- -

Explicit Stabilized Integrators for Stiff Optimal Control Problems

- -

Almuslimani, Ibrahim; Vilmart, Gilles

Explicit stabilized integrators for stiff optimal control problems

Ibrahim Almuslimani[1] and Gilles Vilmart[1]

November 24, 2020

Abstract

Explicit stabilized methods are an efficient alternative to implicit schemes for the time integration of stiff systems of differential equations in large dimension. In this paper we derive explicit stabilized integrators of orders one and two for the optimal control of stiff systems. We analyze their favorable stability properties based on the continuous optimality conditions. Furthermore, we study their order of convergence taking advantage of the symplecticity of the corresponding partitioned Runge-Kutta method involved for the adjoint equations. Numerical experiments including the optimal control of a nonlinear diffusion-advection PDE illustrate the efficiency of the new approach.

Keywords: optimal control, RKC, Chebyshev methods, symplectic methods, geometric integration, stability, adjoint control systems, double adjoint, Burgers equation, diffusion-advection PDE.

AMS subject classification: 49M25, 65L04, 65L06

1 Introduction

In this paper, we introduce and analyze numerical methods for the optimal control of systems of ordinary differential equations (ODEs) of the form

$$\min_u \ \Psi(y(T)); \qquad \dot{y}(t) := \frac{dy}{dt}(t) = f(u(t), y(t)), \quad t \in [0, T]; \ \ y(0) = y^0, \tag{1}$$

where for a fixed final time $T > 0$ and a given initial condition $y^0 \in \mathbb{R}^n$, the function $y : [0, T] \to \mathbb{R}^n$ is the unknown state function, $u : [0, T] \to \mathbb{R}^m$ is the unknown control function. Here, $f : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}^n$ is the given vector field and $\Psi : \mathbb{R}^n \to \mathbb{R}$ is the given cost function, which are assumed to be C^∞ mappings. For simplicity of the presentation, we consider the case of autonomous problems (with f independent of time) but we highlight that our approach also applies straightforwardly to non-autonomous problems $\frac{dy}{dt}(t) = f(t, u(t), y(t))$.[2]

There are essentially two approaches for the numerical solution of optimal control problems: the direct approach, which consists in directly discretizing (1) and then applying a minimization method to the corresponding discrete minimization problem, and the indirect approach, which is based on Pontryagin's maximum principle, taking benefit of continuous optimality conditions (adjoint equation). A natural approach for the accurate numerical approximation of such optimal control problem (1) is to consider Runge-Kutta type schemes. It was shown in [11, Theorem 4.1] by studying the continuous and discrete optimality conditions that additional order conditions for

---

[1]Université de Genève, Section de mathématiques, 2-4 rue du Lièvre, CP 64, CH-1211 Genève 4, Switzerland, Ibrahim.Almuslimani@unige.ch, Gilles.Vilmart@unige.ch

[2]A standard approach is to consider the augmented system with $z(t) = t$, i.e. $\frac{dz}{dt} = 1$, $z(0) = 0$ and define $\tilde{y}(t) = (y(t), z(t))^T$, see e.g. [12, Chap. III] for details.

the convergence rate are required in general by Runge-Kutta methods when applied to optimal control problems, compared to the integration of standard initial valued ordinary differential equations and conditions up to order 4 were derived. In [10], general order conditions were derived, in addition to identifying symplecticity properties. This result is related to the order of symplectic partitioned Runge-Kutta methods, and it implies in particular that applying naively a Runge-Kutta method to (1) yields in general an order reduction phenomenon. This analysis was then extended to other classes of Runge-Kutta type schemes in [17, 18, 14], see also [15, 8] in the context of hyperbolic problems and multistep methods. The use of symplectic integrators is motivated by the recent publication [20] which proves the convergence of forward-backward iterative algorithm (Algorithm 2.3 in the present paper), to implement discretized optimal control problems, when using a symplectic Runge-Kutta method. The work done in [20] generalizes that of [19] in which the authors prove the global convergence of the algorithm in the continuous time case. We also mention the paper [27] where automatic differentiation can be efficiently applied for computing the gradient of the cost function under the assumption that optimal control order conditions are satisfied. In our algorithms, the Jacobian of the vector field is given as an input, however the idea of automatic differentiation could be coupled with our approach to compute derivatives automatically, but this is not the purpose of the present paper.

In the case where the vector field f in (1) is stiff, due for instance to the multiscale nature of the model, or due to the spatial discretization of a diffusion operator in a partial differential equation (PDE) model, standard explicit integrators face in general a severe time step restriction making standard explicit methods unreasonable to be used due to their dramatic cost. A standard approach in this stiff case is to consider indirect implicit methods with good stability properties, as studied in [14] in the context of implicit-explicit (IMEX) Runge-Kutta methods for stiff optimal control problems. Note however that, already for initial value ODEs, such implicit methods can become very costly for nonlinear stiff problems in large dimension, requiring the usage of Newton-type methods and sophisticated linear algebra tools (preconditioners, etc.). Alternatively to using implicit methods, in this paper we focus on fully explicit indirect methods, and introduce new families of explicit stabilized methods for stiff optimal control problems. The proposed methods rely on the so-called Runge-Kutta-Chebyshev methods of order one and its extension RKC of order two [25]. Such explicit stabilized methods are popular in the context of initial value problems of stiff differential equations, particularly in high dimensions in the context of diffusive PDEs, see e.g. the survey [2]. It was extended to the stochastic context first in [4, 5] and recently in [3] for the design of explicit stabilized integrators with optimally large stability domains in the context of mean-square stable stiff and ergodic problems.

This paper is organized as follows. In Section 2, we recall standard tools on explicit stabilized methods and classical results on standard Runge-Kutta methods applied to optimal control problems. In Section 3, we introduce the new explicit stabilized schemes for optimal control problems and analyze their convergence and stability properties. Finally, Section 4 is dedicated to the numerical experiments where we illustrate the efficiency of the new approach.

2 Preliminaries

2.1 Discretization, order conditions, and symplecticity

Let us first recall the definition of a Runge-Kutta method for ordinary differential equations (ODEs),

$$\dot{y}(t) = F(y(t)), \qquad y(0) = y^0, \tag{2}$$

where $y : [0, T] \to \mathbb{R}^n$ is the unknown solution, $F : \mathbb{R}^n \to \mathbb{R}^n$ is a smooth vector field, and $y^0 \in \mathbb{R}^n$ is a given initial condition. We consider for simplicity a uniform discretization of the

interval $[0, T]$ with $N + 1$ points for $N \in \mathbb{N}$, and denote by $h = T/N$ the stepsize. For a given integer s and given real coefficients b_i, a_{ij} $(i, j = 1, \ldots, s)$, an s-stage Runge-Kutta method, $y_k \approx y(t_k)$, $t_k = kh$, to approximate the solution of (2), is defined, for all $k = 0, \ldots, N - 1$, by

$$y_{ki} = y_k + h \sum_{j=1}^{s} a_{ij} F(y_{kj}), \quad i = 1, \ldots, s, \qquad y_{k+1} = y_k + h \sum_{i=1}^{s} b_i F(y_{ki}). \tag{3}$$

The coefficients are usually displayed in a Butcher tableau as follows

$$\begin{array}{c|c} & a_{ij} \\ \hline & b_i \end{array}, \tag{4}$$

and we will sometimes use the notation (a_{ij}, b_i). For more details about the order conditions of Runge-Kutta methods in the context of initial value ODEs, we refer for example to the book [12, Chap. III]. We denote by $y_{k+1} = \Phi_h(y_k)$ the numerical flow of (3), while the time adjoint method Φ_h^* of Φ_h is the inverse map of the original method with reversed time step $-h$, i.e., $\Phi_h^* := \Phi_{-h}^{-1}$ [12, Sect. II.3]. We recall that the time adjoint of an s-stage Runge-Kutta method (a_{ij}, b_i) (3) is again an s-stage Runge-Kutta method with the same order of accuracy and its coefficients (a_{ij}^*, b_i^*) are given by $a_{ij}^* = b_{s+1-j} - a_{s+1-i,s+1-j}$ and $b_i^* = b_{s+1-i}$, where $i, j = 1, \ldots, s$.

If we discretize (1) using a Runge-Kutta discretization as above we naturally get the following discrete optimization problem,

$$\min \quad \Psi(y_N); \quad \text{subject to:}$$
$$y_{k+1} = y_k + h \sum_{i=1}^{s} b_i f(u_{ki}, y_{ki}), \qquad y_{ki} = y_k + h \sum_{j=1}^{s} a_{ij} f(u_{kj}, y_{kj}), \tag{5}$$

where $i = 1 \ldots, s$, $k = 0, \ldots, N - 1$, and $y_0 = y^0$. We denote by p_{ode} the order of accuracy of the method (3) applied to the ODE problem (2) and by p_{oc} the order of the method (5) for solving the optimal control problem (1). Note that we always have $p_{oc} \leq p_{ode}$. In general, $p_{oc} < p_{ode}$ because additional order conditions, described in [11, 10], have to be satisfied.

Let us denote by $H(u, y, p) := p^T f(u, y)$ the pseudo-Hamiltonian of the system where p is the Lagrange multiplier (or the costate) associated to the state y. Applying Pontryagin's maximum (or minimum) principle, the first order optimality conditions of (1) are given by the following boundary value problem,

$$\begin{aligned}
\dot{y}(t) &= f(u(t), y(t)) = \nabla_p H(u(t), y(t), p(t)), \\
\dot{p}(t) &= -\nabla_y f(u(t), y(t)) p = -\nabla_y H(u(t), y(t), p(t)), \\
0 &= \nabla_u H(u(t), y(t), p(t)). \\
t &\in [0, T], \quad y(0) = y^0, \quad p(T) = \nabla \Psi(y(T)).
\end{aligned} \tag{6}$$

Applying a Runge-Kutta integrator naively to (6) as an initial value system of ODEs combined with the classical methodology of shooting methods, would lead to severe instability due to the forward in time integration of the costate equation. For instance, for the optimal control of a diffusion PDE problem such as $\partial_t y(t, x) = \Delta y(t, x) + u(t, x)$, where Δ is the Laplace operator, $y(t, x)$ is the state function, and $u(t, x)$ is the control function (see also the diffusion-convection PDE problem considered in Sect. 4.2), then the costate equation takes the form of a heat equation with the wrong sign, $\partial_t p(t, x) = -\Delta p(t, x)$, which is naturally unstable if integrated forward in time. This makes classical shooting methods not applicable in the context of stiff dissipative optimal control problems considered in this paper. Alternatively, we consider a forward-backward iterative algorithm as described below (Algorithm 2.3).

