Exponential Integrators and Lie group methods

Workshop on Exponential Integrators

Borislav V. Minchev
Borko.Minchev@ii.uib.no
http://www.ii.uib.no/~borko

Department of computer science
University of Bergen, Norway
Outline

- The framework
- The choice of action
- LGI for semilinear problems
- Implementation issues
- Numerical experiments
- Conclusions
- Future work
Lie group methods

- Designed to preserve certain qualitatively properties of the exact flow
Lie group methods

- Designed to preserve certain qualitatively properties of the exact flow
- The freedom in the choice of the action allows to define the basic motions on the manifold in such a way that they provide a good approximation to the flow of the original vector field.
Background theory

Lie group - $(G, e, \star)$

Lie algebra $g = T_e G$ - linear space with bracket

$$[\Theta_1, \Theta_2] = \left. \frac{\partial^2}{\partial s \partial t} \right|_{t=s=0} \gamma_1(s) \star \gamma_2(t) \star \gamma_1(s)^{-1},$$

where $\gamma_1(s)$ and $\gamma_2(t)$ are smooth curves in $G$ such that $\gamma_1(0) = \gamma_2(0) = e$ and $\gamma_1'(0) = \Theta_1, \gamma_2'(0) = \Theta_2$. 
Background theory

Lie group - \((G, e, \star)\)

Lie algebra \(g = T_eG\) - linear space with bracket

\[
[\Theta_1, \Theta_2] = \left. \frac{\partial^2}{\partial s \partial t} \right|_{t = s = 0} \gamma_1(s) \star \gamma_2(t) \star \gamma_1(s)^{-1},
\]

where \(\gamma_1(s)\) and \(\gamma_2(t)\) are smooth curves in \(G\) such that \(\gamma_1(0) = \gamma_2(0) = e\) and \(\gamma'_1(0) = \Theta_1, \gamma'_2(0) = \Theta_2\).

Define the product \(\odot : g \times G \to TG\) by

\[
\Theta \odot g = \left. \frac{d}{dt} \right|_{t = 0} \gamma(t) \star g,
\]

where \(\gamma(t)\) is a smooth curve in \(G\) such that \(\gamma(0) = e\) and \(\gamma'(0) = \Theta\).
The Exp map

The exponential map provides connection between $\mathcal{G}$ and $\mathfrak{g}$. $\text{Exp} : \mathfrak{g} \rightarrow \mathcal{G}$ is defined as $\text{Exp}(\Theta) = \gamma(1)$, where $\gamma(t) \in \mathcal{G}$ satisfies the differential equation

$$\gamma(t)' = \Theta \circ \gamma(t), \quad \gamma(0) = e.$$
The exponential map provides connection between $\mathcal{G}$ and $g$. 

$\text{Exp} : g \to \mathcal{G}$ is defined as $\text{Exp}(\Theta) = \gamma(1)$, where $\gamma(t) \in \mathcal{G}$ satisfies the differential equation

$$\gamma(t)' = \Theta \odot \gamma(t), \quad \gamma(0) = e.$$ 

The differential of the exponential map $d\text{Exp} : g \times g \to g$ is defined as the right trivialized tangent

$$d\text{Exp}(\widehat{\Theta}, \Theta) \text{Exp}(\widehat{\Theta}) = \frac{d}{dt} \bigg|_{t=0} \text{Exp}(\widehat{\Theta} + t\Theta).$$
The Exp map

The exponential map provides connection between $G$ and $g$. $\text{Exp} : g \to G$ is defined as $\text{Exp}(\Theta) = \gamma(1)$, where $\gamma(t) \in G$ satisfies the differential equation

$$\gamma(t)' = \Theta \odot \gamma(t), \quad \gamma(0) = e.$$ 

The differential of the exponential map $d\text{Exp} : g \times g \to g$ is defined as the right trivialized tangent

$$d\text{Exp}(\Theta, \Theta) \text{Exp}(\Theta) = \left. \frac{d}{dt} \right|_{t=0} \text{Exp}(\Theta + t\Theta).$$

$$d\text{Exp}_\Theta(\Theta) = \left. \frac{e^z - 1}{z} \right|_{z = \text{ad}_\Theta(\Theta)} = \sum_{k=0}^{\infty} \frac{1}{(k+1)!} \text{ad}_\Theta^k(\Theta),$$

$$d\text{Exp}_\Theta^{-1}(\Theta) = \left. \frac{z}{e^z - 1} \right|_{z = \text{ad}_\Theta^{-1}(\Theta)} = \sum_{k=0}^{\infty} \frac{B_k}{k!} \text{ad}_\Theta^k(\Theta),$$

where the coefficients $B_k$ are the Bernoulli numbers and

$$\text{ad}_\Theta^k(\Theta) = \text{ad}_\Theta(\text{ad}_\Theta^{k-1}(\Theta)) = [\Theta, \ldots, [\Theta, \Theta]], \quad \text{for } k > 1.$$
Actions on the manifold

