

# Collisional N-Body Dynamics Coupled to Self-Gravitating Magnetohydrodynamics Reveals Dynamical Binary Formation

JOSHUA E. WALL,<sup>1</sup> STEPHEN L. W. McMILLAN,<sup>1</sup> MORDECAI-MARK MAC LOW,<sup>2,3</sup> RALF S. KLESSEN,<sup>4,5</sup> AND SIMON PORTEGIES ZWART<sup>6</sup>

<sup>1</sup>*Drexel University, Department of Physics and Astronomy, Disque Hall, 32 S 32nd St., Philadelphia, PA 19104, USA*

<sup>2</sup>*Department of Astrophysics, American Museum of Natural History, 79th St at Central Park West, New York, NY 10024, USA*

<sup>3</sup>*Center for Computational Astrophysics, Flatiron Institute, 162 Fifth Ave., New York, NY 10010, USA*

<sup>4</sup>*Heidelberg University, Center for Astronomy, Institute for Theoretical Astrophysics, Albert-Ueberle-Str. 2, 69120 Heidelberg, Germany*

<sup>5</sup>*Heidelberg University, Interdisciplinary Center for Scientific Computing, INF 205, 69120, Heidelberg, Germany*

<sup>6</sup>*Leiden Observatory, Leiden University, Niels Bohrweg 2, 2333 Leiden, Netherlands*

(Dated: January 7, 2019; Received December 21, 2018)

Submitted to ApJ

## ABSTRACT

We describe a star cluster formation model that includes individual star formation from self-gravitating, magnetized gas, coupled to collisional stellar dynamics. The model uses the Astrophysical Multi-purpose Software Environment (AMUSE) to integrate an adaptive-mesh magnetohydrodynamics code (FLASH) with a fourth order Hermite N-body code (ph4), a stellar evolution code (SeBa), and a method for resolving binary evolution (multiples). This combination yields unique star formation simulations that allow us to study binaries formed dynamically from interactions with both other stars and dense, magnetized gas subject to stellar feedback during the birth and early evolution of stellar clusters. We find that for massive stars, our simulations are consistent with the observed dynamical binary fractions and mass ratios. However, our binary fraction drops well below observed values for lower mass stars, presumably due to unincluded binary formation during initial star formation. Further, we observe a build up of binaries near the hard-soft boundary that may be an important mechanism driving early cluster contraction.

*Keywords:* star formation — star clusters

## 1. INTRODUCTION

The study of star cluster formation through simulations is a non-linear physical problem with a wide range of scales. Clusters form from turbulent, magnetized molecular clouds that are parsecs across, yet the individual star formation process happens at scales of a single AU or less (Mac Low & Klessen 2004). Further complicating the issue, star formation contains a complex feedback loop in which stars forming in one epoch affect proximal regions of current and future star formation through radiation, winds and supernova feedback. The gravitational contraction of molecular clouds, star formation, stellar evolution, dynamical binary formation,

and cluster assembly and virialization all take place on timescales of 1–10 Myr. Resolving the relevant physical processes on all size and time scales is computationally challenging. As a result, approximations for star formation are used that include sink particles representing entire clusters (Gatto et al. 2017), simplified stellar feedback (Dale et al. 2014), or softened gravitational dynamics for stars (Federrath et al. 2010), or simulations neglect important dynamical agents such as magnetic fields (Rosen et al. 2016) in order to make the problem tractable.

In this study we describe numerical methods to resolve the dynamics of the stars and gas in order to study the formation of star clusters from gas collapse. This includes coupling of the magnetohydrodynamics (MHD) code FLASH (Fryxell et al. 2000), the N-body code ph4 (McMillan, S. et al. 2012), and the stellar evolution code

SeBa (Portegies Zwart & Verbunt 1996) using the Astrophysical Multi-purpose Software Environment (AMUSE Pelupessy et al. 2013), and implementation of a subgrid model for the formation of individual stars from sink particles. Since we focus on cluster formation and evolution as opposed to individual star formation, we have chosen to use the initial mass function (IMF) as an input rather than a result of our simulations. To accomplish this, we sample a Kroupa IMF (Kroupa 2001) using a Poisson process, but still individually form each star in a way that conserves mass both locally and globally.

The natural environment to develop these methods is AMUSE, as the original intention in the development of AMUSE was to allow for the coupling of different astrophysical codes for multiphysics simulations (Portegies Zwart et al. 2013). Further, multiple N-body and stellar evolution codes already exist in AMUSE, allowing us to change methods as needed. For example we could switch between SeBa or MESA (Paxton et al. 2011) for stellar evolution depending on the level of detail desired and computational cost acceptable. This allows us to represent the stars in FLASH, ph4 and SeBa as a single data structure that can be modified by any of the above codes, followed by propagation of the updated information to all other running codes. Interfacing FLASH into the AMUSE environment allows us to couple the gravitational potentials computed by FLASH and ph4 using a gravity bridge (see Sect. 2) directly using code in Python without major rewrites of either code.

In addition to interfacing FLASH with AMUSE, we have also made several additions to FLASH itself. For the heating and cooling of the gas we have modified the methods for atomic heating and cooling of Joung & Mac Low (2006) and Ibáñez-Mejía et al. (2016) with the molecular and dust cooling methods of Seifried et al. (2011), which themselves are based on Neufeld et al. (1995). To do this we have added our own model of heating from the photoelectric effect to dust using either the calculations from Wolfire et al. (2003) or Weingartner & Draine (2001), which can be chosen with a parameter switch. Finally, to solve for the both the degree of ionization and temperature of the gas as well as the dust temperature we have implemented our own integrators based on well known methods.

We reserve a detailed examination of these modifications to a subsequent paper, where we will also detail modifications we have made to include feedback from individual stars. In this work, we focus on the coupling of gravity between FLASH and ph4 using a gravity bridge.

In Sect. 2 we explain the concept of a gravity bridge and how we implement it, while in Sect. 2.2 we verify our implementation. In Sect. 3 we describe our method

for introducing star particles in regions of high gas density, and for handling binary or higher-order systems in Sect. 4. We define a demonstration problem in Sect. 5, and describe dynamical binary formation occurring in our models in Sect. 6. Finally, we summarize our results in Sect. 7.

## 2. GRAVITY BRIDGE

### 2.1. Implementation

Central to our implementation is the requirement to have fully collisional N-body dynamics calculated for stars evolving in gas-rich regions. To allow for physical interaction between the gas in FLASH and the stars in an N-body code, we implement a gravity bridge (Fujii et al. 2007) between the two codes. The method is a “kick-drift-kick,” leapfrog-type integration scheme with roots in the symplectic map method used by Wisdom & Holman (1991) to integrate the motions of planets in the solar system. In Wisdom & Holman (1991), the planets followed an analytic Kepler orbit around the Sun while being perturbed periodically by each other’s gravitational acceleration. The scheme was extended by Fujii et al. (2007) to integrate a star cluster subject to tidal effects inside a parent galaxy. While the method has previously been used to couple gas in smoothed particle hydrodynamics (SPH) to stars contained in an N-body code (e.g. Pelupessy & Portegies Zwart 2012), we have for the first time implemented this method with an Eulerian, adaptive mesh refinement (AMR) grid code. Here we briefly describe the AMUSE bridge method, following Fujii et al. (2007).

