A Perl program for the symbolic manipulation of flows of differential equations and its application to the analysis of defect-based error estimators for splitting methods

Harald Hofstätter

Institute for Analysis and Scientific Computing
Vienna University of Technology, Austria

mailto:hofi@harald-hofstaetter.at

Problem class, splitting methods

Defect and local error

Local error for Lie/Trotter

Local error estimator for Lie/Trotter

Perl modules

TimeExpression data types

SpaceExpression data types

substitute

t_derivative

differential

reduce_order

expand

Elementary differentials

Verification of eqs. (1)-(6)

Problem class and splitting methods

Problem class:

$$\frac{d}{dt}u(t) = H(u(t)) = A(u(t)) + B(u(t)), \quad \text{in general } A,B \text{ both nonlinear}$$
$$u(0) = u_0 \text{ given}$$

Example: cubic nonlinear Schrödinger equation:

$$A(u) = \tfrac{1}{2}\mathrm{i}\Delta u \quad \text{(linear)}$$
$$B(u) = -\mathrm{i}\,V_{\mathrm{ext}}\,u - \mathrm{i}\beta|u|^2 u$$

(or A, B exchanged)
Lie/Trotter splitting:

$$\mathcal{S}(t, u_0) = \mathcal{E}_B(t, \mathcal{E}_A(t, u_0))$$

Strang splitting:

$$\mathcal{S}(t, u_0) = \mathcal{E}_A(\tfrac{1}{2}t, \mathcal{E}_B(t, \mathcal{E}_A(\tfrac{1}{2}t, u_0)))$$

Problem class, splitting methods
Defect and local error
Local error for Lie/Trotter
Local error estimator for Lie/Trotter
Perl modules
TimeExpression data types
SpaceExpression data types
substitute
t_derivative
differential
reduce_order
expand
Elementary differentials
Verification of eqs. (1)-(6)

Defect and local error

Defect:

$$\mathcal{D}(t,u) \;=\; \mathcal{S}^{(1)}(t,u) \;=\; \tfrac{\partial}{\partial t}\mathcal{S}(t,u) - H(\mathcal{S}(t,u)).$$

Local error:

$$\mathcal{L}(t,u) \;=\; \mathcal{S}(t,u) - \mathcal{E}_H(t,u) \;=\; \int_0^t \mathcal{F}(\tau,t,u)\,\mathrm{d}\tau$$

with

$$\mathcal{F}(\tau,t,u) = \partial_2\mathcal{E}_H(t-\tau,\mathcal{S}(\tau,u))\cdot\mathcal{S}^{(1)}(\tau,u).$$

Proof. It holds

$$\begin{aligned}
\tfrac{\partial}{\partial\tau}\mathcal{E}_H(t-\tau,\mathcal{S}(\tau,u)) \;&=\; -H(\mathcal{E}_H(t-\tau,\mathcal{S}(\tau,u))) \\
&\quad\; +\partial_2\mathcal{E}_H(t-\tau,\mathcal{S}(\tau,u))\cdot\tfrac{\partial}{\partial\tau}\mathcal{S}(\tau,u) \\
&=\; -\partial_2\mathcal{E}_H(t-\tau,\mathcal{S}(\tau,u))\cdot H(\mathcal{S}(\tau,u)) \\
&\quad\; +\partial_2\mathcal{E}_H(t-\tau,\mathcal{S}(\tau,u))\cdot\tfrac{\partial}{\partial\tau}\mathcal{S}(\tau,u) \\
&=\; \partial_2\mathcal{E}_H(t-\tau,\mathcal{S}(\tau,u))\cdot\mathcal{S}^{(1)}(\tau,u),
\end{aligned}$$

such that

$$\begin{aligned}
\int_0^t \partial_2\mathcal{E}_H(t-\tau,\mathcal{S}(\tau,u))\cdot\mathcal{S}^{(1)}(\tau,u)\,\mathrm{d}\tau \;&=\; \mathcal{E}_H(t-\tau,\mathcal{S}(\tau,u))\Big|_{\tau=0}^{t} \\
&=\; \mathcal{S}(t,u)-\mathcal{E}_H(t,u) = \mathcal{L}(t,u).
\end{aligned}$$

