ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ УНИТАРНОЕ ПРЕДПРИЯТИЕ
ВСЕРОССИЙСКИЙ НАУЧНО-ИССЛЕДОВАТЕЛЬСКИЙ ИНСТИТУТ
МЕТРОЛОГИЧЕСКОЙ СЛУЖБЫ (ФГУП ВНИИМС)

РЕКОМЕНДАЦИЯ

ГОСУДАРСТВЕННАЯ СИСТЕМА
ОБЕСПЕЧЕНИЯ ЕДИНСТВА ИЗМЕРЕНИЙ

ТИПОВАЯ МЕТОДИКА АТТЕСТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ СРЕДСТВ ИЗМЕРЕНИЙ И ПОРЯДОК ЕЕ ПРОВЕДЕНИЯ

МИ 2955-2005

Москва
2005

gost.e-ecolog.ru

ПРЕДИСЛОВИЕ

РАЗРАБОТАНА Федеральным государственным унитарным предприятием «Всероссийский научно - исследовательский институт метрологической службы» (ФГУП ВНИИМС)

ИСПОЛНИТЕЛИ Ю.Е. Лукашов, к.т.н., (руководитель темы),
Ю.А. Кудеяров, д.ф.-м.н., А.А. Сатановский

УТВЕРЖДЕНА ФГУП ВНИИМС 17 ноября 2005 г.

ЗАРЕГИСТРИРОВАНА ФГУП ВНИИМС 21 ноября 2005 г.

ВВЕДЕНА ВПЕРВЫЕ

СОДЕРЖАНИЕ

1. Назначение и область применения……4
2. Нормативные ссылки……..4
3. Общие положения……..5
4. Порядок проведения аттестации программного обеспечения средств измерений…… ……5
5. Типовая методика аттестации программного обеспечения средств измерений ……..……7
5.1. Методика аттестации и ее основные этапы……..7
5.2. Проверка документации……10
5.3. Проверка структуры……...12
5.4. Проверка соответствия (идентификации)……13
5.5. Оценка погрешности программного обеспечения……..14
5.6. Проверка защиты программного обеспечения……19
Приложение 1. Форма заявки на проведение работ по аттестации ПО СИ…21
Приложение 2. Форма титульного листа методики аттестации ПО СИ……..22
Приложение 3. Рекомендуемая форма акта о результатах испытаний (тестирования) ПО СИ ……23
Приложение 4. Протокол аттестации ПО СИ……25
Приложение 5. Рекомендуемая форма свидетельства об аттестации ПО СИ ……28
Приложение 6. Коэффициент обусловленности (устойчивости) и исполнительная характеристика ПО СИ ……...29
Приложение 7. Метод генерации эталонных данных……33

Государственная система обеспечения
единства измерений

Типовая методика аттестации программного обеспечения средств измерений и порядок ее проведения

РЕКОМЕНДАЦИЯ
МИ 2955-2005

1. НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ

1.1. Настоящая рекомендация устанавливает порядок проведения и типовую методику аттестации программного обеспечения (ПО) средств измерений (СИ) и применяется при его аттестации на соответствие требованиям МИ 2891и другой нормативной документации, указанной в п. 2.

1.2. Рекомендация предназначена для разработчиков и пользователей ПО СИ, экспертов по аттестации ПО СИ, органов государственной метрологической службы, проводящих аттестацию программного обеспечения средств измерений, и государственных центров испытаний СИ.

1.3. Рекомендация распространяется на:

- ПО, являющееся частью измерительных систем, измерительно-вычислительных комплексов, информационно-управляющих систем, методик выполнения измерений и функционирующее на базе персонального компьютера (ПК);
- ПО, являющееся самостоятельным программным продуктом, который может применяться для сбора, обработки, хранения и представления измерительной информации;
- ПО, являющееся неотъемлемой частью СИ (встроенное ПО);
- ПО для контроллеров и вычислительных блоков.

2. НОРМАТИВНЫЕ ССЫЛКИ

ГОСТ Р 8.596-2002 ГСИ. Метрологическое обеспечение измерительных систем. Основные положения.

ГОСТ Р ИСО/МЭК 12119-2000. Информационная технология. Пакеты программ. Требования к качеству и тестирование.

ГОСТ Р ИСО 9127-94. Системы обработки информации. Документация пользователя и информация на упаковке для потребительских программных пакетов.

ПР 50.2.009-94 ГСИ. Порядок проведения испытаний и утверждения типа средств измерений;

МИ 2891-2004 ГСИ. Общие требования к программному обеспечению средств измерений;

МИ 2174-91 ГСИ. Аттестация алгоритмов и программ обработки данных при измерениях. Основные положения.

3. ОБЩИЕ ПОЛОЖЕНИЯ

3.1. Под аттестацией ПО СИ понимается его исследование (тестирование) с целью определения и/или оценки его характеристик и установления их соответствия предъявляемым к ним требованиям с последующей регистрацией полученных результатов исследования (тестирования) в свидетельстве об аттестации (с указанием использованных методов исследования).

3.2. К оцениваемым характеристикам ПО СИ относят:

- степень соответствия (низкая, средняя, высокая (п. 5.1.4.2));
- численные характеристики погрешности:
- относительная погрешность (п. 5.5.10.2);
- исполнительная характеристика (п. 5.5.10.3);
- уровень защищенности (низкий, средний, высокий);
- другие характеристики, согласованные между заказчиком аттестации и ее исполнителем.

3.3. При аттестации ПО СИ выполняются следующие виды испытаний:

- проверка документации;
- проверка соответствия утвержденному типу (идентификация);
- оценка погрешности ПО;
- проверка защищенности ПО и данных, а также другие виды испытаний, согласованные между заказчиком аттестации и ее исполнителем.

3.4. В случае, если разработчиками ПО выделена его часть, подлежащая метрологическому контролю (аттестации), проводится проверка правильности разделения ПО.

3.5. Термины и определения, использующиеся в настоящей рекомендации, те же, что и в методике МИ 2891.

4. ПОРЯДОК ПРОВЕДЕНИЯ АТТЕСТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ СРЕДСТВ ИЗМЕРЕНИЙ

4.1. Аттестацию ПО СИ проводят государственные центры испытаний средств измерений (ГЦИ СИ) и органы государственной метрологической службы, обладающие необходимыми условиями для проведения такой аттестации (п. 4.4).

4.2. При проведении аттестации ПО СИ используются методы определения и оценки его характеристик, основанные на международных и отечественных правилах и рекомендациях, которые позволяют с достаточной степенью достоверности установить их соответствие требованиям нормативной документации, указанной в п. 2, и определить действительные значения этих характеристик.

4.3. Порядок проведения аттестации предусматривает обеспечение конфиденциальности информации, составляющей коммерческую тайну.

4.4. Порядок проведения аттестации программного обеспечения включает:

- подачу заявки на аттестацию;
- принятие решения по заявке на аттестацию с учетом аппаратных, программных и кадровых возможностей для проведения такого рода работы, в том числе назначение экспертов (эксперта) на проведение работы по аттестации;
- оформление договора на проведение работ по аттестации;
- проведение исследования (тестирования) ПО СИ по согласованным с заказчиком аттестации методикам;
- принятие решения о выдаче свидетельства об аттестации;
- выдача свидетельства об аттестации;
- занесение юридического или физического лица и аттестованного ПО в Реестр аттестованного ПО СИ, который ведется ФГУП «ВНИИМС».

4.5. Юридическое или физическое лицо, разрабатывающее (распространяющее, использующее) ПО СИ и желающее его аттестовать, подает заявку в организацию, проводящую аттестацию, по форме, указанной в Приложении 1.

4.6. Документы, подаваемые на аттестацию, составляются в соответствии с требованиями п. 6.1 МИ 2891 и представляются как на бумажном носителе, так и на компакт-диске или на дискете в формате WORD для WINDOWS. Кроме того, в левом углу перечня документов указывается почтовый адрес и название юридического лица (ф.и.о. физического лица), осуществляющего разработку (распространение, использование) ПО СИ.

При высокой жесткости испытаний (п. 5.1.4.2) наряду с документацией, предусмотренной п. 6.1 МИ 2891, представляются также исходные коды ПО.

4.7. Заявка на аттестацию регистрируется организацией, проводящей аттестацию ПО, в журнале регистрации заявок, после чего экспертом организации производится анализ представленных с заявкой документов и принимается решение о проведении аттестации или об отказе в ее проведении.

