Numerical Extension ODE in KeYmaera X

Evelyn Kuo
Differential equations are fundamental to hybrid systems

\[ \begin{align*}
\frac{d}{dt} x(t) &= f(t, x(t)) \\
x(t_0) &= x_0
\end{align*} \]

• Differential equations model the physical world
• Initial value problem

• Analysis of differential equations
  • Mathematics: ODE theory (existence, uniqueness), numerical methods
  • Theorem provers: KeYmaera X
Pros/Cons of KeYmaera X ODE Tactics

Actual Solution
• Easy to prove things knowing exact solution
  \( \{x' = a\} \)
• Not always easy to find or even represent solutions
  \( \{x' = x\} \)
Pros/Cons of KeYmaera X ODE Tactics

Actual Solution
• Easy to prove things knowing exact solution
  \{x' = a\}
• Not always easy to find or even represent solutions
  \{x' = x\}

Differential Invariant
• Analyze trends in how quantities change without knowing exact solution
  \[\{x'=-y, y'=x\}\] \(x^2+y^2 = 1\)
• Trends may not help in proof
  \[\{x'=-x\}\] \(x > 0\)
Pros/Cons of KeYmaera X ODE Tactics

**Differential Ghost**
- Use ghost variable to analyze how quantities change without knowing exact solution

\[ \{x'=-x, \ y'=y/2\}\ x\ y^2 = 1 \]
- Ghost variable solution needs to exist for at least as long as original diffeq

\[
x(t) = e^{-t} \quad y(t) = e^{t/2} \quad x(t)\ y(t)^2 = 1
\]
### Pros/Cons of KeYmaera X ODE Tactics

<table>
<thead>
<tr>
<th>Actual Solution</th>
<th>Differential Invariant</th>
<th>Differential Ghost</th>
</tr>
</thead>
</table>
| • Easy to prove things knowing exact solution  
  \{x' = a\}  
• Not always easy to find or even represent solutions  
  \{x' = x\} | • Analyze trends in how quantities change without knowing exact solution  
  \[{x'=-y, y'=x}\] x^2+y^2 = 1  
• Trends may not help in proof  
  \[{x'=-x}\] x > 0 | • Use ghost variable to analyze how quantities change without knowing exact solution  
  \[{x'=-x, y'=y/2}\] x*y^2 = 1  
• Ghost variable solution needs to exist for at least as long as original diffeq |
Numerical methods can offer approximate solutions which are almost like actual solutions

- Require timestep specification
- Symbolic initial value problem
- How to transform post conditions to account for error

\[
\begin{align*}
\overline{y}_{i+1} &= \overline{y}_i + h \Phi(t_i, \overline{y}_i | h) \\
\overline{y}_0 &= y(t_0)
\end{align*}
\]
Theory: constructing approximations

- Given timestep $h$, numerical methods can yield discrete approximations

\[
\begin{align*}
\overline{y}_{i+1} &= \overline{y}_i + h\Phi(t_i, \overline{y}_i | h) \\
\overline{y}_0 &= y(t_0)
\end{align*}
\]
Theory: global error of discrete approximation

Define error $E_i := |\bar{y}_i - y(t_i)|$, want to find global bound on error
Theory: global error of discrete approximation

Introduce new sequence \( \{y(i+1) = y(t_i) + h\Phi(t_i, y(t_i)|h)\} \) (one step) approximations

Truncation error

\[
\begin{align*}
  y_{i+1} &= y_i + h\Phi(t_i, y_i|h) \\
  y_0 &= y(t_0)
\end{align*}
\]
Theory: global error of discrete approximation

If \( \phi \) sufficiently nice, exists constant \( C \) such that

\[
|\hat{y} - \hat{z}| \leq |y - z| C
\]
Theory: global error of discrete approximation

If $N = T/h$, global error of discrete approximation

$$E_{i+1} \leq CE_i + \tau \leq C^{i+1}E_0 + \tau \sum_{j=0}^{i-1} C^j \leq \frac{C^N - 1}{C - 1}$$

$$E_3 := |\bar{y}_3 - y(t_3)|$$

\[ \begin{align*}
\bar{y}_{i+1} &= \bar{y}_i + h\Phi(t_i, \bar{y}_i | h) \\
\bar{y}_0 &= y(t_0)
\end{align*} \]
Theory: constructing approximations

• Use linear interpolation to convert to continuous approximations

\[
\begin{align*}
\overline{y}_{i+1} &= \overline{y}_i + h \Phi(t_i, \overline{y}_i | h) \\
\overline{y}_0 &= y(t_0)
\end{align*}
\]
Theory: global error of continuous approximation

If $\epsilon_a$ is discrete approximation error, then continuous approximation error of linear interpolation is bounded by $\epsilon_a + hL$

$L$: max derivative of $x$
Theory: global error

Putting it together
If desired error bound is $\epsilon$, Solve for $h$

$$\epsilon = \tau \frac{C^N - 1}{C - 1} + hL$$
Theory: Proof Rule

\[ \exists e > 0 \ \forall t \ (0 \leq t \leq T \ & \ |x - x_{\text{approx}}(x0,t,eps)| < e \rightarrow P(x)) \]

\[ \text{[dApprox]} \]

\[ [t := 0; \ x0 := x; \ \{ t' = 1, \ x' = f(t,x) \ & \ t \leq T \}] \ P(x) \]
Implementation:

\[ x' = f(t, x) \& t \leq T \] 

error tolerance 

\[ x_{\text{approx}}(t, x_0) \]

KeYmaera X

ODE Approximation (numerical method selection)

KeYmaera X
Implementation:

KeYmaera X

x' = f(t,x) & t <= T
error tolerance

ODE Approximation
(numerical method selection)

x_approx(t,x0)

KeYmaera X

timestep (h)
calculation from error tolerance bound

symbolic numerical method

h

discrete approximation

symbolic linear interpolation

x' = f(t,x) & t <= T

continuous approximation
Discussion and Further Work

- Comparison of existing KeYmaera X tools for analyzing ODEs
- Theory of how to convert desired error tolerance into continuous approximation
- Implementation

- Convert python function into KeYmaera X function
- Handle non-initial value symbols in function $x' = f(t, x)$
- Extend to other numerical methods