

**ТЕХНОЛОГИЯ ОРГАНИЗАЦИИ АНАЛИТИКО-ИМИТАЦИОННОГО
МОДЕЛИРОВАНИЯ АСУ АКТИВНЫМИ ПОДВИЖНЫМИ ОБЪЕКТАМИ****С.А. Потрясаев, Б.В. Соколов (Санкт-Петербург)**

Доклад посвящен технологии организации аналитико-имитационного моделирования, применяемого в рамках выполняемого авторами данной статьи проекта. В упомянутом проекте основное внимание уделяется одному из подклассов задачи анализа структурной динамики автоматизированной системы управления активными подвижными объектами (АСУ АПО), а именно задаче расчета, многокритериального оценивания и анализа основных характеристик и показателей качества АСУ АПО.

АПО представляют собой искусственно созданные материальные объекты (приборные комплексы), перемещающиеся в пространстве и осуществляющие взаимодействие (информационное, вещественное, энергетическое) с объектами обслуживания, другими АПО [1-3]. АПО, как показано в работах [1-3], допускает весьма многообразную интерпретацию. Так, например, в качестве АПО может выступать наземное, воздушное, космическое, надводное или подводное средство передвижения с установленной на нем аппаратурой.

На содержательном уровне суть рассматриваемых задач сводится к следующему: известно исходное структурное состояние АСУ АПО, варианты сценариев изменения входных воздействий на элементы и подсистемы АСУ АПО, известны пространственно-временные, технические и технологические ограничения, связанные с процессом ее применения, задан интервал времени. Требуется провести расчет, многокритериальное оценивание и анализ структурной динамики АСУ АПО, в том числе необходимо определить показатели целевых и информационно-технологических возможностей, структурно-топологические показатели, показатели надежности, живучести, устойчивости функционирования основных элементов и подсистем АСУ АПО.

Для решения поставленных задач в ходе выполнения проекта требуется сформировать имитационную систему (ИмС) [4-5]. ИмС – это специальным образом организованный моделирующий комплекс, состоящий из следующих элементов: а) имитационных моделей, отражающих определенную проблемную область; б) аналитических моделей; в) информационной подсистемы, включающей базу данных; г) системы управления и сопряжения, обеспечивающей взаимодействие всех компонент системы и работу с оператором.

Имитационные и аналитические модели, входящие в состав формируемой ИмС, а также алгоритмы и методики расчета, многокритериального оценивания и анализа показателей качества и эффективности функционирования рассматриваемой АСУ АПО были программно реализованы в рамках выполненных ранее работ за последние 20 лет в разных средах и операционных системах, в том числе:

- программный комплекс расчета и многокритериального анализа показателей структурной надежности и устойчивости АСУ АПО (исполняемый модуль для ОС Windows на языке C++);
- программный комплекс расчета показателей пропускной способности АСУ АПО для детерминированных сценариев изменения внешних воздействий (исполняемая оболочка базы данных FoxPro в ОС Windows'98 и исполняемые модули СУБД Clipper для ОС MS-DOS);
- программа расчета и оптимизации показателей робастности и динамической устойчивости программ функционирования АСУ АПО (интерпретируемая программа на языке Python, выполняемая в ОС Linux);

- программный комплекс расчета показателей пропускной способности и ресурсоемкости АСУ АПО для стохастических сценариев изменения внешних воздействий (интерпретируемая программа на языке GPSS, исполняемая в одноименной среде в ОС Windows версии не ниже XP);
- программа расчета показателей эффективности применения АСУ АПО для стохастических сценариев изменения внешних воздействий (реализация аналогична предыдущей);
- программа многокритериального анализа и упорядочения вариантов функционирования АСУ АПО (исполняемый модуль для ОС Windows на C++).

В соответствии с применяемым модульным принципом построения имитационных систем, указанная выше совокупность программ должна быть представлена в виде проблемно-ориентированных вычислительных модулей (подпрограмм), предназначенных для описания отдельных подсистем объекта либо для проведения расчетов определенного типа при решении задач анализа и выбора в рамках ИмС [4-5].

