

КАЧЕСТВО И НАДЕЖНОСТЬ ПРОГРАММНЫХ СРЕДСТВ, ИСПОЛЬЗУЕМЫХ ДЛЯ ОБОСНОВАНИЙ БЕЗОПАСНОСТИ В ОБЛАСТИ ПРОЧНОСТИ И УСТОЙЧИВОСТИ К ВНЕШНИМ ВОЗДЕЙСТВИЯМ

*И.В. Калиберда, кандидат технических наук,
председатель секции № 8 Совета по аттестации программных средств
Госатомнадзора России*

Постановка задачи исследования

Необходимость проведения экспертизы программных средств (ПС), применяемых при обосновании безопасности объектов использования атомной энергии, возникла в середине 80-х годов в связи с активным применением вычислительной техники. По инициативе надзорного органа при разработке Правил ядерной безопасности реакторных установок атомных станций (ПБЯ РУ АС-89) впервые было введено требование о необходимости аттестации ПС. Процедура аттестации ПС введена в 1991 г.

В числе других секций Совета по аттестации ПС Госатомнадзора России была создана секция № 8 «Параметры поведения конструкций, оборудования и трубопроводов при статических и динамических нагрузках».

Для обеспечения деятельности по аттестации ПС возникла необходимость разработки ее методологии. Согласно названию секции, в ее предметную область входили три составляющие:

1. Нагрузки и воздействия на здания, сооружения, конструкции, оборудование и трубопроводы объектов использования атомной энергии.
2. Параметры колебаний, прочность и устойчивость системы «грунт – фундамент – сооружение».
3. Параметры колебаний, прочность и устойчивость трубопроводов и оборудования АЭС.

Кроме решения организационных задач по аттестации ПС (они здесь не рассматриваются), необходимо было:

- изучить подходы к обеспечению качества ПС;
- выявить и классифицировать причины, влияющие на результаты расчета;
- разработать требования к верификационному отчету;
- определить содержание и предмет верификации;
- разработать тесты для аттестации ПС в области учета взаимодействия сооружения с основанием, расчетов трубопроводов, оборудования и конструкций;
- разработать основы информационного банка данных экспериментальных исследований и натурных испытаний зданий и сооружений на динамические воздействия.

Подходы к решению проблемы качества и надежности ПС

Качество ПС всегда было и остается одной из основных проблем в теории и практике программирования. От качества программ зависят их жизнеспособность и конкурентоспособность, т.е. уровень эффективного и полноценного применения средств вычислительной техники в различных областях человеческой деятельности. Обеспечение надежности, как важнейшего качества программ, неизменно остается ключевым направлением развития современной методологии программирования. Традиционный способ обеспечения надежности программ путем тестирования не может полностью удовлетворить возрастающие требования практики. Качественно новый уровень в решении этой проблемы может быть достигнут сочетанием методов тестирования и верификации программ.

В общем случае верификация есть подтверждение корректности программы (отсутствия ошибок в программе) и может рассматриваться как дальнейшее развитие идей тестирования в новом аспекте [1, 2]. Если тестирование ограничено исследованием отдельных выполнений программы (для некоторых путей вычислений), то верификация – анализ свойств всех допустимых выполнений программы с помощью формальных доказательств присутствия требуемых свойств.

Верификация и подтверждение правильности являются основными средствами гарантии качества ПС. Роль верификации становится решающей, поскольку обработка данных используется во всех ответственных областях жизни, таких, как здравоохранение, транспорт, ядерная энергетика, космос и др., где отказ может иметь катастрофические последствия. Роль верификации с экономической точки зрения можно проследить по статистике тестирования программ – самой известной в данном вопросе деятельности. Качественно оценивая усилия по тестированию программы после ее написания, можно сказать, что интеллектуальные затраты на тестирование приблизительно равны затратам на создание программы и эти затраты будут увеличиваться. Поэтому вполне закономерно сделать вывод о том, что тестирование и верификацию программ следует начинать на самых ранних этапах их разработки [1].

Программа, претендующая стать продукцией, проходит через следующие три стадии: разработка, производство и использование. Наиболее ответственной является стадия разработки, в процессе которой формируются свойства, составляющие качество создаваемой программы. Для обеспечения ее жизнеспособности имеет значение также стадия производства, так как на нее возлагается вся ответственность по доведению разработанной программы до уровня продукции производственно-технического назначения и поддержке ее. Очевидно, использование – целевая стадия жизненного цикла программы. Понятие «жизненный цикл» неразрывно связано с понятием «технология программирования» (организованная совокуп-

ность методологических положений, методов, способов, средств автоматизации, направленных на организацию и осуществление процессов разработки, производства и использования программ). Цель технологии программирования состоит в обеспечении требуемых показателей качества программы в заданных условиях ее создания и использования.

