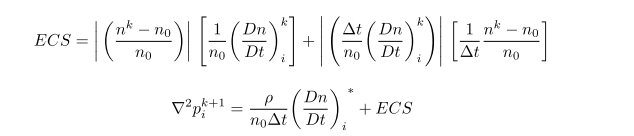
**3.3.3 Numerical stability**

Khayyer and Gotoh [36] came up with a PPE’s source term with error-compensating parts to enhance even further pressure and velocity ﬁeld calculations. The compensating parts should be measures for instantaneous and accumulative violations of ﬂuid incompressibility. subsubsection 17 shows the new terms, and Equation 18 shows the complete modiﬁed pressure calculation equation.

Хайер и Гото [36] придумали термин источника PPE с частями, компенсирующими ошибки, для еще большего улучшения расчетов поля давления и скорости. Компенсирующие части должны быть мерами для мгновенных и накопительных нарушений несжимаемости жидкости. подраздел 17 показывает новые термины, а уравнение 18 показывает полное модифицированное уравнение расчета давления.



**4 Implementation**

This work implements the MPS through the C/C++ programming languages. OpenMP [25] and CUDA [26] were used to take advantage of the many cores in the CPU and GPU, hence accelerating the program execution. Lastly, there is a graphical user interface (GUI) to aid in the system usage, developed through the Windows Forms graphical class library from the Microsoft .NET framework [56].

Эта работа реализует MPS с помощью языков программирования C/C++. OpenMP [25] и CUDA [26] использовались для использования преимуществ множества ядер в процессоре и графическом процессоре, что ускоряло выполнение программы. Наконец, существует графический пользовательский интерфейс (GUI), помогающий в использовании системы, разработанный с помощью библиотеки графических классов Windows Forms от Microsoft.СЕТЕВАЯ структура [56].

**4.1 Neighbouring search algorithms**

This study adopts the ”cell-linked list” strategy for the neighbourhood search [43]. It relies on a background Cartesian grid that divides the whole domain into cells. They have sides equal to r e , which is the inﬂuence radius of a particle. In every iteration, the particles of the simulation are allocated in a speciﬁc cell, depending on their position. Thereby, when searching for a particle’s neighbours, it is only necessary to look in the cell of the particle itself and adjacent cells. Thus, there is a list of particles that remain constant for that entire step, which offers a considerable decrease in the complexity of the neighbouring search function to O(n log n).

В этом исследовании используется стратегия ”связанного с ячейками списка” для поиска по соседству [43]. Он основан на фоновой декартовой сетке, которая делит весь домен на ячейки. Они имеют стороны, равные r e, что является радиусом влияния частицы. На каждой итерации частицы моделирования распределяются в определенной ячейке в зависимости от их положения. Таким образом, при поиске соседей частицы необходимо только заглянуть в ячейку самой частицы и соседние ячейки. Таким образом, существует список частиц, которые остаются постоянными для всего этого шага, что значительно снижает сложность функции поиска по соседству до O(n log n).

**4.2 OpenMP**

The machine used in the present study holds an Intel ® Core™ i7-6820HK CPU @ 2.70 GHz [57] with 32 GB of installed RAM, a 64-bit operating system (x64), with 4 cores. Listing 1 shows an example of the particle number density function main loop parallelized just by adding one line of code, highlighted in red (line 1).

4.2 Открытый формат

Машина, используемая в настоящем исследовании, содержит процессор Intel ® Core ™ i7-6820HK с частотой 2,70 ГГц [57] с 32 ГБ установленной оперативной памяти, 64-разрядную операционную систему (x64) с 4 ядрами. В листинге 1 показан пример основного цикла функции плотности числа частиц, распараллеленного простым добавлением одной строки кода, выделенной красным цветом (строка 1).

**4.3 CUDA**

The GPU code was developed based on the fully sequential and the OpenMP versions previously presented using CUDA C/C++. The utilized machine has an NVIDIA GeForce GTX 1080 Ti, which contains a total of 3, 584 CUDA cores and 11 GB of video memory. The number of threads per block is set to 256. As for the implementations, while the parallelization process of some functions was straightforward, some others needed adaptations for a parallelized version.

4.3 CUDA

Код GPU был разработан на основе полностью последовательных и OpenMP версий, ранее представленных с использованием CUDA C/C++. Используемая машина оснащена NVIDIA GeForce GTX 1080 Ti, которая содержит в общей сложности 3 584 ядра CUDA и 11 ГБ видеопамяти. Количество потоков в блоке равно 256. Что касается реализаций, то, хотя процесс распараллеливания некоторых функций был простым, некоторые другие нуждались в адаптации для распараллеленной версии.

A tricky parallelization step in the fully incompressible model of the MPS is the PPE solution. For this work, the standard ICCG solver used by Koshizuka and his colleagues [45] was parallelized in both OpenMP and CUDA versions using raw array formats for the linear system elements, such as the coefﬁcient matrix and right side source vector. Also, to decrease memory usage by data structures, the total number of possible neighbours (N neigh ) of a particle was limited to 300, which is the value used in [8]. Doing so allowed assembling a coefﬁcient matrix with N × N neigh elements with N being the total number of particles, rather than an N × N matrix. Listing 2 shows the PPE assembly in a CUDA kernel.

Сложным шагом распараллеливания в полностью несжимаемой модели MPS является решение PPE. Для этой работы стандартный решатель ICCG, используемый Кошизукой и его коллегами [45], был распараллелен как в версиях OpenMP, так и в CUDA, используя форматы необработанных массивов для элементов линейной системы, таких как матрица коэффициентов и вектор источника правой стороны. Кроме того, чтобы уменьшить использование памяти структурами данных, общее число возможных соседей (N) частицы было ограничено 300, что является значением, используемым в [8]. Это позволило собрать матрицу коэффициентов с N × N соседними элементами, где N - общее количество частиц, а не матрица N×N. В листинге 2 показана сборка КАНАЛА в ядре CUDA.

Another signiﬁcant step of the method, performance-wise, is the neighbourhood search function. This step occupies a large portion of the simulation running time in either pressure сalculation models, making its parallelization essential to the practicability of massive simulations. In this work, the implementation of the parallelized neighbourhood search is also based on the cell linked list approach described in subsection 4.1, but, its parallelization process is based on the one from Ref. [58], which presents a high scalability capacity.