Problem class, splitting methods
Defect and local error
Local error for Lie/Trotter
Local error estimator for Lie/Trotter
Perl modules
TimeExpression data types
SpaceExpression data types
substitute
t_derivative
differential
reduce_order
expand
Elementary differentials
Verification of eqs. (1)-(6)

Local error for Lie/Trotter splitting

Goal: Show

$$\mathcal{S}^{(1)}(t,u) = O(t) \;\Rightarrow\; \mathcal{L}(t,u) = O(t^2).$$

$$\mathcal{S}(t,u) = \mathcal{E}_B(t, \mathcal{E}_A(t,u)).$$

$$\tfrac{\partial}{\partial t}\mathcal{S}(t,u) = \partial_2\mathcal{S}(t,u)\cdot A(u) + B(\mathcal{S}(t,u)).$$

$$\begin{aligned}
\mathcal{S}^{(1)}(t,u) &= \mathcal{D}(t,u) = \tfrac{\partial}{\partial t}\mathcal{S}(t,u) - H(\mathcal{S}(t,u)) \\
&= \partial_2\mathcal{S}(t,u)\cdot A(u) - A(\mathcal{S}(t,u)) \\
&= \partial_2\mathcal{E}_B(t,\mathcal{E}_A(t,u))\cdot A(\mathcal{E}_A(t,u)) - A(\mathcal{E}_B(t,\mathcal{E}_A(t,u))) \\
&= \tilde{\mathcal{S}}^{(1)}(t,\mathcal{E}_A(t,u)) \qquad (1)
\end{aligned}$$

with

$$\tilde{\mathcal{S}}^{(1)}(t,v) = \partial_2\mathcal{E}_B(t,v)\cdot A(v) - A(\mathcal{E}_B(t,v)).$$

$\tilde{\mathcal{S}}^{(1)}(t,v)$ satisfies

$$\begin{aligned}
\tfrac{\partial}{\partial t}\tilde{\mathcal{S}}^{(1)}(t,v) &= B'(\mathcal{E}_B(t,v))\cdot\tilde{\mathcal{S}}^{(1)}(t,v) + [B,A](\mathcal{E}_B(t,v)) \qquad (2)\\
\tilde{\mathcal{S}}^{(1)}(0,v) &= 0
\end{aligned}$$

Problem class, splitting methods

Defect and local error

Local error for Lie/Trotter

Local error estimator for Lie/Trotter

Perl modules

TimeExpression data types

SpaceExpression data types

substitute

t_derivative

differential

reduce_order

expand

Elementary differentials

Verification of eqs. (1)-(6)

$$\frac{\partial}{\partial t}\mathcal{E}_F(t,u) = F(\mathcal{E}_F(t,u))$$
$$\Rightarrow \quad \frac{\partial}{\partial t}\partial_2\mathcal{E}_F(t,u)\cdot v = F'(\mathcal{E}_F(t,u))\cdot \partial_2\mathcal{E}_F(t,u)\cdot v$$

$\Rightarrow$ $\partial_2\mathcal{E}_F(t,u)$ is a "fundamental systen" of the linear differential equation

$$\frac{\partial}{\partial t}X(t,u) = F'(\mathcal{E}_F(t,u))\cdot X(t,u).$$

$$\partial_2\mathcal{E}_F(t,u)^{-1} = \partial_2\mathcal{E}_F(-t,\mathcal{E}_F(t,u)).$$

$\Rightarrow$ Variation of constants formula:

$$\begin{aligned}&\frac{\partial}{\partial t}X(t,u) = F'(\mathcal{E}_F(t,u))\cdot X(t,u) + R(t,u),\\ &X(0,u) = X_0(u)\end{aligned}$$

has the solution

$$X(t,u) = \partial_2\mathcal{E}_F(t,u)\cdot\left(X_0(u) + \int_0^t \partial_2\mathcal{E}_F(-\tau,\mathcal{E}_F(\tau,u))\cdot R(\tau,u)\,\mathrm{d}\tau\right).$$

