**БЕЗОПАСНОСТЬ СТРОИТЕЛЬНЫХ СИСТЕМ.
ЭКОЛОГИЧЕСКИЕ ПРОБЛЕМЫ В СТРОИТЕЛЬСТВЕ.
ГЕОЭКОЛОГИЯ**

УДК 627.13

V.V. Degtyarev, D.I. Ershov

Sibstrin

В.В. Дегтярев, Д.И. Ершов

ФГБОУ ВПО «ИГАСУ (Сибстрин)»

**ЧИСЛЕННОЕ
МОДЕЛИРОВАНИЕ
ПРОЦЕССА
ОСАДКОНАКОПЛЕНИЯ
ТЕРРИГЕННОГО МАТЕРИАЛА
В УСТЬЕВЫХ ОБЛАСТЯХ РЕК**

**NUMERICAL MODELING
OF SEDIMENTATION
OF TERRIGENOUS MATERIAL
IN ESTUARINE AREAS
OF RIVERS**

Рассмотрены некоторые аспекты и общие проблемы, связанные с организацией параллельных вычислений в задачах гидродинамики. В качестве примера используется задача о вычислении поля скоростей в области смешения речных и морских вод и моделировании процесса осадконакопления терригенного материала в северо-восточном секторе Арктики с целью повышения эффективности и экологической безопасности судоходства.

Ключевые слова: параллельные вычисления, метод сплавленных частиц, численное моделирование, устьевые области, судоходство, экологическая безопасность, Арктика.

Решению транспортных проблем на берегах изолированных арктических рек Яны, Индигирки, Колымы и других, к сожалению, пока уделено недостаточно внимания, и конкретные предложения, планы и стратегии развития водного транспорта в арктической зоне отсутствуют.

Кардинальное решение проблем развития судоходства вдоль побережья Северного Ледовитого океана невозможно без необходимого обоснования трассирования судоходства на устьевых барах, для чего требуются методы

This article considers some aspects and common problems associated with the organization of parallel calculations in the problems of hydrodynamics. As an example the authors consider the calculation problem of the velocity field in the area of river and sea water mixing and the simulation of sedimentation of terrigenous material.

Key words: parallel computing, smoothed particle method, numerical simulation, estuarine areas, shipping, ecological safety, Arctic region.

Unfortunately insufficient attention is paid to the traffic problems in the regions on the isolated shores of the Arctic rivers Yana, Indigirka, Kolyma, and others, and there are no specific proposals, plans and strategies for the development of water transport in the Arctic zone.

The radical solution to the problems of shipping development along the coast of the Arctic ocean is impossible without the necessary justification of tracing the shipping routes on estuarine bars, which requires the methods of numerical simulation of the hydrophysical processes. In particular, the hydrodynamics of flow and sedimentation processes in the mixing zone of fresh river and saline marine waters.

The calculation distribution is transparent and in general an extremely

численного моделирования происходящих на них гидрофизических процессов. В частности, гидродинамики потока и процессов седиментации в зоне смешения пресных речных и осолоненных морских вод.

Распределение вычислений является прозрачным и в общем случае крайне эффективным способом решения многих задач в независимости от области применения, позволяя значительно повышать производительность и масштабируемость вычислений.

Задачей, на примере которой рассматривается оптимизация параллельных вычислений в данной статье, является расчет скоростного поля в области смешения речных и морских вод. Возможно, стоило бы дать перечень более абстрактных рекомендаций, но таковой информации достаточно в открытом доступе, включая базовую документацию на инструменты и стандарты разработки параллельных вычислительных систем. В любом случае оптимизация и рекомендации тем полезнее и эффективнее, чем ближе они к контексту конкретной задачи, что, конечно же, не исключает последующую экстраполяцию на иные области знаний.

Ранее авторами подробно рассматривалась поставленная задача и в качестве метода решения был выбран *SPH* — метод гидродинамики сглаженных частиц.