A group action on a manifold $\mathcal{M}$ is a smooth map $\cdot : \mathcal{G} \times \mathcal{M} \to \mathcal{M}$ satisfying
\[
\begin{align*}
e \cdot p & = p \quad \forall p \in \mathcal{M}, \\
g \cdot (k \cdot p) & = (g \ast k) \cdot p \quad \forall g, k \in \mathcal{G}, \; p \in \mathcal{M}.
\end{align*}
\]
A group action on a manifold $\mathcal{M}$ is a smooth map $\cdot : \mathcal{G} \times \mathcal{M} \to \mathcal{M}$ satisfying

$$e \cdot p = p \quad \forall \, p \in \mathcal{M},$$

$$g \cdot (k \cdot p) = (g \ast k) \cdot p \quad \forall \, g, k \in \mathcal{G}, \, p \in \mathcal{M}.$$ 

An algebra action $\ast : \mathcal{g} \times \mathcal{M} \to \mathcal{M}$ on $\mathcal{M}$ is given by

$$\Theta \ast p = \exp(\Theta) \cdot p.$$
Actions on the manifold

A group action on a manifold $\mathcal{M}$ is a smooth map $\cdot : \mathcal{G} \times \mathcal{M} \rightarrow \mathcal{M}$ satisfying

$$
e \cdot p = p \quad \forall \, p \in \mathcal{M},$$

$$g \cdot (k \cdot p) = (g \ast k) \cdot p \quad \forall \, g, k \in \mathcal{G}, \, p \in \mathcal{M}.$$ 

An algebra action $\ast : \mathfrak{g} \times \mathcal{M} \rightarrow \mathcal{M}$ on $\mathcal{M}$ is given by

$$\Theta \ast p = \text{Exp}(\Theta) \cdot p.$$ 

Note: The algebra action is not uniquely determined by the group action. Every diffeomorphism

$$\Psi : \mathfrak{g} \rightarrow \mathcal{G},$$

such that $\Psi(0) = e$ and $\Psi'(0) = I$, where $I$ is the identity of the algebra, defines an algebra action by the formula $\Theta \ast p = \Psi(\Theta) \cdot p.$
The product \( \otimes : g \times \mathcal{M} \rightarrow TM \) between \( g \) and \( \mathcal{M} \) is defined by

\[
\Theta \otimes p = \left. \frac{d}{dt} \right|_{t=0} \gamma(t) \cdot p,
\]

where \( \gamma(t) \) is a smooth curve in \( \mathcal{G} \) such that \( \gamma(0) = e \) and \( \gamma'(0) = \Theta \).
The product $\otimes: g \times M \to TM$ between $g$ and $M$ is defined by

$$\Theta \otimes p = \left. \frac{d}{dt} \right|_{t=0} \gamma(t) \cdot p,$$

where $\gamma(t)$ is a smooth curve in $G$ such that $\gamma(0) = e$ and $\gamma'(0) = \Theta$.

For a fix $\Theta \in g$ the product $\otimes$ gives a vector field on $M$

$$\mathcal{F}_\Theta(p) = \Theta \otimes p = \left. \frac{d}{dt} \right|_{t=0} \text{Exp}(t\Theta) \cdot p.$$

The vector field $\mathcal{F}_\Theta$ is called a frozen vector field.
The product $\otimes : g \times \mathcal{M} \to T\mathcal{M}$ between $g$ and $\mathcal{M}$ is defined by

$$
\Theta \otimes p = \frac{d}{dt} \bigg|_{t=0} \gamma(t) \cdot p,
$$

where $\gamma(t)$ is a smooth curve in $\mathcal{G}$ such that $\gamma(0) = e$ and $\gamma'(0) = \Theta$.

For a fix $\Theta \in g$ the product $\Theta \otimes p$ gives a vector field on $\mathcal{M}$

$$
\mathcal{F}_\Theta(p) = \Theta \otimes p = \frac{d}{dt} \bigg|_{t=0} \text{Exp}(t\Theta) \cdot p.
$$

The vector field $\mathcal{F}_\Theta$ is called a frozen vector field.

Every differential equation evolving on a homogeneous space $\mathcal{M}$ can always be written as

$$
u'(t) = F(u) \otimes u, \quad u(t_0) = u_0,
$$

where $F : \mathcal{M} \to g$. 
Lie group integrators

Algorithm. (Crouch–Grossman’93)

for $i = 1, \ldots, s$ do

\[
U_i = \text{Exp}(h\alpha_i F_s) \cdots \text{Exp}(h\alpha_1 F_1) \cdot u_n
\]

\[
F_i = F(U_i)
\]
end

\[
u_{n+1} = \text{Exp}(h\beta_s F_s) \cdots \text{Exp}(h\beta_1 F_1) \cdot u_n
\]
Lie group integrators

Algorithm. (Crouch–Grossman'93)

for $i = 1, \ldots, s$ do

\[
U_i = (h\alpha_i F_s) \cdots (h\alpha_1 F_1) * u_n
\]
\[
F_i = F(U_i)
\]
end

$u_{n+1} = (h\beta_s F_s) \cdots (h\beta_1 F_1) * u_n$
Lie group integrators

Algorithm. (Crouch–Grossman’93)

for $i = 1, \ldots, s$ do

\[ U_i = (h\alpha_i F_s) \ast \cdots \ast (h\alpha_1 F_1) \ast u_n \]
\[ F_i = F(U_i) \]

end

\[ u_{n+1} = (h\beta_s F_s) \ast \cdots \ast (h\beta_1 F_1) \ast u_n \]

