

ВЫЧИСЛИТЕЛЬНЫЕ ЭКСПЕРИМЕНТЫ В РЕАЛЬНОМ ВРЕМЕНИ, РАЗДЕЛЕНИЕ МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ С АЛГОРИТМАМИ ИНТЕРАКТИВНОГО УПРАВЛЕНИЯ И ВИЗУАЛИЗАЦИИ

А. Б. Дегтярев^{1,2,}, В. Н. Храмушин^{2,**}*

¹ Санкт-Петербургский государственный университет, Санкт-Петербург, Россия

² Научно-инженерное общество судостроителей им. А. Н. Крылова,
Санкт-Петербург, Россия

При постановке и проектировании вычислительного эксперимента представляются актуальными вопросы интерактивного управления ресурсоемкими алгоритмами. Вместе с тем важна возможность динамической перенастройки моделей с независимым визуальным контролем трехмерных физических явлений и процессов в реальном масштабе времени. Прямой вычислительный эксперимент позволяет достигать практических инженерных решений без традиционных аналитических ограничений. Оптимизационным условием являются лишь объем пространственных сеток и точность аппроксимации изучаемых инженерных объектов.

The issues of interactive control of resource-intensive algorithms seem to be topical in the formulation and design of simulation. The possibility of dynamic reconfiguration of models with independent visual control of three-dimensional physical phenomena and processes in real time is important. Direct simulation allows one to achieve practical engineering solutions without traditional analytical limitations. Only the volume of spatial meshes and the accuracy of approximation of the studied engineering objects is an optimization condition.

PACS: 89.20.Ff; 07.05.Tr

ВВЕДЕНИЕ

Современные вычислительные системы во многом обеспечивают прямое имитационное моделирование процессов и явлений для ситуаций, когда физическое описание не вызывает сомнений [1]. В этом случае возможно проведение численных экспериментов в реальном времени взамен физических экспериментов с той же степенью адекватности в по-

* E-mail: a.degtyarev@spbu.ru

** E-mail: Khram@mail.ru

лучении результатов. При этом преимущество такого подхода очевидно. Во-первых, отсутствует масштабный эффект по сравнению с модельным экспериментом. Во-вторых, эксперименты проводятся в условиях точного знания состояния объекта и внешней среды. В-третьих, условия проведения эксперимента могут быть обеспечены любые, включая экстремальные, их нельзя достигнуть в рамках натурального эксперимента по соображениям безопасности или в рамках модельного эксперимента в силу ограниченных возможностей экспериментальных установок.

В то же время проведение любого эксперимента подразумевает возможность его планирования, управления им, вмешательства в ход его проведения, изменения тех или иных параметров, рассмотрения более подробно, как через лупу, отдельных компонентов, в некоторых случаях повторения эксперимента с определенного момента и т. д. Будем рассматривать данный вопрос на примере задач гидроаэродинамики. В случае прямого имитационного моделирования накладываются определенные требования на сами модели и алгоритмы моделирования [2–4].

Таким образом, фактически речь идет о разработках особых вычислительных моделей, которые без каких-либо противоречий сводятся к строго определенным комплексам функциональных объектов и алгоритмических операций и которые в случае успеха претендуют на особое описание физических явлений [2]. Соответственно, потребуются и особые языковые средства, пригодные для строгого описания результатов, нацеленных на их последующее теоретическое и технологическое развитие [3].

В целом вычислительный эксперимент образуется тремя условно независимыми компонентами [4]:

- разделением решений по физическим процессам в математическом представлении процессов гидромеханики (физики);
- независимым от вычислительной среды многооконным инструментарием графической визуализации моделируемых явлений и процессов;
- системой синхронизации во времени с интерактивным управлением параметрами вычислительных моделей.

Эти три логически независимых компонента вычислительного эксперимента алгоритмически связаны. Далее рассматриваются особенности проектирования и реализации вычислительных экспериментов для интерактивного моделирования в реальном времени.

РАЗДЕЛЕНИЕ ВО ВРЕМЕНИ ВЫЧИСЛИТЕЛЬНЫХ ПРОЦЕССОВ С ИНТЕРАКТИВНЫМ ИНТЕРФЕЙСОМ

При реализации вышеизложенных задач допустимо разделение программно-технических разработок по трем условно независимым направлениям.

