
АКТУАЛЬНЫЕ ПРОБЛЕМЫ АНАЛИЗА КИБЕРПРОСТРАНСТВА

Предлагаются процесс-модели получения квазиоптимального детерминированного многозначного решения в n -мерном логическом векторном кибернетическом пространстве. Рассматриваются теоретические основы векторных логических вычислений для получения решения, оцениваемого неарифметическим критерием качества взаимодействия объектов в пространстве. Формулируются проблемы создания индивидуального киберпространства, распознавания образов, принятия решений, вакцинации программных продуктов, актуальные для рынка информационных технологий. Общность метрики оценивания качества решения на основе хог-операции объединяет достаточно разнородные по составу проблемы анализа информации.

1. Введение

Киберпространство (cyberspace) – метафорическая абстракция, используемая в философии и в компьютерах, – является виртуальной реальностью, которая представляет Ноосферу или Второй мир «внутри» компьютеров и сетей. Это определение взято из Википедии и согласуется с многочисленными публикациями, авторами которых в абсолютном большинстве являются философы. Возможно, уже настало время взглянуть на определение и сущность киберпространства с позиции технологической и математической культуры. Основная аксиома, заложенная в определении, – виртуальность пространства.

Виртуальность (virtus — потенциальный, возможный) – вымышленный, воображаемый объект или явление, не присутствующий в реальном мире, а созданный игрой воображения человеческой мысли, либо смоделированный при помощи других объектов. Реальные компьютеры создают «нереальное пространство», а точнее, Второй мир, который может и хочет стать первым. Не хватает малого – целенаправленного самосовершенствования компьютерного сообщества, способного на равных с человеком участвовать в создании и формировании киберпространства. Информация – реальность и сущность пространства, компьютер – ее носитель или форма.

Киберпространство – совокупность взаимосвязанных компонентов в виде информационных процессов или явлений с носителем в форме компьютера, взаимодействие которых между собой оценивается соответствующей метрикой.

Метрика – способ измерения расстояния в пространстве процессов или явлений. Что такое расстояние в киберпространстве? Самое простое есть самое верное, это – кодовое расстояние по Хэммингу между парой векторов, обозначающих процесс или явление. Производная (булева) – степень изменения или различия двух объектов. Киберпространство хорошо тем, что понятие близости объектов означает, насколько они отличаются друг от друга. Как измерить различия? Сравнить – применить хог-операцию, проще не бывает. Жизнь в киберпространстве уже имеет свои законы. Сохранение своей индивидуальности важно не только в Первом мире. Клоны только с первого взгляда есть хорошее дело в обоих мирах. Но если человека размножить, то прообраз просто потеряется в своих копиях. Прimitивно, но мало кто пожелает делить славу, почет, деньги с незаслужившими это копиями. Но хорошо делить ответственность за деструктивные действия оригинала. Аналогично, копировать и продавать клоны своего продукта есть путь к богатству. Но это может сделать и не автор продукта, продавая нелегальные копии, по сути, занимаясь воровством. В пределе – клонирование есть плохо в обоих мирах. Альтернатива уже сейчас прослеживается в киберпространстве, называемая, возможно, не совсем удачно, облачными вычислениями. Благодаря развитию телекоммуникаций, можно утверждать, что все объекты (субъекты) в киберпространстве находятся рядом в индивидуальном компьютере. Это означает, что необходимо иметь программу (функциональность) в одном экземпляре, которая доступна всему миру, первому и второму благодаря облачной технологии. Защитить одну копию – продукт на собственном сервере – намного проще задача,

чем сделать это для сотен и тысяч клонов. Аргументом в пользу появления облачных технологий является несанкционированное снятие защиты с лицензионной копии программного продукта и дальнейшее его использование, в том числе и на продажу. Востребованность продукта есть успех в бизнесе, здесь все как в первом мире – востребованность индивидуума – основа его состоятельности, признания, богатства. Таким образом, можно сделать некоторые выводы. 1) Второй мир стремится к повторению Первого – клонов не будет, но будет доступность каждого продукта, как компонента киберпространства для всех субъектов обоих миров. 2) Каждый субъект Первого мира будет иметь в идеале один образ (сайт) во Втором мире. 3) Аутентичность – желаемый идеал для всех конструктивных (законопослушных) субъектов обоих миров. 4) Сайт не олицетворяет сущности образа, который должен быть для индивидуума (прообраза) больше, чем отражение в зеркале. Это – история, действительность, метрические и медицинские данные, привычки, бизнес, быт, отдых, будущее, контакты, пристрастия. 5) Такой образ – индивидуальное киберпространство – должно быть сверхнадежно закрыто от несанкционированного доступа. 6) Защищать от деструктивных компонентов и действий нужно будет не компьютер, а субъект (программный продукт) киберпространства на основе предлагаемой технологии вакцинации. Повысить иммунитет субъекта за счет внедрения в программу агента – вакцинной избыточности, взаимодействующей с сервером Лаборатории Касперского (ЛК) – значит сохранить работоспособность продукта. Сервер ЛК генерирует облако услуг по защите субъектов киберпространства, которое в пределе может покрыть всю цифровую планету. 7) Расстояние (производная) – бинарное отношение объектов в киберпространстве есть степень изменения, различия в процессах или явлениях, определяемое с помощью (векторной) xor-операции. Производную можно применять ко всему, что шевелится или нет. Лишь бы было выполнено единственное условие – бинарность (два состояния в процессе, два объекта в пространстве, два явления в природе, два адреса, две части целого). Равно как и утверждение – сравнивать (измерять, соизмерять, оценивать, распознавать) можно все, что угодно, лишь бы этого было не менее двух. Семантический изоморфизм глагола «сравнивать» можно далее распространять на действия: принимать решение, тестировать, диагностировать, идентифицировать, управлять, прогнозировать,

Резюме – нужен мощный мультиматричный процессор (ММП), где каждая команда (and, or, xor, slc) обрабатывает параллельно и предельно быстро только одну бинарную операцию на матрицах (двумерные массивы данных). Количество командно-ориентированных матриц-примитивов создает систему – гетерогенный мультиматричный процессор бинарных операций с буфером М (рис. 1).

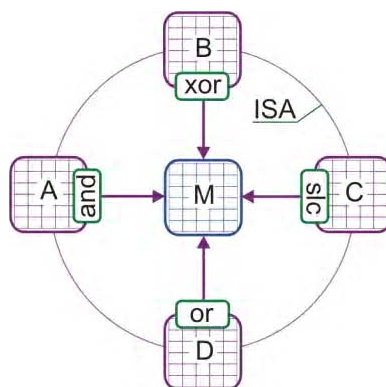


Рис. 1. Мультиматричный процессор бинарных операций

Мультиматричный модуль процессора включает 4 блока памяти со встроенными в них операциями (A – and, B – xor, C – slc – shift left crowding, D – or) и буферную память М. Модуль ориентирован на параллельное выполнение в данном случае одной из четырех инструкций (ISA – Instruction Set Architecture), оперирующих матрицами двоичных данных одинаковой размерности: $M = M \{and, or, xor, slc\} \{A, B, C, D\}$ с занесением результата в буфер М. Особенность ММП в том, что не ячейка матрицы имеет систему команд из

четырёх операций, а каждая команда имеет собственную матрицу ячеек в качестве данных для параллельной обработки, что существенно упрощает структуру управления и устройства в целом. Вся сложность ММП перенесена на структуры данных, где память матрицы имеет одну аппаратно-реализованную встроенную команду, что позволяет иметь примитивную систему управления параллельными вычислительными процессами (SIMD – Single Instruction Multiple Data), последовательностную по своей сути, а значит, нет необходимости создавать сверхсложные компиляторы, ориентированные на распараллеливание вычислительных процессов.

Здесь каждый матричный процессор выполняет одну операцию, встроенную в запоминающие элементы матрицы. Но возникают ситуации, когда матричный уровень (M-level) задания данных избыточен для выполнения, например операций над булевыми (B-level) или регистровыми (R-level) переменными. Для такого случая необходимо иметь иерархию по уровням представления данных, что иллюстрируется рис. 2.

