Scalable Quantum Control with Semi-Automatic Differentiation

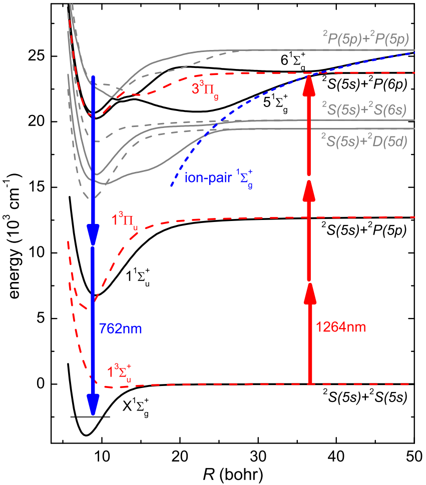
Michael H. Goerz, Sebastián C. Carrasco, Vladimir S. Malinovsky

DEVCOM Army Research Lab

CQE Workshop on Scalable Quantum Control

August 15, 2022

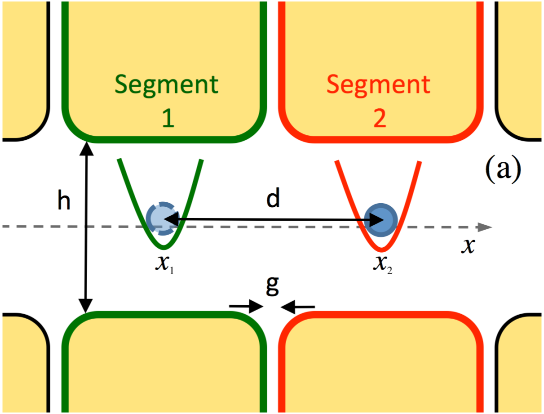
Optimal Control Tasks



Tomza *et al.* Phys. Rev. A 86, 043424 (2012)

- Photoassociation

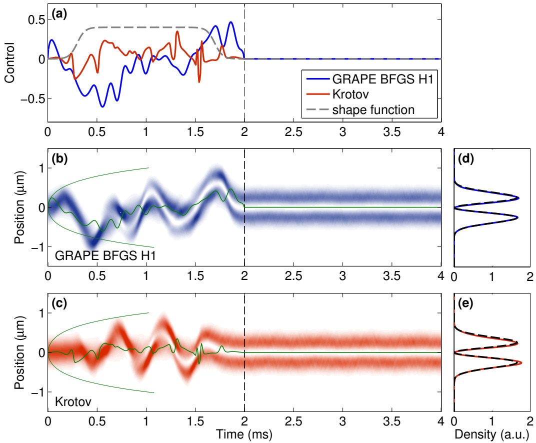
Optimal Control Tasks



Fürst *et al.* New J. Phys. 16, 075007 (2014)

- Photoassociation
- Ion transport

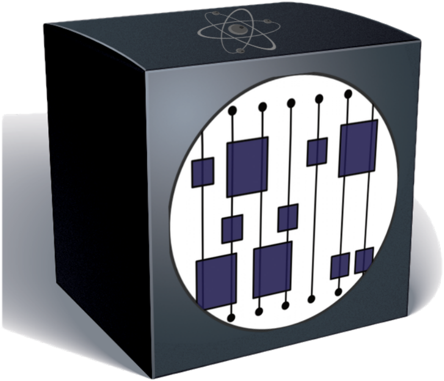
Optimal Control Tasks



Jäger *et al.* Phys. Rev. A 90, 033628 (2014)

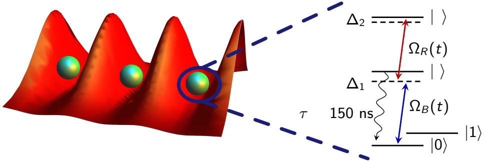
- Photoassociation
- Ion transport
- BEC wave function splitting

Optimal Control Tasks



- Photoassociation
- Ion transport
- BEC wave function splitting
- Quantum gates

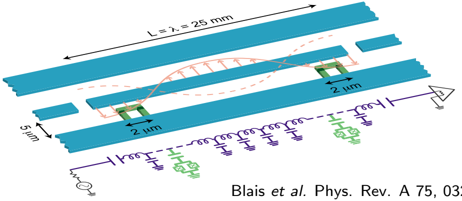
Optimal Control Tasks



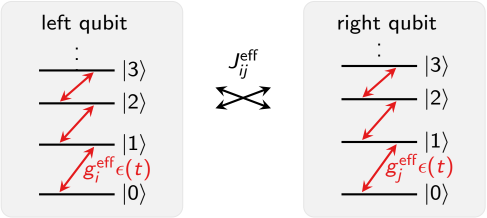
Goerz *et al.* Phys. Rev. A 90, 032329 (2014)

