

УДК 004.43

## **Effectiveness Evaluation of Static Analysis for Searching Natural Semantic Defects of Program Objects**

**Dmitry S. Victorov and Roman E. Zhidkov\***

*Military Academy of Aero-Space Defense  
named after the Marshal of Soviet Union G.K. Zhukov  
50 Zhigareva Str., Tver, 170022, Russia*

Received 24.05.2018, received in revised form 21.08.2018, accepted 12.10.2018

---

*The paper evaluates the effectiveness of static analysis for searching a new type of functional defects called program objects natural semantic defects in terms of completeness and accuracy on the basis of the probability theory mathematical apparatus. Calculation of performance indicators is carried out by graph modeling of the defects localization variants in the software code constructions for various groups of operations (arithmetic, assignments, and comparisons). The possible state of the analyzer is modeled for each terminal situation on the basis of a priori known checking construction natural semantic result value statement (true trigger, conditionally true trigger, skip, and norm). The obtained dependencies allow to determine the feasibility of using static analysis to find considered defect's type on the basis of statistical characteristics of software code.*

*Keywords: functional defects, static analysis, verification, natural semantic, analyzer, modeling.*

---

Citation: Victorov D.S., Zhidkov R.E. Effectiveness evaluation of static analysis for searching natural semantic defects of program objects, J. Sib. Fed. Univ. Eng. technol., 2019, 12(1), 6-17. DOI: 10.17516/1999-494X-0091.

---

---

© Siberian Federal University. All rights reserved

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).

\* Corresponding author E-mail address: r-zhdkv@yandex.ru

## **Оценка эффективности статического анализа для поиска дефектов естественной семантики программных объектов**

**Д.С. Викторов, Р.Е. Жидков**

*Военная академия воздушно-космической обороны  
им. Маршала Советского Союза Г.К. Жукова  
Россия, 170022, Тверь, ул. Жигарева, 50*

---

*В статье проведена оценка эффективности статического анализа для поиска нового типа функциональных дефектов – дефектов естественной семантики программных объектов по показателям полноты и точности на основе математического аппарата теории вероятностей. Расчет показателей эффективности осуществлялся путем графового моделирования вариантов локализации дефектов в конструкциях программного кода для различных групп операций: арифметических, присваивания и сравнения. Для каждой конечной ситуации, создаваемой в модели на основе утверждения об априорно известном результирующем значении естественной семантики проверяемой конструкции, моделируется возможное состояние анализатора (истинное или условно истинное срабатывание, пропуск и норма). Полученные зависимости позволяют определить целесообразность использования статического анализа для поиска дефектов рассматриваемого типа на основе статистических характеристик программного кода.*

*Ключевые слова: функциональные дефекты, статический анализ, верификация, естественная семантика, анализатор, моделирование.*

---

### **Введение**

Одной из важнейших задач, возникающих при разработке программного обеспечения (ПО) автоматизированных систем (АС) военного назначения (ВН), выступает обеспечение требуемого качества. Наличие дефектов ПО оказывает существенное влияние на характеристики качества создаваемого продукта: функциональность, надежность, эффективность и др. Обнаружение и регистрация дефектов ПО осуществляется путем проведения верификации, которая является трудоемким процессом с низкой степенью автоматизации, требующим более 15 % бюджета на разработку ПО [1, 2]. Указанная проблема усугубляется ростом размера и сложности ПО, исследование которого проводится в условиях финансовых и временных ограничений.

Дефекты содержатся как в спецификации и документации к ПО, так и непосредственно в самих программах. Поиск дефектов спецификации и документации осуществляется путем проведения различного рода экспертиз, которые по определению проводятся экспертом «вручную» с различной степенью формализации процедуры исследования и являются высокотрудоемким методом верификации [3]. Программные дефекты (ПД) определяют в основном тестированием, которое в общем случае не может быть автоматизировано и требует разработки тестового плана, сценария и процедур тестирования, а также выполнения анализа результатов его проведения. Однако для поиска некоторой части ПД – нефункциональных дефектов (не связанных с нарушением спецификации) – на практике широкое применение получили инструменты ста-

тического анализа (СА), которые автоматизированно, а в некоторых случаях автоматически, на основе различных представлений программного кода обнаруживают дефекты и указывают их точную локализацию, обеспечивая полное покрытие исходного кода проверками. Применение программного кода в качестве входных данных анализатора дает возможность использовать СА на ранних стадиях разработки ПО, что в целом позволяет получить выигрыш в снижении затрат на его создание [1, 3]. Применение СА только для поиска нефункциональных дефектов ПО ограничивает эффективность данной методологии и не дает минимизировать суммарные затраты на разработку. Исследование возможностей СА для поиска функциональных дефектов (ФД) является актуальной научной задачей.

