Formal Approaches to Computer Networks

Mohammad Mousavi (KCL)

Joint work with: Georgiana Caltais (UT), Hossein Hojjat (TelAS), Hünkar Can Tunç (AU)

OPCT 2023, 26-30 June 2023, Bertinoro, Italy

Contributions

- Minimalist formal language for Software Defined Networks enabling
  - Dynamic network reconfigurations / flow table updates
  - Interaction between control plane and data plane
  - Scalable analysis for realistic cases using equational reasoning

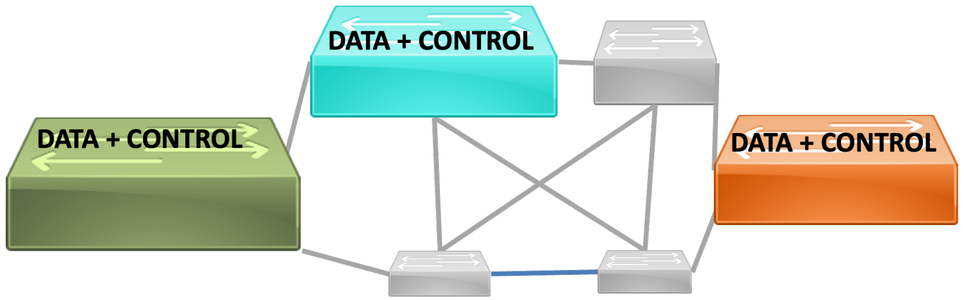
Software Defined Networks (SDNs)

What?

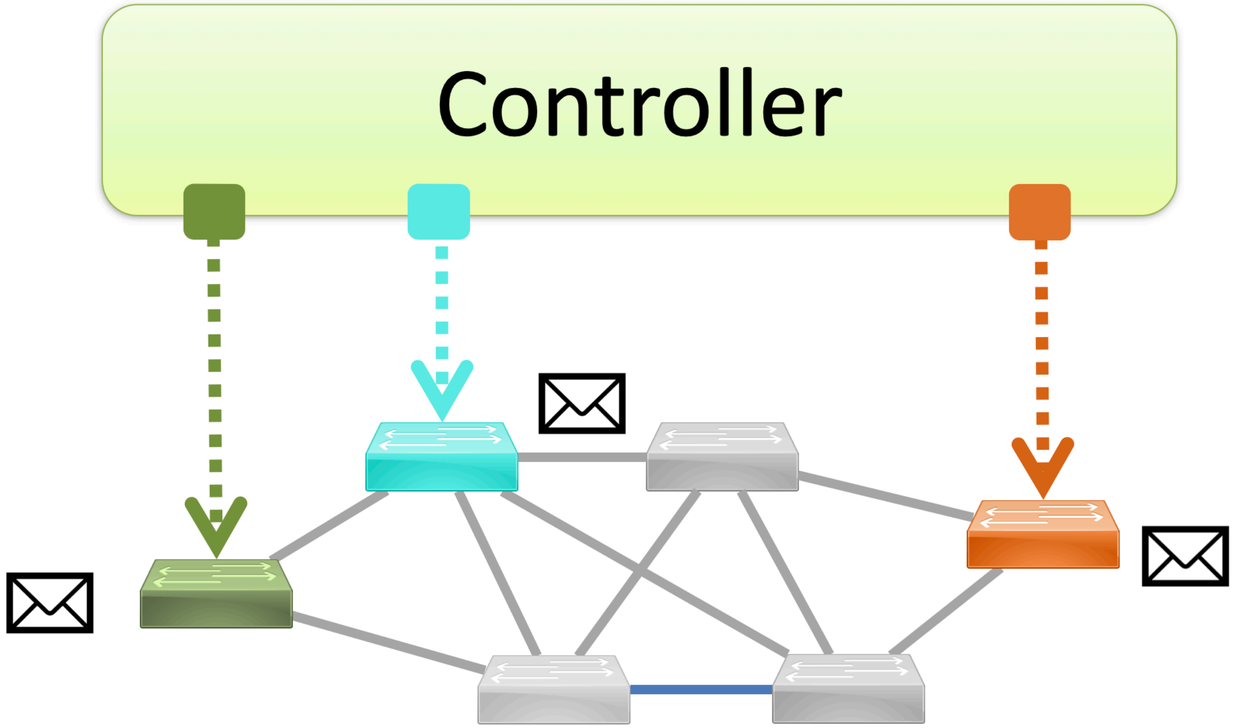
- A major trend in industry
  - Extensively applied by various cloud service providers (Google, Microsoft)
- Make networks programmable
  - Efficient network management
    - Improve scalability, adaptability
  - Cost savings
    - Shift from running proprietary code to open software run on commodity hardware