Problem class, splitting methods
Defect and local error
Local error for Lie/Trotter
Local error estimator for Lie/Trotter
Perl modules
TimeExpression data types
SpaceExpression data types
substitute
t_derivative
differential
reduce_order
expand
Elementary differentials
Verification of eqs. (1)-(6)

Local error for Lie/Trotter splitting

$$\mathcal{S}^{(1)}(t,u) \;=\; \mathcal{D}(t,u) \;=\; \tfrac{\partial}{\partial t}\mathcal{S}(t,u) - H(\mathcal{S}(t,u)) \;=\; \tilde{\mathcal{S}}^{(1)}(t,\mathcal{E}_A(t,u))$$

where

$$\tilde{\mathcal{S}}^{(1)}(t,v) = \partial_2\mathcal{E}_B(t,v)\cdot\int_0^t \partial_2\mathcal{E}_B(-\tau,\mathcal{E}_B(\tau,v))\cdot[B,A](\mathcal{E}_B(\tau,v))\,\mathrm{d}\tau$$

From this integral representation it follows

$$\mathcal{D}(t,u) = O(t)$$

and

$$\mathcal{L}(t,u) = \int_0^t \partial_2\mathcal{E}_H(t-\tau,\mathcal{S}(\tau,u))\cdot\mathcal{D}(\tau,u)\,\mathrm{d}\tau = O(t^2)$$

Problem class, splitting methods

Defect and local error

Local error for Lie/Trotter

Local error estimator for Lie/Trotter

Perl modules

TimeExpression data types

SpaceExpression data types

substitute

t_derivative

differential

reduce_order

expand

Elementary differentials

Verification of eqs. (1)-(6)

A posteriori local error estimator for Lie/Trotter splitting

Error estimator . . . approximation of

$$\mathcal{L}(t,u) \;=\; \int_0^t \mathcal{F}(\tau,t,u)\,\mathrm{d}\tau \;=\; \int_0^t \partial_2\mathcal{E}_H(t-\tau,\mathcal{S}(\tau,u))\cdot\mathcal{S}^{(1)}(\tau,u)\,\mathrm{d}\tau.$$

by the trapezoidal rule:

$$\mathcal{P}(t,u) = \tfrac{1}{2}t\mathcal{F}(t,t,u) = \tfrac{1}{2}t\mathcal{D}(t,u) = \tfrac{1}{2}t\mathcal{S}^{(1)}(t,u).$$

Error of error estimate . . . Peano representation

$$\mathcal{P}(t,u)-\mathcal{L}(t,u) = \int_0^t K_1(\tau,t)\tfrac{\partial}{\partial\tau}\mathcal{F}(\tau,t,u)\,\mathrm{d}\tau$$

with kernel

$$K_1(\tau,t) = \tau - \tfrac{1}{2}t = O(t).$$

Goal: show that

$$\mathcal{P}(t,u)-\mathcal{L}(t,u) = O(t^3) \quad \Rightarrow \text{ asymptotical correctnesss}$$

Problem class, splitting methods
Defect and local error
Local error for Lie/Trotter
Local error estimator for Lie/Trotter
Perl modules
TimeExpression data types
SpaceExpression data types
substitute
t_derivative
differential
reduce_order
expand
Elementary differentials
Verification of eqs. (1)-(6)

A posteriori local error estimator for Lie/Trotter splitting

$$\begin{aligned}
\frac{\partial}{\partial\tau}\mathcal{F}(\tau,t,u) &= \partial_2\mathcal{E}_H(t-\tau,\mathcal{S}(\tau,u))\cdot\mathcal{S}^{(2)}(\tau,u) \\
&\quad +\partial_2^2\mathcal{E}_H(t-\tau,\mathcal{S}(\tau,u))(\mathcal{S}^{(1)}(\tau,u),\mathcal{S}^{(1)}(\tau,u)) \qquad (3)\\
&= \partial_2\mathcal{E}_H(t-\tau,\mathcal{S}(\tau,u))\cdot\mathcal{S}^{(2)}(\tau,u)+O(t)
\end{aligned}$$

