High order well-balanced finite volume methods for multi-dimensional systems of hyperbolic balance laws

Jonas P. Berberich [a],[*], Praveen Chandrashekar [b], Christian Klingenberg [a]

[a] *Department of Mathematics, University of Würzburg, Emil-Fischer-Straße 40, 97074 Würzburg, Germany*
[b] *Centre for Applicable Mathematics, Tata Institute of Fundamental Research, Bengaluru-560065, India*

ARTICLE INFO

ABSTRACT

We introduce a general framework for the construction of well-balanced finite volume methods for hyperbolic balance laws. We use the phrase *well-balancing* in a broader sense, since our proposed method can be applied to exactly follow any solution of any system of hyperbolic balance laws in multiple spatial dimensions and not only time independent solutions. The solution has to be known a priori, either as an analytical expression or as discrete data. The proposed framework modifies the standard finite volume approach such that the well-balancing property is obtained and in case the method is high order accurate, this is maintained under our modification. We present numerical tests for the compressible Euler equations with and without gravity source term and with different equations of state, and for the equations of compressible ideal magnetohydrodynamics.

© 2021 Elsevier Ltd. All rights reserved.

1. Introduction

Several problems in engineering and science are modeled by conservation principles and lead to non-linear partial differential equation which are hyperbolic. These equations can rarely be solved exactly and we must resort to some form of numerical approximation. One successful numerical approach for solving hyperbolic PDE is the finite volume method based on Godunov's idea [1]. As soon as external forces enter the modeled system, a source term has to be added to the hyperbolic conservation laws turning these into hyperbolic balance laws. Although numerical methods for hyperbolic balance laws might admit some discrete stationary states, they are in general grid dependent and different from the stationary states of the PDEs. This gives rise to the need to develop so-called well-balanced methods, i.e. methods which are designed to be exact on special stationary solutions of the system.

In the well-known shallow water equations with non-flat bottom topography, the most widely considered static state, which is the lake-at-rest solution, can be formulated in a closed form. This favors the construction of well-balanced methods for this system (e.g. [2–8] and references therein) and related systems like the Ripa model ([9,10] and references therein). The relevance for methods for non-static stationary states (e.g. [11]) has been pointed out

in [12]. For the Euler equations with gravitational source term, on the other hand, static solutions have to be found by solving a differential equation for density and pressure together with an equation of state (EoS). This makes the construction of well-balanced methods more challenging and typically restricts the resulting method to some special cases. Many methods have been developed for some classes of hydrostatic states assuming an ideal gas EoS (e.g. [13–21] and references therein) and there are also high order methods, see e.g. [22–24].

While the hydrostatic equation for compressible Euler equations with gravity is basically one-dimensional, the spatial structure of this relation is much richer for compressible ideal magnetohydrodynamics (MHD) equations with gravity since it includes off-diagonal terms. In [25], a well-balanced method for MHD is derived to compute waves on the stationary background. This method is designed to balance isothermal hydrostatic states of the Euler equations together with a magnetic field, which satisfies certain stationarity conditions and is known a priori. Part of this method, namely considering deviations to a background magnetic field, goes back to Tanaka [26] and is also used by Powell et al. [27]. To do so, the background magnetic field is assumed to be static and free of rotation as well as divergence.

There are different approaches to obtain the well-balancing property. Some methods are based on a relaxation approach, in which the hydrostatic equation is included in the relaxation system [28–30]. Path-conservative methods are introduced in [31–33]. Another widespread idea is the application of hydrostatic reconstruction, i.e. the reconstruction of variables which are constant if

---

* Corresponding author.
E-mail addresses: jonas.berberich@mathematik.uni-wuerzburg.de (J.P. Berberich), praveen@tifrbng.res.in (P. Chandrashekar), klingen@mathematik.uni-wuerzburg.de (C. Klingenberg).

the system is in the considered stationary state. An early example of this method is [3] for the shallow water system; for Euler equations, this approach has e.g. been used in [16–18,20]. These methods for Euler equations are restricted to a certain EoS and certain classes of hydrostatic solutions. For astrophysical applications, for example, this restriction is a severe limitation since the equations of state describing physics in the stellar interior are much more complex than the EoS of an ideal gas.

More general methods have been developed in [34–37]. The well-balanced methods introduced in these publications can be applied for any EoS. They are exact on certain hydrostatic solutions, and in all other cases they balance a second order approximation of the considered hydrostatic solution. In [38,39], a second order well-balanced method for Euler equations with gravity is introduced. This method can be applied for any hydrostatic solution of Euler equations with any EoS if the hydrostatic solution is known. The method is then exact up to machine precision and it has been extended to high order in [23]. Notably, there is also an extension to stationary states with non-zero velocity in the same article. Similar techniques can be found in the context of numerical atmospheric modeling (e.g. [17,40,41]). Those well-balanced schemes strongly rely on the structure of the discretized equations or the static solutions to be well-balanced.

The method we present in this paper is designed in the manner of the method in [39]. It uses the idea of hydrostatic reconstruction and a modification of the source term discretization to obtain the well-balanced property. The main point in which it differs from all of the methods mentioned above is that our method is not restricted to a certain system of hyperbolic balance laws. Instead, we present a general framework which modifies finite volume methods for any hyperbolic conservation or balance laws such that they obtain the well-balancing property. Also, the method can be used to balance any known solution which is either given by an analytical expression or as discrete data. Additionally, unlike the majority of the well-balanced methods mentioned above which deal with time independent hydrostatic solutions, our well-balanced method can be used to follow known time-dependent solutions exactly, as we will show in this paper. We will refer to the solution that is to be exactly captured by the scheme as the *target* solution. Depending on the application, the target solution can be a time-independent hydrostatic solution, or some general time-dependent solution of interest. Our method is also general in the sense that it is possible to combine it with other modules of a finite volume scheme: It can be applied on any grid system, with any numerical flux function, reconstruction routine, source term discretization, and ODE solver for time-discretization. It allows for high order in the sense that, if all these components are high order accurate, the resulting method is also high order accurate.

There are several applications in which the hydrostatic solution is known a priori, e.g. in stellar astrophysics, but the EoS is often given in the form of a table. Consequently, hydrostatic solutions can only be found numerically and are available in the form of discrete data. While methods which incorporate analytical expressions are not able to exactly maintain these hydrostatic solutions, it is very well possible with the methods in [23,39] and the method we present in this paper. Especially, if we consider the better approximation of stellar structure which is given by a stationary state including rotation, our method can be applied to maintain this stationary solution. Another example from astrophysical application are rotating Keplerian disks. These are two-dimensional disks of matter which follows Newton's laws of motion in the gravitational field of a massive attractor. One way to describe this disk is a stationary solution of Euler equations with gravity including non-zero velocities. Since the velocity is not zero in such a solution, conventional well-balanced methods cannot preserve this solution. A special method designed for this application is presented in [42]. In

this paper, we will show that our method is also able to preserve this solution on different grids. Besides the applicability to any system of hyperbolic balance laws, the balancing of moving and time-depending solutions is one of the key features of our method.

The rest of the paper is structured as follows. In Section 2 we introduce the standard finite volume framework for multi-dimensional systems of hyperbolic conservation laws on arbitrary grids. In Section 3, we introduce our general well-balanced modification for this framework. The well-balanced property we claim for our method is then shown in Section 4. The validity of the well-balanced property also depends on a consistent choice of boundary conditions. Therefore, we add a discussion about well-balanced boundary conditions in Section 5. To emphasize how simple it is to add our method to an existing finite volume code, we comment on the implementation of the method in Section 6. Finally, in Section 7, we show a variety of numerical tests. The range of applications goes from Euler equations to ideal magnetohydrodynamics (MHD) equations. They include classical well-balanced tests on the balance laws and also tests on the homogeneous hyperbolic conservation laws. Different equations of state are used for the Euler equations. We include a test in which the well-balanced solution is not analytically known but has been obtained numerically. Also, we present tests in which the well-balanced solution depends on time. We verify high order accuracy for solutions close to and far away from the well-balanced solution numerically. A simple example for using a target solution which is obtained via numerical simulation is given. The robustness of our approach is validated in a shock tube on a hydrostatic solution for Euler equations with gravity. To show the efficiency of the method, we present CPU time comparisons of simulations with and without the well-balanced modification in Section 8.

2. A standard finite volume method

In this section we present the standard high order finite volume framework for three-dimensional hyperbolic balance laws [43,44]. Consider the 3-D system of hyperbolic balance laws

$$\partial_t \mathbf{q}(\mathbf{x}, t) + \nabla \cdot \mathcal{F}(\mathbf{q}(\mathbf{x}, t)) = \mathbf{s}(\mathbf{q}(\mathbf{x}, t), \mathbf{x}, t) \tag{1}$$

with $\mathcal{F} = (\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3)$, where $\mathbf{f}_l$ is the flux in l-direction. The spatial domain is partitioned by a mesh consisting of N non-overlapping control volumes. For the ith control volume Ω_i ($i \in \{1, \ldots, N\}$), we define the cell-average

$$\mathbf{Q}_i(t) := \frac{1}{V_i} \int_{\Omega_i} \mathbf{q}(\mathbf{x}, t) d\mathbf{x}, \tag{2}$$

where $V_i = |\Omega_i|$ is the control cell volume. Integrating Eq. (1) over Ω_i, applying the divergence theorem, approximating fluxes by numerical fluxes, and applying quadrature yields the standard semi-discrete methods

$$\frac{d}{dt}\mathbf{Q}_i(t) = -\frac{1}{V_i} \sum_{k \in N(i)} \left(\sum_{j=1}^M \omega_j \mathbf{F}\left(\mathbf{Q}_i^{\text{rec}}(\mathbf{x}_{ikj}, t), \mathbf{Q}_k^{\text{rec}}(\mathbf{x}_{ikj}, t), \mathbf{n}(\mathbf{x}_{ikj})\right) \right)$$

$$+ \frac{1}{V_i} I_{\mathbf{x} \in \Omega_i}\left[\mathbf{s}\left(\mathbf{Q}_i^{\text{rec}}, \mathbf{x}, t\right)\right]. \tag{3}$$

Here, $N(i)$ is the set of indices of all control volumes sharing an interface with Ω_i, M is the number of quadrature points at the interfaces, $\mathbf{x}_{ikj}$ are the M quadrature points at the ik interface and ω_j are the corresponding weights. $\mathbf{F}(\cdot, \cdot, \mathbf{n})$ is a numerical flux function consistent with $\mathbf{n} \cdot \mathcal{F}$, i.e. it is Lipschitz continuous in the first two arguments and the relation $\mathbf{F}(\mathbf{q}, \mathbf{q}, \mathbf{n}) = \mathbf{n} \cdot \mathcal{F}(\mathbf{q})$ holds for all unit vectors $\mathbf{n}$. the reconstructed functions $\mathbf{Q}_i^{\text{rec}}, \mathbf{Q}_k^{\text{rec}}$ are obtained using a consistent conservative reconstruction routine on the cell average values $\mathbf{Q}$. Examples for popular consistent conservative reconstruction routines can be found in [44–47]. The symbol $I_{\mathbf{x} \in \Omega}[\cdot]$

denotes a consistent discretization of the integral $\int_\Omega \cdot \, d\mathbf{x}$, which is realized by sufficiently high order accurate Gauß–Legendre quadrature rules in our numerical experiments. The source term discretizations applied in our numerical experiments are described in Appendix A.