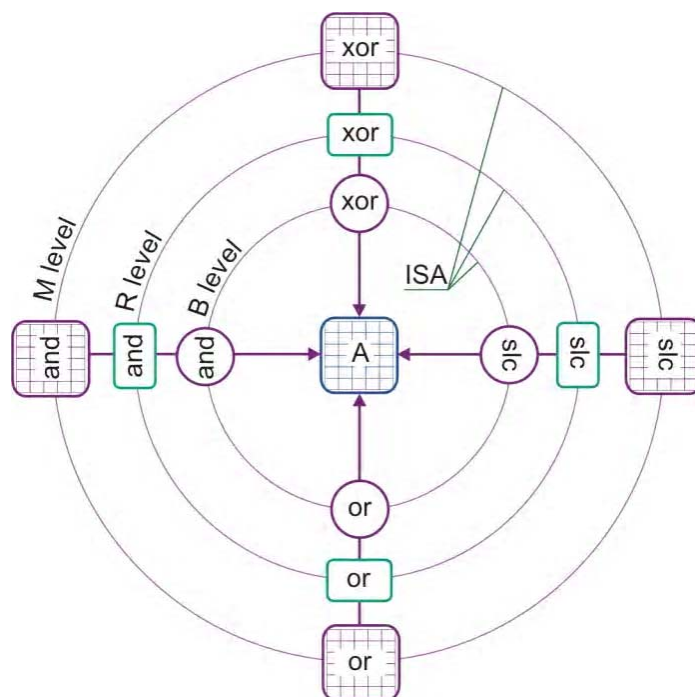


Рис. 2. Мультиуровневый по данным процессор

Здесь система команд (ISA) одна и та же, но операнды – различные по информационному объему, буфер $A = \{B, R, M\}$ представлен компонентами всех уровней иерархии. Мультиуровневая структура данных (Hierarchical SIMD) эффективна только для регистровых и матричных вычислений (переменных), где наблюдается высокий уровень параллельной обработки данных. Формат элементарной вычислительной процедуры для одной из операций имеет вид: $A = A \{and, or, xor, slc\} \{B, R, M\}$.

Источники: 1. Актуальные задачи анализа объектов в киберпространстве [1-3] 2. Метрика киберпространства [4, 5]. 3. Аппаратура и матричные процессоры [6-13, 14-17, 18-28].

2. Транзитивное замыкание дружественного для пользователя пространства

Многообразие сетевых структур (глобальные, корпоративные, социальные, профессиональные, технологические, культурные, бытовые) доставляет не только позитивное влияние на качество работы и отдыха, но и сильнейший прессинг каждому пользователю со стороны киберпространства. Вывод напрашивается сам собой. Необходимо создавать дружественное индивидуальное киберпространство (ИКП) с фильтрами, настроенными на повышение качества жизни каждого человека. Вход в киберпространство должен быть инвариантным по отношению к средству общения (компьютер, мобильный телефон, смар-

телефон, планшет, Apple iPad, Samsung Galaxy S, Apple MacBook Air, Logitech Revue, Google Nexus One (HTC Desire), Apple iPhone 4, Apple TV, Toshiba Libretto W100, Microsoft Kinect, Nook Color). Это означает, что индивидуальное пространство в будущем не должно быть связано с персональным компьютером и зависимо от него. ИКП позиционируется как внешний компонент по отношению к средству управления вычислительными процессами и отображения информации (monitored remote control). При этом вычислительные процессы реализуются на мощных серверах, а пользователь имеет технологию облачных вычислений для всех сфер человеческой деятельности. Такое упрощение функций персонального компьютера и необходимость создания внешнего индивидуального пространства, подключенного к вычислительным ресурсам планеты, ставит новые проблемы разработки средств защиты и сервисного обслуживания ИКП путем проектирования фильтров, структур данных, библиотек и технологий измерения взаимодействия процессов и явлений в киберпространстве. В частности, средства защиты информации следует создавать не только для программных и аппаратных продуктов, компьютеров и сетей, но и для ИКП.

Цель – существенное повышение качества сервисов, доставляемых со стороны программных, аппаратных изделий, сетевых структур и уменьшение стоимости эксплуатационных расходов за счет создания стандартизованных инфраструктур сервисного обслуживания, обеспечивающих дружественную эксплуатацию, тестирование и устранение функциональных нарушений.

Задачи: 1) Разработка математического аппарата анализа кибернетического пространства, ориентированного на создание моделей и методов сервисного обслуживания программных и аппаратных продуктов. 2) Создание процесс-моделей и критериев взаимодействия вредоносных компонентов с программными кодами полезных функциональностей. 3) Описание актуальных для рынка проблем, подлежащих решению в векторном логическом пространстве, определяемом бета-метрикой взаимодействия объектов.

Общность задач синтеза и анализа основана на использовании равенства нулю триады равноценных компонентов, соединенных операцией хог $m \oplus A \oplus Q = 0$, формулирующей условия позитивного решения проблемы. Здесь первый компонент m – входные условия, второй A – эталонные модели процесса или явления, третий Q – результат взаимодействия первых двух, который может вырождаться в критерий качества отношения или принятия решения, оценку распознавания объектов или образов. В общем случае третий компонент Q есть критерий качества решения многих задач синтеза или анализа, имеющий единый с первыми двумя компонентами формат данных – вектор или матрицу. При синтезе входные условия m есть спецификация, а модель A представлена элементами, которые создают условия покрытия специфицированной функциональности. При этом полнота покрытия оценивается третьим компонентом Q , исполняющим роль критерия качества. Следует помнить, что хог-триада, равная нулю, есть условие или метрика оценивания взаимодействия объектов в киберпространстве. В то время как стратегия, технология, модель или алгоритм вычислительных процессов, приводящих к заданной цели, есть самостоятельная проблема для каждой области знаний, которая в данном случае остается за кадром. Аналогично можно использовать триадное уравнение для тестирования знаний студентов, равно как и для создания валидных тестовых заданий (T – тест, S – объект или студент):

$$T \oplus S \oplus Q = 0; \quad T \oplus S_i = \min_i Q_i \rightarrow x \oplus m \oplus Q = 0,$$

$$T \oplus S_1 = Q_1 \approx \begin{matrix} 00 \\ 01 \\ 10 \\ 11 \end{matrix} \begin{bmatrix} 1 & . & 1 & 1 \\ . & 1 & . & . \\ . & 1 & . & 1 \\ 1 & . & . & 1 \end{bmatrix} \oplus \begin{bmatrix} 1 & . & 1 & 1 \\ . & 1 & 1 & 1 \\ . & . & . & . \\ . & . & . & 1 \end{bmatrix} = \begin{bmatrix} . & . & . & . \\ . & . & 1 & 1 \\ . & 1 & . & 1 \\ 1 & . & . & . \end{bmatrix},$$

$$T \oplus S_2 = Q_2 \approx \begin{matrix} 00 \\ 01 \\ 10 \\ 11 \end{matrix} \begin{bmatrix} 1 & . & 1 & 1 \\ . & 1 & . & . \\ . & 1 & . & 1 \\ 1 & . & . & 1 \end{bmatrix} \oplus \begin{bmatrix} 1 & . & 1 & 1 \\ . & 1 & 1 & 1 \\ . & 1 & . & 1 \\ . & . & . & 1 \end{bmatrix} = \begin{bmatrix} . & . & . & . \\ . & . & 1 & 1 \\ . & . & . & . \\ 1 & . & . & . \end{bmatrix},$$

$$T \oplus S_3 = Q_3 \approx \begin{matrix} 00 \\ 01 \\ 10 \\ 11 \end{matrix} \begin{bmatrix} 1 & . & 1 & 1 \\ . & 1 & . & . \\ . & 1 & . & 1 \\ 1 & . & . & 1 \end{bmatrix} \oplus \begin{bmatrix} 1 & . & 1 & 1 \\ . & 1 & 1 & . \\ . & 1 & . & 1 \\ 1 & . & . & 1 \end{bmatrix} = \begin{bmatrix} . & . & . & . \\ . & . & 1 & . \\ . & . & . & . \\ . & . & . & . \end{bmatrix}.$$

В медицине процесс идентификации нарушений сводится к сравнению эталона органа или организма с существующими состояниями упомянутых компонентов конкретного человека. В результате формируется критерий качества, указывающий на критические места организма, подлежащие коррекции:

$$T \oplus S = Q \approx \begin{matrix} 00 \\ 01 \\ 10 \\ 11 \end{matrix} \begin{bmatrix} 1 & . & 1 & 1 \\ . & 1 & . & . \\ . & 1 & . & 1 \\ 1 & . & . & 1 \end{bmatrix} \oplus \begin{bmatrix} 1 & . & 1 & 1 \\ . & 1 & 1 & 1 \\ . & . & . & . \\ . & . & . & 1 \end{bmatrix} = \begin{bmatrix} . & . & . & . \\ . & . & 1 & 1 \\ . & 1 & . & 1 \\ 1 & . & . & . \end{bmatrix}.$$

