

УДК 621.039:004.05

В.И. ДУЖИЙ, В.С. ХАРЧЕНКО, А.А. СТАРОВ, Д.О. РУСИН

Национальный аэрокосмический университет им. Н.Е. Жуковского "ХАИ", Украина

ИССЛЕДОВАНИЕ СЕРВИСОВ СПОРТИВНОГО ПРОГРАММИРОВАНИЯ КАК МНОГОВЕРСИОННЫХ ПРОЕКТОВ

Рассмотрена одна из проблем многоверсионных проектов (МВП), заключающаяся в сложности получения и анализа информации для их оценки диверсности. Предложено использовать конкурсные задачи, решаемые в спортивном программировании как базу для исследования МВП. Выполнен анализ требований, предъявляемых к экспериментальным данным для исследования показателей диверсности. Определены критерии выбора и выполнен анализ сервисов спортивного программирования. Отмечено, что наиболее подходящим сервисом является TopCoder®. Приведены схемы получения результатов решения задач, которые могут рассматриваться как МВП.

Ключевые слова: многоверсионный проект, оценка многоверсионных проектов, показатель диверсности, графовая модель, сервисы спортивного программирования, TopCoder®.

Введение

Постановка проблемы. К системам критического применения, разрабатываемым и эксплуатируемым в настоящее время, предъявляются особые требования надежности и функциональной безопасности [1]. Одним из основных способов обеспечения необходимых показателей в системах критического применения является внедрение комплекса мероприятий, нацеленных на достижение требуемых параметров информационно-управляющих систем (ИУС), входящих в состав таких комплексов. Использование структурной и версионной избыточности (диверсности) позволяет достичь необходимых значений показателей надежности и функциональной безопасности ИУС при выполнении заданных функциональных требований. Использование принципа диверсности определяется соответствующими нормативными документами и подтверждается практикой применения. Более широкое использование принципа диверсности отчасти сдерживается наличием проверенных методик количественной оценки значения показателя диверсности, а также его вклада в общие показатели надежности и функциональной безопасности ИУС. Эти методики должны соответствовать следующим требованиям: базироваться на теоретических моделях; подтверждаться результатами практического использования диверсности; позволять сопоставить теоретические расчеты и практические результаты.

В существующих публикациях количественные оценки для реальных проектов ИУС, использующих принцип диверсности, как правило, либо не приводятся, либо отсутствует количественная база для

вычисления показателей. Это объясняется соображениями безопасности, уникальным характером и сложностью проектирования многоверсионных ИУС, а также отсутствием нормативных показателей диверсности, которые необходимо использовать для оценки ИУС.

Анализ последних исследований и публикаций. Использование принципа диверсности позволяет уменьшить вероятность отказа по общей причине (ООП). Однако, учитывая уникальность многоверсионных проектов и их инфраструктурный характер, весьма сложной проблемой является их оценка. Обзор методов оценки МВП приведен в работе [2].

В работах [3-5] выполнен анализ метрик диверсности, используемых в метрико-вероятностном подходе. В работе [6] рассматривается оценка показателей МВП на основе модели β -фактора. Эти методы позволяют оценивать МВП на основании ошибок, полученных в процессе его эксплуатации. Недостатком такого подхода является возможность определения показателя диверсности *после* разработки МВП. Применение этого подхода на этапе разработки затруднительно. Кроме того, для сбора статистики достаточного объема требуется длительное время, а исправление ошибок – значительных затрат.

В работе [7] предложена классификация типов многоверсионности, учитывающая этапы жизненного цикла, на которых вносится версионная избыточность, предложено разделить ее на внешнюю и внутреннюю, а формирование МВП выполнять в два этапа, последовательно используя внешнюю и внутреннюю версионную избыточность.