Introducing Lagrange multipliers for the finite dimensional optimization problem (5), and supposing that $b_i \neq 0$ for all $i = 1 \ldots, s$, a calculation [11, 10] yields the following discrete optimality conditions

$$y_{k+1} = y_k + h \sum_{i=1}^{s} b_i f(u_{ki}, y_{ki}), \qquad y_{ki} = y_k + h \sum_{j=1}^{s} a_{ij} f(u_{kj}, y_{kj}),$$

$$p_{k+1} = p_k - h \sum_{i=1}^{s} \hat{b}_i \nabla_y H(u_{ki}, y_{ki}, p_{ki}), \; p_{ki} = p_k - h \sum_{j=1}^{s} \hat{a}_{ij} \nabla_y H(u_{kj}, y_{kj}, p_{kj}), \qquad (7)$$

$$0 = \nabla_u H(u_{ki}, y_{ki}, p_{ki}), \quad k = 0, \ldots, N-1 \quad i = 1, \ldots, s,$$

$$y_0 = y^0, \quad p_N = \nabla \Psi(y_N)$$

where the coefficients $\hat{b}_i$ and $\hat{a}_{ij}$ are defined by the following relations which, as observed in [10], correspond to the symplecticity conditions of partitioned Runge-Kutta methods for ODEs,

$$\hat{b}_i := b_i, \qquad \hat{a}_{ij} := b_j - \frac{b_j}{b_i} a_{ji}, \qquad i = 1, \ldots s, j = 1, \ldots, s. \qquad (8)$$

Note that the vectors p_k and p_{ki} are the Lagrange multipliers associated to y_k and y_{ki} respectively. Assuming that the Hessian matrix $\nabla_u^2 H(u, y, p) \in \mathbb{R}^{m \times m}$ is invertible along the trajectory of the exact solution, by the implicit function theorem there exists a C^∞ function ϕ such that $u = \phi(y, p)$ and then (7) is equivalent to a partitioned Runge-Kutta (PRK) method. As noticed in [10], if we consider the problem (6) as a Hamiltonian system, with the Hamiltonian function $\mathcal{H}(y, p) := H(\Psi(y, p), y, p)$, then the obtained PRK (7) scheme is symplectic thanks to the relations (8).

Theorem 2.1 (Theorem 4.1 in [11]). *Consider a Runge-Kutta method (a_{ij}, b_i) of order p_{ode} for ODEs, where $b_i \neq 0$, for all $i = 1, \ldots, s$, applied to the optimal control problem (1). Consider the optimality conditions (6), and assume that $\nabla_u^2 H(u, y, p)$ is invertible in a neighborhood of the solution, then we have the following theorem. If we discretize (6) using an s-stage partitioned Runge-Kutta method $(a_{ij}, b_i) - (\hat{a}_{ij}, \hat{b}_i)$ of order $p_{ode}^\star$ for ODEs (as partitioned RK method), and the condition (8) is satisfied, then the order p_{oc} of (5) satisfies $p_{oc} = p_{ode}^\star \leq p_{ode}$ and the schemes (5) and (7) are equivalent. In particular, for $p_{ode} \geq 2$, equivalently $\sum_{i=1}^{s} b_i = 1$ and $\sum_{i,j=1}^{s} b_i a_{ij} = \frac{1}{2}$, we get $p_{ode}^\star \geq 2$ and $p_{oc} \geq 2$.*

The proof of Theorem 2.1 relies on the commutativity of the following diagram [10, Sect. 2] which means that methods (5) and (7), colorredwhere (7) is a symplectic partitioned Runge-Kutta method, yield exactly the same outputs (up to round-off errors) if derived for Runge-Kutta discretizations of (1) and (6) respectively. We also refer to the article [22] where the role of symplectic partitioned Runge-Kutta methods involved in this commutative diagram is discussed.

$$
\begin{array}{ccc}
(1) & \xrightarrow{\text{discretization}} & (5) \\
\text{optimality conditions} \downarrow & & \downarrow \text{optimality conditions} \\
(6) & \xrightarrow[\text{discretization}]{} & (7)
\end{array}
$$

Remark that in (7), if the method (a_{ij}, b_i) is explicit, then $(\hat{a}_{ij}, \hat{b}_i)$ is in contrast an implicit method. Hence it is useful to consider the costate equation backward in time and use the time adjoint of $(\hat{a}_{ij}, \hat{b}_i)$ which turns out to be explicit as shown in Proposition 3.1 in Section 3. Indeed, consider method (7) and proceed as in [11, 14],

$$p_{k+1} + h \sum_{i=1}^{s} \hat{b}_i \nabla_y H(u_{ki}, y_{ki}, p_{ki}) = p_{ki} + h \sum_{j=1}^{s} \hat{a}_{ij} \nabla_y H(u_{kj}, y_{kj}, p_{kj}),$$

we then deduce from the identity $b_j - \hat{a}_{ij} = \frac{b_j}{b_i} a_{ji}$ the following formulation where p_N serves to initialize the induction on $k = N - 1, \ldots, 0$,

$$p_k = p_{k+1} + h \sum_{i=1}^{s} b_i \nabla_y H(u_{ki}, y_{ki}, p_{ki}), \quad p_{ki} = p_{k+1} + h \sum_{j=1}^{s} \frac{b_j}{b_i} a_{ji} \nabla_y H(u_{kj}, y_{kj}, p_{kj}).$$

The above Runge-Kutta method $(\tilde{a}_{ij}, \tilde{b}_i) := (\frac{b_j}{b_i} a_{ji}, b_i)$ for the costate is in fact the time adjoint of $(\hat{a}_{ij}, \hat{b}_i)$. Since the method $(\hat{a}_{ij}, \hat{b}_i)$ is called in the literature the adjoint method in the sense of optimal control because it is applied to the adjoint equation (costate), we call the method $(\tilde{a}_{ij}, \tilde{b}_i) := (\frac{b_j}{b_i} a_{ji}, b_i)$ the **double adjoint** of (a_{ij}, b_i), and we rewrite method (7) as

$$y_{k+1} = y_k + h \sum_{i=1}^{s} b_i f(u_{ki}, y_{ki}), \quad k = 0, \ldots, N-1$$

$$y_{ki} = y_k + h \sum_{j=1}^{s} a_{ij} f(u_{kj}, y_{kj}), \quad k = 0, \ldots, N-1 \quad i = 1, \ldots, s$$

$$p_k = p_{k+1} + h \sum_{i=1}^{s} \tilde{b}_i \nabla_y H(u_{ki}, y_{ki}, p_{ki}), \quad k = N-1, \ldots, 0 \tag{9}$$

$$p_{ki} = p_{k+1} + h \sum_{j=1}^{s} \tilde{a}_{ij} \nabla_y H(u_{kj}, y_{kj}, p_{kj}), \quad k = N-1, \ldots, 0 \quad i = s, \ldots, 1$$

$$0 = \nabla_u H(u_{ki,ki}, p_{ki}), \quad k = 0, \ldots, N-1 \quad i = 1, \ldots, s$$

$$y_0 = y^0, \quad p_N = \nabla \Psi(y_N).$$

Note that we integrate the state forward in time (increasing indices k) and the costate backward in time (decreasing k).

An immediate consequence of Theorem 2.1 is that applying naively a Runge-Kutta method yields in general an order reduction, as stated in the following remark.

Remark 2.2. *Consider a Runge-Kutta method (a_{ij}, b_i) of order $p_{ode} = 2$ and define $(\tilde{a}_{ij}, \tilde{b}_i) := (a_{ij}, b_i)$, in general the obtained partitioned Runge-Kutta method (9) is not of order $p_{oc} = 2$. Indeed, the coupling order conditions $\sum_{i,j=1}^{s} b_i \hat{a}_{ij} = \frac{1}{2}$ and $\sum_{i,j=1}^{s} \hat{b}_i a_{ij} = \frac{1}{2}$ are not automatically satisfied in general. In particular, for (a_{ij}, b_i) being the standard order two RKC method studied in the next section below, it can be checked that $p_{oc} = 1$. This makes non trivial the construction of an explicit stabilized scheme of order 2 for optimal control problems, as described in section 3.3. We will see that the notion of double adjoint of a Runge-Kutta method, as described above, is an essential tool in our study.*

To implement (9) (equivalent to (7)), we shall use the following classical iterative algorithm which was proposed as a parallel algorithm with N sub-problems in [21, Algo. 4]. For simplicity of the presentation, we only recall the non parallel algorithm, but emphasize that the parallel version could also be used in our context with explicit stabilized schemes.

Algorithm 2.3. *(see for instance [21, Algo. 4]). First start with an initial guess for the internal stages of the control $U^0 = (u_{ki}^0)_{k=0\ldots,N-1}^{i=1,\ldots,s}$ where $u_{ki}^0 \in \mathbb{R}^m$ for all k and i. Denote by $y^l = (y_k^l)_{k=0,\ldots,N-1}$ and $Y^l = (y_{ki}^l)_{k=0\ldots,N-1}^{i=1,\ldots,s}$ the collection of the state values and its internal stages respectively at iteration l, and analogously we use the notations p^l and P^l for costate and its internal stages at iteration l. Suppose that at the iteration l, U^l is known. For the next iteration $l+1$, the computation of U^{l+1} is achieved as follows.*

1. *Compute Y^l, P^l, y^l, p^l as in (9), the computation is done forward in time for the state y_k and backward in time for the costate p_k.*

2. *Compute $\tilde{u}_{ki}^{l+1}$ solving the system $\nabla_u H(\tilde{u}_{ki}^{l+1}, y_{ki}^l, p_{ki}^l) = 0$, for all k and i using an analytical formula if available, or a Newton method for instance.*

3. *Denoting $\tilde{U}^l = (\tilde{u}_{ki}^l)_{k=0...,N-1}^{i=1,...,s}$, define U^{l+1} by $U^{l+1} = (1-\theta^l)U^l + \theta^l \tilde{U}^{l+1}$, where θ^l is defined to minimize the scalar function $\theta \mapsto \Psi(U^{l+1})$ using a simple trisection method for instance, where $\Psi : U \mapsto \Psi(y^N)$.*

We stop when $\|U^{l+1} - U^l\| \leq tol$, where tol is a prescribed small tolerance.

In the recent paper [20], it was shown that the forward-backward sweep iteration defined in Algorithm 2.3 used in implementing discretized optimal control problems converges when using a symplectic Runge-Kutta discretization, which strengthens the interest of such symplectic methods.

For simplicity, we assume in the rest of the paper that $y_0 = y^0$ always holds in (7).

2.2 Explicit stabilized methods

Stability is a crucial property of numerical integrators for solving stiff problems and we refer to the book [13]. A Runge-Kutta method is said to be stable if the numerical solution stays bounded along the integration process. Applying a Runge-Kutta method (3) to the linear test problem (with fixed parameter $\lambda \in \mathbb{C}$),

$$\dot{y} = \lambda y, \quad y(0) = y_0, \tag{10}$$

with stepsize h yields a recurrence of the form $y_{k+1} = R(h\lambda)y_k$ and by induction we get $y_k = R(h\lambda)^k y_0$. The function $R(z)$ is called the stability function of the method and the stability domain is defined as $\mathcal{S} := \{z \in \mathbb{C}; |R(z)| \leq 1\}$, and y_k remains bounded if and only if $h\lambda \in \mathcal{S}$. The same result also applies to the internal stages of the Runge-Kutta method, for all $i = 1, \ldots, s$, where s is the number of internal stages, $y_{ki} = R_i(h\lambda)y_k$, for some function R_i. Remark that $R(z)$ is a rational function for implicit methods, but in the case of explicit methods the stability function $R(z)$ reduces to a polynomial. The simplest Runge-Kutta type method to integrate ODEs (2) is the explicit Euler method $y_{k+1} = y_k + hf(y_k)$ with stability polynomial $R(z) = 1+z$. However, its stability domain $\mathcal{S}$ is small (it reduces to the disc of center -1 and radius 1 in the complex plane) which yields a severe time step restriction and makes it very expensive for stiff problems.

