Parallel-in-Time for Parabolic Optimal Control Problems Using PFASST

Sebastian Götschel and Michael L. Minion

Abstract In gradient-based methods for parabolic optimal control problems, it is necessary to solve both the state equation and a backward-in-time adjoint equation in each iteration of the optimization method. In order to facilitate fully parallel gradient-type and nonlinear conjugate gradient methods for the solution of such optimal control problems, we discuss the application of the parallel-in-time method PFASST to adjoint gradient computation. In addition to enabling time parallelism, PFASST provides high flexibility for handling nonlinear equations, as well as potential extra computational savings from reusing previous solutions in the optimization loop. The approach is demonstrated here for a model reaction-diffusion optimal control problem.

1 Introduction

Gradient-based methods for parabolic optimal control problems are computationally expensive due to the need to solve both a forward state equation and a backward-in-time adjoint equation to compute gradient information in each iteration of the optimization procedure. One potential way to reduce the overall computational time is to employ parallel-in-time (PinT) methods for solving state and adjoint equations. Attempts to construct PinT methods for the solution of differential equations date back more than 50 years and have gained increasing interest in the last 15 years [8]. More recently, the application of space-time parallel methods to the solution of optimization problems governed by PDEs has become an active research area, with approaches including multiple shooting (e.g. [11] and the references therein), Schwarz

Sebastian Götschel
Zuse Institute Berlin, Takustr. 7, 14195 Berlin, Germany, e-mail: goetschel@zib.de

Michael Minion
Lawrence Berkeley National Laboratory, 1 Cyclotron Rd., Berkeley, CA, 94720, USA, e-mail: mlminion@lbl.gov

methods [1, 9], the application of parareal preconditioners [14, 18], and space-time parallel multigrid methods [10].

Here we apply the PFASST method [7] ("Parallel Full Approximation Scheme in Space and Time") to both the state and adjoint equation to provide a fully time-parallel gradient- or nonlinear conjugate gradient method. This approach is somewhat related to the time-parallel gradient type method presented in [5]. There the time interval of interest is subdivided into time steps, which are solved in parallel using quantities from the previous optimization iteration as input. This leads to jumps in the solutions of state and adjoint equation such that these equations are not satisfied during optimization. While convergence is demonstrated in [5] if sufficiently small step sizes for updating the control are used, it is unclear how to automatically select such a step size. In our approach, the usual line search criteria, e.g. the (strong) Wolfe conditions, can be used to guarantee convergence.

2 Background

2.1 SDC, MLSDC, and PFASST

A distinguishing factor of the PFASST algorithm compared to other PinT methods is that, in each iteration, the solution on a given time step is improved using a deferred correction approach rather than being computed in full using a given ODE method. The correction sweeps are based on spectral deferred corrections (SDC) [6] and are applied on a hierarchy of space-time representations of the problem as in multi-level SDC (MLSDC) methods [17]. PFASST exposes parallelism in the time direction because MLSDC iterations are pipelined so that SDC sweeps are done concurrently on all but the coarsest level.

One advantage of SDC methods is the flexibility in choosing the type of substepping for the correction sweep. In the numerical example, we will use both a semi-implicit or IMEX approach [15] (wherein one component of the solution is treated explicitly and one implicitly) and a multi-implicit (MISDC) approach [2] (wherein one component of the solution is treated explicitly and two components implicitly but uncoupled). The motivation for using IMEX and MISDC variants are to replace the solution of coupled nonlinear systems in the time stepping by simpler linear equations (see Sect. 4.1 for further discussion).

Finally, PFASST is an iterative method, and the typical way in which the solution is initialized on each parallel time slice is by serial time stepping on the coarsest level. In optimal control problems, an alternative is to use the solution from the previous optimization iteration as the initial guess for the next state and adjoint equation solve. As the optimization procedure converges, the initial solutions improve in quality, and hence the number of PFASST iterations needed for convergence should decrease. We demonstrate this savings in Sect. 4.

