И.Б.Бурдонов, А.С.Косачев.
Тестирование с преобразованием семантик.
Научный сервис в сети Интернет: суперкомпьютерные центры и задачи: Труды Всероссийской суперкомпьютерной конференции (21-24 сентября 2010 г., г. Новороссийск). - М.: Изд-во МГУ, 2010. с. 420-424.
5 стр.

Тестирование с преобразованием семантик

Бурдонов И.Б., Косачев А.С.

1. Введение

Тестирование на основе моделей проверяет соответствие между моделью тестируемой программы (реализацией) и моделью требований к ней (спецификацией). Обычно это соответствие определяется в предположении, что обе модели определены на одном уровне абстракции. Это означает, что они используют одни и те же понятия тестовых воздействий и наблюдений за ответным поведением реализации.

На практике модели реализации и спецификации оказываются определенными на разных уровнях абстракции, которые нужно согласовывать при тестировании. Для этого тестовое воздействие на реализацию преобразовывается с уровня абстракции спецификации на уровень абстракции реализации. Наоборот, каждое ответное наблюдение преобразовывается с уровня абстракции реализации на уровень абстракции спецификации. Эти преобразования в рамках тестовой системы выполняются специальной программой, называемой *медиатором*.

В данной работе исследуется вопрос о том, как меняется соответствие между реализацией и спецификацией с учетом медиаторного преобразования. Такое преобразование понимается как преобразование семантик взаимодействия.

2. Теория конформности без преобразования семантик

2.1. Формализация тестирования

Тестирование – это проверка в процессе эксперимента соответствия (конформности) реализации требованиям, заданным в виде спецификации. Это соответствие определяется семантикой тестового взаимодействия, которая описывает возможные тестовые воздействия и возможные наблюдения ответного поведения реализации. Будем задавать семантику парой (***T***,***N***), где ***T*** – множество (тестовых) воздействий, ***N*** – множество допустимых наблюдений.

Результатом каждого тестового эксперимента является чередующаяся последовательность воздействий и наблюдений, начинающаяся с воздействия и заканчивающаяся наблюдением, которую мы будем называть *историей*. Множество всех конечных историй обозначим через X. Определим *модель* реализации или спецификации как префикс-замкнутое подмножество $\Sigma \subseteq X$ историй: вместе с каждой историей множество Σ содержит любой ее префикс,

И.Б.Бурдонов, А.С.Косачев.
Тестирование с преобразованием семантик.
Научный сервис в сети Интернет: суперкомпьютерные центры и задачи: Труды Всероссийской суперкомпьютерной конференции (21-24 сентября 2010 г., г. Новороссийск). - М.: Изд-во МГУ, 2010. с. 420-424.
5 стр.

являющийся историей. Обозначим множество наблюдений после истории $\sigma\in\Sigma$ и воздействия t: $f_\Sigma(\sigma,t) =_{\text{def}} \{n\in\boldsymbol{N}|\sigma\cdot t\cdot n\in\Sigma\}$. Отображение f_Σ является одним из способов задания модели:

$\Sigma = \boldsymbol{h}(f_\Sigma) =_{\text{def}} \{\sigma|\forall i=2,4,...,|\sigma|\ \sigma(i)\in f(\sigma[1..i-2],\sigma(i-1))\}$.

Будем называть воздействие t *безопасным* после истории $\sigma\in\Sigma$, если $f_\Sigma(\sigma,t)\neq\varnothing$; в противном случае – опасным. Мы предполагаем, что если при тестировании после трассы σ выполняется безопасное воздействие t, то одно из таких наблюдений n обязательно будет получено через конечное время. Тестирование, при котором используются только безопасные воздействия, будем называть *безопасным*. Модель Σ содержит только безопасные истории.

Для того, чтобы тестирование было безопасным, реализация должна удовлетворять следующей *гипотезе о безопасности*: если после общей истории реализации $\Sigma^\wedge$ и спецификации Σ воздействие безопасно в спецификации, то оно безопасно в реализации (такая реализация называется *безопасно-тестируемой*): $\Sigma^\wedge$ ***safe for*** $\Sigma =_{\text{def}} \forall\sigma\in\Sigma\cap\Sigma^\wedge\ \forall t\in\boldsymbol{T}\ (f_\Sigma(\sigma,t)\neq\varnothing \Rightarrow f_{\Sigma^\wedge}(\sigma,t)\neq\varnothing)$.

