Mon. Not. R. Astron. Soc. **000**, 1–9 (2011) Printed 13 December 2011 (MN LATEX style file v2.2)

Smoothed Particle Hydrodynamics: Theory, Implementation, and Application to Toy Stars

Philip Mocz

Applied Math 205 Final Project, Harvard University, Fall 2011, Prof. Knezevic

ABSTRACT

In this final project, we discuss the theory of Smoothed Particle Hydrodynamics (SPH) and its implementation in MATLAB. SPH is a particle-based method for computational fluid dynamics. We apply SPH to study the time evolution of a toy star model and find its equilibrium state. The goal of this paper is to demonstrate the SPH works successfully, therefore we apply it to problems where analytic solutions are known to exist and compare numerical and theoretical answers. We also consider a toy collision problem of two polytropic bodies. SPH is found to give reliable answers and is a useful method for computational fluid dynamics.

Key words: computational fluid dynamics – numerical analysis – smoothed particle hydrodynamics – stars: general

1 INTRODUCTION

Smoothed particle hydrodynamics (SPH) is a particle-based method for computational fluid dynamics. It was originally invented to simulate polytropic stellar models under non-axisymmetric conditions (the original papers are Gingold & Monaghan (1977) and Lucy (1977)). The SPH method can easily be extended to deal with a large variety of complex physical models. SPH works independently of any grid, unlike finite-difference methods, and interactions between volume elements such as the pressure gradient are represented as a force between particles. The method is purely Lagrangian, meaning that interactions and derivatives are all evaluated in a coordinate system attached to a moving fluid element. The two main ideas of SPH are (1) to evolve the positions and velocities of particles according to the calculation of the forces on each particle at each time-step, and (2) to use an interpolating/smoothing kernel to calculate forces and spatial derivatives.

Since its inception, SPH has found a number of applications in astrophysics, engineering, and computer graphics (see reviews Monaghan (1992, 2005); Rosswog (2009)). In astrophysics, in particular, SPH is very useful to model fluids where boundary conditions are absent. It also has the advantage of simultaneous conservation of linear and angular momentum, regardless of the geometry of the physical system (Price 2011). In addition, particles may be divided into types of particles which interact differently with different types, representing a mixture of various fluids. In the original paper, Gingold & Monaghan (1977) developed and used SPH to simulate the equilibrium structure of uniformly rotating polytropic stars with magnetic fields and compared it to analytic calculations to show that the method is robust. Lucy (1977) (who developed SPH independently) used SPH to study a rotating polytropic fluid undergoing fission to produce a binary system. SPH since has, among other applications, been used to study criteria for fragmentation in collapsing molecular clouds (Miyama, Hayashi & Narita 1984), collision problems (e.g. impact with the Moon (Benz, Slattery & Cameron 1986)), and the structure of dark matter via large scale parallelized cosmological simulations combined with N-body (Springel 2005). Other notable applications of SPH include its usage to simulate Gollum falling into the lava in 'The Lord of the Rings: The Return of the King' (Monaghan 2005).

In the present paper, we discuss the theory of SPH, its implementation in MATLAB, and applications to toy models of polytropic stars. The mathematical theory is outlined in § 2. In § 3 we discuss the physics of the simple toy star, which is based on the paper Monaghan & Price (2004) (see also the later work Monaghan & Price (2006)). Our choices for initial conditions and parameters are influenced by the work by Braune & Lewiner (2009). We discuss the MATLAB code we wrote in § 4. In § 5 we describe the initial conditions and parameters of our simulations. In § 6 we present results of our simulations of polytropic stars in 2 and 3 dimensions and compare the equilibrium states to the analytic solutions. We test two different smoothing kernels and the robustness of SPH under variations in some of the parameters we are free to choose (e.g. number of particles, kernel smoothing length). In addition, we also apply SPH to study the collision of two toy stars. In § 7 we discuss the results and ways in which we could expand the work, and also compare SPH to other numerical methods for fluid dynamics. Our conclusions are stated in § 8.

2 THEORY OF THE SPH METHOD

In SPH, a fluid is modeled as a collection of fluid elements (particles) each of which have associated to them relevant physical quantities (position, velocity, mass, density, etc.). The Euler equation for an ideal fluid without dissipation tells us:

$$\rho \frac{d\mathbf{v}}{dt} = -\nabla P + \mathbf{f} \tag{1}$$

where $\mathbf{v}$ is the velocity, ρ is density, P is pressure, and $\mathbf{f}$ can be some external force per unit volume applied to the fluid (e.g. gravity). In general P will be some function of ρ and thermal energy. In this work we will assume the fluid is **polytropic**, meaning we have the simple relationship between pressure and density:

$$P = k\rho^{1+1/n} \tag{2}$$

where k some constant and n is the polytropic index.

The main goal of the SPH method will be to evaluate the acceleration $\frac{d\mathbf{v}}{dt}$ (the equation of motion) of each particle (which will involve looking at pairwise interactions between particles) and update the velocities and positions of the particles with each time step using an integrator scheme (more on this later). The largest challenge is developing a method to evaluate spatial derivatives. This challenge is overcome by first considering the identity:

$$A(\mathbf{r}) = \int A(\mathbf{r}')\delta(\mathbf{r}-\mathbf{r}')\, d\mathbf{r}' \tag{3}$$

where $A(\mathbf{r})$ is some arbitrary function that maps into $\mathbb{R}$ and $\delta(\mathbf{r})$ is the Dirac Delta function. In the SPH scheme, δ is replaced with an approximation: a smoothing kernel $W(\mathbf{r};\ h)$, where h is the smoothing length scale. The smoothing kernel must have the properties

$$\int W(\mathbf{r})\, d\mathbf{r} = 1 \tag{4}$$

and

$$\lim_{h\to 0} W(\mathbf{r};\ h) \to \delta(\mathbf{r}). \tag{5}$$

The kernel should also be non-negative and invariant under parity.

Two common choices of the smoothing kernel are the Gaussian kernel:

$$W(\mathbf{r};\ h) = \left(\frac{1}{h\sqrt{\pi}}\right)^d \exp(-\|\mathbf{r}\|^2/h^2) \tag{6}$$

and the cubic spline approximation:

$$W(\mathbf{r};\ h) = C_h \begin{cases} (2-q)^3 - 4(1-q)^3 & 0 \le q < 1 \\ (2-q)^3 & 1 \le q < 2 \\ 0 & q \ge 2 \end{cases} \tag{7}$$

where d is the dimension of the problem ($d = 1$, 2 or 3), $q = \|\mathbf{r}\|/h$, and the normalization constant $C_h = \frac{1}{6h}$, $\frac{5}{14\pi h^2}$, $\frac{1}{4\pi h^3}$ for one, two, three dimensions respectively. The Gaussian is the best choice for reproducing physical answers (Gingold & Monaghan 1977) but the spline kernel has the advantage of being non-zero only on a compact support, which reduces computation time since we will see that SPH involves calculating pairwise particle interactions using the smoothing kernel and using a kernel with compact support

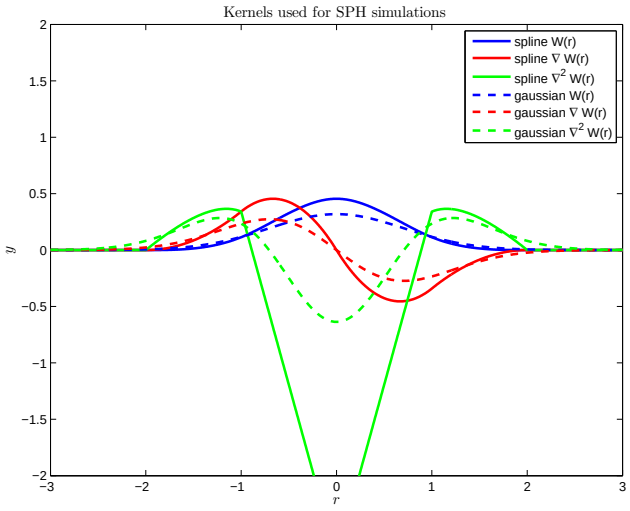

Figure 1. Plot of the Gaussian and spline smoothing kernels, the magnitude of their gradients, and their Laplacians ($h = 1$, $d = 2$).

means the interaction is simply 0 if two particles are sufficiently far apart. The two kernels, the magnitude of their gradients, and their Laplacians are plotted in Figure 1.

Having replaced the Delta function in equation (3) to estimate $A(\mathbf{r})$, now have an estimate $A_1(\mathbf{r})$ of $A(\mathbf{r})$:

$$A(\mathbf{r}) \simeq A_1(\mathbf{r}) := \int A(\mathbf{r}')W(\mathbf{r}-\mathbf{r}';\ h)\, d\mathbf{r}' \tag{8}$$

We then apply a second approximation, namely discretization (i.e., we can only work with a finite number of particles $j = 1, ..., N$). Our new approximation $A_2(\mathbf{r})$ of $A(\mathbf{r})$ becomes:

$$A(\mathbf{r}) \simeq A_2(\mathbf{r}) := \sum_j A(\mathbf{x}_j)W(\mathbf{r}-\mathbf{r}_j;\ h)\Delta V_j \tag{9}$$

where $\Delta V_j m_j/\rho_j$, $\mathbf{r}_j$ is the position of particle j, m_j is its mass, and ρ_j is its density. This is quite useful because now we can approximate derivatives:

$$\nabla A_2(\mathbf{r}) = \sum_j A(\mathbf{x}_j)\nabla W(\mathbf{r}-\mathbf{r}_j;\ h)\Delta V_j \tag{10}$$

and

$$\nabla^2 A_2(\mathbf{r}) = \sum_j A(\mathbf{x}_j)\nabla^2 W(\mathbf{r}-\mathbf{r}_j;\ h)\Delta V_j \tag{11}$$

simply by knowing the gradient and Laplacian of the smoothing kernel.

