Sparse Switching Times Optimization and a Sweeping Hessian Proximal Method

Alberto De Marchi and Matthias Gerdts

Abstract The switching times optimization problem for switched dynamical systems, with fixed initial state, is considered. A nonnegative cost term for changing dynamics is introduced to induce a sparse switching structure, that is, to reduce the number of switches. To deal with such problems, an inexact Newton-type arc search proximal method, based on a parametric local quadratic model of the cost function, is proposed. Numerical investigations and comparisons on a small-scale benchmark problem are presented and discussed.

Keywords Switched dynamical systems · Switching time optimization · Sparse optimization · Cardinality · Proximal methods

MSC 2010: 90C26, 90C53, 49M27

1 Introduction

We focus on the switching times optimization (STO) problem for switched dynamical systems, which consists in computing the optimal time instants for changing the system dynamics in order to minimize a given objective function. A cost term penalizing changes of the continuous dynamics, whose sequence is given, is added to encourage a sparse switching structure. In this paper, for the sake of simplicity and without loss of generality, we consider problems with autonomous dynamical systems, cost functions in Mayer form and fixed final time. Building upon a cardinality-based formulation of the switching cost [3], in Sect. 2 an equivalent composite nonconvex, nonsmooth optimization problem is introduced, which is amenable to proximal methods [5, 6]. In Sect. 3 we propose a novel proximal arc search method, which builds upon both proximal gradient and Newton-type

A. De Marchi (✉) · M. Gerdts
Universität der Bundeswehr München, Neubiberg, Germany
e-mail: alberto.demarchi@unibw.de; matthias.gerdts@unibw.de

J. S. Neufeld et al. (eds.), *Operations Research Proceedings 2019*, Operations Research Proceedings, https://doi.org/10.1007/978-3-030-48439-2_11

methods, aiming at *fast* and *safe* iterates. Numerical tests in Sect. 4 show that it consistently performs well compared to established methods on several instances of a benchmark problem.

2 Problem

Let us consider a time interval $[0, T]$, with final time $T > 0$, and a dynamical system switching between $N > 1$ modes, with initial state $\mathbf{x}_0 \in \mathbb{R}^n$. Consider switching times $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_{N+1})^\top$ and switching intervals $\boldsymbol{\delta} = (\delta_1, \ldots, \delta_N)^\top$, satisfying $0 = \tau_1 \le \tau_2 \le \ldots \le \tau_{N+1} = T$ and $\delta_i = \tau_{i+1} - \tau_i$ for $i = 1, \ldots, N$. Hence, the set Δ of feasible vectors $\boldsymbol{\delta}$ is the simplex of size T in $\mathbb{R}^N$. Our goal is to find feasible switching intervals $\boldsymbol{\delta}^\star$ minimizing an objective functional in composite form, consisting of a Mayer term m and a switching cost term S, weighted by a scalar $\sigma > 0$. The STO problem reads

$$\begin{aligned} \underset{\boldsymbol{\delta}\in\Delta}{\text{minimize}} \quad & m(\mathbf{x}(T)) + \sigma S(\boldsymbol{\delta}) \\ \text{subject to} \quad & \dot{\mathbf{x}}(t) = \mathbf{f}_i(\mathbf{x}(t)), \quad t \in [\tau_i, \tau_{i+1}), \quad i = 1, \ldots, N \\ & \mathbf{x}(0) = \mathbf{x}_0 \end{aligned} \tag{1}$$

with each $\mathbf{f}_i : \mathbb{R}^n \to \mathbb{R}^n$ assumed differentiable [9]. The cost $S(\boldsymbol{\delta})$ can be expressed as the cardinality of the support of vector $\boldsymbol{\delta}$, for any $\boldsymbol{\delta} \in \Delta$, that is, the number of nonzero elements in $\boldsymbol{\delta}$, as proposed in [3]. The direct single shooting approach yields a reformulation of problem (1) without constraints, even though it may be at a disadvantage compared to the multiple shooting approach [7]. Due to initial conditions and dynamics in (1), a unique state trajectory $\mathbf{x}_\delta$ is obtained for any feasible $\boldsymbol{\delta} \in \Delta$, and the smooth term M can be defined as $M(\boldsymbol{\delta}) := m(\mathbf{x}_\delta(T))$. Then, problem (1) can be equivalently rewritten as a finite dimensional problem, namely

