

УДК 004.94+629.58

ПРОГРАММНЫЙ МОДЕЛИРУЮЩИЙ КОМПЛЕКС ДЛЯ АВТОНОМНЫХ ПОДВОДНЫХ АППАРАТОВ НА БАЗЕ МНОГОПРОЦЕССОРНОЙ АРХИТЕКТУРЫ

**С.В. Мельман¹, В.А. Бобков¹,
А.В. Инзарцев^{1,2}, А.М. Павин^{1,2},
А.С. Черкашин³**

Федеральное государственное автономное образовательное учреждение высшего профессионального образования «Дальневосточный федеральный университет»¹
Федеральное государственное бюджетное учреждение науки Институт проблем морских технологий ДВО РАН²
Федеральное государственное бюджетное учреждение науки Институт автоматики и процессов управления ДВО РАН³

Моделирование в реальном времени ресурсоемких задач, связанных с отладкой алгоритмов «интеллектуального» управления автономными подводными роботами (АПР), влечёт необходимость разработки программных моделирующих комплексов с распределёнными вычислениями. В описываемом комплексе исследования алгоритмов ведутся путём имитационного моделирования широкого спектра бортовых датчиков, сенсоров, динамики АПР и физических процессов подводной среды. Отличительной особенностью комплекса является возможность проведения отладки алгоритмов непосредственно в среде системы управления выбранного АПР. Архитектура программного комплекса организована с использованием клиент-серверной технологии и централизованной БД с возможностью подключения в качестве вычислительных узлов как персональных компьютеров, так и супер-ЭВМ. Используемый модульный подход обеспечивает возможность расширения алгоритмической и функциональной базы за счет подключения дополнительных модулей по технологии plug-in. Описываются функциональные возможности комплекса, схема взаимодействия модулей в условиях распределенных вычислений, протоколы передачи и хранения данных с учетом специфики моделируемых процессов. Результаты тестирования комплекса на разных вычислительных конфигурациях демонстрируют возможность моделирования процессов управления АПР в реальном времени при больших вычислительных нагрузках.

1. Введение

Работы по созданию имитационных моделирующих комплексов (МК) начались с появлением первых рабочих подводных роботов, решающих практические задачи. Замена реальных испытаний виртуальными стала одним из решений по ускорению и удешевлению разработки программного интеллекта (алгоритмов управления) автономных подводных роботов (АПР).

Использование имитационных моделирующих комплексов предполагает решение ряда практических задач, которые можно в общем случае определить как изучение поведения одного или нескольких роботов в различных условиях. Такая постановка дает свободу выбора путей решения

проблемы. Большинство моделирующих систем можно классифицировать по используемым методам имитационного моделирования, набору решаемых задач, степени реалистичности отдельных процессов и др. К основным решаемым задачам можно отнести моделирование:

- одного или нескольких АПР;
- коммуникаций между различными АПР (с учетом времени задержки, зашумленности, дальности, пропускной способности);
- внешних объектов (рельеф, статические и динамические объекты);
- окружающей среды (течения, температура – в статике, динамика);
- датчиков (сонары, фото-, видеокамеры и др.).

Кроме того, системы моделирования характеризуются:

- различными уровнями моделирования (НП, real-time, off line);
- возможностью реализации распределенных вычислений;
- предоставлением удобного интерфейса визуализации полученных результатов.

В зависимости от потребностей того или иного исследовательского коллектива реализуются необходимые функциональные возможности для достижения различных целей.

¹ 690091, Владивосток, ул. Суханова, 8. Тел: (423) 275-44-24. E-mail: melman.sv@dvvfu.ru

² 690091, Владивосток, ул. Суханова, 5а. Тел.: (423)221-55-45. E-mail: inzar@marine.febras.ru

³ 690041, Владивосток, ул. Радио, 5. Тел.: (423) 231-37-76. E-mail: alexandr@cherkashin.ws

В ранних работах [1, 2] описываются моделирующие комплексы, решающие конкретные задачи: испытание действующей модели подводного робота или использование комплекса как тренажера операторов телеуправляемых роботов. Коллектив авторов [3] в 1997 году сделал попытку обобщить цели имитационных моделирующих комплексов. Авторами предложены идея универсального МК и подход к визуализации результатов моделирования с учетом тенденций того времени и технологий для многопользовательских интерфейсов на базе VRML. Уже тогда стало ясно, что процессы моделирования и визуализации результатов можно и даже необходимо разделять. В работе [4] даётся довольно полный обзор современного состояния в области моделирующих комплексов, а также предложен метод классификации МК по функциональным возможностям. Там же делается вывод об отсутствии подходящего МК для решения поставленной задачи. Далее формулируются новые требования к МК, которые заключаются в возможности моделирования миссии с несколькими аппаратами одновременно и необходимости подключения реальных аппаратов с помощью специальных интерфейсов в среду симулятора (Hardware-In-The-Loop /HIL/). Затем авторы описывают разработку симулятора Thetis, особенностью которого является использование вычислительных мощностей нескольких ЭВМ, объединенных сетью для моделирования сенсоров аппарата. Таким образом, подводится черта под одомашинными моделирующими комплексами, которые просто не в состоянии справиться со все возрастающими требованиями к реализму моделирования и количеством бортового оборудования, особенно в режимах HIL.