We can write Euler's equation (1) as:

$$\frac{d\mathbf{v}}{dt} = -\frac{1}{\rho}\nabla P + \frac{1}{\rho}\mathbf{f} \tag{12}$$

and we can calculate:

$$\frac{1}{\rho}\nabla P = \nabla\left(\frac{P}{\rho}\right) + \frac{P}{\rho^2}\nabla\rho \tag{13}$$

which, applying equation (10), yields the SPH approximation:

$$\frac{d\mathbf{v}_i}{dt} = -\sum_{j,j\neq i} m_j \left(\frac{P_i}{\rho_i^2} + \frac{P_j}{\rho_j^2}\right)\nabla W(\mathbf{r}_i - \mathbf{r}_j;\ h) + \mathbf{b}_i \tag{14}$$

where $\mathbf{b}_i$ is the acceleration of particle i due to external

forces (to be calculated). In the equation above, we may also add any other physical forces (damping, viscous, Coriolis) that are present in the physical system. The density can be estimated as:

$$\rho_i = \sum_j m_j W(\mathbf{r}_i - \mathbf{r}_j;\ h). \tag{15}$$

The SPH approximation equation (14) is the main equation needed to be solved to obtain the time evolution of the fluid.

2.1 Leap Frog time integration

In SPH, we must provide initial conditions – the positions and velocities of the N particles. Then equation (14) may be solved with a time integration method, such as Runge-Kutta or Leap Frog. The Leap Frog method is often preferred because it is explicit, second order (making it better than Euler's method), and symplectic (meaning it conserves energy). The Leap Frog scheme is as follows:

$$\mathbf{v}(t + \Delta t/2) = \mathbf{v}(t - \Delta t/2) + \mathbf{a}(t)\Delta t \tag{16}$$

$$\mathbf{x}(t + \Delta t/2) = \mathbf{x}(t) + \mathbf{v}(t + \Delta t/2)\Delta t \tag{17}$$

that is, we calculate positions and velocities at interleaved time points. There are two important caveats. First, at the start we only know the initial conditions $\mathbf{x}(0)$ and $\mathbf{v}(0)$, and must use Euler's method to step back half a time step and find $\mathbf{v}(-\Delta t/2)$. Second, in equation (16) we see that $\mathbf{a}(t)$ may actually depend on $\mathbf{v}(t)$, which would make the equation implicit. However, a common approximation in SPH is to assume that the velocity at the current time plays a minor role in the computation of the acceleration (i.e. velocity changes are small with each time step) and then the following approximation can be made:

$$\mathbf{v}(t + \Delta t) = \frac{\mathbf{v}(t - \Delta t/2) + \mathbf{v}(t + \Delta t/2)}{2}, \tag{18}$$

which keeps the method explicit and hence more computationally efficient.

2.2 Choosing time steps and scale lengths

In SPH, we need to choose a value for the time step Δt and the kernel smoothing length h and here we discuss some strategies for setting their values.

The time step needs to be satisfy

$$\Delta t \leq \min\{\frac{h}{v_{\max}}, \sqrt{\frac{h}{F_{\max}}}\} \tag{19}$$

to meet the CFL condition for stability (Gingold & Monaghan 1977), where $v_{\max}$ is the maximum velocity and $F_{\max}$ is the maximum force on a particle at a given time. In the present work, we choose a fixed small enough value for Δt that yielded accurate physical results. In larger scale problems, an adaptive time stepping scheme is often employed (Monaghan 1992).

The value of h needs to be chosen carefully in SPH, especially if one is working with a kernel with compact support (such as the cubic spline kernel). If the kernel has compact support and the smoothing length is too small, then the particle may not interact with other particles. One strategy for choosing a kernel length (which proved to be sufficient for our physical simulations of toy stars) is to choose h to have the same order of magnitude as:

$$(\langle \mathbf{r}^2 \rangle - \langle \mathbf{r} \rangle^2)^{1/2} \tag{20}$$

where $\langle , \rangle$ denotes averaging over the particles (Gingold & Monaghan 1977). A more sophisticated approach may be to use an adaptive scheme where a different h_i is chosen for each particle such that a ball of radius h_i around particle i contains k neighbors (Monaghan 1992).

3 TOY STARS

The toy star model of Monaghan & Price (2004) is described by:

$$\frac{d\mathbf{v}_i}{dt} = -\nu \mathbf{v}_i - \sum_{j, j \neq i} m_j \left(\frac{P_i}{\rho_i^2} + \frac{P_j}{\rho_j^2} \right) \nabla W(\mathbf{r}_i - \mathbf{r}_j;\ h) - \lambda \mathbf{x}_i \tag{21}$$

in the CM frame, where the pressure obeys the polytropic equation (2), ν is a damping coefficient added so that the simulation reaches an equilibrium state, and the λ term is a simplified gravitational term. In the static case ($\mathbf{v} = 0$) we can solve the differential equation

$$0 = -\frac{1}{\rho}\nabla P - \lambda \mathbf{x} = -\frac{k(1 + 1/n)}{\rho} \rho^{1/n} \nabla \rho - \lambda \mathbf{x} \tag{22}$$

analytically and the density is given by a simple profile:

$$\rho(r) = \left(\frac{\lambda}{2k(1+n)} (R^2 - r^2) \right)^n \tag{23}$$

where r is radial position and R is the radius of the toy star, beyond which the density is 0. For example, in the case that $n = 1$, we have the simple density profile:

$$\rho(r) = \frac{\lambda}{4k}(R^2 - r^2). \tag{24}$$

If we give the star a radius R and a mass M, then the density must integrate up to the mass, i.e.:

$$M = \int d^d \mathbf{r} \rho(r) \tag{25}$$

where d is the dimension of the problem, which allows us to solve for λ:

$$\lambda = \begin{cases} 2k\pi^{-1/n} \left(M(1+n)/R^2 \right)^{1+1/n} / M & d = 2 \\ 2k(1+n)\pi^{-3/(2n)} \left(\frac{M\Gamma(\frac{5}{2}+n)}{R^3 \Gamma(1+n)} \right)^{1/n} / R^2 & d = 3 \end{cases} \tag{26}$$

In summary, we expect SPH to yield these static solutions when the particle motions are damped. Comparison with analytic solutions provides a means to test the accuracy of SPH.

4 IMPLEMENTATION OF SPH

We developed a SPH code in MATLAB and ran all simulations on a 2.50 GHz processor dual-core laptop. Each simulation completed in under half an hour. In this section we present some pseudo-code highlighting the main aspects of the SPH algorithm.

Densities are calculated as follows:

```
Calculate_Density(x, m, h){
for i = 1 : N
    % initialize density with i = j contribution
    rho(i) = m * kernel(0, h);
    for j = i + 1 : N
        % calculate vector between two particles
        uij = x(i, :) − x(j, :);
        rho_ij = m * kernel(uij, h);
        % add contribution to density
        rho(i)+ = rho_ij;
        rho(j)+ = rho_ij;
    end
end
}
```

where $\mathbf{x}$ are the particle positions, m is the mass of each particle, and h is the smoothing length.

The accelerations due to pressure, damping, and gravity are calculated as follows:

```
Calculate_Acceleration(x, v, m, rho, P, nu, lambda, h){
% initialize accelerations
a = zeros(N, dim);
% add damping and gravity
for i = 1 : N
    a(i, :)+ = −nu * v(i, :) − lambda * x(i, :);
end
% add pressure
for i = 1 : N
    for j = i + 1 : N
        % calculate vector between two particles
        uij = x(i, :) − x(j, :);
        % calculate acceleration due to pressure
        p_a = −m * (P(i)/rho(i)^2 + P(j)/rho(j)^2) * gradkernel(uij, h);
        a(i, :)+ = p_a;
        a(j, :)+ = −p_a;
    end
end
}
```

where $\mathbf{v}$ are the particle velocities and $\mathbf{P}$ are the calculated pressures from the density.

The main loop in the SPH algorithm (the Leap Frog scheme) looks like:

```
Main_Loop
for i = 1 : max_time_step
    v_phalf = v_mhalf + a * dt;
    x+ = v_phalf * dt;
    v = 0.5 * (v_mhalf + v_phalf);
    v_mhalf = v_phalf;
    % update densities, pressures, accelerations
    rho = Calculate_Density(x, m, h);
    P = k * rho. ∧ (1 + 1/npoly);
    a = Calculate_Acceleration(x, v, m, rho, P, nu, lambda, h);
end
```

With the simple algorithm we have prescribed above, we see that given N particles the algorithm is order N^2 because we have to look at pairwise interactions between particles. Due to the symmetric nature of the pairwise interaction, the amount of computations is reduced by a factor of 2. Alternative methods have been developed in literature to make the computations more efficient, such as computing interactions only between the k nearest neighbors and using a hierarchical tree algorithm (see, e.g. Hernquist & Katz (1989)). In this project, we will only deal with < 1000 particles and therefore can calculate all pairwise interactions in a reasonable time.

