Fast Integrators for Scalable Quantum Molecular Dynamics

Alina Kononov¹, Emil Constantinescu², André Schleife¹

¹University of Illinois at Urbana-Champaign
²Argonne National Laboratory

This material is based upon work supported by the National Science Foundation under Grant No. OAC-17-40219. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

This research is part of the Blue Waters sustained-petascale computing project, which is supported by the National Science Foundation (awards OCI-0725070 and ACI-1238993) and the state of Illinois. Blue Waters is a joint effort of the University of Illinois at Urbana-Champaign and its National Center for Supercomputing Applications.
Target Problem: 2D Materials Under Irradiation

Motivation:
- Radiation hardness for space and nuclear applications
- Ion beam techniques for materials imaging and patterning

\[ t = 0 \]

Few-layer material

100s of atoms
100s – 1000s of electrons
Target Problem: 2D Materials Under Irradiation

Motivation:

- Radiation hardness for space and nuclear applications
- Ion beam techniques for materials imaging and patterning

$t \approx 0.5 \text{ fs}$
Target Problem: 2D Materials Under Irradiation

Motivation:
- Radiation hardness for space and nuclear applications
- Ion beam techniques for materials imaging and patterning

$t \approx 2 \text{ fs}$
Real-Time Time-Dependent Density Functional Theory

Exact many-body quantum dynamics described by time-dependent Schrödinger equation

- PDE in 3N+1 variables
- Computationally intractable for most systems

Instead:

- Approximate with single-particle orbitals $\phi_i(\mathbf{r}, t)$
- Re-formulate in terms of electron density $n(\mathbf{r}, t)$
- N coupled PDEs, each in 3+1 variables:

\[
\hat{H}[n, t] = -\frac{1}{2} \nabla^2 + V_{\text{ext}}(\mathbf{r}, t) + V_{\text{Har}}[n] + V_{\text{XC}}[n]
\]

\[
\frac{\partial}{\partial t} \phi_i(\mathbf{r}, t) = \hat{H}[n(\mathbf{r}, t), t] \phi_i(\mathbf{r}, t)
\]

- At $t=0$, start with initial conditions $\phi_i(\mathbf{r}, 0)$
- Integrate numerically to obtain time evolution

\[
n(\mathbf{r}, t) = \sum_{i=1}^{N_e} f_i |\phi_i(\mathbf{r}, t)|^2
\]

Goal: find “best” numerical integrator

- rank $\sim 10^9$

Qbox/Qb@ll code: Schleife et al., Journal of Chemical Physics, 2012
Legacy Integrator: Fourth Order Runge-Kutta (FORK)

\[ i \frac{\partial}{\partial t} \phi_i(\mathbf{r}, t) = \hat{H}[n(\mathbf{r}, t), t] \phi_i(\mathbf{r}, t) \]

Use current state to compute four intermediate stages:

\[ |k_1\rangle = -\frac{i}{\hat{\hbar}} \Delta t \hat{H}[n_{\phi(t)}] |\phi(t)\rangle, \]
\[ |k_2\rangle = -\frac{i}{\hat{\hbar}} \Delta t \hat{H}[n_{\phi(t)+0.5 \cdot k_1}] |\phi(t) + 0.5 \cdot k_1\rangle, \]
\[ |k_3\rangle = -\frac{i}{\hat{\hbar}} \Delta t \hat{H}[n_{\phi(t)+0.5 \cdot k_2}] |\phi(t) + 0.5 \cdot k_2\rangle, \]
\[ |k_4\rangle = -\frac{i}{\hat{\hbar}} \Delta t \hat{H}[n_{\phi(t)+k_3}] |\phi(t) + k_3\rangle, \]

Combine into final estimate of next state:

\[ |\phi(t + \Delta t)\rangle = \left| \phi(t) + \frac{1}{6}k_1 + \frac{1}{3}k_2 + \frac{1}{3}k_3 + \frac{1}{6}k_4 \right| \]

Schleife et al., Journal of Chemical Physics, 2012
New Integrator: Enforced Time-Reversal Symmetry (ETRS)

\[ i \frac{\partial}{\partial t} \phi_i(\mathbf{r}, t) = \hat{H}[n(\mathbf{r}, t), t] \phi_i(\mathbf{r}, t) \]

Approximate next state using current state:

\[ \phi(\mathbf{r}, t + \Delta t) \approx \exp \left( -i \Delta t \hat{H}[n(\mathbf{r}, t)] \right) \phi(\mathbf{r}, t) \]

