

EMBEDDING LAGRANGIAN SINK PARTICLES IN EULERIAN GRIDS

MARK R. KRUMHOLZ

Department of Physics, University of California, Berkeley, CA 94720; krumholz@astron.berkeley.edu

CHRISTOPHER F. McKEE

Departments of Physics and Astronomy, University of California, Berkeley, CA 94720; cmckee@astron.berkeley.edu

AND

RICHARD I. KLEIN

Department of Astronomy, University of California, Berkeley, CA 94720; and Lawrence Livermore National Laboratory,
P.O. Box 808, L-23, Livermore, CA 94550; klein@astron.berkeley.edu

Received 2003 November 24; accepted 2004 April 15

ABSTRACT

We introduce a new computational method for embedding Lagrangian sink particles into a Eulerian calculation. Simulations of gravitational collapse or accretion generally produce regions whose density greatly exceeds the mean density in the simulation. These dense regions require extremely small time steps to maintain numerical stability. Smoothed particle hydrodynamics (SPH) codes approach this problem by introducing nongaseous, accreting sink particles, and Eulerian codes may introduce fixed sink cells. However, until now there has been no approach that allows Eulerian codes to follow accretion onto multiple, moving objects. We have removed that limitation by extending the sink particle capability to Eulerian hydrodynamics codes. We have tested this new method and found that it produces excellent agreement with analytic solutions. In analyzing our sink particle method, we present a method for evaluating the disk viscosity parameter α due to the numerical viscosity of a hydrodynamics code and use it to compute α for our Cartesian adaptive mesh refinement (AMR) code. We also present a simple application of this new method: studying the transition from Bondi to Bondi-Hoyle accretion that occurs when a shock hits a particle undergoing Bondi accretion.

Subject headings: accretion, accretion disks — hydrodynamics — methods: numerical — shock waves

Online material: color figures

1. INTRODUCTION

Simulations of gaseous collapse and accretion are ubiquitous in astrophysics, and all of them face a common difficulty: gravitational collapse leads to the formation of structures on length scales very small compared to the initial collapsing object. This leads to an enormous dynamic range that generally makes the full problem computationally infeasible. Simulations of star formation, for example, generally start with observed molecular cloud cores that are ~ 0.1 – 1.0 pc in size (Williams et al. 2000), while the stars that are the endpoints of the calculation are of order a solar radius ($\sim 10^{11}$ cm) in size, a dynamic range of $\sim 10^7$ in length.

Dynamic range is expensive for two distinct reasons. First, one requires enough resolution elements (cells for gridded codes, particles for gridless codes) to resolve both the largest and smallest structures in the problem. For a Eulerian code with no adaptivity, such as the widely used ZEUS package (Stone & Norman 1992a, 1992b; Stone et al. 1992), increasing the linear resolution of a calculation by a factor f requires increasing the number of cells in each dimension f , thereby increasing the total number of cells by a factor of f^N , where N is the number of dimensions. Lagrangian approaches, such as smoothed particle hydrodynamics (SPH; Gingold & Monaghan 1977; Lucy 1977), and adaptive Eulerian approaches, such as adaptive mesh refinement (AMR; Berger & Olinger 1984; Berger & Collela 1989; Bell et al. 1994), fare significantly better in this regard by following the mass and adding resolution elements only in regions of interest.

The second reason a large dynamic range is expensive is that the smallest and largest structures present in a problem often evolve on very disparate timescales, requiring a simulation to take an inordinate number of very small time steps. In explicit hydrodynamics codes this problem is embodied in the Courant condition (Richtmyer & Morton 1967), which requires that the time step be less than the signal-crossing time of a resolution element. Increasing the linear resolution by a factor f therefore generally requires multiplying the number of time steps by f as well. Again, adaptive methods that allow different time steps for different resolution elements (Bate et al. 1995) do somewhat better.

Even with the improvements in computational efficiency made possible by adaptivity, however, many interesting problems require more dynamic range than any code can handle. The time stepping constraint in particular is a significant barrier because, unlike additional cells, the additional time steps cannot easily be distributed on a parallel machine. Therefore simulators have introduced sinks: regions of a flow that accrete incoming material but that have no internal structure and therefore no requirements for high resolution in either time or space. Sinks provide a way to stop following collapse at a pre-chosen scale (hopefully) without damaging the rest of the calculation, thereby preventing the time step from grinding to zero and the number of resolution elements from running off to infinity.

Bate et al. (1995) introduced the technique of sink particles in SPH codes. Sink particles are absorbers that accrete other particles that approach within a certain distance and meet

various other criteria. The accretion radius sets the smallest scale that the simulation can resolve. The boundary pressure of a sink particle is determined by extrapolation from the particles around it. While this technique has proven extremely useful and has been adopted in numerous places, SPH codes are not appropriate for all problems. To date there is no widely used, well-tested, SPH code that includes magnetohydrodynamics or radiative transfer (although see Price & Monaghan 2004a, 2004b for a recent implementation of MHD with SPH that is still in the testing phase). In addition, SPH codes require significant artificial viscosity, which tends to cause artificial heating and smearing at shocks and interfaces. SPH is therefore of limited utility in problems where these features are important (Shapiro et al. 1996).

In contrast, Eulerian codes including MHD or radiative transfer are reasonably mature and, using Godunov schemes, require much less artificial viscosity (Truelove et al. 1998). Eulerian codes have included sinks in the form of sink cells (Boss & Black 1982). Similar to sink particles, sink cells allow mass to enter but not to leave, and their boundary pressures are found by extrapolation from neighboring cells. The disadvantage to this approach is that sink cells are fixed in the grid and are therefore inapplicable in cases where there are multiple accretors moving relative to one another (for example, a binary system) or where one does not know in advance where an accretion center will form (for example, in star formation in a turbulent medium).

In this paper we introduce a technique to embed a Lagrangian sink particle in a Eulerian code. This technique allows us to use Eulerian codes for cases where they are preferred while retaining the flexibility of a moving sink center. While previous work has combined nonfluid particles with Eulerian hydrodynamics (Kravtsov 2003), our approach is unique in that it allows the nonfluid particles to accrete from the gas and therefore truly act as sinks. In § 2 we introduce the Eulerian AMR code and describe the method we use to embed Lagrangian sink particles within it. In § 3 we describe tests that we have done to evaluate the accuracy of this method. In discussing our test of a sink particle in the context of a disk, in § 3.4.2 we estimate the standard viscosity parameter α (Shakura & Sunyaev 1973). We then consider a simple application of our technique in § 4: modeling the process of a flow changing from Bondi accretion to Bondi-Hoyle accretion as a result of an external shock. Finally, we discuss our conclusions in § 5.

2. COMPUTATIONAL METHODOLOGY

2.1. The Eulerian Code

For the calculations presented in this paper we use our three-dimensional AMR code. The code includes hydrodynamics, gravity, and radiative transfer, but we refer only to the first two components in this paper. The hydrodynamics module solves the Euler equations for a compressible, multifluid system,

$$\frac{\partial \rho^i}{\partial t} + \nabla \cdot (\rho^i \mathbf{v}) = 0, \quad (1)$$

$$\frac{\partial}{\partial t} (\rho^i \mathbf{v}) + \nabla \cdot (\rho^i \mathbf{v} \mathbf{v}) = -\nabla \sum_i P^i - \rho^i \nabla \phi, \quad (2)$$

$$\frac{\partial}{\partial t} (\rho^i e^i) + \nabla \cdot [(\rho^i e^i + P^i) \mathbf{v}] = \rho^i \mathbf{v} \cdot \nabla \phi, \quad (3)$$

where ρ^i , P^i , and e^i are the density, thermal pressure, and nongravitational specific energy for fluid i , and \mathbf{v} is the vector velocity (taken to be the same over all fluids). We determine the potential ϕ by solving the Poisson equation as described below. The code solves these equations using a conservative high-order Godunov scheme with an optimized approximate Riemann solver (Toro 1997). The algorithm is second-order accurate in both space and time for smooth flows, and it provides robust treatment of shocks and discontinuities.

The gravitational module solves the Poisson equation

$$\nabla^2 \phi = 4\pi G \sum_i \rho^i \quad (4)$$

on an adaptive grid hierarchy (as described below) to find the gravitational potential from a given density distribution. In each time step we compute the potential and then use it as a source term in the hydrodynamics equations as shown above (Truelove et al. 1998). The gravity module uses a multigrid iteration scheme to solve the linearized Poisson equation on each level of the adaptive hierarchy.

Each physics module operates within the AMR framework (Berger & Oliger 1984; Berger & Collela 1989; Bell et al. 1994). We discretize the problem domain onto a base, coarse level, denoted level 0. We dynamically create finer levels, numbered 1, 2, . . . , n , nested within that coarse level as needed. The nesting process is recursive, so each fine level may contain even finer levels, providing no theoretical upper limit to the maximum resolution. In practice, limits of computational resources require that we select a maximum level of refinement allowed for a given calculation. The process for a time step is similarly recursive: one advances level 0 through a single time step Δt_0 , then advances each subsequent level for the same amount of time. Each level has its own time step, and in general $\Delta t_{i+1} < \Delta t_i$, so after advancing level 0 we must advance level 1 through several steps of size Δt_1 , until it has advanced a total time Δt_0 as well. At that point we apply a synchronization procedure to guarantee conservation of mass, momentum, and energy across the boundary between levels 0 and 1. However, each time we advance level 1 through time Δt_1 , we must advance level 2 through several steps of size Δt_2 and so forth to the finest level present.

The sink particle framework we present below is not dependent on any of the details of the AMR framework and may be applied equally well to fixed-grid codes. Therefore, we suppress discussion of AMR details. However, we note here that we add a condition to our refinement criteria requiring that a sink particle's accretion zone (see § 2.4.2) always be refined to the highest allowable level for a given calculation. If the sink particle is moving, the refined patch will move with it. We also refine a buffer region around the accretion zone to guarantee that the accretion zone cannot move out of the refined area in the interval between recalculations of the grid. When we refer to cell spacings and time steps in what follows, the cell spacings are always those of the finest AMR level. In addition, for simplicity we also suppress all discussion of multifluid issues.

2.2. Creation of Sink Particles

We can either introduce sink particles in the initial conditions for a calculation or create them when necessary. Once we introduce a sink particle, we lose all knowledge of the flow in some region around it and assume that the gas within that

region will continue to collapse beyond the scale resolved by our simulation. We therefore wish to introduce sinks only when there is good physical reason to believe that continuing the calculation without the sink will give inaccurate results and that the gas in the vicinity of the sink is likely to continue collapsing past the scales resolved in our calculation.