SPH (Smoothed Particles Hydrodynamics) — подход, рассматривающий жидкость как аппроксимацию ее физических свойств [1—3] на множестве дискретных объектов — частиц. Ни в коем случае нельзя путать эти объекты с молекулами или с трассировкой линий тока. Частицы в *SPH* — это именно узлы аппроксимации.

Любой физический параметр, который может быть сопоставлен точке сплошной среды, сопоставляется и частице — давление, плотность, вязкость и прочее. Значение любой физической величины A

effective way to solve many problems regardless of the application field, allowing significantly improving the performance and scalability of computing.

The task, which considered the optimization of parallel computations in this paper, is the calculation of the velocity field in the region of river and sea waters mixing. Perhaps it would be worthwhile to give a list of more abstract recommendations, but there is enough information in the public domain, including basic documentation on tools and standards for the development of parallel computing systems. In any case, optimization and recommendations are the better and the more effective, the closer they are to the context of a specific task, which of course does not preclude subsequent extrapolation to other areas of knowledge.

Previously, the authors discussed in detail the stated task and *SPH* was chosen as a solution method — a method of smoothed particle hydrodynamics.

SPH (Smoothed Particles Hydrodynamics) is an approach, considering the fluid as an approximation of its physical properties on the set of discrete objects — particles [1—3]. We shouldn't confuse these objects with molecules or marking of current flow lines. The particles in *SPH* are exactly the nodes of approximation.

Any physical parameter which can be compared to a point of continuous medium, is also compared to a particle — pressure, density, viscosity, etc. The value of any physical quantity A

для точки, заданной радиус-вектором r , может быть вычислено по следующей формуле [4—9]:

$$A(r) = \sum_j m_j \frac{A_j}{\rho_j} W(|r - r_j|, h), \quad (1)$$

где m_j — масса отдельной частицы; A_j — значение искомой величины, сопоставленной точке; ρ_j — плотность, сопоставленная частице; h — длина сглаживания данной частицы.

Далее введем следующие упрощения.

Рассматриваемые среды предполагаются несжимаемыми, поэтому для всех частиц значение плотности одинаково: $\forall_j : \rho_j = \rho = \text{const}$.

Длина сглаживания при расчете методом *SPH*, вообще говоря, является величиной непостоянной [10—12] и зависит от требуемой степени разрешения. Выбор критерия распределения этой величины не существенен в рамках данной статьи, поэтому примем величину длины сглаживания постоянной.

Также будем считать, что множество частиц состоит из равнозначных элементов и примем их массы равными.

Таким образом, формула (1) сводится к следующему выражению:

$$A = \sum_j m \frac{A_j}{\rho} W(|r - r_j|, h) = G \sum_j A_j F(dr_j), \quad (2)$$

где G — константа, зависящая от природы физической величины; $F(dr_j)$ — функция, зависящая только от расстояния между двумя частицами. Конкретный вид функции также зависит от природы физической величины.

Путем замены $f(dr_j) = CF(dr_j)$ получаем

$$A = \sum_j A_j f(dr_j). \quad (3)$$

Поскольку A_j и A имеют одинаковую размерность, то $f(dr_j)$ является

for a point specified by the radius vector r can be calculated by the following formula [4—9]:

where m_j is the mass of a single particle, A_j is the desired value that is compared to a point, ρ_j is the density compared to the particle, h is the smoothing length of a given particle;

Next, we introduce the following simplification:

Consider the fluid is assumed to be incompressible, so the density is the same for all the particles: $\forall_j : \rho_j = \rho = \text{const}$.

The smoothing length at calculating by the *SPH* method in general is not constant [10—12] and depends on the required degree of resolution. The selection of distribution criterion of this value is not significant in the context of this article, so we will take a smoothing length value as constant.

Also we assume that the set of particles consists of equivalent elements. Accept their masses are equal.

Thus the formula (1) is reduced to the following expression:

where G is a constant, dependent on the nature of the physical magnitude, $F(dr_j)$ is a function that depends only on the distance between the two particles. A specific kind of function also depends on the nature of physical quantities.