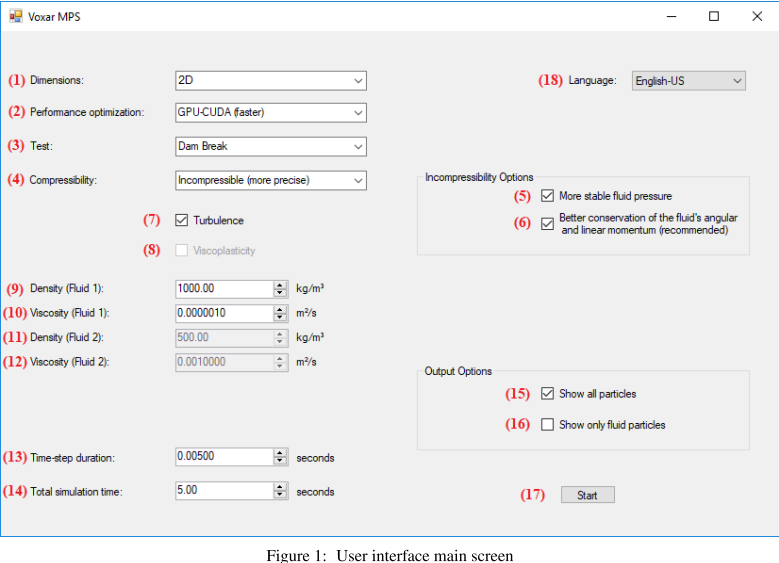
Еще одним важным шагом метода с точки зрения производительности является функция поиска окрестностей. Этот шаг занимает большую часть времени выполнения моделирования в обеих моделях расчета давления, что делает его распараллеливание необходимым для практической реализации масштабного моделирования. В этой работе реализация распараллеленного поиска окрестностей также основана на подходе к списку связанных ячеек, описанном в подразделе 4.1, но процесс его распараллеливания основан на методе из ссылки [58], который обеспечивает высокую масштабируемость.

**4.4 Graphical user interface**

The user interaction may happen through a GUI. Figure 1 shows its layout and design. In it, (1) is a combo box from which the user can choose whether the simulation will have two or three dimensions; in (2) the user can choose how the code will run: sequentially, in parallel in CPU (through OpenMP) or parallel in GPU (through CUDA); in (3) the user will select a previously built simulation scenario; in (4) two approaches of pressure calculation can be chosen: weakly compressible or fully incompressible. By selecting the latter, (5) and (6) will be available, which are options that, if checked, enable Khayyer and Gotoh subsection 3.3 models of pressure gradient calculations to better conserve the linear and angular momentum of the ﬂuid ﬂow. The checking of (7) employs the SPS-LES turbulence model; (8) is only available if the test scenario chosen allows multiphase interaction. If checked, it enables viscoplastic properties in the second ﬂuid in the simulation. (9) and (11) are the density values [Kg/m 3 ] of the ﬂuids in the simulation, and (10) and (12) are their kinematic viscosity [m 2 /s]; (13) sets the time duration between two steps of the simulation and (14) sets how long it will last, both in seconds. If checked, (15) creates a folder in the executable ﬁle directory containing all the particles’ information in each time step. In contrast, (16) creates a folder with the ﬂuid particles’ information in each time step. (17) starts generating the simulation, and (18) allows the user to switch between languages.

4.4 Графический пользовательский интерфейс

Взаимодействие с пользователем может происходить через графический интерфейс. На рисунке 1 показана его компоновка и дизайн. В нем (1) есть поле со списком, из которого пользователь может выбрать, будет ли моделирование иметь два или три измерения; в (2) пользователь может выбрать, как будет выполняться код: последовательно, параллельно в CPU (через OpenMP) или параллельно в GPU (через CUDA); в (3) пользователь выберет ранее созданный сценарий моделирования; в (4) можно выбрать два подхода к расчету давления: слабо сжимаемый или полностью несжимаемый. При выборе последнего будут доступны (5) и (6), которые являются опциями, которые, если они проверены, позволяют моделям Хайера и Гото подраздел 3.3 расчетов градиента давления лучше сохранять линейный и угловой момент потока жидкости. Проверка (7) использует модель турбулентности SPS-LES; (8) доступна только в том случае, если выбранный сценарий тестирования допускает многофазное взаимодействие. Если этот флажок установлен, он включает вязкопластические свойства во второй жидкости в моделировании. (9) и (11) - значения плотности [Кг/м3] жидкостей в моделировании, а (10) и (12) - их кинематическая вязкость [м2/с]; (13) устанавливает продолжительность времени между двумя этапами моделирования и (14) устанавливает, как долго это будет продолжаться, как в секундах. Если флажок установлен, (15) создает папку в каталоге исполняемых файлов, содержащую всю информацию о частицах на каждом временном шаге. Напротив, (16) создает папку с информацией о частицах жидкости на каждом временном шаге. (17) запускает генерацию моделирования и (18) позволяет пользователю переключаться между языками.



**5 Evaluation & discussion**

The aim here is to validate the models and properties of the developed method, such as incompressibility, numerical precision, turbulent ﬂow, and multiphase interaction. Another goal in this section is to assess the performance gain and frame rates of the parallelized implementations of the weakly compressible and fully incompressible versions of the MPS and discuss them.

5 Оценка и обсуждение

Цель здесь состоит в проверке моделей и свойств разработанного метода, таких как несжимаемость, численная точность, турбулентный поток и многофазное взаимодействие. Другая цель в этом разделе - оценить прирост производительности и частоту кадров распараллеленных реализаций слабо сжимаемых и полностью несжимаемых версий MPS и обсудить их.

**5.1 Water drop**

The examination of the evolution of an elliptic water drop is a common test case to validate incompressibility models of a particle-based ﬂuid simulation method [47] [59] [43]. The usual test consists of a two-dimensional water drop, beginning the simulation in the shape of a circle, with a predeﬁned velocity ﬁeld of (−100x, 100y) m/s so that its format evolves into an elliptical shape over time. Figure 2 shows a sketch of the test’s geometry. The water drop has a circle radius of 9 × 10(-2) m and an average particle distance of 5 × 10(−2) m, implying a total of 1257 particles. The inﬂuence radius is r e = 3l 0 , where l 0 is the average particle distance, and the adopted time was 10(−5) s. The used conﬁguration for this test was the default one, as presented in Figure 1, with a ﬂuid density equal to 10 3 Kg/m 3 and ﬂuid viscosity 10(−6) m 2 /s, only differing in the ﬁeld (3) since it is the Water drop test.