- Photoassociation
- Ion transport
- BEC wave function splitting
- Quantum gates
    - Rydberg atoms

Optimal Control Tasks



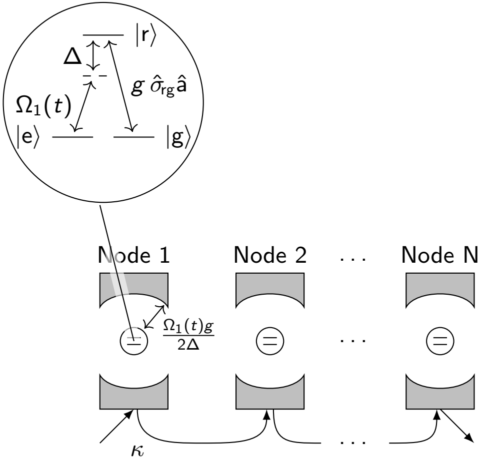
Blais *et al.* Phys. Rev. A 75, 032329 (2007)



Goerz *et al.* EPJ Quantum Tech. 2, 21 (2015)
Goerz *et al.* npj Quantum Information 3, 37 (2017)

- Photoassociation
- Ion transport
- BEC wave function splitting
- Quantum gates
    - Rydberg atoms
    - Superconducting qubits

Optimal Control Tasks

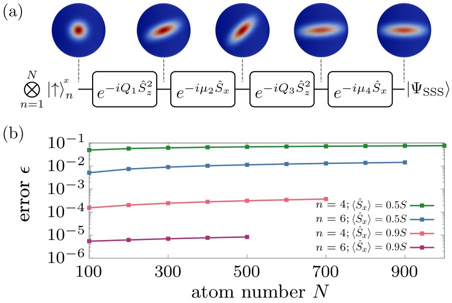


Goerz, Jacobs. Qu. Sci. Technol. 3, 045005 (2018)

- Photoassociation
- Ion transport
- BEC wave function splitting
- Quantum gates
  - Rydberg atoms
  - Superconducting qubits
- Entanglement in quantum networks

UNCLASSIFIED

Optimal Control Tasks



Carrasco *et al*, Phys. Rev. Applied 17, 064050 (2022)

- Photoassociation
- Ion transport
- BEC wave function splitting
- Quantum gates
  - Rydberg atoms
  - Superconducting qubits
- Entanglement in quantum networks
- Spin-squeezed states

Quantum Control Problem

"Pulse-level" control

- Bunch of states: $\{|\Psi_k(t)\rangle\}$
  — e.g. two-qubit gate: $|00\rangle$, $|01\rangle$, $|10\rangle$, $|11\rangle$
- Hamiltonian(s) with control fields: $\hat{\mathrm{H}}_k(\{\epsilon_l(t)\})$ $\rightarrow$ time propagation
  — assume piecewise-constant: ϵ_{ln} for n'th time interval of l'th control

Functional

$$J(\{\epsilon_{nl}\}) = J_T(\{|\Psi_k(T)\rangle\}) + \int_0^T g_a(\{\epsilon_l(t)\}, t)\,\mathrm{d}t + \int_0^T g_b(\{|\Psi_k(t)\rangle\}, t)\,\mathrm{d}t$$

Gradient-based "open loop" optimization

$$(\nabla J)_{ln} \equiv \frac{\partial J}{\partial \epsilon_{ln}} \qquad \Rightarrow \qquad \text{L-BFGS-B}$$

Scalability

Driving cutting-edge quantum technology with optimal control?

- Bigger (open) systems — hard numerics
- More flexibility — better functionals, novel methods

Efficient Quantum Control

1984

An accurate and efficient scheme for propagating the time dependent Schrödinger equation

1986

Coherent pulse sequence induced control of selectivity of reactions: Exact quantum mechanical calculations

1988

J. Phys. Chem. **1988**, *92*, 2087–2100 **2087**

Time-Dependent Quantum-Mechanical Methods for Molecular Dynamics

1992

J. Phys. A: Math. Gen. **25** (1992) 1283–1307. Printed in the UK

Solution of the time-dependent Liouville–von Neumann

1994

Annu. Rev. Phys. Chem. 1994. 45: 145–78
Copyright © 1994 by Annual Reviews Inc. All rights reserved