Конформность определяется для безопасно-тестируемых реализаций и означает, что любое наблюдение, возможное в реализации при безопасном тестировании, разрешается спецификацией (возможно в ней) в аналогичной ситуации, то есть в ответ на то же самое воздействие после той же самой истории: $\Sigma^\wedge$ ***saco*** $\Sigma =_{\text{def}} \Sigma^\wedge$ ***safe for*** Σ & $\forall\sigma\in\Sigma\cap\Sigma^\wedge\ \forall t\in\boldsymbol{T}\ f_{\Sigma^\wedge}(\sigma,t)\subseteq f_\Sigma(\sigma,t)$.

2.2. LTS-модель

Множество Σ или отображение f_Σ являются теоретически безупречными, но практически неудобными способами задания модели, поскольку, как правило, число историй бесконечно. Более удобной моделью является LTS (Labelled Transition System) – ориентированный граф, в котором вершины называются состояниями, одно состояние выделено как начальное, а дуги помечены символами из некоторого алфавита $\boldsymbol{L}$ и называются переходами. В качестве таких символов мы выбираем пары (воздействие, наблюдение). Кроме того, используется символ τ, моделирующий самопроизвольное (независимое от воздействий) и ненаблюдаемое изменение состояния. Для различения опасных и безопасных воздействий вводится специальное наблюдение $\gamma\notin\boldsymbol{N}$, которое называется *разрушением*. Оно моделирует любое запрещённое при тестировании поведение реализации.

Формально, LTS-модель в $(\boldsymbol{T},\boldsymbol{N})$-семантике – это совокупность $S=\text{LTS}(V_S,\boldsymbol{L},E_S,s_0)$, где V_S – непустое множество состояний, $\boldsymbol{L}=\boldsymbol{T}\times(\boldsymbol{N}\cup\{\gamma\})$ – алфавит, $E_S\subseteq V_S\times(\boldsymbol{L}\cup\{\tau\})\times V_S$ – множество переходов, $s_0\in V_S$ – начальное

И.Б.Бурдонов, А.С.Косачев.
Тестирование с преобразованием семантик.
Научный сервис в сети Интернет: суперкомпьютерные центры и задачи: Труды Всероссийской суперкомпьютерной конференции (21-24 сентября 2010 г., г. Новороссийск). - М.: Изд-во МГУ, 2010. с. 420-424.
5 стр.

состояние. Для $a,b \in V_S$, $z \in \boldsymbol{L} \cup \{\tau\}$ обозначим: $a \text{—} z \rightarrow b =_{\text{def}} (a,z,b) \in E_S$ и $a \text{—} z \rightarrow =_{\text{def}} \exists b\ a \text{—} z \rightarrow b$.

Маршрутом LTS называется последовательность смежных переходов: начало каждого перехода, кроме первого, совпадает с концом предыдущего перехода. Когда к LTS применяется воздействие t, она выполняет переход $a \text{—}(t,n) \rightarrow b$ с наблюдением n, перед и после которого могут выполняться ненаблюдаемые τ-переходы. *История маршрута* – это последовательность воздействий и наблюдений, которыми помечены его переходы (с пропуском τ-переходов). Через S ***after*** σ обозначается множество состояний LTS S после истории σ, то есть множество конечных состояний всех маршрутов LTS S, начинающихся в начальном состоянии и имеющих историю σ.

Воздействие безопасно в реализации (в состоянии или после истории), если оно не может привести к разрушению и через конечное время гарантировано получение наблюдения. Можно выделить три условия опасности воздействия.

1) Если в состоянии a определен переход $a \text{—}(t,\gamma) \rightarrow$, то воздействие t опасно в состоянии a, а также в любом состоянии, из которого состояние a достижимо по τ-переходам.

