Stabilizing Stiffness in Relativistic Hydrodynamics: A Comparative Study of RK2, Density Regulator Augmentation, and GRK2

Leo Kim(260914383), Prof. Sangyong Jeon

McGill University Department of Physics

April 10, 2025

Abstract

Stiffness in relativistic viscous hydrodynamics arises when sharp gradients develop in low-density or "vacuum" regions, triggering potential numerical instability in explicit solvers. In this work, we compare three methods for handling stiff behavior: a standard second-order Runge–Kutta (RK2) scheme, RK2 augmented with a density regulator, and a newly proposed Gaussian Runge–Kutta 2 (GRK2) approach. Our numerical tests show that plain RK2 can diverge if the time step is not kept extremely small, whereas adding the density regulator reliably suppresses numerical blow-ups without requiring fine-tuning. GRK2 provides a more fundamental remedy by embedding the stiff dynamics in its integration, thereby maintaining stability even under extreme gradients. These findings suggest that both the regulated RK2 and GRK2 methods achieve robust performance in relativistic hydrodynamic simulations, laying a foundation for future work on more complex fluid systems.

Contents

1 Introduction

A system of differential equations is called *stiff* when there are rapidly changing components (or terms) alongside more slowly varying ones, forcing very small time steps in certain standard numerical methods[1] to maintain stability. In other words, widely varying timescales render many explicit methods inefficient or unstable unless the time step is severely reduced.

Stiff systems appear in many fields and many forms. A straightforward example is the Van der Pol oscillator, defined by

$$\begin{cases} x'(t) = y(t), \\ y'(t) = \mu(1 - x^2)y - x, \end{cases} \tag{1}$$

which is a classic nonlinear oscillator used in various fields such as electrical engineering, biology, and mechanics. This oscillator illustrates self-sustaining oscillations and nonlinear damping, and is used for any modeling that shows relaxation-oscillator behaviour [1].

In our previous work [2], we observed that a standard explicit numerical solver–the second-order Runge-Kutta method (RK2)–can easily diverge if the time steps become too large, because of the large disparity in growth or decay rates of the two variables (see Figure 1).

A more complex stiff system appears in relativistic viscous hydrodynamics, where one studies fluid dynamics subject to relativistic effects and dissipative processes under extreme conditions. In such systems, an undesirable instability arises in numerical implementations using explicit methods, with certain components of the stress-energy tensor diverging at (nearly) empty locations. That is, this problem typically occurs when there is an abrupt transition from a region of finite matter density to vacuum.

To analyze fluid dynamics numerically and understand how stiffness emerges, let us consider how hydrodynamic equations are numerically solved. The first step is to specify the initial stress-energy tensor $T^{\mu\nu}$ at $t = 0$, denoted by $T_0^{\mu\nu}$. Then one solves the Landau matching condition [3]

$$T^{\mu\nu}u_\nu = \epsilon u^\mu, \tag{2}$$

where the stress-energy tensor $T^{\mu\nu}$ is defined by

$$T^{\mu\nu} = \epsilon u^\mu u^\nu - p\Delta^{\mu\nu} + \pi^{\mu\nu}. \tag{3}$$

Here, u^μ is the flow velocity 4-vector, p is the local thermodynamic pressure of the fluid, and ϵ is the local energy density. Our metric is $g_{\mu\nu} = \text{diag}(1, -1, -1, -1)$ and u_ν is defined as $u_\nu = g_{\nu\mu}u^\mu$. The last term $\pi^{\mu\nu}$ is the shear tensor. Once u^μ and ϵ are obtained by solving

---

[1]An explicit method is a numerical scheme in which the next value of the unknown is computed directly from known quantities at earlier steps without solving coupled equations at the new time level.
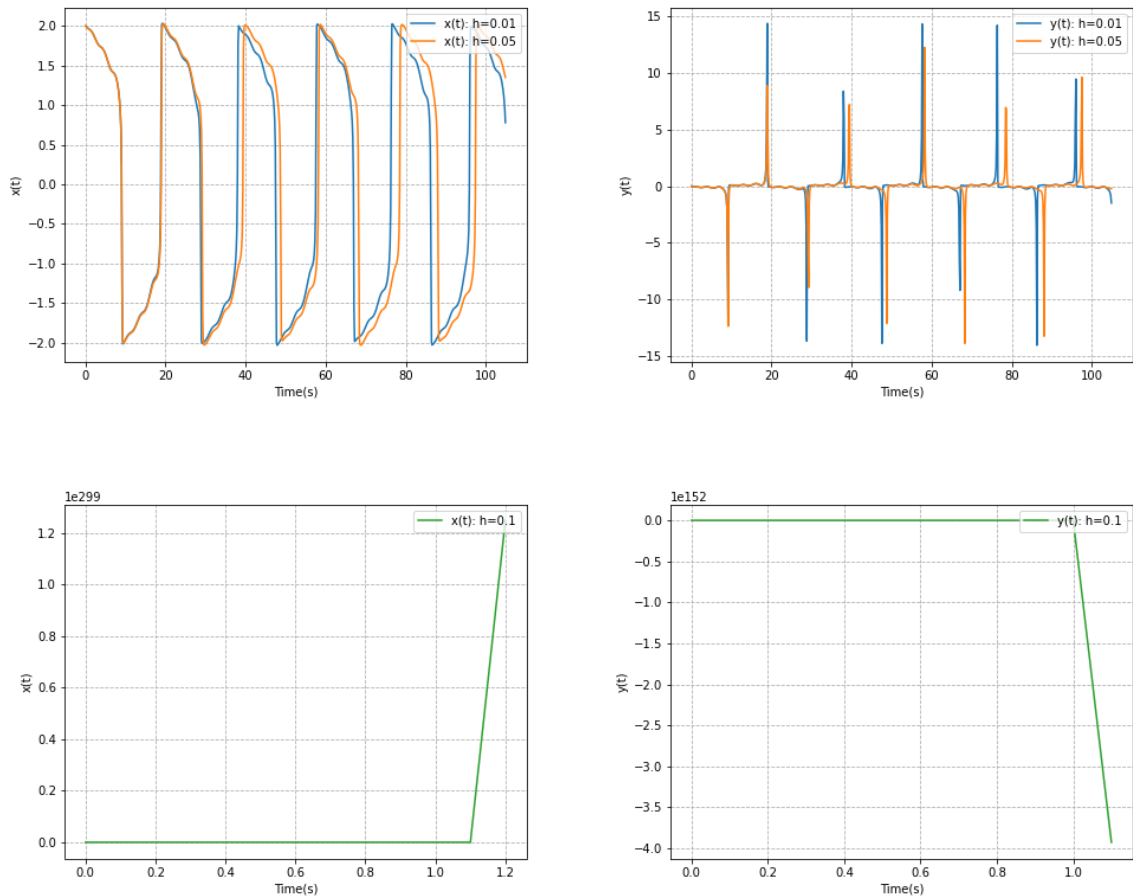
Figure 1: Van der Pol solutions $x(t)$ (left) and $y(t)$ (right) computed with second-order Runge–Kutta (RK2) using three step sizes: $h = 0.01$ (blue), 0.05 (orange), and 0.1 (green). As the step size grows, the numerical solutions deviate more noticeably from the exact solution, gradually damping along with a phase error. The solutions ultimately diverge for $h = 0.1$ only within one time step which suppresses the trend in the finite region.