Еще одной современной разработкой является проект USARSim [5] с акцентом на точную симуляцию среды передачи акустического сигнала между группой подводных роботов и водолазов для координации совместных действий. В этой работе также используются многомашинные вычисления, но не обеспечивается полный универсализм, который позволял бы легко добавлять новые возможности и расширять набор моделируемых датчиков. Это замечание относится и к некоторым другим разработкам, не упоминаемым в данном кратком обзоре. Таким образом, можно отметить, что важной характеристикой МК является возможность его функционирования в режиме реального времени в контексте выполняемых операций. Учитывая то, что современные АПР могут оснащаться большим набором сенсорной аппаратуры, целесообразно разрабатывать универсальные МК на базе высокопроизводительных архитектур с организацией параллельно-распределенной обработки данных.

Предлагаемый МК является развитием предыдущей работы авторов по созданию комплекса, эксплуатируемого на одной ЭВМ [6, 7, 8]. Комплекс создавался с ориентацией на моделирование программно-аппаратной части системы управления АПР разработки ИПМТ ДВО РАН. К отличительным особенностям разработанного ранее комплекса можно отнести широкий спектр моделируемых данных, а также возможность проведения отладки разрабатываемых алгоритмов непосредственно в среде системы управления выбранного АПР. Модульная архитектура позволяла наращивать алгоритмическую базу и расширять ряд моделируемых сенсоров, но со временем стало

понятно, что мощности одного персонального компьютера (ПК) недостаточно для имитационного моделирования всего спектра бортового оборудования в режиме реального времени. Например, задача моделирования процесса инспекции подводных коммуникаций на основе обработки информации от гидролокатора бокового обзора [9] решалась в десятки раз медленнее реального времени. Оснащение АПР новыми сенсорами предполагает появление дополнительных возможностей для реализации «интеллектуального» поведения АПР. Использование многолучевых эхолотов или гидролокаторов переднего обзора (рис. 1, 2) потенциально позволяет решать на борту АПР задачи по распознаванию и классификации объектов или препятствий на акустических снимках с организацией последующей адекватной реакции робота на эти события. Однако возросшие объемы информации от подобных сенсоров ещё больше обостряют проблему имитационного моделирования всего спектра бортового оборудования в режиме реального времени с использованием вычислительных мощностей одного ПК. Кроме того, режим реального времени необходим для подключения бортового оборудования АПР к имитационному комплексу в режиме HIL.

Таким образом, возникла необходимость создания версии МК с



Рис. 1. АПР ММТ-2012 с установленным многолучевым эхолотом GeoSwath



Рис. 2. АПР «Пилигрим». Подобный аппарат предполагается оснастить гидролокатором переднего обзора

многопроцессорной архитектурой, позволяющей реализовать параллельно-распределенные вычисления. В свою очередь, это потребовало разработки нового подхода к архитектуре МК, развития многопользовательского интерфейса, новых функциональных возможностей и соответственно применения новых технологий программной реализации. Решению этих задач и посвящена настоящая статья.

2. Функциональные возможности МК

Применение МК значительно упрощает решение задач по разработке, исследованию и отладке алгоритмов и методов, используемых в интеллектуальной начинке робота. Примерами таких задач могут служить:

- управление движением АПР;
- автоматическая навигация, в том числе решение задачи SLAM (Simultaneous Location and Mapping);
- машинное зрение, распознавание объектов на ГБО (гидролокатор бокового обзора), ГСО (гидролокатор секторного обзора), ГПО (гидролокатор переднего обзора), БГБО (батиметрический ГБО), ЛЧМ ГБО (ГБО с линейно-частотной модуляцией), МЛЭ (многолучевой эхолот) и на

бортовых видеокамерах с целью использовать эти результаты при картографировании, поиске затонувших объектов, движении вдоль протяженных объектов, обследовании пирсов, судов и т.п.;

- “интеллектуальное поведение” (обход препятствий, перепланирование миссий, отработка внештатных ситуаций).

Это возможно благодаря имитации всех бортовых систем подводного аппарата и окружающей среды в МК. Другие функциональные возможности позволяют:

- использовать МК в качестве тренажерного комплекса для операторов АПР;
- задавать и исследовать миссии АПР с учетом возникновения нештатных ситуаций и обстоятельств;
- тестировать работоспособность оборудования и бортового ПО подводного аппарата при его подключении к виртуальной среде моделирующего комплекса в режиме НЛ.

Система предоставляет наряду с возможностями моделирования и управления миссиями также развитые инструменты визуализации результатов моделирования, позволяя выбрать любой момент миссии для изучения как информации с сенсоров, так и визуальной информации в 3D. Наличие централизованной БД позволяет реализовать многопользовательский доступ к результатам моделирования для совместного анализа результатов моделирования миссии несколькими операторами.