2) Состояние a *дивергентно*, если в нём начинается бесконечный τ-маршрут, то есть маршрут, состоящий из τ-переходов, (в частности, τ-цикл). В противном случае состояние *конвергентно* (обозначается $a\!\downarrow$). Если LTS оказалась в дивергентном состоянии, то при любом воздействии она может выполнять только τ-переходы, и никакого наблюдения не будет. Поэтому в дивергентном состоянии любое воздействие считается опасным.

3) Состояние *стабильно*, если в нем не определены τ-переходы. Воздействие t считается опасным в стабильном состоянии a, если в этом состоянии не определены переходы по этому воздействию, то есть переходы вида $a \text{—}(t,z) \rightarrow$b, где $z \in \boldsymbol{N} \cup \{\gamma\}$.

В реализации воздействие опасно после истории, если оно опасно по любому из условий 1-3 хотя бы в одном состоянии после этой истории, поскольку реализация может оказаться в этом состоянии.

В спецификации для безопасности воздействия после истории также должны быть выполнены условия 1 и 2 для каждого состояния после истории, а условие 3 должно быть выполнено хотя бы в одном состоянии, поскольку в этом состоянии история спецификации имеет продолжение, которое будет проверяться при тестировании.

История безопасна, если в ней каждое воздействие безопасно после непосредственно предшествующего ему префикса. Множество безопасных историй LTS S, рассматриваемой как спецификация, обозначим ***sh***(S), а LTS S, рассматриваемой как реализация, обозначим $\boldsymbol{sh}^{\wedge}(S)$.

И.Б.Бурдонов, А.С.Косачев.
Тестирование с преобразованием семантик.
Научный сервис в сети Интернет: суперкомпьютерные центры и задачи: Труды Всероссийской суперкомпьютерной конференции (21-24 сентября 2010 г., г. Новороссийск). - М.: Изд-во МГУ, 2010. с. 420-424.
5 стр.

Для LTS S определим отображение f_S для спецификации и отображение $f_{S^\wedge}$ для реализации: $\forall\sigma\ \forall t{\in}\boldsymbol{T}\ \forall n{\in}\boldsymbol{N}$

$f_S(\sigma,t) = \{n{\in}\boldsymbol{N} | \exists a{\in}(S\ \textbf{\textit{after}}\ \sigma)\ a\text{—}(t,n)\text{→}\ \&\ \forall b{\in}(S\ \textbf{\textit{after}}\ \sigma)\ b{\downarrow}\ \&\ \neg(\text{b—}(t,\gamma)\text{→})\}$,

$f_{S^\wedge}(\sigma,t) = \{n{\in}f_S(\sigma,t) | \forall a{\in}(S\ \textbf{\textit{after}}\ \sigma)\ a\text{—}\tau\text{→}\ \vee\ \exists m\ a\text{—}(t,m)\text{→}\}$.

Тогда $\boldsymbol{sh}(S) = \boldsymbol{h}(f_S)$ и $\boldsymbol{sh}^\wedge(S) = \boldsymbol{h}(f_{S^\wedge})$.

2.3. Генерация тестов

Тест – это инструкция, в каждом пункте которой указывается воздействие, которое нужно выполнить, и для каждого наблюдения – пункт инструкции, который должен выполняться следующим, или вердикт (***pass*** или ***fail***), если тестирование нужно закончить. В [1] такая инструкция соответствует формальному определению *управляемого* теста, который однозначно определяет тестирование (без лишнего недетерминизма).

Реализация *проходит* тест, если её тестирование всегда заканчивается с вердиктом ***pass***. Реализация проходит набор тестов, если она проходит каждый тест из набора. Набор тестов *значимый*, если каждая конформная реализация его проходит; *исчерпывающий*, если каждая неконформная реализация его не проходит; *полный*, если он значимый и исчерпывающий. Задача заключается в генерации полного набора тестов по спецификации.

