



A New Lagrange-Newton-Krylov Solver for PDE-constrained Nonlinear Model Predictive Control

Christiansen, Lasse Hjuler; Jørgensen, John Bagterp

Published in:
I F A C Workshop Series

Link to article, DOI:
[10.1016/j.ifacol.2018.11.053](https://doi.org/10.1016/j.ifacol.2018.11.053)

Publication date:
2018

Document Version
Publisher's PDF, also known as Version of record

[Link back to DTU Orbit](#)

Citation (APA):
Christiansen, L. H., & Jørgensen, J. B. (2018). A New Lagrange-Newton-Krylov Solver for PDE-constrained Nonlinear Model Predictive Control. *I F A C Workshop Series*, 51(20), 325-330.
<https://doi.org/10.1016/j.ifacol.2018.11.053>

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

A New Lagrange-Newton-Krylov Solver for PDE-constrained Nonlinear Model Predictive Control

Lasse Hjuler Christiansen, John Bagterp Jørgensen

*Department of Applied Mathematics and Computer Science &
Center for Energy Resources Engineering,
Technical University of Denmark, DK-2800 Kgs. Lyngby, Denmark
(e-mails: {lhch, jbjjo}@dtu.dk)*

Abstract: Real-time optimization of systems governed by partial differential equations (PDEs) presents significant computational challenges to nonlinear model predictive control (NMPC). The large-scale nature of PDEs often limits the use of standard nested black-box optimizers that require repeated forward simulations and expensive gradient computations. Hence, to ensure online solutions at relevant time-scales, large-scale NMPC algorithms typically require powerful, customized PDE-constrained optimization solvers. To this end, this paper proposes a new Lagrange-Newton-Krylov (LNK) method that targets the class of time-dependent nonlinear diffusion-reaction systems arising from chemical processes. The LNK solver combines a high-order spectral Petrov-Galerkin (SPG) method with a new, parallel preconditioner tailored for the large-scale saddle-point systems that form subproblems of Sequential Quadratic Programming (SQP) methods. To establish proof-of-concept, a case study uses a simple parallel MATLAB implementation of the preconditioner with 10 cores. As a step towards real-time control, the results demonstrate that large-scale diffusion-reaction optimization problems with more than 10^6 unknowns can be solved efficiently in less than a minute.

© 2018, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.

Keywords: Optimal control, Model-based control, Nonlinear control, Partial differential equations, Large-scale systems, Iterative methods.

1. INTRODUCTION

Nonlinear model predictive control (NMPC) is a well-established control technique with many applications in science and engineering (Allgower et al., 1999; Mayne et al., 2000; Badgwell and Qin, 2001). However, for large-scale processes governed by partial differential equations (PDEs), the need for repeated real-time solution of PDE-constrained optimization problems pose persistent computational challenges (Bock et al., 2000; Diehl et al., 2002; Biegler, 2007; Borzi and Schulz, 2009; Leugering et al., 2014). To solve large-scale optimization problems, the literature distinguishes between two solution paradigms, *Nested-Analysis-and-Design* (NAND) and *Simultaneous-Analysis-and-Design* (SAND) (Biegler, 2000; Herzog and Kunisch, 2010). The NAND method uses a reduced space approach that eliminates the state variables from the optimization problem. This has the advantages of reducing the original problem into a sequence of comparably small optimization problems for the decision variables. As a drawback, NAND methods require repeated computation of adjoint sensitivities and solution of the state equations. For large scale problems, where the number of unknowns readily exceeds 10^6 , these requirements become prohibitively expensive. This issue is particularly pronounced for distributed control problems, where the number of decision variables is comparable to the number of states. In contrast to NAND, methods of SAND consider both the states and

the controls as independent optimization variables that are coupled through the PDE constraints. To solve the corresponding optimization problem, SAND methods solve the Karush-Kuhn-Tucker (KKT) system of first-order optimality conditions for the states, controls, and multipliers simultaneously. By solving for all dependent variables at once, SAND methods avoid the need to repeatedly solve the state equations and the corresponding sensitivity problem. In theory, this makes SAND approaches fast and ideal for real-time optimization. However, in practice, the solution of the full KKT system is a highly non-trivial task. In particular, the KKT conditions constitute a nonlinear system of PDEs coupled in both space and time. As a consequence, the optimization problem becomes huge to the point where most standard black-box optimizers are rendered computationally intractable. Hence, to realize the potential of SAND methods and promote online dynamic large-scale optimization, customized preconditioned iterative solvers become imperative.

As a contribution in this direction, this paper proposes a new parallel Lagrange-Newton-Krylov (LNK) scheme that has been tailored for a class of time-dependent nonlinear diffusion-reaction systems that arise from chemical processes (Griesse and Volkwein, 2005; Pearson and Stoll, 2013). Following the traditional LNK framework proposed by Biros and Ghattas (2005), the scheme solves the full nonlinear KKT system by a Sequentially Quadratic Programming (SQP) approach, where solution of the

associated large-scale saddle-point subproblems rely on a matrix-free preconditioned iterative Krylov subspace method. In this context, this paper contributes with the construction of a new preconditioner tailored for the DR application. At its core, the preconditioner relies on a spectral Petrov-Galerkin (SPG) discretization scheme that is inspired by the work of Shen and Wang (2007). By exploiting the structure of the SPG method, inversion of the preconditioner reduces to solution of independent subproblems that only involve small scale, sparse matrices. The independent nature of the subproblems makes the preconditioner scalable and amenable to parallelization. To establish proof-of-concept, a case study uses the new LNK scheme to solve large-scale diffusion-reaction optimization problems with more than 10^6 unknowns. The study uses a parallel implementation of the preconditioner distributed on 10 cores.

The paper is organized as follows. Section 2 introduces the optimal control problem. The new LNK solver is described in Section 3. Section 4 presents numerical results and conclusions are made in Section 5.