3

Eq.(2), we can calculate the Navier-Stokes tensor

$$2\eta\sigma^{ij} \;=\; \eta(\epsilon)(\partial^i u^j + \partial^j u^i - \frac{2}{3}\delta^{ij}\nabla\cdot u) \tag{4}$$

This equation is the definition of the Navier-Stokes tensor σ^{ij} in the rest frame of the fluid cell at a certain spatial position in 3D, denoted by $\mathbf{x}$. Here, $\eta(\epsilon)$ is the shear viscosity coefficient which is a function of the local energy density ϵ. The moving frame Navier-Stokes tensor $\sigma^{\mu\nu}$ can be obtained by boosting Eq.(4) with the flow velocity u^μ. The next step is to solve the Israel-Stewart relaxation equation for the shear tensor

$$\tau_\pi\dot{\pi}^{ij} + \pi^{ij} \;=\; 2\eta\sigma^{ij}. \tag{5}$$

Again, this is the equation in the rest frame of fluid cell and can be boosted to the general frame. This equation shows how π^{ij} evolves towards its Navier-Stokes limit, $2\eta\sigma^{ij}$, over a characteristic relaxation time τ_π. When π^{ij} is non-zero, the system is out of equilibrium and Eq.(5) represents an entropy-increasing dissipative process that moves the system towards equilibrium. At the initial time step, $\pi^{\mu\nu}$ is assumed to vanish (denoted $\pi_0^{\mu\nu}$). Using $\pi_0^{\mu\nu}$ and $2\eta\sigma^{\mu\nu}$ from Eq.(4) into Eq.(5), one can advance $\pi_0^{\mu\nu}$ by one time step to obtain $\pi_1^{\mu\nu}$. In that sense, $\sigma^{\mu\nu}$ can be regarded as the "source" term driving $\pi^{\mu\nu}$. The next step is to update $T^{\mu\nu}$. Rewriting Eq.(3) as

$$T^{\mu\nu} \;=\; T_{ideal}^{\mu\nu} + \pi^{\mu\nu}, \tag{6}$$

we can update ϵ and u^μ by solving the energy-momentum conservation law

$$\partial_u T^{\mu\nu} \;=\; \partial_\mu T_{ideal}^{\mu\nu} + \partial_\mu\pi^{\mu\nu} \;=\; 0. \tag{7}$$

To do so, we can re-arrange Eq.(7)

$$\partial_\mu T_{ideal}^{\mu\nu} \;=\; -\partial_\mu\pi^{\mu\nu}, \tag{8}$$

and solve for ϵ_1 and u_1^μ using the equation of state $p = p(\epsilon)$ and treating $-\partial_\mu\pi_1^{\mu\nu}$ as the source term. In this way, we obtain

$$T_1^{\mu\nu} = T_{ideal,1}^{\mu\nu} + \pi_1^{\mu\nu}. \tag{9}$$

We can then go back to Eq.(2) to begin the next time step until reaching the desired final time.

Within this procedure, one observes the need to compute $\sigma^{\mu\nu}$ (the source component) prior to $\pi^{\mu\nu}$ (the physical component). Should $\sigma^{\mu\nu}$ become very large because of the abrupt change in the flow velocity, $\pi^{\mu\nu}$ will become very large as well, triggering a runaway reaction. This is precisely the major pitfall in conventional explicit methods, and addressing it is the main goal of our research.

There are multiple ways to handle stiff systems. As discussed earlier, a straightforward option is to use an explicit solver, such as the second-order Runge-Kutta scheme (Heun's

4

method or RK2), given by [4]

$$x_{n+1} = x_n + \frac{h}{2}\Big[f(t_n, x_n) + f(t_n + h, x_n + hf(t_n, x_n))\Big], \tag{10}$$

where f generally refers to the right hand side of a differential equation of the form $\dot{x} = f(t, x)$. Although explicit methods can produce very accurate solutions if the time step h is kept sufficiently small, this severely hampers efficiency. In relativistic hydrodynamics, however, the resolution needed for phenomenological study is often much coarser than the size of time step required to keep the stiff equations stable. This observation suggests looking for a solver that can remain stable at larger h yet still maintain acceptable accuracy.

In the context of relativistic viscous hydrodynamics, one strategy is to add a regulator on top of RK2. For instance, Qiu introduced a density-limiting regulator [5]

$$\chi(\rho) = \frac{\tanh(\rho)}{\rho}, \tag{11}$$

where

$$\rho = \frac{\sqrt{\pi^{\mu\nu}\,\pi_{\mu\nu}}}{\rho_{\max}\,\sqrt{\epsilon^2 + 3\,p^2}}. \tag{12}$$

This ensures the regulated shear tensor $\hat{\pi}^{\mu\nu} = \pi^{\mu\nu}\chi(\rho)$ never exceeds $\rho_{max}\sqrt{\epsilon^2 + 3p^2}$, where ρ_{max} is a constant parameter. Although such an approach can successfully prevent the shear tensor from diverging, it is effectively an *ad-hoc* fix that suppresses spikes without "numerically" taming the unphysical divergences. Consequently, a solver that allows reasonably large time steps by design would be preferable.

A recognized approach to alleviating severe step-size restrictions is to split the differential equation into a stiff component (treated exactly via some advanced strategy) and a non-stiff or "source" part (handled using explicit methods). A classic example is *Exponential Time Differencing* (ETD), introduced by Cox and Matthews [6]. The idea is to consider an equation

$$\frac{dx}{dt} = -p_n x + S, \tag{13}$$

where p_n is an approximation of the stiff coefficient over the nth time interval, and S contains the remaining non-stiff terms [7]. In the second-order variant, ETD2RK, the stiff term $-p_n$ is approximated as a constant, while the non-stiff term $S(t, x)$ is approximated via a Runge–Kutta-like approximation. Following Cox and Matthews, the formal solution of Eq.(13) for one time step is

$$x(t_{n+1}) = x(t_n)e^{p_n h} + e^{p_n h}\int_0^h e^{-p_n \tau} S(x(t_n + \tau), t_n + \tau)d\tau, \tag{14}$$

where h again is the time step, and the integral accounts for the contribution of the source term S over the interval between t_n and $t_{n+1} = t_n + h$ [6]. The advantage of ETD2RK is

5

that it allows a somewhat larger h than standard explicit methods. However, if $p(t)$ varies significantly within a single step, treating it as a constant p_n can introduce amplitude or phase errors, and even lead to instability.

To address this limitation, a newer scheme called *Gaussian Runge–Kutta 2 (GRK2)* has been proposed. Rather than assuming $p(t)$ remains constant over each step, GRK2 uses a linear interpolation [7],

$$p(\tau) = p_n + \frac{p_{n+1} - p_n}{h}(\tau - t_n), \tag{15}$$

where $p_n = p(t_n)$ is the stiff coefficient at the start of the interval, and $p_{n+1} = p(t_{n+1})$ may come from a predictor step. Allowing $p(t)$ to vary linearly in $[t_n, t_{n+1}]$ enables the solver to adapt to an exact integration of the linear part, leading to an error function expression instead of a single exponential factor in Eq.(14). This enables it to adapt more accurately to large changes (stiffness) in $p(t)$. Consequently, the formal solution in GRK2 is

$$x_{n+1} = e^{-\frac{p_n + p_{n+1}}{2}h}x_n + \int_0^h d\rho\, e^{-b\rho - a\rho^2} S(\rho), \tag{16}$$

where $\rho = t_{n+1} - t$, $a = \frac{p_n - p_{n+1}}{2h}$, and $b = p_{n+1}$, again referencing [7]. GRK2 thus provides a more robust alternative to ETD2RK if $p(t)$ is changing rapidly.

In our work, we implement three schemes–RK2, the density regulator (DR), and GRK2– and compare them under various settings: RK2 alone, RK2 + DR, and GRK2 alone. Contrasting GRK2 with these established approaches clarifies its viability in complex systems such as relativistic viscous hydrodynamics. In the following section, we discuss how each method is integrated into the algorithm.

2 Methods

We implemented all three schemes in C language, each advancing the solution by a single time step. For RK2 (Heun's method), we simply use the two-stage predictor–corrector formula in Eq.(10).