The semi-discrete scheme Eq. (3) is kth order accurate if the applied reconstruction routine, interface flux quadrature and source term discretization are all at least kth order accurate. It can then be evolved in time using a kth order accurate ODE solver to obtain a kth order accurate fully discrete scheme.

3. The well-balanced modification of the standard finite volume method

In this section we will introduce a well-balanced modification for the three-dimensional finite volume method presented in Section 2. Reducing it to one or two spatial dimensions is straight forward.

Let $\tilde{\mathbf{q}}$ be a given continuous and sufficiently smooth solution of Eq. (1), which we refer to as *target solution* in the following. The equation

$$\partial_t \Delta \mathbf{q}(\mathbf{x}, t) + \nabla \cdot \mathcal{F}(\tilde{\mathbf{q}}(\mathbf{x}, t) + \Delta \mathbf{q}(\mathbf{x}, t)) - \nabla \cdot \mathcal{F}(\tilde{\mathbf{q}}(\mathbf{x}, t))$$
$$= \mathbf{s}(\tilde{\mathbf{q}}(\mathbf{x}, t) + \Delta \mathbf{q}(\mathbf{x}, t), \mathbf{x}, t) - \mathbf{s}(\tilde{\mathbf{q}}(\mathbf{x}, t), \mathbf{x}, t), \tag{4}$$

which is obtained by plugging $\tilde{\mathbf{q}}$ into Eq. (1), subtracting it from Eq. (1) itself, and rewrite it in terms of the deviation $\Delta \mathbf{q} := \mathbf{q} - \tilde{\mathbf{q}}$ from the target solution. Applying our discretization techniques to Eq. (4) instead of Eq. (1) yields the well-balanced semi-discrete scheme

$$\frac{d}{dt}(\Delta \mathbf{Q}_i(t)) = -\frac{1}{V_i} \sum_{k \in N(i)} \left(\sum_{j=1}^M \omega_j \Delta \hat{\mathcal{F}} \big(\Delta \mathbf{Q}_i^{\text{rec}}(\mathbf{x}_{ikj}, t), \right.$$
$$\left. \Delta \mathbf{Q}_k^{\text{rec}}(\mathbf{x}_{ikj}, t), \tilde{\mathbf{q}}(\mathbf{x}_{ikj}, t), \mathbf{n}(\mathbf{x}_{ikj}) \big) \right)$$
$$+ \frac{1}{V_i} I_{\mathbf{x} \in \Omega_i} \big[\mathbf{s}((\Delta \mathbf{Q}_i^{\text{rec}} + \tilde{\mathbf{q}})(\mathbf{x}, t), \mathbf{x}, t) \big]$$
$$- \frac{1}{V_i} I_{\mathbf{x} \in \Omega_i} [\mathbf{s}(\tilde{\mathbf{q}}(\mathbf{x}, t), \mathbf{x}, t)], \tag{5}$$

where $\Delta \mathbf{Q}_i$ is the cell-average of $\Delta \mathbf{q}$ in the ith cell and the reconstructed deviations $\Delta \mathbf{Q}_i^{\text{rec}}$ have been obtained using a consistent conservative reconstruction routine. Note that using a consistent reconstruction on the deviations $\Delta \mathbf{Q}$ is equivalent to the application of a hydrostatic reconstruction as e.g. in [3,48]. A typical hydrostatic reconstruction consists of a transformation to hydrostatic variables (i.e. a set of variables that is constant in the hydrostatic case), a consistent reconstruction, and a transformation back to conservative variables. In our description of the method the deviations $\Delta \mathbf{Q}$ correspond to the hydrostatic states. Hence, no transformation is required.

The numerical flux difference approximation $\Delta \hat{\mathcal{F}}$ is defined by

$$\Delta \hat{\mathcal{F}} \big(\Delta \mathbf{Q}^L, \Delta \mathbf{Q}^R, \tilde{\mathbf{q}}, \mathbf{n} \big) := \mathbf{F}(\Delta \mathbf{Q}^L + \tilde{\mathbf{q}}, \Delta \mathbf{Q}^R + \tilde{\mathbf{q}}, \mathbf{n}) - \mathbf{n} \cdot \mathcal{F}(\tilde{\mathbf{q}})$$
$$\approx \mathbf{n} \cdot (\mathcal{F}(\Delta \mathbf{q} + \tilde{\mathbf{q}}) - \mathcal{F}(\tilde{\mathbf{q}})), \tag{6}$$

where $\mathbf{F}(\cdot, \cdot, \mathbf{n})$ is a numerical flux function consistent with $\mathbf{n} \cdot \mathcal{F}$.

If the source term $\mathbf{s}$ is linear in the first argument, the following relation holds

$$\mathbf{s}((\tilde{\mathbf{q}} + \Delta \mathbf{q})(t, \mathbf{x}), \mathbf{x}, t) - \mathbf{s}(\tilde{\mathbf{q}}(t, \mathbf{x}), \mathbf{x}, t) = \mathbf{s}(\Delta \mathbf{q}(t, \mathbf{x}), \mathbf{x}, t). \tag{7}$$

Due to the linearity of the corresponding source term discretizations, this relation then also holds for the discretized source terms, which further simplifies the scheme. For example, this is the case for the gravitational source term in Euler or ideal MHD equations

and the bottom topography source term in the shallow water equations.

As in the standard method, this semi-discrete scheme Eq. (5) is kth order accurate if the applied reconstruction routine, interface flux quadrature and source term discretization are all at least kth order accurate. It can then be evolved in time using an at least kth order accurate ODE solver to obtain a kth order accurate fully discrete scheme.

Remark 3.1. In the description of the method, we assume the target solution $\tilde{\mathbf{q}}$ to be smooth. In the case of discontinuous $\tilde{\mathbf{q}}$, the two values $\tilde{\mathbf{q}}^L$ and $\tilde{\mathbf{q}}^R$ which are different if a discontinuity is present at the interface, have to be given instead of only one value. Eq. (6) has then to be modified to

$$\Delta \hat{\mathcal{F}} \big(\Delta \mathbf{Q}^L, \Delta \mathbf{Q}^R, \tilde{\mathbf{q}}^L, \tilde{\mathbf{q}}^R, \mathbf{n} \big)$$
$$:= \mathbf{F}(\Delta \mathbf{Q}^L + \tilde{\mathbf{q}}^L, \Delta \mathbf{Q}^R + \tilde{\mathbf{q}}^R, \mathbf{n}) - \mathbf{F}(\tilde{\mathbf{q}}^L, \tilde{\mathbf{q}}^R, \mathbf{n}), \tag{8}$$

i.e. the numerical flux function is also applied to the target solution. If $\tilde{\mathbf{q}}$ is continuous, Eq. (8) reduces to Eq. (6) due to the consistency of the numerical flux function $\mathbf{F}$. However, in the case of a discontinuous target solution $\tilde{\mathbf{q}}$ no high order convergence can be expected and the computational cost of the well-balanced modification increases.

4. Proof of the well-balanced property

In this section we show the well-balanced property of our method.

Theorem 4.1. *The modified finite volume method introduced in Section 3*
satisfies the following property: If

$$\Delta \mathbf{Q}_i = 0 \qquad \forall i \in \{1, \ldots, N\} \tag{9}$$

at initial time, then this holds for all $t > 0$. Consequently, if the initial condition $\mathbf{Q}_i(t = 0)$, $i = 1, \ldots, N$, equals the cell averages of the target solution $\tilde{\mathbf{Q}}_i(t = 0)$, $i = 1, \ldots, N$, the computed solution equals the target solution for all time.

Proof. Let $\Delta \mathbf{Q}_i = 0$ for all $i \in \{1, \ldots, N\}$. The consistency of the applied reconstruction leads to $\Delta \mathbf{Q}_i^{\text{rec}} \equiv 0$ at all flux quadrature points. The flux consistency then yields

$$\Delta \hat{\mathcal{F}} \big(\Delta \mathbf{Q}^L, \Delta \mathbf{Q}^R, \tilde{\mathbf{q}}, \mathbf{n} \big) = \Delta \hat{\mathcal{F}}(0, 0, \tilde{\mathbf{q}}, \mathbf{n}) = \mathbf{F}(\tilde{\mathbf{q}}, \tilde{\mathbf{q}}, \mathbf{n}) - \mathbf{n} \cdot \mathcal{F}(\tilde{\mathbf{q}})$$
$$= \mathbf{n} \cdot \mathcal{F}(\tilde{\mathbf{q}}) - \mathbf{n} \cdot \mathcal{F}(\tilde{\mathbf{q}}) = 0. \tag{10}$$

Now, consider the contribution from the source term: With $\Delta \mathbf{Q}_i = 0$ the source term discretization in Eq. (5) reduces to

$$I_{\mathbf{x} \in \Omega_i} \big[\mathbf{s}((\Delta \mathbf{Q}_i^{\text{rec}} + \tilde{\mathbf{q}})(\mathbf{x}, t), \mathbf{x}, t) \big] - I_{\mathbf{x} \in \Omega_i} [\mathbf{s}(\tilde{\mathbf{q}}(\mathbf{x}, t), \mathbf{x}, t)]$$
$$= I_{\mathbf{x} \in \Omega_i} [\mathbf{s}(\tilde{\mathbf{q}}(\mathbf{x}, t), \mathbf{x}, t)] - I_{\mathbf{x} \in \Omega_i} [\mathbf{s}(\tilde{\mathbf{q}}(\mathbf{x}, t), \mathbf{x}, t)] = 0. \tag{11}$$

We have shown that the right hand side in Eq. (5) vanishes and thus the initial data $\Delta \mathbf{Q}_i = 0$ are conserved for all time. The second part of the theorem follows easily. $\square$

Remark 4.2. If a stationary solution is chosen as target solution (which is the case for classical well-balancing applications), the time derivative of the target solution vanishes by definition. This leads to $\frac{d}{dt} \mathbf{Q}_i = \frac{d}{dt}(\Delta \mathbf{Q}_i)$. The described method can then also be used to directly evolve the $\mathbf{Q}_i$ in time instead of $\Delta \mathbf{Q}_i$.

