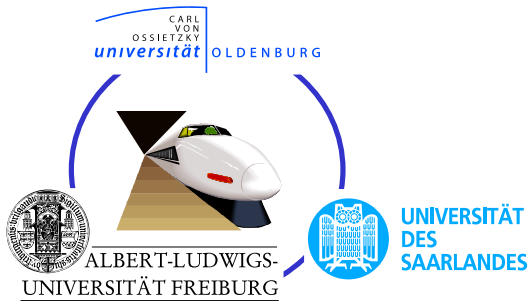
Combining Deduction and Algebraic Constraints for Hybrid System Analysis

André Platzer

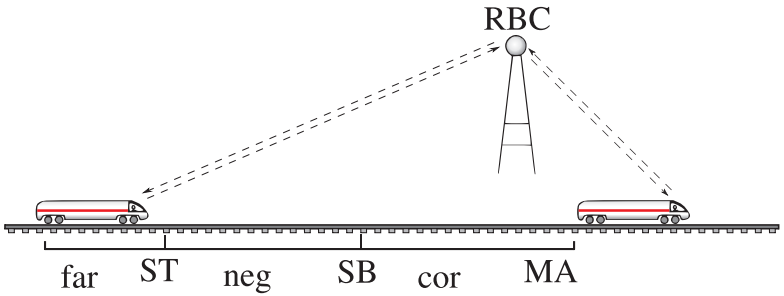
University of Oldenburg, Department of Computing Science, Germany

Verify'07 at CADE'07

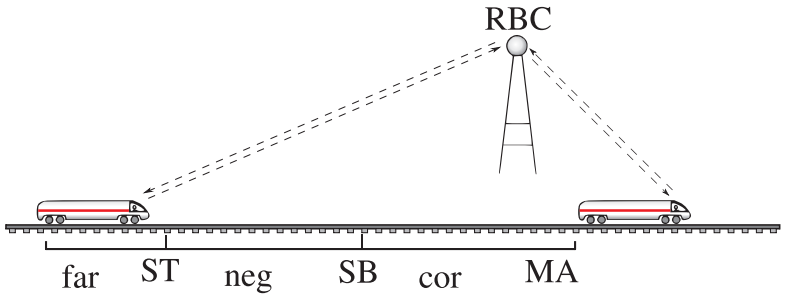


Outline

Deductively Verifying Hybrid Systems

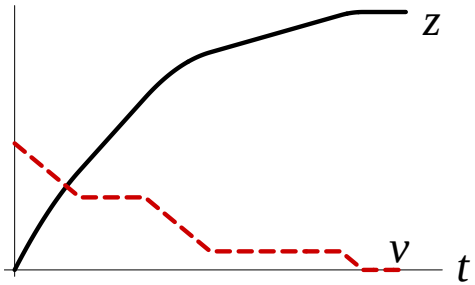


Deductively Verifying Hybrid Systems

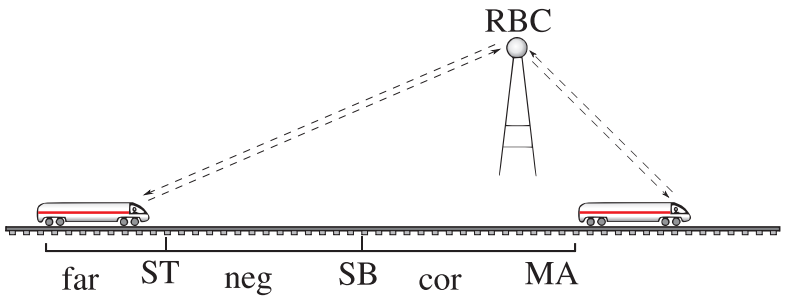


Hybrid Systems

continuous evolution along differential equations + discrete change



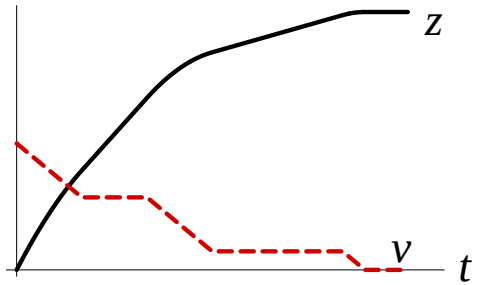
Deductively Verifying Hybrid Systems



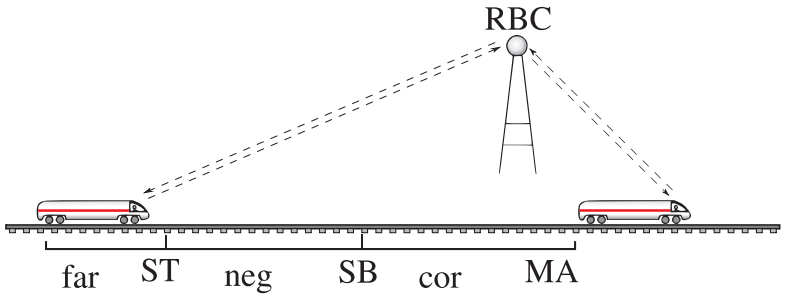
Hybrid Systems

continuous evolution along differential equations + discrete change

- Standard paradigm: model checking
- HyTech, CheckMate, PHAVer, . . . find bugs
- Verification is difficult, because of
  - numerical issues, numerical approximation
  - termination of abstraction refinement
  - unbounded regions
  - Parameter $SB = 10000$?



Deductively Verifying Hybrid Systems



Hybrid Systems

continuous evolution along differential equations + discrete change

differential dynamic logic

$$\mathrm{d}\mathcal{L} = \mathrm{DL} + \mathrm{HP}$$

Outline

Outline

Differential Logic d$\mathcal{L}$: Syntax

Definition (Hybrid program α)

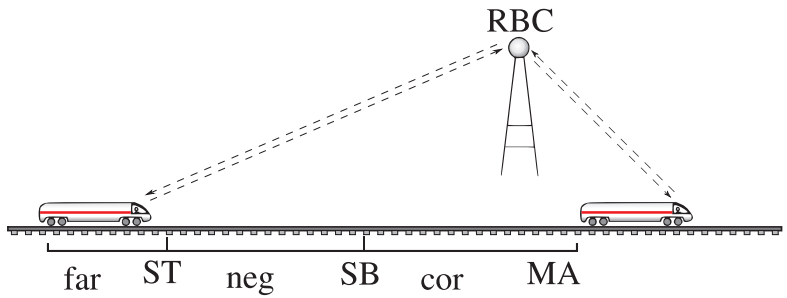
$x' = f(x)$	(continuous evolution)
$x := \theta$	(discrete jump)
$?\chi$	(conditional execution)
$\alpha; \beta$	(seq. composition)
$\alpha \cup \beta$	(nondet. choice)
α^*	(nondet. repetition)

Differential Logic dℒ: Syntax

Definition (Hybrid program α)

$x' = f(x)$	(continuous evolution)
$x := \theta$	(discrete jump)
$?\chi$	(conditional execution)
$\alpha; \beta$	(seq. composition)
$\alpha \cup \beta$	(nondet. choice)
α^*	(nondet. repetition)

$$ETCS \equiv (ctrl\,;drive)^*$$
$$ctrl \equiv (?MA - z \leq SB; a := -b)$$
$$\cup\,(?MA - z \geq SB; a := \ldots)$$
$$drive \equiv z'' = a$$



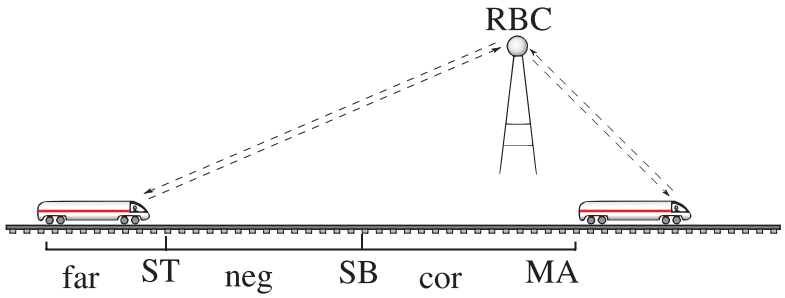
Differential Logic dℒ: Syntax

Definition (Formulas ϕ)

$\neg, \wedge, \vee, \rightarrow, \ \forall x\,, \exists x\,, \ \ =, \leq, \ +, \cdot$ ($\mathbb{R}$-first-order part)
$[\alpha]\phi, \quad \langle\alpha\rangle\phi$ (dynamic part)

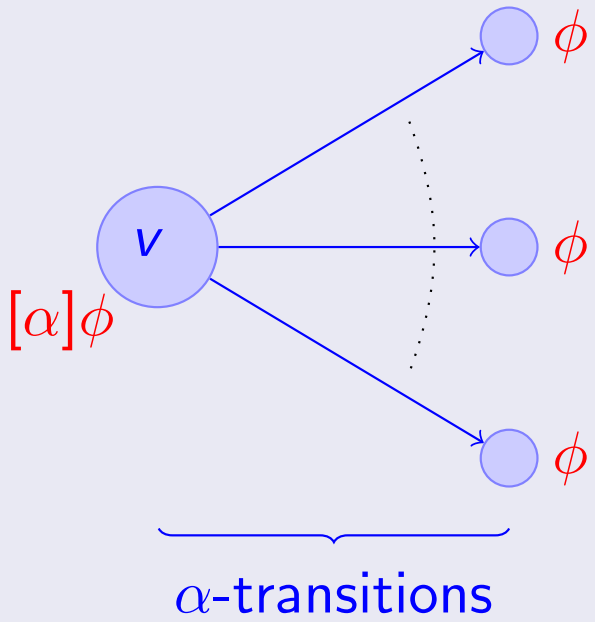
$$\psi \ \rightarrow \ [(\mathit{ctrl}\,;\mathit{drive})^*]\, z \leq \mathit{MA}$$

All trains respect *MA*
$\Rightarrow$ system safe



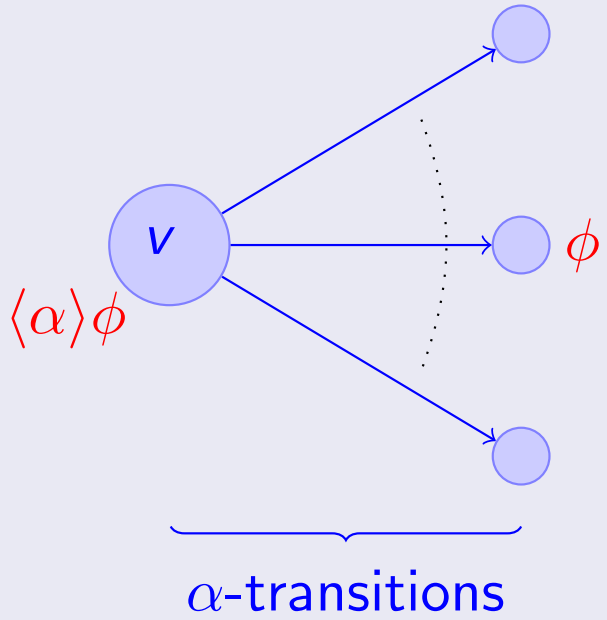
Differential Logic dℒ: Semantics

Definition (Formulas ϕ)