2. OPTIMAL CONTROL OF DIFFUSION-REACTION PROCESSES

Nonlinear diffusion-reaction systems are ubiquitous in chemical engineering and mathematical biology, where they are used to model processes for which chemical reactants or biological species evolve by means of mutual interaction. As an example, this paper considers the chemical reaction $A+B \rightarrow C$, where it is assumed that the backward reaction $C \rightarrow A+B$ is negligible. To model this process, let $\Omega \subset \mathbb{R}^d$, $d = 1, 2, 3$ be a bounded rectangular domain with boundary Γ , take $T > 0$ to be a fixed final time, and consider the space-time cylinders $Q := \Omega \times (0, T)$ and $\Sigma := \Gamma \times (0, T)$. Under the assumption that the forward reaction occurs with a constant rate, the evolution of the reactant concentrations, y, v , can be described by a system of coupled nonlinear diffusion-reaction equations in the form:

$$\partial_t y - D_1 \Delta y + k_1 y + \gamma_1 y v = f_1 \quad \text{in } Q, \quad (1a)$$

$$\partial_t v - D_2 \Delta v + k_2 v + \gamma_2 y v = f_2 \quad \text{in } Q, \quad (1b)$$

$$y = v = 0 \quad \text{on } \Sigma, \quad (1c)$$

$$y(x, 0) = y_0(x) \quad v(x, 0) = v_0(x) \quad \text{in } \Omega. \quad (1d)$$

Here D_1, D_2, k_1, k_2 are appropriate chosen constants and f_1, f_2 are given source terms. The nonlinear coupling terms, $\gamma_1 y v$ and $\gamma_2 y v$, describe the conversion rate of the reactants, where γ_i can be modeled by the *Arrhenius kinetics* expressions

$$\gamma_i = k_0 e^{-\frac{E}{\mathcal{R}T}}, \quad i \in \{1, 2\}. \quad (2)$$

Here k_0 is a proportionality constant, E is the activation energy, \mathcal{R} is the universal gas constant, and T is the time-independent reaction temperature (Warnatz et al., 2006).

2.1 The optimal control problem

To control DR processes of the type (1), this paper considers the PDE-constrained optimization problem:

$$\min_{y, v, u} J(y, v, u) \quad (3a)$$

$$\text{s.t.} \quad \partial_t y - D_1 \Delta y + k_1 y + \gamma_1 y v = \beta u + f_1 \quad \text{in } Q, \quad (3b)$$

$$\partial_t v - D_2 \Delta v + k_2 v + \gamma_2 y v = f_2 \quad \text{in } Q, \quad (3c)$$

$$y = v = 0 \quad \text{on } \Sigma, \quad (3d)$$

$$y(x, 0) = y_0(x) \quad v(x, 0) = v_0(x) \quad \text{in } \Omega. \quad (3e)$$

The problem (3) seeks to determine the optimal control input, $u \in L^2(Q)$, such that the concentrations of the reactants, y, v , track pre-specified desired states, $y_d, v_d \in L^2(\Omega)$. To this end, the objective (3a) is given by the tracking-type functional

$$J(y, v, u) = \frac{1}{2} \|y - y_d\|_2^2 + \frac{1}{2} \|v - v_d\|_2^2 + \frac{\lambda}{2} \|u\|_2^2, \quad (4)$$

where $\|\cdot\|_2$ denotes the $L^2(Q)$ norm and $\lambda > 0$ is the Tikhonov regularization parameter. As a means to control (3), this paper considers a distributed control mechanism, $u \in L^2(Q)$, that acts within the domain Ω . The control can be restricted to any given subdomain, $\Omega_c \subset \Omega$, by choosing the coefficient, $\beta = \beta(x, t)$, as an appropriate characteristic function, i.e. $\beta(x, t) := \chi_{\Omega_c}(x, t)$. In physical terms, the control models injection or suction of individual reactants. Alternatively, for chemical reactions involving ionic species, the control may represent an electric field applied to steer the system towards a desired conductivity pattern (Borzi and Griese, 2005, 2006).

3. A NEW LAGRANGE-NEWTON-KRYLOV SOLVER

As a step towards real-time dynamic optimization of PDE-constrained systems, the following introduces a new Lagrange-Newton-Krylov (LNK) solver tailored for large-scale DR problems of the type (3). In combination with appropriate state estimation tools, the solver is intended to serve as an important building block for closed-loop controllers. Motivated by the discussion of the introduction, the LNK scheme follows the approach of Simultaneous-Analysis-And-Design (SAND) that solves (3) by finding a solution to the Karuhn-Kush-Tucker (KKT) optimality system:

$$\partial_t \bar{y} - D_1 \Delta \bar{y} + k_1 \bar{y} + \gamma_1 \bar{y} \bar{v} = \beta \bar{u} + f_1 \quad \text{in } Q, \quad (5a)$$

$$\partial_t \bar{v} - D_2 \Delta \bar{v} + k_2 \bar{v} + \gamma_2 \bar{y} \bar{v} = f_2 \quad \text{in } Q, \quad (5b)$$

$$-\partial_t \bar{p} - D_1 \Delta \bar{p} + k_1 \bar{p} + \gamma_1 \bar{v} \bar{p} + \gamma_2 \bar{y} \bar{p} = y_d - \bar{y} \quad \text{in } Q, \quad (5c)$$

$$-\partial_t \bar{q} - D_2 \Delta \bar{q} + k_2 \bar{q} + \gamma_1 \bar{y} \bar{p} + \gamma_2 \bar{y} \bar{q} = v_d - \bar{v} \quad \text{in } Q, \quad (5d)$$

$$\lambda \bar{u} - \beta \bar{p} = 0 \quad \text{in } Q, \quad (5e)$$

$$y = v = p = q = 0 \quad \text{on } \Sigma, \quad (5f)$$

$$\bar{y}(x, 0) = y_0(x) \quad \bar{v}(x, 0) = v_0(x) \quad \text{in } \Omega. \quad (5g)$$

$$\bar{p}(x, T) = \bar{q}(x, T) = 0 \quad \text{in } \Omega. \quad (5h)$$

Following a Newton-Krylov strategy, the LNK scheme solves the KKT system in an iterative fashion that consists of *outer* and *inner* iterations:

- *Outer iterations* - The LNK scheme uses a Sequential Quadratic Programming (SQP) approach that applies Newton's method to divide the full nonlinear problem (5) into a sequence of linearized subproblems that determine the outer iterates, x_k .
- *Inner iterations* - To be memory-efficient and scalable, the LNK method solves each individual subproblem using an iterative Krylov subspace (KSP) method that approximates the solution of the outer SQP subproblem, x_k , by the inner iterates, x_k^k .