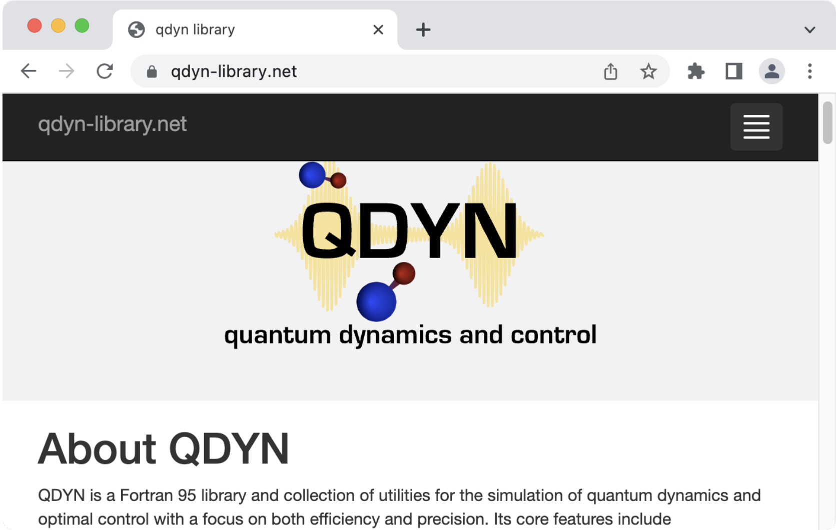
PROPAGATION METHODS FOR QUANTUM MOLECULAR DYNAMICS

Ronnie Kosloff

Efficient Quantum Control

- Get your data structures right
  — grid representation (FFT), sparsity
- Get your propagation right
  — polynomial expansions, in-place BLAS
- Set up simultaneous "objectives" via states
  — parallelization

QDYN



C. Koch group
FU Berlin

Fortran

Scalability

Driving cutting-edge quantum technology with optimal control?

- Bigger (open) systems — hard numerics
- More flexibility — better functionals, novel methods

Automatic Differentiation (AD)

PHYSICAL REVIEW A **95**, 042318 (2017)

Speedup for quantum optimal control from automatic differentiation based on graphics processing units

Nelson Leung,[1,*] Mohamed Abdelhafez,[1] Jens Koch,[2] and David Schuster[1]

PHYSICAL REVIEW A **99**, 052327 (2019)

Gradient-based optimal control of open quantum systems using quantum trajectories and automatic differentiation

Mohamed Abdelhafez,[1,*] David I. Schuster,[1] and Jens Koch[2]

PHYSICAL REVIEW A **101**, 022321 (2020)

Universal gates for protected superconducting qubits using optimal control

Mohamed Abdelhafez,[1] Brian Baker,[2] András Gyenis,[3] Pranav Mundada,[3] Andrew A. Houck,[3]
David Schuster,[1] and Jens Koch[2]

Automatic Differentiation (AD)

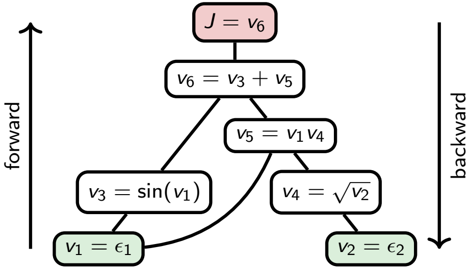
backward-mode "adjoint"

$$\bar{v}_j \equiv \frac{\partial J}{\partial v_j} = \sum_i \bar{v}_i \frac{\partial v_i}{\partial v_j}$$

sum over all v_i

which depend on v_j

$J(\epsilon_1, \epsilon_2) = \sin(\epsilon_1) + \epsilon_1 \sqrt{\epsilon_2}$



Automatic Differentiation (AD)

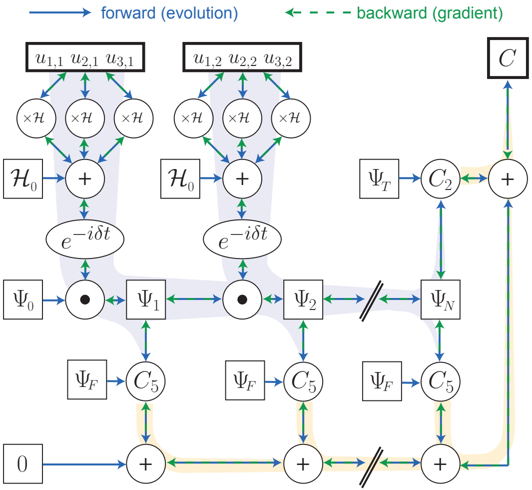


Fig. 2 in Leung *et al.* Phys. Rev. A 95, 042318 (2017)