Указанные методы оценки МВП базируются на использовании информации о дефектах версий (прямые методы) и обладают следующими недостатками: отсутствие возможности оценки МВП на всех этапах жизненного цикла, сложность определения степени корреляции теоретических оценок и практических результатов, а также длительное время, необходимое для их получения. Кроме того, недостаточный объем данных о дефектах не позволяет с уверенностью судить об их репрезентативности. В связи с этим необходим поиск подходов, которые могли бы дать возможность применять косвенные методы оценки диверсности, опираясь на репрезентативные источники данных.

Целью данной работы является анализ возможности использования сервисов спортивного программирования (ССП) как МВП для получения экспериментальных данных и их последующего исследования для разработки и верификации методик оценки диверсности.

Для достижения поставленной цели необходимо решить следующие задачи:

- сформировать требования к программам-решениям конкурсных задач с точки зрения возможности их использования в качестве версий для МВП;
- проанализировать сервисы спортивного программирования как аналог виртуальных МВП;
- сформулировать критерии выбора конкурсных задач для исследования различных аспектов разработки МВП;
- оценить качество полученных экспериментальных данных;
- сделать вывод о возможности использования экспериментальных данных для анализа показателей и разработки методик оценки диверсности в МВП.

1. Анализ сервисов спортивного программирования

1.1. Общие сведения о спортивном программировании

Спортивное программирование – направление профессионального программирования, нацеленное на решение алгоритмических задач повышенной сложности. *Конкурсные задачи* спортивного программирования являются авторской разработкой, имеют увлекательную завуалированную формулировку задания, для решения которого необходимо применить знание одного или нескольких нетривиальных алгоритмов, использующих сложные структуры данных [8]. В настоящее время классифицировано около 25-30 классов задач [9]. Задания формулируются на английском (чаще всего) или русском языке.

При решении конкурсных задач обычно соблюдаются следующие *ограничения*: не требуется знания новейших технологий программирования; решения представляются в виде консольного приложения; для ввода и вывода данных используются либо стандартные устройства ввода и вывода, либо текстовый файл оговоренного формата; формат входных и выходных данных для каждой задачи жестко фиксируется и его несоблюдение является ошибкой; вместе с формулировкой задачи предоставляется некоторый набор открытых тестов; программа должна удовлетворять требованиям по быстройдействию (время выполнения) и объему памяти для данных; задача считается решенной только после прохождения *всех* открытых и закрытых тестов.

Программисты, занимающиеся спортивным программированием, – мотивированные личности, имеющие углубленные знания в специальных разделах математики, непрерывно повышающие свою квалификацию и регулярно участвующие в соревнованиях. В соревнованиях по спортивному программированию участвуют отдельные участники или команды, включающие 3 человека. Спортивные программисты – лучшие программисты своего региона, имеющие опыт командной работы.

1.2. Анализ ССП

Наиболее популярные сервисы спортивного программирования, приведены в табл. 1. ССП содержат сайты, на которых выложены конкурсные задачи, и имеют средства для автоматизированного тестирования решений, получаемых от зарегистрированных пользователей.

ССП – это международные проекты, существующие длительное время, предоставляющие для решения сотни конкурсных задач, обслуживающие тысячи зарегистрированных пользователей, позволяющие решать выбранные задачи на одном из поддерживаемых языков программирования, используя один из имеющихся компиляторов. Для решения конкурсных задач необходима предварительная регистрация участников на сайте.

Для решения конкурсных задач ССП предоставляют следующие виды сервисов: автоматическая проверка решения задачи (сервис Online Judge), участие в соревнованиях (сервис SRM) и длительные соревнования (сервис Marathon). Как правило, каждый из ССП предоставляет только один вид услуг (Timus), хотя есть и такие, которые предоставляют несколько сервисов (TopCoder®).

Сервис *Online Judge* позволяет пользователю выбрать задачу для решения, решить задачу на локальной машине, проверив ее на открытых тестах, и отослать текст программы для проверки сервисом.