5.1 Капля воды

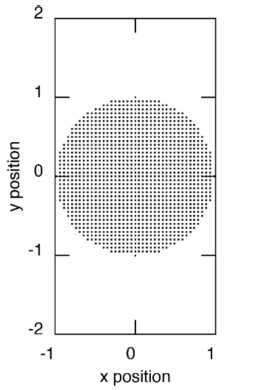
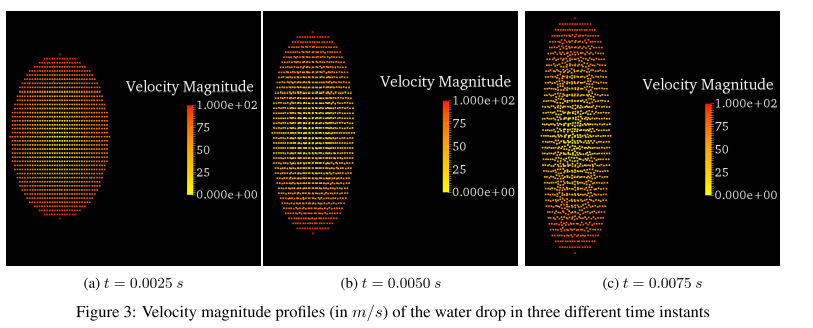
Изучение эволюции эллиптической капли воды является обычным тестовым случаем для проверки моделей несжимаемости метода моделирования жидкости на основе частиц [47] [59] [43]. Обычный тест состоит из двумерной капли воды, начинающей моделирование в форме круга, с заданным полем скорости (- 100x, 100y) м/с, так что его формат со временем превращается в эллиптическую форму. На рисунке 2 показан эскиз геометрии теста. Капля воды имеет радиус окружности 9 × 10(-2) м и среднее расстояние между частицами 5 × 10(-2) м, что означает в общей сложности 1257 частиц. Радиус воздействия равен r e = 3l 0, где l 0 - среднее расстояние между частицами, а принятое время составило 10(-5) с. Используемая конфигурация для этого испытания была стандартной, как показано на рисунке 1, с плотностью жидкости, равной 10 3 кг/м3, и вязкостью жидкости 10(-6) м2/с, отличающейся только в поле (3), поскольку это испытание на падение воды.

As in [43], Figure 3 shows three time instants of the simulation. The kernel function used here is in Equation 4.

As noted by Monaghan [47], the condition in this test to measure incompressibility is that ab is constant throughout the simulation, where a is the semi-minor axis, and b is the semi-major axis of the ellipse. This simulation ends when the size of b becomes twice the value it was initially, and, in that instant, the ab value is compared to its initial value. Errors are within less than 0.3 %. Table 1 compares this result to [47] which keep errors in less than 2 % and [43] which keep errors in less than 0.2 %. This comparison shows the achieved result matches those from previous works.

Как и в [43], на рисунке 3 показаны три временных момента моделирования. Используемая здесь функция ядра приведена в уравнении 4.

Как отметил Монаган [47], условие в этом тесте для измерения несжимаемости состоит в том, что ab является постоянным на протяжении всего моделирования, где a – малая полу ось, а b – большая полу ось эллипса. Это моделирование заканчивается, когда размер b становится вдвое больше, чем было изначально, и в этот момент значение ab сравнивается с его начальным значением. Погрешности находятся в пределах менее 0,3 %. В таблице 1 этот результат сравнивается с [47], в которых ошибки составляют менее 2%, и [43], в которых ошибки составляют менее 0,2%. Это сравнение показывает, что достигнутый результат соответствует результатам предыдущих работ.

**5.2 Dam break**

The collapse of a water column has been widely used in the literature to validate the numerical precision of various ﬂuid simulation techniques.

As the test performed by [8], in this work, the water column height is two times bigger than the water column length L. The ﬂoor in the model employed here is also four times the length of the water column. The size of the water column varies depending on how many particles the simulation has. The average particle distance is 10(−2) m, and the time step of the simulation is 5 × 10(−3) s. The total number of particles is 1122.

5.2 Прорыв плотины

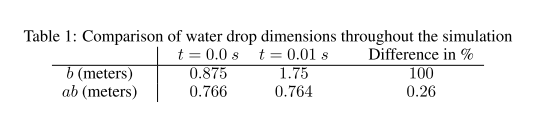
Разрушение водяного столба широко используется в литературе для проверки численной точности различных методов моделирования жидкости.

Как показал тест, проведенный [8], в этой работе высота водяного столба в два раза больше длины водяного столба L. Пол в используемой здесь модели также в четыре раза превышает длину водяного столба. Размер водяного столба варьируется в зависимости от того, сколько частиц содержится в моделировании. Среднее расстояние между частицами составляет 10(-2) м, а временной шаг моделирования составляет 5 ×10(-3) с. Общее количество частиц составляет 1122.

The authors of the original MPS, referred here as standard MPS, put it to test by comparing it to a volume of ﬂuid (VOF) approach [60] and experimental data from three different experiments [61, 62]. This test adopts the standard dam break model [8] and compares the simulation results obtained with the experimental data and the other simulation results. This comparison is possible by examining the water leading-edge position over time, since the dam burst until it hits the right wall. The leading edge is the front of the collapsing water column running on the ﬂoor (bottom wall).

Авторы оригинального MPS, именуемого здесь стандартными MPS, проверили его, сравнив с подходом к объему жидкости (VOF) [60] и экспериментальными данными из трех различных экспериментов [61, 62]. В этом тесте используется стандартная модель разрушения плотины [8] и сравниваются полученные результаты моделирования с экспериментальными данными и другими результатами моделирования. Это сравнение возможно путем изучения положения переднего края воды с течением времени, поскольку плотина прорвалась, пока не уперлась в правую стену. Передний край - это передняя часть разрушающегося столба воды, протекающего по полу (нижняя стена).

Figure 4 compares directly experimental data, other methods results, with the proposed technique. The conﬁguration used here is the default as well, as depicted in Figure 1. The leading edge position is dimensionless, Z/L, where L is



the water column’s initial length, which for this test is equal to 0.18 m. If the water column size changes, the value of 0.18 m also changes, respecting the average particle distance l 0 . The time axis in the chart is dimensionless as well, t p 2g/L, where t is the time in seconds and g the gravitational acceleration, which is equal to 9.8 m/s 2.

На рисунке 4 непосредственно сравниваются экспериментальные данные, результаты других методов, с предложенной методикой. Используемая здесь конфигурация также используется по умолчанию, как показано на рисунке 1. Положение передней кромки безразмерно, Z/L, где L - начальная длина водяного столба, которая для данного испытания равна 0,18 м. Если размер столба воды изменяется, значение 0,18 м также изменяется с учетом среднего расстояния частиц l0. Ось времени на графике также безразмерна, t p 2g/L, где t - время в секундах, а g - ускорение свободного падения, равное 9,8 м/с2.

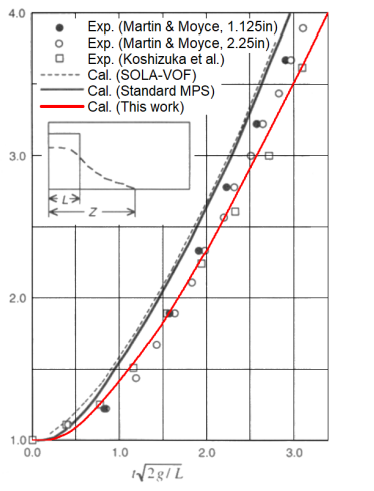
Figure 4: Direct comparison of the evolution of leading edge position between the developed technique (in red), and the experimental results and other methods. Background image extracted from [8]

Рисунок 4: Прямое сравнение эволюции положения передней кромки между разработанной методикой (красным цветом) и результатами эксперимента и другими методами. Фоновое изображение, извлеченное из [8]

Figure 4 shows that the developed method gets a lot closer to the experimental data than the others. Interestingly, it almost overlaps the experiment represented by the small empty squares from [61]. It is important to note that, differently from a technique, the experiments do not become outdated since the setup did not change over the years. Therefore, this corroborates the assumption of a higher precision of the implemented method in this study.

