



## КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ

УДК 004.021:004.946

### О РАСПАРАЛЛЕЛИВАНИИ ВЫЧИСЛИТЕЛЬНОГО ПРОЦЕССА ФОРМИРОВАНИЯ ПОВЕРХНОСТИ ЖИДКОСТИ В РЕЖИМЕ РЕАЛЬНОГО ВРЕМЕНИ

**Е. Ю. ТОРГОНИН**

*Белгородский  
государственный  
технологический  
университет  
им. В.Г. Шухова*

*e-mail:  
e.yudek@gmail.com*

В статье предлагается метод распараллеливания вычислений выполняемых для формирования геометрии поверхности жидкости, в режиме реального времени на высоко распараллеленных вычислительных системах. Проведённый анализ показал, что данный метод демонстрирует высокую производительность и обеспечивает возможность применения уравнений физически обоснованного моделирования жидкости в режиме реального времени на современных вычислительных системах. Приведены результаты эксперимента. Данные могут использоваться при создании технических систем, в компьютерных симуляторах, обучающих программах и мультимедиа приложениях.

Ключевые слова: формирование поверхности жидкостей, высоко распараллеленные вычислительные системы, шумовые функции Перлина, физически обоснованное моделирование, системы реального времени.

В связи с быстрым развитием электроники, постоянным высоким ростом производительности вычислительных устройств и сферы информационных технологий в целом, требования пользователей в самых различных областях давно превзошли физические возможности одного процессора, работающего по принципу одна задача в один момент времени. Особенно данное положение вещей относится к области компьютерного моделирования физических процессов в системах виртуальной реальности, таких как различные симуляторы и тренажёры. Поэтому одной из самых актуальных проблем в настоящее время является проблема реализации массовых параллельных вычислительных процессов.

Физически обоснованное моделирование динамики жидкостей со свободной поверхностью в режиме реального времени с достаточно высокой детализацией и точностью невыполнимо в настоящее время. Следовательно, необходимо детализировано и качественно, с высокой точностью моделировать только те участки среды, где учёт мелких деталей действительно нужен для получения реалистичного результата (например, морской прибой или след за движущимся кораблём). Большинство океаниче-



ских волн возникают и перемещаются достаточно независимо под влиянием ветра, но на них могут оказывать влияние и другие факторы, например течения.

В данной работе описан разработанный мною метод распараллеливания вычислительного процесса формирования поверхности жидкой среды. Метод учитывает широко используемое упрощение [3] для представления жидкости состоящее в том, что за исключением больших и высоких волн, а также прибоев, водная поверхность может быть представлена как двумерный объект или сетка. Сетка задаётся через карты высот и нормалей, что является самым простым и высокопроизводительным решением для любого современного графического ускорителя (GPU). Причём графическому ускорителю могут передаваться лишь контуры данной поверхности, а высокая детализация достигается за счёт тесселяции поверхности с использованием возможностей самих GPU [1]. Для наиболее оптимальной реализации описываемой методики с использованием таких устройств, как современные многоядерные центральные процессоры (CPU) и GPU, необходимо создавать текстуры низкого разрешения для хранения волновой модели поведения жидкости. В данной работе при анализе эффективности разработанного метода используется стохастический метод Тессендорфа [3]. При использовании самого метода [3] мы не сможем моделировать в режиме реального времени действительно большие пространства из-за его большой вычислительной сложности. Если говорить о симуляторах, то это пространства, выходящие за пределы области видимости наблюдателя/пользователя, например сцены с моделью распространения волн цунами на большие расстояния.

Основываясь на представлении жидкости как рекурсивной среды и опираясь на текстуры, созданные с помощью выбранного подхода [3], как восстанавливаемые функции, предлагаю использовать генерацию нескольких октав шума Перлина [1] с различными значениями амплитуд и частот волн. Полученные данные необходимо объединить с данными, полученными с применением физически обоснованной или стохастической моделью поведения волн. Итоговое значение каждого элемента карты поверхности жидкости (узла сетки) рассчитывается по формуле:

$$f(x) = \sum_{i=0}^{N-1} \alpha^i \cdot f(2^i \cdot x), \quad (1)$$

где  $\alpha$  - амплитуда волны, а  $f(2^i \cdot x)$  либо результат расчёта шумовой функции, либо метода Тессендорфа при заданной частоте волны  $2^i$ . Комбинируя несколько таких карт высот в пространстве и времени, мы можем получить достаточно сложную и реалистичную мультипликацию поверхности.

