

УДК 004.415.5

А.А. АНДРАШОВ, Ю.А. КРЕМЕНЧУЦКИЙ, В.С. ХАРЧЕНКО

Национальный аэрокосмический университет им. Н.Е. Жуковского «ХАИ», Украина

АНАЛИЗ МОДЕЛЕЙ ПРЕДСТАВЛЕНИЯ ТРЕБОВАНИЙ К ПРОГРАММНОМУ ОБЕСПЕЧЕНИЮ ПРИ ИХ ПРОФИЛИРОВАНИИ

Рассмотрены модели формального представления требований в контексте разработки метода профилирования требований программно обеспечения. Выделены критерии сравнения моделей. Определены достоинства и недостатки обозначенных моделей. Предложена схема профилирования требований ПО с использованием семантических фасетно-иерархических структур. Приведенная схема профилирования может быть положена в основу разработки информационной технологии создания проектно-ориентированных профилей требований к ПО.

Ключевые слова: профилирование, формальное описание, дефекты профилирования, требования к ПО, семантический анализ.

Введение

Разработка требований является одним из ключевых этапов жизненного цикла программного обеспечения (далее по тексту ПО). Анализ причин неудачного завершения проектов по разработке коммерческого ПО показывает, что в 30% случаев главной причиной провала явились ошибки определения спецификации [1]. Для успешно завершенных проектов затраты на исправление ошибок в определении требований составляют 25 - 40% от бюджета [2]. Для систем критического применения (ракетно-космическая техника, АЭС) риски, связанные с дефектами требований, являются одними из самых высоких.

Задачу разработки требований ПО можно рассматривать, как процесс формирования частного профиля требований (ЧП) на основе базового профиля требований (БП). Под базовым профилем требований понимается подмножество и/или комбинацию базовых стандартов информационных технологий, необходимых для реализации в проектируемой системе требуемых наборов функций. Базовыми стандартами могут являться официально принятые международные стандарты [3], отраслевые стандарты [4], а так же стандарты предприятий [5]. Частный профиль определяет требования к конкретному проекту ПО.

Для формализации процесса профилирования требований используется аппарат фасетно-иерархических структур (ФИС) [6]. ФИС позволяют представить формальную модель профиля требований в виде иерархической или фасетной структуры. Родительской вершиной ФИС является проектируемая система, как абстракция высокого уровня, корни

(таксоны) - конкретные требования к компонентам системы, причем, каждый уровень иерархии соответствует определенному классификационному признаку.

При таком подходе к работе с требованиями возможны три группы ошибок.

1. Ошибки полноты профиля (рис 1, а):

– неучтенная информация - значимое требование или реакция ПО либо не включены в профиль, либо не определены;

– посторонняя информация – информация, представленная в требовании, не требуется или не используется.

2. Ошибки глубины профиля (рис. 1, б):

– неоднозначность информации – требование имеет неоднозначную интерпретацию из-за использования нескольких терминов для определения одной характеристики или использования терминов имеющих различное значение в разных контекстах;

– противоречивость информации – два или более требований противоречат друг другу.

3. Ошибки формулировки требований (рис. 1, в):

– ложный факт – требование декларирует факт невыполнимый в условиях, определенных для системы;

– ошибочное структурирование – ошибки, связанные с включением требования в несоответствующий ему раздел.

ФИС обеспечивают локализацию ошибок полноты профилирования за счет формализации представления и операций над профилями требований с возможностью верификации множества таксонов и классификационных признаков. Для выявления ошибок глубины и формулировки требований необходимо усовершенствовать модель профилирования

на базе ФИС путем введения семантической составляющей.

Так как зачастую требования ПО представлены в виде тестов на естественном языке (ЕЯ), реализация семантической составляющей метода профилирования требует разработки модели формального представления текста.