For the DR, we follow Eq.(11) and (12), computing ρ from $\pi^{\mu\nu}$ and then applying $\chi(\rho)$ to ensure that the magnitude of $\pi^{\mu\nu}$ remains bounded. Immediately after each RK2 update, $\pi^{\mu\nu}$ is replaced by $\hat{\pi}^{\mu\nu} = \pi^{\mu\nu}\chi(\rho)$. This step prevents unphysical spikes in the shear tensor.

To build up to GRK2, one can start with the ETD2RK algorithm. According to Cox and Matthews, ETD2RK integrates the stiff component with an exponential factor while treating the non-stiff part using a second-order scheme [6]. The update rule is

$$x_{n+1} = a_n + \left(S(a_n, t_n + h) - S_n\right)\frac{e^{p_n h} - 1 - hc}{hp_n^2}, \tag{17}$$

where

$$a_n = x_n e^{p_n h} + S_n \frac{e^{p_n h} - 1}{p_n}, \tag{18}$$

and the correction accounts for changes in the predicted source $\tilde{S}$ (denoted by $S(a_n, t_n + h)$ in the equation) over the time step [6]. Although this approach allows moderately larger h than explicit methods, it can introduce phase errors or instability if $p(t)$ varies substantially within a single step.

GRK2 refines ETD2RK by allowing the stiff coefficient $p(t)$ to vary linearly over each time step instead of assuming it is constant, described by Eq.(15). Integrating the stiff term under this linear interpolation leads to an integral in terms of error functions instead of a single exponential factor. Meanwhile, the non-stiff portion $S(t, x)$ is handled with an RK2-like approach, similar to ETD2RK but accommodating the changing $p(\tau)$. In practice, we introduce predicted values $\tilde{p}_{n+1}$ and $\tilde{S}_{n+1}$. Again, tilde is used to indicate that these quantities are evaluated using a predictor estimate of x_{n+1}. Specifically, $S_n = S(x_n, t_n)$ is the non-stiff term at the current step, $\tilde{S}_{n+1} = S(\tilde{x}_{n+1}, t_{n+1})$ is the non-stiff term evaluated at a predicted solution $\tilde{x}_{n+1}$, $p_n = p(x_n, t_n)$ is the stiff coefficient at the current step, and $\tilde{p}_{n+1} = p(\tilde{x}_{n+1}, t_{n+1})$ is the stiff coefficient at the predictor's estimate $\tilde{x}_{n+1}$.

After obtaining $\tilde{p}_{n+1}$ and $\tilde{S}_{n+1}$ from this predictor step, the GRK2 method applies the following "corrector" update to compute x_{n+1}

$$x_{n+1} = e^{-\frac{p_n + \tilde{p}_{n+1}}{2} h} x_n + \tilde{S}_{n+1} I_0(p_n, \tilde{p}_{n+1}) + \frac{S_n - \tilde{S}_{n+1}}{h} I_1(p_n, \tilde{p}_{n+1}). \tag{19}$$

Here, the functions I_0 and I_1 (see A for the definitions) are special integrals involving the error function, accounting for the variation of $p(t)$ over the step [7]. In principle, GRK2 is more stable than using a single exponential factor $e^{-p_n h}$ because it adjusts to how $p(t)$ evolves throughout the step.

In our implementation, we precompute the error function expressions for I_0 and I_1 and evaluate them numerically using standard routines (e.g., `erf` in C), keeping the method explicit.

Using these implementations, we examine both the source component $\sigma^{\mu\nu}$ and the physical component $\pi^{\mu\nu}$ in our fluid equations. This allows us to track how well each solver handles any divergence in $\sigma^{\mu\nu}$ and, in turn, how that influences the stability and magnitude of $\pi^{\mu\nu}$.

(a)

(b)

(c)

(d)
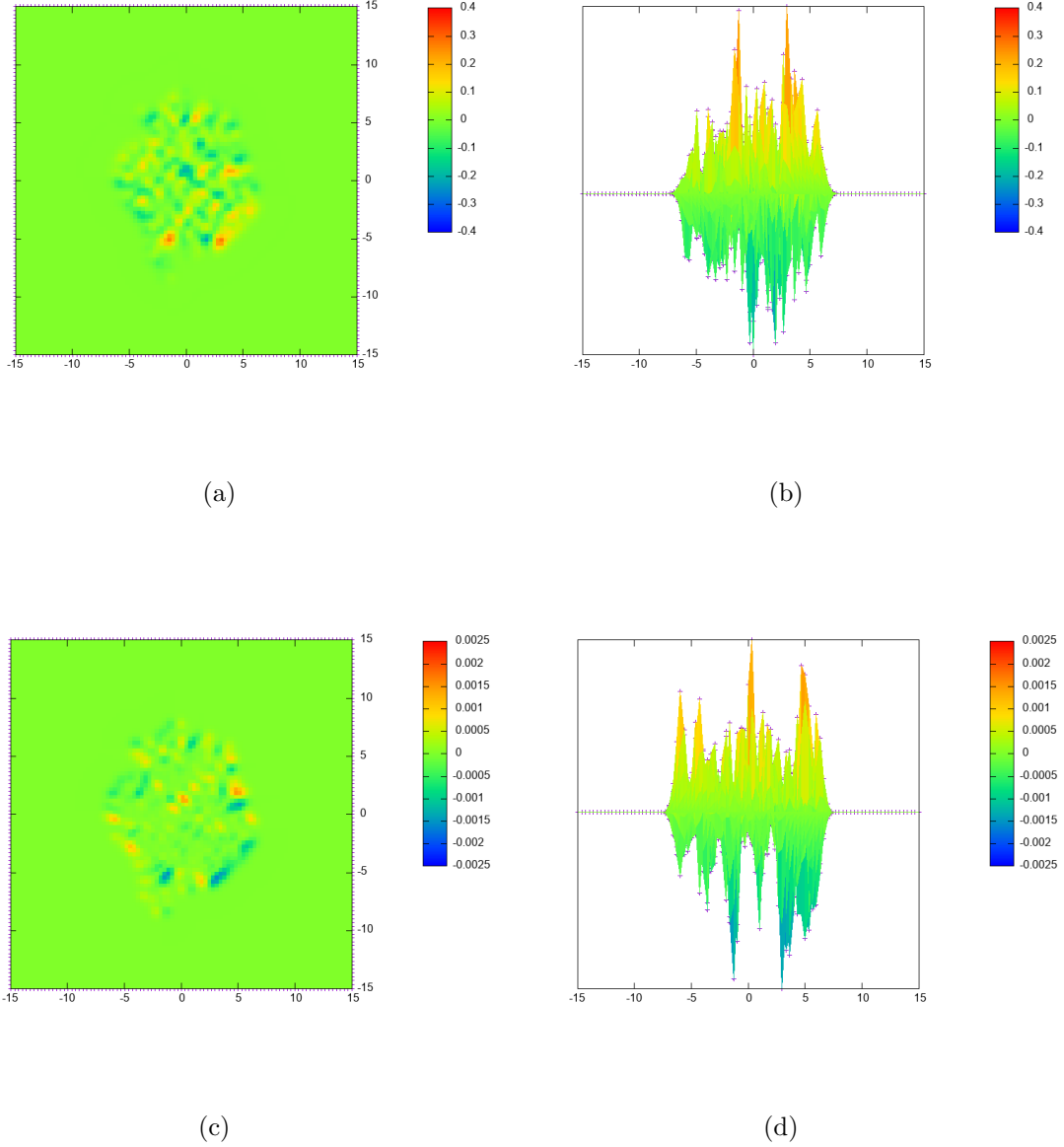
Figure 2: Distributions of the Navier-Stokes tensor $\sigma^{\mu\nu}$ and the shear tensor $\pi^{\mu\nu}$ over the x–y grid after the first time step ($t = $ h, the initial condition), viewed from the top and the front: (a) $\sigma^{\mu\nu}$ from the top, (b) $\sigma^{\mu\nu}$ from the front, (c) $\pi^{\mu\nu}$ from the top, and (d) $\pi^{\mu\nu}$ from the front. In the top-view plots (a) and (c), the horizontal axis is x, the vertical axis is y with distributions shown via rainbow color scales. In the front-view plots, the horizontal axis is x, and the vertical axis represents $\sigma^{\mu\nu}$ and $\pi^{\mu\nu}$ for (b) and (d) respectively. Both $\sigma^{\mu\nu}$ and $\pi^{\mu\nu}$ begin with the same distribution across all solvers to ensure a fair comparison.
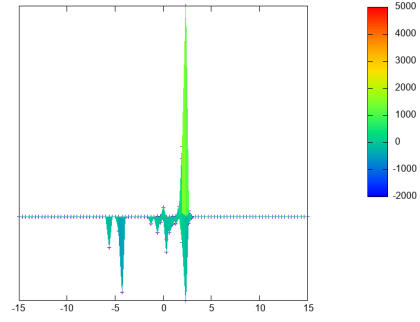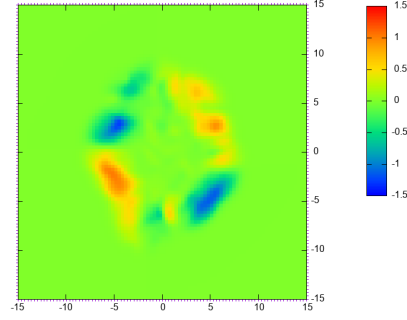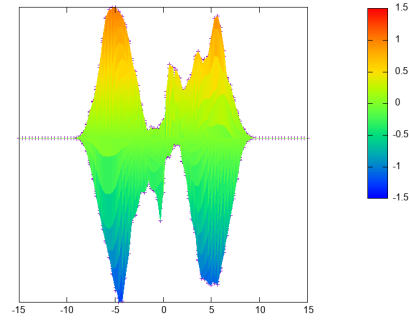
8

