

AN EXACT INTEGRATION SCHEME FOR RADIATIVE COOLING IN HYDRODYNAMICAL SIMULATIONS

R. H. D. TOWNSEND

Department of Astronomy, University of Wisconsin–Madison, 5534 Sterling Hall, 475 N. Charter Street, Madison, WI 53706, USA; townsend@astro.wisc.edu

Received 2008 September 22; accepted 2009 January 19; published 2009 March 12

ABSTRACT

A new scheme for incorporating radiative cooling in hydrodynamical codes is presented, centered around exact integration of the governing semidiscrete cooling equation. Using benchmark calculations based on the cooling downstream of a radiative shock, I demonstrate that the new scheme outperforms traditional explicit and implicit approaches in terms of accuracy, while remaining competitive in terms of execution speed.

Key words: hydrodynamics – methods: numerical – radiation mechanisms: thermal – shock waves

1. INTRODUCTION

Cooling by optically thin radiative emission plays an important role in many different types of astrophysical flows. This is especially true of radiative shocks, which arise in a wide variety of contexts (e.g., Strickland & Blondin 1995; Mignone 2005, and references therein). In this paper, I present a new scheme for incorporating radiative cooling in hydrodynamical codes, centered around exact integration (EI) of the governing cooling equation.

To lay the necessary groundwork for the subsequent discussion, Section 2 presents a derivation of the semi- and fully discrete forms for the cooling equation. Section 3 then reviews various schemes used to solve this equation, culminating with the introduction of the new EI scheme. These schemes are benchmarked in Section 4 to explore the relative trade-offs between accuracy and execution speed, and I conclude with brief remarks in Section 5.

2. THE COOLING EQUATION

The hydrodynamical equation of energy conservation for an ideal gas can be written as

$$\frac{dP}{dt} - \frac{\gamma P}{\rho} \frac{d\rho}{dt} = -(\gamma - 1)n_e n_H \Lambda(T). \quad (1)$$

Here, P is the pressure, ρ is the density, T is the temperature, γ is the ratio of specific heats, n_e and n_H are the electron and hydrogen number densities, respectively, and d/dt denotes the Lagrangian (total) time derivative. The function $\Lambda(T)$ represents the electron cooling efficiency, and is typically obtained in tabular form from detailed modeling (see, e.g., Raymond et al. 1976; Sutherland & Dopita 1993; Gnat & Sternberg 2007). The dependence of Λ on temperature alone is ultimately what makes the EI scheme possible, but is also somewhat of an idealization of the underlying physics. More sophisticated treatments incorporate additional explicit dependences on ionization balance, by tracking a time-varying network of ionic abundances (see, e.g., Raga et al. 2000; Mignone et al. 2007). It is not yet clear how the EI scheme might be extended to these treatments.

In Eulerian-based, finite-difference hydrodynamic codes, it is common to implement energy conservation (1) using an *operator splitting* approach. The rate of pressure change is divided into

a component associated with adiabatic expansion/contraction

$$\left. \frac{dP}{dt} \right|_{\text{ad}} = \frac{\gamma P}{\rho} \frac{d\rho}{dt}, \quad (2)$$

and a component associated with radiative cooling,

$$\left. \frac{dP}{dt} \right|_{\text{cool}} = -(\gamma - 1)n_e n_H \Lambda(T). \quad (3)$$

The pressure change due to the adiabatic component (2) is typically applied in the advection stage of the code, during which the density and velocity are also updated in accordance with the mass and momentum conservation equations.¹ The pressure change due to the cooling component (3) is then applied in a subsequent stage, during which the density is held constant (e.g., Mignone 2005). The isochoric nature of this latter stage does not preclude simulations of isobaric systems such as cooling flows (e.g., Peterson & Fabian 2006); in these cases, the components (2) and (3) are equal and opposite, resulting in no net pressure variations.

The ideal gas law

$$P = nkT \quad (4)$$

is used to recast the cooling equation (3) in terms of temperature; here, n is the total number density of particles, and k is Boltzmann’s constant. The mean molecular weight $\mu \equiv \rho/n$ is assumed to remain constant; as Gnat & Sternberg (2007) argue, this is a reasonable approximation for temperatures $\gtrsim 10^4$ K (although it can break down in circumstances where departures from ionization equilibrium are significant; see Teşileanu et al. 2008). The cooling equation then becomes

$$\frac{dT}{dt} = -\frac{(\gamma - 1)\rho\mu}{k\mu_e\mu_H}\Lambda(T), \quad (5)$$

where $\mu_e \equiv \rho/n_e$ and $\mu_H \equiv \rho/n_H$ are the effective molecular weights per electron and per hydrogen atom/ion. In the regime of full ionization, the molecular weights appearing in this expression are given by

$$\mu = \frac{u}{2X + 3(1 - X - Z)/4 + Z/2}, \quad (6)$$

¹ By maintaining an *adiabatic* advection stage, it remains possible to use numerical schemes derived from Godunov’s (1959) characteristic-based approach, such as the popular Piecewise Parabolic Method (PPM) of Colella & Woodward (1984).

$$\mu_e = \frac{2u}{1+X}, \quad (7)$$

$$\mu_H = \frac{u}{X}; \quad (8)$$

here, X and Z are the usual hydrogen and metal mass fractions, and u is the atomic mass unit.

