УДК 004.052.42

Б.М. КОНОРЕВ, В.В. СЕРГИЕНКО, Г.Н. ЧЕРТКОВ, Ю.Г. АЛЕКСЕЕВ

Сертификационный центр АСУ, Харьков, Украина

ДОКАЗАТЕЛЬНАЯ НЕЗАВИСИМАЯ ВЕРИФИКАЦИЯ И ОЦЕНКА СКРЫТЫХ ДЕФЕКТОВ КРИТИЧЕСКОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ НА ОСНОВЕ ДИВЕРСИФИЦИРОВАННОГО ИЗМЕРЕНИЯ ИНВАРИАНТОВ

Статья описывает дальнейшее развитие работ в рамках общего направления разработки целевой технологии независимой верификации и прогнозирования скрытых дефектов программного обеспечения, важного для безопасности. Особое внимание уделено чрезвычайно важному вопросу калибровки чувствительности методов тестирования программного обеспечения (ПО) и полноты охвата исходного кода проверками. Предлагается концепция по нахождению адекватного профиля дефектов для конкретного проекта ПО при процедуре калибровки. Обосновывается уровень структуры ПО, на котором производится внесение тестовых дефектов для калибровки. Предлагаются принцип калибровки, модели и элементы практической теории, обеспечивающие практическое решение поставленных задач.

Ключевые слова: *программное обеспечение, тестирование, верификация, инвариант, калибровка, профиль дефектов*

Введение

Безопасность АЭС гарантируется путем реализации концепции глубокой защиты, основанной на применении физических барьеров на пути распространения радиоактивных веществ в окружающую среду и системы технических и организационных мер по обеспечению этих барьеров.

Ключевая роль в обеспечении безопасности АЭС принадлежит информационно-управляющим системам (ИУС), выполняющим функции предотвращения, защиты и ликвидации последствий аварийных ситуаций. Аномальное функционирование ИУС вследствие недостатков их качества (дефектов) представляет фактор риска возникновения аварийных ситуаций с тяжестью последствий по критериям «материальные потери – ущерб окружающей среде – угроза здоровью и жизни людей».

Современная тенденция в сфере ИУС состоит в росте объемов и удельной части программно-реализуемых критических функций ИУС АЭС, связанных с обеспечением безопасности. Дефекты критического ПО ИУС могут быть причиной аномального функционирования (отказа) таких систем.

По результатам исследований каждый пятый отказ оборудования АЭС связан с неисправностями оборудования ИУС, также как и каждая пятая авария в ракетно-космической технике обусловлена неисправностями компьютерной системы управления. 6 из 7 отказов компьютерных систем управления, приведших к авариям ракетно-космических комплексов, вызваны дефектами программных средств. В связи с этим критическое ПО ИУС представляет важный объект нормативного регулирования.

Такая ситуация является общей для случаев реализации ИУС как на основе микропроцессоров, так и для ПЛИС-реализации (аппаратная реализация предварительно разработанной программной реализации алгоритмов ИУС – так называемая «жесткая логика»).

Обязательная квалификация и сертификация критического ПО ИУС является заключительной процедурой гарантирования социально допустимых уровней проектных рисков возникновения аварийных состояний из-за скрытых дефектов ПО.

Наиболее значимым для квалификации и сертификации критического ПО ИУС являются результаты выполнения фундаментальных методик: независимой верификации и валидации (НВиВ) и прогнозирования вероятности скрытых дефектов.

Независимость подразумевает эффективную реализацию принципов технологического и административного разнообразия, как средства повышения достоверности (снижения степени неопределенности) результатов оценки качества критического ПО.

Достижение реальной (а не декларируемой) независимости означает предоставление объективных доказательств разнообразия и эффективности используемых методов верификации критического ПО.

Независимая верификация должна быть дока-

зательной. Это означает использование метрик количественной оценки степени разнообразия используемых методов, которые устанавливают степень неопределенности и достоверности итоговой оценки вероятности скрытых дефектов ПО.