Методология тестирования программных средств

В [1, 3, 4] тестирование является одним из важных условий повышения надежности программы, хотя само по себе оно надежность не повышает. Надежность программы определяется правильностью этапов проектирования. Роль тестирования заключается в выявлении тех имеющихся в программе немногочисленных ошибок, которые еще остаются в хорошо спроектированной программе. Попытки с помощью тестирования повысить надежность плохо спроектированной программы обречены на неудачу. Для тестирования программ применяется целый ряд различных методов. По методам применения тестирование можно разделить на детерминированное, при котором известно и регистрируется каждое значение теста, и стохастическое, когда характеристики тестов описываются некоторыми распределенными или статистическими параметрами. Возможно использование обоих методов, но при подготовке и в рабочем функционировании программы применяют преимущественно детерминированный. На завершающих этапах при комплексном подходе проводится стохастическое тестирование. Основные задачи детерминированного тестирования – установление факта работоспособности программы, соответствие их требованиям технического задания и выявление ошибок. Стохастическая проверка программы преследует следующие цели:

- обнаружение ошибок, отладку программ и обеспечение характеристик программы в соответствии с требованиями технического задания;
- испытание программы в реальных условиях.

Тестирование призвано гарантировать правильность решения требуемой задачи. Известно, что цена обнаруженной и исправленной ошибки растет для каждого очередного этапа цикла разработки. По этой причине при тестировании обычно применяется подход “строительных блоков”, начиная с элементарных и стандартных программ. На данном уровне можно, кроме того, проверить большее количество логических путей программы. Процесс тестирования состоит из проверки и сбора статистических данных о производительности в ходе работы программы. Тестирование можно условно разделить на планирование тестирования, проектирование тестирования, выполнение тестов и оценку результатов. Каждый тест проектируется и документируется в тестовых спецификациях. Для каждого теста предусматриваются свой план, порядок проведения и отчет. Порядок проведения теста описывает последовательность выполнения теста, входные данные теста, базу данных, определенную конфигурацию ПС, а также требуемый персонал и его функции. Тест обычно заверяется лицами, ответственными за гарантию качества. Фиксируются все замеченные нарушения. Выходные результаты теста передаются для последующей оценки. Наблюдения при выполнении теста и оценка выходных результатов теста составляют базис, на основе которого определяются, выполнены ли задачи теста, выполнены ли нужные требования. Ручной контроль ошибок сам по себе является потенциальным источником ошибок. Это один из главных аргументов в пользу разработки автоматизированных средств тестирования, полезность и важность которых становятся несомненными.

За исключением наиболее простых программ, к тестированию разумно подходить последовательно, начиная с уровня программных модулей. Большой накопленный практический опыт свидетельствует о том, что обнаружение и исправление ошибок дешевле всего обходятся на ранних этапах цикла разработки. В данном случае легче всего полностью охватить и просмотреть программный код. На этом уровне упор в тестировании делается на верификацию логики вычислений, манипулирование данными, временные соотношения и требования к размерам. После завершения тестирования модули соединяются для определения их совместной функции. При комплексном тестировании ПС рассматривается как состоящее из программных компонент и не принимается во внимание детальная логика программ, которая была предметом тестирования модулей. Поэтому основной упор делается на взаимодействие программных компонент и на их интерфейсы. До данного момента тестирование чаще всего выполняет сам разработчик. Оно протекает в моделируемой среде. Поэтому нельзя с уверенностью сказать, что программная система сможет работать в условиях реальной эксплуатации. Понадобится несколько тестов на месте эксплуатации, т.е. полевые испытания. Проводить их желательно в службах, занимающихся тестированием и испытаниями готового программного продукта.

Автоматическое тестирование

По мере роста размеров и сложности программных систем затраты на их проверку возрастают еще быстрее. Возникает потребность в массовом привлечении людских ресурсов. Это и дорого, и недостаточно эффективно с точки зрения надежности. Несмотря на расходование половины бюджета программной разработки на тестирование, в программном обеспечении остается значительное количество ошибок, которые часто очень серьезно нарушают нормальную работу системы. Эта ситуация привела к разработке автоматических средств тестирования, способствующих созданию эффективных тестов и облегчающих анализ результатов теста.

Автоматические средства тестирования обладают следующими характерными свойствами [1, 4, 5, 6]:

- улучшенная организация тестирования при автоматизации;

- измерение объемов тестирования;
- повышенная надежность.

Как показала практика, тестирование очень больших программ с привлечением автоматизированных средств выполнено с сокращением на 30% машинного времени и трудовых затрат и с повышением достоверности проверки. Автоматизация делает процесс более строгим. Но применение автоматизированных средств полезно при работе с большими объемами тестовых данных.

Примерная классификация средств по функциональному признаку:

1. Анализ проекта системы:

а) средства автоматизированного проектирования, поддерживающие стройную методологию написания проектных требований;

б) автоматизированные средства моделирования аппаратного и программного обеспечения системы с целью изучения их характеристик.