Получив задачу, сервис выполняет ее компиляцию, проводит автоматическое тестирование на полном наборе закрытых тестов, оценивает время выполнения и объем занимаемой памяти, и принимает решение о том, засчитывается ли решение данной задачи,

о чем сообщает пользователю. В случае неудачи пользователь имеет возможность исправить решение, и повторить попытку. Количество попыток и время, выделяемое на решение задачи, неограниченно. Задачи решаются одним участником.

Таблица 1

Сервисы спортивного программирования

| Название сервиса | Timus | Valladolid | Sphere | Lviv | SGU | TopCoder® | | |
|---|---------------|------------|--------|------|---------|------------------------|----------|---|
| Вид | Judge | | | | Contest | SRM | Marathon | |
| Статус | Международный | | | | | | | |
| Год создания | 2000 | 1997 | 2008 | 2004 | 2002 | 2001 | 2001 | |
| Количество задач | 738 | 2623 | 5230 | 260 | 397 | 624*3 192*3 86*3 | 67 | |
| Количество зарегистрированных пользователей | 43078 | 91428 | 59315 | 640 | 12243 | 6739 | 580 | |
| Доступные языки решения | C/C++ | + | + | + | + | + | + | + |
| | C# | + | - | + | - | + | + | + |
| | Java | + | + | + | - | + | + | + |
| | Pascal | + | + | + | + | + | - | - |
| | Другие | - | - | + | - | - | - | - |

Сервис *SRM* проводит соревнования среди пользователей, зарегистрировавшихся для участия в них. Участники соревнований должны решить несколько задач (обычно три) в течение ограниченного промежутка времени (обычно 75 мин). Проверка решений начинается после завершения соревнований, требования к решениям – такие же, что и при автоматизированной проверке задач Online Judge. Задачи решаются индивидуально.

Сервис *Marathon* предлагает в течение длительного периода времени индивидуально или в составе команды решить конкурсную задачу повышенной сложности оптимизационного характера, которая, как правило, не имеет детерминированного решения. Проверка решений проводится на большом объеме тестов, победители награждаются денежными призами, а спонсором соревнований является компания или организация с мировым именем. Современной тенденцией проведения марафонов является участие в них интернациональных команд, географически распределенных по всему земному шару, которые формируются организаторами из числа зарегистрированных участников. Есть все основания полагать, что это станет устоявшейся практикой. Спонсорами марафонов были NSA, NASA, компании Intel, nVidia и др.

1.3. ССП как МВП

Из проведенного анализа следует, что для каждого конкурсного задания t_i имеется множество программ-решений s_{ij} , обладающими следующими свойствами: они разработаны разными программистами, проживающими в различных государствах и обучающихся (закончившими) в различных ВУЗах; они

написаны на разных языках программирования. Наличие этих признаков позволяет сделать вывод о том, что *конкурсное задание и множество его решений является МВП*. Следовательно, каждый ССП представляет собой множество МВП, направленных на решение конкурсных задач. Согласно таксономии версионной избыточности (ВИ) [2], МВП на базе ССП обладают субъектной и программной ВИ. *Субъектная ВИ* ССП означает, что каждая программа-решение разрабатывается независимыми исполнителями, не знакомыми друг с другом, обучающихся в разных ВУЗах и имеющих разные национальности. *Программная ВИ* основана на использовании различных алгоритмов и логики их реализации, различных языков программирования, компиляторов, операционных систем и сред разработки (табл. 1).

Сведения, предоставляемые пользователям каждым ССП, различаются как по характеру, так и по объему (табл. 2). Существует, однако, минимальный набор данных, предоставляемых всеми ССП, – это информация о задачах и пользователях, зарегистрированных в системе. Интерфейсы ССП существенно различаются по удобству и функциональности, и в значительной мере определяют популярность того или иного ССП.

