Journal of Process Control 83 (2019) 20–29

Contents lists available at ScienceDirect

Journal of Process Control

journal homepage: www.elsevier.com/locate/jprocont

Dynamic optimization of processes with time varying hydraulic delays☆

Charles-Henri Clerget[a,*], Nicolas Petit[b]

[a] Massachusetts Institute of Technology, 77 Massachusetts Ave, Cambridge, MA 02139, United States of America
[b] MINES ParisTech, PSL Research University, CAS – Centre automatique et systèmes, 60 bd St Michel, 75006 Paris, France

ARTICLE INFO

Article history:
Received 15 October 2018
Received in revised form 21 March 2019
Accepted 25 April 2019
Available online 17 September 2019

Keywords:
Optimal control
Time varying delays
Numerical methods
Convergence analysis

ABSTRACT

In this article, we propose a numerical algorithm achieving dynamic optimization of a class of processes with input-dependent hydraulic delays. Such delays are often observed in process industries. We use the stationarity conditions to derive an iterative algorithm approaching the solution of this problem by solving a series of simpler auxiliary instances. Interestingly, the algorithm is able to leverage state-of-the-art numerical optimization tools such as IPOPT. The proof of convergence is sketched, highlighting the relevance of the chosen algorithmic structure as a form of gradient descent in a functional space. The practical interest of the algorithm is evidenced using two numerical examples to show its desirable properties of convergence and numerical efficiency.

1. Introduction

In process control, delays are a widespread issue that can have several root causes. A first problem stems from data acquisition, the limitations of the Information Technology (IT) structure and the induced mis-synchronization between networks that lead to communication delays and errors in measurements time-stamping (see [1–3] or [4]). Another common issue in process control is that the analysis of material samples may require a significant time to be performed. As evidenced for instance in [5] or [6], an important source of lags in the control is the computation times of advanced process control algorithms such as Model Predictive Control (MPC *e.g.* [7–10]) and its recent variants (adaptive MPC [11,12], economic MPC [13], distributed and cooperative MPC [14–16] or [17], among others) which consider fairly advanced models of plants. As for the intrinsic sources of delays in the processes themselves, complex chemical schemes involving activation times and the associated reaction lags are well-documented problems in chemical engineering (see [18]).

Many applications in the field of process control use dynamic optimization in order to address systems with simple time delays (*cf* [19]). For this reason, most commercial MPC tools routinely take into account fixed time delays, and implementations are common place in industrial applications. Practically, delays are usually treated directly in the time-discretization schemes by simple shifts of indexes.

The mathematical underpinnings of the optimal control of delay systems have been the subject of a large body of literature. Formally, the optimality conditions have been investigated early on by the control system community, see [20–24]. A detailed panorama of the available stationarity conditions (including the case of state-constrained problems) can be found in [25] and [26] which also propose numerical methods for implementation. These works cover cases of multiple input and state delays in Pontryagin's maximum principle. Besides MPC techniques, other approaches have focused on optimal synthesis, for fixed delays, resulting in feedback control laws. Many research efforts have focused on related numerical aspects, *e.g.* investigations regarding the stability of some orthogonal collocation schemes with regard to the transcription of systems of delay algebraic equations have also been carried (see [27]). Interestingly enough, it appears that fewer attention has been given to dynamic optimization problems under varying delays. While this topic is not new, since the seminal work of [28], most research efforts have focused on closed-form solutions to LQR problems for dynamics impacted by time-varying delays, see [29]. Practically, in most applications where delays are known to be variable, this information is simply ignored and the delays are assumed to be fixed.

☆ This works was partially funded by Total Refining and Chemicals and Total R&D Division. The authors especially wish to acknowledge Jean-Philippe Grimaldi and Mériam Chèbre for their fruitful support.

* Corresponding author.
E-mail addresses: cclerget@mit.edu (C.-H. Clerget), nicolas.petit@mines-paristech.fr (N. Petit).

https://doi.org/10.1016/j.jprocont.2019.04.013
0959-1524/

However, some type of delay phenomena are intrinsically time varying and cannot be adequately approximated by fixed delays. This is especially relevant to delays whose variability is not given *a priori* (whether in a deterministic or stochastic sense), but is rather a function of the process variables. Importantly, such situations occur when material transport phenomena give birth to delays in plants where several parts of the process are not collocated. This is typically the case when some feeds are relatively far from the core of the reactor or when complex piping and recycling architectures are present. Such situations have been documented in the control of processes for a large spectrum of applications such as biological processes (*e.g.* [30]), oil refining (*e.g.* [31–35]), polymerization (*e.g.* [36]), automotive engine regulation (*e.g.* [37,38]) or energy generation (*e.g.* [39]). Furthermore, handling these phenomena is central to the development of multi-unit and plant wide dynamic optimization strategies. Indeed, during major production transients, the setpoint changes and disturbances that cascade through the plant also affect the communication delays between the various units. In order to finely optimize the plant response, this must be taken into account to properly synchronize corrective actions.

The first theoretical results regarding the stationarity conditions of this type of systems were laid out in [40]. However, as evidenced in [41], a straightforward direct simultaneous approach using orthogonal collocations (*e.g.* [42,43] or [44]) fails on this type of applications. Indeed, the dependency of the delay with respect to the input does not allow to transcript the continuous optimal control problem as a smooth NLP.[1] To circumvent this issue, [41] attempted to discretize spatially the underlying plug-flow advection partial differential equation (PDE) to proceed with the direct optimization of the subsequent finite dimension model, as has been studied for the modelling of more complex transport systems *e.g.* [45]. However, in their numerical findings, the authors emphasize the sensitivity of the results with respect to the choice of the descretization scheme of the PDE. Overall, the numerical performances are not fully satisfactory as a good numerical accuracy of the PDE's discretization must be paid by an increased state size leading to a large computational load and high index algebraic equations if state constraints are to be imposed. Typical resolution times range from tens of seconds to a couple of minutes. The method is also shown to be prone to numerical difficulties as a refined spatial discretization of the PDE leads to the ill-conditioning of the Lagrangian's Hessian that the solver must invert. This leads to a malicious game where the numerical performances of the solver deteriorate as the dimension of the problem, and its inherent difficulty, increases.

In this paper, we lay out a first step toward addressing directly the dynamic optimization of these systems by developing an iterative procedure to solve the optimal control problem of systems displaying input-dependant input delays. We will begin by introducing the stationarity conditions of the optimal control problem that we consider. We will then use those conditions to derive a candidate iterative algorithm to solve the optimal control problem. We will sketch its proof of convergence by showing that it can be viewed as successive first order approximations of the delay dynamics and, in a limit case, boils down to a gradient descent algorithm in a functional space. Finally, we will present and discuss numerical results, based on a simple benchmark problem from [38] illustrating the convergence properties of the algorithm and a process example taken from [40] to demonstrate its practical interest. Finally, we sketch the possibility to generalize this approach to a larger class of problems, such as systems displaying state delays. We hope that our approach, which we try to present in a tutorial spirit, will encourage further research to develop new algorithms tackling these broader classes of systems of great practical importance.

[1] In the sense that this dependency leads to index commutations in the equations of the discretized dynamics.

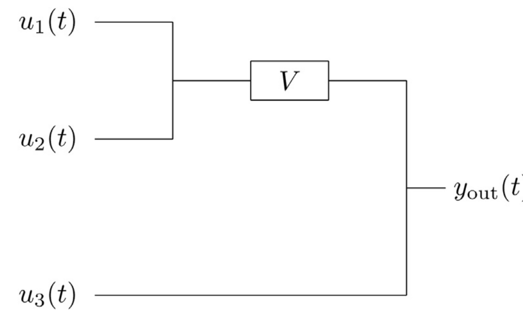


Fig. 1. Three batch mixing unit with a dead volume *V*: the "paint" problem.

2. Notations and problem statement

Mixing processes illustrate simply how hydraulic input delays arise in process control (*e.g.* [46]). Indeed, in many piping networks of industrial processes, some of the streams are merged ahead of the mixing unit. This reduces investment costs and improves compactness of the plant. However, this also creates delays in the system that must be addressed in the control algorithms.