5. Boundary conditions

The validity of the well-balanced property also depends on the correct choice of boundary conditions. In this section, we describe some boundary conditions that are compatible with the well-balancing property and support the potentially high order accuracy of the scheme. Some of the proposed numerical boundary

conditions require knowledge of the target solution outside the domain. If this is not available, one can simply extrapolate the target solution to the ghost cells using a sufficiently high order accurate extrapolation. The well-balanced scheme will in that case exactly balance the target solution including the extrapolated values. Consequently, this will neither affect the well-balanced property nor the order of accuracy.

Extrapolation boundary conditions: High order extrapolation of data in the domain to ghost cells supports the high order accuracy of the applied scheme. In our method we extrapolate the deviations $\Delta\mathbf{Q}$ such that the method is also well-balanced at the boundaries.

Wall and periodic boundary conditions: Periodic boundary conditions and wall boundary conditions as e.g. described in [49] can be applied to the deviations $\Delta\mathbf{Q}$ whenever physically valid. It is easy to show that these choices of boundary conditions supports the well-balanced property.

6. Notes on the implementation

We have seen that our well-balanced method can be applied for a wide range of problems. In this section we will discuss another property useful for applications: The fact that the method is also easy to be implemented. This holds especially if there is an existing finite volume code evolving Eq. (3). It can be easily modified to evolve Eq. (5) and hence obtain a well-balancing capability. The changes that should be introduced in a typical finite volume scheme are the following:

1. Implement a function able to return the target solution $\tilde{\mathbf{q}}$ in any point $(\mathbf{x}, t)$, or a structure containing the cell averages and all the values needed at each quadrature point if $\tilde{\mathbf{q}}$ is time-independent.
2. When the main routine receives the initial cell-averages $\mathbf{Q}_i$, it should transform it to the deviations $\Delta\mathbf{Q}_i = \mathbf{Q}_i - \tilde{\mathbf{Q}}_i$ and work directly with the deviation.
3. In the routine evaluating the numerical flux, it has to be computed $\Delta\hat{\mathcal{F}}\big(\Delta\mathbf{Q}^L, \Delta\mathbf{Q}^R, \tilde{\mathbf{q}}, \mathbf{n}\big)$ instead of $\mathbf{n} \cdot \mathcal{F}(\Delta\mathbf{q} + \tilde{\mathbf{q}})$. Basically, this just means subtracting the exact flux after the evaluation of the standard numerical flux.
4. This step is only necessary if the source term is not linear in $\mathbf{q}$: Evaluate the source term at the states $\tilde{\mathbf{Q}}_i + \Delta\mathbf{Q}_i$ and $\tilde{\mathbf{Q}}_i$. The difference of these source terms is computed and used to evolve the approximate solution as described in Eq. (5).

Let us remind of Remark 4.2 and point out that an alternative implementation could also evolve $\mathbf{Q}$ instead of $\Delta\mathbf{Q}$ in time if $\tilde{\mathbf{q}}$ is time-independent. In the computer implementation, this can be the easier and computationally efficient approach than evolving the deviations.

7. Numerical tests of the scheme

7.1. Hyperbolic systems used in the tests

Since the well-balancing procedure introduced in this article is applicable to any hyperbolic balance law, we present numerical experiments for two different hyperbolic systems. They are introduced in the following.

7.1.1. Compressible Euler equations with gravity source term

The 2-D compressible Euler equations which model the balance laws of mass, momentum, and energy under the influence of gravity are given by

$$\partial_t \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ E \end{bmatrix} + \partial_x \begin{bmatrix} \rho u \\ p + \rho u^2 \\ \rho uv \\ (E + p)u \end{bmatrix} + \partial_y \begin{bmatrix} \rho v \\ \rho uv \\ p + \rho v^2 \\ (E + p)v \end{bmatrix} = \begin{bmatrix} 0 \\ -\rho\partial_x\phi \\ -\rho\partial_y\phi \\ 0 \end{bmatrix} \quad (12)$$

with $\rho, p > 0$. Moreover, $E = \rho\varepsilon + \frac{1}{2}\rho|\mathbf{v}|^2 + \rho\phi$ is the total energy per unit volume with the velocity $\mathbf{v} = (u, v)^T$ and specific internal energy ε. The scalar function ϕ is a given gravitational potential. An additional relation between density, pressure, and specific internal energy is given in the form of an equation of state (EoS). In our tests we will use the ideal gas EoS law $p = (\gamma - 1)\rho\varepsilon$ with $\gamma = 1.4$ if not stated explicitly.

The 2-D Euler equations can be reduced to 1-D Euler equations by removing the y-flux and the ρv equation. It can be reduced to homogeneous Euler equations by removing the right-hand side of Eq. (12).

7.1.2. Homogeneous compressible ideal magnetohydrodynamics

The 2-D compressible ideal magnetohydrodynamics (MHD) equations which model the conservation of mass, momentum, magnetic field, and energy are given by

$$\partial_t \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ B_x \\ B_y \\ E \end{bmatrix} + \partial_x \begin{bmatrix} \rho u \\ \rho u^2 + p + \frac{1}{2}(B_y^2 - B_x^2) \\ \rho uv - B_x B_y \\ 0 \\ B_y u - v B_x \\ u(E + p + \frac{1}{2}B_y^2 - \frac{1}{2}B_x^2) - v B_x B_y \end{bmatrix}$$
$$+ \partial_y \begin{bmatrix} \rho v \\ \rho uv - B_x B_y \\ \rho v^2 + p + \frac{1}{2}(B_x^2 - B_y^2) \\ B_x v - u B_y \\ 0 \\ v(E + p + \frac{1}{2}B_x^2 - \frac{1}{2}B_y^2) - u B_x B_y \end{bmatrix} = 0, \quad (13)$$

where B_x, B_y are the x- and y-component of the magnetic field. The total energy is $E = \rho\varepsilon + \frac{1}{2}\rho|\mathbf{v}|^2 + \frac{1}{2}(B_x^2 + B_y^2)$. All other quantities are defined as for the Euler equations. We use the same EoS as for the Euler equations.

One can also define 2-D compressible ideal MHD equations such that they include the ρw and B_z components. This is in principle reasonable due to the genuine three-dimensional interactions between velocity and magnetic field. In our tests we set ρw and B_z to zero and there is no difference if we omit the corresponding equations.

7.2. Code and numerical methods

We test the methods proposed in this paper using a finite volume code implemented in Python. The code is built in a modular way, such that different schemes can be easily implemented. Note that our well-balanced method is not restricted to the methods we choose to use in the tests and one can for example also use other reconstruction methods or quadrature formulae. One can also apply numerical flux functions designed for special problems (e.g. a low Mach number compliant method for Euler equations like in [50,51]).

Grids: The domain in two-dimensional problems is discretized using a structured grid and in some tests, we use curvilinear grids. Our implementation of curvilinear grids (see Appendix A.1) restricts the overall method to only second order accuracy. Note that our well-balanced modification can also be applied to high order accurate methods on curvilinear grids such as the ones described in [52,53]. Higher order accuracy is demonstrated on a Cartesian grid in our numerical experiments.

Numerical flux function: As numerical flux function we use the local Lax–Friedrichs flux (e.g. [43]), since it is simple and can be applied for any hyperbolic system. In some tests, we use the Roe's approximate Riemann solver for Euler equations [54] to obtain more accurate results.

Table 1

L^1-errors for an isothermal hydrostatic solution of the Euler equations after time $t = 2$ computed using the standard (Std) and well-balanced (WB) methods with different orders of accuracy (O*m* for *m*th order). The setup is described in Section 7.3.

error in	Std-O1	WB-O1	Std-O2	WB-O2	Std-O3	WB-O3	Std-O7	WB-O7
ρ	1.19e-01	0.00e+00	4.60e-04	0.00e+00	9.72e-05	0.00e+00	1.28e-09	0.00e+00
ρu	2.18e-02	0.00e+00	7.24e-04	0.00e+00	1.50e-04	0.00e+00	1.40e-09	0.00e+00
E	1.64e-01	0.00e+00	3.20e-03	0.00e+00	3.92e-04	0.00e+00	3.99e-09	0.00e+00

First order method: To formally obtain a first order method, we use constant reconstruction to obtain the interface values. The numerical fluxes are computed at the center of the interfaces and the source term is evaluated at the cell-center. For the gravity source term used in our tests, we need the gradient $\nabla \phi$ of the given gravitational potential which is evaluated at the cell center using analytical differentiation.

Second order method: To formally obtain a second order method, we use a conservative linear reconstruction (e.g. [44]) with a minmod limiter (e.g. [44]) to obtain the interface values. This is the only difference to the first order method.

Third order method: To formally obtain a third order method, we use a conservative CWENO3 ([47] for 1-D, [55] for 2-D) reconstruction to obtain the interface values. In the two-dimensional case, the numerical fluxes are evaluated at the Gauß–Legendre quadrature points. To compute the source term we multiply the CWENO3 polynomials in momentum with the interpolation polynomial of $\partial_x \phi$ or $\partial_y \phi$ respectively. This results in a polynomial function of the spatial variables whose cell average can be computed by analytical integration.

Seventh order method: To formally obtain a one-dimensional seventh order method, we use a conservative CWENO7 reconstruction [56] to obtain the interface values. To compute the source term we multiply the CWENO7 polynomial in momentum with the interpolation polynomial of $\partial_x \phi$. The resulting source term polynomial is cell-averaged using analytical integration.

Boundary conditions: If the setup has periodic character we use periodic boundary conditions. Otherwise we extrapolate the states to ghost cells with a sufficiently high order using one- or two-dimensional polynomials. If we use the third order method, for example, we extrapolate using parabolas.