Implementing Equation (5) in a hydrodynamical code requires discretization in space and perhaps also time. For simplicity, I focus here on a zeroth-order finite-volume spatial discretization,² which leads to the semidiscrete cooling equation

$$\frac{d\bar{T}_i}{dt} = -\frac{(\gamma-1)\bar{\rho}_i\mu}{k\mu_e\mu_H}\Lambda(\bar{T}_i). \quad (9)$$

Here, $\bar{\rho}_i$ is the volume-averaged density in the numerical zone with integer index i , while \bar{T}_i is the corresponding zone temperature, calculated from $\bar{\rho}_i$ and the volume-averaged pressure \bar{P}_i using the ideal-gas law (4).

The following sections discuss various approaches to solving this equation across a discrete time step Δt . The explicit (Section 3.1) and implicit (Section 3.2) schemes are all based on a fully discrete cooling equation, derived from the semidiscrete form (9) by replacing the temperature rate-of-change with a finite difference:

$$\frac{\bar{T}_i^{n+1} - \bar{T}_i^n}{\Delta t} = -\frac{(\gamma-1)\bar{\rho}_i\mu}{k\mu_e\mu_H}\Lambda(\bar{T}_i). \quad (10)$$

The superscripts n and $n+1$ indicate values at consecutive times t^n and $t^{n+1} \equiv t^n + \Delta t$. Because the finite difference is centered, this equation is second-order accurate in time. To evaluate the cooling efficiency $\Lambda(\bar{T}_i)$ on the right-hand side, either the initial or the updated temperature may be used; the choice differentiates *explicit* schemes from *implicit* schemes.

3. SOLVING THE COOLING EQUATION

3.1. Explicit Schemes

In an explicit scheme, the cooling efficiency in Equation (10) is evaluated using the initial temperature \bar{T}_i^n :

$$\frac{\bar{T}_i^{n+1} - \bar{T}_i^n}{\Delta t} = -\frac{(\gamma-1)\bar{\rho}_i\mu}{k\mu_e\mu_H}\Lambda(\bar{T}_i^n). \quad (11)$$

This is now first-order accurate in time, because the right-hand side is not centered in the interval (t^n, t^{n+1}) . Solving for the updated temperature,

$$\bar{T}_i^{n+1} = \bar{T}_i^n \left[1 - \frac{\Delta t}{t_{\text{cool}}} \right], \quad (12)$$

where

$$t_{\text{cool}} \equiv \frac{k\mu_e\mu_H\bar{T}_i^n}{(\gamma-1)\bar{\rho}_i\mu\Lambda(\bar{T}_i^n)} \quad (13)$$

is the single-point cooling time. This solution, together with the ideal-gas law (4), allows the pressure in each zone to be updated across the time step Δt .

The behavior of the explicit scheme (12) is investigated by calculating the updated temperature \bar{T}_i^{n+1} as a function of

time step, for three different choices of the initial temperature, $\bar{T}_i^n = (10^6 \text{ K}, 10^7 \text{ K}, 10^8 \text{ K})$. The cooling efficiency is obtained from a piecewise power-law fit to the collisional ionization equilibrium (CIE) values tabulated by Gnat & Sternberg (2007). Because the tabulation is truncated at 10^4 K , this temperature is imposed as floor on \bar{T}_i^{n+1} (in an actual simulation, this floor temperature might correspond to the reheating effects of a nearby star). A monatomic gas ($\gamma = 5/3$) and solar abundances ($X = 0.7$, $Z = 0.02$) are assumed here and throughout. The upper panels of Figure 1 plot the results from these calculations. By way of comparison, the panels also show the exact solutions to the semidiscrete cooling equation (9); Section 3.3 discusses how these solutions are obtained.

For Δt approaching t_{cool} , the explicit scheme leads to updated temperatures that depart quite significantly from the exact values. Put simply, this is because the cooling efficiency is fixed at its initial value $\Lambda(\bar{T}_i^n)$, rather than being allowed to evolve in response to the cooling process. This difficulty can be avoided by dividing the time step Δt (typically set during the advection stage by the Courant–Friedrichs–Lewy (CFL) criterion; see Section 4) into a sequence of smaller substeps (see, e.g., Plewa & Rozyczka 1992), and applying Equation (12) multiple times. Alternatively, a higher-order temporal discretization of the cooling equation is possible; for instance, a second-order Runge–Kutta method has

$$\bar{T}_i^{n+1/2} = \bar{T}_i^n \left[1 - \frac{1}{2} \frac{\Delta t}{t_{\text{cool}}} \right], \quad (14)$$

$$\bar{T}_i^{n+1} = \bar{T}_i^n \left[1 - \frac{\Lambda(\bar{T}_i^{n+1/2})}{\Lambda(\bar{T}_i^n)} \frac{\Delta t}{t_{\text{cool}}} \right]. \quad (15)$$

The lower panels of Figure 1 plot the updated temperatures calculated using this second-order scheme. There is a clear improvement over the first-order approach, and further improvements can be gained by moving to even higher orders (e.g., Sutherland et al. 2003). However, with each order added an additional evaluation of the cooling efficiency is required; hence, the computational costs necessarily escalate.