Полный набор тестов всегда существует, в частности, им является набор всех *примитивных* тестов. Такой тест строится по одной выделенной безопасной истории спецификации. Если наблюдение, полученное после воздействия, продолжает историю, тест продолжается. Последнее в истории наблюдение и любое наблюдение, «ведущее в сторону», всегда заканчивают тестирование. Вердикт ***pass*** выносится, если полученная история есть в спецификации, а вердикт ***fail*** – если нет. Такие вердикты соответствуют *строгим* тестам, которые, во-первых, значимые (не ловят ложных ошибок) и, во-вторых, не пропускают обнаруженных ошибок. Любой строгий тест сводится к некоторому множеству примитивных тестов, которые обнаруживают те же самые ошибки.

3. Преобразование воздействий и наблюдений

3.1. Медиатор семантик

В изложенной выше теории конформности предполагается, что реализация и спецификация заданы в одной (***T***,***N***)-семантике. На практике обычно требуется некоторое преобразование спецификационных воздействий в реализационные воздействия и обратное преобразование реализационных наблюдений в спецификационные наблюдения. Программа, осуществляющая эти

И.Б.Бурдонов, А.С.Косачев.
Тестирование с преобразованием семантик.
Научный сервис в сети Интернет: суперкомпьютерные центры и задачи: Труды Всероссийской суперкомпьютерной конференции (21-24 сентября 2010 г., г. Новороссийск). - М.: Изд-во МГУ, 2010. с. 420-424.
5 стр.

преобразования, называется медиатором. Это означает, что реализация может быть задана в другой ($\boldsymbol{T}^\wedge$,$\boldsymbol{N}^\wedge$)-семантике, и медиатор осуществляет преобразование семантик.

В простейшем случае семантики различаются только разными способами представления одних и тех же воздействий и наблюдений в реализации и спецификации. Достаточно взаимно-однозначного преобразования множеств $\boldsymbol{T}\leftrightarrow\boldsymbol{T}^\wedge$ и $\boldsymbol{N}\leftrightarrow\boldsymbol{N}^\wedge$. В общем случае такого преобразования недостаточно, поскольку спецификация, как правило, определяется на более высоком уровне абстракции. Рассмотрим несколько примеров.

1. Передача сообщений. В спецификации может быть определена операция (воздействие) ***send***(*m*) передачи сообщения *m*, а в реализации – ***send***(*m*,*i*) с указанием дополнительного параметра *i* – номера одного из нескольких портов. На уровне спецификации тест принимает любое сообщение от реализации, отсутствие сообщений понимается как особое наблюдение – отказ. На уровне реализации тест может принимать сообщение только из одного порта, и для наблюдения спецификационного отказа нужно выполнить прием по каждому порту и убедиться, что нигде сообщений нет. В этом примере одному спецификационному воздействию ***send***(*m*) соответствует множество реализационных воздействий ***send***(*m*,*i*), каждое из которых нужно выполнить в том же самом состоянии реализации для того, чтобы при полном тестировании осуществить требуемое спецификационное воздействие для всех возможных ситуаций. Реализационный отказ не меняет состояния реализации, но прием сообщения, вообще говоря, меняет состояние, и прием из другого порта можно будет сделать только тогда, когда реализация снова окажется в том же состоянии, а тест объявит прием сообщений. Таким образом медиаторное преобразование, во-первых, адаптивно – реализационные воздействия зависят от полученных реализационных наблюдений (сообщение или отказ), и, во-вторых, зависит от состояния реализации, в котором оно начинает выполняться.

2. Пул элементов с операциями ***alloc*** и ***dealloc***. В спецификации пула определена операция ***dealloc*** без параметров, понимаемая как освобождение любого ранее занятого элемента, а в реализации эта операция требует в качестве параметра указать конкретный освобождаемый элемент ***dealloc***(*i*). Пусть реализация удовлетворяет следующей гипотезе: все состояния пула, соответствующие одному и тому же множеству занятых элементов, эквивалентны. Аналогично предыдущему примеру одному спецификационному воздействию ***dealloc*** соответствует множество (по числу занятых элементов) реализационных воздействий ***dealloc***(*i*), каждое из которых нужно выполнить хотя бы в одном состоянии реализации, соответствующем данному множеству занятых атомов. Есть два отличия от предыдущего примера. 1) Медиаторное преобразование не адаптивно: от наблюдений (удаление *i*-го элемента пула или

И.Б.Бурдонов, А.С.Косачев.
Тестирование с преобразованием семантик.
Научный сервис в сети Интернет: суперкомпьютерные центры и задачи: Труды Всероссийской суперкомпьютерной конференции (21-24 сентября 2010 г., г. Новороссийск). - М.: Изд-во МГУ, 2010. с. 420-424.
5 стр.

