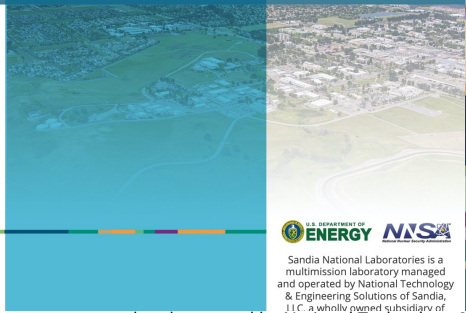
National Laboratories

A Fast Matrix-Free Method for Low-Thrust Trajectory Optimization

Aurya Javeed, Drew Kouri, Denis Ridzal, and I. Michael Ross

2023 AAS/AIAA Astrodynamics Specialist Conference



U.S. DEPARTMENT OF ENERGY NNSA

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of

Sandia National Laboratories

U.S. DEPARTMENT OF ENERGY NNSA

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Secu DE-NA0003525.

SAND2023-08452C

Overview

This talk is about numerics for trajectory optimization.

Spectral collocation methods (also called *pseudospectral methods*) can be accurate, but they produce *dense* matrices that are not thought to scale to large discretizations.

We have developed scalable solves for large spectral collocation discretizations.

Without loss of generality, we focus on initial value problem (IVP) constraints:

$$\begin{aligned} \underset{u,T}{\text{minimize}} \quad & f(x,u,T) \\ \text{subject to} \quad & 0 = c(x,u,T) \triangleq \begin{cases} \dot{x} - g(x,u,t) \\ x(0) - x_i. \end{cases} \end{aligned} \tag{1}$$

- In words, the goal is to a find control $u(t)$ and terminal time T for which the objective f is smallest.

Background

Orbit Transfer

We will use orbit transfer as a running example. Large discretizations are needed to adequately resolve low-thrust problems.

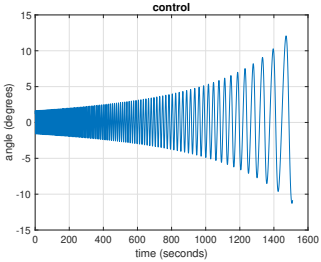
Following [4], we take the physics to be

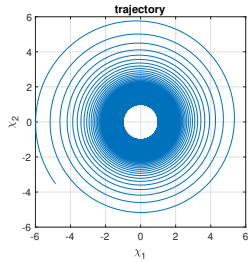
$$x = (r, v_r, v_t), \qquad g(x, u, t) = \begin{cases} v_r \\ \frac{v_t^2}{r} - \frac{1}{r^2} + a \sin u \\ -\frac{v_r v_t}{r} + a \cos u. \end{cases}$$

- u is the angle a thruster makes with the back of a satellite
- a is the thrust amplitude.

Goal: Transfer the satellite from a circular orbit of radius r_i to a larger radius of r_f in as little time as possible ($f = T$).

$$x_i = (r_i, 0, r_i^{-1/2}) \quad \text{and} \quad x_f = (r_f, 0, r_f^{-1/2})$$



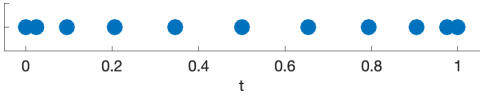


$a = 10^{-4}$

Spectral Collocation (Pseudospectral Methods)

These techniques are *finite element* methods.

Let $c = (c_1, \ldots, c_n)$ be a grid of n times, e.g.,



Most commonly,

The state x of the optimal control problem is approximated as a degree $n-1$ polynomial $p_n(t)$.

$p_n(t)$ is required to satisfy the IVP at the initial time and at $n-1$ of the c_i, i.e.,

$$\begin{cases} p_n(0) & = x_i, \\ \dot{p}_n(c_i) & = g(p_n(c_i), u(c_i), c_i), \quad \|\{i\}\| = n-1. \end{cases}$$

These n conditions define a system of n equations for n unknowns

$$p \triangleq p_n|_c = \big(p_n(c_1), \ldots, p_n(c_n)\big)$$

that uniquely specify $p_n(t)$.

