Efficient method for solving highly oscillatory ordinary differential equations with applications to physical systems

F. J. Agocs [1,2,*] W. J. Handley,[1,2,†] A. N. Lasenby,[1,2,‡] and M. P. Hobson[1,§]

[1]*Astrophysics Group, Cavendish Laboratory, J. J. Thomson Avenue, Cambridge CB3 0HE, United Kingdom*
[2]*Kavli Institute for Cosmology, Madingley Road, Cambridge CB3 0HA, United Kingdom*

We present a numerical routine (OSCODE) with a C++ and Python interface for the efficient solution of one-dimensional, second-order, ordinary differential equations with rapidly oscillating solutions. The method is based on a Runge-Kutta-like stepping procedure that makes use of the Wentzel-Kramers-Brillouin approximation to skip regions of integration where the characteristic frequency varies slowly. In regions where this is not the case, the method is able to switch to a made-to-measure Runge-Kutta integrator that minimizes the total number of function evaluations. We demonstrate the effectiveness of the method with example solutions of the Airy equation and an equation exhibiting a burst of oscillations, discussing the error properties of the method in detail. We then show the method applied to physical systems. First, the one-dimensional, time-independent Schrödinger equation is solved as part of a shooting method to search for the energy eigenvalues for a potential with quartic anharmonicity. Then, the method is used to solve the Mukhanov-Sasaki equation describing the evolution of cosmological perturbations, and the primordial power spectrum of the perturbations is computed in different cosmological scenarios. We compare the performance of our solver in calculating a primordial power spectrum of scalar perturbations to that of BINGO, an efficient code specifically designed for such applications, and find that our method performs better.

I. INTRODUCTION

Runge-Kutta (RK) methods are powerful tools for numerically solving systems of first-order ordinary differential equations, and as such are often the default option in numerical routines for this task. There are cases however when more efficient methods are needed than Runge-Kutta, such as where the solution exhibits rapid oscillations. Problems classified as oscillatory are common in physics, yet the set of tools available to solve oscillatory systems efficiently is small, and problems are often treated on a case-by-case basis, using analytic approximations such as the Wentzel-Kramers-Brillouin (WKB) method [1].

In this paper, we develop a method for a more general solution, motivated by the Mukhanov-Sasaki equation [2], which governs the time evolution of curvature perturbations in the early universe. It has the form of a generalized oscillator with a time-dependent frequency and a first-order derivative term present, the frequency depending on the characteristic

*fa325@cam.ac.uk
†wh260@cam.ac.uk
‡a.n.lasenby@mrao.cam.ac.uk
§mph@mrao.cam.ac.uk

wave number of the perturbation. For inference in cosmology from the cosmic microwave background, it is necessary either to assume an approximate form for the primordial power spectrum of curvature perturbations, or to solve the Mukhanov-Sasaki equation for a range of characteristic wave numbers to compute a spectrum (see, e.g., [3] for a thorough review). In the event of single-field slow-roll inflation, most models lead to a scale-invariant primordial power spectrum [4] which can be obtained analytically [5], but models that introduce features in the primordial power spectrum can improve the fit to cosmic microwave background (CMB) observations [6]. In such cases when one relies on a numerical solution of the Mukhanov-Sasaki equation, in regions where the perturbation is oscillatory, this is a challenging task for Runge-Kutta-based methods. Runge-Kutta solvers such as BINGO [7] can prove efficient for some single-field inflation models by taking a shortcut and not integrating the perturbation throughout its oscillatory phase [8].

There has been a proposal for an algorithm in Ref. [9] that generalizes the Runge-Kutta stepping procedure, but uses the WKB approximation to forecast the solution instead of a Taylor expansion when the solution is highly oscillatory. The proposed algorithm was named RKWKB, and while it served as the theoretical foundation of our present work, there are a number of key differences. Most importantly, we extended the algorithm so that it can be applied to damped oscillators, and to equations without closed-form frequency and first-derivative terms. We also made significant adjustments to the adaptive step-size algorithm and the method to evaluate whether the WKB approximation is applicable at the current

time step. These modifications are outlined in Sec. II and detailed in Appendices A–D.

We present a general purpose solver for differential equations of the form

$$\ddot{x}(t) + 2\gamma(t)\dot{x}(t) + \omega^2(t)x(t) = 0, \tag{1}$$

where γ and ω may or may not be expressed as a closed-form function of time. If they cannot be, but depend on time though a set of "background" variables that can be obtained numerically, they may be supplied to the solver as array-like data structures sampled over time, as detailed in Sec. II E. Since the efficiency of the solver relies on the WKB approximation being valid for a portion of the integration range, the solver is intended for problems where the frequency is slowly varying (relative to the timescales of the problem) for a part of the integration range.

This paper is structured as follows. In the next section we present an overview of the algorithm and the methods used therein, leaving some details to appendices. This is followed by applications of the method to physical systems in Sec. III. Section IV discusses factors the user needs to be aware of that might limit the performance of the solver, as well as future improvements and extensions. We conclude in Sec. V with a short summary.

II. METHODS

A. Overview

The basis for our solver is the generalized stepping approach detailed in [9], which we will summarize here. Having a numerical estimate for the solution x and its derivative $\dot{x}$ at time t, the solution at a later time $t + h$ is obtained. Then, using an error estimate on the proposed step, the step size h is updated such that the error estimate stays within a local tolerance limit. Such adaptive control of the step size is a requirement for robust numerical solvers. Starting from two functions $f_\pm(t)$ that form an appropriate basis set for the true solution of the second-order differential equation and are linearly independent at all t, we match the correct solution and its derivative by linearly combining $f_\pm$ and their derivatives:

$$x(t + h) = A_+ f_+(t + h) + A_- f_-(t + h) \tag{2}$$

and

$$\dot{x}(t + h) = B_+ \dot{f}_+(t + h) + B_- \dot{f}_-(t + h), \tag{3}$$

where

$$A_\pm = \frac{\dot{x}(t)f_\mp(t) - x(t)\dot{f}_\mp(t)}{\dot{f}_\pm(t)f_\mp(t) - \dot{f}_\mp(t)f_\pm(t)} \tag{4}$$

and

$$B_\pm = \frac{\ddot{x}(t)\dot{f}_\mp(t) - \dot{x}(t)\ddot{f}_\mp(t)}{\ddot{f}_\pm(t)\dot{f}_\mp(t) - \ddot{f}_\mp(t)\dot{f}_\pm(t)}. \tag{5}$$

In the above, $\ddot{x}(t)$ may be obtained from the differential equation itself, using $x(t)$ and $\dot{x}(t)$. It it shown in [9] that the above procedure reduces to Euler's method in the limit of vanishing step size h and with the appropriate choice of $f_\pm$. The above approach therefore allows one to pick trial solutions $f_\pm$ that approximate the true solution well over a

larger range than an nth-order polynomial, which would be the choice for $f_\pm$ in the case of an nth-order Runge-Kutta method.

As [9] suggests, the WKB method can be used to derive an analytic approximation to the true solution of a single oscillator on timescales much shorter than $\frac{\omega}{\dot{\omega}}$, the timescale on which the frequency changes. The WKB solutions, detailed in the following subsection, are ideal candidates for $f_\pm$ over such timescales.

In general however, the frequency cannot be expected to vary slowly over the entire range of integration, and the WKB solutions might not always be a good choice for $f_\pm$. To counter this, a dynamic switching mechanism is included in the solver, which consists of attempting two steps of size h simultaneously. First, a Runge-Kutta step of order 5 is calculated (an "RK step" hereafter), then a step using Eqs. (2)–(5), with $f_\pm$ set to the WKB solutions (a "WKB step"). Based on the error estimates on each of these, the next step size, h^*, is computed. The step with the larger next predicted step size is chosen. This is to minimize the number of steps the solver needs to take to achieve a given local accuracy, and hence minimize runtime. The step with the chosen method may be accepted or rejected, and the step size h increased or decreased.

The two methods are described in greater detail in the subsections that follow, with their error estimates discussed in Appendix A. Details of switching between methods and updating the step size can be found in Appendix C. Finally, a step-by-step summary of the algorithm is given in Appendix D.

B. Wentzel-Kramers-Brillouin solutions

Starting from the equation

$$\ddot{x}(t) + 2\gamma(t)\dot{x}(t) + \omega^2(t)x(t) = 0, \tag{6}$$

we wish to derive asymptotic expansions of the two independent solutions, in the limit that ω is slowly varying relative to x, but γ need not be so. In the absence of a first-derivative term the derivation starts by introducing a power-counting parameter T, with $T \gg 1$:

$$\ddot{x} + T^2\omega^2 x = 0. \tag{7}$$

If we now insert γ and allow it to vary on shorter timescales than ω, the equivalent equation to consider is

$$\ddot{x} + 2\gamma\dot{x} + T^2\omega^2 x = 0. \tag{8}$$

Following [1], one can then seek asymptotic approximations in the form of an exponential power series[1]

$$x(t) \sim \exp\left(T \sum_{n=0}^{\infty} S_n(t)T^{-n}\right). \tag{9}$$

---

[1]Note that the following expression appears erroneously in [9], in that T should be replaced with T^{-1}.

Substituting (9) into (8), setting coefficients of powers of T to zero, one arrives at the recursion

$$\dot{S}_0(t) = \pm i\omega,$$
$$\dot{S}_i(t) = -\frac{1}{2S_0'}\left(\ddot{S}_{i-1} + 2\gamma\dot{S}_{i-1} + \sum_{j=1}^{i-1}\dot{S}_j\dot{S}_{i-j}\right). \quad (10)$$

The first four terms in the asymptotic series in the presence of a first-derivative term are

$$S_0 = \pm i\int \omega\, dt,$$
$$S_1 = -\frac{1}{2}\ln\omega - \int \gamma\, dt,$$
$$S_2 = \pm i\int -\frac{1}{2}\frac{\gamma^2}{\omega} - \frac{1}{2}\frac{\dot{\gamma}}{\omega} + \frac{3}{8}\frac{\dot{\omega}^2}{\omega^3} - \frac{1}{4}\frac{\ddot{\omega}}{\omega^2}\, dt, \quad (11)$$
$$S_3 = \frac{1}{4}\frac{\gamma^2}{\omega^2} + \frac{1}{4}\frac{\dot{\gamma}}{\omega^2} - \frac{3}{16}\frac{\dot{\omega}^2}{\omega^4} + \frac{1}{8}\frac{\ddot{\omega}}{\omega^3}.$$

As [1] states, the WKB series is a singular perturbative expansion. The sum in (9) is usually divergent (unless it truncates) and needs to be truncated at some term in order to be a good approximation to $x(t)$. To use (9) as an approximate solution to (6), one needs to set $T = 1$. This is allowed despite having assumed $T \gg 1$ (see, e.g., [1]) as long as the asymptotic inequalities

$$TS_0(t) \gg S_1(t) \gg \cdots \gg T^{1-n}S_n(t) \quad (12)$$

hold uniformly within the interval $[t, t + h]$. If the asymptotic inequalities are satisfied and the first term not included in the asymptotic WKB series is small,

$$T^{-n}S_{n+1}(t) \ll 1, \quad (13)$$

$x(t) \sim \exp\left[T \sum_{i=0}^{n} T^i S_i(t)\right]$ is a good approximation.