На рисунке 4 показано, что разработанный метод намного ближе к экспериментальным данным, чем другие. Интересно, что он почти перекрывает эксперимент, представленный маленькими пустыми квадратами из [61]. Важно отметить, что, в отличие от техники, эксперименты не устаревают, поскольку установка не менялась с годами. Таким образом, это подтверждает предположение о более высокой точности реализованного метода в данном исследовании.

**5.3 R-T instability**

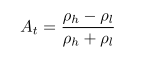
The Rayleigh-Taylor instability test’s goal is to show the multiphase model ability to handle the density stratiﬁcation and to evolve a linear perturbation into nonlinear hydrodynamic turbulence [63].

The test consists of placing the same amount of two immiscible ﬂuids with different densities and the same kinematic viscosity, one on the top of the other. The heavier one will stay on top of the lighter, only inﬂuenced by the gravitational force g inside a two-dimensional rectangular box. When the simulation starts, the denser ﬂuid will tend to go downwards, pushing the lighter upwards. The interface between the ﬂuids will become unstable, and the format of the pattern generated at the interface will say whether the hydrodynamic turbulence was formed or not. The lighter ﬂuid forms a bubble in the shape of a mushroom cap that breaks eventually. The Atwood number, calculated as shown in Equation 19, characterizes the problem.

5.3 R-T нестабильность

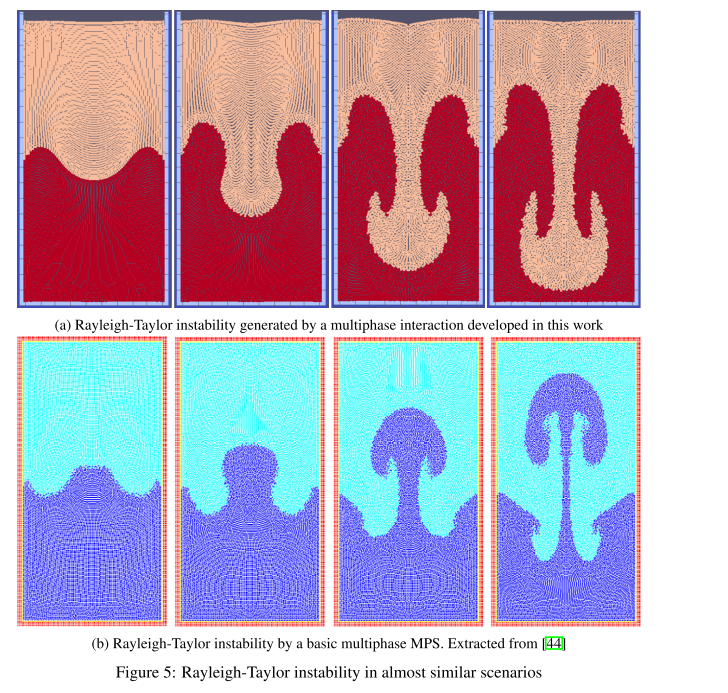
Цель теста на нестабильность Рэлея-Тейлора - показать способность многофазной модели справляться со стратификацией плотности и превращать линейное возмущение в нелинейную гидродинамическую турбулентность [63].

Испытание состоит в размещении одного и того же количества двух несмешивающихся жидкостей с разной плотностью и одинаковой кинематической вязкостью, одна поверх другой. Более тяжелый останется сверху более легкого, только под влиянием гравитационной силы g внутри двумерной прямоугольной коробки. Когда начнется моделирование, более плотная жидкость будет стремиться вниз, подталкивая более лёгкую вверх. Граница раздела между жидкостями станет нестабильной, и формат рисунка, сгенерированного на границе раздела, скажет, была ли сформирована гидродинамическая турбулентность или нет. Легкая жидкость образует пузырь в форме шляпки гриба, который в конце концов лопается. Число Этвуда, рассчитанное, как показано в уравнении 19, характеризует проблему.



ρ h and ρ l refer to the heavier and lighter ﬂuid densities, respectively. For the test used in this study, A t = 1/3 and the kinematic viscosity ν = 0.010 for both ﬂuids, following [44]. The test has 3066 particles. The system conﬁguration used in this scenario differs from the default options shown in Figure 1 with some differences. Here, ﬁeld (3) is R-T instability, (4) is Weakly compressible, and ﬁelds (9) and (11) respect the Atwood number calculation. Figure 5a shows some steps of the generated simulation. Ref. [44] shows a similar test displaying the same phenomenon. For a better context on how other particle-based approaches behave, their results are shown next to the generated simulation in Figure 5b . The main difference between them is that, in the scenario from Ref. [44], there is already an initial perturbation to the ﬂuids’ interface. Another difference is that in the test performed here, the domain top is open. These differences inﬂuence the test, causing the bubble (mushroom cap) to be upside-down.

ρ h и ρ l относятся к плотности более тяжелой и более легкой жидкости соответственно. Для теста, используемого в этом исследовании, t=1/3 и кинематическая вязкость ν=0,010 для обеих жидкостей, следуя [44]. В тесте содержится 3066 частиц. Конфигурация системы, используемая в этом сценарии, отличается от параметров по умолчанию, показанных на рисунке 1, с некоторыми отличиями. Здесь поле (3) является нестабильностью R-T, (4) слабо сжимаемо, а поля (9) и (11) учитывают расчет числа Этвуда. На рисунке 5а показаны некоторые этапы сгенерированного моделирования. Ссылка [44] показывает аналогичный тест, показывающий то же явление. Для лучшего понимания того, как ведут себя другие подходы, основанные на частицах, их результаты показаны рядом с сгенерированным моделированием на рисунке 5b. Основное различие между ними заключается в том, что в сценарии из Ссылки [44] уже существует начальное возмущение на границе раздела жидкостей. Еще одно отличие заключается в том, что в тесте, выполняемом здесь, верхняя часть домена открыта.Эти различия влияют на тест, в результате чего пузырь (шляпка гриба) оказывается перевернутым.



The proposed technique generally shows good agreement regarding the Rayleigh-Taylor instability when compared to another particle-based solution.

Предложенный метод в целом демонстрирует хорошее согласие в отношении неустойчивости Рэлея-Тейлора по сравнению с другим решением на основе частиц.

**5.4 Oil spill**

The Oil spill test makes possible a qualitative analysis and a unique comparison to validate the multiphase model. [40] provides a replicable test of a continuous oil spill due to a damaged tank.