Для автоматизации процесса обнаружения некоторых ФД предлагается подход, основанный на СА исходного кода программ, оперирующих данными с ясным физическим смыслом (ускорение, скорость, длина, плоский угол и т.д.), и заключающийся в контроле выполнения принципа размерной однородности уравнений, нарушение которого предлагается называть дефектом естественной семантики (ДЕС). Естественная семантика – понятие, введенное для обозначения неизменных признаков программных объектов (переменных, функций и т.п.), отличных от семантики, определяемой требованиями спецификации языка программирования. В частности, в данной работе под естественной семантикой понимается размерность физической величины, интерпретируемая в объекте программы. Размерность величин, отталкиваясь от операций со степенями сомножителей формулы размерности физической величины, формализуется в виде вектора естественной семантики (ВЕС) некоторого линейного пространства с определенными операциями поэлементного сложения и умножения на скаляр [4].

### Показатели эффективности СА

Возможность использования вышеописанного подхода к обнаружению ФД должна быть исследована на предмет определения влияющих на него факторов и подкреплена оценкой его эффективности – комплексного свойства, отражающего качество получаемых результатов анализа. Показателями эффективности СА являются:

полнота ( $\Pi_{СА}$ ) – доля истинных среди существующих дефектов;

точность ( $T_{СА}$ ) – доля истинных среди обнаруженных дефектов [5].

Полнота и точность СА для поиска ДЕС, исходя из соотношения множеств существующих и обнаруженных дефектов, может быть оценена до программной реализации алгоритма и представлена в терминах теории вероятностей. Ввиду того, что исследуемый подход не дает ложных срабатываний при анализе, но обладает другим недостатком – условно истинным обнаружением ряда вариантов искажений, предлагается вероятность обнаружения представлять суммой вероятностей истинного и условно истинного срабатывания анализатора. Тогда выражения для показателей эффективности будут иметь вид

$$\Pi_{СА} = \frac{|D_{ИСТ}|}{|D_{СУЩ}|} = \frac{|D_{ИСТ}|}{|D_{ОБН}| + |D_{ПРОП}|} = \frac{P_{ИСТ}}{P_{ОБН} + P_{ПРОП}} = \frac{P_{ИСТ}}{P_{ИСТ} + P_{УСЛ} + P_{ПРОП}}, \quad (1)$$

$$T_{СА} = \frac{|D_{ИСТ}|}{|D_{ОБН}|} = \frac{P_{ИСТ}}{P_{ОБН}} = \frac{P_{ИСТ}}{P_{ИСТ} + P_{УСЛ}}, \quad (2)$$

где  $D_{\text{СУЩ}} = D_{\text{ОБН}} \cup D_{\text{ПРОП}}$  – множество существующих дефектов;  $D_{\text{ОБН}}$  – множество обнаруженных дефектов;  $D_{\text{ПРОП}}$  – множество пропущенных дефектов;  $D_{\text{ИСТ}}$  – множество истинных дефектов;  $P_{\text{ОБН}}$  – вероятность обнаружения дефектов;  $P_{\text{ИСТ}}$  – вероятность истинного обнаружения дефектов;  $P_{\text{УСЛ}}$  – вероятность условно истинного обнаружения дефектов;  $P_{\text{ПРОП}}$  – вероятность пропуска дефектов при анализе.

Расчет показателей полноты и точности СА для поиска ДЕС выполняется на графовых моделях, описывающих состояния анализатора в терминальных вершинах графа и варианты локализации дефектов в конструкциях вида:  $a \otimes b$ , где  $a$  и  $b$  – операнды, представленные программными объектами;  $\otimes$  – бинарная операция (арифметическая, присваивания, сравнения). Предполагается, что в рассматриваемой конструкции имеется только одно искажение в любой из составляющих. Моделирование производится при условии возможности искажения как операндов ( $a \rightarrow def, b \rightarrow def$ ), имеющих ненулевые ВЕС ( $ns(a) \neq 0, ns(b) \neq 0$ ), так и операций ( $\otimes \rightarrow def$ ) с одинаковой вероятностью  $P_{\text{ДЕФ}}$ , и допускает следующие состояния анализатора:

истинное срабатывание – указание на наличие ДЕС (на рис. 1, 3, 5 вершины серого цвета с непрерывным контуром);

условно истинное срабатывание – указание на наличие ДЕС при определенном сочетании составляющих конструкции программного кода (на рис. 1, 3, 5 вершины с прерывистым контуром);

пропуск – отсутствие сигнала при наличии ДЕС (на рис. 1, 3, 5 вершины белого цвета с непрерывным жирным контуром);

норма – отсутствие сигнала при отсутствии ДЕС (на рис. 1, 3, 5 вершины белого цвета с непрерывным контуром).