Восстановление зафиксированных функциональных сбоев возможно и путем воздействия на органы человека через каналы связи информационными символами или образами: пищевыми, тактильными, звуковыми, вербальными, вкусовыми, обонятельными, зрительными. Возможность такой коррекции основана на влиянии (позитивном или негативном) всех указанных компонентов на здоровье человека в процессе жизненного цикла. К 40 годам каждый человек в силу своих привычек формирует собственную карту входных переменных, которые создают, как правило, функциональные нарушения отдельных органов. Путем сравнения реальной карты переменных человека с идеальной для него можно выбрать формулу коррекции нарушений и предложить далее более приемлемую карту (библиотеку) переменных, отражающую позитивные и негативные для организма входные воздействия. Второй компонент библиотеки необходим для создания специальных фильтров, ставящих под запрет доставку человеку деструктивной для него информации. Любое входное воздействие m сравнивается с фильтрами позитивных и негативных воздействий

$A = A^P \vee A^N$. Критерий качества определяет инверсную функцию принадлежности входного воздействия к одному из фильтров:

$$m \oplus A = Q; A = A^P \vee A^N;$$

$$m = \begin{matrix} 00 \\ 01 \\ 10 \\ 11 \end{matrix} \begin{bmatrix} 1 & . & 1 & 1 \\ . & 1 & . & . \\ . & 1 & . & 1 \\ 1 & . & . & 1 \end{bmatrix} \oplus (A^P \vee A^N) = \left(\begin{bmatrix} 1 & . & 1 & 1 \\ . & 1 & 1 & 1 \\ . & . & . & . \\ . & . & . & 1 \end{bmatrix} \vee \begin{bmatrix} 1 & . & 1 & 1 \\ . & 1 & . & . \\ . & . & . & . \\ 1 & . & . & . \end{bmatrix} \right) = Q(A^P \vee A^N) = \left(\begin{bmatrix} . & . & . & . \\ . & . & 1 & 1 \\ . & 1 & . & 1 \\ 1 & . & . & . \end{bmatrix} \vee \begin{bmatrix} . & . & . & . \\ . & . & . & . \\ . & 1 & . & 1 \\ . & . & . & 1 \end{bmatrix} \right).$$

В данном случае входное воздействие является негативным для индивидуума, поскольку $Q(A^P) > Q(A^N)$. Предложенная модель должна стать частью индивидуального кибернетического пространства или компьютера-друга, который, прежде всего, должен ежедневно заботиться о состоянии организма путем наблюдения и управления его поведением с целью выработать привычки относительно правильных входных переменных и корректировать возникающие несоответствия относительно идеальной карты, уникальной для каждого человека, состояния организма. Доказательством возможности коррекции состояния организма может служить эмоциональная зависимость от той информации, которую ему навязывают дома, на работе, в транспорте, по Интернету, телевидению и печати. Информация (образ) может исцелить (известную идею разделяет доктор В.А.Садырин) или убить человека, возвысить или опустить, активизировать или сделать его пассивным. Очень важен авторитет источника, от которого исходит информация. Одни и те же слова по-разному воспринимаются человеком, если они произносятся двумя персонками, имеющими для приемника различный рейтинг. Отсюда вывод – авторитетный для пациента врач уже исцеляет его своим образом и словами без медикаментозных препаратов. Для социума, группы людей, человека существует система позитивных и негативных образов, символов и знаков, которые необходимо найти для создания индивидуального компьютера, способного поддерживать человека на высоком уровне творческой и физической активности путем торможения отношения к деструктивам и активизации – при наличии позитивных образов. Все перечисленное уже имеется в распоряжении спецслужб и психоаналитиков, которые могут манипулировать человеком для решения не всегда гуманистических проблем. Цель, актуальная на рынке сегодня, – сделать каждого человека планеты более счастливым путем информационной коррекции его отношения к действительности, с помощью индивидуального киберпространства или компьютера. Создать дружественный уголок киберпространства лучше с позиции его инвариантности по отношению к типам интер-

фейса <человек–пространство>, которые могут быть представлены любым доступным прибором: компьютер, мобильный телефон, смартфон. Даже при утере физического интерфейса (гаджета) киберпространство каждого индивидуума сохраняется. Как построить такой приватный уголок? Для этого необходимо создать изменяющуюся в пространстве и во времени идеальную компьютерную модель (коррекции) поведения человека, которая способна к самосовершенствованию в процессе жизненного цикла индивидуума, постоянно изучая поведение последнего на уровне всех возможных каналов его реакций на окружающую действительность. Кто формирует здоровье нации, группы людей и человека? Не только пища и экология, и не столько перечисленные компоненты. Но авторитеты для каждого индивидуума: руководитель страны, города, предприятия, учреждения, отдела, группы. Чем ближе к индивидууму и выше рейтинг руководителя, тем сильнее его негативное влияние на подчиненного отрицательными образами, словами, символами. Не сотвори себе кумира! Равно как и наоборот, позитивный знак или образ приумножают силы и активность подчиненных на решение сложных задач. С другой стороны, каждый из нас ежедневно должен заботиться о своем имидже успешного, креативного, жизнерадостного и здорового человека, который своим образом, приятным для окружающих, должен порождать в них аналогичные качества.

3. Технология вакцинации программных продуктов

Востребованной на рынке является решение проблемы создания теории, методов и архитектуры параллельного анализа информации, представленной в виде аналитических, графовых и табличных форм ассоциативных отношений для поиска, распознавания, диагностирования деструктивных компонентов и принятия решений в n -мерном векторном дискретном пространстве. Здесь целесообразно использовать векторно-логические процесс-модели актуальных прикладных задач, в том числе – диагностирование вирусов и восстановление работоспособности программно-аппаратных компонентов компьютерных систем, качество решения которых оценивается неарифметической метрикой взаимодействия булевых векторов.

Решение проблемы ориентировано на поиск, распознавание, диагностирование деструктивных компонентов аппаратно-программными методами в дискретном кибернетическом пространстве. Общность представленной теории синтеза и анализа кибернетического пространства основана на использовании равенства нулю триады равноценных компонентов, соединенных операцией $m \oplus A \oplus Q = 0$, формулирующей условия решения проблемы. Здесь первый компонент m – входной код программы, второй A – эталонные модели деструктивов, третий Q – результат взаимодействия первых двух, который может вырождаться в критерий качества отношения или принятия решения, оценку распознавания объектов или образов.

Цель – существенное повышение качества программных продуктов и уменьшение стоимости эксплуатационных расходов за счет их вакцинации путем введения в код встроенной программной избыточности в виде инфраструктуры сервисного обслуживания, обеспечивающей тестирование, диагностирование и устранение вредоносных компонентов, классифицируемых в библиотеках.

Объект исследования – кибернетическое пространство, представленное информацией, ее носителями и преобразователями, а также деструктивными компонентами, наносящими вред функциональностям, улучшающим качество жизни человека.

Субъект исследования – инфраструктура сервисного обслуживания в виде встроенной программной избыточности, работающей в реальном масштабе времени, обеспечивающей тестирование, диагностирование и устранение вредоносных компонентов, описанных в соответствующих библиотеках.

Мотивация: 1) Отсутствие на рынке антивирусной защиты встроенных средств тестирования, диагностирования и удаления вредоносных компонентов, составляющих инфраструктуру сервисного обслуживания, подобной тому, как в цифровых системах на кристаллах существуют стандарты граничного сканирования, а в программных продуктах – ассерционная избыточность, ориентированные на встроенное тестирование дефектов и ошибок с последующим восстановлением работоспособности аппаратных или программных изде-

лий. 2) Наличие теоретических разработок, связанных с технологией алгебрологического векторного анализа информационных данных, ориентированных на высокое быстродействие решения и оценивания задач распознавания образов, принятия решений и тестирования объектов [1-9]. 3) Наличие образцовой производственной и маркетинговой инфраструктуры (Лаборатория Касперского), способной поддержать проект создания технологии вакцинации программных продуктов и авторитетно предложить его рынку информационных технологий. 4) Миниатюризация цифровых и телекоммуникационных систем (телефоны, смартфоны, IP-фоны, планшеты) требует постоянной защиты от несанкционированного доступа путем внедрения встроенных антивирусных средств, контролирующих информационный обмен.

Задачи: 1) Разработка математического аппарата анализа кибернетического пространства, ориентированного на создание моделей и методов сервисного обслуживания программных продуктов для тестирования, диагностирования и устранения вредоносных компонентов. 2) Создание типовых процесс-моделей и критериев взаимодействия вредоносных компонентов с программными кодами полезных функциональностей. 3) Разработка технологии анализа структуры программного кода для определения критических точек и установки в них ассерционных операторов наблюдения и управления в процессе его функционирования. 4) Создание инфраструктуры сервисного обслуживания функциональных программ для встроенного тестирования, диагностирования и устранения вредоносных компонентов из программного кода функциональности на основе использования библиотеки деструктивных элементов. 5) Тестирование и верификация встроенной инфраструктуры сервисного обслуживания функциональностей, защищающей программный код от вредоносных компонентов.