Time-stepping: The first order accurate scheme is evolved in time using the explicit forward Euler method. For the second and third order accurate semi-discrete scheme we use the explicit third order, four stage Runge–Kutta method from [57] and the explicit tenth order, 17 stage Runge–Kutta method from [58] is used in the seventh order accurate method. Obviously, it would e.g. be sufficient to use a second order accurate Runge–Kutta method in the second order accurate method. However, this article is mainly concerned with spatial discretization and choosing different Runge–Kutta methods can be expected to have few impact on the results if they are sufficiently high order accurate.

7.3. 1-D isothermal hydrostatic solution of the Euler system with gravity

We consider an isothermal hydrostatic solution of the 1-D compressible Euler equations with the ideal gas law and the gravitational source term given by

$$\phi(x) = \sin(2\pi x), \quad \tilde{\rho}(x) = \tilde{p}(x) = \exp(-\phi(x)), \quad \tilde{u} \equiv 0. \quad (14)$$

We set these data on a 1-D grid with 128 grid cells on the domain [0,1]. These initial data are evolved up to the final time $t = 2$ using the first, second, and third order method with the standard method and the well-balanced method each. In the well-balanced method we set the initial data Eq. (14) as time-independent target

solution. The L^1-errors at final time compared to the initial grid can be seen in Table 1. We see that there is no error when the well-balanced method is applied.

Remark 7.1. Most other well-balanced methods balance fluxes against the source term which leads to machine errors. In our method we balance fluxes against fluxes and source term against source term. Thus, the differences can cancel out exactly and the error can be exactly zero.

7.4. Perturbed 1-D isothermal hydrostatic solution of the Euler system with gravity

We add a perturbation to the pressure such that our initial conditions are

$$\rho(x) = \tilde{\rho}(x), \quad u(x) = \tilde{u}(x), \quad p(x) = \tilde{p}(x) + \eta \exp\left(-100\left(x - \frac{1}{2}\right)^2\right) \quad (15)$$

in the domain [0,1]. We choose $\eta = 0.1$ to test the convergence of our method. We evolve this initial setup up to time $t = 0.2$ using our well-balanced method (first to third order and seventh order). The results and convergence rates are shown in Table 2. As a reference solution we use a numerical solution computed with the seventh order standard scheme on a grid with 4096 cells. All convergence rates match our expectations. The convergence rate for the seventh order scheme drops in the last step, since the error approaches machine precision. In Fig. 1, density deviations at time $t = 0.2$ for the test with $\eta = 10^{-5}$ are shown. The discretization error on the hydrostatic background is significant and a rather high resolution is required in order to resolve the perturbation. On the other hand, it gets evident that the second order well-balanced method is capable of correctly resolving the perturbation even on a coarse grid.

7.5. Riemann problem on a 1-D isothermal hydrostatic solution of the Euler system with gravity

To test the robustness of our well-balanced methods in combination with CWENO reconstruction we use the initial data

$$\rho(x) := \begin{cases} \exp\left(-\frac{1}{2}\phi(x)\right) & \text{if } x < 0.125, \\ \exp(-\phi(x)) & \text{if } x \geq 0.125, \end{cases}$$

$$p(x) := \begin{cases} 2\exp\left(-\frac{1}{2}\phi(x)\right) & \text{if } x < 0.125, \\ \exp(-\phi(x)) & \text{if } x \geq 0.125, \end{cases} \quad u(x) := 0 \quad (16)$$

with $\phi(x) := -10x$. Eq. (16) describes a piecewise isothermal hydrostatic solution with a jump, which includes all three waves of the Euler equations. We set these initial data on the domain [0,0.25] and evolve them to the final time $t = 0.02$ using our third and seventh order well-balanced method with CWENO reconstruction and Roe's approximate Riemann solver on 128 grid cells. As target solution for the well-balanced methods we choose

$$\tilde{\rho}(x) := \exp(-\phi(x)) \quad \tilde{p}(x) := \exp(-\phi(x)) \quad \tilde{u}(x) := 0. \quad (17)$$

The results at final time are presented in Fig. 2. As a reference we use a result obtained with a first order standard method on

Table 2

L^1-errors and convergence rates in total energy for a small pressure perturbation on an isothermal hydrostatic solution of the Euler equations after time $t = 0.2$. Different well-balanced methods are used. The setup is described in Section 7.4.

N	WB-O1		WB-O2		WB-O3		WB-O7	
	E error	rate	E error	rate	E error	rate	E error	rate
256	5.73e-03	0.9	5.98e-05	2.0	6.93e-05	2.6	1.32e-09	6.5
512	3.08e-03	0.9	1.49e-05	2.0	1.18e-05	2.7	1.46e-11	6.6
1024	1.60e-03	1.0	3.73e-06	2.0	1.78e-06	2.8	1.46e-13	4.1
2048	8.15e-04		9.36e-07		2.53e-07		8.41e-15	
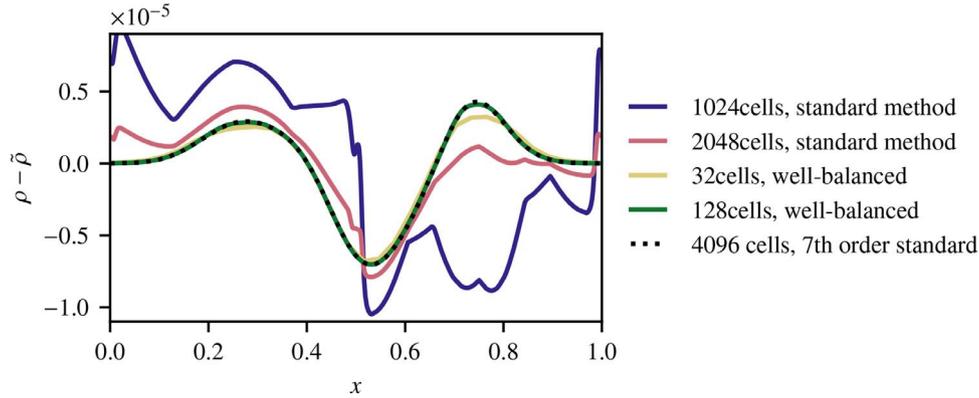
Fig. 1. Perturbation on a hydrostatic atmosphere. The test setup is described in Section 7.4. The density deviation from the hydrostatic background is shown at time $t = 0.2$ for the perturbation $\eta = 1e - 5$. The second order (if not stated explicitly) standard and well-balanced method are used with different resolutions.
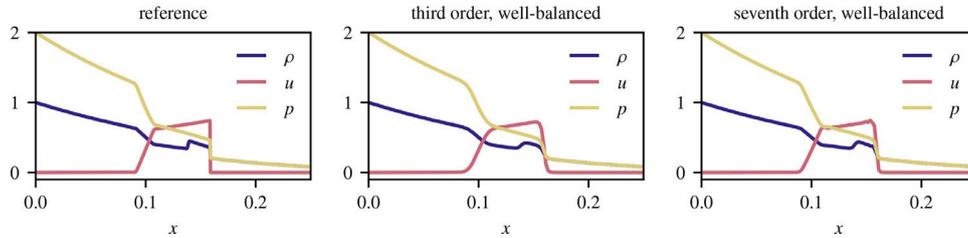
Fig. 2. Riemann problem on an isothermal hydrostatic solution test case from Section 7.5. The three panels show numerical results at final time $t = 0.02$ as described in the text.

8192 grid cells (left panel). Neither the third order method (central panel) nor the seventh order method (right panel) show significant oscillations. The wave structure is captured correctly by both methods.

7.6. 2-D numerically approximated hydrostatic solution of the Euler system with gravity

In stellar astrophysics applications, the hydrostatic state of the star can often be given in a discrete form. In this test, we will show that our well-balanced method can be used if the target solution is given in the form of discrete data in a table.

The thermodynamical quantities shall be related by the EoS for an ideal gas with radiation pressure, which is given by Chandrasekhar [59]

$$p = \rho T + T^4, \qquad \varepsilon = \frac{T}{\gamma - 1} + \frac{3}{\rho} T^4, \qquad (18)$$

which relates temperature and pressure by an implicit relation. We assume the following data is given. Let the gravitational potential be $\phi(\mathbf{x}) = \phi(x, y) = x + y$ and the hydrostatic temperature profile

is $\bar{T}(\mathbf{x}) = 1 - 0.1\phi(\mathbf{x})$. Using Chebfun [60] in the numerical software MATLAB we solve the 1-D hydrostatic equation and EoS for density and pressure corresponding to the given temperature profile for the 1-D gravitational potential $\phi_{1-D}(x) = \sqrt{2}x$. The data are stored as point values on a fine grid (10,000 data points). To obtain the 2-D hydrostatic data on the grid we apply the following procedure:

We use a cubic spline interpolation to construct a continuous function $\mathbf{q}_{1-D}^{hs}(x)$ (e.g. [61]) from the data points. Then we extend the one-dimensional hydrostatic solution $\mathbf{q}_{1-D}^{hs}$ to a two-dimensional solution via $\mathbf{q}^{hs}(\mathbf{x}) := \mathbf{q}_{1-D}^{hs}(x + y)$. The values for the target solution at interface quadrature points can then be evaluated pointwise as $\tilde{\mathbf{q}}(\mathbf{x}, t) = \mathbf{q}^{hs}(\mathbf{x})$. The cell average values of the target solution are computed using a third order accurate 2-D Gauß–Legendre quadrature rule.

We use a 64×64 grid to evolve the hydrostatic initial condition to the final time $t = 2$. For the conversion between pressure and internal energy we use Newton's method to solve for the temperature. The L^1-errors at final time are shown in Table 3. In all tests using the well-balanced modification, there is no error at the final time.

Table 3

L^1-errors in total energy for the numerically integrated hydrostatic solution of the Euler equations with radiation pressure after time $t = 2$ computed using different methods. The setup is described in Section 7.6.