Table 1. Parameters for the toy star simulations

Simulation 1	typical star collapse into equilibrium
Parameter	**Value**
number of particles	$N = 100$
dimension	$d = 2$
star mass	$M = 2$
star radius	$R = 0.75$
smoothing length	$h = 0.04/\sqrt{N/1000}$
time step	$\Delta t = 0.04$
damping	$\nu = 1$
pressure constant	$k = 0.1$
polytropic index	$n = 1$
max time steps	400
kernel	spline
initial config.	random inside circle radius R
Simulation 2	reduced number of particles
number of particles	$N = 50$
Simulation 3	increased number of particles
number of particles	$N = 400$
Simulation 4	Gaussian kernel
kernel	Gaussian
Simulation 5	3-D simulation
number of particles	$N = 300$
dimension	$d = 3$
Simulation 6	altered polytropic index
polytropic index	$n = 2$
Simulation 7	more damping
damping	$\nu = 4$
Simulation 8	increased smoothing length
smoothing length	$h = 0.06/\sqrt{N/1000}$
Simulation 9	separated material init. condition
number of particles	$N = 500$
initial config.	1/5 of particles offset by distance 2.4

5 SIMULATIONS

We performed 11 simulations of toy stars in dimensionless units. In most of the simulations, we wished to find the equilibrium state of a collapsing polytropic star from random initial conditions. We tested the effect of altering the number of particles N in the simulation, the type of kernel, the dimension of the problem, the polytropic index, the effect of the size of damping, the size of the smoothing length h, and the effect of initial conditions. We also simulated collisions of two toy stars to see where the material from the initial stars end up in the final merged star and what are the physical conditions during the evolution. The simulation parameters and setup are described in Tables 1 and 2. If a parameter is not listed under a simulation, it takes the same value as the first simulation in the table. All simulations are performed in the center of mass frame. Included with the report are videos (*.wmv files) of some of the simulations.

Table 2. Parameters for the collision simulations

Simulation 10	soft collision of 2 stars – head on
Parameter	**Value**
number of particles	$N = 500$
dimension	$d = 2$
total mass	$M = 2$
final star radius	$R = 0.75$
particles per star	$400, 100$
smoothing length	$h = 0.04/\sqrt{N/1000}$
time step	$\Delta t = 0.04$
damping	$\nu = 6$
pressure constant	$k = 0.1$
polytropic index	$n = 1$
max time steps	400
kernel	spline
initial config.	two stars in equilibrium
	center distance 1.2 apart
	relative velocity 0.4
Simulation 11	soft collision of 2 stars – low damping
Parameter	**Value**
damping	$\nu = 2$
initial config.	two stars in equilibrium
	center distance 1.2 apart
	relative velocity 0.8
	impact param. 0.6

The parameter value for λ (gravitational strength) is given by equation (26) and the mass per particle is $m = M/N$.

6 RESULTS

We carry out the simulations with the setups mentioned in the previous section. Snapshots of simulations 1, 3, 5, 9, 10, and 11 are shown in Figures 2, 3, 4, 5, 6, and 7. I do not show all the simulations because a number of them are visually similar. The simulations were carried out until equilibrium was reached (up to 400 time steps may be required), which was determined by monitoring the central density of the star (see Figure 8). The central density was found to oscillate, but dampen with time until the equilibrium value was reached. The plot is based on Figure 1 of Gingold & Monaghan (1977).

In addition we find the radial density profile along the x-axis (sampled at 100 points x_i between 0 and R) of the equilibrium states of simulations 1–9 and compare it with what is expected from analytic theory. Plots of the density profiles of simulations 1–9 are shown in Figures 9, 10, 11, and 12. We calculate an error associated with each SPH simulation, namely:

$$\text{error} = \frac{\sum_{i=1}^{100} \|\rho_{\rm SPH}(x_i) - \rho_{\rm theory}(x_i)\|_2^2}{100} \qquad (27)$$

The errors in reported in Table 3 and the simulations closely match the theoretical profile. The results will be more carefully discussed in § 7.

For the collision sequences (simulations 10 and 11), we plot the maximum pressure in the system as a function of time in Figure 13 to give an example of how SPH can be used to track the time dependent evolution of physical quantities. Its significance will be discussed in § 7.

Table 3. Errors of SPH simulations compared to analytic solutions

simulation	error
1	0.002302
2	0.004032
3	0.003423
4	0.003429
5	0.005435
6	0.005856
7	0.002393
8	0.005080
9	0.002497

7 DISCUSSION

7.1 Discussion of results

First we explore what the different simulations 1–9 of collapsing toy stars reaching equilibrium tell us about the SPH method. Simulation 1 is our original setup, which we take and vary one parameter at a time to create simulations 2–9. In Figure 9 we plot the equilibrium density profile of a polytropic toy star simulated with 100, 50, and 400 particles (simulations 1, 2, 3). We notice a few key features in the SPH profile versus the analytic profile. First of all, we reproduce the analytic solution quite well. At $r = 0$, the central density $\rho(0)$ is best estimated using larger number of particles, N. We see that using only $N = 50$ particles gives a density that is slightly small due to the approximations we are making. Gingold & Monaghan (1977) encountered the same phenomenon, performing simulations with $N = 50$ and $N = 80$ particles, and to account for this error normalized all densities by the central densities. We see however, that we can expect to obtain the correct actual densities by using a larger number of particles, as expected by the theory of SPH. There is another effect we notice in Figure 9. Notably, the density profiles deviate at $r = R$, where the density goes to zero in the analytic model. Here the shape of the kernel affects the density profile and this is an unavoidable artifact of SPH. The smoothing length h may be altered to give a better result. Generally, h can be reduced to provide better answers as long as there are enough particles so that the neighborhood of radius h around each particle contains other particles. Reducing h to obtain better answers again means that we need a larger number of particles.

There does not seem to be a significant difference between the Gaussian kernel (simulation 4) and the spline kernel in the determination of the equilibrium density profile, as seen from Figure 10. Gingold & Monaghan (1977) recommends using the Gaussian kernel to obtain the best physical answers, but we see that a spline kernel can be effectively used, and makes calculations more efficient since it has compact support.

The final density profile is also independent of damping, as one would hope, as seen by comparing simulations 1 and 7 in Figure 10. Increased damping can significantly reduce the time it takes for the star to equilibrize (Figure 8). With no damping, the star would continue to oscillate and different oscillatory modes could be excited by initial conditions (c.f. Monaghan & Price (2004)).

An increased smoothing length slightly smooths out the density profile, especially near $r = R$ (compare simulations

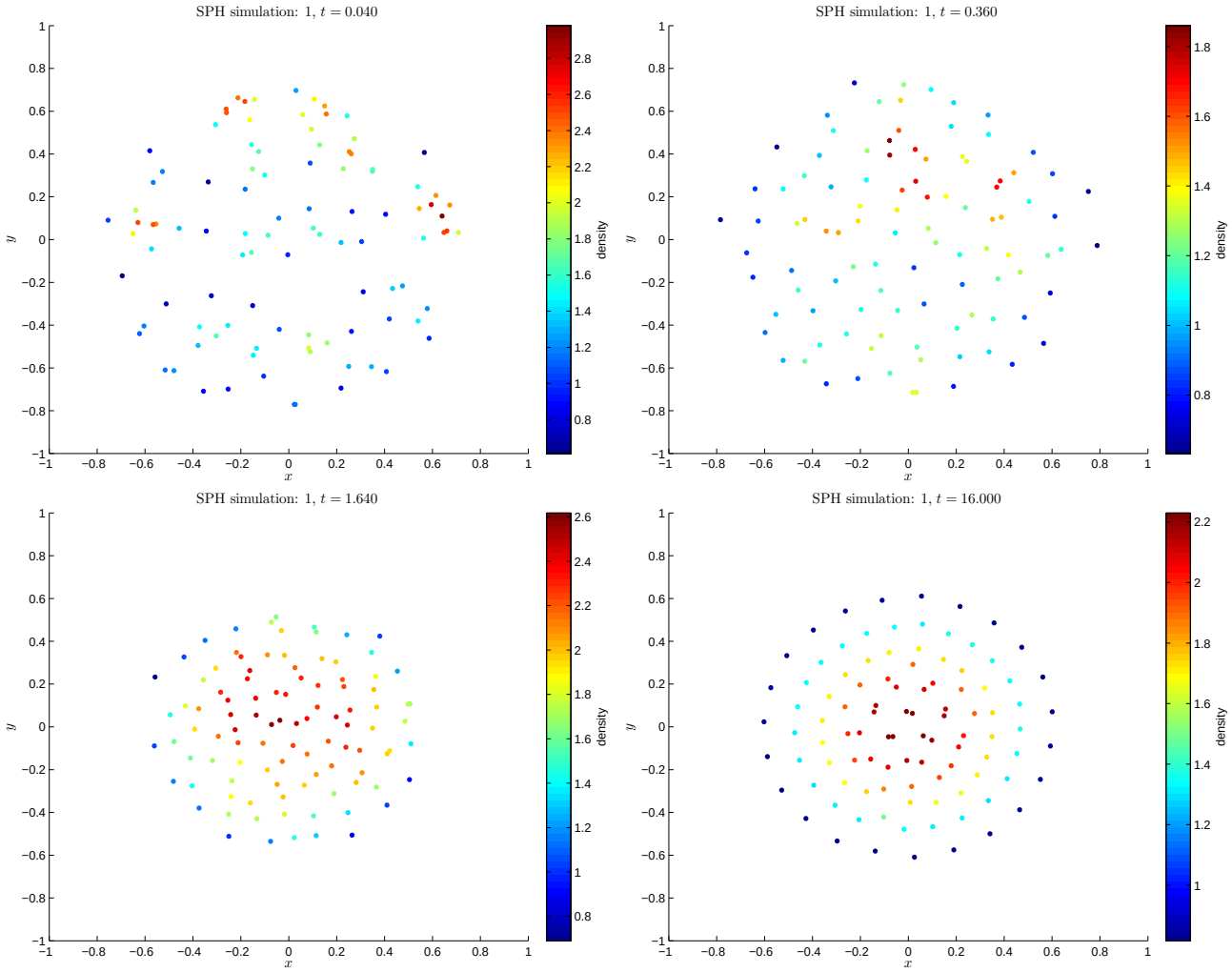

Figure 2. A few steps of the progressive ordering of particles towards equilibrium in Simulation 1 of the toy star.

1 and 8 in Figure 10), although one can obtain reasonable answers for a number of smoothing length scales. The best value to use for h is still an open problem (Monaghan 1992, 2005).

We can reach the equilibrium state from any initial state. In simulation 9 we separate the fluid into two disjoint random distributions and show that we recover the same single star density profile (see Figure 10). This is similar to a collision problem, except that here each disjoint set of particles is not in equilibrium initially.