where

$$\begin{aligned}
\mathcal{S}^{(2)}(t,u) &= \frac{\partial}{\partial t}\mathcal{S}^{(1)}(t,u)-H'(\mathcal{S}(t,u))\cdot\mathcal{S}^{(1)}(t,u)\\
&= \tilde{\mathcal{S}}^{(2)}(t,\mathcal{E}_A(t,u)) \qquad (4)
\end{aligned}$$

with

$$\tilde{\mathcal{S}}^{(2)}(t,v)=\partial_2\tilde{\mathcal{S}}^{(1)}(t,v)\cdot A(v)-A'(\mathcal{E}_B(t,v))\cdot\tilde{\mathcal{S}}^{(1)}(t,v)+[B,A](\mathcal{E}_B(t,v)).$$

$\tilde{\mathcal{S}}^{(2)}(t,v)$ satisfies

$$\begin{aligned}
\frac{\partial}{\partial t}\tilde{\mathcal{S}}^{(2)}(t,v) &= B'(\mathcal{E}_B(t,v))\cdot\tilde{\mathcal{S}}^{(2)}(t,v)\\
&\quad +B''(\mathcal{E}_B(t,v))(\tilde{\mathcal{S}}^{(1)}(t,v),\tilde{\mathcal{S}}^{(1)}(t,v))\\
&\quad -[B,[B,A]](\mathcal{E}_B(t,v))-[A,[B,A]](\mathcal{E}_B(t,v))\\
&\quad +2[B,A]'(\mathcal{E}_B(t,v))\cdot\tilde{\mathcal{S}}^{(1)}(t,v), \qquad (5)\\
\tilde{\mathcal{S}}^{(2)}(0,v) &= [B,A](v).
\end{aligned}$$

Problem class, splitting methods
Defect and local error
Local error for Lie/Trotter
Local error estimator for Lie/Trotter
Perl modules
TimeExpression data types
SpaceExpression data types
substitute
t_derivative
differential
reduce_order
expand
Elementary differentials
Verification of eqs. (1)-(6)

$\Rightarrow$ integral representation

$$\tilde{\mathcal{S}}^{(2)}(t,v) = \partial_2\mathcal{E}_B(t,v)\cdot[B,A](v) + \partial_2\mathcal{E}_B(t,v)\cdot\int_0^t \partial_2\mathcal{E}_B(-\tau,\mathcal{E}_B(\tau,v))\cdot \Big(B''(\mathcal{E}_B(\tau,v))(\tilde{\mathcal{S}}^{(1)}(\tau,v),\tilde{\mathcal{S}}^{(1)}(\tau,v)) - [B,[B,A]](\mathcal{E}_B(\tau,v)) - [A,[B,A]](\mathcal{E}_B(\tau,v)) + 2[B,A]'(\mathcal{E}_B(\tau,v))\cdot\tilde{\mathcal{S}}^{(1)}(\tau,v)\Big)\,\mathrm{d}\tau$$

$$\Rightarrow\ \mathcal{S}^{(2)}(\tau,u) = \partial_2\mathcal{E}_B(\tau,\mathcal{E}_A(\tau,u))\cdot[B,A](\mathcal{E}_A(\tau,u)) + O(t)$$

Altogether,

$$\mathcal{P}(t,u)-\mathcal{L}(t,u) = \int_0^t K_1(\tau,t)\tfrac{\partial}{\partial\tau}\mathcal{F}(\tau,t,u)\,\mathrm{d}\tau$$

$$= \int_0^t K_1(\tau,t)\partial_2\mathcal{E}_H(t-\tau,\mathcal{S}(\tau,u))\cdot\partial_2\mathcal{E}_B(\tau,\mathcal{E}_A(\tau,u))\cdot[B,A](\mathcal{E}_A(\tau,u))\,\mathrm{d}\tau + O(t^3)$$

$$= \int_0^t K_2(\tau,t)\tfrac{\partial}{\partial\tau}\Big(\partial_2\mathcal{E}_H(t-\tau,\mathcal{S}(\tau,u))\cdot\partial_2\mathcal{E}_B(\tau,\mathcal{E}_A(\tau,u))\cdot[B,A](\mathcal{E}_A(\tau,u))\Big)\,\mathrm{d}\tau + O(t^3)$$

where $K_2(\tau,t)=\frac{1}{2}\tau(t-\tau)=O(t^2)$ by partial integration.