Differential Logic dℒ: Semantics

Definition (Formulas ϕ)



11 dynamic rules

(D1) $\dfrac{\phi \wedge \psi}{\langle ?\phi \rangle \psi}$

(D2) $\dfrac{\phi \rightarrow \psi}{[?\phi]\psi}$

(D3) $\dfrac{\langle \alpha \rangle \phi \vee \langle \beta \rangle \phi}{\langle \alpha \cup \beta \rangle \phi}$

(D4) $\dfrac{[\alpha]\phi \wedge [\beta]\phi}{[\alpha \cup \beta]\phi}$

(D5) $\dfrac{\phi \vee \langle \alpha ; \alpha^* \rangle \phi}{\langle \alpha^* \rangle \phi}$

(D6) $\dfrac{\phi \wedge [\alpha ; \alpha^*]\phi}{[\alpha^*]\phi}$

(D7) $\dfrac{\langle \alpha \rangle \langle \beta \rangle \phi}{\langle \alpha ; \beta \rangle \phi}$

(D8) $\dfrac{\phi_x^\theta}{\langle x := \theta \rangle \phi}$

(D9) $\dfrac{\exists t \geq 0 \, (\bar{\chi} \wedge \langle x := y_x$
$\langle x' = \theta \,\&\, \chi \rangle$

(D10) $\dfrac{\forall t \geq 0 \, (\bar{\chi} \rightarrow [x := y$
$[x' = \theta \,\&\, \chi]$

(D11) $\dfrac{\vdash p \qquad \vdash [\alpha^*](p \rightarrow [\alpha]p)}{\vdash [\alpha^*]p}$

Verification Calculus for d$\mathcal{L}$

Propositional/Quantifier Rules

9 propositional rules + 4 quantifier rules

(P1) $\dfrac{\vdash \phi}{\neg\phi \vdash}$ (P4) $\dfrac{\phi, \psi \vdash}{\phi \wedge \psi \vdash}$ (P7) $\dfrac{\phi \vdash \qquad \psi \vdash}{\phi \vee \psi \vdash}$

(P2) $\dfrac{\phi \vdash}{\vdash \neg\phi}$ (P5) $\dfrac{\vdash \phi \qquad \vdash \psi}{\vdash \phi \wedge \psi}$ (P8) $\dfrac{\vdash \phi, \psi}{\vdash \phi \vee \psi}$

(P3) $\dfrac{\phi \vdash \psi}{\vdash \phi \to \psi}$ (P6) $\dfrac{\vdash \phi \qquad \psi \vdash}{\phi \to \psi \vdash}$ (P9) $\dfrac{}{\phi \vdash \phi}$

(F1) $\dfrac{\mathrm{QE}(\exists x \bigwedge_i (\Gamma_i \vdash \Delta_i))}{\Gamma \vdash \Delta, \exists x\, \phi}$ (F3) $\dfrac{\mathrm{QE}(\forall x \bigwedge_i (\Gamma_i \vdash \Delta_i))}{\Gamma \vdash \Delta, \forall x\, \phi}$

(F2) $\dfrac{\mathrm{QE}(\forall x \bigwedge_i (\Gamma_i \vdash \Delta_i))}{\Gamma, \exists x\, \phi \vdash \Delta}$ (F4) $\dfrac{\mathrm{QE}(\exists x \bigwedge_i (\Gamma_i \vdash \Delta_i))}{\Gamma, \forall x\, \phi \vdash \Delta}$

Concise Theory! But End of the Story?

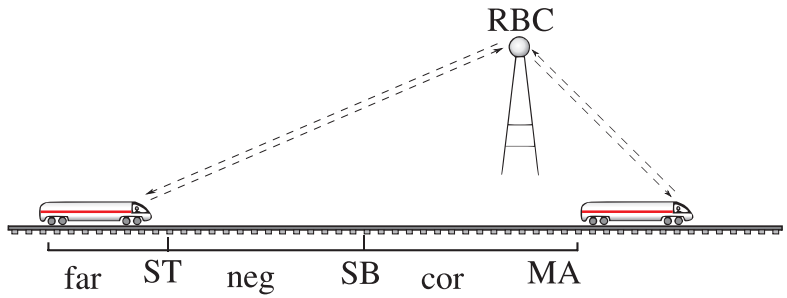
Outline

Analysing European Train Control System (ETCS)

$$\psi \rightarrow [(ctrl\,;drive)^*]\,z \leq MA$$

$$ctrl \equiv (?MA - z < SB; a := -b) \cup (?MA - z \geq SB; a := 0)$$

$$drive \equiv \tau := 0; z' = v, v' = a, \tau' = 1 \;\&\; v \geq 0 \wedge \tau \leq \varepsilon$$



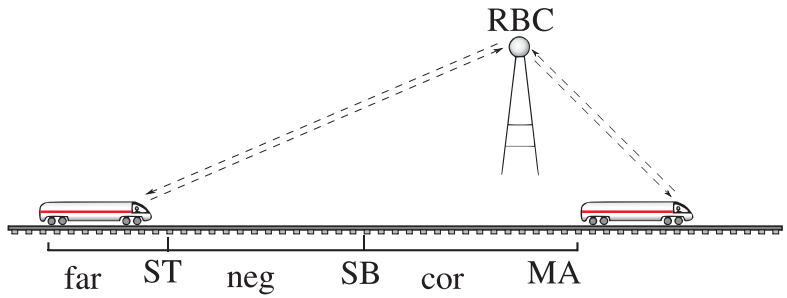
Analysing European Train Control System (ETCS)

provable automatically using invariant!

$$\psi \rightarrow [(ctrl\,;drive)^*]\, z \leq MA$$

$$ctrl \equiv (?MA - z < SB; a := -b) \cup (?MA - z \geq SB; a := 0)$$

$$drive \equiv \tau := 0; z' = v, v' = a, \tau' = 1 \;\&\; v \geq 0 \wedge \tau \leq \varepsilon$$



$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{p\vdash \forall t{\geq}0\,(\langle v := -bt + v\rangle v \geq 0 \rightarrow \langle z := -\frac{b}{2}t^2 + vt + z; v := -bt + v\rangle p)}\,*}{p\vdash [z' = v, v' = -b \,\&\, v \geq 0]p}}{p\vdash \langle a := -b\rangle [drive]p}}{p\vdash [?MA{-}z{\geq}SB; a := 0][drive]p}}{p\vdash [ctrl][drive]p}}{p\vdash [ctrl\,;drive]p}$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{p,\, MA{-}z{\geq}SB\vdash v^2 \leq 2b(MA - \varepsilon v - z)}\,\ldots}{p,\, MA{-}z{\geq}SB\vdash \forall t{\geq}0\,(\langle \tau := t\rangle \tau \leq \varepsilon \rightarrow \langle z := vt}}{p,\, MA{-}z{\geq}SB\vdash \langle \tau := 0\rangle \forall t{\geq}0\,(\langle \tau := t + \tau\rangle \tau \leq \varepsilon}}{p,\, MA{-}z{\geq}SB\vdash \langle \tau := 0\rangle [z' = v, v' = 0, \tau' = 1 \,\&}}{p,\, MA{-}z{\geq}SB\vdash \langle a := 0\rangle \langle \tau := 0\rangle [z' = v, v' = a, \tau}}{p,\, MA{-}z{\geq}SB\vdash \langle a := 0\rangle [drive]p}$$

Full European Train Control System (ETCS)

$$
\begin{array}{lcl}
\text{system} & : & \big(\text{poll};\ (\text{negot} \cup (\text{speedControl};\ \text{atp};\ \text{move}))\big)^* \\
\text{init} & : & \mathit{drive} := 0;\ \mathit{brake} := 1 \\
\text{poll} & : & \mathit{SB} := \frac{v^2-d^2}{2b} + \big(\frac{a_{max}}{b}+1\big)\big(\frac{a_{max}}{2}\varepsilon^2 + \varepsilon v\big);\ \mathit{ST} := * \\
\text{negot} & : & (?m - z > \mathit{ST}) \cup (?m - z \leq \mathit{ST};\ \text{rbc}) \\
\text{rbc} & : & (\mathit{vdes} := *;\ ?\mathit{vdes} > 0) \cup (\mathit{state} := \mathit{brake}) \\
 & & \cup \big(d_{old} := d;\ m_{old} := m;\ m := *;\ d := *; \\
 & & \quad ?d \geq 0 \wedge d_{old}^2 - d^2 \leq 2b(m - m_{old})\big) \\
\text{speedCtrl} & : & (?\mathit{state} = \mathit{brake};\ a := -b) \\
 & & \cup \Big(?\mathit{state} = \mathit{drive}; \\
 & & \quad \big((?v \leq v_{des};\ a := *;\ ? - b \leq a \leq a_{max}) \\
 & & \quad \cup (?v \geq v_{des};\ a := *;\ ?0 > a \geq -b)\big)\Big) \\
\text{atp} & : & (?m - z \leq \mathit{SB};\ a := -b) \cup (?m - z > \mathit{SB}) \\
\text{move} & : & t := 0;\ \{\dot{z} = v, \dot{v} = a, \dot{t} = 1, (v \geq 0 \wedge t \leq \varepsilon)\}
\end{array}
$$

Full European Train Control System (ETCS)

not provable automatically!

52 user interactions!

$$
\begin{array}{lcl}
\text{system} & : & \big(\text{poll};\ (\text{negot} \cup (\text{speedControl};\ \text{atp};\ \text{move}))\big)^* \\
\text{init} & : & \mathit{drive} := 0;\ \mathit{brake} := 1 \\
\text{poll} & : & \mathit{SB} := \frac{v^2-d^2}{2b} + \left(\frac{a_{max}}{b}+1\right)\left(\frac{a_{max}}{2}\varepsilon^2 + \varepsilon v\right);\ \mathit{ST} := * \\
\text{negot} & : & (?m - z > \mathit{ST}) \cup (?m - z \leq \mathit{ST};\ \text{rbc}) \\
\text{rbc} & : & (\mathit{vdes} := *;\ ?\mathit{vdes} > 0) \cup (\mathit{state} := \mathit{brake}) \\
 & & \cup \big(d_{old} := d;\ m_{old} := m;\ m := *;\ d := *; \\
 & & \quad ?d \geq 0 \wedge d_{old}^2 - d^2 \leq 2b(m - m_{old})\big) \\
\text{speedCtrl} & : & (?\mathit{state} = \mathit{brake};\ a := -b) \\
 & & \cup \Big(?\mathit{state} = \mathit{drive}; \\
 & & \quad \big((?v \leq v_{des};\ a := a_{max}) \\
 & & \quad \cup (?v \geq v_{des};\ a := -b)\big)\Big) \\
\text{atp} & : & (?m - z \leq \mathit{SB};\ a := -b) \cup (?m - z > \mathit{SB}) \\
\text{move} & : & t := 0;\ \{\dot{z} = v, \dot{v} = a, \dot{t} = 1, (v \geq 0 \wedge t \leq \varepsilon)\}
\end{array}
$$