2. Статический анализ исходной программы:

а) средства анализа программного кода, выполняющие синтаксический анализ исходного кода и определяющие сомнительные конструкции;

б) средства контроля структуры программы, строящие графы и определяющие структурные изъяны;

в) средства контроля за правильностью взаимодействия модулей, обнаруживающие несогласованности в описаниях структур данных и неправильные связи между модулями;

г) средства контроля последовательности событий в программе (например, последовательность ввода-вывода).

3. Динамический анализ исходной программы:

а) средства текущего контроля за поведением программы в рабочее время;

б) средства автоматизированной генерации тестовых данных, помогающие подбирать тестовые данные для проверки программного кода.

4. Эффективность:

а) средства изменения структуры программы, оказывающие помощь в реорганизации программы в целях оптимизации.

5. Средства аттестации качества программ, определяющие уровень их качества на основе сравнения с желаемыми характерными свойствами.

Данная классификация приведена только в порядке освещения диапазона применения автоматизированных средств. Подобные средства получают информацию из исходного текста программы, которая анализируется, и могут быть названы анализаторами исходных программ.

Анализатор исходных программ выполняет пять основных функций:

1) анализ исходного кода и создание баз данных;

2) генерацию протоколов статического анализа исходного кода;

3) внесение программных вставок в исходный код;

4) анализ результатов тестов и генерацию отчетов;

5) генерацию отчетов для поддержки испытаний, помогающих при организации отчетов тестирования.

Система анализа исходного текста программы является типичным примером средств, используемых для оценки качества.

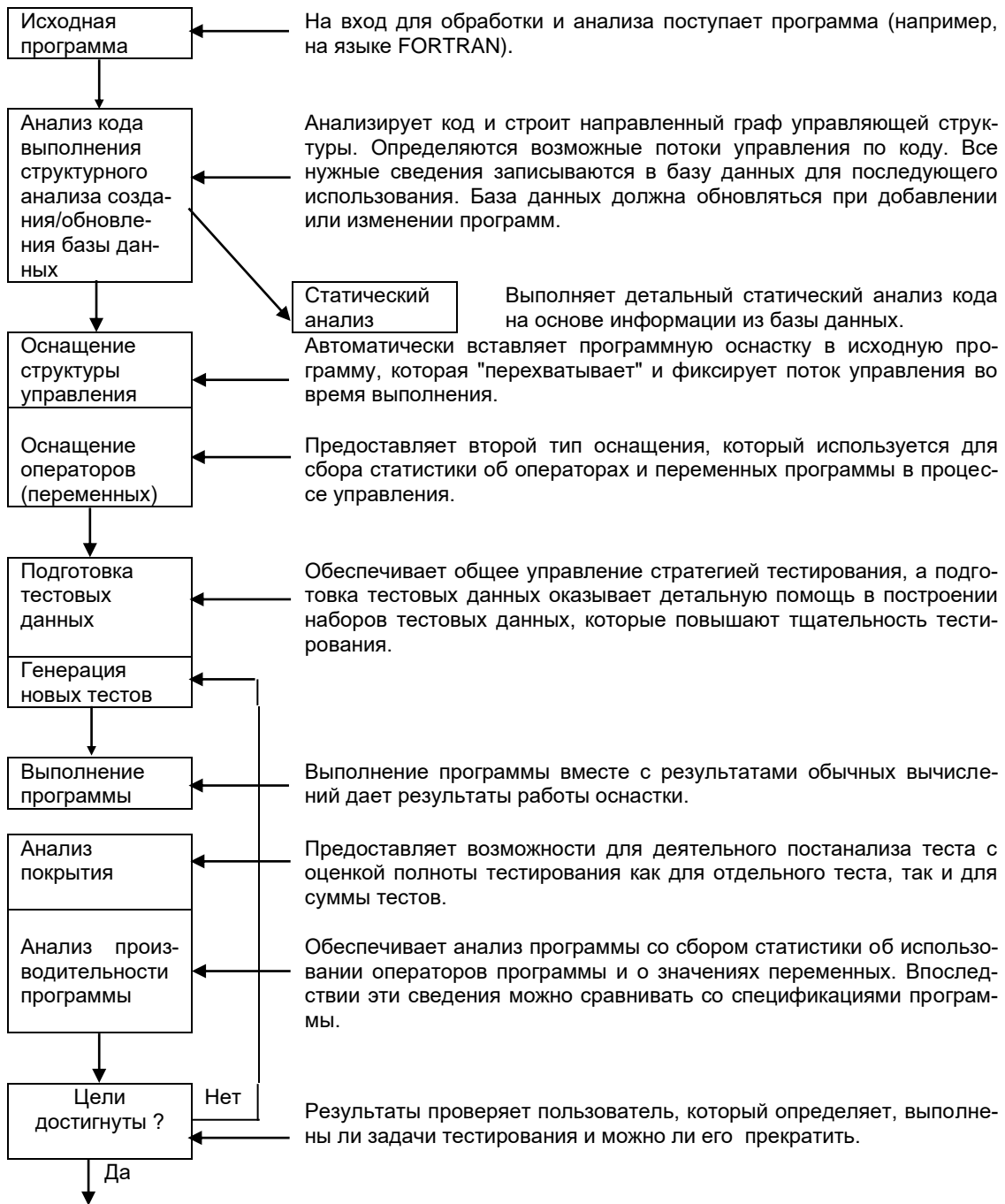
Завершением процесса тестирования считается положение, когда программа выдает правильные результаты на наборе тестов, удовлетворяющем заранее зафиксированному критерию или критериям.