To utilize the WKB solutions, we set $f_\pm(t)$ to $x(t)$ according to (9) with $T = 1$. Computing a WKB step from t to $t + h$ thus involves

$$S_i(t + h) - S_i(t) = \int_t^{t+h} \dot{S}_i(t')\, dt'. \quad (14)$$

If the solver enters an integration region suitable for being approximated by WKB solutions, the step size h is expected to increase, and the error on the integrals (14) is expected to dominate the error on x and $\dot{x}$ in WKB steps. Although in these regions ω changes slowly, care needs to be taken to evaluate the integrals accurately. As $\omega(t)$ and $\gamma(t)$ may not be available in closed form, the integrals are computed numerically.

C. Numerical integration and differentiation

We chose to calculate the integrals (14) using a method from the Gaussian quadrature family [10], Gauss-Lobatto integration [11].

Gaussian quadrature formulas work by modeling the integrand as a linear combination of appropriately chosen mutually orthogonal polynomials. As a side effect, a certain class of integrands makes the integral exact (in the Gauss-Lobatto case, polynomials of degree $2n - 3$, where n is the number of abscissas). Typically the remainder in such methods is proportional to a higher-order ($2n - 2$ for Gauss-Lobatto) derivative of the integrand, which we expect to be small in integration regions of interest, where WKB is a good approximation and several oscillations can be stepped over, such that $h \gg \frac{2\pi}{\omega}$. This property makes Gaussian quadrature superior to integrating the $\dot{S}_i$ with a Runge-Kutta step, as the latter would approximate the integral from t to $t + h$ with a Taylor expansion around t, with an error as some power of h. Gaussian quadrature methods are also desirable because they converge exponentially fast with n, due to the order of the method increasing with n as well as the density of points of evaluation [12]. This makes them a better choice than Newton-Cotes methods with equally spaced abscissas, such as the trapezoidal rule or Simpson's method.

Gauss-Lobatto integration with $n = 6$ was chosen in particular because the abscissas it uses include the beginning and end points of integration, t and $t + h$. This makes it an FSAL (first same as last) method, and one could design a 5th-order, 6-stage Runge-Kutta formula based on the same abscissas, minimizing the number of evaluations of $\omega(t)$ and $\gamma(t)$ during a single step of the algorithm (WKB and RK). The remainder on a Gauss-Lobatto integral is given analytically, but since it involves the $(2n - 2)$th derivative of the integrand, it is more common to be estimated as the difference between the results with n and $n - 1$ abscissas.

The integrands in (11) contain derivatives of ω and γ, which may not be available in closed form, and hence will also be calculated numerically. Since we already need to evaluate ω and γ at a total of 9 distinct points (Gauss-Lobatto abscissas) for the integrals in S_0 and S_1, it is worth reusing these values and deriving finite-difference formulas using them as stencil points [13], by solving

$$\begin{bmatrix} w_1 \\ \vdots \\ w_n \end{bmatrix} = \frac{1}{h^D}\begin{bmatrix} s_1^0 & \cdots & s_n^0 \\ \vdots & \ddots & \vdots \\ s_1^{n-1} & \cdots & s_n^{n-1} \end{bmatrix}^{-1}\begin{bmatrix} 0 \\ \vdots \\ D! \\ \vdots \\ 0 \end{bmatrix}, \quad (15)$$

where the s_i define the stencil such the points of evaluation are $t_i = t + s_i h$, D is the order of derivative desired, and the $D!$ is the $(D + 1)$th entry in the vector on the right-hand side. The w_i are the resulting weights of the function evaluations:

$$\left.\frac{d^D f}{dt^D}\right|_t = \sum_{i=1}^{n} w_i f(t_i). \quad (16)$$

The finite-difference formulas above work by canceling the first D terms in the Taylor expansion of f around t and setting the coefficient of the $(D + 1)$th term to 1. These give $D + 1$ constraints, but since we are free to choose the weights of n_s evaluations of f (where n_s is the number of stencil points), we can cancel a further $n_s - D - 1$ terms in the Taylor series and thus get a result that is accurate to $O(h^{n_s-D})$. While h is expected to be large in a region of integration where WKB is a good approximation, the coefficient multiplying h^{n_s-D} is expected to be small (since the derivatives of ω are expected to be small), and therefore we argue that finite differences is an acceptable way to estimate derivatives of ω and γ.

TABLE I. Butcher tableau for an explicit Runge-Kutta method.

0					
c_2	a_{21}				
c_3	a_{31}	a_{32}			
$\vdots$	$\vdots$	$\vdots$	$\ddots$		
c_s	a_{s1}	a_{s2}	$\cdots$	$a_{s,s-1}$	
	b_1	b_2	$\cdots$	b_{s-1}	b_s

D. Explicit Runge-Kutta formulas based on Gauss-Lobatto stencil points

Runge-Kutta methods are a wide family of solvers that approximate the solution to a system of first-order ordinary differential equations,

$$\dot{\mathbf{y}}(\mathbf{t}) = \mathbf{f}(\mathbf{y}, t). \tag{17}$$

They do so by expressing the solution at a later time $t + h$ as

$$\mathbf{y}(t + h) = \mathbf{y}(t) + \sum_{i=1}^{s} b_i \mathbf{F}_i, \tag{18}$$

$$\mathbf{F}_i = \mathbf{f}(\mathbf{Y}_i, t + c_i h), \tag{19}$$

$$\mathbf{Y}_i = \mathbf{y}(t) + h \sum_{j=1}^{s} a_{ij} \mathbf{F}_j. \tag{20}$$

For the present problem, the system to be solved is

$$\mathbf{y} = (x, \dot{x}) = (y_1, y_2), \tag{21}$$

$$\mathbf{F} = (y_2, -\omega^2(t)y_1 - 2\gamma(t)y_2). \tag{22}$$

Explicit formulas are a subset of the family for which the sum in (20) on j runs until $j < i$. These are of particular interest because the $\mathbf{Y}_i$ can be calculated in an iterative manner (rather than having to solve a system of equations for them). The coefficients a_{ij}, c_i, and b_i fully determine the method, and can be compactly summarized in a Butcher tableau, shown in Table I.

Although there exist implicit methods with few intermediate points (or stages, s) that are based on Gaussian quadrature [14], most Runge-Kutta formulas work on the basis of Taylor-expanding both $\mathbf{y}$ and the linear combination of function evaluations on the right-hand side of (18) around t by an amount h, then matching coefficients of powers of h up until a given order. The equations resulting from counting powers of h are called order constraints, and can be derived with the help of graph theory (as detailed in [14]). Particularly efficient (so-called embedded) algorithms use the same function evaluations $\mathbf{F}_i$ to match coefficients to order N and $N - 1$, thus producing two estimates on $\mathbf{y}(t + h)$ whose difference can be used as an error estimate.

For most combinations of the number of stages s and desired order of accuracy N, the order constraints do not pin down all entries in the Butcher tableau, and the leftover degrees of freedom are often fixed by minimizing the coefficient of the leading-order term in the local error. An efficient embedded (4, 5) pair developed by [15] demonstrates this, and is used in RKSUITE [16] (used by the NAG Library [17]) as

one of the possible Runge-Kutta formulations. For the present problem, we are interested in solving the order constraints of a 6-stage, 5th-order method,[2] with the c_i set to the Gauss-Lobatto abscissas for $n = 6$, and a 4-stage, 4th-order method with its c_i equal to the Gauss-Lobatto abscissas for $n = 5$ with the exception of the midpoint. This way we can recycle the evaluations of ω and γ at the abscissas to calculate the integrals in (14), estimate their errors, take a Runge-Kutta step in $x, \dot{x}$, and get their error estimates all at the same time. The order constraints for this system can be solved symbolically with no leftover degrees of freedom, demonstrated in [18]. The resulting coefficients are summarized in the form of Butcher tableaux in Appendix B.

E. Defining $\omega(t)$ and $\gamma(t)$

In many problems of interest, the frequency and the friction term will not be explicit functions of time, but functions of variables that depend on time through a set of differential equations that may only be solved numerically. The algorithm requires the values of $\omega(t)$ and $\gamma(t)$ to be known at 9 distinct points in each step along the solution, but is otherwise blind to how the functions are defined. In order for the solver (and in particular Gauss-Lobatto integration) to work reliably, the frequency and friction terms need to be known at any time point within the integration range to high (at least 1 in 10^9) accuracy.

For convenience the solver has been set up such that the user can provide values of the functions (or their natural logarithms) as vectors evaluated on an evenly spaced, monotonically increasing grid over time. It will then carry out linear interpolation whenever a function evaluation is required. The even spacing in the independent variable is a requirement for the sake of speed, as it simplifies the search for the nearest grid points ahead of the interpolation.

If evaluation on an evenly spaced grid is not possible or the grid cannot be made fine enough for linear interpolation to be sufficiently accurate, the user may define $\omega(t)$ and $\gamma(t)$ as interpolated functions using a suitable interpolation method.

III. APPLICATIONS

A. Airy equation

We first demonstrate the efficiency of the solver when applied to the Airy equation,

$$\ddot{x} + tx = 0, \tag{23}$$

which has the solution $\mathrm{Ai}(-t) + i\,\mathrm{Bi}(-t)$. All derivatives of ω decrease with time, hence the algorithm is expected initially to perform RK steps and at a later time switch to WKB, making the Airy equation an ideal example to test both the accuracy of the WKB steps and the step-size-update procedure. This behavior is illustrated qualitatively in Fig. 1. The second panel in Fig. 1 then details the error properties of the RK and WKB phases, and shows that while the global relative error grows in RK steps, it levels off once the WKB phase is entered.

---

[2]The highest order a 6-stage method can achieve is 5, as proven in [14].
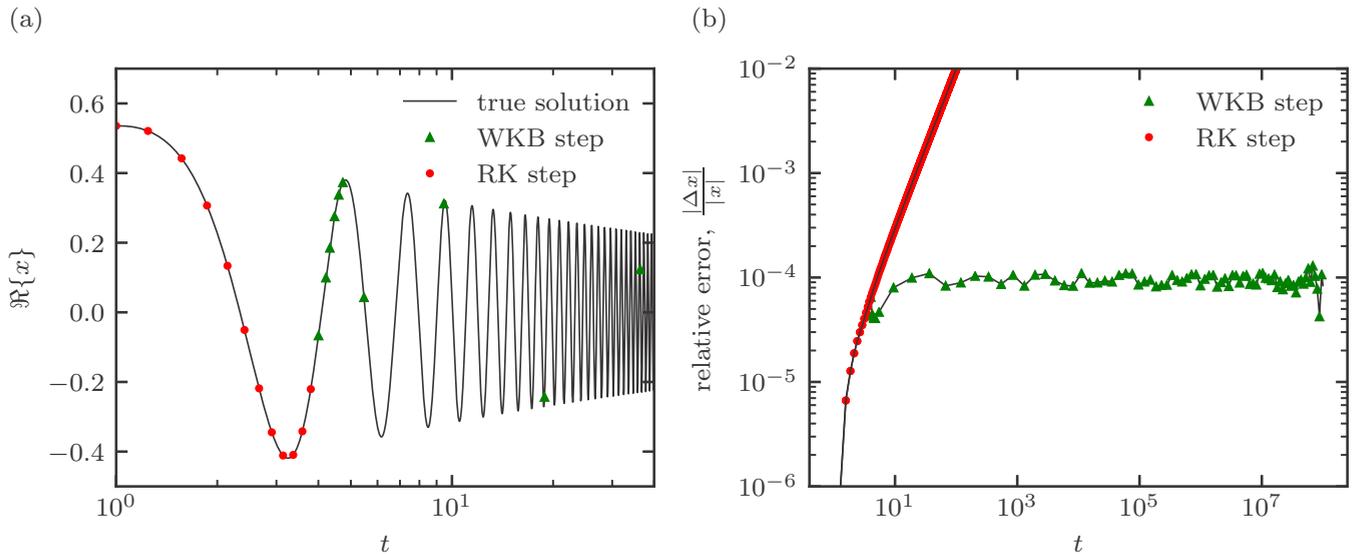
(a)

(b)

FIG. 1. Numerical solution of the Airy equation obtained with the solver (dots and triangles), overlaid on the true solution as computed by the BOOST math library. The algorithm exhibits a clear switch from taking RK steps to WKB steps at around $t \approx 4$, as expected. Despite the t axis being logarithmic, the step-size increase is clearly visible as time increases, and the rate of change of ω decreases. Also shown is the accumulation of relative error during the numerical solution of the Airy equation until late times, showing the difference between a purely RK-based approach and RKWKB (OSCODE). The relative tolerance was set to be 10^{-4}, which the RKWKB solution does not exceed, but navigates such that the largest possible steps are taken while staying within this limit. In contrast, a solver taking only RK steps quickly decreases its steps while accumulating error.