$$\underset{\boldsymbol{\delta}\in\Delta}{\text{minimize}} \quad M(\boldsymbol{\delta}) + \sigma S(\boldsymbol{\delta}) \tag{$\mathcal{P}_\sigma$}$$

which is composite nonsmooth nonconvex with a compact convex feasible set.

3 Methods

Let us consider the finite dimensional optimization problem $\mathcal{P}_\sigma$ with $\sigma > 0$. This can be handled by proximal methods [1, 5, 6], which in general require at least the gradient of the smooth term M and the proximal operator of the nonsmooth term S. Feasibility can be ensured at each iteration by considering the constraints in the proximal operator itself, so that the proximal point is always feasible [3]. Instead, for $\sigma = 0$, problem $\mathcal{P}_\sigma$ turns into a standard nonlinear program (NLP). Even in this case, standard NLP solvers may end up in local minimizers of $\mathcal{P}_\sigma$, as STO problems are often nonconvex [7].

Remark 1 (Smooth Cost and Gradient) Evaluating the gradient of the smooth term M requires computing the sensitivity of the state trajectory $\mathbf{x}_\delta(T)$ [4]. This can be achieved, e.g., by using the sensitivity equation or by linearization of the dynamics over a background time grid and direct derivation. In the numerical tests the latter approach is adopted, which can readily give second-order information too; for more details refer to [9].

Remark 2 (Proximal Operator) Given $\sigma > 0$, the proximal operator for problem $\mathcal{P}_\sigma$ is a possibly set-valued mapping [6], defined as

$$\operatorname{prox}_\gamma(\mathbf{x}) = \arg\min_{\mathbf{u}\in\Delta} \left\{ \sigma S(\mathbf{u}) + \frac{1}{2\gamma} \|\mathbf{u} - \mathbf{x}\|_2^2 \right\}, \quad \text{for any } \gamma > 0. \tag{2}$$

For $\Delta = \mathbb{R}^N$ and $\Delta = \mathbb{R}^N_{\geq 0}$, the proximal point can be expressed analytically and computed entrywise in that the optimization problem is separable. Instead, for the simplex-constrained case, entrywise optimization is not possible due to the coupling among entries. An efficient method for its evaluation is discussed and tested in [2], with accompanying code, and adopted in [3].

3.1 Sweeping Hessian Proximal Method

Let us consider a composite function $\phi := f + g$ and the problem of finding a vector $\mathbf{x}$ minimizing $\phi(\mathbf{x})$, provided an initial guess $\mathbf{x}_0$, with function f smooth, function g possibly extended real-valued, and ϕ lower bounded; further assumptions are discussed below. We propose a Sweeping HEssian ProXimal (SHEPX) method, which is an iterative proximal arc search method, inspired by the proximal arc search procedure in [5] and the averaging line search in [8]. At the k-th iteration, $k = 0, 1, 2, \ldots$, we build a local, parametric, quadratic model $\breve{f}_k^t$ of the smooth term f around the current vector $\mathbf{x}_k$, namely

$$\breve{f}_k^t(\mathbf{x}) := f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^\top (\mathbf{x} - \mathbf{x}_k) + \frac{1}{2}(\mathbf{x} - \mathbf{x}_k)^\top B_k^t (\mathbf{x} - \mathbf{x}_k) \tag{3}$$

with B_k^t a symmetric matrix. Parameter t allows to generate a family of quadratic models, depending on B_k^t, which we define as a weighted combination