Основная идея генерации шума Перлина заключается в получении для заданной точки псевдослучайного шума. Для этого находим четыре соседние узловые точки, получаем псевдослучайные значения шума в них, выполняем интерполяцию для получения результата в искомой точке пространства.

Данная идея хорошо подходит для создания так называемых «юбок» на границах секторов (рисунок 1.б), создаваемых при разбиении поверхности на отдельные участки с целью распределить вычисления по множеству вычислительных устройств. Различные сектора могут формироваться с различной точностью и соответственно скоростью. Получаемые значения в соседних узлах сетки могут существенно отличаться (рисунок 1.б: между блоками типов А, В, С). Данный эффект получающихся «швов» необходимо сгладить, а полученный шум используется в дальнейшем для получения реалистичного изображения.

Одни текстуры, отвечающие за высокую детализацию, должны обновляться каждый кадр, другие нет. Можно вынести сложные вычисления в отдельные потоки на центральном процессоре или GPU, проводя вычисления асинхронно и подменяя участки полученных карт по мере необходимости (рис. 1).

Таким образом, вся поверхность представляет собой множество связанных сегментов или секторов, количество которых увеличивается пропорционально количеству

ву вычислительных устройств и размеру самой поверхности (рисунок 1.б). На рисунке 1.а показана сетка поверхности жидкой среды. Если позиция наблюдателя находится в самом центре, то поверхность можно разделить на сектора, которые могут находиться в одном из трёх состояний:

1) Не активен - сектор расположен достаточно далеко от наблюдателя и в ближайшее время, скорее всего, не будет визуализироваться. В такой области необходимо асинхронно выполнять только основные, общие вычисления модели распространения волн на сетке с низким разрешением (например, распространения волны цунами), если таковые имеются.

В вычислительной системе приоритет потоков, выполняющих вычисления в секторах с таким состоянием, низкий и вычисления выполняются значительно реже и возможно с меньшей точностью, чем на других секторах.

2) Готов – сектор расположен вне поля зрения наблюдателя, но в самое ближайшее время может стать видимым, что требует выполнения всех вычислений, которые будут необходимы для достоверной визуализации.

Вычисления в секторах с данным состоянием так же асинхронны, выполняются без просчёта визуальных эффектов и с меньшей точностью.

3) Сектора данной области видимы, а каждый из них будем считать активным. Активные в данный момент времени сектора находятся в поле зрения наблюдателя и должны быть визуализированы вне зависимости от того, просчитана ли физически обоснованная модель поведения жидкости или нет. Если корректная модель не рассчитана, то используются шумовые функции.

Вычисления синхронизированы, так как в реальном времени необходимо рассчитывать не менее 30 кадров в секунду. Для формирования поверхности жидкости и визуализации выделяются потоки с высоким приоритетом.