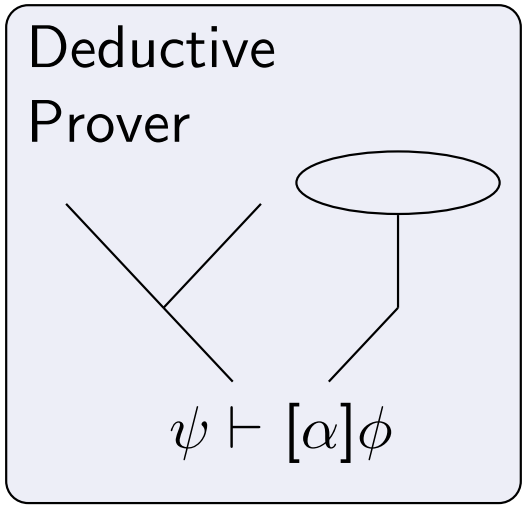
Full European Train Control System (ETCS)

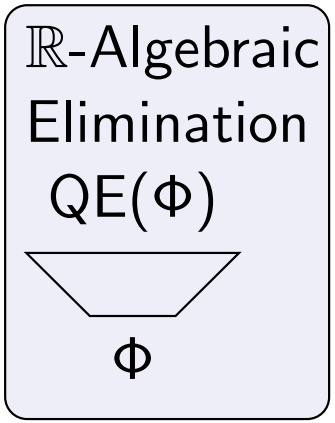
```
 state = 0,
 2 * b * (m - z) >= v ^ 2 - d ^ 2,
 v >= 0, d >= 0, v >= 0, ep >  0, b >  0, amax >  0, d >= 0
==>
    v <= vdes
 -> \forall R a_3;
      (   a_3 >= 0 & a_3 <= amax
       ->   (      m - z
              <= (amax / b + 1) * ep * v
               + (v ^ 2 - d ^ 2) / (2 * b)
               + (amax / b + 1) * amax * ep ^ 2 / 2
            -> \forall R t0;
                 (   t0 >= 0
                  -> \forall R ts0;  (0 <= ts0 & ts0 <= t0 -> -b * ts0 + v >= 0 & ts0 + 0 <= ep)
                  ->      2 * b * (m - 1 / 2 * (-b * t0 ^ 2 + 2 * t0 * v + 2 * z))
                       >= (-b * t0 + v) ^ 2
                        - d ^ 2
                     & -b * t0 + v >= 0
                     & d >= 0))
        & (      m - z
             >  (amax / b + 1) * ep * v
              + (v ^ 2 - d ^ 2) / (2 * b)
              + (amax / b + 1) * amax * ep ^ 2 / 2
           -> \forall R t2;
                (   t2 >= 0
                 -> \forall R ts2;  (0 <= ts2 & ts2 <= t2 -> a_3 * ts2 + v >= 0 & ts2 + 0 <= ep)
                 ->      2 * b * (m - 1 / 2 * (a_3 * t2 ^ 2 + 2 * t2 * v + 2 * z))
                      >= (a_3 * t2 + v) ^ 2
                       - d ^ 2
                    & a_3 * t2 + v >= 0
                    & d >= 0)))
```

Practice Seems Disturbingly Bad!

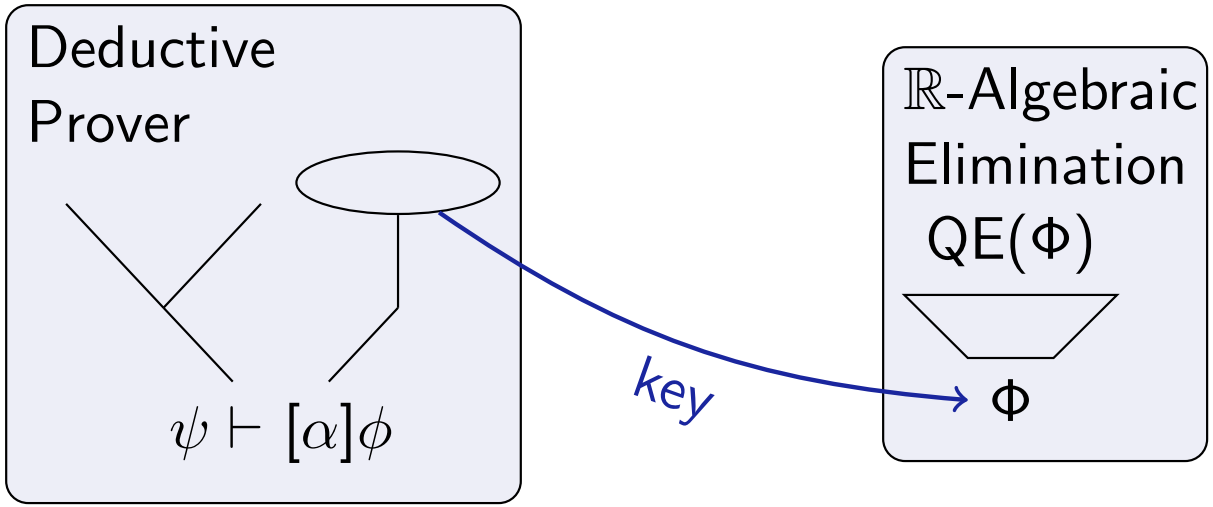
Outline

Modular Combination of Provers

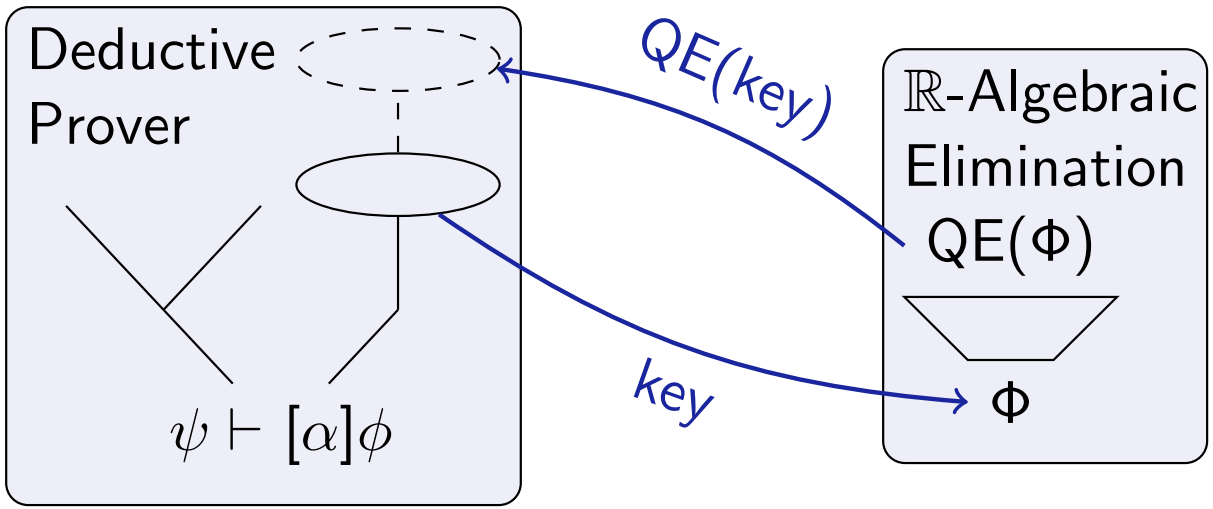




Modular Combination of Provers



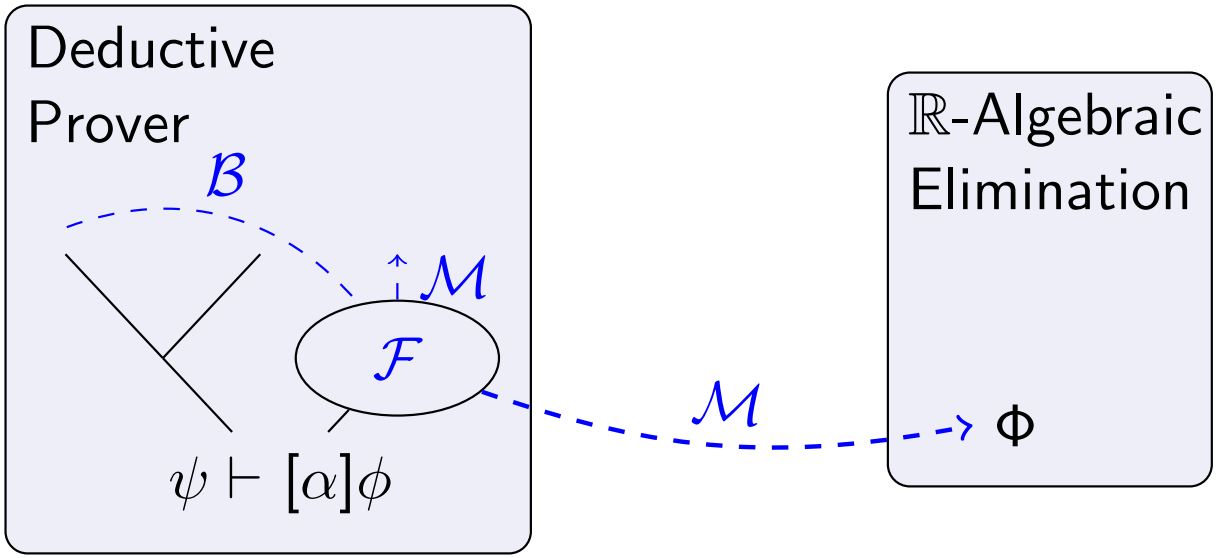
Modular Combination of Provers



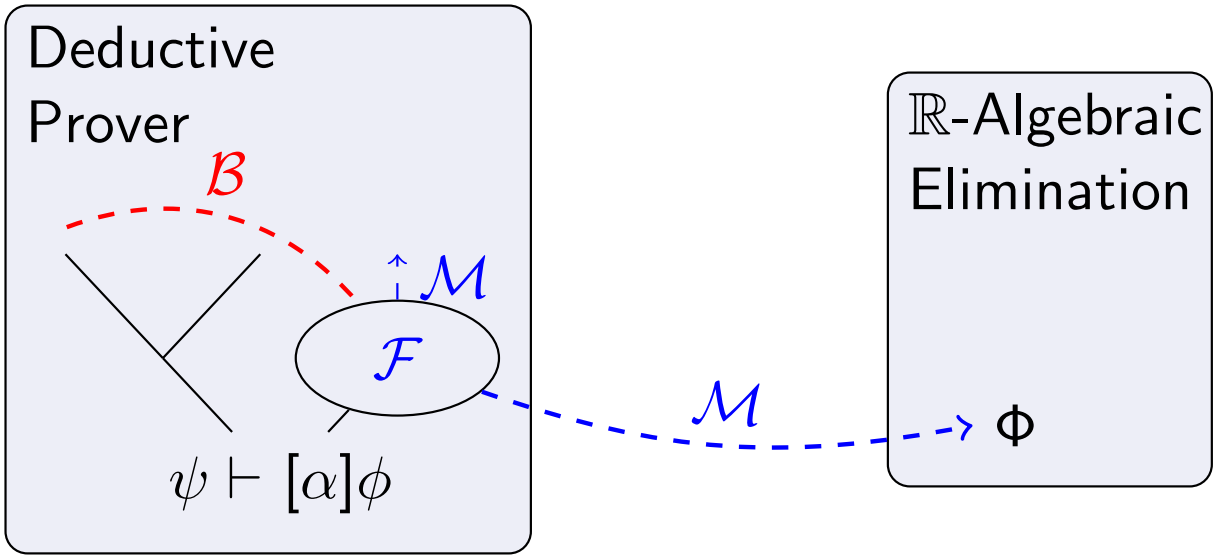
Tableaux Procedure for d$\mathcal{L}$

```
while tableaux T has open branches do
  B := selectBranch(T)          (* B–nondeterminism *)
  M := selectMode(B)            (* M–nondeterminism *)
  F := selectFormulas(B,M)      (* F–nondeterminism *)
  if M = foreground then
    B2 := result of applying a D–rule or P–rule to F i
    replace B by B2 in T
  else
    send key F to background decision procedure QE
    receive result R from QE
    apply a rule F3–F4 to T with QE–result R
  end if
end while
```