2.2.1 Optimal first order Chebyshev methods

The idea of explicit stabilized methods (as introduced in [25], see the survey [2]) is to construct explicit Runge-Kutta integrators with extended stability domain that grows quadratically with the number of stages s of the method along the negative real axis, and hence allows to use large time steps typically for problems arising from diffusion partial differential equations. The family of methods considered in [25] is known as "Chebyshev methods" since its construction relies on Chebyshev polynomials $T_s(x)$ satisfying $T_s(\cos\theta) = \cos(s\theta)$. These polynomials allow us to obtain a two-step recurrence formula and hence low memory requirements and good internal stability with respect to round-off errors. The order one Chebyshev method for solving a stiff ODE (2) is defined as an explicit s-stage Runge-Kutta method by the recurrence

$$\begin{aligned}
y_{k0} &= y_k, \quad y_{k1} = y_k + \mu_1 hF(y_{k0}), \\
y_{ki} &= \mu_i hF(y_{k,i-1}) + \nu_i y_{k,i-1} + (1-\nu_i)y_{k,i-2}, \quad j = 2, \ldots, s
\end{aligned} \tag{11}$$

$$y_{k+1} = y_{ks},$$

where

$$\omega_0 := 1 + \frac{\eta}{s^2}, \quad \omega_1 := \frac{T_s(\omega_0)}{T_s'(\omega_0)}, \quad \mu_1 := \frac{\omega_1}{\omega_0}, \tag{12}$$

where η is called the damping parameter and is used to make the stability of the method robust with respect to small perturbations as described below. Finally, for all $i = 2, \dots, s$,

$$\mu_i := \frac{2\omega_1 T_{i-1}(\omega_0)}{T_i(\omega_0)}, \quad \nu_i := \frac{2\omega_0 T_{i-1}(\omega_0)}{T_i(\omega_0)}. \tag{13}$$

One can easily check that the (family) of methods (11) has the same first order of accuracy as the explicit Euler method (recovered for $s = 1$). Note that instead of the standard Runge-Kutta method formulation (3) with coefficients (a_{ij}, b_i), the one step method $y_{k+1} = \Phi_h(y_k)$ in (11) should be implemented using a recurrence relation (indexed by j) inspired from the relation (14) on Chebyshev polynomials

$$T_j(z) = 2z T_{j-1}(z) - T_{j-2}(z), \tag{14}$$

where $T_0(z) = 1, T_1(z) = z$. This implementation (11) yields a good stability [25] of the scheme with respect to round-off errors. The most interesting feature of this scheme is its stability behavior. Indeed, the method (11) applied to (10) yields, with $z = \lambda h$, $y_{k+1} = R_s^\eta(z) y_k = \frac{T_s(\omega_0 + \omega_1 z)}{T_s(\omega_0)} y_k$. A large real negative interval $(-C_\eta s^2, 0)$ is included in the stability domain of the method $\mathcal{S} := \{z \in \mathbb{C}; |R_s^\eta(z)| \le 1\}$. For the internal stages, we have analogously $y_{ki} = R_{s,i}^\eta(z) y_k = \frac{T_i(\omega_0 + \omega_1 z)}{T_i(\omega_0)} y_k$. The constant $C_\eta = 2 - 4/3\,\eta + \mathcal{O}(\eta^2)$ depends on the so-called damping parameter η and for $\eta = 0$, it reaches the maximal value $C_0 = 2$ (also optimal with respect to all possible stability polynomials for explicit schemes of order 1). Hence, given the stepsize h, for dissipative vector fields with a Jacobian having large real negative eigenvalues (such as diffusion problems) with spectral radius $\lambda_{\max}$ at y_n, the parameter s for the next step y_{n+1} can be chosen adaptively as[1]

$$s := \left\lceil \sqrt{\frac{h\lambda_{\max} + 1.5}{2 - 4/3\,\eta}} + 0.5 \right\rceil, \tag{15}$$

see [1] in the context of stabilized schemes of order two with adaptive stepsizes. The method (11) is much more efficient as its stability domain increases *quadratically* with the number s of function evaluations while a composition of s explicit Euler steps (same cost) has a stability domain that only increases *linearly* with s. In Figure 1 we plot the internal stages for $s = 10$ and different values $\eta = 0$ and $\eta = 0.05$, respectively. We observe that in the absence of damping ($\eta = 0$), the stability function (here a polynomial) is bounded by 1 in the large real interval $[-2s^2, 0]$ of width $2 \cdot 10^2 = 200$. However, for all z that are local extrema of the stability function, where $|R_s^\eta(z)| = 1$, the stability domain is very narrow in the complex plane. Here the importance of some damping appears, to make the scheme robust with respect to small perturbations of the eigenvalues. A typical recommended value for the damping parameter is $\eta = 0.05$, see [26, 2]. The advantage of this damping is that the stability polynomial is now **strictly** bounded by 1 and the stability domain includes a neighborhood of the negative interval $(-C_\eta s^2, 0)$. This improvement costs a slight reduction of the stability domain length from $2s^2$ to $C_\eta s^2$ where $C_\eta \ge 2 - \frac{4}{3}\eta$.

---

[1]The notation $[x]$ stands for the integer rounding of real numbers.
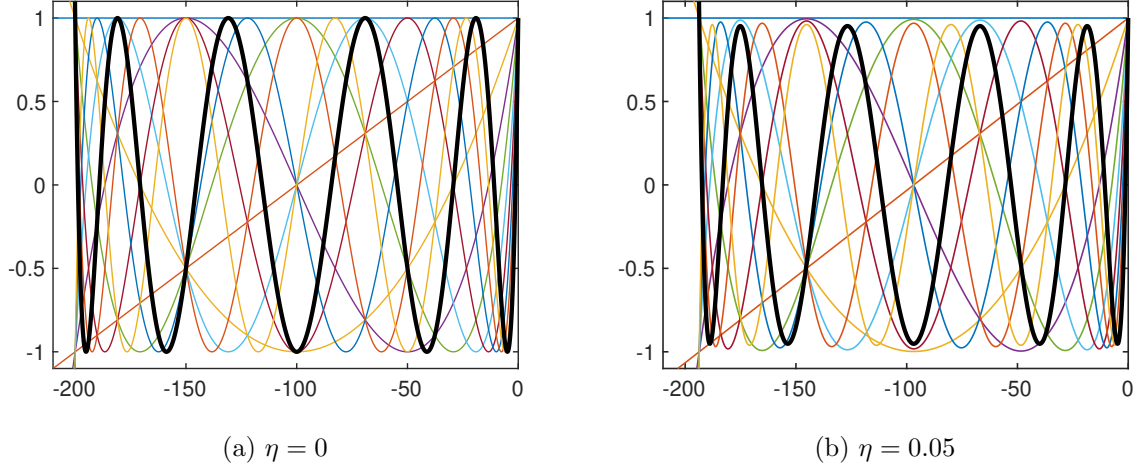
(a) $\eta = 0$ (b) $\eta = 0.05$

Figure 1: Internal stages (thin curves) and stability polynomials (bold curves) of the Chebyshev method (11) for $s = 10$ with and without damping.

2.2.2 Second order RKC methods

To design a second order method, we need the stability polynomial to satisfy[2] $R(z) = 1 + z + \frac{z^2}{2} + \mathcal{O}(z^3)$. In [9], Bakker introduced a correction to the first order shifted Chebyshev polynomials to get the following second order polynomial

$$R_s^\eta(z) = a_s + b_s T_s(\omega_0 + \omega_2 z), \tag{16}$$

where,

$$a_s = 1 - b_s T_s(\omega_0), \quad b_s = \frac{T_s''(\omega_0)}{(T_s'(\omega_0)^2)}, \quad \omega_0 = 1 + \frac{\eta}{s^2}, \quad \omega_2 = \frac{T_s'(\omega 0)}{T_s''(\omega_0)}, \quad \eta = 0.15. \tag{17}$$

For each s, $|R_s^\eta(z)|$ remains bounded by $a_b + b_s = 1 - \eta/3 + \mathcal{O}(\eta^2)$ for z in the stability interval (except for a small interval near the origin). The stability interval along the negative real axis is approximately $[-0.65s^2, 0]$, and covers about 80% of the optimal stability interval for second order stability polynomials, and the formula now for calculating s for a given time step h is

$$s := \left\lceil \sqrt{\frac{h\lambda_{\max} + 1.5}{0.65}} + 0.5 \right\rceil . \tag{18}$$

Using the recurrence relation of the Chebyshev polynomials, the RKC method as introduced in [25] is defined by

$$
\begin{aligned}
y_{k0} &= y_k, \quad y_{k1} = y_{k0} + hb_1\omega_2 f(y_{k0}), \\
y_{ki} &= y_{k0} + \mu_i' h(f(y_{k,i-1}) - a_{i-1}f(y_{k0})) + \nu_i'(y_{k,i-1} - y_{k0}) \\
&\quad + \kappa_i'(y_{k,i-2} - y_{k0}), \\
y_{k+1} &= y_{ks},
\end{aligned}
\tag{19}
$$

where,

$$\mu_i' = \frac{2b_i\omega_2}{b_{i-1}}, \quad \nu_i' = \frac{2b_i\omega_0}{b_{i-1}}, \quad \kappa_i' = -\frac{b_i}{b_{i-2}}, \quad b_i = \frac{T_i''(\omega_0)}{(T_i'(\omega_0)^2)}, \quad a_i = 1 - b_i T_i(\omega_0), \tag{20}$$

---

[2]Indeed, up to order two, the order conditions for nonlinear problems are the same as the order conditions for linear problems [12, Chap. III].
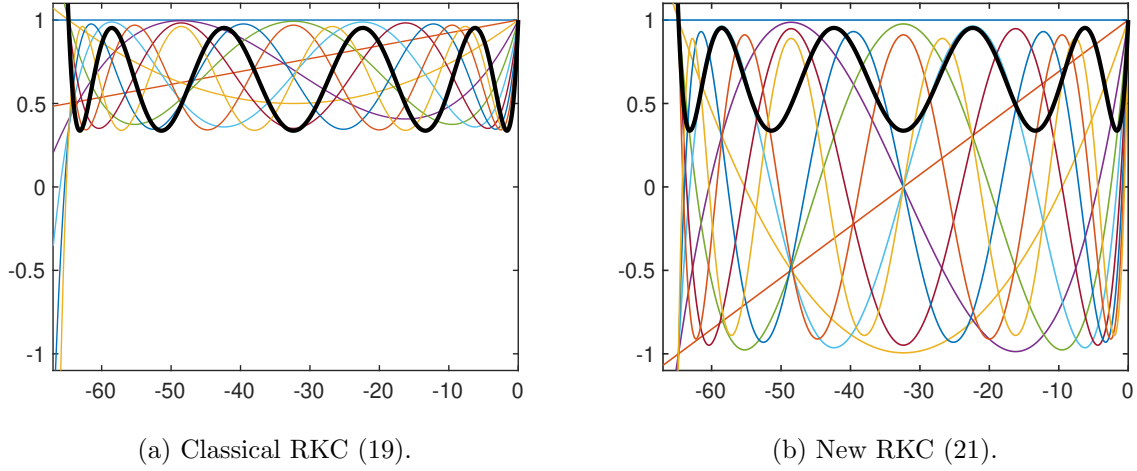
(a) Classical RKC (19).   (b) New RKC (21).