$$B_k^t := t B_k + \frac{1-t}{t} I, \quad t \in (0, 1], \tag{4}$$

between the identity matrix I and a symmetric matrix B_k which models the curvature of f in a neighborhood of $\mathbf{x}_k$; this can be the exact Hessian $\nabla^2 f(\mathbf{x}_k)$ or, e.g., a BFGS approximation [5]. Given (3) and (4), the method generates sequences $\{t_k\}_k$, $\{\mathbf{x}_k\}_k$ such that each update is a solution to a composite subproblem, namely

$$\mathbf{x}_{k+1} = \arg\min_{\mathbf{x}} \left\{ \breve{f}_k^{t_k}(\mathbf{x}) + g(\mathbf{x}) \right\}, \tag{5}$$

which is amenable to (accelerated) proximal gradient methods. Concurrently, a backtracking arc search procedure finds $t_k = \beta^{i_k}$, $\beta \in (0, 1)$, with i_k the lowest nonnegative integer such that the sufficient descent condition

$$\phi(\mathbf{x}_{k+1}) < \phi(\mathbf{x}_k) - \frac{\eta}{2} t_k \|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2^2 \tag{6}$$

is satisfied, for some $\eta \geq 0$. Warm-starting the composite subproblems (5) could greatly reduce the computational requirements; however, this issue is not further developed in the following, where the current vector $\mathbf{x}_k$ is chosen as initial guess.

Remark 3 Lee et al. [5] adopted a backtracking line search procedure to select a step length that satisfies a sufficient descent condition, given a search direction obtained with $B_k^t := B_k$. Also, they mentioned a proximal arc search procedure, which has some benefits and drawbacks over the line search, such as the fact that an arc search step is an optimal solution to a subproblem but requires more computational effort. As a model for the proximal arc search, they considered $B_k^t := B_k/t$ [5, Eq. 2.20], for decreasing values of $t \in (0, 1]$, in place of (4).

For $t \to 0^+$, the model proposed in (4) yields $B_k^t \approx I/t$, which corresponds to what is assumed by proximal gradient methods. Hence, for sufficiently small $t > 0$, solutions to subproblem (5) converge on the proximal gradient step, with stepsize controlled by t, with no need to additionally estimate the Lipschitz constant of ∇f [5, 6]. On the other hand, for $t = 1$, the second-order information is fully exploited, as $B_k^1 = B_k$, possibly accelerating convergence. Thanks to these features, SHEPX seamlessly combines proximal gradient and Newton-type methods, exploiting faster convergence rate of the latter while retaining the convergence guarantees of the former [1, 5, 6]. Adopting a quasi-Newton scheme for B_k and adaptive stopping conditions for subproblems (5), as discussed in [5], makes SHEPX an inexact Newton-type proximal arc search method.

Remark 4 A detailed analysis and further development of the algorithm are ongoing research. Currently, we are interested in the requirements for having global convergence to a (local) minimizer. To this end, the forward-backward envelope could be used as a merit function to select updates with sufficient decrease, as in [8, Eq. 9], to handle nonconvex problems.

4 Numerical Results

We consider several instances of an exemplary problem and adopt different methods and variants to solve them: FISTA, an accelerated proximal gradient method [1], PNOPT, a proximal Newton-type line search method [5], and SHEPX, the aforementioned sweeping Hessian proximal method. Both exact Hessian and BFGS approximation are tested. As initial guess for problem $\mathcal{P}_\sigma$ with $\sigma > 0$, we use the solution to $\mathcal{P}_\sigma$ with $\sigma = 0$, obtained via the `fmincon` MATLAB routine, with interior-point method and initial guess $\delta_i = T/N$, $i = 1, \ldots, N$. We stress that, in general, as both terms in the composite cost function are nonconvex, only local minimizers can be detected. The results are obtained with MATLAB 2018b, on Ubuntu 16.04, with Intel Core i7-8700 3.2 GHz and 16 GB of RAM.