Using substitution $f(dr_j) = CF(dr_j)$ we get

Since A_j and A have the same dimension, $f(dr_j)$ is a dimensionless

безразмерной величиной. Таким образом, значение любой физической величины является взвешенной суммой от значений данной величины в частицах ее окрестности.

Приняв в качестве величины A давление p , путем несложных преобразований уравнения Навье — Стокса (дискретизации) получим

$$a_i = F_i^{press} + F_i^{visc} + g, \quad (4)$$

где a_i — ускорение i -й частицы; $F_i^{press} = \frac{-1}{2} \sum_j (p_j + p_j) \nabla f(dr_{i-j})$ — составляющая силы, являющаяся дискретизацией компоненты уравнения, включающей градиент давления; $F_i^{visc} = k \sum_j (u_j - u_j) \Delta f(dr_{i-j})$; g — постоянный вектор потенциального поля сил.

В совокупности с начальными условиями (исходными значениями положений и скоростей частиц) формула (4) позволяет реализовать интересующую модель движения жидкости. Однако для построения схемы расчета исключим несущественные с точки зрения алгоритмизации элементы формулы (4) и ее компонент, а также обратн абстрагируемся от физического смысла величины A .

Исключив все константные величины и отбросив связь между $f(dr_j)$ и ее производными, получим

$$\begin{cases} r_i^{k+1} = r_i^k - r_i^{k-1} + \tau^2 a_i^k; \\ u_i^{k+1} = \frac{r_i^{k+1} - r_i^k}{\tau}; \\ a_i^k = \sum_j (A_i^k + A_j^k) G_1(|r_i^k - r_j^k|) + (u_j^k - u_i^k) G_2(|r_i^k - r_j^k|). \end{cases} \quad (5)$$

Моделирование движения частиц растворенного вещества осуществляется аналогично, совместно с вычислением скоростных характеристик. Ключевые отличия модели раство-

quantity. Thus the value of any physical quantity is the weighted sum of the values of this magnitude in the particles of its vicinity.

Taking the pressure p as the parameter A by a simple transformation of the Navier-Stokes equations (discretization), we obtain

where a_i is the acceleration of the i -th particle. $F_i^{press} = \frac{-1}{2} \sum_j (p_j + p_j) \times \nabla f(dr_{i-j})$ — a component of the force, which is the discretization of the equation components, including the pressure gradient $F_i^{visc} = k \sum_j (u_j - u_j) \Delta f(dr_{i-j})$, g is a constant vector of potential forces field.

Together with the initial conditions (initial values of the positions and velocities of the particles) the formula (4) allows implementing the fluid flow model of interest. However, for calculation plotting we exclude the formula elements irrelevant from the point of view of algorithmization (4) and its component, as well as we abstract back from the physical sense of the value A .

Excluding all the constant values and discarding the relationship between $f(dr_j)$ and its derivatives we will receive

Modeling the motion of solute particles is the same, together with the calculation of speed characteristics. The key differences of the solute model from the approximated liquid are:

ренного вещества от аппроксимируемой жидкости:

частицы растворенного вещества не участвуют в формулах, определяющих ускорение сглаженных частиц;

частицы растворенного вещества не являются сглаженными [13];

обработка столкновения с границей для частиц растворенного вещества выражается в накоплении осадка — изменения формы поверхности границы.

Для построения траектории частиц использован метод интегрирования Верле [14]. Полученной системы достаточно для построения схемы вычислений, но предварительно следует отметить следующее:

в качестве начальных условий требуется совокупность (r_i^0, u_i^0) для каждой частицы. Значение r_i^{-1} может быть вычислено путем обращения формулы для u_i^0 ;

величина шага по времени τ предполагается в каждый момент времени постоянной;

функции G_1 и G_2 известны;

величины A_i^k рассчитываются перед каждой итерацией в соответствии с известным алгоритмом;

граничные условия выражаются в обработке столкновения сглаженных частиц с заданной поверхностью — дном [15].

