

**ОБ АВТОМАТИЗИРОВАННОМ АНАЛИЗЕ
РЕЗУЛЬТАТОВ ОТРАБОТКИ ИСПЫТАТЕЛЬНОГО
ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ
АВТОМАТИЧЕСКОГО КОСМИЧЕСКОГО АППАРАТА**

© 2016 г. Карасёв А.А.

Конструкторское бюро «Арсенал» имени М.В. Фрунзе (КБ «Арсенал»)
Ул. Комсомола, 1-3, г. Санкт-Петербург, Российская Федерация, 195009,
e-mail: kbarsenal@kbarsenal.ru

В настоящее время испытания автоматических космических аппаратов проводятся в автоматизированном режиме с помощью соответствующего программного обеспечения на автоматизированном испытательном комплексе. На этапе выполнения опытно-конструкторских работ возникает задача оценки отработанности конструкторской документации. Такую оценку можно проводить с помощью автоматически генерируемых протоколов испытаний, которые необходимо проанализировать. Их количество слишком велико, чтобы выполнять эту работу вручную. Для решения этой проблемы было разработано вспомогательное программное обеспечение, которое может провести такую оценку по данным протоколов и испытательных программ.

При использовании такого подхода для одного из опытных космических аппаратов разработки конструкторского бюро «Арсенал» удалось получить результаты, коррелирующиеся с представлениями о качественной оценке отработки. Таким образом, предложенный метод доказал свою эффективность при косвенной оценке результатов испытаний космических аппаратов.

Предложенный подход позволил решить поставленную задачу по оценке результатов с представлением статистических величин, которые ранее были достоверно неизвестны.

Ключевые слова: *испытания, испытательное программное обеспечение, анализ результатов испытаний, автоматический космический аппарат.*

**ON THE AUTOMATED ANALYSIS
OF THE DEVELOPMENT RESULTS OF THE TEST SOFTWARE
FOR UNMANNED SPACECRAFT**

Karasev A.A.

Design Bureau Arsenal named after M.V. Frunze (DB Arsenal)
1-3 Komsomola str., Saint-Petersburg, 195009, Russian Federation, e-mail: kbarsenal@kbarsenal.ru

Currently the tests of unmanned spacecraft are conducted in the automated mode using the corresponding software at the computer-aided test facility. In the development phase there is a problem of evaluating the execution of design documentation. Such an evaluation can be made using automatically generated test protocols to be analyzed. Their quantity is large enough to perform this operation manually. To solve this problem the support software which can make such an evaluation by data of protocols and test programs was developed.

When using this approach for one of the spacecraft prototypes developed by design bureau Arsenal one can get the results correlating with the requirements for qualitative evaluation of the development test. Thus, the proposed method proved its efficiency during indirect evaluation of the spacecraft test results.

The proposed approach allowed to solve the problem of evaluating the results with presentation of statistical values which earlier were not known for a fact.

Key words: *tests, test software, test results analysis, unmanned spacecraft.*



КАРАСЁВ А.А.

КАРАСЁВ Александр Александрович — начальник сектора КБ «Арсенал», e-mail: kbarsenal@kbarsenal.ru
 KARASEV Alexander Alexandrovich — Head of Subdepartment at DB Arsenal, e-mail: kbarsenal@kbarsenal.ru

Введение

Современный автоматический космический аппарат (АКА) представляет собой часть сложной технической системы, созданной для достижения конкретной цели. Это могут быть наблюдение за поверхностью Земли, создание навигационного поля, выполнение научной миссии либо обеспечение инфо-коммуникационных услуг.

АКА — это сложная система, которая является совокупностью приборов и подсистем, объединенных в одну информационную структуру, представимую локальной вычислительной сетью.

Взаимодействие внутри сети может различаться в зависимости от выбранного протокола обмена. Одно можно сказать совершенно определенно — управление АКА осуществляется в бортовом контуре управления с помощью высоконадежного бортового цифрового вычислителя.

Использование современной элементной базы в составе АКА предполагает увеличение его программной сложности. Под этим термином будем понимать увеличение размера исполняемого кода, увеличение состава алгоритмического обеспечения.