Compute next state using both current and approximate next states:

\[ \phi(\mathbf{r}, t + \Delta t) = \exp \left( -\frac{i \Delta t}{2} \hat{H}[n(\mathbf{r}, t + \Delta t)] \right) \exp \left( -\frac{i \Delta t}{2} \hat{H}[n(\mathbf{r}, t)] \right) \phi(\mathbf{r}, t) \]

Approximate exponentials with fourth-order Taylor expansions

Draeger et al., IEEE International Parallel and Distributed Processing Symposium, 2016
Challenges: Accuracy and Speed

- Test system: unperturbed Al sheet (300 atoms) propagated for ~3 fs
  - fourth-order Runge-Kutta (FORK) accumulates unacceptable errors
- Reduce time step?
  - already very small (~0.3 attosecond)
  - requires ~10K steps and ~300K CPU-hours
- Speed up code?
  - already ~ideal parallel scaling
- Better integrator?
  - enforced time-reversal symmetry (ETRS) is suitably accurate, but same cost
  - can we find an integrator that improves time-to-solution?
Comparison of Integrators within Qb@ll

- Smaller production-scale system: H+ irradiated graphene (112 atoms)
- Combine accuracy and speed into “figure of merit”
  \[
  \text{FOM} = \frac{dt}{\Delta Q \Delta E \, dT}
  \]
- Figure of merit (FOM) penalizes...
  - small time step $dt$
  - large error in total energy $\Delta E$ per simulation time
  - large error in net charge $\Delta Q$ per simulation time
  - large wall time per simulation time $dT$
- SSPRK improves stability over FORK
- ETRS is by far most accurate
- Still want faster time-to-solution

Kang, Kononov, Lee, et al., Computational Materials Science, 2019
Interfaced Qb@ll with PETSc

- Interfaced Qb@ll with Portable, Extensible Toolkit for Scientific Computation (PETSc) library
  - \( \phi_i(r, t), \frac{\partial}{\partial t} \phi_i(r, t) \rightarrow \text{PETSc} \rightarrow \phi(r, t + \Delta t) \)
  - required data structure conversions
    - \( \phi_i(r, t) \leftrightarrow \text{contiguous array} \leftrightarrow \text{PETSc vector} \)
    - \( \sim 10\% \) overhead
- Seamless access to wide array of integrators
  - advanced Runge-Kutta schemes
  - error estimation and adaptive time stepping
  - implicit methods

Abhyankar, Brown, Constantinescu, et al., arXiv: 1806.01437, 2018
Preliminary PETSc Results

- Tested all Runge-Kutta schemes available within PETSc
- Test system: Na dimer with atoms displaced from equilibrium positions
  - measured errors in total energy ($\Delta E$) and net charge ($\Delta Q$) per simulation time
- ETRS still outperforms other options
  - more accurate than all but rk5bs at some time steps
  - allows largest time step and shortest time-to-solution
- Combine information about accuracy and speed into “figure of merit”
- Figure of merit penalizes...
  - small time step $dt$
  - large error in total energy $\Delta E$ per simulation time
  - large error in net charge $\Delta Q$ per simulation time
  - large wall time per simulation time $dT$

\[
FOM = \frac{dt}{\Delta Q \Delta E \, dT}
\]
Summary and Outlook

- ETRS remains among the best options
  - exceptionally low error
  - largest allowable time steps

- Ongoing efforts:
  - optimize interface
  - further testing of promising candidates (rk5bs, rk5dp, rk5f)
  - adaptive time-stepping
  - other integrators beyond RK
Extra Material
Integrator: Strong Stability Preserving Runge-Kutta (SSPRK)

\[
i \frac{\partial}{\partial t} \phi_i(r, t) = \hat{H}[n(r, t), t] \phi_i(r, t)
\]

General Explicit Runge-Kutta:

Compute \( m \) stages. Each stage depends on previous stages:

\[
|k_0\rangle = |\phi(t)\rangle
\]

\[
|k_i\rangle = \sum_{j=0}^{i-1} \alpha_{i,j} |k_j\rangle - i \Delta t \beta_{i,j} H[n_{k_j}] |k_j\rangle
\]

Use last stage as solution at next time step:  \(|\phi(t + \Delta t)\rangle = |k_m\rangle\)

