Astrophysics with the Computer: Finding the Gravitational Potential of a Galaxy

Joachim Köppen Kiel/Strasbourg 2006/07

1 Astrophysics

The movement of stars within a galaxy is governed by the gravitational potential in the galaxy which is produced by all the visible and nonvisible matter in the galaxy. So the stars of the disk, of the bulge, and of the halo as well as the gas in the disk influence the kinematics of any star, say in the disk.

If we watch a star far away from the galactic centre, the gravitational attraction can well be approximated by assuming that all the mass of the galaxy is concentrated in the galactic centre. Hence the potential will be that of a point mass, like that of the sun; the star will move like a planet around the sun, in an orbit that has the shape of a conic section. The escape velocity from a given radius will be a certain multiple of the circular velocity. But what happens, if we are nearer to the galactic disk or even within it? We need to take into account the mass distribution.

In this **experimental project** we shall compute the gravitational potential of a given distribution of mass in a galaxy, by solving Poisson's equation, which is a partial differential equation in three dimensions. However, since a 3D treatment would take up far too much of computing time, we shall do this very crudely in 2D to compute the potential in the equatorial plane. Although we shall not get the correct solution, we have the chance to try out some nice numerical techniques.

2 The Equation

Poisson's equation relates the gravitational potential Φ to the density ρ of matter

$$\nabla\Phi = 4\pi G\rho \tag{1}$$

where G is the gravitational constant. This is a second order partial differential equation, which we can discretize on a cartesian grid with constant step widths Δx, Δy, and Δz. Thus, the coordinate x is represented by the discrete values x_i. The second derivative of a function $f(x)$ would be

$$\frac{d^2 f(x_i)}{dx^2} = \frac{f'(x_{i+1/2}) - f'(x_{i-1/2})}{\Delta x} \tag{2}$$

and the first order derivative is

$$f'_{i+1/2} = \frac{df(x_{i+1/2})}{dx} = \frac{f(x_{i+1}) - f(x_i)}{\Delta x} \tag{3}$$

which combines into

$$\frac{d^2f(x_i)}{dx^2} = \frac{f(x_{i+1}) - 2f(x_i - f(x_{i-1})}{(\Delta x)^2} \tag{4}$$

With this expression we can write the discretized form of Poisson's equation as, using the abbreviation $\Phi_{i,j,k} = \Phi(x_i, y_j, z_k)$ and normalizing to $G = 1$:

$$\begin{aligned}
4\pi\rho_{i,j,k} &= \frac{\Phi_{i+1,j,k} - 2\Phi_{i,j,k} + \Phi_{i-1,j,k}}{(\Delta x)^2} && (5)\\
&+\frac{\Phi_{i,j+1,k} - 2\Phi_{i,j,k} + \Phi_{i,j-1,k}}{(\Delta y)^2} && (6)\\
&+\frac{\Phi_{i,j,k+1} - 2\Phi_{i,j,k} + \Phi_{i,j,k-1}}{(\Delta z)^2} && (7)
\end{aligned}$$

Here we have a system of linear equations for the unknown values of the potential in all points of the spatial grid. Together with the normalization that the potential shall be zero at infinity, it should give us the solution ...

However, if we took a spatial grid of 100 points in each direction, there would be 1000000 points, and to solve numerically a system of linear equations of so many variable and equations is not possible with straight-forward methods. In methods such as Gaussian elimination, the rounding errors accumulate in the large number of operations, so that even in double precision the results of more than 100 equations can be completely useless!

Therefore, we shall treat the problem in two dimensions only: We shall pretend that we can discard the derivative in z direction, and that the volume density $\rho = \Sigma/H$ can be substituted by the column density Σ:

$$(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2})\Phi = 2\pi G\Sigma/H \tag{8}$$

with

$$\Sigma = \int_{-\infty}^{\infty} \rho dz \tag{9}$$

and a typical thickness $2H$ of this disk. Please note that this is **not correct**, but simply assumed to permit us to get some results which demonstrate the problem and the numerical technique! With this thought in mind, we get this substantially smaller system:

$$\begin{aligned}
2\pi\Sigma/H &= \frac{\Phi_{i+1,j} - 2\Phi_{i,j} + \Phi_{i-1,j}}{(\Delta x)^2} && (10)\\
&+\frac{\Phi_{i,j+1} - 2\Phi_{i,j} + \Phi_{i,j-1}}{(\Delta y)^2} && (11)
\end{aligned}$$

But see section 4.1 how to perform strict tests with the program.