Problem class, splitting methods
Defect and local error
Local error for Lie/Trotter
Local error estimator for Lie/Trotter
Perl modules
TimeExpression data types
SpaceExpression data types
substitute
t_derivative
differential
reduce_order
expand
Elementary differentials
Verification of eqs. (1)-(6)

A posteriori local error estimator for Lie/Trotter splitting

I.e., we have to show

$$\frac{\partial}{\partial\tau}\Big(\partial_2\mathcal{E}_H(t-\tau,\mathcal{S}(\tau,u))\cdot\partial_2\mathcal{E}_B(\tau,\mathcal{E}_A(\tau,u))\cdot[B,A](\mathcal{E}_A(\tau,u))\Big)=O(1),$$

and this holds because

$$\begin{aligned}
&\frac{\partial}{\partial\tau}\Big(\partial_2\mathcal{E}_H(t-\tau,\mathcal{S}(\tau,u))\cdot\partial_2\mathcal{E}_B(\tau,\mathcal{E}_A(\tau,u))\cdot[B,A](\mathcal{E}_A(\tau,u))\Big)\\
&\qquad=\Big[\partial_2\mathcal{E}_H(t-\tau,\mathcal{E}_B(\tau,v))\cdot\partial_2\mathcal{E}_B(\tau,v)\cdot[[B,A],A](v)\\
&\qquad\quad+\partial_2\mathcal{E}_H(t-\tau,\mathcal{E}_B(\tau,v))\cdot\partial_2\tilde{\mathcal{S}}^{(1)}(\tau,v)\cdot[B,A](v)\\
&\qquad\quad+\partial_2^2\mathcal{E}_H(t-\tau,\mathcal{E}_B(\tau,v))\big(\tilde{\mathcal{S}}^{(1)}(\tau,v),\partial_2\mathcal{E}_B(\tau,v)\cdot[B,A](v)\big)\Big]_{v=\mathcal{E}_{\boldsymbol{A}}(\boldsymbol{t},\boldsymbol{u})}
\end{aligned}\tag{6}$$

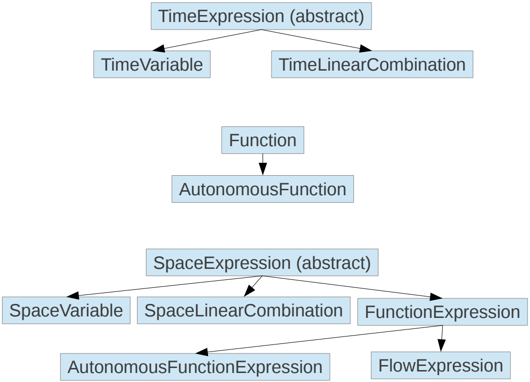
□

Problem class, splitting methods
Defect and local error
Local error for Lie/Trotter
Local error estimator for Lie/Trotter
Perl modules
TimeExpression data types
SpaceExpression data types
substitute
t_derivative
differential
reduce_order
expand
Elementary differentials
Verification of eqs. (1)-(6)

Perl modules for the symbolic manipulation of flows of differential equations

Functionality provided by 2 Perl modules `TimeExpression.pm` and `SpaceExpression.pm`

Data types / class hierarchy:



Problem class, splitting methods
Defect and local error
Local error for Lie/Trotter
Local error estimator for Lie/Trotter
Perl modules
TimeExpression data types
SpaceExpression data types
substitute
t_derivative
differential
reduce_order
expand
Elementary differentials
Verification of eqs. (1)-(6)

Usage of TimeExpression data types

```
use TimeExpression;
my $t = TimeVariable->new('t');
my $tau = TimeVariable->new('tau', '\tau');
my $tt1 = TimeLinearCombination->new($t, 2, $tau, 3);
my $tt2 = 2*$t + 3*$tau;      #overloaded operators
my $tt3 = TimeLinearCombination->new($tt1, 2, $t, -1);
my $tt4 = $tt1 - $tt2;
print $tt1->str() . "\n";
print $tt1->latex() . "\n";
print $tt2->str() . "\n";
print $tt3->str() . "\n";
print $tt4->str() . "\n";
```