AD Advantages

- Arbitrary functionals

μ	Cost-function contribution	$C_\mu(\mathbf{u})$
1	Target-gate infidelity	$1 - \lvert \operatorname{tr}(K_T^\dagger K_N)/D\rvert^2$
2	Target-state infidelity	$1 - \lvert\langle\Psi_T\vert\Psi_N\rangle\rvert^2$
3	Control amplitudes	$\lvert\mathbf{u}\rvert^2$
4	Control variations	$\sum_{j,k} \lvert u_{k,j} - u_{k,j-1}\rvert^2$
5	Occupation of forbidden state	$\sum_j \lvert\langle\Psi_F\vert\Psi_j\rangle\rvert^2$
6	Evolution time (target gate)	$1 - \frac{1}{N}\sum_j \lvert \operatorname{tr}(K_T^\dagger K_j)/D\rvert^2$
7	Evolution time (target state)	$1 - \frac{1}{N}\sum_j \lvert\langle\Psi_T\vert\Psi_j\rangle\rvert^2$

Table 1 in Leung *et al.* Phys. Rev. A 95, 042318 (2017)

- Arbitrary equations of motion
  e.g., quantum trajectories — Abdelhafez *et al.* Phys. Rev. A 99, 052327 (2019)
- GPU support

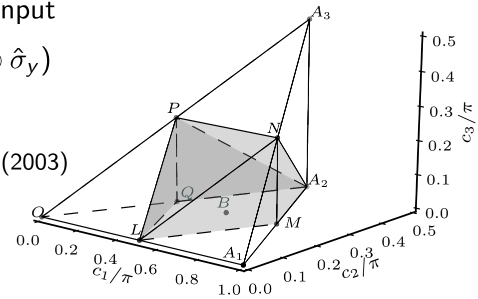
Entanglement Measures

Quantum Gate Concurrence

Max concurrence that can be generated for a separable input

1. $c_1, c_2, c_3 \propto \text{eigvals}\left(\hat{\mathrm{U}}\tilde{\mathrm{U}}\right)$; $\quad \tilde{\mathrm{U}} = (\hat{\sigma}_y \otimes \hat{\sigma}_y)\,\hat{\mathrm{U}}\,(\hat{\sigma}_y \otimes \hat{\sigma}_y)$
2. $C(\hat{\mathrm{U}}) = \max\left|\sin(c_{1,2,3} \pm c_{3,1,2})\right|$

Childs *et al.* Phys. Rev. A 68, 052311 (2003)



Not analytic!

UNCLASSIFIED

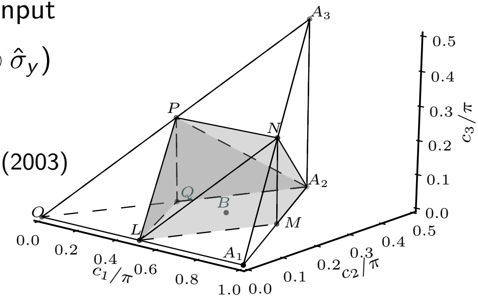
Entanglement Measures

Quantum Gate Concurrence

Max concurrence that can be generated for a separable input

1. $c_1, c_2, c_3 \propto \text{eigvals}\left(\hat{\mathrm{U}}\tilde{\mathrm{U}}\right)$; $\quad \tilde{\mathrm{U}} = (\hat{\sigma}_y \otimes \hat{\sigma}_y)\,\hat{\mathrm{U}}\,(\hat{\sigma}_y \otimes \hat{\sigma}_y)$
2. $C(\hat{\mathrm{U}}) = \max\left|\sin(c_{1,2,3} \pm c_{3,1,2})\right|$

Childs *et al.* Phys. Rev. A 68, 052311 (2003)



Perfect Entanglers Functional

Find a two-qubit gate with maximum entangling power

$$
\begin{aligned}
F_{PE} = &\left(\frac{1}{\det U_B}\right)\left(\frac{1}{4}(\mathrm{tr}^2[U_B^T U_B] - \mathrm{tr}[U_B^T U_B U_B^T U_B])\right)\left(\frac{1}{16}\mathrm{Re}^2[\mathrm{tr}[U_B^T U_B]]\right) + \\
&+\left(\frac{2}{\det U_B}\right)\left(\frac{1}{4}(\mathrm{tr}^2[U_B^T U_B] - \mathrm{tr}[U_B^T U_B U_B^T U_B])\right)\left(\frac{1}{16}\mathrm{Im}^2[\mathrm{tr}[U_B^T U_B]]\right) \\
&\left(\frac{1}{16}\mathrm{Re}[\mathrm{tr}^2[U_B^T U_B]]\right)
\end{aligned}
$$