Именно доказательная НВиВ критического ПО и достоверный прогноз скрытых дефектов на ее основе определяют реальные возможности учета и регулирования рисков, связанных с недостаточным качеством критического ПО ИУС, при нормативном регулировании безопасности АЭС.

Статья описывает продвижение (дальнейшее развитие) работ в рамках общего направления разработки целевой технологии независимой верификации и прогнозирования скрытых дефектов программного обеспечения, важного для безопасности [1-3]. Особое внимание уделено чрезвычайно важному вопросу калибровки чувствительности методов тестирования ПО и полноты охвата исходного кода проверками. Предлагается концепция по нахождению адекватного профиля дефектов для конкретного проекта ПО при процедуре калибровки. Устанавливается уровень внесения дефектов для калибровки, принцип калибровки, предлагаются модели и элементы практической теории, обеспечивающие практическое решение поставленных задач.

1. Цель работы

Работа направлена на создание прогрессивной информационной технологии доказательной независимой верификации и прогнозирования вероятности скрытых дефектов критического ПО ИУС, связанных с безопасностью АЭС.

Целью является разработка методологии и инструментальной среды (ИС), обеспечивающих доказательность и достоверность результатов и управление рентабельностью квалификации критического ПО.

Главными задачами являются:

– прогноз вероятности скрытых дефектов;

– оценка полноты тестового покрытия;

– управление рентабельностью (в смысле, достижения требуемого уровня (величины) степени неопределенности оценок при минимальных затратах ресурсов.

Решаемые задачи являются существенной частью risk-informed подходов к регулированию безопасности АЭС.

ИС при квалификационных испытаниях критического ПО обеспечивает поддержку сценария доказательной независимой верификации (НВ) и прогнозирования скрытых дефектов. С помощью ИС реализуется высокий уровень компьютеризации, автоматизации и достоверности результатов квалификации критического ПО. ИС включает комплекс утилит поддержки методик (процедур) сценария на аналитическом, информационном и организационном уровнях.

Общим контекстом реализации методологии доказательной НВ и прогнозирования скрытых дефектов являются квалификационные испытания ИУС, включающие реализацию фундаментальных методик FMECA, FTA, PHA, HSIA и др., определенных современной нормативной базой (стандартами).

Результаты работ могут быть использованы соответствующими организациями при разработке, сертификации, квалификации и нормативном регулировании (разрешительной деятельности) критического ПО ИУС, связанных с безопасностью АЭС

2. Концепция доказательной независимой верификации

Концепция заключается в реализации следующих основных положений:

– использование методологии статического анализа исходного ПО как платформы доказательной независимой верификации и прогнозирования скрытых дефектов;

– использование model-checking подхода для диверсифицированного инвариантно-ориентированного измерения и оценки качества критического ПО [4];

– формальная верификация критического ПО с использованием критерия сохранности (неизменности) инвариантов для всей области возможных вариантов использования, определенной ТЗ;

– экспериментальная калибровка методом посева тестовых дефектов чувствительности и степени разнообразия инвариантно-ориентированных моделей исходного ПО для количественной оценки вероятности скрытых дефектов;

– использование для калибровки метода «капельной» инъекции тестовых дефектов для исключения эффекта «интерференция-мутация» на адресном поле исходного ПО.

Теоретический базис предлагаемой технологии представлен рядом математических моделей, включая:

– функциональную модель сценария;

– модель диверсифицированного измерения инвариантов ПО;

– модель оценки остаточных дефектов ПО для композиции диверсных методов измерения инвариантов;

– модель экспериментальной калибровки чувствительности и степени разнообразия методов из-

мерения инвариантов на основе (с использованием) «капельного» посева тестовых дефектов.