Output:

```
2t+3tau
2t+3\tau
2t+3tau
3t+6tau
0
```

Problem class, splitting methods
Defect and local error
Local error for Lie/Trotter
Local error estimator for Lie/Trotter
Perl modules
TimeExpression data types
SpaceExpression data types
substitute
t_derivative
differential
reduce_order
expand
Elementary differentials
Verification of eqs. (1)-(6)

Usage of SpaceExpression data types

Define expressions

$$\mathcal{S}(t,u) = \mathcal{E}_B(t, \mathcal{E}_A(t,u))$$

$$C(u) = [A,B](u) = A'(u) \cdot B(u) - B'(u) \cdot A(u)$$

```
use TimeExpression;
use SpaceExpression;
my $t = TimeVariable->new('t');
my $u = SpaceVariable->new('u');
my $A = AutonomousFunction->new('A');
my $B = AutonomousFunction->new('B');
my $Stu = FlowExpression->new($B, $t,
            FlowExpression->new($A, $t, $u));
my $Cu = AutonomousFunctionExpression->new($A, $u,
            AutonomousFunctionExpression->new($B, $u))
        -AutonomousFunctionExpression->new($B, $u,
            AutonomousFunctionExpression->new($A, $u));
print $Stu->str() . "\n";
print $Cu->str() . "\n";
```

Output:

```
E_B[t,E_A[t,u]]
-B{1}[u](B[u])+A{1}[u](B[u])
```

Method `substitute`

Define $\mathcal{S}(t,u)$ by substituting $v \to \mathcal{E}_A(t,u)$ in $\mathcal{E}_B(t,v)$:

```
my $v = SpaceVariable->new('v');
my $E_Atu = FlowExpression->new($A, $t, $u);
my $E_Btv = FlowExpression->new($B, $t, $v);
my $Stu = $E_Btv->substitute($v, $E_Atu);
print $Stu->str() . "\n";
```

Output:

```
E_B[t,E_A[t,u]]
```

Define $D(u) = [A,[A,B]](u)$ by substituting $B \to [A,B]$ in $C(u) = [A,B](u)$:

```
# $Cu already defined
my $Du = $Cu->substitute($B, $Cu, $u);
print $Du->str() . "\n";
```

Output:

```
-A{2}[u](B[u],A{1}[u](B[u])-B{1}[u](B[u]))-A{1}[u](B{1}
[u](A{1}[u](B[u])-B{1}[u](B[u])))+B{2}[u](B[u],A{1}[u](
B[u])-B{1}[u](B[u]))+B{1}[u](B{1}[u](A{1}[u](B[u])-B{1}
[u](B[u]))+A{1}[u](A{1}[u](B[u])-B{1}[u](B[u]))
```

Method `t_derivative`

Define $\mathcal{S}^{(1)}(t,u) = \frac{\partial}{\partial t}\mathcal{S}(t,u) - A(\mathcal{S}(t,u)) - B(\mathcal{S}(t,u))$:

```
my $S1tu = $Stu->t_derivative($t)
          -AutonomousFunctionExpresssion->new($A, $Stu)
          -AutonomousFunctionExpresssion->new($B, $Stu);
print $S1tu->str() . "\n";
```

Output:

```
E_B{0,1}[t,E_A[t,u]](A[E_A[t,u]])-A[E_B[t,E_A[t,u]]]
```

Note that the output corresponds to

$$\mathcal{S}^{(1)}(t,u) = \partial_2\mathcal{E}_B(t,\mathcal{E}_A(t,u)) \cdot A(\mathcal{E}_A(t,u)) - A(\mathcal{E}_B(t,\mathcal{E}_A(t,u)))$$

$\rightsquigarrow$ (1)

Method `differential` (1)