В таблице приводится структура применения анализатора исходных программ в тестировании программ.

Методы верификации программ и их перспективы

Согласно [1, 2, 4, 7], главный недостаток метода тестирования - невозможность гарантировать отсутствие ошибок в программе. Некоторые разработчики предполагают, что программы, почти не содержащие ошибок, можно конструировать, используя определенные организационные методы, и что правильность этих программ может быть доказана в статическом режиме (без реального выполнения) с помощью математических доказательств. Этот конструктивный подход к разработке программ используется в настоящее время в форме структурного проектирования и программирования. В статическом подходе используется "доказательство правильности программ", которое служит предметом исследований. При доказательстве правильности программы операторы исходной программы используются для доказательства математических теорем, описывающих поведение программы. Ожидаемая реакция программы представляется в виде ряда утверждений. Предположения проектировщиков отражаются в этих

Применение анализатора исходных программ в тестировании программ



утверждениях в виде значений переменных в промежуточных точках программы, затем программа рассматривается как теорема и выполняется доказательство утверждений. Одна из трудностей использования указанного метода заключается в необходимости разработки утверждений, которые надо доказывать. Главная причина, ограничивающая широкое использование этих методик, – превышение иногда требуемого объема вычислений размера самой программы. Также следует отметить, что математическое доказательство не является методом, свободным от недостатков. Как и в любом другом методе, в данном случае могут использоваться неверные выводы. Но у этого метода есть и преимущества: при доказательстве могут быть обнаружены такие ошибки, которые не удалось бы выявить с использованием методов тестирования.

Постановка задачи – первый и один из важнейших этапов в общем цикле разработки программы, и его роль в создании качественных программ трудно переоценить.

Задача всегда формулируется вначале на естественном языке некоторой проблемной области, к которой она относится. Однако этот уровень постановки задачи недостаточен для ее успешного программирования. Статистика показывает, что около 65% ошибок, сделанных программистом, вызваны неверным пониманием задачи. Указанные особенности приводят к целесообразности выделения промежуточного уровня между естественными языками и языками программирования. На этом промежуточном уровне находятся языки спецификаций.

Основная идея верификации программы состоит в том, чтобы формально доказать соответствие между текстом программы (на языке программирования) и спецификацией задачи (на языке спецификаций). Спецификация – любое описание ожидаемой функции, выполняемой программой, содержащее соотношение между аргументами и значениями ожидаемой функции.

Языки спецификаций – целое семейство языков, призванных конкретизировать постановку задачи и облегчить их последующее программирование и сопровождение. Различают два вида спецификаций: внешние (языки спецификаций задач) и внутренние (языки спецификаций свойств программ). Принято выделять в языках спецификаций задач два класса: языки описания требований и языки функциональных спецификаций. Языки описания требований – обычно полужформальные языки, подобные языкам блок-схем программ. Используются для описания данных, их потоков.

Языки функциональных спецификаций – формальные языки полного описания функции, которую должна реализовывать программа или ее части. Функциональные спецификации могут служить не только точным формализованным заданием на программирование, но и удобным средством для тестирования и верификации задачи до ее программирования. Языки функциональных спецификаций с указанными свойствами могут быть созданы на прочной математической основе.

Сопоставление спецификации задачи и спецификации свойств программы позволяет проводить коррекцию задачи или программы при последующих итерациях. Это создает основу для эффективных технологий надежного проектирования программ.

Следует также отметить, что верификация программ требует от программиста значительно более высокой квалификации, чем тестирование. Верификация нетривиальной программы очень сложна и трудоемка. В одной из работ было показано, что требуется около месяца труда квалифицированного специалиста для доказательства правильности программы, состоящей из 100 строк. Существуют некоторые автоматизированные системы верификации, например, для языка ПАСКАЛЬ, имеющего аксиоматическое построение.

Как показывает накопленный опыт верификации программ, методы верификации практически можно приложить к любым проблемным областям, в том числе нечисловой природы (логические задачи, обработка строк, текстов, файлов и др.). Основными препятствиями применения методов верификации обычно являются не принципиальные (теоретические) ограничения, а несовершенство традиционных технологий, отсутствие подготовленных кадров, владеющих соответствующим аппаратом и методикой, и инструментальных средств поддержки верификации.

В итоге можно сказать, что методы верификации в настоящее время продолжают находиться в стадии исследований. В то же время полученные результаты исследований по верификации заслуживают того, чтобы посмотреть на эту проблему как на составную часть перспективной технологии программирования и методологии экспертизы ПС.