Both of these conditions are met in cells that violate the Jeans criterion (Truelove et al. 1997, 1998):

$$\Delta x < J\lambda_J = J\sqrt{\frac{\pi c_s^2}{G\rho}}. \quad (5)$$

Here, J is a constant of order unity, λ_J is the Jeans length in a cell of length Δx , and c_s and ρ are the sound speed and density in the cell. Truelove et al. (1997) found that $J = 0.25$ is sufficient to prevent artificial fragmentation in most of the problems they considered. We can also write this as a condition on the density for fixed Δx ,

$$\rho_J < J^2 \frac{\pi c_s^2}{G\Delta x^2}. \quad (6)$$

If the density in any cell does exceed ρ_J with $J = 0.25$, we create in the center of that cell a sink particle with mass

$$m_{\text{sink}} = [\rho - \rho_J(0.25)]\Delta x^3, \quad (7)$$

so the density of the gas remaining in the cell is $\rho_J(0.25)$. We also transfer a proportional amount of momentum and energy from the gas to the sink particle.

Truelove et al. (1997) have shown that continuing a calculation that has violated the Jeans criterion will lead to artificial fragmentation. By creating sink particles in cells that violate the Jeans condition we prevent this from occurring. Also, since the density in the cell must have been lower in the time step before the sink particle has appeared, it follows from continuity that $\nabla \cdot \mathbf{v} < 0$ in that cell. The fact that the cell violates the Jeans criterion indicates that its self-gravity has begun to become important. It is therefore likely that the gas in that cell will continue collapsing indefinitely. Thus, creating sinks in cells that violate the Jeans criterion meets the conditions that we create sinks only when necessary and only when we are confident that indefinite collapse is a valid approximation to the true behavior of the system. Note that these criteria are roughly analogous to those used by SPH codes (Bate et al. 1995; Bromm et al. 2002) to create sink particles: the particle must be in a region of converging flow that is gravitationally bound.

There is one cautionary note: for a cell to violate the Jeans criterion, its mass must be at least

$$m_{\text{cell}} > \rho\Delta x^3 = \pi^{3/2}J^3 \frac{c_s^3}{(G^3\rho)^{1/2}}, \quad (8)$$

where in the last step we eliminate Δx using equation (5). In comparison, the maximum mass possible for a stable, self-gravitating, isothermal object is the Bonnor-Ebert mass (Ebert 1955; Bonnor 1956), $m_{\text{BE}} = 1.18c_s^3/(G^3\rho)^{1/2}$. A Jeans-violating cell (for $J = 0.25$) therefore has a minimum mass of

$$m_{\text{cell}} > \frac{\pi^{3/2}}{1.18}J^3m_{\text{BE}} \approx 0.07m_{\text{BE}}. \quad (9)$$

While the addition of neighboring cells will likely raise the total mass in the region above the Bonnor-Ebert mass, it is therefore still possible to create a sink particle in a region that is stable against collapse. However, a newly created sink particle has a very low accretion rate (see § 2.4.1). If the region is truly stable, gas will not continue to accrete onto the sink and the sink's mass will remain very low. We thus have an ex post facto check on the validity of our sink particle creation method. If such a situation occurs, there is no way to do the calculation at the chosen level of resolution without violating the Jeans condition or creating a suspiciously small sink particle. The only choice is to redo the calculation at a higher resolution.

2.3. Merging Sink Particles

In a region of gravitational collapse, we often find that in a single time step a block of contiguous cells increases in density so that they all violate the Jeans condition and create sink particles. When this happens we wish to merge these particles, since allowing them to remain unmerged and possibly separate would risk a solution that contains resolution-dependent artificial fragmentation. In addition, when a sink particle is first created, gas usually continues to flow into the sink particle's host cell and its neighbors. Since a newly created sink particle's mass and rate of gas accretion (see § 2.4.1) are very low, after each time step or two these cells will once again violate the Jeans condition. As a result, the code will create more sink particles, which ought to be merged into the already existing one. (This process generally continues until the sink particle is massive enough that its gas accretion rate prevents the density in the host cell from rising above the Jeans density.)

To deal with this phenomenon, at the end of each time step we group all the sink particles present in the calculation using a friends-of-friends (FOF) algorithm (Davis et al. 1985) with a linking length equal to the radius of the accretion zone (see § 2.4.2). We then merge all groups of particles that the FOF algorithm finds. We replace the merged group by a single particle at the center of mass of the group, and we add all particle quantities conservatively. Later on in the calculation, if two independently formed objects pass near one another for a short period, we may not wish to merge them. In this case we can temporarily reduce the radius of the accretion zone (which for technical reasons must be no larger than the merger radius) during the close passage and increase it again once the objects are sufficiently far apart. Since sink particles moving under gravity will spend very little time during a close passage, this will have a negligible effect on the overall accretion rate or final mass. This adjustment to the merger radius can be handled either manually or by an automated algorithm.

2.4. Accretion onto Sink Particles

2.4.1. The Accretion Rate

Once it appears, a sink particle accretes gas from the surrounding cells. The gas in which the sink particle is embedded continues to evolve according to the Euler equations. Setting the accretion rate is therefore critical in cases where the flow onto the sink particle is subsonic, because the rate at which mass flows from Eulerian cells to the pressureless particle determines the amount of back-pressure opposing the accretion flow. The accretion formalism thus serves a function analogous to the extrapolation procedure used to find boundary pressure in SPH sink particle or Eulerian sink cell formalisms.

We characterize the relative importance of pressure versus gravity via the particle's Bondi-Hoyle radius (Bondi 1952),

$$r_{\text{BH}} = \frac{GM}{v_{\infty}^2 + c_{\infty}^2}, \quad (10)$$

where M is the particle's mass and v_{∞} and c_{∞} are the velocity and sound speed of the gas far from the sink particle.

In the limit where the Bondi-Hoyle radius is much larger than a cell spacing, the choice of accretion rate and hence the pressure is irrelevant because the flow is supersonic near the sink particle. Even if the accretion rate is set too small, gas will flow into the sink particle's host cell until its density is high enough to violate the Jeans condition. Once that happens, the code will create new sink particles that will immediately merge with the existing particle, thus setting an effective accretion rate higher than that given by the formula. In the opposite limit, $r_{\text{BH}} \ll \Delta x$, the sink particle is simply a point mass moving through a uniform gas. Its gravity is relevant only on scales smaller than we resolve in the simulation. This is just the classical Bondi-Hoyle-Lyttleton problem, which has analytic solutions in the limits $v_{\infty} \ll c_{\infty}$ and $v_{\infty} \gg c_{\infty}$ (Hoyle & Lyttleton 1939, 1940a, 1940b, 1940c; Bondi 1952). In between these two limits, Ruffert (1994a) and Ruffert & Arnett (1994) give the approximate formula

$$\dot{M} = 4\pi\rho_{\infty}G^2M^2 \left[\frac{\lambda^2 c_{\infty}^2 + v_{\infty}^2}{(c_{\infty}^2 + v_{\infty}^2)^4} \right]^{1/2} = 4\pi\rho_{\infty}r_{\text{BH}}^2 (\lambda^2 c_{\infty}^2 + v_{\infty}^2)^{1/2}. \quad (11)$$

Here λ is a constant of order unity that depends on the equation of state of the gas. For an isothermal gas, $\lambda = e^{3/2}/4 \approx 1.120$, and we use that value throughout this work.

For a real simulation with complex, turbulent flows, there is no obvious way to choose v_{∞} . For symmetric accretion flows, the gas in the sink particle's host cell is generally comoving with the background at large distances. Since we have no better alternative, we therefore take v_{∞} to be the relative velocity of the sink particle and the gas in its host cell. Similarly, we use sound speed in the host cell for c_{∞} . The choice of ρ_{∞} requires more discussion. We let ourselves be guided by the behavior we expect when simulating simple Bondi accretion. In that case, the density profile, which we denote by $\alpha(x) \equiv \rho(x)/\rho_{\infty}$, is the solution to a pair of coupled nonlinear ordinary differential equations (Bondi 1952). Here $x \equiv r/r_{\text{BH}}$ is the dimensionless radius. The density in the central cell should be $\sim \rho_{\infty}\alpha(\Delta x/r_{\text{BH}})$. We therefore set

$$\rho_{\infty} = \frac{\bar{\rho}}{\alpha(1.2\Delta x/r_{\text{BH}})}, \quad (12)$$

where $\bar{\rho}$ is the weighted mean density in the accretion region (see § 2.4.2). In the limit $\Delta x \gg r_{\text{BH}}$, $\alpha(1.2\Delta x/r_{\text{BH}}) \rightarrow 1$, so we recover the correct behavior for this case. We inserted the factor of 1.2 because we found that it gave improved results in the intermediate range $\Delta x \sim r_{\text{BH}}$ (see § 3.2). The accretion rate is reduced in the presence of rotation, as described in § 2.4.2.

As a final note, Ruffert (1994b) gives a somewhat more complex formula than equation (11) that provides a slightly better fit to accretion rates found in their numerical simulations. However, even that formula is off by as much as 60% for isothermal flows with intermediate Mach numbers (Ruffert

1996), and thus we decided against the additional complexity involved in implementing it. Fortunately, as we discuss in § 3.3, errors in the accretion formula tend to be self-correcting in an actual simulation, and thus the details of the accretion formula are not critical.

2.4.2. The Accretion Zone

We wish the accretion rate to change smoothly as the sink particle moves across cell boundaries. Therefore we define an accretion zone around each sink particle. We set the accretion rate based on average properties in the accretion zone, and when the sink particle accretes mass, it does so from all cells within the accretion zone. In choosing the size of the accretion zone, there are two competing factors. Since the solution within the accretion zone is artificially affected by the accretion process, the larger the accretion zone, the larger the region in which we give up on the accuracy of the solution. However, in order to compute accurately the rate at which mass enters the accretion zone, we must have adequate resolution on its boundary. In addition, the size of the accretion zone will determine our ability to resolve anisotropies in the accretion flow. We define the accretion region as all the cells within a radius r_{acc} of the sink particle's host cell. Based on experiments with different sizes, we adopt a value $r_{\text{acc}} = 4\Delta x$ throughout this work. However, our implementation of the sink particle algorithm leaves the radius of the accretion region as a free parameter to be set at run time.

In cases where the particle's Bondi-Hoyle radius is smaller than the accretion zone, it would be incorrect to set the accretion rate based on a uniform average of all cells, however. Therefore we define an accretion kernel with radius

$$r_K = \begin{cases} \Delta x/4 & r_{\text{BH}} < \Delta x/4, \\ r_{\text{BH}} & \Delta x/4 \leq r_{\text{BH}} \leq r_{\text{acc}}/2, \\ r_{\text{acc}}/2 & r_{\text{BH}} > r_{\text{acc}}/2. \end{cases} \quad (13)$$

Within the accretion zone we assign each cell a weight

$$w \propto \exp(-r^2/r_K^2), \quad (14)$$

where r is the distance from the cell center to the sink particle; cells outside the accretion zone have a weight of 0. The minimum value of $\Delta x/4$ for r_K ensures that the accretion changes smoothly as the particle crosses cell boundaries even when r_{BH} is small. The maximum value of $r_{\text{acc}}/2$ ensures that cells at the edge of the accretion region have little weight and thus there are no sudden changes in the accretion rate as cells enter or leave the accretion region. Once we have assigned weights to all cells in the accretion zone, we use a weighted average to set $\bar{\rho}$ in equation (12). We then compute the accretion rate for this time step from equation (11). Note that this procedure becomes undefined if the accretion zones of multiple sink particles overlap; for this reason we require that the sink particle merger radius always be greater than or equal to the accretion zone radius.