A simplified example of such systems was given in [40], which displayed the control of a unit mixing three products. A schematic can be found in Fig. 1. Three batches are filled with products that we wish to mix. Products 1 and 2 are first mixed before going through an additional dead volume *V*. At the outlet of this pipeline, they are blended with product 3 and we wish to control the composition y of this final mixture. The total flow rate of the unit, F_{tot}, being fixed, the control variables are the injection ratios of each product at a given time $u = (u_1, u_2, u_3)$ with

$$u_1 + u_2 + u_3 = 1 \tag{1}$$

If we assume perfect mixing when streams are merged and a plug-flow regime otherwise, we find that the composition y depends both on past and present values of the control variables according to

$$y(t) = \Gamma(u(t - D_u(t)))\, u(t) \tag{2}$$

where

$$\Gamma(u) = \begin{pmatrix} \dfrac{u_1}{u_1+u_2} & \dfrac{u_1}{u_1+u_2} & 0 \\ \dfrac{u_2}{u_1+u_2} & \dfrac{u_2}{u_1+u_2} & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{3}$$

and D_u is a time varying hydraulic delay implicitly defined by

$$\int_{t-D_u(t)}^{t} F_{\text{tot}}(u_1(\tau) + u_2(\tau))\, d\tau = V \tag{4}$$

In the following, we will introduce mathematical notations to present a formal problem statement addressing the optimal control of systems presenting such hydraulic delays.

Given a set of numbers A, we denote $A^* \triangleq A\ \{0\}$.

Given $p \in \mathbb{N}^*$, we denote $\mathcal{M}_p(\mathbb{R})$ the set of real-valued square matrices of dimension p.

Let $T > 0$ and $n \in \mathbb{N}^*$, we note $L^2([0; T], \mathbb{R}^n)$ the space of square integrable functions over the interval $[0\,; T]$ and $D^1([0; T], \mathbb{R}^n)$ the space of continuous and differentiable functions on the interval $[0\,; T]$.

Problem statement: Our goal in this paper is to solve optimal control problems (generalization of a Bolza problem) of the following form

$$\begin{aligned} \mathcal{P}: \min_{v} \int_0^T & L(t,x(t),u(t)) + \frac{1}{2}v(t)^T P v(t)\,\mathrm{d}t \triangleq J(v) \\ s.t. \quad & \forall t \in [0;T], \quad \dot{x}(t) = f(t,x(t),u(t),u(r_u(t))) \\ & \forall t \in [0;T], \quad \dot{u}(t) = v(t) \\ & x(0)=x_0, \quad u_{[r_0;0]} = u_0, \quad v_{[r_0;0]} = v_0 \end{aligned} \tag{5}$$

where f and L are smooth functions, $P \in \mathcal{M}_p(\mathbb{R})$ is a symmetric definite positive matrix, x is the vector of state variables, u is the vector of control variables and $r_u(t) \triangleq t - D_u(t)$ is implicitly defined from the hydraulic delay by the relation

$$\int_{r_u(t)}^{t} \phi(u(\tau))\,\mathrm{d}\tau = 1 \tag{6}$$

with $\phi : \mathbb{R}^p \to \mathbb{R}_+^*$. $(r_0, v_0, u_0) \in \mathbb{R}_-^* \times L^2([r_0;0],\mathbb{R}^p) \times D^1([r_0;0],\mathbb{R}^p)$ and initial conditions verifying

$$\int_{r_0}^{0} \phi(u_0(\tau))\,\mathrm{d}\tau = 1 \tag{7}$$

and

$$\forall t \in [r_0;0], \quad u_0(t) = u_0(0) + \int_0^t v_0(\tau)\,\mathrm{d}\tau \tag{8}$$

Remark 1. In the simple case of Eq. (4), ϕ would boil down to the linear function

$$\phi(u_1,u_2) = F_{tot}\frac{u_1+u_2}{V} \tag{9}$$

In the following, in addition with the classic notations $\frac{\partial f}{\partial x}$ and $\frac{\partial f}{\partial u}$, we will denote $\frac{\partial f}{\partial u_r}$ the vector of partial derivatives of f w.r.t. its last argument.

Throughout the rest of the discussion, the following assumptions are considered:

Assumption 1. L is twice continuously differentiable while f, ϕ are continuously differentiable. There exists $K \geq 0$ such that $\nabla^2 L$, ∇f, $\nabla\phi$ are bounded by K and K-Lipschitz continuous.

Assumption 2. J admits a global finite lower bound, J^*.

Assumption 3. ϕ admits a strictly positive lower bound, $\phi_{\min}$.

Remark 2. Assumptions 1 and 2 are classic in optimization to insure that the problem admits a well defined global minimum and that local minima can practically be characterized using gradient-based optimization.

Assumption 3 is usually considered for systems with input varying delays of hydraulic type (see [38]) so that r_u' be bounded away from zero and the input keep on reaching the plant.

3. Algorithm design

After some lines of calculus, one can extend the results of [40] to show that the stationarity conditions of $\mathcal{P}$ are given by a two point boundary value problem

$$\begin{aligned} (u,x,\lambda,v) &= \mathfrak{P}(v) \\ Pv + v &= 0 \end{aligned} \tag{10}$$

with the operator $\mathfrak{P}$[2] defined as[3]

$$\dot{u}(t) = v(t), \quad u_{[r_0;0]} = u_0 \tag{11}$$

$$\dot{x}(t) = f(t,x(t),u(t),u(r_u(t))), \quad x(0) = x_0 \tag{12}$$

$$\dot{\lambda}(t) = -\frac{\partial L}{\partial x}(t,x(t),u(t))^T - \frac{\partial f}{\partial x}(t,x(t),u_n(t),u(r_u(t)))^T\lambda(t) \tag{13}$$

$$\begin{aligned} \lambda(T) = \;& 0 \\ \dot{v}(t) = \;& -\frac{\partial L}{\partial u}(t,x(t),u(t))^T - \frac{\partial f}{\partial u}(t,x(t),u(t),u(r_u(t)))^T \cdot \lambda(t) \\ & -\mathbb{1}_{[0;r_u(T)]}(t)(r_u^{-1})'(t)\cdot \\ & \frac{\partial f}{\partial u_r}(r_u^{-1}(t),x(r_u^{-1}(t)),u(r_u^{-1}(t)),u(t))^T\lambda(r_u^{-1}(t)) \\ & -\int_t^{r_u^{-1}(\min(t,r_u(T)))} \lambda(\tau)^T\frac{\partial f}{\partial u_r}(\tau,x(\tau),u(\tau),u(r_u(\tau)))\cdot \\ & \frac{v(r_u(\tau))}{\phi(u(r_u(\tau)))}\,\mathrm{d}\tau\,\frac{\partial\phi}{\partial u}(u(t)) \end{aligned} \tag{14}$$

$$v(T) = 0 \tag{14}$$

As mentioned previously, the input-dependency of the delay makes $\mathcal{P}$ impractical to solve directly using orthogonal collocations. Instead, we would much prefer to solve a sequence of simpler auxiliary problems and a natural idea would be to define a sequence of problems displaying a delay law which would be time-varying, but in a fixed fashion based on the value of v_n found at the nth step of the sequence so that at iteration $n+1$, one would solve[4]

$$\begin{aligned} \min_{v_{n+1}} \int_0^T & L(t,X_{n+1}(t),u_{n+1}(t)) + \frac{1}{2}v_{n+1}(t)^T P v_{n+1}(t)\,\mathrm{d}t \\ s.t. \quad & \dot{X}_{n+1}(t) = f(t,X_{n+1}(t),u_{n+1}(t),u_{n+1}(r_{u_n}(t))) \\ & \dot{u}_{n+1}(t) = v_{n+1}(t) \\ & x(0) = x_0, \quad u_{n+1\,[r_0;0]} = u_0, \quad v_{n+1\,[r_0;0]} = v_0 \end{aligned} \tag{15}$$