(a)

(b)

(c)

(d)

(e)

(f)
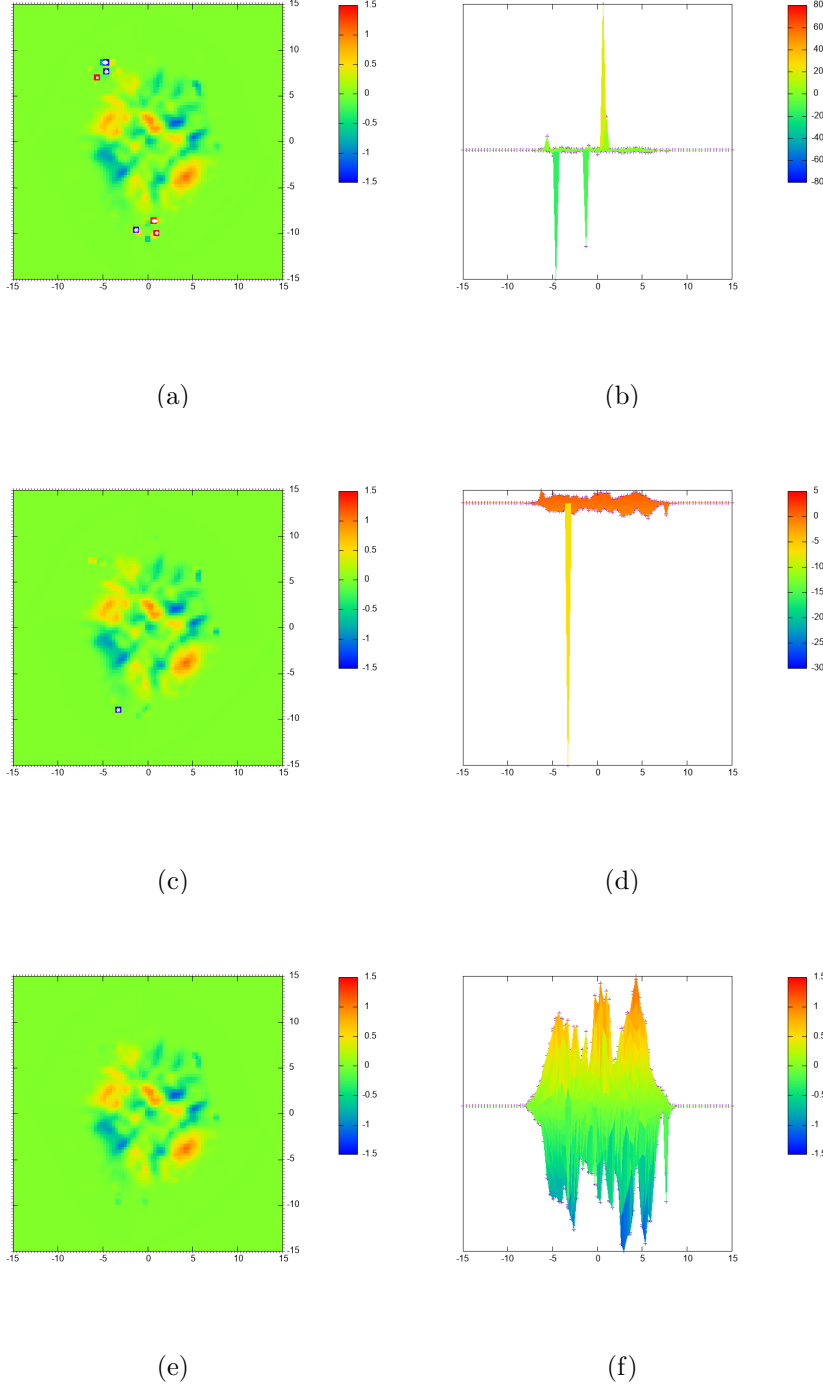
Figure 3: Distributions of $\sigma^{\mu\nu}$ over the x–y grid at $t = 150$h, comparing top and front views for each method: RK2 (a,b), RK2 + DR (c,d), and GRK2 (e,f). The top-view plots (a), (c) and (e) are placed on a comparable scale of z-axis to emphasize excessive growth (spikes) in certain regions. The front-view plots (b), (d) and (f) reflect the actual $\sigma^{\mu\nu}$ values. In (a) and (c), the blue and red dots mark negative and positive spikes, respectively.
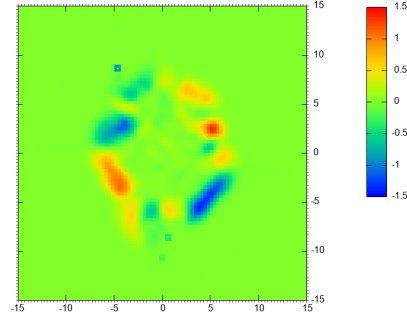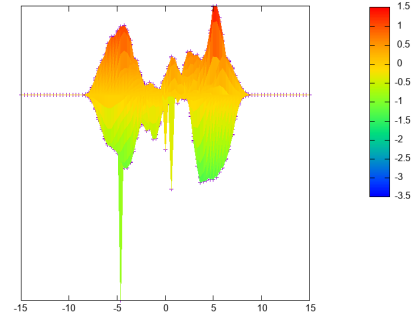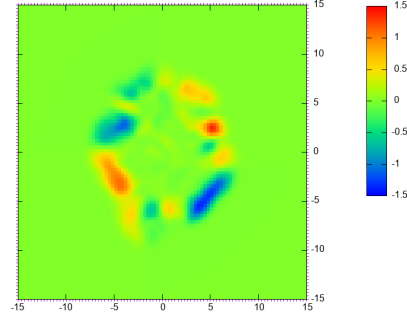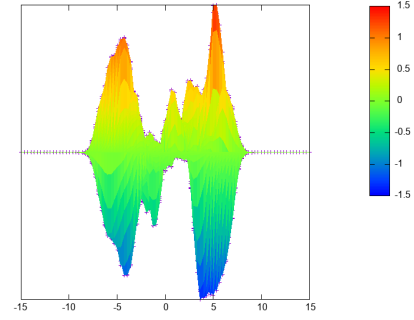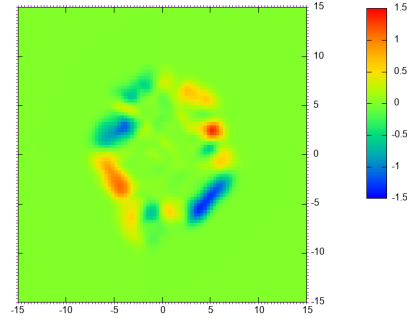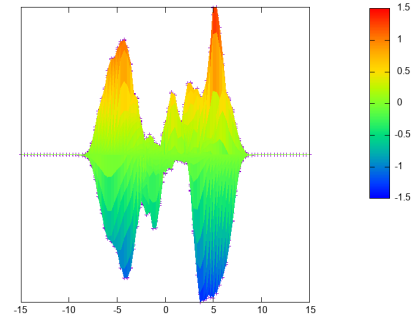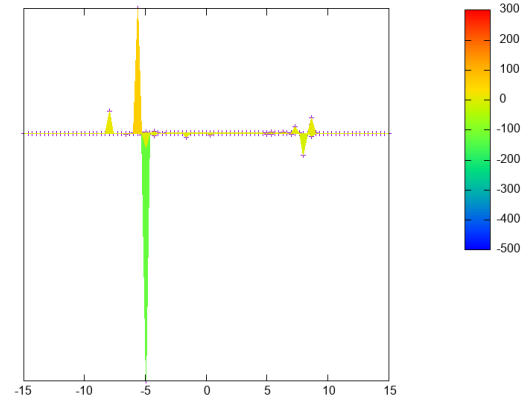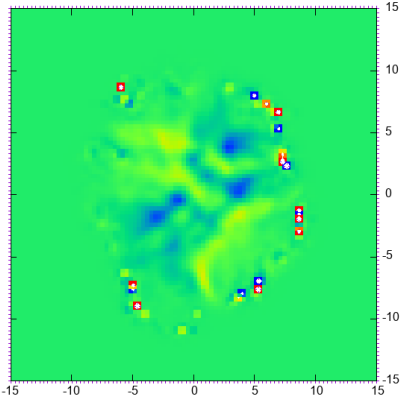
Figure 4: Distributions of $\pi^{\mu\nu}$ over the x–y grid at $t = 150$h, comparing top and front views for each method: RK2 (a,b), RK2 + DR (c,d), and GRK2 (e,f). $\pi^{\mu\nu}$ experiences excessive growth for RK2, indicated by the blue dot at around $(-5\,\text{fm}, 9\,\text{fm})$ in (a).