Логика определения состояния анализатора в терминальных вершинах графа модели ДЕС рассматриваемой конструкции  $a \otimes b$  строится на утверждении об априорной известности результирующего ВЕС  $ns(rec)$  для выражения преобразования естественной семантики:  $ns(a) \otimes ns(b)$ , где  $ns(a)$  и  $ns(b)$  – ВЕС операндов программных объектов;  $\oplus = \{+, -, =\}$  – операция с ВЕС. Состояния анализатора в терминальной вершине устанавливаются исходя из условия  $ns(rec) = ns(a) \otimes ns(b)$  и варианта локализации ДЕС в исходной конструкции  $a \otimes b$ , зависящего от возможности совпадения ВЕС истинного и искажающего программных объектов, а также от типа операции с программными объектами (арифметическая, присваивания и сравнения).

Для задания условия корректности моделей используется свойство, согласно которому сумма вероятностей несовместных событий, образующих полную группу: дефект существует, который может быть либо обнаружен ( $P_{\text{ОБН}} = P_{\text{ИСТ}} + P_{\text{УСЛ}}$ ), либо пропущен ( $P_{\text{ПРОП}}$ ), и дефект отсутствует ( $P_{\text{ОТСУТ}}$ ) – должна равняться единице:

$$P_{\text{ИСТ}} + P_{\text{УСЛ}} + P_{\text{ПРОП}} + P_{\text{ОТСУТ}} = 1. \quad (3)$$

Цель данной работы – разработка и использование для оценки эффективности СА графовых моделей ДЕС программных объектов для операций, которым возможно задать условие корректности согласно принципу размерной однородности физических уравнений.

### Модель ДЕС программных объектов в арифметических конструкциях

Граф модели ДЕС для конструкций с арифметическими операциями  $\{+, -, \%, *, /\}$  представлен на рис. 1. Пусть вершина  $a$ , соответствующая левому операнду бинарной операции, является начальной, из которой с вероятностью  $P_{ДЕФ}$  возможен переход в вершину  $a \rightarrow def$ , отражающую ситуацию искажения операнда  $a$ . Распознавание искажения операнда возможно в случае изменения его ВЕС  $ns(a) \neq ns(def)$ . Ситуация, когда дефект операнда существует, но его семантика осталась неизменной  $ns(a) = ns(def)$ , происходит с вероятностью  $P_{СВ}$  и не обнаруживается. Вероятность  $P_{СВ}$  определяется диапазоном возможных значений естественной семантики программных объектов, т.е. чем больше объектов одной физической размерности в исходном коде программы, тем выше вероятность их появления в качестве искажающего воздействия.

Альтернативой ситуации наличия искажения левого операнда  $a \rightarrow def$  служит событие в вершине  $a \rightarrow a$ , иллюстрирующее отсутствие дефекта в  $a$  и происходящее с вероятностью  $1 - P_{ДЕФ}$ . Событие искажения правого операнда соответствует вершине  $b \rightarrow def$ , происходит с вероятностью  $P_{ДЕФ}$  и является аналогичным искажению левого операнда.

Вершина  $\otimes$  соответствует ситуации возможного дефекта операции, которая исключает дефекты операндов и проявляется с вероятностью  $1 - P_{ДЕФ}$ . Возникновение конкретной арифметической операции происходит с вероятностью  $P_i$ , каждая из которых может быть искажена с одинаковой вероятностью  $P_j^i$ , показывающей изменение  $i$ -й операции на  $j$ -ю, где  $i, j \in \{+, -, \%, *, /\}$ . Отсутствие дефекта операции проявляется с вероятностью  $P_i^i$ .