Таким образом, получаем следующую последовательность расчета:

1) начало k -й итерации ($k > 0$). Расчет величин A_i^k ;

2) расчет значений a_i^k ;

3) расчет (r_i^{k+1}, u_i^{k+1}) . Возврат к пункту 1.

Как видно из формулы (5), первые две операции являются вариантом метода *Map* параллельных вычислений и могут быть реализованы посредством стандартных инструментов любой платформы.

the solute particles do not participate in the formulas determining the acceleration of the smoothed particles;

the particles of the solute are not smoothed [13];

handling collisions with the boundary for the solute particles is expressed in the accumulation of precipitate — change of the surface shape of the border.

In order to build the trajectories of the particles the Verlet integration method was used [14]. The resulting system is enough to build a computation scheme, but beforehand we should note the following:

as the initial conditions the set of (r_i^0, u_i^0) is required for each particle. The value r_i^{-1} can be calculated by applying the formula for u_i^0 ;

the step size in time τ is assumed constant in each point in time;

the functions G_1 and G_2 are known;

the A_i^k values are calculated before each iteration in accordance with the known algorithm;

the boundary conditions are expressed in handling the collision of smoothed particles with a given surface — the bottom.

Thus, we get the following sequence of calculations:

1) the beginning of the k -th iteration ($k > 0$). The calculation of A_i^k values;

2) calculation of a_i^k values;

3) the calculation of (r_i^{k+1}, u_i^{k+1}) .

Return to step 1.

As it can be seen from the formula (5) the first two steps are variants of the *Map* method of the parallel calculations and can be implemented through standard tools of any platform.

Расчет же величин a_i^k является сочетанием методов *Map* и *Reduce* [16—18]. При этом *Reduce* производится на множестве элементов, зависящем от рассматриваемого узла и итерации вычисления, а именно — конфигурации узлов в окрестности. Под окрестностью подразумевается множество узлов, для которых справедливо равенство

$$|r_i - r_j| < 2h, \quad (6)$$

где h — постоянная величина.

Для осуществления редукции использовался алгоритм для ассоциативного оператора, так как именно этот вариант был ближе к архитектуре использованной платформы.

Реализация модели производилась для общего случая. Результатом стала трехуровневая библиотека разработки. Для осуществления параллельных вычислений использовалась технология *OpenCL*.

Состав логических уровней библиотеки

1. Объектно-ориентированная оболочка. Логика *OpenCL* организована в традициях императивной парадигмы, что делает крайне неудобным организацию емких вычислений [19, 20]. Например, покомандная обработка ошибок может привести к огромному объему исходного кода для решения относительно простых задач. Данный уровень библиотеки вводит самостоятельные классы, представляющие собственно платформу вычислений, контекст используемой памяти, контекст вычислений и прочее.

2. Абстракция вычислителя. Задача данного уровня инкапсулировать подготовку данных и контекста исполнения непосредственно перед осуществлением вычислений. Также на данный уровень возложена типиза-

The calculation of a_i^k values is a combination of *Map* and *Reduce* methods [16—18]. The *Reduce* is performed on the set of elements which depends on the considered node and iteration of the calculation, namely the configuration of nodes in the neighborhood. By the neighborhood to the set of nodes is implied, for which the equality is true

where h is a constant value.

For the reduction the algorithm for associative operator was used, since this version was closer to the architecture of the used platform.

The implementation of the model was performed for the general case. The result was a three-level development library. For implementation of parallel computing the technology *OpenCL* was used.

The composition of the logical levels of the library:

1. Object-oriented shell. The *OpenCL* logic is organized in the tradition of the imperative paradigm, which makes the organization of intensive computing extremely inconvenient [19, 20]. For example, sequential processing of errors can lead to huge amount of source code for solving relatively simple problems. This library level introduces independent classes that represent the actual computing platform, the space context, the context of computing and so on.