In contrast, for a pure RK method, the step size decreases while the relative error continues growing. The RKWKB-based solver (OSCODE) has no difficulty stepping through the Airy solution until times as late as 10^8, at which point the step size becomes too large to store $[S_i]_t^{t+h}$ with the required precision. This limitation is discussed in Sec. IV.

B. Burst equation

To illustrate the switching mechanism between RK/WKB steps, we next apply the solver to the equation

$$\ddot{x} + \frac{n^2 - 1}{(1 + t^2)^2} x = 0. \quad (24)$$

A solution for this system is

$$x(t) = \frac{\sqrt{1 + t^2}}{n} [\cos(n \arctan t) + i \sin(n \arctan t)], \quad (25)$$

characterized by a burst of approximately $n/2$ oscillations in the region $|t| < n$. The exact solution and numerical estimates of the solution at the steps taken by our solver are shown in Fig. 2.

Figure 2 also shows the error accumulated in the numerical solution of this example. This clearly shows that once the burst of oscillations is encountered, taking WKB steps becomes more efficient, and the solver allows the step size to grow until the local error reaches its tolerance limit. It then keeps the local error at this limit while traversing as many oscillations as possible. The global error is also seen to level off, at a slightly higher value than the local tolerance. To demonstrate that as many oscillations are stepped over as possible, Fig. 3 shows the number of oscillations traversed during a single step of the

solver as a function of time having a sharp peak near $t = 0$, where it is able to leap through 10^4 oscillations.

The robustness of the algorithm was tested by monitoring the numerical error as a function of time for relative tolerances ranging from 10^{-6} to 10^{-4}, shown in Fig. 4. The global error in all of the above examples reaches a constant value of $\sim 10 \times$ rtol by the end of the oscillatory phase.

Finally, we show that the algorithm is efficient over a range of values of n (which determine the total number of oscillations) and tolerances in Figs. 5 and 6. The algorithm shows a slow, 4-fold runtime increase over 9 orders of magnitude change in the number of oscillations, which is due to the increase in WKB steps needed to traverse the oscillatory region, shown in Fig. 7. Figure 6 also reveals that the algorithm is most efficient in the relative tolerance range of 10^{-6} to 10^{-4}. For tolerances lower than this, a $(4, 5)$th-order RK pair is not generally recommended.

C. Schrödinger equation

The one-dimensional time-independent Schrödinger equation for a potential $V(x)$ takes the form

$$\Psi''(x) + 2m[E - V(x)]\Psi(x) = 0, \quad (26)$$

where we set $\hbar = 1$. The WKB method's original use was to compute approximate solutions of (26), which suggests that our solver can be used as an alternative to traditional methods (such as the Numerov method [19]) to calculate fast numerical solutions. Starting with an analytic example, Fig. 8 shows the numerical evaluation of the energy eigenfunction Ψ_n for the nth energy level in a harmonic potential well, for a range of n's including high-energy excited states. Figure 8 clearly shows that OSCODE only needs to take a few steps once inside
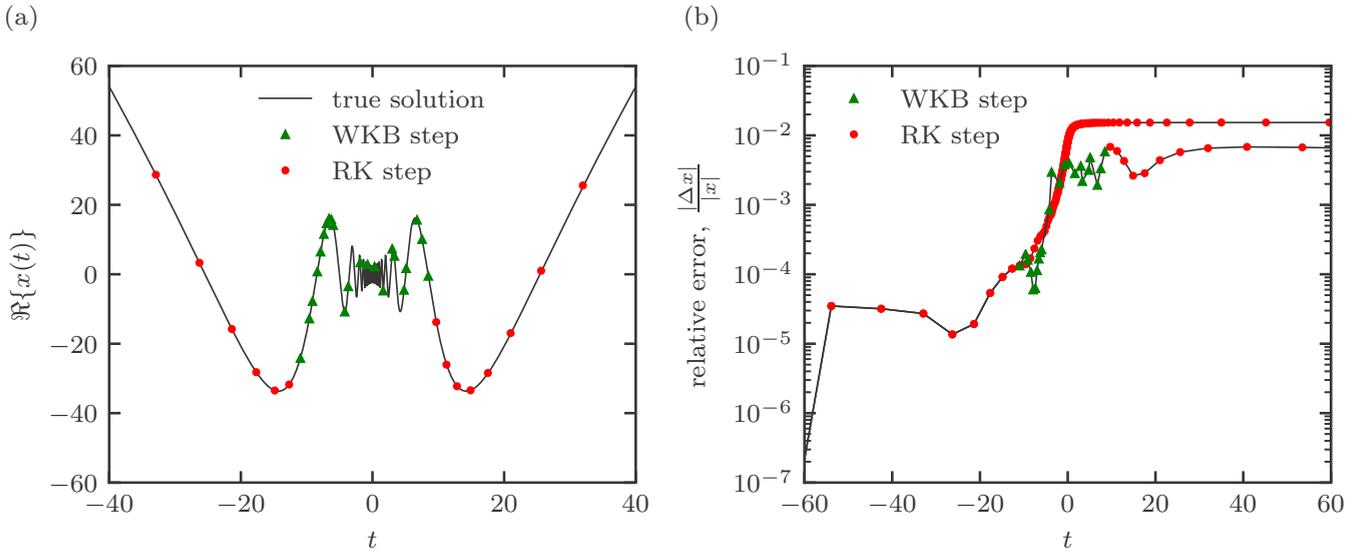
(a)

(b)

FIG. 2. Numerical solution of the burst equation with $n = 40$. A relative tolerance of 10^{-3} was set, and the equation was solved from $t = -2n$ to $t = 2n$.

the potential well, suggesting that computation time is greatly reduced relative to purely Runge-Kutta-based approaches. In this example the analytic solutions for the eigenfunctions were available and were used to set the values of Ψ and Ψ' at the integration boundaries.

In a general potential well, analytic solutions are not accessible and the energy eigenvalues are unknowns to be computed. Shooting methods [20] are frequently used to estimate the eigenvalues in such cases. We employ one such method to find the energies of the quantum harmonic oscillator with quartic anharmonicity, which has the potential

$$V(x) = x^2 + \lambda x^4. \tag{27}$$

An initial guess for the eigenvalue, E, is made. We start integration from points $\pm x_0$ outside the potential on either side of $x = 0$, where $E \ll V(x)$, using the initial conditions

$\Psi(\pm x_0) = 0$ and $\Psi'(\pm x_0) = 1$. We integrate toward the inside of the potential well in order to avoid contamination of the exponentially decaying solution by the growing mode when one integrates away from the well. The first initial condition is a good approximation far outside the potential well, and Ψ' can be chosen arbitrarily as it accounts for a choice of normalization. The two numerical solutions, Ψ_L and Ψ_R, meet at an intermediate point x_1. At x_1, both Ψ and Ψ' must be continuous if E is an eigenvalue. Therefore the normalization-independent quantity

$$\frac{\Psi_L'}{\Psi_L} - \frac{\Psi_R'}{\Psi_R} \tag{28}$$

is minimized as a function of E. A few examples of the eigenvalues thus computed are presented in Table II, alongside their matching values from [21]. Note that in order to get
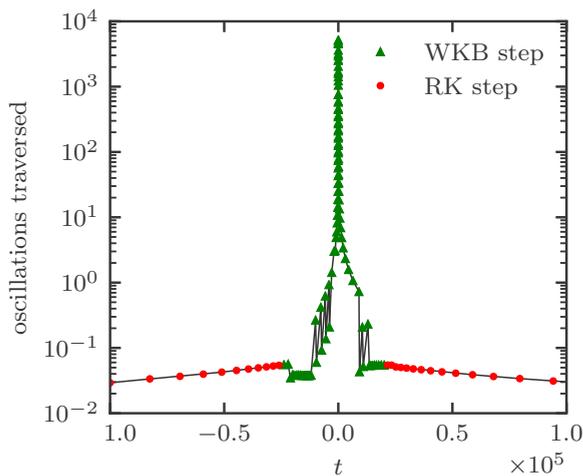

FIG. 3. Number of oscillations stepped over in a single step, while solving the burst equation with $n = 10^5$, and a relative tolerance of 10^{-4}.
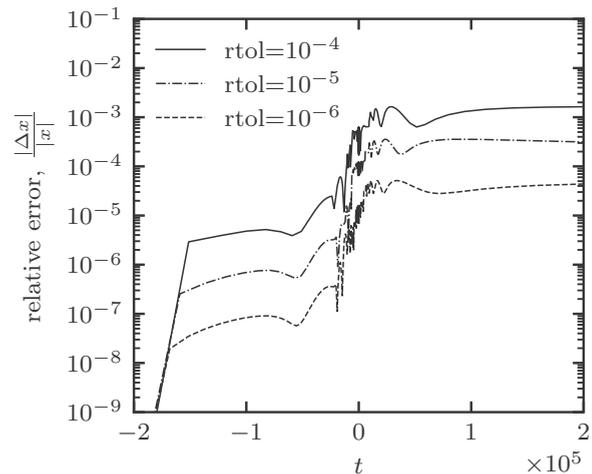

FIG. 4. Progression of the relative error in the burst equation with $n = 10^5$, with different settings of the local relative tolerance "rtol".
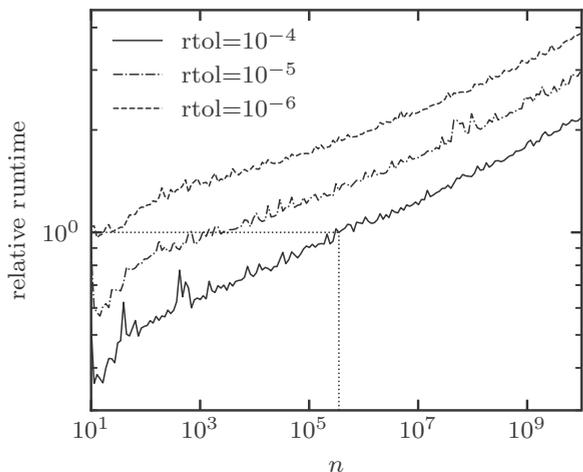
FIG. 5. Relative runtime of solving the burst equation from $t = -2n$ to $t = 2n$, with n varying from 10^1 to 10^{10}, and the relative tolerance "rtol" from 10^{-4} to 10^{-6}. The runtimes are referenced to the median of the n range and relative tolerance of 10^{-4}, as indicated by the dotted lines.

equivalent eigenvalues, we set $m = 0.5$. The eigenvalues are in good agreement up to highly excited states.