отказ, когда *i*-ый элемент свободен) зависит только соответствующее спецификационное наблюдение, возвращаемое в тест, но не последующие реализационные воздействия, так как для каждого воздействия ***dealloc*** выполняется только одно воздействие ***dealloc***(*i*). 2) Преобразование зависит от множества занятых элементов, вычисляемое по предыстории взаимодействия. По гипотезе, зависимость от реализационного состояния можно не учитывать, поскольку состояния с одним множеством занятых элементов эквивалентны.

3. Стационарное тестирование. Спецификация описывает конечный автомат, каждый переход которого помечен парой (стимул $x\in I$, реакция $y\in O$); спецификационное воздействие $P(x)=\{(x,y)|y\in O\}$ означает посылку стимула x и прием любой реакции $y\in O$. Реализация представляет собой LTS, каждый (не τ-) переход которой помечен либо стимулом $x\in I$, либо реакцией $y^\wedge\in O^\wedge$; реализационные воздействия: $P^\wedge(x)=\{x\}$, означающее посылку стимула x, и $P^\wedge=O^\wedge$, означающее прием любой реакции. Медиатор посылает стимул x и либо возвращает отказ (стимул не принят, так как в состоянии реализации не определен переход по этому стимулу), либо принимает реакции до тех пор, пока они поступают от реализации, а затем для полученной (быть может, пустой) последовательности $y^\wedge_1,\ldots,y^\wedge_n$ реализационных реакций выдает как наблюдение спецификационную реакцию $y=M(y^\wedge_1,\ldots,y^\wedge_n)$. Функция M как раз и определяет медиаторное преобразование. Тестирование через такой медиатор иногда называют стационарным тестированием, поскольку стимулы подаются только в стационарных состояниях реализации – состояниях, в которых нет реакций [3]. Спецификационному воздействию соответствует последовательность реализационных воздействий, в процессе выполнения которой состояние реализации меняется, но, в отличие от первого примера, нас интересует состояние только в конце последовательности, соответствующее полному выполнению спецификационного воздействия. Преобразование адаптивно, но не зависит от реализационных состояний.

В общем случае медиатор может оказаться сложной программой, осуществляющей медиативные функции между реализацией в $(\boldsymbol{T}^\wedge,\boldsymbol{N}^\wedge)$-семантике и тестом, генерируемым по спецификации в $(\boldsymbol{T},\boldsymbol{N})$-семантике. Медиатор компонуется с тестом; результат такой композиции рассматривается как тест в $(\boldsymbol{T}^\wedge,\boldsymbol{N}^\wedge)$-семантике, взаимодействующий напрямую с реализацией.

3.2. Устройство теста и медиатора

Когда тест непосредственно взаимодействует с реализацией, все воздействия и наблюдения реализационные. При тестировании через медиатор с реализацией непосредственно взаимодействует медиатор, а интерфейс медиатора и теста будет другой. Работа теста, генерируемого по спецификации

И.Б.Бурдонов, А.С.Косачев.
Тестирование с преобразованием семантик.
Научный сервис в сети Интернет: суперкомпьютерные центры и задачи: Труды Всероссийской суперкомпьютерной конференции (21-24 сентября 2010 г., г. Новороссийск). - М.: Изд-во МГУ, 2010. с. 420-424.
5 стр.

в $(\boldsymbol{T},\boldsymbol{N})$-семантике, – это чередующаяся последовательность тестовых воздействий $t\in\boldsymbol{T}$ и полученных в ответ наблюдений $n\in\boldsymbol{N}$. В медиаторе между воздействием $t\in\boldsymbol{T}$ и наблюдением $n\in\boldsymbol{N}$ происходит взаимодействие медиатора с реализацией в $(\boldsymbol{T}^\wedge,\boldsymbol{N}^\wedge)$-семантике. Это взаимодействие отличается от взаимодействия теста и реализации в одной $(\boldsymbol{T}^\wedge,\boldsymbol{N}^\wedge)$-семантике: 1) вместо вынесения вердикта выдается спецификационное наблюдение, 2) взаимодействие начинается после получения спецификационного воздействия.