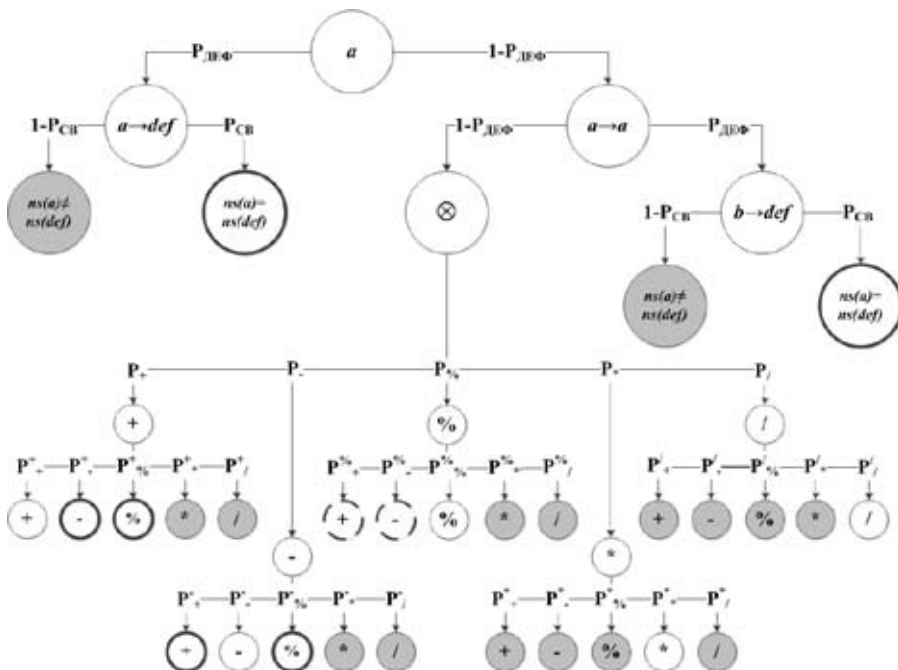


Рис. 1. Граф модели ДЕС программных объектов в арифметических конструкциях

Fig. 1. Model graph of software objects NSD for arithmetic construction

Ситуации искажения арифметических операций  $\{+, -\}$ , не изменяющих ВЕС программных объектов при преобразовании  $(P_+^+, P_+^-, P_-^+, P_-^-)$ , не распознаются, так как  $ns(res) = ns(a) = ns(b)$ . В случае замены операциями  $\{+, -\}$  исходной операции  $\{\%, \}$ , корректность преобразования естественной семантики которой контролируется по условию  $ns(res) = ns(a)$ , возможны варианты появления условно истинных срабатываний анализатора при  $ns(a) = ns(b)$ .

Вероятность существования и отсутствия дефекта арифметической операции в конструкциях программного кода при условии равной вероятности их искажения и равенства вероятностей искажения операций и операндов имеет вид:

$$P_{\text{деф}} = 4P_j^i, P_j^i = \text{const}, i \neq j, \quad (4)$$

$$1 - P_{\text{деф}} = P_i, \forall i \in \{+, -, \%, *, /\}. \quad (5)$$

Вероятности истинного и условно истинного обнаружения дефектов в арифметических конструкциях, а также вероятность пропуска дефекта на основе графа модели ДЕС и соотношения (4) равны:

$$P_{\text{ист}}^A = P_{\text{деф}} \left( (1 - P_{\text{св}})(2 - P_{\text{деф}}) + \frac{1}{2}(1 - P_{\text{деф}})^2(1 + P_* + P_/) \right), \quad (6)$$

$$P_{\text{усл}}^A = \frac{1}{2}P_{\%}P_{\text{деф}}(1 - P_{\text{деф}})^2, \quad (7)$$

$$P_{\text{проп}}^A = P_{\text{деф}} \left( P_{\text{св}} + (1 - P_{\text{деф}})P_{\text{св}} + \frac{1}{2}(1 - P_{\text{деф}})^2(P_+ + P_-) \right). \quad (8)$$

Вероятность отсутствия дефекта в арифметической конструкции, исходя из соотношений (4) и (5), представляет собой

$$P_{\text{отсут}}^A = (1 - P_{\text{деф}})^2(P_+P_+^+ + P_-P_-^- + P_{\%}P_{\%}^{\%} + P_*P_*^* + P_/P_/' ) = (1 - P_{\text{деф}})^3. \quad (9)$$

Корректность модели ДЕС программных объектов для арифметических конструкций подтверждается путем подстановки в соотношение (3) выражений (6), (7), (8) и (9).

Исходя из (1), (2), (3), (7) и (8), выражения для полноты и точности СА для поиска ДЕС в арифметических конструкциях следующие:

$$P_{\text{са}}^A = \frac{(1 - P_{\text{св}})(2 - P_{\text{деф}}) + \frac{1}{2}(1 - P_{\text{деф}})^2(1 + P_* + P_/)}{P_{\text{деф}}^2 - 3P_{\text{деф}} + 3}, \quad (10)$$

$$T_{\text{са}}^A = \frac{(1 - P_{\text{св}})(2 - P_{\text{деф}}) + \frac{1}{2}(1 - P_{\text{деф}})^2(1 + P_* + P_/)}{(1 - P_{\text{св}})(2 - P_{\text{деф}}) + \frac{1}{2}(1 - P_{\text{деф}})^2(1 + P_* + P_/ + P_{\%})}. \quad (11)$$

Зависимость полноты и точности СА для поиска ДЕС в арифметических конструкциях от вероятности наличия дефекта операнда или операции при фиксированных значениях вероятности совпадения ВЕС и вероятностей появления операций  $\{*, /\}$  и  $\{\%\}$  в коде представлена на рис. 2.