Mathematically, the stationarity conditions of each of these problems could then be expressed as special cases of (11)–(14) in which ϕ would no longer be a function of u but instead of t alone. However, examining Eq. (14) shows that the distributed term that it involves (the last term of (14)) would vanish (exactly, at each step) and that if the sequence was ever to converge, its solution would not verify the original stationarity conditions of $\mathcal{P}$, but a *biased* version of them. Following this remark, given an initial profile v_1 and $\alpha \in \mathbb{R}_+$, we define a sequence of auxiliary problems for all $n \geq 1$ using the following recursion

$$\begin{aligned} \mathcal{P}_{n+1}: \min_{v_{n+1}} \int_0^T & L(t,X_{n+1}(t),u_{n+1}(t)) + \frac{1}{2}v_{n+1}(t)^T P v_{n+1}(t) \\ & + \mathcal{S}_n(t)(u_{n+1}(t)-u_n(t)) + \frac{\alpha}{2}\|v_{n+1}(t)-v_n(t)\|_2^2\,\mathrm{d}t \\ s.t. \quad & \dot{X}_{n+1} = f(t,X_{n+1}(t),u_{n+1}(t),u_{n+1}(r_{u_n}(t))) \\ & \dot{u}_{n+1} = v_{n+1} \\ & X_{n+1}(0) = x_0, \quad u_{n+1\,[r_0;0]} = u_0, \\ & v_{n+1\,[r_u(0);0]} = v_0 \end{aligned} \tag{16}$$

where

[2] Technically, $\mathfrak{P} : L^2([0;T],\mathbb{R}^p) \to D^1([0;T],\mathbb{R}^p) \times D^1([0;T],\mathbb{R}^m)^2 \times D^1([0;T],\mathbb{R}^p)$.

[3] On the interval $[0;r_u(T)]$ covered by the indicator function, the function r_u^{-1} employed in (14) is well defined.

[4] We use the capital letter notation X to outline the fact that in this case, the state variable follows a different differential equation from the original problem.

$$\mathcal{S}_n(t) = \int_t^{r_{u_n}^{-1}(\min(t, r_{u_n}(T)))} \lambda_n(\tau)^T \frac{\partial f}{\partial u_r}(\tau, x_n(\tau), u_n(\tau), u_n(r_{u_n}(\tau))) \cdot \frac{v_n(r_{u_n}(\tau))}{\phi(u_n(r_{u_n}(\tau)))} \, d\tau \frac{\partial \phi}{\partial u}(u_n(t)) \tag{17}$$

is the sensitivity of the objective with respect to the change of the delay law caused by a change of the control input as derived from the calculus of variations. In the definition of $\mathcal{S}_n$ and the general statement of $\mathcal{P}_n$, $(u_n, x_n, \lambda_n, \nu_n)$ are defined as

$$v_n \mapsto (u_n, x_n, \lambda_n, \nu_n) \triangleq \mathfrak{P}(v_n) \tag{18}$$

Definition 1. Consider a sequence $(v_n)_{n\in\mathbb{N}^*}$ and $\alpha \ge 0$, (v_n) is called α-admissible if for all $n \ge 2$, v_n is a solution (possibly local) of $\mathcal{P}_n$.

Let us define

$$\mathcal{X} \triangleq \{v \in L^2([0;T]), \exists R_v \in \mathbb{R}_+, \forall w \in L^2([0;T]), \\ J(w) \le J(v) \Rightarrow \|w\|_2 \le R_v\} \tag{19}$$

the set of L^2 functions such that their J-level set is included in a ball of L^2 and note

$$g_v \triangleq Pv + \nu \tag{20}$$

The main result concerning the sequence $(\mathcal{P}_n)$ is as follows

Theorem 1. *Under Assumptions 1–3, given any α-admissible sequence $(v_n)_{n\in\mathbb{N}^*}$ such that $v_1 \in \mathcal{X}$, if α is large enough then (v_n) satisfies*

$$\lim_{n\to\infty} \|g_{v_n}\|_2 = 0 \tag{21}$$

and

$$\lim_{n\to\infty} \|v_{n+1} - v_n\|_2 = 0 \tag{22}$$

Furthermore, the sequence $(J(v_n))_{n\in\mathbb{N}^*}$ is monotonically decreasing.

Proof. Given $n \in \mathbb{N}^*$, let us assume that $v_n \in \mathcal{X}$ (which is true for $n = 1$ by assumption) and, by extension of (19), define

$$\mathcal{X}_n \triangleq \{v \in L^2([0;T]), \quad J(v) \le J(v_n)\} \subset \mathcal{X} \tag{23}$$

which is a bounded set in the sense of the L^2 norm, *i.e.* there exists $R_n > 0$ such that

$$\forall v \in \mathcal{X}_n, \quad \|v\|_2 \le R_n \tag{24}$$

Consider the operator defined by $\mathfrak{Q}(v, w) = (u, q, x, \lambda)$[5] with

$$\dot{u}(t) = v(t), \quad u_{[r_0;0]} = u_0 \tag{25}$$

$$\dot{q}(t) = w(t), \quad q_{[r_0;0]} = u_0 \tag{26}$$

$$\dot{x}(t) = f(t, x(t), u(t), u(r_q(t))), \quad x(0) = x_0 \tag{27}$$

$$\dot{\lambda}(t) = -\frac{\partial L}{\partial x}(t, x(t), u(t))^T - \frac{\partial f}{\partial x}(t, x(t), u(t), u(r_q(t)))^T \lambda(t) \tag{28}$$

$$\lambda(T) = 0 \qquad (28)$$

Note the slight (but important) differences between $\mathfrak{P}$ defined by (11)–(14) and $\mathfrak{Q}$. The second argument of $\mathfrak{Q}$ is used to define the time-varying delay appearing in the right-hand side of Eqs. (27) and (28).

[5] Technically, $\mathfrak{Q} : L^2([0;T], \mathbb{R}^p)^2 \to D^1([0;T], \mathbb{R}^p)^2 \times D^1([0;T], \mathbb{R}^m)^2$.

The newly defined operator $\mathfrak{Q}$ plays a key role with respect to the sequence (v_n). Indeed, the stationarity conditions of $\mathcal{P}_{n+1}$ are given by

$$\begin{aligned}
&(u_{n+1}, X_{n+1}, \Lambda_{n+1}) = \mathfrak{Q}(v_{n+1}, v_n) \\
&\dot{N}_{n+1}(t) = \\
&\quad - \frac{\partial L}{\partial u}(t, X_{n+1}(t), u_{n+1}(t))^T \\
&\quad - \frac{\partial f}{\partial u}(t, X_{n+1}(t), u_{n+1}(t), u_{n+1}(r_{u_n}(t)))^T \Lambda_{n+1}(t) \\
&\quad - \mathbb{1}_{[0;r_{u_n}(T)]}(t)(r_{u_n}^{-1})'(t) \cdot \\
&\qquad \frac{\partial f}{\partial u_r}(r_{u_n}^{-1}(t), X_{n+1}(r_{u_n}^{-1}(t)), u_{n+1}(r_{u_n}^{-1}(t)), u_{n+1}(t))^T \cdot \\
&\qquad \Lambda_{n+1}(r_{u_n}^{-1}(t)) - \mathcal{S}_n(t)^T \\
&N_{n+1}(T) = 0 \\
&0 = Pv_{n+1} + N_{n+1} + \alpha(v_{n+1} - v_n)
\end{aligned} \tag{29}$$

Noticing that

$$(\mathfrak{Q}(v, v), N(\mathfrak{Q}(v, v), v)) = \mathfrak{P}(v) \tag{30}$$

it is clear from the structure of $\mathcal{P}_{n+1}$ that if an α-admissible sequence (v_n) converges, its limit will satisfy Eqs. (11)–(14).

