А. Ю. Сизов

ЗАЩИТА ИНФОРМАЦИОННЫХ ПРОЦЕССОВ МЕТОДОМ КОНТРОЛЯ ЭКВИВАЛЕНТНОСТИ В ФИКСИРОВАННЫХ КЛАССАХ (ОБ ОДНОМ МЕТОДЕ УСТАНОВЛЕНИЯ ЭКВИВАЛЕНТНОСТИ ИНФОРМАЦИОННЫХ ПРОЦЕССОВ)

Введение

Объект информационного процесса или информационной технологии как объекта защиты информации в последнее время все чаще рассматривается как отдельный объект защиты в информационных системах, наряду с классическими понятиями защиты данных, каналов и ПЭВМ. Впервые информационные технологии как процессы создания и жизни объекта, несущего в себе некий информационный факт, были упомянуты в работах В. А. Конявского в 2000 году [1, 2]. Данные работы являются базой, в которой формализуется понятие информационной технологии, приводится обоснование роли такого объекта и количественные характеристики повышения уровня защищенности при внедрении механизмов контроля информационных технологий.

Следуя изложенной в этих исследованиях позиции, мы будем определять защиту информационных процессов или технологий как возможность установления последовательности методов, способов и операций, примененных к некоторому множеству информационных ресурсов, включающему в себя и объект, являющийся результатом реализации информационной технологии, гарантированность неизменности этой последовательности, и возможность установления ее легитимности в рамках вычислительной системы.

Общие вопросы эквивалентности математических структур рассматриваются в различных областях теоретических и прикладных наук с древних времен. Наиболее существенными и основополагающими среди них можно назвать труды А. А. Маркова, Э. Поста. В своих трудах [3, 4] Марков дает представления об ассоциативных исчислениях, в которых неразрешима проблема установления эквивалентности.

Исследование вопросов эквивалентности в настоящее время нашло свое отражение в исследованиях эквивалентных преобразований схем программ (э.п. программ) Р. И. Подловченко [5, 6] и В. А. Захарова [7], эквивалентности многоленточных автоматов В. Е. Хачатряна [8, 9], вирусологии В.А. Захарова.

Цель этой работы — предложить механизм обеспечения защищенности информационных процессов методом установления эквивалентности независимо образованных объектов информационной систем (сообщений), несущих в себе идентичный информационный факт.

1. Информационный процесс

Введем основные понятия.

Информационная технология по [2] — это информационный процесс создания или обработки объекта электронного взаимодействия, определяемый последовательностью методов, способов и операций, продекларированных в рамках информационной системы.

Зададим абстрактное отображение ζ, ставящее в соответствие операции информационной среды o из O (создания, обработки и пр.) некоторый атрибут $a \in$ A:

$$\zeta : o \to a,$$

являющееся биекцией. Вследствие применения этого отображения образуется последовательность атрибутов $a_1 a_2 \ldots a_n$.

Дополним сказанное отображением

$$\xi : a \to t, \quad t \in \mathrm{T},$$

устанавливающим взаимно однозначное соответствие между атрибутами записи информационной технологии и символами некоторого алфавита T.

Таким образом, возможная интерпретация есть представление информационных технологий как последовательности применения операций над сообщением и фиксации уникальных атрибутов — $a_1a_2... a_n$ как слов $t_1t_2... t_n$ в некотором алфавите T в структуре сообщения.

Заметим, что формирование слова в алфавите T производится естественным образом: необходимость фиксации применения некоторого метода к сообщению влечет за собой добавление нового символа справа в запись слова.

Описание правил сравнения и установления эквивалентности двух абстрактных информационных технологий, определяющих равнозначность применения последовательностей операций и представленных словами $c^1{}_1...c^1{}_t$ и $d^1{}_1...d^1{}_s$, в рамках заданных требований защищенности осуществляется заданием полного набора допустимых атрибутов $t \in T$, а система преобразований Θ имеет вид:

$$c^1{}_1...c^1{}_t \sim d^1{}_1...d^1{}_s$$
$$\dots$$
$$c^v{}_1...c^v{}_t \sim d^v{}_1...d^v{}_s,$$

где $c^i{}_j, d^i{}_k \in T, \quad 1 \le i \le v, 1 \le j \le t, 1 \le k \le s$.

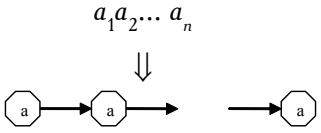
Рассмотрим представление элементов системы отношений установления эквивалентности в виде графа:

Основываясь на последовательном формировании атрибутов, правила представления можно описать следующим образом:

1) каждому атрибуту информационной технологии ставится в соответствие вершина графа;

2) последовательность атрибутов обозначается дугой, направленной в сторону формирования информационной технологии.