3. Предлагаемая архитектура МК и его функционирование

Процесс моделирования робота в среде описывается как простая циклическая последовательность «действие робота» – «реакция сре-

ды», выполняемая в течение каждого кванта времени. «Действие робота» делится на последовательные этапы: восприятие среды сенсорами, интеллектуальная обработка показаний сенсоров, принятие решения и выработка управляющих сигналов, движение и/или другая деятельность. В этой последовательности только последний пункт «движение» тесно связан с «реакцией среды», поэтому появляется возможность рассматривать некоторые этапы моделирования как независимые, с возможностью параллельного выполнения. Если речь идет о моделировании группы роботов, то обмен информацией между роботами происходит через «реакцию среды». Иными словами, для одного робота все остальные являются частью среды, из которой робот извлекает данные с помощью сенсоров. В этом случае обработка действий роботов в группе может происходить параллельно, так же, как и моделирование набора сенсоров на борту одного робота. Это обусловлено тем, что сенсоры не влияют друг на друга в течение данного кванта времени, они лишь создают «возмущение» в среде, которое повлияет на другие сенсоры в следующий квант времени.

Описанная общая схема моделирования позволяет учитывать все современные требования к моделирующим комплексам, встречающиеся в литературе.

Если рассматривать моделирование миссии как некоторый технологический процесс, то в целом МК может быть представлен как SCADA-система (supervisory control and data acquisition). В состав системы входит БД (база данных) для выполнения следующих задач:

- интеграция в единую структуру хранилища всех данных о техническом процессе,

независимо от их источника, вида или времени внесения в базу;

- получение немедленного доступа к упорядоченной по времени информации;
- извлечение исторических данных в сочетании с информацией о конфигурации, событиях и итоговыми данными.

Каждый пункт приведенной выше последовательности операций МК содержит множество процессов, которые могут выполняться параллельно:

1) восприятие среды сенсорами: каждый сенсор оперирует только данными о собственной конфигурации и окружающей среде, поэтому моделирование работы сенсоров идет независимо и может идти параллельно;

2) интеллектуальная обработка показаний сенсоров, извлечение более высокоуровневой информации из показаний сенсоров: распознавание образов, задачи позиционирования и т.д. могут идти параллельно для сенсоров или групп сенсоров;

3) принятие решения о действиях на базе собранной информации (отработка миссии);

4) выработка управляющих сигналов на двигатели и другие исполнительные устройства согласно миссии: 3- и 4-й пункты не имеют ярко выраженного потенциала для распределенных вычислений;

5) движение и/или другая деятельность по возмущению среды;

6) реакция среды, динамические перемещения объектов сре-

ды, роботов, изменение параметров среды: параллельно могут обрабатываться разные параметры среды, расчеты течений, распространения акустических сигналов и т.д.

В основу предыдущей версии МК был заложен модульный принцип, согласно которому все функциональные блоки были реализованы в качестве отдельных программных динамических модулей (бортовые сенсоры, расчет движения, управление и т.д.). Необходимые динамические модули при моделировании подключались к МК автоматически. Такой подход позволил наращивать алгоритмическую базу без изменений основного приложения программного комплекса.

В архитектуре описываемой версии МК этот подход находит своё дальнейшее развитие. В целом архитектура характеризуется использованием следующих подходов:

- распределенные вычисления – для обеспечения вычислений в реальном времени;
- клиент-серверная модель – для централизованного управления комплексом;
- база данных (БД) – для централизации и унификации потоков данных;
- plugin-технология;
- гибридный параллелизм (GPGPU+CPU) в функциональных блоках.

Указанные решения позволяют в значительной степени использовать предыдущие наработки,

включая модули динамики и бортовых сенсоров. Клиент-серверная часть МК состоит из трех функциональных программных приложений (рис. 3).

- Сервер – обеспечивает связь клиентов с базой данных и супервизором, который координирует работу всей распределенной системы.

- Диспетчер – части распределенной системы, обеспечивающие обмен данными между отдельными модулями и сервером (супервизором). Эти приложения располагаются на различных ПК, объединенных сетью, и выступают посредниками между отдельными модулями и сервером, объединяя весь комплекс в единую распределенную систему. Также диспетчеры содержат копию данных БД последнего этапа моделирования.

- Интерфейс – программное приложение, позволяющее пользователю управлять процессом моделирования, визуализировать результаты в режиме реального времени и в режиме записи. Также интерфейс выступает в качестве терминала для ручного управления в режиме тренажера.

Архитектура распределенного МК позволяет работать одновременно нескольким пользователям (например, при реализации режима «учитель-ученик» при работе МК в качестве тренажёра оператора [10]). Интерфейс для каждого пользователя может быть сконфигурирован индивидуально, что позволяет одновременно нескольким операторам с разных ПК наблюдать за ходом миссии или анализировать сохраненные результаты моделирования независимо друг от друга.