Отказ в проведении аттестации возможен в случае, если представленные заявителем материалы показывают его неготовность к проведению аттестации ПО СИ.

4.8. В случае положительного решения оформляется договор на проведение работ по аттестации и определяется комиссия экспертов (эксперт) для проведения аттестации.

4.9. Эксперты, входящие в комиссию, проводят испытания (тестирование) ПО, представленного на аттестацию. Испытание (тестирование) ПО проводится по методикам аттестации, согласованным между организацией, осуществляющей аттестацию, и заявителем (п. 5. 1).

4.10. По решению руководителя организации, проводящей аттестацию, в состав комиссии по проведению аттестации могут входить как эксперты организации, проводящей аттестацию, так и эксперты других организаций.

4.11. В случае принятия решения об участии в работе комиссии по аттестации ПО экспертов других организаций в эти организации направляется письмо с просьбой обеспечить их участие в работе комиссии по аттестации и техническое задание на проведение аттестации, утвержденное руководителем организации, проводящей аттестацию.

4.12. Аттестация ПО может проводиться как непосредственно у заявителя или исполнителя, так и в любой другой организации в соответствии с договором между организацией, проводящей аттестацию, и заявителем.

4.13. В ходе аттестации комиссия экспертов (эксперт):

- проверяет достоверность сведений, изложенных в приложениях к заявке на аттестацию;
- проводит испытание (тестирование) ПО СИ по согласованным с заказчиком аттестации методикам (рекомендуемая форма титульного листа методики аттестации приведена в Приложении 2).

4.14. По результатам испытаний (тестирования) ПО комиссия экспертов (эксперт) составляет акт о результатах испытаний (тестирования) (Приложение 3), являющийся основанием для принятия решения о возможности выдачи (или не выдаче) свидетельства об аттестации ПО СИ.

Акт о результатах испытаний (тестирования), выводом которого является решение о возможности выдачи свидетельства об аттестации ПО СИ, может содержать рекомендации по совершенствованию ПО.

4.15. Акт о результатах испытаний (тестирования), подписанный членами экспертной комиссии (экспертом) и завизированный представителем разработчика (распространителя, пользователя), передается руководству организации, проводящей аттестацию, для оформления свидетельства об аттестации (форма свидетельства приведена в Приложении 5).

4.16. Если в результате испытаний (тестирования) ПО выявилось его несоответствие требованиям нормативной документации, приведенной в п. 2, свидетельство об аттестации не выдается. Заказчику аттестации в этом случае выдается акт о результатах испытаний (тестирования) ПО с обоснованием отказа в выдаче свидетельства об аттестации.

4.17. При возникновении разногласий между организацией, проводящей аттестацию, и заказчиком аттестации они решаются в установленном законодательством РФ порядке.

5. ТИПОВАЯ МЕТОДИКА АТТЕСТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ СРЕДСТВ ИЗМЕРЕНИЙ

5.1. МЕТОДИКА АТТЕСТАЦИИ И ЕЕ ОСНОВНЫЕ ЭТАПЫ

5.1.1. При проведении аттестации комиссией экспертов (экспертом) организации, проводящей аттестацию, составляется методика аттестации, содержащая детальное описание всех действий, выполняемых в процессе атте-

стации. При этом в методике аттестации должны быть предусмотрены следующие основные этапы:

- выбор методов испытаний (тестирования) в соответствии с целями испытаний и степенью их жесткости (п. 5.1.4.2);
- составление тестовых заданий в соответствии с требованиями к ПО и с указанием контролируемых параметров и характеристик, исходных данных и критериев, которым должны удовлетворять результаты, полученные тестируемым ПО;
- проведение испытаний (тестирования) и получение результатов функционирования испытываемого (тестируемого) ПО в соответствии с тестовыми заданиями;
- обработка результатов испытаний (тестирования) (сравнение результатов, полученных испытываемым (тестируемым) ПО, с эталонными).

5.1.2. Методика аттестации оформляется для каждого отдельного ПО СИ, с учетом его назначения, функциональных особенностей и применяемых методов испытаний (тестирования).

5.1.3. Испытания (тестирование) ПО СИ при выбранной жесткости могут проводиться на основе изучения документации, сопровождающей ПО, испытаний (тестирования) с использованием аппаратных и программных средств, а также на комплексной основе с использованием как документации, так и аппаратных и программных средств.

5.1.4. При выполнении этапов, указанных в п. 5.1.1, т.е. при составлении методики аттестации, рекомендуется придерживаться следующей последовательности действий:

5.1.4.1. По согласованию с разработчиком (заказчиком аттестации) устанавливается жесткость испытаний ПО СИ с учетом

- области применения;
- риска намеренных искажений результатов измерений;
- требуемой степени соответствия утвержденному типу;
- специфических задач аттестации.

5.1.4.2. При выбранной жесткости в программу испытаний рекомендуется включать следующие виды испытаний:

Низкая жесткость	Средняя жесткость	Высокая жесткость
1	2	3
1. Проверка документации.	1. Проверка документации.	1. Проверка документации.
2. Проверка разделения ПО и наличия защищенных интерфейсов (на основе анализа документации).	2. Проверка разделения ПО и наличия защищенных интерфейсов (на основе анализа документации и с использованием программных средств).	2. Проверка разделения ПО и наличия защищенных интерфейсов(с использованием программных средств, в том числе на основе анализа исходного кода).

3. Проверка соответствия утвержденному типу (на основе анализа документации).	3. Проверка соответствия утвержденному типу (идентификация) (на основе анализа документации и с использованием программных средств).	3. Проверка соответствия утвержденному типу (идентификация) (на основе анализа документации и с использованием программных средств, в том числе на основе анализа исходного кода).
4. Испытание (тестирование) функций ПО (с использованием аппаратных и программных средств).	4. Испытание (тестирование) ПО с целью установления правильности функционирования и определения его свойств и характеристик, в том числе тестирование обработки данных (с использованием аппаратных и программных средств).	4. Испытание (тестирование) ПО с целью установления правильности функционирования и определения его свойств и характеристик и тестирование обработки данных (с использованием аппаратных и программных средств, в том числе на основе анализа исходного кода).
5. Установление степени защищенности ПО и данных (на основе анализа документации).	5. Тестирование защищенности ПО и данных (с использованием программных средств).	5. Тестирование защищенности ПО и данных (с использованием программных средств, в том числе на основе анализа исходного кода).

Примечание - Высокая жесткость испытаний с использованием анализа исходного кода применяется в тех случаях, когда имеют дело с ПО сложных и ответственных измерительных систем, в том числе систем, используемых при коммерческих расчетах, или когда к таким системам предъявляются исключительные требования по безопасности их функционирования. В обычных случаях ограничиваются низкой и средней жесткостью испытаний, при которых тестирование ПО осуществляется методом «черного ящика».

5.1.4.3. Определяются методы испытаний в соответствии с установленной жесткостью испытаний, которые должны найти отражение в тестовых заданиях.

5.1.4.4. Составляются тестовые задания с учетом того, что эти задания и/или исходные данные для их формирования, а также методы их выполнения должны обеспечить проверку всех основных режимов функционирования испытываемого (тестируемого) ПО, а также его соответствие требованиям нормативной документации (п. 2).

5.1.4.5. Тестовые задания разрабатываются с учетом установленных видов испытаний (п.п. 3.3, 3.4 и 5.1.4.2), руководства пользователя, а также другой необходимой технической документации, представляемой на аттестацию. Каждому требованию и/или контролируемой функции должно соответствовать не менее одного тестового задания.

5.1.4.6. В тестовых заданиях определяются контролируемые свойства, параметры и характеристики ПО, методы испытаний тестируемого ПО, необ-

ходимые для этого исходные данные, программные продукты, с которыми сравниваются результаты, получаемые в процессе испытаний (эталонное ПО), а также критерии, позволяющие производить оценку характеристик тестируемого ПО.

5.1.4.7. Описывается последовательность действий при проведении процедуры испытаний (тестирования) в соответствии с установленной жесткостью испытаний и перечнем разработанных тестовых заданий.

5.1.5. Результаты испытаний ПО признаются положительными, если:

- программа соответствует функциональным возможностям, описанным в документации, и выполняет в рамках тестовых заданий все функции, декларируемые в документации;
- полученные значения параметров и характеристик программы при выполнении всех тестовых заданий удовлетворяют установленным критериям или находятся в допустимых пределах отклонений от них;
- в программе реализованы необходимые методы защиты и идентификации в соответствии с требованиями;
- программная документация соответствует требованиям нормативных документов (п. 2).