If we define the equation of evolution for our solution  $f(q(t), p(t); t)$  in terms of the Poisson bracket

$$\frac{df}{dt} = \{f, H\}, \quad (1)$$

where  $H$  is the Hamiltonian of the system, and define an evolution operator  $D_H$

$$D_H = \frac{d}{dt} = \{ \cdot, H \}, \quad (2)$$

the formal solution for  $f(t)$  is

$$f(t + \Delta t) = e^{\Delta t D_H} f(t). \quad (3)$$

Yoshida (1990) noted that if  $H$  (and therefore  $D_H$ ) is separated into kinetic and potential energy terms,  $H = K(p) + U(q)$  (with coordinates  $q$  and momenta  $p$ ), and  $D_K$  and  $D_U$  are defined as in Eq. 2, then the exponential in Eq. 3 can be approximated as

$$e^{\Delta t D_H} = e^{\Delta t (D_K + D_U)} \quad (4)$$

$$= e^{\Delta t} \prod_{k=1}^l e^{a_k D_K} e^{b_k D_U} + O(\Delta t^n), \quad (5)$$

for suitable  $l$ ,  $n$  and constants  $a_k, b_k$ . In the simplest case,  $l = 2, n = 2, a_1 = a_2 = 1/2, b_1 = 1$ , and  $b_2 = 0$ , and the total evolution of  $f(t)$  becomes a second-order integration scheme upon Taylor expansion

$$f(t + \Delta t) = e^{\frac{\Delta t}{2} D_K} e^{\Delta t D_U} e^{\frac{\Delta t}{2} D_K} f(t) \quad (6)$$

$$= \left(1 + \frac{\Delta t}{2} D_K\right) (1 + \Delta t D_U) \left(1 + \frac{\Delta t}{2} D_K\right) f(t). \quad (7)$$

We immediately recover the familiar kick-drift-kick formulation of the leapfrog integrator, since

$$D_U q_i = \{q_i, H_U\} = \frac{p_i}{m_i} = \dot{q}_i = v_i \quad (8)$$

$$D_K p_i = \{p_i, H_K\} = -m_i \nabla V_K = F_g = m_i a_g, \quad (9)$$

and the evolution of the system reduces to

$$v'_i = v_i + a_i(x) \frac{1}{2} \Delta t \quad (10)$$

$$x'_i = x_i + v'_i \Delta t \quad (11)$$

$$v'_i = v'_i + a_i(x') \frac{1}{2} \Delta t. \quad (12)$$

Wisdom & Holman (1991) noted that the Hamiltonian of a system comprising two or more coupled subsystems can alternatively be split into a set of secular evolution terms describing the internal evolution of each subsystem and perturbation terms consisting of delta functions, representing the interactions between the subsystems. Following Wisdom & Holman (1991) and Fujii et al. (2007), we split our Hamiltonian for each system into a sum of terms,  $D_K$  and  $D_D$ , representing, respectively, the kick due to the external perturbation and the drift due to internal (unperturbed) evolution. Regardless of the split, the Yoshida (1990) decomposition (Eq. 5) still applies, and the evolution of the system can be described by a scheme of the same form as Eq. 7.

In our simulations, the subsystems are the stars (modeled using **ph4**) and the gas (modeled using **FLASH**), so  $D_K$  is computed for the stars using the gravitational acceleration due to the gas, and for the gas using the gravitational acceleration due to the stars. For the drift operators, instead of deriving the drift from the Hamiltonian as was done in Eq. 8, we use each subsystem's internal integration scheme as shown in Figure 1. This means we now also introduce any error from the internal schemes (fourth-order for **ph4** and second-order for **FLASH**) to the formally symplectic integration of the bridge, but we gain the ability to couple the codes gravitationally. This error is found to be generally small, even for fairly large bridge timesteps (see Sect. 2.2). Our

integration scheme for stars is

$$\text{kick} \quad v'_{s,0} = v_{s,0} + \frac{\Delta t}{2} a_{g \rightarrow s,0} \quad (13)$$

$$\text{drift} \quad x_{s,1}, v'_{s,1} = \text{ph4}(x_{s,0}, v'_{s,0}, \Delta t) \quad (14)$$

$$\text{kick} \quad v_{s,1} = v'_{s,1} + \frac{\Delta t}{2} a_{g \rightarrow s,1} \quad (15)$$

where  $a_{g \rightarrow s}$  is the gravitational acceleration on the stars due to the gas. The stars receive an initial velocity kick from the gas, then drift alone, then get a final velocity kick from the gas. The same considerations lead to a similar procedure for the gas:

$$v'_{g,0} = v_{g,0} + \frac{\Delta t}{2} a_{s \rightarrow g,0} \quad (16)$$

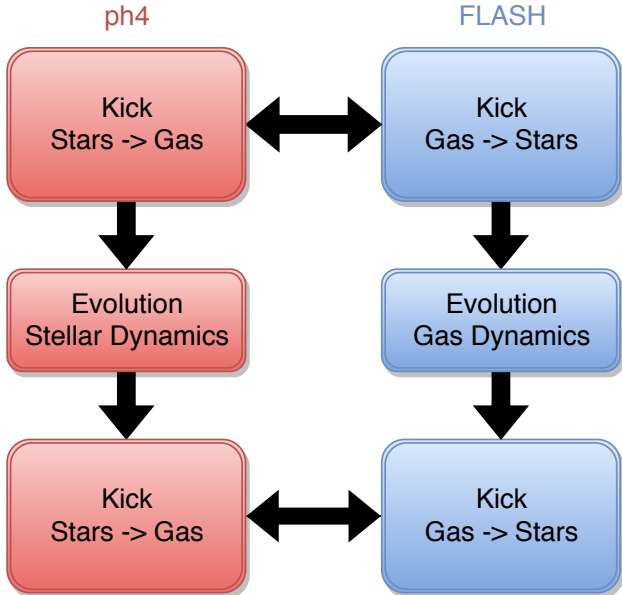
$$x_{g,1}, v'_{g,1} = \text{FLASH}(x_{g,0}, v'_{g,0}, \Delta t) \quad (17)$$

$$v_{g,1} = v'_{g,1} + \frac{\Delta t}{2} a_{s \rightarrow g,1}, \quad (18)$$

where  $a_{s \rightarrow g}$  is the gravitational acceleration on the gas due to the stars.

At each bridge step, the gravitational acceleration due to gas in each cell of the MHD code on the stars  $a_{g \rightarrow s}$  is calculated at the locations of each star in **ph4**, and the gravitational acceleration of each star on the gas  $a_{s \rightarrow g}$  is calculated at each cell site in **FLASH**. For obtaining the gravitational acceleration of the gas in **FLASH**, we use the first order finite differences of the potential calculated by the **FLASH** multigrid solver (Ricker 2008). For the acceleration of the particles on the gas, we initially used the acceleration directly from **ph4**. However testing showed that combining two different methods for stars on gas and gas on stars led to violations of Newton's Third Law. We have therefore included a cloud in cell mapping of the stellar masses onto the grid itself followed by the same multigrid potential and acceleration solution method, allowing us to use the same solver in both bridge directions to properly conserve momentum during the interactions.

The method averages the gravitational acceleration of one code on the other over the bridge time step, so the error in the bridge depends on the time step  $\Delta t$ . Testing with different timesteps has shown that  $\Delta t \sim t_{\text{ff}}/100$  is accurate enough to pass the tests presented in the next section, where  $t_{\text{ff}}$  is the minimum free-fall time in the simulation, although in practice we set bridge timesteps much smaller than this, generally  $\sim 2.5 \Delta t_h$ , where  $\Delta t_h$  is the hydro time step. Runs at this time step still mean that each code is taking numerous steps independently, making the whole simulation more efficient overall. Also, since the two codes drift independently, they can be in principle evolved in parallel for another improvement in speed.



**Figure 1.** The bridge scheme implemented in AMUSE using FLASH for hydrodynamics, ph4 for N-body, and SeBa for stellar evolution.

## 2.2. Verification

To test the gravity bridge we perform the test used by Federrath et al. (2010) for sink particles when they were first incorporated into FLASH. This consists of embedding three particles at different radii on circular orbits centered on a cloud of gas. The gas does not evolve and acts as a static potential, This tests the actual interaction between gas and particles, unlike imposing a background static potential without representation on the grid. The density of the gas varies as

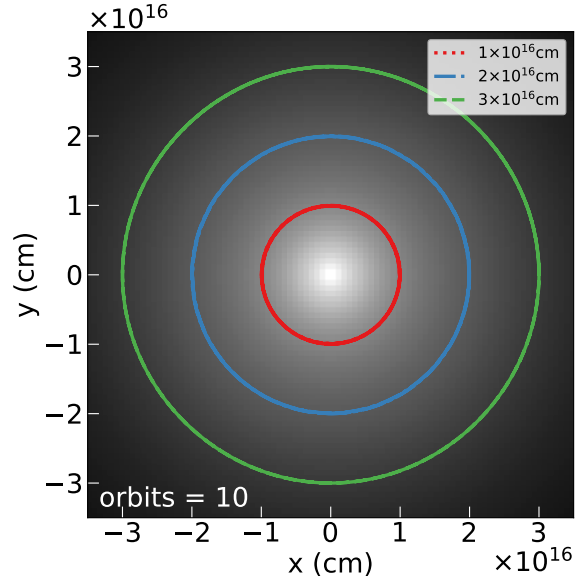
$$\rho(r) = \rho(r_o)(r_o/r)^2, \quad (19)$$

where  $\rho(r_o) = 3.82 \times 10^{-18} \text{ g cm}^{-3}$  and  $r_o = 5 \times 10^{16} \text{ cm}$ , which implies a gas mass of roughly  $3 M_\odot$  inside  $r_o$ . The three particles are placed at distances of  $10^{16} \text{ cm}$ ,  $2 \times 10^{16} \text{ cm}$  and  $3 \times 10^{16} \text{ cm}$  from the center of the gas cloud and have masses  $10^{-10} M_\odot$  such that the inter-particle gravity is very small compared to that of the gas. Each particle starts with a translational velocity for circular orbits