При таком представлении информационная технология будет иметь вид ветви:



Система преобразований отношений Θ в общем виде примет вид неориентированного графа, возможно содержащего замкнутые пути, с сохранением пометки вершин и существованием дуги при наличии эквивалентной связи.

Рассмотрим характерные особенности информационных технологий:

Утверждение 1 (ограничение линейности).

Ограничением формирования атрибутов информационной технологии (добавления нового символа в запись слова), является **ограничение линейности** фиксации применяемых к некоторому сообщению операций.

Пусть информационная технология описана последовательностью $t_1...t_{k-1}$ применения некоторых операций над сообщением. Тогда какова бы ни была очередная операция, характеризуемая символом t_k, его запись в слово в ходе формирования информационной технологии не способно изменить никакой элемент последовательности $t_1...t_{k-1,}$...

Данное утверждение является естественным в силу определения информационной технологии, данного в [2].

Определение 1. Две различные информационные технологии, в процессе применения которых создаются некоторые сообщения, несущие идентичные сведения или информационные факты, называются эквивалентными, если соответствующие им последовательности атрибутов

$a_1...a_n$, а, следовательно, и последовательность $t_1...t_n$, формирующихся в соответствии с операциями из O, будут эквивалентны, как слова в алфавите T.

Утверждение 1' (правило левого сокращения). Для эквивалентных информационных технологий применимо правило левого сокращения: если в эквивалентных информационных технологиях эквивалентны их префиксы, тогда эквивалентны и суффиксы этих информационных технологий, полученные отбрасыванием этих префиксов.

Определение 2. Операция модификации информационной технологии, определенная при заданном множестве допустимых операций O и системы отношений установления эквивалентности Θ, есть операция замещения последовательности $t^1{}_1...t^1{}_n$ на $t^2{}_1...t^2{}_m$ или наоборот, при $t^1{}_1...t^1{}_n \sim t^2{}_1...t^2{}_m \in \Theta$.

Определение 3 (условная эквивалентность для системы Θ).

Отношения установления эквивалентности виде $c_1...c_t \sim d_1...d_s$, при условии удаления первых идентичных символов записи такого отношения $c_i = d_i$, где $1 \le i \le k, k < \min(t,s)$, формирует условную эквивалентность $c_{k+1}...c_t \tilde{\sim} d_{k+1}...d_s$, а $c_1...c_t \sim d_1...d_s$ — условие этой эквивалентности.

Показано, что применение таких структур при преобразовании исходных информационных технологий допустимо только в случае установления вхождения условия в описание информационной технологии.

Утверждение 2. Фиксация конкретной пары множеств T и Θ является достаточным условием определения правил защищенности информационных технологий.

Данное утверждение очевидно в силу определения защищенности, приведенного во введении, и того факта, что задание пары (T, Θ) является заданием вариационного исчисления.

Определение 4. Класс информационных технологий — множество информационных технологий, для которых при:

1) заданном множестве допустимых операций $o \in O$ (или $\{t_1...t_n\}\ t \in T$);

2) заданном множестве отношений установления эквивалентности операций Θ;

3) заданном экземпляре информационной технологии выполнено ℓ:

для любой информационной технологии, принадлежащей классу, существует цепочка преобразований модификации из отношений Θ, переводящей ее в экземпляр ℓ.

2. Алгоритм установления эквивалентности информационных технологий

Проблема установления эквивалентности двух эквивалентных информационных технологий сводится к поиску конечного алгоритма, строящего цепочку преобразований на основании системы Θ и переводящего одну информационную технологию в другую. Если же информационные технологии не эквивалентны, алгоритм должен обнаруживать это.

Заданы две информационные технологии $ИТ_1 = a_1...a_n$, $ИТ_2 = b_1...b_m$ в рамках общего алфавита T.

Система отношений эквивалентности Θ имеет вид:

$c^1{}_1...c^1{}_t \sim d^1{}_1...d^1{}_s$

…

$c^v{}_1...c^v{}_t \sim d^v{}_1...d^v{}_s$

Проверять эквивалентность будем методом сведения одной информационной технологии к другой, используя отношения из Θ и введенное правило модификации из определения 2. Не ограничивая общности, будем сводить к $ИТ_2$.

Описание алгоритма (шаг алгоритма):

проверяем идентичность символов ($a_i = b_j$), где $1 \le i \le n, 1 \le j \le m$, (для первого шага, $i = 1, j = 1$).

1 если символы идентичны — вычеркиваем эти символы из обоих слов и переходим к следующему шагу алгоритма с $i = i+1, j = j+1$,

2 если символы не идентичны, из Θ находим все отношения эквивалентности, первый символ которого есть b_j (например, $b_1c_2...c^r{}_t \sim d^r{}_1...d^r{}_s$)

2.1 если такие отношения эквивалентности существуют, переходим к подзадаче определения эквивалентности вида $a_i...a_n$ и $d^r{}_1...d^r{}_s$.