Функциональная модель сценария целевой технологии независимой верификации (более детально см. [1]) включает три базовых методики:

– нормализации проекта ПО;

– измерения инвариантов при статическом анализе исходных кодов ПО;

– калибровки методов измерения инвариантов. Прогнозирование скрытых дефектов. Интегральная оценка характеристик ПО. Рентабельность.

Методика «Нормализация проекта ПО» включает:

– формирование и верификацию нормативного профиля (опорной модели) для оцениваемого проекта;

– раскрытие спецификаций измеряемых атрибутов проекта ПО (инвариантов) и формирование схемы измерений;

– верификацию инвариантно-ориентированной модели оценивания, прямую и инверсную трассировку элементов проекта ПО «Спецификации требований к ПО (ТЗ)», «Спецификации инвариантов», «Нормативного профиля проекта», «Спецификации проектных определений и обоснований (ТД проекта)»;

– оценку полноты тестового покрытия проекта ПО.

Методика «Измерение инвариантов при статическом анализе исходных кодов ПО» включает:

– инструментирование исходного ПО на базе model-checking подхода. Формирование оценочной модели пригодной для измерения инвариантов;

– измерение семантических, интервально – точностных и логических инвариантов в режиме интерпретации инструментированной версии ПО;

– обработку результатов измерения инвариантов и диагностирование дефектов ПО.

Методика «Калибровка методов измерения инвариантов. Прогнозирование скрытых дефектов. Интегральная оценка характеристик ПО. Рентабельность» включает:

– калибровку чувствительности и степени разнообразия методов измерения инвариантов ПО;

– прогнозирование скрытых дефектов;

– оценку полноты тестового покрытия;

– интегральную оценку характеристик качества проекта ПО.

3. Модель диверсифицированного измерения инвариантов ПО

Уточненная модель диверсифицированного измерения инвариантов ПО [2] иллюстрирует интегральную проверяющую способность композиции диверсных технологий верификации, основанных на измерении числовых, семантических, логических и др. инвариантов ПО (рис. 1).

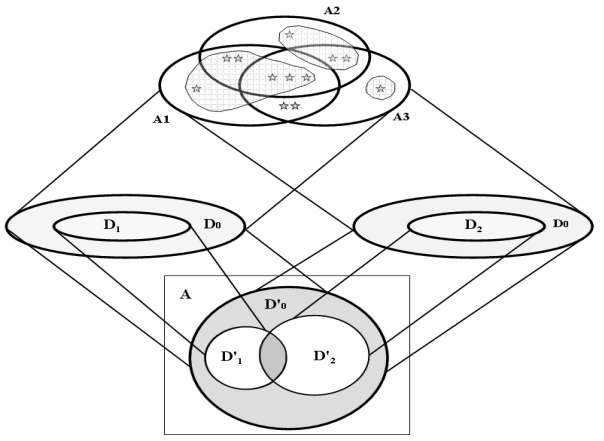

Рис. 1 Модель диверсифицированного измерения инвариантов ПО

Ai – представленные на экспертизу материалы;

D0 – множество дефектов, возможных в ПО;

Di – множество дефектов, обнаруживаемых методом измерения i-го типа инварианта;

A – единое поле для оценки разнообразия использованных методов;

D'i – проекция Di на поле A.

Модель представляет спецификацию множеств дефектов ПО, обнаруживаемых различными методами измерения инвариантов и позволяет оценить чувствительность диверсных методов измерения инвариантов (ДМИИ) к дефектам.

UA_i – представленные на экспертизу материалы (техническая и нормативная документация, программное обеспечение в виде исходных и исполнимых кодов) – область определений, опорная модель для анализа и оценки характеристик ПО

Для проверки одного отдельного инварианта из представленного на экспертизу материала выбирается (элиминируется) только информация, относящаяся к данному инварианту.