Реализация программного обеспечения позволяет запускать приложения также и на кластере. Пользователь самостоятельно выбирает модули, которые будут

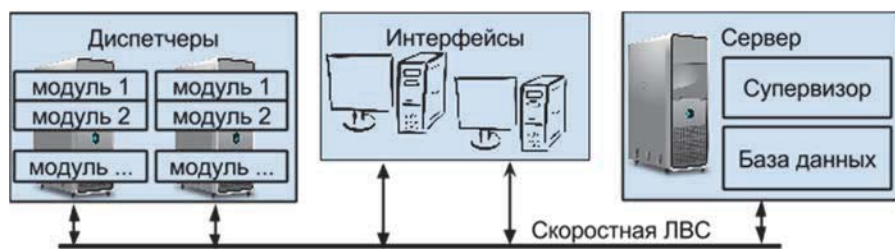


Рис. 3. Схема распределенного моделирующего комплекса

работать в составе приложений-диспетчеров, таким образом распределяя вычислительную нагрузку на узлы.

Функционирование МК происходит в три этапа. На первом этапе выполняется конфигурация логической структуры комплекса. Для этого необходимо определить рабочую схему комплекса (используемые ПК и соответствующие приложения), затем выбрать и запустить сервер, диспетчеры и интерфейсы пользователя. Пользователь обеспечивает диспетчеры набором необходимых модулей для моделирования. На втором этапе с использованием интерфейса пользователя выполняется загрузка файлов миссии на сервер, а также загрузка всех необходимых данных о среде моделирования и подводном аппарате. На этом этапе делается автоматическая проверка на наличие необходимых вычислительных и моделирующих модулей в системе, иначе моделирование невозможно. На третьем этапе пользователь запускает процесс моделирования и может наблюдать за ним в режиме реального времени. На мониторе пользователя могут отображаться общие виды подводной обстановки, показания видеокамер, сенсоров, датчиков, бортовых приборов и т.д. Конфигурацию видов датчиков пользователь определяет по собственному усмотрению. Во время выполнения миссии все данные моделирования заносятся в БД и после завершения миссии доступны для анализа. Пользователь может просмотреть результаты на любой момент времени как видеозапись, перемещаясь по временной шкале и запуская воспроизведение.

Помимо трех основных функциональных программных приложений в состав МК входят также дополнительные программные приложения для конфигурирова-

ния аппарата и создания макета подводной обстановки. Они носят вспомогательный характер, так как конфигурационные файлы хранятся в текстовом формате XML, что позволяет их легко редактировать в любом текстовом редакторе.

3.1. Сервер МК

Серверная часть комплекса реализована в программном приложении, объединяющем в себе базу данных и супервизор.

База данных используется для централизованного хранения результатов моделирования. От начала моделирования в МК запускается таймер “выполнения миссии”, как если бы реальный аппарат работал в реальной среде – время модели. За каждый такт модельного времени в БД заносится новый вектор состояния АПР (положение, скорости, ориентация), полученный при помощи модуля расчёта динамики аппарата.

Датчики и бортовые приборы условно можно разделить на два типа по периоду обновления данных: высокоскоростные и медленные. Высокоскоростные (например, датчик глубины) могут сообщать свои показания в любой момент времени. Их показания заносятся в БД на каждом такте моделирования. Для получения измерений от медленных датчиков требуется время на формирование отклика. Эти данные заносятся в БД в момент получения показаний с меткой “времени модели” их получения. В данном случае имеется в виду время формирования отклика, которое бы потребовалось реальному сенсору в реальных условиях, а не время, затраченное ПК на моделирование. К медленным сенсорам можно отнести, например, все виды сонаров. В базе данных хранится также информация об окружающей среде и о по-

ложении динамических объектов среды в каждый момент времени. Таким образом, сохранение состояния виртуальной среды и АПР позволяет производить как дальнейший анализ хода выполнения миссии, так и обмен информацией между распределенными модулями МК.

Для унификации все виды хранимых в БД данных преобразуются к одномерному байтовому массиву – пакету данных, размеры которого зависят от типа моделируемого датчика. Формат этих пакетов зафиксирован для каждого датчика и может быть расшифрован любым модулем МК. Извлечение нужной информации из БД происходит посредством драйвера, который обеспечивает универсальный интерфейс обмена данными между модулями МК, независимо от типа и структуры БД (по “идентификатору данных” и “времени их получения”). Если информация запрошена от медленного датчика на момент времени, когда отклик еще не сформирован, то из БД извлекается его последнее показание.

Супервизор координирует работу распределенной системы, выполняя на каждом такте следующую последовательность действий:

- 1) увеличение счетчика тактов и запрос на моделирование перемещения АПР, моделирование динамических объектов среды;
- 2) формирование списка опрашиваемых датчиков, запрос и ожидание окончания моделирования датчиков от всех диспетчеров на данном такте;
- 3) работу алгоритмов интеллектуального управления АПР и формирование сигналов управления.

Все действия выполняются последовательно, но модули моделирования на каждом из этих шагов

могут работать независимо. Таким образом, можно добиться максимальной производительности, распределяя при конфигурировании логической структуры МК модули, относящиеся к одному шагу, между разными диспетчерами.