U_B: projection into logical subspace, in Bell basis

Watts *et al.* Phys. Rev. A 91, 062306 (2015)
Goerz *et al.* Phys. Rev. A 91, 062307 (2015)

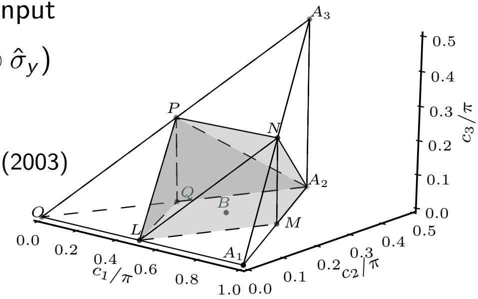
Entanglement Measures

Quantum Gate Concurrence

Max concurrence that can be generated for a separable input

1. $c_1, c_2, c_3 \propto \text{eigvals}\left(\hat{\mathsf{U}}\tilde{\mathsf{U}}\right)$; $\quad \tilde{\mathsf{U}} = (\hat{\sigma}_y \otimes \hat{\sigma}_y)\,\hat{\mathsf{U}}\,(\hat{\sigma}_y \otimes \hat{\sigma}_y)$
2. $C(\hat{\mathsf{U}}) = \max\left|\sin(c_{1,2,3} \pm c_{3,1,2})\right|$

Childs *et al.* Phys. Rev. A 68, 052311 (2003)



To a computer, everything is analytic!

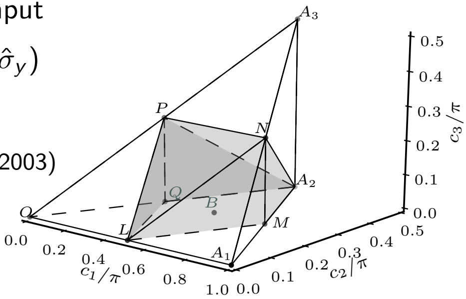
Entanglement Measures

Quantum Gate Concurrence

Max concurrence that can be generated for a separable input

1. $c_1, c_2, c_3 \propto \text{eigvals}\left(\hat{U}\tilde{U}\right)$; $\tilde{U} = (\hat{\sigma}_y \otimes \hat{\sigma}_y)\,\hat{U}\,(\hat{\sigma}_y \otimes \hat{\sigma}_y)$
2. $C(\hat{U}) = \max \left|\sin(c_{1,2,3} \pm c_{3,1,2})\right|$

Childs *et al.* Phys. Rev. A 68, 052311 (2003)



To a computer, everything is analytic!

Quantum Fisher Information

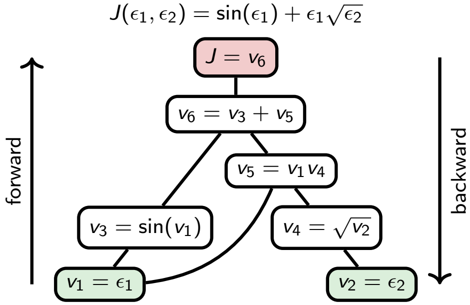
$$F(\hat{\rho}) = \sum_{i \neq j} \frac{2(p_i - p_j)^2}{p_i + p_j} \left|\left\langle \phi_i \middle| \hat{S}_z \middle| \phi_j \right\rangle\right|^2$$

where p_i, $|\phi_i\rangle$ are eigenvalues / eigenstates of $\hat{\rho}$

— Ma *et al.*. Phys. Rep. 509, 89 (2011)

AD Compromises

1. Numerical scaling
   - AD memory overhead
   - computational overhead (at least on CPU)
2. Framework limitations
   - Complex numbers?
   - In-place operations?
   - Double-precision?
3. Code reuse
   - Re-implement propagation methods?
   - Re-use existing GRAPE implementation?



We don't have to compromise!

Semi-Automatic Differentiation

arXiv:2205.15044

Quantum Optimal Control via Semi-Automatic Differentiation

Michael H. Goerz, Sebastián C. Carrasco, and Vladimir S. Malinovsky

DEVCOM Army Research Laboratory, 2800 Powder Mill Road, Adelphi, MD 20783, USA

ant-ph] 27 May 2022

We develop a framework of "semi-automatic differentiation" that combines existing gradient-based methods of quantum optimal control with automatic differentiation. The approach allows to optimize practically any computable functional and is implemented in two open source Julia packages, `GRAPE.jl` and `Krotov.jl`, part of the `QuantumControl.jl` framework. Our method is based on formally rewriting the optimization functional in terms of propagated states, overlaps with target states, or quantum gates. An analytical application of the chain rule then allows to separate the time propagation and the evaluation of the functional when calculating the gradient. The former can be evaluated with great efficiency via a modified GRAPE scheme. The latter is evaluated with automatic differenti-