Модульное построение математического обеспечения ИмС оказывается очень удобным, так как позволяет с помощью относительно небольшого числа вычислительных модулей строить большое число вычислительных программ (совокупности вычислительных модулей) для моделирования объекта с различной степенью детализации в рамках одной и той же ИмС.

Наряду со специальным программно-математическим обеспечением (СПМО) важную роль в ИмС играет система управления, сопряжения и интерпретации (СУСИ). В состав СУСИ в общем случае входят локальные системы управления и сопряжения (синтеза) проблемно-ориентированных и стандартных вычислительных модулей для решения конкретных задач моделирования, блоки обработки, анализа и интерпретации результатов моделирования, выработки рекомендаций по организации дальнейшего моделирования, блок формирования сценариев моделирования, диалоговая система управления моделированием.

Еще одной важнейшей компонентой ИмС является ее информационное обеспечение, основу которого составляет база данных (в перспективе база знаний) о предметной области моделирования, включающая в себя сведения об объекте моделирования, среде, в которой он функционирует, сценарии, а также исходные данные для моделирования и его результаты [4-5]. Таким образом, СУСИ и информационное обеспечение совместно обеспечивают согласование моделей по исходным данным.

На этапе практической реализации ИмС в рамках выполняемого авторами проекта возник ряд трудностей, связанных, во-первых, с существенной гетерогенностью применяемого СПМО, во-вторых, с организацией взаимодействия между вычислительными модулями. Перечисленные выше унаследованные программные комплексы представляют собой законченные решения, прошедшие валидацию и верификацию. Разработка подобных систем «с нуля» представляет собой крайне трудозатратный и экономически не обоснованный процесс. В связи с этим требовалось создать СУСИ таким образом, чтобы обеспечить беспрепятственный обмен согласованными исходными данными и выходным результатом между всеми модулями ИмС.

Наиболее хорошо зарекомендовавшим себя подходом к построению модульных систем на сегодняшний день является сервис-ориентированная архитектура (ServiceOrientedArchitecture, SOA) [6]. Данный подход ориентирован на слабую связность, распределенность используемых модулей. В то же время SOA требует использования стандартизированных интерфейсов модулей и работы по стандартизированным протоколам, что изначально не было реализовано в унаследованных программных продуктах.

С учетом обозначенных ранее особенностей используемых унаследованных программных комплексов была предложена следующая архитектура создания ИмС. Во-первых, для преодоления проблем гетерогенности, с одной стороны, и удобства развертывания ИмС, с другой стороны, разместить все модули с несовместимыми требованиями к среде исполнения на различных виртуальных машинах в рамках одного аппаратного сервера. Во-вторых, с учетом перспектив дальнейшего развития ИмС, обеспечить взаимодействие модулей посредством сетевого обмена данными. Для этих целей предлагается создать программные «обертки», преобразующие частную систему ввода-вывода каждого модуля в стандартизированный интерфейс обмена данными. В-третьих, разработать центральную часть СУСИ: программный комплекс, обеспечивающий вызов всех остальных модулей, снабжение их согласованными исходными данными, сбор и интерпретацию результатов. И, в-четвертых, создать интегрированный пользовательский интерфейс, позволяющий удаленно использовать все возможности ИмС.

На сегодняшний день на рынке программных комплексов для создания виртуальных машин (гипервизоров) насчитываются десятки решений. Имитационные и аналитические вычисления обычно сопровождаются высокой нагрузкой на аппаратные ресурсы, поэтому при наличии ограничений на время выполнения моделирования реализация ИмС требует использования как можно более производительного гипервизора. Основываясь на этом, выбор был сделан в пользу решения XenCloudPlatform. Данный гипервизор с открытым исходным кодом использует технологию паравиртуализации, что позволяет приблизить скорость работы виртуальной машины к реальной. Применение версии CloudPlatform, обеспечивающей работу виртуальных машин на нескольких аппаратных серверах с возможностью их «горячей» замены, обосновано необходимостью масштабирования и обеспечения отказоустойчивости создаваемой ИмС.