Ожидаемые результаты и их рыночная привлекательность: 1) Инфраструктура встроенной защиты программного кода от несанкционированной модификации, приводящей к изменению функциональности. 2) Избыточность инфраструктуры программного кода, которая автоматически синтезируется на стадии проектирования и верификации, составляет не более 5% от специфицированной функциональности. 3) Рыночная привлекательность инфраструктуры, определяемая многообразием программных продуктов, умноженной на уровень продаж каждого изделия, составляет в год порядка одного миллиарда экземпляров. 4) Стоимость создания инфраструктуры для программного продукта составляет 5–10% затрат от разработки функционального кода. Если уровень продаж – не менее 500 копий, то затраты на создание встроенного антивируса вполне окупаемы в течение года. 5) Внедрение запатентованной технологии вакцинации программных продуктов при их рождении может принести компании порядка миллиарда долларов. 6) Маркетинговая проблема глобальной компании (Лаборатория Касперского) заключается в убеждении разработчиков программных продуктов имплементировать существующие внешние антивирусы вовнутрь кода полезной функциональности.

4. Инфраструктура программного продукта

Проблема создания эффективной инфраструктуры кибернетического пространства (Cyber Space), а также саморазвивающейся информационно-компьютерной экосистемы (ИКЭС) планеты особенно важна для глобальных компаний, таких как Лаборатория Касперского, Google, Microsoft. Экосистема – совокупность популяций, взаимодействующих между собой и окружающей их средой неопределённо долгое время, имеющая сходство относительно протекающих в них энергетических (информационных) процессов.

Кибернетическое пространство как объект природы также подвержено влиянию деструктивных компонентов, влияющих на работоспособность субъектов, которыми являются компьютеры, системы и сети. Поэтому сейчас и в будущем важной проблемой остается стандартизация пространства и специализация всех взаимодействующих субъектов, включая негативные, как неотъемлемую часть экосистемы. Данная акция есть постоянно действующая во времени, цель которой – не отставать, но на один шаг опережать появление новых вредоносных компонентов. путем создания инфраструктуры кибернетического пространства, обеспечивающей функционирование компьютерной экосистемы планеты и качество жизни каждого человека, включающей следующие модули:

1) Наблюдение за состоянием внутренних и выходных линий компьютерных систем в процессе функционирования, верификации и тестирования штатных функциональных блоков на основе использования стандартов граничного сканирования.

2) Тестирование функциональных модулей путем подачи проверяющих наборов от различных тестовых генераторов, ориентированных на проверку вредоносных или исправного поведения.

3) Диагностирование отказов и вредоносных путем анализа информации, полученной на стадии тестирования и использования специальных методов встроенного поиска деструктивов на основе стандартов граничного сканирования или ассерционной избыточности, ориентированной на обнаружение вредоносного кода, что позволит идентифицировать и устранять деструктивы программных продуктов без использования внешних средств. Таким образом, можно будет обходиться без сложных внешних программ моделирования, тестирования и диагностирования путем прививки каждого программного изделия тестопригодной интеллектуальной избыточностью кода на стадии его создания. При этом следует использовать предикат узнавания буквы a : $x^a = x \oplus a$, который оперирует не только булевыми, но регистровыми и матричными переменными, что делает его практически значимым в формальной записи уравнений диагноза или распознавания:

$$x^a \approx x \oplus a = 0 \vee \min_i Q_i \rightarrow x \oplus a \oplus Q = 0;$$

$$x^m \approx x \oplus m = 0 \vee \min_i Q_i \rightarrow x \oplus m \oplus Q = 0;$$

$$T \oplus S = Q \approx \begin{matrix} 00 \\ 01 \\ 10 \\ 11 \end{matrix} \begin{bmatrix} 1 & . & 1 & 1 \\ . & 1 & . & . \\ . & 1 & . & 1 \\ 1 & . & . & 1 \end{bmatrix} \Delta \begin{bmatrix} 1 & . & 1 & 1 \\ . & 1 & 1 & 1 \\ . & . & . & . \\ . & . & . & 1 \end{bmatrix} = \begin{bmatrix} . & . & . & . \\ . & . & 1 & 1 \\ . & 1 & . & 1 \\ 1 & . & . & . \end{bmatrix}.$$

На основе предиката узнавания m -образа любой сложности, природы и формы можно создавать достаточно компактные уравнения предикатов, формирующие интеллектуальные решения в области распознавания образов, принятий решений, тестирования знаний и технических объектов, диагностирования (узнавания) вирусных компонентов в программном коде.

4) Восстановление работоспособности функциональных модулей после фиксации отрицательного результата тестирования и определения места и вида вредоноса при выполнении фазы диагностирования внутренними средствами, создающими инфраструктуру функциональности.

5) Измерение сигнатуры, основных характеристик и параметров функционирования изделия на основе встроенных средств, позволяющих производить временные и функциональные измерения для идентификации состояния программного продукта.

6) Надежность и отказоустойчивость функционирования изделия в процессе эксплуатации, которая достигается диверсификацией функциональных блоков, их дублированием и восстановлением работоспособности в реальном масштабе времени, а также встроенными средствами сервисного обслуживания, работающими в реальном масштабе времени.

В связи с этим предложенная инфраструктура кибернетического пространства, метрика его измерения и процесс-модели анализа и синтеза субъектов дают возможность создавать эффективные решения компьютерных изделий, ориентированных на быстрый поиск, распознавание, диагностирование не только позитивных, но и негативных субъектов. Конкретно, предложенная инфраструктура может решать задачи: 1) Описание многообразия деструктивных компонентов кибернетического пространства. 2) Формализация процессов взаимодействия триады компонентов <программа, деструктивности, тесты>. 3) Диагностирование и устранение деструктивных компонентов. 4) Создание и эффективное использование базы деструктивных данных. 5) Создание быстродействующих интеллектуальных саморазвивающихся средств сервисного обслуживания и защиты кибернетического пространства.

Инфраструктура верификации и тестирования проектов цифровых систем и программных продуктов есть актуальная проблема для рынка информационных технологий. Ее

комплексное решение, основанное на наличии исходной информации в качестве спецификации, включает следующие пункты: 1. Синтез транзакционного графа (ТГ) HDL-кода, где вершины есть примитивы или структуры памяти, а дуги – операторы или строки HDL-кода. 2. Оценивание тестопригодности проекта на основе анализа ТГ и определение критических точек, покрываемых механизмом ассерций для наблюдения (управления) процесса верификации. 3. Оценивание диагностируемости проекта по ТГ, как количественной характеристики глубины поиска функциональных нарушений (ФН) в коде HDL-модели. Улучшение оценки диагностируемости путем введения дополнительных ассерций, позволяющих довести упомянутую характеристику до требований пользователя. 4. Определение функционального покрытия (functional coverage) для синтеза теста и оценивания его качества в виде полноты покрытия. 5. Синтез таблицы функциональных нарушений (ТФН), как матрицы контрдостижимостей вершин и дуг, задающих структуры памяти HDL-кода и блоки операторов соответственно, проверяемых на тестовых сегментах. 6. Создание методов и средств диагностирования ФН на основе векторного параллельного анализа строк и столбцов ТФН. 7. Разработка методов и средств восстановления работоспособности HDL-кода за счет введения избыточности в проект в виде ТГ, механизма ассерций и использования моделей и методов, направленных на: синтез ТГ, оценивание тестопригодности и диагностируемости, построение ТФН, анализ ТФН для поиска функциональных несоответствий.