Optimize coefficients \( \alpha_{ij} \) and \( \beta_{ij} \) for stability

Tested 5-stage and 10-stage 4th-order methods

Optimal five stage, fourth-order method:

\[
\begin{align*}
    u^{(1)} &= u^n + 0.391752226571890 \Delta t F(u^n), \\
    u^{(2)} &= 0.444370493651235 u^n + 0.555629506348765 u^{(1)} \\
    &+ 0.368410593050971 \Delta t F(u^{(1)}), \\
    u^{(3)} &= 0.620101851488403 u^n + 0.379898148511597 u^{(2)} \\
    &+ 0.251891774271694 \Delta t F(u^{(2)}), \\
    u^{(4)} &= 0.178079954393132 u^n + 0.821920045606868 u^{(3)} \\
    &+ 0.544974750228521 \Delta t F(u^{(3)}), \\
    u^{n+1} &= 0.517231671970585 u^{(2)} \\
    &+ 0.096059710526147 u^{(3)} + 0.063692468666290 \Delta t F(u^{(3)}) \\
    &+ 0.386708617503269 u^{(4)} + 0.226007483236906 \Delta t F(u^{(4)})
\end{align*}
\]

Optimal ten stage, fourth-order method:

\[
\begin{align*}
    u^{(i)} &= \sum_{k=0}^{i-1} \left( \alpha_{i,k} u^{(k)} + \Delta t \beta_{i,k} F(u^{(k)}) \right) \\
    \alpha_{i,i-1} &= \begin{cases} 
        1 & i \in \{1..4, 6..9\}, \\
        \frac{2}{5} & i = 5, \\
        \frac{3}{5} & i = 10,
    \end{cases} \\
    \beta_{i,i-1} &= \begin{cases} 
        \frac{1}{6} & i \in \{1..4, 6..9\}, \\
        \frac{1}{15} & i = 5, \\
        \frac{1}{10} & i = 10,
    \end{cases} \\
    \beta_{10,4} &= \frac{3}{50}, \\
    \alpha_{10,4} &= \frac{9}{25}, \\
    \alpha_{5,0} &= \frac{3}{5}, \\
    \alpha_{10,0} &= \frac{1}{25}.
\end{align*}
\]

### Stability of SSPRK Methods

$c$ measures stability
relative to forward Euler

Computational cost scales with $m$

$c_{\text{eff}}$ measures effective efficiency

<table>
<thead>
<tr>
<th>$m$</th>
<th>$p$</th>
<th>$c_{\text{eff}} = c/m$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>2</td>
<td>0.5</td>
<td>–</td>
</tr>
<tr>
<td>3</td>
<td>0.67</td>
<td>0.33</td>
</tr>
<tr>
<td>4</td>
<td>0.75</td>
<td>0.5</td>
</tr>
<tr>
<td>5</td>
<td>0.8</td>
<td>0.53</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.30</td>
</tr>
<tr>
<td>6</td>
<td>0.83</td>
<td>0.59</td>
</tr>
<tr>
<td>7</td>
<td>0.86</td>
<td>0.61</td>
</tr>
<tr>
<td>8</td>
<td>0.88</td>
<td>0.64</td>
</tr>
<tr>
<td>9</td>
<td>0.89</td>
<td>0.67</td>
</tr>
<tr>
<td>10</td>
<td>0.9</td>
<td>0.68</td>
</tr>
<tr>
<td>11</td>
<td>0.91</td>
<td>0.69</td>
</tr>
</tbody>
</table>

Integration error in electron density...

- evolves over time
- varies spatially
- increases with vacuum length
  - affects convergence
- is difficult to predict
## PETSc abbreviations

- **rk1fe**: 1st order forward Euler
- **rk2a**: 2nd order Runge-Kutta
- **rk3**: 3rd order Runge-Kutta
- **rk3bs**: 3rd order Bogacki-Shampine Runge-Kutta with 2nd order embedded method
- **rk4**: 4th order Runge-Kutta
- **rk5bs**: 5th order Bogacki-Shampine Runge-Kutta with 4th order embedded method
- **rk5dp**: 5th order Dormand-Prince Runge-Kutta with 4th order embedded method
- **rk5f**: 5th order Fehlberg Runge-Kutta with 4th order embedded method
- **ssprk104**: 10-stage, 4th order strong stability preserving Runge-Kutta
- **etrs**: enforced time-reversal symmetry