2. The abstraction of the calculator. The objective of this level is to encapsulate preparation of data and execution context directly before performing calculations. Also this level is responsible for typing of memory objects on the user side of the library. The garbage collection of the subsidiary object is implemented.

ция объектов памяти на стороне пользователя библиотеки. Реализована сборка мусора вспомогательных объектов.

3. Инструментарий. Данный уровень представляет собой еще один уровень объектно-ориентированной абстракции, инкапсулирующий работу с векторами, матрицами и прочее. Также разработан отдельный интерфейс для реализации вычислений методом *SPH*.

Метод *SPH* показал себя достаточно гибким и производительным, позволяя с легкостью внедрять новые гидрометеорологические факторы. На данный момент разработана альфа-версия библиотеки, позволяющей осуществлять разработку приложений для моделирования движения жидкости в устьевых областях рек и осадконакопления терригенного материала.

Библиографический список

1. *Anderson J.D., Jr.* Computational Fluid Dynamics. The basics with applications. 1 edition. McGraw-Hill Science/Engineering/Math, February 1, 1995. 574 p.
2. *Takeda H., Miyama S., Sekiya M.* Numerical simulation of viscosity using SPH // *Prog. Theor. Phys.* 1994. Vol. 92. Pp. 939—960.
3. *Verlet L.* Computer Experiments on Classical Fluids I: Thermodynamical Properties of Lennard-Jones Molecules // *Physics Review.* 1967. Vol. 159. Pp. 98—103.
4. *Kelager M.* Lagrangian Fluid Dynamics Using Smoothed Particle Hydrodynamics. University of Copenhagen : Department of Computer Science, 2006. 81 p.
5. *Kruger J., Westermann R.* Linear algebra operators for GPU implementation of numerical algorithms // *ACM Transactions on Graphics.* 2003. Vol. 22. No. 3. Pp. 908—916.
6. *Liu G.R., Liu M.B.* Smoothed particles hydrodynamics. A meshfree particle method. National University of Singapore : World Scientific Publishing, 2003. 473 p.

3. Toolkit. This level represents another level of object-oriented abstraction that encapsulates the work with vectors, matrices, etc. Also a separate interface for the implementation of calculations by *SPH* method is developed.

SPH method proved to be highly flexible and efficient, allowing easily implementing new hydrometeorological factors. At the moment an alpha version of the library is developed that enables the development of applications for modeling the fluid flow in the estuarine areas of the rivers and sedimentation of terrigenous material.

References

1. *Anderson J.D., Jr.* Computational Fluid Dynamics. The Basics with Applications. 1 edition. McGraw-Hill Science/Engineering/Math, February 1, 1995, 574 p.
2. *Takeda H., Miyama S., Sekiya M.* Numerical Simulation of Viscosity Using SPH. *Prog. Theor. Phys.* 1994, vol. 92, pp. 939—960.
3. *Verlet L.* Computer Experiments on Classical Fluids I: Thermodynamical Properties of Lennard-Jones Molecules. *Physics Review.* 1967, vol. 159, pp. 98—103. DOI: <http://dx.doi.org/10.1103/PhysRev.159.98>.
4. *Kelager M.* Lagrangian Fluid Dynamics Using Smoothed Particle Hydrodynamics. University of Copenhagen : Department of Computer Science, 2006, 81 p.
5. *Kruger J., Westermann R.* Linear Algebra Operators for GPU Implementation of Numerical Algorithms. *ACM Transactions on Graphics.* 2003, vol. 22, no. 3, pp. 908—916. DOI: <http://dx.doi.org/10.1145/1201775.882363>.
6. *Liu G.R., Liu M.B.* Smoothed Particles Hydrodynamics. A Meshfree Particle Method. National University of Singapore, World Scientific Publishing, 2003, 473 p.
7. *Monaghan J.J.* Particle Methods for Hydrodynamics. *Comput. Phys.* 1985, Rep. 3. Pp. 71—124.
8. *Monaghan J.J.* SPH and Riemann solvers. *J. Comput. Phys.* 1997, vol. 136,