5. Проблема оценивания качества решения

1. Можно ли уйти от численной оценки качества двух или нескольких проектов (решений)? Ответ позитивный. Процесс-модель поиска оценки лучшего решения (рис. 3) имеет следующие пункты. 1) Первоначально в вектор-результат Q , в котором будет сохранено лучшее решение, заносятся единичные значения во все координаты (худшее решение) и одновременно осуществляется операция slc сдвига влево и уплотнения всех единиц текущего вектора Q_i . 2) Затем выполняется сравнение двух векторов: Q и очередной оценки Q_i из списка решений. 3) После этого реализуется векторная операция $and (Q \wedge Q_i)$, результат которой сравнивается с содержимым вектора Q , что дает возможность изменить его, если вектор Q_i имеет меньшее число единичных значений. 4) Процедура поиска оценки лучшего решения повторяется n раз.

$$Q = Q(\overline{((Q \wedge Q_i) \oplus Q)}) \vee Q_i(\vee((Q \wedge Q_i) \oplus Q));$$

$$Y = \vee((Q \wedge Q_i) \oplus Q);$$

$$Q = Q\bar{Y} \vee Q_i Y.$$

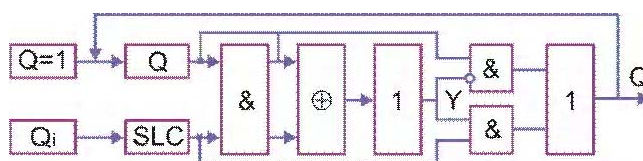


Рис. 3. Процесс-модель выбора решения

2. Можно ли найти замену фаззи-логике, чтобы уйти от численного интервала и арифметических операций? Исходя из сущности фаззи-логики, как гибрида чисел в интервале $[0,1]$ и логических (теоретико-множественных) операций, можно предложить более рациональную алгебраическую структуру, в которой нет «тяжеловесных» арифметических операций и чисел. Лингвистическая или логическая переменная определена любым фиксированным количеством градаций, кратным степени двойки (2, 4, 8, 16, 32, 64, ...). Логические операции над лингвистическими переменными определяются многозначными таблицами или диаграммой Хассе. Последняя дает возможность ранжировать все состояния лингвистической переменной для определения операций пересечения и объединения в виде быстрого поиска по графу минимума или максимума функций принадлежности, заданных кодами-состояниями двух любых вершин. Для определения функции принадлежности некоторого входного компонента необходимо иметь эталон модели, относительно которой путем сравнения формируется степень соответствия объекта эталону. Вместо численного интервала $[0,1]$

для задания функции принадлежности используется вектор двоичных или многозначных переменных:

$$\mu(m \in A) = (a_1, a_2, \dots, a_i, \dots, a_n), a_i = \{0, 1\} \vee \{\alpha_1, \alpha_2, \dots, \alpha_i, \dots, \alpha_k\}.$$

Вектор может вырождаться в одну переменную, равно как и усложняться до матричной структуры, если она есть технологичная форма для описания эталона и входного объекта. Таким образом, функция принадлежности должна быть идентичной по структуре анализируемым компонентам. Нуль-вектор (-матрица, -переменная) будет означать полное совпадение входного объекта с эталоном.

Пример 1. Имеются три подмножества температур: $C = \{9, 10, 11\}$, $W = \{11, 12, 13\}$, $H = \{13, 14, 15\}$. Определить функции принадлежности для входного подмножества $m = \{10, 11, 12\}$. Подмножества представлены тремя векторами:

$$C = (1110000), W = (0011100), H = (0000111).$$

Функции принадлежности определяются выполнением хог-операции над входным вектором $m = (0111000)$ и тремя эталонами:

$$\mu(m \in C) = m \oplus C = (0111000) \oplus (1110000) = (1001000);$$

$$\mu(m \in W) = m \oplus W = (0111000) \oplus (0011100) = (0100100);$$

$$\mu(m \in H) = m \oplus H = (0111000) \oplus (0000111) = (0111111).$$

Для выбора лучшего решения следует использовать схему, представленную на рис. 3, которая путем попарного сравнения определит, что минимальную оценку имеет $\mu(m \in C)$, как первое решение, которое будет встречено в процессе вычислений, из двух оптимальных $\mu(m \in C), \mu(m \in W)$, имеющих одинаковое и минимальное число несовпадений или единиц, равное двум. Кроме того, векторная оценка принадлежности дает еще дополнительную информацию – по каким координатам или переменным произошло несовпадение, что полностью скрыто в скалярном интервальном оценивании функций принадлежности.

Если считать достоинством непустое пересечение значений лингвистических переменных, которое позволяет относить результат к двум или нескольким подмножествам, то для целей определения принадлежности результата логической операции к одному из заданных подмножеств можно использовать диаграмму Хассе, как идеальный инструмент выполнения теоретико-множественных операций пересечения и объединения. На рис. 4 представлена диаграмма, содержащая 4 примитива: 00, 01, 10, 11. Здесь нет «тяжелых» арифметических операций, но только логические, которые имеют высокое быстроедействие, что является несомненным преимуществом использования диаграммы для определения результата и принадлежности входного объекта к одному из заданных подмножеств.

Пример 2. Построить диаграмму карьерной иерархии. Для этого используется четыре непересекающихся примитива культур: эстетическая, математическая, технологическая и социальная: $A = \{A_e = Q, A_m = E, A_t = H, A_s = J\}$. Все виды теоретико-множественного взаимодействия данных примитивов представлены диаграммой Хассе на рис. 4, которые задают 4 уровня состоятельности человека. Каждая вершина есть лингвистическая переменная, содержащая от нуля до четырех значений. Вычисление операций (объединения, пересечения) над любыми двумя вершинами или символами сводится к нахождению ближайшей общей вершины (верхнего, нижнего) уровня. Определение функции принадлежности сводится к вычислению площади взаимодействия между m и примитивами: $\mu(m \in A) = m \oplus \{A_e, A_m, A_t, A_s\}$ или m и производными от примитивов – любой вершиной диаграммы Хассе.

Для перевода теоретико-множественной модели в плоскость булевой векторной алгебры, где основные операции есть and, or, not, необходимо выполнить кодирование всех многозначных символов двоичными векторами:

$\emptyset$	Q	E	H	J	O	I	A	B	S	P	C	F	L	V	Y
0000	1000	0100	0010	0001	1010	0101	1100	0011	1001	0110	0111	1011	1101	0111	1111

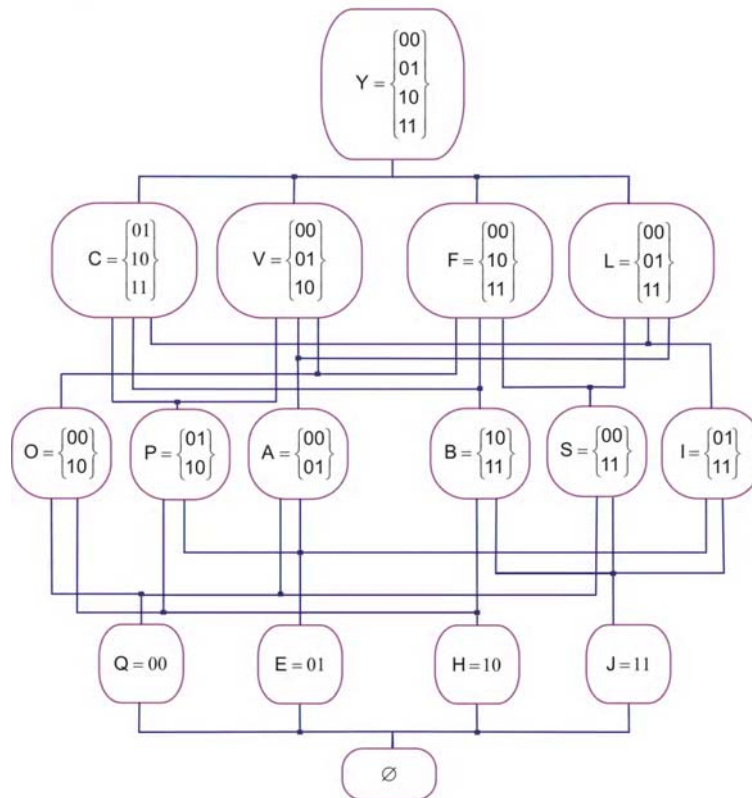


Рис. 4. Диаграмма Хассе для четырех примитивов

Алгебра векторной логики дает возможность существенно повысить быстродействие выполнения операций по сравнению с теоретико-множественными, которые технологически трудно сделать векторными и параллельными процедурами. На кодах символов шестнадцатеричного алфавита выполняются все операции алгебры логики, относительно которых кодовый алфавит является замкнутым, как и исходный теоретико-множественный булеан. В общем случае вектор объекта $A = (A_1, A_2, \dots, A_i, \dots, A_k)$ киберпространства может содержать разнородные лингвистические переменные, каждая из которых может быть адекватно и однозначно представлена вектором из n переменных в логическом булевом пространстве $n = \log_2 |A_i|$. Для понимания человеком лингвистические переменные, заданные в многозначном алфавите, более предпочтительны. Однако все вычислительные процессы необходимо ориентировать на векторы-коды значений упомянутых переменных в булевом пространстве.

Мощность универсума примитивов определяется моделью проблемы, подлежащей решению. Для универсума, содержащего 2 примитива, оптимальная структура будет представлена в виде четырехзначной диаграммы $A = \{A_m = 0, A_t = 1, A_x = X = \{0, 1, \emptyset\}\}$, изображенной на рис. 5.

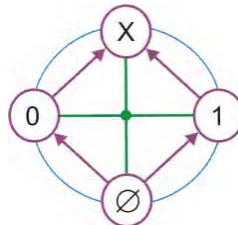


Рис. 5. Диаграмма Хассе для двух примитивов

Восьмизначный алфавит на универсуме из трех элементов представлен следующим множеством всех подмножеств: $A = \{\emptyset, \{a\}, \{b\}, \{c\}, \{a, b\}, \{a, c\}, \{b, c\}, \{a, b, c\}\}$, которому соответствует диаграмма, представленная на рис. 6.