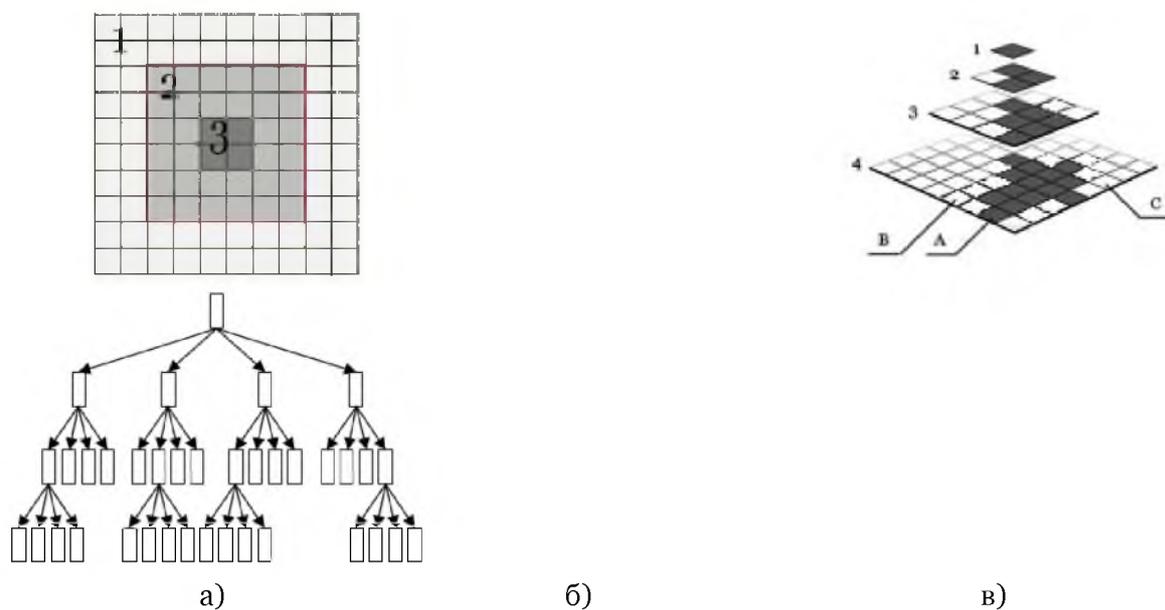


Рис. 1. Асинхронное вычисление модели поведения поверхности жидкости

Физически обоснованное формирование поверхности жидкости очень сложный и ресурсоёмкий процесс, поэтому может занимать большое количество времени и ресурсов процессоров, при этом в большинстве случаев не подчиняется общей системе синхронизации, так как вычисления могут продолжаться достаточно длительное время. Данный случай показан на рис. 2, где потоки 1 и 2 требуют времени больше, чем

выделено на обработку 1 кадра. Мною разработана система со свободным шагом для реализации параллельных вычислений. Это означает, что не требуется синхронизация по всем потокам по завершении каждого следующего кадра и часть потоков получает право на продолжение трудоёмких вычислений, но в результате испытаний выяснилось, что данная задержка не должна продолжаться более 2-3 кадров. В противном случае накапливаются большие ошибки.



Рис. 2. Синхронизация и потоки со свободным шагом выполнения

Проведенные мною исследования показали, что для поставленной задачи формирования поверхности жидкости на больших открытых пространствах оптимальной методологией проектирования системы планирования и диспетчеризации потоков, выполняющих вычисления модели поведения на различных секторах, является подход с минимизацией взаимоблокировок [2] или так называемый lock-less подход. Подход подразумевает сведение к минимуму объёма разделяемых ресурсов и соответственно сводит к минимуму вероятность «гонок».

Все потоки работают со своей локальной копией данных и сохраняют результаты в отдельной памяти, сообщая планировщику о завершении работы, который в свою очередь, сообщает всем заинтересованным системам о необходимости обновить данные по новому адресу.

Данный подход позволяет при небольшом перерасходе памяти, получить иногда существенный прирост производительности за счёт экономии времени на работу средств синхронизации повсеместно используемых в традиционных lock-based методиках.

Расчёт физической модели поведения предлагается возлагать на некоторое множество процессоров, доступное на используемой системе, для чего создаётся специальный пул потоков. А данные, поступающие на обработку, разбиваются на задачи (рис. 3), которые в свою очередь сами могут иметь подзадачи (рис. 1.в).

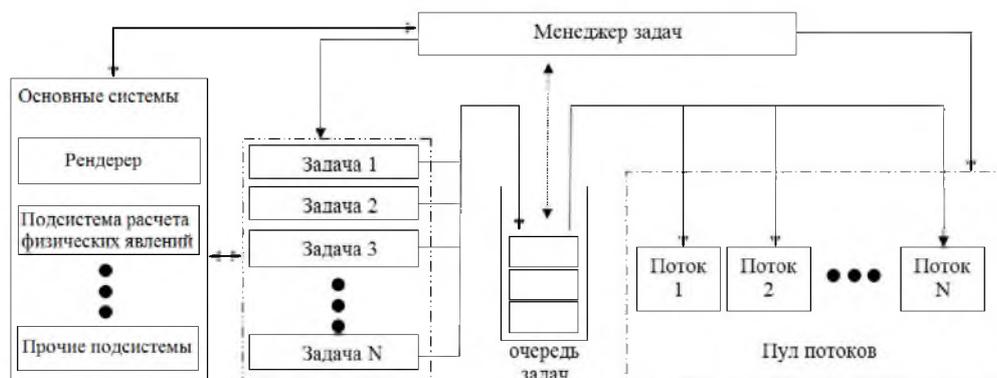


Рис. 3. Организация процесса многопоточных вычислений



Синхронизация доступа к подзадачам осуществляется только при записи/чтении очереди задач. Очередь становится разделяемым ресурсом. Проведенные мною исследования показали, что прирост производительности составляет до 15% по сравнению с lock-based методиками при решении поставленной задачи.

В итоге, для поставленной задачи представленный метод достаточно эффективен и позволяет создавать большие и хорошо детализированные сцены при приемлемой скорости визуализации в режиме реального времени.