Tableaux Procedure for d$\mathcal{L}$: Nondeterminisms

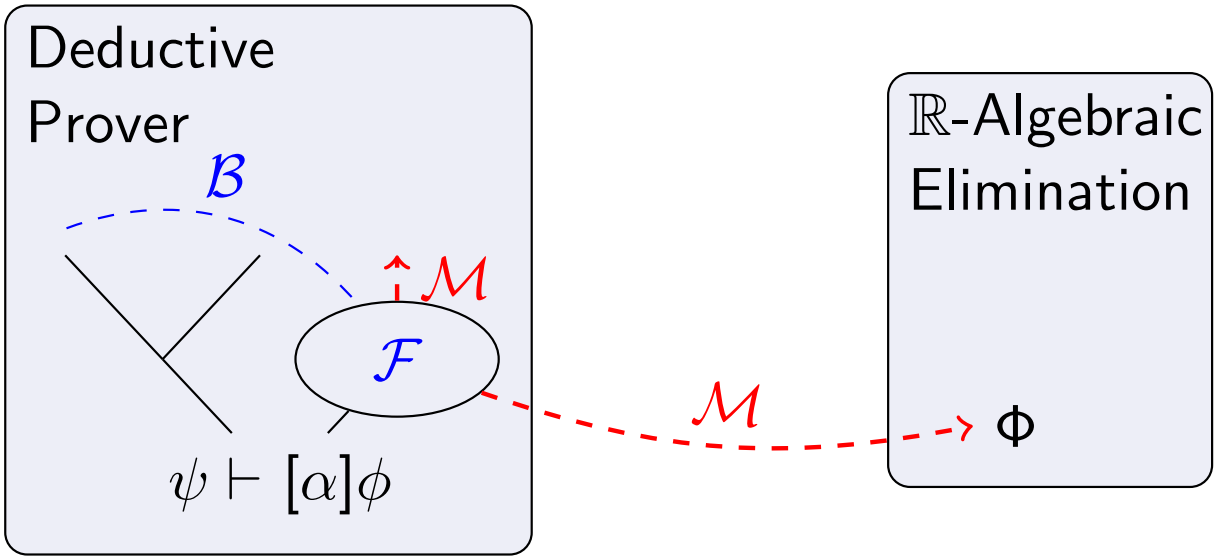


Tableaux Procedure for dℒ: Nondeterminisms



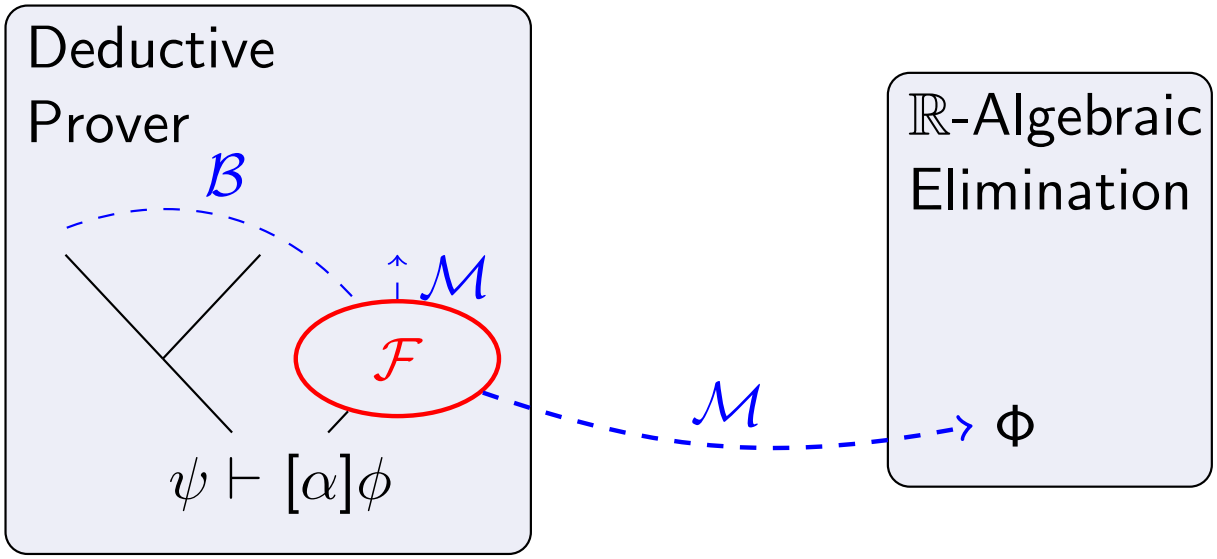
$\mathcal{B}$ branch selection

Tableaux Procedure for dℒ: Nondeterminisms



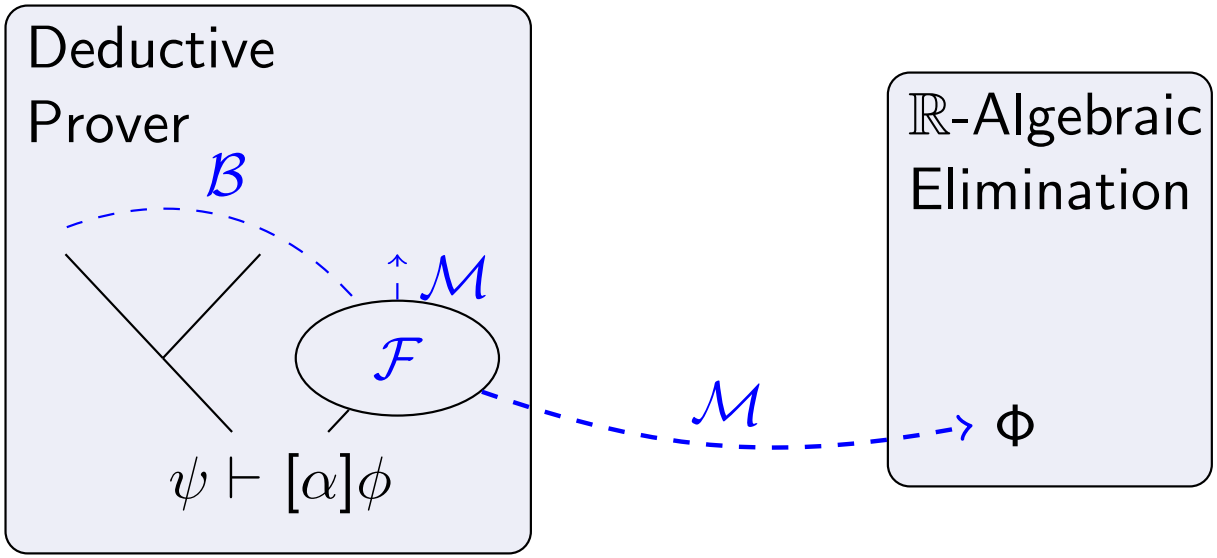
$\mathcal{M}$ mode selection

Tableaux Procedure for d$\mathcal{L}$: Nondeterminisms



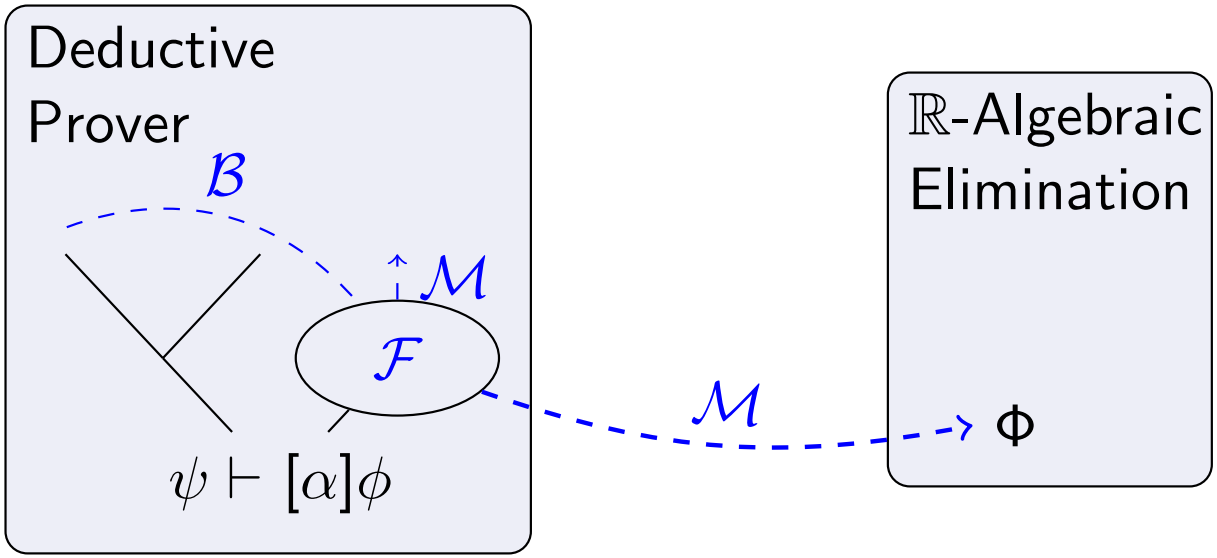
$\mathcal{F}$ formula selection

Tableaux Procedure for dℒ: Nondeterminisms



no nondeterminism from closing substitutions

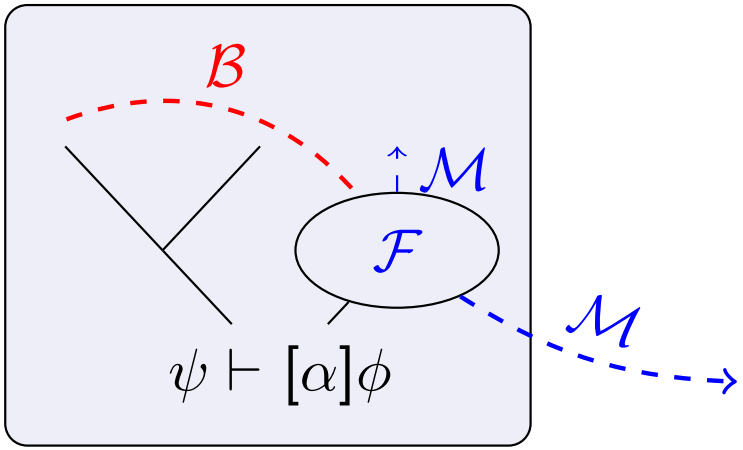
Tableaux Procedure for d$\mathcal{L}$: Nondeterminisms



uninterpreted FOL	interpreted d$\mathcal{L}$
uninterpreted symbols	interpreted symbols
close by substitution	close by arithmetic
close needs backtracking	equivalent QE elimination
closing is cheap	arithmetic is $O(2^{2^n})$