Thus far our algorithm has not included the effects of angular momentum, which may substantially reduce the accretion rate relative to the spherically symmetric case. Furthermore, in a rotating flow, low angular momentum gas along the polar axis accretes more easily than high angular momentum gas in the equatorial plane. Thus, it would be incorrect to use an accretion algorithm that accretes equally quickly from all cells regardless of their place in the rotating flow. We

therefore choose a strategy that will both reduce the accretion rate in the presence of rotation and allow accretion to occur anisotropically from within the accretion zone.

To include rotation, we first divide the mass to be transferred to the sink particle (computed via eq. [11]) among the cells in the accretion zone so that each cell contributes an amount of mass proportional to its weight (computed via eq. [14]). We then divide each cell into 8^3 identical point particles arranged in a uniform grid throughout the cell, each with $1/8^3$ the mass, momentum, and energy of the cell. For each point particle we compute its distance of closest approach to the sink particle if it were to travel on a purely ballistic trajectory while the sink particle moved at constant velocity. For a point particle with specific angular momentum j_{sp} and specific energy e_{sp} (kinetic plus gravitational) in the sink particle's rest frame, this distance is

$$r_{\min} = -\frac{GM}{2e_{\text{sp}}} \left[1 - \sqrt{1 + \frac{2j_{\text{sp}}e_{\text{sp}}}{(GM)^2}} \right]. \quad (15)$$

If the point particle is not bound to the sink particle, so that $e_{\text{sp}} > 0$, we take $r_{\min} = \infty$. We do not want the sink particle to accrete material that has too much angular momentum to reach its ‘‘surface.’’ We therefore count up the number n of point particles for which $r_{\min} > \Delta x/4$ and reduce the amount of mass to be accreted from that cell by a factor $n/8^3$. (The factor of 4 in r_{\min} is to ensure that the sink particle has an effective size smaller than the size of a cell; experimentation with different values from 0.1 to 0.5 produced no noticeable differences in behavior). For the host cell, since we cannot compute a meaningful specific angular momentum, we set n equal to the maximum of the values of n in the cells bordering the host cell if $r_K \geq \Delta x/4$, or $n = 0$ (i.e., uninhibited accretion) if $r_K < \Delta x/4$. Once this is done we subtract the appropriate amount of mass from each cell and add it to the sink particle. To ensure stability, we also set an absolute cap that no more than 25% of the mass may be removed from a cell in any single time step.

Next we must compute the amount of linear momentum the sink particle accretes. Since it represents an object far smaller than the grid size in the calculation, it should accrete negligible angular momentum and exert no torques on the gas. We therefore divide each cell's momentum, taken in the sink particle's rest frame, into components parallel and transverse to the radial vector connecting that cell to the sink particle. We reduce the cell's radial momentum by a factor equal to the fraction of the cell's mass that we have accreted, while we leave its transverse momentum unchanged. Thus, accretion preserves the *radial velocity* and the *angular momentum* of the gas. To ensure linear momentum conservation, we change the sink particle's momentum by an amount equal and opposite to the total change in the momentum of the gas cells. Finally, we compute the new total energy of each gas cell by keeping the cell's specific thermal energy constant while computing a new kinetic energy based on its new density and momentum.

We find that this accretion procedure conserves mass, momentum, and angular momentum to machine precision. However, for more discussion of angular momentum conservation, see § 3.4.

2.5. Motion of Sink Particles

In each time step we update the position of each sink particle based on its current momentum, and we modify its momentum

through accretion (as described in § 2.4.1) and through gravity. For reasons of algorithmic speed, we handle the position and momentum update of sink particles in two steps. First, we change the momenta of sink particles due to their gravitational interactions with the gas by an amount $F_{\text{gas-part}}\Delta t$, where Δt is the time step and $F_{\text{gas-part}}$ is the gas-particle gravitational force. To ensure accuracy, we constrain the time step to require that

$$\max(v_{\text{part}})\Delta t < C\Delta x, \quad (16)$$

where $\max(v_{\text{part}})$ is the largest particle velocity in the calculation, C is a constant of order unity (for our runs generally 0.5), and Δx is a cell spacing. This restriction is usually less stringent than the ordinary gas Courant condition. Particles have velocities comparable to the gas out of which they form, and the collapsing gas from which sink particles form is usually not the gas that has the highest velocity relative to the grid.

To compute the force on a particle, we use a Plummer law (Aarseth 1963),

$$\mathbf{F}_{\text{gas-part}} = -Gm \int \frac{\rho}{r^2 + \epsilon^2} \frac{\mathbf{r}}{r} dV, \quad (17)$$

where m is the mass of the particle, ρ is the gas density, \mathbf{r} is the vector from the sink particle to a given cell or particle, and ϵ is the softening length, which we leave as a parameter that may be set at run time. In general, the softening length should be smaller than the size of the accretion region, to ensure that softening does not alter the rate at which gas crosses its boundary. Choosing a smaller softening length, however, increases the maximum velocity the gas will attain within the accretion region as it falls onto a sink particle, which will in turn decrease the time step due to the Courant condition. For our default choice of $r_{\text{acc}} = 4\Delta x$, we therefore set a default value of $\epsilon = 2\Delta x$. Since the number of particles is generally small, we compute gas-particle forces via a direct sum. In computing the force between a particle and the gas, we treat all cells except the particle's host cell and its neighbors as point masses located at the cell center. To compute the gravitational force between a sink particle, its host cell, and its neighboring cells, we subdivide each cell into 8^3 identical point particles, as described in § 2.4.2. We set the force between the cell and the sink particle equal to the sum of the forces between the sink particle and the 8^3 point particles.

The second step is to update the positions and momenta of the particles, including the effects of particle-particle interactions. We handle this step separately because particles may occasionally pass close to one another, within a few cell spacings. When this happens, a time step chosen via equation (16) may allow the particles to change their separation by a significant amount in a single update. In this case, a simple first- or second-order position and momentum update will not provide an accurate integration of the particle orbits. However, since computing gas updates is far more expensive than computing particle updates, we do not wish to reduce our overall time step and compute more gas updates. Instead, we integrate the particles forward through a time Δt using a Bulirsch-Stoer method with an adaptive time step and error control (Press et al. 1992, p. 724). During this integration, we consider only particle-particle gravitational interactions, which, as with the particle-gas interactions, we compute via a direct sum. Unlike with particle-gas interactions, we do not use a softened force law for particle-particle interactions. We have tested this method

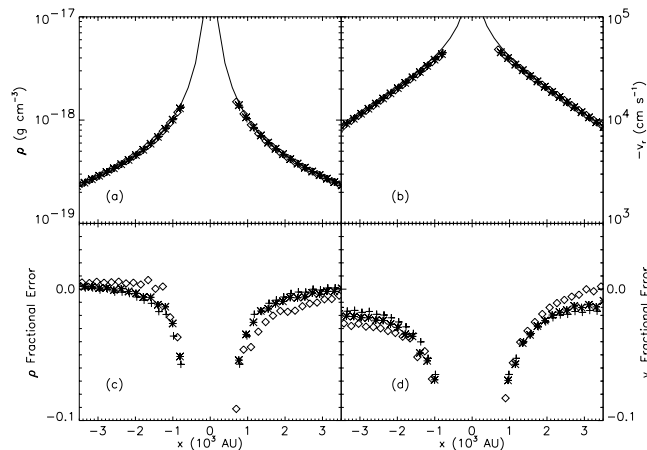


FIG. 1.—Results of our isothermal sphere test. (a, b) Density and velocity vs. radial distance in the sink particle’s rest frame. (c, d) Fractional error in density and velocity. All quantities are plotted along the direction of the sink particle’s motion. In the top panels, the analytic solution is the solid line. In all panels the crosses show values from the unadvected run, the asterisks show values from the run advected at Mach 0.5, and the diamonds show values from the run advected at Mach 2. The simulation data stop at the edge of the sink region. All the plots are at time $t = 2.0 \times 10^{12}$ s.

by placing two particles in an extremely eccentric orbit ($e = 0.998$), where the closest approach of the particles is $\sim \Delta x/100$ and the gas density and temperature are chosen so that accretion is negligible. In this test we found that after several orbits the semimajor axis was conserved to $\sim 1\%$ and the eccentricity to $\sim 0.1\%$.

3. TESTS OF THE METHODOLOGY

3.1. The Collapse of an Isothermal Sphere

We have tested this method against the analytic solution for the collapse of an isothermal sphere with a density 10% above the critical value (Shu 1977). We consider a $1 M_{\odot}$ sphere of molecular H and He mixed in the standard cosmic abundance (mean particle mass of $2.33 m_p$) with a sound speed of 0.18 km s^{-1} , appropriate for $\sim 10 \text{ K}$ gas. To avoid the singular initial configuration of the Shu solution, we set our initial density and velocity profile to their analytic values after a time $t = 1.3 \times 10^{12}$ s, when the expansion wave has propagated 2.4×10^{16} cm, or eight cells, from the origin. There are 64 cells in the radius of the sphere, and the sphere is motionless and centered on the origin. To ensure that there are no problems when the sink particle crosses cell boundaries, we ran two more tests with identical setups except that we gave the isothermal sphere an initial uniform velocity relative to the grid. In one test we used an advection velocity of half the sound speed, and in the other twice the sound speed.

The density and velocity profiles that we find from these tests are shown in Figure 1, and the mass of the sink particle versus time is shown in Figure 2. The calculation reproduces the analytic solution extremely well. Errors in the velocity and density profiles are a few percent in the cells adjacent to the accretion region, dropping rapidly to $\sim 1\%$ as one moves farther away. The advected cases show a small initial transient during which the accretion rate deviates from the theoretical value, but after a time of less than the sound-crossing time of the accretion region the flow settles into a steady state. Once this happens, the error in the accretion rate is $\sim 1\%$ in all three runs. There is also a slight asymmetry near the sink particle

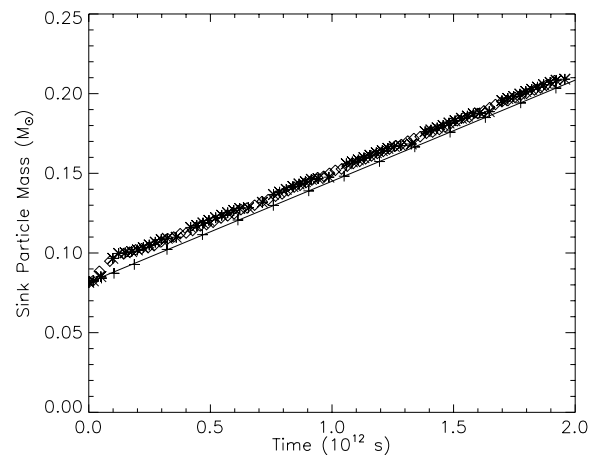


FIG. 2.—Mass of the sink particle vs. time. *Solid line*: Theoretical result. *Crosses*: Values from the unadvected run. *Asterisks*: Values from the run advected at Mach 0.5. *Diamonds*: Values from the run advected at Mach 2.

visible in the test advected at Mach 2. Relative to the unadvected and subsonically advected cases, there is a few percent higher density and lower velocity in the region trailing the sink particle (relative to its motion on the grid), and a difference of the opposite sign ahead of the sink particle. However, the mass flux into the accretion region seems to be fairly symmetric, since the errors in density and velocity are of similar magnitude and opposite sign. The asymmetry in the supersonically advected run does not have any noticeable effect on the time-averaged accretion rate, which differs from the other two runs and from the theoretical solution by only $\sim 1\%$.

