Bull. Nov. Comp. Center, Num. Anal., 15 (2011), 61–71

Computation of gravitational potential of isolated systems in cylindrical coordinates*

N.V. Snytnikov

Abstract. We present the new method for computation of a gravitational potential for isolated systems in cylindrical coordinates. This method solves the main difficulty arising when treating isolated systems: in order to correctly state the Dirichlet problem for the Poisson equation at the boundary of a finite computational domain, one must provide the boundary values of a gravitational potential which are unknown.

To develop this method, we adapt the ideas of the convolution method and the James algorithm to the Cartesian coordinates and rectangular computational domains. To solve the Dirichlet problem for the Poisson equation, we use a finite difference 7-point stencil. System of linear equations obtained after a difference approximation is solved by means of Fast Fourier transform applied to the azimuthal coordinate, Fast sine transform for the vertical coordinate and 3-diagonal elimination to determine the radial component of potential.

1. Introduction

In order to simulate the dynamics of galaxies or circumstellar disks it is needed to solve systems of differential equations for the motion of a medium. Although the motion may have a different nature for different processes (e.g. collisionless or hydrodynamic), a common part is that the force is produced by gravitation, which is described by the Poisson equation. Its right-hand side, the density of the matter, has a finite support (density is non-zero only for some bounded domain) and the natural boundary conditions are written down in the form:

$$\Delta\Phi(\boldsymbol{r}) = 4\pi G\rho(\boldsymbol{r}), \qquad \Phi|_{\boldsymbol{r}\to\infty} = 0, \tag{1}$$

where G is a gravitational constant.

Formally, the Dirichlet problem for the Poisson equation is correctly stated, but in practice, in numerical experiments, one deals with a domain bounded in space, so the natural boundary condition cannot be directly used. One popular approach is to use a well-known fundamental solution of the Poisson equation:

*Supported by the Russian Foundation for Basic Research under Grant 08-01-615 and by the SB RAS Integration Project 26.

$$\Phi(\boldsymbol{r}) = -G \int \frac{\rho(\boldsymbol{r}')\, d\boldsymbol{r}'}{|\boldsymbol{r} - \boldsymbol{r}'|}. \tag{2}$$

The direct computation of this integral requires $O(N^3)$ operations for one spatial point $\boldsymbol{r}$ (where N is taken to be the number of grid nodes along one direction; for simplicity we assume the number of nodes to be of the same order for all the directions). If a potential is needed for any point of a 3D computational domain, then the total number of operations is equal to $O(N^6)$.

This great number of computations can be significantly decreased to $O(N^3 \log N)$ with Hockney's method based on the convolution theorem and Fast Fourier transform, which requires doubling a grid size for each direction and modifying a gravitational potential at close distances [1]. However, the straightforward implementation for cylindrical coordinates has a complexity of $O(N^4 \log N)$ [2], because the convolution theorem can be applied only for azimuthal and vertical coordinates but not for radial coordinates (this will be discussed in more detail in Section 3).

Another approach to solve (1) is to specify values of the gravitational potential Φ_Γ for the boundary Γ of the given computational domain Ω:

$$\Delta\Phi(\boldsymbol{r}) = 4\pi G\rho(\boldsymbol{r}), \quad \boldsymbol{r} \in \Omega, \qquad \Phi|_\Gamma = \Phi_\Gamma, \tag{3}$$

and then to apply one of the well-known methods of solving the Dirichlet problem for the Poisson equation [3, 4]. Apparently, the complexity of the computation Φ_Γ must be of the same order as that of solving (3).

There are three different ways to compute Φ_Γ:

Direct calculation. Using formulas (2), we can compute only boundary values of the potential. It takes $O(N^5)$ operations for an arbitrary density distribution and can hardly be used for the general case. However, if density is known to be non-zero only in some plane, then complexity is decreased to $O(N^4)$ [5].

Computation of approximated values using multipole moments. Under a reasonable assumption of the mass sources concentration near to the center of a domain, we can compute approximated boundary values of a potential using its multipole expansion [6–8]. The method takes $O(N^3)$, but its main drawback is that to attain desired precision (for example, 1 %), boundaries should be moved back by a factor of 3 from the place, where the main part of mass sources is concentrated. Another drawback of this approach is in the difficulty of using it in the domain decomposition techniques.