Software Defined Networks (SDNs)

Old-Fashioned



SDNs



Source: medium.com

Verification of Networks

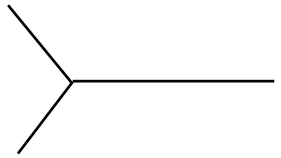
Mathematically inspired PL

- Flowlog [Nelson & al., NSDI'14]
- Kinetic [Kim & al., NSDI'15]
- NetKAT [Anderson & al., POPL'14][Foster & al., POPL'15]
- WNetKAT [Larsen & al., OPODIS'16]
- AllSynth [Larsen & al., FASE'22]

NetKAT

Kleene Algebra with Tests + Network Primitives

- NetKAT policies encode
  - Packet forwarding (+ . *)
  - Packet classification (= . + ¬)
  - Packet modification (f ← n)



KAT: Regular Expressions + Boolean Algebra

Packets σ, e.g.:

$$
\begin{aligned}
&\mathit{Fields} ::= f_1 \mid \ldots \mid f_k \\
&\qquad \sigma \triangleq \{f_1 = n_1, \ldots, f_k = n_k\} \\
&\qquad \sigma \triangleq \{sw = A,\ pt = 2,\ type = SSH,\ dst = H_2\}
\end{aligned}
$$

NetKAT
Topologies

Packets σ, e.g.:

$$
\begin{aligned}
\mathit{Fields} &::= f_1 \mid \ldots \mid f_k \\
\sigma &\triangleq \{f_1 = n_1, \ldots, f_k = n_k\} \\
\text{✉} &\triangleq \{sw = A,\ pt = 2,\ type = SSH,\ dst = H_2\}
\end{aligned}
$$

$$
\begin{aligned}
t \triangleq\ & sw = A \cdot pt = 2 \cdot sw \leftarrow B \cdot pt \leftarrow 1+ \\
& sw = B \cdot pt = 1 \cdot sw \leftarrow A \cdot pt \leftarrow 2+ \\
& \ldots
\end{aligned}
$$

NetKAT

Forwarding Tables

Packets σ, e.g.:

$$
\begin{aligned}
\mathit{Fields} &::= f_1 \mid \ldots \mid f_k \\
\sigma &\triangleq \{f_1 = n_1, \ldots, f_k = n_k\} \\
\text{✉} &\triangleq \{sw = A,\ pt = 2,\ type = SSH,\ dst = H_2\}
\end{aligned}
$$

Pattern	Action
$type = SSH$	$true$
$*$	$false$

$$p \triangleq type = SSH$$

NetKAT

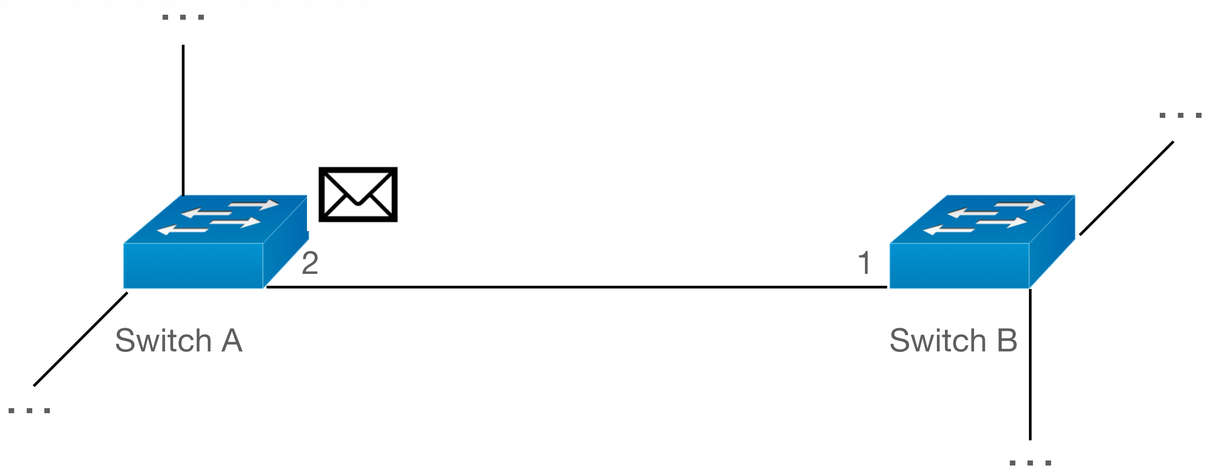
“Monitor SSH traffic between Switch A and Switch B”

$$✉ \triangleq \{sw = A,\ pt = 2,\ type = SSH,\ dst = H_2\}$$

$$p \triangleq type = SSH$$

$$(p \cdot t)^*$$

$$\begin{aligned} t \triangleq\ & sw = A \cdot pt = 2 \cdot sw \leftarrow B \cdot pt \leftarrow 1+ \\ & sw = B \cdot pt = 1 \cdot sw \leftarrow A \cdot pt \leftarrow 2+ \\ & \cdots \end{aligned}$$