7. *Monaghan J.J.* Particle methods for hydrodynamics // *Comput. Phys.* 1985. Rep. 3. Pp. 71—124.
8. *Monaghan J.J.* SPH and Riemann solvers // *J. Comput. Phys.* 1997. Vol. 136. No. 2. Pp. 298—307.
9. *Monaghan J.J.* SPH without a tensile instability // *J. Comput. Phys.* 2000. Vol. 159. No. 2. Pp. 290—311.
10. *Monaghan J.J.* Smoothed particles hydrodynamics. Monash University, Australia : School of Mathematical Sciences, 1992. 68 p.
11. *Muller M., Solenthaler B., Keiser R., Gross M.* Particle-based fluid-fluid interaction // *Proc. of SIGGRAPH Symposium on Computer Animation.* 2005. Pp. 237—244.
12. *Rapaport D.C.* The art of molecular dynamics simulation. Cambridge University Press, 1995. 564 p.
13. *Hernquist L., Katz N.* TREESPH — A unification of SPH with the hierarchical tree method // *Astrophys. J. (Suppl.).* 1989. Vol. 70. Pp. 419—446.
14. *Harlow F.H.* The particle-in-cell method for numerical solution of problems in fluid dynamics // *Proceedings of Symposia in Applied Mathematics.* 1963. Vol. 15. No. 10. Pp. 269—288.
15. *Balsara D.S.* Von Neumann stability analysis of smooth particle hydrodynamics — suggestions for optimal algorithms // *J. Comput. Phys.* 1995. Vol. 121. No. 2. Pp. 357—372.
16. *Hoover W.G.* Isomorphism linking smooth particles and embedded atoms // *Physica A: Statistical Mechanics and its Applications.* 1998. Vol. 260. No. 3—4. Pp. 244—254.
17. *Hoover W.G.* Computational Statistical Mechanics. Amsterdam : Elsevier Science Publisher, 1991. 324 p.
18. *Johnson G.R., Beissel S.R.* Normalized smoothed functions for SPH impact computations // *International Journal for Numerical Methods in Engineering.* 1996. Vol. 39. No. 16. Pp. 2725—2741.
19. *Harlow F.H.* The particle-in-cell computing method for fluid dynamics // *Methods in Computational Physics, Vol. 3 : Numerical Methods in Hydrodynamics.* New York : Academic Press, 1964. Pp. 319—343.
- no. 2, pp. 298—307. DOI: <http://dx.doi.org/10.1006/jcph.1997.5732>.
9. *Monaghan J.J.* SPH without a Tensile Instability. *J. Comput. Phys.* 2000, vol. 159, no. 2, pp. 290—311. DOI: <http://dx.doi.org/10.1006/jcph.2000.6439>.
10. *Monaghan J.J.* Smoothed Particles Hydrodynamics. Monash University, Australia, School of Mathematical Sciences, 1992, 68 p.
11. *Muller M., Solenthaler B., Keiser R., Gross M.* Particle-Based Fluid-Fluid Interaction. *Proc. of SIGGRAPH Symposium on Computer Animation.* 2005, pp. 237—244. DOI: <http://dx.doi.org/10.1145/1073368.1073402>.
12. *Rapaport D.C.* The Art of Molecular Dynamics Simulation. Cambridge University Press, 1995, 564 p.
13. *Hernquist L., Katz N.* TREESPH — A Unification of SPH with the Hierarchical Tree Method. *Astrophys. J. (Suppl.).* 1989, vol. 70, pp. 419—446. DOI: <http://dx.doi.org/10.1086/191344>.
14. *Harlow F.H.* The Particle-In-Cell Method for Numerical Solution of Problems in Fluid Dynamics. *Proceedings of Symposia in Applied Mathematics.* 1963, vol. 15, no. 10, pp. 269—288.
15. *Balsara D.S.* Von Neumann Stability Analysis of Smooth Particle Hydrodynamics — Suggestions for Optimal Algorithms. *J. Comput. Phys.* 1995, vol. 121, no. 2, pp. 357—372. DOI: [http://dx.doi.org/10.1016/S0021-9991\(95\)90221-X](http://dx.doi.org/10.1016/S0021-9991(95)90221-X).
16. *Hoover W.G.* Isomorphism Linking Smooth Particles and Embedded Atoms. *Physica A: Statistical Mechanics and its Applications.* 1998, vol. 260, no. 3—4, pp. 244—254. DOI: [http://dx.doi.org/10.1016/S0378-4371\(98\)00357-4](http://dx.doi.org/10.1016/S0378-4371(98)00357-4).
17. *Hoover W.G.* Computational Statistical Mechanics. Amsterdam, Elsevier Science Publisher, 1991, 324 p.
18. *Johnson G.R., Beissel S.R.* Normalized Smoothed Functions for SPH Impact Computations. *International Journal for Numerical Methods in Engineering.* 1996, vol. 39, no. 16, pp. 2725—2741. DOI: [http://dx.doi.org/10.1002/\(SICI\)1097-0207\(19960830\)39:16<2725::AID-NME973>3.0.CO;2-9](http://dx.doi.org/10.1002/(SICI)1097-0207(19960830)39:16<2725::AID-NME973>3.0.CO;2-9).