N	Std-O1	WB-O1	Std-O2	WB-O2	Std-O3	WB-O3
64	5.10e-03	0.00e+00	4.38e-05	0.00e+00	1.36e-07	0.00e+00
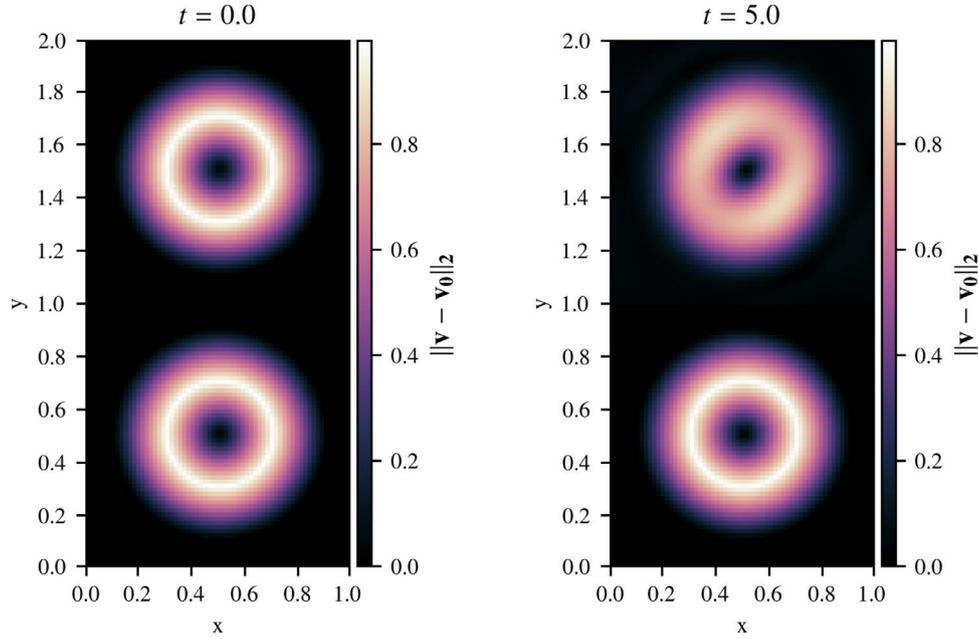
Fig. 3. Illustration for the double Gresho vortex test from Section 7.7. The absolute velocity after subtraction of the constant advection velocity is shown for the initial (left panel) and final (right panel) time. The vortex which is included in the target solution (which is the bottom vortex in both panels) is preserved while the other one is diffused and deformed.

7.7. Double Gresho vortex

In this test we use a vortex for homogeneous 2-D Euler equations first introduced in [62]. The pressure and angular velocity of this vortex in dependence of the distance to the center are given by

$$\left(\hat{u}(r), \hat{p}(r)\right)$$
$$= \begin{cases} \left(5r, 5 + \frac{25}{2}r^2\right), & \text{if } 0 \leq r < 0.2, \\ \left(2 - 5r, 9 - 4\ln(0.2) + \frac{25}{2}r^2 - 20r + 4\ln(r)\right), & \text{if } 0.2 \leq r < 0.4, \\ \left(0, 3 + 4\ln(2)\right), & \text{if } 0.4 \leq r, \end{cases} \quad (19)$$

The radial velocity is zero and the density is $\rho \equiv 1$. In our test we set up the domain $[0, 1] \times [0, 2]$ with two Gresho vortices centered at $(0.5, 0.5)$ and $(0.5, 1.5)$ respectively. The vortices are advected with the velocity $\mathbf{v}_0 = (u_0, v_0)^T = (0.2, 0.4)^T$ and the boundaries are periodic. At time $t = 5$ the exact solution of this initial data equals the initial setup. We apply our well-balanced method on a 64×128 grid to evolve the initial condition up to final time $t = 5$. We use Roe's numerical flux functions and a linear reconstruction. Only the vortex initially (and finally) centered at $(0.5, 0.5)$ is included in the target solution. The result is illustrated in Fig. 3.

7.8. 2-D Euler wave in gravitational field

To demonstrate that we can follow time-dependent solutions exactly with our method we use a problem from Xing and Shu [22] and Chandrashekar and Zenk [63] which involves a known exact solution of the 2-D Euler equations with gravity given by

$$\tilde{\rho}(t, x, y) = 1 + \frac{1}{5}\sin(\pi(x + y - t(u_0 + v_0))), \qquad \tilde{u}(t, x, y) = u_0,$$
$$\tilde{v}(t, x, y) = v_0,$$

$$\tilde{p}(t, x, y) = p_0 + t(u_0 + v_0) - x - y$$
$$+ \frac{1}{5\pi}\cos(\pi(x + y - t(u_0 + v_0))). \quad (20)$$

The gravitational potential is $\phi(\mathbf{x}) = x + y$, the EoS is the ideal gas EoS. In accordance to Xing and Shu [22] and Chandrashekar and Zenk [63] we choose $u_0 = v_0 = 1$, $p_0 = 4.5$ on the domain $[0, 1]^2$. We use the first, second, and third order accurate well-balanced method to evolve the initial data with $t = 0$ to a final time $t = 0.1$ on a 64×64 Cartesian grid and the second order well-balanced method on a 64×64 polar grid. The L^1-error in every component of the state vector is exactly zero in each of the tests. We omit showing a table since it does not provide additional insight.

7.9. Perturbation on the 2-D Euler wave in gravitational field

In this test we want to verify the order of accuracy for perturbations to time-dependent target solutions if the well-balanced method is used. For this we use the initial setup from Eq. (20) and add a pressure perturbation:

$$\rho(t = 0, x, y) = \tilde{\rho}(t = 0, x, y), \ u(t = 0, x, y) = \tilde{u}(t = 0, x, y),$$
$$v(t = 0, x, y) = \tilde{v}(t = 0, x, y),$$

$$p(t = 0, x, y) = \tilde{p}(t = 0, x, y)$$
$$+ \eta\exp\left(-100\left(\left(x - \frac{1}{2}\right)^2 + \left(y - \frac{1}{2}\right)^2\right)\right). \quad (21)$$

We evolve these initial data to time $t = 0.1$ using the third order standard and well-balanced method with $\eta = 0.1$. The L^1 errors and corresponding convergence rates are presented in Table 4. As reference solution for determining the error we use a numerically approximated solution computed using the third order standard method on a 1024^2 grid. In this test we use exact boundary

Table 4

L^1-errors and convergence rates in total energy for different pressure perturbations ($\eta = 0.1, 10^{-5}$) on the wave in a gravitational field solution of the 2-D Euler equations after time $t = 0.1$. The third order standard and well-balanced method are used. The setup is described in Section 7.9.

grid cells	$\eta = 0.1$				$\eta = 10^{-5}$			
	Std-O3		WB-O3		Std-O3		WB-O3	
	E error	rate	E error	rate	E error	rate	E error	rate
64×64	3.96e-04	2.8	3.84e-04	2.8	4.38e-05	2.7	9.38e-08	2.7
128×128	5.81e-05	3.0	5.65e-05	3.0	6.68e-06	3.0	1.49e-08	2.9
256×256	7.50e-06	3.1	7.30e-06	3.1	7.72e-07	3.1	1.93e-09	3.2
512×512	8.50e-07		8.30e-07		5.46e-08		2.17e-10	

conditions for the standard method, which means that we evaluate the states in the ghost cells at any time from Eq. (20). We see third order convergence for both methods. However, there seems to be no significant benefit from using the well-balanced method in this test. The choice of $\eta = 0.1$ leads to a large discretization error in the perturbation which seems to dominate the total error. Choosing a large perturbation is necessary since we use a solution computed from the standard method as a reference to compute the errors. For smaller perturbations the standard method fails to provide a sufficiently accurate reference solution. To yet show the improved accuracy of the well-balanced modification we add a convergence test with a small perturbation of $\eta = 10^{-5}$ for which a sufficiently accurate reference solution is produced using the third order well-balanced method on a 1024^2 grid. The errors and convergence rates for the third order accurate standard and well-balanced methods can also be seen in Table 4 and it gets evident that the well-balanced method is significantly more accurate on the small perturbation.

7.10. 2-D Keplerian disk

Consider a stationary solution given by Gaburro et al. [21]

$$\tilde{\rho} \equiv 1, \quad \tilde{u}(x, y) = -\sin(\alpha(x,y))\sqrt{\frac{Gm_S}{r(x,y)}},$$

$$\tilde{v}(x, y) = \cos(\alpha(x,y))\sqrt{\frac{Gm_S}{r(x,y)}}, \quad \tilde{p} \equiv 1 \tag{22}$$

with the gravitational potential $\phi(r) = -\frac{Gm_S}{r}$ and $r = \sqrt{x^2 + y^2}$, $\alpha = \arctan(\frac{y}{x})$, $G = m_S = 1$. We use the initial conditions

$$\rho(x, y) = \begin{cases} 2 & \text{if } (x - 1.2)^2 + (y - 1.0)^2 < 0.15^2 \\ \tilde{\rho} & \text{else} \end{cases} \tag{23}$$

and $u = \tilde{u}, v = \tilde{v}, p = \tilde{p}$ on the domain $[-2, 2] \times [-2, 2]$. We chose the domain such that we omit the singularity in the velocity at $(x, y) = (0, 0)$. In Fig. 4 results of numerical tests are illustrated. The second order standard and well-balanced methods are applied on a polar grid with 32×256 cells and a Cartesian grid with 128×128 cells. In the Cartesian grid we take out the center with $r < 1$ using Dirichlet boundary conditions. We also use Dirichlet boundary conditions at all outer boundaries. Since there is a discontinuity in the initial setup we apply a minmod slope limiter to the linear reconstruction of the conserved variables in the standard method or the deviations in conserved variables in the well-balanced method. The density at time $t = 2.5$ for each simulation is shown in Fig. 4 together with the exact solution. Since this is an purely advective problem and there is no radial component to the velocity, the quantity $\|(\rho - 1)r\|_1$, which describes the average distance of the density perturbation to the center, is conserved for all time in the exact solution. For our simulations we measure the quantity $d = \|(\rho(t = 2.5) - 1)r\|_1 / \|(\rho(t = 0) - 1)r\|_1$ as a measure of the quality of the numerical solutions. For the exact solution we

have $d = 1$ for all time. The values of d are shown in the center of the plots in Fig. 4.

In the tests with the standard method we see discretization errors in the Keplerian disk solution Eq. (22). This introduces radial velocities, the advection of the spot of increased density has a component towards the center. In the tests using our well-balanced methods the result is free of discretization errors in the Keplerian disk solution Eq. (22). The advection is more accurate, the only errors are diffusion errors. The polar grid is more suitable for this test problem, since it is adapted to the radial geometry. The test using our well-balanced method on the Cartesian grid is more diffusive than the one on the polar grid, yet we see that the well-balanced modification improves the result significantly on both grids.