Наилучшим решением для испытаний такого рода систем является их проведение в автоматизированном режиме по специальным испытательным программам.

Совокупность таких программ и документации, созданных с целью проведения автоматизированных испытаний АКА, далее будем называть испытательным программным обеспечением (ИПО).

На этапе выполнения опытно-конструкторской работы перед разработчиком АКА ставится задача отработки документации. В эту задачу также входит подзадача оценки отработки ИПО на АКА. Как правило, качественно эта подзадача решается по результату, полученному при проведении испытаний.

Для получения количественной оценки требуется обработать большой объем данных,

и ручную, без соответствующего программного обеспечения, сделать это весьма проблематично.

Данная работа посвящена вопросу получения количественной оценки отработки ИПО на опытном АКА.

Анализ предметной области

Анализ процедуры испытаний АКА. При изучении вопроса получения количественной оценки отработки ИПО АКА рассмотрим процедуру автоматизированных испытаний АКА (рис. 1).



Рис. 1. Процедура автоматизированных испытаний

Примечание. АИК — автоматизированный испытательный комплекс.

Оператор автоматизированного испытательного комплекса (АИК), действуя согласно своей инструкции, выбирает испытательную программу для проведения работы с АКА. Эта программа по разработанному алгоритму осуществляет взаимодействие с системами АКА и с контрольно-проверочным оборудованием для выяснения их технического состояния. В ходе этой операции формируется протокол испытаний, где фиксируются все воздействия и ответные реакции от АКА с привязкой ко времени проведения работ.

АИК является основным инструментом в руках испытателей для проведения автоматизированных электрических проверок АКА. В его состав входят необходимые модули сопряжения и интерфейсы для осуществления взаимодействия с АКА. Помимо этого, АИК и другое контрольно-проверочное оборудование объединены в локальную вычислительную сеть. Это позволяет получить доступ к дополнительной информации о техническом состоянии объекта испытаний и провести ее обработку.

В завершение работ протокол испытаний анализируется операторами систем для оценки исправности бортовой аппаратуры. Как правило, положительное заключение дается по отсутствию оценки «ненорма» в тексте протокола.

Анализ ИПО АКА. ИПО АКА предназначено для автоматизации процесса его электрических испытаний. ИПО создается на специальном проблемно-ориентированном языке программирования и исполняется в специальной среде испытаний АИК. Эта среда позволяет в шаговом и автоматическом режимах исполнять алгоритмы проверок, выдавать вручную все воздействия, протоколировать результаты испытаний.

Рассматривая состав ИПО, можно сделать вывод, что оно различается в зависимости от типа АКА и задействованного в испытаниях проверочного оборудования. Количество программ также зависит от подхода, которого придерживается разработчик для испытаний своей системы. Так, для приборов бортового специального комплекса космической системы, включаемых периодически, можно выделить шаблон — проведение испытаний сеансами. Для каждого такого испытания может быть создана отдельная испытательная программа.

Обратим внимание на проблемно-ориентированный язык испытаний, используемый для проверок АКА с помощью АИК.

Практика его использования на предприятии началась с 1980-х гг., а разработку этого языка можно отнести к 1960-м гг. Именно этим можно объяснить его построение в виде таблиц, где каждая строка — это определенная макрокоманда.

Для управления объектом испытаний необходимо использовать эти макрокоманды. Совокупность макрокоманд объединяется разработчиком в логическую единицу, именуемую входом программы. Входы программ не разделены между собой, поэтому один вход может использовать код другого. Такой подход требует большой дисциплины от разработчика, чтобы соблюдать принципы структурного программирования.

В структуре ИПО можно выделить определенную иерархию. В ее основе лежат макрокоманды, выше идут входы программ и подпрограммы. Их совокупность образует целую программу. Множество всех программ для испытаний АКА называется формированием. Структурно такая организация представлена на рис. 2.