Algorithm. (Runge–Kutta Munthe-Kaas’99)

for $i = 1, \ldots, s$ do

\[ \Theta_i = h \sum_{j=1}^{s} \alpha_{ij} K_j \]
\[ F_i = F(\Psi(\Theta_i) \cdot u_n) \]
\[ K_i = \sigma \Psi^{-1}(F_i) \]

end

\[ u_{n+1} = \Psi(h \sum_{i=1}^{s} \beta_i K_i) \cdot u_n \]
Lie group integrators

Algorithm. (Crouch–Grossman'93)

for $i = 1, \ldots, s$ do
    $U_i = (h\alpha_is F_s) \cdots (h\alpha_1 F_1) * u_n$
    $F_i = F(U_i)$
end

$u_{n+1} = (h\beta_s F_s) \cdots (h\beta_1 F_1) * u_n$

Algorithm. (Runge–Kutta Munthe–Kaas'99)

for $i = 1, \ldots, s$ do
    $\Theta_i = h \sum_{j=1}^s \alpha_{ij} K_j$
    $F_i = F((\Theta_i) * u_n)$
    $K_i = \sigma \Psi^{-1}(F_i)$
end

$u_{n+1} = (h \sum_{i=1}^s \beta_i K_i) * u_n$
Lie group integrators

Algorithm. *(Crouch–Grossman'93)*

for \( i = 1, \ldots, s \) do

\[
U_i = (h\alpha_i F_s) \cdot \ldots \cdot (h\alpha_1 F_1) \cdot u_n
\]

\[
F_i = F(U_i)
\]

end

\[
u_{n+1} = (h\beta_s F_s) \cdot \ldots \cdot (h\beta_1 F_1) \cdot u_n
\]

Algorithm. *(Hunze–Kutta Munthe-Kaas'99)*

for \( i = 1, \ldots, s \) do

\[
\Theta_i = h \sum_{j=1}^{s} \alpha_{ij} K_j
\]

\[
F_i = F((\Theta_i) \cdot u_n)
\]

\[
K_i = \alpha \Psi^{-1}(F_i)
\]

end

\[
u_{n+1} = (h \sum_{i=1}^{s} \beta_i K_i) \cdot u_n
\]

The values on \( M \) can be computed via the formula \( U_i = \Theta_i \cdot u_n \).
Lie group integrators

Algorithm. Commutator-free Lie group method (Celledoni, Marthinsen, Owren 03)

for $i = 1, \ldots, s$ do

\[
\begin{align*}
U_i &= \text{Exp}(h \sum_{k=1}^{s} \alpha_{ij}^k F_k) \cdots \text{Exp}(h \sum_{k=1}^{s} \alpha_{i1}^k F_k) \cdot u_n \\
F_i &= F(U_i)
\end{align*}
\]
end

\[
\begin{align*}
u_{n+1} &= \text{Exp}(h \sum_{k=1}^{s} \beta_{ij}^k F_k) \cdots \text{Exp}(h \sum_{k=1}^{s} \beta_{i1}^k F_k) \cdot u_n
\end{align*}
\]
Lie group integrators

Algorithm. Commutator-free Lie group method (Celledoni, Marthinsen, Owren 03)

for $i = 1, \ldots, s$ do

\[ U_i = (h \sum_{k=1}^{s} \alpha_{ij}^k F_k) \ast \cdots \ast (h \sum_{k=1}^{s} \alpha_{i1}^k F_k) \ast u_n \]

\[ F_i = F(U_i) \]

end

\[ u_{n+1} = (h \sum_{k=1}^{s} \beta_{ij}^k F_k) \ast \cdots \ast (h \sum_{k=1}^{s} \beta_{i1}^k F_k) \ast u_n \]
Lie group integrators

**Algorithm.** Commutator-free Lie group method (Celledoni, Marthinsen, Owren 03)

for $i = 1, \ldots, s$ do

$$U_i = (h \sum_{k=1}^{s} \alpha_{iJ}^k F_k) \ast \cdots \ast (h \sum_{k=1}^{s} \alpha_{i1}^k F_k) \ast u_n$$

$$F_i = F(U_i)$$

end

$$u_{n+1} = (h \sum_{k=1}^{s} \beta_{iJ}^k F_k) \ast \cdots \ast (h \sum_{k=1}^{s} \beta_{i1}^k F_k) \ast u_n$$

An example of a fourth order method, based on the classical fourth order Runge–Kutta method

\[
\begin{array}{c|cccc}
0 & & & & \\
\frac{1}{2} & & & & \\
\frac{1}{2} & & \frac{1}{2} & & \\
\frac{1}{2} & & 0 & \frac{1}{2} & \\
\frac{1}{2} & & \frac{1}{2} & 0 & 0 \\
\frac{1}{2} & & -\frac{1}{2} & 0 & 1 \\
\hline
\frac{1}{4} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & -\frac{1}{12} \\
\frac{1}{12} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{4}
\end{array}
\]
Consider the following nonautonomous problem defined on $\mathbb{R}^d$

$$u' = f(u, t), \quad u(t_0) = u_0.$$ 

By adding the trivial differential equation $t' = 1$, we can rewrite it in the form

$$y' = f(y(t)), \quad y(t_0) = y_0,$$

where

$$f = \begin{bmatrix} f(u, t) \\ 1 \end{bmatrix}, \quad y = \begin{bmatrix} u \\ t \end{bmatrix}.$$
Consider the following nonautonomous problem defined on $\mathbb{R}^d$

$$u' = f(u, t), \quad u(t_0) = u_0.$$ 

By adding the trivial differential equation $t' = 1$, we can rewrite it in the form

$$y' = f(y(t)), \quad y(t_0) = y_0.$$ 

Define:
- the basic movements on $\mathcal{M}$ to be given by the solution of a simpler diff. equation

$$(2) \quad y' = \mathcal{F}_\Theta(y), \quad y(t_0) = y_0,$$

where $\mathcal{F}_\Theta(y)$ approximates $f(y(t))$.
- the Lie algebra $\mathfrak{g}$ to be the set of all coefficients $\Theta$ of the frozen vector fields $\mathcal{F}_\Theta$.
- the algebra action $h\Theta * y_0$ to be the solution of (2) at time $t_0 + h$. 