Figure 2: Internal stages (thin curves) and stability polynomials (bold curves) of the classical (19) and the new (21) RKC implementations for $s = 10$ internal stages.

for $i = 2, \ldots, s$. As in (16), the stability functions of the internal stages are given by $R_i^\eta(z) = a_i + b_i T_i(\omega_0 + \omega_2 z)$, where $i = 0, \ldots, s - 1$, and the parameters a_i and b_i are chosen such that the above stages are consistent $R_i^\eta(z) = 1 + \mathcal{O}(z)$. The parameters b_0 and b_1 are free ($R_0^\eta(z)$ is constant and only order 1 is possible for $R_1^\eta(z)$) and the values $b_0 = b_1 = b_2$ are suggested in [24]. In this paper, to facilitate the analysis of the internal stability of the optimal control methods, making the internal stages of the RKC method analogous to the Chebyshev method (11) of order one, we introduce a new implementation of RKC method

$$
\begin{aligned}
y_{k0} &= y_k, \quad y_{k1} = y_k + \mu_1 h F(y_{k0}), \\
y_{ki} &= \mu_i h F(y_{k,i-1}) + \nu_i y_{k,i-1} + (1 - \nu_i) y_{k,i-2}, \quad i = 2, \ldots, s \\
y_{k+1} &= a_s y_{k0} + b_s T_s(\omega_o) y_{ks},
\end{aligned}
\tag{21}
$$

where $\mu_1 = \frac{\omega_2}{\omega_0}$, a_s, b_s are given in (17), and the parameters μ_i and ν_i are defined by (analogously to (13), using ω_2 instead of ω_1), $\mu_i = \frac{2\omega_2 T_{i-1}(\omega_0)}{T_i(\omega_0)}$, $\nu_i = \frac{2\omega_0 T_{i-1}(\omega_0)}{T_i(\omega_0)}$, for $i = 2, \ldots, s$. This new formulation (21) yields the same stability function $R_s^\eta(z)$ in (16) but different internal stages, and it will be helpful when we introduce the double adjoint of RKC in Section 3. We recall that for an accurate implementation, one should not use the standard Runge-Kutta formulations with coefficients (a_{ij}, b_i) for (11) and (19) since they are unstable due to the accumulated round-off error for large values of s. In contrast, the *low memory* induction formulations (11) and (19) are easy to implement and very stable with respect to round-off errors [25].

Note that (21) is not the same Runge-Kutta method as the standard RKC (19) from [25], it has different internal stages but the same stability function $R_s^\eta(z)$ in (16) and hence order $p_{ode} = 2$. In Figure 2 we can see that the internal stages of the new formulation (21) of RKC have an analogous behavior compared to those of the first order Chebyshev method oscillating around zero (see Figure 1), in contrast to those of the standard RKC (19), oscillating around the value $a_s > 0$. In addition, comparing Figures 2b and 1b, we see that the internal stages of the new RKC method (21) are the same as the order one Chebyshev method (11) up to a horizontal rescaling ω_2/ω_1. This is because the s internal stages of the methods have the stability function $T_i(\omega_0 + \omega_j z)/T_i(\omega_0)$, $i = 1, \ldots, s$ for $j = 1, 2$ respectively.

Notice however that this modification of the standard RKC method (19) deteriorates the order two of accuracy of the internal stages of the method, useful for PDEs with non homogeneous
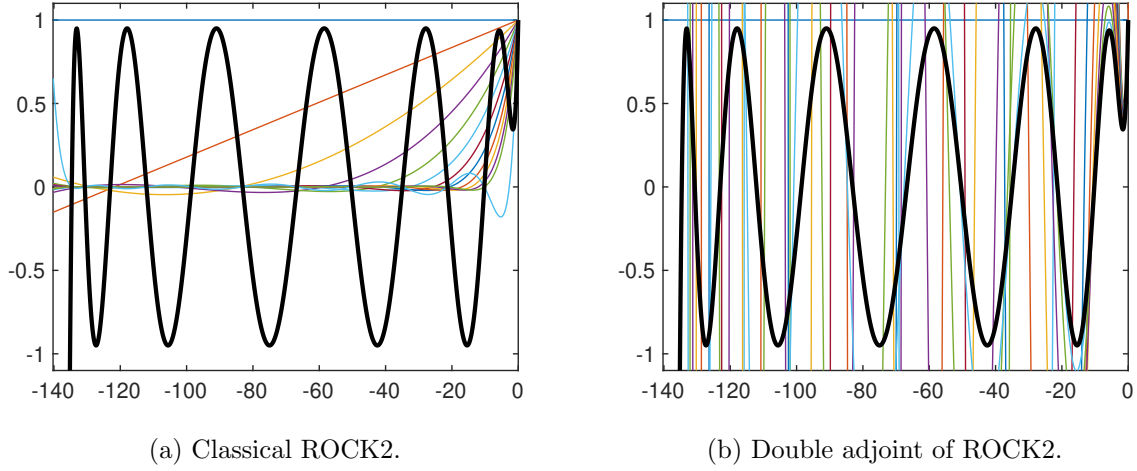
(a) Classical ROCK2.

(b) Double adjoint of ROCK2.

Figure 3: Internal stages (thin curves) and stability polynomials (bold curves) of the ROCK2 method [6] and its double adjoint for optimal control for $s = 13$ stages.

boundary conditions [16, Chap. V].

Remark 2.4. *Analogously to the standard RKC method* (19), *the new RKC formulation* (21) *can be equipped with an error estimator to allow a variable time step control. Since the new formulation* (21) *has the same stability function, one can use the same error estimator as proposed in* [23, Sect. 3.1]. *In this paper we consider only a constant time step for simplicity of the presentation but emphasize that a variable time step h_n can be used for the new optimal control method of order two.*

Remark 2.5. *Second order Runge-Kutta Orthogonal Chebyshev (ROCK2) methods, as introduced in* [6], *are second order explicit stabilized methods for which the stability domain contains an interval that covers around* 98% *of the optimal one for second order explicit methods. It would be interesting to extend such second order methods with nearly optimally large stability domain to the context of optimal control problems. It turns out however that such an extension based on ROCK2 (or its order four extension ROCK4 [1]) is difficult and not analyzed in the present paper. This difficulty arises from the severe instability of the internal stages of the double adjoint of the standard ROCK2 method (see Figure 3) which would introduce large round-off errors for stiff problems (large values of s), making the obtained optimal control method not reliable.*

3 Explicit stabilized methods for optimal control

In this section, we derive new two term recurrence relations of the double adjoints of Chebyshev method (11) and RKC method (21) that are numerically stable. Indeed, one cannot rely on standard Runge-Kutta coefficients for the implementation of explicit stabilized schemes.

3.1 Double adjoint of a general Runge-Kutta method

Recall from (9) that the Butcher tableau of the double adjoint $(\tilde{a}_{ij}, \tilde{b}_i)$ of (4) is

$$\frac{\quad}{\tilde{b}_i}\begin{array}{|c} \tilde{a}_{ji} \\ \hline \tilde{b}_i \end{array} := \frac{\quad}{b_i}\begin{array}{|c} \frac{b_j}{b_i}a_{ji} \\ \hline b_i \end{array}. \tag{22}$$

Proposition 3.1. *If a Runge-Kutta method (a_{ij}, b_i) (4) is explicit, then its double adjoint (22) is explicit as well.*

Proof. For an explicit Runge-Kutta method we have that $a_{ij} = 0$ for all $j \geq i$ i.e the matrix (a_{ij}) is strictly lower triangular. Permuting the internal stages in (22) for $i, j = s, \ldots, 2, 1$ does not modify the method but yields the following Butcher tableau

$$
\begin{array}{c|c}
 & \frac{b_{s+1-j}}{b_{s+1-i}} a_{s+1-j, s+1-i} \\
\hline
 & b_{s+1-i}
\end{array}
\tag{23}
$$

which is strictly lower triangular, and thus the method (22) is again explicit. $\qquad\square$

An immediate consequence of Proposition 3.1 is that for explicit methods, the stability function of the double adjoint is again a polynomial. In fact it turns out, as stated in Theorem 3.2 below, that for any Runge-Kutta method, the double adjoint $(\tilde{a}_{ij}, \tilde{b}_i)$ has exactly the same stability function as (a_{ij}, b_i). Note however that this result does not hold in general for the internal stages (see Remark 2.5 about ROCK2).

Theorem 3.2. *A Runge-Kutta method (a_{ij}, b_i) and its double adjoint $(\tilde{a}_{ij}, \tilde{b}_i)$ in (22) share the same stability function $R(z)$.*

Proof. Let $A = (a_{ij})$, $A_d = (a_{ji} b_j / b_i)$, and $b = (b_i)$, $i, j = 1 \ldots s$. We recall the formula for the stability function of the Runge-Kutta method (a_{ij}, b_i),

$$
R(z) = \frac{\det(I - zA + z\mathbb{1}b^T)}{\det(I - zA)},
\tag{24}
$$

where $\mathbb{1} \in \mathbb{R}^s$ is the line vector of size s containing only ones. Using a simple calculation, one can show that $A_d^T = DAD^{-1}$ where $D = \mathrm{diag}(b_i)$. This implies that $I - zA_d^T = D(I - zA)D^{-1}$, and since $I - zA_d^T = (I - zA_d)^T$, thus $\det(I - zA) = \det(I - zA_d)$. Using the same diagonal matrix D we have $D\mathbb{1}b^T D^{-1} = \mathbb{1}^T b$, hence $I - zA_d^T + z\mathbb{1}b = D(I - zA + z\mathbb{1}b)D^{-1}$ and $\det(I - zA_d^T + z\mathbb{1}b) = \det(I - zA + z\mathbb{1}b)$ and hence the stability function of $(\tilde{a}_{ij}, \tilde{b}_i)$ is again (24). $\qquad\square$

3.2 Chebyshev method of order one for optimal control problems

For clarity of presentation, we first study Chebyshev method of order one for optimal control problems before introducing the second order RKC method. Applying the order one Chebyshev method (11) to the problem (1) we get

$$
\begin{aligned}
&\min \ \Psi(y_N), \ \ such \ that \\
&y_{k0} = y_k, \quad y_{k1} = y_{k0} + \mu_1 h f(u_{k0}, y_{k0}), \\
&y_{ki} = \mu_i h f(u_{k,i-1}, y_{k,i-1}) + \nu_i y_{k,i-1} + (1 - \nu_i) y_{k,i-2}, \quad i = 2, \ldots, s \\
&y_{k+1} = y_{ks},
\end{aligned}
\tag{25}
$$

where, $k = 0, \ldots, N-1$, $\eta > 0$ is fixed, and the parameters μ_i, ν_i are defined as in (12) and (13).

For the implementation of Algorithm 2.3 based on the order one Chebyshev method (25) for the state equation, the costate equation can be implemented efficiently using the recurrence relations given by the following theorem.

Theorem 3.3. *The double adjoint of scheme* (25) *is given by the recurrence*

$$p_N = \nabla\Psi(y_N), \quad p_{ks} = p_{k+1}$$

$$p_{k,s-1} = p_{ks} + \frac{\mu_s}{\nu_s} h\nabla_y H(u_{k,s-1}, y_{k,s-1}, p_{ks})$$

$$p_{k,s-j} = \frac{\mu_{s-j+1}\alpha_{s-j+1}}{\alpha_{s-j}} h\nabla_y H(u_{k,s-j}, y_{k,s-j}, p_{k,s-j+1})$$

$$+ \frac{\nu_{s-j+1}\alpha_{s-j+1}}{\alpha_{s-j}} p_{k,s-j+1}$$

$$+ \frac{(1-\nu_{s-j+2})\alpha_{s-j+2}}{\alpha_{s-j}} p_{k,s-j+2}, \quad j = 2\ldots, s-1,$$ (26)

$$p_{k0} = \mu_1\alpha_1 h\nabla_y H(u_{k0}, y_{k0}, p_{k1}) + \alpha_1 p_{k1} + (1-\nu_2)\alpha_2 p_{k2}$$

$$p_k = p_{k0}$$

$$\nabla_u H(u_{k,s-j}, y_{k,s-j}, p_{k,s-j+1}) = 0, \quad j = 1, \ldots, s.$$

where $k = N-1, \ldots, 2, 1, 0$ *and the coefficients* α_j *are defined by induction as*

$$\alpha_s = 1, \quad \alpha_{s-1} = \nu_s,$$

$$\alpha_{s-j} = \nu_{s-j+1}\alpha_{s-j+1} + (1-\nu_{s-j+2})\alpha_{s-j+2}, \quad j = 2\ldots s-1.$$ (27)

The proof of Theorem 3.3 uses similar arguments to the proof of Theorem 2.1, with the exception that we now rely on the recurrence formula (25) instead of the standard Runge-Kutta formulation (6) to avoid numerical instability.