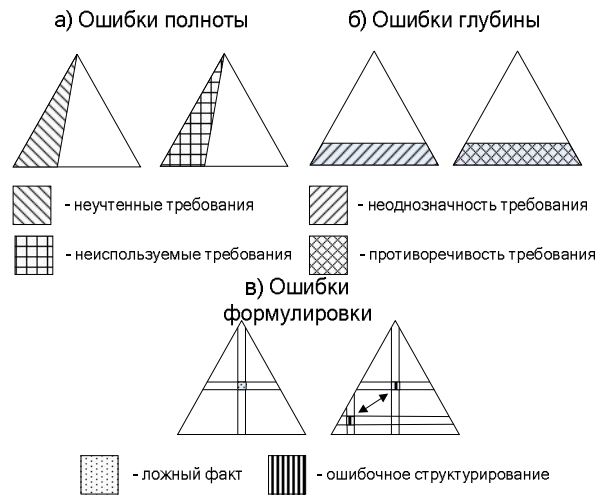


Рис. 1. Ошибки профилирования требований ПО

Задача формального представления информации на естественном языке является частной задачей представления знаний в экспертных системах, т.е. можно построить логическую цепочку: знания – информация – текст на ЕЯ. Исходя из выше сказанного, **целью статьи** является анализ моделей формального представления знаний на ЕЯ для использования при профилировании требований к ПО на базе ФИС.

1. Критерии выбора модели

Определим критерии, которым должна удовлетворять искомая модель.

1. Адаптируемость математического аппарата, используемого в модели. Математический аппарат искомой модели должен быть легко адаптируемым к использованию в нотации таксономических структур, обеспечивать корректное и полное описание всех свойств, необходимых для представления профилируемых множеств.

2. Наглядность. Модель должна обеспечивать естественность и наглядность в представлении объектов предметной области.

3. Апробированность. Модель должна быть хорошо изучена и широко используется, что позволит применить существующий опыт реализации и избежать возникновения инструментальных и методических ошибок.

4. Простота программной реализации. Модель должна обеспечивать возможность программной реализации в объектно-ориентированной парадигме программирования с применением реляционной базы данных.

5. Размерность. Модель должна минимизировать требования к объему памяти для хранения элемента модели при программной реализации.

2. Анализ моделей формального представления знаний на ЕЯ

Первые попытки формального представления знаний на ЕЯ были предприняты в 1949 году и были связаны с разработкой системы машинного перевода [7]. С увеличением объемов информации развивался как круг задач, так и модели представления знаний на ЕЯ. На сегодняшний день можно выделить 4 группы моделей формального представления знаний на ЕЯ: логические модели, продукционные модели, сетевые модели, фреймовые модели.

Проанализируем особенности указанных моделей в контексте задачи профилирования требований к ПО.

2.1. Логические модели

Логическая (предикатная) модель представления знаний основана на алгебре высказываний и предикатов, на системе аксиом этой алгебры и ее правилах вывода [8].

В основе логического способа представления лежит идея описания знаний о предметной области в виде некоторого множества утверждений, выраженных в виде логических формул, и получение решения путем вывода в некоторой формальной (дедуктивной) системе.

Знания, которые могут быть представлены с помощью логики предикатов, являются либо фактами, либо правилами. При использовании логических методов сначала анализируется структура предметной области, затем выбираются соответствующие обозначения и в заключении формируются логические формулы, представляющие собой закономерности рассматриваемой области. Множество таких формул является логической программой, содержащей информацию о предметной области.

Основные преимущества использования логики предикатов для представления знаний заключаются в том, что обладающий хорошо понятными математическими свойствами мощный механизм вывода может быть непосредственно запрограммирован; имеется возможность контроля логической целостности базы знаний, т.е. ее непротиворечивости и

полноты; весьма просто реализуется запись фактов.

Главным недостатком логической модели является отсутствие четких принципов организации фактов в базе знаний. В связи с этим достаточно большие базы трудно поддаются анализу и обработке.

Объем памяти, необходимый для хранения элемента знаний – предиката или предикатной формулы, мал в силу его простоты. Однако, база знаний, описывающая реальную, даже не очень сложную предметную область, должна содержать значительное количество указанных элементов.

Модель привлекает разработчиков высокой модульностью, легкостью внесения в систему дополнений и изменений. В качестве языка логического программирования можно использовать «ПРОЛОГ» [9].

Таким образом, логические модели представления знаний целесообразно использовать в тех предметных областях, где база знаний невелика по объему и относительно проста по структуре.

2.2. Продукционные модели

Продукционная модель наиболее адекватно соответствует процедурному характеру знаний. Эта модель описывает знания с помощью так называемых правил продукции, которые, по существу, являются программами, состоящими из одного оператора вида «ЕСЛИ <условие>, ТО<действие>.»