D. Mukhanov-Sasaki equation

In the previous two toy examples, there was only a frequency ω term present in the differential equation to be solved, and it was available to arbitrary precision. This may not always be the case, as (1) one may want to switch to a more physically meaningful independent-dependent variable pair, which can introduce a friction term γ, and (2) the frequency and friction terms might themselves be available only through numerically solving a set of differential equations. The Mukhanov-Sasaki equation illustrates both of these cases. In the brief introduction to the background of the equation to
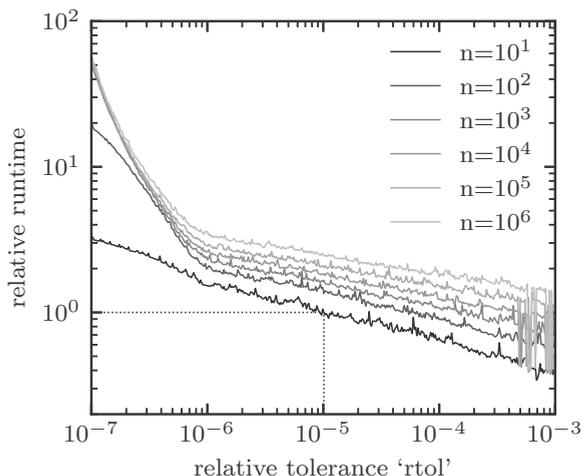

FIG. 6. Relative runtime of solving the burst equation, as a function of the relative tolerance set (with the absolute tolerance set to 0). The different curves show different values of n, proportional to the total number of oscillations traversed.
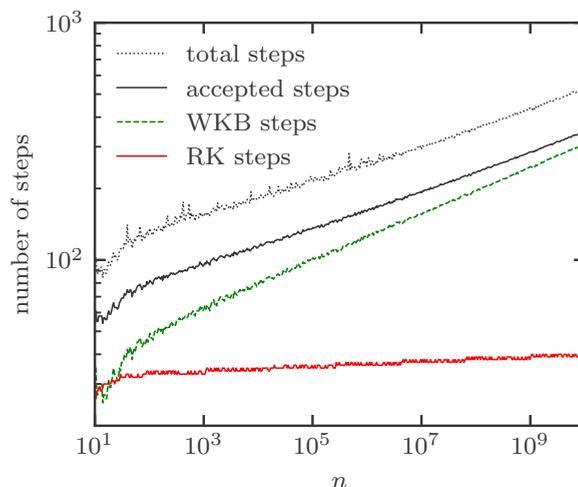

FIG. 7. Step breakdown in solving the burst equation from $t = -2n$ to $t = 2n$, with n varying from 10^1 to 10^{10}, and the relative tolerance "rtol" set to 10^{-4}.

follow, we use Planck units

$$c = \hbar = k_B = G = 1,$$

and set the Planck mass to 1, $m_P = 1$.

The Mukhanov-Sasaki equation describes the time-evolution of perturbations in a homogeneous, isotropic, "background" universe. This background, in the simplest models, assumes the presence of a single time-dependent scalar field $\phi(t)$ (the inflaton field). The field has self-interactions described by the potential $V(\phi)$, and its dynamics are defined by the action

$$\mathcal{S} = \int d^4x \sqrt{-g} \left[\frac{1}{2}R + \frac{1}{2}g^{\mu\nu}\partial_\mu\phi\partial_\nu\phi - V(\phi) \right]. \quad (29)$$

Assuming a metric of the Friedmann-Robertson-Walker form, the above action leads to the equations of motion

$$H^2 + \frac{K}{a^2} = \frac{1}{3}\left[\frac{1}{2}\dot{\phi}^2 + V(\phi) \right], \quad (30)$$

$$\dot{H} + H^2 = -\frac{1}{3}\left[\dot{\phi}^2 - V(\phi) \right], \quad (31)$$

$$0 = \ddot{\phi} + 3H\dot{\phi} + V_{,\phi}, \quad (32)$$

out of which only two are independent. In the above, $a(t)$ is the scale factor, $H(t)$ is the Hubble parameter defined as $H = \frac{\dot{a}}{a}$, and K is the curvature, taking values 0, ± 1 for flat, closed, and open universes. In this section we consider flat and closed universe models, starting with the flat case. In what follows, $K = 0$ until stated otherwise. Perturbing the field and the metric, and introducing the gauge-invariant scalar $\mathcal{R}$ (called the comoving curvature perturbation), one can then arrive at the Mukhanov-Sasaki equation, which we write as

$$\ddot{\mathcal{R}}_k + 2\left(\frac{\ddot{\phi}}{\dot{\phi}} - \frac{1}{2}\dot{\phi}^2 + \frac{3}{2} \right)\dot{\mathcal{R}}_k + \left(\frac{k}{aH} \right)^2 \mathcal{R}_k = 0. \quad (33)$$

In the above equation, the overdot denotes differentiation with respect to $N = \ln a$, and $\mathcal{R}_k$ is the mode with wave number k in the Fourier decomposition of $\mathcal{R}$. N measures the
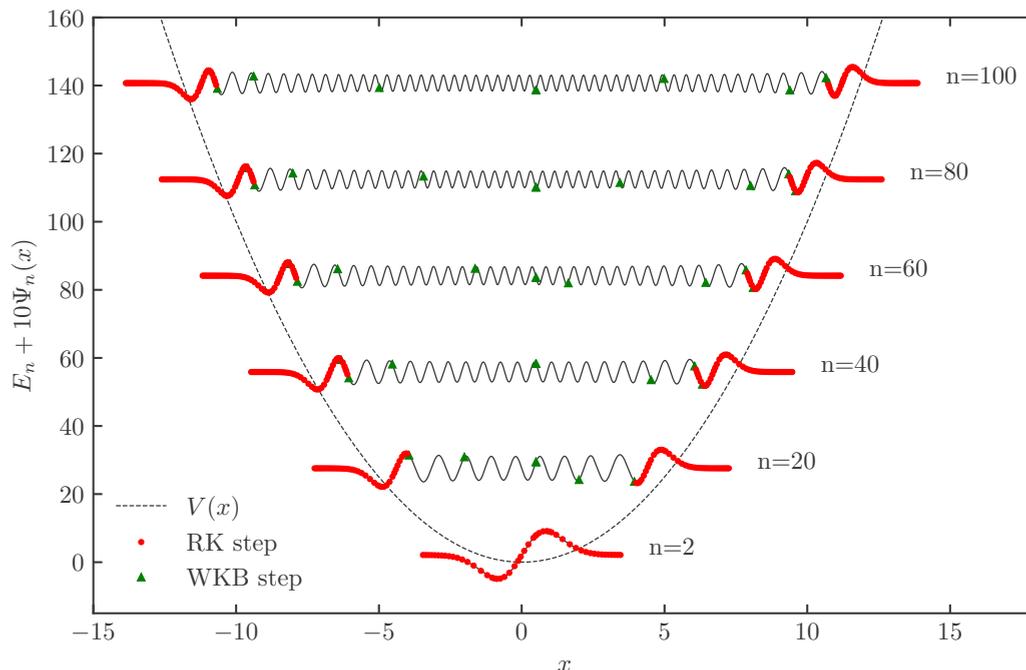
FIG. 8. Energy eigenfunctions in a harmonic potential well. In units of $m = \hbar = 1$ and with a potential $V(x) = x^2$, the nth level has energy $\sqrt{2}(n - 1/2)$. The wave functions in this potential are given analytically in terms of the Hermite polynomials, and are plotted in black. Numerical integration was started from both sides of $x = 0$, from well outside the potential [where $E \ll V(x)$], until $x = 0.5$. The initial conditions were set using the analytic solution for Ψ and Ψ'. The relative tolerance was set to be 10^{-3}.

amount of expansion the universe goes through. Even in this simple model, the single scalar field ϕ is enough to trigger an accelerated expansion of the universe, inflation ([5,22]). During inflation, $a(t) \sim e^{Ht}$ and H is approximately constant, hence N is a natural independent-variable candidate. Another important characteristic of inflation is that the quantity $\frac{1}{aH}$, called the comoving Hubble horizon, shrinks. The Hubble horizon plays a crucial role in governing the dynamics of perturbations, which will be described later.

TABLE II. Energy eigenvalues of the quantum harmonic oscillator with quartic anharmonicity. The E_n column shows the eigenvalues found with our method, to be compared with the E_n^* column, which lists the (rounded) results of [21].

n	E_n	E_n^*
0	1.392353	1.392352
1	4.648815	4.648813
2	8.6550501	8.6550500
3	13.156806	13.156804
4	18.0577	18.0576
15	88.6104	88.6103
16	96.1291	96.1296
17	103.793	103.795
18	111.6025	111.6020
19	119.5440	119.5442
50	417.05620	417.05626
100	1035.5440	1035.5442
1000	21932.7848	21932.7840
10000	471103.81	471103.80

In the limit of *slow-roll* inflation, defined by $\frac{1}{2}\dot{\phi}^2 \ll V(\phi)$, the background equations (30)–(32) admit analytic solutions. They also do in the opposite limit, $\frac{1}{2}\dot{\phi}^2 \gg V(\phi)$, called *kinetic dominance* (see [23]). Kinetic dominance has been shown to be the limit the universe emerges from in most single-field models [24]. In kinetic dominance the comoving Hubble horizon grows, then shrinks again as slow-roll inflation is entered. Both limits can thus be used to set initial conditions to Eqs. (30)–(32), which can then be integrated numerically.

The Mukhanov-Sasaki equation in the flat case can also be solved analytically if for all k modes of interest, $k \gg aH$. Since k^{-1} is the characteristic length scale of a perturbation mode, this means that all modes of interest are assumed to be well inside the Hubble horizon. Letting the Mukhanov-Sasaki equation emerge from this limit is equivalent to choosing a vacuum state (see [25]) which, together with a normalization condition, are enough to provide initial conditions for the mode functions $\mathcal{R}_k$. This choice of vacuum and the initial conditions are referred to as *Bunch-Davies*. With a different choice of vacuum, it is possible to set initial conditions on the $\mathcal{R}_k$ in kinetic dominance, when modes are not necessarily inside the Hubble horizon. The form of $\mathcal{R}_k$ in the kinetically dominated limit are derived in [26]. In the models investigated, we shall consider both slow-roll and kinetically dominated initial conditions for the background and the perturbations.