The average particle distance is 4 × 10−3 m, the water and oil’s kinematic viscosity ν water and ν oil are, respectively, 10−6 m 2 /s and 5×10 −5 m 2 /s and conﬁguration here, relative to Figure 1, differs in ﬁelds (3), which is Oil Spill, (4), Weakly compressible, (11), which is 897.0Kg/m 3 and (12), which is 5 × 10−5 m 2 /s. This test has a total of 152, 779 particles.

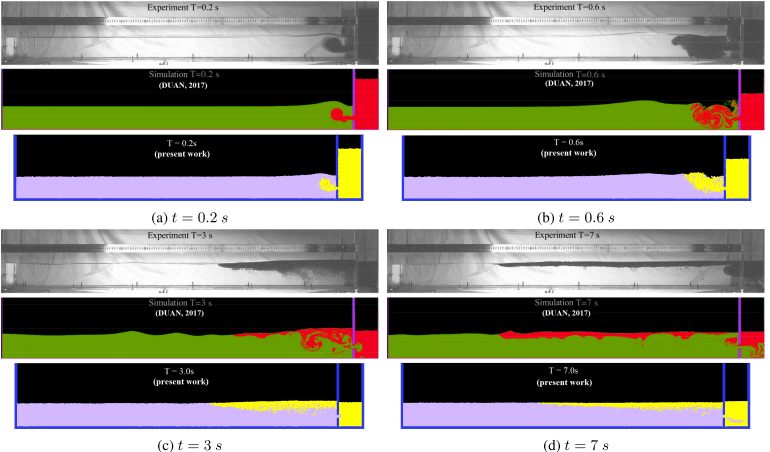
5.4 Разлив нефти

Испытание на разлив нефти позволяет провести качественный анализ и уникальное сравнение для проверки многофазной модели. [40] обеспечивает воспроизводимый тест непрерывного разлива нефти из-за поврежденного резервуара.

Среднее расстояние между частицами составляет 4×10-3 м, кинематическая вязкость воды и масла ν вода и ν масло составляют соответственно 10-6 м2/с и 5×10-5 м2/с, и конфигурация здесь, относительно рисунка 1, отличается в полях (3), что является разливом нефти, (4), Слабо сжимаемой, (11), что составляет 897,0 кг/м3 и (12), что составляет 5×10-5 м2/с. Этот тест содержит в общей сложности 152 779 частиц.

This leakage simulation is qualitatively compared to experimental results and to [40], which is a fully incompressible multiphase MPS. Figure 6 shows ﬁve time instants of the experiments and the simulations. In each time instants, there is: an image of the experiment; the simulation by [40]; and the generated simulation, in that order.

Это моделирование утечки качественно сравнивается с экспериментальными результатами и с [40], которое представляет собой полностью несжимаемый многофазный MPS. На рисунке 6 показаны пять временных моментов экспериментов и моделирования. В каждый момент времени есть: изображение эксперимента; моделирование с помощью [40]; и сгенерированное моделирование, в таком порядке.



The authors in [40] claim that FS-MMPS is quite precise and accurate. On the other hand, it is possible to observe that the use of a weakly compressible approach of the combined method developed here can better predict certain situations, such as the more straight and even ﬂow of the oil, in yellow, over the denser ﬂuid, mainly in Figure 6c and Figure 6d. In a general manner, the oil proﬁle and the waves generated by it are in reasonable agreement with the generated by the experiment, with fewer computations than the FS-MMPS, since, in this case, a WC model is used, which signiﬁcantly diminishes the pressure calculation complexity.

Авторы в [40] утверждают, что FS-MMPS достаточно точен и точен. С другой стороны, можно заметить, что использование слабо сжимаемого подхода комбинированного метода, разработанного здесь, может лучше прогнозировать определенные ситуации, такие как более прямой и равномерный поток масла, выделенный желтым цветом, по более плотной жидкости, в основном на рисунке 6c и рисунке 6d. В общем случае профиль масла и генерируемые им волны находятся в разумном согласии с полученными в результате эксперимента, с меньшим количеством вычислений, чем FS-MMPS, поскольку в этом случае используется модель WC, что значительно снижает сложность расчета давления.

**5.5 Pressure ﬁeld**

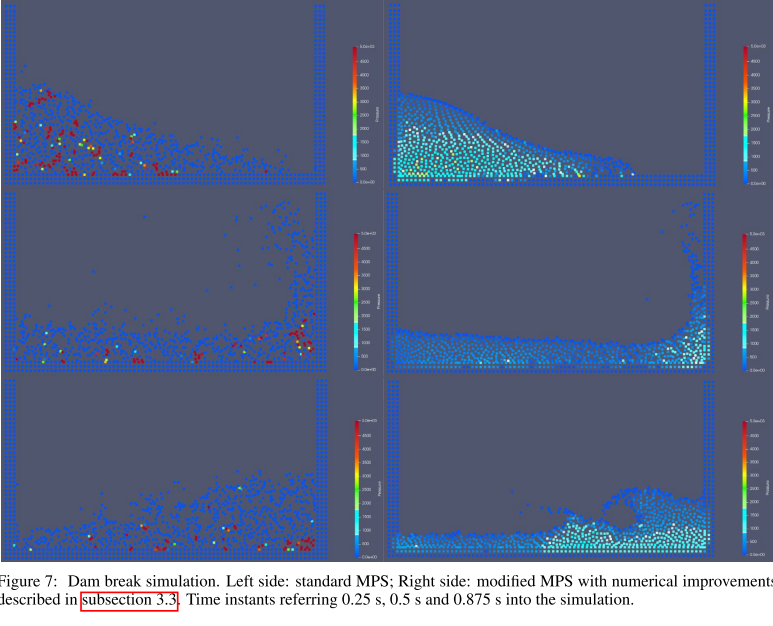
It is essential to show how the previously presented numerical improvements impact the pressure ﬁeld of the ﬂuid. This test makes use of the traditional dam break scenario described in subsection 5.2. Figure 7 shows three time instants of the same simulation. The left side refers to the simulation using the standard MPS, and the right side, the simulation with the numerical improvements described in subsection 3.3.

5.5 Поле давления

Важно показать, как ранее представленные численные улучшения влияют на поле давления жидкости. В этом тесте используется традиционный сценарий прорыва плотины, описанный в подразделе 5.2. На рисунке 7 показаны три момента времени одного и того же моделирования. Левая сторона относится к моделированию с использованием стандартных MPS, а правая сторона - к моделированию с численными улучшениями, описанными в подразделе 3.3

It is apparent in Figure 7 that the numerical improvements signiﬁcantly diminish the spurious pressure oscillations and can provide a more stable pressure ﬁeld during the simulation, even after the ﬂuid suffers large deformations.

На рисунке 7 видно, что численные улучшения значительно уменьшают паразитные колебания давления и могут обеспечить более стабильное поле давления во время моделирования, даже после того, как жидкость испытывает большие деформации.