From (29), we directly deduce that the solutions of $\mathcal{P}_n$ and $\mathcal{P}_{n+1}$ are related by

$$v_{n+1} = v_n - \frac{1}{\alpha} g_{v_n} + \frac{1}{\alpha} \epsilon_{n+1} \tag{31}$$

with

$$\epsilon_{n+1} = -P(v_{n+1} - v_n) - (N_{n+1} - \nu_n) \tag{32}$$

In turn, the cost variation between v_n and v_{n+1} is given by

$$J(v_{n+1}) - J(v_n) = \int_0^1 G'(s) \, ds \tag{33}$$

where

$$G(s) = J(v_n + (v_{n+1} - v_n)s) \tag{34}$$

Using the adjoint state method (see *e.g.* [47]), one computes, after a few lines of calculus,

$$J(v_{n+1}) - J(v_n) = \int_0^1 \int_0^T g_{v_n + (v_{n+1} - v_n)s}(t)^T (v_{n+1}(t) - v_n(t)) \, dt \, ds \tag{35}$$

which gives

$$\begin{aligned}
J(v_{n+1}) - J(v_n) = \; & -\frac{1}{\alpha} \|g_{v_n}\|_2^2 + \frac{1}{\alpha} \langle g_{v_n}, \epsilon_{n+1} \rangle \\
& + \int_0^1 \langle g_{v_n + (v_{n+1} - v_n)s} - g_{v_n}, v_{n+1} - v_n \rangle \, ds
\end{aligned} \tag{36}$$

Finally

$$\begin{aligned}
J(v_{n+1}) - J(v_n) \le \; & -\frac{1}{\alpha} \|g_{v_n}\|_2^2 + \frac{1}{\alpha} \|g_{v_n}\|_2 \|\epsilon_{n+1}\|_2 \\
& + \int_0^1 \|g_{v_n + (v_{n+1} - v_n)s} - g_{v_n}\|_2 \|v_{n+1} - v_n\|_2 \, ds
\end{aligned} \tag{37}$$

At this point, the two main technical results remaining to establish the desired results are to show that $\|\epsilon_{n+1}\|_2$ admits an upper bound proportional to $\|g_{v_n}\|_2$ and that the function $v \mapsto g_v$ is Lipschitz continuous with respect to the L^2 norm on any bounded set. For the sake of briefness, the explicit derivation of these properties is not considered here. The reader can refer to [48], Chapter 5 and Appendix B for a complete proof.

Then, one shows that for α large enough, $J(v_{n+1}) - J(v_n) \le 0$. In particular, this guarantees that $v_{n+1} \in \mathcal{X}_n$. By induction, this implies that if one picks a value $\alpha = \alpha_1$ such that it guarantees a decrease in cost at $n = 1$, then for all rank n, $v_n \in \mathcal{X}_1$ and there exists $\beta > 0$ such that

$$\forall n \in \mathbb{N}^*, \quad J(v_{n+1}) - J(v_n) \le -\beta \|g_{v_n}\|_2^2 \tag{38}$$

This leads to

$$\sum_{i=0}^{N} \|g_{v_i}\|_2^2 \le \beta(J(v_0) - J(v_{n+1})) \tag{39}$$

Finally we derive

$$\sum_{i=0}^{N} \|g_{v_i}\|_2^2 \le \beta(J(v_0) - J^*) \tag{40}$$

and the convergence of the series yields

$$\lim_{n\to\infty} \|g_{v_n}\|_2 = 0 \tag{41}$$

which gives the conclusion.

Practically, the algorithm is implemented by choosing an initialization value v_1, a tuning parameter α, and for all $n \ge 1$ applying the following steps:

- given v_n, integrate to compute u_n and derive the delay law r_n from (6)
- compute (x_n, λ_n) from (12) and (13) to derive $\mathcal{S}_n$
- solve $\mathcal{P}_{n+1}$ and obtain v_{n+1}
- check if the termination criteria has been reached

Different termination criteria can be considered. In real-time applications, where time is of the essence, it could simply be reaching a given computing budget. In other cases, a criteria pertaining to the stationarity conditions such as bringing $\|g_{v_n}\|_2$ under a given threshold would be more relevant.

The choice of an appropriate value for the parameter α is important for the numerical implementation of the algorithm. From the proof of the theorem, we see that this parameter implicitly controls the size of the steps taken by the algorithms, in a way very similar to a trust-region method (*e.g.* [49]). Practically, α must be chosen large enough to insure convergence, but small enough to avoid slowing it down. While we did not explore the theoretical underpinning of this option, empirical evidence shows that α should be adapted during the optimization based on a set of step acceptance rules. Minimally, it should be adjusted to guarantee a monotonic cost decrease of the sequence.

4. Numerical examples

4.1. Illustration of the convergence results

For the sake of illustration, we treat a benchmark problem already considered in [38]. Consider a second order unstable system with dynamics given by

$$\begin{aligned} \ddot{z}(t) - \dot{z}(t) + z(t) &= u(r_u(t)) \\ \dot{u}(t) &= v(t) \end{aligned} \tag{42}$$

and

$$\int_{r_u(t)}^{t} u(\tau)\,\mathrm{d}\tau = 1 \tag{43}$$

with the following initial conditions

$$\begin{aligned} &z(0) = 1, \quad \dot{z}(0) = 0 \\ &u_{[r_0;0]} = 1, \quad v_{[r_0;0]} = 0 \end{aligned} \tag{44}$$

This can equivalently be recast as

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(r_u(t)) \\ \dot{u}(t) &= v(t) \end{aligned} \tag{45}$$

where

$$x = \begin{pmatrix} z \\ \dot{z} \end{pmatrix}, \quad A = \begin{pmatrix} 0 & 1 \\ -1 & 1 \end{pmatrix}, \quad B = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \tag{46}$$

The control of this system is challenging due to its unstable character that hampers existing robust feedback laws based on small gain assumptions (see [38]). The optimal control problem is

$$\begin{aligned} \mathcal{P} : \min_{v} \int_0^T & \|z(t) - z_r\|_2^2 + w_u \|u(t) - u_r\|_2^2 + w_v \|v(t)\|_2^2 \,\mathrm{d}t \\ s.t. \quad & \dot{x}(t) = Ax(t) + Bu(r_u(t)) \\ & \dot{u}(t) = v(t) \end{aligned} \tag{47}$$

with $T = 10$, $w_u = 0.1$, $w_v = 0.1$ and $z_r = u_r = 1.5$. Taking $\alpha = 5$,[6] we pick the trivial initialization value $v_1 = 0$ and apply the algorithms detailed at the end of Section 3 to iteratively approach a solution of $\mathcal{P}$ by constructing an α-admissible sequence (v_n).

Practically, the resolution of the sequence of auxiliary problems $(\mathcal{P}_n)$ is performed using a direct collocation transcription method, AMPL as algebraic modelling language and IPOPT 3.11.8 as NLP solver. The time horizon is divided into 100 finite elements of equal size, each of them containing 3 Radau collocation points. The results are presented in Figs. 2–5.

Figs. 2–4 display the optimal trajectory that is computed and the associated delay law (the values are obtained for $n = 100$). The inflexions of the input profile around $t = 0.9$ and $t = 2.4$ are typical of systems having variable delays. Fig. 3 pictorially shows how this trajectory is progressively approached by the sequence. Fig. 5 exhibits some indicators regarding the convergence properties of the algorithm as n grows: the cost J along with the relative steps size measured by $\log_{10}(\Delta v) \triangleq \log_{10}\left(\frac{\|v_n - v_{n-1}\|_2}{\|v_n\|_2}\right)$ and $\log_{10}(\Delta J) \triangleq \log_{10}\left(\frac{\|J_n - J_{n-1}\|_2}{\|J_n\|_2}\right)$ at successive iterations. As expected, the cost decreases monotonically and the linear shape of the cost decrease on the semi-log plot is evocative of a first order steepest descent-like method. The total computation time for the first 100 iterations displayed in Fig. 5 using a 2.60 GHz Intel(R) Core(TM) i7-4720HQ processor on a 64 bits system with a 16.0Go RAM is equal to 12.6 s, 9.8 s being actually spent in the solver.

4.2. Mixing processes

Here, we come back to the example of a mixing process that was introduced at the beginning of Section 2 (see Eqs. (1)–(4)).

