Differential-Algebraic Dynamic Logic for KeYmaera X
CPS Grand Prix

Benjamin Lim    Yao Chong Lim

School of Computer Science, Carnegie Mellon University

December 11, 2018
Motivation

Imprecision is everywhere in actual Cyber-Physical systems...
Motivation

Imprecision is everywhere in actual Cyber-Physical systems... but how do we precisely model its semantics and prove guarantees?
\[ x \leq m \rightarrow \left[ \{ x' = v, v' = a \& v \geq 0 \} \right] x \leq m \]
Differential-Algebraic Dynamic Logic (dALC)

\[ x \leq m \rightarrow [\{x' = v, v' = a \& v \geq 0\}] x \leq m \]
Differential-Algebraic Dynamic Logic (dALC)

\[ x \leq m \implies \exists \delta. (x' = v, v' = a + \delta \land v \geq 0 \land \delta^2 \leq \frac{|v|}{100}) \]
Differential-Algebraic Dynamic Logic (dALC)

\[ dALC = dL + \text{existentially quantified ODEs} \]

\[ \{ \exists \bar{y}. (x_1' = \theta_1, x_2' = \theta_2, \cdots, x_n' = \theta_n \& Q) \} \]
$x^2 + y^2 = 1 \rightarrow [\{x' = -y, y' = x\}] \ x^2 + y^2 \leq 1$
dAL Example (Perturbed Circular Motion)

\[ x^2 + y^2 = 1 \rightarrow \left[ \exists e. (x' = -y + e, y' = x \& x \cdot e \leq 0) \right] x^2 + y^2 \leq 1 \]
Uniform Substitution (Abridged)

- $[x := e]P(x) \rightarrow P(e)$ \hspace{1cm} Axiom Schema

- Side conditions necessary...
  - but which ones?
  - Key observation: Never bind a free variable that was free!
Uniform Substitution (Abridged)

- $[x := e]P(x) \rightarrow P(e)$  Axiom Schema
- $[x := x + x][y := 3]x > 0 \rightarrow [y := 3]x + x > 0$

Valid instance!

$[x := x + y][y := 3]x > 0 \rightarrow [y := 3]x + y > 0$

Invalid instance!

Side conditions necessary...

but which ones?

Key observation: Never bind a free variable that was free!
Uniform Substitution (Abridged)

- $[x := e]P(x) \rightarrow P(e)$  Axiom Schema
- $[x := x + x][y := 3]x > 0 \rightarrow [y := 3]x + x > 0$  Valid instance!
Uniform Substitution (Abridged)

- $[x := e]P(x) \rightarrow P(e)$  Axiom Schema
- $[x := x + x][y := 3]x > 0 \rightarrow [y := 3]x + x > 0$  Valid instance!
- $[x := x + y][y := 3]x > 0 \rightarrow [y := 3]x + y > 0$
Uniform Substitution (Abridged)

- \([x := e]P(x) \rightarrow P(e)\)  Axiom Schema
- \([x := x + x][y := 3]x > 0 \rightarrow [y := 3]x + x > 0\)  Valid instance!
- \([x := x + y][y := 3]x > 0 \rightarrow [y := 3]x + y > 0\)  Invalid instance!
Uniform Substitution (Abridged)

- \([x := e] P(x) \rightarrow P(e)\)  
  Axiom Schema

- \([x := x + x][y := 3] x > 0 \rightarrow [y := 3] x + x > 0\)  
  Valid instance!

- \([x := x + y][y := 3] x > 0 \rightarrow [y := 3] x + y > 0\)  
  Invalid instance!
Uniform Substitution (Abridged)

- $[x := e]P(x) \rightarrow P(e)$  Axiom Schema
- $[x := x + x][y := 3]x > 0 \rightarrow [y := 3]x + x > 0$  Valid instance!
- $[x := x + y][y := 3]x > 0 \rightarrow [y := 3]x + y > 0$  Invalid instance!
- Side conditions necessary...
Uniform Substitution (Abridged)

- \([x := e]P(x) \rightarrow P(e)\)  Axiom Schema
- \([x := x + x][y := 3]x > 0 \rightarrow [y := 3]x + x > 0\)  Valid instance!
- \([x := x + y][y := 3]x > 0 \rightarrow [y := 3]x + y > 0\)  Invalid instance!
- Side conditions necessary...but which ones?