The James method. The fastest and most powerful method to deal with the potential of isolated systems in the Cartesian coordinates is the James method [9] based on the well-known phenomenon of electrostatics. If a charged body is encompassed by an earthen conductor, then charges in this conductor will be redistributed in such a way as to provide the total zero

potential in the conductor (these charges will be further denoted as *screening charges*). This phenomenon combined with Hockney's algorithm leads to the following method:

1. Solve the Poisson equation with zero Dirichlet boundary conditions imposed on Γ:

$$\Delta\Phi_0(\boldsymbol{r}) = 4\pi G\rho(\boldsymbol{r}), \quad \boldsymbol{r} \in \Omega, \qquad \Phi_0|_\Gamma = 0. \tag{4}$$

2. Taking into account the fact that Φ_0 is zero outside of Ω, apply a difference Laplace operator to Φ_0 at the boundary nodes of Γ to obtain density of screening charges.
3. Compute potential Φ_1 in Ω generated by the surface screening charges using the idea of Hockney's algorithm (the convolution method).
4. The desired potential $\Phi = \Phi_0 - \Phi_1$.

In other words, the James algorithm reduces the computation of a boundary potential generated by the density $\rho \in \Omega$ to the computation of a volume potential generated by screening charges located on the surface Γ. The latter turns out to be simpler because of the convolution method adaptation (but of course it is not evident at first glance).

However, as a matter of fact, for cylindrical coordinates, Step 3 of the James algorithm cannot be directly employed because of the abovementioned impossibility of using the convolution theorem.

The present paper is focused on the adaptation of the James method for the cylindrical coordinates. Also, we present our implementation of this method for solving the Dirichlet problem for the Poisson equation (Section 2). The method of screening charges computation is described in Section 3, while convolution method and its application to our needs is discussed in Section 4. A brief description of the combined algorithm and its complexity is discussed in Section 5. The main results and application of the created method are summarized in conclusion.

2. The Poisson equation solver

Assume that we have a cylindrical computational domain:

$$\Omega = \big\{(r, \phi, z) : \ r \in [R_0, R_{\max}], \ \phi \in [0, 2\pi], \ z \in [Z_0, Z_{\max}]\big\}.$$

If $R_0 > 0$, this means that the central cylindrical domain is cut off and the boundary conditions should be set for both the exterior ($R_{\max}$) and the interior (R_0) boundary faces of the cylinder.

The Poisson equation in the cylindrical coordinates is written down as

$$\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial \Phi}{\partial r}\right)+\frac{1}{r^2}\frac{\partial^2 \Phi}{\partial \phi^2}+\frac{\partial^2 \Phi}{\partial z^2}=\rho, \tag{5}$$

where the right-hand side ρ already contains the factor 4π, and the gravitational constant G is taken to be unity in dimensionless variables. We consider the Dirichlet conditions to be specified for the boundary Γ.

To solve this equation, we use a finite difference method. In the domain Ω, we introduce a grid with $N_r \times N_\phi \times N_z$ nodes:

$$\begin{aligned} r_{i-1/2} &= R_0 + (i-1/2)h_r, & i &= 0,\ldots,N_r, & h_r &= \frac{R_{\max}-R_0}{N_r},\\ \phi_k &= kh_\phi, & k &= 0,\ldots,N_\phi, & h_\phi &= \frac{2\pi}{N_\phi},\\ z_l &= Z_0 + lh_z, & l &= 0,\ldots,N_z, & h_z &= \frac{Z_{\max}-Z_0}{N_z}. \end{aligned}$$