С учетом унаследованной гетерогенности и вынужденной распределенной архитектуры, в качестве базовой технологии создания ИмС выбрана технология веб-служб. В отличие от аналогичных решений, например, DCOM или CORBA, веб-службы основаны на открытых стандартах и протоколах, а благодаря использованию текстового формата XML для обмена данными достигается простота их разработки и отладки.

Для создания программных «обертки» был выбран язык программирования Python. Выбор обоснован кроссплатформенностью языка, ориентацией его на скорость разработки и читаемость кода.

Описанная ранее СУСИ в терминах сервис-ориентированной архитектуры соответствует понятию сервисной шины предприятия (enterpriseservicebus, ESB). Это связующее программное обеспечение централизованного и унифицированного событийно-ориентированного обмена сообщениями между различными информационными системами. В выполняемом проекте реализация СУСИ основана на свободно распространяемом программном каркасе Zato с открытым исходным кодом. Zato представляет возможность быстрой разработки сервера приложений, реализующего сервисную шину предприятия. Информационная подсистема реализована в виде СУБД PostgreSQL, свободно распространяемой с открытыми исходными кодами. В соответствующей базе данных хранятся наборы исходных данных для имитационного и аналитического моделирования, а также сведения о конфигурациях ИмС и сценариях моделирования. Указанные сведения, как упоминалось выше, позволяют строить различные вычислительные программы для моделирования объекта с различной степенью детализации.

Подсистема управления моделированием, визуализации и интерпретации результатов представляет собой пользовательский интерфейс, реализованный в виде веб-клиента (отображается в браузере). Клиентская часть основана на свободно распространяемом программном каркасе DHTMLX, который предоставляет возможность быстрого создания

Windows-подобных пользовательских интерфейсов из набора готовых графических компонентов. Несмотря на внешнюю схожесть с настольными приложениями, клиентская часть фактически реализована на языках HTML и Javascript, что позволяет использовать ее в большинстве современных операционных систем без необходимости преобразования исходного кода. Для повышения наглядности результатов моделирования применяется свободно распространяемый программный каркас PhiloGL, ориентированный на визуализацию данных на основе развивающейся технологии WebGL.

Взаимодействие интерфейса пользователя с СУСИ на стороне сервера обеспечивается программным каркасом WEB2PY.

Описанная выше архитектура ИмС схематично представлена на рисунке 1.