Key observation: Never bind a free variable that was free!
Uniform Substitution (Abridged)

- \([x := e]P(x) \rightarrow P(e)\)  \text{ Axiom Schema}
- \([x := x + x][y := 3]x > 0 \rightarrow [y := 3]x + x > 0\)  \text{ Valid instance!}
- \([x := x + y][y := 3]x > 0 \rightarrow [y := 3]x + y > 0\)  \text{ Invalid instance!}
- Side conditions necessary...but which ones?
- Key observation: Never bind a free variable that was free!
Uniform Substitution (Abridged)

Instead of schema each with their own unique side conditions...

\[ [x := e] P(x) \rightarrow P(e) \quad (\text{+some set of side conditions}) \]
Uniform Substitution (Abridged)

Instead of schema each with their own unique side conditions...

\[ [x := e]P(x) \rightarrow P(e) \quad (\text{+some set of side conditions}) \]

You have substitution axioms (without side conditions)...

\[ [x := c()]p(x) \rightarrow p(c()) \]
Uniform Substitution (Abridged)

Instead of schema each with their own unique side conditions...

\[ [x := e]P(x) \rightarrow P(e) \quad (\text{+some set of side conditions}) \]

You have substitution axioms (without side conditions)...

\[ [x := c()]p(x) \rightarrow p(c()) \]

and generic admissibility rules for each logical construct (checked recursively) preventing capture of free variables.
Uniform Substitution (Abridged)

Upshot: A significantly reduced soundness-critical core that is easier to maintain and understand
Plan of Attack

- Modernize $d\mathcal{AL}$, providing a uniform substitution calculus for it similar to that for $d\mathcal{L}$.
Plan of Attack

- Modernize $d\mathcal{L}$, providing a uniform substitution calculus for it similar to that for $d\mathcal{L}$.
- Implement uniform substitution axioms into KeYmaeraX.
Plan of Attack

- Modernize $\mathcal{L}$, providing a uniform substitution calculus for it similar to that for $\mathcal{L}$.
- Implement uniform substitution axioms into KeYmaeraX.
- Implement derived axioms and tactics into KeYmaeraX.
Plan of Attack

- Modernize $\text{dA}\mathcal{L}$, providing a uniform substitution calculus for it similar to that for $\text{dL}$.
- Implement uniform substitution axioms into KeYmaeraX.
- Implement derived axioms and tactics into KeYmaeraX.
- Prove stuff!
dŁ Recap

**DW** \([c \land q(\bar{x})]p(\bar{x}) \leftrightarrow [c \land q(\bar{x})](q(\bar{x}) \rightarrow p(\bar{x}))\)

**DC** \(([c \land q(\bar{x})]p(\bar{x}) \leftrightarrow [c \land q(\bar{x}) \land r(\bar{x})]p(\bar{x})) \leftarrow [c \land q(\bar{x})]r(\bar{x})\)

**DE** \([x' = f(\bar{x}), c \land q(\bar{x})]p(\bar{x}) \leftrightarrow [x' = f(\bar{x}), c \land q(\bar{x})][x' := f(\bar{x})]p(\bar{x})\)

**DI** \(([c \land q(\bar{x})]p(\bar{x}) \leftrightarrow [?q(\bar{x})]p(\bar{x})) \leftarrow [c \land q(\bar{x})](p(\bar{x}))'\)
Recap

\[ \text{DW} \quad [c \& q(\bar{x})]p(\bar{x}) \leftrightarrow [c \& q(\bar{x})](q(\bar{x}) \rightarrow p(\bar{x})) \]

\[ \text{DC} \quad ([c \& q(\bar{x})]p(\bar{x}) \leftrightarrow [c \& q(\bar{x}) \land r(\bar{x})]p(\bar{x})) \leftarrow [c \& q(\bar{x})]r(\bar{x}) \]

\[ \text{DE} \quad [x' = f(\bar{x}), c \& q(\bar{x})]p(\bar{x}) \leftrightarrow [x' = f(\bar{x}), c \& q(\bar{x})][x' := f(\bar{x})]p(\bar{x}) \]

\[ \text{DI} \quad ([c \& q(\bar{x})]p(\bar{x}) \leftrightarrow [?q(\bar{x})]p(\bar{x})) \leftarrow [c \& q(\bar{x})](p(\bar{x}))' \]
dŁ Recap

DW \([c \land q(\vec{x})]p(\vec{x}) \leftrightarrow [c \land q(\vec{x})](q(\vec{x}) \rightarrow p(\vec{x}))\)

DC \([(c \land q(\vec{x}))p(\vec{x}) \leftrightarrow [c \land q(\vec{x}) \land r(\vec{x})]p(\vec{x})] \leftarrow [c \land q(\vec{x})]r(\vec{x})\)