5.1.6. Если в процессе выполнения какого-либо из тестовых заданий произойдет отказ программы, ее «зависание» или искажение результатов, то данное тестовое задание может быть модифицировано для подтверждения ошибки функционирования и повторено. Выявленные ошибки фиксируются в протоколе испытаний.

5.1.7. По результатам испытаний ПО СИ составляется акт о результатах его исследования (тестирования) (п.п. 4.14, 4.15), неотъемлемой частью которого является протокол испытаний, подписанный непосредственными исполнителями испытаний (тестирования) (Приложение 4).

5.1.8. Дальнейшие разделы рекомендации конкретизируют виды испытаний (тестирования), приведенные в п. 5.1.4.2.

5.2. ПРОВЕРКА ДОКУМЕНТАЦИИ

5.2.1. Представление всей необходимой документации на аттестацию в соответствии с требованиями рекомендации МИ 2891, а также стандартов ГОСТ Р ИСО/МЭК 12119 и ГОСТ Р ИСО 9127, является необходимым условием ее проведения.

Рекомендуемые виды проверки документации применяются при всех степенях жесткости испытаний.

5.2.2. В соответствии с требованиями указанных нормативных документов проверяется наличие, достаточность и правильность представленной документации. При этом проверяются следующие разделы документации:

5.2.2.1. Описание структуры ПО и последовательности обработки данных.

Структура ПО может быть представлена в виде одного или нескольких взаимосвязанных модулей, реализующих функции ПО, с учетом разделения ПО. Описание структуры ПО может быть осуществлено в графическом виде с пояснениями и/или в текстовой форме.

Описание каждого модуля ПО должно включать:

- наименование модуля;
- функции, реализуемые модулем, и реализованные в модуле алгоритмы;
- данные, используемые и/или изменяемые модулем;
- названия связанных (вызываемых) модулей.

Проверяется описание всех входных и выходных данных, последовательность обработки входных данных в структуре ПО и других программно-аппаратных компонентах системы.

Данный раздел документации должен содержать информацию о методе связи (интерфейсе) СИ и ПО, о данных, получаемых от и передаваемых в СИ программным обеспечением, описание всех аппаратных и программных компонент СИ, а также описание исполняемых файлов (название, размер в мегабайтах).

5.2.2.2. Описание всех выполняемых функций и параметров ПО, с учетом выделения частей, подлежащих метрологическому контролю (аттестации) с внесением контролируемых функций и параметров в отдельный список.

Для всех функций ПО должны быть описаны входные и выходные данные, а также результат выполнения.

5.2.2.3. Описание реализованных в ПО расчетных алгоритмов, а также их блок-схемы.

Должны быть приведены описания или логические схемы алгоритмов, функции, реализуемые алгоритмами в ПО, а также все величины, рассчитываемые с их помощью, и их формулы, данные о степени округления при расчетах (точность алгоритмов).

5.2.2.4. Описание интерфейсов, меню, диалогов и перечень команд для каждого интерфейса, включая заявление об их полноте, а также список, значение и действие всех команд, получаемых от клавиатуры, мыши и других устройств ввода.

5.2.2.5. Описание реализованного метода идентификации ПО.

Проверяется наличие информации о методе (алгоритме) идентификации ПО, способах идентификации ПО в соответствии с реализованным методом, о системе кодификации номера версии.

5.2.2.6. Описание реализованных методов защиты ПО и данных.

Проверяется описание реализованных методов (авторизация пользователя, журнал событий, кодирование данных и т.д.) защиты ПО и данных от недопустимых изменений и искажений, а также наличие сообщений об ошибках.

Примечания

1 Некоторые требования к документации в МИ 2891-04 охватывают требования, приведенные в стандартах ГОСТ Р ИСО/МЭК 12119-2000, ГОСТ Р ИСО 9127-94. Поэтому при проверке соответствия документации указанным стандартам, проверяются те требования к документации ПО, которые не регламентируются МИ 2891-04.

2 Вся документация ПО проверяется на однозначность понимания и непротиворечивость.

5.2.3. Все данные, в том числе выявленные несоответствия, полученные при анализе документации ПО, заносятся в протоколы испытаний.

5.2.4. Перечень документов, сопровождающих ПО, объем и методы проверки документации могут корректироваться соглашением между исполнителем и заказчиком аттестации ПО.

5.3. ПРОВЕРКА СТРУКТУРЫ

Под проверкой структуры ПО понимается проверка правильности выделения его частей, подлежащих метрологическому контролю (аттестации), а также проверка наличия и правильности функционирования защищенных интерфейсов.

5.3.1. Проверка правильности разделения ПО.

5.3.1.1. При низкой и средней жесткости испытаний правильность разделения ПО проверяется посредством анализа документации, представленной заявителем аттестации, а также с помощью программных средств.

Все функции и параметры ПО, которые используются при обработке результатов измерений, влияют на них, или используются во вспомогательных функциях должны быть отнесены к части ПО, подлежащей метрологическому контролю (аттестации).

На основе документации устанавливается реализованный метод разделения ПО или отсутствие разделения.

5.3.1.2. Проверяются сведения о наличии защищенного интерфейса между частями ПО, подлежащими и не подлежащими метрологическому контролю (аттестации).

5.3.1.3. В случае отсутствия разделения, все ПО подлежит метрологическому контролю (аттестации), однако документация (см. п. 5.2.2.2) должна содержать перечень функций и параметров, относящихся к контролируемым. Этот перечень функций и параметров также подлежит проверке.

5.3.1.4. Для встроенного ПО его разделение и проверка разделения не проводятся.

5.3.1.5. При высокой жесткости испытаний правильность разделения ПО дополнительно проверяется при помощи анализа его исходного кода.

5.3.1.6. По исходному коду ПО проверяется правильность реализации защищенного интерфейса. Все модули, подпрограммы, процедуры, функции, параметры, переменные и т.д., участвующие в обработке информации должны быть охвачены защищенным интерфейсом.

5.3.1.7. По исходному коду рекомендуется составить структурную схему

ПО (граф), отображающую потоки данных в ПО и способы их управления. Это позволяет проверить поступление измерительной информации только в части ПО, подлежащие метрологическому контролю (аттестации), а также убедиться в возможности обращения к функциям ПО, к его параметрам и к данным, подлежащим метрологическому контролю (аттестации), только через защищенный интерфейс.

5.3.1.8. Сведения о разделении ПО заносятся в протокол испытаний. По результатам проверки может быть сделано замечание о неправильном разделении ПО и дана рекомендация о соответствующих коррективах.

5.3.2. Проверка интерфейса(ов) ПО.

5.3.2.1. Проверяется возможность недопустимого воздействия на ПО и данные через интерфейс(ы) ПО (СИ).

5.3.2.2. При низкой/средней жесткости, испытания, указанные в п. 5.3.2.1, проводятся на основе анализа документации и тестирования ПО с использованием программных средств.

5.3.3.3. С помощью программных средств выполняются все функции ПО, подлежащие метрологическому контролю (аттестации), и определяется их соответствие представленной документации.

Исследуются меню, подменю, диалоги ПО с целью выявления не соответствующих документации функций, параметров и команд.

5.3.3.4. Проверяется фильтрация ввода данных через интерфейс и возможность введения недопустимых данных.

Для этой проверки в ПО могут вводиться данные:

- из диапазона допустимых значений;
- из диапазона недопустимых значений;
- данные на границах ограничений диапазона ввода;
- комбинации вышеуказанных категорий данных.

5.3.3.5. При высокой жесткости испытания проводятся на основе анализа исходного кода ПО в соответствии с п. 5.3.1.5. При этом:

- выявляются недокументированные функции, процедуры, параметры, переменные, команды и их использование в ПО;

- проверяется соответствие действия вводимых команд (с помощью диалогов, командной строки и т.д.) представленной документации.

5.3.3.6. Возможно использование других методов проверки правильности функционирования интерфейса(ов) ПО.

5.4. ПРОВЕРКА СООТВЕТСТВИЯ (ИДЕНТИФИКАЦИЯ)

Под проверкой соответствия понимается проверка конкретного ПО на соответствие тому типу, который был утвержден (зафиксирован) при его аттестации, если такая аттестация ранее проводилась, а также проверка методов и способов идентификации.