In addition, we demonstrate that our code can easily be extended to 3D. In simulation 5 we recover the analytic density profile with SPH (Figure 11). We also tested that the SPH simulation gives the correct density profile (both shape and magnitude) for a different polytropic index (see simulation 6 in Figure 12).

We have also simulated the collision of two toy stars and recorded the evolution of their maximum pressures (Figure 13). The purpose was not to draw astrophysical conclusions (since we are working with a simplified model and arbitrary units) but to demonstrate how SPH can easily be used to find the evolution of physical quantities (pressure, energy, temperature, etc.) which may otherwise be difficult or impossible to compute analytically. SPH also gives us insight into the structures that may form given initial conditions and the time scales certain changes happen. An astrophysicist, for example, may test a hypothesis about how the Moon was formed with soft-body collision using SPH, and see what initial conditions may have been required to complete the process in a certain amount of time or create necessary physical conditions that explain the composition of the Moon.

7.2 Possible extensions of the work

The SPH algorithm can be extended in a variety of ways to study fluid dynamics. Notably, the equations of motion (21) can be changed to reflect the physics of a certain setup. For example, the gravitational force may be replaced by a downwards constant force to model uniform gravity on the Earth's surface (this is built into our code, although we do not use it for modeling toy stars). We could then use the SPH code to simulate, for example, drops of water, waves, or a breaking dam.

Often, a term is added into the equations of motion to

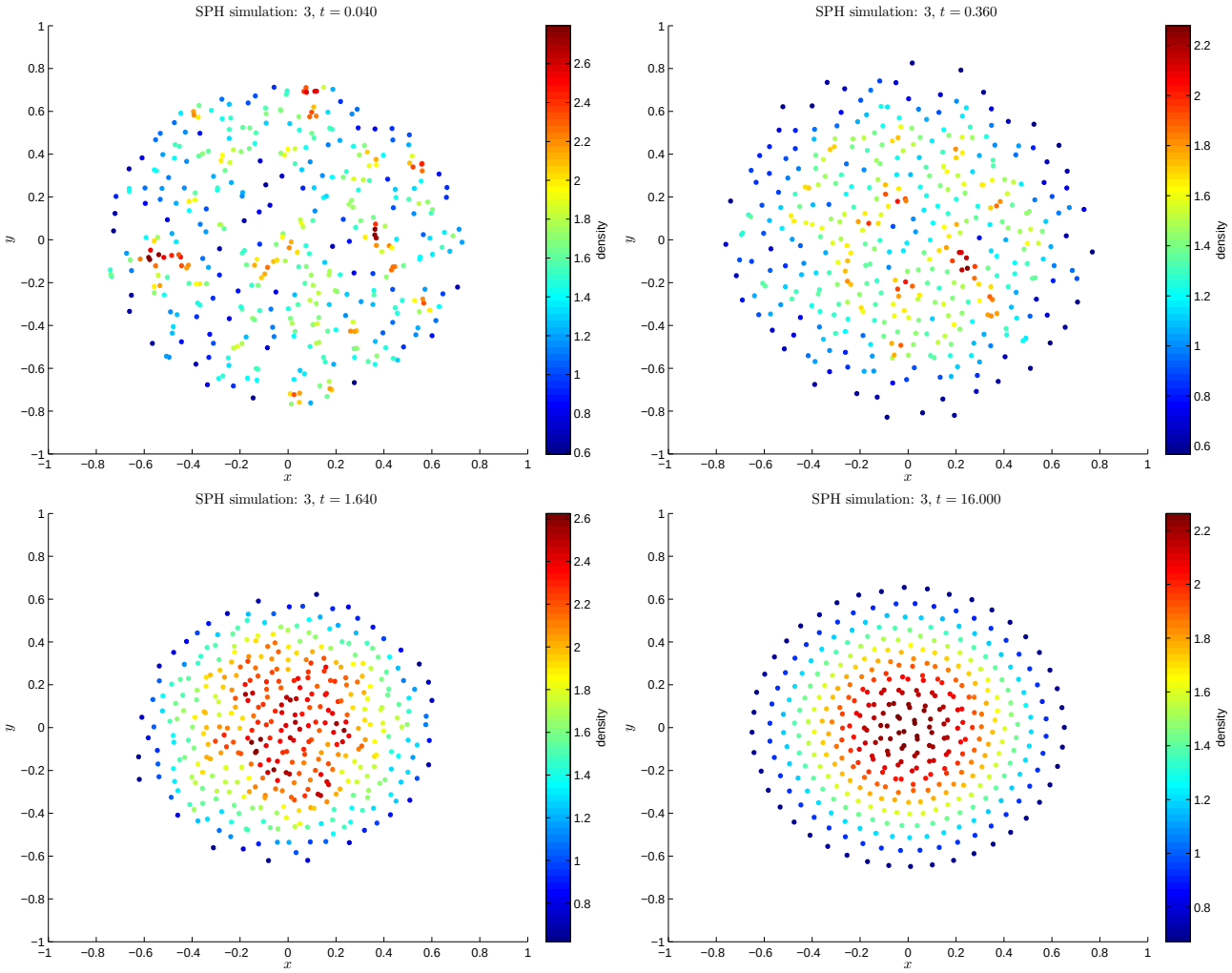

Figure 3. A few steps of the progressive ordering of particles towards equilibrium in Simulation 3 of the toy star.

represent viscous forces. Most commonly used is:

$$-\sum_{j,j\neq i} m_j \left(\Pi_{ij}\right) \nabla W(\mathbf{r}_i - \mathbf{r}_j;\ h) \tag{28}$$

where Π_{ij} has the form

$$\Pi_{ij} = \begin{cases} -\frac{-\alpha c \mu_{ij} + \beta \mu_{ij}^2}{\rho_{ij}} & \mathbf{v}_{ij} \cdot \mathbf{r}_{ij} < 0 \\ 0 & \mathbf{v}_{ij} \cdot \mathbf{r}_{ij} > 0 \end{cases} \tag{29}$$

where

$$\mu_{ij} = \frac{h \mathbf{v}_{ij} \cdot \mathbf{r}_{ij}}{\mathbf{r}_{ij}^2 + \eta^2} \tag{30}$$

where $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$, $\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j$, $\rho_{ij} = \rho_i - \rho_j$ (see Monaghan (1992)).

It is also possible to add a boundary to the SPH simulation which serves as a wall. This was added into our SPH code, although was not required for the toy star simulations. A boundary wall can be simulated as a spring. It gives any particle inside the wall a force proportional to the depth of penetration and directed perpendicular to the wall. An additional damping factor may also be added.

With an increased number of particles and lots of computing power, very complicated hydrodynamic processes, from galaxy dynamics to planetary disks, may be studied using SPH, which is one of the primary scientific applications of the method.

7.3 Comparison of SPH to grid-based methods

The SPH method, unlike finite difference grid-based methods, is inherently non-dissipative. This means that in the absence of explicitly added dissipative terms, the numerical solution will neither dissipate nor diffuse energy artificially (Price 2011). This is one of the strengths of SPH in modeling certain fluid flows. This can also be a disadvantage in some cases. Dissipative terms need to be added explicitly to capture physical processes such as shocks (for example, the viscosity term equation (28) which contains non-physical terms) (Price 2011).

Unlike grid-based methods, the Hamiltonian that describes the equations of motion of particles in SPH guarantees that there is a particle arrangement that minimizes the Lagrangian (Price 2011). This makes SPH very useful for studying the evolution of a system reaching a stable equilibrium state as we have done. A stable and noise-free particle arrangement is a usually a good indicator of the accuracy