The Fuller's control problem has a solution which shows chattering behaviour, making it a small-scale benchmark problem [7]. We consider $N = 40$ modes, and the i-th dynamics read $\dot{x}_1 = x_2$, $\dot{x}_2 = v_i$, with the discrete-valued control v_i taking values in the given sequence $\{v^1, v^2, v^3, v^4, v^1, v^2, \ldots\}$, with values $v^1 = 1$, $v^2 = 0.5$, $v^3 = -1$ and $v^4 = -2$. Initial state $\mathbf{x}_0 = (0.01, 0)^\top$ and final time $T = 1$ are fixed. The cost functional, $\int_0^T x_1^2(t)\mathrm{d}t + \|\mathbf{x}(T) - \mathbf{x}_0\|_2^2$, can be transformed in Mayer form by augmenting the dynamics. We choose the background time grid with 100 time points [9], a maximal number of iterations (200, or 1000 for FISTA, for a fair comparison, because it is a first-order method and does not consider second-order information), and a stepsize tolerance ($\|\boldsymbol{\delta}_{k+1} - \boldsymbol{\delta}_k\|_2 < 10^{-6}$). For SHEPX, we set $\beta = 0.1$ and $\eta = 0$.

Table 1 summarizes the solutions found for different values of the switching cost σ, in terms of cost and cardinality of $\boldsymbol{\delta}^\star$. Statistics regarding the optimization process are also reported, such as required iterations and time. In Fig. 1 the state trajectories are depicted for two cases, highlighting the sparsity-inducing effect of the switching cost. The results show that SHEPX performs similarly to FISTA and better than PNOPT in terms of solution quality. We argue the line search procedure adopted by PNOPT is detrimental for cardinality optimization problems, which benefit from updating by solving a proximal subproblem. Also, SHEPX requires much less iterations than FISTA, meaning that some second-order information is exploited. Interestingly, the quasi-Newton variant of PNOPT seems to work better than the one with exact Hessian, while it holds the opposite for SHEPX. The latter might be able to exploit the second-order information which the former cannot handle with the line search, for which the positive-definite approximation obtained via BFGS is beneficial.

5 Outlook

We proposed a proximal Newton-type arc search method for dealing with cardinality optimization problems. Numerical tests on a sparse switching times optimization problem with switching cost have demonstrated the viability of the approach. A

Table 1 Solutions and computational performances, with different methods, for switching cost $\sigma \in \{10^{(i/3)-3} \mid i = 0, 1, 2, 3\}$

σ	Method	Cost value	Cardinality	Iterations	CPU time [s]
0.001	Initial guess	0.0400	40	402	4.85
	FISTA	0.0340 {0.0340}	34 {34}	200* {1000*}	5.90 {28.16}
	PNOPT	**0.0150** (0.0340)	15 (34)	200* (6)	3.85 (**0.30**)
	SHEPX	0.0330 (0.0340)	33 (34)	17 (148)	0.42 (3.84)
0.0022	Initial guess	0.0880	40	402	4.81
	FISTA	0.0726 {0.0726}	33 {33}	200* {1000*}	5.96 {27.72}
	PNOPT	0.0220 (0.0311)	10 (14)	200* (14)	3.90 (**0.43**)
	SHEPX	**0.0176** (0.0229)	8 (10)	52 (200*)	1.56 (5.08)
0.0046	Initial guess	0.1840	40	402	4.89
	FISTA	0.0329 {**0.0236**}	7 {5}	200* {351}	5.39 {8.86}
	PNOPT	0.0330 (0.0470)	7 (10)	200* (5)	3.96 (0.37)
	SHEPX	**0.0236** (0.0333)	5 (7)	12 (200*)	**0.28** (5.15)
0.01	Initial guess	0.4000	40	402	4.82
	FISTA	0.0509 {**0.0306**}	5 {3}	200* {449}	5.24 {11.14}
	PNOPT	0.0513 (0.0712)	5 (7)	200* (4)	3.90 (0.36)
	SHEPX	**0.0306** (0.0515)	3 (5)	10 (200*)	**0.26** (4.99)

Variant with more iterations in { }, and with BFGS in ( ). Symbol * denotes that the iteration limit is reached. Boldface highlights best cost value and CPU time