2.2 Optimal Control of Parabolic PDEs

We consider optimal controls problems

$$\min_{y\in Y, u\in U} J(y,u) \text{ subject to } c(y,u)=0, \tag{1}$$

with $c: Y \times U \to Z^\star$ a semi-linear parabolic PDE on Banach spaces Y, Z and Hilbert space U over a spatial domain $\Omega \subset \mathbb{R}^d$. We assume that there exists a unique solution $y = y(u) \in Y$ of the state equation $c(y,u) = 0$ for each control $u \in U$. To avoid a full, typically 4D, discretization of this problem, methods working on the reduced functional

$$\min_{u\in U} j(u) := J(y(u),u) \tag{2}$$

are often employed. Under standard assumptions, the reduced gradient is given by

$$j'(u) = J_u(y(u),u) + c_u(y(u),u)^\star p(u),$$

where p solves the adjoint equation

$$J_y(y(u),u) + c_y(y(u),u)^\star p(u) = 0, \tag{3}$$

which is backward in time, see, e.g., [13] for details. Due to the occurrence of $-J_y(y(u),u)$ as a source term, and—in the nonlinear case—the dependence of $c_y(y(u),u)$ on the state solution $y(u)$, the adjoint gradient computation consists of three steps:

1. solve $c(y,u) = 0$ for $y \in Y$ and store the solution trajectory,
2. solve $c_y(y,u)^\star p = -J_y(y,u)$ for $p \in Z$,
3. set $j'(u) = J_u(y,u) + c_u(y,u)^\star p$.

In order to facilitate fully parallel algorithms to solve the optimal control problem (1), state and adjoint equations need to be solved using PinT methods.

3 PFASST for Optimal Control

Minimizing the objective function (2) is done via gradient- or nonlinear conjugate gradient (ncg) methods

$$\begin{aligned} u_{k+1} &= u_k + \alpha_k d_k \\ d_{k+1} &= -j'(u_{k+1}) + \beta_k d_k, \end{aligned}$$

where $d_0 = -j'(u_0)$, α_k denotes the step size, required to satisfy the (strong) Wolfe conditions [16], and the choice of β_k defines the type of method ($\beta_k = 0$ for the gradient method; various possibilities for β_k leading to different ncg methods, see [4] for a brief overview and the method used in the experiments). For the numerical

solution we apply a method of lines approach, discretizing first in space, then in time.

Parallelization in time for these methods requires three ingredients: time-parallel computation of inner products, step size selection, and the solution of state and adjoint equations. The first two ingredients are straightforward: on each time interval, local scalar products are computed, and then communicated to all other processors, summing them up. These scalar products are used to compute β_k, as well as to check sufficient decrease and curvature conditions during step size selection. For the time-parallel solution of state and adjoint equations we propose two different strategies using PFASST. In the first approach, the state and adjoint problems are solved separately using PFASST for both. The state solution at each time step and quadrature node is stored for subsequent use in the solution of the adjoint equation. Alternatively, PFASST could be used to solve the state and adjoint equation at the same time. Each SDC sweep of the state equation would be followed by a backward sweep of the adjoint equation on the same nodes, leading to more complicated communication patterns. In the numerical example, we focus on the first approach. Details and results for the second approach will be reported elsewhere.

4 Numerical Example

Here we consider the following optimal control problem ([3, 12]) governed by a semi-linear reaction-diffusion equation on $\Omega = (0,20)$:

$$\min_{y,u} \frac{1}{2}\int_0^T \int_\Omega (y-y_d)^2 \, dx\, dt + \frac{\lambda}{2}\int_0^T \int_\Omega u^2 \, dx\, dt,$$

where $T = 5$, and $y(x,t)$ is subject to

$$\begin{aligned} y_t - y_{xx} + y(\frac{1}{3}y^2 - 1) &= u(x,t) \quad \text{in } \Omega \times (0,T) \\ y(x,0) &= y_0(x) \quad \text{in } \Omega, \end{aligned} \tag{4}$$

with homogeneous Neumann boundary conditions. The initial condition and desired state are

$$y_0(x) = \begin{cases} 1.2\sqrt{3}, & x \in [9,11] \\ 0, & \text{elsewhere} \end{cases} \quad \text{and} \quad y_d(x,t) = \begin{cases} y_{\text{nat}}(x,t), & t \in [0,2.5] \\ y_{\text{nat}}(x,2.5), & t \in (2.5,T], \end{cases}$$

where y_{nat} denotes the solution to the PDE (4) for $u \equiv 0$. For $\lambda = 0$, an exact optimal control is known:

$$u_{\text{exact}} = \begin{cases} 0, & t \le 2.5 \\ \frac{1}{3}y_{\text{nat}}^3(x,2.5) - y_{\text{nat}}(x,2.5) - \frac{\partial^2}{\partial x^2} y_{\text{nat}}(x,2.5), & t > 2.5. \end{cases}$$