2.2 если такого отношения эквивалентности не существует — две представленные информационные не эквивалентны.

3 Применяем алгоритм к подзадачам до тех пор, пока не получаем соотношение вида $a_{r+1}...a_n \sim 0$, что свидетельствует о существовании цепочки преобразований от $a_i...a_n$ к $d_1...d_r a_{r+1}...a_n$. Заменяем в основной задаче (или подзадаче) $a_1...a_r$ на $b_1...c^r{}_t$, и возвращаемся к основной задаче (подзадаче более высокого уровня).

При этом на каждом уровне подзадач могут возникать дополнительные задачи вида $c_2...c^r{}_t \sim d^r{}_2...d^r{}_s$ подзадачи $b_1...c^r{}_t \sim d^r{}_1...d^r{}_s$. Т. е. любая подзадача некоторого уровня может приносить дополнительные задачи.

Конец алгоритма.

Приведем пример работы данного алгоритма на примере: пусть $ИТ_1$ = abcde и $ИТ_2$ = akph, Θ = { cde ~ fh, bf~kp }. Будем обозначать пункты *x.y...* работы алгоритма как *x* — номер шага алгоритма входной задачи, *y* — номер шага подзадачи.

Работа алгоритма:

1. abcde и akph *(первые символы равны — удаляем)*

2. bcde и kph *(выбираем из Θ kp~bf — подзадача для получения kp)*

2.1 bcde и bf *(первые символы равны — удаляем)*

2.2 cde и f *(выбираем fh~cde — подзадача для получения fh)*

2.2.1 cde и cde *(получил символы $ИТ_2$, кроме символа h, который внесется в запись при переходах к подзадачам более высокого уровня)*

⇒ эквивалентность технологий установлена в силу сведения к .

Теорема 1.

Представленный алгоритм корректно устанавливает эквивалентность *ИТ*.

Доказательство: очевидно в силу определения информационной технологии и построения алгоритма, проводящего проверку всех отношений из Θ. Проведем доказательство от противного:

Пусть ∃ информационная технология *ИТ*, эквивалентная ℓ, но не существует цепочки преобразований, получаемой в результате применения описанного алгоритма, переводящего *ИТ* в ℓ. Тогда алгоритм на некотором шаге должен не обнаружить отношение из Θ, осуществляющего переведение *i*-го символа *ИТ* в *j*-ий символ ℓ. В силу того что, на шаге алгоритма 2.1 производится выбор всех возможных отношений, удовлетворяющих поиску, тогда множество отношений системы Θ задано не верно, и *ИТ* не эквивалентна ℓ в рамках заданного класса.

3. Необходимые условия эквивалентности информационных процессов

Рассмотрим вопрос конечности алгоритма, приведенного в разделе 3. Исследование будем проводить методом варьирования значений переменных, характеризующих некоторую задачу установления эквивалентности информационных технологий. В терминах условия алгоритма (1):

1) отсутствие ограничений на длины информационных технологий *n* и *m*;

2) последовательном рассмотрении совокупности параметров max(*s*, *t*) — максимум длины левой или правой части отношений в Θ и *υ* — количества отношений в Θ.

Рассмотрим несколько вариантов варьирования переменных:

1) пусть $\max(s, t)=1$ и $\forall v$, тогда работа алгоритма будет бесконечна при существовании в Θ отношений $\{b\sim c, c\sim d, d\sim b\}$ на входных $ИТ_1 = a...$ и $ИТ_2 = b...$

2) пусть $\max(s, t)=2$ и $v = 2$, тогда исключая случай из 1), находим еще одну конструкцию из Θ – $\{c\sim dc, f\sim df\}$, которая на словах $ИТ_1 = f...$ и $ИТ_2 = c...$ переведет работу алгоритма в бесконечный цикл.

Проведенный анализ дает основания заключить, что представленный алгоритм не является конечным, и выделяются два характерных случая, описывающие это факт:

1) «бесконечный поиск» – пункта 2.1 алгоритма;

2) «бесконечная модификация» – пункт 3 алгоритма, формирующая подзадачу некоторого уровня, эквивалентную задаче более высокого уровня при неизменности сводимой последовательности $a_1...a_n$, начиная с получения первой такой подзадачи.

Пусть все отношения в Θ пронумерованы, тогда множество$_1 = \{t_i\}$ – все операции из левой и правой части соотношения.

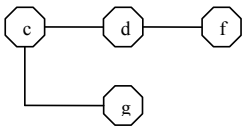
Определение 5. Множество отношений установления эквивалентности Θ будем называть **связанным**, если ∃ непустое пересечение любых двух определенных выше множеств.

Утверждение 3. Для пустого или несвязанного множества Θ данный алгоритм конечен.