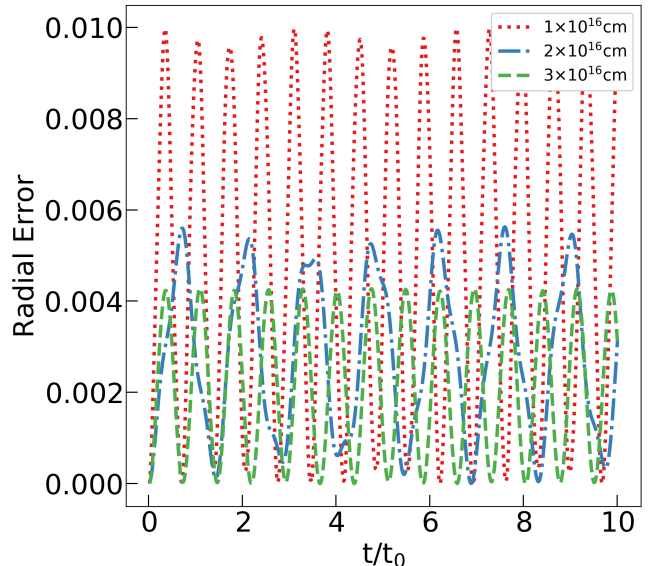
$$v = (GM(r)/r)^{1/2} = 895 \text{ m s}^{-1}, \quad (20)$$

which we lower by 2.3, 1.1, and 0.8% respectively to account for the non-singular nature of gravity on the grid at the origin (Federrath et al. 2010).

To check energy conservation we integrated 10 orbits of the innermost particle to compare against Federrath et al. (2010), with the result shown in Fig. 2. Our integration appears to close the orbits as well as the integration in Federrath et al. (2010), which used a second order

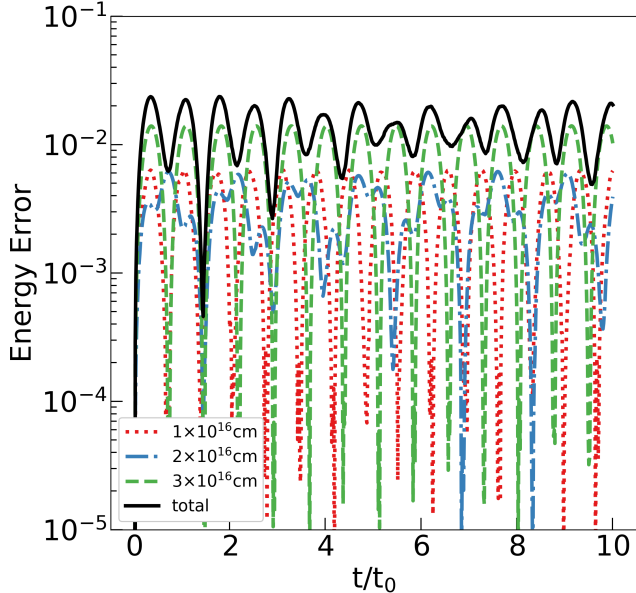


**Figure 2.** Orbital paths of three test particles after 10 orbits in an isothermal density profile where the gravitational acceleration of the particles is due to the bridge.



**Figure 3.** The fractional absolute error in radius of the three test particles.

leapfrog scheme. However their integration produced larger errors in the outermost orbit, while ours shows the most error in the innermost orbit. Federrath et al. (2010) attributed the error in the outer orbit to the finite effects of the grid (deviations from spherical symmetry) at the edges of the grid. Our model does not show this effect strongly as the density drops smoothly to the edge



**Figure 4.** The fractional energy error of the three test particles.

of the computational domain, while in the Federrath test the cloud has a sharp edge at  $\sim 4 \times 10^{15}$  cm where the density changes by three orders of magnitude.

Although the orbits in Figure 2 are well closed, they do oscillate slightly about the proper path. This is more clearly seen in a plot of the fractional radial error (Fig. 3). The resulting energy error, shown in Fig. 4, never rises above  $\sim 2\%$ . The larger radius error in the inner orbit corresponds to the larger angular distance covered by the inner particle between kicks, which in this test case were delivered at fixed time intervals  $\sim 10$  yr. The expected stability of symplectic integrators is evident, and the energy error does not grow noticeably with time.

### 3. STAR FORMATION

Capturing the range of scales in simulations is one of the core challenges to overcome in conducting studies of star cluster formation and the ISM in general. In order to account for the effects of the surrounding medium, including its large scale turbulence, magnetic fields and feedback, simulation boxes need to have sizes of tens to hundreds of parsecs. However in order to properly capture star formation for stars as small as a solar mass, including binary star formation, simulations need to be able to resolve the Jeans length  $\lambda_J$

$$\lambda_J = (\pi c_s^2 / G\rho)^{1/2} \quad (21)$$

which is on the order of or below a single AU. (Recent work does suggest that perhaps only hundreds of AU need be resolved; Sadavoy & Stahler 2017.)

To resolve the Jeans length in pure Eulerian hydrodynamics  $\lambda_J$  must be resolved by at least four grid cells (Truelove et al. 1997), while in MHD at least six cells per Jeans length are needed to resolve Alfvén waves (Heitsch et al. 2001), and as many as 32 cells per Jeans length would be needed to properly resolve self-consistent formation of magnetic fields through the microturbulent dynamo (Federrath et al. 2011). These requirements generally set the physical scales of the simulation, with the computational expense increasing with dynamical range.

Many authors overcome this difficulty when simulating star formation in large clouds by adding so called sink particles that represent entire clusters, essentially truncating the small scales. Clusters are created from Jeans unstable gas that is collected in sink particles (Bate et al. 1995; Krumholz et al. 2004; Federrath et al. 2010) on the grid (Dale et al. 2014; Gatto et al. 2017). This method requires taking random samples from the IMF, which in turn requires that enough gas be collected that the high-mass end of the IMF can be sampled appropriately. This typically means that around  $100 M_\odot$  to  $150 M_\odot$  must be collected in a sink particle, which once sampled for a cluster population becomes a single point source for all of the cluster’s feedback.

The second difficulty in modeling star formation comes from the effects of feedback at the protostellar and pre-main sequence phases. During the protostellar disk phase, accretion luminosity reduces fragmentation (Krumholz et al. 2007; Bate 2009; Peters et al. 2010, 2011). This luminosity, along with protostellar jet driving, is expected to reduce the efficiency of envelope accretion (Matzner & Jumper 2015). All of these will have an effect on the final main sequence star that results. Generally these effects are replaced by a local star formation efficiency parameter usually on the order of 0.1 to 0.5, which represents the fraction of gas that survived the accretion process from the stars initial outer gas envelope (e.g. Padoan et al. 2014).

Here we use an efficiency parameter to account for proto- and pre-stellar effects, creating our stars as main sequence objects. But to actually create the stars we take a different, and perhaps more historical, approach compared to recent simulations. Instead of sampling an IMF directly after collecting mass, we choose instead to take a Poisson sampling of the number of stars in a given mass bin in the IMF for any given star forming region as proposed by Sormani et al. (2017),

$$P_i(n) = e^{-\lambda_i} \lambda_i^{n_i} / n_i!, \quad (22)$$

where  $\lambda_i = f_i M / \langle m_i \rangle$ ,  $M$  is the total mass for a specific sample,  $\langle m \rangle$  is the average mass in the  $i^{th}$  bin for the



total range of the IMF sampled over,  $f_i$  is the fraction of the total mass in the  $i^{\text{th}}$  bin for the IMF range, and  $n$  is the number of stars for a specific sampling for which the probability  $P$  is returned.