Eq. (2) does not directly fit into the framework of the problem we have previously studied in Section 3. This could easily be circumvented by considering this dynamics as the limit case of

$$\frac{1}{\epsilon}\dot{y}(t) = \Gamma(u(r_u(t)))u(t) - y(t) \tag{48}$$

when ϵ goes to zero. However, we will illustrate the generality of our method by solving the optimal control directly based on Eq. (2). Let us then address the following optimal control problem

$$\mathcal{P}_m : \min_{u} \int_0^T L(t, u(t), u(r_u(t), \dot{u}(t))\,\mathrm{d}t \tag{49}$$

[6] This value was achieved by trial and error.

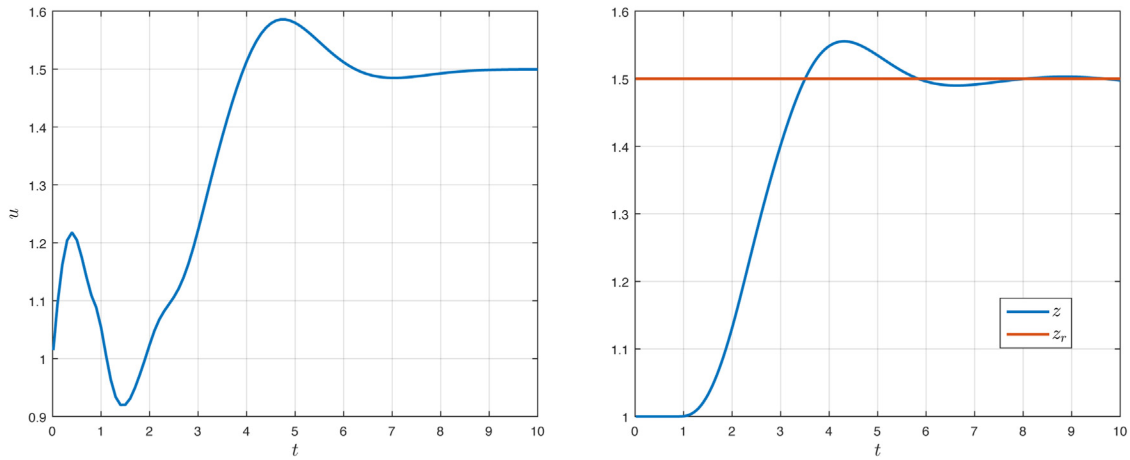


Fig. 2. Optimal trajectory computed for $\mathcal{P}$.

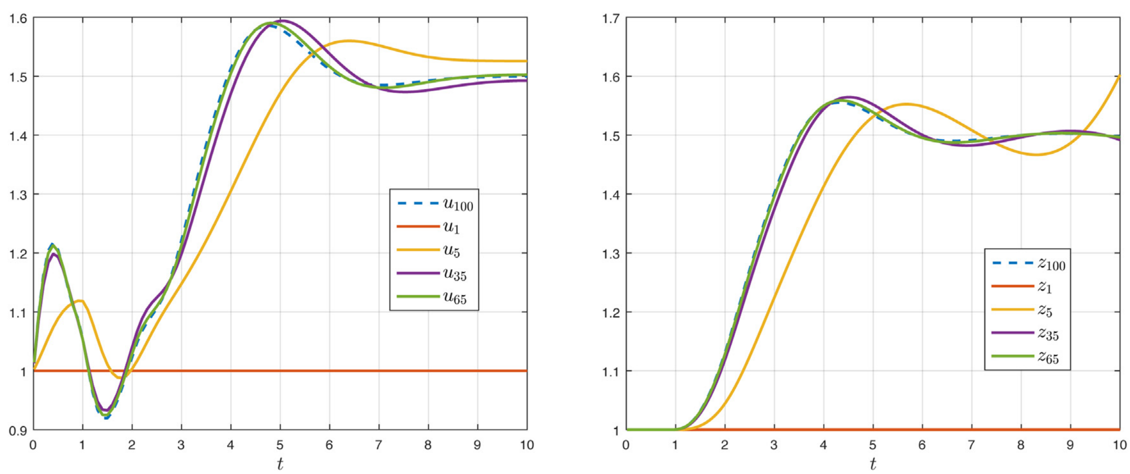


Fig. 3. Successive approximations of the optimal trajectory.

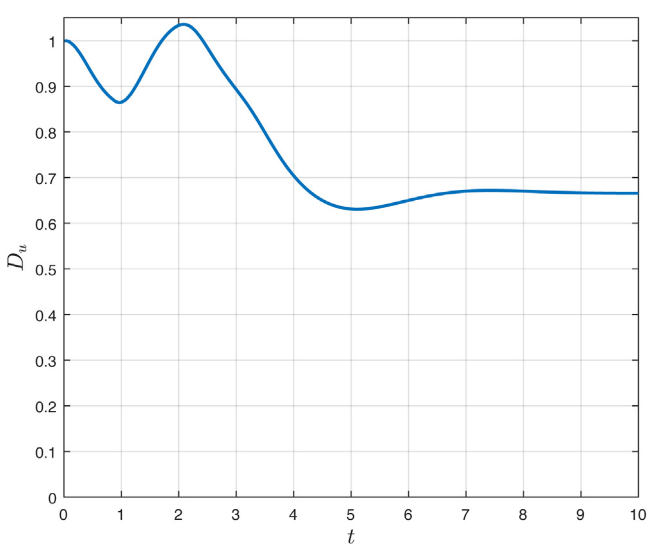


Fig. 4. Delay law of the optimal trajectory.

with

$$L(t, u(t), u(r_u(t)), \dot{u}(t)) = \|\Gamma(u(r_u(t)))u(t) - y_{\text{ref}}(t)\|_2^2 + w_{\dot{u}}\|\dot{u}(t)\|_2^2 \tag{50}$$

where y_{ref} is a desired reference trajectory (practically, we will consider a smooth transition between two operating points). Using the calculus of variations detailed in [40], we propose to solve a sequence of problems where u_{n+1} is deduced from u_n by

$$\min_{u_{n+1}} \int_0^T L(t, u_{n+1}(t), u_{n+1}(r_{u_{n+1}}(t)), \dot{u}_{n+1}(t)) + \mathcal{S}_n(t)u_{n+1}(t)\,\mathrm{d}t \tag{51}$$

with

$$\mathcal{S}_n(t) = \int_t^{r_{u_n}^{-1}(\min(t, r_{u_n}(T)))} \frac{\partial L}{\partial u_r}(\tau, x_n(\tau), u_n(\tau), u_n(r_{u_n}(\tau)))\cdot \frac{\dot{u}_n(r_{u_n}(\tau))}{\phi(u_n(r_{u_n}(\tau)))}\,\mathrm{d}\tau\,\frac{\partial \phi}{\partial u}(u_n(t)) \tag{52}$$

For our numerical experiment, we chose $w_{\dot{u}} = 10^{-3}$, $T = 6$. We denote $u_{\text{ref}} = y_{\text{ref}}$ the control sequence that would be optimal if we considered a case with a negligible delay. This trajectory is used to initialize the optimization procedure.

On this example, we converge after 5 iterations, representing 2.2 s of CPU time. The optimal trajectory is displayed in Fig. 6. One sees that after a first phase during which the initial content of the pipe needs to be flushed out, a proper pre-compensation strategy allows the system to reach its reference value. Interestingly, in this case, convergence is obtained without using a damped formulation (*i.e.* $\alpha = 0$).