4.1 IMEX and MISDC Formulations

As mentioned in Sect. 2.1, there is great flexibility in how the substepping procedure in SDC is constructed since it need only be a first-order approximation. For our numerical example, two strategies are investigated, IMEX SDC and multi-implicit SDC. In both cases, the diffusion term in (4) is treated implicitly to avoid the severe time-step restriction inherent in explicit temporal methods. In the IMEX strategy, only the diffusion term is treated implicitly, while in the MISDC method, both diffusion and reaction are treated implicitly, but the implicit solutions are done independently as in operator splitting methods. In addition, we employ a lagging of nonlinear terms in MISDC iteration to turn the nonlinear solve into a linear problem.

Methods that employ operator splitting are desirable when the reduced cost of split implicit solvers compared to coupled solvers is significant. The overall accuracy of PFASST (assuming convergence of SDC iterations) does not depend on the form of the substepping, rather on the choice of number and type of integration nodes. Hence, the main concern in terms of efficiency is the computational cost of each SDC iteration and the number of iterations required for convergence.

The IMEX and MISDC approach are explained by examining a single substep of an SDC sweep. Letting k denote the SDC iteration, m the substep index, and D^2 the discretization of the second derivative term, then the correction equation for a single fully implicit, backward-Euler type discretization of the substep for (4) will take the form

$$y_{m+1}^{[k+1]} = y_m^{[k+1]} + \Delta t_m (D^2 y_{m+1}^{[k+1]} - y_{m+1}^{[k+1]} (\frac{1}{3}(y_{m+1}^{[k+1]})^2 - 1)) + S_j^{[k]}, \tag{5}$$

where the term $S_j^{[k]}$ contains terms that either depend on the previous iteration $[k]$ or values at iteration $[k+1]$ already computed at substep $j < m+1$, including the control terms arising from the discretization of $u(x,t)$. Note that the implicit equation couples nonlinear reaction and diffusion terms and hence would require a global nonlinear solver in each substep. For problems in which the reaction terms are nonstiff and can be treated explicitly, the reaction terms at node $m+1$ do not appear in the implicit equation, giving the form

$$y_{m+1}^{[k+1]} = y_m^{[k+1]} + \Delta t_m (D^2 y_{m+1}^{[k+1]} - y_m^{[k+1]} (\frac{1}{3}(y_m^{[k+1]})^2 - 1)) + S_j^{[k]}. \tag{6}$$

Each substep now requires only the solution of a linear implicit equation, and hence is computationally cheaper than a fully implicit approach, assuming that the explicit treatment of the reaction term does not impose an additional time step restriction.

When the reaction term is stiff, and hence it is advantageous to treat it implicitly, a standard MISDC approach applies an operator splitting between diffusion and reaction in the correction equation. For example,

$$y^* = y_m^{[k+1]} + \Delta t_m D^2 y^* + S_j^{*,[k]}, \tag{7}$$

$$y_{m+1}^{[k+1]} = y_m^{[k+1]} + \Delta t_m (D^2 y^* - y_{m+1}^{[k+1]} (\frac{1}{3}(y_{m+1}^{[k+1]})^2 - 1)) + S_j^{[k]}. \tag{8}$$

For the numerical methods here, the MISDC approach is further modified, so that the nonlinear solve for reaction in (8) is made linear by lagging terms in $[k]$:

$$y_{m+1}^{[k+1]} = y_m^{[k+1]} + \Delta t_m (D^2 y^* - y_{m+1}^{[k+1]} (\frac{1}{3}(y^*)^2 - 1)) + S_j^{[k]}. \tag{9}$$

This form creates an implicit solve with roughly the same cost as treating reaction explicitly but is more stable. In all the numerical examples presented here, a DIRK type approach [19] is used so that the generic form of $S_j^{[k]}$ contains both the usual SDC terms from iteration $[k]$ and a linear combination of previously computed right-hand-side terms from SDC iteration $[k+1]$.

4.2 Results

In this section we show the results using IMEX and MISDC approaches to solve the state and adjoint equation. In both cases, a method of lines is employed by using a spectral discretization in space with spatial derivatives computed with the fast Fourier transform. The PFASST iterations are stopped when the relative or absolute residual falls below 10^{-11}. For solving the optimization problem we set $\lambda = 10^{-6}$ and use the ncg method from [4], with initial control $u_0 = 0.5 u_{\text{exact}}$. As described in [3], the ncg method converges quite slowly for this particular problem; it was stopped after at most 200 iterations.