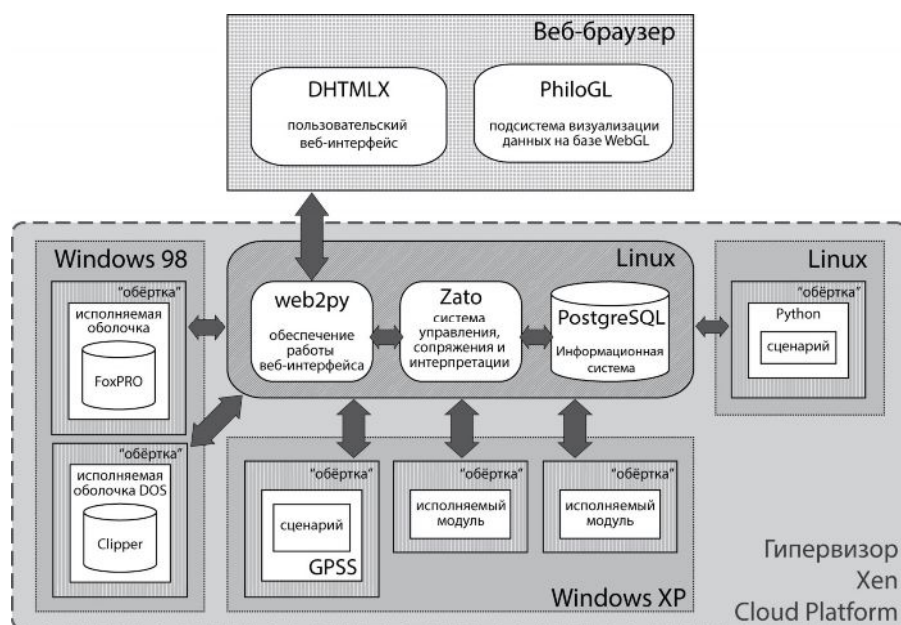


Рис. 1. Архитектура имитационной системы

Выводы

Предложенная архитектура дает ряд преимуществ по сравнению с использованием набора разрозненных приложений. Основные преимущества приведены ниже.

1. Возможность подключения существующих программных комплексов в качестве модулей, реализующих различные классы математических моделей, а также применение режимов сценарной работы СУСИ для проведения автоматических многократных просчетов позволяют повысить обоснованность принимаемых с использованием ИмС решений.

2. Применение единого гибкого настраиваемого веб-интерфейса для всех используемых унаследованных приложений позволяет предоставить оператору (обычно лицу, принимающему решения, специалисту в предметной области) только необходимую информацию на понятном предметном языке. Кроме того, использование 2D и 3D визуализации данных позволяет бороться со сложностью интерпретации результатов моделирования.

3. Использование сервисной шины предприятия в качестве СУСИ предоставляет возможность контролировать ход выполнения моделирования. Каждый предлагаемый системой сервис (в терминологии SOA) может быть представлен несколькими модулями, имеющими свою специфику. Например, для решения одной и той же подзадачи первый модуль способен дать приемлемое решение в кратчайшие сроки, второй модуль готов предоставить точное решение за более длительный интервал времени. В этих условиях

СУСИ имеет возможность самостоятельно выбрать используемый модуль в зависимости от обобщенных предпочтений оператора (например, «считать точно» или «считать быстро»).

4. Заложённая в основу ИмС сервис-ориентированная архитектура обуславливает способность этой системы к развитию. В процессе эксплуатации ИмС в зависимости от складывающейся обстановки может выбирать наиболее пригодный в силу своих свойств тип моделей, входящих в состав ИмС. Использование стандартизированных интерфейсов даёт возможность в будущем постепенно обновлять отдельные модули, не перестраивая ИмС целиком.

5. Использование программных «оберток» предоставляет возможность повторного использования в составе современной гибкой ИмС унаследованного программного обеспечения, имеющего не приемлемый для эксперта в предметной области интерфейс (например, командная строка). Это позволяет в условиях дефицита времени существенно сократить трудозатраты на «реинжиниринг» программного обеспечения, устаревшего внешне, но реализующего классические неизменные математические методы и алгоритмы.

Литература

1. **Калинин В.Н., Резников Б.А.** Теория систем и управления (структурно-математический подход). – Л.: ВИКИ им. А.Ф. Можайского, 1978. – 417 с.
2. **Калинин В.Н.** О задаче оптимального управления активными подвижными объектами // Дифференциальные уравнения. – 1981. – № 12. – С. 2136–2143.
3. **Калинин В.Н., Соколов Б.В.** Оптимальное планирование процесса взаимодействия активных подвижных объектов // Дифференциальные уравнения. – 1985. – С. 752–757.
4. **Охтилев М.Ю., Соколов Б.В.** Теоретические и прикладные проблемы разработки и применения автоматизированных систем мониторинга состояния сложных технических объектов // Тр. СПИИРАН. Отделение информационных технологий и вычислительных систем. – СПб. – 2002. – Вып. 1. – Т. 1. – С.167–180.
5. **Охтилев М.Ю., Соколов Б.В., Юсупов Р.М.** Интеллектуальные технологии мониторинга и управления структурной динамикой сложных технических объектов. – М.: Наука, 2006. – 410 с.
6. **Власов С. А., Девятков В. В., Девятков Т. В.** Универсальная моделирующая среда для разработки имитационных приложений // Информационные технологии и вычислительные системы. – 2009. – № 2. – С. 5–12.