Time integration for MCTDH

Christian Lubich
Univ. Tübingen

Exploiting New Advances in Mathematics to Improve Calculations in Quantum Molecular Dynamics,
BIRS, Banff, 25 January 2016

Talk based on: Time integration in the multiconfiguration time-dependent Hartree method of molecular quantum dynamics, Appl. Math. Res. Express 2015, 311-328.

Outline

MCTDH recap

Projector-splitting integrator for the matrix case

MCTDH integrator

Outline

MCTDH recap

Projector-splitting integrator for the matrix case

MCTDH integrator

Setting

Time-dependent Schrödinger equation

$$i\frac{\partial \psi}{\partial t} = \mathcal{H}\psi$$

for the wavefunction $\psi = \psi(x_1, \dots, x_d, t)$

MCTDH: model reduction via low-rank tensor approximation

Meyer, Manthe & Cederbaum 1990 (first MCTDH paper)
Meyer, Gatti & Worth 2009 (MCTDH book)

Galerkin method / Full configuration interaction

With L^2-orthonormal basis functions φ_j^n (for $j = 1, \ldots, K$ in each mode $n = 1, \ldots, d$), approximate the wave function by

$$\psi(x_1, \ldots, x_d, t) \approx \sum_{i_1=1}^{K} \cdots \sum_{i_d=1}^{K} a_{i_1,\ldots,i_d}(t)\, \varphi_{i_1}^1(x_1) \ldots \varphi_{i_d}^d(x_d),$$

where the time-dependent coefficient tensor

$$A(t) = (a_{i_1,\ldots,i_d}(t)) \in \mathbb{C}^{K \times \cdots \times K} = \mathbb{C}^{K^d}$$

satisfies a linear tensor differential equation

$$i\dot{A}(t) = H[A(t)]$$

with a discrete Hamiltonian $H : \mathbb{C}^{K^d} \to \mathbb{C}^{K^d}$.

This system is not directly tractable because of its sheer size.

Tucker tensor format

Approximate, with $r \ll K$,

$$a_{i_1,\dots,i_d} \approx \sum_{j_1=1}^{r} \cdots \sum_{j_d=1}^{r} c_{j_1,\dots,j_d}\, u^1_{i_1,j_1} \dots u^d_{i_d,j_d}.$$

Single-particle matrices $\boldsymbol{U}_n = (u^n_{ij}) \in \mathbb{C}^{K \times r}$ (for $n = 1, \dots, d$) have orthonormal columns $\boldsymbol{u}^n_j \in \mathbb{C}^K$.

Core tensor $C = (c_{j_1,\dots,j_d}) \in \mathbb{C}^{r \times \cdots \times r}$.

Storage is reduced from K^d to $r^d + dKr$ entries.

Shorthand tensor notation

$$A \approx Y = C \times_1 \boldsymbol{U}_1 \cdots \times_d \boldsymbol{U}_d = C \bigtimes_{n=1}^{d} \boldsymbol{U}_n.$$

MCTDH

The MCTDH method combines

- low-rank tensor approximation in the Tucker format with the
- Dirac-Frenkel time-dependent variational principle.

MCTDH

$\mathcal{M}_r =$ manifold of all Tucker tensors where each single-mode matrix unfolding of the core tensor is of full rank r

$T_Y\mathcal{M}_r =$ tangent space of $\mathcal{M}_r$ at $Y \in \mathcal{M}_r$

Approximate $A(t) \approx Y(t) \in \mathcal{M}_r$ by

$$\left\langle i\dot{Y}(t) - H[Y(t)], \delta Y \right\rangle = 0 \qquad \text{for all } \ \delta Y \in T_{Y(t)}\mathcal{M}_r.$$