2.1 Boundary conditions

An important issue is to impose the normalization that the potential shall vanish at infinity: We have to specify the conditions at the outer boundary of the computational region.

Let us assume that we use a cartesian grid centered in the centre of the galaxy. Since we know that at large distances, the gravitational potential will approach the value computed, as if all the mass were concentrated in a point at the galactic centre, we can thus estimate the potential at large distances from the galaxy. Thus we do not need to compute up to infinity, but only up to a sufficiently large distance, where we shall know the potential. How far is that? It depends on the accuracy which we want to have for the potential. We must try out with different values.

So let us take the grid to go from $x = -L$ to $+L$, and similar in y direction. At the outer rim we shall fix the potential:

$$\Phi(x, y) = -\frac{GM_{\mathrm{gal}}}{x^2 + y^2} \tag{12}$$

for all those points (with $i = 1$ and $i = nx$ and $k = 1$ and $k = ny$, if you use the FORTRAN index notation; also I use k for the other index, because j can too easily be confused by i ...).

2.2 Numbering of points

On all the other points of the grid, the potential values need to be determined. It is practical – and also essential for the iterative method – to number these unknown points: a simple and straight-forward scheme is like this:

i =	1	2	3	4	5
k=1	–	–	–	–	–
k=2	–	1	2	3	–
k=3	–	4	5	6	–
k=4	–	7	8	9	–
k=5	–	–	–	–	–

The boundary points are marked with ”–”, their values are known.

3 Numerical Methods

3.1 Direct Inversion

Although the direct inversion, by Gaussian elimination, is not a useful or efficient method, you might try it, but only if you really want to: This teaches you the problems when large matrices need to be inverted. You find such a routine in the ’Numerical Recipes’ or maybe on the Web ... but always check that the routine works exactly as it should ... and check that you don't confuse row with columns.

Then we simply fill the matrix with the coefficients from the left hand side (to make writing easier, we set $\Delta = \Delta x = \Delta y$):

$$-4\Phi_{i,k} + \Phi_{i-1,k} + \Phi_{i+1,k} + \Phi_{i,k-1} + \Phi_{i,k+1} = 2\pi\Sigma_{i,k}/H\,\Delta^2 \tag{13}$$

and fill the corresponding vector with the terms of the right-hand side. But please note that here we go over the points (i,k) of the inner computational region, and that for the points near the boundary also involve the boundary values. For instance, for $i = 2$ and $1 < k < ny$:

$$-4\Phi_{2,k} + \Phi_{3,k} + \Phi_{2,k-1} + \Phi_{2,k+1} = 2\pi\Sigma_{2,k}/H\,\Delta^2 - \Phi_{1,k} \tag{14}$$

since $\Phi_{1,k}$ is the **known** value at the boundary. It is completely simple, but be very careful ... it is a good idea to write down the equations by hand or to make yourself a drawing of the grid points.

For a few configurations analytical solution of the potential and the corresponding density distribution are known. Use them to compare with the results. When you use the column densities from the Toomre-Kuzmin disk potential, you will find that the result will differ from the theoretical potential, but that one gets a similar curve. The results depend unfortunately strongly on the number of grid points, and more than about 20 by 20 will probably give nonesense results, because of the rounding errors.

Suggestion: what is the largest number of grid points which permits to obtain a solution? How does the computing time go up with increasing number of points?