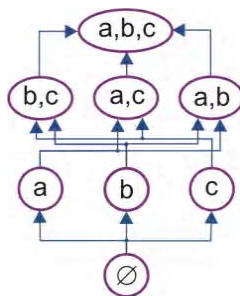


Рис. 6. Диаграмма Хассе для трех примитивов

Такая диаграмма может моделировать процессы, связанные с определением принадлежности входного воздействия одному из компонентов библиотеки. Например, заданы три перекрывающихся интервала роста человека, названные: низкий (1,2,3), средний (3,4,5), высокий (5,6,7). Векторное задание каждого из интервалов на универсуме из 7 двоичных переменных $Y=(1,2,3,4,5,6,7)$ имеет следующий вид: $A_1=(1110000)$; $A_2=(0011100)$; $A_3=(0000111)$. Полная модель логического взаимодействия данных примитивов относительно операций объединения и пересечения определяется трехпримитивной диаграммой Хассе (см. рис. 4), где представлены все возможные (a, b, c, ab, bc) и невозможные (ac, abc) сочетания для решения задачи вычисления принадлежности входного слова m одному из восьми состояний вершины. Чтобы вычислить функции принадлежности $m=(1101001)$ для трех примитивов графа Хассе, нужно применить операцию хог:

$$\mu(m \in A_1) = m \oplus A_1 = (1101001) \oplus (1110000) = (0011001);$$

$$\mu(m \in A_2) = m \oplus A_2 = (1101001) \oplus (0011100) = (1110101);$$

$$\mu(m \in A_3) = m \oplus A_3 = (1101001) \oplus (0000111) = (1101110).$$

Вывод: входной сигнал или условие m имеет минимум различий с объектом под номером 1.

Таким образом, функция принадлежности может быть идентифицирована скалярной величиной или одной переменной, градуированной мощностью примитивных символов диаграммы Хассе; одномерным вектором двоичных или многозначных переменных; двумерной матрицей двоичных или многозначных переменных. Если площадь взаимодействия между m и A формирует функцию принадлежности $\mu(m \in A) = m \oplus A$, то некоторые варианты принадлежности не могут быть адекватно представлены интервальной метрикой. Например, рис. 7 имеет скалярную неразличимость в интервальной арифметике первого и третьего, второго и четвертого вариантов отношений между m и A.

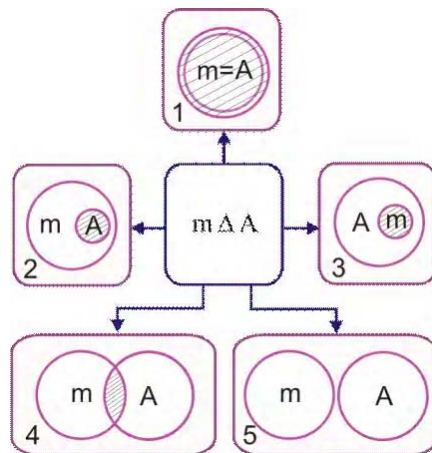


Рис. 7. Типы взаимодействия двух объектов

6. Распознавание образов на основе хог-операции

Если разделить синтаксис и семантику постановки технологической задачи, то многообразие интересных для практики проблем можно свести к универсальной в некотором смысле модели или форме. Так, множество задач можно привести к модели треугольника, когда обобщенная структура представляется уравнением (входные образы, эталоны, критерии сходства): $m \oplus A \oplus Q = 0$, где по любым двум известным компонентам следует найти третий. В формате триады компонентов модели описания процесса или явления простое решение задачи распознавания образов может быть получено, если воспользоваться вектором переменных $m = (m_1, m_2, \dots, m_i, \dots, m_n)$, где каждая из них $m_i \in B(Y)$, $Y = (Q = 00, E = 01, H = 10, J = 11)$ определена, в общем случае, более чем двумя значениями или булеаном на универсуме примитивов $B(Y) = \{Q, E, H, J, O = \{Q, H\}, I = \{E, J\}, A = \{Q, E\}, B = \{H, J\}, S = \{Q, J\}, P = \{E, H\}, C = \{E, H, J\}, F = \{Q, H, J\}, L = \{Q, E, J\}, V = \{Q, E, H\}, Y = \{Q, E, H, J\}, U\}$. Тогда взаимодействие двух векторов по правилам теоретико-множественной операции симметрической разности $m_i \Delta A_i = (m_i \cap \tilde{A}_i) \cup (\tilde{m}_i \cap A_i)$, например, будет иметь следующий вид: $m \Delta A = (SPQF) \Delta (QEAL) = (JHEP)$. Графическая интерпретация векторов и результата операции оценивания их сходства имеет следующий вид:

$$m \Delta A = (SPQF) \Delta (QEAL) = (JHEP) = \begin{matrix} 00 \\ 01 \\ 10 \\ 11 \end{matrix} \begin{bmatrix} 1 & . & 1 & 1 \\ . & 1 & . & . \\ . & 1 & 1 & 1 \\ 1 & . & . & 1 \end{bmatrix} \Delta \begin{matrix} 1 & . & 1 & 1 \\ . & 1 & 1 & 1 \\ . & . & . & . \\ . & . & . & 1 \end{matrix} = \begin{bmatrix} . & . & . & . \\ . & . & 1 & 1 \\ . & 1 & . & 1 \\ 1 & . & . & . \end{bmatrix}.$$

Здесь вектор-столбец (1..1) идентифицирует или кодирует пару элементов {00, 11} одним символом S, а точки обозначают отсутствие соответствующих примитивов, в данном случае {01, 10}. Естественно, что сложение всех матриц путем применения операции симметрической разности дает матрицу пустых значений:

$$(SPQF) \Delta (QEAL) \Delta (JHEP) = (\emptyset \emptyset \emptyset \emptyset) \rightarrow \begin{matrix} 00 \\ 01 \\ 10 \\ 11 \end{matrix} \begin{bmatrix} 1 & . & 1 & 1 \\ . & 1 & . & . \\ . & 1 & 1 & 1 \\ 1 & . & . & 1 \end{bmatrix} \Delta \begin{matrix} 1 & . & 1 & 1 \\ . & 1 & 1 & 1 \\ . & . & . & . \\ . & . & . & 1 \end{matrix} \Delta \begin{matrix} . & . & . & . \\ . & . & 1 & 1 \\ . & 1 & . & 1 \\ 1 & . & . & . \end{matrix} = \begin{bmatrix} . & . & . & . \\ . & . & . & . \\ . & . & . & . \\ . & . & . & . \end{bmatrix}.$$

Выполнение операции симметрической разности можно представить набором тождеств:

$$\Delta = \begin{cases} 0 \leftarrow (m = 0 \wedge A = \emptyset) \vee (m = 1 \wedge A = x) \vee m = x \wedge A = 1 \vee (m = \emptyset \wedge A = 0); \\ 1 \leftarrow (m = 0 \wedge A = x) \vee (m = 1 \wedge A = \emptyset) \vee m = x \wedge A = 0 \vee (m = \emptyset \wedge A = 1); \\ x \leftarrow (m = 0 \wedge A = 1) \vee (m = 1 \wedge A = 0) \vee m = x \wedge A = \emptyset \vee (m = \emptyset \wedge A = x); \\ \emptyset \leftarrow (m = 0 \wedge A = 0) \vee (m = 1 \wedge A = 0) \vee m = x \wedge A = x \vee (m = \emptyset \wedge A = \emptyset). \end{cases}$$

Они регулируют все отношения в четырехзначном алфавите Кантора $A = \{0, 1, x = \{0, 1\}, \emptyset\}$ для определения оценки взаимодействия или сходства двух объектов, например:

$$m \Delta A = (01xx) \Delta (10x1) = (xx00) = \begin{matrix} 0 \\ 1 \end{matrix} \begin{bmatrix} 1 & . & 1 & 1 \\ . & 1 & 1 & 1 \end{bmatrix} \Delta \begin{matrix} . & 1 & 1 & . \\ 1 & . & 1 & 1 \end{matrix} = \begin{bmatrix} 1 & 1 & . & 1 \\ 1 & 1 & . & . \end{bmatrix}.$$

Очевидно, что значность алфавита увеличивает число строк, спрятанных в многозначном векторе, которые можно одновременно анализировать в процессе распознавания образов, описанных упорядоченной совокупностью логических векторов.