DE \([x' = f(\vec{x}), c \land q(\vec{x})]p(\vec{x}) \leftrightarrow [x' = f(\vec{x}), c \land q(\vec{x})][x' := f(\vec{x})]p(\vec{x})\)

DI \([(c \land q(\vec{x})]p(\vec{x}) \leftrightarrow [?q(\vec{x})]p(\vec{x})]) \leftarrow [c \land q(\vec{x})](p(\vec{x}))'\)

\[
\begin{aligned}
x' &= f(x) &\land & Q
\end{aligned}
\]
dL Recap

\[\begin{align*}
\text{DW} & \quad [c \land q(\bar{x})]p(\bar{x}) \leftrightarrow [c \land q(\bar{x})](q(\bar{x}) \to p(\bar{x})) \\
\text{DC} & \quad ([c \land q(\bar{x})]p(\bar{x}) \leftrightarrow [c \land q(\bar{x}) \land r(\bar{x})]p(\bar{x})) \leftarrow [c \land q(\bar{x})]r(\bar{x}) \\
\text{DE} & \quad [\bar{x}' = f(\bar{x}), c \land q(\bar{x})]p(\bar{x}) \leftrightarrow [\bar{x}' = f(\bar{x}), c \land q(\bar{x})][\bar{x}' := f(\bar{x})]p(\bar{x}) \\
\text{DI} & \quad ([c \land q(\bar{x})]p(\bar{x}) \leftrightarrow [?q(\bar{x})]p(\bar{x})) \leftarrow [c \land q(\bar{x})](p(\bar{x}))'
\end{align*}\]
Attempts at Uniformity

\begin{align*}
\text{DW} & \quad [c \& q(\bar{x})]p(\bar{x}) \iff [c \& q(\bar{x})](q(\bar{x}) \rightarrow p(\bar{x})) \\
\text{DC} & \quad ([c \& q(\bar{x})]p(\bar{x}) \iff [c \& q(\bar{x}) \land r(\bar{x})]p(\bar{x})) \iff [c \& q(\bar{x})]r(\bar{x}) \\
\text{DE} & \quad [x' = f(\bar{x}), c \& q(\bar{x})]p(\bar{x}) \iff [x' = f(\bar{x}), c \& q(\bar{x})][x' := f(\bar{x})]p(\bar{x}) \\
\text{DI} & \quad ([c \& q(\bar{x})]p(\bar{x}) \iff [?q(\bar{x})]p(\bar{x})) \iff [c \& q(\bar{x})](p(\bar{x}))'
\end{align*}
Attempts at Uniformity

\[ \text{DW} \quad [c \land q(\vec{x})]p(\vec{x}) \leftrightarrow [c \land q(\vec{x})](q(\vec{x}) \rightarrow p(\vec{x})) \]

\[ \text{DC} \quad ([c \land q(\vec{x})]p(\vec{x}) \leftrightarrow [c \land q(\vec{x}) \land r(\vec{x})]p(\vec{x})) \leftarrow [c \land q(\vec{x})]r(\vec{x}) \]

\[ \text{DE} \quad [x' = f(\vec{x}), c \land q(\vec{x})]p(\vec{x}) \leftrightarrow [x' = f(\vec{x}), c \land q(\vec{x})][x' := f(\vec{x})]p(\vec{x}) \]

\[ \text{DI} \quad ([c \land q(\vec{x})]p(\vec{x}) \leftrightarrow [?q(\vec{x})]p(\vec{x})) \leftarrow [c \land q(\vec{x})](p(\vec{x}))' \]
Attempts at Uniformity

\[\text{DW} \quad [c \land q(\bar{x})] p(\bar{x}) \leftrightarrow [c \land q(\bar{x})](q(\bar{x}) \to p(\bar{x}))\]

\[\text{DC} \quad ([c \land q(\bar{x})] p(\bar{x}) \leftrightarrow [c \land q(\bar{x}) \land r(\bar{x})] p(\bar{x})) \leftarrow [c \land q(\bar{x})] r(\bar{x})\]

\[\text{DE} \quad [\bar{x}' = f(\bar{x}), c \land q(\bar{x})] p(\bar{x}) \leftrightarrow [\bar{x}' = f(\bar{x}), c \land q(\bar{x})][\bar{x}' := f(\bar{x})] p(\bar{x})\]

\[\text{DI} \quad ([c \land q(\bar{x})] p(\bar{x}) \leftrightarrow [\forall q(\bar{x})] p(\bar{x})) \leftarrow [c \land q(\bar{x})](p(\bar{x}))'\]