Compute higher-order Fréchet derivatives of

$$Z(u) = F(\mathcal{E}_F(t,u)) - \partial_2\mathcal{E}_F(t,u)\cdot F(u) = 0$$

```perl
my $F = AutonomousFunction->new('F');
my $Zu = AutonomousFunctionExpression->new($F,
            FlowExpression->new($F, $t, $u))
        -FlowExpression->new($F, $t, $u,
            AutonomousFunctionExpression->new($F, $u));
my $Z1uv = $Zu->differential($u, $v);
my $w = SpaceVariable->new('w');
my $Z2uvw = $Z1uv->differential($u, $w);
print $Z1uv->str() . "\n";
print $Z2uvw->str() . "\n";
```

Output:

```
-E_F{0,1}[t,u](F{1}[u](v))+F{1}[E_F[t,u]](E_F{0,1}[t,u]
(v))-E_F{0,2}[t,u](F[u],v)
-E_F{0,1}[t,u](F{2}[u](v,w))+F{2}[E_F[t,u]](E_F{0,1}[t,
u](v),E_F{0,1}[t,u](w))-E_F{0,2}[t,u](F{1}[u](w),v)+F{1
}[E_F[t,u]](E_F{0,2}[t,u](v,w))-E_F{0,2}[t,u](F{1}[u](v
),w)-E_F{0,3}[t,u](F[u],v,w)
```


Problem class, splitting methods
Defect and local error
Local error for Lie/Trotter
Local error estimator for Lie/Trotter
Perl modules
TimeExpression data types
SpaceExpression data types
substitute
t_derivative
differential
reduce_order
expand
Elementary differentials
Verification of eqs. (1)-(6)

Methods `differential` (2) and `reduce_order`

We obtain the identities

$$F(\mathcal{E}_F(t,u)) - \partial_2\mathcal{E}_F(t,u)\cdot F(u) = 0$$

$$F'(\mathcal{E}_F(t,u))\cdot\partial_2\mathcal{E}_F(t,u)\cdot v - \partial_2^2\mathcal{E}_F(t,u)(F(u),v) - \partial_2\mathcal{E}_F(t,u)\cdot F'(u)\cdot v = 0$$

$$\begin{aligned}F''(\mathcal{E}_F(t,u))(\partial_2\mathcal{E}_F(t,u)\cdot v, \partial_2\mathcal{E}_F(t,u)\cdot w) + F'(\mathcal{E}_F(t,u))\cdot\partial_2^2\mathcal{E}_F(t,u)(v,w)\\ -\partial_2^3\mathcal{E}_F(t,u)(F(u),v,w) - \partial_2^2\mathcal{E}_F(t,u)(F'(u)\cdot v, w)\\ -\partial_2^2\mathcal{E}_F(t,u)(F'(u)\cdot w, v) - \partial_2\mathcal{E}_F(t,u)\cdot F''(u)(v,w) = 0\end{aligned}$$

The method `reduce_order` transforms expressions of the form of the highest order derivative in these identities by means of these identities:

$$\partial_2\mathcal{E}_F(t,u)\cdot F(u) \;\to\; F(\mathcal{E}_F(t,u))$$

$$\partial_2^2\mathcal{E}_F(t,u)(F(u),v) \;\to\; F'(\mathcal{E}_F(t,u))\cdot\partial_2\mathcal{E}_F(t,u)\cdot v - \partial_2\mathcal{E}_F(t,u)\cdot F'(u)\cdot v$$

$$\begin{aligned}\partial_2^3\mathcal{E}_F(t,u)(F(u),v,w) \;\to\; & F''(\mathcal{E}_F(t,u))(\partial_2\mathcal{E}_F(t,u)\cdot v, \partial_2\mathcal{E}_F(t,u)\cdot w)\\ & + F'(\mathcal{E}_F(t,u))\cdot\partial_2^2\mathcal{E}_F(t,u)(v,w) - \partial_2^2\mathcal{E}_F(t,u)(F'(u)\cdot v, w)\\ & -\partial_2^2\mathcal{E}_F(t,u)(F'(u)\cdot w, v) - \partial_2\mathcal{E}_F(t,u)\cdot F''(u)(v,w)\end{aligned}$$