Nondeterminisms in Branch Selection

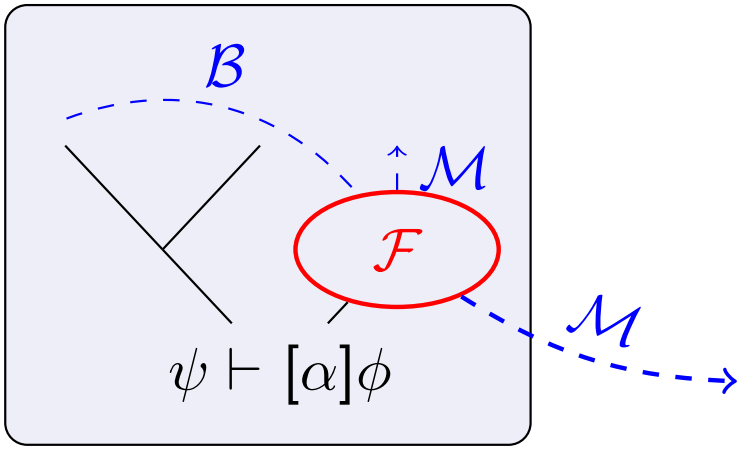
- harmless
  because no closing substitutions



Nondeterminisms in Formula Selection

- In principle: simple

$$\Phi \text{ closes} \quad \Rightarrow \quad \Psi \supseteq \Phi \text{ closes}$$



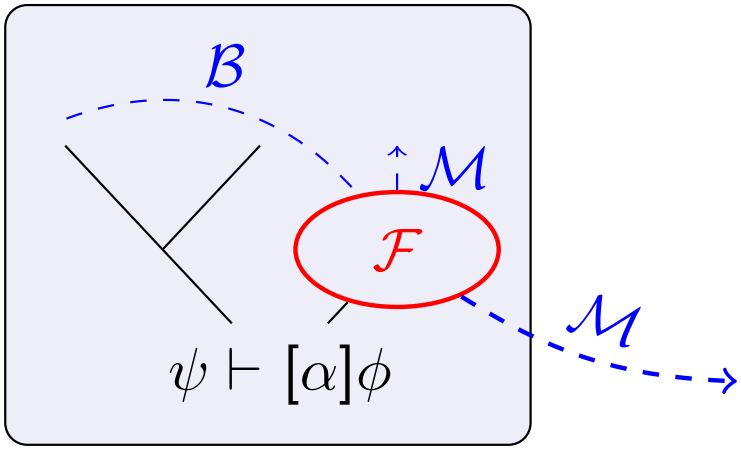
Nondeterminisms in Formula Selection

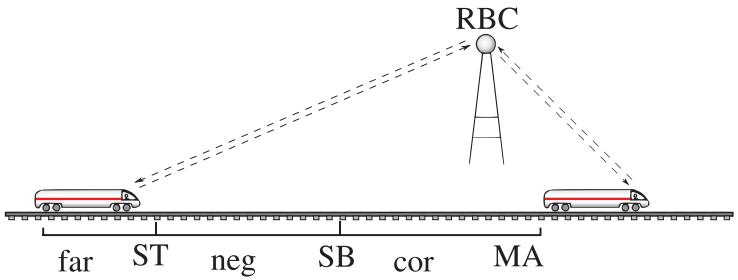
- In principle: simple

$$\Phi \text{ closes} \quad \Rightarrow \quad \Psi \supseteq \Phi \text{ closes}$$

- In practice: irrelevant formulas distract QE considerably
- Partially necessary ETCS constraint:

$$SB \geq \frac{v^2}{2b} + \left(\frac{a}{b} + 1\right)\left(\frac{a}{2}\varepsilon^2 + \varepsilon v\right)$$



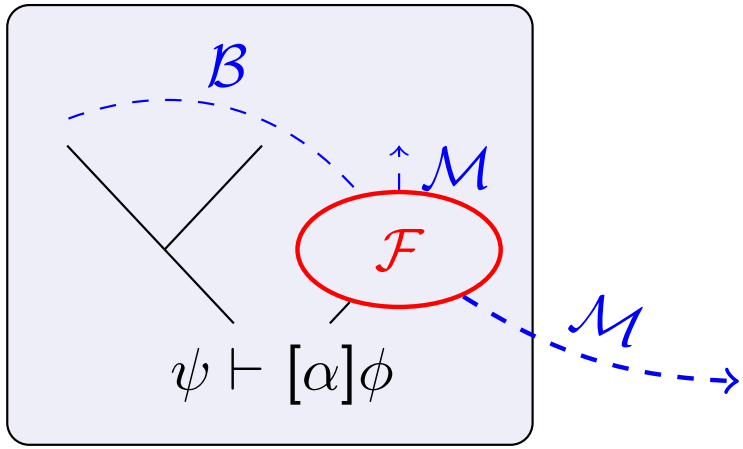


Nondeterminisms in Formula Selection

- In principle: simple

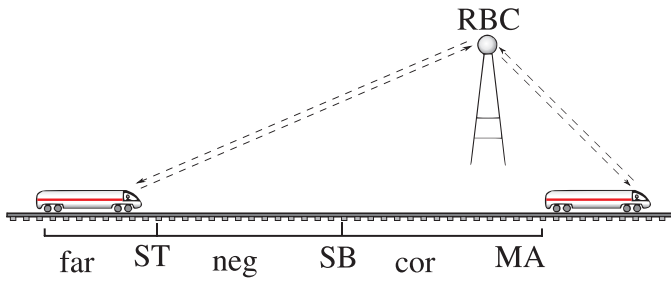
$$\Phi \text{ closes} \quad \Rightarrow \quad \Psi \supseteq \Phi \text{ closes}$$

- In practice: irrelevant formulas distract QE considerably



$\gg 24h$

$$t > 0,\ a + 1/tv \geq 0,\ \varepsilon \geq t,\ t \geq 0,$$
$$m - z \geq v^2/(2b) + (a/b + 1)(a/2\varepsilon^2 + \varepsilon v),$$
$$2b(m - z) \geq v^2,\ v \geq 0,$$
$$2b(m - z_0) \geq v_0^2,\ v_0 \geq 0,$$
$$\varepsilon \geq 0,\ b > 0,\ a \geq 0$$
$$\vdash (at + v)^2 \leq 2b(m - 1/2(at^2 + 2tv + 2z))$$

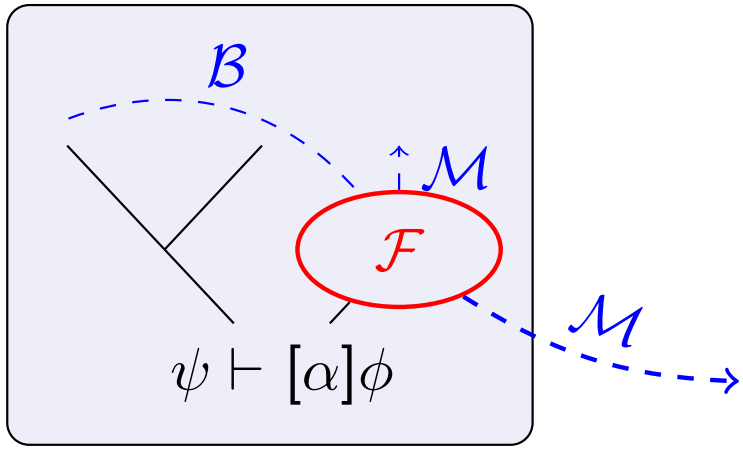


Nondeterminisms in Formula Selection

- In principle: simple

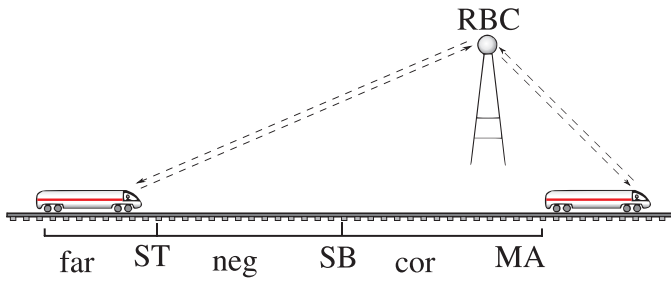
$$\Phi \text{ closes} \quad \Rightarrow \quad \Psi \supseteq \Phi \text{ closes}$$

- In practice: irrelevant formulas distract QE considerably



$\gg 24h$

$$t > 0,\ a + 1/tv \geq 0,\ \varepsilon \geq t,\ t \geq 0,$$
$$m - z \geq v^2/(2b) + (a/b + 1)(a/2\varepsilon^2 + \varepsilon v),$$
$$2b(m - z) \geq v^2,\ v \geq 0,$$
$$2b(m - z_0) \geq v_0^2,\ v_0 \geq 0, \qquad \text{(initial state)}$$
$$\varepsilon \geq 0,\ b > 0,\ a \geq 0$$
$$\vdash (at + v)^2 \leq 2b(m - 1/2(at^2 + 2tv + 2z))$$

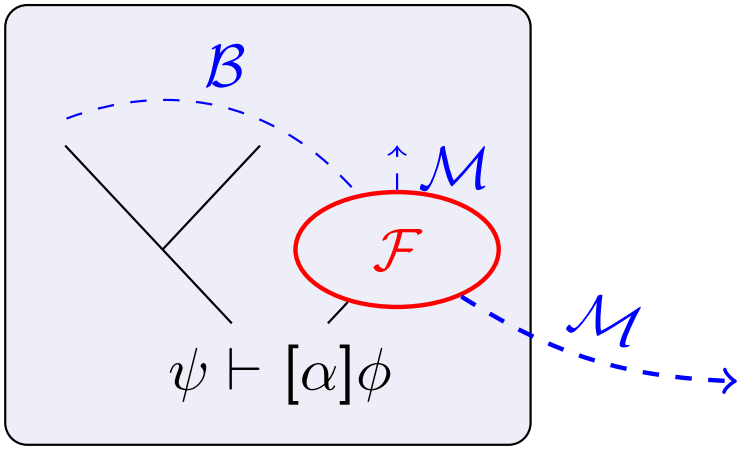


Nondeterminisms in Formula Selection

- In principle: simple

$$\Phi \text{ closes} \quad \Rightarrow \quad \Psi \supseteq \Phi \text{ closes}$$

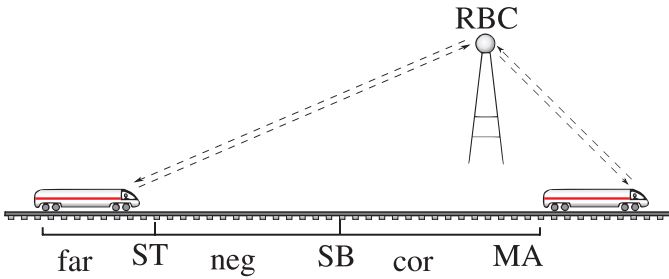
- In practice: irrelevant formulas distract QE considerably