С помощью свертываемых в многозначный вектор единичных матриц можно легко создавать эталоны символов, цифр, знаков, геометрических фигур, в дальнейшем используемые для создания более сложных образных эталонов. Некоторые из них (N, I, C, O) представлены ниже:

$$A = \begin{matrix} 00 \\ 01 \\ 10 \\ 11 \end{matrix} \begin{bmatrix} 1 & . & . & 1 \\ 1 & 1 & . & 1 \\ 1 & . & 1 & 1 \\ 1 & . & . & 1 \end{bmatrix} \begin{bmatrix} . & 1 & 1 & 1 \\ . & . & 1 & . \\ . & . & 1 & . \\ . & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & . & . & . \\ 1 & . & . & . \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & . & . & 1 \\ 1 & . & . & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}.$$

Распознавание некорректно написанного тестового символа C на множестве эталонов дает матричный результат:

$$\begin{aligned}
& \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & . & . \\ 1 & 1 & . & . \\ 1 & 1 & 1 & 1 \end{bmatrix} \Delta \begin{bmatrix} 1 & . & . & 1 \\ 1 & 1 & . & 1 \\ 1 & . & 1 & 1 \\ 1 & . & . & 1 \end{bmatrix} \begin{bmatrix} . & 1 & 1 & 1 \\ . & . & 1 & . \\ . & . & 1 & . \\ . & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & . & . & . \\ 1 & . & . & . \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & . & . & 1 \\ 1 & . & . & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} = \\
& = \begin{bmatrix} . & 1 & . & . \\ . & . & . & 1 \\ . & 1 & 1 & 1 \\ . & 1 & . & . \end{bmatrix} \begin{bmatrix} 1 & . & . & . \\ 1 & 1 & 1 & . \\ 1 & 1 & 1 & . \\ 1 & . & . & . \end{bmatrix} \begin{bmatrix} . & . & . & . \\ . & 1 & . & . \\ . & 1 & . & . \\ . & . & . & . \end{bmatrix} \begin{bmatrix} . & . & . & . \\ . & 1 & . & 1 \\ . & 1 & . & 1 \\ . & . & . & . \end{bmatrix} \rightarrow \\
& \mu = \min(\mu_1, \mu_2, \mu_3, \mu_4) = \begin{bmatrix} . & . & . & . \\ . & 1 & . & . \\ . & 1 & . & . \\ . & . & . & . \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & . & . & . \\ 1 & . & . & . \\ 1 & 1 & 1 & 1 \end{bmatrix} \rightarrow C.
\end{aligned}$$

В аналитической форме последнее выражение можно записать в следующем виде: $\mu = \min(\mu_1, \mu_2, \mu_3, \mu_4) = [\emptyset P \emptyset \emptyset] \rightarrow [YSSS] \rightarrow C$, что позволяет идентифицировать входное изображение как букву С.

Таким образом, функция распознавания входного образа m относительно имеющихся в наличии эталонов множества A может быть представлена в следующем виде:

$$\mu(m, A) = A_i[\min(m \Delta A_i)]_{i=1}^n.$$

Иначе, функционал, определяющий максимально сходный с эталоном образ, обращает критерий качества взаимодействия в минимальное число пиксельных или координатных несовпадений. В двоичном варианте описания образов симметрическая разность вырождается в хог-функцию:

$$\mu(m, A) = A_i[\min(m \oplus A_i)]_{i=1}^n.$$

Практическая значимость и проблемы для решения:

1. Аналитическая и матричная оценки степени сходства преобразов или взаимодействия объектов любой природы вычисляются параллельно, путем обработки векторов или матриц, использующих операцию симметрической разности, вычисляемую за один временной такт благодаря использованию мультимикропроцессора.

2. Наглядность матричной оценки в фрагментах изображения, где существуют различия, дает возможность корректировать не только анализируемый объект, но и эталонные примитивы на стадии создания библиотеки образов.

3. Технология распознавания образов, основанная на матричном (векторном) представлении изображений, ориентирована на использование матричных параллельных вычислений путем применения микропроцессора или неарифметического микропроцессора.

4. Переход от матричного к аналитическому образу и наоборот обеспечивает как визуализацию процесса распознавания образов, так и высокое быстродействие параллельного выполнения основной и единственной векторной операции симметрической разности (исключающее ИЛИ для двоичного алфавита).

5. Модели и метод векторно-матричного распознавания образов допускают сканируемое по галсам распознавание масштабных изображений окном допустимой для пользователя размерности.

6. Векторно-логическое представление изображения дает возможность существенно минимизировать объем памяти для его хранения, если соседние векторы имеют незначительные отличия друг от друга.

7. Нормализация изображений для последующего сравнения образов сводится к выполнению операций сдвига по трем степеням свободы и масштабированию к размерности эталонных примитивов, представленных в библиотеке.

8. Матричный образ, представленный двоичным кодом, является одним из возможных вариантов применения технологии распознавания, в то время как все объекты в кибернетическом пространстве, по сути, есть цифровые образы, к которым применима универсальная модель векторно-логического процесса распознавания, предложенная выше.

9. Стандартизация моделей образов. Разбиение образа сеткой на сегменты позволяет масштабировать рисунки любой размерности для их приведения к матрице одной размерности, имеющей одинаковое число координат, которые идентифицируются двоичными или многозначными значениями, составляющими сигнатуру изображения. Это дает возможность стандартизировать не только модели образов, пригодные для сравнения, но и модели критериев сходства, которые имеют такой же формат данных. Иначе, модели входных образов, эталонов и критериев должны иметь одинаковый формат данных. Имея два любых из упомянутых компонентов, можно легко восстановить третий. Например, взаимодействие матриц одной размерности теста, функциональности и дефектов в техническом изделии может быть представлено тремя полезными равенствами:

$$T \oplus M \oplus F = 0 \rightarrow \begin{cases} T = M \oplus F; \\ M = T \oplus F; \\ F = T \oplus M. \end{cases}$$

В общем случае уравнение взаимодействия трех объектов:

$$m \oplus A \oplus Q = 0 \rightarrow \begin{cases} Q = m \oplus A; \\ m = A \oplus Q; \\ A = m \oplus Q \end{cases}$$

– образа, эталона и критерия качества отношения между ними – формулирует, а значит, решает сотни практически полезных задач, включая: принятие решений, распознавание образов, тестирование и диагностику процессов и явлений.

Области применения векторно-логической технологии анализа процессов или явлений: 1) распознавание текстов в регистрационных картах пересечения границы; 2) идентификация личности по фотографиям, близким к стандартным снимкам для получения визовых документов; 3) поиск аналогов в Интернете по заданным образам; 4) сортировка изображений в базе данных по классам и признакам; 5) дактилоскопия и создание классифицированной библиотеки спецслужб; 6) распознавание и классификация программных вирусов; 7) распознавание почерков и идентификация личности по существенным признакам написания букв; 8) идентификация наземных целей и движущихся объектов (самолеты, корабли, машины); 9) синтез или корректировка образов по характерным существующим признакам; 10) применение технологии векторно-матричного (-логического) распознавания для управления робототехнических комплексов; 11) распознавание образов обоняния, вкуса, звуковых, тепловых и радиочастотных излучений; 12) распознавание лингвистических конструкций и примитивов, а также их оценивание при сравнении с эталонами; 13) динамическая визуализация эффектов последовательного и мягкого трансформирования одного образа в любой другой.

7. Заключение

1. Кибернетическое пространство как объект природы подвержен влиянию деструктивных средств, влияющих на работоспособность субъектов, которыми являются компьютеры, системы и сети. Поэтому важной проблемой остается стандартизация и спецификация пространства и всех взаимодействующих субъектов, включая негативные. Данная акция есть постоянно действующая во времени, цель которой – не отставать, но на один шаг опережать появление новых деструктивных компонентов, что обеспечит функционирование компьютерной экосистемы планеты и качества жизни человечества.

2. В связи с этим инфраструктура кибернетического пространства, метрика его измерения и процесс-модели анализа и синтеза субъектов дают возможность создавать эффек-

тивные решения для компьютерных изделий, ориентированных на быстрый поиск, распознавание, диагностирование не только позитивных, но и негативных субъектов. Конкретно, инфраструктура призвана решать задачи: 1) Описание многообразия деструктивных компонентов кибернетического пространства. 2) Формализация процессов взаимодействия триады компонентов <программа (аппаратура), функциональные нарушения, тесты>. 3) Диагностирование и устранение функциональных нарушений. 4) Создание и эффективное использование базы деструктивных данных или функциональных нарушений. 5) Создание быстродействующих интеллектуальных саморазвивающихся средств сервисного обслуживания и защиты кибернетического пространства.

3. Компьютер, как субъект экосистемы, имеет собственный путь развития во взаимодействии с остальным миром. Ученым не следует заниматься проблемой очеловечивания компьютера. Но компьютер нуждается в помощи человека в части наделения его способностью к саморазвитию (творчеству) путем создания стандартизированной инфраструктуры информационно-компьютерного пространства, включающей опыт человечества, оформленный в виде всемирной библиотеки стандартизированной математической, технологической и интеллектуальной культуры.