With the orthogonal projection $P(Y)$ onto the tangent space $T_Y\mathcal{M}_r$, this can be equivalently stated as

$$i\dot{Y}(t) = P(Y(t))\, H[Y(t)].$$

MCTDH equations of motion

$$i\dot{C} = H[Y] \bigtimes_{n=1}^{d} \boldsymbol{U}_n^*$$

$$i\dot{\boldsymbol{U}}_n = (\boldsymbol{I} - \boldsymbol{U}_n\boldsymbol{U}_n^*)\,\mathrm{mat}_n\big(H[Y] \bigtimes_{k\neq n} \boldsymbol{U}_n^*\big)\, \boldsymbol{C}_{(n)}^+$$

with the pseudo-inverse $\boldsymbol{C}_{(n)}^+ = \boldsymbol{C}_n^*\big(\boldsymbol{C}_{(n)}\boldsymbol{C}_{(n)}^*\big)^{-1}$ of the n-mode matricization of the core tensor $\boldsymbol{C}_{(n)} = \mathrm{mat}_n(C)$, and with

$$Y(t) = C(t) \bigtimes_{n=1}^{d} \boldsymbol{U}_n(t),$$

which is taken as the approximation to $A(t)$.

These differential equations need to be solved numerically.

Ill-conditioned MCTDH density matrices

MCTDH equations contain the inverse of the density matrices

$$\boldsymbol{\rho}_n = \boldsymbol{C}_{(n)} \boldsymbol{C}_{(n)}^*.$$

These matrices are typically ill-conditioned. This leads to a severe stepsize restriction with usual numerical integrators.

Ad-hoc remedy: regularization of the density matrices $\boldsymbol{\rho}_n = \boldsymbol{C}_{(n)} \boldsymbol{C}_{(n)}^*$ to $\boldsymbol{\rho}_n + \sigma^2 \boldsymbol{I}$ with a not too small $\sigma > 0$.

Novelty in this talk: Numerical integrator for the MCTDH equations of motion which can use stepsizes that are not restricted by ill-conditioned density matrices, without any regularization.

MCTDH integrator

A step of the integrator alternates between

- orthogonal matrix decompositions and
- solving linear systems of differential equations (by Lanczos).

The MCTDH density matrices are nowhere computed, nor are their inverses.

Outline

MCTDH recap

Projector-splitting integrator for the matrix case

MCTDH integrator

Equivalent formulations of dynamical low-rank approximation

- $\dot{Y} \in T_Y\mathcal{M}_r$ such that $\|\dot{Y} - \dot{A}\| = \min!$
- $\langle \dot{Y} - \dot{A}, \delta Y \rangle = 0$ for all $\delta Y \in T_Y\mathcal{M}_r$
- $\dot{Y} = P(Y)\dot{A}$ with $P(Y) =$ orth. projection onto $T_Y\mathcal{M}_r$:

$$P(Y)\dot{A} = \dot{A}P_{\mathcal{R}(Y^T)} - P_{\mathcal{R}(Y)}\dot{A}P_{\mathcal{R}(Y^T)} + P_{\mathcal{R}(Y)}\dot{A}$$

Idea: split the projection

L. & Oseledets 2014

Splitting integrator, abstract form

1. Solve the differential equation

$$\dot{Y}_I = \dot{A} P_{\mathcal{R}(Y_I^T)}$$

   with initial value $Y_I(t_0) = Y_0$ for $t_0 \le t \le t_1$.

2. Solve

$$\dot{Y}_{II} = -P_{\mathcal{R}(Y_{II})} \dot{A} P_{\mathcal{R}(Y_{II}^T)}$$

   with initial value $Y_{II}(t_0) = Y_I(t_1)$ for $t_0 \le t \le t_1$.

3. Solve

$$\dot{Y}_{III} = P_{\mathcal{R}(Y_{III})} \dot{A}$$

   with initial value $Y_{III}(t_0) = Y_{II}(t_1)$ for $t_0 \le t \le t_1$.

Finally, take $Y_1 = Y_{III}(t_1)$ as an approximation to $Y(t_1)$.