Basic motions on $\mathcal{M}$
The choice of action

- The simplest case
  - \( \mathfrak{g} = \{ \mathbf{b} \in \mathbb{R}^{d+1} \} = \{ (b^{[0]}, \lambda) : b^{[0]} \in \mathbb{R}^d, \lambda \in \mathbb{R} \} \).
  - The generic function is given by \( F \left( \begin{bmatrix} u_0 \\ t_0 \end{bmatrix} \right) = (f(u_0, t_0), 1) = (b^{[0]}, 1). \)
  - The frozen vector field is \( \mathcal{F}_{(b^{[0]}, \lambda)} \left( \begin{bmatrix} u \\ t \end{bmatrix} \right) = \begin{bmatrix} b^{[0]} \\ \lambda \end{bmatrix} \).
  - The algebra action is \( h(b^{[0]}, \lambda) \ast \begin{bmatrix} u_0 \\ t_0 \end{bmatrix} = \begin{bmatrix} u_0 + h b^{[0]} \\ t_0 + h \lambda \end{bmatrix} \) (translations).
  - The commutators are given by \([\Theta_1, \Theta_2] = (0, 0)\). In this case we recover the traditional integration schemes.
The choice of action

- The simplest case
  - $g = \{ b \in \mathbb{R}^{d+1} \} = \{ (b^{[0]}, \lambda) : b^{[0]} \in \mathbb{R}^d, \lambda \in \mathbb{R} \}$.
  - The generic function is given by $F\left(\begin{bmatrix} u_0 \\ t_0 \end{bmatrix}\right) = (f(u_0, t_0), 1) = (b^{[0]}, 1)$.
  - The frozen vector field is $F_{(b^{[0]}, \lambda)} \left(\begin{bmatrix} u \\ t \end{bmatrix}\right) = \begin{bmatrix} b^{[0]} \\ \lambda \end{bmatrix}$.
  - The algebra action is $h(b^{[0]}, \lambda) * \begin{bmatrix} u_0 \\ t_0 \end{bmatrix} = \begin{bmatrix} u_0 + h b^{[0]} \\ t_0 + h \lambda \end{bmatrix}$ (translations).
  - The commutators are given by $[\Theta_1, \Theta_2] = (0, 0)$.

In this case we recover the traditional integration schemes.

- When $f(u, t) = L(u, t)u + N(u, t)$ (such a representation is always possible)
The choice of action

- The simplest case
  - $g = \{ b \in \mathbb{R}^{d+1} \} = \{ (b^0, \lambda) : b^0 \in \mathbb{R}^d, \lambda \in \mathbb{R} \}$.
  - The generic function is given by $F\left( \begin{bmatrix} u_0 \\ t_0 \end{bmatrix} \right) = (f(u_0, t_0), 1) = (b^0, 1)$.
  - The frozen vector field is $F_{b^0,\lambda}\left( \begin{bmatrix} u \\ t \end{bmatrix} \right) = \begin{bmatrix} b^0 \\ \lambda \end{bmatrix}$.
  - The algebra action is $h(b^0, \lambda) \ast \begin{bmatrix} u_0 \\ t_0 \end{bmatrix} = \begin{bmatrix} u_0 + h b^0 \\ t_0 + h \lambda \end{bmatrix}$ (translations).
  - The commutators are given by $[\Theta_1, \Theta_2] = (0, 0)$.
  In this case we recover the traditional integration schemes.

- When $f(u, t) = L(u, t)u + N(u, t)$ (such a representation is always possible)
  - $g = \{ (A, b) \in \mathbb{R}^{d+1 \times d+1} \times \mathbb{R}^{d+1} \}$, with
    $A = \begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix}$, $b = \begin{bmatrix} b^0 \\ \lambda \end{bmatrix}$,
The choice of action

- The simplest case

  - $g = \{ b \in \mathbb{R}^{d+1} \} = \{(b^{[0]}, \lambda) : b^{[0]} \in \mathbb{R}^d, \lambda \in \mathbb{R} \}$.
  - The generic function is given by $F([u_0 \atop t_0]) = (f(u_0, t_0), 1) = (b^{[0]}, 1)$.
  - The frozen vector field is $F(b^{[0]}, \lambda) (\begin{bmatrix} u \\ t \end{bmatrix}) = \begin{bmatrix} b^{[0]} \\ \lambda \end{bmatrix}$.
  - The algebra action is $h(b^{[0]}, \lambda) * \begin{bmatrix} u_0 \\ t_0 \end{bmatrix} = \begin{bmatrix} u_0 + h b^{[0]} \\ t_0 + h \lambda \end{bmatrix}$ (translations).
  - The commutators are given by $[\Theta_1, \Theta_2] = (0, 0)$.