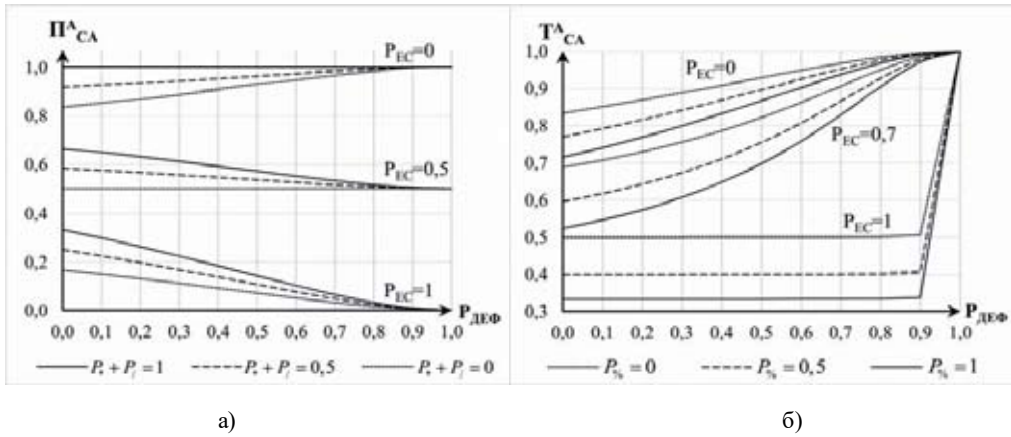


Рис. 2. Графики зависимости полноты (а) и точности (б) СА для поиска ДЕС в арифметических конструкциях от вероятности наличия дефекта операции или операнда

Fig. 2. Schedules of dependence SA completeness (a) and accuracy (б) for searching NSD in arithmetic construction from probability of operation or operand defect existence

Полнота анализа арифметических конструкций возрастает при росте доли операций  $\{*, /\}$  в программном коде, для которых обнаруживаются все возможные варианты искажений. Точность анализа увеличивается при снижении доли операций  $\{\%, \}$ , так как для них существует возможность условно истинного срабатывания анализатора. Кроме того, сужение диапазона возможных ВЕС программных объектов, т.е. увеличение вероятности их совпадения, ведет к уменьшению показателей эффективности СА.

### Модель ДЕС программных объектов в конструкциях присваивания

Для конструкций присваивания  $\{=, +=, -=, \% =, *=, /=\}$  графовая модель ДЕС программных объектов изображена на рис. 3. Данная модель характеризуется наличием операций, которые с точки зрения принципа размерной однородности и принятых допущений моделей ДЕС являются некорректными  $\{*=, /=\}$ , так как при ненулевом ВЕС второго операнда в данных конструкциях будет наблюдаться дефект:  $ns(res) = ns(a) \neq ns(a) \pm ns(b)$ ,  $ns(b) \neq 0$ . Значит, вероятность наличия дефекта для операций  $\{*=, /=\}$  следующая:

$$P_{ДЕФ}^{**} = 6P_j^i = 1, P_j^i = const, \forall i \in \{*=, /=\}. \quad (12)$$

Вероятности существования и отсутствия дефекта операций  $\{=, +=, -=, \% =\}$ , исключая заведомо неправильные конструкции с операциями  $\{*=, /=\}$ , при допущении о равенстве вероятностей дефекта операнда и операции представляет собой:

$$P_{ДЕФ} = 5P_j^i, P_j^i = const, i \neq j, \quad (13)$$

$$1 - P_{ДЕФ} = P_i^i, \forall i \in \{=, +=, -=, \% =\}. \quad (14)$$

Анализатор пропускает взаимное искажение операций  $\{=, +=, -=, \% =\}$  ввиду невозможности выявить несоответствие априорно известного результирующего ВЕС и результата

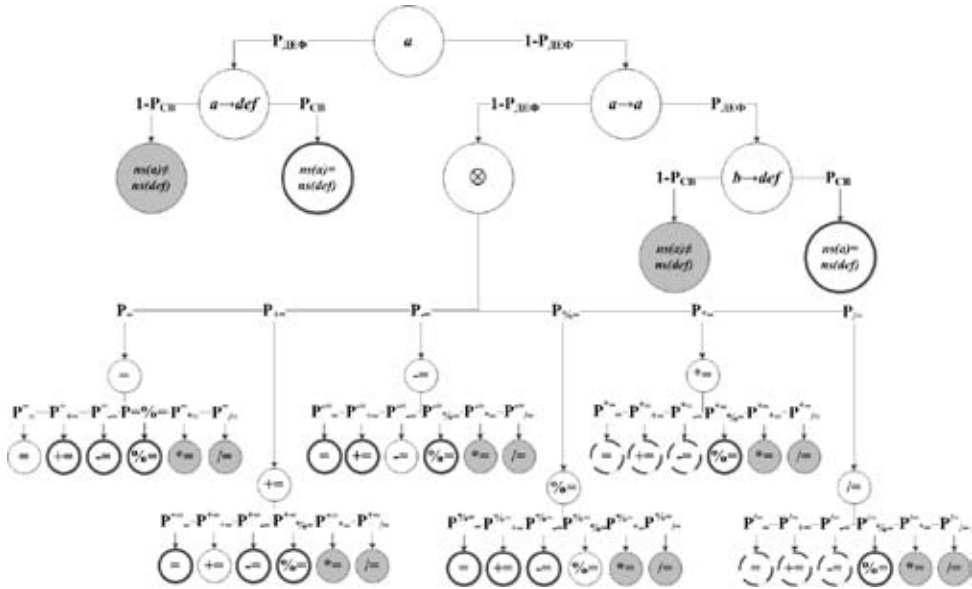


Рис. 3. Граф модели ДЕС программных объектов в конструкциях присваивания  
 Fig. 3. Model graph of software objects NSD for assignment construction

преобразования ВЕС операндов для операций, имеющих одинаковое условие корректности  $ns(res) = ns(a) = ns(b)$ .