Состояния медиатора будем называть m-состояниями, они бывают трех типов. В m-состоянии 1-го типа принимаются спецификационные воздействия, причем медиатор готов принять любое воздействие $t\in\boldsymbol{T}$. После получения воздействия медиатор взаимодействует с реализацией в m-состояниях 2-го типа, и через конечное число шагов переходит в m-состояние 3-го типа, где выдает спецификационное наблюдение $n\in\boldsymbol{N}$ с переходом в m-состояние 1-го типа.

После получения при тестировании реализационной истории $\sigma^\wedge$ тест обращается к медиатору со спецификационным воздействием t, далее медиатор взаимодействует с реализацией, которая проходит после истории $\sigma^\wedge$ историю $\lambda^\wedge$, после чего медиатор возвращает тесту спецификационное наблюдение n, которое, вообще говоря, зависит от $\sigma^\wedge$, t и $\lambda^\wedge$. Это означает, что медиатор формально можно задать в виде двух отображений $\boldsymbol{M}_1{:}\mathrm{X}^\wedge\times\boldsymbol{T}\to 2^{\mathrm{X}^\wedge}$ и $\boldsymbol{M}_2{:}\mathrm{X}^\wedge\times\boldsymbol{T}\times\mathrm{X}^\wedge\to\boldsymbol{N}$, где X и $\mathrm{X}^\wedge$ – множества историй в $(\boldsymbol{T},\boldsymbol{N})$- и $(\boldsymbol{T}^\wedge,\boldsymbol{N}^\wedge)$-семантиках: $\mu^\wedge=\sigma^\wedge\cdot\lambda^\wedge\in\boldsymbol{M}_1(\sigma^\wedge,t)$ и $n=\boldsymbol{M}_2(\sigma^\wedge,t,\mu^\wedge)$. Предполагается, что медиатор, как и тест в $(\boldsymbol{T}^\wedge,\boldsymbol{N}^\wedge)$-семантике, готов получать от реализации любые наблюдения из $\boldsymbol{N}^\wedge$. Это означает, что замыкание по взятию префикса множества историй $\mu^\wedge$, генерируемых отображением $\boldsymbol{M}_1(\sigma^\wedge,t)$, ветвится в каждой точке после $\sigma^\wedge$ по всем наблюдениям из $\boldsymbol{N}^\wedge$.

Медиаторное преобразование может зависеть от состояния LTS-реализации. Если такой зависимости нет, достаточно иметь одно m-состояние 1-го типа. Если есть, таких m-состояний несколько, и одному m-состоянию 1-го типа соответствует множество эквивалентных состояний реализации. В этом случае должна быть возможность опросить реализационное состояние. Тестирование с такой возможностью называется тестированием с открытым состоянием [2,4]. Предполагается, что опрос состояния достоверен: в каждом состоянии возвращается именно это состояние. Это можно понимать как добавление воздействия $q^\wedge$ опроса состояния к $\boldsymbol{T}^\wedge$. Множество $\boldsymbol{N}^\wedge$ расширяется состояниями реализации. В каждом состоянии i реализации добавляется переход-петля по опросу состояния $i\text{—}(q^\wedge,i)\to i$. Тогда зависимость медиатора от состояния реализации сводится к адаптивности, то есть зависимости от наблюдений.

И.Б.Бурдонов, А.С.Косачев.
Тестирование с преобразованием семантик.
Научный сервис в сети Интернет: суперкомпьютерные центры и задачи: Труды Всероссийской суперкомпьютерной конференции (21-24 сентября 2010 г., г. Новороссийск). - М.: Изд-во МГУ, 2010. с. 420-424.
5 стр.