5.4.1. Проверка соответствия.

5.4.1.1. При низкой и средней жесткости испытаний проверка соответствия утвержденному типу проводится на основе анализа представленной до-

кументации, а также с использованием программных средств. При этом может приниматься во внимание заявление заказчика аттестации об отсутствии изменений и модификаций части ПО, подлежащей метрологическому контролю (аттестации).

5.4.1.2. В случаях, когда имеются сомнения в полном соответствии конкретного ПО утвержденному типу, может быть проведено выборочное тестирование тех функций и частей ПО, относительно которых имеются указанные сомнения.

5.4.2. Проверка методов и способов идентификации программного обеспечения.

5.4.2.1. При низкой и средней жесткости испытаний методы и средства идентификации ПО проверяются на основе документации и выполнения функциональных испытаний (тестирования) ПО.

5.4.2.2. На основе документации проверяется описание реализованного метода идентификации и способа работы с ним. В случае разделения ПО на программном уровне устанавливается соответствие метода идентификации принципу разделения ПО.

5.4.2.3. На основе анализа документации получают информацию обо всех частях (модулях) ПО (с учетом его разделения), охватываемых идентификацией. Все части ПО, подлежащие метрологическому контролю (аттестации) должны быть охвачены алгоритмом идентификации.

5.4.2.4. Оценивается соответствие метода идентификации уровню требуемой защиты. В качестве метода идентификации ПО может быть рекомендован номер версии с учетом п. 6.3.2.1. МИ 2891.

5.4.2.5. Проверяется наличие номинального значения, идентифицирующего ПО (номер версии, контрольная сумма и др.). Номинальное значение идентификации ПО вносится в свидетельство об аттестации.

5.4.2.6. Проверяется функция идентификации ПО, например, выводом номера версии на экран.

5.4.2.7. При высокой жесткости испытаний проводятся дополнительные испытания на основе анализа исходного кода ПО. При этом устанавливается:

- охват идентификацией всех частей ПО, подлежащих метрологическому контролю (аттестации), например, формирование номинального закрепляемого значения идентификации;
- охват идентификацией частей ПО, не подлежащих метрологическому контролю (аттестации);
- соответствие метода идентификации представленной документации.

Примечание - В случае изменения файлов, входящих в состав ПО и охваченных алгоритмом идентификации, изменение значения, идентифицирующего ПО, устанавливается при испытаниях по проверке защиты от искажений ПО и данных.

5.5. ОЦЕНКА ПОГРЕШНОСТИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

5.5.1. Погрешность ПО представляет собой отличие результатов, полученных с помощью испытываемого (тестируемого) ПО, от результатов, по-

лученных при тех же условиях эталонным ПО. При этом под эталонным понимается ПО, отвечающее наивысшим требованиям к его точностным и функциональным характеристикам, подтвержденным (в ряде случаев независимыми методами) при его неоднократном тестировании.

5.5.2. Критерии отнесения ПО к эталонному условны и при проведении аттестации ПО СИ являются предметом соглашения между заказчиками аттестации и ее исполнителями. В качестве эталонного может быть использовано аттестованное ПО.

5.5.3. К основным источниками погрешностей ПО относятся:

- программные ошибки вследствие неправильной записи исходного текста программы на выбранном языке программирования, а также ошибки трансляции программы в объектный код;
- алгоритмические ошибки, связанные с неполной или ошибочной формулировкой необходимых условий решения;
- применение неустойчивых алгоритмов при решении плохо обусловленных* измерительных задач;
- ошибки программного преобразования входных данных перед обработкой и ошибки обратной процедуры, если она проводится;
- ошибки округления и др.

5.5.4. Порядок проведения испытаний (тестирования) ПО определяется тестовым заданием (п.п. 5.1.4.4 – 5.1.4.7) и может включать в себя:

- анализ ПО и его алгоритмов, выбор контролируемых параметров, характеристик и свойств;
- определение методов испытаний (тестирования) в соответствии с выбранной жесткостью испытаний;
- определение критериев оценки погрешности, характеристик и параметров ПО;
- выбор (или разработка) эталонного ПО;
- выбор (определение) исходных данных и/или их получение методом генерации или какими-либо другими методами;
- получение результатов обработки исходных данных в испытываемом (тестируемом) ПО (получение тестовых результатов);
- получение оценки погрешности ПО посредством обработки результатов испытаний (тестирования) (сравнения тестовых результатов с эталонными);
- дополнительные исследования свойств, параметров и характеристик используемых алгоритмов (адекватность измерительной задаче, область устойчивости, время, затрачиваемое на обработку результатов измерений и т.п.).

5.5.5. Основными методами, применяемыми при оценке погрешности ПО, являются:

---

* Обусловленность является мерой чувствительности решения к изменениям входных данных. Хорошо обусловленная задача обладает свойством малого изменения решения при малом же изменении входных данных, в то время как плохо обусловленная задача при тех же условиях повлечет за собой существенные изменения решения.

- сравнительные испытания с применением эталонного (аттестованного) ПО;
- в отсутствие эталонного (аттестованного) ПО сравнительные испытания с использованием моделей исходных данных или с применением метода генерации эталонных данных (Приложение 7);
- испытания на основе исходного кода ПО, а также комбинации указанных методов.

Метод оценки погрешности ПО выбирают с учетом наличия или возможности разработки того или иного вида эталонного ПО, а также возможности применения указанных методов в каждом конкретном случае.

При низкой/средней жесткости, испытания проводятся с применением одного из методов сравнительных испытаний.

При высокой жесткости испытания проводятся на основе исходного кода ПО.

5.5.6. Сравнительные испытания с применением эталонного (аттестованного) ПО.

5.5.6.1. Данный метод испытания применяется при наличии эталонного (аттестованного) ПО, с помощью которого могут быть идентично воспроизведены функции испытываемого (тестируемого) ПО. В качестве эталонного ПО при оценке погрешности ПО может быть применено:

- аттестованное ПО СИ аналогичного функционального назначения;
- специально разработанное ПО с функциями, идентичными тестируемому;
- стандартные пакеты коммерческого ПО (например, электронные таблицы, пакеты математического и статистического ПО и т.д.).

5.5.6.2. К разработке эталонного ПО прибегают в тех случаях, когда испытываемое (тестируемое) ПО является не очень сложным, а его алгоритмы достаточно просты. Это означает, что затраты на разработку эталонного ПО должны быть разумными и сопоставимыми со стоимостью работ по аттестации ПО. Данный метод позволяет максимально учитывать особенности аттестуемого ПО, а также метрологические характеристики соответствующего СИ, и может быть рекомендован, в частности, при аттестации встроенного ПО.

5.5.6.3. Разрабатываемое эталонное ПО может содержать только функции и параметры, подлежащие метрологическому контролю (аттестации). При этом в некоторых случаях могут не учитываться особенности графического интерфейса пользователя, а также функции, не участвующие в обработке результатов измерений (например, функции отображения, хранения данных и т.д.).

5.5.6.4. К разработке эталонного ПО могут привлекаться специалисты организаций, занимающихся разработкой программных средств.

5.5.7. Сравнительные испытания с использованием моделей исходных данных.

5.5.7.1. Метод аттестации с использованием моделей исходных данных рекомендуется методикой МИ 2174 для аттестации алгоритмов обработки результатов измерений. Метод позволяет оценить погрешность алгоритмов сравнением результатов обработки тестируемыми алгоритмами заданных моделей исходных данных с параметрами этих моделей.

5.5.7.2. Наборы моделей исходных данных выбираются таким образом, чтобы они соответствовали частной измерительной задаче, решаемой тестируемыми алгоритмами.

5.5.8. Генерация эталонных наборов данных

5.5.8.1. Метод генерации эталонных наборов данных применяется как альтернатива использованию эталонного ПО (в случае его отсутствия или невозможности использования) при оценке отдельных функций, реализуемых тестируемым ПО, и, по сути, имитирует случайный характер измерительного процесса.

5.5.8.2. Эталонные данные получают путем генерации таких данных с помощью специально разработанной программы – генератора эталонных данных, который представляет собой алгоритм, предназначенный для моделирования эталонных данных на основе выбранных (заданных) исходных данных.

Генератор эталонных данных реализуют на одном из алгоритмических языков программирования или при помощи стандартного математического или статистического программного пакета (Приложение 7).