The idea of Poisson sampling for mass values has been used before to choose from the IMF (Elmegreen 1997). It has the added mathematical benefit that even when sampling one star at a time the sum of all the samples will always reproduce the parent sample, since the product of the subset Poisson distributions of  $n_i, n_j$  with mean values  $\lambda_i, \lambda_j$  is equal to the Poisson distribution of the entire set  $N$  with the mean being  $\lambda_i + \lambda_j$ :

$$P(N) = \sum_{i=0}^N P(n_i, \lambda_i | n_j, \lambda_j) \quad (23)$$

$$= \sum_{i=0}^N \frac{\lambda_i^{n_i}}{n_i!} e^{-\lambda_i} \frac{\lambda_j^{n_j}}{n_j!} e^{-\lambda_j} \quad (24)$$

$$= \frac{(\lambda_i + \lambda_j)^N}{N!} e^{\lambda_i + \lambda_j}, \quad (25)$$

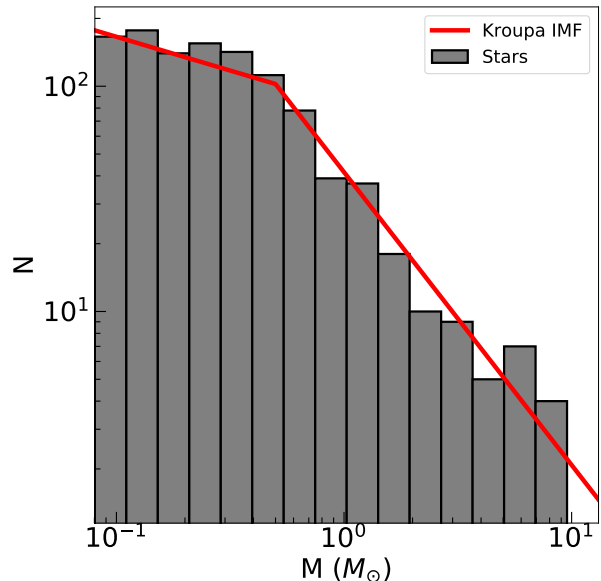
from the binomial theorem.

On the face of it, it would seem that the same considerations of having enough mass to sample each mass bin appropriately would apply to our Poisson sampling as well (see for example Sormani et al. 2017), since even an input of  $1 M_\odot$  of gas can result in an unphysical  $20 M_\odot$  star, even if we lack enough mass in the simulation to create it. Suggestions to overcome this difficulty in star formation methods have been to violate local mass conservation by sampling from all sink particles at once, or to simply sample all gas over a given density threshold throughout the simulation (Fujii & Portegies Zwart 2015).

Instead we *invert* this process. We use the sink particle routines of Federrath et al. (2010) in FLASH to identify star forming regions. Every time a sink particle forms because a region has become Jeans unstable, we create a list of stellar masses for that sink particle by sampling the IMF with our Poisson process with  $10^4 M_\odot$  of stars created at once, before the sink accretes any gas. The number of stars from the Poisson sampling in each mass bin is returned, and then we randomly sample from the Kroupa (2001) IMF in each bin to give actual masses to every star, with the sampled mass bracketed between  $0.08 M_\odot$  to  $150 M_\odot$ . We choose a minimum mass of  $0.08 M_\odot$  for the IMF. We then randomize the entire list of stars created. Once the sink particle obtains enough mass to create the first star in the list, this star is removed from the list and placed into the simulation, after which the sink must then gather enough mass for the second star in the list.

This method allows us to form particles star by star, without any violation of mass conservation. Each particle can take on the local momentum, mass and velocity of the sink at the time of formation. Also if a massive star forms, it has the chance to shut down local (or non local) star formation in the simulation by preventing further accretion, allowing the effects of feedback on star formation to be properly analyzed. Furthermore, since stars are formed individually, gravitational interactions between stars and with the surrounding gas can lead to binary formation and stellar ejections that can have important dynamical effects on the clusters and their surrounding natal gas clouds.

Each gas-gathering sink particle has an accretion radius of 2.5 times the smallest grid cell, to capture the local flow for accretion of the gas (Federrath et al. 2010). Since this is the best resolved location we have for star formation, star particles are placed randomly within this radius using an isothermal spherical density profile (Binney & Tremaine 2011). This allows some stars to form on the edges of these regions, but with smaller probability. We sample the velocity for the star from a Gaussian profile centered on the sink velocity, using the local gas sound speed as the variance. Fig. 5 is the resulting mass distribution for a typical small cloud ( $10^3 M_\odot$ ) run using this method.



**Figure 5.** The mass function of stars from run M3f presented in the Results section. The Kroupa IMF is shown for comparison, normalized to the same number of total stars as in the simulation, here 1100.

#### 4. MULTIPLE STARS

The formation of close binaries and higher-order systems can lead to the effective time step shrinking to a small fraction of the binary orbital period, preventing further integration of the solution due to the extreme computational expense. Normally, in N-body codes, the solution to this problem is to introduce a specialized treatment of close encounters, through regularization of the equations of motion (Aarseth & Zare 1974) or some other approximate treatment of close encounters (Portegies Zwart et al. 1999). Several of the N-body modules in AMUSE have the ability to incorporate such treatments, but for a general and minimally intrusive solution within the AMUSE framework, we prefer to handle close encounters using an external module, as we now describe.

The basic simplification in the approach we use is that the N-body code manages only the centers of mass of stable multiple systems. These include binaries, a stable hierarchical triples according to the Mardling (2008) criterion, or a higher-order multiple systems in which the Mardling criterion applied to the outermost orbits indicates stability. Close encounters are resolved using the `multiples` module (Portegies Zwart & McMillan 2019), which keeps track of the internal structure of all multiple systems and manages interactions between them. To operate with this module, an N-body code must be modified to detect close encounters and return immediately to the top-level AMUSE script controlling the simulation, where appropriate means are taken to resolve the encounter. Such functionality is straightforward to add, and is applied at the end of every N-body step.

In our case, the `ph4` module checks for pairs of particles that satisfy the stopping conditions that (1) they are approaching, (2) they have separations less than twice the sum of their effective dynamical diameters (a tunable parameter set at runtime to be 100 AU for all stars and twice the semi-major axis of a binary), and (3) they are relatively unperturbed by their next nearest neighbor (with the ratio of accelerations  $\gamma_p < 0.02$  in the terminology discussed below in Sect. 6 and represented by Eq. 26). Once the stopping condition is triggered, any internal structure in the two interacting particles is restored, and the entire system is moved to a separate code designed for small-N encounters, aptly named `smallN` (Hut et al. 1995; McMillan & Hut 1996). The `smallN` code models the encounter as a scattering experiment, terminating when the system has resolved itself into a collection of mutually unbound single stars or stable multiples (as just defined). The internal structure of the stable multiples is saved, their centers of mass are placed back in the N-body code, and the integra-

tion continues. In this way, arbitrarily complex hierarchical configurations can form and interact, and their dynamical histories can easily be monitored. This treatment of multiples is unusual in the N-body community, but similar implementations are widely used in Monte Carlo models of cluster dynamics (Chatterjee et al. 2010; Hypki & Giersz 2013).

Currently, the internal structure of a multiple is simply frozen until its next close encounter. Secular internal evolution or perturbations due to encounters too wide to trigger a stopping condition are not included. Binaries on wide or strongly perturbed orbits are not merged into their center of mass; instead, their components are returned directly to the N-body code. We note that, although `ph4` evolves only the centers of mass of multiple star systems, for all feedback and bridge calculations the individual component stars are used directly.

#### 5. DEMONSTRATION PROBLEM

As a demonstration of our method, we simulate star formation in turbulent spheres of gas. For gas dynamics in FLASH we use the unsplit MHD solver (Lee 2013) with third order PPM reconstruction (Colella & Woodward 1984) and the HLLD Riemann solver (Miyoshi & Kusano 2005), while for solving Poisson’s equation for gravity we use the multigrid solver of Ricker (2008). We include feedback in these runs from radiation, winds and supernovae that we will describe in a subsequent paper, since the results we describe here are not strongly affected by the feedback. We initialize the density field with the commonly-used, initially spherically symmetric, Gaussian, gas distribution of Bate et al. (1995), while the velocity field is generated with a turbulent Kolmogorov velocity spectrum (Wünsch 2015) for the dense gas. All runs have eight levels of AMR refinement with the exception of M3f, which has seven, with refinement triggered by the Jean’s criterion described in Federrath et al. (2010). All the runs except M3V2 have all three stellar feedback methods (winds, radiation and supernovae) switched on, although no run has yet to experience a supernova event.