Для использования информации ССП в исследованиях она должна отвечать следующим требованиям: полнота, качество, достоверность. *Полнота* информации означает, что сервис хранит все необходимые данные о присланном на проверку решении для однозначной и корректной его оценки. Наиболее важными сведениями являются: тексты программ, перечень тестов, которые прошла или не прошла данная задача, а также ошибки, возникшие во время выполнения вследствие дефектов. *Качество* информации о решении основано на следующих факторах: высокая квалификация про-

граммистов, их высокая мотивация, большой набор тестов, автоматическое тестирование программ, проверка отсутствия копий решений, возможность пересмотра решения (rejudge) в случае нахождения ошибки в тестах. *Достоверность* означает, что имеющихся данных о программе-решении достаточно, чтобы однозначно и в реальном масштабе времени выполнить ее оценку, вынести решение – "решено – не решено".

Достаточным условием использования информации ССП в научных исследованиях является ее репре-

зентативность и доступность. Значительное количество задач и программ-решений позволяют говорить о *репрезентативности* имеющихся данных, позволяя применять к ним методы научного анализа, в том числе выполнять их статистическую обработку. *Доступность* информации означает возможность исследователю иметь доступ к той части информации ССП, которая обычно недоступна участникам, например, программам-решениям. К счастью, некоторые виды ССП такие данные предоставляют.

Таблица 2

Информация, предоставляемая пользователям различными ССП

| Название сервиса | Timus | Valladolid | Sphere | Lviv | SGU | TopCoder® | |
|--|-------|------------|--------|------|---------|-----------|----------|
| Вид | Judge | | | | Contest | SRM | Marathon |
| Сохраняются решения удачные | + | ? | + | + | – | + | + |
| Сохраняются решения неудачные | + | ? | + | + | – | + | + |
| Доступны чужие решения | – | – | – | – | – | + | + |
| Известно количество тестов | + | – | – | + | – | + | + |
| Известно количество посылок решений | – | + | + | + | + | – | – |
| Можно получить все не пройденные тесты | – | – | – | – | – | + | + |
| Есть информация о времени решения | + | + | + | + | + | + | – |
| Есть информация об используемой памяти | + | – | + | + | + | + | – |

Показатели диверсности можно оценивать при помощи прямых и косвенных метрик. Прямые метрики основаны на результатах тестирования программ, а косвенные – на основе анализа текста программ (табл. 3). Вывод о том, что наличие данных о номерах пройденных и не пройденных тестов, фактически тождественно данным о дефектах, базируется на следующем суждении. Если программа-версия П1 и программа-версия П2 проверяются тестами ТА и ТБ, и по результатам проверки тест ТА дал негативный результат для программ П1 и П2, это означает, что обе версии имеют сходные аномалии (групповые дефекты). В данном случае не важно, какой именно дефект (дефекты) являются причиной этого. Позитивный результат теста ТБ для версий означает лишь тот факт, что в них отсутствуют дефекты, обнаруживаемые тестом ТБ. К сожалению, известные ССП не предоставляют открытую информацию о том, – отличаются ли результаты негативного тестирования (различимы ли дефекты версий), либо не предоставляют её в явном виде.

Таблица 3

Необходимая для вычисления метрик диверсности информация, предоставляемая ССП

| Тип метрик | Тексты верных и ошибочных программ | Тесты пройденные и не пройденные |
|------------|------------------------------------|----------------------------------|
| Прямые | – | + |
| Косвенные | + | – |

Анализ табл. 3 показывает, что всю необходимую информацию для оценки показателя диверсно-

сти при помощи прямых и косвенных метрик зарегистрированным пользователям предоставляет только один ССП – TopCoder®. По этой причине в качестве источника репрезентативных данных для исследования показателей диверсности МВП рекомендуется использовать этот сервис.

2. Анализ схем получения экспериментальных данных с помощью ССП

С точки зрения исследования многоверсионных технологий соревнования по спортивному программированию можно рассматривать как эксперимент, направленный на получение практических данных о МВП, а сами ССП – в качестве источника таких экспериментальных данных. Перед тем, как использовать в исследованиях данные эксперимента, необходимо: формально оценить методику проведения эксперимента каждым ССП; выполнить их сравнительный анализ; определить показатель диверсности для каждого МВП; сопоставить степень соответствия правил проведения соревнований жизненному циклу (ЖЦ) реальных МВП. Схемы получения экспериментальных данных для различных ССП приведены на рис. 1-3.