Классификация причин, влияющих на результаты расчета

Результаты расчета являются следствием ряда небесспорных операций, каждая из которых чревата новыми отклонениями результатов в ту или иную сторону от физически точных. Среди этих этапов выделим основные.

А. Переход от объекта к физико-математической модели.

На этом этапе делается ряд физических допущений, таких как, например:

- линейность физических законов, описывающих работу материала;
- вообще вид законов (вязкость, пластичность, трещины и т.д.);
- отсутствие сугубо нелинейных макроэффектов типа падения одной части конструкции на другую, отрыва и проскальзывания по границе грунт – фундамент;
- вообще перечень рассматриваемых эффектов;
- разнообразные допущения о воздействиях (природе учитываемых воздействий, их сочетаниях и взаимной зависимости) и т.п.

Конкретный набор допущений зависит от программы и далеко не всегда осознается расчетчиком. Отказ от этих допущений чаще всего выводит задачу за границы возможностей программы.

Б. Переход от математической модели к расчетной схеме.

На этом этапе вводятся дополнительные допущения, но теперь они в большей мере зависят от расчетчика. Эти допущения можно разделить на две группы.

К первой относится выбор типа расчетной схемы. Например, предположение, что пространственная конструкция деформируется с соблюдением гипотез Кирхгофа-Лява (и, следовательно, может быть описана как классический стержень), вводится расчетчиком практически явно, хотя и не всегда обосновывается. Сюда же относится предположение о жесткости фундамента, вертикальности падающей сейсмической волны. Отдельно для строительной механики следует выделить предположение о малости деформаций, лежащее в основе линейной теории упругости. Отказ от этих предположений (кроме последнего) обычно не выводит задачу за пределы возможностей программы, хотя и делает расчетную схему более

громоздкой (например, пространственная модель вместо стрессовой, учет податливости фундамента, учет неперпендикулярности сейсмических волн и т.д.).

Вторая группа предположений, которая обычно не осознается вовсе, – подбор исходных данных для расчетной схемы. Возможно, эти предположения следовало бы отнести к этапу А или выделить в отдельную группу. Отказ от них не влияет явно на вид расчетной схемы. Пример такого предположения: жесткость грунта в расчетной схеме следует определять на основании результатов сейсморазведки (скорости волн) и лабораторного анализа образцов (плотности грунта). Вместо этого предположения можно было бы постулировать, что жесткость грунта в той же расчетной схеме определяется на основании резонансных динамических испытаний образцов грунта. В обоих случаях мы фактически вводим предположение, что в условиях рассматриваемой задачи (например, землетрясения на площадке АЭС) грунт будет вести себя так же, как при экспериментах, из которых берутся исходные данные для расчетной схемы. Выбор одного из двух названных источников исходных данных в данном примере существенно влияет на результат расчета, хотя расчетная схема не меняется.

В. Переход от расчетной схемы к алгоритму получения результата.

Этот этап можно пройти и без дополнительных предположений, но лишь для очень простых задач. Большинство же задач, даже полностью и корректно заданных математически, не имеют замкнутого решения. Для получения результата применяются приближенные методы (например, разложение в ряд с удержанием конечного числа членов, приближенное вычисление интегралов, интегрирование по времени с конечным шагом, итерационные методы и т.д.). Такие же допущения вводятся практически при применении метода конечного элемента: используемые типы элементов и сетки моделей сводят исходную задачу к некоторой другой. Здесь многое зависит от расчетчика, поскольку, например, с помощью одной и той же конечно-элементной программы можно получать для одной задачи очень разные результаты в зависимости от сетки и типов элементов. Чаще всего у расчетчика есть в распоряжении определенные средства контроля точности расчетов (шаг по времени, размеры конечного элемента и т.д.). По отношению к дополнительному отклонению результатов в силу предположений этого этапа обычно применяют термин “погрешность метода”. В то же время известны ситуации, когда указанная погрешность специально увеличивается для компенсации погрешности расчетной схемы.

Г. Реализация алгоритма в программе и (отдельно) в расчете.

Здесь речь идет о так называемых “погрешностях вычислений” и о возможных ошибках в программе и (отдельно) в расчете, т.е. при задании исходных данных и диалоге с программой. Известны ситуации, когда погрешности вычислений обесценивают результаты, полученные по правильному в принципе алгоритму.

Содержание и предмет верификации результатов расчета

С точки зрения конечного результата, т.е. верификации результатов расчета, на их отклонение от истинных физических влияют причины всех четырех этапов (А, Б, В, Г). Однако с организационной точки зрения ответственность расчетчика и автора ПС должна быть четко разделена. Автор ПС не может нести ответственность за правильность допущений группы А (при условии, что они четко оговорены в документации на ПС), как и за допущения группы Б. В группе В автор ПС должен нести ответственность за “погрешности метода”, недоступные корректировке пользователем. В группе Г допущений автор ПС несет полную и исключительную ответственность за ошибки, допущенные в ПС при реализации алгоритма. Таким образом, верификация ПС является лишь достаточно ограниченной частью верификации результатов расчета.