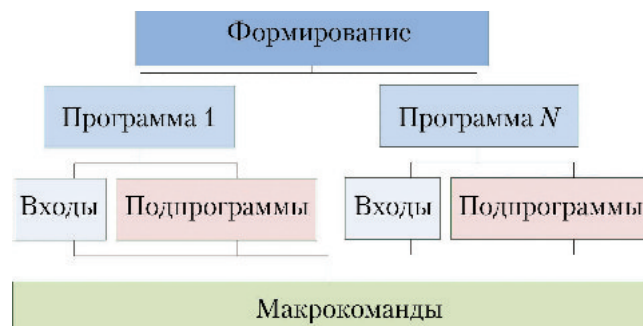


Рис. 2. Иерархическая организация испытательного программного обеспечения

Среда испытаний в АИК работает с формированием ИПО, проводя его контрольное суммирование. По разным причинам, которыми могут быть модернизация приборов в АКА, их ремонт либо обнаружение ошибок в ИПО, в процессе испытаний возникает необходимость в корректировке отдельных макрокоманд в испытательных программах, либо добавление новых строк с ними. Это ведет к изменению контрольной суммы в программе и, как следствие этого, изменяется также контрольная сумма формирования. Помимо этого, в процессе испытаний может возникнуть ситуация с необходимостью добавления новых программ для уточнения методики испытаний. Все это говорит о том, что формирование ИПО не является статичным и имеет свою собственную динамику в процессе проведения испытаний.

При анализе отработки ИПО и получения ее количественных характеристик это явление необходимо учитывать.

При разработке программ с использованием программы-оболочки испытатель приближен к заданию кодов программ вручную. Следствием этого является наличие нескольких недостатков:

- код недостаточно структурирован;
- разработчик волен задать пропуски строк в тексте программ (в качестве резерва), что ведет к увеличению занимаемого места в запоминающем устройстве АИК.

Для решения проблемы создания ИПО в терминах макрокоманд КБ «Арсенал» был создан проблемно-ориентированный язык программирования «БАЗИС» [1]. С его использованием испытатели ушли от составления

макрокоманд вручную, предоставив эту работу транслятору. Такой подход позволил существенно сократить затраты на разработку испытательного программного обеспечения.

Помимо этого, код, полученный в результате машинного перевода из текста проблемно-ориентированного языка в текст макрокоманд, получился без пропусков строк и лучше структурирован.

Анализ протоколов автоматизированных испытаний АКА. При проведении автоматизированных электрических проверок АКА перемещается в своем фазовом пространстве из начального (исходного) состояния S_0 в некоторое состояние S_N . Этот путь отмечается в протоколах испытаний путем фиксации исполнения директив оператора и выполнения строк макрокоманд. Эти события привязываются к шкале времени АИК.

С помощью испытательной программы разработчик может управлять протоколом испытаний. У испытателя есть возможность часть строк в программе отметить без вывода на печать в протокол. В этом случае подтвердить выполнение таких строк в программе по протоколу будет невозможно. Это обстоятельство накладывает определенные ограничения на автоматизированную оценку отработки ИПО по протоколам АИК.

На общем фоне вклад таких строк в суммарный объем протоколов сравнительно невелик, и далее при рассмотрении им можно пренебречь.

В начале протокола следует его заголовочная часть, в которой указывается служебная информация, содержащая дату и время составления протокола, его название. В некоторых случаях здесь может быть помещена информация об АИК. Пример такого содержания приводится ниже:

ИМЯ КАТАЛОГА ПРОТОКОЛА ИСПЫТАНИЙ - 150603_1436o_1

...

ДАТА ПРОВЕДЕНИЯ ИСПЫТАНИЙ 3/06/2015

ВРЕМЯ ПРОВЕДЕНИЯ ИСПЫТАНИЙ 14ч36м52с343мс

Эти данные определяют атрибуты протокола, которые можно использовать при автоматизированном анализе.

Теперь обратим внимание на содержательную часть протокола. В основном это результаты отработки макрокоманд в испытательных программах. Такие строки содержат следующую информацию:

- номер программы;
- номер строки;
- шифр макрокоманды;
- время АИК;
- результат отработки макрокоманды;
- оценка норма/ненорма (если оценка предусматривается командой).