In this case we recover the traditional integration schemes.

- When $f(u, t) = L(u, t) u + N(u, t)$ (such a representation is always possible)

  - $g = \{(A, b) \in \mathbb{R}^{d+1 \times d+1} \times \mathbb{R}^{d+1} \} = \{(A, b^{[0]}, \lambda) : A \in \mathbb{R}^{d \times d}, b^{[0]} \in \mathbb{R}^d, \lambda \in \mathbb{R} \}$.
  - The generic function is given by $F([u_0 \atop t_0]) = (L(u_0, t_0), N(u_0, t_0), 1) = (A, b^{[0]}, 1)$.
  - The frozen vector field is $F(A, b^{[0]}, \lambda) (\begin{bmatrix} u \\ t \end{bmatrix}) = \begin{bmatrix} A u + b^{[0]} \\ \lambda \end{bmatrix}$.
  - The algebra action is given by $h(A, b^{[0]}, \lambda) * \begin{bmatrix} u_0 \\ t_0 \end{bmatrix} = \begin{bmatrix} e^{hA} u_0 + h b^{[0]} \phi^{[1]}(h A) \\ t_0 + h \lambda \end{bmatrix}$, where $e^{hA}$ denotes the matrix exponential and $\phi^{[1]}$ is the first ETD $\phi^{[2]}$ function.
  - The commutators are given by $[\Theta_1, \Theta_2] = (A_1 A_2 - A_2 A_1, A_1 b_2^{[0]} - A_2 b_1^{[0]}, 0)$.

In this case we recover the affine algebra action proposed by Munthe-Kaas'99.
Similarly, when \( f(u, t) = L(u, t)u + N^{[0]}(u, t) + t N^{[1]}(u, t) \).

\[
g = \{(A, b) \in \mathbb{R}^{d+1 \times d+1} \times \mathbb{R}^{d+1}\}, \text{ with}
\]

\[
A = \begin{bmatrix}
A & b^{[1]} \\
0 & 0
\end{bmatrix}, \quad b = \begin{bmatrix}
b^{[0]} \\
\lambda
\end{bmatrix},
\]
Nonautonomous frozen vector fields

- Similarly, when $f(u, t) = L(u, t)u + N^0(u, t) + tN^1(u, t)$.

- $g = \{(A, b) \in \mathbb{R}^{d+1} \times \mathbb{R}^{d+1} \times \mathbb{R}^{d+1} \}$
  $$= \{(A, b^{[1]}, b^{[0]}, \lambda) : A \in \mathbb{R}^{d \times d}, b^{[1]}, b^{[0]} \in \mathbb{R}^d, \lambda \in \mathbb{R} \}.$$

- The generic function is given by
  $$F\left([u_0 \atop t_0]\right) = (L(u_0, t_0), N^{[1]}(u_0, t_0), N^{[0]}(u_0, t_0), 1) = (A, b^{[1]}, b^{[0]}, 1).$$

- The frozen vector field is
  $$\mathcal{F}(A, b^{[1]}, b^{[0]}, \lambda) \left([u \atop t]\right) = (A, b^{[1]}, b^{[0]}, \lambda) \circ \left[u \atop t\right] = [Au + c_0 + tc_1, \lambda],$$
  where $c_0 = b^{[0]} + t_0(1 - \lambda)b^{[1]}$ and $c_1 = \lambda b^{[1]}$.

- The algebra action is given by
  $$h(A, b^{[1]}, b^{[0]}, \lambda) \star \left[u_0 \atop t_0\right] = \left[e^{hA}u_0 + h(b^{[0]} + t_0b^{[1]})\phi^{[1]}(hA) + h^2\lambda b^{[1]}\phi^{[2]}(hA)\right].$$

- The commutators are given by
  $$[\Theta_1, \Theta_2] = \left([A_1, A_2], A_1b_2^{[1]} - A_2b_1^{[1]}, A_1b_2^{[0]} - A_2b_1^{[0]} + \lambda_2b_1^{[1]} - \lambda_1b_2^{[1]}, 0\right).$$
Nonautonomous frozen vector fields

- Similarly, when \( f(u, t) = L(u, t)u + N^{[0]}(u, t) + tN^{[1]}(u, t) \).

- \( g = \{(A, b) \in \mathbb{R}^{d+1 \times d+1} \times \mathbb{R}^{d+1}\} = \{(A, b^{[1]}, b^{[0]}, \lambda) : A \in \mathbb{R}^{d \times d}, b^{[1]}, b^{[0]} \in \mathbb{R}^d, \lambda \in \mathbb{R}\} \).

- The generic function is given by
  \[
  F \left( \begin{bmatrix} u_0 \\ t_0 \end{bmatrix} \right) = (L(u_0, t_0), N^{[1]}(u_0, t_0), N^{[0]}(u_0, t_0), 1) = (A, b^{[1]}, b^{[0]}, 1).
  \]

- The frozen vector field is
  \[
  F(A, b^{[1]}, b^{[0]}, \lambda) \left( \begin{bmatrix} u \\ t \end{bmatrix} \right) = (A, b^{[1]}, b^{[0]}, \lambda) \otimes \begin{bmatrix} u \\ t \end{bmatrix} = \begin{bmatrix} Au + c_0 + tc_1 \\ \lambda \end{bmatrix},
  \]
  where \( c_0 = b^{[0]} + t_0(1 - \lambda)b^{[1]} \) and \( c_1 = \lambda b^{[1]} \).