3.2 The properties of a potential

The gravitational potential (as well as the electrostatic one) have a nice property: the value at every point in space is the average of its values in the neighbourhood. For our 2D cartesian grid this means:

$$\Phi_{i,k} = \left(\Phi_{i-1,k} + \Phi_{i,k-1} + \Phi_{i+1,k} + \Phi_{i,k+1}\right)/4 \tag{15}$$

It is worth verifying this with one of the potentials of the test cases below! This characteristic is the basis of the powerful and efficient relaxation methods we shall discuss now:

3.3 Jacobi's Method

The direct inversion is limited to too few points that one cannot get a solution with a useful high spatial resolution. A more attractive method is an iterative solution of solving the system of equations. We apply this formula:

$$\begin{aligned} \Phi^{n+1}_{i,k} &= -2\pi\Sigma_{i,k}/H\,\Delta^2 && (16)\\ &\quad + \left(\Phi^n_{i-1,k} + \Phi^n_{i,k-1} + \Phi^n_{i+1,k} + \Phi^n_{i,k+1}\right)/4 && (17) \end{aligned}$$

Here the index n counts the number of iterations. We see that the potential is corrected by the average over the neighbouring four points. Again, we apply this formula only to the points of the inner computational region; the points on the boundary are set to the known values. This schema is also pretty simple, but be very careful with the indices ... otherwise you spend a lot of time of tracing the errors!

The iteration needs a start solution: we can simply use $\Phi = 0$ everywhere, but also anything better.

In 'Numerical Recipes' this method is said to be slow and not practical. Try it out, compare it with the others, and tell us your opinion!

3.4 Gauss-Seidel method

If one uses values already computed as one goes through the grid of points, one can devise a faster method:

$$\begin{aligned} \Phi^{n+1}_{i,k} &= -2\pi\Sigma_{i,k}/H\,\Delta^2 && (18)\\ &+ \left(\Phi^{n+1}_{i-1,k} + \Phi^{n+1}_{i,k-1} + \Phi^{n}_{i+1,k} + \Phi^{n}_{i,k+1}\right)/4 && (19) \end{aligned}$$

One notes that on the right hand side, one uses already values from the new iteration. This is easily done by going through the grid in the same way we had described the numbering of equations. What do you think of this?

3.5 Simultaneous Over-Relaxation Method

Finally, one can cook up this formula:

$$\begin{aligned} \Phi^{n+1}_{i,k} &= \Phi^{n}_{i,k} + \frac{A}{4}\left(-4\Phi^{n}_{i,k} - 2\pi\Sigma_{i,k}/H\,\Delta^2\right. && (20)\\ &\left. + \Phi^{n+1}_{i-1,k} + \Phi^{n+1}_{i,k-1} + \Phi^{n}_{i+1,k} + \Phi^{n}_{i,k+1}\right) && (21) \end{aligned}$$

which contains a factor A to control the convergence. The smaller the value is, the slower is the convergence. Values from 0 to less than 2 can be used. In our problem it is not critical.

You will see that the iteration is well behaved, and converges towards a curve similar to the theoretical potential, but again, it will differ quite a bit. This is our crude neglect of the third dimension! Convince yourself that the deviation is not due to the number of grid points, the relaxation parameter A, or the initial trial solution!

Suggestion: you can try various starting solutions. How well does the iteration cope with that? How does the number of grid points influence the results? How does this method compare in computing time with the other methods?

4 Test cases

The simplest system is a point mass M at the centre. The potential is simply

$$\Phi(r) = -\frac{GM}{r} \qquad (22)$$

The escape velocity follows directly. From this one obtains the centripetal acceleration by taking the radial derivative. The circular velocity follows from the equality of centripetal and centrifugal forces.

A description often used for spherical self-gravitating systems of stars is the Plummer sphere. It has the potential

$$\Phi(r) = -\frac{GM}{\sqrt{r^2 + a^2}} \tag{23}$$

and it is produced by the gravitation of stars distributed in space with the density

$$\rho(r) = \frac{3M}{4\pi a^3}\left(1 + (\frac{r}{a})^2\right)^{-5/2} \tag{24}$$

where r is the distance to the centre. The constant a is the characteristic radius of the density distribution. This case is a good test case for a 1-D integration over the radial coordinate r.