$\ll 1s$

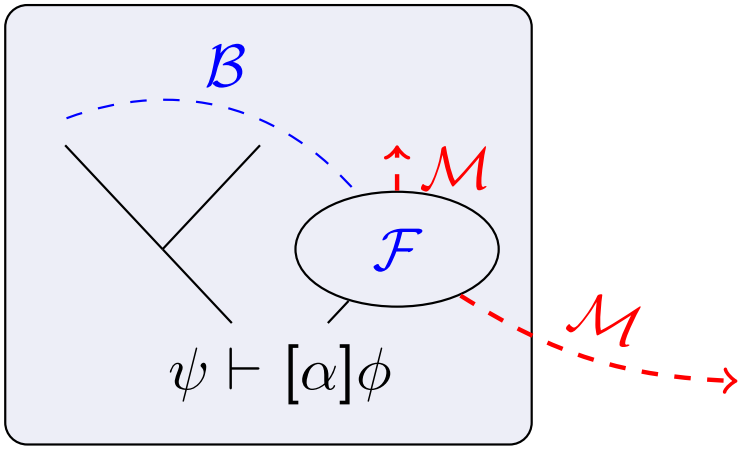
$$t > 0,\ a + 1/tv \geq 0,\ \varepsilon \geq t,\ t \geq 0,$$
$$m - z \geq v^2/(2b) + (a/b + 1)(a/2\varepsilon^2 + \varepsilon v),$$
$$2b(m - z) \geq v^2,\ v \geq 0,$$

$$\varepsilon \geq 0,\ b > 0,\ a \geq 0$$
$$\vdash (at + v)^2 \leq 2b(m - 1/2(at^2 + 2tv + 2z))$$



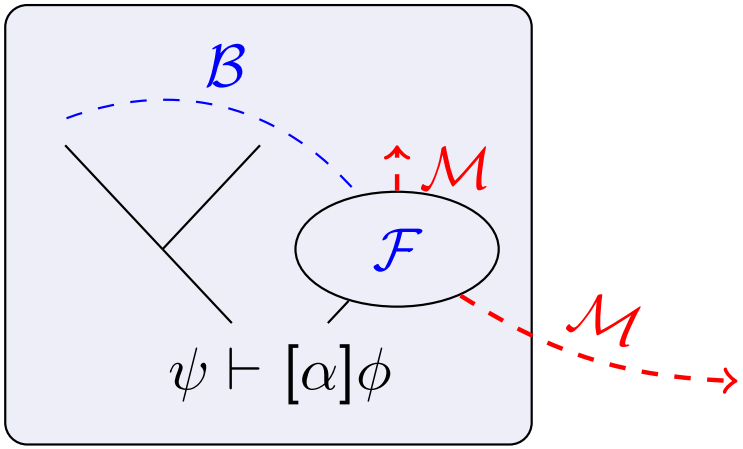
Nondeterminisms in Mode Selection

- In principle: only background closure, anything could close



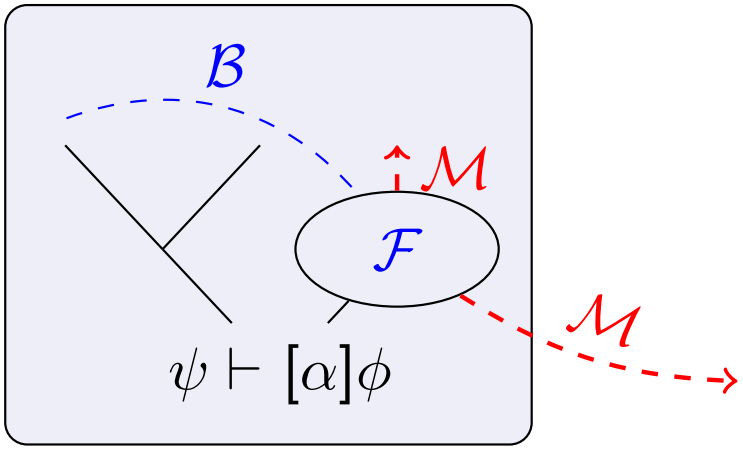
Nondeterminisms in Mode Selection

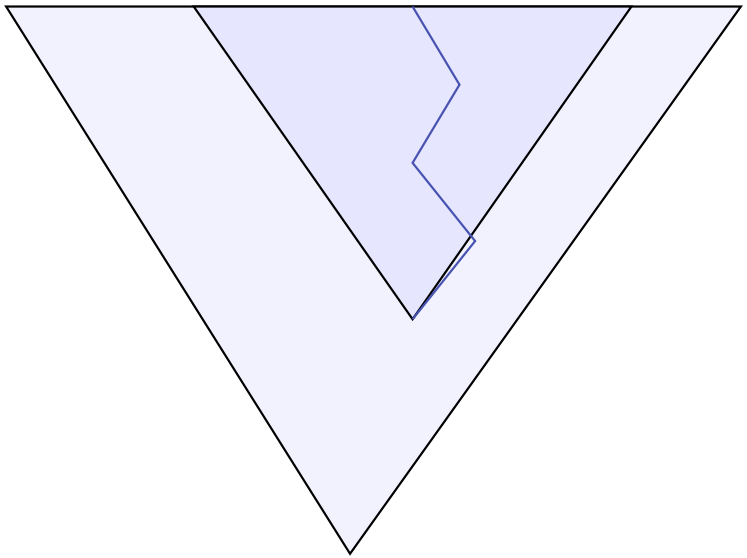
- In principle: only background closure, anything could close
- In practice: some QE "never" terminate



Nondeterminisms in Mode Selection

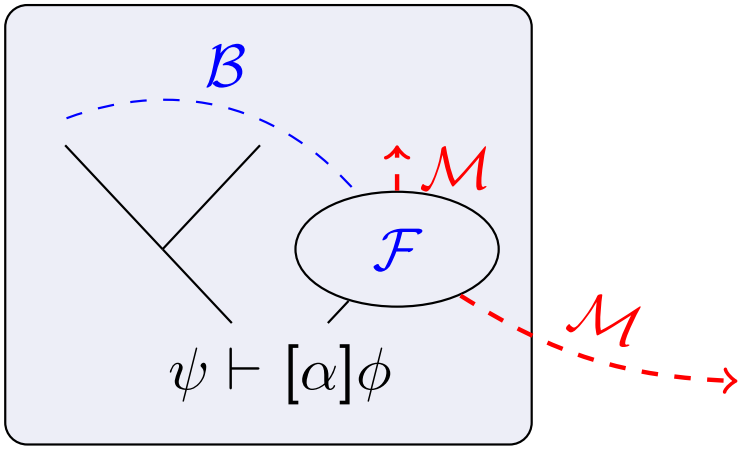
- In principle: only background closure, anything could close
- In practice: some QE "never" terminate

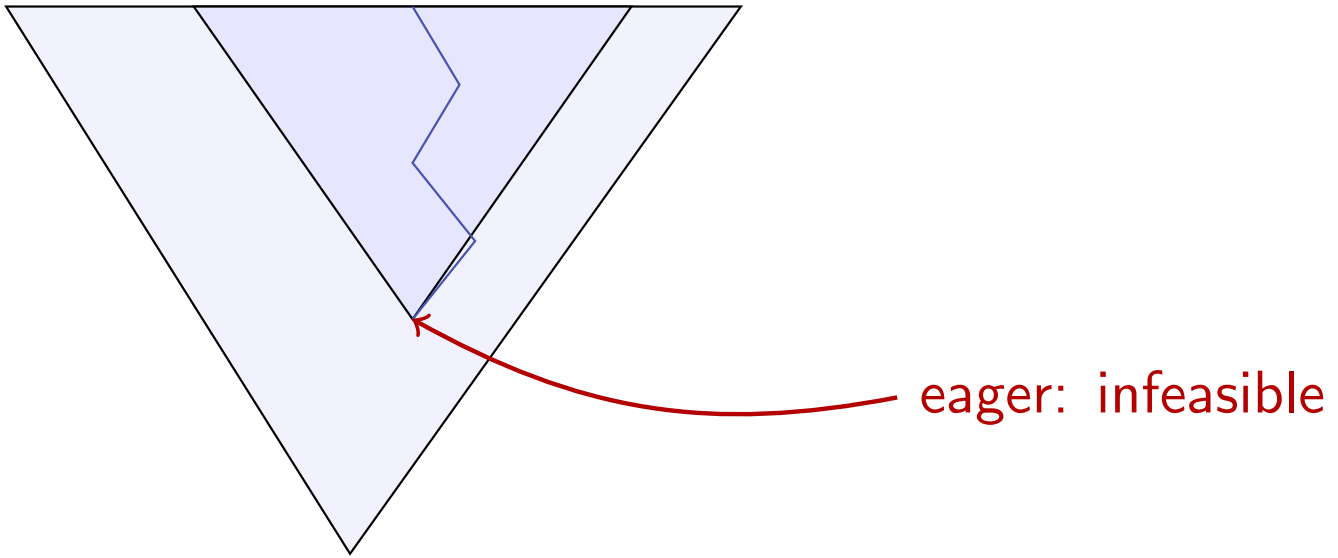




Nondeterminisms in Mode Selection

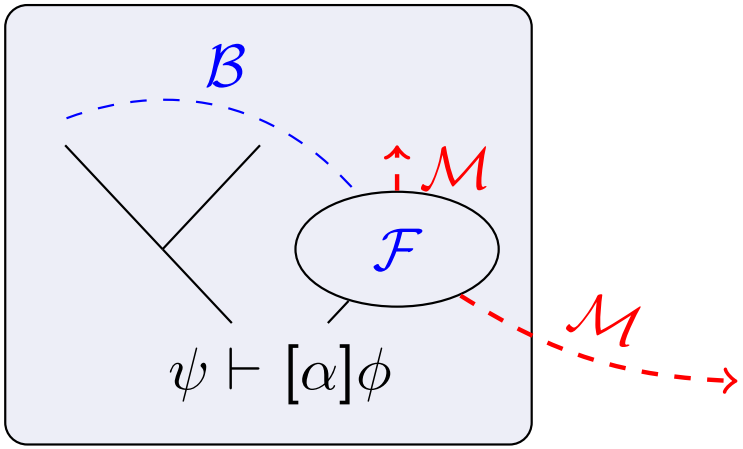
- In principle: only background closure, anything could close
- In practice: some QE "never" terminate