Proof of Theorem 3.3. The Lagrangian associated to the discrete optimization problem (25) is

$$\mathcal{L} = \Psi(y_N) + p_0 \cdot (y^0 - y_0) + \sum_{k=0}^{N-1} \left\{ p_{k+1} \cdot (y_{ks} - y_{k+1}) - p_{k0} \cdot (y_k - y_{k0}) \right.$$

$$+ p_{k1} \cdot (y_{k0} + \mu_1 h f(u_{k0}, y_{k0}) - y_{k1})$$

$$+ \sum_{i=2}^{s} p_{ki} \cdot (\mu_i h f(u_{k,i-1}, y_{k,i-1}) + \nu_i y_{k,i-1}$$

$$\left. + (1-\nu_i)y_{ki-2} - y_{ki}) \right\}.$$

Here p_{k+1}, p_{ki}, and p_0 are the Lagrange multipliers. The optimality necessary conditions are thus given by

$$\frac{\partial\mathcal{L}}{\partial y_k} = 0, \quad \frac{\partial\mathcal{L}}{\partial y_{ki}} = 0, \quad \frac{\partial\mathcal{L}}{\partial p_k} = 0, \quad \frac{\partial\mathcal{L}}{\partial p_{ki}} = 0, \quad \frac{\partial\mathcal{L}}{\partial u_{ki}} = 0,$$ (28)

where $k = 0, \ldots, N-1$ and $i = 0, \ldots, s$. By a direct calculation, we obtain the following system,

$$y_{k0} = y_k, \quad y_{k1} = y_{k0} + \mu_1 h f(u_{k0}, y_{k0}),$$

$$y_{ki} = \mu_i h f(u_{k,i-1}, y_{k,i-1}) + \nu_i y_{k,i-1} + (1-\nu_i)y_{k,i-2}, \quad i = 2, \ldots, s,$$

$$y_{k+1} = y_{ks},$$

$$p_N = \nabla\Psi(y_N), \quad p_{ks} = p_{k+1},$$

$$p_{k,s-1} = \mu_s h\nabla_y H(u_{k,s-1}, y_{k,s-1}, p_{ks}) + \nu_s p_{ks},$$

$$p_{k,s-j} = \mu_{s-j+1} h\nabla_y H(u_{k,s-j}, y_{k,s-j}, p_{k,s-j+1}) + \nu_{s-j+1} p_{k,s-j+1}$$

$$+ (1-\nu_{s-j+2})p_{k,s-j+2}, \quad j = 2, \ldots, s-1,$$ (29)

$$p_{k0} = \mu_1 h\nabla_y H(u_{k0}, y_{k0}, p_{k1}) + p_{k1} + (1-\nu_2)p_{k2},$$

$$p_k = p_{k0},$$

$$\nabla_u H(u_{k,s-j}, y_{k,s-j}, p_{k,s-j+1}) = 0, \quad j = 1, \ldots, s,$$

where $k = 0, \ldots, N - 1$. In the above system, observe that the steps p_{ki} of the double adjoint are not internal stages of a Runge-Kutta method, that is because they are not $\mathcal{O}(h)$ perturbations of the p_{k+1}, i.e. $p_{ki} \neq p_{k+1} + \mathcal{O}(h)$, for instance, already for the first step $p_{k,s-1} = \nu_s p_{k+1} + \mathcal{O}(h)$ with $\nu_s = 2 + \mathcal{O}(\eta)$. Since the pseudo-Hamiltonian $H(u, y, p)$ is linear in p, we can rescale the internal stages of the costate by a factor α_j such that for $\hat{p}_{kj} := \alpha_j^{-1} p_{kj}$, we obtain $\hat{p}_{kj} = p_{k+1} + \mathcal{O}(h)$. We define

$$\hat{p}_{ks} := p_{ks}, \quad \hat{p}_{k,s-1} := \frac{p_{k,s-1}}{\nu_s} = \hat{p}_{ks} + \frac{\mu_s}{\nu_s} h \nabla_y H(u_{k,s-1}, y_{k,s-1}, \hat{p}_{ks}).$$

Substituting $\hat{p}_{k,s-2}$ in (29), we obtain

$$p_{k,s-2} = \mu_{s-1} \nu_s h \nabla_y H(u_{k,s-2}, y_{k,s-2}, \hat{p}_{k,s-1}) + \nu_{s-1} \nu_s \hat{p}_{k,s-1} + (1 - \nu_s) \hat{p}_{ks},$$

the quantities $\hat{p}_{k,s-1}$ and $\hat{p}_{ks}$ are equal to $p_{k+1} + \mathcal{O}(h)$, hence $p_{k,s-2} = (\nu_{s-1}\nu_s + 1 - \nu_s) p_{k+1} + \mathcal{O}(h)$, this implies that $\alpha_{s-2} = \nu_{s-1}\nu_s + 1 - \nu_s$ and therefore

$$\hat{p}_{k,s-2} := \frac{p_{k,s-2}}{\nu_{s-1}\nu_s + 1 - \nu_s} = \frac{p_{k,s-2}}{\nu_s(\nu_{s-1} - 1) + 1}.$$

Following this procedure for $p_{k,s-j}, j = 2, \ldots, s - 1$, we arrive at the Runge-Kutta formulation (26) of the double adjoint of scheme (25), where we go back to the notation p_{ki} instead of $\hat{p}_{ki}$. $\quad\square$

Remark 3.4. *A straightforward calculation yields that without damping (for $\eta = 0$), we have $\alpha_{s-j} = j + 1$, and method (25)-(26) reduces to the following recurrence*

$$y_{k0} = y_k, \quad y_{k1} = y_{k0} + \frac{h}{s^2} f(u_{k0}, y_{k0}),$$

$$y_{ki} = \frac{2h}{s^2} f(u_{k,i-1}, y_{k,i-1}) + 2y_{k,i-1} - y_{k,i-2}, \quad i = 2, \ldots, s,$$

$$y_{k+1} = y_{ks},$$

$$p_N = \nabla\Psi(y_N), \quad p_{ks} = p_{k+1},$$

$$p_{k,s-1} = p_{ks} + \frac{h}{s^2} \nabla_y H(u_{k,s-1}, y_{k,s-1}, p_{ks}),$$

$$p_{k,s-j} = \frac{2j}{(j+1)s^2} h \nabla_y H(u_{k,s-j}, y_{k,s-j}, p_{k,s-j+1}) + \frac{2j}{j+1} p_{k,s-i+1} \tag{30}$$

$$+ \frac{1-j}{j+1} p_{k,s-j+2}, \quad j = 2, \ldots, s-1,$$

$$p_{k0} = \frac{h}{s} \nabla_y H(u_{k0}, y_{k0}, p_{k1}) + s p_{k1} + (1-s) p_{k2},$$

$$p_k = p_{k0},$$

$$\nabla_u H(u_{k,s-j}, y_{k,s-j}, p_{k,s-j+1}) = 0, \quad j = 1, \ldots, s.$$

where $k = 0, \ldots, N - 1$. In Section 3.4, we shall study the stability of (30) (without damping) and of (25)-(26) (with damping).

3.3 RKC method of order 2

We consider the new implementation (21) of the RKC method applied to (1) for which the internal stages behave similarly to that of the order one method, given by

$$\min \, \Psi(y_N), \textit{ such that}$$

$$y_{k0} = y_k, \quad y_{k1} = y_{k0} + \mu_1 h f(u_{k0}, y_{k0}),$$

$$y_{ki} = \mu_i h f(u_{k,i-1}, y_{k,i-1}) + \nu_i y_{k,i-1} + (1 - \nu_i) y_{k,i-2}, \quad i = 2, \ldots, s \tag{31}$$

$$y_{k+1} = a_s y_{k0} + b_s T_s(\omega_0) y_{ks},$$

where, $k = 0, \ldots, N - 1$, $\eta = 0.15$, and again all the parameters are defined as for the Chebyshev method using ω_2 instead of ω_1. The order two RKC method with formulation (31) for the state equation can be implemented using Algorithm 2.3, the costate equation being implemented using the recurrence relations given by the following Theorem 3.5. Its proof is analogous to that of Theorem 3.3 and thus omitted.

Theorem 3.5. *The double adjoint of the scheme* (31) *is given by the recurrence*

$$
\begin{aligned}
p_N &= \nabla \Psi(y_N), \quad p_{ks} = p_{k+1}, \\
p_{k,s-1} &= p_{ks} + \frac{\mu_s}{\nu_s} h \nabla_y H(u_{k,s-1}, y_{k,s-1}, p_{ks}), \\
p_{k,s-j} &= \frac{\mu_{s-j+1} \alpha_{s-j+1}}{\alpha_{s-j}} h \nabla_y H(u_{k,s-j}, y_{k,s-j}, p_{k,s-j+1}) \\
&\quad + \frac{\nu_{s-j+1} \alpha_{s-j+1}}{\alpha_{s-j}} p_{k,s-j+1}, \\
&\quad + \frac{(1 - \nu_{s-j+2}) \alpha_{s-j+2}}{\alpha_{s-j}} p_{k,s-j+2}, \quad j = 2, \ldots, s - 1, \\
p_{k0} &= \mu_1 \alpha_1 h \nabla_y H(u_{k0}, y_{k0}, p_{k1}) + \alpha_1 p_{k1} + (1 - \nu_2) \alpha_2 p_{k2} + a_s p_{k+1}, \\
p_k &= p_{k0}, \\
\nabla_u H&(u_{k,s-j}, y_{k,s-j}, p_{k,s-j+1}) = 0, \quad j = 1, \ldots, s,
\end{aligned}
\tag{32}
$$

where the coefficients α_j *are defined using the induction*

$$
\begin{aligned}
\alpha_s &= b_s T_s(\omega_0), \quad \alpha_{s-1} = \nu_s \alpha_s, \\
\alpha_{s-j} &= \nu_{s-j+1} \alpha_{s-j+1} + (1 - \nu_{s-j+2}) \alpha_{s-j+2}, \quad j = 2 \ldots s - 1.
\end{aligned}
\tag{33}
$$

In Figure 4, we plot the stability function and the internal stages of the double adjoint (26) of Chebyshev (11) and the double adjoint (32) of RKC (21). Comparing with Figures 1b and 2b, we observe that the internal stages are not the same for the double adjoint methods compared to the (11) and (21), while the stability function itself is identical as shown in Theorem 3.2 for a general Runge-Kutta method.

## 3.4   Stability and convergence analysis

In this section, we study the stability of the double adjoint of the Chebyshev method (25) and the RKC method (31). We recall that for the Chebyshev method of order one (resp. RKC of order two) the stability domain contains the interval $[-\beta(s, \eta), 0]$ where $\beta_{Cheb}(s, \eta) \approx (2 - 4\eta/3)s^2$ (resp. $\beta_{RKC}(s, \eta) \approx 0.653 s^2$ for $\eta = 0.15$).

Theorem 3.6. *Consider the Chebyshev* (25) *and the RKC* (31) *methods. For* $\eta = 0$, *the stability functions of the internal stages* $R_{s,i}(z)$ *of the Chebyshev (resp. RKC) double adjoint* (26) *(resp.* (32)*), are bounded by 1 for all* $z \in [-2s^2, 0]$ *(resp.* $[-\frac{2}{3}s^2 + \frac{2}{3}, 0]$*) and all* $s \in \mathbb{N}$.
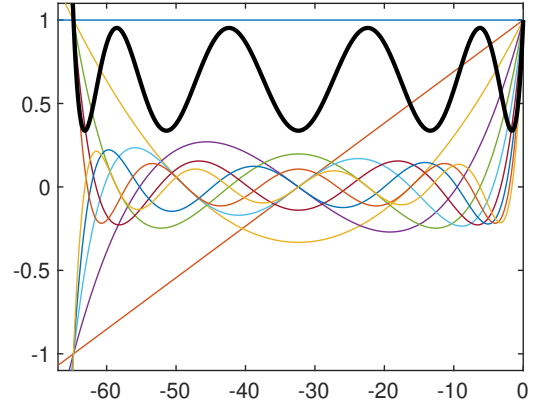
The proof of Theorem 3.6 relies on the following lemma.