5.5.8.3. Исходные данные для испытаний, в том числе и для генерации эталонных данных, формируются с учетом свойств реализованных алгоритмов. При этом наборы исходных данных должны охватывать как можно больший диапазон возможных значений, поступающих на обработку.

5.5.8.4. В наборы исходных данных могут быть включены:

- данные, близкие к наибольшим и наименьшим значениям, а также ряд промежуточных значений;

- особые значения входных переменных - точки резкого возрастания или разрыва производных, нулевые, единичные и предельно малые численные значения переменных и т.п.

5.5.8.5. Если значения некоторой переменной зависят от значения другой переменной, то испытания проводят при особых сочетаниях этих переменных таких, как равенство обеих переменных, малое и предельно большое их различие, нулевые и единичные значения и т.п.

5.5.9. Испытания на основе исходного кода ПО

5.5.9.1. При испытаниях на основе исходного кода ПО проверяется:

- соответствие структуры алгоритма представленной документации;

- запись алгоритма на выбранном языке программирования;

- адекватность выбранного алгоритма измерительной задаче (выявление неустойчивых алгоритмов).

5.5.9.2. При проверке соответствия структуры алгоритма представленной документации, по тексту программы может быть составлена блок-схема реализуемого алгоритма, которая сравнивается с алгоритмом, изложенным в

документации. В случае нахождения различий в структуре алгоритмов проводится дополнительный анализ элементов блок-схемы, в которых обнаружены различия.

5.5.9.3. Проверяется правильность записи алгоритма на выбранном языке программирования. При этом устанавливается соответствие кода правилам программирования, наличие тупиков и висящих вершин, пустых переменных, операторов, правильность организации циклов и т.д.

5.5.9.3. Соответствие выбранного алгоритма измерительной задаче осуществляется путем математического анализа реализованного алгоритма обработки данных. При этом исследуются логические и точностные характеристики реализованных алгоритмов, анализируется пригодность и оптимальность примененных численных методов к решению измерительной задачи.

5.5.9.4. Методы оценки погрешности ПО на основе исходного кода могут корректироваться исполнителем и заказчиком аттестации ПО.

5.5.10. Обработка результатов испытаний (тестирования) ПО.

5.5.10.1. На основе полученных результатов испытаний (тестирования) рассчитывают характеристики точности тестируемых алгоритмов: относительную погрешность алгоритма и его исполнительную характеристику.

Могут быть оценены и другие характеристики алгоритмов такие, как их сложность, скорость исполнения, адекватность измерительной задаче, выбор численной схемы расчета, коэффициент обусловленности (устойчивости), область устойчивости и т.п.

5.5.10.2. Относительная погрешность алгоритма.

Вычисление относительной погрешности алгоритма производится по формуле:

$$\delta = \frac{|Y_{тест} - Y_{эт}|}{Y_{эт}}, \qquad (1)$$

где $Y_{тест}$ - результат, полученный испытываемым ПО (результат обработки),

$Y_{эт}$ - соответствующее значение, полученное эталонным ПО (эталонный результат).

5.5.10.3. Исполнительная характеристика алгоритма (Приложение 6).

Исполнительная характеристика алгоритма вычисляется по формуле:

$$P(\vec{x}) = \lg(1 + \frac{1}{k(\vec{x})\eta} \cdot \frac{\|\Delta\vec{y}\|}{\|\vec{y}^{(эт)}\|}), \qquad (2)$$

где $k(\vec{x})$ - коэффициент обусловленности (устойчивости),

η - относительная вычислительная точность,

$\Delta\vec{y}$ - отличие тестируемых результатов от эталонных,

$\|\vec{y}^{(эт)}\|$ - норма (длина) вектора эталонных результатов (Приложение 6).

Исполнительная характеристика показывает число потерянных цифр точности в тестируемом программном обеспечении по сравнению с эталонным.

5.5.10.4. Перечень характеристик аттестуемого ПО может корректироваться соглашением между исполнителем и заказчиком аттестации ПО.

5.5.10.5. Все определенные и оцененные характеристики и свойства алгоритмов заносятся в протокол испытаний.

5.5.10.6. Критерии, которым должны удовлетворять определенные и оцененные характеристики алгоритмов ПО, а также допускаемые значения характеристик могут быть установлены на основе требований к точности решения измерительной задачи, если они имеются, точности выполняемых расчетов (степени округления) и.т.п. Критерии и допуски на значения характеристик фиксируются в тестовых заданиях и согласовываются в рамках методики аттестации ПО с ее заказчиком.

5.6. ПРОВЕРКА ЗАЩИТЫ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

5.6.1. Проверка защиты от сбоев.

5.6.1.1. К сбоям в работе ПО относятся ситуации, приводящие к прекращению выполнения основных функций ПО на длительное или неопределенное время, нарушению нормального хода вычислительного процесса, стиранию или искажению данных в оперативной или долговременной памяти и т.п.

5.6.1.2. Целью испытаний является проверка средств защиты ПО от сбоев, от использования вследствие сбоя искаженных программ и данных, а также средств восстановления.

5.6.1.3. При сбое может предусматриваться запись события о сбое в программном журнале событий, в который записывается информация о времени сбоя, а также о результатах срабатывания средств восстановления.

При данных испытаниях может применяться расчет контрольных сумм по коду программы и данным.

5.6.1.4. При низкой/средней жесткости испытания проводятся на основе анализа документации и функциональных испытаний. При высокой жесткости испытаний проводится проверка реализации средств защиты и восстановления от сбоев на основе исходного кода ПО.

5.6.1.5. При испытаниях проверяется функционирование средств программного контроля и оперативного восстановления ПО. С этой целью разрабатываются сценарии имитации программно-аппаратных сбоев, искажений исходных данных и программных ошибок. При этом могут имитироваться следующие ситуации:

- параллельный запуск нескольких экземпляров приложения;
- досрочное завершение работающего ПО, вызванное, например, отключением питания компьютера, нарушением соединения компьютера с СИ и т.п.;
- тестирование при больших объемах данных.

При реализации каждой из этих ситуаций фиксируется результат имитации сбойной ситуации, проверяется работа программного журнала событий (наличие записи о сбое), выполняется идентификация ПО, проверятся целостность ПО и данных (например, с помощью расчета контрольных сумм).

5.6.1.6. При высокой жесткости испытаний на основе исходного кода ПО проверяются средства защиты и восстановления от сбоев. При этом устанавливается правильность расчета и сверки контрольных сумм по коду и данным.

5.6.1.7. Могут применяться и другие методы проверки защиты ПО от сбоев.

5.6.2. Защита программного обеспечения и данных от изменений.

5.6.2.1. На основе анализа документации проверяются наличие и достаточность средств защиты ПО и данных от изменений.

Средства защиты могут включать в себя:

- запрос авторизации (допуска) у пользователя;
- кодирование или электронную подпись (контрольную сумму) файлов с данными (настройками, параметрами), контрольные суммы по другим файлам ПО, относящимся к частям, подлежащим метрологическому контролю (аттестации);
- программный журнал событий.

5.6.2.2. При данной проверке могут выполняться следующие испытания:

а) Проверка авторизации пользователя:

- проверяется возможность обхода авторизации;
- проверяется реакция ПО на неоднократный ввод неправильной информации учетной записи пользователя;
- проверяются функции ввода новых пользователей и назначения их прав.

б) Проверка программного журнала внесения изменений:

- проверяется появление соответствующей записи в журнале изменений после внесения изменений в настройки ПО;
- проверяется возможность правки информации журнала изменений с помощью, например, текстового редактора*.

в) Проверка защиты данных и ПО.

Проверяется работа средств защиты ПО и данных от искажений после внесения, например с помощью текстового редактора, изменений в различные файлы ПО и файлы данных.

5.6.2.3. Могут применяться и другие методы проверки защиты ПО и данных от изменений.

Приложение 1

* В соответствии с требованиями МИ 2891 журнал изменений должен быть защищен от изменений.

Форма заявки на проведение работ по аттестации ПО СИ

Руководителю организации,
проводящей аттестацию ПО СИ
от

_____

(наименование предприятия-заявителя)

З А Я В К А

Прошу провести аттестацию _____
_____,
название программного продукта
разработанного (распространяемого, используемого) нашей организацией.

Перечень документов, предъявляемых на аттестацию, подготовленный в соответствии с требованиями п. 6.1 МИ 2891, указан в Приложении.