3.2. Bondi Accretion

To test how well our sink particle formalism works in cases where our resolution is marginal, so that $r_{\text{BH}} \sim \Delta x$, we ran a series of simulations of simple Bondi accretion using a variety of values for $r_{\text{BH}}/\Delta x$. In each case we use a sphere of mixed H and He at 10 K with a radius of 1.21×10^{19} cm. For 32 cells across the radius of the sphere and a $1 M_{\odot}$ central star this gives $\Delta x = r_{\text{BH}}$. We initialize the density and velocity profile of the sphere to the analytic solution for the Bondi problem and allow the calculation to run until the accretion rate reaches steady state. We then repeat the calculation with different sink particle masses and compare the accretions rates to the theoretical predictions. The results are shown in Table 1.

As the table indicates, there can be a substantial error in the accretion rate when the Bondi radius of the sink particle is

TABLE 1
SIMULATED VERSUS THEORETICAL ACCRETION RATES FOR BONDI ACCRETION

M_{sink} (M_{\odot}) (1)	$r_{\text{B}}/\Delta x$ (2)	\dot{M}_{theor} ($M_{\odot} \text{ yr}^{-1}$) (3)	\dot{M}_{sim} ($M_{\odot} \text{ yr}^{-1}$) (4)	\dot{M} Error (5)
0.1.....	0.1	5.94×10^{-13}	6.01×10^{-13}	0.011
0.316.....	0.316	5.94×10^{-12}	5.90×10^{-12}	-0.006
1.0.....	1.0	5.94×10^{-11}	5.21×10^{-11}	-0.122
3.16.....	3.16	5.94×10^{-10}	4.49×10^{-10}	-0.244
10.0.....	10.0	5.94×10^{-9}	5.76×10^{-9}	-0.023

NOTES.—Col. (1): Sink particle mass. Col. (2): Ratio of the Bondi radius to a grid spacing. Cols. (3) and (4): Theoretical and simulated accretion rates. Col. (5): Fractional error in the simulated accretion rate.

comparable to a cell spacing, but the error drops off rapidly as the Bondi radius either increases or decreases. The peak error of $\sim 25\%$ seems to occur when the Bondi radius is approximately equal to the radius of the accretion zone, i.e., the case $r_B = 10^{1/2} \Delta x \approx r_{\text{acc}}$. For a factor of 10 difference in r_B and r_{acc} the error is $\sim 1\%$, while for even a factor of a few difference the error is only $\sim 10\%$. We experimented with several factors of order unity in equation (12), where we set ρ_∞ to see if we could decrease the error. We found that 1.2 gave the best results but that other factors between 0.5 and 2.0 increased the error in the accretion rate by no more than $\sim 10\%$.

This substantial error when $r_B \approx r_{\text{acc}}$ is not particularly surprising. The Bondi radius is the point at which the flow transitions from subsonic to supersonic. If that is equal to the accretion radius, inside which we are artificially altering the physics, then the transition is not well resolved and the density and velocity in cells near the sink particle will have substantial errors. These errors lead the code to set an incorrect accretion rate, which in turn compounds the problem by setting a back-pressure that is too large. When the flow near the sink particle is either subsonic or supersonic, this problem does not occur and the errors are far smaller. Nonetheless, this provides an important caveat to our method: one ought not use it in cases where the flow at the sink particle is transitioning from subsonic to supersonic.

3.3. Bondi-Hoyle Accretion

We tested the ability of our sink particle method to handle accretion from a moving medium by simulating Bondi-Hoyle accretion. We place a $1 M_\odot$ sink particle in an initially uniform gas, composed of a standard interstellar mix of H and He (mean particle mass of $2.33 m_p$) at 10 K, with a density of $10^{-25} \text{ g cm}^{-3}$. We performed two versions of this simulation. In one, the sink particle is initially stationary with respect to the computational grid, and the gas flows past it at Mach 3; we apply inflow boundary conditions in the upstream direction and outflow boundary conditions in the downstream direction. In the second version, the gas is initially at rest with respect to the grid, and the sink particle moves at Mach 3 at an angle of 30° relative to the x -axis, in the xy -plane. In this case, we used symmetry boundary conditions. In both cases the resolution of the finest cells in the calculation is $7.4 \times 10^{14} \text{ cm} = r_{\text{BH}}/50$, where r_{BH} is as defined by equation (10).

Using equation (11), the expected accretion rate is $\dot{M}_{\text{BH}} = 1.7 \times 10^{-12} M_\odot \text{ yr}^{-1}$. However, this simulation is in a regime where the interpolation formula works poorly. Ruffert (1996) performed a simulation very similar to ours (the run labeled GS, which has a small accretor, gas with polytropic index $\gamma = 1.01$, $\mathcal{M} = 3$) and found an accretion rate of $2.0 \times 10^{-12} M_\odot \text{ yr}^{-1} = 1.17 \dot{M}_{\text{BH}}$, where we have scaled their dimensionless result to our dimensional parameters. Ruffert (1996) also found that both the accretion rate and the flow pattern are time dependent and that the flow pattern shows substantial deviations from axial symmetry on scales comparable to r_{BH} . The mechanism for disrupting steady, symmetric flow is not fully understood, but Foglizzo & Ruffert (1999) suggest Rayleigh-Taylor and Kelvin-Helmholtz instabilities at the shock front as the probable cause.

Figure 3 shows the system near the end of our simulations. As expected, the flow is time dependent and unstable, with no axial symmetry. One can clearly see the gravitational focusing of streamlines into a shock behind the accretor that is characteristic of Bondi-Hoyle accretion. Since the exact morphology is a product of instabilities, we do not expect the two simu-

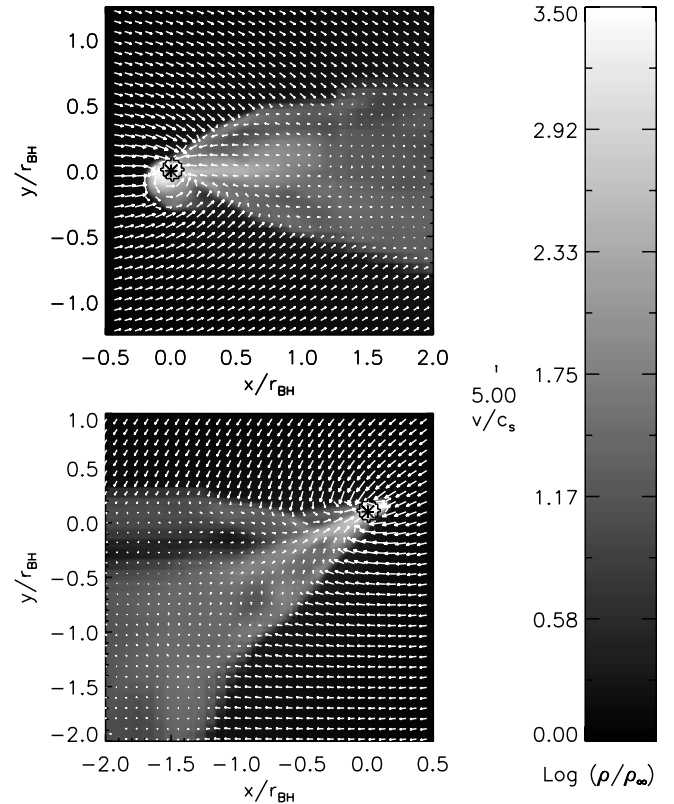


FIG. 3.—Density and velocity fields in the xy -plane at $t \approx 12t_{\text{BH}}$. *Top*: Run with the sink particle at rest. *Bottom*: Run with the sink particle moving relative to the grid. The bottom plot is in a coordinate system comoving with the sink particle. The black borders indicate the boundary of the accretion zone. [See the electronic edition of the Journal for a color version of this figure.]

lations to have identical appearances. However, the overall structure of the flow appears to be very similar in the two simulations. Both runs show clear Mach cones, with similar opening angles, similar densities, and similar velocities.

Figure 4 shows the accretion rates as a function of time. In both cases the accretion rate requires several Bondi-Hoyle times, defined by $t_{\text{BH}} \equiv r_{\text{BH}}/c_s = 6.35 \times 10^4 \text{ yr}$, to reach an equilibrium value, and even then it shows substantial fluctuations. In the run with the sink particle initially at rest, the accretion rate starts increasing sooner, but later on the other run passes it and shows a somewhat higher accretion rate. In both cases, the accretion rate appears to reach equilibrium after $\sim 6t_{\text{BH}}$; the average accretion rate after that point is $1.95 \times 10^{-12} M_\odot \text{ yr}^{-1} = 1.15 \dot{M}_{\text{BH}}$ in the run with the stationary sink particle and $2.14 \times 10^{-12} M_\odot \text{ yr}^{-1} = 1.26 \dot{M}_{\text{BH}}$ in the run with the moving sink particle. The former agrees with the results of Ruffert (1996) to 2.1%, and the latter to 7.8%. In both cases the difference between the mean accretion rate we find and the Ruffert (1996) result is smaller than the fluctuations in the accretion rate (which are typically a few tens of percent), so the agreement is good.