**5.6 Computational performance analysis**

It is noticeable that switching between the incompressibility models - weakly compressible (WC) or fully incompressible (FI) - had a material impact on computational performance, both in memory usage and in runtime performance. This discrepancy happens because the FI approach requires the solution of a linear system of order N × N neigh , to ensure incompressibility and, thus, obtain more precise values for the pressures. Storing it in memory and solving it are both costly steps for the algorithm. The chosen incompressibility model inﬂuences the most on the runtime, regardless of the selected combination of enhancements.

5.6 Анализ вычислительной производительности

Заметно, что переключение между моделями несжимаемости - слабо сжимаемой (WC) или полностью несжимаемой (FI) - оказало существенное влияние на производительность вычислений, как в использовании памяти, так и в производительности во время выполнения. Это несоответствие происходит потому, что подход FI требует решения линейной системы порядка N × N, чтобы обеспечить несжимаемость и, таким образом, получить более точные значения давлений. Сохранение его в памяти и его решение являются дорогостоящими шагами для алгоритма. Выбранная модель несжимаемости оказывает наибольшее влияние на время выполнения, независимо от выбранной комбинации улучшений.

To evaluate the computational performance, this section details the memory usage and the runtime proﬁle of two main approaches (WC and FI) combined with three different types of execution: completely sequential; parallelized through OpenMP; parallelized through CUDA. The entire runtime for each test size was measured to calculate the speedups. Finally, it is possible to calculate each version’s frame rate by running a set of tests in which they vary in the number of particles.

Для оценки вычислительной производительности в этом разделе подробно описывается использование памяти и профиль времени выполнения двух основных подходов (WC и FI) в сочетании с тремя различными типами выполнения: полностью последовательным; распараллеленным с помощью OpenMP; распараллеленным с помощью CUDA. Для расчета ускорений было измерено все время выполнения для каждого размера теста. Наконец, можно рассчитать частоту кадров каждой версии, выполнив набор тестов, в которых они различаются по количеству частиц.

**5.6.1 Memory usage**

The Performance and Diagnostics tool of Visual Studio 2019 [64] provides a detailed report of the sequential and OpenMP executions to evaluate CPU memory usage in all versions. The NVIDIA Visual proﬁler [65] provided the majority of information regarding the GPU, such as the whole simulation runtime and time spent in each CUDA kernel during the execution. As for the GPU memory usage, GPU-Z [66] provided the minimum and maximum occupancy throughout the simulation runtime.

5.6.1 Использование памяти

Инструмент производительности и диагностики Visual Studio 2019 [64] предоставляет подробный отчет о последовательном выполнении и выполнении OpenMP для оценки использования памяти ЦП во всех версиях. Визуальный профилировщик NVIDIA [65] предоставил большую часть информации о графическом процессоре, такой как время выполнения всего моделирования и время, проведенное в каждом ядре CUDA во время выполнения. Что касается использования памяти GPU, GPU-Z[66] обеспечивал минимальное и максимальное заполнение в течение всего времени выполнения моделирования.

Table 2 shows the memory usage of the WC approach of the MPS for a standard 2D dam break test, like the one presented in subsection 5.2. For this case, there is a total of 1, 054, 122 particles.

В таблице 2 показано использование памяти WC-подхода MPS для стандартного 2D-теста на разрыв плотины, подобного тому, который представлен в подразделе 5.2. В этом случае в общей сложности 1 054 122 частицы.

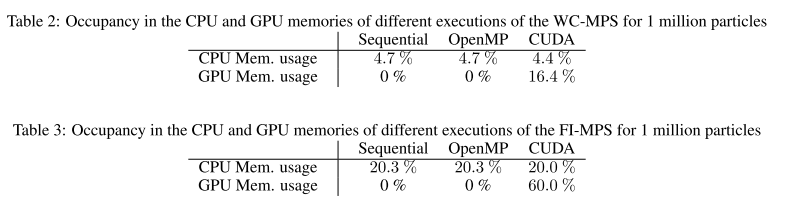


Table 3 shows the memory usage relative to the FI approach for the same test described above and in subsection 5.2. One million particles can be considered a high amount of particles for a FI approach running on GPU since a linear system has to be fully loaded. This number of particles is an improvement compared to previous works such as [67, 42].

В таблице 3 показано использование памяти относительно подхода FI для того же теста, описанного выше и в подразделе 5.2. Миллион частиц можно считать большим количеством частиц для подхода FI, работающего на графическом процессоре, поскольку линейная система должна быть полностью загружена. Это количество частиц является улучшением по сравнению с предыдущими работами, такими как [67, 42].

Note that, in both approaches, the CUDA versions require not only the GPU memory but also CPU memory, which is also smaller compared to OpenMP and the fully sequential executions. This difference may happen since some information must exist in host memory (CPU) and in device memory (GPU) to transfer particle input information from the CPU to the GPU.

Обратите внимание, что в обоих подходах версии CUDA требуют не только памяти графического процессора, но и памяти центрального процессора, которая также меньше по сравнению с OpenMP и полностью последовательными исполнениями. Это различие может произойти, поскольку некоторая информация должна существовать в памяти хоста (CPU) и в памяти устройства (GPU) для передачи информации о вводе частиц из центрального процессора в графический процессор.

**5.6.2 Speedup**

Due to the scalability of the GPU-accelerated neighbourhood search strategy, the system was able to reach promising speedup values. Figure 8 shows that this is especially true in more massive simulations. The OpenMP version did not achieve considerable speedups since its neighbourhood calculation is not optimally parallelized.

5.6.2 Ускорение

Благодаря масштабируемости стратегии поиска окрестностей, ускоряемой графическим процессором, система смогла достичь многообещающих значений ускорения. На рисунке 8 показано, что это особенно верно при более масштабном моделировании. Версия OpenMP не достигла значительного ускорения, так как ее вычисление окрестностей не оптимально распараллелено.

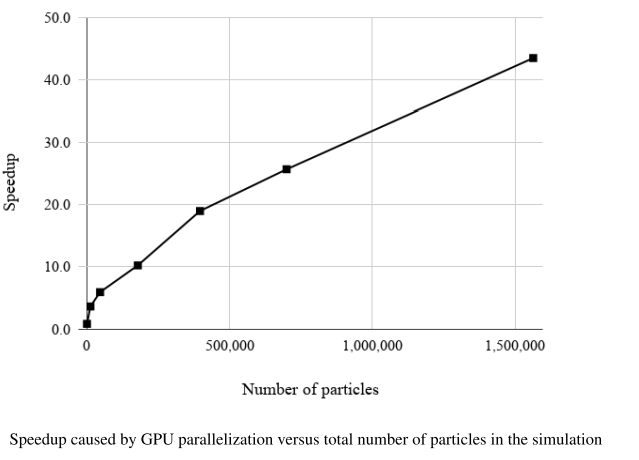


Figure 8 shows that a case with 1, 563, 658 particles can achieved a massive speedup of 43.5 times. The speedup curve shows that simulations with more particles could yield even higher speedup rates. On the other hand, for a case with 1, 022 particles, the speedup rate obtained was 0.8 times, in other words, running this scenario in the GPU is slower than running it in sequentially in the CPU. That shows it is not efﬁcient to employ parallel programming for simulations with a small number of particles. One of the causes for this is that the data transfer between the host and device reduces the speedup signiﬁcantly.