7.11. Stationary MHD vortex - long time

We consider the following exact solution of the homogeneous 2-D ideal MHD equations:

$$\hat{x} = x - tu_0, \qquad \hat{y} = y - tv_0, \qquad r^2 = \hat{x}^2 + \hat{y}^2,$$

$$u = u_0 - k_p e^{\frac{1-r^2}{2}}\hat{y}, \quad v = v_0 + k_p e^{\frac{1-r^2}{2}}\hat{x}, \quad \rho = 1,$$

$$B_x = -m_p e^{\frac{1-r^2}{2}}\hat{y}, \quad B_y = m_p e^{\frac{1-r^2}{2}}\hat{x}, \quad p = 1 + \left(\frac{m_p^2}{2}(1 - r^2) - \frac{k_p^2}{2}\right)e^{1-r^2}. \tag{24}$$

This setup describes a stationary vortex which is advected through the domain with the velocity (u_0, v_0). The domain is $[-5, 5] \times [-5, 5]$. One vortex turnover-time is $t_{turnover} = \frac{2\pi}{\sqrt{e}k_p} \approx \frac{3.81}{k_p}$. In a first test we set $m_p = k_p = 0.1$, $u_0 = v_0 = 0$ and run the test up to $t = 100t_{turnover}$ on a 32×32 grid. We use the well-balanced method and the target solution equals the initial data. The numerical error at final time compared to the initial setup is exactly zero in all conservative variables.

7.12. Stationary MHD vortex - numerical target solution

In this test we present a simple application in which the result of a high-resolution simulation is used as target solution for low-resolution simulations. For this purpose, we once more use the stationary MHD vortex test case described in Section 7.11. The parameters are $k_p = m_p = 0.1$ and $u_0 = v_0 = 0.1$, the final time is $t_{final} = 5$. First, we compute a target solution using our third order non-well-balanced method with 128×128 grid cells with parabolic extrapolation boundary conditions. Every time-step is stored. The discrete target solution is made available for low resolution simulations by mapping the data at each time step to the coarse grid with

$$\tilde{\mathbf{Q}}_{ij}(t^n) = \frac{1}{\tilde{V}_{ij}} \sum_{\hat{x}_{kl} \in \tilde{\Omega}_{ij}} \hat{V}_{kl}\hat{\mathbf{Q}}_{kl}^n, \tag{25}$$

where all quantities with $\tilde{}$ correspond to the coarse grid and all quantities with $\hat{}$ correspond to the fine grid. The values of $\tilde{\mathbf{Q}}_{ij}$ at
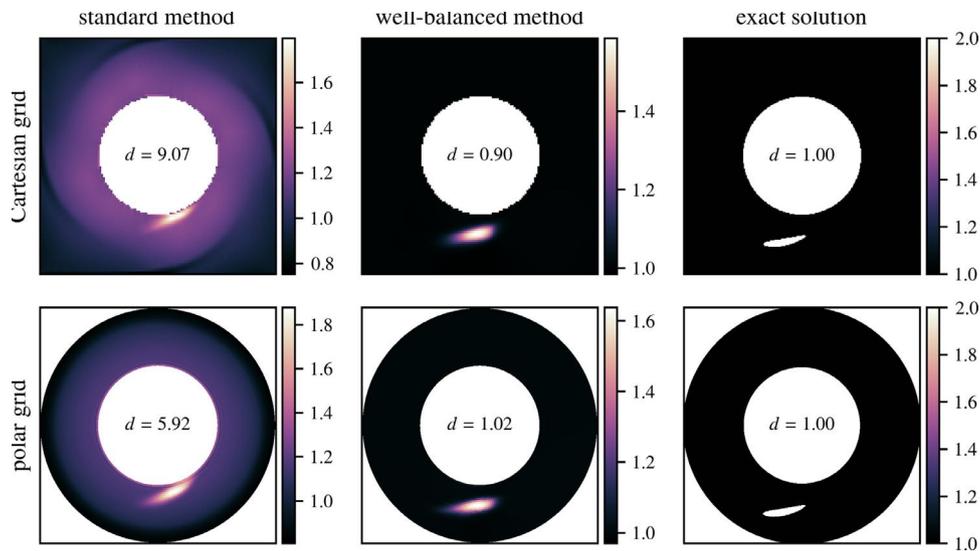
Fig. 4. Mass advection on a Keplerian disk using second order standard (left column) and well-balanced (center column) methods on a Cartesian 128×128 grid (top row) and a polar 32×256 grid (bottom row). The setup is given in Section 7.10. Results can be compared to the exact solutions shown in the right column. The meaning of d is described in the text. The domain shown in each panel is $[-2, 2]^2$. The x-coordinate increases to the right, the y-coordinate increases to the top.
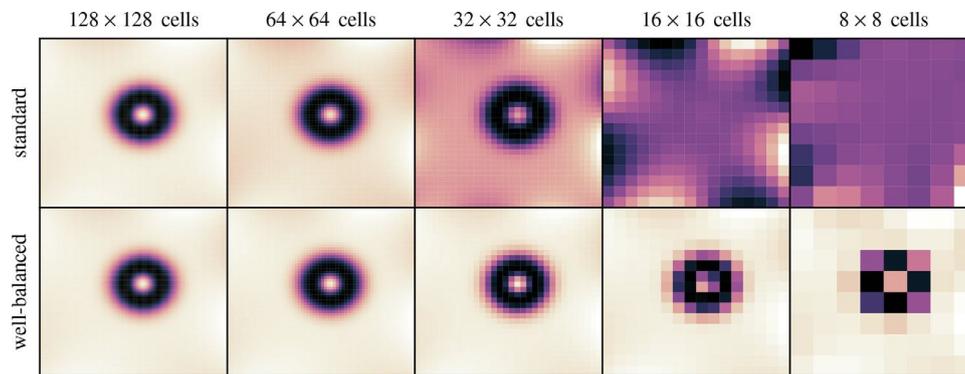
Fig. 5. Pressure of the moving stationary MHD vortex as described in Section 7.12 at the final time. The target solution for the well-balanced method is the numerical solution computed with the standard method on a 128×128 cells grid (upper left panel). In the upper panels the standard method is used, in the lower panels the well-balanced method is used. Different columns correspond to different resolutions.

intermediate times are obtained via a third order accurate interpolation in time.

The value of the target solution at all quadrature points required in the scheme are evaluated using a third order accurate interpolation on the cell-centered point values $\tilde{\mathbf{q}}^{\mathrm{rec}}(\mathbf{x}_{ij})$ that have been computed using a third order accurate conservative reconstruction on the cell-averages $\tilde{\mathbf{Q}}_{ij}$.

The resulting pressure for well-balanced and standard methods on different Cartesian meshes is shown in Fig. 5. All methods use CWENO3 reconstruction and parabolic extrapolation boundary conditions. On the 128×128 grid the solutions for the well-balanced and non-well-balanced method are exactly the same, since the solution from the standard method is used as target solution in the well-balanced method. For smaller resolutions the standard method is too diffusive to resolve the vortex. The quality of the results obtained with the well-balanced method is the same for all resolutions, since all of them use the same 128×128 simulation as target solution.

8. Computational cost of the modification

Well-balanced methods are constructed to improve the accuracy with which solutions of balance laws are approximated. In the previous section we have shown that usage of our well-balanced

modification can improve the accuracy of a simulation significantly. On the other hand, an increase of computational effort can countervail the gain in accuracy, if it is too high. In this section we will compare the computation times of simulations using our well-balanced modification to simulations using the corresponding standard method and show that the increase in CPU time is moderate.

8.1. The procedure

To compare the methods, we will run tests with different setups and grid resolutions using a standard method and the corresponding well-balanced method. We use the simple python code described in Section 7 on a single CPU. Each test is repeated 20 times and the wall clock times are measured. We compute the average wall-clock time and standard deviations of the single runs for every test. The ratio of average wall clock time for the well-balanced compared to the standard method is visualized depending on the grid resolution.

Note that we use final times that are significantly smaller than the final times used in the corresponding tests above. The reason for this is that after some time the solutions obtain with and without well-balancing differ. This results in different sizes for the time steps which significantly influences the wall-clock time necessary to reach the final time. Since we aim to compare the efficiency
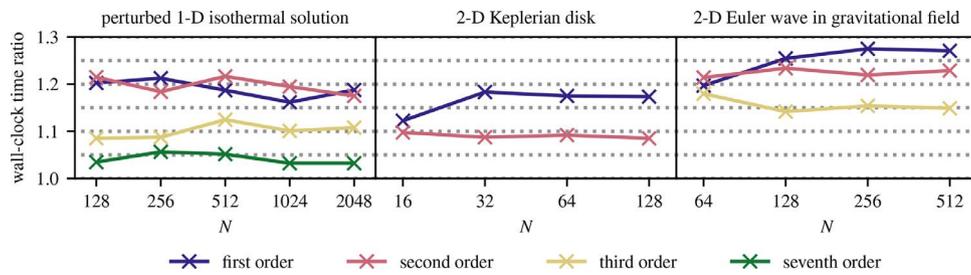
Fig. 6. Ratio of the average wall-clock times for the well-balanced and the standard method. Values and errors are determined as described in Section 8.1 with N cells. *Left*: One-dimensional test case with static target solution from Section 7.4 *Center*: Two-dimensional test case with stationary target solution from Section 7.10 on a polar grid with $N \times 8N$ cells. *Right*: Two-dimensional test case with time-dependent target solution from Section 7.9 on a Cartesian grid with N^2 cells.

of the methods without taking the quality of the solution into account, we use reduced final times.

8.2. The tests

To compare the runtimes we use test setups described in Section 7 using exactly the same methods. In a first test, we use the perturbed one-dimensional isothermal solution described in Section 7.4 with a final time $t = 0.02$. The first, second, third, and seventh order 1-D methods are applied. The ratio of wall clock times for the tests with and without the well-balanced modification can be seen in the left panel of Fig. 6. The target solution $\tilde{q}$ used in the well-balanced modification is constant in time in this test case. It is hence computed once and stored in an array. From the figure we see that we can expect an increase in CPU time of about 20% for the first and second order accurate methods. The difference in runtime reduces significantly for methods with higher order of accuracy. For the seventh order methods the increase in wall-clock time is only about 5%.