3.2. Implicit Schemes

To overcome the drawbacks of explicit schemes when $\Delta t \gtrsim t_{\text{cool}}$, a number of authors (e.g., Strickland & Blondin 1995; Stone et al. 1997; Pittard et al. 2004) instead opt for an implicit scheme. The cooling efficiency in Equation (10) is then evaluated using the updated temperature \bar{T}_i^{n+1} :

$$\frac{\bar{T}_i^{n+1} - \bar{T}_i^n}{\Delta t} = -\frac{(\gamma-1)\bar{\rho}_i\mu}{k\mu_e\mu_H}\Lambda(\bar{T}_i^{n+1}). \quad (16)$$

The solution can be written in a standard form similar to the explicit case (see Equation (12)),

$$\bar{T}_i^{n+1} = \bar{T}_i^n \left[1 - \frac{\Lambda(\bar{T}_i^{n+1})}{\Lambda(\bar{T}_i^n)} \frac{\Delta t}{t_{\text{cool}}} \right], \quad (17)$$

but the appearance of \bar{T}_i^{n+1} on the right-hand side means that this equation must now be solved numerically, typically using a root-finding algorithm.

Figure 2 investigates the behavior of this implicit scheme, plotting \bar{T}_i^{n+1} as a function of Δt for the same parameters as in Figure 1. The upper panels use a secant algorithm to solve Equation (17), with a fallback to bisection when the most recent

² See Strickland & Blondin (1995) for a demonstration of how a higher-order discretization might be constructed.

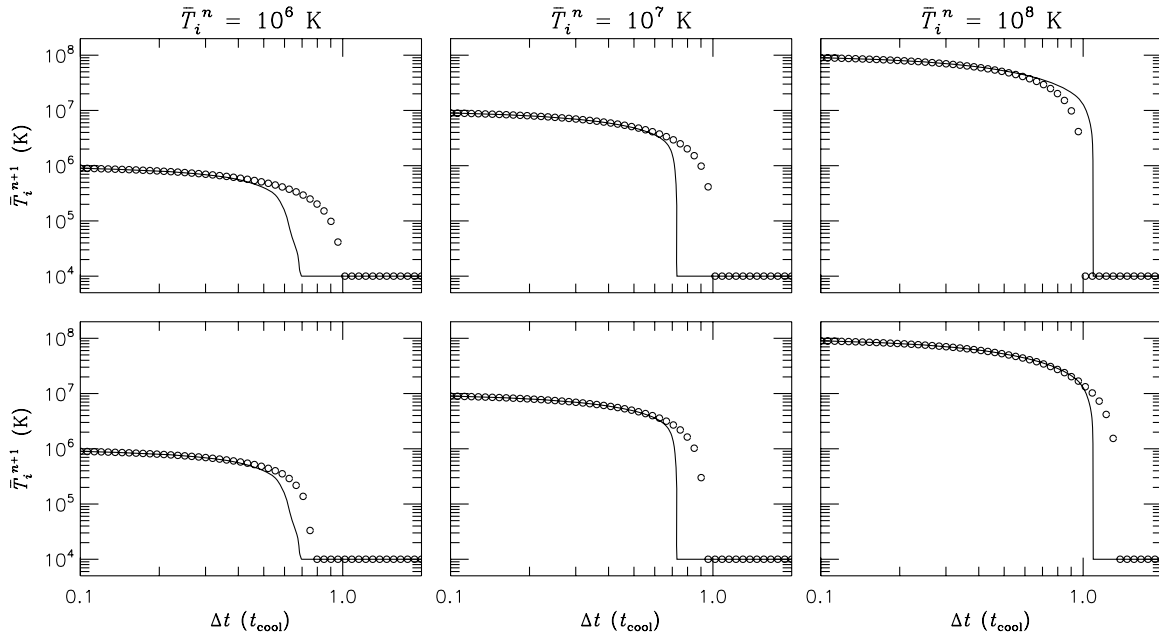


Figure 1. Updated temperature \bar{T}_i^{n+1} plotted as a function of time step Δt (in units of the cooling time t_{cool}), for three differing choices (left to right) of initial temperature \bar{T}_i^n . The circles in the top (bottom) panels indicate values calculated using the first-order (second-order) explicit scheme. The solid lines show the corresponding exact solutions.