\[([\exists \bar{y}. (c \land q(\bar{x}, \bar{y}))] p(\bar{x}) \leftrightarrow \forall \bar{y}. [\forall q(\bar{x}, \bar{y})] p(\bar{x})) \leftarrow \forall \bar{y}. [c \land q(\bar{x}, \bar{y})](p(\bar{x}))'\]
Attempts at Uniformity

\[ \text{DW} \quad [c \land q(\bar{x})]p(\bar{x}) \leftrightarrow [c \land q(\bar{x})](q(\bar{x}) \rightarrow p(\bar{x})) \]

\[ \text{DC} \quad ([c \land q(\bar{x})]p(\bar{x}) \leftrightarrow [c \land q(\bar{x}) \land r(\bar{x})]p(\bar{x})) \leftrightarrow [c \land q(\bar{x})]r(\bar{x}) \]

\[ \text{DE} \quad [x' = f(\bar{x}), c \land q(\bar{x})]p(\bar{x}) \leftrightarrow [x' = f(\bar{x}), c \land q(\bar{x})][x' := f(\bar{x})]p(\bar{x}) \]

\[ \text{DI} \quad ([c \land q(\bar{x})]p(\bar{x}) \leftrightarrow [\neg q(\bar{x})]p(\bar{x})) \leftrightarrow [c \land q(\bar{x})](p(\bar{x}))' \]

\[ ([\exists \bar{y}. (c \land q(\bar{x}, \bar{y}))]p(\bar{x}) \leftrightarrow \forall \bar{y}. [\neg q(\bar{x}, \bar{y})]p(\bar{x})) \leftrightarrow \forall \bar{y}. [c \land q(\bar{x}, \bar{y})](p(\bar{x}))' \]

Wrong!
Attempts at Uniformity

\[
([\exists \bar{y}. (c \& q(\bar{x}, \bar{y}))]p(\bar{x}) \leftrightarrow \forall \bar{y}. [?q(\bar{x}, \bar{y})]p(\bar{x})) \leftarrow \forall \bar{y}. [c \& q(\bar{x}, \bar{y})](p(\bar{x}))'
\]
Attempts at Uniformity

\(
([\exists \bar{y}. (c \& q(\bar{x}, \bar{y}))]p(\bar{x}) \leftrightarrow \forall \bar{y}. [?q(\bar{x}, \bar{y})]p(\bar{x})) \leftarrow \forall \bar{y}. [c \& q(\bar{x}, \bar{y})](p(\bar{x}))'
\)

Counterexample:

\(
( [\exists y. (x' = y, z' = -1 \& y \geq z)] x \geq 0 \leftrightarrow \forall y. [?y \geq z] x \geq 0) \\
\leftarrow \forall y. [\{x' = y, z' = -1 \& y \geq z\}] (x \geq 0)'
\)
Attempts at Uniformity

$$(\exists \bar{y}. (c \& q(\bar{x}, \bar{y})))p(\bar{x}) \iff \forall \bar{y}. (q(\bar{x}, \bar{y}))p(\bar{x}) \iff \forall \bar{y}. (c \& q(\bar{x}, \bar{y}))(p(\bar{x}))'$$

Counterexample:

$$(\{\exists y. (x' = y, z' = -1 \& y \geq z)\}) x \geq 0 \iff \forall y. (y \geq z) x \geq 0)$$

$$\iff \forall y. (\{x' = y, z' = -1 \& y \geq z\}) (x \geq 0)'$$

Pick a state with $x \geq 0 \land z \geq 0$: 
Attempts at Uniformity

\[
([\exists \bar{y}. (c \& q(\bar{x}, \bar{y}))] p(\bar{x}) \leftrightarrow \forall \bar{y}. [?q(\bar{x}, \bar{y})] p(\bar{x})) \leftrightarrow \forall \bar{y}. [c \& q(\bar{x}, \bar{y})](p(\bar{x})))'
\]

Counterexample:

\[
([\exists y. (x' = y, z' = -1 \& y \geq z)] x \geq 0 \leftrightarrow \forall y. [?y \geq z] x \geq 0) \\
\phantom{[\exists y. (x' = y, z' = -1 \& y \geq z)] x \geq 0} \leftrightarrow \forall y. [\{x' = y, z' = -1 \& y \geq z\}] (x \geq 0)'
\]

Pick a state with \(x \geq 0 \& z \geq 0\):