Модель позволяет определить:

$D'_1 \cup D'_2$ – суперпозицию множеств дефектов, обнаруживаемых 1-м или 2-м методом в адресном поле ПО, – определяет реальную степень разнообразия методов измерения инвариантов ПО;

$D'_0 \setminus D'_1 \cup D'_2$ – множество дефектов, необнаруживаемых 1-м и 2-м методами, что обуславливается нечувствительностью обоих методов.

4. Модель остаточных и скрытых дефектов ПО для композиции ДМИИ

Получила дальнейшее развитие модель остаточных и скрытых дефектов ПО для композиции диверсных

методов измерения инвариантов ДМИИ [2]. Модель (см. рис.2) иллюстрирует возможные взаимные расположения множеств остаточных и скрытых (латентных) дефектов и позволяет оценить выигрыш от использования ДМИИ для различных вариантов расположения. Остаточные дефекты для ДМИИ – дефекты, не обнаруживаемые данным ДМИИ. Скрытый (латентный) дефект – дефект в элементе, который может привести к одному или более отказам непосредственно элемента или другого связанного оборудования, не выявленный при разработке и тестировании ПО.

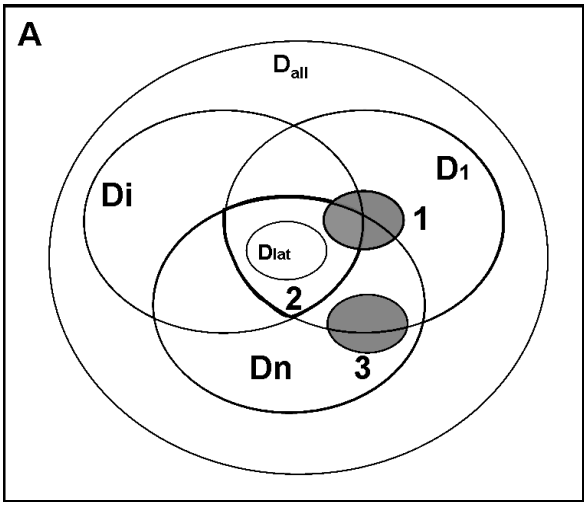

Рис. 2 Модель остаточных и скрытых дефектов ПО

Dall – множество возможных дефектов в пространстве;

A; Di – множество дефектов, обнаруживаемых методом измерения i-го типа инварианта;

Dlat – множество скрытых дефектов.

Индикатором оценки достигаемого эффекта от использования методов измерения инвариантов является величина (в %), на которую уменьшается вероятность скрытых (латентных) дефектов P (D_{lat}) в процессе последовательной реализации композиции диверсных методов измерения инвариантов при независимой верификации:

$$I = \frac{\left| D_{lat} \setminus \bigcap_{i=1}^{n} D_i \right|}{|D_{lat}|} \bullet P(D_{lat})$$

где n – общее количество использованных ДММИ.

Возможны варианты:

а) теоретически возможный случай уменьшения вероятности скрытых дефектов на 100%:

$$\left(\bigcap_{i=1}^{n} D_i \right) \cap D_{lat} = \varnothing \quad I=1 \quad \text{(позиция 3)}$$

б) максимально неблагоприятный вариант:

$$D_{lat} \subset \bigcap_{i=1}^{n} D_i \quad I=0 \quad \text{(позиция 2)}$$

в) общий случай:

$$\left(\bigcap_{i=1}^{n} D_i \right) \cap D_{lat} \neq \varnothing, \quad I = \overline{0,1} \quad \text{(позиция 1)}$$

Если при независимой верификации не будут обнаружены скрытые дефекты (элементы множества D_{lat}), то, хотя выигрыш I = 0, подмножество $\cap D_i$ может использоваться как граничная область нахождения скрытых дефектов (является рамочной оценкой вероятности скрытых дефектов). Устанавливается область, где дефекты гарантированно отсутствуют – $\overline{\bigcap_{i=1}^{n} D_i}$.

Для улучшения или уточнения оценки усилия необходимо сосредоточить на анализе области $\cap D_i$ и поисках дефектов внутри этой области.