Если текущая ситуация (факты) в задаче удовлетворяет или согласуется с частью правила ЕСЛИ, то выполняется действие, определяемое частью ТО [10]. Это действие может оказаться воздействием на окружающий мир или же повлиять на управление программой (например, вызвать проверку и запуск некоторого набора других правил), или может сводиться к указанию системе добавить новый факт или гипотезу в базу данных.

Сопоставление частей ЕСЛИ правил с фактами может породить так называемую цепочку выводов – дерево решений [11]. Один из главных недостатков метода представления знаний с помощью правил – значительные затраты времени на построение цепочки вывода. При частом использовании какого-либо дерева решений система редуцирует ("сжимает") дерево решений до нового правила и вводит его в базу знаний. Это действие называют продукцией правил. Правило такого рода имеет значительно большую размерность, чем исходные правила. Системы, построенные на основе продукционных моделей, более эффективны по затратам памяти и по быстродействию, чем системы, основанные просто на правилах.

Модель направлена на решение простых, однородных задач и приводит к резкому падению эффективности решения таких проблем, которые состоят

из нескольких разнородных задач.

Серьезнейшим недостатком является невозможность эффективно описать правила с исключениями. Объем памяти, необходимый для хранения элемента знаний модели – конструкции ЕСЛИ – ТО, мал в силу его простоты. Однако база знаний, описывающая реальную, даже не очень сложную задачу, должна содержать сотни и тысячи правил [12].

2.3. Сетевые модели

Наиболее простым и универсальным средством представления знаний в системах искусственного интеллекта является семантическая сеть (СС). Семантические сети, разработанные как общий аппарат представления знаний, с самого начала активно использовались для построения систем обработки естественного языка. Один из ранних примеров такого использования можно найти в [13]

В общем случае СС представляет собой ориентированный граф, вершины которого обозначают сущности (объекты), а ребра – отношения (связи) между ними [14]. Имена вершин и ребер обычно совпадают с именами соответствующих сущностей и отношений, используемыми в естественном языке. Ребро и две связываемые им вершины представляют минимальную смысловую информацию – факт наличия связи определенного типа между соответствующими объектами.

По роду хранимой информации выделяют два типа семантических сетей: А-сети (концептуальные, интенциональные) и К-сети (фактуальные, экстенциональные). Первые содержат множество объектов и отношений, допустимых в данной предметной области; вторые – множество объектов и отношений, присутствующих в описании конкретной ситуации.

СС обеспечивают наглядность отображения объектов, связей и отношений, а также гибкость представления знаний за счет ассоциативности и иерархичности наследования в сети, т.е. элементы более низкого уровня в сети могут наследовать свойства элементов более высокого уровня, что экономит память, поскольку информацию о наследуемых свойствах не нужно повторять в каждом узле сети. Такие модели хорошо приспособлены для хранения в реляционных базах данных. К недостаткам модели можно отнести тот факт, что для описания даже небольшой предметной области размер концептуальной сети будет весьма велик.

Модель универсальна и легко настраивается. СС активно используются на практике, примером удачной реализации является система SNePS широко применяемая, как средство разработки экспериментальных приложений, использующих естественный язык [15].