This test demonstrates that our method produces correct results for nonsymmetric, time-dependent flows. It also illustrates an important point regarding our accretion formula, equation (11): the method is self-correcting, and thus the exact details of the accretion formula do not make much difference as long as the Bondi-Hoyle radius is well resolved. The formula uses our best guesses for v_∞ and ρ_∞ based on the characteristics of the flow; however, it would be unreasonable to expect our method to correctly guess these values to the level of

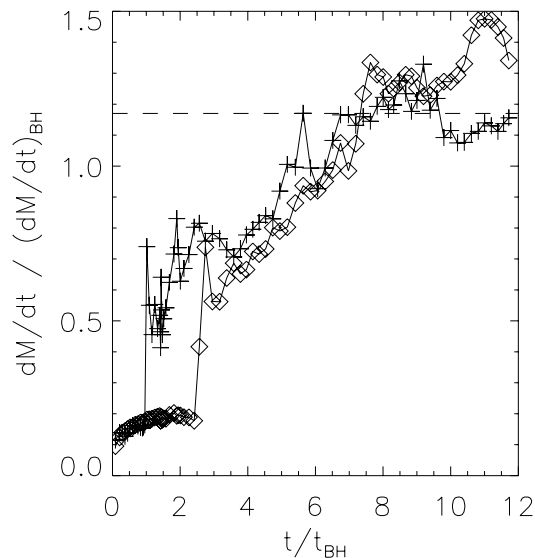


FIG. 4.—Accretion rate as a function of time in our simulations of Bondi-Hoyle accretion. The points are sampled at intervals of $\sim t_{\text{BH}}/5$. Time is plotted in units of $t_{\text{BH}} = 6.35 \times 10^4$ yr, and accretion rate is plotted in units of $\dot{M}_{\text{BH}} = 1.7 \times 10^{-12} M_{\odot} \text{ yr}^{-1}$. *Crosses*: Simulation with the sink particle stationary. *Diamonds*: Simulation with the sink particle moving. *Dashed horizontal line*: Accretion rate predicted by Ruffert (1996).

accuracy necessary for the formula to reproduce the Ruffert (1996) result. Even if our guesses for v_{∞} and ρ_{∞} were perfectly accurate, equation (11) would still predict an accretion rate 15% lower than Ruffert (1996) found and we find in our simulations. Instead, the accretion rate is ultimately dictated by the rate at which the ordinary, unaltered hydrodynamics of our simulation brings gas into the accretion region. If equation (11) sets an accretion rate that is too low, gas will enter the region faster than it is removed by accretion, so our guess for ρ_{∞} will increase and the sink particle will consume gas more quickly. The opposite effect happens if equation (11) dictates that gas be removed too quickly. Thus, our accretion rate is self-correcting.

3.4. Rotating Flows

3.4.1. Sink Particle Results

Finally, to ensure that our method does not cause artificial accretion or angular momentum transport, we tested the evolution of a disk around a sink particle. We place a $1 M_{\odot}$ sink particle at the center of a thin 10 K gas disk with a radius $r_0 = 2 \times 10^{15}$ cm and a power-law surface density profile $\Sigma = \Sigma_0 (r_0/r)^{k_{\rho}}$, with $\Sigma_0 = 0.1 \text{ g cm}^{-2}$ and $k_{\rho} = 1$. The grid is chosen so that the finest cells are 2.1×10^{13} cm in length. The disk is in Keplerian rotation, and so the surface density should not change with time. We simulated the evolution of the disk for at least 50 orbital periods at the edge of the accretion region. We ran two versions of the test, one with the sink particle at rest relative to the grid, and one with the disk and sink particle advected across the grid at Mach 10 at an angle of $(\theta, \phi) = (30^{\circ}, 30^{\circ})$.

We also attempted a run with the disk advected at Mach 100 but found that the accumulation of truncation errors caused by the rapid motion of the disk across the grid leads to distortions in the disk shape after 30–40 orbits. The distortions are small near the accretion region and larger farther away, and do not appear to be related to the sink particle; this instead appears to

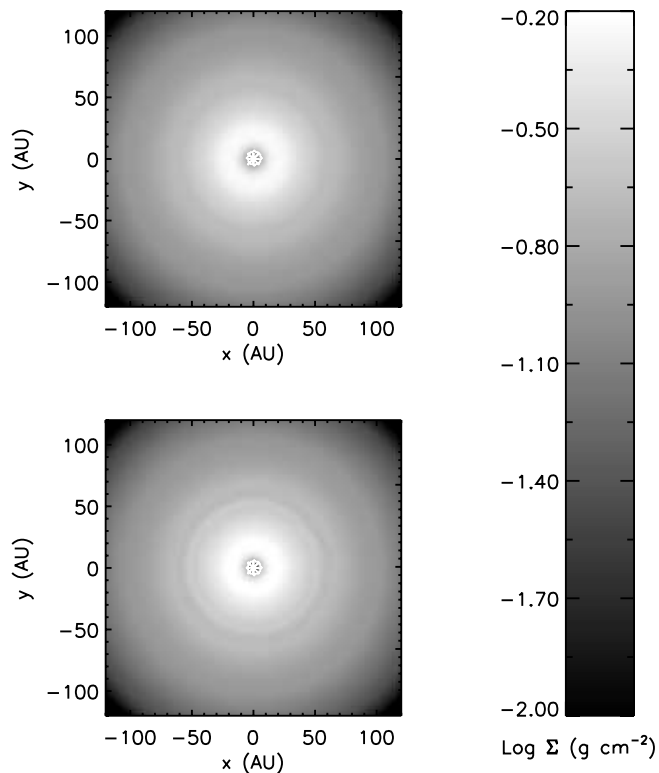


FIG. 5.—Surface density of our simulated disks after a time of ≈ 50 orbital periods at the edge of the accretion zone, $r = 5.55$ AU. *Top*: Run with the sink particle stationary on the grid. *Bottom*: Run with the sink particle advected across the grid. The white outlines in the center of each plot show the accretion zone. In the bottom panel, x - and y -coordinates are measured relative to the sink particle's position. [See the electronic edition of the *Journal* for a color version of this figure.]

be a limitation in the ability of our hydrodynamic code to handle extremely supersonic advection. Thus, we do not discuss that test further.

In analyzing this test, it is crucial to separate the effects of the sink particle from those of ordinary numerical viscosity. As one approaches the sink particle, the circles in which the gas is flowing are resolved by fewer and fewer cells. Numerical viscosity therefore becomes significant and will cause angular momentum transport. This effect causes evacuation of the gas in the disk near the sink particle, with some of the gas falling toward the center and the rest pushed outward. Nelson et al. (2000) and Okamoto et al. (2003) report an analogous phenomenon in SPH calculations, as do Kuznetsov et al. (1999) in a three-dimensional Eulerian code gridded inhomogeneously in spherical coordinates. To disentangle this effect from possible artificial angular momentum transport due to the sink particle, we ran a third simulation in which we disabled accretion onto the sink particle; we only performed this run with the sink particle at rest relative to the grid, not with an advected sink particle. Because the sink is far more massive than the disk, it does not move significantly during the calculation. Thus, the sink particle's sole effect in this run is to impose a point gravitational potential.

First, we compare the advected and unadvected runs with accretion turned on to ensure that there are no ill effects from motion of the sink particle across the grid. Figure 5 shows the surface density of the two runs at similar times. One can see in both images that the disks have remained circular and that there is no visible distortion near the sink particle. Figure 6 compares

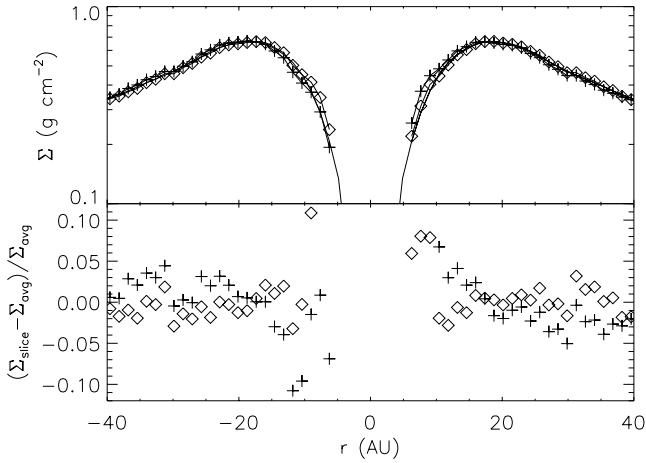


FIG. 6.—*Top*: Azimuthally averaged surface density (unmarked line), surface density sliced along the sink particle’s direction of motion (line with crosses), and surface density sliced transverse to the direction of motion (line with diamonds) for our run with the sink particle advected across the grid. *Bottom*: Fractional difference between the slice and the azimuthal average. The data for the slices end at the edge of the accretion zone. The distances r are measured relative to the position of the sink particle. The plots are for a time of ≈ 50 orbital periods at the edge of the accretion zone, $r = 5.55$ AU.

the azimuthally averaged surface density and slices of surface density in the directions along and transverse to the sink particle’s direction of motion for the run with advection. The difference between the surface density slices is a measure of the asymmetry induced by motion of the sink particle. As the plots show, the asymmetry can be tens of percent in cells adjacent to the accretion zone but quickly drops to only a few percent as one moves farther away. The difference between the slice and the azimuthal average is not substantially larger along the advection direction than transverse to it. Thus, a moving sink particle does not appear to induce substantial asymmetries.

Second, we compare the runs with and without accretion, with the sink particle at rest relative to the grid, to determine whether the sink particle has caused artificial angular momentum transport. Figure 7 shows the surface density versus radius at nearly identical times in our three calculations. In the simulation without accretion, there is a clear dip in the surface density between about 5 and 20 AU. Some of this gas has fallen into an unresolved hydrostatic object extending a few cells from the origin, leading to a density enhancement there. The rest has been pushed out farther into the disk, leading to the alternating enhanced and diminished density between 45 and 70 AU. Examination of surface density profiles at other times shows that this phenomenon is a wave moving out from the origin, a result of material being pushed away from the sink particle by numerical viscosity.

In both calculations with sink particle accretion, the surface density matches the initial surface density well except within about 20 AU of the sink particle, where it falls sharply. In these cases, the gas being evacuated from the low-resolution region around the sink particle has mostly been accreted. There is a slight density enhancement just outside the evacuated region, but it is smaller both in magnitude and in extent than in the calculation with no accretion. Similarly, the wave caused by material pushed outward from the origin is far smaller in the simulations with accretion. In the calculations with and without accretion the evacuated region is approximately the same size. The small decrease in density at radii greater than 80 AU apparent in all the simulations is a result of the pressure

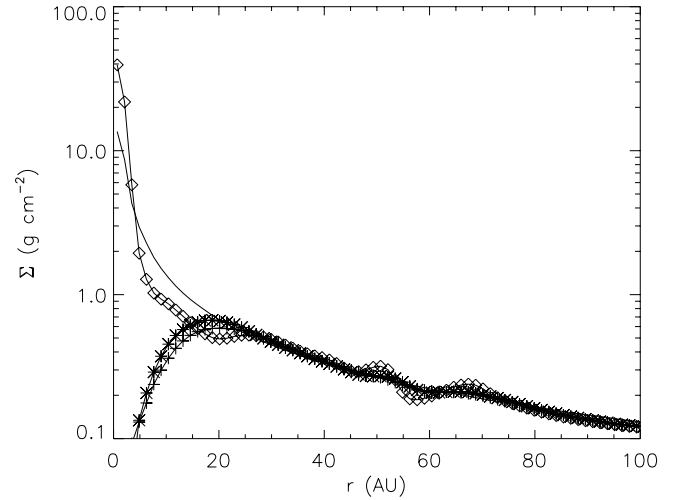


FIG. 7.—Azimuthally averaged surface density vs. radius after 50 orbital periods at the edge of the accretion region, $r = 5.55$ AU, in simulations of disk evolution around a sink particle. *Line marked with diamonds*: Run without sink particle accretion. *Line marked with crosses*: Run with accretion where the sink particle is stationary with respect to the grid. *Line marked with asterisks*: Run with the sink particle advected relative to the grid. *Unmarked line*: Initial surface density profile. The interval in radius between data points is one cell.

boundary conditions on the disk and is unrelated to the sink particle.