Integration-Based Collocation

Instead of differentiating $p_n(t)$, we prefer

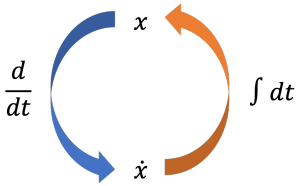
(1) enforcing the integral form of the IVP:

$$x(t) = x_i + \int^t g(x(t), u(t), t)\, dt.$$

This approach is algebraically equivalent to

(2) discretizing the derivative $v = \dot{x}$ and enforcing the differental form of the IVP:

$$v(t) = g(x_i + \int^t v(t)\, dt, u(t), t).$$



Both (1) and (2) are single-interval implicit Runge-Kutta formulations of collocation [3].

A dense integration matrix B is key. **B maps values on c to the integral of their polynomial interpolant on c.**

Trajectory Optimization

Derivative-based optimization algorithms are, at their core, Newton's method applied to a set of conditions $F(\zeta) = 0$.

Let $JF(\zeta)$ be the Jacobian of F at ζ. The Newton direction η is the solution of

$$\big(JF(\zeta)\big)\eta = -F(\zeta). \tag{2}$$

Let u be the values of the control at the collocation points: $u = \big(u(c_1), \ldots, u(c_n)\big)$.

Consider the toy problem of solving the collocation equations for p (values of the polynomial approximation of the state) given u and T:

$$F(p) = p - \big(x_i + Bg(p, u, c)\big) \quad \text{and} \quad JF(p) = I - BJ_p g(p, u, c).$$

We developed ingredients for solving this problem quickly.

These same ingredients scale bona fide Newton solves for trajectory optimization.

Matrix-Free Solves

Matrix-Free Linear Algebra

The Newton system

$$\big(I - BJ_p g(p,u,c)\big)\eta = \big(x_i + Bg(p,u,c)\big) - p$$

is an $n \times n$ linear problem

$$Ax = b.$$

We can solve this problem quickly with *matrix-free* linear algebra, i.e.,

we never instantiate A as an array of values.

We use an iterative solver that only requires applications of A to vectors.

Generalized Minimal Residual Method (GMRES) [6]

The Cayley-Hamilton theorem implies that the solution x belongs to a Krylov subspace of A, i.e.,

$$x \in \mathcal{K}_k \triangleq \text{span}(b, Ab, \ldots, A^k b) \quad \text{for some} \quad k \leq \ell.$$

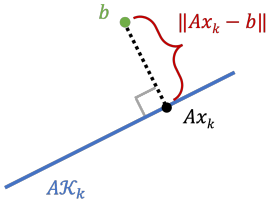
GMRES uses an orthonormal basis $\{q_i\}_{i=1}^k$ for $\mathcal{K}_k$ to cheaply compute

$$x_k = \arg\min_{\tilde{x} \in \mathcal{K}_k} \|A\tilde{x} - b\|.$$

Until $\|Ax_k - b\|$ is sufficiently small, $k \to k+1$:

- $\mathcal{K}_k \to \mathcal{K}_{k+1}$ by multiplying Aq_k,
- orthogonalize Aq_k against $\{q_i\}_{i=1}^k$ to get q_{k+1}:

$$AQ_k = Q_{k+1}H_k. \quad \text{(Arnoldi)}$$



Algorithm 1 GMRES

$q = b/\|b\|$
for $k = 1, 2, \ldots$ **do**
  step k of Arnoldi
  $y = \arg\min_y \left\|H_k y - \|b\| e_1\right\|$
  $x = Q_k y$
end for

Fast Matrix-Free Solves

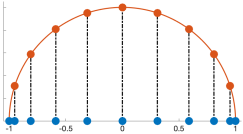
Ingredient 1: Fast Matrix-Vector Multiplies (MVMs) [2]

$$Ax = b, \quad \text{where} \quad A = I - BJ_p g(p, u, c).$$

The integration matrix B is dense, so A is dense.

We collocate at the Chebyshev points

$$c_k = \cos\left(\frac{k-1}{n-1}\pi\right).$$



Hence we can

"apply the integral operator [B and hence A] in $\mathcal{O}(n \log n)$ operations, making iterative methods more attractive".