На рисунке 8 показано, что в случае с 1, 563, 658 частицами можно добиться значительного ускорения в 43,5 раза. Кривая ускорения показывает, что моделирование с большим количеством частиц может привести к еще более высоким скоростям ускорения. С другой стороны, для случая с 1 022 частицами скорость ускорения составила 0,8 раза, другими словами, выполнение этого сценария в графическом процессоре медленнее, чем его последовательное выполнение в центральном процессоре. Это показывает, что неэффективно использовать параллельное программирование для моделирования с небольшим количеством частиц. Одной из причин этого является то, что передача данных между хостом и устройством значительно снижает скорость.

**5.6.3 System limits**

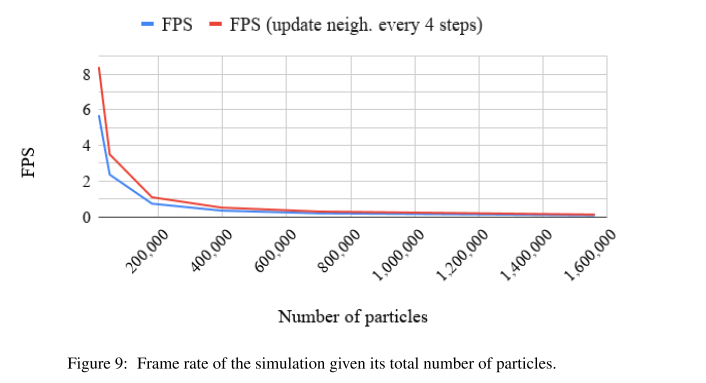
Data related to performance and memory limitations are presented here, such as frame rate achieved for different numbers of particles and the maximum number of particles in a simulation.

5.6.3 Системные ограничения

Здесь представлены данные, связанные с ограничениями производительности и памяти, такие как частота кадров, достигнутая для разного количества частиц, и максимальное количество частиц в моделировании.

Figure 9 shows how many frames per second (FPS) the GPU-accelerated application achieves, given the total number of particles in the simulation. The blue frame rate curve represents a regular execution where the neighbourhood is updated every step, prioritizing a more accurate simulation. The red frame rate curve represents an execution where the list of neighbours is only updated every four steps during the simulation, aiming for high computational performance.

На рисунке 9 показано, сколько кадров в секунду (кадров в секунду) достигает приложение с ускорением GPU, учитывая общее количество частиц в моделировании. Синяя кривая частоты кадров представляет собой регулярное выполнение, при котором окрестности обновляются на каждом шаге, уделяя приоритетное внимание более точному моделированию. Красная кривая частоты кадров представляет выполнение, при котором список соседей обновляется только каждые четыре шага во время моделирования, что обеспечивает высокую вычислительную производительность.



Hence, in a case that computational performance is the priority, Figure 9 shows that it is possible to simulate a scenario with approximately 200, 000 particles at one frame per second. It also displays a case with 46, 768 particles run at approximately 4 fps. Both of them are already considered interactive frame rates [68, 69].

Следовательно, в случае, когда вычислительная производительность является приоритетом, на рисунке 9 показано, что можно смоделировать сценарий с примерно 200 000 частицами со скоростью одного кадра в секунду. Он также отображает случай с 46 768 частицами, работающими со скоростью примерно 4 кадра в секунду. Оба они уже считаются интерактивными частотами кадров [68, 69].

Regarding the simulation size limit, it is noteworthy that the available memory plays a fundamental role. The GPU used in this work provides 11 GB of video memory and allows for a simulation with 1, 563, 658 before the application runs out of memory. The largest simulation size tested for the fully sequential version when adopting the fully incompressible approach had a total of 3, 677, 442 particles. The CPU process of the latter used 25 of the 32 GB available of RAM. When adopting the weakly compressible model, the total number of particles could reach 16, 657, 962 by using 26 GB of the available RAM. The last two simulation loads show how expensive it is in terms of memory, just loading the PPE into the RAM, not to mention the computational load required to solve it.

Что касается ограничения размера моделирования, следует отметить, что доступная память играет фундаментальную роль. Графический процессор, используемый в этой работе, обеспечивает 11 ГБ видеопамяти и позволяет моделировать с 1, 563, 658 до того, как у приложения закончится память. Самый большой размер моделирования, протестированный для полностью последовательной версии при использовании полностью несжимаемого подхода, имел в общей сложности 3 677 442 частицы. Процесс процессора последнего использовал 25 из 32 ГБ доступной оперативной памяти. При использовании слабо сжимаемой модели общее количество частиц может достигать 16 657 962 при использовании 26 ГБ доступной оперативной памяти. Последние две загрузки моделирования показывают, насколько это дорого с точки зрения памяти, просто загружая КАНАЛ в оперативную память, не говоря уже о вычислительной нагрузке, необходимой для его решения.

**6 Conclusions**

As previously discussed, the study of ﬂuid ﬂow simulation is of great importance in mitigating the consequences of environmental disasters and accidents. It has applications in a wide range of engineering problems, computer graphics, and virtual and augmented reality software. Meshless methods like the MPS are a great alternative to deal with large deformations and free-surface ﬂow, situations where the traditional mesh-based approaches usually perform inefﬁciently.

6 Выводы

Как обсуждалось ранее, изучение моделирования потока жидкости имеет большое значение для смягчения последствий экологических катастроф и аварий. Он имеет приложения в широком спектре инженерных задач, компьютерной графики, а также программного обеспечения виртуальной и дополненной реальности. Методы без сетки, такие как MPS, являются отличной альтернативой для решения проблем с большими деформациями и потоком со свободной поверхностью, в ситуациях, когда традиционные подходы на основе сетки обычно работают неэффективно.

Throughout the development of his work, it became clear that the community is continually improving the MPS, both in numerical performance and computation efﬁciency, despite suffering from a few pressure instability problems. The literature shows its usability in a large number of scenarios. The considerable amount of referenced works also shows the complexity of this task, the importance of the method, and the great potential to simulate, increasingly more realistically, ﬂuid ﬂows.

На протяжении всей его работы стало ясно, что сообщество постоянно совершенствует MPS, как в числовой производительности, так и в эффективности вычислений, несмотря на некоторые проблемы с нестабильностью давления. В литературе показано его использование в большом количестве сценариев. Значительное количество ссылочных работ также показывает сложность этой задачи, важность метода и большой потенциал для моделирования, все более реалистичного, потоков жидкости.