Funding

DEVCOM Army Research Laboratory, Cooperative Agreement Numbers W911NF-16-2-0147, W911NF-21-2-0037; DIRA-TRC No. DTR19-CI-019

arXiv:2205.15044

Semi-Automatic Differentiation

$$J(\{\epsilon_{nl}\}) = J_T(\{\psi_k(T)\}) + \dots$$

$$\nabla J = \frac{\partial J_T}{\partial \epsilon_{nl}}$$

$$\frac{\partial J_T}{\partial \epsilon_{nl}} = 2\,\mathrm{Re}\left[\sum_k \underbrace{\frac{\partial J_T}{\partial |\psi_k(T)\rangle}}_{\langle \chi_k(T)|} \cdot \frac{\partial |\psi_k(T)\rangle}{\epsilon_{nl}}\right] \; ; \quad |\chi_k(T)\rangle = \frac{\partial J_T}{\partial \langle \psi_k(T)|}$$

$$= 2\,\mathrm{Re}\left[\sum_k \frac{\partial}{\partial \epsilon_{nl}} \langle \chi_k(T) | \psi_k(T)\rangle\right]$$

arXiv:2205.15044

Semi-Automatic Differentiation

$$\frac{\partial}{\partial \epsilon_{nl}} \langle \chi_k(T) | \hat{U}_N \dots \hat{U}_1 | \psi_k(0) \rangle \quad ; \quad \hat{U}_n = e^{-i \hat{H}_n dt}$$

$$= \underbrace{\langle \chi_k(T) | \hat{U}_N \dots \hat{U}_{n+1}}_{\text{bw}} \frac{\partial \hat{U}_n}{\partial \epsilon_{nl}} \underbrace{\hat{U}_{n-1} \dots \hat{U}_1 | \psi_k(0) \rangle}_{\text{fw}}$$

$$|\chi_k(T)\rangle = \frac{\partial J_T}{\partial \langle \psi_k(T) |}$$

$$J_T(\hat{A}) \qquad \hat{A}_{ik} = \langle \phi_i | \psi_k \rangle$$

$$\frac{\partial J_T}{\partial \langle \psi_k(T) |} = \sum_i \frac{\partial J_T}{\partial u_{ik}} |\phi_i\rangle$$

Gradient of Time Evolution Operator

$$
\begin{pmatrix}
\frac{\partial \hat{\mathrm{U}}_n^\dagger}{\partial \epsilon_{n1}} \vert \chi_k(t_n) \rangle \\
\vdots \\
\frac{\partial \hat{\mathrm{U}}_n^\dagger}{\partial \epsilon_{nL}} \vert \chi_k(t_n) \rangle \\
\hat{\mathrm{U}}_n^\dagger \vert \chi_k(t_n) \rangle
\end{pmatrix}
= \exp \left[-\mathrm{i}
\begin{pmatrix}
\hat{\mathrm{H}}_n^\dagger & 0 & \dots & 0 & \hat{\mathrm{H}}_n^{(1)\dagger} \\
0 & \hat{\mathrm{H}}_n^\dagger & \dots & 0 & \hat{\mathrm{H}}_n^{(2)\dagger} \\
\vdots & & \ddots & & \vdots \\
0 & 0 & \dots & \hat{\mathrm{H}}_n^\dagger & \hat{\mathrm{H}}_n^{(L)\dagger} \\
0 & 0 & \dots & 0 & \hat{\mathrm{H}}_n^\dagger \,,
\end{pmatrix}
dt_n \right]
\begin{pmatrix}
0 \\
\vdots \\
0 \\
\vert \chi_k(t_n) \rangle
\end{pmatrix}
$$

$$
\hat{\mathrm{U}}_n = \exp[-\mathrm{i}\hat{\mathrm{H}}_n dt_n]\,; \qquad \hat{\mathrm{H}}_n^{(l)} = \frac{\partial \hat{\mathrm{H}}_n}{\partial \epsilon_l(t)}
$$