Схема проведения эксперимента для ССП Online Judge приведена на рис. 1, а). Ее анализ показывает, что тексты конкурсных заданий публикуются до начала регистрации участника, он может ознакомиться с ним задолго до начала решения, и серьезно к нему подготовится. Задачи решаются индивидуально. Количество открытых тестов ограничено, а

число попыток для решения задачи, равно как и отведенное время, – нет (∞). Модель ЖЦ, используемая данным ССП, (рис. 1, в) соответствует автономной отладке системы на полном наборе тестов. Практика показывает, что в проекте все же могут остаться скрытые дефекты, и поэтому некоторые ССП практикуют пересмотр тестов в случае обнаружения в них ошибок и последующий пересмотр соответствующих решений.

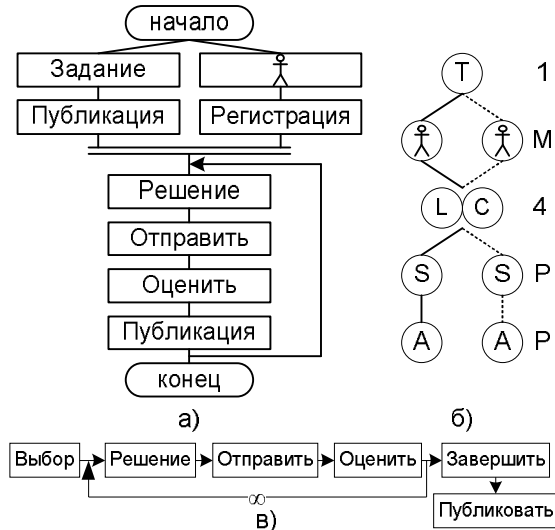


Рис. 1. Схема проведения получения данных с помощью ССП Online Judge

МВП на основе ССП Online Judge описывается графовой моделью, приведенной на рис. 1, б). Данная графовая модель показывает, что решается одна задача ($T=1$) несколькими пользователями (M), возможно использование четырех языков программирования (L) и компиляторов для них (C) (табл. 1). Решенная пользователем задача (S) оценивается сервисом (A) и в случае успеха публикуется сообщение об этом. Каждый пользователь решает не все задачи, поэтому $S < M$. Таким образом, множество решений P каждой задачи T , выполненное разными пользователями, составляет МВП.

Проведение эксперимента на основе ССП SRM (Single Round Match) представлено на рис. 2, а. В отличие от предыдущего эксперимента тексты заданий становятся известны только после начала состязаний. Для решения предлагается несколько конкурсных задач, которые должны быть решены индивидуально в течение ограниченного промежутка времени (75 мин). Количество открытых тестов также ограничено. Тексты решений участники отправляют на сервис для оценки, которая, однако, будет выполнена только после окончания соревнований, а результаты решений опубликованы. Поскольку результаты решений других участников становятся известны только по окончании состязания, даже косвенная оценка сложности решаемых

задач невозможна. Такая модель ЖЦ системы соответствует отладке и эксплуатации необслуживаемого изделия (рис. 2, в).

Графовая модель для МВП, полученного в результате проведения соревнований при помощи ССП SRM, представлена на рис. 2, б). Несколько пользователей (M) решают по три задачи ($T=3$), используя до четырех языков программирования (L) и компиляторов для них (C). Комплект из трех решений конкурсных заданий ($S=3$) оценивается ($A=3$) и публикуется на сервисе. Таким образом, МВП представляет собой P решений трех задач без возможности исправления обнаруженных ошибок.

