Bull. Nov. Comp. Center, Num. Anal., 16 (2013), 93–99

A scalable parallel algorithm of solving the Poisson equation for stellar dynamics problems*

N.V. Snytnikov

Abstract. We present a new parallel algorithm for solving the Poisson equation in the context of non-stationary stellar dynamics problems, e.g. rotating galaxies or circumstellar disks. This allows us to conduct numerical experiments on a mesh with 10–100 billion of nodes and to use more than 10 thousand of processors. This algorithm is based on a finite difference method, 3D domain decomposition, partial pre-computation of Green's function for subdomains boundaries and parallel implementation of the James method. All the computations have been conducted in the Siberian Supercomputer Center (Novosibirsk) and the Joint Supercomputer Center (Moscow).

1. Introduction

The particle-in-cell (PIC) method is among those that are used [1, 2] for solving the stellar dynamics equations, consisting of the collisionless Boltzmann equation (also known as the Vlasov equation)

$$\frac{\partial f}{\partial t} + \boldsymbol{u}\frac{\partial f}{\partial \boldsymbol{r}} - \nabla\Phi(t,\boldsymbol{r})\frac{\partial f}{\partial \boldsymbol{u}} = 0, \quad f(0,\boldsymbol{r},\boldsymbol{u}) = f^0(\boldsymbol{r},\boldsymbol{u})$$

and the Poisson equation for isolated systems with a special Dirichlet condition

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

Here $f = f(t,\boldsymbol{r},\boldsymbol{u})$ is the distribution function of a matter (the stars in galaxies or the dust in circumstellar disks), $\Phi = \Phi(t,\boldsymbol{r})$ is the gravitational potential, and $\rho = \rho(t,\boldsymbol{r})$ is the density, which can be calculated using the following equation:

$$\rho(t,\boldsymbol{r}) = \int f(t,\boldsymbol{r},\boldsymbol{u})\,d\boldsymbol{u}.$$

The PIC method requires the computation of individual movements of a large number of particles that represent the distribution function f, and of the self-consistent gravitational field using a finite-difference method on a regular mesh.

The present paper is focused on the latter problem — solving the Poisson equation, which in this context has the following difficulties to address:

*Supported by RFBR under Grants 12-07-00065, 14-01-31088 and Integration Grant of SB RAS No. 26.

- Non-stationary processes in galaxies or circumstellar disks require calculation of a gravitational potential for each time step. And the total number of steps may exceed hundreds of thousands.
- It is needed to use a fine mesh (of order of 1000^3 nodes) for simulating physical processes, like planet formation, in a correct way.
- It is necessary to conduct tens or hundreds of numerical experiments with different input parameters.

The only way to meet all these requirements is to develop a scalable parallel algorithm aimed at running on a supercomputer with more than 10^4 processors.

There are several methods to deal with the Dirichlet problem for isolated systems. Among them are *potential approximation* on the domain boundaries [3], *convolution method* [4], and the most efficient sequential algorithm *James method* [5], which was also recently adopted for the use in cylindrical coordinates [6].

Parallel algorithms for solving the Poisson equation for isolated systems can be divided into the following groups:

1. potential approximation on a finite domain boundary and parallel solution of finite difference methods [7];
2. parallel convolution method using a parallel Fourier transform [8, 9];
3. method of local corrections [10].

This paper describes a new parallel numerical algorithm which employs a method of partial pre-computation of Green's function (which is similar to an idea proposed in [11] for the parallel solution of three-diagonal linear algebraic equations), calculation of a single-layer potential in a spirit of [12] and the use a modification of the James method [6].

In Section 2, we describe the Green's function partial pre-computation algorithm, 2D domain decomposition methods being given in Section 3. The general algorithm for 3D domain decomposition along with testing results are described in Section 4. Section 5 outlines a parallel modification of the James method.