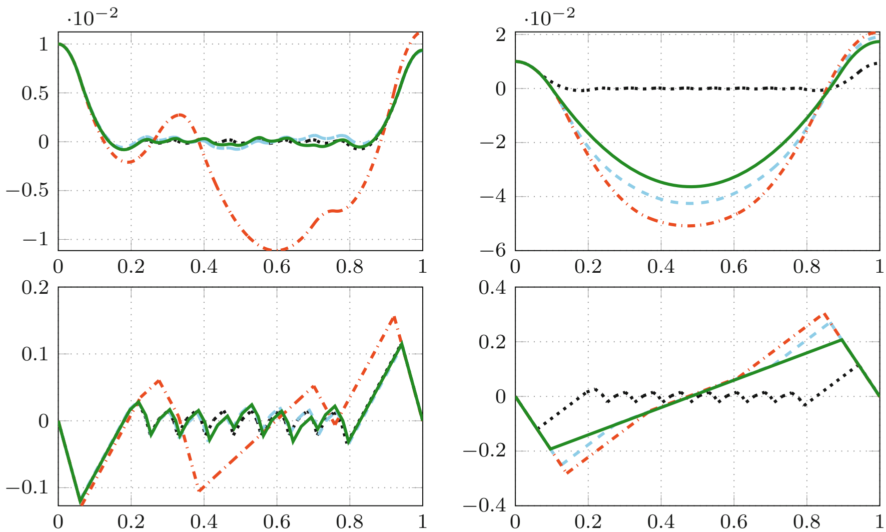

Fig. 1 Differential states x_1 (top) and x_2 (bottom) versus time t, for switching cost $\sigma = 0.001$ (left) and $\sigma = 0.01$ (right): initial guess (dotted black), FISTA (200 iterations, dashed blue), PNOPT (dash-dotted orange) and SHEPX (solid green)

comparison to other proximal methods, in terms of computation time and solution quality, has shown its effectiveness. Future research needs to further analyze the proposed method and to extend the present work to a more general class of problems. In particular, we aim at embedding proximal methods in the augmented Lagrangian framework for dealing with constraints and eventually tackling mixed-integer optimal control problems.

References

1. Beck, A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM J. Imag. Sci. **2**(1), 183–202 (2009). https://doi.org/10.1137/080716542
2. De Marchi, A.: Cardinality, Simplex and Proximal Operator (2019). https://doi.org/10.5281/zenodo.3334538
3. De Marchi, A.: On the mixed-integer linear-quadratic optimal control with switching cost. IEEE Control Syst. Lett. **3**(4), 990–995 (2019). https://doi.org/10.1109/LCSYS.2019.2920425
4. Gerdts, M.: Optimal Control of ODEs and DAEs. De Gruyter (2011). https://doi.org/10.1515/9783110249996
5. Lee, J.D., Sun, Y., Saunders, M.A.: Proximal Newton-type methods for minimizing composite functions. SIAM J. Optim. **24**(3), 1420–1443 (2014). https://doi.org/10.1137/130921428
6. Parikh, N., Boyd, S.: Proximal algorithms. Found. Trends® Optim. **1**(3), 127–239 (2014). https://doi.org/10.1561/2400000003
7. Sager, S.: Numerical methods for mixed-integer optimal control problems, Ph.D. thesis. University of Heidelberg, Heidelberg (2005). Interdisciplinary Center for Scientific Computing
8. Stella, L., Themelis, A., Sopasakis, P., Patrinos, P.: A simple and efficient algorithm for nonlinear model predictive control. In: 56th IEEE Conference on Decision and Control (CDC), pp. 1939–1944. IEEE, New York (2017). https://doi.org/10.1109/CDC.2017.8263933
9. Stellato, B., Ober-Blöbaum, S., Goulart, P.J.: Second-order switching time optimization for switched dynamical systems. IEEE Trans. Autom. Control **62**(10), 5407–5414 (2017). https://doi.org/10.1109/TAC.2017.2697681