We assume that N_r, N_ϕ, N_z are of the same order (denoted as N). To approximate equation (5), the following difference scheme is used:

$$\begin{aligned} &\frac{1}{h_r^2 r_{i-1/2}}\left[r_i(\Phi_{i+1/2,k,l}-\Phi_{i-1/2,k,l})-r_{i-1}(\Phi_{i-1/2,k,l}-\Phi_{i-3/2,k,l})\right]+\\ &\quad\frac{1}{h_\phi^2 r_{i-1/2}^2}\left(\Phi_{i-1/2,k+1,l}-2\Phi_{i-1/2,k,l}+\Phi_{i-1/2,k-1,l}\right)+\\ &\qquad\frac{1}{h_z^2}\left(\Phi_{i-1/2,k,l+1}-2\Phi_{i-1/2,k,l}+\Phi_{i-1/2,k,l-1}\right)=\rho_{i-1/2,k,l},\\ &\qquad\qquad i=1,\ldots,N_r,\ \ k=1,\ldots,N_\phi,\ \ l=1,\ldots,N_z. \end{aligned} \tag{6}$$

From the boundary conditions it follows that the values $\Phi_{-1/2,k,l}$, $\Phi_{N_r-1/2,k,l}$, $\Phi_{i-1/2,k,0}$, $\Phi_{i-1/2,k,N_z}$ are known and $\Phi_{i,0,l}=\Phi_{i,N_\phi,l}$ for $i=0,\ldots,N_r$, $l=0,\ldots,N_z$.

If $R_0=0$, then a cylinder domain does not contain a cut off area in its center. In this case, the interior values $\Phi_{-1/2,k,l}$ are not needed at all (the property of a difference scheme shifted at $1/2$ of a node [11]).

Linear system of equations (6) is solved in a way based on the ideas of [6,10].

First we construct a homogeneous problem with zero boundary conditions by modifying the right-hand side:

$$\begin{aligned} \tilde{\rho}_{1/2,k,l} &= \rho_{1/2,k,l}-\frac{1}{h_r^2}\frac{r_0}{r_{1/2}}\Phi_{-1/2,k,l},\\ \tilde{\rho}_{N_r-3/2,k,l} &= \rho_{N_r-3/2,k,l}-\frac{1}{h_r^2}\frac{r_{N_r-1}}{r_{N_r-3/2}}\Phi_{N_r-1/2,k,l}, \end{aligned}$$

$$\tilde{\rho}_{i-1/2,k,1} = \rho_{i-1/2,k,1} - \frac{1}{h_z^2}\Phi_{i-1/2,k,1},$$
$$\tilde{\rho}_{i-1/2,k,N_z-1} = \rho_{i-1/2,k,N_z-1} - \frac{1}{h_z^2}\Phi_{i-1/2,k,N_z-1}.$$

Then we write discrete Fourier series for Φ and ρ:

$$\Phi_{i-1/2,k,l} = \sum_{m=0}^{N_\phi-1} F_{i-1/2,l}(m)\exp\left(-\frac{2\pi\bar{i}km}{N_\phi}\right),$$
$$\rho_{i-1/2,k,l} = \sum_{m=0}^{N_\phi-1} G_{i-1/2,l}(m)\exp\left(-\frac{2\pi\bar{i}km}{N_\phi}\right)$$

(where $\bar{i}$ denotes an imaginary unit) and substitute them into equation (6) thus obtaining a difference scheme with regard to the complex azimuthal harmonics:

$$\frac{1}{h_r^2 r_{i-1/2}}\left[r_i(F_{i+1/2,l}(m) - F_{i-1/2,l}(m)) - r_{i-1}(F_{i-1/2,l}(m) - F_{i-3/2,l}(m))\right] -$$
$$\frac{1}{h_\phi^2 r_{i-1/2}^2}F_{i-1/2,l}\sin^2\frac{\pi m}{N_\phi} +$$
$$\frac{1}{h_z^2}\left(F_{i-1/2,l+1}(m) - 2F_{i-1/2,l}(m) + F_{i-1/2,l-1}(m)\right)$$
$$= 4\pi G_{i-1/2,l}(m), \tag{7}$$
$$i = 1,\ldots,N_r,\ \ m = 0,\ldots,N_\phi - 1,\ \ l = 1,\ldots,N_z.$$

The true number of independent equations is $N_\phi/2$, because a harmonics with wavenumbers exceeding $N_\phi/2$ are a complex conjugation for harmonics with wavenumbers less than $N_\phi/2$, so the corresponding equations are redundant. Harmonics with wavenumbers 0 and $N_\phi/2$ are purely real (both properties follow from the properties of the complex discrete Fourier series for purely real functions).