Расчетчик в принципе несет ответственность за все остальное, хотя, учитывая реальную ситуацию, проектировщики в силу различных причин (в первую очередь, загруженности другой работой) не могут реально представить себе последствия тех или иных допущений. Здесь они должны привлекать себе на помощь исследователей, которые на основании расчетов и экспериментов должны оценивать допустимость тех или иных предположений в конкретном случае. Очень редко появляется возможность четко очерчивать количественно границы допустимости тех или иных предположений и давать рекомендации расчетчику в однозначном виде.

Таким образом, верификация результатов расчета должна включать в себя:

- а) верификацию ПС на предмет отсутствия ошибок;
- б) верификацию ПС на предмет погрешности метода применительно к конкретному случаю;
- в) верификацию исходных данных на предмет ошибок расчетчика;
- г) верификацию исходных данных на предмет контроля точности;
- д) верификацию исходных данных на предмет их источников;
- е) верификацию расчетной схемы на предмет ее применимости к конкретной ситуации;
- ж) верификацию физических допущений этапа А на предмет их применимости к конкретной ситуации.

Рассмотрим последовательно все семь перечисленных составляющих верификации расчета, и остановимся на средствах реализации каждой из них.

а) Верификация ПС на предмет отсутствия ошибок производится сопоставлением результатов либо с теоретическими решениями (если они имеются), либо с результатами, полученными с помощью других ПС при тех же допущениях. Надо учитывать, что для сложных программ полная верификация реально недостижима, так что приходится заботиться о представительности разбираемого ограниченного

круга примеров. Часть данной верификации обязан делать автор ПС еще до вынесения ПС на аттестацию. Привлечение эксперимента к этой части верификации лишено смысла.

б) Верификация ПС на предмет погрешности метода производится также с помощью либо теоретических решений, либо других ПС, работающих с такой же расчетной схемой, но другими методами. Эксперимент здесь может играть некоторую роль (например, исследования на фотоупругих моделях), если свойства модельных материалов известны и совпадают с заложенными в расчетные схемы.

в) Верификация исходных данных на предмет ошибок, к сожалению, практически нереальна. Остается полагаться на возможность экспертной оценки промежуточных результатов (например, собственных частот и форм). Такой верификации служит, в частности, визуализация расчетных схем и результатов (форм, деформированного состояния).

г) Верификация исходных данных на предмет контроля точности может проводиться самим расчетчиком либо путем сравнения двух расчетов (например, одного с определенным шагом по времени и второго с вдвое меньшим), на основании априорных оценок (зная частоту среза, можно оценивать требуемый шаг по времени в долях минимального периода) или тестовых расчетов, в которых продемонстрирована сходимости результатов в зависимости от параметров точности. Эксперимент здесь также лишен смысла.

д) Верификация исходных данных на предмет контроля их источников должна основываться на результатах экспериментов. Эксперименты должны показывать, во-первых, разброс данных в зависимости от техники их получения, во-вторых, какая методика получения данных наиболее подходит для конкретного расчетного случая. Эти эксперименты могут выполняться как на натуре, так и в лаборатории, но участвовать в них должны реальные (не модельные) материалы. Программа экспериментов должна составляться каждый раз заново. Расчеты будут играть вспомогательную роль.

е) Для верификации расчетной схемы на предмет ее применимости к конкретной ситуации могут служить как расчеты, так и эксперименты. Расчеты следует проводить по альтернативным, желательно более точным расчетным схемам (для данного случая или аналогичного, по мнению эксперта). Здесь же могут быть использованы результаты как лабораторных экспериментов на моделях, так и натуральных наблюдений (это особенно ценно), а также натуральных экспериментов (если удастся достаточно точно воспроизвести нагрузки). Так, например, для исследования вопроса о применимости стержневых схем к расчету сооружений АЭС может быть использовано сравнение как результатов расчета по стержневым схемам с результатами расчета по пространственным схемам, так и данных испытаний моделей, а также данные натуральных наблюдений (во время землетрясений) и натуральных экспериментов (при решении вопроса о воздействиях).

ж) Верификация физических допущений первого этапа – наиболее сложная часть работы. Здесь должны использоваться как расчетные, так и все три вида экспериментальных исследований: лабораторные исследования на моделях, лабораторные исследования с реальными материалами на фрагментах, натурные эксперименты и наблюдения. Расчетные исследования могут использоваться в двух направлениях. Во-первых, в самом ПС могут содержаться встроенные ограничители (например, если нелинейность проявляется выше некоторых пороговых значений, то в чисто линейном расчете могут выводиться максимальные величины, которые позволят определить, достигаются ли пороговые значения в рассматриваемом конкретном случае). Во-вторых, дополнительные расчеты могут проводиться по другим ПС, которые способны учитывать ряд дополнительных эффектов (на практике это менее реально, поскольку такой расчет дороже исходного и зачастую просто недоступен, но принципиально такие ситуации возможны). Главная проблема с лабораторными экспериментами на моделях – соответствие характеристик моделей натуре с учетом масштабных эффектов. Более перспективным представляется другое направление – лабораторные испытания фрагментов из реальных материалов при нагрузках, близких к реальным, с параллельным расчетом по тому же ПС не всей конструкции, а именно рассматриваемых в эксперименте фрагментов. Наконец, особую ценность представляют натурные эксперименты в сочетании с расчетами не на природные, а именно на синтезированные к целям эксперимента нагрузки.