Figure 8 shows the size of the evacuated region of the disk, r_{evac} , versus time. We define the edge of the evacuated region as the smallest radius for which the surface density is 90% or more of the initial surface density at that radius. In the simulation without accretion, we exclude the inner four cells where the hydrostatic gas has accumulated. As the plot shows, the radius of the evacuated region is slightly less in the runs with sink accretion than without. This indicates that the sink particle is not causing enhanced accretion or angular momentum transport. The sink particle may actually lead to better results by preventing material that is artificially pushed outward by numerical viscosity from escaping the accretion region and affecting the rest of the calculation.

The size of the evacuated region is generally smaller by a cell or two in the run advected relative to the grid than in the

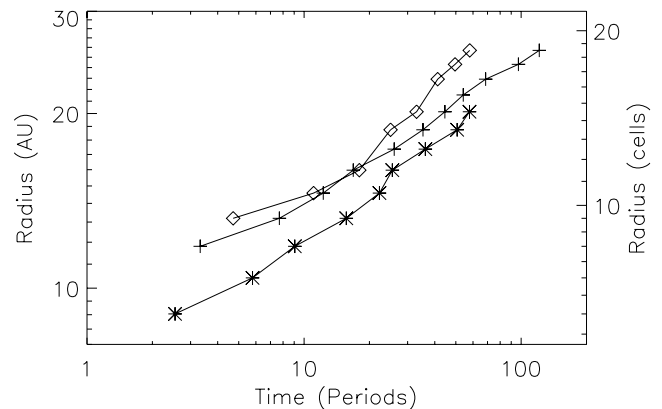


FIG. 8.—Radius of the evacuated region vs. time in simulations of disk evolution around a sink particle. *Line marked with diamonds*: Run without sink particle accretion. *Line marked with crosses*: Run with accretion that is stationary relative to the grid. *Line marked with asterisks*: Run with the sink particle advected relative to the grid.

run that is stationary with respect to the grid. However, the slope of the $\log r_{\text{evac}} - \log t$ relation is similar in the two runs. The difference in the evacuation radius is therefore most likely a result of advection causing an initial transient in the accretion rate, similar to the phenomenon shown in Figure 2. Once the transient passes, the evacuation radius moves outward in the same manner in both the advected and unadvected runs.

3.4.2. Estimating α Due to Numerical Viscosity

As a side note, in the calculation with accretion, a power-law fit to the radius of the evacuated region versus time gives

$$\frac{r_{\text{evac}}}{\Delta x} = 6.1 \left[\frac{\Omega(r_{\text{acc}})t}{2\pi} \right]^{0.23}, \quad (18)$$

where $\Omega(r_{\text{acc}})$ is the angular velocity of the disk at the edge of the accretion region; recall that we use $r_{\text{acc}} = 4\Delta x$ in this work. We can interpret this relation as a resolution requirement for the number of cells $r_{\text{evac}}/\Delta x$ as a function of total run time t that we need to ensure that numerical viscosity does not affect structures at a specified distance r_{evac} from the sink particle. We can also compute an approximate viscosity parameter α for this effect. For an isothermal Keplerian disk orbiting around an object of mass M , the accretion timescale at a distance r from the central object is

$$t_{\text{acc}} \approx \frac{r^2}{\nu}, \quad (19)$$

where ν is the kinematic viscosity (Lynden-Bell & Pringle 1974). The standard α -parameter is defined (Shakura & Sunyaev 1973) so that

$$\nu = \frac{\alpha c_s^2}{|r(d\Omega/dr)|} = \frac{\alpha c_s^2}{(3/2)\sqrt{GM/r^3}}, \quad (20)$$

where c_s is the sound speed in the disk and Ω is the disk angular velocity. Eliminating ν ,

$$\alpha \approx \frac{3}{2} \frac{(GMr)^{1/2}}{t_{\text{acc}} c_s^2} = \frac{3}{2} \frac{GM}{rc_s^2} \frac{1}{\Omega t_{\text{acc}}}. \quad (21)$$

We can identify the evacuation radius and time of equation (18) with the accretion radius and timescale of equation (21). Doing so, we find that the numerical viscosity as a function of radius is

$$\alpha \approx 78 \frac{GM}{rc_s^2} \left(\frac{r}{\Delta x} \right)^{-2.85} = 78 \frac{r_B}{\Delta x} \left(\frac{r}{\Delta x} \right)^{-3.85}, \quad (22)$$

where $r_B \equiv GM/c_s^2$ is the standard Bondi radius. Note that the exponent of the relation for α is -2.85 rather than $-4.35 = -1/0.23$ because $\Omega \propto r^{-3/2}$. To give some feel for the consequences of this relation, for $r_B/\Delta x = 10$ the numerical viscosity drops to $\alpha = 0.01$ at $r = 18.6\Delta x$.

We also computed an effective α using a method introduced by Artymowicz & Lubow (1994) for SPH simulations. We simulated an initially thin (two cells wide) ring of material orbiting around a sink particle of mass M_\odot . The temperature in the ring is 10 K, and the mean molecular mass is $2.33m_p$. As the ring orbits, numerical viscosity causes it to widen. Pringle (1981) has analytically solved the problem of an initially thin ring of pressureless material spreading due to viscosity as it

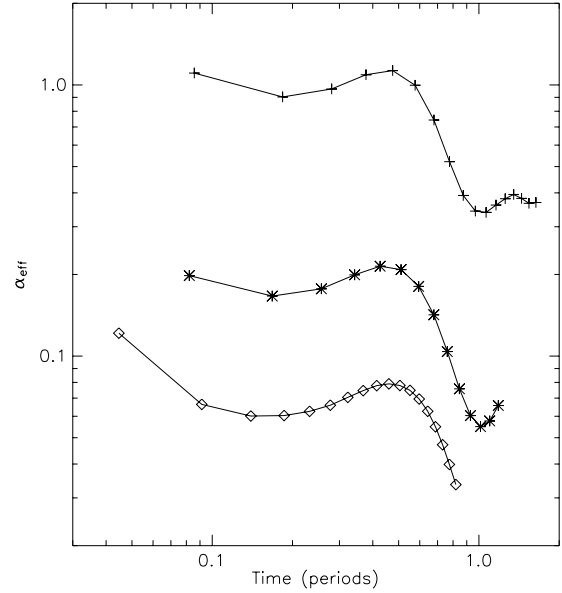


FIG. 9.—Best-fit value of α vs. time for our ring test. The curves marked with crosses, asterisks, and diamonds are for $r/\Delta x = 20, 40,$ and $60,$ respectively. The x -axis measures time in units of orbital periods at the initial radius of the ring.

orbits around a point mass. He shows that for a ring of initial mass m , radius R_0 , and viscosity ν , the surface density as a function of time is given by

$$\Sigma(x, \tau) = \frac{m}{\pi R_0^2} \tau^{-1} x^{-1/4} \exp\left(-\frac{1+x^2}{\tau}\right) I_{1/4}\left(\frac{2x}{\tau}\right), \quad (23)$$

where $x = R/R_0$ and $\tau = 12\nu t/R_0^2$ are the dimensionless radius and time and $I_{1/4}$ is the modified Bessel function of order $1/4$. We simulated the evolution of a ring, and at a series of times during its evolution we fitted the surface density profile to the form given by equation (23) to obtain a best-fit value for ν , from which we infer a value of α using equation (20). We repeated this procedure for rings with radii of 20, 40, and 60 cells, for cells 2.1×10^{13} cm = 1.85 AU across.

Figure 9 shows the value of α that we infer as a function of time for each of our test runs. This is not constant; instead, it varies in a curve that is the same for each ring radius. The variation may be due to the changing numerical viscosity that different parcels of gas experience as the ring spreads over a range of radii, it may be a result of the varying resolution with which we resolve the ring during its spread, or it may simply be an indication that the behavior of viscosity in our code is more complex than can be captured with a simple α -disk approach. It is almost certainly not simply an initial transient, since over the course of the evolution the peak surface density in the ring has decreased by several e -foldings. The variation of α with time makes comparison with our disk evacuation radius method, which yields a single value of α , problematic. To facilitate comparison, we compute a time-averaged value of α in the ring test over a period from 0.09 to 0.9 orbital periods, where we have data for each ring radius. While this period may not be fully representative of the value of α as it evolves over time, it provides a rough estimate of the outcome of our ring tests.

Figure 10 compares a time-averaged estimate of α from the ring test with the values predicted by our disk evacuation

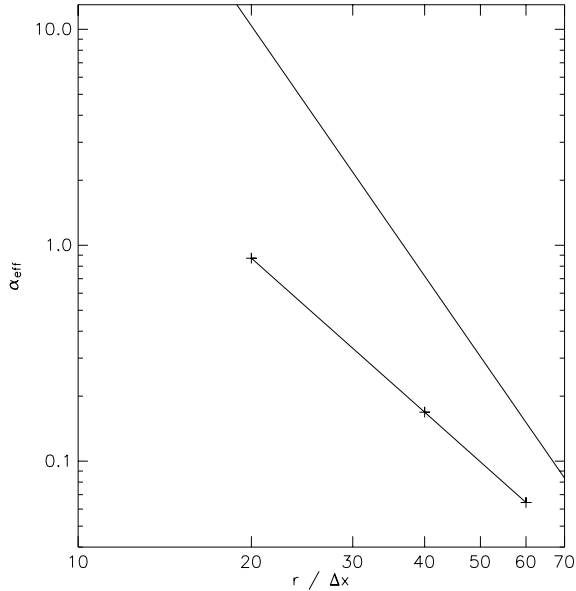


FIG. 10.—Plot showing α approximated using our disk evacuation method (upper, unmarked line) and using our ring method (lower line, marked with crosses).

method using equation (22). In all three cases that we ran, the ring tests show substantially smaller values of α , but the slope of α versus radius appears to be shallower than indicated by the disk evacuation radius method. We fit the time-averaged value of α versus radius shown in Figure 10 and find a best fit

$$\alpha \approx 1100 \left(\frac{r}{\Delta x} \right)^{-2.37} \quad (24)$$

for $r_B/\Delta x = 1.36 \times 10^4$. (We chose this very large value of $r_B/\Delta x$ to ensure that the ring was thin in the vertical direction.) We have not performed tests for other values of $r_B/\Delta x$, so it is not clear how α scales with $r_B/\Delta x$. For our chosen value of $r_B/\Delta x$, our estimate of α derived from the radius of the evacuated zone is more conservative for values of $\alpha > 0.016$, which occur for radii of $r/\Delta x < 107$. For larger radii or lower values of α , we can use equation (24) as a more conservative limit.