1. *Собственно математическое моделирование*: задействуется только арифметико-логический блок с множеством вычислительных ядер

и общим доступом к большим объемам оперативной памяти для быстрого считывания и модификации.

2. *Автоматическое управление* вычислительным экспериментом осуществляется в процедурах обработки прерываний по одному или группе интервальных таймеров.

3. *Графическая визуализация результатов* организуется и выполняется в интерактивном взаимодействии с внешней аппаратной периферией, включающей прерывания от таймеров, курсорказателей и клавиатуры компьютера.

При таком раскладе допускается изначальная ориентация на построение предельно эффективных алгоритмов собственно численного моделирования.

Традиционный вычислительный эксперимент выстраивается по моделированию процессов развития и трансформации сложных физических явлений во времени, где каждый такт вычислений безусловно мал. Обработка прерываний от таймеров, так же как и от внешней периферии, может выполняться в порядке очередности в свободные от моделирования интервалы времени.

Всегда есть особенности в работе процедур графической визуализации, в которых при параллельном выполнении задействуются массивы данных из оперативной памяти, которые одновременно модифицируются в ходе вычислительного эксперимента. Но обычно это не приносит больших искажений в текущие результаты.

Существенно сложнее происходит согласование внутри процедур интерактивной обработки прерываний от внешних периферийных устройств. В случае изменения графических сцен или перестройки локальных фрагментов изображения вычислительный эксперимент может не приостанавливаться. Однако при изменении параметров вычислительного эксперимента, особенно при полной смене состояния моделируемой окружающей среды, может потребоваться либо моментальная приостановка моделирования, либо полный перезапуск всего вычислительного эксперимента.

КОМПЛЕКС ПРОЦЕДУР ИНТЕРВАЛЬНОГО ТАЙМЕРА

Рассмотрим две реализации интерактивного управления вычислительным экспериментом.

Вариант А. Основное внимание обращено на исполнение ресурсоемкого эксперимента. Обычно такой вариант характерен для ситуации, когда интерактивное управление параметрами математических моделей является первичным.

Вариант Б. Ориентирован на работу с группой таймеров. Этот вариант оптимален для постановки численного моделирования в реальном времени.

В варианте А подстройка графической визуализации текущих результатов реализуется по необходимости. Формально здесь достаточно одного

интервального таймера (WaitTime). Интерактивное управление ходом эксперимента и графическими процедурами выполняется последовательно в моменты приостановки собственно численного моделирования.

```
long WaitTime(
    long wait,           // delay for independent interrupt processing
    bool( *inFree )() = null, // free function of the computational experiment
    long work = 0 );    // control time to execute the calculation cycle [ms]
```

Для динамического управления вычислительными процессами, к примеру для оптимизации величин интервалов времени wait и work, в программной среде доопределяется прямой доступ к компьютерным часам.

```
long StartTime,      // computer start time of the whole program
RealTime,           // current execution time of the inFree procedure within
                    WaitTime
GetTime(),          // query the exact time in milliseconds (GetTickCount)
ElapsedTime();     // querying the program runtime (µsec)
```

Управляющие операции, в том числе позиционирование графической сцены OpenGL, могут выполняться просто с клавиатуры, тем не менее хотя бы одно графическое окно должно быть предварительно открыто и активировано.

```
byte Window::WaitKey() // stop and waiting of new symbol from keyboard
byte Window::GetKey()  // querying and selecting a symbol without stopping
                        the program
byte Window::ScanKey() // symbol polling without stopping and without
                        sampling from the queue
byte Window::ScanStatus() // getting associated keys code from buffer
```

Вариант В. Применяемый инструментарий ориентирован в большей степени на параллельные потоки для математических моделей, графическую визуализацию и интерактивное управление. Здесь возможно задействование окон в качестве классов объектно-ориентированного программирования.

Такое же разделение требуется для независимой настройки контекстуально зависимого графического пакета OpenGL с аналогичным обособлением физических явлений и процессов собственно вычислительного эксперимента.

```
virtual Window& Window::Timer() // virtual procedure for timer processing
Window& Window::SetTimer( µsec, bool( *inFree )()=null ) // interval
                                                           and transaction
Window& Window::KillTimer() // timer reset
```

Здесь процедура Window::SetTimer также активирует виртуальное обращение к процедуре обработки прерываний Window::Timer() непосредственно внутри числового объекта, что корректно поддерживает автоматическую обработку алгоритмов пролога и эпилога по началу и завершению реентерабельных процедур и делает возможным согласование по логическим флагам асинхронного исполнения прерываний от различных внешних устройств. Процедура Window::KillTimer() прекращает цикл

прерываний интервального таймера, предварительно обеспечив выполнению всех текущих прерываний внутри для активной программы в целом.