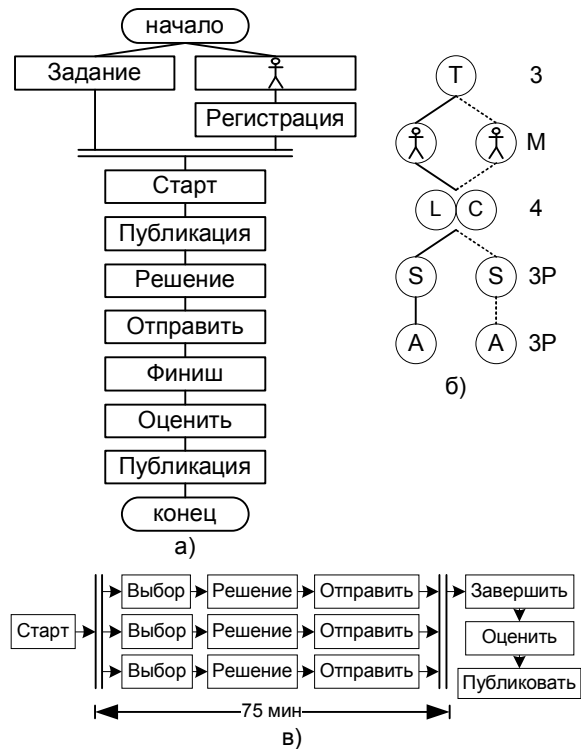


Рис. 2. Схема проведения получения данных с помощью ССП SRM

Схема проведения эксперимента для сервиса Marathon (рис. 3, а) представляет собой международный проект, к которому дважды применяется принцип диверсности: первый раз – на этапе формирования команды, а второй – при решении задачи. На начальном этапе формируются команды, которые будут участвовать в состязаниях. Для этого все зарегистрированные участники разделяются на группы в зависимости от стран (N) и университетов (U), в которых они обучаются. Далее все участники объединяются в команды, состоящие из пяти человек, проживающих в различных странах и обучаемых в разных университетах. Четыре команды составляют одну комнату (Room). Таким образом, в состязаниях участвуют команды, состоящие из 5 участников, территориально распределенных по всему земному шару.

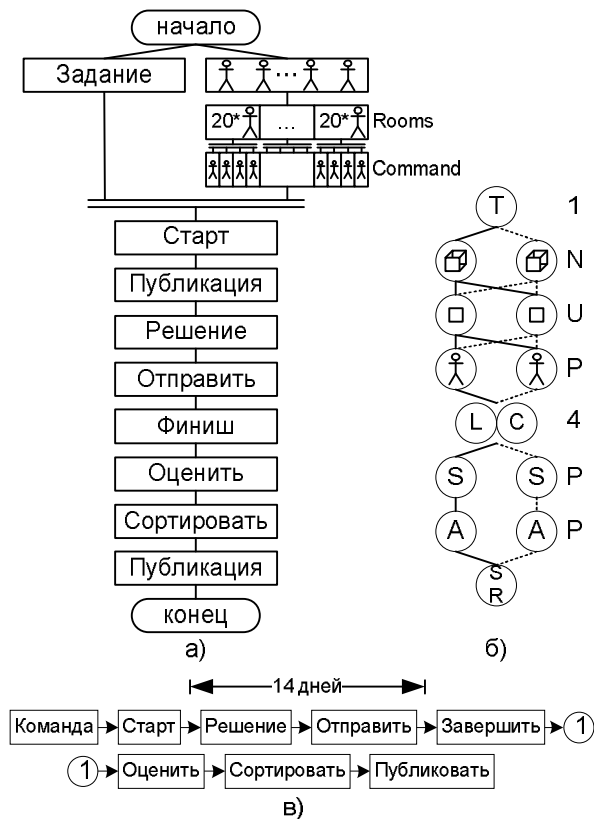


Рис. 3. Схема проведения получения данных с помощью ССП Marathon

Текст конкурсной задачи публикуется после начала состязания (рис. 3, а). Задание решается командой в течение 10-14 дней. Длительный срок решения задания обусловлен ее сложностью и распределенностью участников команды. Для проверки полученного решения используется большое количество открытых тестов, предоставляемых ССП. Решенная задача отправляется на сервис, который проверяет ее на некотором подмножестве исходных тестов. Полученные результаты от различных команд проверяются сервисом, после чего сортируются по комнатам и среди всех участников. Оценивается не только правильность решения задания, но и его оптимальность по нескольким заданным критериям. Такая модель ЖЦ системы (рис. 3, в) соответствует отладке и эксплуатации необслуживаемой изделия с повышенными требованиями к его надежности и эффективности.