Состояние условно истинного срабатывания анализатора достигается для операций  $\{*=, /=\}$  при замене их на  $\{=, +=, -=\}$  вследствие неоднозначности результата проверки условия корректности для искажающих операций  $ns(res) = ns(a) = ns(b)$ . Ситуация наблюдается при совпадении ВЕС операндов  $ns(a) = ns(b)$ .

Исходя из соотношения (13), состояния истинного и условно истинного срабатывания анализатора, а также пропуск дефекта для конструкций присваивания происходят со следующими вероятностями:

$$P_{\text{ИСТ}}^{\Pi} = P_{\text{ДЕФ}} (1 - P_{\text{СВ}}) (2 - P_{\text{ДЕФ}}) + (1 - P_{\text{ДЕФ}})^2 \times \left( \frac{2}{5} P_{\text{ДЕФ}} (P_{=} + P_{+=} + P_{-=} + P_{\% =}) + \frac{1}{3} P_{\text{ДЕФ}}^{**} (P_{* =} + P_{/=}) \right), \quad (15)$$

$$P_{\text{УСЛ}}^{\Pi} = \frac{1}{2} P_{\text{ДЕФ}}^{**} (1 - P_{\text{ДЕФ}})^2 (P_{* =} + P_{/=}), \quad (16)$$

$$P_{\text{ПРОП}}^{\Pi} = P_{\text{ДЕФ}} P_{\text{СВ}} (2 - P_{\text{ДЕФ}}) + (1 - P_{\text{ДЕФ}})^2 \times \left( \frac{3}{5} P_{\text{ДЕФ}} (P_{=} + P_{+=} + P_{-=} + P_{\% =}) + \frac{1}{6} P_{\text{ДЕФ}}^{**} (P_{* =} + P_{/=}) \right). \quad (17)$$

Вероятность отсутствия дефекта в конструкции присваивания, основываясь на выражениях (12), (13), (14), имеет вид

$$P_{\text{ОТСУТ}}^{\Pi} = (1 - P_{\text{ДЕФ}})^2 (P_{=} P_{=} + P_{+=} P_{+=} + P_{-=} P_{-=} + P_{\% =} P_{\% =}) = (1 - P_{\text{ДЕФ}})^3 (P_{=} + P_{+=} + P_{-=} + P_{\% =}). \quad (18)$$



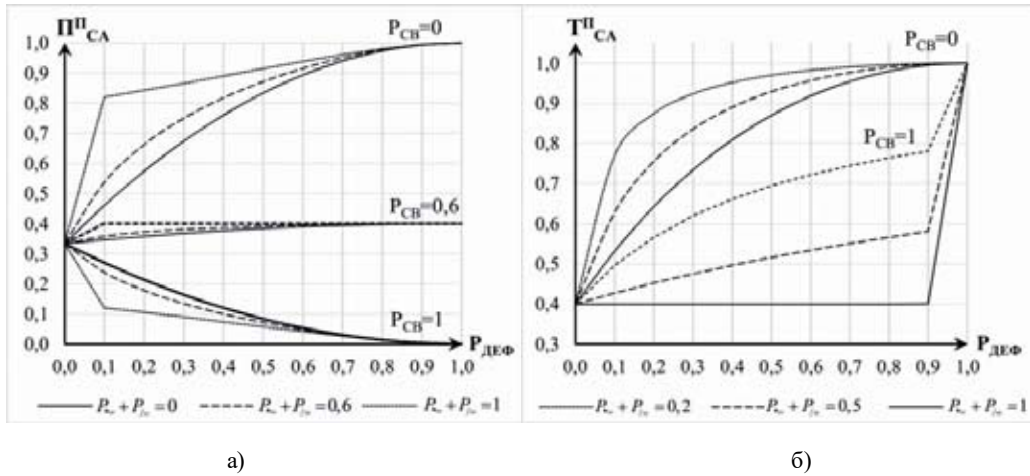


Рис. 4. Графики зависимости полноты (а) и точности (б) СА для поиска ДЕС в конструкциях присваивания от вероятности наличия дефекта операции или операнда

Fig. 4. Schedules of dependence SA completeness (a) and accuracy (б) for searching NSD in assignment construction from probability of operation or operand defect existence