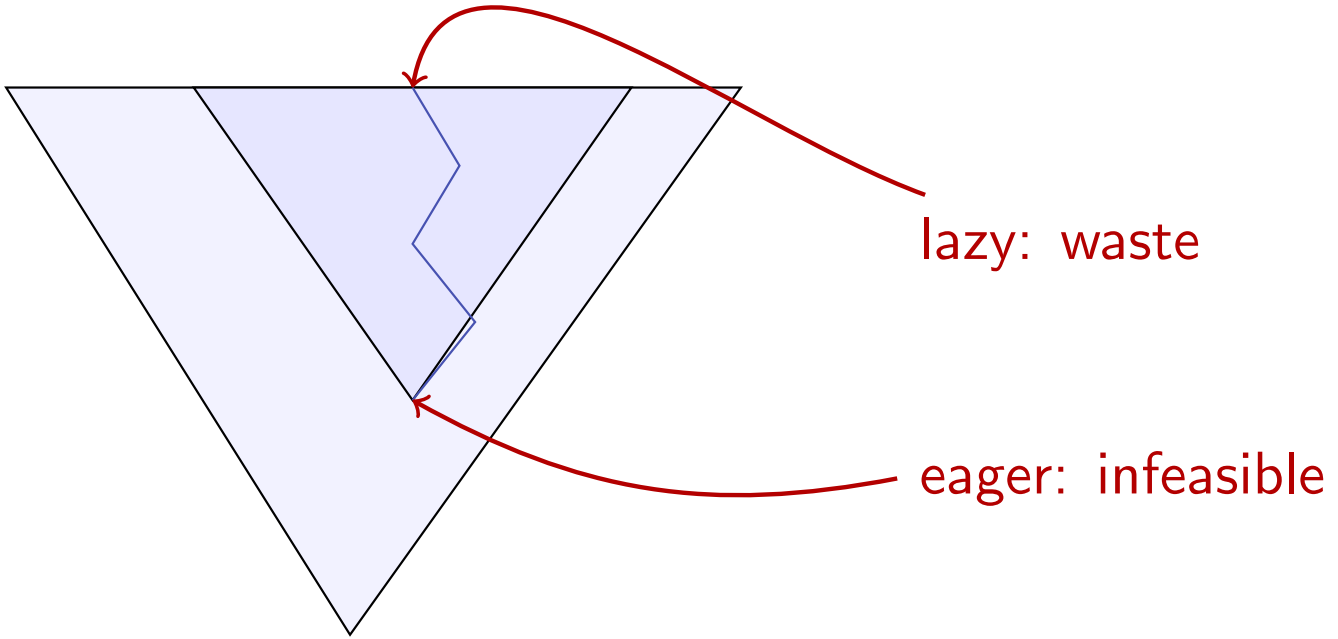




Nondeterminisms in Mode Selection

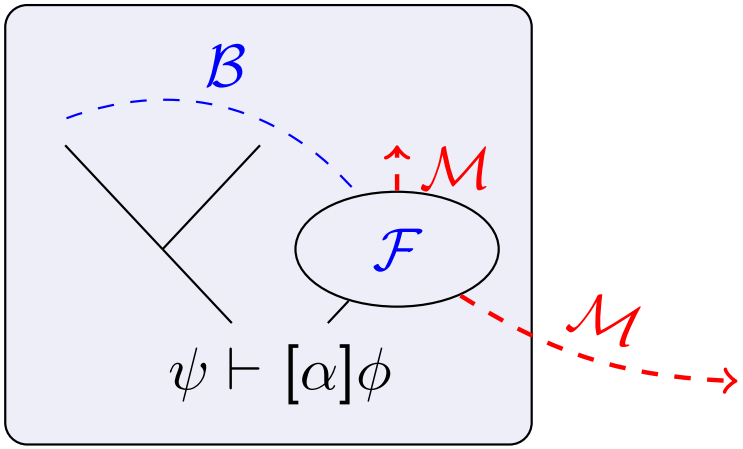
- In principle: only background closure, anything could close
- In practice: some QE "never" terminate

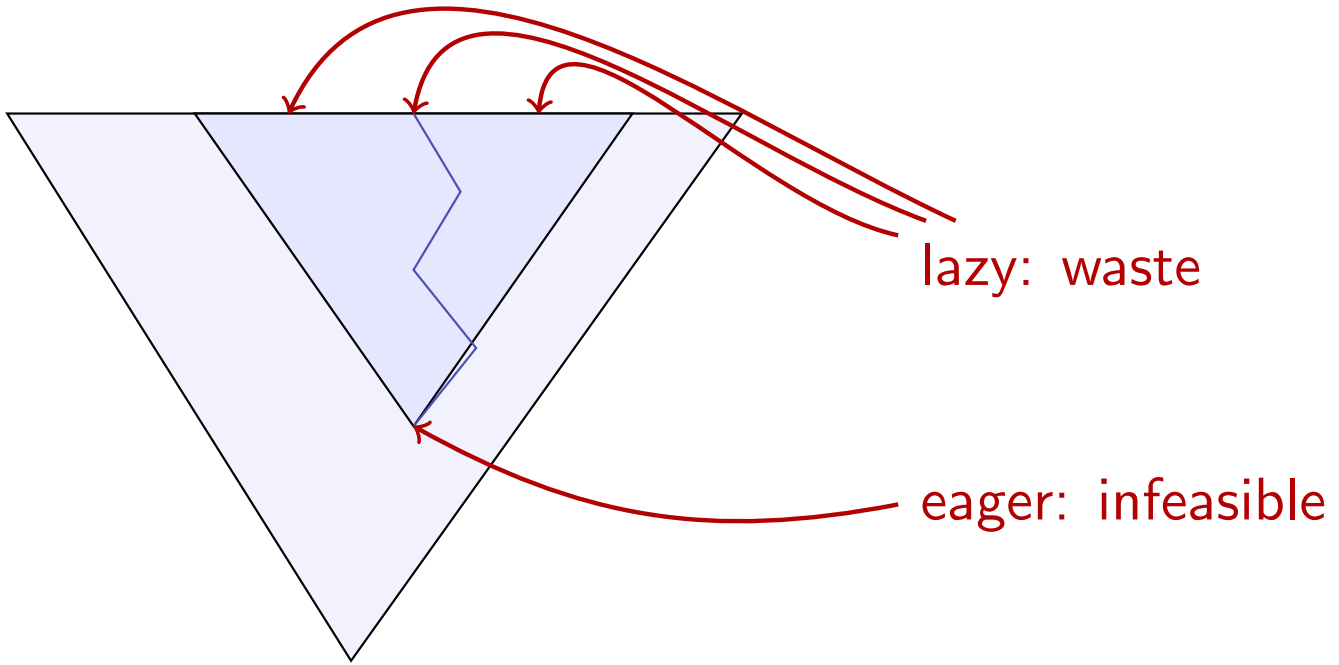




Nondeterminisms in Mode Selection

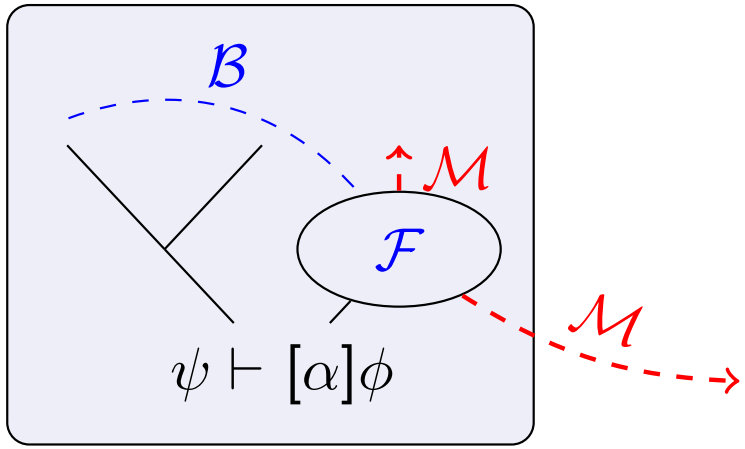
- In principle: only background closure, anything could close
- In practice: some QE "never" terminate





Nondeterminisms in Mode Selection

- In principle: only background closure, anything could close
- In practice: some QE "never" terminate
- Syntactic representational redundancy

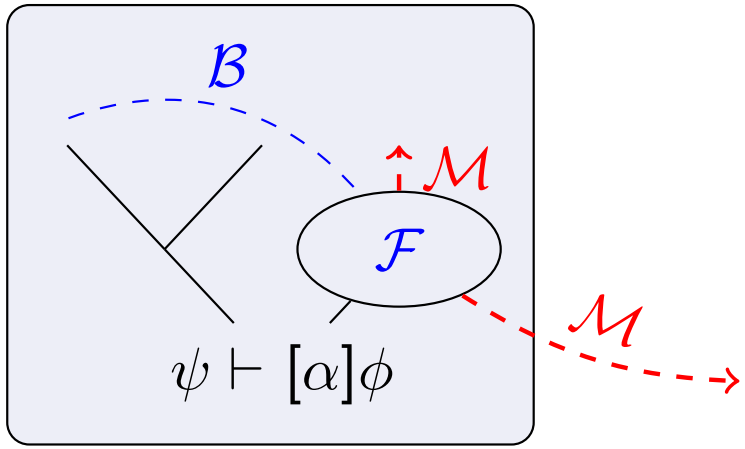


$$\frac{\psi \vdash v^2 \leq 2b(m-z) \qquad \psi \vdash (z \geq 0 \rightarrow v \leq 0)}{\psi \vdash v^2 \leq 2b(m-z) \wedge (z \geq 0 \rightarrow v \leq 0)}$$

redundant duplication or case distinction improvement?

Nondeterminisms in Mode Selection

- In principle: only background closure, anything could close
- In practice: some QE "never" terminate
- Syntactic representational redundancy
- Semantic representational redundancy



$$\frac{\vdash b \geq v^2/(2m-2z) \vee m \leq z}{z < m \quad \vdash v^2 \leq 2b(m-z)}$$

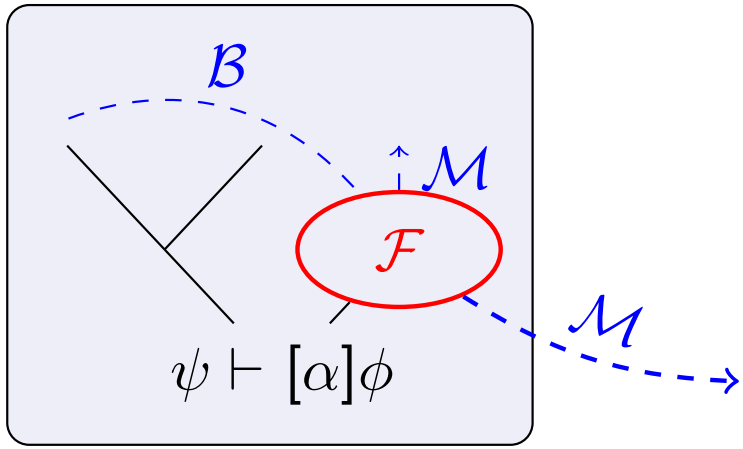
valid "reduction" but perfectly useless ($\Rightarrow$ proof loops)

How to Navigate among Nondeterminisms?