- Premise 1: \(\forall y. [\{x' = y, z' = -1 \& y \geq z\}] (x \geq 0)\)' True!
Attempts at Uniformity

\[ ([\exists \bar{y}. (c \land q(\bar{x}, \bar{y}))] p(\bar{x}) \leftrightarrow \forall \bar{y}. [? q(\bar{x}, \bar{y})] p(\bar{x})] \leftrightarrow \forall \bar{y}. [c \land q(\bar{x}, \bar{y})] (p(\bar{x}))') \]

Counterexample:

\[ (\{ \exists y. (x' = y, z' = -1 \land y \geq z) \}) \ x \geq 0 \leftrightarrow \forall y. [? y \geq z] x \geq 0 \]
\[ \leftrightarrow \forall y. [\{ x' = y, z' = -1 \land y \geq z \}] (x \geq 0)' \]

Pick a state with \( x \geq 0 \land z \geq 0 \):

- **Premise 1:** \( \forall y. \{ x' = y, z' = -1 \land y \geq z \} \) \( (x \geq 0)' \) \hspace{1cm} True!
- **Premise 2:** \( \forall y. [? y \geq z] x \geq 0 \) \hspace{1cm} True!
Attempts at Uniformity

\[
([\exists \bar{y}. (c \& q(\bar{x}, \bar{y}))]p(\bar{x}) \leftrightarrow \forall \bar{y}. [?q(\bar{x}, \bar{y})]p(\bar{x})) \leftarrow \forall \bar{y}. [c \& q(\bar{x}, \bar{y})](p(\bar{x}))
\]

Counterexample:

\[
([\exists y. (x' = y, z' = -1 \& y \geq z)] \land x \geq 0) \leftrightarrow \forall y. [?y \geq z]x \geq 0
\]

\[
\leftarrow \forall y. [\{x' = y, z' = -1 \& y \geq z\}] (x \geq 0)
\]

Pick a state with \(x \geq 0 \land z \geq 0\):

- Premise 1: \(\forall y. [\{x' = y, z' = -1 \& y \geq z\}] (x \geq 0)'\) True!
- Premise 2: \(\forall y. [?y \geq z]x \geq 0\) True!
- Conclusion: \([\{\exists y. (x' = y, z' = -1 \& y \geq z)\}]x \geq 0\) False??
Uniform Substitution for d\(\mathcal{AL}\)

\[
\begin{align*}
DC & \quad ([c \& q]p \leftrightarrow [c \& q \land r]p) \leftrightarrow [c \& q]r \\
DI & \quad ([c \& q]p \leftrightarrow [?q]p) \leftrightarrow [c \& q](p)'
\end{align*}
\]

\[
\begin{align*}
DW & \quad [c \& q]p \leftrightarrow [c \& q](q \rightarrow p) \\
DE & \quad [x' = f, c \& q]p \leftrightarrow [x' = f, c \& q][x' := f]p
\end{align*}
\]

\[
\begin{align*}
DAC & \quad ([\forall \bar{y}.(c \& q)] p \leftrightarrow [\forall \bar{y}.(c \& q \land r)] p) \leftrightarrow [\forall \bar{y}.(c \& q)] r \\
DAI & \quad ([\forall \bar{y}.(c \& q)] p \leftrightarrow \forall \bar{y}. [?q] p) \leftrightarrow [\forall \bar{y}.(c \& q)] (p)'
\end{align*}
\]

\[
\begin{align*}
DAW & \quad [\forall \bar{y}.(c \& q)] p \leftrightarrow [\forall \bar{y}.(c \& q)] (q \rightarrow p) \\
DAE & \quad [\forall \bar{y}.(x' = f, c \& q)] p \leftrightarrow \forall \bar{y}. [\forall \bar{y}.(x' = f, c \& q)] [x' := f] p
\end{align*}
\]
Uniform Substitution for $dAL$

$$
\text{DC} \quad ([c \& q]p \leftrightarrow [c \& q \land r]p) \leftarrow [c \& q]r
$$

$$
\text{DI} \quad ([c \& q]p \leftrightarrow [?q]p) \leftarrow [c \& q](p)'
$$

$$
\text{DW} \quad [c \& q]p \leftrightarrow [c \& q](q \rightarrow p)
$$

$$
\text{DE} \quad [x' = f, c \& q]p \leftrightarrow [x' = f, c \& q][x' := f]p
$$

$$
\text{DAC} \quad ([\exists \bar{y}.(c \& q)]) p \leftrightarrow ([\exists \bar{y}.(c \& q \land r)]) p) \leftarrow ([\exists \bar{y}.(c \& q)]) r
$$