We use total masses of  $M = 10^3, 10^4$  and  $10^5 M_\odot$  and Gaussian density profiles with variance  $r_o = 5, 10$  and 50 pc respectively. These length scales are chosen to roughly match the average density scales of clouds of these masses (Stahler & Palla 2008). Note this means the larger clouds have significantly longer free fall times. Outside of the sphere the density is chosen to roughly match the ISM density for a containing medium based on the size and density of the sphere itself (i.e. for the  $10^3 M_\odot$  sphere, with higher density, the containing medium was assumed to be cold neutral medium, while

for the  $10^5 M_\odot$  sphere the containing medium is warm and ionized). Then the temperatures are chosen to keep the sphere and containing medium in pressure equilibrium. The physical grid domain sizes  $D$ , listed in Tab. 1, are  $\sim 1.3$  to  $1.5$  times the Gaussian width  $r_0$  in each case. All models reported here were initialized with virial ratio of kinetic to potential energy of 0.2. We choose this initially low virial parameter to ensure quick cloud collapse even before all of the turbulence decays within a free-fall time (Mac Low et al. 1998). As expected our spheres rapidly collapse into filamentary structures and begin forming stars (see Fig. 6).

The four models we analyze are the current snapshots of our first runs, listed in Tab. 1 and shown in Fig. 6. Note although M5f is both more massive and significantly older, it also contains a  $97 M_\odot$  star that is shutting down star formation in a large volume. Therefore the number of stars is comparable to the much younger star forming regions in other runs. The larger runs have larger minimum cell sizes, since all runs have the same number of refinement levels, but they also have more individual filaments and cluster forming regions. Finally we note that simulations of this nature are in general highly stochastic and therefore only predictable in a statistical sense.

## 6. BINARIES

Given the collisional nature of our coupled code, the possibility exists of dynamical binary formation by interactions between stars and with the gas. Indeed, all four runs examined here formed binaries. Note that here when we refer to binaries, we mean any particles that are bound. Not all of these will necessarily be merged into root particles in `multiples` (Sec. 4). The `multiples` code is a numerical solution to the problem of prohibitively short timesteps, but will only act on the tightest physical multiple systems.

To identify binaries in our simulations, we first calculate the relative energies of all stars with respect to each other, keeping only those that are bound to each other. We identify those that are mutually bound, which is our initial list of binary candidates. We then test each binary, consisting of stars with masses  $m_1$  and  $m_2$  and semi-major axis  $a$ , for perturbation by any star with mass  $m_p$  and distance  $d$  from the binary center of mass,

$$\left| \frac{m_p m_1}{(d-a)^2} - \frac{m_p m_2}{(d+a)^2} \right| < \gamma_p \frac{m_1 m_2}{4a^2}, \quad (26)$$

where  $\gamma_p$  is the chosen limiting ratio of accelerations. As a guide for choosing  $\gamma_p$ , we consider cases in which all three stars have equal mass. Then for ratios  $d/a = 2$ , 5, and 10, inverting Eqn. 26 gives  $\gamma_p \sim 3.0$ , 0.14, and

0.016 respectively. For the preliminary analysis that we present here, we chose  $\gamma_p = 3.0$  and combine the results of all four runs, which yields 85 binaries.

The multiplicity fraction

$$f_{bin} = \frac{B}{S+B} \quad (27)$$

for each mass bin is shown in Fig. 8, where  $B$  is the total number of binary systems, and  $S$  is the total number of single stars. For comparison we include observations of  $f_{bin}$  compiled by Duchêne & Kraus (2013).

The lack of low mass binaries is due to the fact that we do not include any primordial binaries as we form stars, nor do we have high enough resolution to capture the gas dynamics that may lead to primordial binary formation, such as core fragmentation at small scales (Bate 2012). Fig. 9 shows that in absolute numbers, most binaries are close to  $1 M_\odot$ , with a steep decline for more massive stars following the IMF. We have no binaries containing a primary star with mass below  $0.1 M_\odot$ , although our IMF goes down to  $0.08 M_\odot$  in all runs.

The value of  $f_{bin}$  at the massive end appears remarkably consistent with the observations, despite our neglect of primordial binaries.

Indeed, all of our massive binaries have separations  $r \gtrsim 1 \text{ AU}$ , consistent with dynamical formation, as shown in Fig. 10. This agrees with observations that show the majority of massive stars occur in hierarchical systems consisting of tight, presumably primordial, binaries orbiting on wide orbits consistent with the dynamical binaries formed in our system (Karl et al. 2018).

For binaries consisting of a primary mass  $M_p$  and secondary mass  $M_s$ , the mass ratio  $q = M_s/M_p$ , is shown in Figs. 11 and 12. Comparing with observations in our mass range, we find our  $q$  distribution consistent with Kouwenhoven et al. (2005), with a large peak for  $q < 0.2$  followed by a power law drop. It is interesting to note that both our data and Kouwenhoven et al. (2005) appear to have moderate peaks near  $q \sim 0.5$  and  $q > 0.8$ . To test the robustness of the peaks in our histogram, we also examine the data as a cumulative distribution and using kernel density estimation. The multi-modal appearance of the data is evident in all three methods. The width of the bins for the histogram was calculated with the method described by Doane (1976), which works well for small data sets and does not assume the data is strictly Gaussian. For the kernel density estimation we use the cross validation technique of leaving out one data point for computing the bandwidth  $\sigma$  of the Gaussian. We evenly sample  $\sigma$  from  $10^{-3}$  to 1 for each rotation through all the data, comparing the mean integrated square error of each fit with the full data set, to find the appropriate bandwidth.



**Table 1.** Parameters for each of the four runs described here including mass  $M$ , total number of stars  $N_s$  at end of run  $t_{\text{end}}$  when analysis was performed, time at first star-forming event  $t_{\text{sf}}$ , the cell size  $\Delta x$  at maximum refinement and the domain size  $D$ . Note that M3 and M3f used different random turbulent patterns initially, explaining their different values of  $t_{\text{sf}}$ .

Run <sup>a</sup>	$M$ ( $M_\odot$ )	$N_s$	$t_{\text{sf}}$ (Myr)	$t_{\text{end}}$ (Myr)	$\Delta x$ (pc)	$D$ (pc)
M3	$10^3$	1100	2.86	4.38	0.01	10
M3f	$10^3$	1062	2.31	3.90	0.02	10
M4f	$10^4$	589	7.76	9.14	0.05	12
M5f	$10^5$	1144	15.38	17.82	0.2	110

<sup>a</sup>Runs ending in “f” include radiation, winds, and supernovae.

With fully collisional N-body dynamics, we expect to see a separation of our binaries into hard and soft regimes following the Heggie-Hills law (Hills 1975; Heggie 1975). For an average effective cluster thermal energy of

$$\frac{3}{2}NkT = \frac{1}{2}\langle m \rangle \sigma_v^2, \quad (28)$$

and a binary energy of

$$x = \frac{Gm_1m_2}{2a}, \quad (29)$$

with soft binaries having  $x/kT \lesssim 1$  and hard binaries having  $x/kT \gtrsim 1$ , the separation between the two types grows with time. In our runs, we indeed see distinct hard and soft populations well separated by a boundary at  $\sigma_v \sim 3 \text{ km s}^{-1}$ , the stellar velocity dispersion averaged across all four runs. It also appears that the soft binaries accumulate near the hard/soft boundary, which should be an important energy sink for the clusters as these binaries are disrupted. Gas dynamical friction could drive binaries to build up in this way, after which they could be disrupted near the maximal soft binary energy and thereby drive the entire cluster to contract (Leigh et al. 2014). Higher resolution runs will be needed to confirm if this effect occurs at the smallest binary separations.

## 7. SUMMARY

In this work we have coupled the Eulerian MHD code FLASH with stellar evolution and collisional N-body dynamics using the ph4 and SeBa codes in the AMUSE software framework. We then used AMUSE to couple the two gravity calculations using a gravity bridge to allow for interaction between the gas and stars, allowing us to simulate open cluster formation and early evolution in spherical, turbulent clouds of masses  $10^3$ – $10^5 M_\odot$ .