Similarly for higher derivatives of analogous form

Method `expand`

(Higher-order) Fréchet derivative ... (multi-)linear map

(multi-)linear map applied to linear combination(s)
$\longrightarrow$ linear combination of (multi-)linear maps

Examples:

$$\partial_2\mathcal{E}_A(t,u)\cdot(2v+3w) \;\longrightarrow\; 2\partial_2\mathcal{E}_A(t,u)\cdot v + 3\partial_2\mathcal{E}_A(t,u)\cdot w$$

$$A''(u)(v+w,v+w) \;\longrightarrow\; A''(u)(v,v)+2A''(u)(v,w)+A''(u)(w,w)$$

Problem class, splitting methods
Defect and local error
Local error for Lie/Trotter
Local error estimator for Lie/Trotter
Perl modules
TimeExpression data types
SpaceExpression data types
substitute
t_derivative
differential
reduce_order
expand
Elementary differentials
Verification of eqs. (1)-(6)

Demonstration: elementary differentials

$$
\begin{aligned}
y'(t) &= F(y(t)) \\
y''(t) &= F'(y(t)) \cdot F(y(t)) \\
y'''(t) &= F''(y(t))(F(y(t), F(y(t)) + F'(y(t)) \cdot F'(y(t)) \cdot F(y(t)) \\
&\vdots
\end{aligned}
$$

terms = # elementary differentials (Butcher trees) of given order

available from the literature:

order	1	2	3	4	5	6	7	8	9	10	11
# terms	1	1	2	4	9	20	48	115	286	719	1842

$\rightsquigarrow$ `elementary_differentials.pl`

Problem class, splitting methods
Defect and local error
Local error for Lie/Trotter
Local error estimator for Lie/Trotter
Perl modules
TimeExpression data types
SpaceExpression data types
substitute
t_derivative
differential
reduce_order
expand
Elementary differentials
Verification of eqs. (1)-(6)

Demonstration: verification of equations (1)-(6)

$\rightsquigarrow$ `defect_lie_trotter.pl`

Problem class, splitting methods
Defect and local error
Local error for Lie/Trotter
Local error estimator for Lie/Trotter
Perl modules
TimeExpression data types
SpaceExpression data types
substitute
t_derivative
differential
reduce_order
expand
Elementary differentials
Verification of eqs. (1)-(6)

Thank you for your attention!

A Perl program for the symbolic manipulation of flows of differential equations and its application to the analysis of defect-based error estimators for splitting methods

Harald Hofstätter

Institute for Analysis and Scientific Computing
Vienna University of Technology, Austria

mailto:hofi@harald-hofstaetter.at

Fakultät für Mathematik, Informatik und Physik
Universität Innsbruck
Institut für Mathematik

Technikerstraße 19a
A-6020 Innsbruck
Telefon: +43 512 507 53801

E-Mail: mathematik@uibk.ac.at
http://www.uibk.ac.at/mathematik/
Fax: +43 512 507 53899

MATHEMATIKKOLLOQUIUM

Das Institut für Mathematik lädt zu folgendem Vortrag ein:

Harald Hofstätter

TU Wien

A Perl program for the symbolic manipulation of flows of differential equations and its application to the analysis of defect-based error estimators for splitting methods

For the proof of the asymptotical correctness of certain defect-based a posteriori local error estimators for splitting methods, suitable (integral) representations of local error expansions have to be derived, from which the order of the local error can directly be inferred. In the linear case this is achieved by tedious but relatively straightforward manipulations of operator exponentials.
In the nonlinear case, however, to obtain these local error representaions explicitly, complicated expressions involving nonlinear flows of differential equations and higher-order Frechet derivatives of such flows have to be handled. Performing these tedious and error prone calculations manually very quickly becomes unreasonable or even impossible. We describe an especially tailored tool for performing such manipulations symbolically, whose development was strongly facilitated by employing the object-orientedness and the superior string and hash manipulation features of the Perl programming language.

Zeit: **Donnerstag, den 24. Januar 2013 um 17:15 Uhr**

Ort: **Bauing.-Gebäude, Technikerstraße 13, HSB 7**

Gäste sind herzlich willkommen!

Mechthild Thalhammer