At the next step, we apply a discrete sine transform in the vertical direction:

$$F_{i-1/2,l}(m) = \sum_{n=1}^{N_z-1} H_{i-1/2}(m,n)\sin\frac{\pi ln}{N_z},$$
$$G_{i-1/2,l}(m) = \sum_{n=1}^{N_z-1} R_{i-1/2}(m,n)\sin\frac{\pi ln}{N_z}.$$

Substituting into(7) we obtain $(N_z-1)\times N_\phi$ independent three-diagonal systems of equations:

$$\frac{r_i}{h_r^2 r_{i-1/2}} H_{i+1/2}(m,n) +$$
$$\left(-\frac{2}{h_r^2} - \frac{1}{h_\phi^2 r_{i-1/2}^2} 4\sin^2\frac{\pi m}{N_\phi} - \frac{1}{h_z} 4\sin^2\frac{\pi n}{2N_z}\right) H_{i-1/2}(m,n) +$$
$$\frac{r_{i-1}}{h_r^2 r_{i-1/2}} H_{i-3/2}(m,n) = R_{i-1/2}(m,n). \tag{8}$$
$$i = 1,\ldots,N_r,\ \ m = 0,\ldots,\frac{N_\phi}{2} - 1,\ \ n = 1,\ldots,N_z - 1,$$

that are solved by a three-diagonal elimination method.

Now we can summarize the steps of the Poisson equation solver:

1. Modify the initial inhomogeneous boundary conditions to homogeneous conditions.
2. Apply the Fast Fourier transform to $\rho(r_{i-1/2}, \phi_k, z_l)$ and obtain $G(r_{i-1/2}, z_l)(m)$.
3. Apply the Fast sine transform to $G(r_{i-1/2}, z_l)(m)$ and obtain $R(r_{i-1/2})(m,n)$.
4. Solve independent linear systems of equations with a three-diagonal elimination and obtain $H(r_{i-1/2})(m,n)$.
5. Apply the inverse Fast sine transform to $H(r_{i-1/2})(m,n)$ and obtain $F(r_{i-1/2}, z_l)(m)$.
6. Apply the inverse Fast Fourier transform to $F(r_{i-1/2}, z_l)(m)$ and obtain $\Phi(r_{i-1/2}, \phi_k, z_l)$.

3. Screening charges calculation

Assume that we have a solution Φ_0 of homogeneous problem (4). The simplest way to calculate screening charges is to apply a difference Laplace operator (the left-hand side of (6)) to Φ_0 at the boundary Γ (i.e., to the nodes $i = N_r$, $l = 0, N_z$), taking into account the fact that Φ_0 is equal to zero at Γ and outside of Ω.

However, it is more convenient to use the following formulas for the surface density of screening charges [12]:

$$\sigma(\Gamma) = -\frac{1}{4\pi}\frac{\partial \Phi_0}{\partial \boldsymbol{n}}, \tag{9}$$

where $\boldsymbol{n}$ is an outward surface normal. We can approximate it by the directed second order finite difference, which is written, for example, for the lateral exterior boundary Γ_{R_0} and the "floor" boundary Γ_{Z_0}in the following way:

$$\begin{aligned}\sigma(\Gamma_{R_0}) = -\frac{1}{4\pi}\frac{1}{2h_r}\big(&\Phi_0(r_{N_r-5/2},\phi_k,z_l)- \\ &4\Phi_0(r_{N_r-3/2},\phi_k,z_l)+3\Phi_0(r_{N_r-1/2},\phi_k,z_l)\big), \\ \sigma(\Gamma_{Z_0}) = -\frac{1}{4\pi}\frac{1}{2h_z}\big(&-\Phi_0(r_{i-1/2},\phi_k,z_2)+ \\ &4\Phi_0(r_{i-1/2},\phi_k,z_1)-3\Phi_0(r_{i-1/2},\phi_k,z_0)\big).\end{aligned} \tag{10}$$

4. Calculation of the potential generated by screening charges

The key idea of the James algorithm is the method of a volume potential calculation generated by the screening charges. To use this idea in the cylindrical coordinates we have reformulated the problem in the following way.