Исходя из (1), (2), (15), (16) и (17), выражения для полноты и точности СА для поиска ДЕС в конструкциях присваивания следующие:

$$\Pi_{СА}^П = \frac{P_{ДЕФ}(1 - P_{СВ})(2 - P_{ДЕФ}) + (1 - P_{ДЕФ})^2 \left( \frac{2}{5} P_{ДЕФ} + (P_{*} + P_{/=}) \left( \frac{1}{3} - \frac{2}{5} P_{ДЕФ} \right) \right)}{P_{ДЕФ}(2 - P_{ДЕФ}) + (1 - P_{ДЕФ})^2 (P_{ДЕФ} + (P_{*} + P_{/=})(1 - P_{ДЕФ}))}, \quad (19)$$

$$T_{СА}^П = \frac{P_{ДЕФ}(1 - P_{СВ})(2 - P_{ДЕФ}) + (1 - P_{ДЕФ})^2 \left( \frac{2}{5} P_{ДЕФ} + (P_{*} + P_{/=}) \left( \frac{1}{3} - \frac{2}{5} P_{ДЕФ} \right) \right)}{P_{ДЕФ}(1 - P_{СВ})(2 - P_{ДЕФ}) + (1 - P_{ДЕФ})^2 \left( \frac{2}{5} P_{ДЕФ} + (P_{*} + P_{/=}) \left( \frac{5}{6} - \frac{2}{5} P_{ДЕФ} \right) \right)}. \quad (20)$$

С использованием выражений (19) и (20) получена зависимость полноты и точности СА для поиска ДЕС в конструкциях присваивания от вероятности наличия дефекта операнда или операции при фиксированных значениях вероятности совпадения ВЕС и вероятностей появления операций  $\{*=, /=\}$  в коде (рис. 4).

Полнота СА конструкций присваивания растет при уменьшении вероятности совпадения ВЕС. До уровня  $P_{СВ} \approx 0,675$  полнота анализа убывает при уменьшении вероятности появления в коде операций  $\{*=, /=\}$ , после данного значения зависимость принимает обратный характер. Точность СА конструкций присваивания зависит от частоты появления операций  $\{*=, /=\}$  и увеличивается при снижении их доли в коде программы.

### Модель ДЕС программных объектов в конструкциях сравнения

Операции сравнения  $\{<, >, <=, >=, ==, !=\}$  применяются только к операндам, обладающим одинаковыми ВЕС (рис. 5), откуда следует условие корректности для данных конструкций

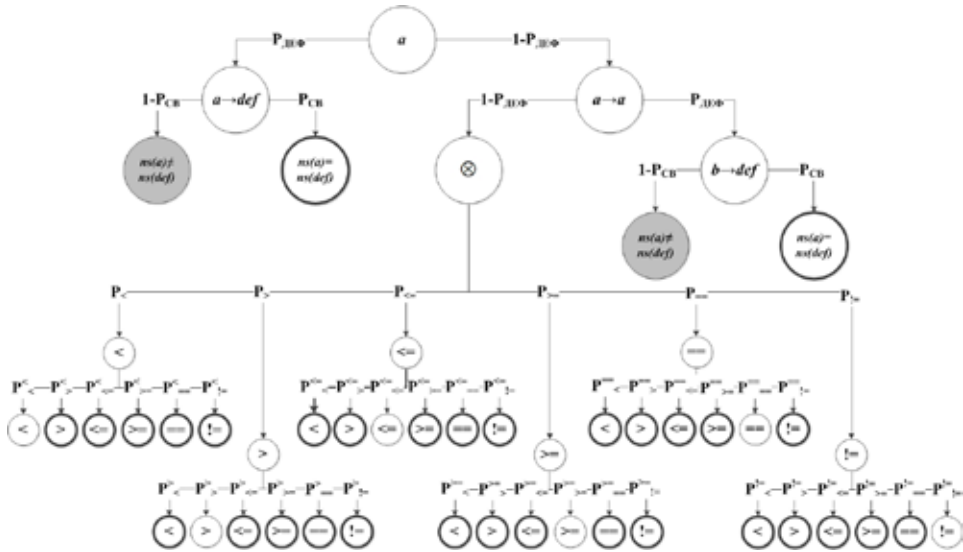


Рис. 5. Граф модели ДЕС программных объектов в конструкциях сравнения

Fig. 5. Model graph of software objects NSD for comparison construction

$ns(a) = ns(b)$ . Результатом выполнения операций сравнения является значение истина или ложь, что не позволяет определить наличие дефекта самой операции. Эффективность анализа конструкций сравнения зависит только от вероятности совпадения ВЕС операндов при искажении ( $P_{СВ}$ ) и не зависит от представленного в коде диапазона операций сравнения.