IMEX. Since the reaction terms in our example are not highly stiff, an IMEX approach can be used for the state and adjoint equations. PFASST is employed using three levels (32/64/128 spatial points and 3/5/9 LobattoIIIA nodes in time) with 20 parallel time intervals. Note the temporal method is formally 16th order. Running on 20 processors in parallel, the final objective function value after 200 ncg iterations is $2.4 \cdot 10^{-3}$, and the computed control has a relative L^2-error of 0.15 compared to u_{exact}. In contrast, the sequential version stops with a slightly worse objective function value of $3.2 \cdot 10^{-3}$, and a relative L^2-error of 0.15 in the computed control. A plot of the computed control, the error in the computed control, and the corresponding computed optimal state can be found in Fig. 1. By parallel execution, the overall runtime was reduced by a factor 3.8, yielding a parallel efficiency of 19%.

MISDC. For testing MISDC we used 20 parallel time intervals with two PFASST levels consisting of 64/128 spatial points and 5/9 LobattoIIIA nodes. After 200 ncg iterations, the sequential version reached an objective function value of $3.8 \cdot 10^{-3}$ and a relative L^2-error of 0.15 in the computed control. Running in parallel reduced the computation time by a rather small factor 2, but lead to improved results (objective: $1.8 \cdot 10^{-3}$, control: relative L^2-error 0.14). Initializing the state solution at the collocation nodes in optimization iteration k with their values from iteration $k-1$ ("warm start") reduced the required sweeps by 48% while reaching an objective function value of $1.5 \cdot 10^{-3}$ and relative L^2-error in the control of 0.13. The reduc-

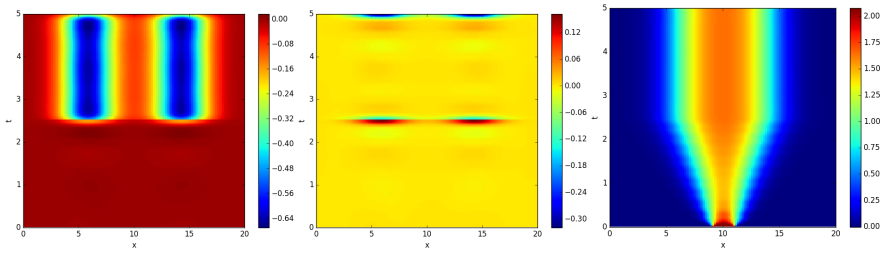

Fig. 1 Computed optimal control with $\lambda = 10^{-6}$ (left), difference to exact control for $\lambda = 0$ (middle) and optimal state (right) using IMEX.

tion in sweeps translates to a significant reduction of overall computation time by 45%. This is in contrast to the IMEX experiment, where for a reduction in sweeps by 39% the gain in overall speed was a mere 7%. Lagged linearization as in (9) increases the total number of PFASST sweeps for a state equation solve from 570 (on average 28.5 per time step) to 642 (avg. 32.1/time step). Using smaller time steps (40 parallel intervals), the average number of iterations is 28.3 in both cases.

For this example, it is unreasonable to attempt to compare the IMEX and MISDC approaches in terms of overall efficiency since MISDC is designed for problems where both diffusion and reaction components are stiff. The pertinent point here is that employing the MISDC procedure with a lagged linearization of reaction terms does not appear to increase the number of PFASST iterations needed for convergence substantially, thus offering the possibility of greatly reducing the cost of implicit substepping compared to fully implicit methods.

5 Discussion

An approach using PFASST for the time-parallel solution of PDE-constrained optimization problems has been presented, and non-trivial parallel speedup and efficiency have been obtained. It is important to note that the parallel efficiency of PFASST is improved when solutions on coarsest levels are much cheaper than in finer levels, and spatial coarsening has a larger effect in multiple dimensions compared to the one-dimensional example used here. In addition, applying PFASST simultaneously to state and adjoint equations with proper handling of communication offers further improved parallel speedup. The flexibility of SDC/PFASST has been used to reduce the cost of implicit solutions in the substepping and also to re-use information from previous optimization iterations. Future research will, for example, deal with adaptive control of the accuracy for inexact gradient computations, and different strategies for storing or recomputing state solutions for the adjoint solve.