Let us consider an arbitrary domain Ω_1 with the boundary Γ_1 such as $\Omega_1 \supset \Omega$. The task is to find the potential Φ_{Γ_1} produced by surface screening charges located in Γ.

It is easy to see that the direct calculation takes $O(N^4)$ operations, that is not satisfactory. To compute the potential Φ_{Γ_1} efficiently, we have to use the convolution theorem [13]. First let us rewrite (2) in the following way:

$$\Phi_(\boldsymbol{r}) = -\int \rho(\boldsymbol{r}')K(|\boldsymbol{r}-\boldsymbol{r}'|)\,d\boldsymbol{r}' = -\rho * K, \tag{11}$$

where $K(|\boldsymbol{r}-\boldsymbol{r}'|) = \dfrac{1}{|\boldsymbol{r}-\boldsymbol{r}'|}$ and $\rho * K$ denotes the integral convolution.

If an integrand in (11) is absolutely integrable and the following integrals exist (i.e., there are bounding constants C_1 and C_2):

$$\begin{aligned}\int \rho(\boldsymbol{r}')\,d\boldsymbol{r}' < C_1, \\ \int K(\boldsymbol{r}')\,d\boldsymbol{r}' < C_2,\end{aligned} \tag{12}$$

then it follows that

$$\mathrm{FT}[\Phi](\boldsymbol{k}) = -\,\mathrm{FT}[\rho](\boldsymbol{k})\cdot\mathrm{FT}[K](\boldsymbol{k}), \quad \Phi = -\,\mathrm{FT}^{-1}\left[\mathrm{FT}[\rho]\cdot\mathrm{FT}[K]\right], \tag{13}$$

where $FT[\cdot]$ is a Fourier transform. This means that in order to compute integral (11), we have to compute the Fourier coefficients for the functions K and ρ, multiply them and apply the inverse Fourier transform. This theorem holds for discrete functions and discrete Fourier transform as well.

It is very useful if one needs to compute the integral for the values of Φ defined on the same grid as values of ρ. In this case, the total number

of operations can be reduced to $O(N)\log N$ because of the Fast Fourier transform algorithm, while the direct computation takes $O(N^2)$.

Thus if we apply a discrete analog of the convolution theorem to (2) defined on the Cartesian grid, then the computation complexity could be reduced to $O(N^3 \log N)$. However it is not easy to do it, because of the two important obstacles. The first obstacle is a divergence of integral (12), when $\boldsymbol{r}' = 0$. To overcome this problem some authors [1, 2, 14] used a modified Newtonian potential, by cutting off the interaction force at close distances to provide $K(0) < \infty$.

The second obstacle appears due to the fact that a discrete Fourier transform deals with periodic functions, while both K and ρ are not periodic. To walk around this problem, Hockney [15] proposed to extend a grid in all directions by a factor of 2 and defined both functions in a specific way. It was proved that this method gave a correct result for the potential Φ defined on the initial grid.

Now let us consider the convolution method in the cylindrical coordinates. Integral (2) takes the form

$$\Phi(r,\phi,z) = -\int_0^{\infty}\int_0^{2\pi}\int_{-\infty}^{\infty} \frac{r'\rho(r',\phi',z')\,dr'\,d\phi'\,dz'}{\sqrt{r^2 + r'^2 - 2rr'\cos(\phi-\phi') + (z-z')^2}}. \qquad (14)$$

Its kernel can be represented in the form of (11) only for z and ϕ coordinates. So, the computation complexity will be of order $O(N^4)\log N$, because the potential values for a radial coordinate can be computed only by the direct summation.