The main feature that separates the LNK scheme from conventional methods is the use of a custom-made spectral Petrov-Galerkin (SPG) discretization scheme to construct new, parallel and matrix-free preconditioners that accelerate convergence of the inner KSP iterations. As opposed to traditional high-order collocation methods (Biegler, 1984), the SPG scheme uses carefully designed modal bases whose special orthogonal properties lead to well-conditioned,

sparse and banded mass- and stiffness matrices (Shen and Wang, 2007). In turn, these properties are central to construction of the preconditioners and ensure efficient solution of the outer SQP subproblems.

The presentation of the LNK solver divides into three parts. The first part introduces key aspects of the SPG discretization. The second part presents the outer SQP subproblems, while the third part discusses how to speed up the inner iterations by using the SPG bases to build efficient preconditioners for the outer SQP problems.

3.1 The SPG discretization

To introduce the SPG method, let \mathbb{P}_M , $M \in \mathbb{N}$ be the set of all univariate, real-valued polynomials of degree less than or equal to M and define the spatial and temporal discretization spaces:

$$V_N = \{v \in \mathbb{P}_N : v(\pm 1) = 0\}. \quad (6)$$

$$\mathbb{T}_N := \{v \in \mathbb{P}_N : v(-1) = 0\}, \quad \mathbb{T}_N^* := \{v \in \mathbb{P}_N : v(1) = 0\}. \quad (7)$$

To be efficient, the LNK scheme relies on construction of appropriate SPG bases

$$\{\omega_k\}_{k=0}^{N_t-1} \in \mathbb{T}_{N_t}, \quad \{\omega_k^*\}_{k=0}^{N_t-1} \in \mathbb{T}_{N_t}^*, \quad \{\psi_k\}_{k=0}^{N-2} \in V_N. \quad (8)$$

The basic idea is to construct the bases such that the associated mass- and stiffness matrices obtain structured sparsity patterns that can be exploited to devise efficient preconditioners for the outer SQP problems. For the sake of brevity, the presentation of the SPG bases only covers the basic definitions and the key properties that are relevant to the preconditioners. In turn, the papers (Shen, 1994; Shen and Wang, 2007; Christiansen and Jørgensen, 2017), provide full derivations and implementation details. To construct bases for \mathbb{T}_{N_t} and $\mathbb{T}_{N_t}^*$, the SPG method uses linear combinations of the orthogonal Legendre polynomials, $\{L_k(\cdot)\}_{k=0}^N$, that have been adapted to satisfy the forward-backward conditions (5h):

$$\omega_k(t) := L_k(t) + L_{k+1}(t), \quad 0 \leq k \leq N_t - 1, \quad (9a)$$

$$\omega_k^*(t) := L_k(t) - L_{k+1}(t), \quad 0 \leq k \leq N_t - 1. \quad (9b)$$

These choices of bases imply that the matrix, D_t ,

$$d_{ij} = \langle \omega_j', \omega_i^* \rangle, \quad D_t = (d_{ij})_{i,j=0,\dots,N_t-2}, \quad (10)$$

becomes diagonal, whereas the matrices, B, By, Bp ,

$$b_{ij} = \langle \omega_j, \omega_i^* \rangle, \quad B = (b_{ij})_{i,j=0,\dots,N_t-1}. \quad (11a)$$

$$(by)_{ij} = \langle \omega_j, \omega_i \rangle, \quad By = ((by)_{ij})_{i,j=0,\dots,N_t-1}, \quad (11b)$$

$$(bp)_{ij} = \langle \omega_j^*, \omega_i^* \rangle, \quad Bp = ((bp)_{ij})_{i,j=0,\dots,N_t-1}. \quad (11c)$$

become tridiagonal. To generate a basis for V_N , the SPG scheme uses the Fourier-like (FL) basis

$$\psi_k(x) = \sum_{j=0}^{N-2} q_{jk} \phi_j(x), \quad 0 \leq k \leq N-2. \quad (12)$$

Here the functions $\{\phi_k(\cdot)\}_{k=0}^{N-2}$ are defined by:

$$\phi_j(x) = c_j (L_j(x) - L_{j+2}(x)), \quad 0 \leq j \leq N-2, \quad c_j = \frac{1}{\sqrt{4j+6}}, \quad (13)$$

where $\{L_k(\cdot)\}_{k=0}^N$ denote the orthogonal Legendre polynomials. The coefficients, q_{ij} , are given by the elements of the matrix $Q = (q_{ij})$, whose columns form the eigenvectors of the associated mass matrix $M_A = ((\phi_j, \phi_i))_{ij}$. As its key property, the FL basis satisfies the orthogonality relations

$$\langle \psi_i, \psi_j \rangle = \lambda_j \delta_{j,i}, \quad \langle \psi_i', \psi_j' \rangle = \delta_{j,i}, \quad (14)$$

where $\{\lambda_i\}_{i=1}^{N-2}$ are the eigenvalues of the matrix, M_A . As a consequence, the FL basis leads to diagonal stiffness- and mass matrices, S and M ,

$$s_{ij} = \langle \psi_j', \psi_i' \rangle, \quad S = (s_{ij})_{i,j=0,\dots,N-2}, \quad (15a)$$

$$m_{ij} = \langle \psi_j, \psi_i \rangle, \quad M = (m_{ij})_{i,j=0,\dots,N-2}. \quad (15b)$$