Solving the split differential equations

Write rank-r matrix $Y \in \mathbb{C}^{m\times n}$ (non-uniquely) as

$$Y = USV^*$$

where $U \in \mathbb{C}^{m\times r}$ and $V \in \mathbb{C}^{n\times r}$ have orthonormal columns, and $S \in \mathbb{C}^{r\times r}$. Then, the projection becomes

$$P(Y)\dot{A} = \dot{A}VV^* - UU^*\dot{A}VV^* + UU^*\dot{A}.$$

The solution of 1. is given by

$$Y_I = U_I S_I V_I^T \quad \text{with} \quad (U_I S_I)^{\dot{}} = \dot{A}V_I, \quad \dot{V}_I = 0:$$

$$U_I(t)S_I(t) = U_I(t_0)S_I(t_0) + (A(t) - A(t_0))V_I(t_0), \qquad V_I(t) = V_I(t_0)$$

and similarly for 2. and 3.

Splitting integrator, practical form

Start from $Y_0 = U_0 S_0 V_0^T \in \mathcal{M}_r$.

1. With the increment $\Delta A = A(t_1) - A(t_0)$, set

   $$K_1 = U_0 S_0 + \Delta A\, V_0$$

   and orthogonalize:

   $$K_1 = U_1 \widetilde{S}_1,$$

   where $U_1 \in \mathbb{R}^{m \times r}$ has orthonormal columns, and $\widetilde{S}_1 \in \mathbb{R}^{r \times r}$.
2. Set $\widetilde{S}_0 = \widetilde{S}_1 - U_1^T\, \Delta A\, V_0$.
3. Set $L_1 = V_0 \widetilde{S}_0^T + \Delta A^T U_1$ and orthogonalize:

   $$L_1 = V_1 S_1^T,$$

   where $V_1 \in \mathbb{R}^{n \times r}$ has orthonormal columns, and $S_1 \in \mathbb{R}^{r \times r}$.

The algorithm computes a factorization of the rank-r matrix

$$Y_1 = U_1 S_1 V_1^T \approx Y(t_1).$$

Splitting integrator, cont.

- Use symmetrized variant (Strang splitting)
- For a matrix differential equation $i\dot{A} = H[A]$:
  in substep 1. solve

  $$i\dot{K} = H[KV_0^T]V_0, \qquad K(t_0) = U_0 S_0$$

  by a step of a numerical method (e.g., Lanczos),
  and similarly in substeps 2. and 3.

ODEs for dynamical low-rank approximation

$$Y = USV^T$$

with

$$\begin{aligned}
\dot{U} &= (I_m - UU^T)\dot{A}VS^{-1} \\
\dot{V} &= (I_n - VV^T)\dot{A}^TUS^{-T} \\
\dot{S} &= U^T\dot{A}V
\end{aligned}$$

What if S is ill-conditioned? (effective rank smaller than r)

An exactness result for the splitting method

If $A(t)$ has rank r, then the splitting integrator is exact:

$$Y_1 = A(t_1)$$

Ordering of the splitting is essential! (KSL, not KLS)

Approximation is robust to small singular values

CL, Ivan Oseledets, A projector-splitting integrator for dynamical low-rank approximation, BIT 54 (2014), 171-188.

E. Kieri, CL, Hanna Walach, Discretized dynamical low-rank approximation in the presence of small singular values, Preprint 2015, submitted.

Remarks on the proof

The method splits $P(Y) = P_I(Y) - P_{II}(Y) + P_{III}(Y)$ in

$$\dot{Y} = P(Y)F(t, Y).$$

Difficulty: cannot use the Lipschitz continuity of the tangent space projection $P(\cdot)$ and its subprojections, because the Lipschitz constants become large for small singular values.