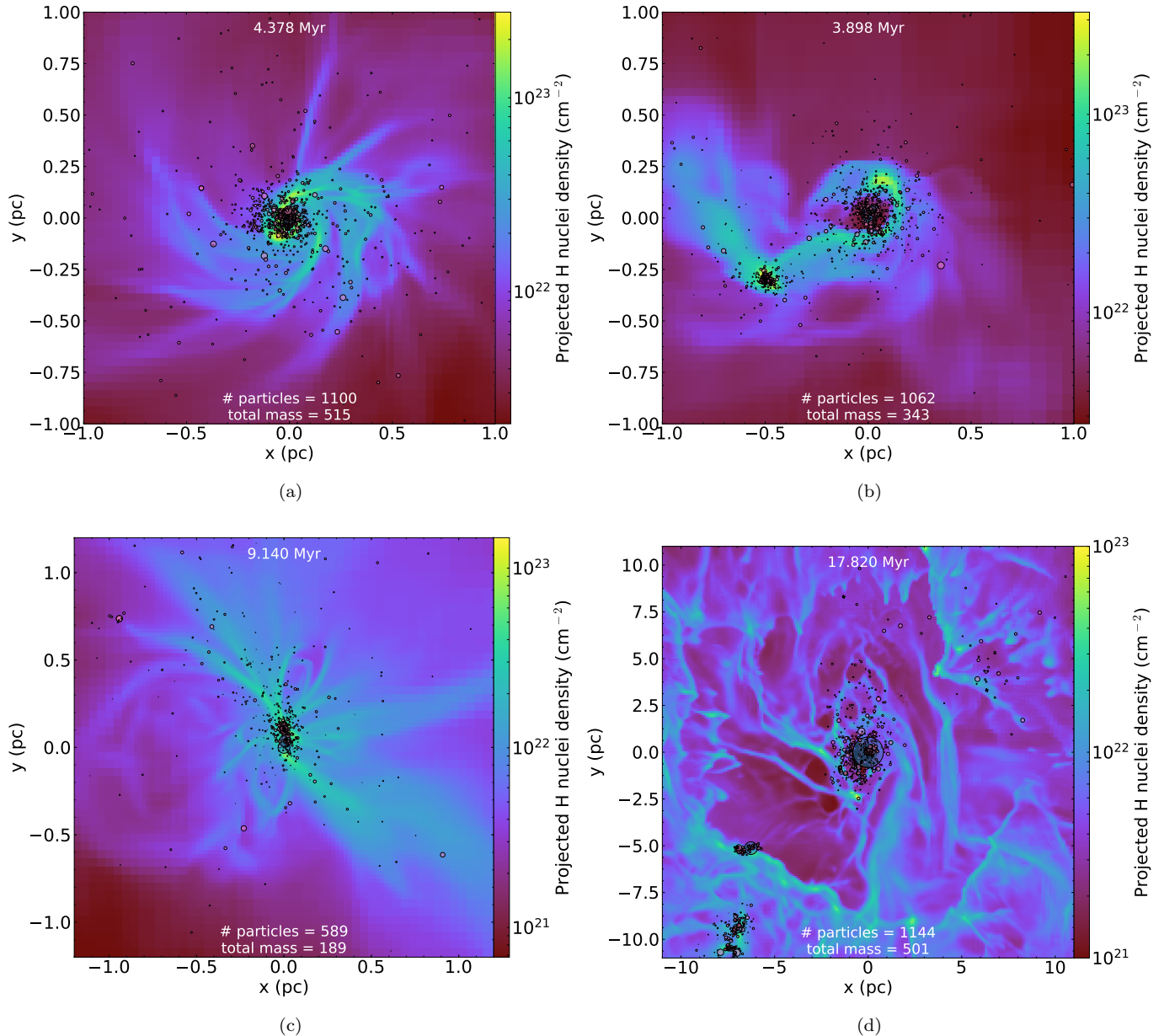
We have examined the binary populations produced in our demonstration runs. Despite not injecting any pri-

mordial binary population from core or disk fragmentation, we find a large number of wide binaries with properties that suggest they formed by interaction with the gas.

We find that the mass ratios of these binaries appear consistent with observations, and that the binary fraction of massive binaries is close to that observed. The lack of low-mass or tight binaries that we find suggests that those populations are predominantly produced by primordial core or disk fragmentation, but that the wide hierarchical multiple systems in which massive stars occur may be formed by this dynamical mechanism acting on primordial binaries. We find well separated hard and soft binary populations as predicted by the Heggie-Hills law, with evidence of a buildup of soft binaries near the boundary between the groups. Our results suggest that the hitherto little considered interaction of stars with gas during the early evolution of stellar clusters while their natal gas remains present, may explain much of the wide binary and multiple population, particularly for the most massive stars.

With publication of this work we make public our interface for the FLASH code in the AMUSE framework, in order to allow reproduction of this work. We hope our interface inspires others to use the coupling ideas behind this work in ways we might never consider ourselves, in the spirit of scientific discovery. The interface can be found within the FLASH directory of the AMUSE code at <https://github.com/amusecode/amuse>. Specific implementation details are available from the first author upon request.

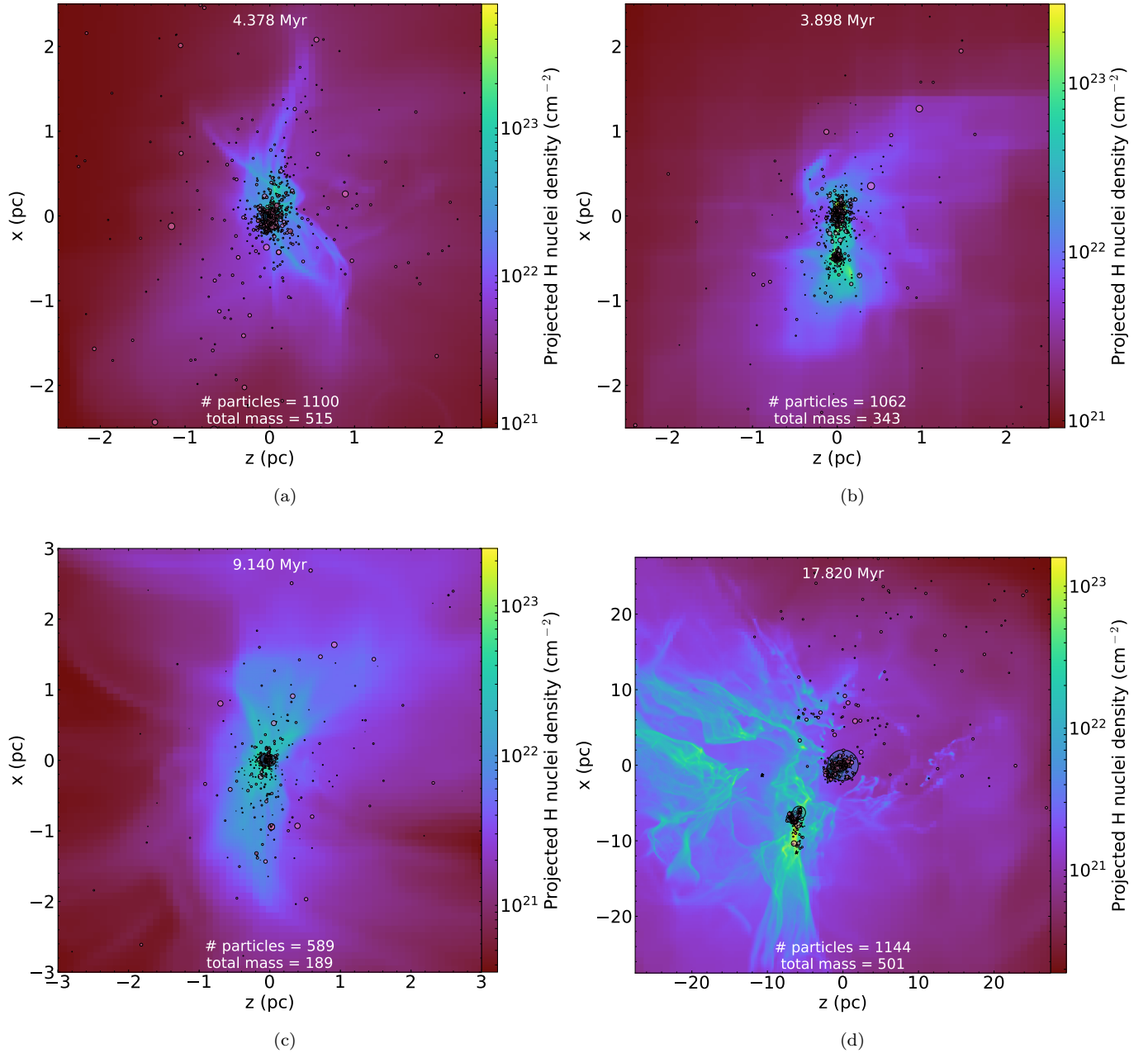
We acknowledge M. Davis, C. Federrath, S. Glover, A. Hill, J. Moreno, & E. Pellegrini for useful discussions, and A. van Elteran and I. Pelupessy for assistance with AMUSE, R. Banerjee and D. Seifried for kindly provid-