Lemma 3.7. *Let* $s \geq 1$, *and consider the double sequence* $\tilde{\gamma}_j^i$ *indexed by* i *and* j,

$$
\begin{aligned}
\tilde{\gamma}_j^i &= 0 \ \forall \ j > i, \quad i = 0, \ldots, s - 1, \\
\tilde{\gamma}_0^0 &= 1, \quad \tilde{\gamma}_0^1 = 0, \quad \tilde{\gamma}_1^1 = 2,
\end{aligned}
\tag{34}
$$
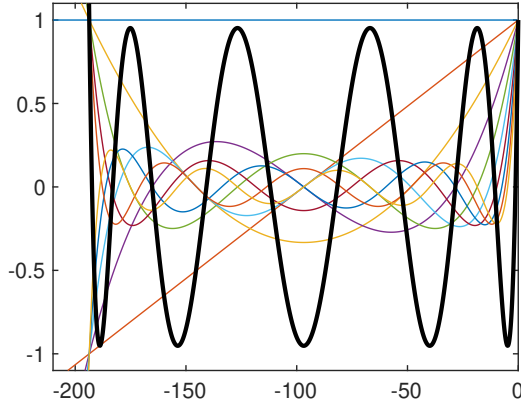
(a) Double adjoint (26) for $\eta = 0.05$

(b) Double adjoint (32) for $\eta = 0.15$

Figure 4: Internal stages (thin curves) and stability polynomial (bold curve) of the double adjoint of the Chebyshev method (26) of order one and the RKC method (32) of order two for $s = 10$ internal stages.

$$
\begin{aligned}
&\tilde{\gamma}_0^i = \tilde{\gamma}_1^{i-1} - \tilde{\gamma}_0^{i-2}, \quad \tilde{\gamma}_1^i = 2\tilde{\gamma}_0^{i-1} + \tilde{\gamma}_2^{i-1} - \tilde{\gamma}_1^{i-2} \qquad i = 2, \ldots, s-1, \\
&\tilde{\gamma}_j^i = \tilde{\gamma}_{j-1}^{i-1} + \tilde{\gamma}_{j+1}^{i-1} - \tilde{\gamma}_j^{i-2} \qquad i = 2, \ldots, s-1, \ j = 2, \ldots, i,
\end{aligned}
\tag{35}
$$

Then,

$$
\begin{aligned}
&\tilde{\gamma}_j^i = 0 \ \forall \ j > i, \\
&\tilde{\gamma}_0^i = \begin{cases} 1 & \text{if } i \text{ is even} \\ 0 & \text{if } i \text{ is odd} \end{cases} \qquad \tilde{\gamma}_j^i = \begin{cases} 2 & \text{if } i - j \text{ is even} \\ 0 & \text{otherwise} \end{cases}
\end{aligned}
\tag{36}
$$

where $i = 0, \ldots, s-1, \ j = 0, \ldots, i$.

Proof. It can be checked that the coefficients defined in (36) verify the induction (35). Hence using the fact that they have the same initial terms (34), we conclude that they coincide by induction on i and j. $\qquad\square$

Proof of Theorem 3.6. We first consider the Chebyshev method without damping applied to the linear test problem $y' = \lambda y$, $\lambda \in \mathcal{C}$, $t \in (0, T]$, $y(0) = 1$, with a uniform subdivision $x_0 = 0 < x_1 < \cdots < x_N = T$ of stepsize h. Using Remark 3.4, we obtain for $k = 0$:

$$
\begin{aligned}
&y_{k0} = 1, \quad y_{k1} = y_{k0} + \frac{h\lambda}{s^2} y_{k0}, \\
&y_{ki} = \frac{2ih\lambda}{(i+1)s^2} y_{k,i-1} + \frac{2i}{i+1} y_{k,i-1} + \frac{1-i}{i+1} y_{k,i-2}, \quad i = 2, \ldots, s-1, \\
&y_{ks} = \frac{h\lambda}{s} y_{k,s-1} + s y_{k,s-1} + (1-s) y_{k,s-2}, \\
&y_1 = y_{ks}.
\end{aligned}
\tag{37}
$$

First, notice that since $\frac{2i}{i+1} + \frac{1-i}{i+1} = 1$, we have that for all $i = 0, \ldots, s$, $y_{ki} = (1 + \mathcal{O}(h))$ (by induction). Setting $z = h\lambda$, it is sufficient to prove the identity

$$
y_{ki} = \sum_{j=0}^{i} \gamma_j^i T_j \left(1 + \frac{z}{s^2}\right)
\tag{38}
$$

15

where y_{ki} is a convex combination of the polynomials $T_j(1 + \frac{z}{s^2})$,

$$\sum_{j=0}^{i} \gamma_j^i = 1 \text{ and } \gamma_j^i \geq 0 \; \forall \; i, j = 1, \ldots s - 1, \tag{39}$$

because $|T_j(1 + \frac{z}{s^2})| \leq 1$ for all $j = 0, \ldots, s$ and $z \in [-\beta(s, 0), 0] = [-2s^2, 0]$.

Since the Chebyshev polynomials form a basis of the vector space of polynomials, this already justifies the existence of the expansion (38) with some real coefficients γ_j^i. The identity $\sum_{j=0}^{i} \gamma_j^i = 1$ follows from the fact that $y_{ki} = 1 + \mathcal{O}(h)$ and $T_j(1 + \frac{z}{s^2}) = 1 + \mathcal{O}(h)$ for all $i, j = 0, \ldots, s$. Now we can calculate these coefficients for the first two internal stages, $R_{s,0}(z) = y_{k0} = y_k = 1 = T_0(1 + \frac{z}{s^2})$, thus $\gamma_0^0 = 1$. Analogously, $R_{s,1}(z) = y_{k1} = y_{k0} + \frac{h\lambda}{s^2}y_{k0} = 1 + \frac{z}{s^2} = T_1(1 + \frac{z}{s^2})$ we obtain $\gamma_0^1 = 0$, $\gamma_1^1 = 1$.

It remains to prove the positivity of the coefficients γ_j^i. Coupling (38) and (39), we obtain

$$
\begin{aligned}
R_{s,i}(z) = y_{ki} &= \frac{2i}{i+1}(1 + \frac{z}{s^2})y_{k,i-1} + \frac{1-i}{i+1}y_{k,i-2} \\
&= \frac{2i}{i+1}(1 + \frac{z}{s^2})\sum_{j=0}^{i-1}\gamma_j^{i-1}T_j(1 + \frac{z}{s^2}) + \frac{1-i}{i+1}\sum_{j=0}^{i-2}\gamma_j^{i-2}T_j(1 + \frac{z}{s^2}) \\
&= \frac{2i}{i+1}\gamma_0^{i-1}T_1(1 + \frac{z}{s^2}) + \sum_{j=2}^{i}\frac{i}{i+1}\gamma_{j-1}^{i-1}T_j(1 + \frac{z}{s^2}) \\
&\quad + \sum_{j=2}^{i}\left(\frac{i}{i+1}\gamma_{j-1}^{i-1} - \frac{i-1}{i+1}\gamma_{j-2}^{i-2}\right)T_{j-2}(1 + \frac{z}{s^2})
\end{aligned}
$$

where we used (14). By comparison with (38), we obtain the following recurrence

$$
\begin{aligned}
&\gamma_j^i = 0 \; \forall \; j > i, \qquad i = 0, \ldots, s - 1, \\
&\gamma_0^0 = 1, \quad \gamma_0^1 = 0, \quad \gamma_1^1 = 1, \quad \gamma_0^i = \frac{i}{i+1}\gamma_1^{i-1} - \frac{i-1}{i+1}\gamma_0^{i-2} \qquad i = 2, \ldots, s - 1, \\
&\gamma_1^i = \frac{2i}{i+1}\gamma_0^{i-1} + \frac{i}{i+1}\gamma_2^{i-1} - \frac{i-1}{i+1}\gamma_1^{i-2} \qquad i = 2, \ldots, s - 1, \\
&\gamma_j^i = \frac{i}{i+1}\gamma_{j-1}^{i-1} + \frac{i}{i+1}\gamma_{j+1}^{i-1} - \frac{i-1}{i+1}\gamma_j^{i-2} \qquad i = 2, \ldots, s - 1, \; j = 2, \ldots, i.
\end{aligned}
$$

Now defining $\tilde{\gamma}_j^i = (i+1)\gamma_j^i$, the above induction relations simplify to (34) and (35). The positivity of $\tilde{\gamma}_j^i$, and hence of γ_j^i, follows from Lemma 3.7. For $i = s$, the stability is a consequence of Theorem 3.2.

Analogously, the RKC method reads for $\eta = 0$,

$$
\begin{aligned}
&y_{k0} = 1, \quad y_{k1} = y_{k0} + \frac{3h\lambda}{s^2 - 1}y_{k0}, \\
&y_{ki} = \frac{6ih\lambda}{(i+1)(s^2 - 1)}y_{k,i-1} + \frac{2i}{i+1}y_{k,i-1} + \frac{1-i}{i+1}y_{k,i-2}, \quad i = 2, \ldots, s - 1, \\
&y_{ks} = \frac{h\lambda}{s}y_{k,s-1} + \frac{s^2 - 1}{3s}y_{k,s-1} - \frac{s^3 - s^2 - s + 1}{3s^2}y_{k,s-2} + \frac{2s^2 + 1}{3s^2}y_0, \\
&y_1 = y_{ks},
\end{aligned} \tag{40}
$$

and we follow the same methodology as above. Note that for RKC we have $|T_j(1 + \frac{3}{s^2-1}z)| \leq 1$ for all $j = 0, \ldots, s$ and $z \in [-\beta(s, 0), 0] = [-\frac{2}{3}(s^2 - 1), 0]$. Using the same notations we search for

coefficients satisfying the following

$$y_{ki} = \sum_{j=0}^{i} \gamma_j^i T_j(1 + \frac{3}{s^2-1}z), \text{ where } \sum_{j=0}^{i} \gamma_j^i = 1 \text{ and } \gamma_j^i \geq 0 \ \forall \ i,j = 1,\ldots s-1. \qquad (41)$$

Remark that $R_{s,0}(z) = y_{k0} = y_k = 1 = T_0(1 + \frac{3}{s^2-1}z)$, hence $\gamma_0^0 = 1$. Analogously, $R_{s,1}(z) = y_{k1} = y_{k0} + \frac{3h\lambda}{s^2-1}y_{k0} = 1 + \frac{3z}{s^2-1} = T_1(1 + \frac{3}{s^2-1}z)$, and we deduce that $\gamma_0^1 = 0$, $\gamma_1^1 = 1$.
Again we find a relation between these new coefficients to prove their positivity using $y_{ki} = \frac{2i}{i+1}(1 + \frac{3}{s^2-1}z)y_{k,i-1} + \frac{1-i}{i+1}y_{k,i-2}$. We get a recurrence of the same form as in the Chebyshev double adjoint method (37) but with different parameter, proceeding in the same we obtain exactly the same coefficients γ_j^i, which concludes the proof. $\qquad \square$

Remark 3.8. *For the case of positive damping $\eta > 0$, the coefficients get very complicated and it is difficult to find a recurrence relation between them in order to prove their positivity. However, observing that all the coefficients in the recurrence relations of the internal stages of the methods are continuous functions of η, then for all s, there exists $\eta_0(s)$ such that the internal stages are stable (bounded) for all $\eta \in [0, \eta_0(s)]$. Numerical investigations suggest that Theorem 3.6 remains valid for all $\eta > 0$ i.e the methods remain stable with the stability functions of the internal stages bounded by 1, for all integers $s \geq 1$ for Chebyshev and $s \geq 2$ for RKC, and all $\eta > 0$. We have verified this numerically for $s \leq 200$.*

We conclude this section by the following convergence theorem for the new explicit stabilized methods for stiff optimal control problems.