3.3. Гипотеза о безопасности и конформность

Медиаторное отображение $\boldsymbol{M}_1$ генерирует отображение историй $\boldsymbol{M}_3$:X→$2^{X^\wedge}$, дающее то множество реализационных историй, которые могут быть получены при прохождении данной спецификационной истории. Оно определяется следующими правилами:

1) $\boldsymbol{M}_3(\epsilon)=\{\epsilon\}$,

2) $\forall\sigma\in X\ \forall t\in\boldsymbol{T}\ \forall n\in\boldsymbol{N}\ \boldsymbol{M}_3(\sigma\cdot t\cdot n)=\{\mu^\wedge|\exists\sigma^\wedge\in\boldsymbol{M}_3(\sigma)\ \mu^\wedge\in\boldsymbol{M}_1(\sigma^\wedge,t)\}$.

На основе отображения историй переопределим гипотезу о безопасности и конформность для тестирования через медиатор:

$\Sigma^\wedge$ ***safe for*** $\Sigma =_{\text{def}} \forall\sigma\in\Sigma\ \forall\sigma^\wedge\in\boldsymbol{M}_3(\sigma)\cap\Sigma^\wedge\ \forall t\in\boldsymbol{T}\ (f_\Sigma(\sigma,t)\neq\varnothing \Rightarrow \boldsymbol{M}_1(\sigma^\wedge,t)\cap\Sigma^\wedge\neq\varnothing)$.

$\Sigma^\wedge$ ***saco*** $\Sigma =_{\text{def}}$ $\Sigma^\wedge$ ***safe for*** Σ

& $\forall\sigma\in\Sigma\ \forall\sigma^\wedge\in\boldsymbol{M}_3(\sigma)\cap\Sigma^\wedge\ \forall t\in\boldsymbol{T}\ \forall\mu^\wedge\in\boldsymbol{M}_1(\sigma^\wedge,t)\cap\Sigma^\wedge\ \boldsymbol{M}_2(\sigma^\wedge,t,\mu^\wedge)\in f_\Sigma(\sigma,t)$.

3.4. Преобразование состояний

Если используется LTS-модель и в реализации определена операция опроса состояния, то, кроме воздействий и наблюдений, медиатор может преобразовывать также и состояния: из реализационного в спецификационное. В этом случае предполагается, что на множестве состояний реализации задано отношение эквивалентности, и состояние спецификации соответствует классу эквивалентных состояний реализации. Проверка этого соответствия выполняется во время тестирования наряду с проверкой наблюдений.

Медиатор преобразует состояние в ответ на дополнительное спецификационное воздействие q «опрос спецификационного состояния», добавляемое к множеству $\boldsymbol{T}$. Множество $\boldsymbol{N}$ расширяется состояниями спецификации. В состоянии s спецификации добавляется переход-петля по опросу состояния s—(q,s)→s. Медиаторное отображение, гипотеза о безопасности и конформность для измененных семантики и спецификации определяются как обычно.

Литература

1. Бурдонов И.Б. Теория конформности для функционального тестирования программных систем на основе формальных моделей. Диссертация на соискание учёной степени д.ф.-м.н., Москва, 2008.
http://www.ispras.ru/~RedVerst/RedVerst/Publications/TR-01-2007.pdf

И.Б.Бурдонов, А.С.Косачев.
Тестирование с преобразованием семантик.
Научный сервис в сети Интернет: суперкомпьютерные центры и задачи: Труды Всероссийской суперкомпьютерной конференции (21-24 сентября 2010 г., г. Новороссийск). - М.: Изд-во МГУ, 2010. с. 420-424.
5 стр.

---

2. Бурдонов И.Б., Косачев А.С. Полное тестирование с открытым состоянием ограниченно недетерминированных систем. "Программирование", -2009, №6–стр 3-18.
3. Г.В.Ключников, А.С.Косачев, Н.В.Пакулин, А.К.Петренко, В.З.Шнитман. Применение формальных методов для тестирования реализации IPv6. Труды Института системного программирования РАН, N 4, 2003. стр.121-140.
4. Lee D., Yannakakis M. Principles and Methods of Testing Finite State Machines – A Survey. Proceedings of the IEEE 84, No. 8, 1090–1123, 1996.