3.2 Outer iterations - The SQP subproblems

The outer iterations of the LNK scheme uses an SQP approach to approximate the solution to the KKT system (5) by a sequence of Newton iterates $x_i := (y_i, v_i, p_i, q_i)$, $1 \leq i \leq k$. Given, x_k , the next iterate, $x_{k+1} := (y, v, p, q)$, is determined by solution of the associated linearized optimality system:

$$Ly + \gamma_1 y_k v - \lambda^{-1} \beta^2 p = F_1(x_k), \quad (16a)$$

$$Lv + \gamma_2 v_k y = F_2(x_k), \quad (16b)$$

$$Lp + \gamma_1 p_k v + \gamma_2 C(v, q) + y = F_3(x_k), \quad (16c)$$

$$Lq + \gamma_2 q_k y + \gamma_1 C(y, p) + v = F_4(x_k). \quad (16d)$$

Here the differential operators, Ly, Lv, Lp and Lq are defined by

$$Ly := \partial_t y - D_1 \Delta y + k_1 y + \gamma_1 v_k y, \quad (17a)$$

$$Lv := \partial_t v - D_2 \Delta v + k_2 v + \gamma_2 y_k v, \quad (17b)$$

$$Lp := -\partial_t p - D_1 \Delta p + k_1 p + \gamma_1 v_k p, \quad (17c)$$

$$Lq := -\partial_t q - D_2 \Delta q + k_2 q + \gamma_2 y_k q, \quad (17d)$$

and $C(a, b) := (a_k b + b_k a)$. The right-hand side is given by

$$F_1(x_k) := f_1 + \gamma_1 v_k y_k, \quad (18a)$$

$$F_2(x_k) := f_1 + \gamma_1 v_k y_k, \quad (18b)$$

$$F_3(x_k) := y_d + \gamma_1 v_k p_k + \gamma_2 v_k q_k, \quad (18c)$$

$$F_4(x_k) := v_d + \gamma_1 y_k p_k + \gamma_2 y_k q_k. \quad (18d)$$

To discretize (16) using the SPG method, let $K := N \cdot N_t$ and introduce the combined space-time discretization spaces

$$S_K := V_N \times \mathbb{T}_{N_t} \quad \text{and} \quad S_K^* := V_N \times \mathbb{T}_{N_t}^*. \quad (19)$$

Now, consider the series representations

$$y = \sum_{k=0}^{N-2} \sum_{j=0}^{N_t-1} \hat{y}_{jk} \phi_k(x) \omega_j(t), \quad p = \sum_{k=0}^{N-2} \sum_{j=0}^{N_t-1} \hat{p}_{jk} \phi_k(x) \omega_j^*(t), \quad (20a)$$

$$v = \sum_{k=0}^{N-2} \sum_{j=0}^{N_t-1} \hat{v}_{jk} \phi_k(x) \omega_j(t), \quad q = \sum_{k=0}^{N-2} \sum_{j=0}^{N_t-1} \hat{q}_{jk} \phi_k(x) \omega_j^*(t), \quad (20b)$$

where $\{\psi_k\}_{k=0}^{N-2} \in V_N$, $\{\omega_k\}_{k=0}^{N_t-1} \in \mathbb{T}_{N_t}$, $\{\omega_k^*\}_{k=0}^{N_t-1} \in \mathbb{T}_{N_t}^*$ are the SPG bases. The finite-dimensional counterpart to the SQP subproblem (16) then becomes

$$\text{Find } y, v \in S_K, p, q \in S_K^* : \quad (21a)$$

$$\langle Ly + \gamma_1 y_k v - \lambda^{-1} \beta^2 p, w \rangle = \langle F_1, w \rangle \quad \forall w \in S_K^*, \quad (21b)$$

$$\langle Lv + \gamma_2 v_k y, w \rangle = \langle F_2, w \rangle \quad \forall w \in S_K^*, \quad (21c)$$

$$\langle Lp + \gamma_1 p_k v + \gamma_2 C(v, q) + y, w \rangle = \langle F_3, w \rangle \quad \forall w \in S_K, \quad (21d)$$

$$\langle Lq + \gamma_2 q_k y + \gamma_1 C(y, p) + v, w \rangle = \langle F_4, w \rangle \quad \forall w \in S_K. \quad (21e)$$

Next, arrange the the unknown expansion coefficients using the ordering

$$\hat{y} := (\hat{y}_0, \dots, \hat{y}_{N-2}), \quad \hat{y}_j = \{\hat{y}_{jk}\}_{k=0}^{N_t-1}, \quad (22a)$$

$$\hat{v} := (\hat{v}_0, \dots, \hat{v}_{N-2}), \quad \hat{v}_j = \{\hat{v}_{jk}\}_{k=0}^{N_t-1}, \quad (22b)$$

$$\hat{p} := (\hat{p}_0, \dots, \hat{p}_{N-2}), \quad \hat{p}_j = \{\hat{p}_{jk}\}_{k=0}^{N_t-1}, \quad (22c)$$

$$\hat{q} := (\hat{q}_0, \dots, \hat{q}_{N-2}), \quad \hat{q}_j = \{\hat{q}_{jk}\}_{k=0}^{N_t-1}, \quad (22d)$$

and define the variable coefficient mass matrices, B_y, B_v, BY_p, BY_q , and BY_β , by the entries

$$(by)_{ij} = \langle y_k \psi_k \omega_m, \psi_l \omega_n^* \rangle, \quad (23a)$$

$$(bv)_{ij} = \langle v_k \psi_k \omega_m, \psi_l \omega_n^* \rangle, \quad (23b)$$

$$(b\beta)_{ij} = \langle \beta^2 \psi_k \omega_m^*, \psi_l \omega_n^* \rangle, \quad (23c)$$

$$(byp)_{ij} = \langle p_k \psi_k \omega_m, \psi_l \omega_n \rangle, \quad (23d)$$