$$
\text{DAI} \quad ([\exists \bar{y}.(c \& q)]) p \leftrightarrow \forall \bar{y}. [?q] p) \leftarrow ([\exists \bar{y}.(c \& q)]) (p)'
$$

$$
\text{DAW} \quad ([\exists \bar{y}.(c \& q)]) p \leftrightarrow ([\exists \bar{y}.(c \& q)]) (q \rightarrow p)
$$

$$
\text{DAE} \quad [\exists \bar{y}.(x' = f, c \& q)]) p \leftrightarrow \forall \bar{y}. [\exists \bar{y}.(x' = f, c \& q)]) [x' := f] p
$$
Uniform Substitution for d\(\mathcal{A}\)

\[
\begin{align*}
\text{DC} & \quad ([c \& q]p \leftrightarrow [c \& q \land r]p) \leftarrow [c \& q]r \\
\text{DI} & \quad ([c \& q]p \leftrightarrow [?q]p) \leftarrow [c \& q](p)'
\end{align*}
\]

\[
\begin{align*}
\text{DW} & \quad [c \& q]p \leftrightarrow [c \& q](q \rightarrow p) \\
\text{DE} & \quad [x' = f, c \& q]p \leftrightarrow [x' = f, c \& q][x' := f]p
\end{align*}
\]

\[
\begin{align*}
\text{DAC} & \quad ([\exists \bar{y}.(c \& q)]p \leftrightarrow [\exists \bar{y}.(c \& q \land r)]p) \leftarrow [\exists \bar{y}.(c \& q)]r \\
\text{DAI} & \quad ([\exists \bar{y}.(c \& q)]p \leftrightarrow \forall \bar{y}.[?q]p) \leftarrow [\exists \bar{y}.(c \& q)](p)'
\end{align*}
\]

\[
\begin{align*}
\text{DAW} & \quad [\exists \bar{y}.(c \& q)]p \leftrightarrow [\exists \bar{y}.(c \& q)](q \rightarrow p) \\
\text{DAE} & \quad [\exists \bar{y}.(x' = f, c \& q)]p \leftrightarrow \forall \bar{y}.[\exists \bar{y}.(x' = f, c \& q)] [x' := f]p
\end{align*}
\]
Uniform Substitution for dAL

\(\text{DC} \quad ([c \& q]p \leftrightarrow [c \& q \land r]p) \leftrightarrow [c \& q]r\)

\(\text{DI} \quad ([c \& q]p \leftrightarrow [?q]p) \leftrightarrow [c \& q](p)’\)

\(\text{DW} \quad [c \& q]p \leftrightarrow [c \& q](q \rightarrow p)\)

\(\text{DE} \quad [x' = f, c \& q]p \leftrightarrow [x' = f, c \& q][x' := f]p\)

\(\text{DAC} \quad ([\exists \bar{y}.(c \& q)]p \leftrightarrow [\exists \bar{y}.(c \& q \land r)]p) \leftrightarrow [\exists \bar{y}.(c \& q)]r\)

\(\text{DAI} \quad ([\exists \bar{y}.(c \& q)]p \leftrightarrow \forall \bar{y}. [?q]p) \leftrightarrow [\exists \bar{y}.(c \& q)](p)’\)

\(\text{DAW} \quad [\exists \bar{y}.(c \& q)]p \leftrightarrow [\exists \bar{y}.(c \& q)](q \rightarrow p)\)

\(\text{DAE} \quad [\exists \bar{y}.(x' = f, c \& q)]p \leftrightarrow \forall \bar{y}. [\exists \bar{y}.(x' = f, c \& q)] [x' := f]p\)
Uniform Substitution for dAL

DAC \quad (\{\exists \bar{y}.(c & q)\} \ p \leftrightarrow \{\exists \bar{y}.(c & q \land r)\} \ p) \leftarrow \{\exists \bar{y}.(c & q)\} \ r

DAI \quad (\{\exists \bar{y}.(c & q)\} \ p \leftrightarrow \forall \bar{y}. [?q \ p] \leftarrow \{\exists \bar{y}.(c & q)\} \ (p)'

DAW \quad \{\exists \bar{y}.(c & q)\} \ p \leftrightarrow \{\exists \bar{y}.(c & q)\} \ (q \rightarrow p)

DAE \quad \{\exists \bar{y}.(x' = f, c & q)\} \ p \leftrightarrow \forall \bar{y}. \{\exists \bar{y}.(x' = f, c & q)\} \ [x' := f] \ p
Uniform Substitution for dAL

DAC \quad ([\{\exists y. (c \land q)\}] \ p \leftrightarrow [\{\exists y. (c \land q \land r)\}] \ p) \leftarrow [\{\exists y. (c \land q)\}] \ r