**Figure 6.** Projected number density along the  $z$ -axis for runs (a) M3 (b) M3f (c) M4f and (d) M5f at the last data file from each run. The area of the circles representing stars are proportional to their mass, while the locations of sink particles are shown by white star symbols. Feedback is most effective in run (b) where multiple feedback stars have sunk together to the center of the cluster and in (d) due to the  $97 M_{\odot}$  star in the center of the image.

ing the base code for dust and molecular cooling, and R. Wunsch for kindly providing a helper script for the initial conditions. Analysis and plotting for this work was done using `yt` (Turk et al. 2011). M-MML was additionally supported by the Alexander-von-Humboldt Stiftung. We acknowledge NASA grant NNX14AP27G, which supported this work, and the Dutch National Supercomputing Center SURFSara grant 15520 that pro-

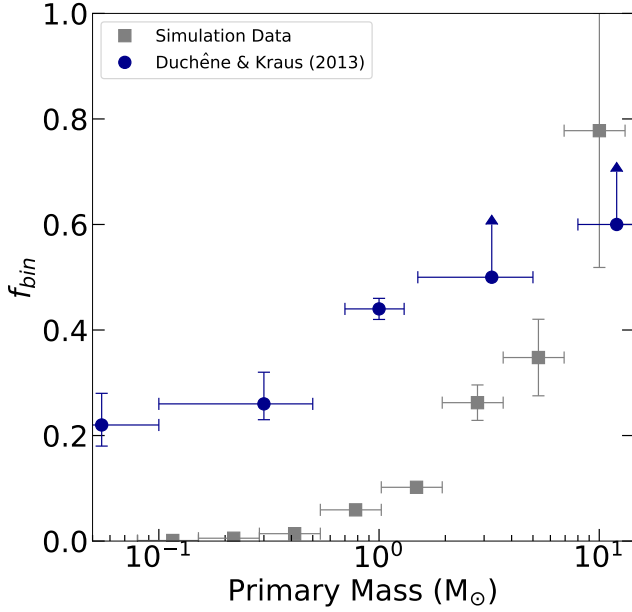
vided computing resources for our simulations. RSK acknowledges support from the Deutsche Forschungsgemeinschaft (DFG) via SFB 881 "The Milky Way System" (sub-projects B1, B2 and B8), and SPP 1573 "Physics of the ISM". Furthermore RSK thanks the European Research Council for funding under the European Communitys Seventh Framework Programme via the ERC Advanced Grant "STARLIGHT" (project number 339177).



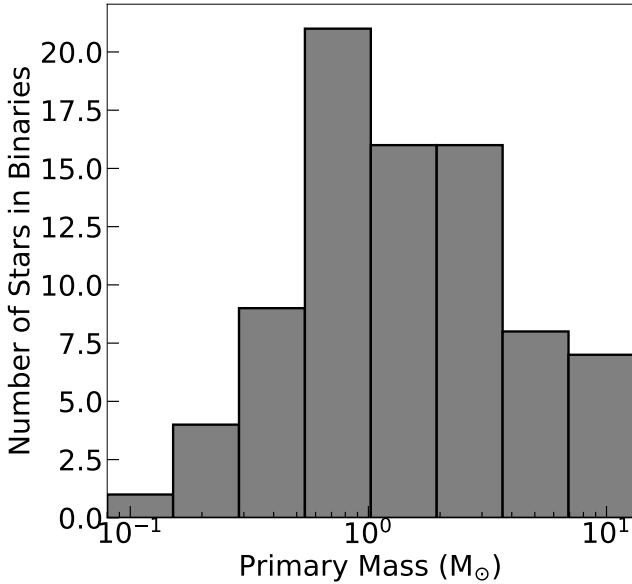
**Figure 7.** Projected number density along the  $y$ -axis for runs (a) M3 (b) M3f (c) M4f and (d) M5f at the last data file from each run. These images are zoomed out by a factor of  $\sim 2$  compared to Fig. 6 to better show the overall structure. (a) and (c) have fully collapsed and merged, while (b) is in the process of merging two subclusters, and (d) is still scattered. (d) also shows signs of partial disruption as the cloud was destroyed above and right of the  $97 M_{\odot}$  star, causing many cluster members to become unbound.

## REFERENCES

- Aarseth, S. J., & Zare, K. 1974, *Celestial Mechanics*, 10, 185
- Bate, M. R. 2009, *MNRAS*, 392, 1363
- . 2012, *MNRAS*, 419, 3115
- Bate, M. R., Bonnell, I. A., & Price, N. M. 1995, *MNRAS*, 277, 362
- Binney, J., & Tremaine, S. 2011, *Galactic Dynamics: (Second Edition)*, Princeton Series in Astrophysics (Princeton University Press)
- Chatterjee, S., Fregeau, J. M., Umbreit, S., & Rasio, F. A. 2010, *ApJ*, 719, 915



**Figure 8.** The fraction of all stars in binaries by stellar mass. Our simulations produce massive binaries at a rate consistent with observations, but very few low-mass binaries. The mass errors shown are the bin widths, while the  $f_{bin}$  error is given by the Poisson statistical error  $N_s^{-1/2}$ .



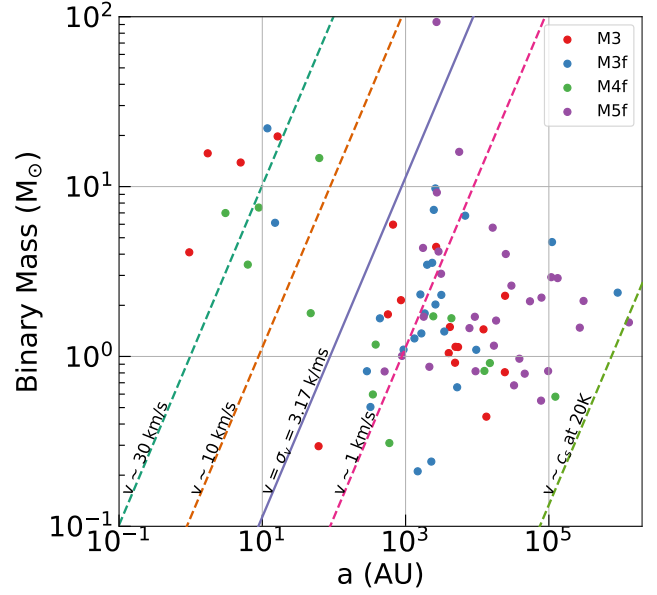
**Figure 9.** The number of stars in binaries binned by stellar mass.

Colella, P., & Woodward, P. R. 1984, *J. Comput. Phys.*, **54**, 174

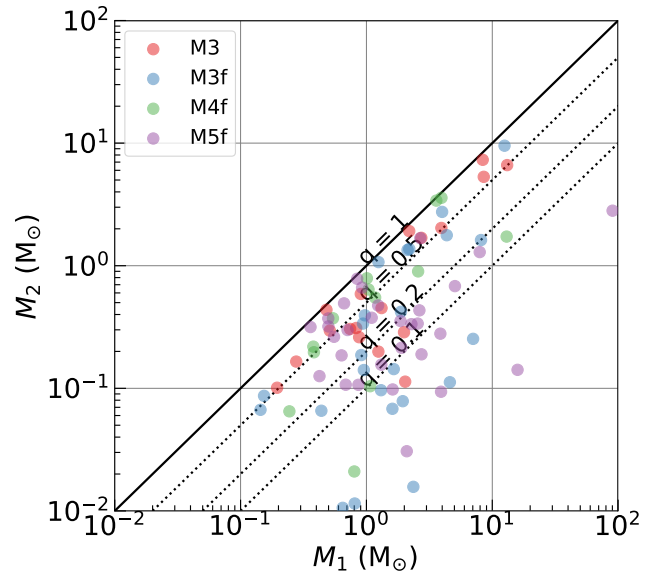
Dale, J. E., Ngoumou, J., Ercolano, B., & Bonnell, I. A. 2014, *MNRAS*, **442**, 694

Doane, D. P. 1976, *The American Statistician*, **30**, 181

Duchène, G., & Kraus, A. 2013, *ARA&A*, **51**, 269



**Figure 10.** The binary mass and semi-major axis, with lines representing several cluster thermal energies overplotted on their corresponding binary masses and radii. Also shown is the mean thermal energy of all the stars averaged across all four simulations. There is a clear separation into hard and soft binaries, as expected from the Heggie-Hills Law.