Обязуемся оплатить все расходы, связанные с процедурой аттестации.

Приложение: 1. ПО, подлежащее аттестации.
2. Перечень документации ПО, представляемого на аттестацию.

Руководитель организации-заявителя _____ _____
подпись
Ф.И.О.

Главный бухгалтер _____ _____
подпись
Ф.И.О.

М.П.

«____» _____ 200 __ г.

Приложение 2

Форма титульного листа методики аттестации ПО СИ

_____

_____

(название организации, проводящей аттестацию ПО СИ)

СОГЛАСОВАНО

Руководитель организации – заявителя

_____

«____» _____ 200 __ г.

УТВЕРЖДАЮ

Руководитель организации, проводящей аттестацию ПО СИ

_____

«___» _____ 200 __ г.

МЕТОДИКА АТТЕСТАЦИИ

_____,

(название программного обеспечения)

представленного на аттестацию _____

(название организации-заявителя)

Город, где проводится аттестация

200 __ г.

Методика аттестации после изложения содержательной части подписывается разработчиками - комиссией экспертов (экспертом) организации, проводящей аттестацию ПО СИ.

Приложение 3

Рекомендуемая форма акта о результатах испытаний (тестирования) ПО СИ

А К Т

испытаний (тестирования)_____ ,
(наименование программного обеспечения)
представленного _____.
(наименование организации-заявителя)

1. Комиссия экспертов (эксперт) _____
(наименование организации, проводившей испытания)
провела(ел) испытания (тестирование)_____,
(наименование программного обеспечения)
разработанного _____
(наименование организации-разработчика (если это возможно установить))
Испытания (тестирование) проведены в период с «__» по «__» _____ 200__г.
на основании_____ .
(номер договора с заявителем аттестации)
в _____.
(место проведения испытаний)

2. Организации, проводившей аттестацию, были представлены образцы ПО СИ _____
(название программного обеспечения и его назначение)

3. Ознакомившись с предъявленными образцами ПО и рассмотрев сопровождающую документацию, комиссия экспертов (эксперт) _____признала
(название организации, проводящей аттестацию ПО СИ)
предъявленные материалы достаточными для проведения испытаний (тестирования).

4. Комиссия экспертов (эксперт) _____провела(ел)
(название организации, проводящей аттестацию ПО СИ)
испытания (тестирование) _____
(наименование программного обеспечения)
в соответствии с методикой аттестации, утвержденной руководителем
_____
(название организации, проводящей аттестацию ПО СИ)
«____» _____ 200__ г.

5. В результате проведенных испытаний (тестирования) на основании представленных протоколов испытаний, являющихся неотъемлемой частью данного акта, было установлено, что _____
(наименование программного обеспечения)
обладает следующими основными характеристиками (перечисляются основные характеристики ПО, установленные в ходе испытаний).

6. В процессе испытаний (тестирования) отмечены следующие недостатки:

_____.

(перечень отмеченных недостатков, в т.ч. по документам, представленным на испытание)

7. На основании результатов проведенных испытаний (тестирования), отраженных в протоколах испытаний, комиссия экспертов (эксперт) рекомендует:

_____

(рекомендация о выдаче (не выдаче) свидетельства об аттестации)

(В случае рекомендации о невыдаче свидетельства об аттестации ПО должно быть приведено обоснование такой рекомендации)

Акт составлен в 2-х экземплярах _____ _____

подписи(сь) членов комиссии экспертов (эксперта) инициалы, фамилия

С актом ознакомлен.

Руководитель организации-заявителя

_____ _____

подпись инициалы, фамилия

«___»_____ 200__ г.

Приложение 4

Протокол аттестации программного обеспечения средств измерений

Название программы _____

Назначение программы _____

_____

Версия / Идентификационный номер _____

Организация разработчик _____

1. Документация				
Требование		Присутствует	Отсутствует	Примечание
1.1	Структура ПО, последовательность обработки данных (текст, граф)			
1.1.1	Описание связи ПО с аппаратной частью (СИ)			
1.2	Описание всех функций, параметров ПО			
1.2.1	Описание всех законодательно контролируемых функций и параметров			
1.3	Описание расчетных алгоритмов ПО			
1.4	Описание всех меню, диалогов, перечень команд для каждого интерфейса			
1.5	Описание метода идентификации ПО			
1.6	Описание методов защиты ПО и данных от недопустимых изменений (искажений)			
1.6.1	Сообщения об ошибках и отказах, корректирующие и восстанавливающие действия			
1.7	Документация в целом соответствуют стандартам ГОСТ Р ИСО 9127 и ГОСТ Р ИСО/МЭК 12119			
1.8	Представлен исходный код ПО			
Заключение: _____				

Требование		Соответствует	Не соответствует	Примечание
2. Структура				
2.1	Разделение ПО			
2.1.1	Разделение ПО выполнено			

2.1.2	Метод разделения			
2.1.3	Защитный интерфейс реализован			
2.1.4	Все контролируемые функции и параметры охвачены защитным интерфейсом			
2.1.5	Отсутствует открытый доступ к контролируемым функциям и параметрам из неконтролируемых частей			
2.2	Интерфейс ПО			
2.2.1	Интерфейс ПО (меню, диалоги, вводимые параметры) соответствует документации			
2.2.2	Осуществляется фильтрация ввода			
2.2.3	Отсутствуют недокументированные возможности (по исходному коду)			
2.2.4	Действие вводимых команд соответствует спецификации			
Заключение:				

Требование		Выполняется	Не выполняется	Примечание
3. Соответствие				
3.1.	Соответствие ПО версии, установленной (зафиксированной) при испытаниях с целью утверждения типа			
3.2	Осуществляется идентификация ПО			
3.3	Функция идентификации может быть вызвана по команде пользователя			
3.4	Метод идентификации			
3.5	Учет при идентификации разделения ПО			
3.6	Все законодательно контролируемые части ПО охвачены идентификацией			
3.7	Номинальное значение идентификации			
Заключение:				

4. Погрешность ПО

4.1 Аттестация проводились методом _____

4.2. Определены следующие характеристики алгоритмов:

Алгоритм вычислений	Относительная погрешность, δ	Исполнительная характеристика, *P(x)*

Примечание - Результаты расчетов, с указанием диапазонов испытаний приводятся в приложении к протоколу.

4.3 Заключение: _____

5. Защита

Требование		Соответствует	Не соответствует	Примечание
5.1 Защита ПО и данных от сбоев				
5.1.1	ПО содержит средства защиты и восстановления от сбоев			
5.1.2	Реализован журнал событий			

5.1.3 Выполнены следующие испытания защиты и восстановления ПО от сбоев

№	Имитация сбоя	Результат	Примечание

Требование		Соответствует	Не соответствует	Примечание
5.2 Защита ПО и данных от искажений				
5.2.1	Осуществляется авторизация пользователя при запуске			
5.2.2	Осуществляется защита ПО и данных (контрольная сумма, кодирование)			

5.2.2 Выполнены следующие испытания защищенности ПО и данных от искажений

№	Испытание	Результат	Примечание

5.3 Заключение: _____

Заключение по результатам аттестации: _____

Исполнители (должности, фамилии, инициалы, подписи):

Дата проведения испытаний:

Приложение 5

Рекомендуемая форма свидетельства об аттестации ПО СИ

_____

(название организации, проводящей аттестацию ПО СИ)

СВИДЕТЕЛЬСТВО № ___

(регистрационный номер в Реестре аттестованного ПО СИ)

об аттестации программного обеспечения средств измерений

Название программы: _____

_____

Назначение и область применения программы: _____

_____

Организация - разработчик программы

_____

Результаты аттестации:

Название и обозначение характеристики:_____

Метод испытаний (тестирования) _____

Полученное значение характеристики: _____

По результатам аттестации (акт аттестационных испытаний от «___» _____ 200___ г.)

программа _____

допускается к применению с указанными в настоящем свидетельстве характеристиками.

Дата выдачи свидетельства: _____ _____ 200___ г.

Руководитель организации, проводившей аттестацию

_____ _____

подпись инициалы, фамилия

М.П.

_____

Результаты аттестации приводятся для каждой характеристики.

Допускается приводить результаты в виде сводной таблицы, содержащей указанные сведения.

Возможно оформление таблицы на отдельном листе или на обороте свидетельства.

Коэффициент обусловленности (устойчивости) и исполнительная характеристика ПО СИ

Содержание этого Приложения основано на разработках Национальной физической лаборатории (Великобритания).