DAI \quad ([\{\exists y. (c \land q)\}] \ p \leftrightarrow \forall y. [?q \ p] \leftarrow [\{\exists y. (c \land q)\}] (p)'

DAW \quad [\{\exists y. (c \land q)\}] \ p \leftrightarrow [\{\exists y. (c \land q)\}] (q \rightarrow p)

DAE \quad [\{\exists y. (x' = f, c \land q)\}] \ p \leftrightarrow \forall y. [\{\exists y. (x' = f, c \land q)\}] [x' := f] \ p
Uniform Substitution for dAL

DAC \quad ([\{\exists \bar{y}.(c \land q)\}] p \leftrightarrow [\{\exists \bar{y}.(c \land q \land r)\}] p) \leftrightarrow [\{\exists \bar{y}.(c \land q)\}] r

DAI \quad ([\{\exists \bar{y}.(c \land q)\}] p \leftrightarrow \forall \bar{y}. [?q] p) \leftrightarrow [\{\exists \bar{y}.(c \land q)\}] (p)'

DAW \quad [\{\exists \bar{y}.(c \land q)\}] p \leftrightarrow [\{\exists \bar{y}.(c \land q)\}] (q \rightarrow p)

DAE \quad [\{\exists \bar{y}.(x' = f, c \land q)\}] p \leftrightarrow \forall \bar{y}. [\{\exists \bar{y}.(x' = f, c \land q)\}] [x' := f] p

DAE too weak for technical reasons, DAW actually unnecessary!
Uniform Substitution for $\mathcal{DAL}$

DAC  $([\exists \bar{y}. (c \& q)] \rightarrow [\exists \bar{y}. (c \& q \land r)] p) \leftarrow [\exists \bar{y}. (c \& q)] r$

DAI $([\exists \bar{y}. (c \& q)] \rightarrow \forall \bar{y}. ?q p) \leftarrow [\exists \bar{y}. (c \& q)] (p)'$

DAS $[\exists \bar{y}. (c \& q)] \rightarrow \forall \bar{y}. [\exists \bar{y}. (c \& q)] [\{c \& q\}] p$

What we actually need is a 'differential algebraic stutter' axiom (DAS)!
Uniform Substitution for \(d\mathcal{AL}\)

DAC \( (\{\exists \bar{y}.(c \& q)\}\) p ⇔ \(\{\exists \bar{y}.(c \& q \land r)\}\) p) ⇔ \(\{\exists \bar{y}.(c \& q)\}\) r

DAI \( (\{\exists \bar{y}.(c \& q)\}\) p ⇔ \(\forall \bar{y}. [?q] p) ⇔ \(\{\exists \bar{y}.(c \& q)\}\) (p)′

DAS \( \{\exists \bar{y}.(c \& q)\}\) p ⇔ \(\forall \bar{y}. \{\exists \bar{y}.(c \& q)\}\) \(\{c \& q\}\) p

Theorem (Soundness of Uniform Substitution Calculus for \(d\mathcal{AL}\))

The above substitution axioms for \(d\mathcal{AL}\) are sound.
Uniform Substitution for dAL

DAC \quad ([\exists \bar{y}.(c \land q)] p \leftrightarrow [\exists \bar{y}.(c \land q \land r)] p) \leftrightarrow [\exists \bar{y}.(c \land q)] r

DAI \quad ([\exists \bar{y}.(c \land q)] p \leftrightarrow \forall \bar{y}. [?q] p) \leftrightarrow [\exists \bar{y}.(c \land q)] (p)'

DAS \quad ([\exists \bar{y}.(c \land q)] p \leftrightarrow \forall \bar{y}. [\exists \bar{y}.(c \land q)] [c \land q] p

Theorem (Soundness of Uniform Substitution Calculus for dAL)

The above substitution axioms for dAL are sound.