$\dfrac{\left| \bigcap_{i=1}^{n} D_i \right|}{\left| \bigcup_{i=1}^{n} D_i \right|}$ – величина отношения, – определяет относительный выигрыш от использования ДММИ.

5. Модель калибровки чувствительности и степени разнообразия диверсных методов на основе капельной инъекции тестовых дефектов

Для оценки полноты покрытия исходного кода проверками, взаимной чувствительности методов измерения инвариантов используется калибровка – посев тестовых дефектов (см. рис. 3).

Также с помощью калибровки определяется оптимальный по ресурсоемкости набор методов для испытаний конкретного проекта ПО.

Суть калибровки состоит в «капельной инъекции» одиночного тестового дефекта определенного типа и циклического выполнения процедуры «инъекция – обнаружение дефекта» статистически значимое число раз.

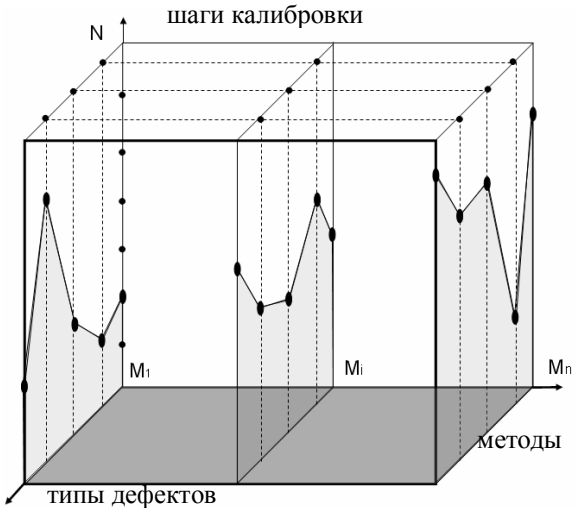

Рис. 3. Модель калибровки

Результатом калибровки является определение чувствительности j–го метода к i-му типу дефектов из подмножества дефектов – парциальная чувствительность $S_{парц\ ij}$ (множество остаточных дефектов после использования метода).

На каждом шаге N_i для каждого метода M_j выполняется засев или «инъекция» дефектов определенных типов. Возможные типы дефектов определяются для конкретного проекта ПО на основании анализа использованных языковых конструкций и их возможных искажений.

Последовательность действий при калибровке на каждом шаге (слое):

1. последовательно выбирается тип дефекта;
2. вносится в проверяемое ПО дефект выбранного типа. Место внесения выбирается случайным образом;
3. ПО проверяется методами, чувствительными к данному типу дефектов;
4. определяется для каждого метода обнаружение/необнаружение искажения ПО;
5. ПО возвращается к исходному начальному состоянию (до внесения дефекта).

Процедура выполняется циклически для всех типов дефектов, возможных для конкретного проекта.

Для получения результатов, позволяющих определить реальную чувствительность методов набирается статистика – количество шагов калибровки должно быть статистически значимым.

По результатам калибровки определяется:

1) интегральная чувствительность каждого метода:

$$S = \bigcup_{i=1}^{k} S_{парц i}$$

где k – общее количество типов дефектов,

$S_{парц\ i}$ – парциальная чувствительность метода к i-му типу дефекта (чувствительность к конкретному типу дефекта)

2) попарно для всех Mi (каждого с каждым) степень разнообразия:

$$m_{ij} = 1 - \frac{S_i \cap S_j}{S_i \cup S_j}$$

где i и j принимают значения от 1 до n,

n – количество использованных методов.

и формируется матрица разнообразия:

	M_1	M_2	…	M_n
M_1	**0**		m_{ij}	
M_2		**0**		
…	m_{ji}		**0**	
M_n				**0**

На основании анализа матрицы разнообразия устанавливается оптимальный набор методов для достижения заданного покрытия и чувствительности.