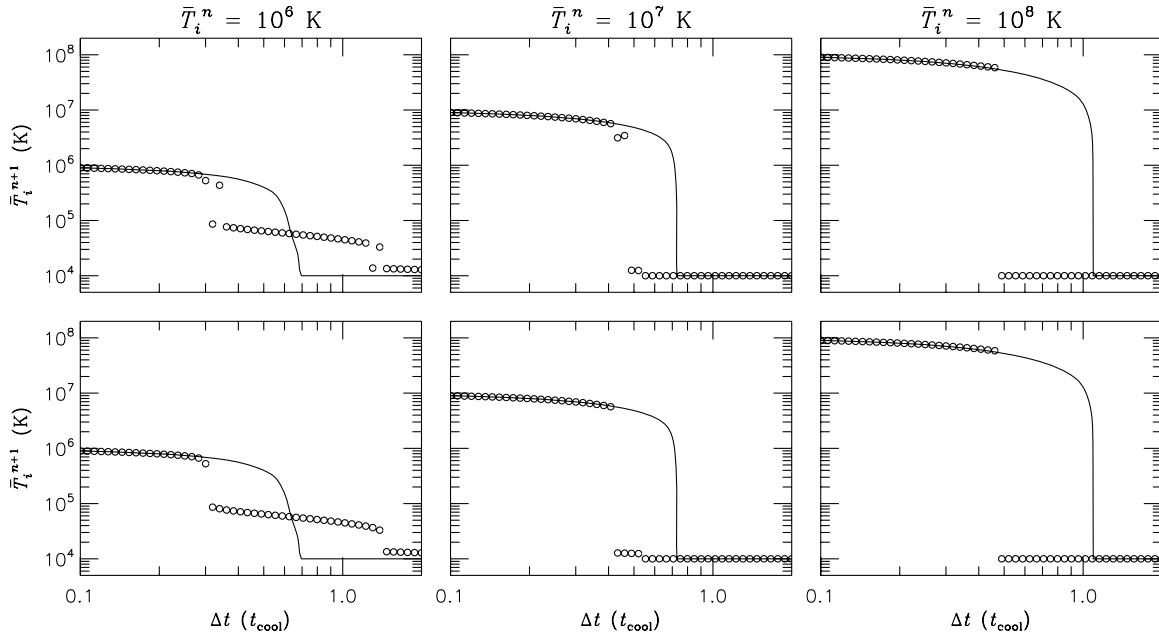


Figure 2. As in Figure 1, except that the circles in the top (bottom) panels now indicate values calculated using the secant (Brent) first-order implicit scheme.

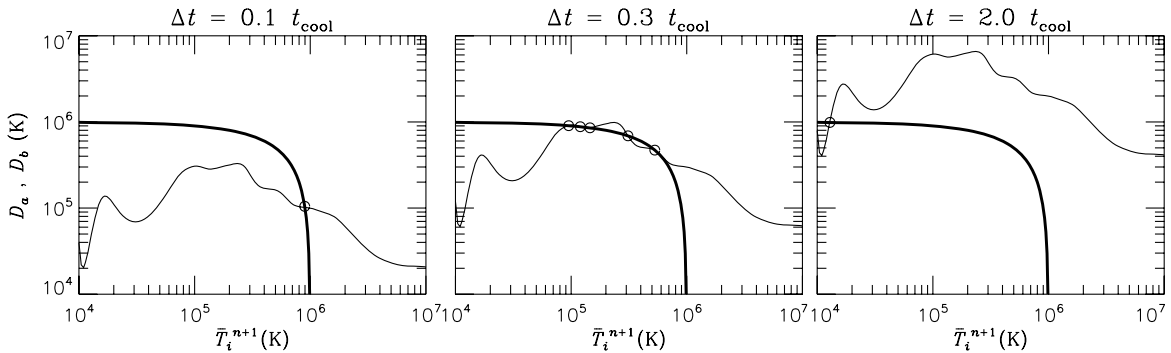


Figure 3. Discriminants D_a (thick) and D_b (thin) plotted as a function of updated temperature \bar{T}_i^{n+1} , for three differing choices (left to right) of the time step Δt . Intersections between the two curves are highlighted by circles.

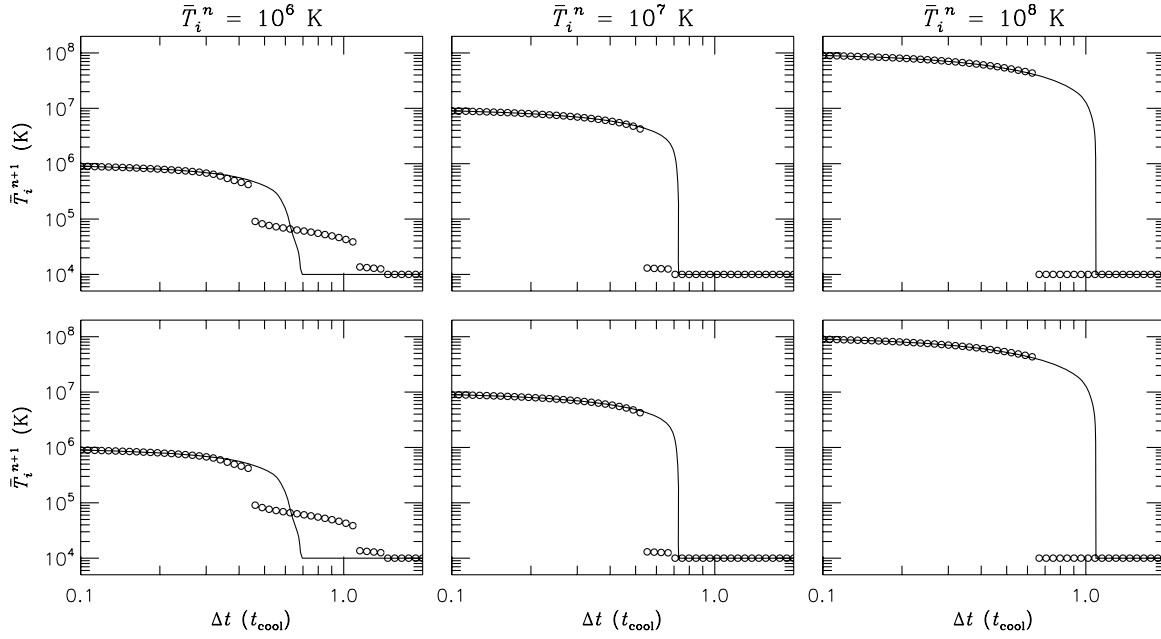


Figure 4. As in Figure 1, except that the circles in the top (bottom) panels now indicate values calculated using the secant (Brent) second-order implicit scheme.

iterate for \bar{T}_i^{n+1} falls outside the bounds of the $\Lambda(T)$ tabulation. Conversely, the lower panels use Brent's algorithm (Press et al. 1992). In both cases, solutions are iterated until the fractional change in \bar{T}_i^{n+1} drops below 10^{-4} .