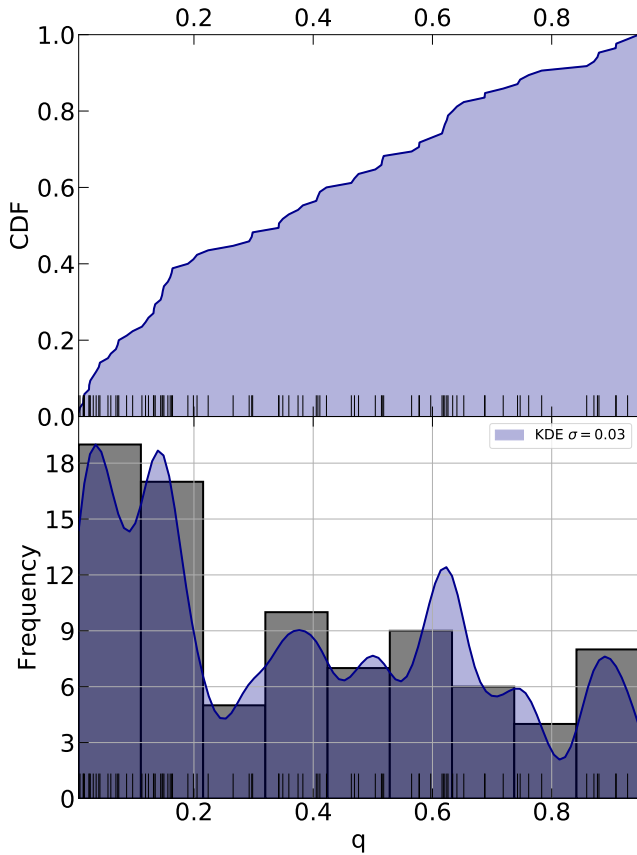


**Figure 11.** The mass ratio of the primary to the secondary of the 77 binaries we have found in our runs so far. Also shown are several lines of constant mass ratio  $q$  for comparison.

Elmegreen, B. G. 1997, *ApJ*, **486**, 944

Federrath, C., Banerjee, R., Clark, P. C., & Klessen, R. S. 2010, *ApJ*, **713**, 269





**Figure 12.** The distribution of mass ratios. In the top panel we show a cumulative distribution, while in the bottom panel we show a histogram overplotted with a kernel density estimation. The Gaussian kernel for this estimate has a bandwidth  $\sigma = 0.03$ .

Federrath, C., Sur, S., Schleicher, D. R. G., Banerjee, R., & Klessen, R. S. 2011, *ApJ*, **731**, 62

Fryxell, B., Olson, K., Ricker, P., et al. 2000, *ApJS*, **131**, 273

Fujii, M., Iwasawa, M., Funato, Y., & Makino, J. 2007, *PASJ*, **59**, 1095

Fujii, M. S., & Portegies Zwart, S. 2015, *MNRAS*, **449**, 726

Gatto, A., Walch, S., Naab, T., et al. 2017, *MNRAS*, **466**, 1903

Heggie, D. C. 1975, *MNRAS*, **173**, 729

Heitsch, F., Mac Low, M.-M., & Klessen, R. S. 2001, *ApJ*, **547**, 280

Hills, J. G. 1975, *AJ*, **80**, 809

Hut, P., Makino, J., & McMillan, S. 1995, *ApJ*, **443**, L93

Hypki, A., & Giersz, M. 2013, *MNRAS*, **429**, 1221

Ibáñez-Mejía, J. C., Mac Low, M.-M., Klessen, R. S., & Baczynski, C. 2016, *ApJ*, **824**, 41

Joung, M. K. R., & Mac Low, M.-M. 2006, *ApJ*, **653**, 1266

Karl, M., Pfuhl, O., Eisenhauer, F., et al. 2018, [arXiv:1809.10376](https://arxiv.org/abs/1809.10376) [astro-ph]

Kouwenhoven, M. B. N., Brown, A. G. A., Zinnecker, H., Kaper, L., & Portegies Zwart, S. F. 2005, *A&A*, **430**, 137

Kroupa, P. 2001, *MNRAS*, **322**, 231

Krumholz, M. R., Klein, R. I., & McKee, C. F. 2007, *ApJ*, **656**, 959

Krumholz, M. R., McKee, C. F., & Klein, R. I. 2004, *ApJ*, **611**, 399

Lee, D. 2013, *J. Comput. Phys.*, **243**, 269

Leigh, N. W. C., Mastrobuono-Battisti, A., Perets, H. B., & Böker, T. 2014, *MNRAS*, **441**, 919

Mac Low, M.-M., & Klessen, R. S. 2004, *Rev. Mod. Phys.*, **76**, 125

Mac Low, M.-M., Klessen, R. S., Burkert, A., & Smith, M. D. 1998, *PhRvL*, **80**, 2754

Mardling, R. A. 2008, in *IAU Symposium, Vol. 246, Dynamical Evolution of Dense Stellar Systems*, ed. E. Vesperini, M. Giersz, & A. Sills, 199

Matzner, C. D., & Jumper, P. H. 2015, *ApJ*, **815**, 68

McMillan, S. L. W., & Hut, P. 1996, *ApJ*, **467**, 348

McMillan, S., van Elteran, A., & Whitehead, A. 2012, in *Astronomical Society of the Pacific Conference Series, Vol. 453*, 129

Miyoshi, T., & Kusano, K. 2005, *J. Comput. Phys.*, **208**, 315

Neufeld, D. A., Lepp, S., & Melnick, G. J. 1995, *ApJS*, **100**, 132

Padoan, P., Haugbølle, T., & Nordlund, Å. 2014, *ApJ*, **797**, 32

Paxton, B., Bildsten, L., Dotter, A., et al. 2011, *ApJS*, **192**, 3

Pelupessy, F. I., & Portegies Zwart, S. 2012, *MNRAS*, **420**, 1503

Pelupessy, F. I., van Elteren, A., de Vries, N., et al. 2013, *A&A*, **557**, A84

Peters, T., Banerjee, R., Klessen, R. S., & Mac Low, M.-M. 2011, *ApJ*, **729**, 72

Peters, T., Klessen, R. S., Mac Low, M.-M., & Banerjee, R. 2010, *ApJ*, **725**, 134

Portegies Zwart, S., McMillan, S., van Elteren, A., Pelupessy, I., & de Vries, N. 2013, *Comput. Phys. Comm.*, **184**, 456

Portegies Zwart, S., & McMillan, S. L. W. 2019, *Astrophysical Recipes: The Art of Amuse* (Institute of Physics Publishing)

Portegies Zwart, S. F., Makino, J., McMillan, S. L. W., & Hut, P. 1999, *A&A*, **348**, 117

Portegies Zwart, S. F., & Verbunt, F. 1996, *A&A*, **309**, 179

Ricker, P. M. 2008, *ApJS*, **176**, 293

Rosen, A. L., Krumholz, M. R., McKee, C. F., & Klein, R. I. 2016, *MNRAS*, **463**, 2553

Sadavoy, S. I., & Stahler, S. W. 2017, *MNRAS*, **469**, 3881

- Seifried, D., Banerjee, R., Klessen, R. S., Duffin, D., & Pudritz, R. E. 2011, *MNRAS*, **417**, 1054
- Sormani, M. C., Treß, R. G., Klessen, R. S., & Glover, S. C. O. 2017, *MNRAS*, **466**, 407
- Stahler, S., & Palla, F. 2008, *The Formation of Stars* (Wiley)
- Truelove, J. K., Klein, R. I., McKee, C. F., et al. 1997, *ApJLetters*, **489**, L179
- Turk, M. J., Smith, B. D., Oishi, J. S., et al. 2011, *ApJS*, **192**, 9
- Weingartner, J. C., & Draine, B. T. 2001, *ApJS*, **134**, 263
- Wisdom, J., & Holman, M. 1991, *AJ*, **102**, 1528
- Wolfire, M. G., McKee, C. F., Hollenbach, D., & Tielens, A. G. G. M. 2003, *ApJ*, **587**, 278
- Wünsch, R. 2015, personal communication
- Yoshida, H. 1990, *Physics Letters A*, **150**, 262