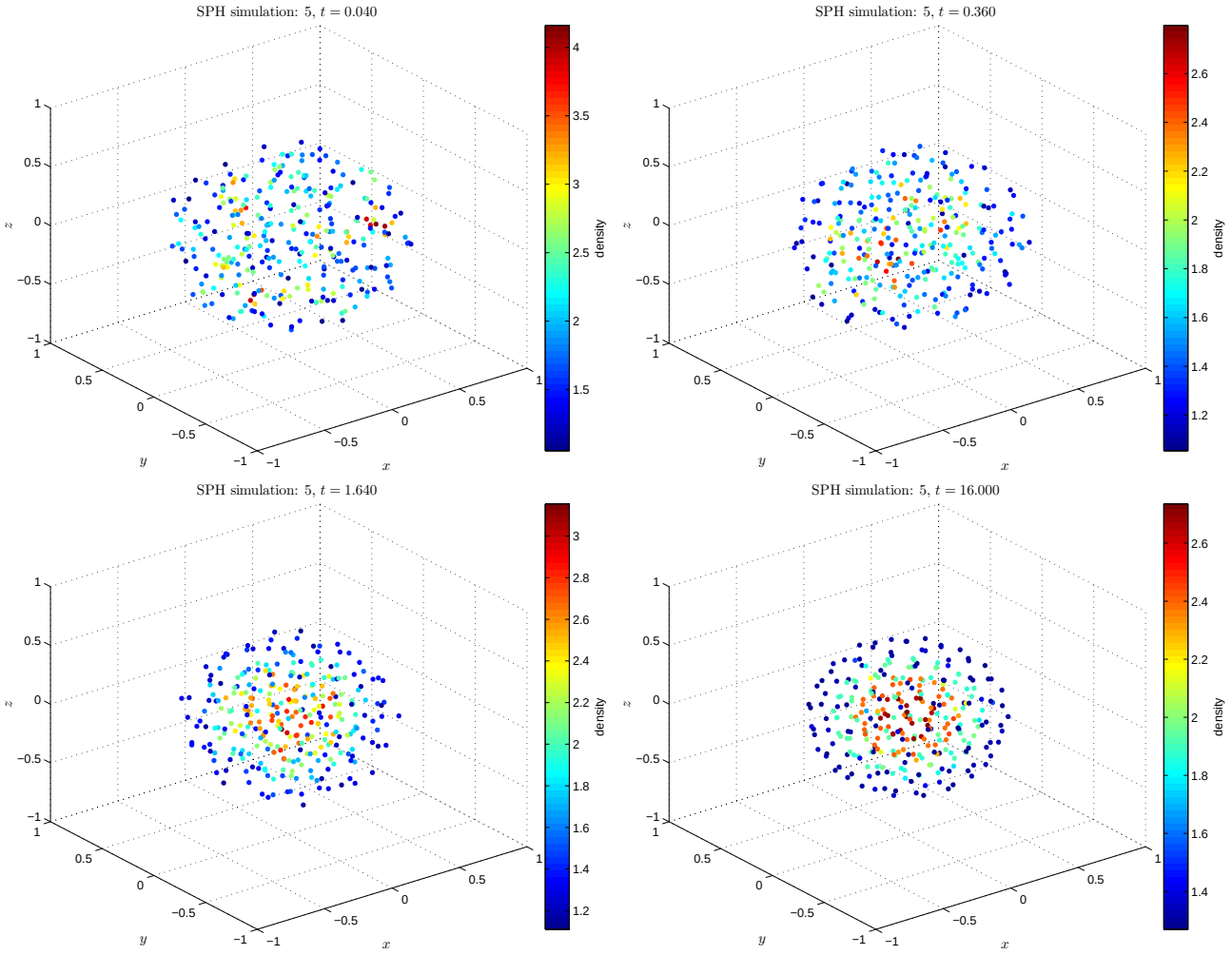

Figure 4. A few steps of the progressive ordering of particles towards equilibrium in Simulation 5 of the toy star.

of SPH solutions (Price 2011). We see in our simulations (Figures 2, 3, 4, 5, 6, and 7) that this is the case.

The resolution of SPH is determined by the masses of the particles (Price 2011) instead of grid size in finite difference methods. This is very useful in astrophysics, making SPH a great choice to study structure formation and gravitational collapse. However, in cases where the number of SPH particles required to resolve a phenomena needs to be the same order of magnitude as the number of cells in a grid-based version of the problem, the grid-based method is more computationally efficient, by a factor of around 10 in practice (Price 2011).

Much research and progress has been done with both particle and grid-based algorithms. One recent development, which is sort of a mix of the two, has been to use a mesh which distorts with time with the local fluid velocity, resulting in an effectively Lagrangian method (Duffell & MacFadyen 2011). The algorithm by Duffell & MacFadyen (2011) uses Voronoi tessellations to generate a numerical mesh which moves its mesh-generating points in the tessellation.

8 CONCLUSION

In conclusion, we have demonstrated that SPH can reliably reproduce expected physical results in fluid dynamics. We have used SPH to determine the equilibrium density profile of polytropic toy stars, which agreed well with analytic profiles. We also explored the collision sequence of two polytropic stars to demonstrate how SPH can be used to track the evolution of physical quantities. We also discussed SPH compared to finite difference methods. Both approaches have their own strengths.

REFERENCES

Benz W., Slattery W. L., Cameron A. G. W., 1986, Icarus, 66, 515
Braune L., Lewiner T., 2009, Dept. Math, PUC-Rio, 11, 1
Duffell P. C., MacFadyen A. I., 2011, ApJS, 197, 15
Gingold R. A., Monaghan J. J., 1977, MNRAS, 181, 375
Hernquist L., Katz N., 1989, ApJS, 70, 419
Lucy L. B., 1977, AJ, 82, 1013
Miyama S. M., Hayashi C., Narita S., 1984, ApJ, 279, 621
Monaghan J. J., 1992, ARA&A, 30, 543
—, 2005, Reports on Progress in Physics, 68, 1703

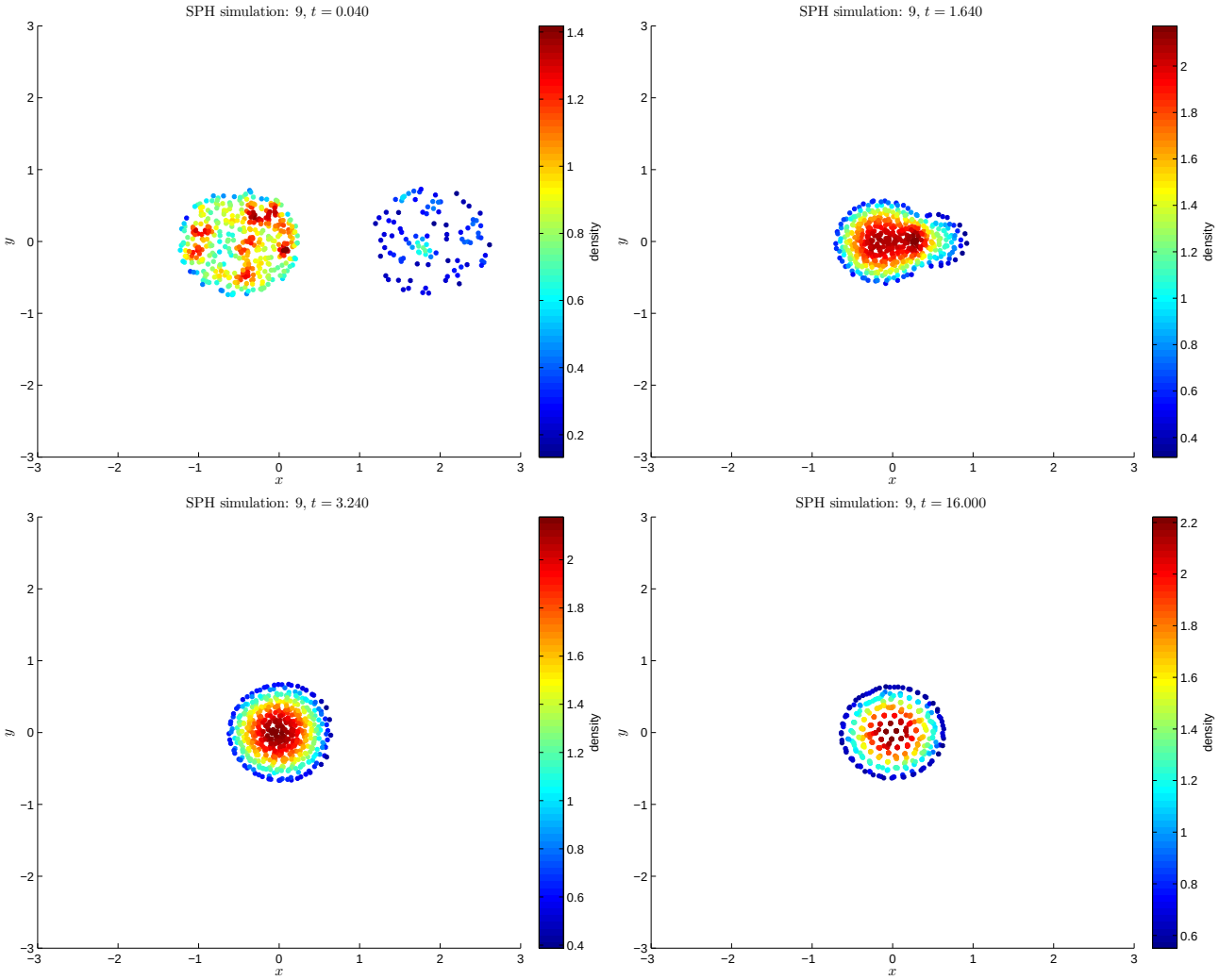

Figure 5. A few steps of the progressive ordering of particles towards equilibrium in Simulation 9 of the toy star.

Monaghan J. J., Price D. J., 2004, MNRAS, 350, 1449
—, 2006, MNRAS, 365, 991
Price D. J., 2011, ArXiv e-prints
Rosswog S., 2009, NAR, 53, 78
Springel V., 2005, MNRAS, 364, 1105

This paper has been typeset from a TEX/ LATEX file prepared by the author.