The figure reveals problems with the implicit schemes. For instance, in the $\bar{T}_i^n = 10^6$ K case, \bar{T}_i^{n+1} is significantly underestimated for $0.3 t_{\text{cool}} \lesssim \Delta t \lesssim 0.7 t_{\text{cool}}$, and overestimated for $\Delta t \gtrsim 0.7 t_{\text{cool}}$. Moreover, rather than varying smoothly as Δt is increased (as one might hope for a stable scheme), \bar{T}_i^{n+1} exhibits abrupt jumps.

To explore the origin of these jumps, I introduce the twin discriminants

$$D_a = \bar{T}_i^n - \bar{T}_i^{n+1} \quad (18)$$

and

$$D_b = \bar{T}_i^n \frac{\Lambda(\bar{T}_i^{n+1})}{\Lambda(\bar{T}_i^n)} \frac{\Delta t}{t_{\text{cool}}}, \quad (19)$$

such that the implicit equation (17) corresponds to the condition $D_a = D_b$. Figure 3 plots the discriminants together as a function of \bar{T}_i^{n+1} , for $\bar{T}_i^n = 10^6$ K and three choices of time step. The middle, $\Delta t = 0.3 t_{\text{cool}}$ panel shows that the curves intersect multiple times, corresponding to multiple, distinct solutions (in this case, five) to the implicit equation. Abrupt switching between these solutions, due both to the convergence behavior of the particular root-finding algorithm, and to the appearance or disappearance of solutions as Δt is varied, is responsible for the jumps seen in Figure 2.

To underscore further that solution jumping is an intrinsic property of implicit schemes, Figure 4 illustrates solutions of the cooling equation using the Crank–Nicholson method,

$$\frac{\bar{T}_i^{n+1} - \bar{T}_i^n}{\Delta t} = -\frac{(\gamma - 1)\bar{\rho}_i \mu}{k\mu_e \mu_H} \frac{\Lambda(\bar{T}_i^{n+1}) + \Lambda(\bar{T}_i^n)}{2}, \quad (20)$$

which is now second-order implicit. In the standard form, this becomes

$$\bar{T}_i^{n+1} = \bar{T}_i^n \left[1 - \frac{\Lambda(\bar{T}_i^{n+1}) + \Lambda(\bar{T}_i^n)}{2\Lambda(\bar{T}_i^n)} \frac{\Delta t}{t_{\text{cool}}} \right]. \quad (21)$$

While the data plotted in the figure differ from the first-order cases shown in Figure 2, they still exhibit abrupt jumps arising from the existence of multiple solutions—although the range of Δt values over which jumping occurs is somewhat reduced.

In spite of these various issues, implicit schemes have proven popular in the literature. This stems in part from their reputation for stability; for instance, Strickland & Blondin (1995) remark that their implicit cooling scheme “*is unconditionally stable*” (words subsequently echoed by Pittard et al. 2004); likewise, Stone et al. (1997) state that their scheme “*is stable even when the cooling time is much less than the dynamical time.*” However, this confidence appears misplaced. While implicit schemes are stable when used to solve linear equations (as can be demonstrated through a von Neumann stability analysis; see, e.g., Press et al. 1992), this property does not necessarily extend to *nonlinear* systems such as the semidiscrete cooling equation (9). Moreover, even if stability can be established for a given scheme, there are no corresponding guarantees of accuracy or convergence—and it is in these latter capacities that the implicit schemes reviewed here fall short. The jumping between solutions is particularly problematic, because it can cause two neighboring zones with very similar initial states to cool to quite different temperatures. This will establish a strong pressure differential between the zones, in turn generating spurious fluid flows and/or waves.

3.3. The Exact Integration Scheme

The new cooling scheme introduced here avoids the various difficulties outlined above, by going back to the semidiscrete cooling equation (9) and solving it exactly. The equation is first rearranged as

$$\frac{d\bar{T}_i}{\Lambda(\bar{T}_i)} = -\frac{(\gamma - 1)\bar{\rho}_i \mu}{k\mu_e \mu_H} dt, \quad (22)$$

and then integrated across a time step,

$$\int_{\bar{T}_i^n}^{\bar{T}_i^{n+1}} \frac{d\bar{T}_i}{\Lambda(\bar{T}_i)} = -\frac{(\gamma - 1)\bar{\rho}_i \mu}{k\mu_e \mu_H} \Delta t. \quad (23)$$

The dimensionless “temporal evolution function” (TEF) is then introduced as

$$Y(T) = \frac{\Lambda(T_{\text{ref}})}{T_{\text{ref}}} \int_T^{T_{\text{ref}}} \frac{dT'}{\Lambda(T')}, \quad (24)$$

where T_{ref} is an arbitrary reference temperature; the TEF represents a normalized measure of the total time taken to cool from T_{ref} to T . With this definition, the integrated cooling equation (23) becomes

$$\frac{T_{\text{ref}}}{\bar{T}_i^n} \frac{\Lambda(\bar{T}_i^n)}{\Lambda(T_{\text{ref}})} [Y(\bar{T}_i^n) - Y(\bar{T}_i^{n+1})] = -\frac{\Delta t}{t_{\text{cool}}}, \quad (25)$$

where t_{cool} has the same definition as before, and the consequent solution is

$$\bar{T}_i^{n+1} = Y^{-1} \left[Y(\bar{T}_i^n) + \frac{\bar{T}_i^n}{T_{\text{ref}}} \frac{\Lambda(T_{\text{ref}})}{\Lambda(\bar{T}_i^n)} \frac{\Delta t}{t_{\text{cool}}} \right]. \quad (26)$$