Our approach can also be compared to alternative existing methods for solving such problems using finite order discretizations of the advection PDE. In this study, we consider a method of lines using a second order finite volume discretization scheme (see [41] for implementation details). We note n_d the number of cells used

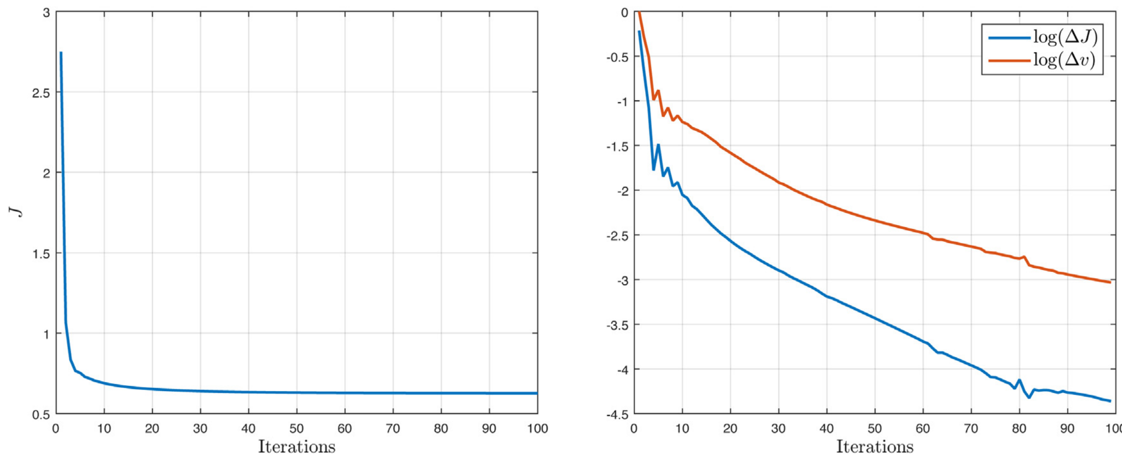


Fig. 5. Convergence properties of the algorithm.

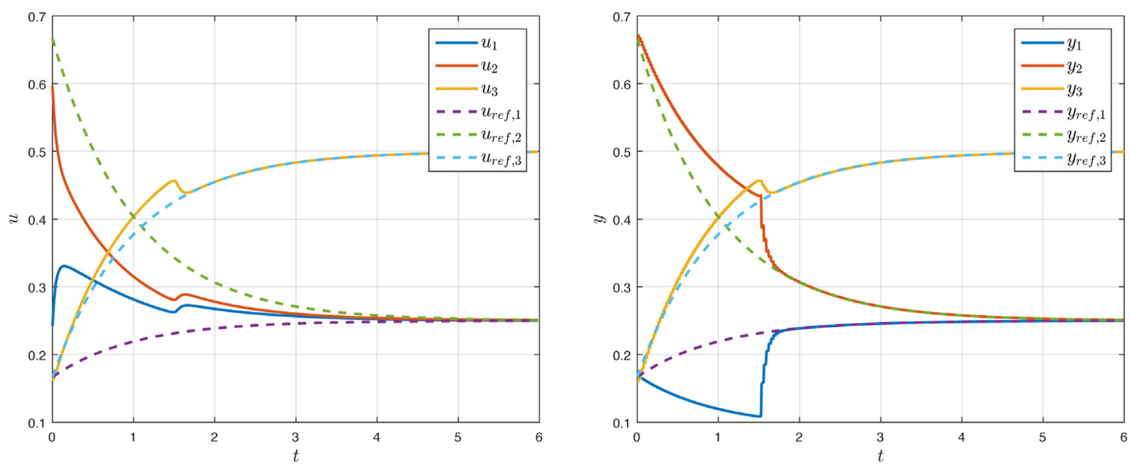


Fig. 6. Optimal trajectory computed for $\mathcal{P}_m$ using an exact resolution.

in the discretization scheme. The results of the optimization are displayed in Fig. 7 for different values of n_d. All computation times are summarized in Figure 8.

For a small number of discretization cells, such as $n_d = 10$, the computation time is already larger than using our exact method and the resulting trajectory suffers from a significant overshoot and subsequent oscillations. The computation load of the optimization sharply increases as the discretization of the advection equation is refined. Asymptotically, the optimal trajectory computed through this method seems to converge toward the solution we directly computed as n_d grows.

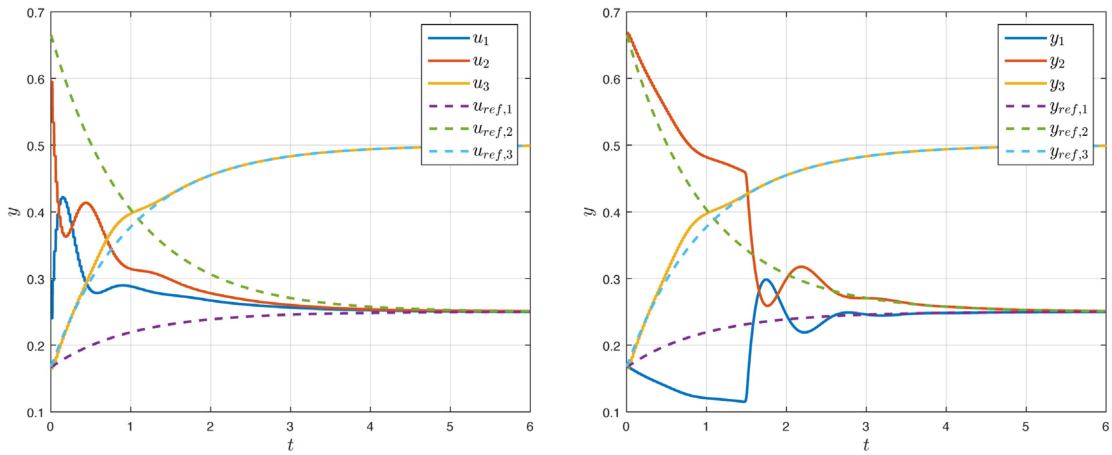
While this case displays an impressive performance gap between the two approaches, it should be noted that it will not always be in favour of our method in a more general setting. Indeed, our method leverages a first order expansion of the delay equation while a standard NLP solvers (such as IPOPT) solving the discretized transport equation makes use of second order derivatives and enjoys a faster convergence rate (quadratic if the problem is not ill-conditioned). Also, our optimization procedure is still rudimentary and an efficient implementation should encompass an adaptive step-size selection akin to that of a trust-region method. As a consequence, when the discretization error of the transport equation does not arise as a primary concern (especially in the case of smooth input signals), the benefits of using to their fullest the efficient optimization algorithms developed to solve standard collocation problems outweigh the extra accuracy of our approach. This might become critical for large scale optimization problems involving many varying delays that must simulateneously be linearized.

The extension of our approach to a second order method would require to compute the second variation of the optimal control problem $\mathcal{P}$. As the Gâteaux differentiability of $\mathcal{P}$ required the control u to be differentiable (which lead us to cast explicitly u as the integral of a non-differentiable auxiliary function v, see [48]), we believe that this extension would require to enforce one more order of differentiability of u, which might lead to practical difficulties.

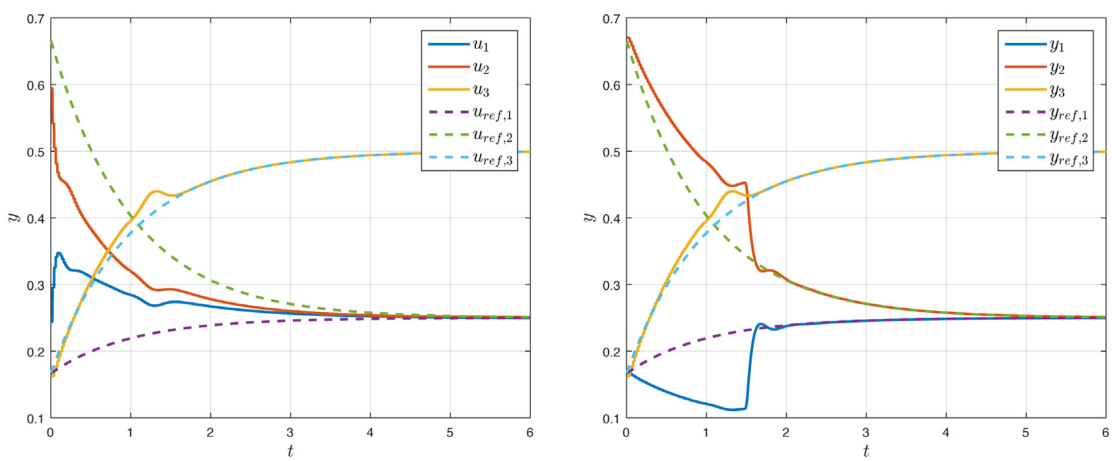
5. Towards an extension for state delays

Let us consider a variation of the problem we studied where the hydraulic delay appears in the state rather than the control