Были получены результаты, показывающие высокую эффективность данного подхода на DirectX 11 совместимых устройствах, а так же системах с аппаратной поддержкой платформы CUDA или Stream при разбиении карты поверхности на сектора размером 256x256 пикселей и выводом на экран с разрешением 1024x768 пикселей.

При анализе использовался компьютер, оборудованный 4-ядерным процессором Intel Core2Quad E6600 2.4 GHz, графическим ускорителем ATI HD4850 и объёмом оперативной памяти в размере 4 Gb. При использовании центрального процессора, как основного вычислительного устройства производительность снижается примерно в 10 раз, по сравнению с использованием GPU по средствам интерфейса OpenCL.

В таблице 1 показано количество кадров в секунду выполняемых при визуализации жидкости (поверхности океана) с геометрической сеткой размером 256x256 вершин. Как видно из таблицы, использование lock-less подхода позволяет повысить производительность. Такое решение хорошо масштабируется и позволяет использовать произвольное количество N потоков исполнения, без необходимости переписывания существующего кода, не опасаясь проблем взаимоблокировок.

Таблица 1

**Анализ эффективности предложенного подхода**

Общий размер карт поверхности	GPU		CPU(2 потока)		CPU(4 потока)	
	Lock-based	Lock-less	Lock-based	Lock-less	Lock-based	Lock-less
8192x8192	~3	~3	~ 0	~ 0	~ 0	~ 0
4096x4096	~7	~7	~ 0	~ 0	~ 0	~ 0
2048x2048	~26	~27	~ 1	~ 1	~ 1	~ 1
1024x1024	~47	~49	~ 1	~ 2	~ 2	~ 2
512x512	~74	~78	~ 5	~ 5	~ 7	~ 8

Использование средств GPU, как основного вычислительного устройства, позволяет достигнуть наиболее высокой эффективности за счёт большого числа вычислительных блоков.

На основании проведенных исследований можно указать преимущества предложенного подхода:

- 1) Ориентированность на современное аппаратное обеспечение, многопоточность и многопроцессорные системы.
- 2) Высокая реалистичность результата в реальном масштабе времени.
- 3) Высокая производительность.

Данный подход может использоваться в следующих областях:

- 1) Кинематография и мультипликация.
- 2) Компьютерные тренажёры и симуляторы.
- 3) Обучающие программы.
- 4) Мультимедиа приложения и компьютерные игры.

С применением данного подхода мы получаем возможность реалистичного и физически обоснованного формирования поверхности жидкости в реальном масштабе времени, что невозможно сделать, используя прочие подходы ориентированные либо на зрелищность результата в реальном времени, либо на точность, но без возможности применения в приложениях реального времени: симуляторах, мультимедиа и обучающих программах.



Следует отметить, что технологии аппаратной поддержки многопоточной визуализации и моделирования, как на CPU, так и на GPU продолжают бурно развиваться и имеется достаточно большое поле для дальнейших исследований, в том числе и при взаимном использовании средств CPU и GPU.

### Литература

1. Торгонин Е. Ю. Моделирование динамики движения жидкостей и оптика на высоко параллельных вычислительных системах // Математические методы в технике и технологиях – ММТТ-23: сб. трудов XXIII Междунар. науч. конф. - Саратов: Саратов. гос. техн. ун-т, 2010. - Т.9. - С. 154-157.
2. Andrews J. Designing the Framework of a Parallel Game Engine [Электронный ресурс] / J. Andrews. – Режим доступа: <http://software.intel.com/en-us/articles/designing-the-framework-of-a-parallel-game-engine/> - Дата обращения: 01.02.2011.
3. Tessendorf J. Simulating Ocean Water [Электронный ресурс] / J. Tessendorf. – Режим доступа: <http://graphics.ucsd.edu/courses/rendering/2005/jdewall/tessendorf.pdf>. - Дата обращения: 01.02.2011.

## **ABOUT MULTITHREADED COMPUTING PROCESS OF CREATION OF THE SURFACE GEOMETRY OF A LIQUID IN REAL-TIME**

**E. Y. TORGONIN**

*Belgorod State Technology  
University named after  
V.G. Shukhov*

*e-mail:  
e.yudek@gmail.com*

In this paper is developed the method of the organization multithreaded calculations for creation of the surface geometry of a liquid in real-time. The made analysis has shown, that the developed method shows high efficiency and gives an opportunity of use physically-based liquid modeling in real time. Results of experiment are presented. Results can be used at creation of technical systems, in the computer simulators, training programs and multimedia applications.

Key words: creation of the surface geometry of a liquids, multithreaded computer systems, Perlin noise, physical based modelling, real-time systems.