Rescue:

- use the previous exactness result
- use the conservation of the subprojections in the substeps

Outline

Relating back to the matrix case

The Tucker tensor $Y = C \bigtimes_{n=1}^{d} \boldsymbol{U}_n$ has the n-mode matrix unfolding

$$\boldsymbol{Y}_{(n)} = \boldsymbol{U}_n \boldsymbol{C}_{(n)} \bigotimes_{k \neq n} \boldsymbol{U}_k^\top,$$

where $\boldsymbol{C}_{(n)} \in \mathbb{C}^{r \times r^{d-1}}$ is the n-mode matrix unfolding of the core tensor $C \in \mathbb{C}^{r^d}$. We orthonormalize

$$\boldsymbol{C}_{(n)}^\top = \boldsymbol{Q}_n \boldsymbol{S}_n^\top,$$

where $\boldsymbol{Q}_n \in \mathbb{C}^{r^{d-1} \times r}$ has orthonormal columns, and $\boldsymbol{S}_n \in \mathbb{C}^{r \times r}$. On introducing $\boldsymbol{V}_n^\top = \boldsymbol{Q}_n^\top \bigotimes_{k \neq n} \boldsymbol{U}_k^\top$, we have, like in the matrix case,

$$\boldsymbol{Y}_{(n)} = \boldsymbol{U}_n \boldsymbol{S}_n \boldsymbol{V}_n^\top.$$

In MCTDH terminology, the columns of $\boldsymbol{V}_n$ represent an orthonormalized set of *single-hole functions*.

Tucker tensor tangent space projector

Tucker tensor $Y = C \bigtimes_{n=1}^{d} \boldsymbol{U}_n$ has the n-mode matrix unfolding

$$\boldsymbol{Y}_{(n)} = \boldsymbol{U}_n \boldsymbol{S}_n \boldsymbol{V}_n^\top,$$

where $\boldsymbol{S}_n \in \mathbb{C}^{r \times r}$, and $\boldsymbol{V}_n = \left(\bigotimes_{k \neq n} \boldsymbol{U}_k\right) \boldsymbol{Q}_n \in \mathbb{C}^{K^{d-1} \times r}$ has orthonormal columns. For $Z \in \mathbb{C}^{K \times \cdots \times K}$ and for $n = 1, \ldots, d$ we denote

$$P_n^+(Y)Z = \mathrm{ten}_n\left(\boldsymbol{Z}_{(n)} \overline{\boldsymbol{V}}_n \boldsymbol{V}_n^\top\right)$$
$$P_n^-(Y)Z = \mathrm{ten}_n\left(\boldsymbol{U}_n \boldsymbol{U}_n^* \boldsymbol{Z}_{(n)} \overline{\boldsymbol{V}}_n \boldsymbol{V}_n^\top\right)$$
$$P_0(Y)Z = Z \bigtimes_{n=1}^{d} \boldsymbol{U}_n \boldsymbol{U}_n^*.$$

Then, the orthogonal projection $P(Y)$ onto the tangent space $T_Y \mathcal{M}_r$ is given as

$$P(Y) = \sum_{n=1}^{d} \Big(P_n^+(Y) - P_n^-(Y)\Big) + P_0(Y).$$

MCTDH projector-splitting integrator

The splitting integrator that results from the above additive decomposition of the tangent space projection alternates between

- orthogonal matrix decompositions and
- solving linear systems of single-particle differential equations, which can be done efficiently by Lanczos approximations.

The splitting integrator can be implemented at a

- computational cost per time step that is about the same as for existing MCTDH integrators, but
- allowing for larger time steps
- without requiring any regularization.

The MCTDH density matrices are nowhere computed, nor are their inverses.

Implementation and extensions

- First implementation and tests by Benedikt Kloss (excellent master student with Irene Burghardt): Python implementation, compact code, observes good behaviour and speedup compared with MCTDH code
- Extension to multilayer MCTDH conceptually straightforward (hierarchical Tucker tensor format)
- Projector-splitting integrator for tensor trains (= MPS) in:

  CL, I. Oseledets, B. Vandereycken, *Time integration of tensor trains*, SIAM J. Numer. Anal. 53 (2015), 917-941.