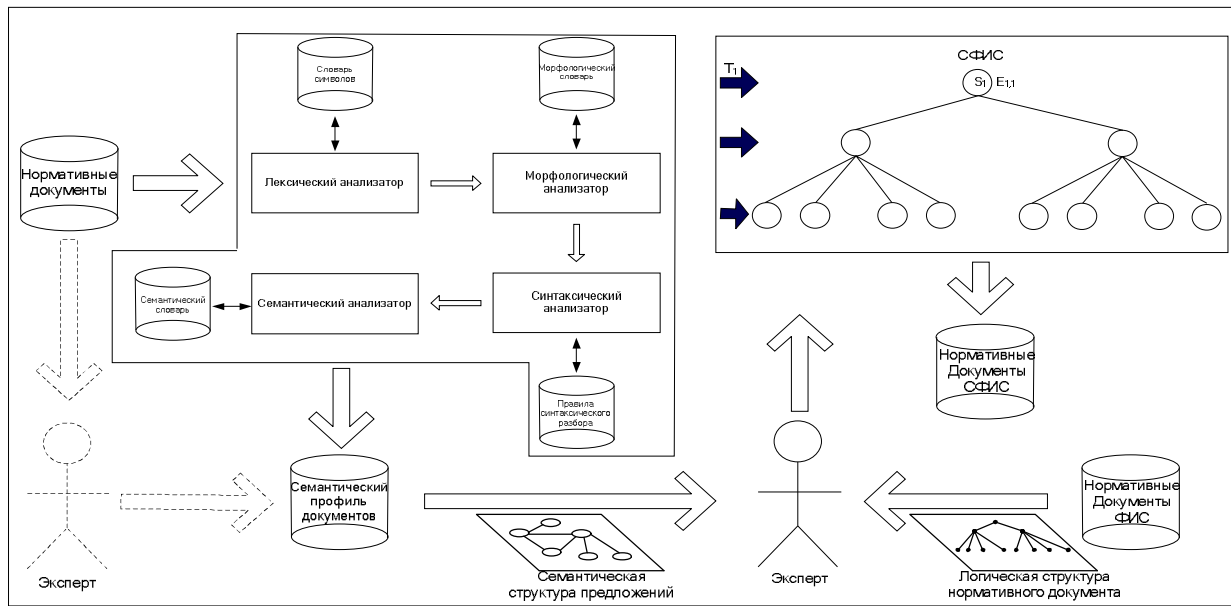


Рис. 2. Схема профилирования требований ПО с использованием СФИС

2.4. Фреймовые модели

Модель представления знаний с помощью фреймов предложена Марвином Минским, который описывает их следующим образом [16]: "Фрейм – это структура данных, представляющая стереотипную ситуацию. Каждому фрейму присоединяются несколько видов информации. Часть этой информации – о том, как использовать фрейм. Часть о том, чего можно ожидать далее. Часть о том, что следует делать, если эти ожидания не подтвердятся".

Фреймовая модель по своей организации во многом похожа на семантическую сеть. Она является сетью узлов и отношений, организованных иерархически: верхние узлы представляют общие понятия, а подчиненные им узлы – частные случаи этих понятий. В системе, основанной на фреймах, понятие в каждом узле определяется набором атрибутов-слотов (например, процесс, объект, свойство) и значениями этих атрибутов (например, верификация, требований, независимая). Каждый слот может быть связан со специальными процедурами, которые выполняются, когда информация в слотах (значения атрибутов) меняется. С каждым слотом можно связать любое число процедур.

Описание некоторой предметной области в виде фреймов обладает высоким уровнем абстрактности. Фреймовая система не только описывает знания, но и позволяет человеку описывать метазнания, т.е. правила и процедуры обработки.

Модель является универсальной, поскольку существуют не только фреймы для обозначения объектов и понятий, но и фреймы-роли (разработчик, эксперт, пользователь), фреймы-ситуации (тре-

вога, рабочий режим устройства) и др.

Достоинства и недостатки фреймовой модели аналогичны сетевым моделям.

3. Вариант схемы профилирования требований ПО

Анализ моделей формального представления знаний на ЕЯ позволил разработать вариант схемы профилирования требований ПО с использованием семантических фасетно-иерархических структур (СФИС) (рис. 2). СФИС – фасетно-иерархическая структура, таксоны которой содержат семантическое описание вербальных требований к системе. Формально СФИС можно представить следующей пятеркой $SFIS = \{E, \Psi_{EE}, T, \Psi_{ET}, S_E\}$, где E – непустое множество таксонов, T – множество классификационных признаков, Ψ_{EE} – функция разметки таксонов (связь между таксонами разных уровней), Ψ_{ET} – функция разметки классификационных признаков, служит для отображения классификационных признаков на таксоны, S_E – семантический код таксона, содержит семантическую структуру предложения или множества предложений, определяющих физический смысл таксона в контексте анализируемого нормативного документа.

Входной информацией для метода профилирования служат нормативные документы, представленные в виде текстовых файлов. Исходный текст попадает на обработку в лексический анализатор, который преобразует его в множество лексических единиц (ЛЕ) (слов, чисел или знаков препинания).

Далее множество лексических единиц поступает в морфологический анализатор. Здесь для каждой лексической единицы, путем сопоставления ЛЕ с содержимым морфологического словаря, определяются ее принадлежность к определенной части речи (существительное, глагол и т.д.) и набор атрибутов, характеризующий эту часть речи (род, число, падеж), а так же выполняется поиск омонимов (слов имеющих одинаковое написание, но разную смысловую нагрузку).