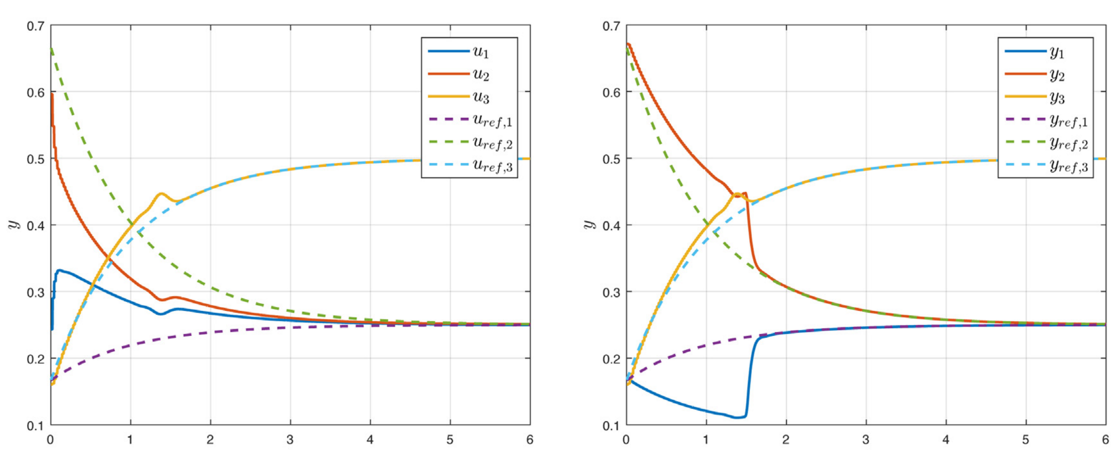
$$\begin{aligned} \tilde{\mathcal{P}}: \min_{v} \int_0^T \quad & L(t, x(t), u(t)) + \frac{1}{2} v(t)^T P v(t)\,\mathrm{d}t \\ s.t. \quad & \forall t \in [0;T], \quad \dot{x}(t) = f(t, x(t), x(r_u(t)), u(t)) \\ & \forall t \in [0;T], \quad \dot{u}(t) = v(t) \\ & x_{[r_0;0]} = x_0, \quad u(0) = u_0 \end{aligned} \tag{53}$$



(a) $n_d = 10$



(b) $n_d = 50$



(c) $n_d = 100$

Fig. 7. Optimal trajectory computed for $\mathcal{P}_m$ using a finite volume discretization of the advection equation.

Modelling	CPU
Exact delay equation	2.2 s
Finite volumes, $n_d = 10$	4.5 s
Finite volumes, $n_d = 50$	59.1 s
Finite volumes, $n_d = 100$	192.9 s

Fig. 8. Numerical results of various resolution schemes.

This class of systems is of practical interest because it models cascades of reactive units and recycle streams. One could show that the stationarity conditions of this problem are given by[7]

$$\dot{u}(t) = v(t), \quad u(0) = u_0 \tag{54}$$

$$\dot{x}(t) = f(t, x(t), x(r_u(t)), u(t)), \quad x_{[r_0;0]} = x_0 \tag{55}$$

$$\begin{aligned}\dot{\lambda}(t) = & -\frac{\partial L}{\partial x}(t, x(t), u(t))^T - \frac{\partial f}{\partial x}(t, x(t), x(r_u(t)), u(t))^T \lambda(t) \\ & -\mathbb{1}_{[0;r_u(T)]}(t)(r_u^{-1})'(t) \cdot \\ & \frac{\partial f}{\partial x_r}(r_u^{-1}(t), x(r_u^{-1}(t)), x(t), u(r_u^{-1}(t)))^T \lambda(r_u^{-1}(t))\end{aligned} \tag{56}$$

$$\lambda(T) = 0 \tag{57}$$

$$\begin{aligned}\dot{v}(t) = & -\frac{\partial L}{\partial u}(t, x(t), u(t))^T - \frac{\partial f}{\partial u}(t, x(t), u(t), u(r_u(t)))^T \lambda(t) \\ & -\int_t^{r_u^{-1}(\min(t, r_u(T)))} \lambda(\tau)^T \frac{\partial f}{\partial x_r}(\tau, x(\tau), x(r_u(\tau)), u(\tau)) \cdot \\ & \frac{\dot{x}(r_u(\tau))}{\phi(u(r_u(\tau)))} \, d\tau \frac{\partial \phi}{\partial u}(u(t))\end{aligned} \tag{57}$$

$$v(T) = 0 \tag{57}$$

Then, it would be natural to propose an iterative optimization algorithm similar to the one we studied in this paper defined by

$$\begin{aligned}\tilde{\mathcal{P}}_{n+1} : \min_{v_{n+1}} \int_0^T & \; L(t, X_{n+1}(t), u_{n+1}(t)) + \frac{1}{2} v_{n+1}(t)^T P v_{n+1}(t) \\ & + \tilde{\mathcal{S}}_n(t)(u_{n+1}(t) - u_n(t)) + \frac{\alpha}{2} \| v_{n+1}(t) - v_n(t) \|_2^2 \, dt \\ \text{s.t.} \quad & \dot{X}_{n+1} = f(t, X_{n+1}(t), X_{n+1}(r_{u_n}(t)), u_{n+1}(t)) \\ & \dot{u}_{n+1} = v_{n+1} \\ & X_{n+1[r_u(0);0]} = x_0, u_{n+1}(0) = u_0\end{aligned} \tag{58}$$

where

$$\begin{aligned}\tilde{\mathcal{S}}_n(t) = & \int_t^{r_{u_n}^{-1}(\min(t, r_{u_n}(T)))} \lambda_n(\tau)^T \frac{\partial f}{\partial x_r}(\tau, x_n(\tau), x_n(r_{u_n}(\tau)), u_n(\tau)) \cdot \\ & \frac{\dot{x}_n(r_{u_n}(\tau))}{\phi(u_n(r_{u_n}(\tau)))} \, d\tau \frac{\partial \phi}{\partial u}(u_n(t))\end{aligned} \tag{59}$$

Despite the strong analogy with the problem we addressed in this paper, a detailed analysis of the convergence of this algorithm proves to be technically more challenging than in the input delay case and is beyond the scope of this paper. One of the main technical difficulty lies in the derivation of the Lipschitz continuity of the function $v \mapsto g_v$. Indeed, it requires a result analogous to the Gronwall lemma when a varying state delay appears in the right hand side of the inequality (see [50]).

[7] See [48] for a full exposition of this result.

6. Conclusion

In this paper, we have proposed an iterative algorithm to solve the problem of optimal control of systems with hydraulic input-dependent input delays. A convergence proof was sketched and numerical results were given illustrating the practical interest of the method. More details will be given in a forthcoming publication.

The first extension to develop is a proof of convergence of the algorithm in the case of hydraulic input-dependent state delays. This case is of great practical importance since it is instrumental in the modelling of recycling loops or cascades of reacting units.

Computationally, two main improvements must be considered. First, exploit trust-region methods theory to elaborate an effective step-size selection scheme. Second, investigate whether the method can be extended to a second order expansion that would enjoy a faster convergence rate.

The current study has focused on the open-loop generation of optimal trajectories for the system. A valuable improvement would be to study the closed-loop behaviour of such a methodology used in a receding horizon framework such as MPC for real-time control applications. We expect technical challenges in the derivation of the convergence proof related to situations in which the system might not be controllable until the control input reaches the plant, especially since that instant itself depends on the control profile through the definition of the hydraulic delay.

Conflicts of interest

The authors declare no conflicts of interest.

References

[1] M. Jankovic, I. Kolmanovsky, Developments in control of time-delay systems for automotive powertrain applications, in: Delay Differential Equations, Recent Advances and New Directions, Springer Science, 2009, pp. 55–92.

[2] S. Noble, Loss, Latency, and Speed – TN0021., Tech. Rep., Data Expedition, Inc., 2012.

[3] L. Magnis, N. Petit, Impact of imprecise dating of measurements for bulk material flow network monitoring Programmable Devices and Embedded Systems, vol. 12, 2013, pp. 274–279.