3.2. БД и протокол обмена данными

Как было сказано выше, все данные в БД заносятся в виде последовательности байтов с метками времени и указателями на источник получения. Таким образом, БД является 2D таблицей, в ячейках которой хранятся массивы байтов. Программная реализация такой БД отличается простотой и высокой скоростью извлечения данных из индексированных массивов. Данные в БД делятся на два типа:

- статические, не меняющиеся на протяжении всего процесса моделирования (визуальные модели среды, АПР, миссия, данные конфигурации среды и др.);
- динамические (получаемые в процессе моделирования).

Протокол обмена данными построен на протоколе TCP/IP с гарантированной доставкой и целостностью данных. Схема формирования пакетов данных состоит из сериализации данных от сенсора с пометкой времени получения данных и метки сенсо-

ра, от которого данные получены. Схемы сериализации закрепляются за каждым типом сенсора или бортового прибора при разработке соответствующего модуля.

Некоторые типы бортового оборудования формируют большое количество данных (см. таблицу). Для таких датчиков в БД реализован механизм сохранения данных, полученных в последний момент времени, а все предыдущие показания сохраняются на жесткий диск. При этом наряду с экономией оперативной памяти сохраняется возможность визуального анализа пользователем результатов моделирования. При такой организации хранения данных за один час моделирования в оперативной памяти накапливается около 1,5 Гб данных, что не является проблемой для современных ПК.

3.3. Взаимодействие модулей с сервером

3.3.1. Диспетчеры

Диспетчеры выступают посредниками между модулями и сервером МК. Во время запуска диспетчер устанавливает связь с сервером, получает из сервера файлы миссии, конфигурации АПР и среды, файлы данных об аппарате и среде, в соответствии

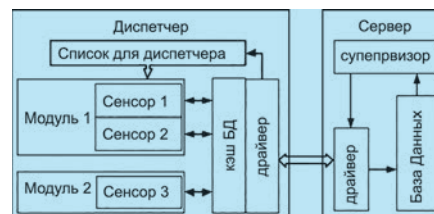


Рис. 4. Схема обмена данными между модулями МК

с конфигурационными файлами инициализирует необходимые подключаемые модули. Также диспетчеры предоставляют модулям интерфейс для обмена данными и обеспечивают и автоматическое восстановление связи при разрывах в передаче данных между сервером и БД.

На рис. 4 показана схема обмена данными в МК. В диспетчере все запросы к драйверу идут через кэш БД, что позволяет минимизировать количество обращений к БД и разгрузить локальную сеть. Динамические записи удаляются из кэша на каждом такте моделирования.

Работа диспетчера происходит следующим образом. Супервизор формирует список датчиков и модулей, которые должны использоваться на очередном шаге, и рассылает всем диспетчерам сигнал о начале очередного шага. Диспетчер посредством драйвера получает этот список (один список для всех диспетчеров) и, сверяясь со списком, выбирает ассоциированные с ним модули, передавая им управление по очереди. Модули возвращают результаты работы на сервер в БД через драйвер. Таким образом, если запись в БД обновилась, то супервизор считает, что модуль и соответствующий датчик отработали, и переходит к следующему шагу.

Список выполняется последовательно, что позволяет все ресурсы ПК сосредоточить на одном модуле для более эффективного параллелизма в рамках этого модуля.

Расчет объема данных, передаваемых в БД (1 такт = 0,01 с)

Тип данных	Байт за передачу	Передач в секунду	Количество приборов	Байт/с	~Мб/ч
Вектор состояния АПР	48	100	1	4800	16
ГБО	1000	5	2	10000	34
ВЧ ГБО	6250	5	2	62500	225
ЛЧМ ГБО	132000	2	2	528000	1800
МЛЭ	—	—	—	—	800
Цветные видекамеры (jpg) 640×480×15 fps	~92160	15	2	~2764800	~10000

3.3.2. Интерфейсы

Всё взаимодействие пользователя с МК происходит через клиентское программное приложение. Для исключения коллизий один клиент управляет моделированием, в то время как остальные могут подключаться только в режиме просмотра результатов моделирования. Пользователь запускает интерфейсное программное приложение, подключается к серверу и выбирает сконфигурированную миссию, после чего на сервер автоматически загружаются все файлы данных, необходимые для моделирования миссии. Далее, если текущая конфигурация МК удовлетворяет требованиям миссии, то может быть начат процесс моделирования. В рамках пользовательского интерфейса можно просматривать данные о текущей миссии, конфигурации АПР и бортовых датчиков. В центральном окне интерфейса можно просматривать вид подводной обстановки, реализующий два режима наблюдения:

- свободный полет (пользователь может свободно перемещаться по подводному пространству);
- следование за АПР (виртуальная камера следует за подводным аппаратом и всегда направлена на него, ракурс выбирает пользователь).

Для отображения данных от датчиков на данном этапе разработаны шесть видов виджетов-мониторов: стрелочный, цифровой, график изменения параметра во времени, виджет для ГБО, виджет для ГСО, виджет для фото и видео.