Определение 6. Графом Γ^1 отношений установления эквивалентности будем называть граф, вершины которого помечены первыми символами из левой и правой части отношений из Θ, а существование дуги из вершины a в вершины b означает существование в Θ отношения вида $aa_2...a_n\sim bb_2...b_m$.

Единственным исключением из этого правила могут быть замкнутые пути длины 1, т. е. – т. е. $a_1a_2...a_n\sim a_1b_2...b_m$ отношения условной эквивалентности.

Например, для $\Theta = \{c,c_2,...c_{n_1}\sim d,d_2,...d_{m_1},\ d,e_2...e_{n_2}\sim f,f_2...f_{m_2},\ c,c_2,...c_{n_3}\sim g,g_2,...g_{m_3}\}$, граф Γ^1 примет вид:



Лемма 1. Отсутствие замкнутых путей в графе Γ^1 обеспечивает конечность алгоритма для случая «бесконечного поиска».

Доказательство очевидно.

Определение 7. Графом Γ^2 отношений установления эквивалентности будем называть граф, являющейся дополнением Γ^1 дугами, существование которой из вершины a в вершины b означает существование в Θ отношения: $aa_2...a_n\sim bb_2...b_m$ или $aa_2...a_n\sim b_1...b...b_m$.

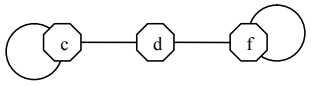
Лемма 2. Отсутствие замкнутых путей в графе Γ^2 обеспечивает конечность алгоритма для случая «бесконечной модификации».

Доказательство аналогично доказательству Леммы 1 и строится на анализе возможной структуры графа Γ^2 и применении к нему алгоритма установления эквивалентности информационных технологий.

Следствие.

Анализируя условие Леммы 2 можно сделать вывод, что данное условие является слишком сильным для ограничения возникновения бесконечного цикла в процессе работы алгоритма.

Рассмотрим граф Γ^2 для $\Theta = \{c\sim dc,\ f\sim df\}$



Данная конструкция характеризует минимальную по длине путей структуру, описывающую рассматриваемый случай. Она задает существованием в графе Γ^2 двух связанных замкнутых путей, причем как минимум один уз этих путей будет отсутствовать в графе Γ^1.

На основании условий Леммы 1 и Леммы 2 формируется класс K с определенным алфавитом T и системой Θ.

Теорема (необходимые условия разрешимости эквивалентности для информационных технологий). Для установления эквивалентности произвольных информационных технологий в некотором классе с заданной системой отношений установления эквивалентности Θ должно быть выполнено: представление системы Θ в виде графа Γ^1 не должно содержать замкнутых путей, а представление системы в виде графа Γ^2 не должно содержать двух связанных замкнутых путей.

Заключение

Проведена формализация понятий информационного процесса, или информационной технологии, для решения задачи установления эквивалентности таких объектов. Описывается один из классов информационных технологий с разрешимой задачей установления эквивалентности, ограниченный жесткими требованиями к структуре формирования системы отношений, описывающей эквивалентные преобразования информационных технологий. Ограничения накладываются на систему отношений, представленную в виде графа, и характеризуются отсутствием замкнутых путей специального вида. Проведено исследование работы алгоритма: выделены условия для входных данных, приводящих к образованию бесконечного цикла. Показано, что данные условия носят вероятностный характер относительно входных слов.

СПИСОК ЛИТЕРАТУРЫ:

1. *Конявский В. А.* Возможное решение проблемы «оригинал-копия» для электронных документов // Безопасность информационных технологий. М., 2000. № 1.
2. *Конявский В. А., Гадасин В. А.* Основы понимания феномена электронного обмена информацией. Мн.: Беллитфонд, 2004.
3. *Марков А. А., Нагорный Н. М.* Теория алгорифмов. М.: Наука, 1984.
4. *Марков А. А.* О неразрешимых алгорифмических проблемах. Матем. сб. Новая сер. 1952.
5. *Подловченко Р. И.* Об одном массовом решении проблемы эквивалентных преобразований схем программ. I // Программирование. 2000. № 1. С. 64–77.
6. *Подловченко Р. И.* Об одном массовом решении проблемы эквивалентных преобразований схем программ. II // Программирование. 2000. № 2. С. 3–11.
7. *Захаров В. А.* Быстрые алгоритмы разрешения эквивалентности операторных программ на уравновешенных шкалах // Математические вопросы кибернетики. М.: Физматлит, 1998. Вып. 7. С.
8. *Хачатрян В. Е.* Полная система эквивалентных преобразований для многоленточных автоматов // Программирование. 2003. № 1. С. 62–77.
9. *Подловченко Р. И., Хачатрян В. Е.* Об одном подходе к разрешению проблемы эквивалентности // Программирование. 2004. № 3. С. 3–20.