10
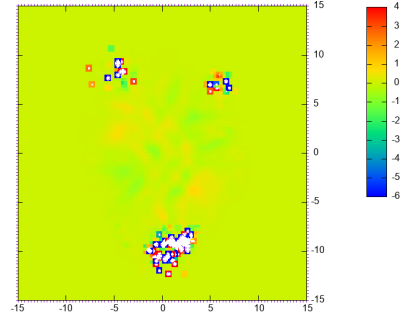
(a)
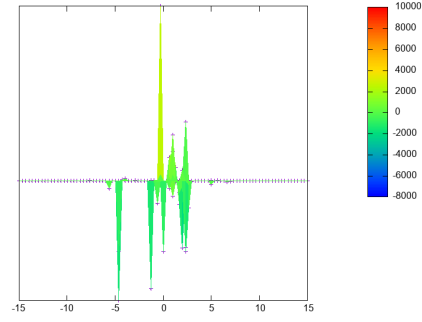
(b)
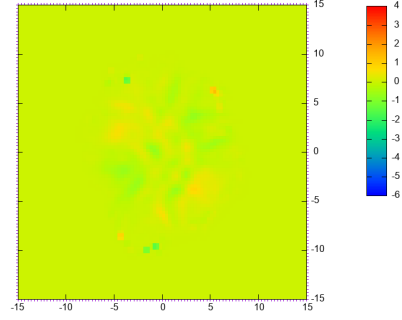
(c)
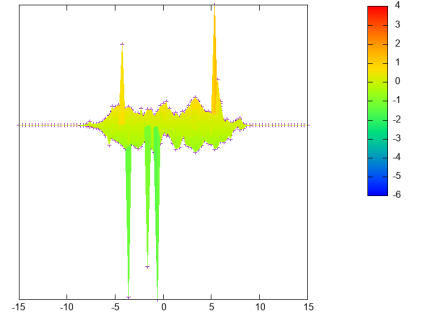
(d)
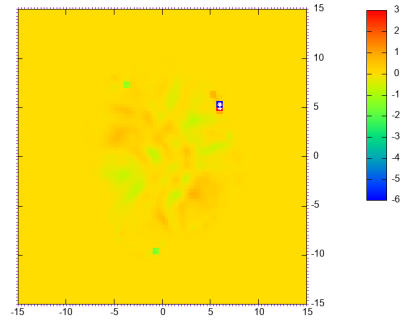
(e)
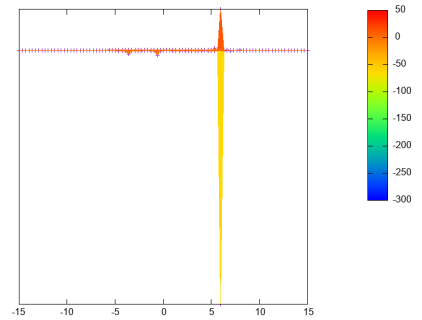
(f)

Figure 5: Distributions of $\sigma^{\mu\nu}$ over the x–y grid at $t = 200h$, comparing top and front views for each method: RK2 (a,b), RK2 + DR (c,d), and GRK2 (e,f). All three methods escalate beyond their stable ranges, but RK2 is significantly larger.
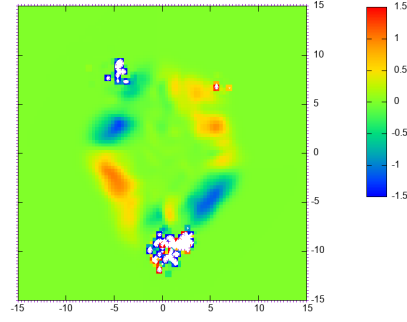
(a)

(b)

(c)
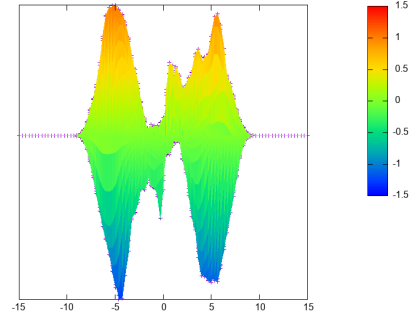
(d)
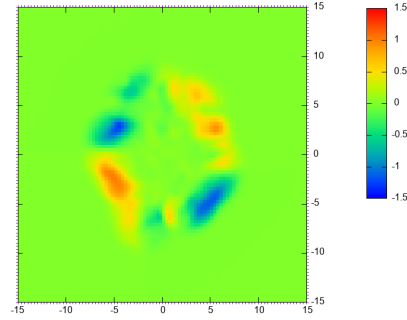
(e)

(f)