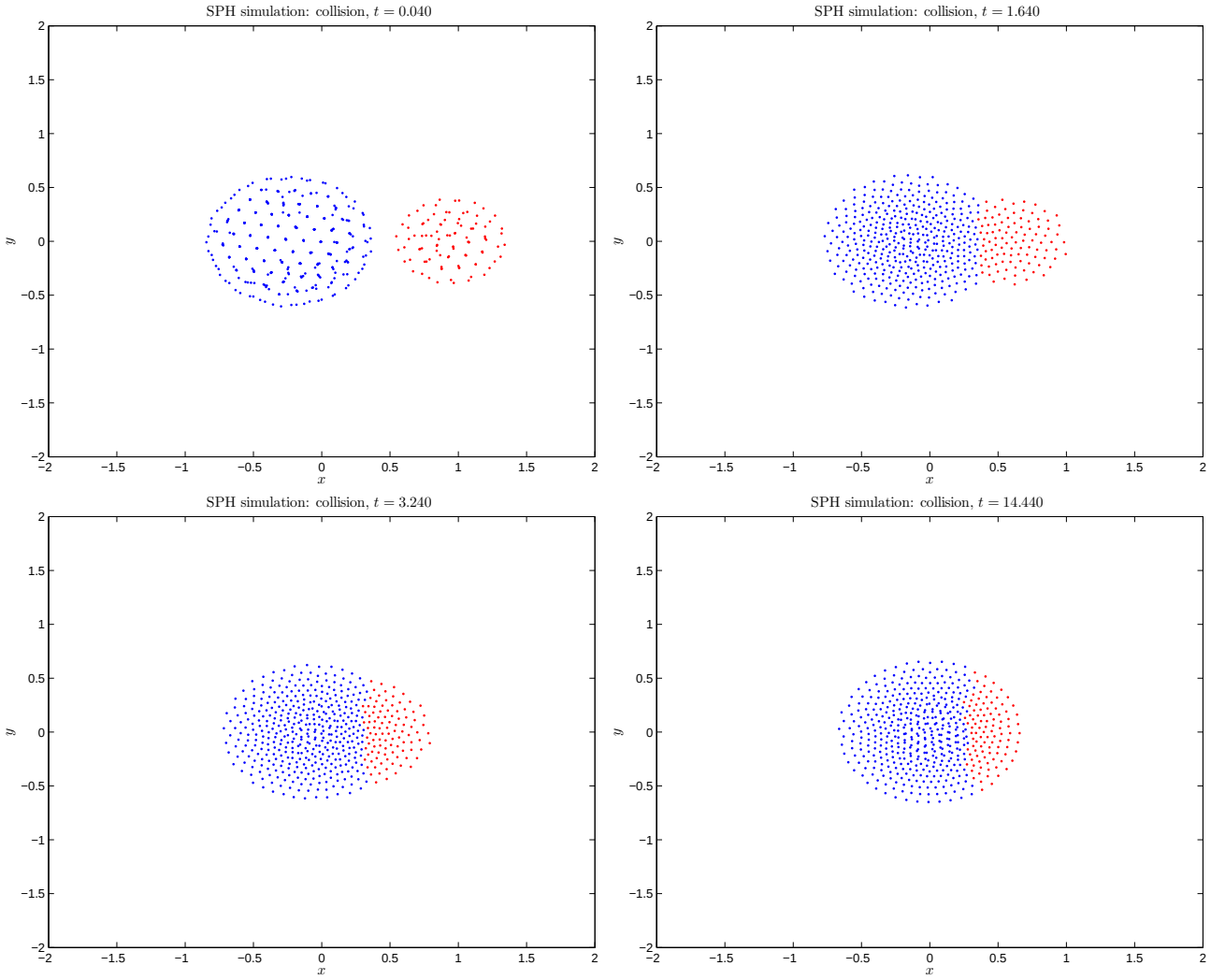

Figure 6. A few steps of the collision in Simulation 10.

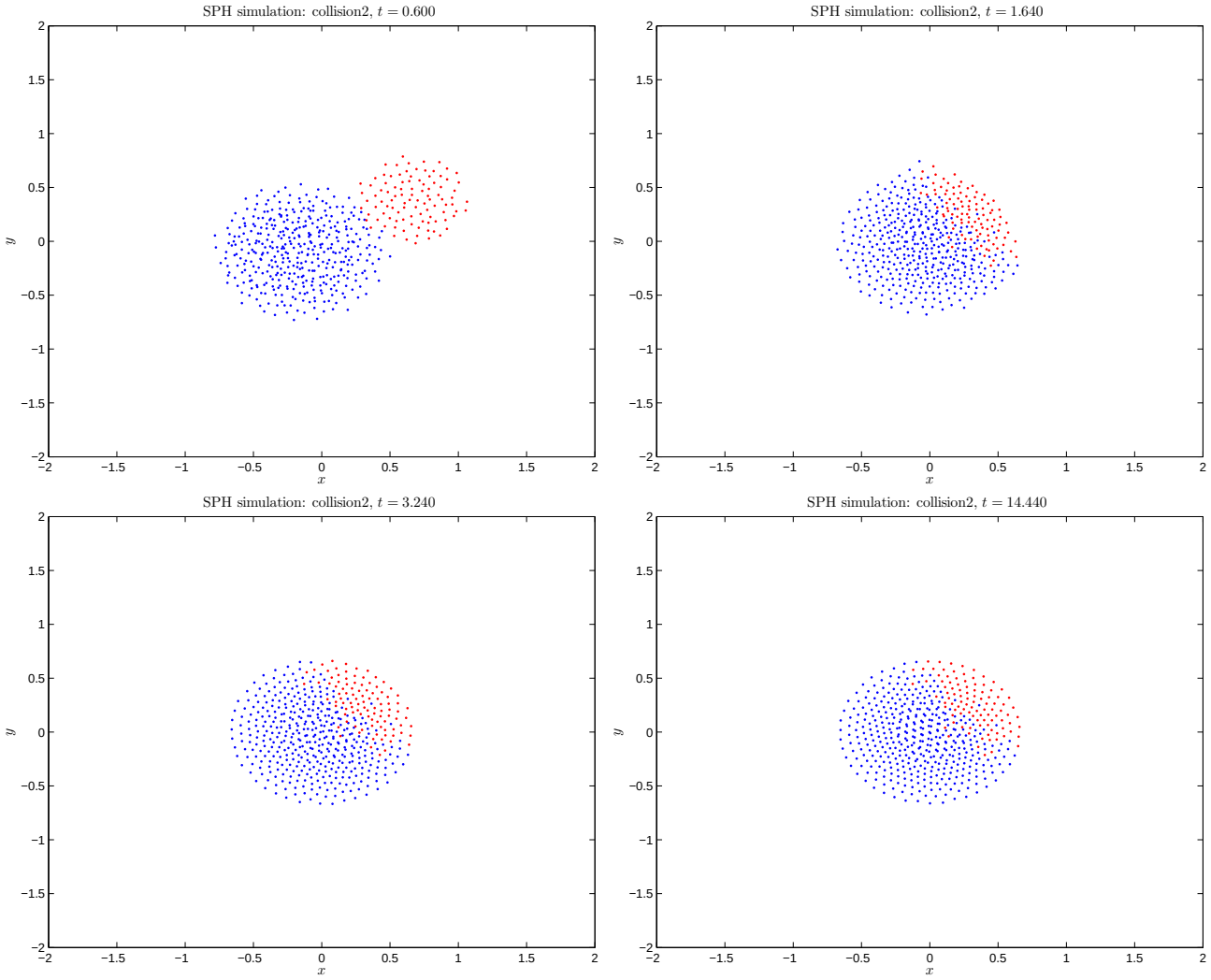

Figure 7. A few steps of the collision in Simulation 11.

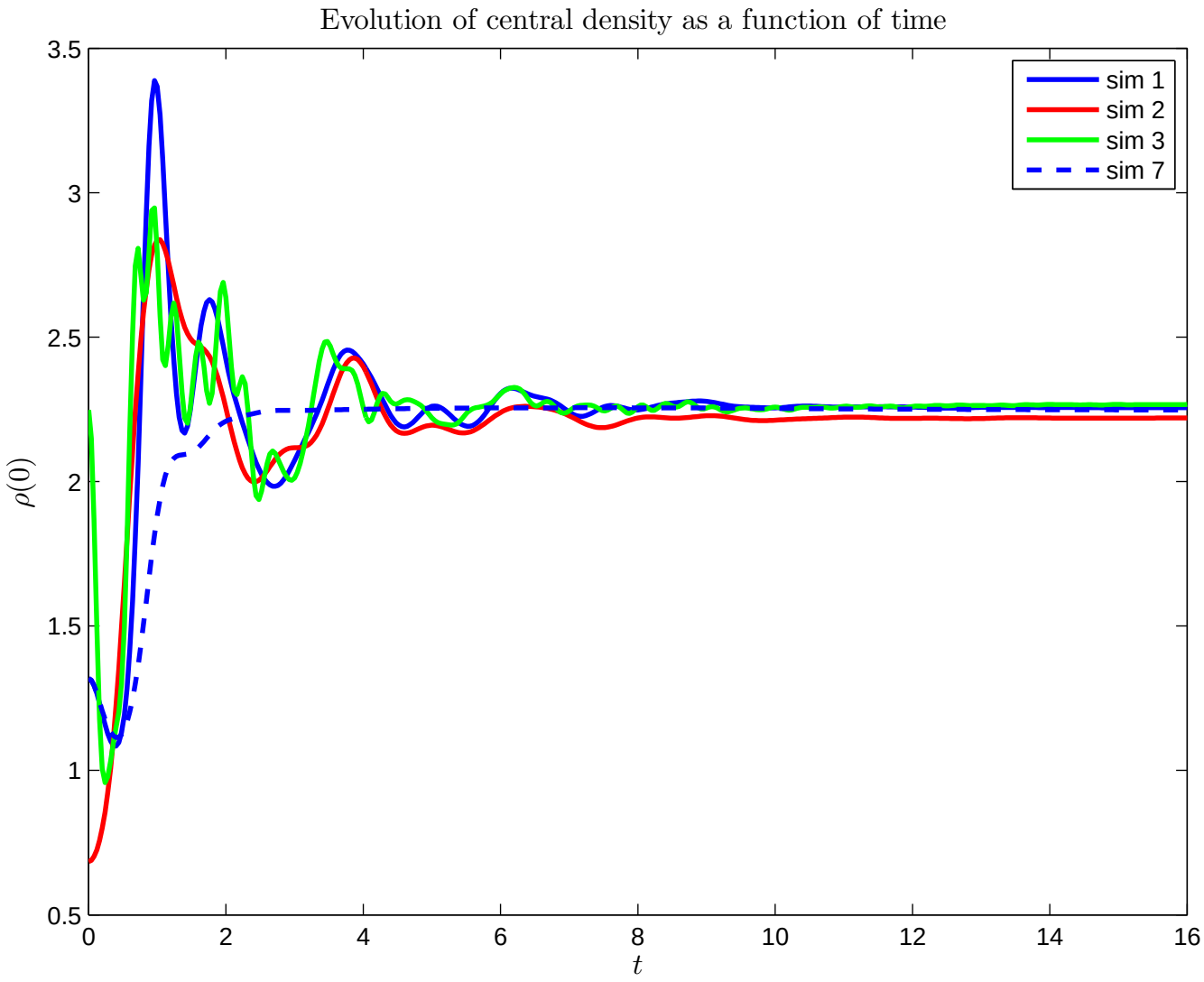

Figure 8. Evolution of central densities as a function of time for simulations 1, 3, 5, 7. We see that the simulations converge to the same final density (with small deviation in simulation 2 due to the low number ($N = 50$) of particles). In simulation 7, the increased damping allows for the star to reach its equilibrium state more quickly.