20. Harris M. Fast fluid dynamics simulation on the GPU // GPU Gems. NVIDIA, 2004. Pp. 637—665.

Поступила в редакцию в декабре 2014 г.

Об авторах: **Дегтярев Владимир Владимирович** — доктор технических наук, профессор, заведующий кафедрой гидротехнических сооружений и гидравлики, проректор по научной работе, **Новосибирский государственный архитектурно-строительный университет (ФГБОУ ВПО «НГАСУ (Сибстрин)»)**, 630008, г. Новосибирск, ул. Ленинградская, д. 113, gts@substrin.ru;

Ершов Дмитрий Игоревич — аспирант кафедры гидротехнических сооружений и гидравлики, **Новосибирский государственный архитектурно-строительный университет (ФГБОУ ВПО «НГАСУ (Сибстрин)»)**, 630008, г. Новосибирск, ул. Ленинградская, д. 113, mostthemain@gmail.com.

Для цитирования: *Дегтярев В.В., Ершов Д.И.* Численное моделирование процесса осадконакопления терригенного материала в устьевых областях рек // Вестник МГСУ. 2015. № 6. С. 64—72.

19. Harlow F.H. The Particle-In-Cell Computing Method for Fluid Dynamics. Methods in Computational Physics. Vol. 3 : Numerical Methods in Hydrodynamics. Academic Press, New York, 1964, pp. 319—343.

20. Harris M. Fast Fluid Dynamics Simulation on the GPU. GPU Gems. NVIDIA, 2004, pp. 637—665.

Received in december 2014.

About the authors: **Degtyarev Vladimir Vladimirovich** — Doctor of Technical Sciences, Professor, chair, Department of Hydraulic Engineering Structures and Hydraulics, **Novosibirsk State University of Architecture and Civil Engineering (Sibstrin)**, 113 Leningradskaya str., Novosibirsk, 630008, Russian Federation; gts@substrin.ru;

Ershov Dmitriy Igorevich — post-graduate student, Department of Hydraulic Engineering Structures and Hydraulics, **Novosibirsk State University of Architecture and Civil Engineering (Sibstrin)**, 113 Leningradskaya str., Novosibirsk, 630008, Russian Federation; mostthemain@gmail.com.

For citation: *Degtyarev V.V., Ershov D.I.* Numerical Modeling of Sedimentation of Terrigenous Material in Estuarine Areas of Rivers. *Vestnik MGSU* [Proceedings of Moscow State University of Civil Engineering]. 2015, no. 6, pp. 64—72.