  J. Haegeman, CL, I. Oseledets, B. Vandereycken, F. Verstraete, *Unifying time evolution and optimization with matrix product states*, arXiv:1408.5056.
- Extension to MCTDHF and MCTDHB for fermions/bosons feasible (needs yet to be done)

Propagation of the basis, forward loop

For $n = 1, \dots, d$ do the following:

1. For the n-mode matrix unfolding $\boldsymbol{C}_{(n)}^{0,n-1} \in \mathbb{C}^{r \times r^{d-1}}$ of the core tensor $C^{0,n-1}$ decompose, using QR or SVD,

$$\left(\boldsymbol{C}_{(n)}^{0,n-1}\right)^\top = \boldsymbol{Q}_n^0 \boldsymbol{S}_n^{0,\top},$$

   where $\boldsymbol{Q}_n^0 \in \mathbb{C}^{r^{d-1} \times r}$ has orthonormal cols., and $\boldsymbol{S}_n^0 \in \mathbb{C}^{r \times r}$.
2. Set $\boldsymbol{K}_n^0 = \boldsymbol{U}_n^0 \boldsymbol{S}_n^0$.
3. With $\boldsymbol{V}_n^{0,\top} = \boldsymbol{Q}_n^{0,\top} \bigotimes_{k<n} \boldsymbol{U}_n^{1/2,\top} \otimes \bigotimes_{k>n} \boldsymbol{U}_n^{0,\top} \in \mathbb{C}^{r \times K^{d-1}}$ solve the linear initial value problem on $\mathbb{C}^{K \times r}$ from t^0 to $t^{1/2}$,

$$i \dot{\boldsymbol{K}}_n(t) = \mathrm{mat}_n H[\mathrm{ten}_n(\boldsymbol{K}_n(t) \boldsymbol{V}_n^{0,\top})] \overline{\boldsymbol{V}_n^0}, \qquad \boldsymbol{K}_n(t^0) = \boldsymbol{K}_n^0.$$

4. Decompose, using QR or SVD,

$$\boldsymbol{K}_n(t^{1/2}) = \boldsymbol{U}_n^{1/2} \widetilde{\boldsymbol{S}}_n^{1/2},$$

   where $\boldsymbol{U}_n^{1/2} \in \mathbb{C}^{K \times r}$ has orthonormal cols., and $\widetilde{\boldsymbol{S}}_n^{1/2} \in \mathbb{C}^{r \times r}$.

Propagation of the basis, forward loop (cont.)

5. Solve the linear initial value problem on $\mathbb{C}^{r\times r}$ backward in time from $t^{1/2}$ to t^0,

   $$i\dot{\boldsymbol{S}}_n(t) = \boldsymbol{U}_n^{1/2,*}\mathrm{mat}_n H[\mathrm{ten}_n(\boldsymbol{U}_n^{1/2}\boldsymbol{S}_n(t)\boldsymbol{V}_n^{0,\top})]\overline{\boldsymbol{V}_n^0}, \quad \boldsymbol{S}_n(t^{1/2}) = \widetilde{\boldsymbol{S}}_n^{1/}$$

   and set $\widetilde{\boldsymbol{S}}_n^0 = \boldsymbol{S}_n(t^0)$.

6. Define the core tensor $C^{0,n} \in \mathbb{C}^{r^d}$ by setting its n-mode matrix unfolding to

   $$\left(\boldsymbol{C}_{(n)}^{0,n}\right)^\top = \boldsymbol{Q}_n^0\widetilde{\boldsymbol{S}}_n^{0,\top}.$$

Propagation of the core tensor

Solve the linear initial value problem on $\mathbb{C}^{r^d}$ from t^0 to t^1,

$$i\dot{C}(t) = H\Big[C(t) \bigtimes_{n=1}^{d} \boldsymbol{U}_n^{1/2}\Big] \bigtimes_{n=1}^{d} \boldsymbol{U}_n^{1/2,*}, \qquad C(t^0) = C^{0,d}.$$

Set $C^{1,d} = C(t^1)$.