Figure 6: Distributions of $\pi^{\mu\nu}$ over the x–y grid at $t = 200$h, comparing top and front views for each method: RK2 (a,b), RK2 + DR (c,d), and GRK2 (e,f). With RK2, $\pi^{\mu\nu}$ diverges entirely, while in the other two methods it remains stable.
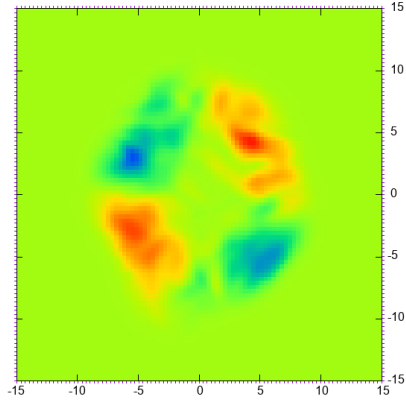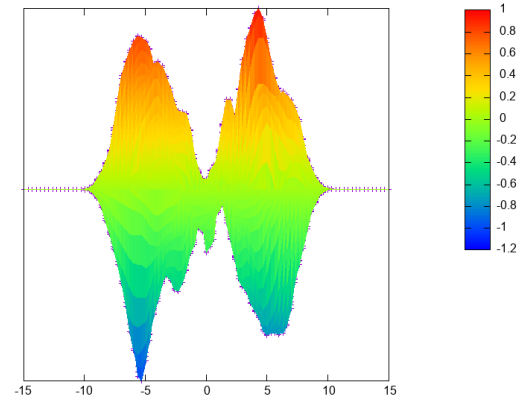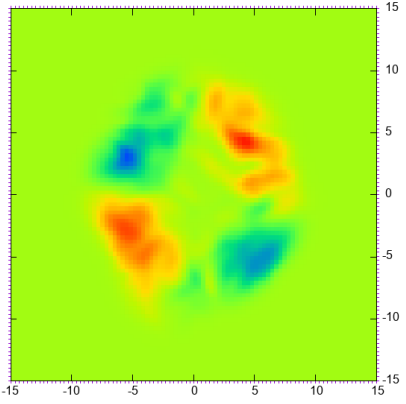
12
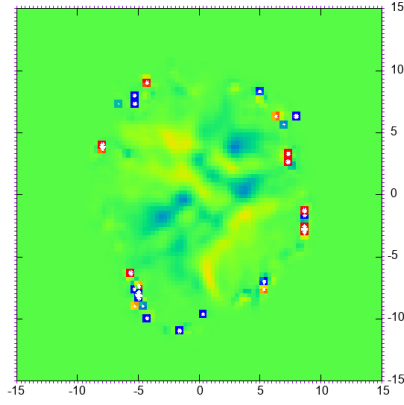
(a)

(b)

(c)
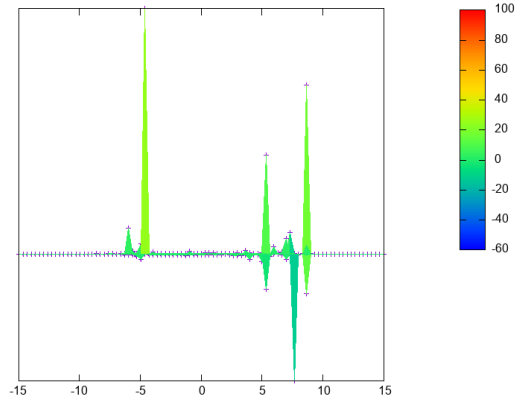
(d)

Figure 7: Distributions of $\sigma^{\mu\nu}$ over the x–y grid at $t = 300$h, comparing top and front views for RK2 + DR (a,b), and GRK2 (c,d). Both exhibit large spikes, but not as severely as RK2 did at $t = 200$h.
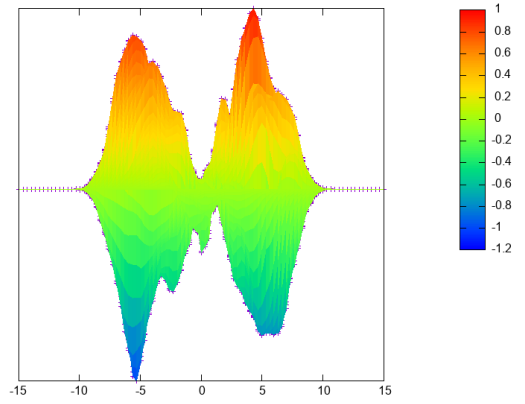
(a)

(b)

(c)

(d)

Figure 8: Distributions of $\pi^{\mu\nu}$ over the x–y grid at $t = 300$h, comparing top and front views for RK2 + DR (a,b), and GRK2 (c,d). Both remain stable and visually similar.

3 Results

In our simulations, we consider a simplified model of relativistic viscous hydrodynamics on a two-dimensional grid in the x-y plane (with $z \equiv 0$). The domain extends $30\,\mathrm{fm}$ in each direction, discretized by 90 cells, giving a cell size of $0.33\,\mathrm{fm}$. A dense fluid region is initialized near the center, simulating the initial condition of a relativistic heavy-ion collision, and evolves from an initial time of $\tau_0 = 0.4\,\mathrm{fm/c}$ to $\tau = 3.4\,\mathrm{fm/c}$. We include shear viscosity while bulk viscosity remains off. The initial energy density is largest in the central region and decreases towards the edges (vacuum).

Physically, we expect the system to expand outward from the center, with its energy density ϵ and shear tensor $\pi^{\mu\nu}$ diminishing over time as the fluid flows into the low-density regions and expand strongly in the longitudinal direction. Our primary goal is to see that each numerical method captures this outward expansion reliably without spurious instabilities or divergences so that $\sigma^{\mu\nu}$ (the source) and $\pi^{\mu\nu}$ (the physical viscous stress) remain well-behaved as the solution transitions from dense matter at the center to vacuum at the boundaries.

At $t = $ h, Figure 2 illustrates that all three methods (RK2, RK2 + DR, and GRK2) begin from the same initial condition for both $\sigma^{\mu\nu}$ and $\pi^{\mu\nu}$. The plots reveal that $\sigma^{\mu\nu}$ and $\pi^{\mu\nu}$ are mainly concentrated near the center, reflecting the chosen input parameters. Because this is just the initial setup (effectively one step of evolution), none of the methods exhibit any risk of divergence yet, and they appear identical.

At $t = 150$h, $\sigma^{\mu\nu}$ starts growing larger for both RK2 and RK2 + DR, while GRK2 remains stable, as shown in Figure 3. In the RK2 case, the excessive growth of $\pi^{\mu\nu}$ appears near $(-5\,\mathrm{fm}, 9\,\mathrm{fm})$, highlighted by the blue dot in Figure 4-(a). Even though $\sigma^{\mu\nu}$ also shows some significant growth when using DR, it suppresses the explosive growth of $\pi^{\mu\nu}$, so it remains stable and appears very similar to the GRK2 result.

At $t = 200$h, $\sigma^{\mu\nu}$ shows large growth for all methods, shown in Figure 5. However, $\sigma^{\mu\nu}$ for the RK2 + DR case scales around 10^0 and for GRK2 around 10^2, whereas RK2 surges to around 10^3 or 10^4, which is no longer acceptable. Hence, $\pi^{\mu\nu}$ escalates beyond the acceptable range for RK2 in Figure 6-(a,b). Meanwhile, the DR and GRK2 remain stable in $\pi^{\mu\nu}$ and continue to appear similar.

At $t = 300$h, RK2 has fully diverged to NaN, so no plots are available. For $\sigma^{\mu\nu}$, both RK2 + DR and GRK2 scaled to around 10^2, as in Figure 7. Despite that, $\pi^{\mu\nu}$ remains stable in both in Figure 8, and appears very similar for the two methods.

Although not the primary focus of our analysis, we have also included plots of the local energy density ϵ–another essential physical quantity in relativistic hydrodynamics–for completeness in Section B.

15

4 Discussion

In comparing how each solver handles the source component $\sigma^{\mu\nu}$ and the physical shear tensor $\pi^{\mu\nu}$, we focus on whether any unphysical growth in $\sigma^{\mu\nu}$ triggers a runaway reaction causing $\pi^{\mu\nu}$ to diverge.