Предположим, что связь между входом $\vec{x}$ и выходом $\vec{y}$ измерительной системы описываются математической моделью

$$\vec{y} = \vec{f}(\vec{x}), \tag{6.1}$$

где $\vec{f}$ обозначает векторную функциональную зависимость между входом и выходом. Эта зависимость может быть разного вида сложности, в частности, она может быть и неявной.

Пусть $\delta\vec{x}$ обозначает малое возмущение входного воздействия, а $\delta\vec{y}$ - соответствующую реакцию выхода. Тогда *коэффициент обусловленности* модели (коэффициент устойчивости) определяется формулой

$$k(\vec{x}) = \frac{\|\delta\vec{y}\|}{\|\vec{y}\|} \Big/ \frac{\|\delta\vec{x}\|}{\|\vec{x}\|}, \tag{6.2}$$

где символ $\|\vec{a}\|$ обозначает *норму* (длину) вектора $\vec{a}$.

Под нормой (длиной) вектора $\vec{a}$ понимают неотрицательное действительное число $\|\vec{a}\| = \sqrt{(\vec{a}\cdot\vec{a})}$. В скобках стоит скалярное произведение вектора $\vec{a}$ самого на себя. В частности, если вектор проведен из начала координат в точку с координатами (a_1, a_2, a_3), то норма такого вектора может быть вычислена по формуле $\|a\| = \sqrt{a_1^2 + a_2^2 + a_3^2}$. Известно, что норма вектора наряду с другими свойствами удовлетворяет неравенству Коши - Буняковского $\left|(\vec{a}\cdot\vec{b})\right| \le \|\vec{a}\|\cdot\|\vec{b}\|$.

Коэффициент обусловленности (устойчивости) характеризует относительное изменение выхода системы по отношению к соответствующему относительному изменению входа.

Приращение функции (6.1) с точностью до членов первого порядка малости записывается в виде

$$\delta\vec{y} = \vec{f}(\vec{x} + \delta\vec{x}) - \vec{f}(\vec{x}) \approx J(\vec{x})\cdot\delta\vec{x}, \tag{6.3}$$

где *функциональная матрица* $J(\vec{x})$ определяется формулой

$$J(\vec{x}) = \left(\frac{\partial f_i}{\partial x_k}\right)\Big|_{P_0} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} \cdots & \frac{\partial f_m}{\partial x_1} \\ \frac{\partial f_1}{\partial x_2} \cdots & \frac{\partial f_m}{\partial x_2} \\ \cdots\cdots & \cdots\cdots \\ \frac{\partial f_1}{\partial x_n} \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix}_{P_0}. \tag{6.4}$$

В случае, когда $n = m$ определитель этой матрицы $J = \det(\frac{\partial f_i}{\partial x_k})$ называется *якобианом* функции $\vec{f}$.

С учетом (6.3) формула (6.2) может быть записана в виде

$$k(\vec{x}) = \frac{\frac{\|J(\vec{x})\cdot\delta\vec{x}\|}{\|\vec{y}\|}}{\frac{\|\delta\vec{x}\|}{\|\vec{x}\|}} \le \frac{\|J(\vec{x})\|\cdot\|\delta\vec{x}\|}{\|\vec{y}\|} \Big/ \frac{\|\delta\vec{x}\|}{\|\vec{x}\|} = \frac{\|J(\vec{x})\|\cdot\|\vec{x}\|}{\|\vec{y}\|}. \qquad (6.5)$$

При записи (6.5) было использовано неравенство Коши – Буняковского.

Пусть теперь $\vec{y}^{(эт)}$ обозначает «эталонный» отклик, соответствующий входному воздействию $\vec{x}$ (отклик на выходе эталонного ПО при воздействии входных данных $\vec{x}$). Предполагается, что эталонное ПО реализует правильное исполнение оптимально устойчивого алгоритма. Для устойчивых алгоритмов коэффициент обусловленности принимает значение порядка единицы. Напомним, что алгоритм называется *устойчивым*, если для любого $\varepsilon > 0$ существует $\delta > 0$ такое, что при погрешности входных данных, меньшей δ, максимальная погрешность выходных данных будет меньше ε.

Далее предполагается, что $\vec{y}^{(тест)}$ – отклик, производимый тестируемым ПО, и пусть

$$\Delta\vec{y} = \vec{y}^{(тест)} - \vec{y}^{(эт)}.$$

Из (6.2) следует

$$\|\Delta\vec{x}\| = \frac{\|\Delta\vec{y}\|}{\|\vec{y}^{(r)}\|}\cdot\frac{\|\vec{x}\|}{k(\vec{x})}, \qquad (6.6)$$

где $\Delta\vec{x}$ - возмущение (погрешность) входных данных.

Из определения устойчивого алгоритма следует, что при оценке его устойчивости необходимо определить значение отклика $\Delta\vec{y}$ на минимальное изменение внешнего воздействия $\Delta\vec{x}$, например, такое, когда $\|\Delta\vec{x}\| \cong \eta\cdot\|\vec{x}\|$, где η - относительная вычислительная точность, которая в большинстве случаев порядка 10^{-16}. Другими словами, «качество» алгоритма (по отношению к проблеме обусловленности) количественно может быть оценено его реакцией $\Delta\vec{y}$ на минимальное изменение входных величин. Такое изменение может, например, возникать при преобразовании входных данных от десятичной системы в двоичную, и наоборот.

Для эталонного ПО можно ожидать, что если

$$\frac{\|\Delta\vec{x}\|}{\eta\cdot\|\vec{x}\|} = O(1), \qquad (6.7)$$

то, как следует из (6.6),

$$\frac{1}{k(\vec{x})\cdot\eta}\cdot\frac{\|\Delta\vec{y}\|}{\|\vec{y}^{(эт)}\|}=O(1), \tag{6.8}$$

где $O(1)$ – величина порядка единицы.

Если же для тестируемого ПО отношение в левой части формулы (6.8) оказываются порядка $O(10^p)$ при том, что входные данные удовлетворяют условию (6.7), это означает, что, как уже отмечалось, тестируемое ПО теряет при вычислениях p значащих десятичных цифр по сравнению с эталонным. Число потерянных цифр p можно определить с помощью исполнительной характеристики

$$P(\vec{x})=\lg(1+\frac{1}{k(\vec{x})\eta}\cdot\frac{\|\Delta\vec{y}\|}{\|\vec{y}^{(эт)}\|}). \tag{6.9}$$

В самом деле, если второе слагаемое под знаком логарифма оказывается порядка 10^p, то даже в самом неблагоприятном случае, когда $p=1$, единицей под знаком логарифма можно пренебречь по сравнению с десятью, и в общем случае с большой степенью точности можно написать, что

$$P(\vec{x})\cong\lg(10^p)=p. \tag{6.10}$$

Если же вычислительные возможности тестируемой и эталонной программ совпадают, то $\Delta\vec{y}=0$, и, следовательно, $P(\vec{x})=0$, т.е. $p=0$ и никакой потери цифр не происходит.

С учетом (6.5) выражение для исполнительной характеристики (6.9) может быть записано в виде

$$P(\vec{x})=\lg(1+\frac{\|\Delta\vec{y}\|}{\|J(\vec{x})\|\cdot\|\vec{x}\|\cdot\eta}). \tag{6.11}$$

Можно показать, что для алгоритма вычисления СКО

$$S^2=\frac{\sum_{i=1}^{n}(x_i-\bar{x})^2}{(n-1)}, \tag{6.12}$$

где $\bar{x}$ среднее значение результатов измерений x_i

$$\bar{x}=\frac{1}{n}\sum_{i=1}^{n}x_i. \tag{6.13}$$

для коэффициента обусловленности получается выражение

$$k^2(\vec{x})=\left(\frac{n-1}{n}\right)^2+\left(\frac{n-1}{n}\right)\left(\frac{\bar{x}}{S}\right)^2. \tag{6.14}$$

Видно, что первое слагаемое в этом выражении является константой, зависящей от объема выборки. При больших объемах значение этого слагаемого с большой степенью точности равно единице. Второе слагаемое

зависит от отношения $\left(\frac{\bar{x}}{S}\right)$ и равно нулю, когда $\bar{x} = 0$. Это означает, что при больших объемах выборки значение коэффициента обусловленности в основном определяется значением указанного отношения и может быть оптимизировано его подбором.