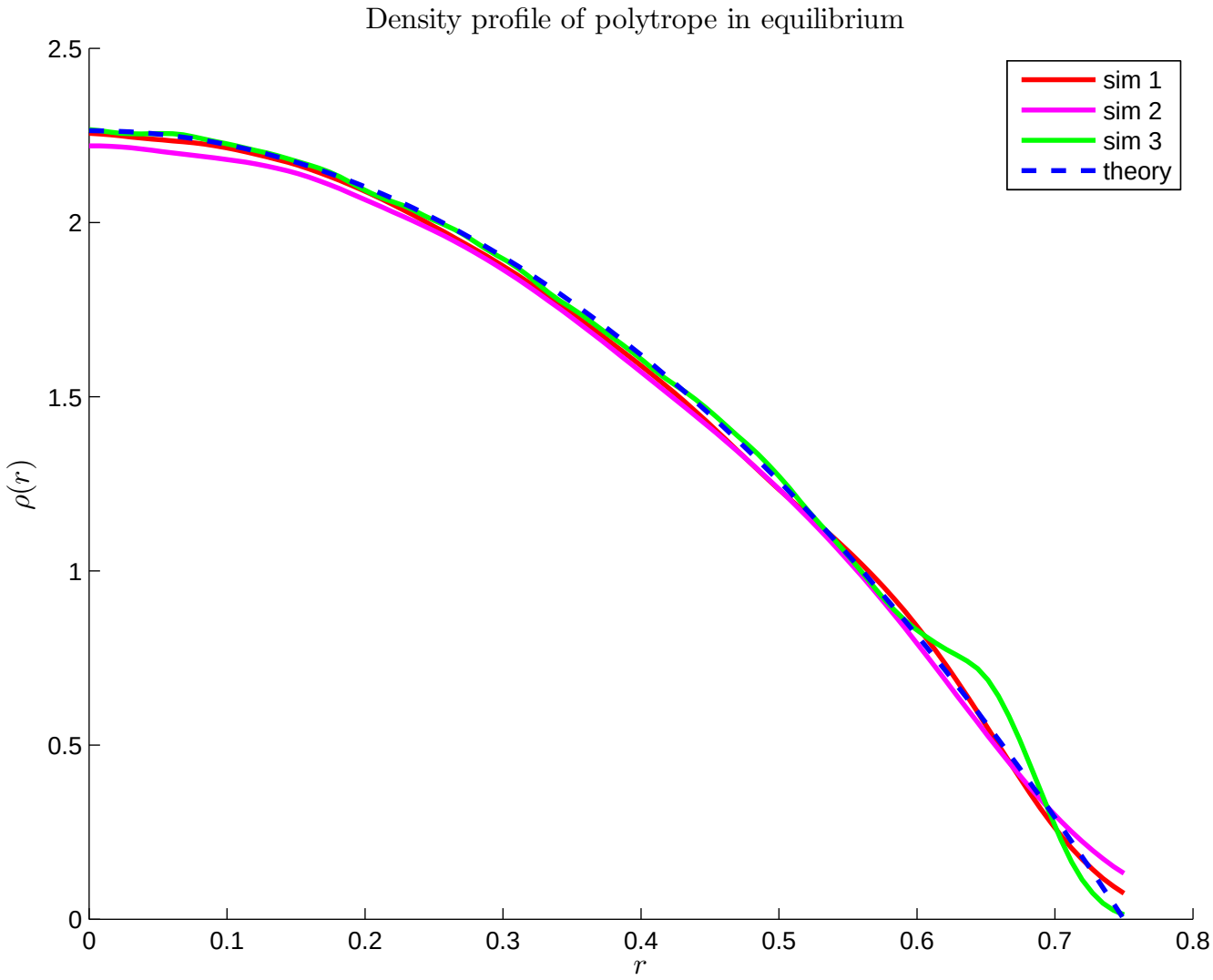

Figure 9. Density profiles of simulations 1, 2, 3 compared to analytic model.

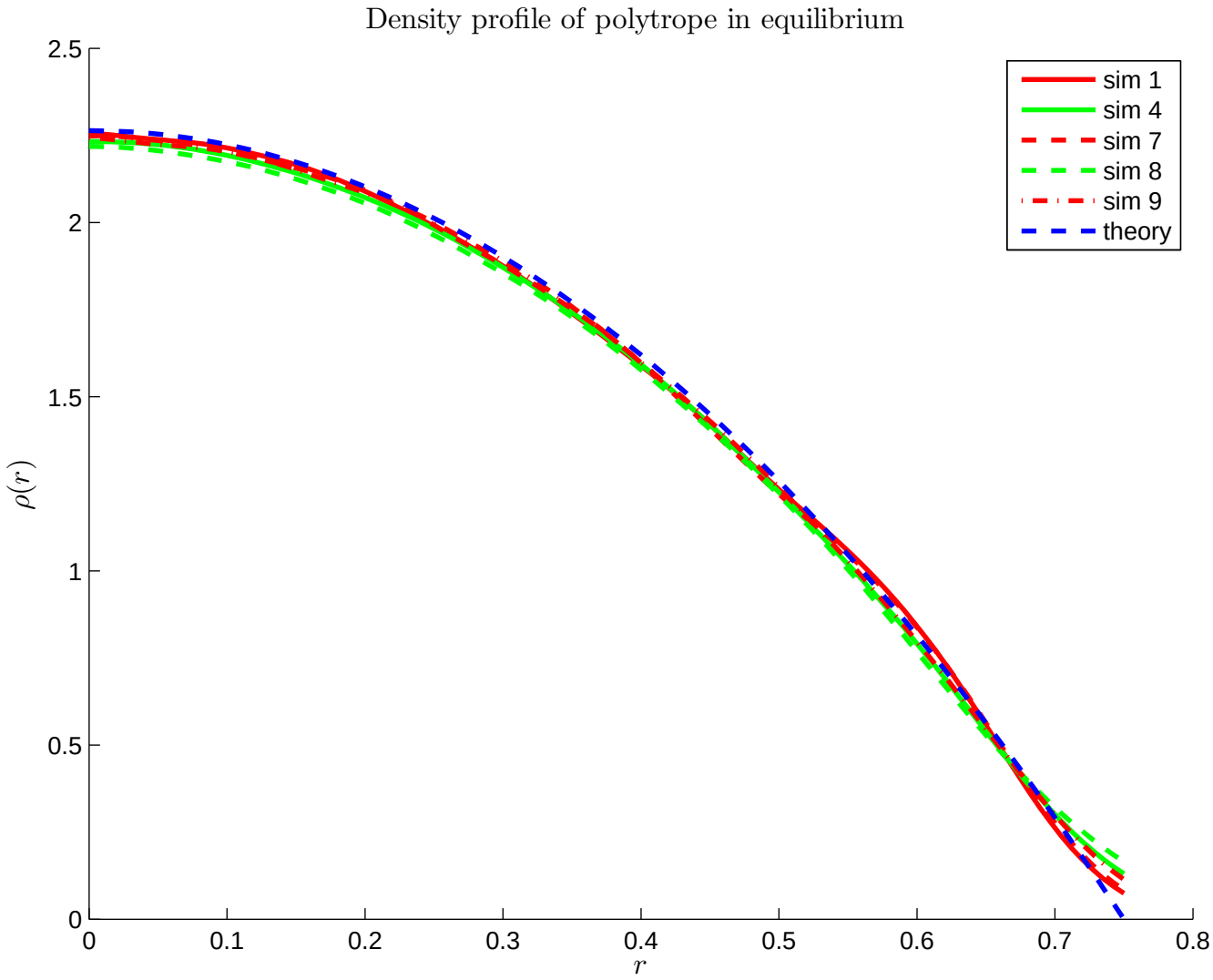

Figure 10. Density profiles of simulations 4, 7, 8, 9 compared to analytic model.