Условно истинных срабатываний анализатора при поиске дефектов в конструкциях сравнения не возникает ( $P_{УСЛ}^C = 0$ ), поэтому точность СА согласно соотношению (2) следующая:

$$T_{СА}^C = 1. \tag{21}$$

События наличия и отсутствия дефекта операции сравнения, а также ее операндов проявляются со следующими вероятностями:

$$P_{ДЕФ} = 5P_j^i, P_j^i = const, i \neq j, \tag{22}$$

$$1 - P_{ДЕФ} = P_i^i, \forall i \in \{<, >, <=, >=, ==, !=\}. \tag{23}$$

Аналогично рассуждениям для вышеописанных моделей ДЕС программных объектов и на основании соотношений (22) и (23) для конструкций сравнения имеем:

$$P_{ИСТ}^C = P_{ДЕФ} (2 - P_{ДЕФ}) (1 - P_{СВ}), \tag{24}$$

$$P_{УСЛ}^C = 0, \tag{25}$$

$$P_{ПРОП}^C = P_{ДЕФ} (P_{СВ} + (1 - P_{ДЕФ}) P_{СВ} + (1 - P_{ДЕФ})^2), \tag{26}$$

$$P_{ОТСУТ}^C = (1 - P_{ДЕФ})^3. \tag{27}$$

Модель ДЕС программных объектов для операций сравнения корректна по соотношению (3) для полученных выражений (24), (25), (26), (27).

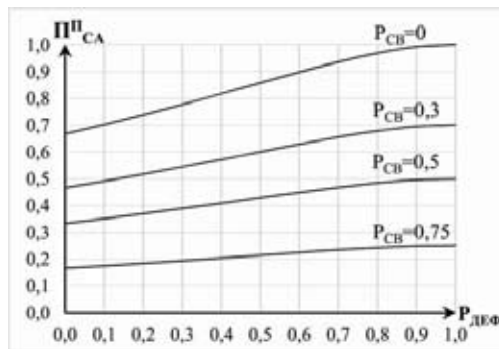


Рис. 6. Графики зависимости полноты СА для поиска ДЕС в конструкциях сравнения от вероятности наличия дефекта операции или операнда

Fig. 6. Schedules of dependence SA completeness for searching NSD in comparison construction from probability of operation or operand defect existence

Полнота СА для поиска ДЕС в конструкциях сравнения, исходя из соотношений (1), (24), (25), (26) такова:

$$\Pi_{СА}^C = \frac{(1 - P_{СВ})(2 - P_{ДЕФ})}{P_{ДЕФ}^2 - 3P_{ДЕФ} + 3}. \quad (28)$$

Согласно выражению (28) получена зависимость полноты анализа конструкций сравнения для поиска ДЕС от вероятности наличия дефекта операнда или операции при фиксированных значениях вероятности совпадения ВЕС (рис. 6).

Из анализа зависимости (рис. 6) следует, что полнота СА конструкций присваивания при уменьшении вероятности совпадения ВЕС повышается.

### Заключение

Таким образом, проведенное исследование позволяет сделать вывод о возможности применения СА для поиска нового типа функциональных дефектов – ДЕС программных объектов. Расчеты показали, что для ПО АС ВН, программные объекты которых интерпретируют широкий диапазон возможных значений естественной семантики, разрабатываемый подход является высокоэффективным. Данный факт открывает новые возможности по снижению затрат на верификацию ПО АС ВН.

### Список литературы

[1] Ицыксон В.М., Моисеев М.Ю., Цесько В.А., Захаров А.В., Ахин М.Х. Алгоритм интервального анализа для обнаружения дефектов в исходном коде программ. *Информационно-управляющие системы*, 2009, 2, 34-41 [Itsykson V.M., Moiseev M.Ju., Tsesko V.A., Zakharov A.V., Akhin M.H. Interval analysis algorithm for detecting defects in the software source code. *Information-control systems*, 2009, 2, 34-41 (in Russian)]

[2] Липаев В.В. *Технико-экономическое обоснование проектов сложных программных средств*. М.: СИНТЕГ, 2004. 284 с. [Lipaev V.V. Feasibility study of complex software projects. Moscow, SINTEG, 2004, 284 p. (in Russian)]

[3] Кулямин В.В. *Методы верификации программного обеспечения*. М.: Институт системного программирования РАН, 2008. 117 с. [Kuljamin V.V. *Methods of software verification*. Moscow, Institut sistemnogo programmirovaniya RAN, 2008, 117 p. (in Russian)]

[4] Жидков Р.Е. Методика верификации программного обеспечения автоматизированных систем военного назначения по семантической модели. *Сборник статей XXVIII Всероссийской научно-технической конференции школы-семинара «Передача, прием, обработка и отображение информации о быстротекающих процессах»*, М.: ИД «Академия Жуковского», 2017, 189-196 [Zhidkov R.E. Military automated system software verification technique by semantic model. *Digest of articles XXVIII All-Russian Scientific Technical Conference school-seminar "Transmission, reception, processing and display of information on fast processes"*, 2017, 189-196 (in Russian)]

[5] Глухих М.И., Ицыксон В.М., Цесько В.А. Использование зависимостей для повышения точности статического анализа программ. *Моделирование и анализ информационных систем*, 2011, 4(1), 68-79 [Glukhikh M.I., Itsykson V.M., Tsesko V.A. The use of dependencies for improving the precision of program static analysis. *Modeling and analysis of information systems*, 2011, 4(1), 68-79 (in Russian)]