Acknowledgements S.G. gratefully acknowledges partial funding by the Deutsche Forschungsgemeinschaft (DFG), Project WE 2937/6-1. The work of M.M. was supported by the Applied

Mathematics Program of the DOE Office of Advanced Scientific Computing Research under the U.S. Department of Energy under contract DE-AC02-05CH11231.

References

1. Barker, A. T., Stoll, M.: Domain decomposition in time for PDE-constrained optimization. Comput. Phys. Commun. **197**, 136–143 (2015)
2. Bourlioux, A., Layton, A. T., Minion, M. L.: High-order multi-implicit spectral deferred correction methods for problems of reactive flow. J. Comput. Phys. **189**(2), 651–675 (2003)
3. Buchholz, R., Engel, H., Kammann, E., Tröltzsch, F.: On the optimal control of the Schlögl-model. Comput. Optim. Appl. **56**(1), 153–185 (2013)
4. Dai, Y. H., Yuan, Y.: A nonlinear conjugate gradient method with a strong global convergence property. SIAM J. Optim. **10**(1), 177–182 (1999)
5. Deng, X., Heinkenschloss, M.: A parallel-in-time gradient-type method for discrete time optimal control problems. Preprint, Department of Computational and Applied Mathematics, Rice University (2016) Available from http://www.caam.rice.edu/∼heinken.
6. Dutt, A., Greengard, L., Rokhlin, V.: Spectral deferred correction methods for ordinary differential equations. BIT Numer. Math. **40**(2), 241–266 (2000)
7. Emmett, M., Minion, M. L.: Toward an efficient parallel in time method for partial differential equations. Comm. App. Math. Comp. Sci. **7**, 105–132 (2012)
8. Gander, M. J.: 50 years of time parallel time integration. In: Carraro, T. et al. (eds.), Multiple Shooting and Time Domain Decomposition Methods: MuS-TDD, Heidelberg, May 6-8, 2013, pp. 69–113. Springer, Cham (2015)
9. Gander, M. J., Kwok, F.: Schwarz methods for the time-parallel solution of parabolic control problems. In: Dickopf, T. et al. (eds.), Domain Decomposition Methods in Science and Engineering XXII, pp. 207–216. Springer, Cham (2016)
10. Günther, S., Gauger, N. R., Schroder, J. B.: A Non-Intrusive Parallel-in-Time Adjoint Solver with the XBraid Library. ArXiv e-prints (2017)
11. Heinkenschloss, M.: A time-domain decomposition iterative method for the solution of distributed linear quadratic optimal control problems. J. Comput. Appl. Math. **173**(1), 169–198 (2005)
12. Herzog, R. Rösch, A., Ulbrich, S., Wollner, W.: OPTPDE: A collection of problems in PDE-constrained optimization. In: Leugering, G. et al. (eds.), Trends in PDE Constrained Optimization, pp. 539–543. Springer, Cham (2014)
13. Hinze, M., Pinnau, R., Ulbrich, M., Ulbrich, S.: Optimization with PDE constraints. Springer (2009)
14. Mathew, T., Sarkis, M., Schaerer, C. E.: Analysis of Block Parareal Preconditioners for Parabolic Optimal Control Problems. SIAM J. Sci. Comput. **32**(3), 1180–1200 (2010)
15. Minion, M. L.: Semi-implicit spectral deferred correction methods for ordinary differential equations. Commun. Math. Sci. **1**(3), 471–500 (2003)
16. Nocedal, J., Wright, S. J.: Numerical Optimization. Springer, New York (2006)
17. Speck, R., Ruprecht, D., Emmett, M., Minion, M. L., Bolten, M., Krause, R.: A multi-level spectral deferred correction method. BIT Numer. Math. **55**, 843–867 (2015)
18. Ulbrich, S.: Preconditioners based on "parareal" time-domain decomposition for time-dependent PDE-constrained optimization. In: Carraro, T. et al. (eds.), Multiple Shooting and Time Domain Decomposition Methods: MuS-TDD, Heidelberg, May 6-8, 2013, pp. 203–232. Springer, Cham (2015)
19. Weiser, M.: Faster SDC convergence on non-equidistant grids by DIRK sweeps. BIT Numer. Math. **55**, 1219–1241 (2013)