We have created a sound and 'minimal' uniform substitution calculus for dAL that we can implement into KeYmaeraX!
Implementation Details

- Modified the parser and core data structures in KeYmaeraX to support dAL.
- We support singly-quantified differential systems (due to lack of vectorial support and significant compatibility changes required).
- Modified to unification, uniform substitution and other necessary algorithms supported by KeYmaeraX.
- Uniform Substitution axioms added to trusted axiom base.
- Derived axioms and derived tactics proven from trusted axioms.
- Tested proving examples using derived axioms and tactics.
Implementation Details

- Modified the parser and core data structures in KeYmaeraX to support d\(\mathcal{L}\)
- We support singly-quantified differential systems (due to lack of vectorial support and significant compatibility changes required)
- Modified to unification, uniform substitution and other necessary algorithms supported by KeYmaeraX
- Uniform Substitution axioms added to trusted axiom base
- Derived axioms and derived tactics proven from trusted axioms
- Tested proving examples using derived axioms and tactics

A successful extension that minimally extends the trusted core!
Derived Tactics

\[
\frac{Q(x, \bar{y}) \vdash P(x)}{\Gamma \vdash [\{\exists \bar{y}. (x' = f(x, \bar{y}) \land Q(x, \bar{y}))\}] P(x), \Delta} \quad \text{dAW}
\]

\[
\frac{Q(x, \bar{y}) \vdash [x' := f(x, \bar{y})] (P(x))'}{P(x) \vdash [\{\exists \bar{y}. (x' = f(x, \bar{y}) \land Q(x, \bar{y}))\}] P(x)} \quad \text{dAI}
\]

\[
\Gamma \vdash [\{\exists \bar{y}. (c \land Q \land R)\}] P, \Delta \quad \Gamma \vdash [\{\exists \bar{y}. (c \land Q)\}] R, \Delta
\]

\[
\Gamma \vdash [\{\exists \bar{y}. (c \land Q)\}] P, \Delta
\]

\[
\Gamma \vdash [\{\exists \bar{y}. (c \land Q)\}] R, \Delta
\]

\[
\text{dAC}
\]
Example (Perturbed Circular Motion)

\[
\begin{align*}
* & \quad x \cdot e \leq 0 \vdash 2x(-y + e) + 2yx \leq 0 \\
\vdash x \cdot e \leq 0 & \vdash [x' := -y + e] [y' := x] 2xx' + 2yy' \leq 0 \\
\vdash x \cdot e \leq 0 & \vdash [x' := -y + e; y' := x] (x^2 + y^2 \leq 1)' \\
\vdash x^2 + y^2 = 1 & \vdash \{\exists e.(x' = -y + e, y' = x & x \cdot e \leq 0)\} x^2 + y^2 \leq 1 \\
\vdash x^2 + y^2 = 1 & \rightarrow \{\exists e.(x' = -y + e, y' = x & x \cdot e \leq 0)\} x^2 + y^2 \leq 1
\end{align*}
\]

\[
\begin{align*}
Q(x, \bar{y}) & \vdash [x' := f(x, \bar{y})] (P(x))' \\
P(x) & \vdash \{\exists \bar{y}.(x' = f(x, \bar{y}) & Q(x, \bar{y}))\} P(x)
\end{align*}
\]
Conclusion and Future Work

- Constructed a uniform substitution calculus for dAL
- Implemented it in KeYmaeraX for a single existentially quantified variable
- Constructed derived axioms and rules for actual use
- Future extensions:
  - Support for multiple quantifiers
  - WebUI support
  - 'Desugared' syntax
  - Support for more derived rules
  - Support for hybrid games with differential-algebraic components
Conclusion and Future Work

- Constructed a uniform substitution calculus for dAL
- Implemented it in KeYmaeraX for a single existentially quantified variable
- Constructed derived axioms and rules for actual use
- Future extensions:
  - Support for multiple quantifiers
  - WebUI support
  - 'Desugared' syntax
  - Support for more derived rules
  - Support for hybrid games with differential-algebraic components
Conclusion and Future Work

- Constructed a uniform substitution calculus for $d\mathcal{L}$
- Implemented it in KeYmaeraX for a single existentially quantified variable
- Constructed derived axioms and rules for actual use
- Future extensions:
  - Support for multiple quantifiers
  - WebUI support
  - 'Desugared' syntax
  - Support for more derived rules
  - Support for hybrid games with differential-algebraic components
Conclusion and Future Work

- Constructed a uniform substitution calculus for $\text{dAL}$
- Implemented it in KeYmaeraX for a single existentially quantified variable
- Constructed derived axioms and rules for actual use
- Future extensions:
  - Support for multiple quantifiers
  - WebUI support
  - ’Desugared’ syntax
  - Support for more derived rules
  - Support for hybrid games with differential-algebraic components
Conclusion and Future Work

- Constructed a uniform substitution calculus for dAL
- Implemented it in KeYmaeraX for a single existentially quantified variable
- Constructed derived axioms and rules for actual use
- Future extensions:
  - Support for multiple quantifiers
  - WebUI support
  - 'Desugared' syntax
  - Support for more derived rules
  - Support for hybrid games with differential-algebraic components
Questions?