решений при проектировании кибер-физических систем с высоким уровнем вариабельности. Рассматриваются типовые механизмы управления вариабельностью. Оценивается влияние вариабельности на основные архитектурно значимые параметры – производительность, готовность, совокупную стоимость владения. Предлагаемый подход ориентирован, прежде всего, на проектирование крупномасштабных распределенных кибер-физических систем. Рассмотренный подход был использован при построении ряда реальных кибер-физических систем, принадлежащих различным предметным доменам – производственным системам, системе мониторинга состояния систем кабельного телевидения и т. д. В настоящее время авторы ведут работы по созданию системы управления образовательным контентом.

Архитектурное проектирование, вариабельность, кибер-физические системы, интернет вещей, туманные платформ

Актуальность. Прогресс техники и технологии позволяет создавать и применять все более сложные антропогенные системы. При этом их сложность выражается не только и не столько в