В ходе выполнения или анализа миссии пользователь выбирает отображаемые сенсоры, распола-

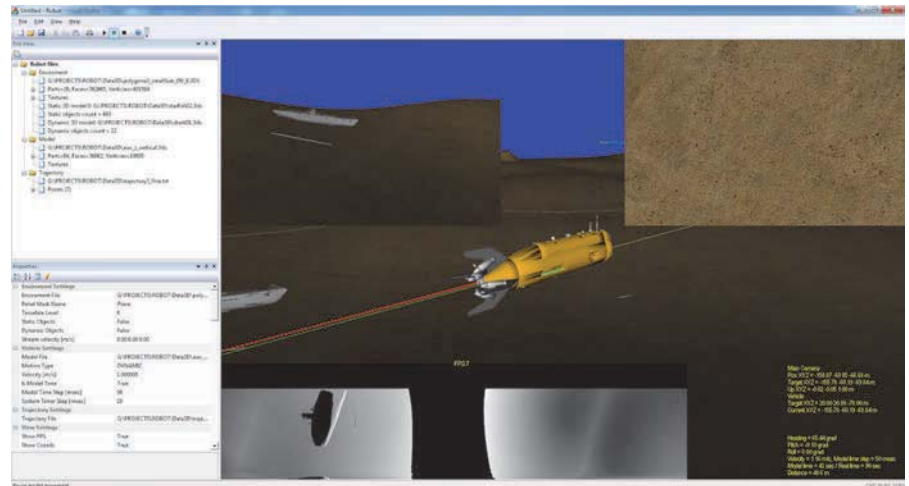


Рис. 5. Вид интерфейса пользователя

гает виджеты на экране и задает параметры отображения.

На рис. 5 показана возможная конфигурация интерфейса пользователя. В левой части отображается информация о дереве и параметрах проекта. В правой части даётся общий вид подводной обстановки в режиме слежения за аппаратом. Отображается также информация с двух бортовых видеокамер, внизу выводятся данные ГБО, а справа внизу – текстовая информация о параметрах камеры, положении и ориентации АПР, пройденном пути и др.

3.4. Функциональные модули

3.4.1. Модуль сетевого обмена – драйвер

Драйвер предоставляет функции по обмену данными между модулями и хранилищем данных, выполняя пересылку данных по сети и обращения к БД. Соединение устанавливается по протоколу ТСР/ПР. Передача данных по сети происходит в асинхронном режиме. Для передачи данных модуль передает идентификатор моделируемого сенсора и сериализован-

ные данные драйверу. Драйвер пересылает по сети эту информацию супервизору. Супервизор прикрепляет к пакету данных от сенсора метку времени и помещает в БД. Для получения данных из БД модуль по идентификатору данных получает пакет данных, сохраненных в БД на предыдущем шаге.

3.4.2. Видеокамеры и фотокамеры

Модуль, реализующий работы видео- или фотокамеры, запрашивает из БД модель окружающей среды, положение АПР в абсолютной системе координат и положение фотоаппарата на борту робота в связанной системе координат. Также запрашивается информация о параметрах устройства из конфигурации АПР. Это делается для установки виртуального окна сенсора в абсолютной системе координат, после чего происходит рендеринг подводной обстановки. При необходимости, для повышения реалистичности, происходит обработка фильтрами для имитации шума, оптических искажений, бликов и так далее. После обработки фильтрами изображение помещается в БД.

3.4.3. Сонары

Моделирование сонаров подробно описано в статье [6]. Алгоритм реализован в виде модулей, имитирующих на данном этапе три вида сонаров: однолучевой дальномер, гидролокатор бокового обзора и гидролокатор секторного обзора.

Имитация работы виртуального сонара основана на принципе, схожем с имитацией фотоаппарата, однако вместо наложения фильтров используется пересчет Z-буфера в строку откликов. Из строки откликов выбирается ближайший превышающий заданный порог отклик и преобразуется в расстояние. Далее эти данные передаются в БД через драйвер. Модуль ГБО передает в БД всю строку откликов. ГСО, помимо вычисления положения в абсолютной системе координат, также делает поворот виртуального окна сенсора. При этом угол поворота зависит от времени, прошедшего с момента старта миссии.

3.4.4. Имитация среды передачи данных

Для имитирования среды передачи данных используется схема, позволяющая как упрощенное моделирование распространения сигналов в водной среде, так и физически более точное и реалистичное. Передающая сторона в каждый момент времени оставляет в БД пакет данных, которые необходимо передать вместе с точным положением и параметрами передающей антенны (известными из данных о конфигурации и динамике АПР). Принимающая сторона, зная положение источника, данные об окружающей среде и параметры принимающего устройства, вносит в исходные

данные шум и искажения, присущие приемопередающим приборам и среде распространения. Таким образом, можно имитировать как простые затухания акустических сигналов подводных модемов, так и более сложные передающие устройства. В простейшем случае, если расстояние между передающей и принимающей антенной превышает некоторый предел, то сигнал считается утерянным. В сложных случаях подобная схема позволяет учитывать не только параметры модемов, но и скорость движения АПР, рельеф местности, температуру, соленость, течения воды, имитировать частичную потерю сигнала и специфическое искажение информации. Такой подход обеспечивает возможность отладки высокоуровневых алгоритмов передачи данных между модемами.