MPS optimization is moderately complex since it is used to replicate real phenomena more reliably, when taking into consideration other meshless approaches. Some works enhance its computational efﬁciency with acceleration structures without losing the precision it offers [23] [22] [67] [42]. In contrast, other works prefer, despite the precision loss, replace the performance and memory bottleneck of the method, the solution of a PPE, with an equation of state to solve the pressures [43] [39] and then, parallelize it through GPU [24] [70]. Another factor to consider is hardware development since manufacturers are continuously building more powerful GPUs, which directly inﬂuence computational performance.

Оптимизация MPS является умеренно сложной, поскольку она используется для более надежного воспроизведения реальных явлений, если учитывать другие подходы без сетки. Некоторые работы повышают его вычислительную эффективность с помощью структур ускорения без потери точности, которую он предлагает [23] [22] [67] [42]. В отличие от этого, другие работы предпочитают, несмотря на потерю точности, заменять узкое место в производительности и памяти метода, решение СИЗ, уравнением состояния для решения давления [43] [39], а затем распараллеливать его через графический процессор [24] [70]. Еще одним фактором, который следует учитывать, является разработка аппаратного обеспечения, поскольку производители постоянно создают более мощные графические процессоры, которые напрямую влияют на производительность вычислений.

The literature also shows that works usually accelerate the standard version of the MPS [8] or the standard weakly compressible version [43][39], with few numerical improvements to the calculation. This work provides a wide variety of models, improvements, and approaches to the MPS technique, which are entirely parallelized, through OpenMP and CUDA, integrated into a single solution. This integration enables a ﬁne-tuning of the system, allowing setups concerned with high precision, fully incompressible ﬂuid ﬂow, and GPU-accelerated multiphase WC ﬂuid simulations. This solution is open-source under the GPLv3 license.

В литературе также показано, что работы обычно ускоряют стандартную версию MPS[8] или стандартную слабо сжимаемую версию [43][39] с небольшими численными улучшениями в расчете. Эта работа предоставляет широкий спектр моделей, улучшений и подходов к технике MPS, которые полностью распараллелены с помощью OpenMP и CUDA, интегрированных в единое решение. Эта интеграция обеспечивает точную настройку системы, позволяя выполнять настройки, связанные с высокой точностью, полностью несжимаемым потоком жидкости и многофазным моделированием жидкости WC с ускорением GPU. Это решение с открытым исходным кодом под лицензией GPLv3.

Regarding the numerical improvements, the techniques proposed by [55] and [44] were combined and extended, which implicated in enhancements in simulation coherence, as presented in the Dam break, Oil spill, and Water drop tests. The developed method shows compatibility with recent works in the Oil Spill test, by qualitatively comparing it with [40], which shows the numerical advances achieved. The GPU-acceleration provides speedups ranging from 3 to 43 times, depending on the total number of particles in the simulation. That permits a simulation with approximately 200, 000 particles to run at one frame per second, and a test case with 46, 768 particles to achieve nearly 4 fps, which both are already considered interactive rates [68, 69].

Что касается численных улучшений, то методы, предложенные в [55] и [44], были объединены и расширены, что привело к улучшению согласованности моделирования, как это было представлено в тестах на прорыв плотины, разлив нефти и падение воды. Разработанный метод демонстрирует совместимость с недавними работами по испытанию на разлив нефти, качественно сравнивая его с [40], в котором показаны достигнутые численные успехи. GPU-ускорение обеспечивает ускорение в диапазоне от 3 до 43 раз, в зависимости от общего количества частиц в моделировании. Это позволяет моделировать примерно 200 000 частиц со скоростью один кадр в секунду, а тестовый случай с 46 768 частицами позволяет достичь почти 4 кадров в секунду, что уже считается интерактивными скоростями [68, 69].

**6.1 Future Work**

There are exciting possibilities for future developments of this work. Inevitably, reﬁnements in every part of the code will lead to a more optimized version, which is the path to a notable real-time simulation.

6.1 Будущая Работа

Существуют захватывающие возможности для дальнейшего развития этой работы. Неизбежно, уточнения в каждой части кода приведут к более оптимизированной версии, что является путем к заметному моделированию в реальном времени.

A possibility that would enhance even further computational performance is to improve the GPU implementation ofthe FI-MPS to be able to run locally on multiple GPU or in a GPU cluster, which could raise the speedup values to new levels. [24] shows the performance of WC-MPS aided by a GPU cluster; fully incompressible versions of the MPS could beneﬁt signiﬁcantly from such structures. [71] presents a domain decomposition strategy for a parallelized MPS for running in a cluster of computers. That enables massive simulations since the simulation domain can be loaded and solved separately in the PC’s memories and then integrated back.

Возможность, которая еще больше повысит производительность вычислений, заключается в улучшении реализации графического процессора FI-MPS, чтобы иметь возможность работать локально на нескольких графических процессорах или в кластере графических процессоров, что может повысить значения ускорения до новых уровней. [24] показывает производительность WC-MPS, поддерживаемую кластером GPU; полностью несжимаемые версии MPS могли бы значительно выиграть от таких структур. [71] представляет стратегию декомпозиции домена для распараллеленной MPS для работы в кластере компьютеров. Это позволяет проводить масштабное моделирование, поскольку область моделирования может быть загружена и решена отдельно в памяти ПК, а затем интегрирована обратно.

Another possibility is to improve the OpenMP parallelization of the neighbourhood search function. Since that function is based on its sequential version, it could not reach decent speedup values. However, with an optimal CPU parallelization of that step, the OpenMP acceleration could reach new levels.

Другая возможность заключается в улучшении распараллеливания OpenMP функции поиска окрестностей. Поскольку эта функция основана на ее последовательной версии, она не смогла достичь приличных значений ускорения. Однако при оптимальном распараллеливании процессора на этом этапе ускорение OpenMP может достичь новых уровней.

Still on the acceleration task, since solving a PPE is costly, an alternative to the code parallelization could be the usage of neural networks. They would learn the usual results of сommonly yielded linear systems by fully incompressible ﬂuid simulations. The work of [72] shows promising ﬂuid simulations using this approach.

Все еще решается задача ускорения, поскольку решение PPE является дорогостоящим, альтернативой распараллеливанию кода может быть использование нейронных сетей. Они изучили бы обычные результаты обычно получаемых линейных систем с помощью моделирования полностью несжимаемой жидкости. В работе [72] показано перспективное моделирование жидкостей с использованием этого подхода.

The addition of models that allow the interaction between ﬂuids and solids, such as ﬂoating bodies, deformable bodies, and viscoelastic ﬂuid, would drastically increase the number of application possibilities.

Добавление моделей, которые допускают взаимодействие между жидкостями и твердыми телами, такими как плавающие тела, деформируемые тела и вязкоупругая жидкость, значительно увеличило бы число возможностей применения.