The Mukhanov-Sasaki equation (33) is of the form of a generalized oscillator with a first-derivative γ term present, with both γ and the frequency ω being (in general, nonanalytic) functions of time as they depend on the cosmological background. It follows that when a k mode

is inside the Hubble horizon, $k > aH$, it oscillates with some varying amplitude and frequency usually proportional to k, and one can show that the mode "freezes out" once outside the Hubble horizon, meaning $\mathcal{R}_k \sim$ constant. The wave-number dependence of the frequency term makes this equation challenging to solve for large values of k without resorting to approximations.

Our goal is to solve the Mukhanov-Sasaki equation for a range of k modes until each mode has a constant amplitude, up to large values of k, in order to obtain the primordial power spectrum

$$\mathcal{P}_{\mathcal{R}}^2(k) = \frac{k^3}{2\pi^2}|\mathcal{R}_k|^2. \tag{34}$$

1. Comparison with BINGO

We shall first adopt the computational strategy employed by many solvers designed to compute primordial power spectra, for example BINGO [7] and MODECODE [27], and compare our solver performance with the former. BINGO is a Fortran-based code for efficient evaluation of the scalar bi-spectrum, that has to calculate the primordial power spectrum of scalar perturbations on the way, but we shall only use it to compute the primordial power spectrum.

BINGO gets around the computational challenges by using a trick: it has been shown that in the case of a single-field inflationary model and assuming the universe emerges from slow-roll inflation, it is sufficient to evolve each curvature perturbation from a time they are well inside the Hubble horizon (from, say, $k/aH = 100$; see [8]), until the perturbation freezes out outside of the Hubble horizon ($k/aH = 10^{-2}$). This avoids integrating the solution through the majority of its oscillatory phase. For the comparison to be fair, we will do the same. First, the cosmological background ($\phi(N), \dot{\phi}(N), \dots$) is computed numerically as a function of N, starting from the slow-roll conditions, set such that the total number of e-folds of inflation $N_{\text{tot}} \sim 60$. The inflationary model used in this example involves a quadratic potential,

$$V(\phi) = \tfrac{1}{2}m^2\phi^2, \tag{35}$$

where we set the inflaton mass to 1, $m = 1$. The initial scale factor is set such that a pivot mode, corresponding to $k = 0.05$ Mpc^{-1}, leaves the Hubble horizon when there are 50 e-folds of inflation left. For each mode, we find the N corresponding to the start and end of integration, then for our solver, we supply the algorithm with $\omega(N)$ and $\gamma(N)$ defined as grids, on which we perform linear interpolation (the grid needs to be sufficiently fine; for the present example we used 5×10^5 equally spaced points between $N = 0$ and $N = 75$). We then solve the mode evolution for each k starting from Bunch-Davies initial conditions. We set the same parameters to BINGO and our solver; in particular we set a relative tolerance of 10^{-4} and an absolute tolerance of 0. The resulting power spectra are identical, as shown in Fig. 9.

The computation time for the solver to obtain $\mathcal{R}_k$ is measured and plotted as a function of k in Figs. 10 and 11. The former shows the ratio of BINGO and our solver's runtimes as a function of k, and the latter just that of our solver, relative to the median k. Together they show that BINGO's runtime is logarithmic in k, whereas our solver's is constant. They also
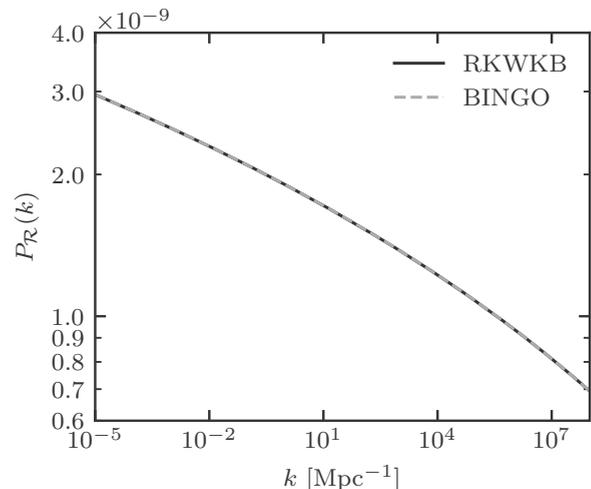

FIG. 9. Primordial power spectra of the gauge-invariant scalar curvature perturbations, generated by BINGO and our solver with identical parameters. The spectra have been computed up to extremely large values of k for the sake of comparing the runtimes of the two codes.

show that OSCODE performs better than BINGO by at least a factor of 2, and at most a factor of 4 in the k range of interest.

This can be explained by looking at (33). The frequency term is a fixed number at the start and end of integration, so it will no longer scale with k. The friction term during inflation is approximately constant. The range of integration, ΔN, is determined by the points where $k/aH = c_0$ (a constant), which during inflation is also roughly constant. Therefore the number of oscillations over the range of wave numbers in the spectrum barely changes, and we expect a WKB-based method to traverse the oscillations in constant time. In reality, the integration range increases slowly with N, and small variations in the friction term cause the oscillations to change in shape, hence the slow increase in the runtime of BINGO. The 2-fold runtime difference present even at the
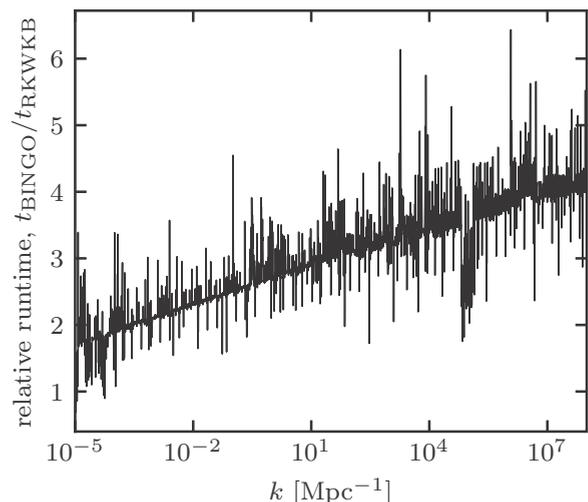

FIG. 10. Ratio of the runtime of BINGO and our solver during the evaluation of a scalar primordial power spectrum, as a function of wave vector.
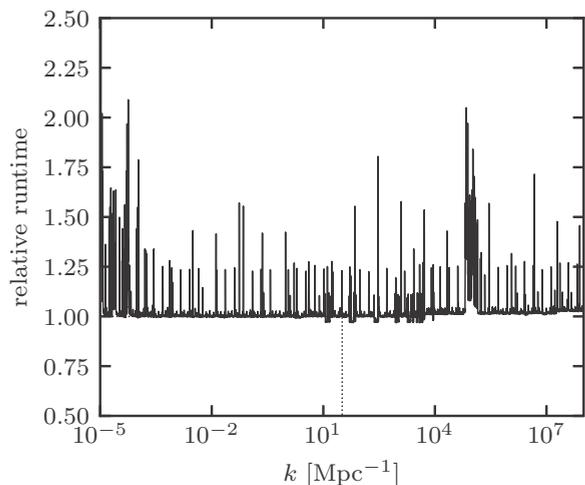
FIG. 11. Progression of our solver's relative runtime with increasing wave number, while calculating a scalar primordial power spectrum. The times are referenced to the computation time corresponding to the median k value, indicated by the dotted lines.

smallest values of k can be explained by the difference in the number of steps taken. Figure 12 shows the intermediate steps taken by RKSUITE, a numerical routine implementing efficient Runge-Kutta methods and used by BINGO, and the intermediate steps taken by OSCODE, while computing the time evolution of a single k mode. OSCODE is able to traverse the oscillatory region of the mode's evolution in significantly fewer steps than the Runge-Kutta method, giving a reduction in computing time.

In models where one has to start integrating the mode equation from deeper within the horizon, the starting fre-
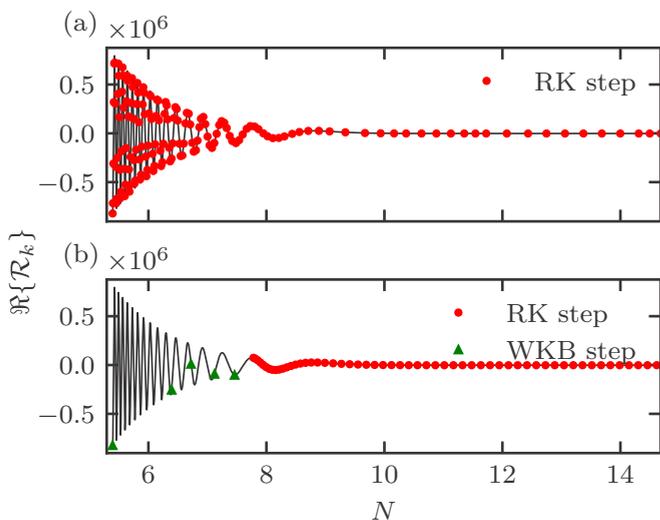

FIG. 12. Comparison of BINGO and our solver in the evolution of a single perturbation with wave vector $k = 10^{-5}$ Mpc^{-1}. The black reference line is a dense solution generated with a Runge-Kutta $(7, 8)$th-order pair. On top of it the top panel shows the steps that RKSUITE's $(4, 5)$th-order Runge-Kutta solver takes (a total of $\sim$150), the bottom panel the steps that our solver takes (a total of $\sim$60). The relative tolerance was set to 10^{-4} for both methods.

quency during the evolution of modes is larger, and the performance difference between an RKWKB-based approach and a Runge-Kutta integrator is even more distinct. Examples include universes emerging from kinetic dominance, axion monodromy models [28], or models with alpha vacua initial conditions [29].

2. A model using kinetic dominance

Inflationary models including kinetic dominance are already being investigated, e.g., by [30] and [31]. In this scenario, the cosmological background in terms of N is integrated from the initial state

$$\phi = \phi_P - \sqrt{6} \ln N, \tag{36}$$

$$\dot{\phi} = -\sqrt{6 - \frac{2V}{H^2}}, \tag{37}$$

$$H = \frac{1}{3} e^{-3N}, \tag{38}$$

where $\dot{\phi}$ contains a contribution from the potential in order to make the system numerically stable. In kinetic dominance, [26] obtains a solution for the perturbation modes, which in terms of N take the form

$$\mathcal{R}_k = \sqrt{\frac{3\pi}{8}} \frac{1}{z} e^N \left[A_k H_0^{(1)} \left(\frac{3}{2} k e^{2N} \right) + B_k H_0^{(2)} \left(\frac{3}{2} k e^{2N} \right) \right], \tag{39}$$

$$\dot{\mathcal{R}}_k = \sqrt{\frac{27\pi}{8}} \frac{k}{z} e^{3N} \left[A_k H_0^{(1)\prime} \left(\frac{3}{2} k e^{2N} \right) \right. $$
$$\left. + B_k H_0^{(2)\prime} \left(\frac{3}{2} k e^{2N} \right) \right] + \left(-\frac{\dot{z}}{z} + 1 \right) \mathcal{R}_k, \tag{40}$$

where $H^{(1)}$ and $H^{(2)}$ are Hankel functions of the first and second kind, and A_k, B_k are constants. We set $A_k = 0$ and $B_k = 1$, and chose parameters such that the total number of e-folds during inflation $N_{\text{tot}} \approx 60$, and the pivot scale corresponding to $k = 0.05$ Mpc^{-1} today leaves the horizon when there are $N_* \approx 54$ e-folds of inflation left. We set the initial conditions for the background at $N = 0$ and for the modes at a constant $N = 1.1$, and integrate until far after horizon exit, as in Sec. III D 1.