Аналогичным способом осуществляется обработка прерываний от внешних устройств.

ПРИМЕРЫ ИСПОЛЬЗОВАНИЯ СРЕДЫ ПРОГРАММИРОВАНИЯ ДЛЯ ВЫЧИСЛИТЕЛЬНОГО ЭКСПЕРИМЕНТА

1) При постановке вычислительного эксперимента в области корпускулярной механики не возникает необходимости согласования физического времени с компьютерными часами. Алгоритмы оптимизируются на предельно эффективные и быстрые вычисления, без потери возможностей детальной визуализации моделируемых физических процессов, с поддержанием быстрой реакции на внешние интерактивные управляющие воздействия.

Прямой вычислительный эксперимент с детальной прорисовкой подвижных пространственных объектов полезен для наглядной интерпретации свойств и особенностей моделей физических явлений, в том числе не проверяемых точными аналитическими решениями. Интерактивное управление вычислительным экспериментом позволяет оценивать адекватность математических моделей, динамически определять области существования физических явлений и устойчивости механических процессов (рис. 1).

2) Построение прямого вычислительного эксперимента для моделирования, к примеру, штормовой гидромеханики корабля принципиально требует согласования всех операций с реальным временем, что в большей

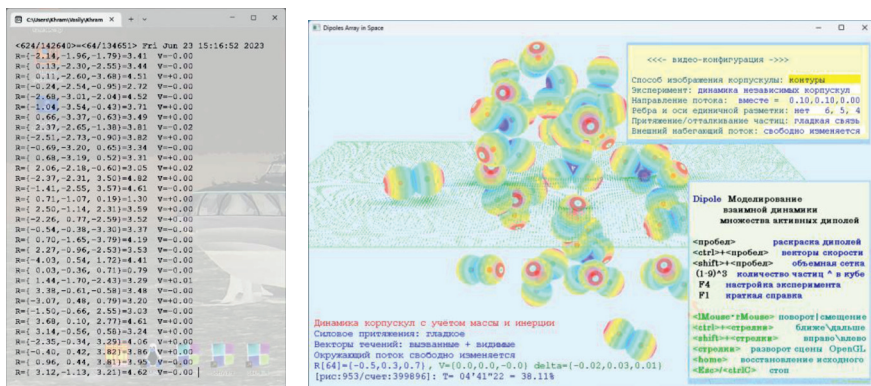


Рис. 1. Облако подвижных и активно взаимодействующих поляризованных частиц. Справа — меню с параметрами вычислительного эксперимента, ниже — подсказка по командам управления; слева — контрольные параметры установления решения, отображаемые без активизации графической среды

степени обуславливается проверкой характера результатов моделирования с представлениями хорошей морской практики. Предполагается, что проведение подобных экспериментов в области штормовой мореходности корабля образует практический полигон для опробования и отработки