Примеры таких строк протокола приведены ниже:

- *пример 1:*

320:0001:HA10:14ч37м51с397:АВТОНОМНАЯ
ОТЛАДКА ИПО КБА;

- *пример 2:*

305:0097: :14ч37м51с483: 00000000: 37577777: 02000040:
02000040: НК : н 15.

Возможно также представление в коротком формате, где перечисляются только номера программ и строк. Пример приведен ниже:

ИП 042 СТР 2167(37);

ИП 305 СТР 0099(44);

ИП 042 СТР 2169(37).

По представленным формам можно сделать предположение о пригодности протоколов для автоматизированного анализа.

Постановка задачи

Как уже отмечалось выше, одна из задач, решаемых во время опытно-конструкторской работы — это отработка документации. Для разработчиков автоматизированных проверок АКА представляет интерес автоматический анализ отработки ИПО по протоколам АИК. Для получения количественной оценки отработки ИПО необходимо разработать соответствующий инструментарий для разработчика-испытателя.

Решение с помощью объектно-ориентированной методологии

Поставленную задачу можно решать несколькими способами:

- путем создания структурированной программы на процедурном языке программирования;
- путем создания объектно-ориентированной программы.

В данной работе было принято решение использовать объектно-ориентированный подход [2], чтобы упростить декомпозицию задачи. В качестве среды разработки программы автоматизированного анализа ИПО применялся *Embarcadero C++ Builder© XE*.

Было выделено несколько типов сущностей:

- формы для взаимодействия с оператором программы;
- классы объектов, реализующих логику программы.

Среди классов выделим два наиболее существенных:

- класс, описывающий протоколы испытаний;
- класс, описывающий испытательные программы.

Класс для работы с протоколами имеет следующие составляющие:

- разреженный массив для хранения статистики отработанных программ и строк в них;
- переменные для хранения атрибутов протоколов;
- метод чтения данных из файлов;
- методы, оперирующие данными класса.

Класс для работы с испытательными программами содержит в себе следующие элементы:

- массив, содержащий изменения в программе;
- массив строк;
- методы чтения испытательных программ из файлов;
- методы анализа протоколов испытаний;
- массив статистической информации по отработке.

При выполнении анализа протоколов использовалась многопоточная модель программирования для решения следующих задач:

- расчет истории изменений испытательных программ;
- расчет отработки программы по протоколам;
- считывание протоколов в банк данных;
- считывание программ в банк данных;
- отрисовка отработки программ.

Для создания потоков в программе использовалась также объектная декомпозиция. От базового класса потока для каждой задачи из перечисленных выше путем наследования определялся производный отдельный класс с описанием исполнения потока.

Таким образом, с применением объектно-ориентированного подхода к решению задачи получения количественной оценки отработки ИПО был получен результат в виде программного обеспечения, наглядно демонстрирующего процесс отработки.

Оценка результатов

Для автоматизированной оценки отработки ИПО использовались данные, полученные в ходе автоматизированных электрических испытаний опытного АКА.

При расчете использовалось более 1 000 протоколов испытаний и более 100 формирований ИПО.

Были получены следующие количественные результаты:

- математическое ожидание отработки программы;
- дисперсия отработки программы;
- графики зависимостей, описывающие процесс отладки ИПО на АКА.

На рис. 3 изображена гистограмма, отражающая процент отработки программ в формировании.

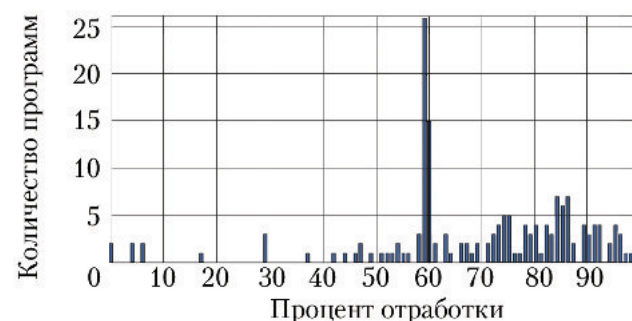


Рис. 3. Гистограмма среднего процента отработки испытательного программного обеспечения

Как видно из рисунка, наибольшее число программ отработано выше 50%. Для выяснения статистических значений, характеризующих случайную величину отработки испытательной программы в формировании, были рассчитаны основные величины.