Theorem 3.9. *The method (25)-(26) (resp. (31)-(32)) has order 1 (resp. 2) of accuracy for the optimal control problem (1).*

Proof. The proof follows immediately from Theorem 2.1 with $p_{oc} = p_{ode} = 2$ for the RKC method. $\qquad \square$

Remark 3.10. *The proposed explicit stabilized integrators for optimal control problems could be combined with the idea of implicit-explicit (IMEX) integrators as proposed in [14], where RKC type methods would replace the implicit part in the IMEX integrator. This idea is already proposed in [28, 7] in the context of advection-diffusion-reaction problems. In [28], the diffusion part is discretized with an RKC method which typically has a large number of internal stages, and the advection-reaction part is integrated using a 4-stage explicit Runge-Kutta method. In [7], the method integrates the diffusion term using ROCK2 method, the advection term using a 3-stage explicit method, and the nonlinear reaction term is solved implicitly. Such an extension is however out of the scope of the paper.*

4 Numerical experiments

In this Section, we illustrate numerically our theoretical findings of convergence and stability of the new fully explicit methods for stiff optimal control problems, first on a stiff three dimensional problem and second on a nonlinear diffusion advection PDE (Burgers equation).
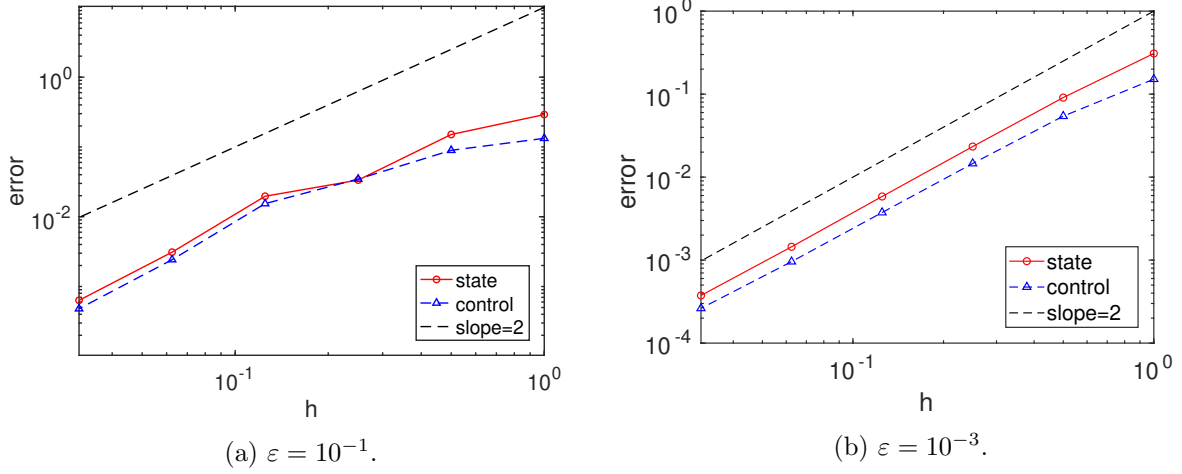
(a) $\varepsilon = 10^{-1}$.   (b) $\varepsilon = 10^{-3}$.

Figure 5: Convergence plot of RKC (31)-(32) applied to problem (43).

4.1   A linear quadratic stiff test problem

We start this section by a simple test problem taken from [11]:

$$\min \ \frac{1}{2} \int_0^1 (u^2(t) + 2x^2(t)) dt \text{ subject to}$$
$$\dot{x}(t) = \frac{1}{2} x(t) + u(t), \ t \in [0,1], \ x(0) = 1. \tag{42}$$

The optimal solution (u^*, x^*) is given by $u^*(t) = \frac{2(e^{3t} - e^3)}{e^{3t/2}(2+e^3)}$, $x^*(t) = \frac{2e^{3t} + e^3}{e^{3t/2}(2+e^3)}$. As studied in [14] we modify problem (42) into a singularly perturbed (stiff) problem to illustrate the good stability properties of our new method. For a fixed $\varepsilon > 0$, we consider the following stiff optimal control problem,

$$\min \ c(1) \text{ subject to}$$
$$\dot{c}(t) = \frac{1}{2}(u^2(t) + x^2(t) + 4z^2(t)), \ c(0) = 0,$$
$$\dot{x}(t) = z(t) + u(t), \ x(0) = 1, \tag{43}$$
$$\dot{z}(t) = \frac{1}{\varepsilon}\left(\frac{1}{2} x(t) - z(t)\right), \ z(0) = \frac{1}{2},$$

Figure 5 shows the convergence behavior, using the new RKC method (31)-(32), of the error in infinity norm between the solutions of the stiff problem (43) for $\varepsilon = 10^{-1}$ and $\varepsilon = 10^{-3}$ and different sizes of the time step $h_i = 2^{-i}, i = 0, \dots, 5$ and the reference solution is obtained with $h = 2^{-7}$. We observe lines of slope 2 which confirms the theoretical order two of accuracy of the scheme (Theorem 3.9). In the stiff case ($\varepsilon = 10^{-3}$), the method uses $s = 4$ to calculate the reference solution and $s = 40, 28, 20, 14, 10, 7$ respectively for the different time steps used to illustrate the convergence, these values coincide with the theoretical values that can be obtained using (18). Analogously to the case of stiff ODEs, the cost of scheme (31)-(32) is $\mathcal{O}(\varepsilon^{-\frac{1}{2}})$ function evaluations of f, while using Euler method with its double adjoint would cost $\mathcal{O}(\varepsilon^{-1})$.

4.2 Optimal control of Burgers equation

To illustrate the performance of the new method, we consider the following optimal control problem of a nonlinear diffusion advection PDE corresponding to the Burgers equation

$$\min_{u \in L^2([0,T];L^2(\Omega))} J(u) = \frac{1}{2}\|y(T) - y^{target}\|^2_{L^2(\Omega)} + \frac{\alpha}{2}\int_0^T \|u(t)\|^2_{L^2(\Omega)}$$

subject to

$$\partial_t y(t,x) = \mu\Delta y(t,x) - \frac{\nu}{2}\partial_x(y^2(t,x)) + u(t,x) \quad \text{in } (0,T) \times \Omega,$$

$$y(0,x) = g(x) \quad \text{in } \Omega,$$

$$y(t,x) = 0 \quad \text{on } \partial\Omega,$$

(44)

where $\mu, \nu > 0$, in dimension $d = 1$ with domain $\Omega = (0,1)$ and the final time is given by $T = 2.5$. Here the control u is a part of the source that we want to adjust in order to achieve a given final state $y^{target} : \Omega \to \mathbb{R}$.

We use a standard central finite difference space discretization for the state equation, and the trapezoid rule to discretize in space the norm $L^2(\Omega)$. We consider $M + 2$ points in space $x_m = m\Delta x$, with grid mesh size $\Delta x = \frac{1}{M+1}$, and we denote by $y_m(t)$ the approximation to $y(t,x_m)$, and define the vector $Y(t) = (y_0(t), y_1(t), \ldots, y_{M+1}(t)) \in \mathbb{R}^{M+2}$. Similar notations are used for U and P. We obtain the following optimal control problem semi discretized in space,

$$\min_{U(t) \in \mathbb{R}^{M+2}} \Psi(c(T), Y(T)) = \frac{1}{2(M+1)} \sum_{m=0}^{M+1} {}' (y_m(T) - y^{target}(x_m))^2 + \alpha c(T)$$

subject to

$$\dot{c}(t) = \frac{1}{2(M+1)} \sum_{m=0}^{M+1} {}' u_m^2(t), \quad c(0) = 0,$$

(45)

$$\dot{y}_m(t) = F_m(U(t), Y(t)) := \frac{\mu}{\Delta x^2}(y_{m+1} - 2y_m + y_{m-1})$$

$$- \frac{\nu}{4\Delta x^2}(y_{m+1}^2 - y_{m-1}^2) + u_m,$$

$$y_m(0) = g(x_m), \quad m = 0, \ldots, M+1,$$

where $m = 0, \ldots M+1$ and the primed sum denotes a normal sum where the first and the last term are divided by 2 and we define $y_0 = y_{M+1} = 0$ to take into account the homogeneous Dirichlet boundary conditions. The function $F : \mathbb{R}^{M+2} \to \mathbb{R}^{M+2}$ with components $F_m : \mathbb{R}^{M+2} \to \mathbb{R}$ is obtained from the standard central finite difference discretization of the right hand side of the state equation (44), and adapted to the boundary conditions. The corresponding adjoint system is

$$\dot{p}_c(t) = 0, \quad p_c(T) = \nabla_c \Psi = \alpha,$$

$$\dot{P}(t) = -\nabla_Y F(U(t), Y(t))P,$$

(46)

where P is a vector of length $M + 2$ containing the costate values p_m, $m = 0, \ldots, M + 1$. In all our experiments we take $\mu = 0.1$, $\nu = 0.02$, $g(x) = \frac{3}{2}x(1-x)^2$, and $y^{target}(x) = \frac{1}{2}\sin(10x)(1-x)$.

In Figure 6 we plot the optimal control function (Fig. 6b) and the corresponding state function (Fig. 6a) obtained using scheme (31)-(32). When we use a small value for α in the model, we allow larger control values and thus a final state very close to the target (Fig. 6c), otherwise the control will be more limited and then the final state will not be that close to the target (Fig. 6d).

(a) State.

(b) Optimal control

(c) Final and target states for $\alpha = 0.01$.