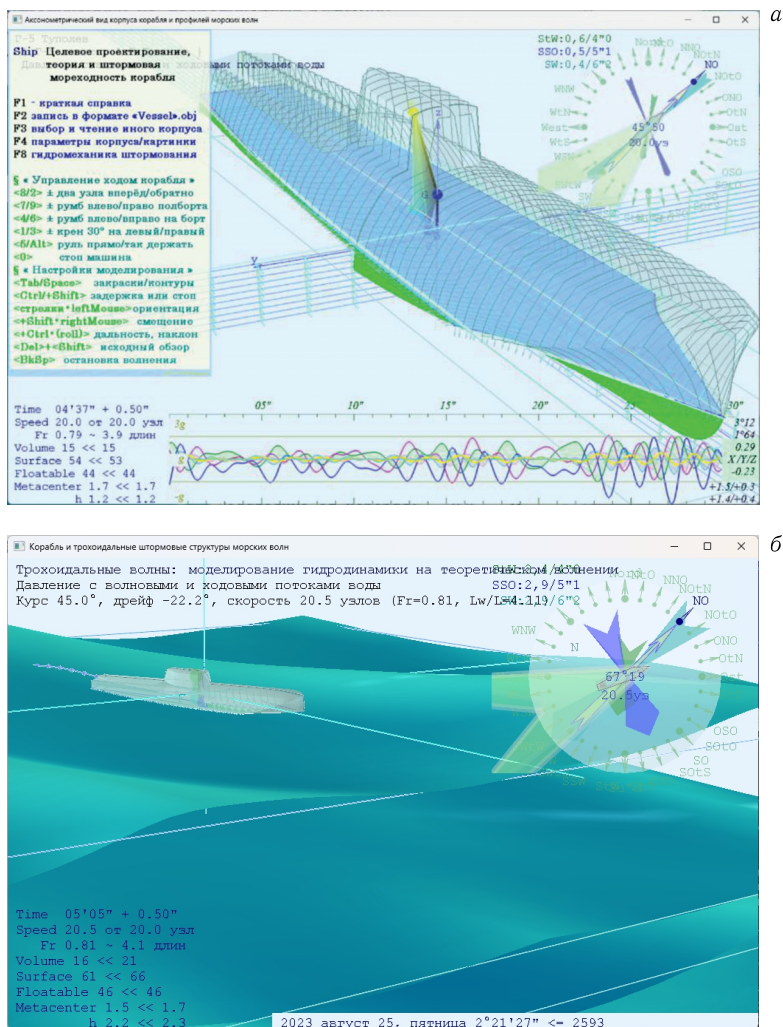


Рис. 2. Представление штормового моря и числовых параметров контроля состояния корабля в локальной корабельной системе отсчетов. а) Слева — подсказки для интерактивного управления экспериментом, ниже — числовая и графическая информация по гидромеханике маневрирования. б) Маневрирующий корабль в представлении относительно морской неподвижной системы координат

маневров в штормовом море, а также для наполнения базы знаний бортовой интеллектуальной системы. Возможно применение для чисто инженерных изысканий по оптимизации формы корпуса и общекорабельной архитектуры в интересах целевого проектирования флота.

В настоящей реализации (рис. 2) на одном графическом поле отображается моделируемый корабль с оценками гидродинамического состояния, в другом графическом окне — окружающее корабль морское волнение с указанием курса, скорости, ходового дрейфа и положения руля, что требуется для интерактивного управления штормовым маневрированием. На текстовой консоли обновляются основные параметры проведения вычислительного эксперимента. В каждом графическом экране действуют обособленные системы интерактивного управления с помощью курсорказателя, клавиатуры и контекстных меню. С каждым графическим окном связываются контекстные подсказки.

ЗАКЛЮЧЕНИЕ

Таким образом, в работе сформулированы минимально допустимые и достаточные инструментальные средства для эффективной постановки и объективного контроля хода и результатов вычислительного эксперимента. Рассмотрены минимально необходимые инструментальные средства для наглядного контроля адекватности выполнения математических моделей. С учетом архитектуры суперкомпьютеров в проектировании экспериментов заложены требования выполнения расчетов и визуализации результатов в реальном времени. С учетом возможностей параллельного и быстрого выполнения графических процедур в среде OpenGL предложены варианты прямых и виртуальных интерактивных интерфейсов. Новый инструментарий необходим для анализа адекватности математического моделирования в локальных подобластях.

Работа выполнена при частичной поддержке СПбГУ (проект NP_GZ_2021-3).

СПИСОК ЛИТЕРАТУРЫ

1. *Bogdanov A., Degtyarev A., Gankevich I., Khramushin V., Korkhov V.* Virtual Testbed: Concept and Applications // LNCS. 2020. V. 12254. P. 3–17.
2. *Bogdanov A., Khramushin V.* Tensor Arithmetic, Geometric and Mathematic Principles of Fluid Mechanics in Implementation of Direct Computational Experiments // Eur. Phys. J. Web Conf. 2016. V. 108, No. 02013.
3. *Degtyarev A., Khramushin V.* Coordinate Systems, Numerical Objects and Algorithmic Operations of Computational Experiment in Fluid Mechanics // Ibid. No. 02018.
4. *Degtyarev A., Khramushin V., Shichkina J.* Tensor Methodology and Computational Geometry in Direct Computational Experiments in Fluid Mechanics // AIP Conf. Proc. 2017. V. 1863, No. 11006.