Equations (22) and (24) provide useful resolution requirements for hydrodynamics codes involving disks, since one can only believe the results of a disk simulation in those regions with resolution high enough that the effective α due to numerical viscosity is much smaller than the α that arises from whatever sources of physical viscosity are present. The scalings of α with r , r_B , and Δx that we have found using our two methods should depend only on geometry and on the physics of α -disks and so is likely to be about the same in any code using Cartesian geometry; the constants of proportionality, however, likely depend on the hydrodynamics algorithm. Hydrodynamics codes using different grid geometries, on the other hand, would likely have very different exponents in equations (22) and (24). In particular, a disk centered on the origin of a cylindrical or spherical grid would probably show much less numerical viscosity.

4. FROM BONDI TO BONDI-HOYLE ACCRETION

4.1. Background

While there have been extensive studies of Bondi-Hoyle accretion for flow with uniform velocities or smooth velocity gradients (Ruffert 1994a, 1994b, 1995, 1996; Ruffert & Arnett

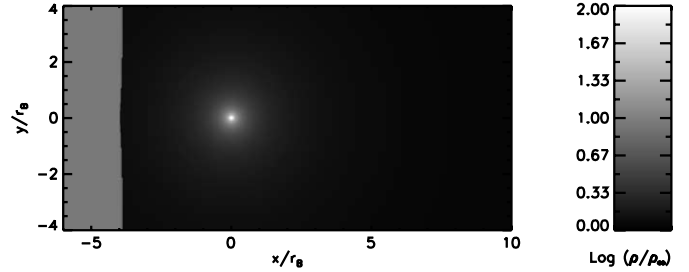


FIG. 11.—Logarithm of density for the initial configuration of the Bondi-Hoyle accretion problem. The image is a slice through the equatorial plane. We do not show the accretion region or sink particle because they are too small to see clearly. The plotted density range has been truncated at the top to bring out lower density features. The small irregularity visible in the shock front is a result of varying resolution: the shock is wider farther from the x -axis because it is spread over ~ 3 cells and the cells are larger farther from the axis. The Bondi radius is $r_B = 0.12$ pc; $\rho_\infty = 10^{-25}$ g cm $^{-3}$. [See the electronic edition of the *Journal* for a color version of this figure.]

1994; Ruffert & Anzer 1995; Fogliizzo & Ruffert 1997, 1999; Fogliizzo 2002), in many cases accretion occurs in a supersonically turbulent medium in which there are numerous shocks present. An example of accretion in a shock-filled medium is the star formation process. Observations of star-forming regions show that they are turbulent, with complex morphologies and velocity structures. Nonthermal line widths within star-forming regions range from transonic for the length scale of a core that forms a single star or small multiple system to highly supersonic on the scale of entire molecular clouds, with a power-law relation between line width and size in between (Larson 1981; Ossenkopf & Mac Low 2002). Simulations of turbulence in star-forming cores show that turbulent motions are able to reproduce the molecular line emissions, aspect ratios, line width gradients, and line width–size relations (Klein et al. 2003; Padoan et al. 2003).

It is therefore interesting to consider what happens when a shock rolls over an accreting object, such as a protostar in a molecular cloud clump. Initially, the accretion rate and density and velocity profiles will look like standard Bondi accretion; after long times, they will be appropriate for Bondi-Hoyle accretion. We are interested in studying the details of the transition and the time-dependence of the accretion rate.

4.2. The Simulation

We simulate a Mach 3 shock impacting an accreting particle. We place a sink particle with a mass of M_\odot in a gas of H and He with a temperature of 10 K. The Bondi radius is therefore $r_B = 0.12$ pc. Figures 11 and 12 show the initial configuration of the problem. We divide the computational domain into two regions. For $x > -4r_B$, the gas initially has a density and velocity profile given by the analytic solution for Bondi accretion with $\rho_\infty = 10^{-25}$ g cm $^{-3}$. This density is unrealistically low for a real star-forming region; we choose it because we wish to ensure that the particle does not accrete enough to substantially change its mass over the course of the simulation. We are neglecting the self-gravity of the gas, and thus as long as the sink particle's mass changes negligibly, the density is not a relevant parameter of the problem and may be scaled to an arbitrary value. For $x < -4r_B$, we generate postshock conditions appropriate for an isothermal shock of Mach number $\mathcal{M} = 3$ moving in the $+x$ -direction into a region with density ρ_∞ that is at rest. Thus, for $x < -4r_B$, we set $\rho = \mathcal{M}^2 \rho_\infty$, $v = (\mathcal{M} - 1/\mathcal{M})c_s \hat{x}$, where c_s is the sound speed.

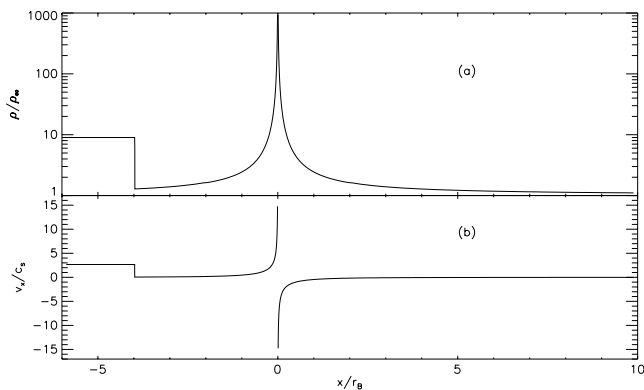


FIG. 12.—(a, b) Density and x -velocity, respectively, along the x -axis in the initial configuration of our Bondi to Bondi-Hoyle accretion problem. The Bondi radius, sound speed, and background density are $r_B = 0.12$ pc, $c_s = 0.19$ km s $^{-1}$, and $\rho_\infty = 10^{-25}$ g cm $^{-3}$.

The domain of the computation goes from $-4r_B$ to $4r_B$ in the y - and z -directions, and from $-6r_B$ to $10r_B$ in the x -direction. The boundary conditions are inflow/outflow in every direction. We choose refinement criteria to guarantee that the region around the sink particle is always resolved such that at a distance r from the sink particle, $r/\Delta x \geq 32$. This continues to a maximum resolution of $\Delta x = r_B/512 = 50$ AU. Note that for $\mathcal{M} = 3$, $r_B = 10r_{\text{BH}}$, so the maximum resolution is 51 cells per Bondi-Hoyle radius.

Before the shock reaches the sink particle, the flow is pure Bondi accretion; for the parameters we use in the simulation, the accretion rate should be $\dot{M}_B = 5.9 \times 10^{-11} M_\odot \text{ yr}^{-1}$. Long after the shock passes the particle, the flow should be Bondi-Hoyle accretion with $\mathcal{M} = 3$ and a background density of $\mathcal{M}^2 \rho_\infty = 9 \times 10^{-25}$ g cm $^{-3}$. Based on our results from § 3.3 and those of Ruffert (1996), the mean accretion rate should then be $\dot{M}_{\text{BH}} = 1.8 \times 10^{-11} M_\odot \text{ yr}^{-1}$.

4.3. Results

Figure 13 shows a series of snapshots of the simulation. As one might expect, the configuration does not change much until the shock approaches within a distance $\sim r_B$ of the particle. At that point, the shock starts to bow, with the part along the axis the farthest forward as it propagates into infalling gas. Figure 13a shows this effect. Figure 13b shows that as the shock passes the particle, a dense cylinder of shocked material accumulates where flows of gas swept up in the shock converge and shock again. This is the beginning of the converging streamlines and Mach cone that are characteristic of Bondi-Hoyle accretion. In Figure 13c the dense cylinder is showing the first signs of the destruction of axial symmetry. In Figure 13d the asymmetry is increasing. By Figure 13e, within one Bondi-Hoyle radius of the sink particle there is a well-developed Mach cone, which is becoming turbulent in its interior. In Figure 13f the Mach cone extends the full length of the range that we plot and appears to be fully turbulent in its interior. However, in Figure 13f the density of material in the Mach cone is still inflated due to the presence of material that was part of the Bondi flow and has been swept up by the shock. The densities in the cone undergo a slow decline until reaching a steady state; however, the morphology of the cone does not change further.

Figure 14 shows the accretion rate as a function of time. As the plot shows, the accretion rate is initially flat and in good agreement with the predicted value for Bondi accretion. When

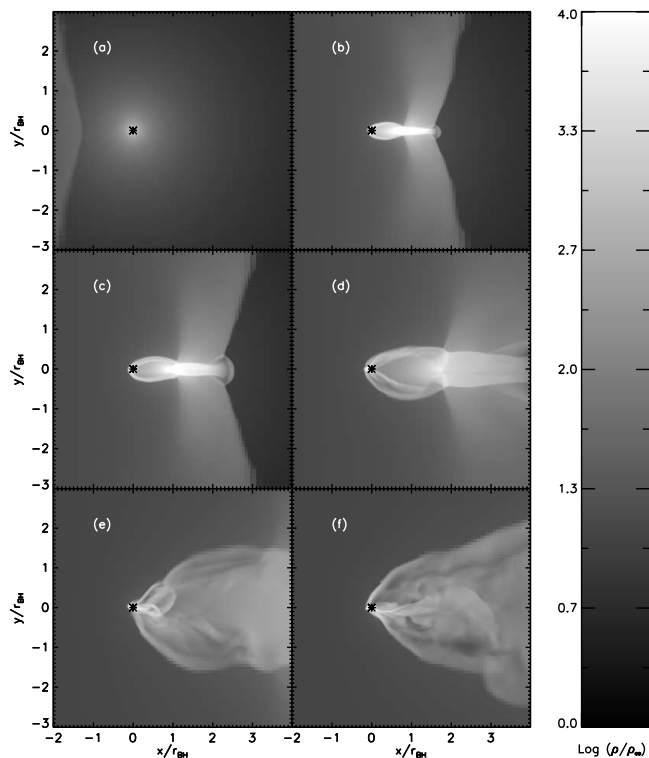


FIG. 13.—Series of slices through the equatorial plane. The density range has been truncated to bring out detail. The asterisks indicate the position of the sink particle; we do not show the accretion region because it is too small to see clearly. We have truncated the density range at the top and the bottom to bring out details. Note that in contrast with Figs. 11 and 12, the length unit of these plots is $r_{\text{BH}} = 0.012$ pc $= r_B/10$. The times shown in each panel are as follows: (a) $-0.25t_{\text{BH}}$, (b) $1.6t_{\text{BH}}$, (c) $1.9t_{\text{BH}}$, (d) $2.7t_{\text{BH}}$, (e) $4.3t_{\text{BH}}$, and (f) $5.9t_{\text{BH}}$, where $t_{\text{BH}} = 6.35 \times 10^4$ yr. [See the electronic edition of the *Journal* for a color version of this figure.]