Об экспериментальной проверке ПС, используемых для расчета параметров колебаний и напряженно-деформированного состояния строительных конструкций, оборудования и трубопроводов

При рассмотрении ПС для решения задач прочности и расчета параметров колебаний зданий, сооружений, трубопроводов и оборудования возникает необходимость экспериментального подтверждения результатов, полученных по данной методике.

Согласно разработанным положениям об экспериментальной проверке ПС, верификация расчета должна включать в себя ряд составляющих, в том числе:

- верификацию исходных данных (этап 1);
- верификацию расчетной схемы на предмет ее применимости к конкретной ситуации (этап 2);
- верификацию физических допущений этапа 1 на предмет их применимости к конкретной ситуации.

При верификации исходных данных должны, во-первых, даваться оценки разброса данных в зависимости от техники их получения, во-вторых, оценки пригодности методик получения данных, а также того, насколько они подходят для конкретного расчетного случая.

Для верификации расчетной схемы (этап 2) на предмет ее применимости к конкретной ситуации, кроме расчетов, могут служить результаты экспериментов на моделях, натуральных наблюдений (что особенно ценно), а также натуральных экспериментов (если удастся достаточно точно воспроизвести нагрузку).

Экспериментальной проверке могут быть подвергнуты результаты расчетов, выполненных с использованием ПС, но не само это средство.

Положения верификационного отчета

Согласно основным требованиям к верификационному отчету, в отчете о верификации должны быть представлены:

1. Анализ математической модели, результаты тестирования, метрологические данные, данные о встроенном диагностическом контроле и т.д.
2. Сопоставление результатов вариантов расчетов по ПС с:
 - экспериментальными данными;
 - аналогичными результатами, полученными по отечественным и зарубежным аттестованным ПС, если таковые имеются;
 - аналитическими решениями и т.п.
3. Анализ причин возможного несовпадения результатов расчета по аттестуемому ПС с представленными в отчете данными, а также влияния этого несовпадения на прочность и устойчивость конструкций.

В верификационном отчете при сопоставлении результатов расчетов по ПС с экспериментальными данными должно быть отражено следующее:

- матрица верификации ПС;
- перечень образцовых экспериментов, используемых для верификации данного ПС, качественные описания этих экспериментов, количественные зависимости сравниваемых параметров и их максимальные отклонения;
- перечень экспериментов, выполненных для проверки ПС в организации-разработчике ПС или в других организациях, описание экспериментальных установок, выполненное согласно соответствующим требованиям, включая погрешности сравниваемых параметров и их максимальные отклонения;
- перечень специально подготовленных для верификации экспериментов на натуральных объектах использования атомной энергии, качественное описание этих экспериментов, максимальные отклонения сравниваемых параметров;
- сведения об анализе чувствительности математической модели к различным вариантам разбиения расчетной схемы на элементы, к неточностям в задании исходных данных.

Информационный банк данных экспериментальных исследований и натуральных испытаний зданий и сооружений АЭС на динамические воздействия

Из-за невозможности в каждом случае экспертизы ПС провести комплексные исследования возникла реальная необходимость разработки информационного банка данных экспериментальных исследований.

Был проведен системный поиск отечественных и зарубежных источников, содержащих сведения о проводимых экспериментальных исследованиях (на моделях, фрагментах и натуре), а также отобраны материалы для включения в информационный банк данных о параметрах колебаний и характеристиках напряженно-деформированного состояния зданий, конструкций, систем и элементов. Критерием отбора стала полнота информации для разработки проверочных тестов. Недостаточными для включения в информационный банк считались материалы, на основании которых не могут быть получены исходные данные для выполнения аналогичного исследования численными методами.

Разработанный банк данных содержит сведения общего порядка, конкретные сведения, специальные сведения.

Сведения общего порядка носят информационный характер. Они должны содержать данные, позволяющие осуществлять поиск конкретной и специальной информации.

Конкретные сведения должны быть достаточными для разработки тестов к проведению численных исследований и надежными для доказательства корректности проводимых расчетных исследований.

Специальные сведения служат дополнительной информацией для анализа достоверности физических констант.

В информационный банк включены результаты модельных и натуральных экспериментальных исследований таких объектов, как конструкция в виде коробчатых замкнутых объемов из плит перегородок и перекрытий главного здания АЭС с реактором типа РБМК-1000, массивный железобетонный короб главного здания Армянской АЭС, реакторное отделение блока 1 Крымской АЭС, цилиндрический барабан защитной оболочки АЭС, толстостенная замкнутая с торцов цилиндрическая оболочка на мягком грунте и др. Вся информация записана на машинных носителях.