2. Partial pre-computation of Green's function

Let us consider the Dirichlet problem for the 2D Poisson equation

$$\Delta\Phi = \rho, \quad \Phi|_\Gamma = 0,$$

which is solved on $L_x \times L_y$ mesh with the spatial steps h_x and h_y using a 5-point approximation stencil (or any other compact stencil). We suppose the density ρ to be equal to zero everywhere, except for a layer with the

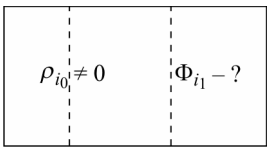

Figure 1. The problem of gravitational potential calculation on a layer with index i_1, which is generated by charges (masses) $\rho_{i_0 k}$ located in the layer i_0

index i_0: $\rho_{i_0 k} \neq 0$. It is required to calculate a gravitational potential $\Phi_{i_1 k}$ on a layer with the index i_1 (Figure 1).

Applying the Fourier transform for the sine functions and taking into account the fact that $\rho_i = 0$, if $i \neq i_0$, we write down

$$\Phi_{i_1 k} = \sum_{n=1}^{L_y - 1} A_{i_0}(n) \cdot B_{i_1, i_0}(n) \cdot \sin \frac{\pi n k}{L_y},$$

where

$$B_{i_1, i_0}(n) = \sum_{m=1}^{L_x - 1} \left[-\frac{4 \sin^2 \frac{\pi m}{2 L_x}}{h_x^2} - \frac{4 \sin^2 \frac{\pi n}{2 L_y}}{h_y^2} \right]^{-1} \sin \frac{\pi m i_0}{L_x} \sin \frac{\pi m i_1}{L_x},$$

$$A_{i_0}(n) = \sum_{k=1}^{L_y - 1} \rho_{i_0 k} \sin \frac{\pi n k}{L_y}.$$

As the auxiliary function $B_{i_1, i_0}(n)$ does not depend on density and can be calculated one time for a given mesh and a stencil, then it appears that the computational complexity of calculating $\Phi_{i_1 k}$ is of order $O(L_y \log_2 L_y)$. Note, that if we solve the same problem with the well-known fast Fourier transform, then the computational complexity will be equal to $O(L_y L_x \log_2 L_y \log_2 L_x)$.

Thus, if we compute the function $B_{i_1, i_0}(n)$ for all necessary values of i_1, i_0 before solving a non-stationary problem, it will drastically reduce the amount of computations to be carried out.

3. The 2D domain decomposition

The method of domain decomposition is based on the idea of a single layer calculation, which is used in a number of algorithms [5, 12].

Let us consider the Dirichlet problem for the 2D Poisson equation in the 2D domain Ω, and assume that we subdivide Ω into 2 subdomains Ω_1, Ω_2 with the boundaries Γ_1, Γ_2 and the common boundary (*interface boundary*) γ (Figure 2).

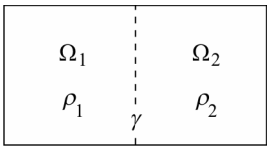

Figure 2. Subdivision of 2D domain into 2 subdomains

The decomposition algorithm consists of the following steps:

1. Assign $\Phi_i|_\gamma = 0$, and then in each of the subdomains solve a subproblem
$$\Delta\Phi_i = 4\pi\rho_i, \quad \Phi_i|_{\Gamma_i} = 0, \quad i = 1, 2.$$
2. Calculate Φ^0 in Ω as a union of Φ_1 in Ω_1 and Φ_2 in Ω_2. Then Φ^0 is a continuous function on the interface γ, but its normal derivative $\frac{\partial\Phi^0}{\partial\vec{n}}$ has a discontinuity on γ.
3. This discontinuity of the normal derivative generates a layer of "screening charges" q, located on γ, which can be calculated by applying a finite difference Laplace operator to the function Φ^0: $q = \Delta\Phi^0(\gamma)$.
4. Screening the charges q generate the gravitational potential Φ^{scr} in the domain Ω.
5. It is sufficient to calculate Φ^{scr} at the nodes of the boundary γ using a partially pre-computed Green's function.
6. For each subdomain we solve the Dirichlet problem for the Laplace equation $\Delta\Phi_i = 0$, $\Phi_i|_{\Gamma_i\setminus\gamma} = 0$, $\Phi_i|_\gamma = \Phi^{\text{scr}}_\gamma$.
7. A required solution to the initial problem is the sum of homogeneous and inhomogeneous Dirichlet problems which were calculated at the first and the last steps.