На следующем шаге множество наборов омонимов передается в лексический анализатор, где на основании набора правил определяются варианты группировки слов в дерево синтаксических зависимостей (предложения). Набор правил определяет порядок следования частей речи с учетом их атрибутов, допустимый для предложений на ЕЯ.

Результаты, полученные на этапе лексического анализа, нуждаются в экспертной верификации, т.е. из множества вариантов разбора дерева зависимостей конкретного предложения, эксперт определяет вариант, наиболее соответствующий исходному тесту.

Семантический анализатор трансформирует дерево синтаксических зависимостей в семантическую сеть - ориентированный граф, вершины которого обозначают сущности (объекты), а ребра - отношения (связи) между ними. Ребро и две связываемые им вершины представляют минимальную смысловую информацию - факт наличия связи определенного типа между соответствующими объектами.

Множество объектов и связей между ними, характерных для данной предметной области определены в семантическом словаре. Результатом работы семантического анализатора является множество семантических структур, соответствующих предложениям из исходного теста. Множество семантических структур нормативного документа формируют семантический профиль документа, хранящийся в базе данных. Семантический профиль нормативного документа нуждается в экспертной верификации, так как качество работы современных семантических анализаторов не обеспечивает адекватного качества разбора текста.

Затем эксперт производит связывание семантических структур с соответствующими таксонами факетно-иерархической модели, описывающей конкретный нормативный документ (алгоритм построения ФИС модели описан в [6]). Эта работа требует хорошего знания нормативной базы предметной области и нотации представления семантических сетей. На выходе процедуры связывания логической структуры нормативного документа и его семантического профиля имеем СФИС нормативного документа. СФИС нормативного документа помещается в базу

данных для дальнейшей обработки. Рассмотрим схему получения частного профиля требований ПО с применением СФИС (рис. 3). База данных содержит множество нормативных документов, соответствующих заданной предметной области, представленных в виде СФИС. Эксперт извлекает из базы данных базовый профиль требований, который содержит множество нормативных документов, регламентирующих разработку ПО в определенной предметной области. Затем, на основании особенностей конкретного проекта (требуемого набора функций), происходит выделение частного профиля требований. Дальнейшее прикладное использование СФИС требует разработки математического аппарата, позволяющего совершать преобразования заданных СФИС.

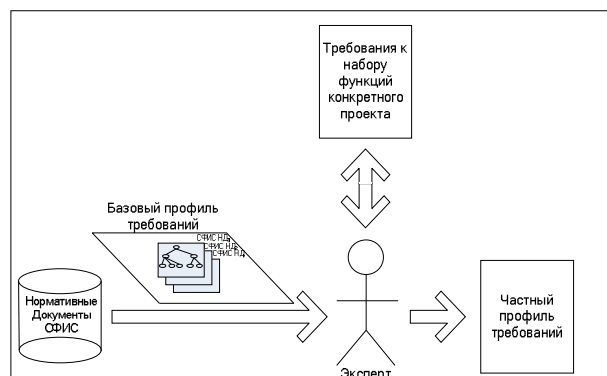


Рис. 3. Схема получения частного профиля требований ПО

Заключение

При рассмотрении типовых моделей представления знаний в контексте задачи профилирования, были выявлены достоинства и недостатки каждой из перечисленных моделей представления. Анализ показал, что для реализации метода профилирования требований ПО целесообразно использовать модель на базе семантических сетей. Этот подход позволяет адекватно (полно и непротиворечиво) описать предметную область, является хорошо изученным и активно применяется на практике, предоставляет достаточный уровень наглядности представления информации, за счет наследования обеспечивает минимизацию памяти, необходимой для хранения элемента модели.

Приведенные схемы профилирования могут быть положены в основу разработки информационной технологии создания проектно-ориентированных профилей требований к ПО.

Литература

1. Davis A. *Just Enough Requirements Management: Where Software Development Meets Marketing* / A. Davis. – New York: Dorset House, 2005. – 240 с.