$$(byq)_{ij} = \langle q_k \psi_k \omega_m, \psi_l \omega_n \rangle, \quad (23e)$$

where i, j satisfy that

$$i = lN_t + (n + 1), \quad j = k(N - 1) + (m + 1), \quad (24a)$$

$$0 \leq k, l \leq N - 2, \quad 0 \leq n, m \leq N_t - 1. \quad (24b)$$

Finally, define $BY = M \otimes By$ and the submatrices, LY, LV, LP and LQ ,

$$LY = 2M \otimes D_{N_t} + D_1(S \otimes B) + k_1(M \otimes B) + \gamma_1 Bv, \quad (25a)$$

$$LV = 2M \otimes D_{N_t} + D_2(S \otimes B) + k_2(M \otimes B) + \gamma_2 Bv, \quad (25b)$$

$$LP = 2M \otimes D_{N_t} + D_1(S \otimes B) + k_1(M \otimes B)^T + \gamma_1 Bv^T, \quad (25c)$$

$$LQ = 2M \otimes D_{N_t} + D_2(S \otimes B) + k_2(M \otimes B)^T + \gamma_2 Bv^T. \quad (25d)$$

Then (21) can be written in matrix form

$$\underbrace{\begin{bmatrix} M^{(1)} & B^{(1)} \\ B^{(2)} & M^{(2)} \end{bmatrix}}_{\mathcal{A}} \underbrace{\begin{bmatrix} \widehat{y}v \\ \widehat{p}q \end{bmatrix}}_x = \underbrace{\begin{bmatrix} \widehat{G} \\ \widehat{H} \end{bmatrix}}_b, \quad (26)$$

where

$$\widehat{y}v := (\widehat{y}_0, \dots, \widehat{y}_{N-2}, \widehat{v}_0, \dots, \widehat{v}_{N-2}), \quad (27a)$$

$$\widehat{p}q := (\widehat{p}_0, \dots, \widehat{p}_{N-2}, \widehat{q}_0, \dots, \widehat{q}_{N-2}), \quad (27b)$$

$$\widehat{G} := (\widehat{g}_0^1, \dots, \widehat{g}_{N-2}^1, \widehat{g}_0^2, \dots, \widehat{g}_{N-2}^2), \quad (27c)$$

$$\widehat{H} := (\widehat{h}_0^3, \dots, \widehat{h}_{N-2}^3, \widehat{h}_0^4, \dots, \widehat{h}_{N-2}^4), \quad (27d)$$

$$\widehat{g}_j^i := \{\langle F_i, \psi_j \omega_k^* \rangle\}_{k=0}^{N_t-1}, \quad i \in \{1, 2\}, \quad (27e)$$

$$\widehat{h}_j^i := \{\langle F_i, \psi_j \omega_k \rangle\}_{k=0}^{N_t-1}, \quad i \in \{3, 4\}, \quad (27f)$$

and

$$M^{(1)} = \begin{bmatrix} BY & \Sigma_{pq} \\ \Sigma_{pq} & BY \end{bmatrix}, \quad B^{(1)} = \begin{bmatrix} LP & \gamma_2 Bv^T \\ \gamma_1 Bv^T & LQ \end{bmatrix}, \quad (28a)$$

$$B^{(2)} = \begin{bmatrix} LY & \gamma_1 Bv \\ \gamma_2 Bv & LV \end{bmatrix}, \quad M^{(2)} = \begin{bmatrix} -\lambda^{-1} BY\beta & \cdot \\ \cdot & \cdot \end{bmatrix}, \quad (28b)$$

with $\Sigma_{pq} := \gamma_1 BY_p + \gamma_2 BY_q$.

3.3 Inner iterations - The preconditioned KSP method

To solve the large-scale saddle-point problems (26) efficiently, the LNK scheme uses KSP methods that approximate the true solution, x , by the inner iterates, x^k , that belong to the k -dimensional Krylov subspaces:

$$\mathcal{K}(\mathcal{A}, r_0) := \text{span} \{r_0, \mathcal{A}r_0, \mathcal{A}^2 r_0, \dots, \mathcal{A}^{k-1} r_0\}. \quad (29)$$

Here $r_0 := b - \mathcal{A}x^0$ represents the initial residual vector. The KSP approach implies that (26) can be solved matrix-free such that the associated large-scale matrices, \mathcal{A} , do not need to be formed and stored (Christiansen and Jørgensen, 2017). However, for KSP methods to be efficient, preconditioning is essential. The goal of preconditioning is to accelerate the convergence of KSP methods by modifying the original system, e.g. by pre-multiplication of a suitable matrix, P^{-1} , i.e.,

$$P^{-1} \mathcal{A} x = P^{-1} b. \quad (30)$$

To precondition the SQP subproblems (26), this paper proposes matrices, P , in the form

$$P = \begin{bmatrix} \widehat{M}^{(1)} & \widehat{B}^{(1)} \\ \widehat{B}^{(2)} & \widehat{M}^{(2)} \end{bmatrix}. \quad (31)$$

To exploit the properties of the SPG bases, the idea is to replace the individual blocks (28) of the original system by

the approximations, $\widehat{M}^{(i)}, \widehat{B}^{(i)}$, $i = 1, 2$, that come from an SPG discretization of the associated constant-coefficient problem:

$$\widehat{L}y + \gamma_1 \overline{y}_k v - 1/2 \lambda^{-1} p = F_1(x_k), \quad (32a)$$

$$\widehat{L}v + \gamma_2 \overline{v}_k y = F_2(x_k), \quad (32b)$$

$$\widehat{L}p + \gamma_1 \overline{p}_k v + \gamma_2 C(v, q) + y = F_3(x_k), \quad (32c)$$

$$\widehat{L}q + \gamma_2 \overline{q}_k y + \gamma_1 C(y, p) + v = F_4(x_k), \quad (32d)$$

where

$$\widehat{L}y := \partial_t y - D_1 \Delta y + k_1 y + \gamma_1 \overline{v}_k y, \quad (33a)$$

$$\widehat{L}v := \partial_t v - D_2 \Delta v + k_2 v + \gamma_2 \overline{y}_k v, \quad (33b)$$

$$\widehat{L}p := -\partial_t p - D_1 \Delta p + k_1 p + \gamma_1 \overline{v}_k p, \quad (33c)$$

$$\widehat{L}q := -\partial_t q - D_2 \Delta q + k_2 q + \gamma_2 \overline{y}_k q, \quad (33d)$$