- The algebra action is given by
  \[
  h(A, b^{[1]}, b^{[0]}, \lambda) \star \begin{bmatrix} u_0 \\ t_0 \end{bmatrix} = \begin{bmatrix} e^{hA}u_0 + h(b^{[0]} + t_0b^{[1]})\phi^{[1]}(hA) + h^2\lambda b^{[1]}\phi^{[2]}(hA) \\ t_0 + h\lambda \end{bmatrix}.
  \]

- The commutators are given by
  \[
  [\Theta_1, \Theta_2] = \left( [A_1, A_2], A_1b_2^{[1]} - A_2b_1^{[1]}, A_1b_2^{[0]} - A_2b_1^{[0]} + \lambda_2b_1^{[1]} - \lambda_1b_2^{[1]}, 0 \right).
  \]

- Can generalize this approach to the case \( f(u, t) = L(u, t)u + \sum_{k=0}^{p} t^k N^{[k]}(u, t) \).

  - append \( p \) trivial differential equations corresponding to \( t, t^2, \ldots, t^p \).
Consider the semilinear problem

(1) \[ u' = Lu + N(u, t), \quad u(t_0) = u_0, \]

- Natural choice for $\ast$ is the affine action.
Consider the semilinear problem

\[ u' = Lu + N(u, t), \ u(t_0) = u_0, \]

- Natural choice for \( * \) is the affine action.

**A third order Crouch–Grossman method**

\[
\begin{bmatrix}
0 & 0 & 0 \\
\frac{3}{4} \phi^{[1]} & 0 & 0 \\
\frac{119}{216} e^{\frac{17}{108} hL} \phi^{[1]} \left( \frac{119}{216} hL \right) & \frac{17}{108} \phi^{[1]} \left( \frac{17}{108} hL \right) & 0 \\
\frac{13}{51} e^{\frac{38}{51} hL} \phi^{[1]} \left( \frac{13}{51} hL \right) & -\frac{2}{3} e^{\frac{24}{17} hL} \phi^{[1]} \left( -\frac{2}{3} hL \right) & \frac{24}{17} \phi^{[1]} \left( \frac{24}{17} hL \right) \\
\end{bmatrix}
\begin{bmatrix}
I \\
e^{\frac{3}{4} hL} \\
e^{\frac{17}{24} hL} \\
e^{hL}
\end{bmatrix}
\]
Consider the semilinear problem

\( u' = Lu + N(u, t), \quad u(t_0) = u_0, \)

- Natural choice for \( \ast \) is the affine action.

A fourth order RKMK method with exact \( \text{Exp} \)

\[
\begin{bmatrix}
0 & 0 & 0 & 0 & 0 \\
\frac{1}{2} \phi^{[1]} & 0 & 0 & 0 & 0 \\
\frac{1}{2} \phi^{[1]} - \frac{1}{2} I & \frac{1}{2} I & 0 & 0 & 0 \\
\hat{\phi}^{[3]^2} \left( \frac{hL}{2} \right) & -\hat{\phi}^{[2]} \hat{\phi}^{[3]} \left( \frac{hL}{2} \right) & \hat{\phi}^{[2]} & 0 & \frac{1}{6} I \\
b_1(hL) & b_2(hL) & b_3(hL) & \frac{1}{6} I & e^{hL}
\end{bmatrix}
\begin{bmatrix}
I \\
\frac{1}{2} hL \\
\frac{1}{2} hL \\
\frac{1}{6} hL \\
e^{hL}
\end{bmatrix}
\]
LGI for semilinear problems

Consider the semilinear problem

(1) \[ u' = Lu + N(u, t), \quad u(t_0) = u_0, \]

- Natural choice for \( \ast \) is the affine action.

A fourth order RKMK method with exact \( \text{Exp} \)

where

\[
\begin{align*}
\hat{\phi}^{[2]}(z) &= \phi^{[1]}(z)\phi^{[1]}^{-1}\left(\frac{z}{2}\right) = \frac{e^{\frac{z}{2}} + I}{2}, \\
\hat{\phi}^{[3]}(z) &= \phi^{[1]}^{-1}(z) - I = \frac{e^z - z - I}{I - e^z}, \\
b_1(hL) &= \frac{1}{6}\phi^{[1]} - \frac{1}{3}\phi^{[1]}\hat{\phi}^{[3]}\left(\frac{hL}{2}\right) - \frac{1}{6}\phi^{[1]}\left(\hat{\phi}^{[3]} - 2I\right)\hat{\phi}^{[3]}^2\left(\frac{hL}{2}\right), \\
b_2(hL) &= \frac{1}{3}\hat{\phi}^{[2]} + \frac{1}{6}\hat{\phi}^{[2]}\left(\hat{\phi}^{[3]} - 2I\right)\hat{\phi}^{[3]}\left(\frac{hL}{2}\right), \\
b_3(hL) &= \frac{1}{3}\hat{\phi}^{[2]} - \frac{1}{6}\hat{\phi}^{[2]}\hat{\phi}^{[3]}. 
\end{align*}
\]
LGI for semilinear problems

Consider the semilinear problem

\[ u' = Lu + N(u, t), \quad u(t_0) = u_0, \]

- Natural choice for \( * \) is the affine action.