At $t = 150$h, $\sigma^{\mu\nu}$ experiences some unphysical growth in both RK2 and RK2 + DR runs, but much more severely under standard RK2. Meanwhile, $\pi^{\mu\nu}$ remains stable for RK2 + DR, yet exhibits a few spikes for plain RK2, indicating the DR effectively suppresses the runaway reaction in $\pi^{\mu\nu}$ as a limiting function.

At $t = 200$h, $\sigma^{\mu\nu}$ grow excessively large again in both RK2 and RK2 + DR (with RK2 diverging much worse), whereas $\pi^{\mu\nu}$ under RK2 is entirely unphysical at this point, effectively failing as a solver. In contrast, RK2 + DR and GRK2 both remain stable.

Extending to $t = 300$h, RK2 + DR and GRK2 continue to match closely, suggesting that, for this specific hydrodynamics example, adding a regulator to RK2 can appear to "fix" the divergence in $\pi^{\mu\nu}$. However, that regulator merely suppresses potential spikes in $\pi^{\mu\nu}$ rather than reflecting genuine behavior of the runaway reaction, so it does not necessarily capture the true physics. GRK2, on the other hand, achieves solutions closer to the reference over the entire evolution without relying on such external damping term, meeting the overall objective of a numerically stable method that can handle the stiff behavior intrinsic to this relativistic viscous hydrodynamics scenario.

5 Conclusion

In this work, we addressed the numerical instability problem in relativistic viscous hydrodynamics, especially near vacuum regions where sharp velocity gradients can cause unphysical growth in the shear tensor. We implemented and compared three numerical schemes: the standard second-order Runge–Kutta method (RK2), RK2 augmented by a density-limiting regulator, and the Gaussian Runge–Kutta 2 (GRK2) method.

Our results show that while plain RK2 can fail under stiff conditions due to unbounded growth in the shear tensor, incorporating the density regulator successfully stabilizes the evolution—without requiring fine-tuned parameters—across a wide range of scenarios. By effectively suppressing excessive growth in the shear tensor, the regulator-based approach keeps the system from blowing up in problematic regions.

In contrast, GRK2 pursues a more fundamental strategy. Rather than limiting divergences through an external regulator, it handles stiffness directly via error-function-based integration and accounts for time-varying relaxation scales. Although this can theoretically introduce a small degree of numerical damping, GRK2 avoids instabilities in a natural way and maintains stability even under extreme gradients.

Overall, our findings suggest that GRK2 provides a principled method to address stiff terms in relativistic hydrodynamics, while RK2 + DR remains a reliably stable and straight-

forward solution that does not blow up in practice. Future studies can further optimize GRK2 performance, extend it to additional physical processes (such as including bulk viscosity), and apply these insights to larger-scale, more realistic systems.

References

[1] B. van der Pol, "A theory of the amplitude of free and forced triode vibrations," *Radio Review (later Wireless World)*, vol. 1, pp. 701–710, 1920. 2

[2] L. Kim and S. Y. Jeon, "Comparative analysis of RK2, ETD2RK, and GRK2 methods for stiff systems: Applications to Lotka–Volterra and Van der Pol," PHYS 479 Interim Report, McGill University, 2025. 2

[3] C. Gale, S. Jeon, and B. Schenke, "Hydrodynamic Modeling of Heavy-Ion Collisions," *Int. J. Mod. Phys. A*, vol. 28, p. 1340011, 2013. 2

[4] T. A. Brorson, "Predictor-corrector methods and runge-kutta," LibreTexts, https://math.libretexts.org/Bookshelves/Differential_Equations/Numerically_ Solving_Ordinary_Differential_Equations_(Brorson)/01%3A_Chapters/1.04% 3A_Predictor-corrector_methods_and_Runge-Kutta, accessed April 4, 2025. 5

[5] Z. Qiu, "Event-by-event Hydrodynamic Simulations for Relativistic Heavy-ion Collisions," `arXiv:1308.2182 [nucl-th]`, 2013, 16 citations counted in INSPIRE as of 04 Apr 2025. 5

[6] S. M. Cox and P. C. Matthews, "Exponential time differencing for stiff systems," *Journal of Computational Physics*, vol. 176, no. 2, pp. 430–455, 2002. 5, 6, 7

[7] S. Jeon, "A new gaussian integrator for stiff differential equations in relativistic hydrodynamics," Unpublished notes. 5, 6, 7

A Definitions of I_0 and I_1

$$I_k(a,b) = \int_0^h d\rho\, e^{-b_n\rho - a_n\rho^2}\rho^k = (-1)^k \frac{\partial^k}{\partial b^k} \int_0^h d\rho\, e^{-b\rho - a\rho^2} \tag{20}$$

for $k = 0, 1$. For k $= 0$,

$$I_0 = a\sqrt{\frac{\pi}{4|a|^3}} e^{\frac{b^2}{4a}} \left(\mathrm{erf}_{\mathrm{sign}(a)}\left(\frac{b+2ah}{2\sqrt{|a|}}\right) - \mathrm{erf}_{\mathrm{sign}(a)}\left(\frac{b}{2\sqrt{|a|}}\right) \right), \tag{21}$$

defining

$$\mathrm{erf}_+(x) = \mathrm{erf}(x) \tag{22}$$

and

$$\mathrm{erf}_-(x) = \mathrm{erfi}(x). \tag{23}$$

Similarly,

$$I_1(a,b) = \frac{1}{4|a|^{\frac{3}{2}}} \left(2\mathrm{sign}(a)\sqrt{|a|}(1-e^{-h(b+ah)}) + be^{\frac{b^2}{4a}}\sqrt{\pi}\left(\mathrm{erf}_{\mathrm{sign}(a)}\left(\frac{b+2ah}{2\sqrt{|a|}}\right) - \mathrm{erf}_{\mathrm{sign}(a)}\left(\frac{b}{2\sqrt{|a|}}\right) \right) \right). \tag{24}$$

B Plots for Local Energy Density (ϵ)