Here the constant coefficients, $\overline{y}_k, \overline{v}_k, \overline{p}_k, \overline{q}_k$, are chosen as the mean of the corresponding iterates, y_k, v_k, p_k, q_k , i.e.,

$$\overline{w}_k := \frac{1}{2} \left(\max_{x \in Q} w_k(x) + \min_{x \in Q} w_k(x) \right), \quad w \in \{y, v, p, q\}. \quad (34)$$

Hence, during each inner iteration, application of the preconditioner requires solution of the linear system of equations

$$\underbrace{\begin{bmatrix} \widehat{M}^{(1)} & \widehat{B}^{(1)} \\ \widehat{B}^{(2)} & \widehat{M}^{(2)} \end{bmatrix}}_P \underbrace{\begin{bmatrix} \widehat{y}v^k \\ \widehat{p}q^k \end{bmatrix}}_{z^k} = \underbrace{\begin{bmatrix} \widehat{G}^k \\ \widehat{H}^k \end{bmatrix}}_{\mathcal{A}x^k}. \quad (35)$$

To ensure low cost inversion of the preconditioner, the LNK scheme relies on the orthogonality relations of the SPG bases. In particular, using the diagonal structure of the matrix D_t defined by (10) and the FL properties (14), it follows that the $4(N-1)N_t \times 4(N-1)N_t$ system (35) decouples into $(N-1)$ independent $4N_t \times 4N_t$ subsystems in the form

$$\begin{bmatrix} \widehat{M}_s^{(1)} & \widehat{B}_s^{(1)} \\ \widehat{B}_s^{(2)} & \widehat{M}_s^{(2)} \end{bmatrix} \begin{bmatrix} \widehat{y}v_j^k \\ \widehat{p}q_j^k \end{bmatrix} = \begin{bmatrix} \widehat{g}_j^k \\ \widehat{h}_j^k \end{bmatrix}, \quad 0 \leq j \leq N-2, \quad (36)$$

where

$$M_s^{(1)} = \begin{bmatrix} \lambda_j By & \widehat{\Sigma}_{pq} \\ \widehat{\Sigma}_{pq} & \lambda_j By \end{bmatrix}, \quad B_s^{(1)} = \begin{bmatrix} \widehat{L}P & \gamma_2 \lambda_j \overline{v}_k B^T \\ \gamma_1 \overline{y}_k B^T & \widehat{L}Q \end{bmatrix}, \quad (37a)$$

$$B_s^{(2)} = \begin{bmatrix} \widehat{L}Y & \gamma_1 \lambda_j \overline{y}_k B \\ \gamma_2 \lambda_j \overline{v}_k B & \widehat{L}V \end{bmatrix}, \quad M_s^{(2)} = \begin{bmatrix} -1/2 \lambda^{-1} \lambda_j Bp & \cdot \\ \cdot & \cdot \end{bmatrix}. \quad (37b)$$

Here $\widehat{\Sigma}_{pq} := \gamma_1 \lambda_j \overline{p}_k By + \gamma_2 \lambda_j \overline{q}_k By$ and

$$\widehat{L}Y = 2\lambda_j D_{N_t} + D_1 B + k_1 \lambda_j B + \gamma_1 \lambda_j \overline{v}_k B, \quad (38a)$$

$$\widehat{L}V = 2\lambda_j D_{N_t} + D_2 B + k_2 \lambda_j B + \gamma_2 \lambda_j \overline{y}_k B, \quad (38b)$$

$$\widehat{L}P = 2\lambda_j D_{N_t} + D_1 B + k_1 \lambda_j B^T + \gamma_1 \lambda_j \overline{v}_k B^T, \quad (38c)$$

$$\widehat{L}Q = 2\lambda_j D_{N_t} + D_2 B + k_2 \lambda_j B^T + \gamma_2 \lambda_j \overline{y}_k B^T. \quad (38d)$$

Similarly, in the two and three dimensional cases, one can show that the corresponding preconditioners, P , reduce to $(N-1)^2$ and $(N-1)^3$ subsystems of dimensions $4N_t \times 4N_t$, respectively. Hence, by exploiting the structure of SPG bases, it is possible to reduce the multi-dimensional preconditioning problem (35) to a sequence of $(N-1)^d$ independent subproblems (36) that only involve blocks of $N_t \times N_t$ tridiagonal matrices, where d denotes the spatial dimension and N, N_t are the number of modes in the space and time directions, respectively. Further, due to high-order accuracy of the SPG bases, the number of modes can usually be chosen small, i.e. $N, N_t < 30$.

This implies that the preconditioner can be inverted efficiently and robustly by solving each subsystem using, e.g., a sparse direct solver. Finally, by the *independent* nature of the subsystems, the preconditioner is readily inverted in parallel by distributing a given portion of the subsystems onto different cores.

To summarize, Algorithm 1 provides a conceptual description of the LNK scheme.