3.4.5. Навигационные датчики

Датчики глубины, акселерометры, компасы и лаги для простоты имитируются, основываясь на данных о положении и ориентации робота в среде, которые точно известны из данных о движении АПР. Для повышения реалистичности в показаниях этих приборов могут вноситься искажения, шумы или имитироваться поломки.

3.4.6. Моделирование динамики АПР

Модуль моделирования движения робота выполняет расчеты при известных управляющих величинах движительного комплекса. Для реалистичного моделирования создается уникальный модуль каждого аппарата, реализующий его динамические характеристики и особенности конструкции [7, 8, 11], хотя существуют и универ-

сальные математические модели подводных аппаратов [12, 13].

3.5. Файлы конфигурации

Моделирование поведения АПР в виртуальном мире требует знаний о геометрии и свойствах объектов мира, о “сенсорах” робота и о взаимовлиянии мира и АПР. Настройка всех этих взаимозависимостей происходит в текстовых файлах XML, описывающих структуру объектов и задействованных в моделировании их свойств. Под структурой подразумевается иерархическая взаимосвязь объектов (например, дно, на котором расположены камни и подводное сооружение, а на сооружении находятся прикрепленные телеметрические буйки). Под свойствами понимаются их геометрическая форма, заданная полигональными сетками, их геометрические атрибуты, цвет, текстура, отражательные способности, температура, если требуется. Также задаются свойства и самой окружающей среды: температура, соленость, течения и т.д.

Конфигурирование требуется для виртуальной среды (рельеф, объекты окружающей среды, их атрибуты и т.д.), аппарата (геометрическая модель, набор бортовых датчиков и их расположение) и датчиков (тактико-технические характеристики). Модели этих объектов с достаточной степенью достоверности описываются иерархической древовидной структурой. На верхнем уровне иерархии находится “проект моделирования” – часть описания, необходимая для объединения на следующем уровне равнозначных узлов “среда”, “аппарат”, “миссия”. Был рассмотрен и альтернативный вариант иерархии, с узлом “миссия” на верхнем уровне. То

есть “миссия” АПР разрабатывается для конкретного “аппарата” и “среды”. Но подобный подход не позволит разрабатывать миссии, не привязанные к конкретным аппаратам и местности. Примером может быть миссия обследования дна с фотографированием и картографированием с использованием ГБО. Таким образом, разработанная миссия может быть применена для любой местности, любого аппарата, потенциально способного выполнить эту миссию, то есть обладающего фотоаппаратом и ГБО.

4. Тестирование МК

Для оценки эффективности распределенного подхода был проведен эксперимент с размещением МК на двух ПК с одинаковыми характеристиками, соединенными локальной сетью со скоростью передачи 100 Мб/с. Параметры используемых ПК: Intel i5-760 @ 2,8 ГГц NVIDIA GTX-470. На первом ПК был запущен сервер, интерфейс пользователя и диспетчер. На втором ПК был запущен диспетчер. Производительность распределенного МК сравнивались с производительностью однопроцессорной версии МК. Исходные данные: виртуальная сцена, один АПР, оснащенный двумя каналами ГБО с включенной моделью зашумления, пятилучевой эхолокационной системой (ЭЛС) и одной цветной видеокамерой. Шаг моделирования 0,01 с. Детализация виртуальной сцены ~800 000 треугольников. Каждый из моделируемых датчиков производит полный рендеринг сцены. Затвор видеокамеры срабатывает 15 раз в секунду, сигнал дальномеров и ГБО обрабатывается 5 раз в секунду. Вычислительная нагрузка между диспетчерами была распределена

следующим образом: первый ПК моделирует работу видеокамеры, ГБО левого борта и два луча ЭЛС, второй моделирует ГБО правого борта и три луча ЭЛС. В результате вычислительных экспериментов было установлено, что в однопроцессорной версии МК на моделирование 100 с миссии потребовалось 263 с машинного времени, что не соответствует режиму реального времени. Распределенному МК на двух ПК потребовалось 164 с, т.е. увеличение аппаратной производительности системы в 2 раза повысило скорость расчетов в ~1,6 раза (эффективность 62%), но не обеспечило режима реального времени.

Второй эксперимент был проведен на конфигурации, состоящей из одного ПК и супер-ЭВМ SMH11 с 8 вычислительными узлами. Конфигурация одного узла SMH11: 2×CPU Xeon L5506, 2.13GHz, 2×NVIDIA Tesla M2050 GPU computing processor– 3 GB, 32 Gb оперативной памяти, Linux (Gentoo)+x-server, сеть со скоростью передачи 1 Гбит/с. На ПК был запущен сервер МК и интерфейс пользователя. На кластере задействовано 3 узла, на которых было запущено по 1 диспетчеру с распределением нагрузки: 1-й узел – видеокамера и 2 ГБО, 2-й узел – три луча ЭЛС, 3-й узел – два луча ЭЛС. На моделирование миссии потребовалось около 80 с, что отвечает режиму реального времени. При использовании всех 8 вычислительных узлов – по одному датчику на узел – время вычислений сократилось до 40 секунд, в этом варианте МК значительная часть времени затрачивается на обмен данными по сети.