For axisymmetric systems (elliptical galaxy, disk of a spiral galaxy) one can generalize this model and obtain the Toomre-Kuzmin disk model:

$$\Phi(\omega, z) = -\frac{GM}{\sqrt{\omega^2 + (a + |z|)^2}} \tag{25}$$

where ω shall denote the radial distance to the axis of symmetry. This potential is produced by stars which are distributed in the galactic plane, having a *surface density*

$$\Sigma(\omega) = \int \rho(\omega, z)dz = \frac{M}{2\pi a^2}\left(1 + (\frac{r}{a})^2\right)^{-3/2} \tag{26}$$

Your program should be able to compute the potential, if you give the surface density profile!

A further generalization gives the Miyamoto-Nagai potential

$$\Phi(\omega, z) = -\frac{GM}{\sqrt{\omega^2 + (a + \sqrt{b^2 + z^2})^2}} \tag{27}$$

which is the result of the mass distribution

$$\rho(\omega, z) = \frac{Mb^2}{4\pi a^2}\frac{a\omega^2 + (a + 3\sqrt{z^2 + b^2})(a + \sqrt{z^2 + b^2})}{[\omega^2 + (a + 3\sqrt{z^2 + b^2})^2]^{5/2}(z^2 + b^2)^{3/2}} \tag{28}$$

You can use this to test your 3-D version of integration over the given mass distribution.

4.1 Correct Solutions?

As stated earlier, our brutal reduction of the 3D problem to two dimensions does not permit to obtain the correct solutions. But how can we test the program in a strict way?

We generate our own analytical density-potential pairs!

The program solves Poisson's equation

$$\nabla\Phi = 4\pi G\rho \tag{29}$$

but in 2D. If we consider axisymmetric problems only, we can write

$$\nabla\Phi = \frac{1}{r}\frac{d}{dr}\left(\frac{d\Phi}{dr}\right) = 4\pi G\rho = 2\pi G\Sigma/H \tag{30}$$

If we now take a function similar in shape to a potential, we can directly obtain the corresponding surface density distribution $\Sigma(r)/H$. Thus we can use that density law to see whether the program finds the assumed potential ...

5 Challenge: 3D

When your program works, and if it is fast enough, maybe you could try to do it in three dimensions. Instead of solving the wrong equation and thus geting results that cannot agree with the theoretical formulae, you could verify the Plummer sphere and the Toomre-Kuzmin disk!

Again, we shall assume that the spatial stepwidths are the same in all three directions ($\Delta = \Delta x = \Delta y = \Delta z$). While for the test case of the Plummer sphere the computational region should be a cube with sidelength of about twice the characteristic radius a, we can treat a disk galaxy in a smaller volume: the height must only be twice the scale height H, hence we need less points in the vertical grid! This trick can make a 3D calculation possible!

It is straightforward to generalize the relaxation formulae to three dimensions.

Note that in 3D we can use the **volume** densities; for the disk galaxy, we can simply modify the Plummer sphere law:

$$\rho(r) = \frac{3M}{4\pi a^3}\left(1 + \frac{x^2+y^2}{a^2} + \frac{z^2}{H^2}\right)^{-5/2} \tag{31}$$

and choosing $H = 0.1a$ which gives a fairly flat disk. The Plummer or Kuzmin potential will still be correct in the exterior parts of the galaxy, but near the centre ($r < H$) we must expect deviations.

Note that for numerical reasons the vertical grid should have a sufficient number of points! But do not be too surprised, if your program finds the solution even with 3 points in z only ... you are cheating, because we must specify the potential on the outer boundary, and if you give the correct formula there, the program will turn out the correct formula!

6 Literature

The books "Galactic Dynamics" by Binney and Tremaine, "Galactic Astronomy" by Binney and Mihalas are available in the MS2 library.

The book "Numerical Recipes" is available in the MS2 library. It also explains Successive Overrelaxation method and the other methods in our problem.

A test version of a Java applet for the Kuzmin disk - but still without any explanations – is available at
`http://astro.u-strasbg.fr/~koppen/galpot2/`