Algorithm 1 Conceptual LNK scheme

```

1: procedure OFFLINE COMPUTATIONS
2:   Fix  $N$  and  $N_t$ .
3:   Compute the SPG basis functions by (9) and (12).
4:   Compute the  $N - 1$  eigenvalues,  $\{\lambda_k\}_{k=0}^{N-2}$ , of  $M_A$ .
5:   Compute the  $N_t \times N_t$  tridiagonal matrices,  $B, B_y, B_p$ .
6: end procedure
7: procedure OUTER SQP ITERATIONS
8:   Choose  $x_0$ . Set  $k := 0$ . Set  $\varepsilon := 1$ .
9:   while  $\varepsilon > \text{tol}$  do
10:    Compute  $\widehat{G}(x_k)$  and  $\widehat{H}(x_k)$ .
11:    Compute  $\widehat{y}_k, \widehat{v}_k, \widehat{p}_k, \widehat{q}_k$  to setup  $P$  by (35).
12:    procedure INNER KSP ITERATIONS
13:      Solve  $PA = Pb$  by GMRES to obtain  $\widehat{y}v_{k+1}, \widehat{p}q_{k+1}$ .
14:       $\triangleright$  Invert  $P$  in parallel using (36).
15:    end procedure
16:    Compute  $x_{k+1} := (y_{k+1}, v_{k+1}, p_{k+1}, q_{k+1})$  using (20).
17:    Compute  $\varepsilon := \|x_{k+1} - x_k\|_0$ .
18:    Set  $k := k + 1$ .
19:  end while
20: end procedure

```

4. NUMERICAL RESULTS

To demonstrate the potential of the LNK scheme as a fast and memory-efficient PDE-constrained optimization solver, the following case study considers the open-loop DR optimal control problem (1) in 3D. The main focus of the case study is to illustrate computational efficiency and robustness of the proposed preconditioner (26). To this end, the study investigates how the number of KSP iterations depend on 1) problem dimensionality and 2) different values of the non-linear coupling parameters γ_1, γ_2 . In this regard, the study considers a slightly modified version of a model problem that was first considered by Pearson and Stoll (2013). Here the domain is $\Omega = [0, 2]^3$ and $T = 1$. The objective is to track the desired states

$$y_d(x_1, x_2, x_3, t) := \begin{cases} 0.7, & (x_1, x_2, x_3) \in [1/2, 3/2]^3 \\ 0.2, & \text{Otherwise} \end{cases}, \quad (39)$$

$$v_d(x_1, x_2, x_3, t) := 0. \quad (40)$$

Note that (39) is unattainable, since it does not satisfy the boundary conditions. The fixed model parameters are given by $D_1 = D_2 = k_1 = k_2 = 1$. All experiments use the regularization parameter $\lambda := 10^{-5}$. The initial concentrations are given by

$$y_0(x_1, x_2, x_3) = v_0(x_1, x_2, x_3) = 0.2 \cdot \chi_{[(1/2, 3/2)^3]}(x_1, x_2, x_3), \quad (41)$$

while the source terms, f_1 and f_2 , are set to zero. All computations and timings are carried out in MATLAB (2017a) using the DTU Cluster¹ application nodes with a total of 10 Core Intel Xeon E5-2660v3 w. The outer SPG algorithm uses a tolerance of $\epsilon := 10^{-4}$. The inner KSP iterations are performed using MATLABs implementation of GMRES with a tolerance of 10^{-6} . The preconditioner is inverted in

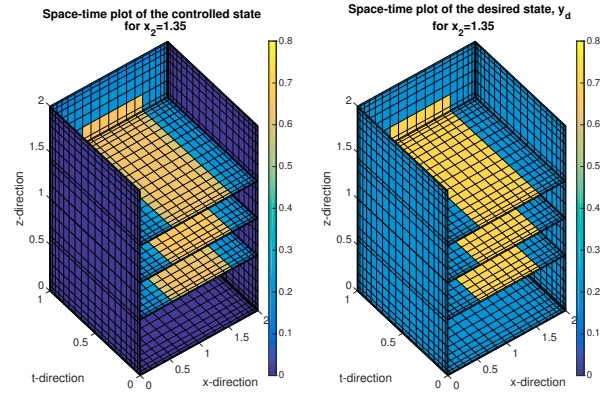


Fig. 1. Comparison of 1) the controlled state, y and 2) the desired state y_d .

parallel by assigning $\lfloor \frac{N-1}{N_c} \rfloor$ subsystems (36) to each of the N_c cores. The remainder, if any, is allocated to the core with index, N_c . Parallelization of the preconditioner uses MATLABs Single-Program-Multiple-Data (SPMD) framework.

4.1 Case study - Open-loop optimal control of chemical processes

Fig. 1 compares the computed concentration of the first reactant to the desired state, y_d , in a space-time plot for $N \times N \times N \times N_t = 20 \times 20 \times 20 \times 20$, where the spatial variable, x_2 , has been fixed for visual convenience. In general, the results show good agreement between the reactant, y , and y_d . Since the control, u , only directly affects y , it is in general difficult to control the concentration, v . In this case study, v remains close to the initial condition (41) for all times. To support computational efficiency, Table 1 shows how 1) CPU time, 2) the number of outer SQP iterations, and 3) the average number of inner KSP iterations required per outer SQP iteration, depend on the number of degrees of freedom (DOF) and the coupling parameters $\gamma_1, \gamma_2 \in \{0.15, 1.5, 15\}$. The DOF increase by doubling the number of temporal modes, N_t . The CPU timings demonstrate fast solution of (1) with a large number of DOF. For such problem sizes, conventional NAND approaches become intractable. Further, in favor of scalability, the timings grow approximately linear in the sense that the required CPU time doubles as the problem size doubles. Note also, compared to Pearson and Stoll (2013) that use similar hardware, the timings show significant improvement. In this regard, it should be mentioned that Pearson and Stoll (2013) consider their implementation prototypic and foresee significant speed-ups given appropriate modifications. Finally, the results support robustness. In particular, Table 1 shows that both the numbers of outer and inner iterations are largely invariant to the problem size and the model parameters, γ_1, γ_2 . However, a minor benign growth in the inner KSP iterations occur when γ_1, γ_2 increase.

5. CONCLUSIONS AND FUTURE WORK

As a step towards development of closed-loop controllers for real-time NMPC of systems governed by PDEs, this paper has introduced a new Lagrange-Newton-Krylov (LNK)

¹ See www.hpc.dtu.dk for a description of the DTU Cluster.

Table 1.