5. Заключение

Тестирование МК на вычислительных комплексах различных

конфигураций подтвердило правильность принятых архитектурных решений и дало обнадёживающие результаты в реализации режима реального времени при моделировании работы бортовых сенсоров АПР. Дальнейшее развитие комплекса видится в следующих направлениях:

- оптимизация архитектурной и сетевой структуры комплекса для уменьшения временных затрат, связанных с обменом данными (например, других протоколов передачи данных);
- развитие методов моделирования перспективных акустических сенсоров;
- развитие подходов к интеграции МК с модифицированной событийно-ориентированной программной средой системы управления АПР;
- отладка алгоритмов “интеллектуального поведения” с использованием новых возможностей, предоставляемых МК.

Предполагается публикация серии статей, посвященных решению указанных выше проблем.

Работа выполнена при частичной финансовой поддержке РФФИ (проект № 15-07-00341), Программы РАН №4 по стратегическим направлениям развития науки «Фундаментальные проблемы математического моделирования» (проект «Разработка и адаптация алгоритмов и математического обеспечения на супер-ЭВМ с экстремальным параллелизмом для решения задач моделирования и прогнозирования чрезвычайных ситуаций на море с применением данных спутникового мониторинга»), Программы «Дальний Восток» (проект 15-I-4-025), а также гранта Российского научного фонда № 14-50-00034 (в постановочной части исследования).

ЛИТЕРАТУРА

1. Kuroda Y., Aramaki K., Ura T. AUV test using real/virtual synthetic world // Proc. of IEEE Conf. on Autonomous Underwater Vehicle Technology. 1996. June 3–6, Monterey, CA. URL: <http://ieeexplore.ieee.org/iel3/3780/11039/00532436.pdf>, accessed on 2014/Dec/08.
2. Choi S.K., Yuh J. Design of advanced underwater robotic vehicle and graphic workstation // Proc. of IEEE Intern. Conf. on Robotics and Automation, 1993. May 2-6, Atlanta, GA. URL: <http://ieeexplore.ieee.org/iel2/904/7227/00292131.pdf>, accessed on 2014/Dec/08.
3. Gracanin D., Valavanis K.P., Matijasevic M. Virtual environment testbed for autonomous underwater vehicles // Proc. of 8th Intern. Conf. on Advanced Robotics. 1998. 7-9 July, Monterey, CA. URL: <http://ieeexplore.ieee.org/iel3/4880/13463/00620272.pdf>, accessed on 2014/Dec/08.
4. Parodi O., Lapiere L., Jouvencel B. Hardware-in-The-Loop Simulators for multi-vehicles scenarios:survey on existing solutions and proposal of a new architecture' // Proc. of the IEEE/RSJ Intern. Conf. on Intelligent Robots and Systems. September 2009. St Louis, MO, USA, 2009.
5. Sehgal A., Cernea D. A multi-AUV missions simulation framework for the USARSim Robotics Simulator // Proc. of 18th Mediterranean Conf. on Control & Automation (MED). 23–25 June 2010. Marrakech, 2010. P. 1188–1193.
6. Бобков В.А., Морозов М.А. Моделирование работы гидролокационных устройств средствами машинной графики // Подводные исследования и робототехника. 2012. № 1(13). С. 47–51.
7. Bobkov V.A., Borisov Yu.S., Inzartsev A.V., Mel'man S.V. Simulation program complex for studying motion control methods for autonomous underwater vehicles // Programming and Computer Software. 2008. Vol. 34, N. 5. P. 257–266.
8. Бобков В.А., Морозов М.А., Багницкий А.В., Инзарцев А.В., Павин А.М., Щербатюк А.Ф., Туфанов И.Е. Имитационный моделирующий комплекс для обследовательского автономного подводного робота // Научная визуализация. 2013. Т. 5, № 4. С. 47–70.
9. Багницкий А.В., Инзарцев А.В., Павин А.М., Мельман С.В., Морозов М.А. Модельное решение задачи автоматической инспекции подводных трубопроводов с помощью гидролокатора бокового обзора // Подводные исследования и робототехника. 2011. № 1(11). С. 17–23.
10. Инзарцев А.В., Сидоренко А.В., Сенин Р.А., Матвиенко В.Ю. Комплексное тестирование программного обеспечения АПР на базе имитационного моделирующего комплекса // Подводные исследования и робототехника. 2009. № 1(7). С. 9–14.
11. Автономные подводные роботы. Системы и технологии / Агеев М.Д., Киселев Л.В., Матвиенко Ю.В. и др.; под общ. ред. акад. М.Д. Агеева. М.: Наука, 2005. 398 с.
12. Fossen, T.I. Guidance and control of ocean vehicles. Chichester: John Wiley & Sons Ltd., 1994. 494 p.
13. Киселев Л.В., Медведев А.В. Сравнительный анализ и оптимизация динамических свойств АНПА различных проектов и конфигураций // Подводные исследования и робототехника. 2012. № 1(13). С. 24–35.