Propagation of the basis, backward loop

For $n = d$ down to 1 do the following:

6'. For the n-mode matrix unfolding $\boldsymbol{C}^{1,n}_{(n)} \in \mathbb{C}^{r \times r^{d-1}}$ of the core tensor $C^{1,n}$ decompose, using QR or SVD,

$$\left(\boldsymbol{C}^{1,n}_{(n)}\right)^\top = \boldsymbol{Q}^1_n \widehat{\boldsymbol{S}}^{1,\top}_n,$$

where $\boldsymbol{Q}^1_n \in \mathbb{C}^{r^{d-1} \times r}$ has orthonormal cols., and $\widehat{\boldsymbol{S}}^1_n \in \mathbb{C}^{r \times r}$.

5'. With the notation $\boldsymbol{V}^{1,\top}_n = \boldsymbol{Q}^{1,\top}_n \bigotimes_{k<n} \boldsymbol{U}^{1/2,\top}_n \otimes \bigotimes_{k>n} \boldsymbol{U}^{1,\top}_n$ solve the linear initial value problem on $\mathbb{C}^{r \times r}$ backward in time from t^1 to $t^{1/2}$,

$$i\dot{\boldsymbol{S}}_n(t) = \boldsymbol{U}^{1/2,*}_n \mathrm{mat}_n H[\mathrm{ten}_n(\boldsymbol{U}^{1/2}_n \boldsymbol{S}_n(t) \boldsymbol{V}^{1,\top}_n)]\overline{\boldsymbol{V}^1_n}, \quad \boldsymbol{S}_n(t^1) = \widehat{\boldsymbol{S}}^1_n,$$

and set $\widehat{\boldsymbol{S}}^{1/2}_n = \boldsymbol{S}_n(t^{1/2})$.

Propagation of the basis, backward loop (cont.)

4'. Set $\boldsymbol{K}_n^{1/2} = \boldsymbol{U}_n^{1/2}\widehat{\boldsymbol{S}}_n^{1/2}$.

3'. Solve the linear initial value problem on $\mathbb{C}^{K\times r}$ from $t^{1/2}$ to t^1,

$$i\dot{\boldsymbol{K}}_n(t) = \mathrm{mat}_n H[\mathrm{ten}_n(\boldsymbol{K}_n(t)\boldsymbol{V}_n^{1,\top})]\overline{\boldsymbol{V}_n^1}, \quad \boldsymbol{K}_n(t^{1/2}) = \boldsymbol{K}_n^{1/2}.$$

2'. Decompose, using QR or SVD,

$$\boldsymbol{K}_n(t^1) = \boldsymbol{U}_n^1\boldsymbol{S}_n^1,$$

where $\boldsymbol{U}_n^{1/2} \in \mathbb{C}^{K\times r}$ has orthonormal columns, and $\boldsymbol{S}_n^1 \in \mathbb{C}^{r\times r}$.

1'. Define the core tensor $C^{1,n-1} \in \mathbb{C}^{r^d}$ by setting its n-mode matrix unfolding to

$$\left(\boldsymbol{C}_{(n)}^{1,n-1}\right)^\top = \boldsymbol{Q}_n^1\boldsymbol{S}_n^{1,\top}.$$

Finally, take the core tensor at time t^1 as $C^1 = C^{1,0}$. The algorithm has thus computed the factors in the Tucker tensor decomposition $Y^1 = C^1 \mathsf{X}_{n=1}^d \boldsymbol{U}_n^1$.

Approximation is robust to small singular values

$$\dot{A} = F(t, A), \qquad A(t_0) = Y_0 \in \mathcal{M}_r$$

- F is locally Lipschitz-continuous
- $\|(I - P(Y))F(t, Y)\| \leq \varepsilon$ for all $Y \in \mathcal{M}_r$.

$Y_n \in \mathcal{M}_r$ result of the projector-splitting integrator after n steps with stepsize h

Theorem

$$\|Y_n - A(t_n)\| \leq c_1\varepsilon + c_2 h \qquad \textit{for} \;\; t_n \leq T,$$

where c_1, c_2 depend only on the local Lipschitz constant and bound of F, and on T.

E. Kieri, CL, Hanna Walach, Discretized dynamical low-rank approximation in the presence of small singular values, Preprint 2015, submitted.