Now let us proceed to the algorithm for an arbitrary number of subdomains P. The main difficulty is that we have $P - 1$ interface boundaries, and each of them has its own layer of screening charges. All those charges generate their "own" gravitational potential.

We have developed special algorithm, which uses binary tree of subdomains, where a root of a tree is the domain Ω, child nodes being 2 "half-domains" of a parent node, and leaf nodes are P subdomains. Starting from the leaf nodes we apply the algorithm proposed for 2 subdomains on each tree level until we reach the tree root.

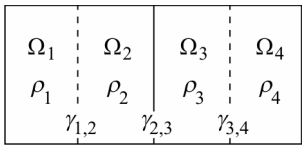

Figure 3. 2D domain subdivision into 4 subdomains

Thus, the boundary conditions for each of the subdomains can be calculated for $\log P$ steps.

Here we give an example, how the algorithm works for $P = 4$ (Figure 3):

1. Solve the Poisson equation with homogeneous boundary conditions. Calculate screening charges $q_{1,2}$ on $\gamma_{1,2}$ and $q_{3,4}$ on $\gamma_{3,4}$.
2. Calculate the gravitational potential, which is generated by $q_{1,2}$ and $q_{3,4}$ on $\gamma_{2,3}$ and is adjacent to $\gamma_{2,3}$ nodes. Calculate screening charges $q_{2,3}$ on $\gamma_{2,3}$.
3. Calculate the gravitational potential, which is generated by $q_{2,3}$ on $\gamma_{2,3}$, $\gamma_{1,2}$, $\gamma_{3,4}$.

4. This gravitational potential defines the boundary conditions of the Dirichlet problem for the Laplace equation.

4. The 3D domain decomposition

Let us describe a 3D domain decomposition method. First, it is worth to note that an "elongated" 3D parallelepiped domain is quite similar to the 2D case. If the required number of processors P is much less than the number of nodes L_x, L_y, or L_z (e.g., $P \leq 128$ and $L_x = 2048$), then a 3D decomposition method is a straightforward generalization of a 2D decomposition. After applying the fast Fourier transform to the direction z and dealing with L_z independent systems of linear algebraic equations with regard to 2D Fourier harmonics of the gravitational potential, we apply a 2D domain decomposition algorithm to each of them.

However, in the case of a "cubic" 3D domain, this algorithm is not suitable, because for a mesh with 1024^3 nodes it is needed to have not less than 1024 processors in order to accomplish one time step in several seconds.

That is why a 3D decomposition must be applied in 2 directions in the following way:

1. Suppose that $P = P_x \times P_y$ and apply a 2D decomposition twice: first subdivide the domain into P_x layers and then subdivide each layer into P_y columns.
2. For each layer solve in parallel the homogeneous Dirichlet problem for the Poisson equation.
3. Apply a fast Fourier transform in the direction z and solve L_z independent systems of linear equations with regard to 2D Fourier harmonics of the gravitational potential.
4. Re-distribute the interface boundaries data: it is required to apply the matrix transposition of $L_y \times L_z$ and "all-to-all" communications in processor groups, each of them comprising of P_y processors.
5. Compute screening charges on the interfaces of x layers and their impact on "parallel" layers.
6. Apply the inverse matrix transposition and fast Fourier transform.
7. Solve the inhomogeneous Dirichlet problem for the Laplace equation and obtain a required solution.