As a second test setup we choose the Keplerian disk from Section 7.10. We evolve it to time $t = 0.01$ with a first and second order method on a polar grid. As in the previous test, the target solution is time-independent and thus only computed once. The wall-clock time ratios are visualized in the central panel of Fig. 6. We observe an increase in CPU time of less than 20% when using the first order well-balanced method and less than 10% when using the second order well-balanced method.

To also test the increase of CPU time consumption for a simulation in which the target solution is time dependent, we use the Euler wave in a gravitational field with perturbation from Section 7.9. We evolve the solution up to the final time $t = 0.01$ with each method. In this test, the target solution is computed from a function every time it is used (which happens in every intermediate step). The result of these tests can be seen in the right panel of Fig. 6. We see an increase in CPU time of less than 30% if the well-balanced method is used. Note that, again, the wall-clock time ratio is smaller for methods with higher order accuracy. For the third order well-balanced method we only observe an increase of about 15% in wall-clock time.

9. Summary and conclusions

We introduced a new general framework for the construction of well-balanced finite volume methods for hyperbolic balance laws. A standard finite volume method is modified such that it evolves the deviation from a target solution instead of the actual solution. This makes the scheme exact on the target solution. The finite volume method can include any consistent reconstruction, numerical flux function, interface quadrature, source term discretization, and ODE solver for time discretization. Thus, it can be arbitrarily high order accurate and the method can be defined on any computational grid geometry. One can view our method as a high order

extension of [39] and [23] to all known solutions of all hyperbolic balance laws.

In numerical tests with Euler and MHD equations on different grids we could verify that the method can successfully be applied to exactly maintain static and stationary solutions or even follow time-dependent solutions. For that, the solution has to be known either analytically or in the form of discrete data. The latter case is especially interesting for complex applications like stellar astrophysics, where static states of the Euler equations with gravity can be obtained numerically but only in few cases analytically. Also, for the case of differentially rotating stars, i.e. stars that are in a rotating stationary state with an angular velocity depending on the distance to the axis of rotation, our method can be applied for well-balancing since it can include non-zero velocities in stationary states. High order accuracy has been confirmed in numerical experiments. Also, in a series of numerical tests we have shown that the increase in computational time is moderate.

The proposed well-balanced method can be easily implemented in existing finite volume codes with minimal effort. However, there are applications in which the well-balanced solution is not known beforehand and the current method cannot be applied. In that case, another well-balanced methods existing in the literature for the balance law under consideration has to be applied, in case such a scheme exists. In all other cases, in which the well-balanced solution is known, our simple framework can be applied to obtain the well-balanced property.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Jonas P. Berberich: Conceptualization, Methodology, Software, Investigation, Visualization, Writing - original draft, Project administration. **Praveen Chandrashekar:** Methodology, Investigation, Validation, Writing - review & editing. **Christian Klingenberg:** Supervision, Writing - review & editing.

Acknowledgments

Appendix A. Details of the applied finite volume schemes

We use structured grids in all the numerical tests and hence in the description of the details, we restrict ourselves to structured grids. Some parts of the scheme, such as the reconstruction methods, are applied to $\mathbf{Q}$ in the standard method but to $\Delta\mathbf{Q}$ in the well-balanced method. We will denote the states with $\mathbf{U}$; depending on the method we have $\mathbf{U} = \mathbf{Q}$ or $\mathbf{U} = \Delta\mathbf{Q}$. Analogously, we will use $\mathbf{u}$ to denote $\mathbf{q}$ or $\Delta\mathbf{q}$.

A1. Curvilinear coordinates

We define a 2-D curvilinear coordinate system. The coordinates in physical space are $\mathbf{x} = (x, y)$, the coordinates in computational space are $\boldsymbol{\xi} = (\xi, \eta)$. The (i, j)th cell is denoted $\Omega_{i,j}$ in the physical space and by $\hat{\Omega}_{i,j}$ in the computational space. We can rewrite Eq. (12) in the computational coordinates as

$$J\partial_t \mathbf{q} + \partial_\xi \hat{\mathbf{f}} + \partial_\eta \hat{\mathbf{g}} = J\mathbf{s}, \tag{A.1}$$

where

$$J := \det\begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{pmatrix}, \quad \hat{\mathbf{f}} := J\left(\frac{\partial \xi}{\partial x}\mathbf{f} + \frac{\partial \xi}{\partial y}\mathbf{g}\right),$$

$$\hat{\mathbf{g}} := J\left(\frac{\partial \eta}{\partial x}\mathbf{f} + \frac{\partial \eta}{\partial y}\mathbf{g}\right). \tag{A.2}$$

To solve Eq. (12) on the curvilinear physical grid, we can now solve Eq. (A.1) on a Cartesian grid. We construct the grid from the nodes and approximate the derivatives of the coordinate transformation using central differences on the nodal coordinates. This implementation of curvilinear grids restricts the scheme to second order accuracy. We can achieve higher order accuracy only on Cartesian grids. More details on the finite volume method on a curvilinear mesh can be found in [39].

Polar grid: The polar grid can be defined by the function

$$\mathbf{x}(\boldsymbol{\xi}) := \begin{pmatrix} \xi \sin(\eta) \\ \xi \cos(\eta) \end{pmatrix} \tag{A.3}$$

for $\xi > 0$, $\eta \in [0, 2\pi)$. Note, that this functions can not be inverted at $\xi = 0$, i.e. $\mathbf{x} = 0$. Hence, the origin in physical coordinates has to be omitted, when this grid is used.

A2. Source term discretization

In some tests we use a gravity source term for Euler equations. The source term component in the momentum equation has to be approximated to sufficiently high order.

Second order source term discretization: For the first and second order method, we use the second order accurate source term discretization

$$-\frac{1}{\Delta_x}\int_{\Omega_i} \rho(x)g(x)dx \approx -\bar{\rho}(\Omega_i)g(x_i) \tag{A.4}$$

in the one-dimensional case and

$$-\frac{1}{|\Omega_{ij}|}\int_{\Omega_{ij}} \rho(\mathbf{x})\mathbf{g}(\mathbf{x})d\mathbf{x} \approx -\bar{\rho}(\Omega_{ij})\mathbf{g}(\mathbf{x}_{ij}) \tag{A.5}$$

in the two-dimensional case in the momentum equation. The cell-averaged density is denoted $\bar{\rho}$ and the gravitational acceleration $\mathbf{g} = \nabla\phi$ is given exactly at the cell-center.

Third and seventh order source term discretization in 1-D: For the one-dimensional methods with CWENO reconstruction we define

$$S_i^{\rho u} := -\frac{1}{|\Omega_i|}\int_{\Omega_i} \rho^{\text{rec}}(x)g^{\text{int}}(x)\, dx,$$

where ρ^{rec} is the density polynomial obtained from the CWENO reconstruction and g^{int} is the gravitational acceleration interpolated

from the cell centered values to third or seventh order respectively. Since the integrand is a polynomial, the integral can be computed exactly.

Third order source term discretization in 2-D: Similar to the 1-D case we define the cell-averaged momentum source terms

$$S_{ij}^{\rho u} := -\frac{1}{|\Omega_{ij}|}\int_{\Omega_{ij}} \rho^{\text{rec}}(\mathbf{x})g_x^{\text{int}}(\mathbf{x})\, d\mathbf{x} \quad \text{and}$$

$$S_{ij}^{\rho v} := -\frac{1}{|\Omega_{ij}|}\int_{\Omega_{ij}} \rho^{\text{rec}}(\mathbf{x})g_y^{\text{int}}(\mathbf{x})\, d\mathbf{x}$$

where ρ^{rec} is obtained from the two-dimensional CWENO3 reconstruction and $\mathbf{g}^{\text{int}} = \left(g_x^{\text{int}}, g_y^{\text{int}}\right)$ is the parabola that satisfies $\mathbf{g}^{\text{int}}(\mathbf{x}_{ij}) = \mathbf{g}_{ij}$, $\mathbf{g}^{\text{int}}(\mathbf{x}_{i\pm1,j\pm1}) = \mathbf{g}_{i\pm1,j\pm1}$, and $\mathbf{g}^{\text{int}}(\mathbf{x}_{i\pm1,j\mp1}) = \mathbf{g}_{i\pm1,j\mp1}$ for the cell-centered point values in coordinate direction and approximates the diagonal values $\mathbf{g}_{i+1,j+1}$, $\mathbf{g}_{i-1,j+1}$, $\mathbf{g}_{i+1,j-1}$, and $\mathbf{g}_{i-1,j-1}$ in the least square sense.

References

[1] Godunov SK. Finite difference method for numerical computation of discontinuous solution of the equations of fluid dynamics. Matematicheskii Sbornik 1959;47:271.

[2] Brufau P, Vázquez-Cendón M, García-Navarro P. A numerical model for the flooding and drying of irregular domains. Int J Numer Methods Fluids 2002;39(3):247–75.

[3] Audusse E, Bouchut F, Bristeau M-O, Klein R, Perthame B. A fast and stable well-balanced scheme with hydrostatic reconstruction for shallow water flows. SIAM J Sci Comput 2004;25(6):2050–65.

[4] Bermudez A, Vázquez ME. Upwind methods for hyperbolic conservation laws with source terms. Comput Fluids 1994;23(8):1049–71.

[5] LeVeque RJ. Balancing source terms and flux gradients in high-resolution Godunov methods: the quasi-steady wave-propagation algorithm. J Comput Phys 1998;146(1):346–65.

[6] Noelle S, Pankratz N, Puppo G, Natvig JR. Well-balanced finite volume schemes of arbitrary order of accuracy for shallow water flows. J Comput Phys 2006;213(2):474–99.

[7] Castro MJ, Semplice M. Third-and fourth-order well-balanced schemes for the shallow water equations based on the CWENO reconstruction. Int J Numer Methods Fluids 2018.

[8] Barsukow W, Berberich JP. A well-balanced active flux scheme for the shallow water equations with wetting and drying. Journal on Scientific Computing 2020. Submitted.

[9] Desveaux V, Zenk M, Berthon C, Klingenberg C. Well-balanced schemes to capture non-explicit steady states: Ripa model. Math Comput 2016;85(300):1571–602.

[10] Touma R, Klingenberg C. Well-balanced central finite volume methods for the Ripa system. Appl Numer Math 2015;97:42–68.