The resulting primordial power spectrum is shown in Fig. 13. Such computations are only possible if the solver used can trace oscillations in the solution extremely efficiently, and indeed we found that calculating a spectrum starting from kinetic dominance and from a fixed fraction of the horizon in slow roll can be carried out on similar timescales using our solver. It is worth noting that a fast solver has been developed specifically for the Mukhanov-Sasaki equation [32] that works on the basis of using analytic approximations for when the frequency is well approximated by an exponential or first-order polynomial. This gives a significant speedup over Runge-Kutta methods, but relies on the Mukhanov-Sasaki equation to be transformable to a form without a first-order derivative term. Closed-universe models do not have this property, but can still be investigated with our method.
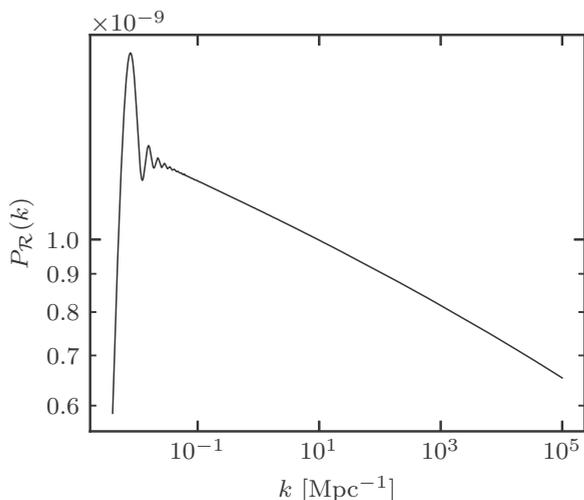
FIG. 13. Scalar primordial power spectrum of perturbations emerging from kinetic dominance. The mode equation was solved from a fixed, early time (well inside kinetic dominance) until long after horizon crossing, which is only feasible if the solver used is capable of traversing many oscillations at once. The relative tolerance was set to be 10^{-4}.

3. A closed universe model

In this example we investigate closed-universe models with curvature $K = 1$. The cosmological background evolution equations (30)–(32) can be cast into a system of linear ODEs,

$$\frac{d \ln |\Omega_k|}{dN} = 4 + |\Omega_k|[4K - 2a^2 V(\phi)], \qquad (41)$$

$$\left(\frac{d\phi}{dN}\right)^2 = 6 + |\Omega_k|[6K - 2a^2 V(\phi)], \qquad (42)$$

where $\Omega_k = \frac{K}{(aH)^2}$. We shall consider a cosmological background emerging from kinetic dominance, such that the Hubble horizon, $(aH)^{-1} = \sqrt{\Omega_k}$, grows until it reaches a maximum $\sqrt{\Omega_k^i}$ at e-folds N_i. From this point the horizon shrinks, and inflation starts. The parameters (Ω_k^i, N_i), together with the requirement $\dot{\Omega}_k(N_i) = 0$, fully fix the background evolution, and hence determine the amount of inflation, N_{tot}. We used Brent's method of root finding [33] to search for the N_i for a given Ω_k^i that yields $N_{\mathrm{tot}} = 60$. Hence the primordial power spectra have all other parameters fixed, with only Ω_k^i, the initial curvature at the start of inflation, changing. Integration of the background is started from N_i and is performed forward until the end of inflation (and backward, if necessary) to cover the integration range of the perturbation modes.

The mode functions obey the generalized Mukhanov-Sasaki equation in the presence of nonzero curvature K, with frequency and first-derivative terms given by [34]

$$\omega^2 = \Omega_k \left[(k_2 - K) - \frac{2Kk_2}{EK + k_2} \frac{\dot{E}}{E} \right], \qquad (43)$$

$$2\gamma = K\Omega_k + 3 - E + \frac{k_2}{EK + k_2} \frac{\dot{E}}{E}, \qquad (44)$$

where $E = \frac{1}{2}\dot{\phi}^2$ and

$$k_2 = \begin{cases} k(k+2) - 3K, & \text{if } K > 0, \\ k^2 - 3K, & \text{otherwise.} \end{cases} \qquad (45)$$

The modes are started from $N = N_i$ using the Bunch-Davies conditions introduced at the start of Sec. III. Although the Bunch-Davies solution has been derived from the Mukhanov-Sasaki equation in a flat universe, its closed-universe equivalent is not yet known. An important feature of closed-universe primordial power spectra is that the values of the *comoving wave number k*, appearing in the above equations, are quantized to only take integer values, with the lowest possible value of $k = 3$ [35]. We relate the comoving wave number, measured in Planck units, to the physical scale of the perturbation today via

$$k_{\mathrm{today}} = \frac{k}{a_0}, \qquad (46)$$

where a_0 is the present-day scale factor, given in terms of the present-day reduced Hubble parameter, $h \equiv H_0/(100 \text{ km s}^{-1} \text{ Mpc}^{-1})$, and the present-day density in curvature, $\Omega_{k,0}$, by

$$a_0 \approx 4.3 \times 10^4 \left(\frac{h}{0.7}\right)^{-1} \left|\frac{\Omega_{k,0}}{0.01}\right|^{-\frac{1}{2}} \text{ Mpc.} \qquad (47)$$

Figure 14 shows the resulting primordial power spectra for various values of initial curvature Ω_k^i, each with an associated spectrum treating comoving k as a continuous variable plotted underneath. Calculating the spectra with the RKWKB method provided roughly three orders of magnitude reduction in computing time compared to Runge-Kutta-like methods.

IV. LIMITATIONS

When applying our solver to a problem, it is worth considering whether the solver's performance would be limited by strict accuracy requirements or by a nonideal choice of independent or dependent variable.

As shown in Fig. 6, the algorithm's runtime scales up gently in the relative tolerance range $[10^{-4}, 10^{-6}]$. The user is therefore advised to use the solver if such accuracies are acceptable for the problem in question. If the problem requires rtol $< 10^{-6}$, one would need a higher-order Runge-Kutta pair as an alternative solver to WKB, such as a (7, 8) pair used by the NAG Library.

As mentioned in Sec. III A, the solver will not be able to fulfil the accuracy requirements if at any time the integral(s) $[S_i]_t^{t+h}$ exceed $\sim 10^{12}$. Care needs to be taken especially with the first term in the WKB series, $\pm i \int_t^{t+h} \omega dt$, as this is expected to be the largest in a region where the WKB approximation is appropriate. The reason underlying this limit is that the solver needs to compute the exponential of this large imaginary term, which requires large accuracy modulo 2π. Storing such large numbers accurately is limited by machine (double) precision, and the solver might start accumulating error. In Sec. III A the step size and frequency become so large at $t > 10^8$ that this limit is reached.

Finally, the solver is only efficient if in some region the frequency is slowly varying; therefore care needs to be taken
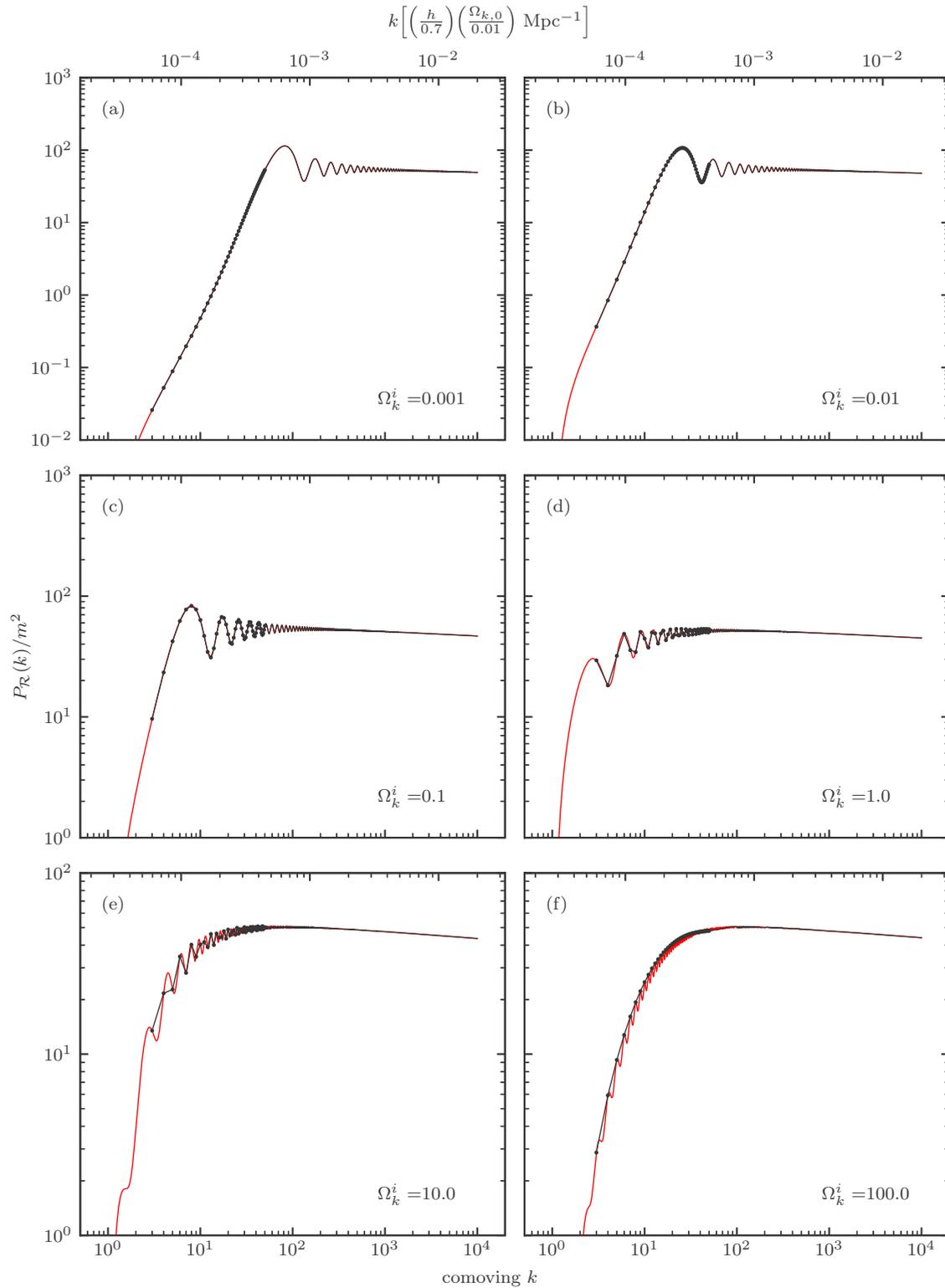
FIG. 14. Scalar primordial power spectra in universes with varying initial curvature. The start of inflation, N_i, is adjusted to vary with the curvature at the start of inflation, Ω_k^i, such that the total e-folds of inflation, $N_{\text{tot}} = 60$, is constant. In curved universes, only integer values of comoving k are allowed, with $k \geqslant 3$ (continuous line with $k \leqslant 50$ highlighted), but for clarity we include the continuous spectrum (dashed line). The modes are started from the Bunch-Davies vacuum.

to choose an appropriate independent-dependent variable pair if the problem allows. In our cosmological examples $\mathcal{R}_k$ was chosen for its freezing-out property which makes the computation easy outside of the horizon, and N instead of cosmic time t because it does not span several orders of magnitude during the integration and gives a remarkably simple $\ln \omega$. The frequency and friction term in terms of N are smooth and slowly varying, which allowed for them to

be well approximated by linear interpolation on a sufficiently fine, evenly spaced grid.