Optimal tracking problem									
	$\gamma_1 = \gamma_2 = 0.15$			$\gamma_1 = \gamma_2 = 1.5$			$\gamma_1 = \gamma_2 = 15$		
DOF	Time (s)	SQP steps	KSP iter.	Time (s)	SQP steps	KSP iter.	Time (s)	SQP steps	KSP iter.
640000	26.8	3	3	28.54	3	4	45.044	3	6
1280000	37.68	3	3	43.02	3	4	64.820	3	6
2560000	75.56	3	3	92.81	3	4	141.80	3	6

solver that targets fast and memory-efficient open-loop optimization of time-dependent, nonlinear, diffusion-reaction systems. As the main contribution, the LNK scheme has introduced a new type of preconditioners that are based on a customized spectral Petrov-Galerkin discretization method. Numerical results have demonstrated computational efficiency of the preconditioners with a potential to efficiently handle large-scale problems with more than 10^6 unknowns in reasonable times using a simple prototypic MATLAB implementation. While this paper considers diffusion-reaction systems with two reactants, the LNK approach generalizes to multiple coupled equations in a straightforward manner. In particular, in the general case, the preconditioning problem decouples into subsystems of dimensions $2N_r N_t \times 2N_r N_t$, where N_r is the number of coupled reactants. Further, while this paper restricts attention to coupled diffusion-reaction systems, the SPG method can also handle scalar equations with more general non-linear reaction kinetics and certain types of diffusion-convection problems (Christiansen and Jørgensen, 2017, 2018). Future work focuses on extending the LNK scheme from distributed control problems to include the case of Neumann boundary control.

REFERENCES

- Allgower, F., Badgwell, T.A., Qin, J.S., Rawlings, J.B., and Wright, S.J. (1999). Nonlinear predictive control and moving horizon estimation - An introductory overview. *Advances in Control. Highlights of Ecc'99*, 391–449, 391–449.
- Badgwell, T.A. and Qin, S.J. (2001). Review of nonlinear model predictive control applications. *Nonlinear Predictive Control Theory and Practice*, 3–32, 3–32.
- Biegler, L. (1984). Solution of dynamic optimization problems by successive quadratic programming and orthogonal collocation. *Computers and Chemical Engineering*, 8(3), 243–247.
- Biegler, L.T. (2007). *Real-time PDE-constrained optimization*, volume 3. Society for Industrial and Applied Mathematics.
- Biegler, L. (2000). Efficient solution of dynamic optimization and NMPC problems. *Prog Syst C*, 26, 219–243.
- Biros, G. and Ghattas, O. (2005). Parallel Lagrange-Newton-Krylov-Schur methods for PDE-constrained optimization. Part I: The Krylov-Schur solver. *SIAM Journal on Scientific Computing*, 27(2), 687–713.
- Bock, H.G., Diehl, M.M., Leineweber, D.B., and Schlöder, J.P. (2000). A direct multiple shooting method for real-time optimization of nonlinear DAE processes. *Nonlinear Model Predictive Control (progress in Systems and Control Theory Vol.26)*, 245–67, 245–267.
- Borzi, A. and Griese, R. (2005). Experiences with a space-time multigrid method for the optimal control of a chemical turbulence model. *International Journal for Numerical Methods in Fluids*, 47(8-9), 879–885.
- Borzi, A. and Griese, R. (2006). Distributed optimal control of lambda-omega systems. *Journal of Numerical Mathematics*, 14(1), 17–40.
- Borzi, A. and Schulz, V. (2009). Multigrid methods for PDE optimization. *SIAM Rev.*, 51(2), 361–395.
- Christiansen, L.H. and Jørgensen, J.B. (2017). A fast and memory-efficient spectral Galerkin scheme for distributed elliptic optimal control problems. *arXiv:1712.08225*.
- Christiansen, L.H. and Jørgensen, J.B. (2018). A fast PDE-constrained optimization solver for nonlinear diffusion-reaction processes. *To appear: Proceedings of 2018 IEEE 57th Annual Conference on Decision and Control*.
- Diehl, M., Bock, H., Schlöder, J., Findeisen, R., Nagy, Z., and Allgower, F. (2002). Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations. *Journal of Process Control*, 12(4), 577–585.
- Griese, R. and Volkwein, S. (2005). A primal-dual active set strategy for optimal boundary control of a nonlinear reaction-diffusion system. *SIAM J. Control Optim.*, 44(2), 467–494.
- Herzog, R. and Kunisch, K. (2010). Algorithms for PDE-constrained optimization. *Gamm Mitteilungen*, 33(2), 163–176.
- Leugering, G., Benner, P., Engell, S., Griewank, A., Harbrecht, H., Hinze, M., Rannacher, R., and Ulbrich, S. (2014). *Trends in PDE Constrained Optimization*. International Series of Numerical Mathematics. Springer International Publishing.
- Mayne, D., Rawlings, J., Rao, C., and Scokaert, P. (2000). Constrained model predictive control: Stability and optimality. *Automatica*, 36(6), 789–814. doi:10.1016/S0005-1098(99)00214-9.
- Pearson, J.W. and Stoll, M. (2013). Fast iterative solution of reaction-diffusion control problems arising from chemical processes. *SIAM J. Sci. Comput.*, 35(5), B987–B1009.
- Shen, J. (1994). Efficient spectral-Galerkin method. I. direct solvers of second- and fourth-order equations using Legendre polynomials. *SIAM J. Sci. Comput.*, 15(6), 1489–505, 1489–1505.
- Shen, J. and Wang, L.L. (2007). Fourierization of the Legendre-Galerkin method and a new space-time spectral method. *Appl. Numer. Math.*, 57(5-7), 710–720.
- Warnatz, J., Maas, U., and Dibble, R.W. (2006). Combustion: Physical and chemical fundamentals, modeling and simulation, experiments, pollutant formation. *Combustion: Physical and Chemical Fundamentals, Modeling and Simulation, Experiments, Pollutant Formation*, 1–378.