RKMK methods with approximations to the Exp map

With appropriate choices for the diffeomorphism \( \Psi \) it is possibel to show that:
- IF RK methods are RKMK methods (Krogstad'03);
- GIF/RK methods are also RKMK methods.
Consider the semilinear problem

\[ u' = Lu + N(u, t), \quad u(t_0) = u_0, \]

- Natural choice for * is the affine action.

A fourth order CF method

\[
\begin{bmatrix}
0 & 0 & 0 & 0 \\
\frac{1}{2} \phi[1] & 0 & 0 & 0 \\
0 & \frac{1}{2} \phi[1] & 0 & 0 \\
\frac{1}{2} \phi[1] & 0 & 0 & 0 \\
-\frac{1}{2} \phi[1] & 0 & \phi[1] & 0 \\
\frac{1}{4} \phi[1] & \frac{1}{6} \phi[1] & \frac{1}{6} \phi[1] & -\frac{1}{12} \phi[1] \\
-\frac{1}{12} \phi[1] & \frac{1}{6} \phi[1] & \frac{1}{6} \phi[1] & \frac{1}{4} \phi[1]
\end{bmatrix}
\begin{bmatrix}
I \\
e^{-\frac{hL}{2}} \\
e^{-\frac{hL}{2}} \\
e^{-\frac{hL}{2}} \\
e^{-\frac{hL}{2}} \\
e^{-\frac{hL}{2}}
\end{bmatrix}
\]
Consider the semilinear problem

\[ u' = Lu + N(u, t), \quad u(t_0) = u_0, \]

- Natural choice for \( * \) is the affine action.

**A fourth order CF method**

\[
\begin{bmatrix}
0 & 0 & 0 & 0 \\
\frac{1}{2} \phi^{[1]} & 0 & 0 & 0 \\
0 & \frac{1}{2} \phi^{[1]} & 0 & 0 \\
\frac{1}{2} \phi^{[1]} \left( \frac{hL}{2} \right) \left( e^{\frac{hL}{2}} - I \right) & 0 & \phi^{[1]} \left( \frac{hL}{2} \right) & 0 \\
\frac{1}{2} \phi^{[1]} - \frac{1}{3} \phi^{[1]} \left( \frac{hL}{2} \right) & \frac{1}{3} \phi^{[1]} & \frac{1}{3} \phi^{[1]} & -\frac{1}{6} \phi^{[1]} + \frac{1}{3} \phi^{[1]} \left( \frac{hL}{2} \right)
\end{bmatrix}
\begin{bmatrix}
I \\
e^{\frac{1}{2} hL} \\
e^{\frac{1}{2} hL} \\
e^{hL} \\
e^{hL}
\end{bmatrix}
\]
LGI for semilinear problems

Consider the semilinear problem

\begin{equation}
  u' = Lu + N(u, t), \quad u(t_0) = u_0,
\end{equation}

- Natural choice for * is the affine action.
- Other possible choice is to use nonautonomous frozen vector fields

\[ N(u, t) = N_n + t \frac{N(u, t) - N_n}{t} = N[0] + t N[1] \]
Consider the semilinear problem

\begin{equation}
    u' = Lu + N(u, t), \quad u(t_0) = u_0,
\end{equation}

- Natural choice for $*$ is the affine action.
- Other possible choice is to use nonautonomous frozen vector fields

\[ N(u, t) = N_n + t \frac{N(u, t) - N_n}{t} = N[0] + t N[1] \]

or

\[ N(u, t) = N_n + t \frac{N_n - N_{n-1}}{t} + t^2 \frac{N(u, t) - 2N_n + N_{n-1}}{t^2}, \]

where $N_n$ and $N_{n-1}$ are the values of $N$ at the end of step number $n$ and $n - 1$.

*Note:* In this way we again obtain GLMs.
Implementation issues

Consider the ETD $\phi^{[i]}$ functions

$$
\phi^{[1]}(z) = \frac{e^z - 1}{z}, \quad \phi^{[i+1]}(z) = \frac{\phi^{[i]}(z) - \frac{1}{i!}}{z}, \quad \text{for} \quad i = 2, 3, \ldots.
$$

A straightforward implementation suffers from cancellation errors (Kassam and Trefethen).
Implementation issues

Consider the ETD $\phi^i$ functions

$$\phi^1(z) = \frac{e^z - 1}{z}, \quad \phi^{i+1}(z) = \frac{\phi^i(z) - \frac{1}{i!}}{z}, \quad \text{for} \ i = 2, 3, \ldots.$$ 

A straightforward implementation suffers from cancellation errors (Kassam and Trefethen).

Numerical techniques

- Decomposition methods
- Krylov subspace approximations
- Cauchy integral approach
Cauchy integral approach

Based on the Cauchy integral formula

\[ \phi^{(\tilde{z})}(A) = \frac{1}{2\pi i} \int_{\Gamma_A} \phi^{(\tilde{z})}(\lambda)(\lambda I - A)^{-1} d\lambda, \]

where \( \Gamma_A \) is a contour in the complex plane that encloses the eigenvalue of \( A \), and it is also well separated from 0. It is practical to choose the contour \( \Gamma_A \) to be a circle centered on the real axis.