NetKAT

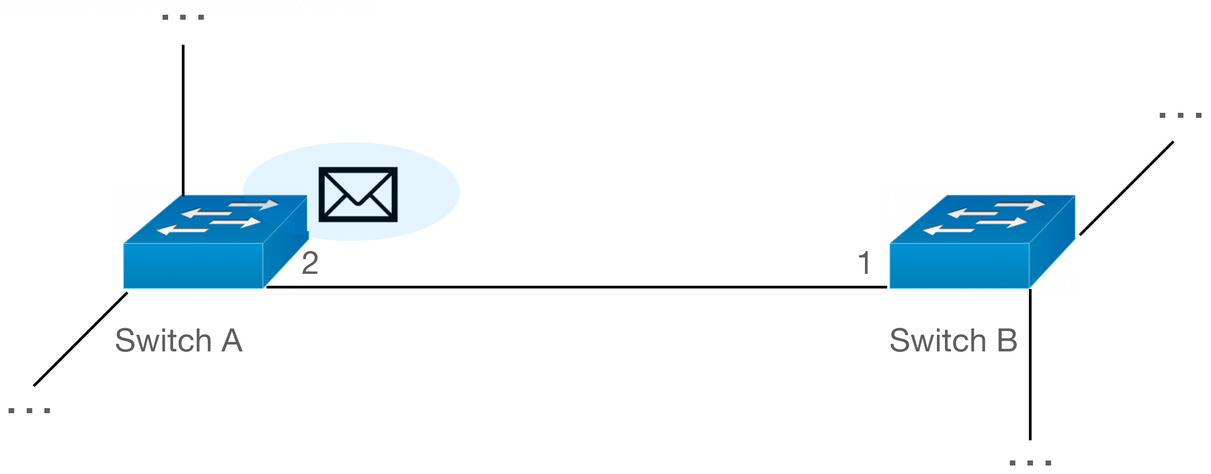
“Monitor SSH traffic between Switch A and Switch B”

$$✉ \triangleq \{sw = A,\ pt = 2,\ type = SSH,\ dst = H_2\}$$

$$p \triangleq type = SSH$$

$$(p \cdot t)^*$$

$$\begin{aligned} t \triangleq\ & sw = A \cdot pt = 2 \cdot sw \leftarrow B \cdot pt \leftarrow 1+ \\ & sw = B \cdot pt = 1 \cdot sw \leftarrow A \cdot pt \leftarrow 2+ \\ & \cdots \end{aligned}$$



NetKAT

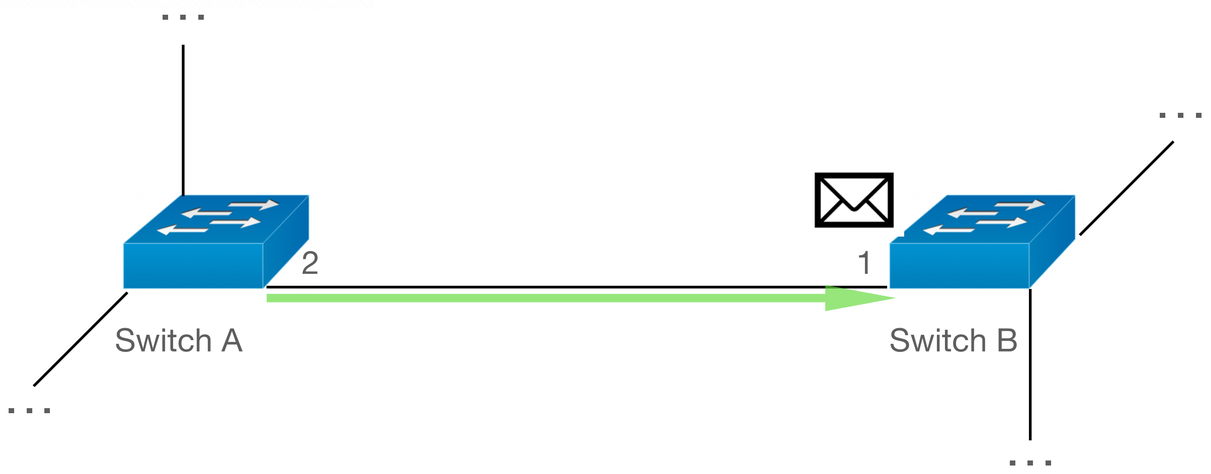
“Monitor SSH traffic between Switch A and Switch B”