6. Концепция решения проблемы выбора профиля дефектов и проведения калибровки

Для калибровки использованных методов автоматизированного тестирования ПО необходимо внести дефекты в код в соответствии с выбранным профилем дефектов. При калибровке существует две главных проблемы:

– выбор уровня в структуре (архитектуре) ПО, на котором производится внесение тестовых дефектов;

– учет специфики конкретного проекта (платформы, языка программирования, спектра операций).

Проблемы при калибровке на исполнимом коде:

1. Способ внесения – изменение исполнимого файла приводит просто к невозможности его запуска (перестает выполняться).
2. Отсутствует однозначная связь с дефектами, которые реально могут присутствовать в коде как результат работы программиста.
3. Внесение дефектов на данном уровне для предложенной независимой верификации не позволяет произвести калибровку, поскольку используемые методы не охватывают исполнимый код.

При инъекции дефектов для калибровки на исходном коде трудно установить адекватный профиль дефектов из-за большого количества используемых языков программирования и сложно клас-

сифицируемого разнообразия типов возможных дефектов.

Для решения задачи выбора адекватного профиля дефектов и уровня структуры ПО для внесения дефектов при калибровке предлагается использовать промежуточный код. Под промежуточным кодом имеется в виду представление ПО в табличном виде после этапа разбора исходного кода при компиляции.

Весь процесс компиляции состоит из 2 этапов:

1. Этап анализа – определение структуры и значений исходного кода.

2. Этап синтеза – построение исполнимого кода.

Для получения промежуточного кода необходимо провести только 1-й этап. Таким образом, есть прямая связь между моделями для контроля инвариантов, которые строятся на этапе анализа и получаемом после этапа синтеза исполнимым кодом, бездефектность которого оценивается. В силу того, что для критических систем используются проверенные (аттестованные) компиляторы, вероятностью внесения ошибок на этапе синтеза можно пренебречь. И можно утверждать, что, оценивая бездефектность промежуточного кода, мы оцениваем бездефектность исполнимого кода.

Подобно компилятору с помощью специальной утилиты (парсера) производится семантический и синтаксический разбор представленного на испытания кода, и полученные результаты заносятся в специальные таблицы базы данных. Профиль дефектов строится на базе полученных таблиц. Таким образом, учитывается специфика конкретного проекта и упрощается процедура внесения искусственных дефектов при калибровке (достаточно внести изменения в соответствующие таблицы).

Предлагаемый подход является универсальным для целого класса языков программирования, за счет получения промежуточного кода в результате разбора. Для использования нового языка необходимо описать правила разбора. При этом не требуется доработок как методов измерений, так и механизма калибровки.

7. Практическая реализация

Практическая часть работ представлена результатами разработки утилиты статического анализа (СА).

Утилита СА предназначен для анализа и оценки качества программного обеспечения с использованием свойств программных инвариантов: размерности, интервала допустимых значений, точности программных переменных, логики выполнения и др.

Инвариант – неизменное свойство ПО на всех этапах жизненного цикла. Контроль производится путем измерения инвариантов ПО и сравнения их с заданными в ТД.

В данный момент Утилита СА предназначена для анализа программного обеспечения, представленного в виде текста, размещенного в одном или нескольких файлах, написанных на языке C/C++ в среде разработки Borland C++ версий 3-5. Техническая реализуемость утилит и готовность к полномасштабной разработке подтверждена интеграционными лабораторными тестами с использованием реальных объектов экспертизы – ПО ИУС АЭС критического применения. Контролируемыми утилитой инвариантами являются:

1. семантический инвариант – соблюдение постоянства физических размерностей переменных;

2. интервально-точностной инвариант – допустимый диапазон и точность представления переменных;

3. корректность использования оперативной памяти:

a) выход за пределы массива при чтении/записи;

b) освобождение некорректного (нераспределенный, освобожденный и т.д) участка памяти;

c) попытка доступа к некорректному участку памяти;

d) утечка ресурсов (некорректное освобождение ресурсов);

e) нецелевое использование памяти.