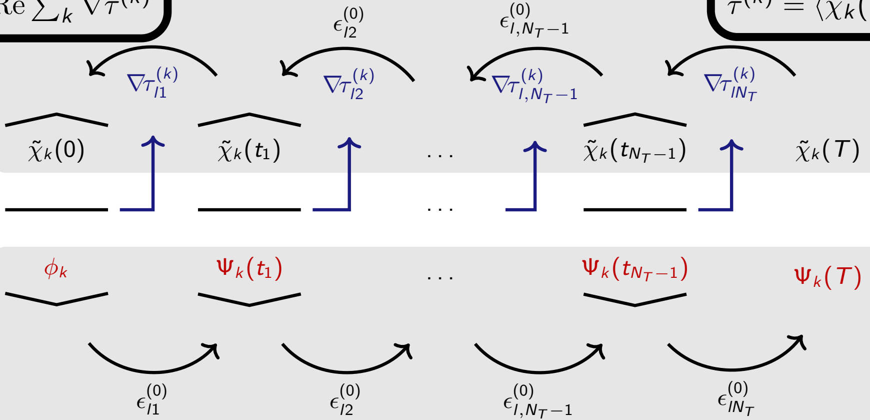
— Goodwin, Kuprov, J. Chem. Phys. 143, 084113 (2015)

Generalized GRAPE scheme

$\boxed{\nabla J_T = 2\mathrm{Re}\sum_k \nabla\tau^{(k)}}$

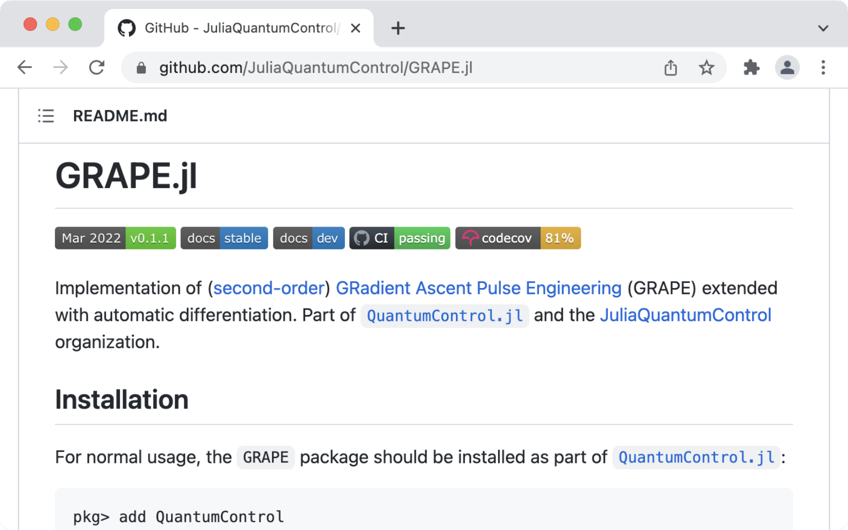
② backward-prop of extended state/gradient

$\boxed{\tau^{(k)} = \langle \chi_k(T) \,|\, \Psi_k(T) \rangle}$



① forward-prop and storage with guess

GRAPE.jl


GitHub - JuliaQuantumControl/

github.com/JuliaQuantumControl/GRAPE.jl

README.md

GRAPE.jl

Mar 2022 v0.1.1 | docs stable | docs dev | CI passing | codecov 81%

Implementation of (second-order) GRadient Ascent Pulse Engineering (GRAPE) extended with automatic differentiation. Part of `QuantumControl.jl` and the JuliaQuantumControl organization.

Installation

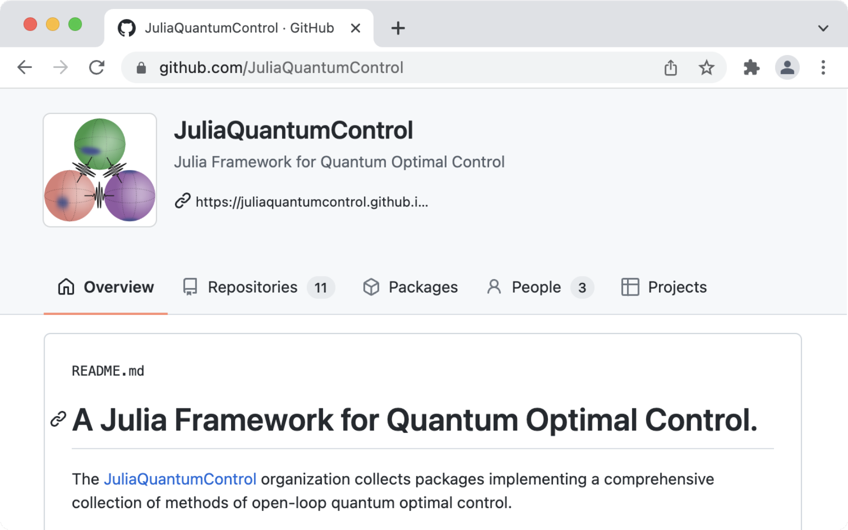
For normal usage, the `GRAPE` package should be installed as part of `QuantumControl.jl`:

```
pkg> add QuantumControl
```