The computational complexity of the algorithm for a cubic 3D domain is

$$O\Big(\frac{N^3 \log N}{P_x \cdot P_y}\Big) + T(P_y, 6N^2) + SR((\log P_x + \log P_y) \cdot 4N^2 \log N),$$

where $T(P_y, 6N^2)$ denotes the transposition of six 2D matrices of $N \times N$ dimension inside the processor group of P_y entries, and SR denotes operations of send/receive data for 2 processors. This estimation shows that interprocessor communications do not require transferring a 3D data array among all the processors.

Profiling, testing and proof-of-concept experiments were carried out on supercomputers of the Siberian Supercomputer Center (Novosibirsk) and the Joint Supercomputer Center (Moscow).

The performance measurements for the algorithm implementation are presented in the table.

Parallel algorithm performance.
Computations have been conducted in the JSCC

Number of proc., $P = P_x \times P_y$	Solving time for $1024 \times 1024 \times 1024$ mesh, seconds			
	Dirichlet	SendRecv	AllToAll	Total
$512 = 32 \times 16$	1.6	0.38	0.33	2.3
$1024 = 64 \times 16$	0.6	0.44	0.33	1.5
$2048 = 64 \times 32$	0.35	0.6	1.2	2.15

5. Parallel implementation of the James method

Let us outline the parallel James method for isolated systems, which is an ongoing focus of the present research:

1. Decompose the domain boundary Γ to "patches" and re-distribute them among processors. Pre-compute all the necessary values of Green's function for each of the patches.
2. Solve a homogeneous Dirichlet problem for the Poisson equation with the algorithm described. Compute screening charges on the domain boundary.
3. Compute the gravitational potential, which is generated by screening charges located on the opposite patches, using a modification of the convolution method [6].
4. Solve the Laplace equation with new boundary conditions. Compute the gravitational potential.

6. Conclusion

We have developed a parallel algorithm for solving the 3D Poisson equation for isolated systems, aimed at the simulation of non-stationary stellar dynamics problems. This algorithm is based on domain decomposition in 2 directions, the subdomain coupling with a screening charges technique,

and a partial pre-computation of Green's function. The proof-of-concept tests have shown that the Poisson equation on a mesh with 1024^3 nodes can be solved in less than 2 seconds using 1024 processors.

References

[1] Berezin Yu.A., Vshivkov V.A. Particle-in-Cell Method in the Plasma Dynamics. — Nauka, 1980.

[2] Snytnikov N. Scalable parallel algorithm for solving the collisionless Boltzmann–Poisson system of equations // Astronomical Society of the Pacific Conference Series. — 2012. — Vol. 453. — P. 393.

[3] Hockney R.W. Gravitational experiments with a cylindrical galaxy // Astrophysical J. — 1967. — Vol. 150. — P. 797.

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

[5] James R.A. The solution of Poisson's equation for isolated source distributions // J. Comput. Phys. — 1977. — Vol. 25. — P. 71.

[6] Snytnikov N.V. Computation of gravitational potential of isolated systems in cylindrical coordinates // Bull. Novosibirsk Computing Center. Ser. Num. Anal. — Novosibirsk, 2011. — Iss. 15. — P. 61–73.

[7] Snytnikov N., Vshivkov V., Snytnikov V. Study of 3D dynamics of gravitating systems using supercomputers: methods and applications // Parallel Computing Technologies. — 2007. — P. 162–173. — (LNCS; 4671).

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

[9] Habib S. et al. The Universe at Extreme Scale: Multi-Petaflop Sky Simulation on the BG/Q. — arXiv:1211.4864

[10] 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.

[11] Terekhov A.V. Parallel dichotomy algorithm for solving tridiagonal system of linear equations with multiple right-hand sides // Parallel Computing. — 2010. — Vol. 36. — P. 423–438.

[12] Huang J., Greengard L. A fast direct solver for elliptic partial differential equations on adaptively refined meshes // SIAM J. Sci. Comput. — 2000. — Vol. 21. — P. 1551–1566.