The most immediate future generalization of the algorithm would involve extending it to several dimensions, so that one could solve a coupled set of oscillatory differential equations. However, exploratory investigation revealed this task to be more difficult than anticipated [36].

A reduction in runtime and simplification of the stepping procedure would be possible if at small step sizes, the step proposed by the algorithm using the WKB approximation reduced to a Runge-Kutta step of similar order. At present, the small-step-size limit of the WKB steps is Euler's method. Euler's method not being efficient enough for practical use makes it necessary to take an alternative higher-order Runge-Kutta step, which adds computational overhead.

It is worth noting that due to the oscillatory nature of equations, OSCODE only obtains the solution at the start and end of integration (t_{start}, t_{end}), and at a set of intermediate points determined by the solver. If the solution is required at a set of specific points, then it can be acquired by running the solver multiple times with t_{start} and t_{end} coinciding with the desired set. We are considering how the solution could be obtained at any given point within $[t_{\text{start}}, t_{\text{end}}]$ from the solutions (2) and (3), and plan to include this feature in a future release.

V. CONCLUSIONS

We have presented a numerical solver for second-order ordinary differential equations that can be written in the form of a one-dimensional oscillator, with a time-varying frequency and friction term that do not necessarily have a closed form. We have shown that the solver is significantly more efficient than other known methods if the frequency varies slowly over some part of the integration range, even if it is extremely large, because the solver can exploit the WKB approximation in these cases to traverse many oscillations at once. We have also shown that the solver can detect regions where the WKB approximation is not valid and can dynamically switch to a Runge-Kutta integrator.

We demonstrated the above properties on several examples, the Airy equation, a more complex "burst" equation, the Schrödinger equation, for which the frequency term can be written as a function of time, and the Mukhanov-Sasaki equation, where both the frequency and the friction term need to be computed numerically in advance. In the case of the Mukhanov-Sasaki equation, we compared the solver's performance to that of BINGO, a highly efficient Fortran code that computes the scalar bi-spectrum by first computing a primordial power spectrum of scalar curvature perturbations using a fast Runge-Kutta solver available from RKSUITE. We measured for each wave number k how long each code takes to compute a solution to the Mukhanov-Sasaki equation from subhorizon ($k/aH = 100$) to superhorizon ($k/aH = 0.01$) times with all parameters identical, and found that our solver takes constant time in k, being approximately twice as fast as BINGO in the observational range. If integration started when modes were deeper inside the Hubble horizon, the performance difference increases dramatically. To prove this, we demonstrated that our solver is capable of integrating each

mode from a single fixed time through horizon entry and exit, starting from kinetically dominated initial conditions for both the smooth, isotropic universe and the perturbations. We further computed primordial power spectra for closed universes with varying initial curvature, a family of models in which the oscillatory equation of motion cannot be transformed into a first-derivative-free form, making it impossible to be solved with the efficient non-Runge-Kutta method developed in [32].

The numerical solver, OSCODE, is available on GitHub [37] and can be accessed via its C++ or Python interface.

APPENDIX A: ESTIMATING THE ERROR IN RK AND WKB STEPS

The RK and WKB steps each give x and $\dot{x}$ (referred to by their subscripts), and the difference between the 4th and 5th order RK steps gives an error on them. Estimating the error on a WKB step is less straightforward, and we decided to use the larger of two error estimates which dominate in different limits, as discussed below.

The obvious equivalent error estimate on WKB steps, Δx_{WKB} and $\Delta \dot{x}_{\text{WKB}}$, is the difference between an Nth and $(N-1)$th order estimate, where N refers to the highest-order S term in (11) included in the WKB expansion. This estimate is a good proxy for the validity of the WKB approximation because it can signal the breakdown of the relations (12), but in a region where they hold, it is expected that the numerical error in the $S_i(t)$ will dominate Δx and $\Delta \dot{x}$. We therefore estimate the error on the WKB step arising from the imperfect numerical integration of $\dot{S}_i(t)$ as

$$\Delta x_{\text{WKB}} = A_+ \Delta f_+ + A_- \Delta f_-, \tag{A1}$$

$$\Delta f_\pm = f_\pm \sum_{i=0}^{n} \Delta [S_i]_t^{t+h}, \tag{A2}$$

and

$$\Delta \dot{x}_{\text{WKB}} = B_+ \Delta \dot{f}_+ + B_- \Delta \dot{f}_-, \tag{A3}$$

$$\Delta \dot{f}_\pm = \Delta f_\pm \frac{\dot{f}_\pm}{f_\pm}. \tag{A4}$$

Note that in the above, f and its derivatives are evaluated at $t + h$ according to (4) and (5), and that it is assumed that the numerical integrations of $\dot{S}_i$ are the *only* sources of error, i.e., that the $\dot{S}_i$ can be acquired perfectly.

APPENDIX B: RUNGE-KUTTA METHODS WITH GAUSS-LOBATTO STENCILS

In this section we present the Butcher tableau of the two Runge-Kutta methods used in the solver. Table III contains the coefficients of the 4-stage, 4th-order method, and Table IV contains those of the 6-stage, 5th-order one.

TABLE III. Butcher tableau for the 4-stage, 4th-order Runge-Kutta method used in the solver, based on 4 out of 5 stencil points of a Gauss-Lobatto quadrature with $n = 5$ stencil points.

0				
$\frac{1}{2}(1 - \sqrt{\frac{3}{7}})$	$\frac{1}{2}(1 - \sqrt{\frac{3}{7}})$			
$\frac{1}{2}(1 + \sqrt{\frac{3}{7}})$	$-\frac{1}{4}(3 + 5\sqrt{\frac{3}{7}})$	$\frac{1}{4}(5 + \sqrt{21})$		
1	$-\frac{1}{4}(3 + 7\sqrt{21})$	$-\frac{1}{4}(21 + 5\sqrt{21})$	$\frac{1}{14}(-1 + \sqrt{\frac{3}{7}})$	
	$-\frac{1}{12}$	$\frac{7}{12}$	$-\frac{1}{12}$	$-\frac{1}{12}$

APPENDIX C: STEPPING PROCEDURE

Let us summarize the different error estimates:

(i) Δx_{RK}, $\Delta \dot{x}_{RK}$: error on RK step;

(ii) Δx_{WKB}, $\Delta \dot{x}_{WKB}$: error on WKB step from computing $[S_i]_t^{t+h}$ numerically;

(iii) Δx^t_{WKB}, $\Delta \dot{x}^t_{WKB}$: error on WKB step from truncation of WKB series.

In order for the solver to switch successfully to the most suitable method dynamically, and adapt the step size to stay within the error bound required, it has to determine two things:

(1) Which step (RK or WKB) to choose that yields the largest possible next step size within acceptable tolerance?

(2) What should the size of the next step be?

TABLE IV. Butcher tableau for the 6-stage, 5th-order Runge-Kutta method used by the solver. The time points of evaluation are the 6 stencil points used for Gauss-Lobatto quadrature with $n = 6$.

c_1	0
c_2	$\frac{1}{2}(1 - \sqrt{\frac{1}{3} + \frac{2\sqrt{7}}{21}})$
c_3	$\frac{1}{2}(1 - \sqrt{\frac{1}{3} - \frac{2\sqrt{7}}{21}})$
c_4	$\frac{1}{2}(1 + \sqrt{\frac{1}{3} - \frac{2\sqrt{7}}{21}})$
c_5	$\frac{1}{2}(1 + \sqrt{\frac{1}{3} + \frac{2\sqrt{7}}{21}})$
c_6	1
a_{21}	0.117472338035267
a_{31}	−0.186247980065150
a_{32}	0.543632221824827
a_{41}	−0.606430388550828
a_{42}	1
a_{43}	0.249046146791150
a_{51}	2.89935654001573
a_{52}	−4.36852561156624
a_{53}	2.13380671478631
a_{54}	0.217890018728924
a_{61}	18.6799634999572
a_{62}	−28.8505778397313
a_{63}	10.7205340842092
a_{64}	1.41474175650804
a_{65}	−0.964661500943270
b_1	0.112755722735172
b_2	0
b_3	0.506557973265535
b_4	0.0483004037699511
b_5	0.378474956297846
b_6	−0.0460890560685063

The answer to question 1 requires forecasting the error progression of both methods with the step size, i.e., requires knowledge of $\Delta x(h)$ and $\Delta \dot{x}(h)$. For the RK step this behavior is known to be a power law, and for WKB steps we shall assume two separate power laws with different exponents n_{WKB} and n'_{WKB}, for when the dominant error on WKB steps arises from the numerical integrals and the truncation of the asymptotic series, respectively. First, the dominant error on each type of step is determined,

$$\Delta_{RK} = \max(\epsilon, \Delta x_{RK}, \Delta \dot{x}_{RK}), \quad (C1)$$

$$\Delta_{WKB} = \max(\epsilon, \Delta x_{WKB}, \Delta \dot{x}_{WKB}, \Delta x^t_{WKB}, \Delta \dot{x}^t_{WKB}), \quad (C2)$$

where ϵ is a small number close to machine precision, for safety. The type of dominant error on the WKB step, *truncation* or *integral*, is recorded. Starting from a current step size h, the largest possible steps within the error bound "tol" are then

$$h_{RK} = h \times \left(\frac{\text{tol}}{\Delta_{RK}} \right)^{1/n_{RK}},$$

$$h_{WKB} = h \begin{cases} \left(\frac{\text{tol}}{\Delta_{WKB}} \right)^{1/n'_{WKB}}, & \text{if } \textit{truncation}, \\ \left(\frac{\text{tol}}{\Delta_{WKB}} \right)^{1/n_{WKB}}, & \text{otherwise}. \end{cases} \quad (C3)$$

The step with the larger step size will then be chosen as a trial step, but is not yet accepted. The next step size is then predicted. If the chosen method is RK, this next step size is simply

$$h_{next} = h_{RK}. \quad (C4)$$

If the chosen method is WKB however (i.e., the truncated WKB series was deemed sufficient to approximate the solution), the error arising from truncation of the WKB series will be ignored:

redefine Δ_{WKB} as

$$\Delta_{WKB} = \max(\epsilon, \Delta x_{WKB}, \Delta \dot{x}_{WKB}), \quad (C5)$$

$$\text{then} \quad h_{next} = h \left(\frac{\text{tol}}{\Delta_{WKB}} \right)^{1/n_{WKB}}. \quad (C6)$$

This process is illustrated in Fig. 15. Finally, if $h_{next} > h$, the current error did not exceed the tolerance limit and the step is accepted. Otherwise the step is rejected and one must ensure that the step is reattempted with sufficiently small h. The new step size in both cases is calculated via

$$h_{next} = h \times \begin{cases} \left(\frac{\text{tol}}{\Delta_{RK}} \right)^{1/(n_{RK}-1)}, \\ \left(\frac{\text{tol}}{\Delta_{WKB}} \right)^{1/(n_{WKB}-1)}, & \text{if } \textit{integral}, \\ \left(\frac{\text{tol}}{\Delta_{WKB}} \right)^{1/(n'_{WKB}-1)}, & \text{if } \textit{truncation}. \end{cases} \quad (C7)$$

This ensures h is decreased after rejected steps and increased following accepted ones.