This result is exact, but requires construction of the TEF and its inverse from $\Lambda(T)$. The Appendix presents analytic expressions for $Y(T)$ and its inverse in the common cases where the cooling efficiency is represented by a power law (Section A.1) and a piecewise power law (Section A.2). The cooling efficiencies used in Figures 1 and 2 fall into the latter category, and the exact solutions plotted in these figures are calculated using the EI scheme described here. Likewise, the cooling efficiency assumed by Mignone (2005) falls into the former category, and in fact his analytic cooling scheme (which foreshadows the present paper) can be derived from this EI formalism.

4. BENCHMARKS

The preceding sections (and in particular, Figures 1 and 2) demonstrate that both explicit and implicit schemes for solving the cooling equation can become inaccurate as the time step approaches the cooling time t_{cool} ; in contrast, the EI scheme gives the exact solution for any value of Δt . However, an important caveat here is that the time step is itself constrained by numerical considerations in the advection stage. Efficiency dictates that Δt be chosen as large as possible (subject to accuracy requirements), but for stability reasons it cannot exceed the limit established by the CFL criterion (see, e.g., Laney 1998).

To explore how the different cooling schemes perform with a CFL-based time step, I consider the problem of a steady,³ one-dimensional radiative shock characterized by an upstream density ρ_{in} , Mach number \mathcal{M}_{in} , and temperature T_{in} . For various combinations of these parameters (to be discussed below), each scheme is benchmarked by repeating the following steps.

1. The run of density and pressure throughout the postshock cooling region are calculated using the approach described by Strickland & Blondin (1995, their Section 4.1). This region is bounded on the upstream side by the shock itself, and on the downstream side by the condition $T = T_{\text{in}}$ (i.e., the gas has cooled back down to its initial temperature).
2. The cooling region is discretized into N equal-sized zones; for each zone, the volume-averaged density $\bar{\rho}_i$ and pressure \bar{P}_i are evaluated, and the corresponding temperature \bar{T}_i is calculated using the ideal-gas law (4).

³ In reality, radiative shocks are often time variable due to the cooling instability discovered by Langer et al. (1981); however, this variability is ignored here since the principal criterion is a well-defined test system, even if it is somewhat idealized.

Table 1
Benchmark Results

Cooling Scheme	$N = 1$		$N = 10$		$N = 100$	
	ε (%)	τ (ns)	ε (%)	τ (ns)	ε (%)	τ (ns)
$\mathcal{M}_{\text{in}} = 3$						
First-order explicit	4.0	95	14.1	111	0.2	100
Second-order explicit	4.0	213	5.9	201	0.0	181
First-order implicit (secant)	26.1	844	4.2	484	0.2	362
First-order implicit (Brent)	26.1	903	4.2	787	0.2	562
Second-order implicit (secant)	22.0	931	2.4	837	0.0	612
Second-order implicit (Brent)	22.0	927	2.4	837	0.0	613
Exact	0.0	213	0.0	192	0.0	173
$\mathcal{M}_{\text{in}} = 10$						
First-order explicit	30.8	94	7.6	112	0.3	101
Second-order explicit	3.0	215	0.4	204	0.0	183
First-order implicit (secant)	24.0	906	6.5	548	0.2	383
First-order implicit (Brent)	24.0	972	6.5	778	0.2	572
Second-order implicit (secant)	12.2	1230	1.3	875	0.0	640
Second-order implicit (Brent)	12.2	1248	1.3	874	0.0	639
Exact	0.0	215	0.0	197	0.0	173
$\mathcal{M}_{\text{in}} = 100$						
First-order explicit	38.0	92	1.8	107	0.1	98
Second-order explicit	12.1	207	0.1	204	0.0	181
First-order implicit (secant)	99.9	266	1.2	523	0.1	387
First-order implicit (Brent)	99.9	258	1.2	725	0.1	544
Second-order implicit (secant)	99.9	315	0.3	796	0.0	598
Second-order implicit (Brent)	99.9	317	0.3	799	0.0	600
Exact	0.0	214	0.0	195	0.0	169

3. The CFL time step is calculated as $\Delta t = \Delta x / c_{\text{max}}$, where Δx is the spatial extent of the zones, and $c_{\text{max}} \equiv \max(\sqrt{\gamma \bar{P}_i / \bar{\rho}_i})$ is the maximum value of the adiabatic sound speed over all zones composing the cooling region.
4. For each zone, the updated temperature \bar{T}_i^{n+1} is evaluated using one of the cooling schemes discussed in the preceding sections. This step is repeated five times, and the average CPU execution time τ (per zone, per repeat) is recorded.
5. The updated temperatures are compared with the exact values $\bar{T}_{i,\text{EI}}^{n+1}$ that result from using the EI scheme; the maximum relative error

$$\varepsilon = \max(|\bar{T}_i^{n+1} - \bar{T}_{i,\text{EI}}^{n+1}| / \bar{T}_{i,\text{EI}}^{n+1}) \quad (27)$$

is recorded.