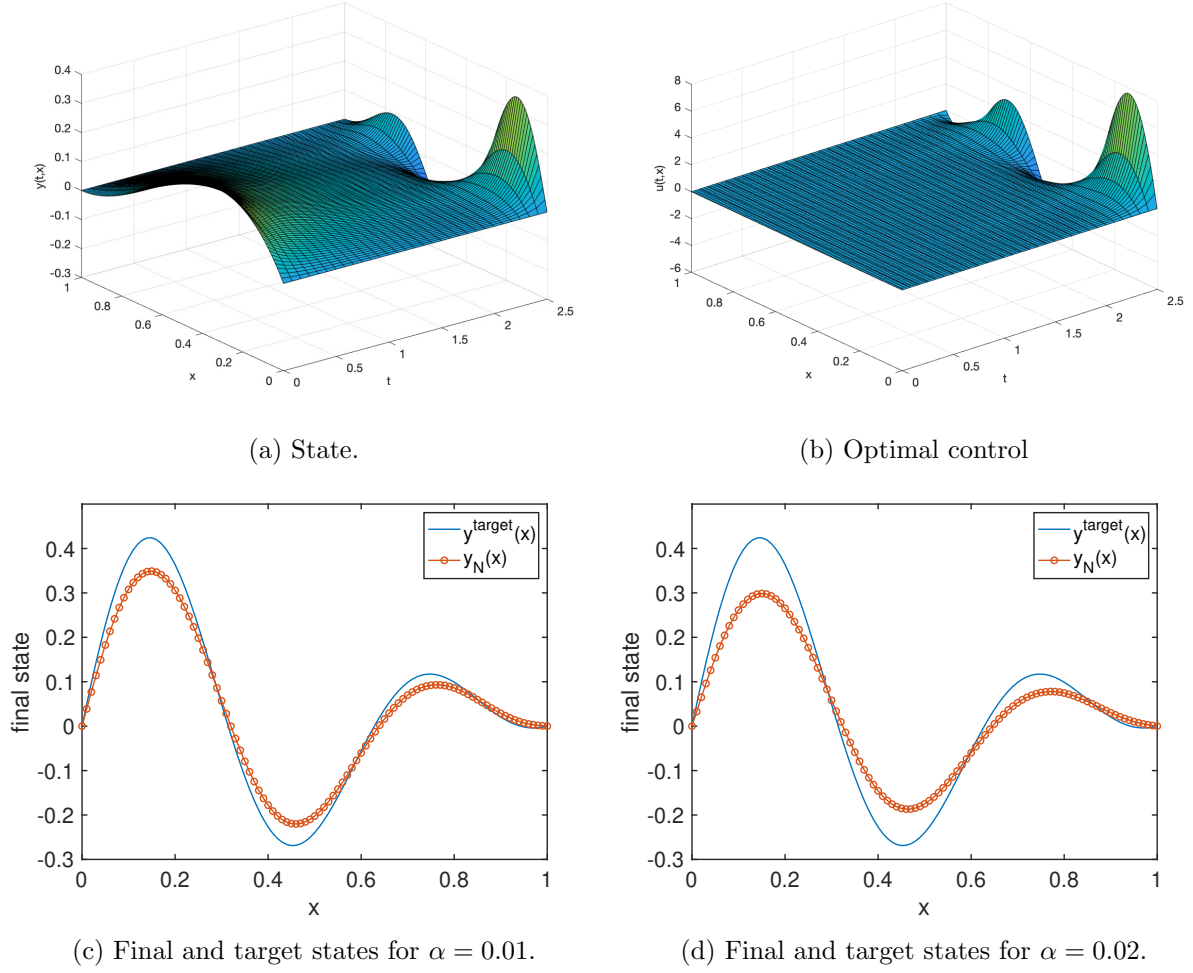(d) Final and target states for $\alpha = 0.02$.

Figure 6: State, final state, and control, of problem (44). Figures (a), (b), and (c) are obtained using $\Delta x = 1/100$, $\Delta t = T/30$, and $s = 24$ stages, and $\alpha = 0.01$. Figure (d) uses the same Δx, g and y^{target} but $\alpha = 0.02$.

Note that the method required $s = 24$ stages for $\Delta x = 1/100$ and $\Delta t = 2.5/30$, while using an Euler method with its double adjoint would require $\Delta t \leq \Delta t_{max,Euler} := \Delta x^2/2$ at most. Hence, the standard Euler method would be $\Delta t/(s\Delta t_{max,Euler}) \simeq 70$ times more expensive in terms of number of function evaluations for $\Delta x = 1/100$, a factor that grows arbitrarily as $\Delta x \to 0$.

In Figure 7, we plot the convergence curves for the state and control functions of the new RKC method (31)-(32) applied to the diffusion problem discretized in space (45), where the number of stages s is computed adaptively using (18). We recover again lines of slope two, which corroborate the order two of the method. Although our convergence analysis is valid only in finite dimensions (Theorem 3.9), this suggests that the convergence of order two persists in the PDE case. For comparison, we also included the results for the following standard diagonally implicit Runge-Kutta method of order two, inspired from [14, Table 5.1] in the context of stiff optimal control problems, and given by the following Butcher tableau where $\gamma = 1 - \sqrt{2}/2$ (making the method L-stable),

$$
\begin{array}{c|cc}
\gamma & \gamma & \\
1 - 2\gamma & \gamma \\
\hline
& 1/2 & 1/2
\end{array}
\tag{47}
$$
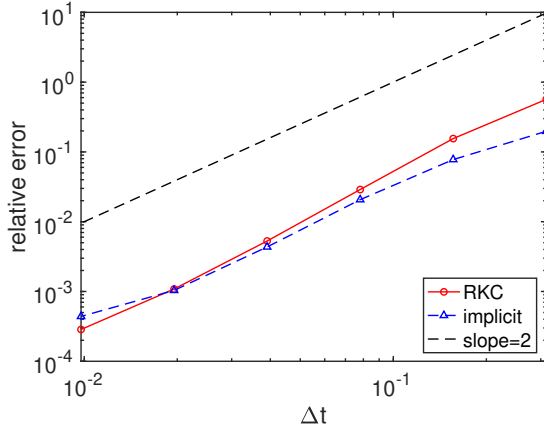
(a) Error in the state.
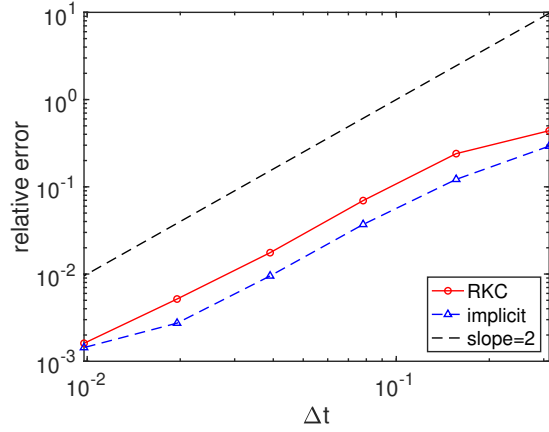
(b) Error in the control.

Figure 7: Convergence plot of the RKC method (31)-(32) and the implicit method (47) applied to problem (45) for many time steps $\Delta t_i = T/2^i$, $i = 3, \ldots, 8$, $\Delta x = 1/100$, and $\alpha = 0.02$, The reference solution is obtained using $\Delta t = T/2^{12}$, $s = 3$.

Although for a fixed timestep, the second order implicit method (47) appears about two times more accurate than the RKC method (31)-(32) for the control and almost of the same accuracy for the state, we emphasize that these convergence plots do not take into account the extra cost of the implicitness of method (47). Indeed, the cost and difficulty of the implementation of the implicit methods (nonlinear iterations, preconditioners, etc.) would typically deteriorate in larger dimensions and for a nonlinear diffusion operator, as it is already the case for initial value PDEs [2], while as an explicit stabilized scheme, the RKC method (31)-(32) can be conveniently implemented in the spirit of the simplest explicit Euler method.

References

[1] A. Abdulle. Fourth order Chebyshev methods with recurrence relation. *SIAM J. Sci. Comput.*, 23(6):2041–2054, 2002.

[2] A. Abdulle. *Explicit Stabilized Runge–Kutta Methods*, pages 460–468. Encyclopedia of Applied and Computational Mathematics, Springer Berlin Heidelberg, 2015.

[3] A. Abdulle, I. Almuslimani, and G. Vilmart. Optimal explicit stabilized integrator of weak order 1 for stiff and ergodic stochastic differential equations. *SIAM/ASA J. Uncertain. Quantif.*, 6(2):937–964, 2018.

[4] A. Abdulle and S. Cirilli. S-ROCK: Chebyshev methods for stiff stochastic differential equations. *SIAM J. Sci. Comput.*, 30(2):997–1014, 2008.

[5] A. Abdulle and T. Li. S-ROCK methods for stiff Ito SDEs. *Commun. Math. Sci.*, 6(4):845–868, 2008.

[6] A. Abdulle and A. Medovikov. Second order Chebyshev methods based on orthogonal polynomials. *Numer. Math.*, 90(1):1–18, 2001.

[7] A. Abdulle and G. Vilmart. PIROCK: a swiss-knife partitioned implicit-explicit orthogonal Runge-Kutta Chebyshev integrator for stiff diffusion-advection-reaction problems with or without noise. *J. Comput. Phys.*, 242:869–888, 2013.

[8] G. Albi, M. Herty, and L. Pareschi. Linear multistep methods for optimal control problems and applications to hyperbolic relaxation systems. *Appl. Math. Comput.*, 354:460–477, 2019.

[9] M. Bakker. Analytical aspects of a minimax problem. 1971. Technical Note TN 62 (in Dutch), Mathematical centre, Amsterdam.

[10] J. F. Bonnans and J. Laurent-Varin. Computation of order conditions for symplectic partitioned Runge-Kutta schemes with application to optimal control. *Numer. Math.*, 103(1):1–10, 2006.

[11] W. W. Hager. Runge-Kutta methods in optimal control and the transformed adjoint system. *Numer. Math.*, 87(2):247–282, 2000.

[12] E. Hairer, C. Lubich, and G. Wanner. *Geometric numerical integration*, volume 31 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, second edition, 2006. Structure-preserving algorithms for ordinary differential equations.

[13] E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems*. Springer Series in Computational Mathematics 14. Springer-Verlag, Berlin, 2 edition, 1996.

[14] M. Herty, L. Pareschi, and S. Steffensen. Implicit-explicit Runge-Kutta schemes for numerical discretization of optimal control problems. *SIAM J. Numer. Anal.*, 51(4):1875–1899, 2013.

[15] M. Herty and V. Schleper. Time discretizations for numerical optimisation of hyperbolic problems. *Appl. Math. Comput.*, 218(1):183–194, 2011.

[16] W. Hundsdorfer and J. Verwer. *Numerical solution of time-dependent advection-diffusion-reaction equations*, volume 33 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 2003.

[17] C. Y. Kaya. Inexact restoration for Runge-Kutta discretization of optimal control problems. *SIAM J. Numer. Anal.*, 48(4):1492–1517, 2010.

[18] J. Lang and J. G. Verwer. W-methods in optimal control. *Numer. Math.*, 124(2):337–360, 2013.

[19] Q. Li, L. Chen, C. Tai, and W. E. Maximum principle based algorithms for deep learning. *Journal of Machine Learning Research*, 18(165):1–29, 2018.

[20] X. Liu and J. Frank. Symplectic Runge-Kutta discretization of a regularized forward-backward sweep iteration for optimal control problems. *Submitted*, arXiv: 1912.07028v1, 2019.

[21] Y. Maday, M.-K. Riahi, and J. Salomon. Parareal in time intermediate targets methods for optimal control problems. In *Control and optimization with PDE constraints*, volume 164 of *Internat. Ser. Numer. Math.*, pages 79–92. Birkhäuser/Springer Basel AG, Basel, 2013.

[22] J. M. Sanz-Serna. Symplectic Runge-Kutta schemes for adjoint equations, automatic differentiation, optimal control, and more. *SIAM Rev.*, 58(1):3–33, 2016.

[23] B. Sommeijer, L. Shampine, and J. Verwer. RKC: an explicit solver for parabolic PDEs. *J. Comput. Appl. Math.*, 88:316–326, 1998.

[24] B. P. Sommeijer and J. G. Verwer. *A performance evaluation of a class of Runge-Kutta-Chebyshev methods for solving semidiscrete parabolic differential equations.* Afdeling Numerieke Wiskunde [Department of Numerical Mathematics], 91. Mathematisch Centrum, Amsterdam, 1980.

[25] P. J. van der Houwen and B. P. Sommeijer. On the internal stability of explicit, m-stage Runge-Kutta methods for large m-values. *Z. Angew. Math. Mech.*, 60(10):479–485, 1980.

[26] J. Verwer. Explicit Runge-Kutta methods for parabolic partial differential equations. *Special issue of Appl. Num. Math.*, 22:359–379, 1996.

[27] A. Walther. Automatic differentiation of explicit Runge-Kutta methods for optimal control. *Comput. Optim. Appl.*, 36(1):83–108, 2007.

[28] C. J. Zbinden. Partitioned Runge-Kutta-Chebyshev methods for diffusion-advection-reaction problems. *SIAM J. Sci. Comput.*, 33(4):1707–1725, 2011.