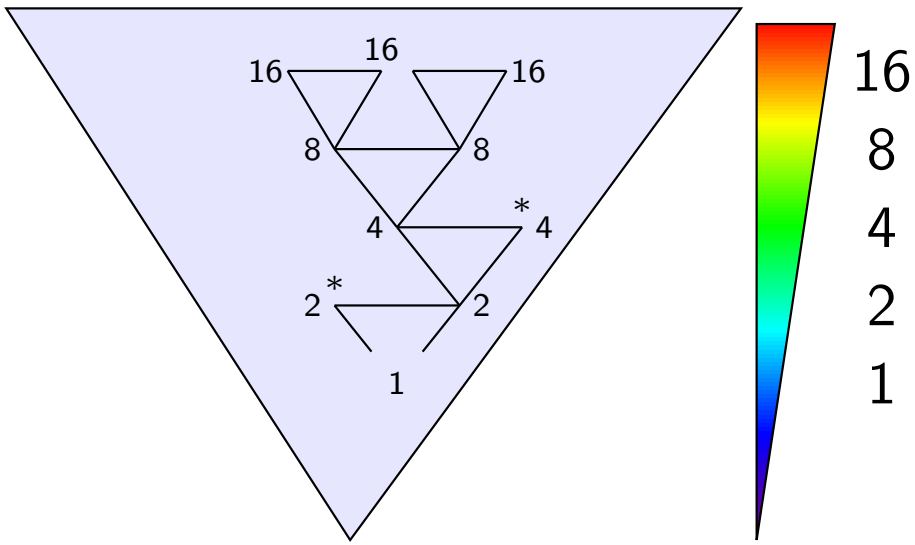
Proof Strategy Priorities for Formula Selection

"accept QE if variable eliminated"
ensures progress and termination

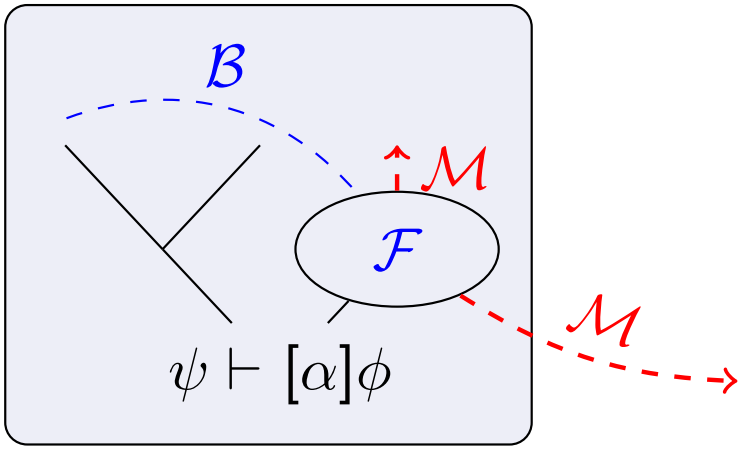
1. *arithmetic rules* if variable eliminated
2. propositional rules P1–P4, P8–P9 on modalities
3. dynamic rules
4. splitting rules P5–P7 on modalities
5. *arithmetic rules* if variable eliminated
6. propositional rules P1–P9 on first-order formulas



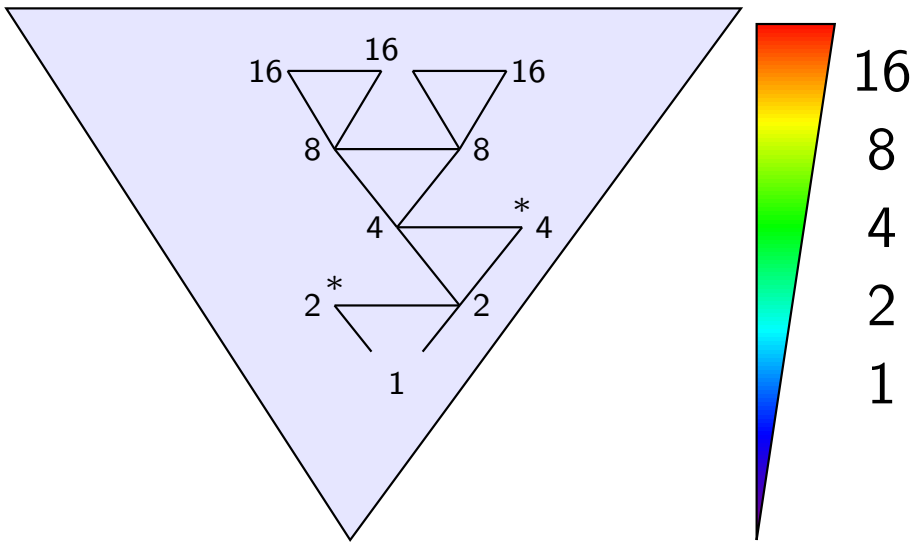
Iterative Background Closure (IBC) Strategy



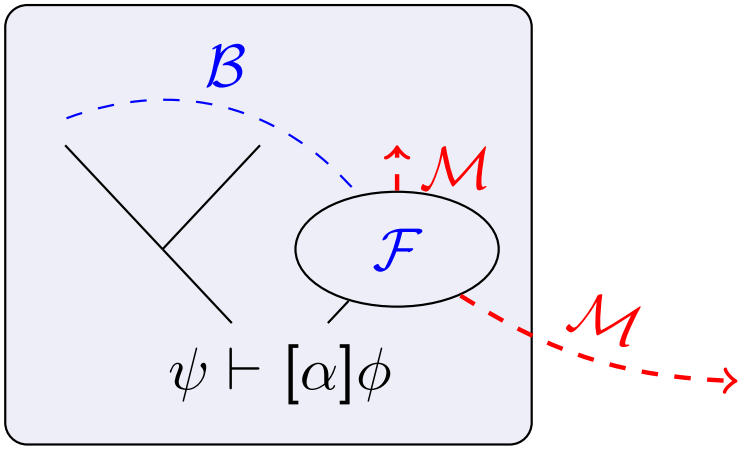
background
timeout



Iterative Background Closure (IBC) Strategy



background
timeout



- Periodical background arithmetic with increasing timeout after split
- Avoid splitting in average case
- Split prohibitively complicated cases

Outline

Full European Train Control System

$$
\begin{array}{lcl}
\text{system} & : & \big(\text{poll};\ (\text{negot} \cup (\text{speedControl};\ \text{atp};\ \text{move}))\big)^* \\
\text{init} & : & \mathit{drive} := 0;\ \mathit{brake} := 1 \\
\text{poll} & : & \mathit{SB} := \frac{v^2-d^2}{2b} + \left(\frac{a_{max}}{b}+1\right)\left(\frac{a_{max}}{2}\varepsilon^2 + \varepsilon v\right);\ \mathit{ST} := * \\
\text{negot} & : & (?m - z > \mathit{ST}) \cup (?m - z \leq \mathit{ST};\ \text{rbc}) \\
\text{rbc} & : & (\mathit{vdes} := *;\ ?\mathit{vdes} > 0) \cup (\mathit{state} := \mathit{brake}) \\
 & & \cup \big(d_{old} := d;\ m_{old} := m;\ m := *;\ d := *; \\
 & & \quad ?d \geq 0 \wedge d_{old}^2 - d^2 \leq 2b(m - m_{old})\big) \\
\text{speedCtrl} & : & (?\mathit{state} = \mathit{brake};\ a := -b) \\
 & & \cup \Big(?\mathit{state} = \mathit{drive}; \\
 & & \quad \big((?v \leq v_{des};\ a := *;\ ? - b \leq a \leq a_{max}) \\
 & & \quad \cup (?v \geq v_{des};\ a := *;\ ?0 > a \geq -b)\big)\Big) \\
\text{atp} & : & (?m - z \leq \mathit{SB};\ a := -b) \cup (?m - z > \mathit{SB}) \\
\text{move} & : & t := 0;\ \{\dot{z} = v, \dot{v} = a, \dot{t} = 1, (v \geq 0 \wedge t \leq \varepsilon)\}
\end{array}
$$

Full European Train Control System

provable automatically with IBC!

only $1 \ll 52$ user interaction + reduced verification time!

$$
\begin{array}{lcl}
\text{system} & : & \big(\text{poll};\ (\text{negot} \cup (\text{speedControl};\ \text{atp};\ \text{move}))\big)^* \\
\text{init} & : & \mathit{drive} := 0;\ \mathit{brake} := 1 \\
\text{poll} & : & \mathit{SB} := \frac{v^2-d^2}{2b} + \left(\frac{a_{max}}{b}+1\right)\left(\frac{a_{max}}{2}\varepsilon^2 + \varepsilon v\right);\ \mathit{ST} := * \\
\text{negot} & : & (?m - z > \mathit{ST}) \cup (?m - z \leq \mathit{ST};\ \text{rbc}) \\
\text{rbc} & : & (\mathit{vdes} := *;\ ?\mathit{vdes} > 0) \cup (\mathit{state} := \mathit{brake}) \\
 & & \cup \big(d_{old} := d;\ m_{old} := m;\ m := *;\ d := *; \\
 & & \quad ?d \geq 0 \wedge d_{old}^2 - d^2 \leq 2b(m - m_{old})\big) \\
\text{speedCtrl} & : & (?\mathit{state} = \mathit{brake};\ a := -b) \\
 & & \cup \Big(?\mathit{state} = \mathit{drive}; \\
 & & \quad \big((?v \leq v_{des};\ a := a_{max}) \\
 & & \quad \cup (?v \geq v_{des};\ a := -b)\big)\Big) \\
\text{atp} & : & (?m - z \leq \mathit{SB};\ a := -b) \cup (?m - z > \mathit{SB}) \\
\text{move} & : & t := 0;\ \{\dot{z} = v, \dot{v} = a, \dot{t} = 1, (v \geq 0 \wedge t \leq \varepsilon)\}
\end{array}
$$

Preliminary Experimental Results

Case Study	Interactions	IBC	No IBC
ETCS-binary	1	89s	>8h
ETCS-binary	2	<89s	1184s
ETCS-binary	3	<89s	30s
ETCS	1	3000s	∞
ETCS	2	500s	∞
ETCS	10		427s
ETCS-optimal	2	>3h	∞
ETCS-binary	1	89s	
ETCS	1	1381s	
ETCS	2	271s	
Water tank	1	4.7s	

Outline

Future Work

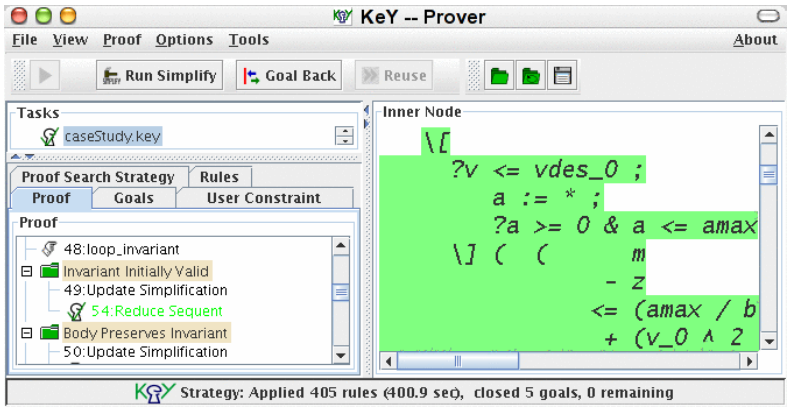
- More experimental evaluation
  - More examples (currently: 4)
  - Effect of strategy variations
- Guide variable selection

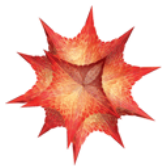
Conclusions

differential dynamic logic

$$d\mathcal{L} = DL + HP$$

- Deductively verify hybrid systems
- Practical perspective
- Surprisingly challenging nondeterminism
- Tremendous impact of Iterative Background Closure (IBC)
- Train control (ETCS) verification
- Verification tool HyKeY





KeY