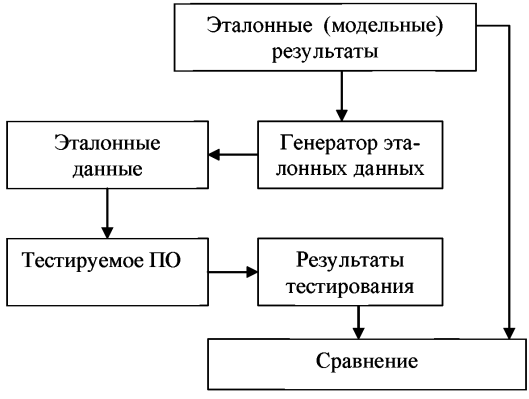
Исполнительная характеристика для СКО получается подстановкой формулы (6.14) в выражение (6.9).

Аналогично могут быть получены выражения для коэффициентов обусловленности и исполнительных характеристик и для других, используемых при обработке измерительной информации алгоритмов.

Метод генерации эталонных данных

Содержание этого Приложения основано на разработках Национальной физической лаборатории (Великобритания).

В излагаемом методе тестирование программного обеспечения происходит в соответствии со следующей схемой



Рассмотрим применение метода генерации эталонных данных на примере задачи простой линейной регрессии.

Известно, что линейный регрессионный анализ объединяет широкий круг задач, связанных с построением функциональных зависимостей между двумя группами числовых переменных: $x_1,\ldots,x_p$ и $y_1,\ldots,y_q$.

На практике широко распространенной является ситуация, когда функция $\vec{y} = f(\vec{x})$ линейно зависит от $\vec{x}$, т.е. предполагается, что эта зависимость имеет вид

$$y_i = b_1 + b_2 x_i + r_i, \tag{7.1}$$

или в векторном виде

$$\vec{y} = A\vec{b} + \vec{r}, \tag{7.2}$$

где матрица наблюдений A и векторы в этой формуле, в том числе вектор остатков $\vec{r}$, имеют вид

$$A=\begin{pmatrix}1 & x_1\\ 1 & x_2\\ . & .\\ . & .\\ . & .\\ 1 & x_m\end{pmatrix},\ \vec{y}=\begin{pmatrix}y_1\\ y_2\\ .\\ .\\ .\\ y_m\end{pmatrix},\quad \vec{b}=\begin{pmatrix}b_1\\ b_2\end{pmatrix},\ \vec{r}=\begin{pmatrix}r_1\\ r_2\\ .\\ .\\ .\\ r_m\end{pmatrix}\quad . \tag{7.3}$$

Задача нахождения значений параметров $\vec{b}$ при заданных (известных из эксперимента) векторах $\vec{y}$ и $\vec{x}$ называется задачей простой линейной регрессии, а ее решение ищется методом наименьших квадратов (МНК).

Задача, рассматриваемая здесь, заключается в построении такого набора эталонных y, который имитировал бы экспериментальные данные при тестировании программных продуктов, реализующих МНК. Эта задача, в частности, решается методом нуль – пространства.

Решение задачи простой линейной регрессии МНК требует, чтобы выполнялось условие (см. формулу (7.2))

$$A^T\vec{r}=0, \tag{7.4}$$

где A^T- матрица, транспонированная по отношению к матрице A. В свою очередь, условие (7.4) эквивалентно выполнению двух условий

$$\begin{aligned}&\sum_{i=1}^{m} r_i = 0,\\ &\sum_{i=1}^{m} x_i r_i = 0.\end{aligned} \tag{7.5}$$

В теории линейных алгебраических уравнений вектор остатков $\vec{r}$, из-за того, что он удовлетворяет уравнению (7.4) – (7.5), рассматривается как вектор, принадлежащий так называемому нуль – пространству матрицы A^T.

Пусть теперь N будет базисом в нуль – пространстве матрицы A^T, т.е.

$$A^T N = 0. \tag{7.6}$$

Этот базис может быть выражен как набор линейно независимых векторов, которые и образуют матрицу N.

Если известен базис нуль – пространства, определяемый формулой (7.6), то вектор $\vec{r}$ может быть представлен в виде линейной комбинации базисных векторов, т.е.

$$\vec{r} = N\cdot\vec{u}, \tag{7.7}$$

где вектор $\vec{u}$ представляет коэффициенты этой линейной комбинации.

Замечательной особенностью рассматриваемого метода является то, что замена вектора $\vec{y}$ в формуле (7.2) на $\vec{y}+\vec{r}$, где вектор $\vec{r}$ определяется формулой (7.4), оставляет неизменным решение $\vec{b}$ линейной регрессионной задачи.

В самом деле, с учетом условия (7.4), получаем

$$A^T(\vec{y}+\vec{r}) = A^T\vec{y} + A^T\vec{r} = A^T\vec{y} + A^T N\cdot\vec{u} = A^T\vec{y}, \qquad (7.8)$$

т.е. ничего не меняется по сравнению со случаем, когда остаточного члена не было. Из написанного соотношения следует, что, построив один набор данных $\vec{y}$, выбором вектора $\vec{u}$ можно построить множество наборов данных, имеющих одно и тоже решение линейной регрессионной задачи $\vec{b}$. Обычно векторы $\vec{u}$ выбираются в виде набора случайных чисел, имитирующих измеренные данные.

Таким образом, процедура генерации эталонных данных в общих чертах состоит из последовательности следующих действий:

1. Предполагая строгую линейную зависимость (линейную модель) откликов системы на входные воздействия, строится модельный вектор наблюдений $\vec{y}_0 = A\vec{b}$, где индексом «0» обозначены эталонные (модельные) результаты, являющиеся «входом» для генерации эталонных данных.
2. С помощью программной функции строится базис N нуль – пространства матрицы A^T (такой стандартный программный пакет, как Matlab, содержит в качестве библиотечной такую функцию).
3. С помощью соотношения $\vec{r} = N\cdot\vec{u}$ строится вектор остатков, при этом компоненты вектора $\vec{u}$ выбираются в виде случайных чисел, имитирующих ошибки измерений. Часто вектор $\vec{r}$ необходимо видоизменять (масштабировать) таким образом, чтобы он представлял распределение случайных чисел с заданными средним значением и СКО.
4. По формуле $\vec{y} = \vec{y}_0 + \vec{r}$ образуется вектор наблюдений, представляющий собой сгенерированные эталонные данные.

Исполнительная характеристика $P(x)$ (формула 6.9) для рассматриваемой задачи определяется соотношением

$$P(x) = \lg\left(1 - \frac{|y_0 - y_{тест}|}{k(x)\cdot\eta\cdot y_0}\right), \qquad (7.9)$$

где $y_{тест}$ - результат, полученный тестируемым ПО. Можно показать, что $\lg k\{x\}$ определяет число цифр точности, теряемых алгоритмом вычисления y_0 при решении задачи линейной регрессии. Исполнительная характеристика, в свою очередь, показывает количество потерянных значащих цифр при обработке эталонных данных тестируемым ПО по сравнению с априори известным модельным решением измерительной задачи (Приложение 6).

Поведение исполнительной характеристики зависит от выбора значений так называемых исполнительных параметров, в качестве которых могут, в частности, выступать:

m - объем выборки;

$\bar{x}$ - среднее значение набора входных воздействий (оно может совпадать с медианным значением);

s - среднее квадратическое отклонение (измерительный шум).

Для каждого исполнительного параметра задается область принципиально возможных значений, а также набор номинальных (наиболее часто встречающихся) значений. При этом набор исполнительных характеристик строится по следующему принципу: фиксируются номинальные значения двух исполнительных параметров (например, объем выборки и СКО) и определяется зависимость исполнительной характеристики от значений третьего исполнительного параметра (в данном случае от среднего значения) во всей области его возможного изменения.

Приведенная выше процедура генерации эталонных данных была реализована в пакете Matlab, в котором был построен генератор данных, примененный для тестирования функций линейной регрессии в Matlab и в некоторых других пакетах. Расчеты велись с точность 10^{-16}. Полученные данные заносились в программный пакет Microsoft Excel, в котором строились соответствующие графики, при этом объем выборки m во всех расчетах брался равным 100. Результаты расчета исполнительной характеристики $P(x)$ для функции линейной регрессии показывают, что в зависимости от значения исполнительных параметров число потерянных цифр точности может изменяться от 0 до 5.

gost.e-ecolog.ru