Принцип построения утилит позволяет нарастить проверки (реализация планируется), в части:

1. Соответствие требованиям стандартов и правилам кодирования (использование рекурсии, степень вложенности процедур, несоблюдение соглашения об именах, метрики сложности, глубина вложенности циклов, метрики Холстеда, объектно-ориентированные метрики для языка программирования C++)

2. Логика выполнения программного кода:

a) наличие неиспользуемых участков кода,

b) контроль прохождения условий перехода

c) формализованные проверки соответствия спецификации:

– установление соответствия базе переменных

– логика включения алгоритмов, в т.ч. реализуемость условий включения

Входными данными для СА являются список файлов, содержащих анализируемый код ПО и характеристики измеряемых инвариантов ПО.

Характеристики измеряемых инвариантов ПО задаются пользователем утилиты СА на основании технической документации, входящей в состав ПО.

Основой оценки ПО являются методы, использующие семантическую, интервальную и точностную алгебры измерения атрибутов базовых операций, входящих в состав модулей, и вызовов функций и методов классов.

Результаты работы утилит СА выдаются в форме отчета, содержащего сводную оценку качества ПО и ее достоверность. Работа с объектами экспертизы выполняется в технологии тонкого клиента. Вся логика обработки располагается на Web сервере.

Заключение

Целевая технология независимой верификации и прогнозирования вероятности скрытых дефектов критического ПО представляет одну из ключевых методик анализа критичности и оценок гарантоспособности и функциональной безопасности при квалификационных испытаниях ИУС критического применения.

Предложенный подход (концепция) по обоснованию профиля дефектов для инъекции тестовых дефектов при калибровке чувствительности и степени разнообразия моделей для контроля инвариантов позволяет сделать результаты независимой верификации с использованием ДМИИ обоснованными.

Эффективность и рентабельность такой технологии в значительной мере определяют реальные возможности достижения необходимых уровней гарантоспособности и функциональной безопасности разрабатываемой ИУС.

В целом предложенный подход (концепция, методология, модели и методы) к построению целевой технологии независимой верификации и прогнозирования скрытых дефектов критического ПО обеспечивает следующие преимущества:

– расширяет реальные возможности организаций – разработчиков и регулирующих органов в части повышения достоверности и точности прогнозирования рисков аномального функционирования ИУС из-за дефектов критического ПО и может использоваться в долгосрочных программах восстановления, модернизации и продления сроков эксплуатации ИУС важных для безопасности;

– обеспечивает возможность количественно оценивать предельные значения и управлять снижением вероятности скрытых дефектов критического ПО в диапазоне 0 – 100% (в пределе на 100% в зависимости от характеристик проекта ПО и приемлемых уровней рентабельности);

– соответствует передовому научно-техническому уровню в сфере квалификации критического ПО;

– представляет методологическую основу для решения актуальной задачи разработки нормативно-методического и инструментального обеспечения оценок гарантоспособности и функциональной безопасности критического ПО ИУС важных для безопасности в атомной энергетике.

Литература

1. Конорев Б.М. Целевая технология рентабельной оценки надежности и функциональной безопасности критического программного обеспечения / Б.М. Конорев, Ю.Г. Алексеев, В.В. Сергиенко, В.С. Харченко, Г.Н. Чертков // Радіоелектронні і комп'ютерні системи. Науково-технічний журнал Харків, "ХАІ", 2007. – №6 (25). – С. 162-170.

2. Конорев Б.М. Квалификационные испытания критического программного обеспечения космических систем: целевая технология независимой верификации и прогнозирования скрытых дефектов / Б.М. Конорев, Ю.Г. Алексеев, С.А. Засуха, Л.П. Семенов, В.С. Харченко, Г.Н. Чертков // Космічна наука і технологія. Науково-практичний журнал Том 14, №4, НКАУ, НАНУ, Київ, 2008. – С. 9- 26.