To cover a representative region of parameter space, I consider three values $\mathcal{M}_{\text{in}} = 3, 10$, and 100 of the Mach number (corresponding to mild, moderate, and strong shocks), and three values $N = 1, 10$, and 100 of the zone count (corresponding to poor, moderate, and good resolution of the shocks). An upstream density $\rho_{\text{in}} = 10^{-15} \text{ g cm}^{-3}$ is assumed throughout, but results do not depend at all on this value. With these choices of parameters, and for each of the cooling schemes considered previously, Table 1 shows the error ε and execution time τ obtained by following the steps above. All calculations were undertaken on a single core of an Intel E5345 quad-core CPU running at 2.33 GHz.

The table reveals a general trend that the error decreases as the zone count increases. When the shock is resolved by only a single zone, the error tends to be large (with the obvious exception of the EI scheme, for which $\varepsilon = 0$ always). For $N = 10$, ε is below 10% in all but one case; and by $N = 100$, it is below 1% in all cases. To explain this trend, the definition

of the CFL time step is used to write

$$\frac{\Delta t}{t_{\text{cool}}} = \frac{\Delta x}{c_{\text{max}} t_{\text{cool}}}. \quad (28)$$

Because the flow downstream of the radiative shock is subsonic, it follows that

$$\frac{\Delta t}{t_{\text{cool}}} \lesssim \frac{\Delta x}{v t_{\text{cool}}}, \quad (29)$$

where $v \lesssim c_{\text{max}}$ is the typical flow velocity in the cooling region. Recognizing that $v t_{\text{cool}}$ approximates the spatial extent of this region, the corollary is that

$$\frac{\Delta t}{t_{\text{cool}}} \lesssim \frac{1}{N}. \quad (30)$$

Thus, the limit $N \gg 1$ implies that $\Delta t \ll t_{\text{cool}}$, which favors accurate cooling irrespective of the choice of scheme. Turning this statement around, all of the cooling schemes apart from the EI scheme tend to be inaccurate when the cooling region is poorly resolved. Of course, this applies only when Δt is tied solely to the CFL time step. It is often desirable to place further constraints on Δt , over and beyond that given by the CFL criterion. For instance, to improve the coupling between thermal and hydrodynamical evolution, Δt can be limited so that the anticipated temperature/pressure change during cooling does not for any zone exceed a specified fraction of its initial value. The price paid in this approach is the greater number of steps that must be taken to cross a given time interval.

Looking now at the relative performance of the different schemes, the τ data in Table 1 reveal that the first-order explicit scheme is the fastest, with an average execution time of ~ 100 ns per zone. The second-order explicit scheme and the EI scheme are both only about a factor of 2 slower than this, with the latter slightly beating the former in all but two tests. The implicit schemes are in every case the slowest, ranging from around 2.5 up to 12 times slower than the first-order explicit scheme.

5. CONCLUDING REMARKS

A criticism that might be leveled at the EI scheme is that it requires the reciprocal of the cooling efficiency, $1/\Lambda(T)$, be analytically integrable. In practice, this is rarely an issue; the most common representations of $\Lambda(T)$ are piecewise power-law or piecewise polynomial fits to detailed models, both of which meet this restriction. In any case, it is always possible to re-fit arbitrary cooling efficiency data with a conforming representation.

The principal strengths of the EI scheme are twofold. On the one hand, it produces exact solutions to the semidiscrete cooling equation (9), irrespective of whether the time step is small or large compared to the cooling time t_{cool} . On the other, it remains very competitive in terms of execution speed, being only two times slower than the (fastest, yet often inaccurate) first-order explicit scheme. While more sophisticated cooling treatments that track ionic abundances (e.g., Mignone et al. 2007) will remain the state-of-the-art in terms of physical fidelity, the strengths of the EI scheme naturally recommend it as the cooling scheme of choice in any hydrodynamical code where a simple, fast, and robust treatment of optically thin radiative losses is desired.

My thanks go to Stan Owocki, for many useful discussions that led to the genesis of the paper, and to the anonymous

referees for their very helpful remarks. I moreover acknowledge support from NASA *Long Term Space Astrophysics* grant NNG05GC36G and NSF grant AST-0507581.

Note Added in Proof. Thomas, Fabian, & Nulsen (1987, MNRAS, 228, 973) incorporate radiative losses in their steady-state models for cooling flows using an integration scheme similar to the EI scheme presented here. I am grateful to these authors for bringing the similarity to my attention.

APPENDIX A

TEMPORAL EVOLUTION FUNCTIONS

A.1. Power Law

I first consider the simple case of a power-law cooling function,

$$\Lambda(T) = \Lambda_{\text{ref}} \left(\frac{T}{T_{\text{ref}}} \right)^{\alpha}, \quad (A1)$$

where T_{ref} is the reference temperature introduced in Section 3.3, and Λ_{ref} and α are constant coefficients. Substituting this into Equation (24) leads to a TEF

$$Y(T) = \begin{cases} \frac{1}{1-\alpha} \left[1 - \left(\frac{T_{\text{ref}}}{T} \right)^{\alpha-1} \right] & \alpha \neq 1, \\ \ln \left(\frac{T_{\text{ref}}}{T} \right) & \alpha = 1. \end{cases} \quad (A2)$$

The corresponding inverse TEF is given by

$$Y^{-1}(Y) = \begin{cases} T_{\text{ref}} [1 - (1 - \alpha)Y]^{1/(1-\alpha)} & \alpha \neq 1, \\ T_{\text{ref}} \exp(-Y) & \alpha = 1. \end{cases} \quad (A3)$$