Графовая модель для МВП, полученного в результате проведения соревнований при помощи ССП Marathon, представлена на рис. 3, б). Несколько команд (P) решают одну задачу (T=1). Имеется несколько отличий данного МВП от предыдущего, которые хорошо видны на графовой модели. Основное отличие заключается в том, что команда формируется с использованием принципа диверсности из представителей разных наций (N) и университетов (U). Вторым отличием является дополнительная количест-

венная оценка решения (SR) согласно заранее определенным критериям эффективности полученных результатов (A) с целью ранжирования участников для последующего определения победителей соревнований в комнате (Room) и среди всех команд.

Заключение

В данной работе решена задача анализа и выбора сервисов спортивного программирования, которые рекомендуется использовать в качестве источника экспериментальных данных о МВП. Для решения этой задачи выполнен сравнительный анализ ССП и предложено использовать сервис TopCoder® в качестве возможного МВП с полной информацией для вычисления показателей диверсности.

Получены следующие научные результаты:

- определены критерии для анализа ССП;
- определены необходимые и достаточные условия использования ССП в качестве источника данных МВП;
- выявлены виды диверсности, присущие ССП;
- исследованы схемы проведения соревнований ССП, разработаны графовые модели МВП и модели ЖЦ для каждого ССП.

Для получения информации с целью вычисления метрик диверсности, необходима предварительная обработка большого объема детальных данных и приведения их к форме, удобной для последующего анализа.

Дальнейшие исследования должны быть направлены на разработку методов и инструментальных средств автоматизированного анализа МВП, представленных в виде программ-решений ССП, усовершенствование методов оценки МВП, которые бы позволили проверить гипотезу о возможности и порядке использования косвенных метрик для оценки диверсности.

Литература

1. Айзенберг Я.Е. Сопоставление принципов обеспечения безопасности систем управления ракето-носителями и атомными электростанциями / Я.Е. Айзенберг, М.А. Ястребенецкий // Космічна наука та технологія. – 2002. – № 1. – С. 55-60.
2. Сиора А.А. (n-m) – версионные системы: таксономия модели и технологии / А.А. Сиора, В.В. Скляр, В.С. Харченко // Вісник Харківського національного університету. Серія "Математичне моделювання. Інформаційні технології. Автоматичні системи управління." – 2008. – № 833. – С. 231-246.
3. Preckshot G.G. Method for Performing Diversity and Defense-in-Depth Analyses of Reactor Protection Systems / G.G. Preckshot // Lawrence Livermore National Laboratory, California USA December, 1994. – 45 p.

4. Liu M. *Software Diversity Metrics and Measurements* / M.R. Liu, J-H. Chen, A. Avižienis // *Proc. IEEE COMPSAC 1992, Chicago, Illinois, September 1992.* – P. 69-78.

5. Харченко В.С. *Метрики диверсності: Класифікація, аналіз і застосування для оцінки надійності та безпеки комп'ютерних систем управління* / В.С. Харченко, І.В. Пискачева, В.В. Скляр // *Открытые информационные и компьютерные интегрированные технологии.* – Харьков: НАКУ «ХАИ», 2001. – № 9. – С. 194-214.

6. Hokstad P. *Common Cause Failure Modeling: Status and Trend* / P. Hokstad, M. Rausand // *Handbook of Performance Engineering* / K.V. Misra (ed.). – Springer, 2008. – P. 621-640.