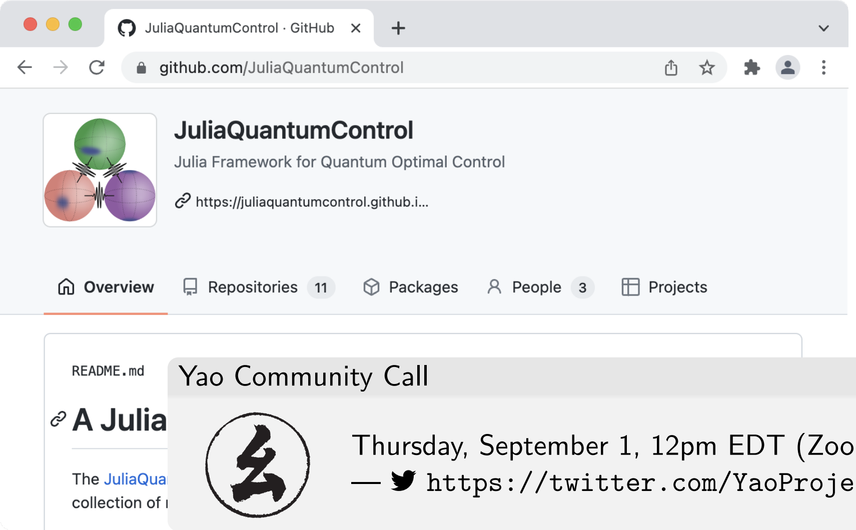
julia

Zygote

JuliaQuantumControl



JuliaQuantumControl



julia

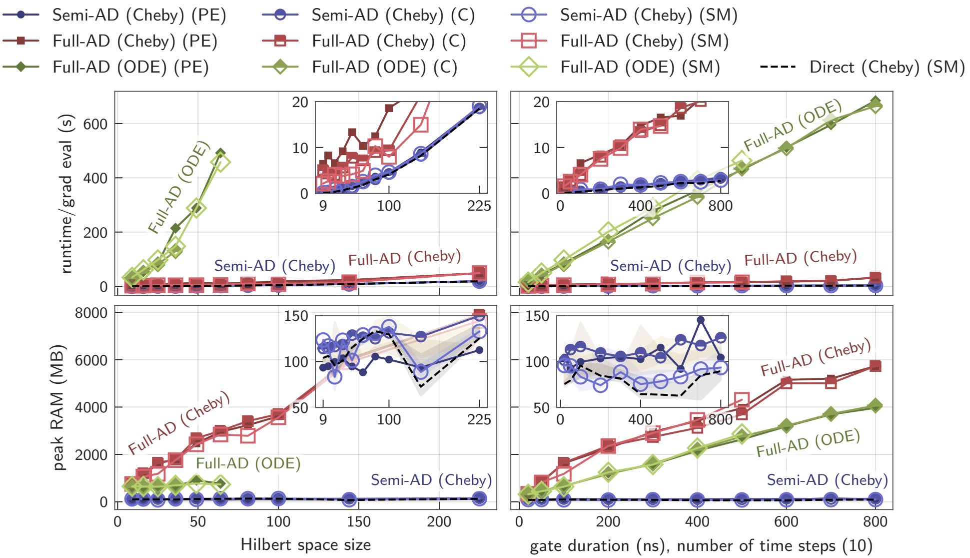
Zygote

Yao Community Call

Thursday, September 1, 12pm EDT (Zoom)

— https://twitter.com/YaoProject

Benchmarks



Semi-AD (Cheby) (PE)
Semi-AD (Cheby) (C)
Semi-AD (Cheby) (SM)
Full-AD (Cheby) (PE)
Full-AD (Cheby) (C)
Full-AD (Cheby) (SM)
Full-AD (ODE) (PE)
Full-AD (ODE) (C)
Full-AD (ODE) (SM)
Direct (Cheby) (SM)
runtime/grad eval (s)
peak RAM (MB)
Full-AD (ODE)
Semi-AD (Cheby)
Full-AD (Cheby)
Hilbert space size
gate duration (ns), number of time steps (10)

Conclusion

AD-enhanced optimal control without compromises! arXiv:2205.15044

- Use optimal data structures
- Use polynomial in-place propagators
- Use semi-AD implementation of GRAPE

- propagation and optimal control are independent
- AD and GPU computing are independent
- Full power of AD with near-zero overhead

Thank you