The slopes of the errors as functions of h in Fig. 15 were not chosen at random. The error in a 6-stage, 5th-order RK method goes as h^5 for $h < 1$ and h^6 for $h > 1$. The error arising from truncation of the WKB series, upon entering a region well approximated by the WKB expansion, is expected to be proportional to h. This can be understood starting from
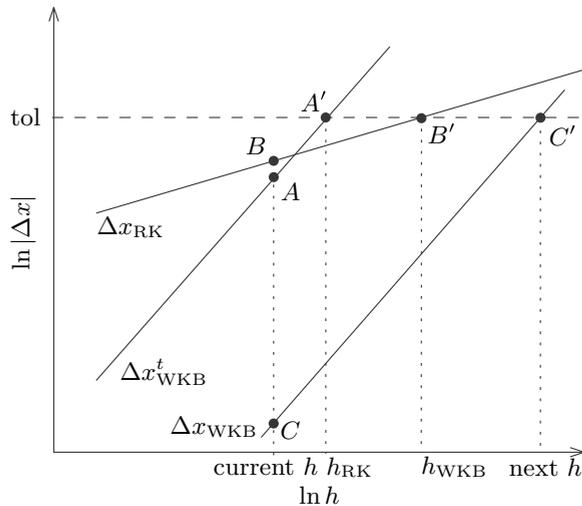
FIG. 15. Schematic plot of the assumed error progression in RK and WKB steps with increasing step size h. After the steps have been calculated from t to $t + h_{\text{current}}$, the errors of each method are shown by points A, B, and C, the latter two arising from the truncation of the WKB asymptotic series and the numerical integrals present in the series, respectively. The dominant type of error on the WKB step in this case is the *truncation*. Assuming power-law behavior in the errors for both steps with different exponents for each type of error, the next largest step size within the required tolerance "tol" would be h_{RK} and h_{WKB}, marked by points A' and B'. Since $h_{\text{WKB}} > h_{\text{RK}}$, the algorithm in this case chooses the WKB step. The size of the next step is therefore determined solely on the basis of the *integral* error, marked by C, and is going to be the projection of C'. Since this next step size is larger than the previous, the step is accepted.

the relative error on x based on [1],

$$\frac{\Delta x}{x} \sim S_{N+1}, \tag{C8}$$

for an Nth-order WKB estimate. For $N \geqslant 1$, S_{N+1} is a numerical integral of a small and nearly constant quantity, and is therefore $\propto \dot{S}_{N+1} h$. The errors on WKB steps arising from the integrals $[S_i]_t^{t+h}$ are on the other hand expected to go roughly as the errors on the integrals themselves [see Eq. (A1)].

Although more difficult to predict, this is expected to be dominated by the imperfect evaluations of the integrands, which contain numerical derivatives. The largest of these are the first derivatives, which will have an error $\propto h^{n_s-1}$. Since the algorithm uses the $n = 6$ Gauss-Lobatto evaluations to calculate all derivatives, we set $n_s = 6$.

By the above reasoning, the algorithm by default has

$$n_{\text{RK}} = 5, \quad n_{\text{WKB}} = 5, \quad n_{\text{WKB}}^t = 2, \tag{C9}$$

but the user can set these parameters to better fit the problem in question. For example, for optimal step acceptance/rejection ratio, for all burst examples in Sec. III B we set $n_{\text{WKB}} = 8$, $n_{\text{WKB}}^t = 1$.

APPENDIX D: SUMMARY

The algorithm goes through the following steps:

(1) Stepping from t to $t + h$, evaluate ω and γ at the Gauss-Lobatto stencil points for $n = 6$ and $n = 5$, a total of 9 different points.

(2) Use the Butcher tableaux of Tables IV and III to construct an RK step in x and $\dot{x}$, and use the difference as the error Δx_{RK}.

(3) Use finite-difference methods to evaluate all necessary derivatives of ω and γ needed for the derivatives of terms $(\dot{S}_i)$ in the WKB series (11).

(4) Use Gauss-Lobatto quadrature with $n = 6$ and $n = 5$ to evaluate the terms in the WKB series and their errors, taken as the difference.

(5) Construct an Nth and $(N - 1)$th order WKB step in x and $\dot{x}$.

(6) Compute the *truncation* error on WKB steps as the difference between the Nth and $(N - 1)$th order estimates, and the *integral* from (A1) and (A3).

(7) Find the dominant error and its type based on (C1), and choose between RK/WKB methods based on (C3).

(8) Predict the next step size, h_{next}, based on (C4)–(C6).

(9) **If** $h_{\text{next}} > h$, accept the step, and update x, $\dot{x}$, and t.

(10) **Otherwise**, reject the step and calculate h_{next} according to (C7).

(11) Update h to h_{next}.

(12) Repeat steps 1–11.

[1] C. M. Bender and S. A. Orszag, *Advanced Mathematical Methods for Scientists and Engineers. I. Asymptotic Methods and Perturbation Theory* (Springer, New York, 1999).

[2] V. F. Mukhanov, H. A. Feldman, and R. H. Brandenberger, Theory of cosmological perturbations, Phys. Rep. **215**, 203 (1992).

[3] D. Baumann, TASI lectures on inflation, arXiv:0907.5424.

[4] S. Dodelson, *Modern Cosmology* (Academic Press, San Diego, 2003).

[5] A. R. Liddle and D. H. Lyth, *Cosmological Inflation and Large-Scale Structure* (Cambridge University Press, Cambridge, 2000).

[6] H. V. Peiris, E. Komatsu, L. Verde, D. N. Spergel, C. L. Bennett, M. Halpern, G. Hinshaw, N. Jarosik, A. Kogut, M. Limon, S. S. Meyer, L. Page, G. S. Tucker, E. Wollack, and E. L. Wright, First-year *Wilkinson Microwave Anisotropy Probe* (*WMAP*) observations: Implications for inflation, Astrophys. J. Suppl. Ser. **148**, 213 (2003).

[7] D. K. Hazra, L. Sriramkumar, and J. Martin, BINGO: A code for the efficient computation of the scalar bi-spectrum, J. Cosmol. Astropart. Phys. 05 (2013) 026.

[8] D. S. Salopek, J. R. Bond, and J. M. Bardeen, Designing density fluctuation spectra in inflation, Phys. Rev. D **40**, 1753 (1989).

[9] W. J. Handley, A. N. Lasenby, and M. P. Hobson, The Runge-Kutta-Wentzel-Kramers-Brillouin method, arXiv:1612.02288.

[10] K. F. Riley, M. P. Hobson, and S. J. Bence, *Mathematical Methods for Physics and Engineering*, 3rd ed. (Cambridge University Press, Cambridge, 2006).

[11] M. Abramowitz, *Handbook of Mathematical Functions, with Formulas, Graphs, and Mathematical Tables* (Dover, New York, 1965).

[12] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes: The Art of Scientific Computing*, 3rd ed. (Cambridge University Press, New York, 2007).

[13] K. Jordán, *Calculus of Finite Differences* (Chelsea Publishing Company, New York, 1965).

[14] J. C. Butcher, *The Numerical Analysis of Ordinary Differential Equations: Runge-Kutta and General Linear Methods* (Wiley, Chichester, 1987).

[15] P. Bogacki and L. Shampine, An efficient Runge-Kutta (4, 5) pair, Comput. Math. Appl. **32**, 15 (1996).

[16] R. Brankin, I. Gladwell, and L. F. Shampine, RKSUITE: A suite of explicit Runge-Kutta codes, in *Contributions in Numerical Mathematics* (World Publishing, Singapore, 1993), pp. 41–53.

[17] The Numerical Algorithms Group (NAG), Oxford, United Kingdom, The NAG Library, http://www.nag.com.

[18] W. Gander and J. Hrebicek, *Solving Problems in Scientific Computing Using Maple and MATLAB*, 4th ed. (Springer, Berlin, 2004).

[19] B. V. Noumerov, A method of extrapolation of perturbations, Mon. Not. R. Astron. Soc. **84**, 592 (1924).

[20] J. Killingbeck, Shooting methods for the Schrödinger equation, J. Phys. A: Math. Gen. **20**, 1411 (1987).

[21] K. Banerjee, S. P. Bhatnagar, V. Choudhry, and S. S. Kanwal, The anharmonic oscillator, Proc. R. Soc. London, Ser. A **360**, 575 (1978).

[22] A. H. Guth, Inflationary universe: A possible solution to the horizon and flatness problems, Phys. Rev. D **23**, 347 (1981).

[23] C. R. Contaldi, M. Peloso, L. Kofman, and A. Linde, Suppressing the lower multipoles in the CMB anisotropies, J. Cosmol. Astropart. Phys. 07 (2003) 002.

[24] W. J. Handley, S. D. Brechet, A. N. Lasenby, and M. P. Hobson, Kinetic initial conditions for inflation, Phys. Rev. D **89**, 063505 (2014).

[25] N. D. Birrell and P. C. W. Davies, *Quantum Fields in Curved Space*, Cambridge Monographs on Mathematical Physics (Cambridge University Press, Cambridge, 1982).

[26] W. J. Handley, A. N. Lasenby, and M. P. Hobson, Novel quantum initial conditions for inflation, Phys. Rev. D **94**, 024041 (2016).

[27] M. J. Mortonson, H. V. Peiris, and R. Easther, Bayesian analysis of inflation: Parameter estimation for single field models, Phys. Rev. D **83**, 043505 (2011).

[28] R. Flauger, L. McAllister, E. Pajer, A. Westphal, and G. Xu, Oscillations in the CMB from axion monodromy inflation, J. Cosmol. Astropart. Phys. 06 (2010) 009.

[29] B. Allen, Vacuum states in de Sitter space, Phys. Rev. D **32**, 3136 (1985).

[30] L. T. Hergt, W. J. Handley, M. P. Hobson, and A. N. Lasenby, Case for kinetically dominated initial conditions for inflation, Phys. Rev. D **100**, 023502 (2019).

[31] L. T. Hergt, W. J. Handley, M. P. Hobson, and A. N. Lasenby, Constraining the kinetically dominated universe, Phys. Rev. D **100**, 023501 (2019).

[32] W. I. J. Haddadin and W. J. Handley, Rapid numerical solutions for the Mukhanov-Sazaki equation, arXiv:1809.11095.

[33] R. P. Brent, *Algorithms for Minimization without Derivatives* (Prentice-Hall, 1973), Chap. 4.

[34] W. Handley, Primordial power spectra for curved inflating universes, Phys. Rev. D **100**, 123517 (2019).

[35] A. Lasenby and C. Doran, Closed universes, de Sitter space, and inflation, Phys. Rev. D **71**, 063502 (2005).

[36] J. Bamber and W. Handley, Beyond the Runge-Kutta-Wentzel-Kramers-Brillouin method, arXiv:1907.11638.

[37] See https://github.com/fruzsinaagocs/oscode.