7. Дужий В.І. *Розробка багатоверсійної ієрархії рішень при проектуванні програмного забезпе-*

чення / В.І. Дужий, І.В. Дужий, А.В. Шостак // *Радіоелектронні і комп'ютерні системи.* – 2007. – №8 (27). – С. 173-176.

8. Кормен Т.Х. *Алгоритмы: построение и анализ* / Т.Х. Кормен, Ч.И. Лейзерсон, Р.Л. Ривест, К. Штайн – М.: Издательский дом "Вильямс", 2005. – 1296 с.

9. Скиена С.С. *Олимпиадные задачи по программированию. Руководство по подготовке к соревнованиям* / С.С. Скиена, М.А. Ревилла – М.: КУДИЦ-ОБРАЗ, 2005. – 416 с.

10. Дужий В.І. *Оценка показателя версионной избыточности при проектировании программного обеспечения* / В.І. Дужий, І.В. Дужий, А.В. Шостак // *Радіоелектронні і комп'ютерні системи.* – 2009. – №6 (40). – С. 159-165.

Поступила в редакцію 12.02.2010

Рецензент: д-р техн. наук, проф. Г.Н. Жолткевич, Харківський національний університет ім. В.Н. Каразіна, Харків.

ДОСЛІДЖЕННЯ СЕРВІСІВ СПОРТИВНОГО ПРОГРАМУВАННЯ ЯК БАГАТОВЕРСІЙНИХ ПРОЄКТІВ

В.І. Дужий, В.С. Харченко, О.О. Старов, Д.О. Русін

Розглянута одна із проблем багатоверсійних проєктів (БВП), яка полягає у складності одержання та аналізу інформації для їх оцінки диверсності. Запропоновано використовувати конкурсні завдання, які вирішуються у спортивному програмуванні як базу для дослідження БВП. Виконано аналіз вимог, які пред'являються до експериментальних даних для дослідження показників диверсності. Визначені критерії вибору, та виконаний аналіз сервісів спортивного програмування. Відзначено, що найбільш придатним сервісом є TopCoder®. Наводяться схеми одержання результатів вирішення завдань, які можуть розглядатись в якості БВП.

Ключові слова: багатоверсійний проєкт, оцінка багатоверсійних проєктів, показник диверсності, графова модель, сервіси спортивного програмування, TopCoder®.

RESEARCH SPORTS PROGRAMMING SERVICES AS MULTIVERSION PROJECTS

V.I. Duzhyi, V.S. Kharchenko, O.O. Starov, D.O. Rusin

Considered one of the problems of multi-version projects (MVP) is the difficulty of obtaining and analyzing information for their evaluation. Proposed use of competitive tasks in sports programming as the basis for research MVP. Completed analysis of the requirements for the experimental data for the study of diversity index. Defined selection criteria and the analysis services of sports programming. It is noted that the most appropriate service is TopCoder®. A scheme of the results of the tasks that can be considered as the MVP.

Key words: multi-version project, score multi-version projects, diversity index, graph model, a sports programming services, TopCoder®.

Дужий Вячеслав Игоревич – старший преподаватель кафедры компьютерных систем и сетей Национального аэрокосмического университета им. Н.Е. Жуковского "ХАИ", Харьков, Украина, e-mail: V.Duzhy@csac.khai.edu.

Харченко Вячеслав Сергеевич – д-р техн. наук, проф., зав. кафедры компьютерных систем и сетей, Харьковский национальный аэрокосмический университет, "ХАИ", Харьков, Украина, e-mail: V.Kharchenko@khai.edu.

Старов Алексей Александрович – студент факультета радиотехнических систем летательных аппаратов Национального аэрокосмического университета им. Н.Е. Жуковского "ХАИ", Харьков, Украина, e-mail: A_Starov@hotmail.com.

Русин Дмитрий Олегович – студент факультета радиотехнических систем летательных аппаратов Национального аэрокосмического университета им. Н.Е. Жуковского "ХАИ", Харьков, Украина, e-mail: D.Rusin@live.ru.