However, we can adapt this method to compute potential values generated by the screening charges. Consider a "ring" of nodes belonging to Γ_1 with the cylindrical coordinates

$$(r_{i_1}, \phi_{k_1}, z_{l_1}), \quad k_1 = 0, \ldots, N_\phi - 1,$$

and a "ring" of nodes belonging to Γ:

$$(r_{i_2}, \phi_{k_2}, z_{l_2}), \quad k_2 = 0, \ldots, N_\phi - 1.$$

Now we denote $\Phi_{i_2,l_2}(r_{i_1}, \phi_{k_1}, z_{l_1})$ as a potential, generated by screening charges located in the ring indices $i2, l2$. We can write a discrete analog of (14) as follows:

$$\begin{aligned}\Phi_{i_2,l_2}(r_{i_1}, \phi_{k_1}, z_{l_1}) &= \sum_{k=0}^{N_\phi-1} \frac{Q_{i_2,k,l_2}}{\sqrt{r_{i_2}^2 + r_{i_1}^2 - 2r_{i_2}r_{i_1}\cos(\phi_k - \phi_{k_1}) + (z_{l_2} - z_{l_1})^2}} \\ &= \sum_{k=0}^{N_\phi-1} Q_{i_2,k,l_2} K(r_{i_1}, r_{i_2}, z_{l_1}, z_{l_2}, \phi_k - \phi_{k_1}), \qquad (15)\end{aligned}$$

where Q_{i_2,k,l_2} is a screening charge located at the node (i_2, k, l_2). Then we apply a discrete convolution theorem with regard to ϕ:

$$\mathrm{FT}[\Phi_{i_2,l_2}(r_{i_1}, z_{l_1})](m) = \mathrm{FT}[Q_{i_2,l_2}](m) \cdot \mathrm{FT}[K(r_{i_1}, r_{i_2}, z_{l_1}, z_{l_2})](m).$$

Henceforth, this means that to evaluate a contribution of the ring from Γ to the potential of the ring Γ_1 it is needed to perform $O(N_\phi \log N_\phi)$ operations. The total potential in the ring $\Phi(r_{i_1}, \phi_{k_1}, z_{l_1})$, $k_1 = 0, \ldots, N_\phi - 1$, is evaluated as

$$\Phi(r_{i_1}, \phi_{k_1}, z_{l_1}) = \sum_{(i_2,l_2)\in\Gamma} \Phi_{i_2,l_2}(r_{i_1}, \phi_{k_1}, z_{l_1}),$$

and it can be done in $O(N^2 \log N)$ operations. Since the total number of rings in Γ_1 is of order $O(N)$, then we find the complexity of the algorithm to be $O(N^3 \log N)$.

Note that both problems mentioned for a classical convolution algorithm (divergence of an integral at close distances and non-periodicity of intergrands) do not exist for the algorithm presented, because Γ and Γ_1 do not contain common points, therefore the kernel's denominator is strictly greater than zero, and both K and ρ are true periodic functions.

5. The James algorithm for cylindrical coordinates

Now we can summarize the James algorithm for cylindrical coordinates.

1. Solve Poisson equation (5) in Ω with zero Dirichlet conditions imposed on the boundary Γ using difference operator (6) and the corresponding method based on Fast Fourier transform and three-diagonal elimination. It takes $O(N^3 \log N)$ operations.
2. Compute screening charges on the boundary Γ using a finite difference discretization of (9). It takes $O(N^2)$ operations.
3. Take arbitrary Ω_1 such as $\Omega_1 \supset \Omega$ and calculate the potential Φ_{Γ_1} (where Γ_1 is a boundary of Ω_1) produced by screening charges located in Γ. The method is based on the convolutions applied to the rings of nodes and direct summation of the contributions from all the rings. In practice, Ω_1 contains slightly more grid nodes than Ω. It takes $O(N^3 \log N)$ operations.
4. Solve Poisson equation (5) in Ω_1 with the Dirichlet boundary conditions on Γ_1. The solution is a desired potential. It takes $O(N^3 \log N)$ operations.

So, the total complexity of the algorithm is $O(N^3 \log N)$ and does not need any approximations of boundary conditions or assumptions about a

density distribution inside Ω, which are vitally important for the previously created algorithms [6, 8, 16].

Although the key steps of the algorithm have the same complexity it is interesting to find an appropriate ratio between them in order to understand whether there is any need for optimization. In our prototype implementation (using FFTW 3.1.5 [17, 18] for Fast Fourier transform), we have found that Steps 3 and 4 take approximately the same time, while Step 1 is about 2–4 times faster depending on the size of a mesh. The most possible reason is that the FFTW algorithm may produce significantly different performance results for good grid sizes and for the general case.