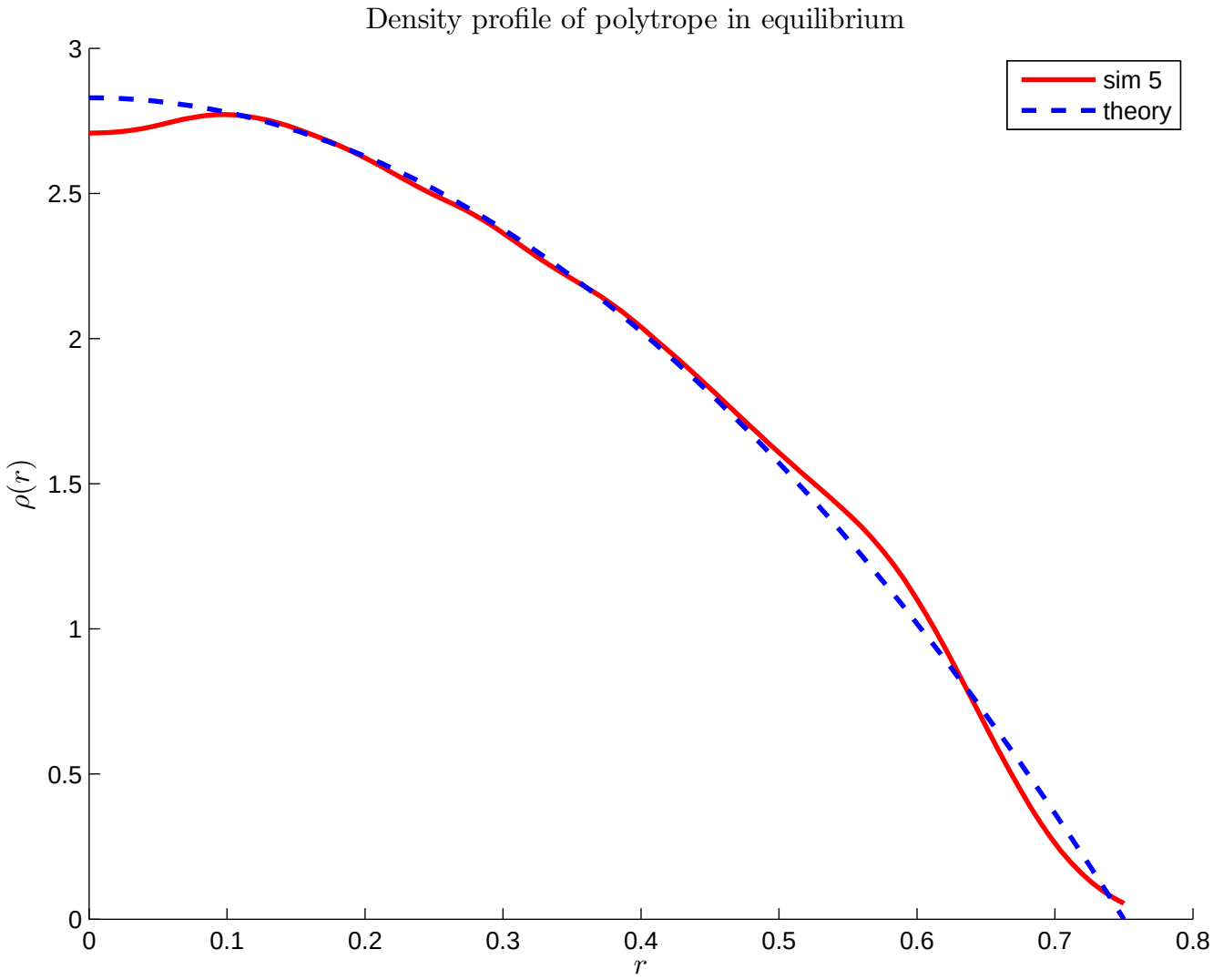

Figure 11. Density profiles of simulation 5 compared to analytic model.

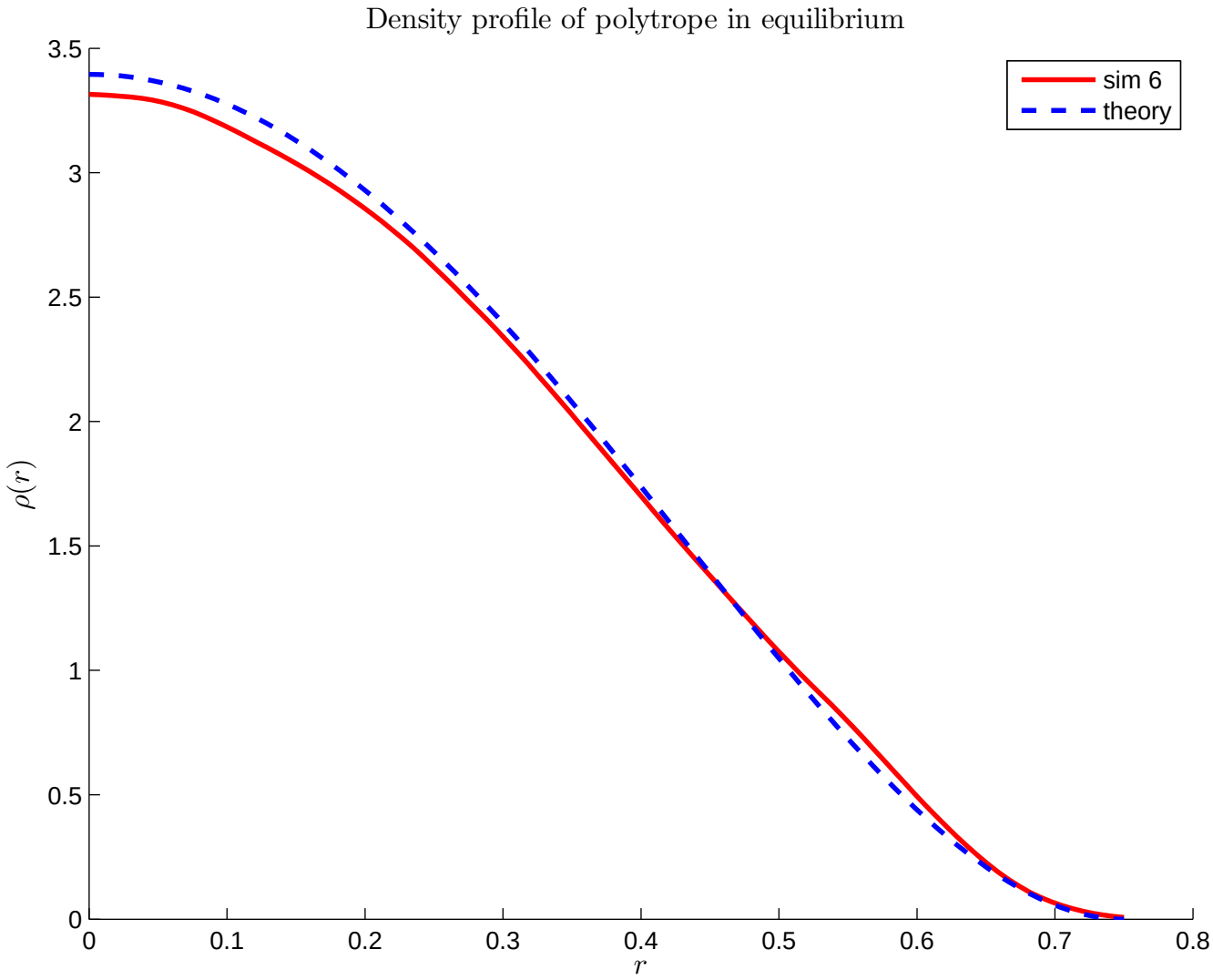

Figure 12. Density profiles of simulation 6 compared to analytic model.

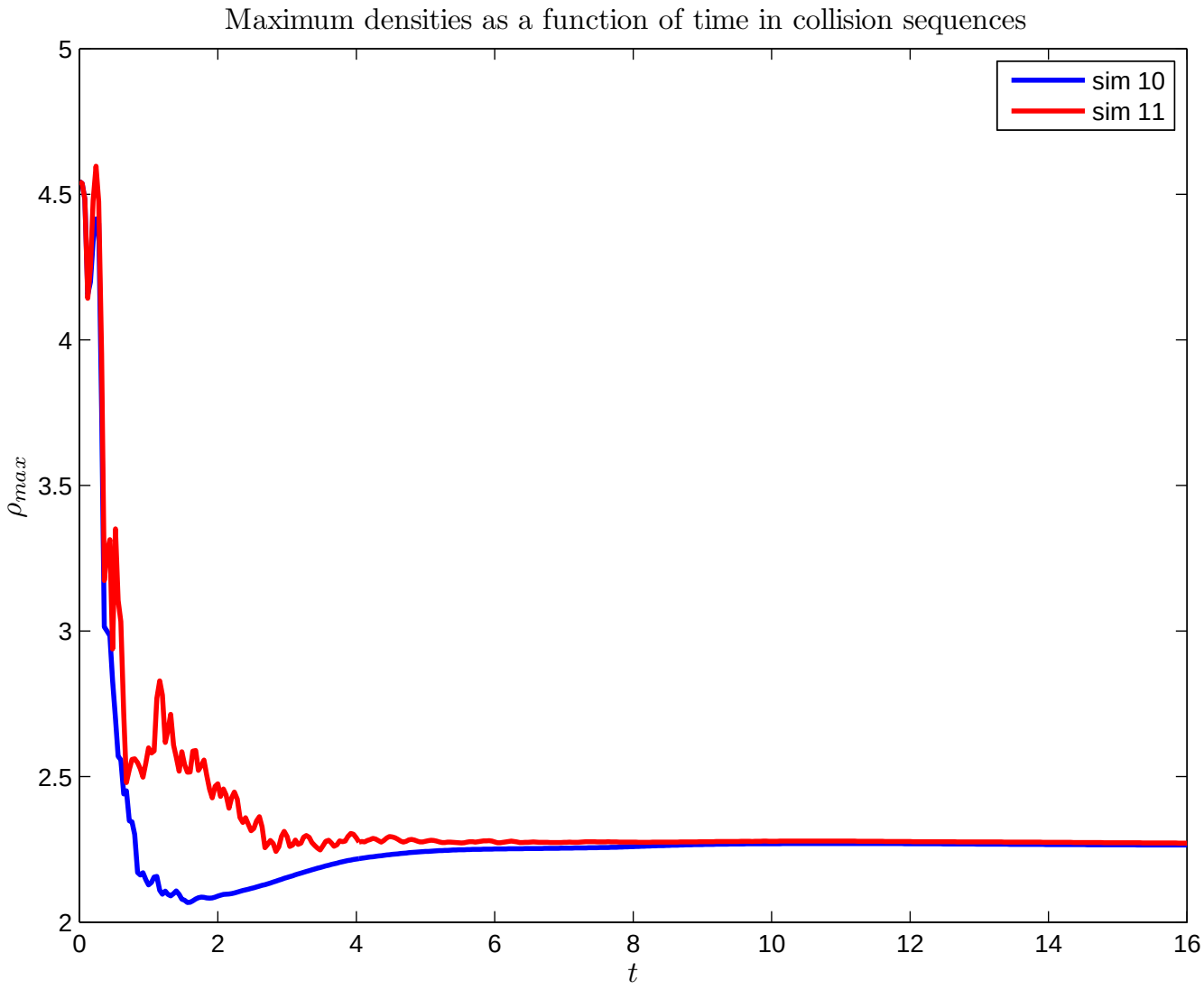

Figure 13. Maximum densities in the collision sequences of simulation 10 and 11.