A degree $n-1$ polynomial expands as

$$p_n(t) = \sum_{j=0}^{n-1} a_j T_j(t),$$

where T_j is the jth Chebyshev polynomial:

$$T_j(t) = \cos(j \operatorname{arccos}(t)).$$

$B : \mathbb{R}^n \to \mathbb{R}^n$

1. interpolate the values $v \in \mathbb{R}^n$ on c	IFFT
2. integrate this interpolant	$\mathcal{O}(n)$
3. evaluate the integral on c	FFT

"Unfortunately, for many situations of interest, complex behavior of the solution causes ... the number of [solver] iterations to be large".

Ingredient 2: Preconditioning

To reduce the number of iterations of GMRES, we right precondition:

We choose an easy to invert preconditioner P, and then solve

$$AP^{-1}y = b$$

followed by

$$Px = y.$$

This x solves $Ax = b$ since $AP^{-1}P = A$.

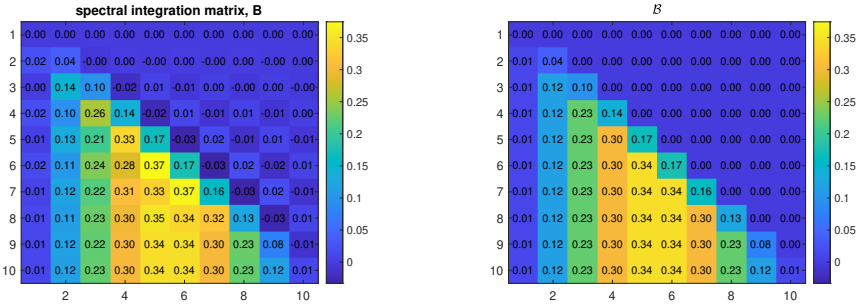
For this strategy to be effective, P should be a "good approximation" of A.

$$\frac{\|Ax_k - b\|}{\|b\|} \leq \inf_{p_k \in P_k} \|p_k(A)\|$$

A Preconditioner Based on B

As $n \to \infty$, the integration matrix B converges to indefinite integration $I(\xi)(t) = \int_0^t \xi(s)ds$ – the continuous analogue of a cumulative sum:

To compute $I(\xi)$, continually add the increment $\xi(t)ds$ to the running tally $\int_0^t \xi(s)ds$



Each subdiagonal column of B_L is constant, equalling a quadrature weight. Hence preconditioner $P = I - B_L\big(J_p g(p, u, c)\big)$ requires only $\mathcal{O}(n)$ operations to invert.

Numerical Results

ALESQP

We solve an orbit transfer problem using an augmented Lagrangian SQP algorithm (ALESQP) [1].

We can think of SQP as Newton's method applied to first-order optimality conditions.

The key computational kernel of ALESQP is a linear solve $\tilde{A}\tilde{x} = \tilde{b}$, where

$$\tilde{A} = \begin{pmatrix} R_x & 0 & A^T \\ 0 & R_u & C^T \\ A & C & 0 \end{pmatrix}.$$

Here, R_x and R_u are diagonal and again $A = I - BJ_p g(p, u, c)$.[1]

From our preconditioner P, we build an approximate *Schur complement preconditioner* [5] for $\tilde{A}$:

$$\tilde{P} = \begin{pmatrix} R_x & 0 & 0 \\ 0 & R_u & 0 \\ 0 & 0 & S \end{pmatrix}, \quad S = PR_x^{-1}P^T. \tag{3}$$

[1] This solve, for example, is a means of computing the least-squares solution to $\begin{pmatrix} A^T \\ C^T \end{pmatrix} x = b$.

Numerics

Method 1: LU Factorizations

n	Run Time	Iterations		
		AL	SQP	CG
1025	425 seconds	11	30	241

- 96% of the run time is spent factoring instances of $\tilde{A}$.

Method 2: Unpreconditioned Matrix-Free

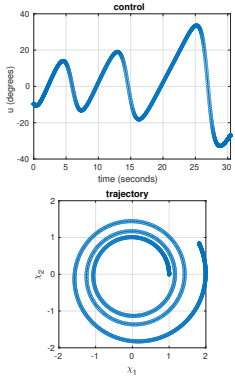
n	Run Time	Iterations		
		AL	SQP	CG
1025	1655 seconds	12	42	661

- $\tilde{A}$ is not well-conditioned, and unlike LU, we do not have a factorization to reuse for different solves with the same $\tilde{A}$.