A.2. Piecewise Power Law

More physically realistic cooling functions are often represented by piecewise power-law fits to detailed models (e.g., Walder & Folini 1996; Kimoto & Chernoff 1997; Caunt & Korpi 2001; Townsend et al. 2007). I assume a fit parameterization of the form

$$\Lambda(T) = \Lambda_k \left(\frac{T}{T_k} \right)^{\alpha_k} \quad T_k \leq T \leq T_{k+1}, \quad (A4)$$

for a set of $N - 1$ temperature intervals (T_k, T_{k+1}) ($k = 1, 2, \dots, N - 1$) and coefficient pairs $\{\Lambda_k, \alpha_k\}$. Substituting this into Equation (24), with a reference temperature chosen as $T_{\text{ref}} = T_N$, leads to the piecewise TEF

$$Y(T) = Y_k + \begin{cases} \frac{1}{1-\alpha_k} \frac{\Lambda_N}{\Lambda_k} \frac{T_k}{T_N} \left[1 - \left(\frac{T_k}{T} \right)^{\alpha_k-1} \right] & \alpha_k \neq 1 \\ \frac{\Lambda_N}{\Lambda_k} \frac{T_k}{T_N} \ln \left(\frac{T_k}{T} \right) & \alpha_k = 1 \end{cases} \quad T_k \leq T \leq T_{k+1}, \quad (A5)$$

where $\Lambda_N \equiv \Lambda_{N-1}(T_N/T_{N-1})^{\alpha_{N-1}}$. The coefficients $Y_k = Y(T_k)$ are constants of integration; the requirement that $Y(T)$ be continuous dictates that

$$Y_k = Y_{k+1} - \begin{cases} \frac{1}{1-\alpha_k} \frac{\Lambda_N}{\Lambda_k} \frac{T_k}{T_N} \left[1 - \left(\frac{T_k}{T_{k+1}} \right)^{\alpha_k-1} \right] & \alpha_k \neq 1, \\ \frac{\Lambda_N}{\Lambda_k} \frac{T_k}{T_N} \ln \left(\frac{T_k}{T_{k+1}} \right) & \alpha_k = 1. \end{cases} \quad (A6)$$

This recurrence can be started by noting that $Y_N = Y(T_{\text{ref}}) = 0$ (see Equation (24)). The inverse TEF is given in each (Y_k, Y_{k+1}) interval by

$$Y^{-1}(Y) = \begin{cases} T_k \left[1 - (1 - \alpha_k) \frac{\Lambda_k}{\Lambda_N} \frac{T_N}{T_k} (Y - Y_k) \right]^{1/(1-\alpha_k)} & \alpha_k \neq 1 \\ T_k \exp \left[-\frac{\Lambda_k}{\Lambda_N} \frac{T_N}{T_k} (Y - Y_k) \right] & \alpha_k = 1 \end{cases} \quad Y_k \leq Y \leq Y_{k+1}. \quad (\text{A7})$$

REFERENCES

- Caunt, S. E., & Korpi, M. J. 2001, [A&A](#), **369**, 706
- Colella, P., & Woodward, P. R. 1984, [J. Comp. Phys.](#), **54**, 174
- Gnat, O., & Sternberg, A. 2007, [ApJS](#), **168**, 213
- Godunov, S. K. 1959, *Math. Sbornik*, **47**, 271
- Kimoto, P. A., & Chernoff, D. F. 1997, [ApJ](#), **485**, 274
- Laney, C. B. 1998, in *Computational Gasdynamics* (Cambridge: Cambridge Univ. Press)
- Langer, S. H., Chanmugam, G., & Shaviv, G. 1981, [ApJ](#), **245**, L23
- Mignone, A. 2005, [ApJ](#), **626**, 373
- Mignone, A., Bodo, G., Massaglia, S., Matsakos, T., Tesileanu, O., Zanni, C., & Ferrari, A. 2007, [ApJS](#), **170**, 228
- Peterson, J. R., & Fabian, A. C. 2006, [Phys. Rep.](#), **427**, 1
- Pittard, J. M., Hartquist, T. W., Ashmore, I., Byfield, A., Dyson, J. E., & Falle, S. A. E. G. 2004, [A&A](#), **414**, 399
- Plewa, T., & Rozyczka, M. 1992, *Acta Astron.*, **42**, 295
- Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. 1992, *Numerical Recipes in Fortran* (2nd ed.; Cambridge: Cambridge Univ. Press)
- Raga, A. C., Navarro-González, R., & Villagrán-Muniz, M. 2000, *RevMexAA*, **36**, 67
- Raymond, J. C., Cox, D. P., & Smith, B. W. 1976, [ApJ](#), **204**, 290
- Stone, J. M., Xu, J., & Hardee, P. 1997, [ApJ](#), **483**, 136
- Strickland, R., & Blondin, J. M. 1995, [ApJ](#), **449**, 727
- Sutherland, R. S., Bicknell, G. V., & Dopita, M. A. 2003, [ApJ](#), **591**, 238
- Sutherland, R. S., & Dopita, M. A. 1993, [ApJS](#), **88**, 253
- Teşileanu, O., Mignone, A., & Massaglia, S. 2008, [A&A](#), **488**, 429
- Townsend, R. H. D., Owocki, S. P., & Ud-Doula, A. 2007, [MNRAS](#), **382**, 139
- Walder, R., & Folini, D. 1996, [A&A](#), **315**, 265