2. Леффингуэлл Д. Принципы работы с требованиями к программному обеспечению. Унифицированный подход / Д. Леффингуэлл, Д. Уидриг. – М.: Вильямс, 2002. – 448 с.
3. ISO/IEC 12207:2008. Systems and software engineering – Software life cycle processes, 2008. – 124 p.
4. НП 306.5.02/3.035-2000. Требования по ядерной и радиационной безопасности к информационным и управляющим системам, важным для безопасности атомных станций, 2000 – 80 с.
5. РД 522-048-2002. Руководство по качеству ИТ СКБ «ПОЛИСВИТ» ГНПО «КОММУНАР», 2002. – 95 с.
6. Гордєєв О.О. Фасетно – ієрархічні структури у задачах оцінки якості програмного забезпечення / О.О. Гордєєв, В.С. Харченко // Інформаційні технології та комп'ютерна інженерія. – 2005. – № 3. – С. 190-196.
7. Weaver W. Translation. Technical Report, 1949. Reprinted in Machine Translation of languages, 1955 / Cambridge, MA: MIT Press. – P. 15-23.
8. Hammond P. Logic programming for expert systems / P. Hammond . - M. Sc. Thesis. Dept. of Computing. Imperial College, Univ of London, England, 1980. - 130 p.
9. Братко И. Алгоритмы искусственного интеллекта на языке PROLOG = Prolog Programming For Artificial Intelligence / И. Братко. - М.: Вильямс, 2004. - 640 с.
10. Уотермен Д. Руководство по экспертным системам: Пер. с англ. / Д. Уотермен. - М.: Мир, 1989. - 388 с.
11. Экспертные системы. Принципы работы и примеры: Пер. с англ. / Под ред. Р. Форсайта. - М.: Радио и связь, 1987. - 224 с.
12. Джексон П. Введение в экспертные системы. / П. Джексон. – М.: Вильямс, 2001. – 624 с.
13. Simmons R.F. Semantic Networks: Their Computation and Use for Understanding English Sentences / R.C. Schank, K.M. Colby (eds.)// Computer Models of Thought and Language. – San Francisco: Freeman, 1973. – P. 63-113.
14. Рубашкин В. Представление и анализ смысла в интеллектуальных информационных системах / В. Рубашкин - М.: Наука, 1989. – 234 с.
15. Shapiro S.C. The SNePS family / S.C. Shapiro, W.J. Rapaport // Computers & Mathematics with Applications. – 1992. - 23:5. - P. 243-275.
16. Минский М. Фреймы для представления знаний / М. Минский. - М.: Мир, 1979. – 152 с.

Поступила в редакцию 15.02.2009

Рецензент: д-р техн. наук, проф., проф. Кафедры Б.М. Конорев, Национальный аэрокосмический университет им. Н.Е. Жуковского «ХАИ», Харьков.

АНАЛІЗ МОДЕЛЕЙ ПРЕДСТАВЛЕННЯ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ПРИ ЇХ ПРОФІЛЮВАННІ

А.О. Андрашов, Ю.О. Кременчуцький, В.С. Харченко

Розглянуто моделі формального представлення вимог в контексті розробки метода профілювання вимог програмного забезпечення. Виділені критерії порівняння моделей. Визначені переваги і недоліки позначених моделей. Запропонована схема профілювання вимог ПЗ з використанням семантичних фасетно-ієрархічних структур. Наведена схема профілювання може бути покладена в основу розробки інформаційної технології створення проектно-орієнтованих профілів вимог ПЗ.

Ключові слова: профілювання, формальний опис, дефекти профілювання, вимоги до програмного забезпечення, семантичний аналіз.

ANALYSIS OF REQUIREMENTS PRESENTATION MODELS TO SOFTWARE DURING THEIR PROFILING

A.A. Andrashov, U.A. Kremenchutskiy, V.S. Kharchenko

Models of formal requirements presentation were considered in the context of development of method of software requirements profiling. Criteria of models comparison were outlined. Advantages and disadvantages for outlined models were established. It was suggested to use software requirements profiling model with application of semantic facet - hierarchy structures. Proposed profiling scheme could be applied as a basis for developing an information technology for creation project-oriented SW requirements profiles.

Key words: profiling, formal description, profiling faults, SW requirements, semantic analysis.

Андрашов Антон Александрович – ассистент кафедры Национального аэрокосмического университета им. Н.Е. Жуковского «ХАИ», Харьков, Украина, e-mail: anton1a@rambler.ru.

Харченко Вячеслав Сергеевич – д-р техн. наук, проф., зав. кафедрой Национального аэрокосмического университета им. Н.Е. Жуковского «ХАИ», Харьков, Украина, e-mail: V.Kharchenko@khai.edu.

Кременчуцкий Юрий Александрович – студент 5 курса Национального аэрокосмического университета им. Н.Е. Жуковского «ХАИ», Харьков, Украина.