Method 3: Preconditioned Matrix-Free

n	Run Time	Iterations		
		AL	SQP	CG
1025	73 seconds	10	34	421

The matrix-free approach we propose is not meant to address the *number* of optimization iterations – only to make those iterations *cheaper*.



thrust amplitude $a = 10^{-2}$

Numerics

Method 3: Preconditioned Matrix-Free

n	Run Time	Iterations		
		AL	SQP	CG
1025	73 seconds	10	34	421

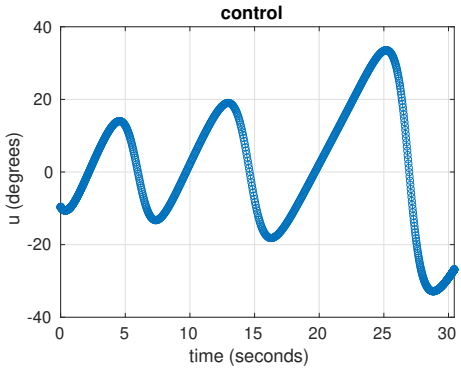
# of Linear Solver Iterations	4575
# of linear Solver Calls	511
Average # of Iterations per Call	8.95

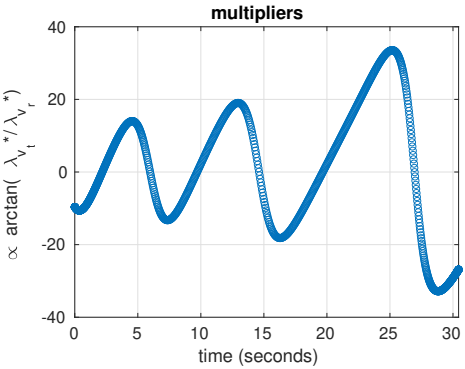
n	Run Time	Iterations		
		AL	SQP	CG
4097	251 seconds	10	32	500

# of Linear Solver Iterations	4847
# of linear Solver Calls	585
Average # of Iterations per Call	8.29

- Matrix factorization is $\mathcal{O}(n^3)$, meaning LU will take roughly 64 times longer for $n = 4097$ (about 7 hours). Our approach, however, takes approximately 4 times longer at this larger value of n.
- For all methods, we use scaled inner products to make the discretization consistent with the infinite dimensional problem. Indeed, we see that the number of AL and SQP iterations are roughly the same.
- The preconditioner P becomes more accurate as n gets larger, which we see in the numerics. The average number of solver iterations per call decreased.

Lagrange Multipliers





Conclusions

- We developed matrix-free solves for large spectral collocation discretizations. These solves scale for an orbit transfer example.
- The two key ingredients are a new preconditioner and a known result: fast FFT MVMs with the integration matrix B.
- Thank you!

References

H. Antil, D. P. Kouri, and D. Ridzal.
ALESQP: An augmented Lagrangian equality-constrained SQP method for optimization with general constraints.
SIAM Journal on Optimization, 33(1):237–266, 2023.

L. Greengard.
Spectral integration and two-point boundary value problems.
SIAM Journal on Numerical Analysis, 28(4):1071–1080, 1991.

E. Hairer, S. P. Nørsett, and G. Wanner.
Solving Ordinary Differential Equations I, Nonstiff Problems.
Springer, 2nd edition, 1993.

N. Koeppen, I. M. Ross, L. C. Wilcox, and R. J. Proulx.
Fast mesh refinement in pseudospectral optimal control.
Journal of Guidance, Control, and Dynamics, 42(4):711–722, 2019.

T. Rees, H. S. Dollar, and A. J. Wathen.
Optimal solvers for PDE-constrained optimization.
SIAM Journal on Scientific Computing, 32(1):271–298, 2010.

Y. Saad and M. H. Schultz.
GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems.
SIAM Journal on Scientific and Statistical Computing, 7(3):856–869, 1986.