(a)   (b)
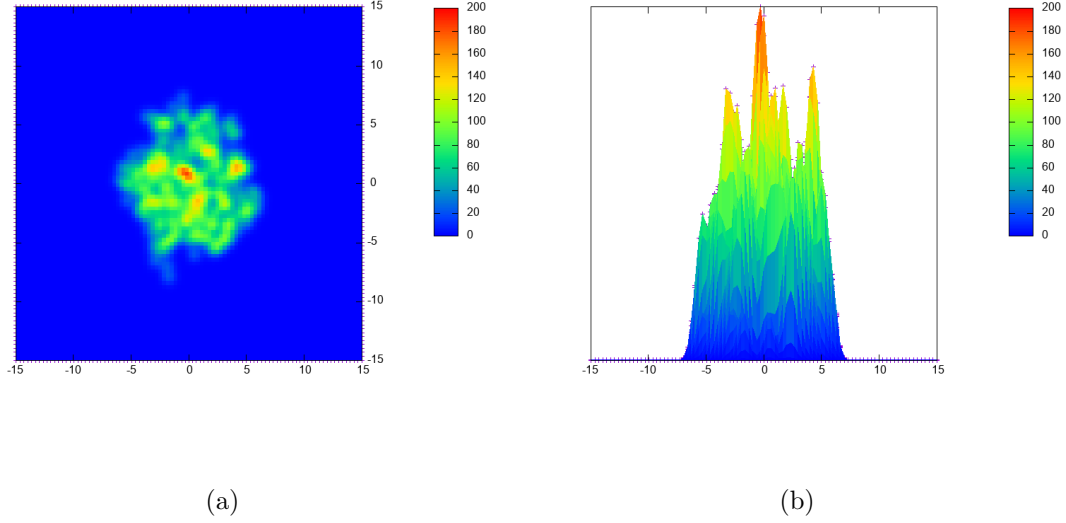
Figure 9: Distributions of the local energy density ϵ over the x-y grid after the first time step ($t = $ h, the initial condition), viewed from the top (a) and the front (b). The same distribution of ϵ is observed across all solvers.
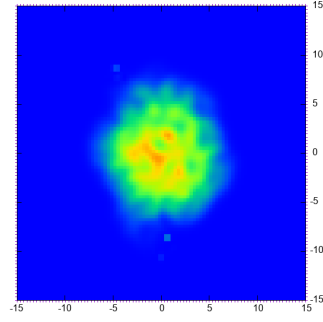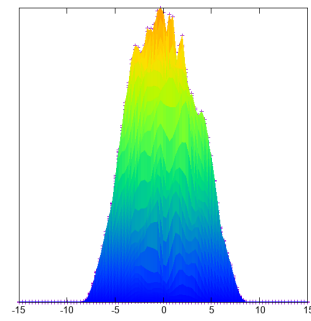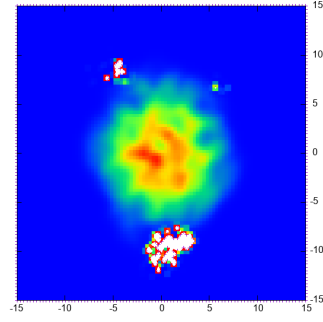
(a)
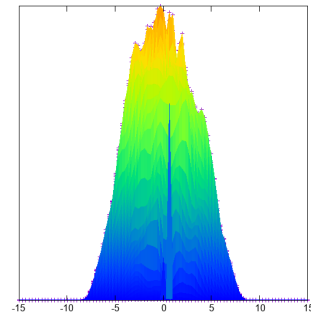
(b)
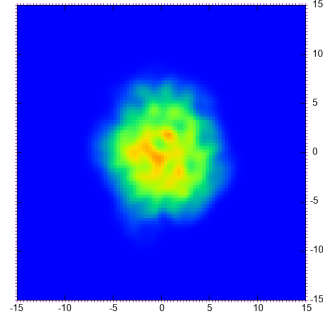
(c)
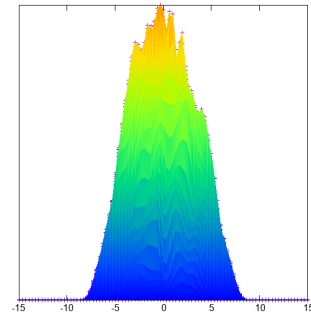
(d)
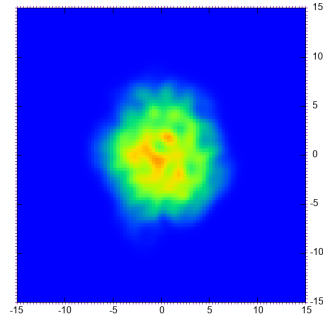
(e)

(f)

Figure 10: Distributions of ϵ over the x–y grid at $t = 150$h, comparing top and front views for each method: RK2 (a,b), RK2 + DR (c,d), and GRK2 (e,f). ϵ starts showing spikes for RK2, indicated by the two green dots in (a).
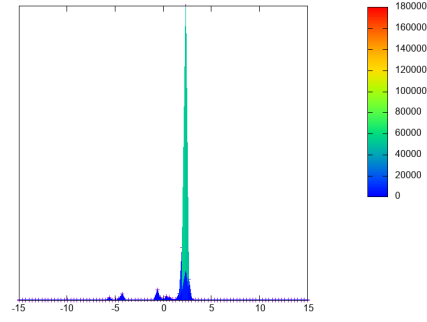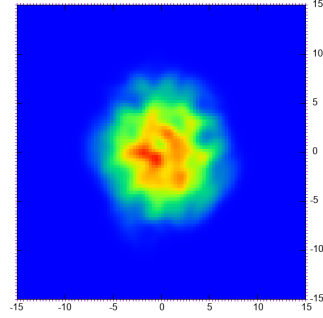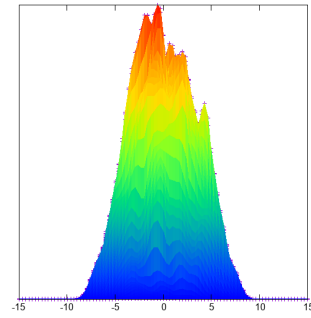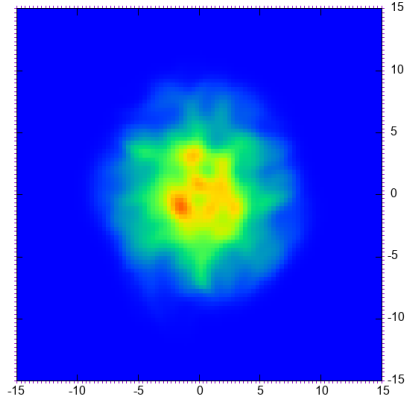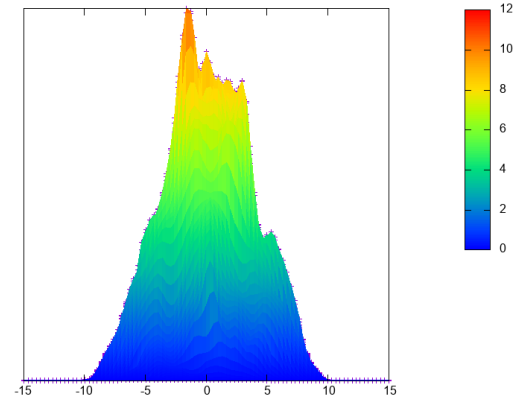
(a)   (b)
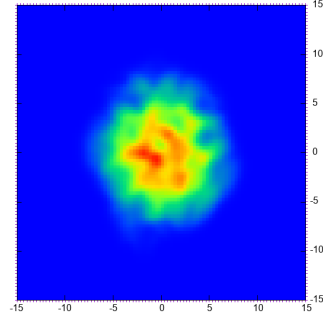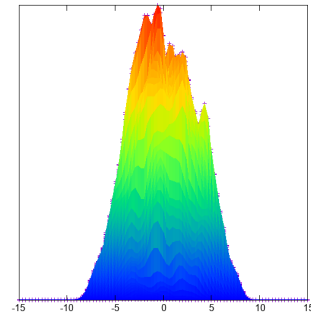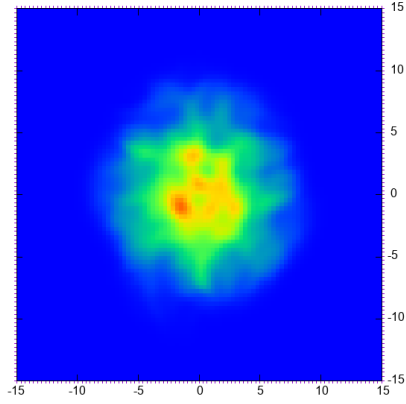
(c)   (d)

(e)   (f)

Figure 11: Distributions of ϵ over the x–y grid at $t = 200$h, comparing top and front views for each method: RK2 (a,b), RK2 + DR (c,d), and GRK2 (e,f). For $RK2$, ϵ diverges entirely, while in the other two methods it remains stable.
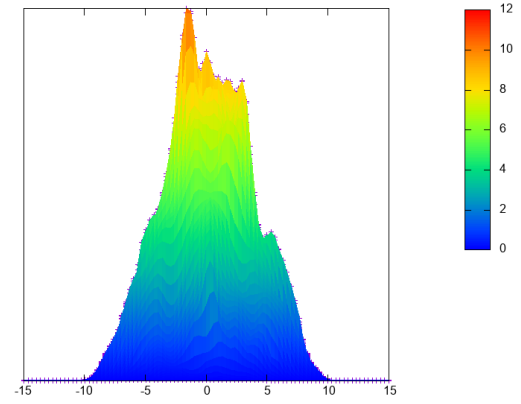
22

(a)

(b)

(c)

(d)

Figure 12: Distributions of ϵ over the x–y grid at $t = 300$h, comparing top and front views for RK2 + DR (a,b), and GRK2 (c,d). Both remain stable and visually similar.