Математическое ожидание [3] отработки ИПО рассчитывалось по формуле:

$$M[X] = \frac{\sum_{i=1}^n x_i}{n},$$

где x_i — это вычисленный процент отработки испытательной программы по протоколам АИК; n — число программ в последнем формировании.

Дисперсия вычислялась по формуле:

$$D[X] = \left[\frac{\sum_{i=1}^n x_i^2}{n} - m_x^2 \right] \frac{n}{n-1},$$

где m_x — математическое ожидание отработки ИПО.

Для рассматриваемого АКА эти величины оказались следующими:

$$M[X] = 0,698; D[X] = 0,041.$$

Из представленной гистограммы видно, что испытательные программы не отработываются полностью. Это связано с избыточностью кода, вызванной необходимостью обработки ошибок и несоответствий состояния АКА заданному. Исходя из этого, нельзя ожидать, что отработка ИПО во время испытаний АКА будет 100%.

Все испытательные программы различаются объемом кода в макрокомандах. Гистограмма, приведенная на рис. 3, этого аспекта не учитывает.

Рис. 4 содержит диаграмму отработки программы и объем кода. На ней каждая программа в формировании представлена точкой в пространстве, описывающем процент отработки и объема кода в испытательной программе.

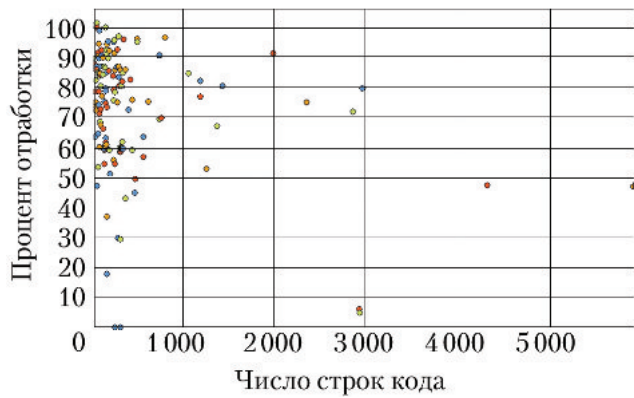


Рис. 4. Диаграмма зависимости средней отработки испытательного программного обеспечения от объема кода

Примечание. Цветные точки на диаграмме — обозначение разных испытательных программ.

Из рис. 4 видно, что основной вклад в ИПО в рассматриваемом АКА составляют небольшие испытательные программы с высокой степенью отработки. Вместе с этим, есть программы с невысоким процентом отработки большого объема. Этим фактором можно объяснить разрыв между средним процентом отработки испытательной программы в формировании и процентом отработки ИПО.

Процедура испытаний устроена таким образом, что частота отработки участков кода распределена неравномерно — чаще всего обрабатываются программы включения/отключения АКА, либо когда какие-то действия с АКА выполняются с определенной цикличностью. При этом интерес вызывает вопрос: с какой частотой обрабатывались строки в формировании? Фрагмент такой зависимости представлен на рис. 5.

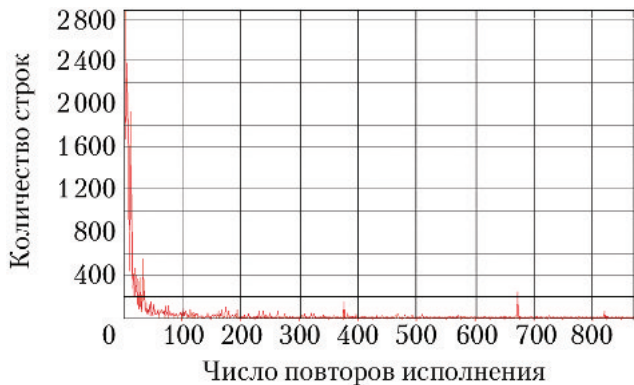


Рис. 5. Фрагмент графика количества строк, обрабатываемых n раз

На рис. 6 показан весь процесс отработки ИПО.