[11] Noelle S, Xing Y, Shu C-W. High-order well-balanced finite volume WENO schemes for shallow water equation with moving water. J Comput Phys 2007;226(1):29–58.

[12] Xing Y, Shu C-W, Noelle S. On the advantage of well-balanced schemes for moving-water equilibria of the shallow water equations. J Sci Comput 2011;48(1–3):339–49.

[13] Cargo P, LeRoux A. A well balanced scheme for a model of atmosphere with gravity. Comptes Rendus de l'Academie des Sciences Serie I-Mathematique 1994;318(1):73–6.

[14] LeVeque R, Bale D. Wave propagation methods for conservation laws with source terms. In: Birkhauser, Basel; 1999. p. pp.609–618.

[15] LeVeque RJ, George DL, Berger MJ. Tsunami modelling with adaptively refined finite volume methods. Acta Numer 2011;20:211–89.

[16] Chandrashekar P, Klingenberg C. A second order well-balanced finite volume scheme for Euler equations with gravity. SIAM J Sci Comput 2015;37(3):B382–402.

[17] Ghosh D, Constantinescu EM. Well-balanced, conservative finite difference algorithm for atmospheric flows. AIAA J 2016.

[18] Touma R, Koley U, Klingenberg C. Well-balanced unstaggered central schemes for the Euler equations with gravitation. SIAM J Sci Comput 2016;38(5):B773–807.

[19] Bermúdez A, López X, Vázquez-Cendón ME. Finite volume methods for multi-component Euler equations with source terms. Comput Fluids 2017;156:113–34.

[20] Chertock A, Cui S, Kurganov A, Özcan ŞN, Tadmor E. Well-balanced schemes for the Euler equations with gravitation: conservative formulation using global fluxes. J Comput Phys 2018.

[21] Gaburro E, Castro MJ, Dumbser M. Well-balanced Arbitrary-Lagrangian-Eulerian finite volume schemes on moving nonconforming meshes for the Euler equations of gas dynamics with gravity. Mon Not R Astron Soc 2018;477(2):2251–75.

[22] Xing Y, Shu C-W. High order well-balanced WENO scheme for the gas dynamics equations under gravitational fields. J Sci Comput 2013;54(2–3):645–62.

[23] Klingenberg C, Puppo G, Semplice M. Arbitrary order finite volume well-balanced schemes for the Euler equations with gravity. SIAM J Sci Comput 2019;41(2):A695–721.

[24] Gaburro E. A unified framework for the solution of hyperbolic PDE systems using high order direct Arbitrary-Lagrangian-Eulerian schemes on moving unstructured meshes with topology change. Arch Comput Methods Eng 2020.

[25] Fuchs FG, McMurry A, Mishra S, Risebro NH, Waagan K. High order well-balanced finite volume schemes for simulating wave propagation in stratified magnetic atmospheres. J Comput Phys 2010;229(11):4033–58.

[26] Tanaka T. Finite volume TVD scheme on an unstructured grid system for three-dimensional MHD simulation of inhomogeneous systems including strong background potential fields. J Comput Phys 1994;111(2):381–9.

[27] Powell KG, Roe PL, Linde TJ, Gombosi TI, de Zeeuw DL. A solution-adaptive upwind scheme for ideal magnetohydrodynamics. J Comput Phys 1999;154:284–309. doi:10.1006/jcph.1999.6299.

[28] Desveaux V, Zenk M, Berthon C, Klingenberg C. A well-balanced scheme for the Euler equation with a gravitational potential. In: Finite Volumes for Complex Applications VII-Methods and Theoretical Aspects. Springer; 2014. p. 217–26.

[29] Desveaux V, Zenk M, Berthon C, Klingenberg C. A well-balanced scheme to capture non-explicit steady states in the Euler equations with gravity. Int J Numer Methods Fluids 2016;81(2):104–27.

[30] Thomann A, Zenk M, Klingenberg C. A second-order positivity-preserving well-balanced finite volume scheme for Euler equations with gravity for arbitrary hydrostatic equilibria. Int J Numer Methods Fluids 2019;89(11):465–82.

[31] Parés C. Numerical methods for nonconservative hyperbolic systems: a theoretical framework.. SIAM J Numer Anal 2006;44(1):300–21.

[32] Castro M, Gallardo JM, López-García JA, Parés C. Well-balanced high order extensions of Godunov's method for semilinear balance laws. SIAM J Numer Anal 2008;46(2):1012–39.

[33] Gaburro E, Dumbser M, Castro MJ. Direct Arbitrary-Lagrangian-Eulerian finite volume schemes on moving nonconforming unstructured meshes. Comput Fluids 2017;159:254–75.

[34] Käppeli R, Mishra S. Well-balanced schemes for the Euler equations with gravitation. J Comput Phys 2014;259:199–219.

[35] Käppeli R, Mishra S. A well-balanced finite volume scheme for the Euler equations with gravitation-the exact preservation of hydrostatic equilibrium with arbitrary entropy stratification. Astron Astrophys 2016;587:A94.

[36] Varma D, Chandrashekar P. A second-order, discretely well-balanced finite volume scheme for Euler equations with gravity. Comput Fluids 2019.

[37] Grosheintz-Laval L, Käppeli R. High-order well-balanced finite volume schemes for the Euler equations with gravitation. J Comput Phys 2019;378:324–43.

[38] Berberich JP, Chandrashekar P, Klingenberg C. A general well-balanced finite volume scheme for Euler equations with gravity. In: C Klingenberg, M Westdickenberg, editors. Theory, numerics and applications of hyperbolic problems I, Springer proceedings in mathematics & statistics 236; 2018. p. 151–63. doi:10.1007/978-3-319-91545-6_12.

[39] Berberich JP, Chandrashekar P, Klingenberg C, Röpke FK. Second order finite volume scheme for Euler equations with gravity which is well-balanced for general equations of state and grid systems. Commun Comput Phys 2019;26:599–630. doi:10.4208/cicp.OA-2018-0152.

[40] Botta N, Klein R, Langenberg S, Lützenkirchen S. Well balanced finite volume methods for nearly hydrostatic flows. J Comput Phys 2004;196(2):539–65.

[41] Giraldo FX, Restelli M. A study of spectral element and discontinuous Galerkin methods for the Navier–Stokes equations in nonhydrostatic mesoscale atmospheric modeling: equation sets and test cases. J Comput Phys 2008;227(8):3849–77.

[42] Gaburro E. Well balanced Arbitrary-Lagrangian-Eulerian finite volume schemes on moving nonconforming meshes for non-conservative hyperbolic systems. University of Trento; 2018. Ph.D. thesis.

[43] LeVeque RJ. Numerical methods for conservation laws. 2nd ed. Basel: Birkhauser; 1992.

[44] Toro EF. Riemann Solvers and Numerical Methods for Fluid Dynamics: a practical Introduction. Berlin Heidelberg: Springer; 2009. ISBN 9783540498346.

[45] Harten A, Osher S. Uniformly high-order accurate nonoscillatory schemes. I. SIAM J Numer Anal 1987;24(2):279–309.

[46] Liu X-D, Osher S, Chan T. Weighted essentially non-oscillatory schemes. J Comput Phys 1994;115:200–12. doi:10.1006/jcph.1994.1187.

[47] Levy D, Puppo G, Russo G. Compact central WENO schemes for multidimensional conservation laws. SIAM J Sci Comput 2000;22(2):656–72.

[48] Berberich JP, Chandrashekar P, Käppeli R, Klingenberg C. High order discretely well-balanced methods for arbitrary hydrostatic atmospheres. Commun Comput Phys 2021. doi:10.13140/RG.2.2.10439.11683/1. Submitted.

[49] Blazek J. Computational fluid dynamics: principles and applications. Butterworth-Heinemann; 2015.

[50] Barsukow W, Edelmann PVF, Klingenberg C, Miczek F, Röpke FK. A numerical scheme for the compressible low-Mach number regime of ideal fluid dynamics. J Sci Comput 2017:1–24. doi:10.1007/s10915-017-0372-4.

[51] Berberich JP, Klingenberg C. Entropy stable numerical fluxes for compressible Euler equations which are suitable for all Mach numbers. Accepted for publication in: SEMA SIMAI series: numerical methods for hyperbolic problems Numhyp 2019; 2020. doi:1013140/RG221207096326.

[52] Colella P, Dorr MR, Hittinger JA, Martin DF. High-order, finite-volume methods in mapped coordinates. J Comput Phys 2011;230(8):2952–76.

[53] Wongwathanarat A, Grimm-Strele H, Müller E. APSARA: a multi-dimensional unsplit fourth-order explicit Eulerian hydrodynamics code for arbitrary curvilinear grids. Astron Astrophys 2016;595:A41.

[54] Roe PL. Approximate Riemann solvers, parameter vectors, and difference schemes. J Comput Phys 1981;43(2):357–72. doi:10.1016/0021-9991(81)90128-5.

[55] Levy D, Puppo G, Russo G. A third order central WENO scheme for 2D conservation laws. Appl Numer Math 2000;33(1):415–22.

[56] Cravero I, Puppo G, Semplice M, Visconti G. CWENO: uniformly accurate reconstructions for balance laws. Math Comput 2018;87(312):1689–719.

[57] Kraaijevanger JFBM. Contractivity of Runge–Kutta methods. BIT Numer Math 1991;31(3):482–528.

[58] Feagin T. A tenth-order Runge–Kutta method with error estimate. In: Proceedings of the IAENG conference on scientific computing; 2007.

[59] Chandrasekhar S. An introduction to the study of stellar structure, vol 2. Courier Corporation; 1958.

[60] Driscoll TA, Hale N, Trefethen LN. Chebfun guide. Pafnuty Publications, Oxford; 2014.

[61] Press WH, Teukolsky S, Vetterling W, Flannery B. Numerical recipes in C: the art of scientific computing. 2nd ed. New York: Cambridge Univ. Press; 1992.

[62] Gresho PM, Chan ST. On the theory of semi-implicit projection methods for viscous incompressible flow and its implementation via a finite element method that also introduces a nearly consistent mass matrix. Part 2: implementation. Int J Numer Methods Fluids 1990;11(5):621–59. doi:10.1002/fld.1650110510.

[63] Chandrasekar P, Zenk M. Well-balanced nodal discontinuous Galerkin method for Euler equations with gravity. J Sci Comput 2017:1–32.