6. Conclusion

We have adapted the James algorithm for the use in the cylindrical coordinates by employing the convolution method to compute the boundary potential generated by the screening charges. The created method needs no assumptions on the density distribution in the computational domain. We also implemented a general method for solving the Dirichlet problem for the Poisson equation, which can be used for cylindrical domains with a cut off central area.

This work is the first step on the development of a scalable parallel algorithm based on domain decomposition, the James method and the method of local corrections [19, 20], but intended for cylindrical coordinates.

References

[1] Hockney R.W., Eastwood J.W. Computer Simulation Using Particles. — New York: McGraw-Hill, 1981.

[2] Pfenniger D., Friedli D. Computational issues connected with 3D N-body simulations // Astronomy and Astrophysics. — 1993. — Vol. 270, No. 1–2. — P. 561–572.

[3] Samarskii A.A. Introduction in Numerical Methods. — Moscow: Nauka, 1987 (In Russian).

[4] Snytnikov A.V. PhD thesis. — Novosibirsk, 2006 (In Russian).

[5] Kuksheva E.A., Snytnikov V.N. Parallel implementation of fundamental solution of Poisson equation // Vychislitelnye tekhnologii. — 2005. — Vol. 10, No. 4. — P. 63–71 (In Russian).

[6] Vshivkov V.A., Snytnikov V.N., Snytnikov N.V. Simulation of three-dimensional dynamics of matter in gravitational field with the use of multiprocessor computer // Vychislitelnye tekhnologii. — 2006. — Vol. 11, No. 2 (In Russian).

[7] Snytnikov N.V. PhD thesis. — Novosibirsk, 2008 (In Russian).

[8] Kuksheva E.A., Snytnikov V.N. Parallel algorithm and program Key7D for solving of nonstationary three dimensional problems of gravitational physics. // Vychislitelnye tekhnologii. — 2007. — Vol. 12, No. 1. — P. 35–44 (In Russian).

[9] James R.A. The solution of Poisson's equation for isolated source distributions // J. Comput. Phys. — 1977. — Vol. 5. — P. 71.

[10] Vshivkov V.A., Snytnikov A.V. Construction of effective parallel method for solving Poisson equation to simulate evolution of protoplanetary disk // Computational Methods and Programming. — 2009. — Vol. 10. –P. 116–122.

[11] Samarskii A.A., Andreev V.B. Difference Methods for Elliptic Equations. — Moscow: Nauka, 1976 (In Russian).

[12] Landau L.D., Lifshitz E.M., Pitaevskii L.P. Electrodynamics of Continuous Media. — Butterworth-Heinemann, 1984. — Vol. 8 (1st ed.).

[13] Kolmogorov A.N., Fomin S.V. Elements of the Theory of Functions and Functional Analysis. — Rochester, N.Y.: Graylock Press, 1961.

[14] Springel V., Yoshida N., White S.D.M. GADGET: a code for collisionless and gasdynamical cosmological simulation // New Astronomy. — 2001. — Vol. 6, Iss. 2. — P. 79–117.

[15] Eastwood J.W., Brownrigg D.R.K. Remarks on the solution of Poisson's equation for isolated systems // J. Comput. Phys. — 1979. — Vol. 32. — P. 24–38.

[16] Snytnikov N.V., Vshivkov V.A., Snytnikov V.N. Study of 3D dynamics of gravitating systems using supercomputers: methods and applications // Parallel Computing Technologies. — 2007. — (LNCS; 4671).

[17] Frigo M., Johnson S.G. The Design and Implementation of FFTW3 // Proc. IEEE. — 2005. — Vol. 93(2). — P. 216–231.

[18] Official site of FFTW software. — http://fftw.org.

[19] McCorquodale P.M., Colella P., Balls G.T., Baden S.B. A local corrections algorithm for solving Poisson's equation in three dimensions // Communications in Applied Mathematics and Computational Science. — 2007. — Vol. 2, No. 1. — P. 56–81.

[20] Balls G.T., Colella P. A finite difference domain decomposition method using local corrections for the solution of Poisson's equation // J. Comput. Phys. — 2002. — Vol. 180. — P. 25–53.