Верхняя линия на графике отображает рост среднего процента отработки испытательных программ в последнем формировании с учетом истории его изменения. Нижняя кривая отображает рост процента отработки всего ИПО.

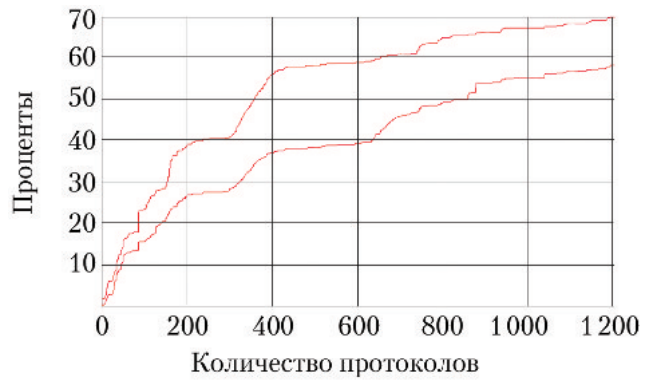


Рис. 6. Процесс отработки испытательного программного обеспечения

Выводы

Исследование процесса отработки ИПО в автоматизированном режиме позволило получить количественную оценку характеристик ИПО.

При оценке этого процесса можно сделать вывод, что он носит случайный характер — зависит как от уровня исправности объекта испытаний, в который входит АКА и сеть КПА, так и от ошибок в макрокомандах, неправильных действий операторов за пультом АИК. Все эти факторы влияют на объем обрабатываемого кода и число протоколов испытаний.

При рассмотрении процесса отработки ИПО можно сделать косвенный вывод об отработанности АКА по фактическому проценту отработки. Для разных типов АКА он будет отличаться в зависимости от заложенных алгоритмов проверки и избыточности кода.

С использованием разработанного программного обеспечения у разработчиков ИПО появилась возможность задавать целевые границы отработанности для оценки проведения испытаний или завершения отладки ИПО АКА.

Список литературы

1. Карасёв А.А. Проблема разработки испытательного программного обеспечения для новых бортовых комплексов дистанционного зондирования Земли в составе автоматического космического аппарата // Итоги диссертационных исследований. Т. 3. Материалы IV Всероссийского конкурса молодых ученых. М.: РАН, 2012. С. 3–10.
2. Буч Г. Объектно-ориентированный анализ и проектирование с примерами приложений на C++. 2-е изд. М.: Бинوم, 1998. 560 с.
3. Гмурман В.Е. Теория вероятностей и математическая статистика. Учеб. пособие для вузов. 9-е изд. М.: Высшая школа, 2003 г. 479 с.

Статья поступила в редакцию 04.12.2015 г.

Reference

1. Karasev A.A. *Problema razrabotki ispytatel'nogo programmno obespecheniya dlya novykh bortovykh kompleksov distantsionnogo zondirovaniya Zemli v sostave avtomaticheskogo kosmicheskogo apparata* [The problem of the development of the test software for new onboard systems of the Earth remote sensing as part of unmanned spacecraft]. *Itogi dissertatsionnykh issledovaniy. Materialy IV Vserossiiskogo konkursa molodykh uchenykh. Moscow, RAS publ., 2012. Vol. 3, pp. 3–10.*
2. Buch G. *Ob"ektno-orientirovanniy analiz i proektirovanie s primerami prilozhenii na C++ 2nd ed.* [The object-oriented analysis and design with examples of applications in C++. 2nd ed.]. *Moscow, Binom publ., 1998. 560 p.*
3. Gmurman V.E. *Teoriya veroyatnostei i matematicheskaya statistika Ucheb. posobie dlya vuzov. 9th ed.* [Theory of probability and mathematical statistics. Text-book for Institutions of Higher Education. 9th ed.]. *Moscow, Vysshaya shkola publ., 2003. 479 p.*