3. Конорев Б.М. Калибровка методов измерения инвариантов критического программного обеспечения: профиль инъектируемых тестовых дефектов / Б.М. Конорев, В.В. Сергиенко, Л. Новы, Г.Н. Чертков // Радіоелектронні і комп'ютерні системи. Науково-технічний журнал, Харків, "ХАІ", 2008. – №6 (25). – С. 161-167.

4. Кларк Э.М. Верификация моделей программ: Model Checking / Э.М. Кларк, О. Грамберг, Д. Пелед Пер. с англ. / Под ред. Р. Смелянского. — М.: МЦНМО, 2002. – 416 с.

Поступила в редакцию 7.02.2009

Рецензент: д-р техн. наук, доц., проф. О.В. Поморова, Хмельницкий национальный университет, Хмельницкий, Украина.

.

ДОКАЗОВА НЕЗАЛЕЖНА ВЕРИФІКАЦІЯ Й ОЦІНКА ПРИХОВАНИХ ДЕФЕКТІВ КРИТИЧНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ НА ОСНОВІ ДИВЕРСИФІКОВАНОГО ВИМІРЮВАННЯ ІНВАРІАНТІВ

Б.М. Конорев, В.В. Сергієнко, Г.М. Чертков, Ю.Г. Алексеєв

Стаття описує подальший розвиток робіт у рамках загального напрямку розробки цільової технології незалежної верифікації й прогнозування прихованих дефектів програмного забезпечення, важливого для безпеки. Особлива увага приділена надзвичайно важливому питанню калібрування чутливості методів тестування ПО й повноти охвата вихідного коду перевірками. Пропонується концепція по встановленню адекватного профілю дефектів для конкретного проекту ПО при процедурі калібрування. Обґрунтовується рівень структури ПО на якому виконується внесення тестових дефектів для калібрування. Пропонуються принцип калібрування, моделі й елементи практичної теорії, що забезпечують практичне рішення поставлених завдань.

Ключові слова: програмне забезпечення, тестування, верифікація, інваріант, калібрування, профіль дефектів.

PROVEN INDEPENDENT VERIFICATION AND LATENT FAULT FORECASTING OF THE CRITICAL SOFTWARE ON THE BASE OF DIVERSE INVARIANT MEASUREMENT

B.M Konorev, V.V. Sergiyenko, G.N. Chertkov, U.G. Alexeev

The article describes the further development of target technology of independent verification and latent defect forecasting of the software, important for system safety. Special attention is given to the important issue of sensitivity calibration for the methods of software testing and completeness of test coverage of source software code. The concept of estimation at calibration procedure of adequate defect profile for the specific software project is offered . The level of the software structure for the test defects injunction during calibration is substantiated. The principle of calibration, models and the elements of the practical theory providing the practical decision of the tasks are offered.

Keywords: software, testing, verification, invariant, calibration, profile of defects.

Конорев Борис Михайлович - докт. техн. наук, профессор кафедры программного обеспечения компьютерных систем Национального аэрокосмического университета им. Н.Е. Жуковского «ХАИ», Харьков, Украина,. e-mail: admin@scasu.com.

Сергиенко Владимир Владимирович - руководитель Испытательной лаборатории информационно-вычислительных систем управления Сертификационного центра АСУ ГП Госцентркачества, Харьков, Украина. e-mail:admin@scasu.com.

Чертков Георгий Николаевич - заслуженный машиностроитель Украины, директор Сертификационного центра АСУ ГП Госцентркачества, Харьков, Украина. e-mail: scasu@scasu.com.

Алексеев Юрий Гаврилович - начальник отдела экспертизы ПО и ПТК Сертификационного центра АСУ Госцентркачества, г. Харьков, Украина. e-mail: ugalex@scasu.com.