Using the trapezoid rule, we obtain the following approximation

\[ \phi^{(\tilde{z})}(A) \approx \frac{1}{k} \sum_{j=1}^{k} \lambda_j \phi^{(\tilde{z})}(\lambda_j)(\lambda_j I - A)^{-1}, \]

where \( k \) is the number of the equally spaced points \( \lambda_j \) along the contour \( \Gamma_A \).
Cauchy integral approach

To achieve computational savings we can use the formula

$$\phi^{[i]}(A) = \phi^{[i]}(\gamma hL) = \frac{1}{2\pi i} \int_{\Gamma} \phi^{[i]}(\gamma \lambda) (\lambda I - hL)^{-1} d\lambda,$$

where the contour $\Gamma$ encloses the eigenvalues of $\gamma hL$ and $\gamma \Gamma$ is well separated from 0 for all $\gamma$ in the integration process.

As before

$$\phi^{[i]}(A) \approx \frac{1}{k} \sum_{j=1}^{k} \lambda_j \phi^{[i]}(\gamma \lambda_j)(\lambda_j I - hL)^{-1},$$

where now $\lambda_j$ are the equally spaced points along the contour $\Gamma$.

Note: The inverse matrices no longer depend of $\gamma$. 
Cauchy integral approach

To achieve computational savings we can use the formula

\[ \phi^{[i]}(A) = \phi^{[i]}(\gamma h L) = \frac{1}{2\pi i} \int_{\Gamma} \phi^{[i]}(\gamma \lambda)(\lambda I - h L)^{-1} d\lambda, \]

where the contour \( \Gamma \) encloses the eigenvalues of \( \gamma h L \) and \( \gamma \Gamma \) is well separated from 0 for all \( \gamma \) in the integration process.

As before

\[ \phi^{[i]}(A) \approx \frac{1}{k} \sum_{j=1}^{k} \lambda_j \phi^{[i]}(\gamma \lambda_j)(\lambda_j I - h L)^{-1}, \]

where now \( \lambda_j \) are the equally spaced points along the contour \( \Gamma \).

When \( L \) arises from a finite difference approximation, we can benefit from its sparse block structure and find the action of the inverse matrices to a given vector by:

- **iterative methods** - *preconditioned conjugate gradient* and *multigrid methods*.
- **direct methods** - *CR, FFT, FACR, LU* factorization.
Numerical experiments

The methods

- **ETD RK4(Kr)** The fourth order method of Krogstad;
- **IF RK4** The fourth order Integrating Factor Runge–Kutta method (classical RK);
- **CF4** The fourth order Commutator Free Lie group method with affine action;
- **CF4A1** A fourth order CF method with action corresponding to nonautonomous FVF.
Kuramoto-Sivashinsky equation

\[ u_t = -u u_x - u_{xx} - u_{xxx}, \quad x \in [0, 32\pi] \]

with periodic boundary conditions and with the initial condition

\[ u(x, 0) = \cos\left(\frac{x}{16}\right)(1 + \sin\left(\frac{x}{16}\right)). \]
Kuramoto-Sivashinsky equation

\[
  u_t = -uu_x - u_{xx} - u_{xxxx}, \quad x \in [0, 32\pi]
\]

with periodic boundary conditions and with the initial condition

\[
  u(x, 0) = \cos\left(\frac{x}{16}\right)(1 + \sin\left(\frac{x}{16}\right)).
\]

We discretise the spatial part using Fourier spectral method. The transformed equation in the Fourier space is

\[
  \hat{u}_t = -\frac{ik}{2} \hat{u}^2 + (k^2 - k^4)\hat{u},
\]

\[
  (L\hat{u})(k) = (k^2 - k^4)\hat{u}(k) \quad \text{and} \quad N(\hat{u}, t) = -\frac{ik}{2}(F(\left(F^{-1}(\hat{u})\right)^2))
\]
Allen-Cahn equation

\[ u_t = \varepsilon u_{xx} + u - u^3, \quad x \in [-1, 1] \]

with \( \varepsilon = 0.01 \) and with boundary and initial conditions

\[ u(-1, t) = -1, \quad u(1, t) = 1, \quad u(x, 0) = .53x + .47 \sin(-1.5\pi x) \]
Allen-Cahn equation

\[ u_t = \varepsilon u_{xx} + u - u^3, \quad x \in [-1, 1] \]

with \( \varepsilon = 0.01 \) and with boundary and initial conditions

\[ u(-1, t) = -1, \quad u(1, t) = 1, \quad u(x, 0) = 0.53x + 0.47 \sin(-1.5\pi x) \]

After discretisation in space.

\[ u_t = Lu + N(u(t)) \]

where \( L = \varepsilon D^2 \), \( N(u(t)) = u - u^3 \) and \( D \) is the Chebyshev differentiation matrix.
Allen-Cahn equation
Conclusions

- Studied the connection between exponential integrators and Lie group methods.
- Proposed how to construct exponential integrators with algebra action arising from the solutions of differential equations with nonautonomous frozen vector fields.
- The proposed approach is promising for non-diagonal examples.
Future work

- Study the connections between the order of the action and the order of the methods.
- How to find a good algebra action?
- Stability analysis.
- Extensive numerical experiments.
References

- S. Krogstad, Generalized integrating factor methods for stiff PDEs, available at: http://www.ii.uib.no/~stein