[4] N. Petit, Analysis of problems induced by imprecise dating of measurements in oil and gas production, ADCHEM 2015, International Symposium on Advanced Control of Chemical Processes (2015).

[5] R. Findeisen, F. Allgöwer, Computational delay in nonlinear model predictive control, IFAC Proc. Vol. 37 (1) (2004) 427–432, 7th International Symposium on Advanced Control of Chemical Processes (ADCHEM 2003), Hong-Kong, 11–14 January 2004.

[6] V.M. Zavala, L.T. Biegler, The advanced-step NMPC controller: optimality, stability and robustness, Automatica 45 (2009) 86–93.

[7] J. Richalet, A. Rault, J.L. Testud, J. Papon, Model predictive heuristic control: application to industrial processes, Automatica 14 (1978) 413–428.

[8] D.Q. Mayne, J.B. Rawlings, C.V. Rao, P.O.M. Scokaert, Constrained model predictive control: stability and optimality, Automatica 36 (2000) 789–814.

[9] M. Diehl, H.G. Bock, J.P. Schölder, R. Findeisen, Z. Nagy, F. Allgöwer, Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations, J. Process Control 12 (4) (2002) 577–585.

[10] S.J. Qin, T. Badgwell, A survey of industrial model predictive control technology, Control Eng. Pract. 11 (2003) 733–764.

[11] G. Pannocchia, J.B. Rawlings, Disturbance models for offset-free model-predictive control, AIChE J. 49 (2) (2003) 426–437, http://dx.doi.org/10.1002/aic.690490213.

[12] G. Pannocchia, A. Bemporad, Combined design of disturbance model and observer for offset-free model predictive control, IEEE Trans. Autom. Control 52 (2007) 1048–1053.

[13] J.B. Rawlings, D.Q. Mayne, Model Predictive Control: Theory and Design, Nob Hill Publishing, 2009.

[14] B.T. Stewart, A.N. Venkat, J.B. Rawlings, S.J. Wright, G. Pannocchia, Cooperative distributed model predictive control, Syst. Control Lett. 59 (2010) 460–469.

[15] H. Scheu, W. Marquardt, Sensitivity-based coordination in distributed model predictive control, J. Process Control 21 (2011) 715–728.

[16] J.M. Maestre, D. Mu noz de la Pe na, E.F. Camacho, T. Alamo, Distributed model predictive control based on agent negotiation, J. Process Control 21 (2011) 685–697.

[17] P.D. Christofides, R. Scattolini, D. Muñoz de la Peña, J. Liu, Distributed model predictive control: a tutorial review and future research directions, Comput. Chem. Eng. 51 (2013) 21–41.

[18] M.R. Roussel, The use of delay differential equations in chemical kinetics, J. Phys. Chem. 100 (20) (1996) 8323–8330.
[19] J.P. Richard, Time-delay systems: an overview of some recent advances and open problems, Automatica (2003).
[20] A. Halanay, Optimal controls for systems with time lag, SIAM J. Control (1975).
[21] M.A. Soliman, W.H. Ray, Optimal control of multivariable systems with pure time delays, Automatica 7 (1971) 681–689.
[22] M. Malez-Zavarei, Suboptimal control of systems with multiple delays, J. Optim. Theory Appl. 30 (1980) 621–633.
[23] M. Basin, J. Rodriguez-Gonzales, Optimal control for linear systems with multiple time delays in control input, IEEE Trans. Autom. Control 51 (2006) 91–97.
[24] G.S.F. Frederico, D.F.M. Torres, Noether's symmetry theorem for variational and optimal control problems with time delay, Numer. Algeb. Control Optim. 2 (2012) 619–630.
[25] L. Göllmann, D. Kern, H. Maurer, Optimal control problems with delays in state and control variables subject to mixed control-state constraints, Optim. Control Appl. Methods 30 (4) (2009) 341–365, http://dx.doi.org/10.1002/oca.843.
[26] L. Göllmann, H. Maurer, Theory and applications of optimal control problems with multiple time-delays, J. Ind. Manag. Optim. 10 (2014) 413–441.
[27] J.T. Betts, S.L. Campbell, K. Thompson, Lobatto IIIA methods, direct transcription, and DAEs with delays, Numer. Algorithms 69 (2) (2015) 291–300.
[28] H.T. Banks, Necessary conditions for control problems with variable time lags, SIAM J. Control 6 (1) (1968) 9–47.
[29] F. Carravetta, P. Palumbo, P. Pepe, Quadratic optimal control of linear systems with time-varying input delay, 49th IEEE Conference on Decision and Control (2010).
[30] J. Harmand, D. Dochain, The optimal design of two interconnected (bio)chemical reactors revisited, Comput. Chem. Eng. 30 (1) (2005) 70–82.
[31] M. Chèbre, G. Pitollat, Feed control for an hydrodesulphurization unit using Anamel blend optimizer, ERTC MaxAsset & Computing Conference (2008).
[32] N. Petit, Systems with uncertain and variable delays in the oil industry: some examples and first solutions (plenary), Proc. of the 2nd IFAC Workshop on Automatic Control of Offshore Oil and Gas Production (2015).
[33] J. Barraud, Commande de procédés à paramètres variables, Ph.D. Thesis, École des Mines de Paris, 2006.
[34] P.D. Khandalekar, Control and Optimization of Fluid Catalytic Cracking Process, Master's Thesis, 1993.
[35] C. Antoniades, P.D. Christofides, Robust control of nonlinear time-delay systems, Int. J. Appl. Math. Comput. Sci. 9 (4) (1999) 811–837.
[36] J. Shi, L.T. Biegler, I. Hamdan, Optimization of grade transitions in polyethylene solution polymerization processes, AIChE J. 62 (4) (2016) 1126–1142.
[37] C. Depcik, D. Assanis, One-dimensional automotive catalyst modeling, Prog. Energy Combust. Sci. 31 (4) (2005) 308–369.
[38] D. Bresch-Pietri, J. Chauvin, N. Petit, Prediction-based stabilization of linear systems subject to input-dependent input delay of integral-type, IEEE Trans. Autom. Control 59 (2014) 2385–2399.
[39] L. Roca, M. Berenguel, L. Yebra, D.C. Alarcón-Padilla, Solar field control for desalination plants, Solar Energy 82 (2008) 772–786.
[40] C.-H. Clerget, J.-P. Grimaldi, M. Chèbre, N. Petit, Optimization of dynamical systems with time-varying or input-varying delays, 55th IEEE Conference on Decision and Control (2016).
[41] C.-H. Clerget, N. Petit, L. Biegler, Dynamic optimization of a system with input-dependant time delays, Chemical Process Control 2017 (2017).
[42] L.T. Biegler, An overview of simultaneous strategies for dynamic optimization, Chem. Eng. Process. Process Intensif. 46 (11) (2007) 1043–1053, special Issue on Process Optimization and Control in Chemical Engineering and Processing.
[43] A. Flores-Tlacuahuac, S. Terrazas Moreno, L.T. Biegler, Global optimization of highly nonlinear dynamic systems, Ind. Eng. Chem. Res. 47 (2008) 2643–2655.
[44] L.T. Biegler, Nonlinear Programming, Concepts, Algorithms, and Applications to Chemical Processes, MOS-SIAM Series on Optimization, 2010.
[45] A. Agarwal, Advanced Strategies for Optimal Design and Operation of Pressure Swing Adsorption Processes, Ph.D. Thesis, 2010.
[46] M. Chèbre, Y. Creff, N. Petit, Feedback control and optimization for the production of commercial fuels by blending, J. Process Control 20 (4) (2010) 441–451.
[47] G. Strang, Computational Science and Engineering, Wellesley-Cambridge Press, 2007.
[48] C.-H. Clerget, Contributions to the Control and Dynamic Optimization of Processes With Varying Delays, Ph.D. Thesis, 2017.
[49] J. Nocedal, S.J. Wright, Numerical Optimization, Springer Series in Operations Research, Springer, 1999.
[50] N. Bekiaris-Liberis, M. Krstic, Nonlinear Control Under Nonconstant Delays, vol. 25, Society for Industrial and Applied Mathematics, 2013.