4. Информационно-компьютерная экосистема планеты есть материальная структура, ориентированная на повышение качества жизни каждого человека. При этом персонализация компьютера трансформируется в индивидуализацию, а далее – в личное киберпространство, что имеет целью создать интеллектуального и верного «друга», способного обслуживать человека 24 часа в сутки по всем сервисам, связанным с бытом, бизнесом, отдыхом, здоровьем.

5. Информационное векторное логическое пространство как подмножество метрического регулирует взаимодействие конечного числа объектов с помощью введенных определений, аксиом тождественности, симметрии и транзитивности треугольника. При этом последнее свойство вырождается в строгое равенство, что дает возможность потенциально уменьшить на треть объемы двоичной информации о взаимодействии объектов, благодаря свертке любого замкнутого логического пространства в нуль-вектор.

6. Рыночная привлекательность обозначенных проблем заключается в ориентации математической и технологической культуры на создание метрики кибернетического пространства, инфраструктуры сервисного обслуживания в виде моделей, методов и средств, включающих быстродействующие мозгоподобные компьютеры, критерии качества взаимодействия объектов в пространстве при поиске, распознавании и принятии решений, а также новые сервисы со стороны кибернетического пространства и индивидуального интеллектуального компьютера (киберпространства).

7. Актуальные проблемы создания саморазвивающейся информационно-компьютерной экосистемы: 1) Синтез минимального количества элементарных функций, способных покрыть все коды изображения для минимизации информационного объема, передаваемого по каналу. 2) Уменьшение объемов информации и минимизация структур путем использования принципа дополнения в избыточности замкнутых алфавитов. 3) Формирование решений в причинно-следственном покрытии на основе максимизации критерия качества взаимодействия вектор-спецификации и библиотечных примитивов, которые должны быть упорядочены по разделам или кодам и представлены в форме двоичного дерева. 4) Разработка формальных методов синтеза функциональностей по таблице истинности, которая в совокупности составляет спецификацию проекта. 5) Создание новых методов сжатия данных, основанных на использовании метрики векторно-логического пространства, для передачи информации по телекоммуникационным системам. 6) Разработка новых методов криптоанализа на основе использования метрики векторно-логического пространства и принципа свертки кодов расстояний в нулевую сумму.

Список литературы: 1. Babulak E. Future Global Office // 12th International Conference “Computer Modelling and Simulation”. 2010. P. 352–356. 2. Caplan K., Sanders J.L. Building an international security standard // IT Professional. Vol. 1, Issue 2. 1999. P. 29–34. 3. Qishi Wu, Ferebee D., Yunyue Lin, Dasgupta D. Visualization of security events using an efficient correlation technique // Computational Intelligence in Cyber Security, CICS '09. 2009. P. 61–68. 4. Инфраструктура мозгоподобных вычислительных процессов / М.Ф. Бондаренко, О.А. Гузь, В.И. Хаханов, Ю.П. Шабанов-Кушнаренко. Харьков: Новое Слово. 2010.

160 с. **5.** Хаханов В.И., Чумаченко С.В. Модели пространств в научных исследованиях // Радиоэлектроника и информатика. 2002. №1. С. 124-132. **6.** IEEE Standard for Reduced-Pin and Enhanced-Functionality Test Access Port and Boundary-Scan Architecture IEEE Std 1149.7-2009. 985 p. **7.** Da Silva F., McLaurin T., Waayers T. The Core Test Wrapper Handbook. Rationale and Application of IEEE Std. 1500™. –Springer. – 2006. XXIX. 276 p. **8.** Marinissen E.J., Yervant Zorian. Guest Editors' Introduction: The Status of IEEE Std 1500. IEEE Design & Test of Computers. 2009. No26(1). P. 6-7. **9.** IEEE Std 1800-2009 IEEE Standard for System Verilog-Unified Hardware Design, Specification, and Verification Language. <http://ieeexplore.ieee.org/servlet/opac?punumber=5354133>. **10.** Marinissen E.J. Testing TSV-based three-dimensional stacked ICs // DATE 2010. 2010. P.1689-1694. **11.** Benso A., Di Carlo S., Prinetto P., Zorian Y. IEEE Standard 1500 Compliance Verification for Embedded Cores // IEEE Trans. VLSI Syst. 2008. No 16(4). P. 397-407. **12.** Ubar R., Kostin S., Raik J. Embedded diagnosis in digital systems // 26th International Conference "Microelectronics", MIEL 2008. 2008. P. 421-424. **13.** Elm M., Wunderlich H.-J. Scan Chain Organization for Embedded Diagnosis // Design, Automation and Test in Europe, DATE '08. 2008. P. 468-473. **14.** Проектирование и тестирование цифровых систем на кристаллах. Verilog & System Verilog / В.И. Хаханов, Е.И. Литвинова, О.А. Гузь. Харьков: ХНУРЭ. 2009. 484с. **15.** Проектирование и верификация цифровых систем на кристаллах / В.И. Хаханов, И.В. Хаханова, Е.И. Литвинова, О.А. Гузь. Харьков: Новое слово. 2010. 528с. **16.** Семенец В.В., Хаханова И.В., Хаханов В.И. Проектирование цифровых систем с использованием языка VHDL. Харьков: ХНУРЭ. 2003. 492 с. **17.** Хаханов В.И., Хаханова И.В. VHDL+Verilog = синтез за минуты. Харьков: ХНУРЭ. 2006. 264с. **18.** Zorian Yervant. Guest Editor's Introduction: Advances in Infrastructure IP // IEEE Design and Test of Computers. 2003. P.49-55. **19.** Bulent I. Dervisoglu. A Unified DFT Architecture for Use with IEEE 1149.1 and VSIA/IEEE P1500 Compliant Test Access Controllers. Proceedings of the Design Automation Conference. 2001. P. 53-58. **20.** Bergeron J. Writing Testbenches using SystemVerilog. Springer US.2006. 414 p. **21.** Shibata T. Implementing brain-like systems using nano functional devices // Ultimate Integration of Silicon, ULIS 2009. 2009.P. 131-134. **22.** Soliman M.I., Al-Junaid A.F. Codevelopment of Multi-level ISA and hardware for an efficient matrix processor // International Conference Computer Engineering & Systems.2009. P. 211-217. **23.** Senning C., Studer C., Luethi P., Fichtner W. Hardware-efficient steering matrix computation architecture for MIMO communication systems // IEEE International Symposium Circuits and Systems. 2008. P. 304-307. **24.** Pedram Ardavan, Daneshalab Masoud, Sedaghati-Mokhtari Nasser, Fakhraie Sied Mehdi. A High-Performance Memory-Efficient Parallel Hardware for Matrix Computation in Signal Processing Applications // International Symposium Communications and Information Technologies. ISCIT '06. 2006. P. 473-478. **25.** Chenlong Hu, Ping Yang, Ying Xiao, Shaoxiong Zhou. Hardware design and realization of matrix converter based on DSP & CPLD // 3rd International Conference Power Electronics Systems and Applications. 2009. P. 1-5. **26.** Dave N., Fleming K., Myron King, Pellauer M., Vijayaraghavan M. Hardware Acceleration of Matrix Multiplication on a Xilinx FPGA // 5th IEEE/ACM International Conference Formal Methods and Models for Codesign. 2007. P.97-100. **27.** Loucks W.M., Snelgrove M., Zaky S.G. A Vector Processor Based on One-Bit Microprocessors // IEEE Micro. Volume 2, Issue 1. 1982. P. 53-62. **28.** Hilewitz Y., Lauradoux C., Lee R.B. Bit matrix multiplication in commodity processors // International Conference Application-Specific Systems, Architectures and Processors. 2008. P. 7-12.

Поступила в редколлегию 16.02.2011

Tiesoura Yves, аспирант кафедры АПВТ ХНУРЭ. Научные интересы: техническая диагностика цифровых систем и сетей. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-326. E-mail: hahanov@kture.kharkov.ua.

Чумаченко Светлана Викторовна, д-р техн. наук, проф. кафедры АПВТ ХНУРЭ. Научные интересы: техническая диагностика цифровых систем и сетей. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-326. E-mail: ri@kture.kharkov.ua.

Хаханов Владимир Иванович, декан факультета КИУ ХНУРЭ, д-р техн. наук, проф. кафедры АПВТ ХНУРЭ. Научные интересы: техническая диагностика цифровых систем, сетей и программных продуктов. Увлечения: баскетбол, футбол, горные лыжи. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-326. E-mail: hahanov@kture.kharkov.ua.