Результаты разработки изложены в [8].

Разработка тестов для целей аттестации ПС

Для целей аттестации ПС были определены и разработаны эталонные решения с применением аналитических методов.

А. Эталонные решения для расчета системы грунт – фундамент – сооружение при динамическом воздействии (авторы Калинин В.В., Полякова И.Б.) [9].

Предложено три модели фундамента и сооружения в виде жесткого тела, деформируемой плиты и жесткого фундамента.

Предложено две модели основания: однородное упругое; вязкоупругое полупространство. Предложены интегральные уравнения для определения напряжений под фундаментом.

При комбинациях моделей сооружения и основания выполнены расчеты динамики фундамента при нестационарных режимах в двухмерной постановке и пространственной постановке и получены эпюры контактных напряжений, зависимости смещения фундамента во времени. Проведены исследования достоверности результатов путем сравнения промежуточных результатов с аналогичными, полученными другими методами.

Б. Эталонные решения для расчета реакторного отделения атомной станции на сейсмические воздействия с учетом совместной работы с основанием (авторы Сеймов В.М., Трофимчук А.Н.) [10].

Для целей аттестации ПС предложены:

а) методика расчета вертикальных колебаний массива на упругом полупространстве при сейсмических воздействиях;

б) методика расчета горизонтально-вращательных колебаний массива на упругом полупространстве при горизонтальных сейсмических воздействиях;

в) методики реализованы в программу расчета для применения при тестировании ПС;

г) получены решения задач для применения их в качестве тестов.

В. Эталонные решения для трубопроводов и конструкций [11].

Заключение

Опыт проведения аттестации ПС в системе Госатомнадзора России отражен в [12].

В настоящей статье обращается внимание на ряд проблем повышения качества и надежности ПС, а также надежности получаемых с их помощью результатов расчетов прочности и устойчивости конструкций, важных для безопасности объектов использования атомной энергии.

Тестирование и верификация ПС являются важнейшими составляющими обеспечения качества ПС. Поэтому эту деятельность разработчиков ПС необходимо специально планировать, учитывая большие на нее трудозатраты и затраты времени. При этом необходимо руководствоваться программой обеспечения качества ПС. На аттестацию в Госатомнадзор России следует представлять ПС, прошедшие тщательное тестирование и верификацию.

Следует уделять специальное внимание верификации исходных данных для расчета и расчетных схем, в том числе с использованием данных экспериментальных исследований. Эта деятельность не входит в процедуру аттестации ПС в области прочности и устойчивости конструкций, но она требует также планирования и контроля качества. Целесообразно для этих целей разрабатывать программу обеспечения качества результатов расчета с применением аттестованных ПС.

Список литературы

1. Майкс С. Дойч. Верификация и подтверждение правильности. Технология программирования. ВЦП - № РИ-56984. Пер. с англ.
2. Непомнящий В., Рякин О. Прикладные методы верификации программ. М.: Радио и связь, 1988.
3. Ван Тассел. Стиль, разработка, эффективность и испытания программы. М.: Мир, 1985.
4. Бахтизин А., Савкин И. Методы тестирования и верификации программ. М.: Машиностроение, 1984.
5. Саркисян А. Повышение качества программ на основе автоматизированных методов. М.: Радио и связь, 1991.
6. Болонья С., Уренбергер У. Проверка и утверждение программ. Пер. с англ. Киев, 1990.
7. Международные стандарты в области качества программных средств. М.: объединение "Алгоритм", 1990.
8. Проведение экспертных работ по аттестации программных средств, используемых при обосновании объектов атомной энергетики. Отчет / "Экозащита", М.: 1991.
9. Разработка эталонных решений для аттестации программных средств для расчета системы грунт – фундамент – сооружение при динамическом воздействии. Отчет / НИИ механики и прикладной математики РГУ. Ростов, 1991.
10. Разработка эталонных решений для аттестации программных средств для расчета реакторного отделения атомной станции на сейсмические воздействия с учетом совместной работы с основанием системы грунт – фундамент – сооружение при динамическом воздействии. Отчет / Институт гидромеханики АН УССР. Киев, 1991.
11. Разработка и применение методов экспертизы прочности, надежности и сейсмостойкости трубопроводных систем, работающих совместно со строительными конструкциями АЭС, с ис-

- пользованием современных программно-вычислительных средств. Отчет / НТЦ ЯРБ при Госпромоматомнадзоре России. М.: 1991.
12. Калиберда И.В., Ковалевич О.М., Рубцов В.С., Туляков П.В., Уголева И.Р. Об аттестации программных средств и экспертизе расчетных обоснований прочности и устойчивости оборудования и трубопроводов ОИАЭ. // 1-я конференция "Методы и программное обеспечение расчетов на прочность". // Россия, г. Туапсе, 9-14 октября 2000 г.