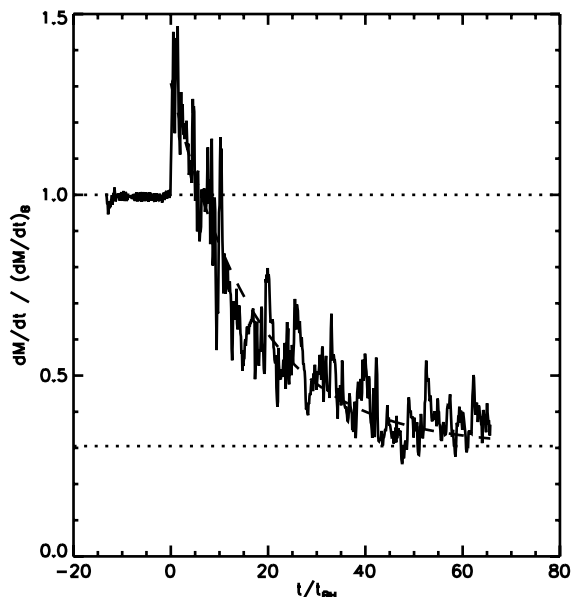


FIG. 14.—Solid curve: Accretion rate onto the sink particle vs. time, sampled at intervals of $t_{\text{BH}}/10$. Short-dashed horizontal lines: Predicted Bondi (upper line) and Bondi-Hoyle (lower line) accretion rates. Long-dashed line: Exponential fit to the accretion rate after the shock hits the sink particle. Time is plotted in units of $t_{\text{BH}} = 6.35 \times 10^4$ yr, and accretion rate is plotted in units of $\dot{M}_B = 5.9 \times 10^{-11} M_\odot \text{ yr}^{-1}$.

the shock hits the sink particle at $t = 0$, the accretion rate immediately increases by $\sim 50\%$ as the particle starts accreting high-density shocked material. The accretion rate then begins to fall off until it approaches its equilibrium value. After some experimentation, we found that the overall shape curve is reasonably well-fitted by an exponential of the form

$$\dot{M} = \dot{M}_{\text{BH}} + (\dot{M}_0 - \dot{M}_{\text{BH}}) \exp\left(-\frac{t}{t_{\text{trans}}}\right), \quad (25)$$

where \dot{M}_0 is the accretion rate immediately after the shock hits the sink particle and t_{trans} is the characteristic timescale for the accretion rate to transition from Bondi to Bondi-Hoyle. Our best-fit values for this case are $\dot{M}_0 = 7.7 \times 10^{-11} M_{\odot} \text{ yr}^{-1} = 1.3\dot{M}_{\text{B}} = 4.3\dot{M}_{\text{BH}}$ and $t_{\text{trans}} = 1.1 \times 10^6 \text{ yr} = 1.7t_{\text{B}} = 17t_{\text{BH}}$, where $t_{\text{B}} = r_{\text{B}}/c_s = 6.35 \times 10^5 \text{ yr}$ and $t_{\text{BH}} = r_{\text{BH}}/c_s = 6.35 \times 10^4 \text{ yr}$.

The characteristic timescale for the transition is closer to the Bondi time than the Bondi-Hoyle time. This is not surprising in retrospect. Before the shock hits, gas out to a distance $\sim r_{\text{B}}$ from the sink particle is inflowing supersonically. In a Bondi-Hoyle flow, gas at distances $\sim r_{\text{B}} \gg r_{\text{BH}}$ does not develop supersonic inflow velocities because it does not spend enough time near the sink particle. However, even after it is shocked, the gas that was originally part of the Bondi flow retains its supersonic infall speed and is therefore likely to find its way down to the sink particle. Since the gas is coming from a distance of order r_{B} , its characteristic timescale to reach the sink particle is of order t_{B} . Once all the leftover gas that was part of the Bondi flow out to $\sim r_{\text{B}}$ has drained onto the accreting particle, the accretion rate drops down to what one would expect for pure Bondi-Hoyle flow. As a result of this effect, the particle accretes an additional amount of mass

$$\Delta M \approx \int (\dot{M} - \dot{M}_{\text{BH}}) dt \approx (\dot{M}_0 - \dot{M}_{\text{BH}}) t_{\text{trans}} \approx 5.6 \dot{M}_{\text{BH}} t_{\text{BH}} \quad (26)$$

beyond what it would have accreted if the accretion rate had instantly shifted from Bondi to Bondi-Hoyle.

5. SUMMARY

We have demonstrated a new method for including moving, Lagrangian sink particles in a Eulerian hydrodynamics code. Our method shows excellent agreement with analytic results for a number of test problems, even in regimes where our formula for the accretion rate is at best a guess. In the process of testing the behavior of our sink particle in the presence of rotating flows, we develop a method to parameterize the effects of numerical viscosity in disk simulations and use it to compute α for our Cartesian AMR code. This method provides a resolution requirement for future disk simulations.

We have also solved a simple example problem using our sink particle method and shown that it produces useful results in that case. We have discovered one limit of our method: sink particles can produce significant errors in the accretion rate when the flow is transitioning from subsonic to supersonic at the accretion radius. It is not clear if SPH sink particles also encounter difficulty in this regime; we have not found any tests in the literature that address the point. Regardless, with AMR one can easily avoid the regime where $r_{\text{B}} \sim r_{\text{acc}}$, since one can simply increase the resolution temporarily until the sink particle accretes enough mass that $r_{\text{B}} > r_{\text{acc}}$.

The new technique will extend the range of Eulerian simulations, allowing them to run for longer times on problems gravitational collapse. Thus, it extends to Eulerian codes one of the heretofore unique advantages of SPH while retaining the accurate treatment of shocks and radiative transfer capability that are found in Eulerian approaches.

The authors thank Fumitaka Nakamura and Robert Fisher for useful discussions. This work was supported by NASA GSRP grant NGT 2-52278 (M. R. K.), NSF grant AST 00-98365 (C. F. M.), NASA ATP grant NAG 5-12042 (C. F. M. and R. I. K.), and the US Department of Energy at the Lawrence Livermore National Laboratory under contract W-7405-Eng-48. This research used resources of the National Energy Research Scientific Computing Center, which is supported by the Office of Science of the US Department of Energy under contract DE-AC03-76SF00098 and the NSF San Diego Supercomputer Center through NPACI program grant UCB267.

REFERENCES

- Aarseth, S. J. 1963, MNRAS, 126, 223
 Artymowicz, P., & Lubow, S. H. 1994, ApJ, 421, 651
 Bate, M. R., Bonnell, I. A., & Price, N. M. 1995, MNRAS, 277, 362
 Bell, J., Berger, M., Saltzman, J., & Welcome, M. 1994, SIAM J. Sci. Stat. Comput., 15, 127
 Berger, M. J., & Collela, P. 1989, J. Comput. Phys., 82, 64
 Berger, M. J., & Oliger, J. 1984, J. Comput. Phys., 53, 484
 Bondi, H. 1952, MNRAS, 112, 195
 Bonnor, W. B. 1956, MNRAS, 116, 351
 Boss, A. P., & Black, D. C. 1982, ApJ, 258, 270
 Bromm, V., Coppi, P. S., & Larson, R. B. 2002, ApJ, 564, 23
 Davis, M., Efstathiou, G., Frenk, C. S., & White, S. D. M. 1985, ApJ, 292, 371
 Ebert, R. 1955, Z. Astrophys., 317, 217
 Foglizzo, T. 2002, A&A, 392, 353
 Foglizzo, T., & Ruffert, M. 1997, A&A, 320, 342
 ———. 1999, A&A, 347, 901
 Gingold, R. A., & Monaghan, J. J. 1977, MNRAS, 181, 375
 Hoyle, F., & Lyttleton, R. A. 1939, Proc. Cambridge Philos. Soc., 35, 405
 ———. 1940a, Proc. Cambridge Philos. Soc., 36, 323
 ———. 1940b, Proc. Cambridge Philos. Soc., 36, 325
 ———. 1940c, Proc. Cambridge Philos. Soc., 36, 424
 Klein, R. I., Fisher, R. T., Krumholz, M. R., & McKee, C. F. 2003, Rev. Mex. AA Ser. Conf., 15, 92
 Kravtsov, A. V. 2003, ApJ, 590, L1
 Kuznetsov, O. A., Lovelace, R. V. E., Romanova, M. M., & Chechetkin, V. M. 1999, ApJ, 514, 691
 Larson, R. B. 1981, MNRAS, 194, 809
 Lucy, L. B. 1977, AJ, 82, 1013
 Lynden-Bell, D., & Pringle, J. E. 1974, MNRAS, 168, 603
 Nelson, A. F., Benz, W., & Ruzmaikina, T. V. 2000, ApJ, 529, 357
 Okamoto, T., Jenkins, A., Eke, V. R., Quilis, V., & Frenk, C. S. 2003, MNRAS, 345, 429
 Ossenkopf, V., & Mac Low, M.-M. 2002, A&A, 390, 307
 Padoan, P., Boldryev, S., Langer, W., & Nordlund, Å. 2003, ApJ, 583, 308
 Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. 1992, Numerical Recipes in C (2nd ed.; New York: Cambridge Univ. Press)
 Price, D. J., & Monaghan, J. J. 2004a, MNRAS, 348, 123
 ———. 2004b, MNRAS, 348, 139
 Pringle, J. E. 1981, ARA&A, 19, 137
 Richtmyer, R. D., & Morton, K. W. 1967, Difference Methods for Initial-Value Problems (New York: Wiley)
 Ruffert, M. 1994a, ApJ, 427, 342
 ———. 1994b, A&AS, 106, 505
 ———. 1995, A&AS, 113, 133
 ———. 1996, A&A, 311, 817
 ———. 1999, A&A, 346, 861
 Ruffert, M., & Anzer, U. 1995, A&A, 295, 108
 Ruffert, M., & Arnett, D. 1994, ApJ, 427, 351

- Shakura, N. I., & Sunyaev, R. A. 1973, *A&A*, 24, 337
- Shapiro, P. R., Martel, H., Villumsen, J. V., & Owen, J. M. 1996, *ApJS*, 103, 269
- Shu, F. H. 1977, *ApJ*, 214, 488
- Stone, J. M., Mihalas, D., & Norman, M. L. 1992, *ApJS*, 80, 819
- Stone, J. M., & Norman, M. L. 1992a, *ApJS*, 80, 753
- . 1992b, *ApJS*, 80, 791
- Toro, E. 1997, *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction* (Berlin: Springer)
- Truelove, J. K., Klein, R. I., McKee, C. F., Holliman, J. H., II, Howell, J. H., & Greenough, J. A. 1997, *ApJ*, 489, L179
- Truelove, J. K., Klein, R. I., McKee, C. F., Holliman, J. H., II, Howell, J. H., Greenough, J. A., & Woods, D. T. 1998, *ApJ*, 495, 821
- Williams, J. P., Blitz, L., & McKee, C. F. 2000, in *Protostars and Planets IV*, ed. V. Mannings, A. P. Boss, & S. S. Russell (Tucson: Univ. Arizona Press), 97