$$✉ \triangleq \{sw = A,\ pt = 2,\ type = SSH,\ dst = H_2\}$$

$$p \triangleq type = SSH$$

$$t \triangleq \begin{array}{l} sw = A \cdot pt = 2 \cdot sw \leftarrow B \cdot pt \leftarrow 1+ \\ sw = B \cdot pt = 1 \cdot sw \leftarrow A \cdot pt \leftarrow 2+ \\ \ldots \end{array}$$

$$(p \cdot t)^*$$



NetKAT

“Monitor SSH traffic between Switch A and Switch B”

$$✉ \triangleq \{sw = A,\ pt = 2,\ type = SSH,\ dst = H_2\}$$

$$p \triangleq type = SSH$$

$$(p \cdot t)^*$$

$$\begin{aligned} t \triangleq\ & sw = A \cdot pt = 2 \cdot sw \leftarrow B \cdot pt \leftarrow 1+ \\ & sw = B \cdot pt = 1 \cdot sw \leftarrow A \cdot pt \leftarrow 2+ \\ & \cdots \end{aligned}$$

NetKAT

“Monitor SSH traffic between Switch A and Switch B”

$$✉ \triangleq \{sw = A,\ pt = 2,\ type = SSH,\ dst = H_2\}$$

$$p \triangleq type = SSH$$

$$(p \cdot t)^*$$

$$t \triangleq \begin{array}{l} sw = A \cdot pt = 2 \cdot sw \leftarrow B \cdot pt \leftarrow 1+ \\ sw = B \cdot pt = 1 \cdot sw \leftarrow A \cdot pt \leftarrow 2+ \\ \ldots \end{array}$$

NetKAT

“Monitor SSH traffic between Switch A and Switch B”

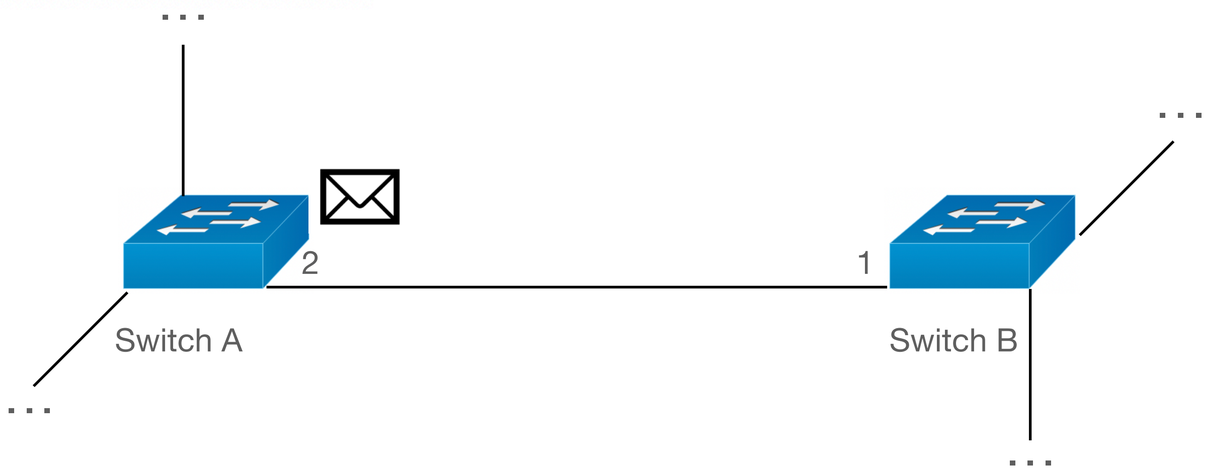
$$✉ \triangleq \{sw = A,\ pt = 2,\ type = SSH,\ dst = H_2\}$$

$$p \triangleq type = SSH$$

$$t \triangleq \begin{array}{l} sw = A \cdot pt = 2 \cdot sw \leftarrow B \cdot pt \leftarrow 1+ \\ sw = B \cdot pt = 1 \cdot sw \leftarrow A \cdot pt \leftarrow 2+ \\ \ldots \end{array}$$

$$(p \cdot t)^*$$

$$\mathbf{dup} \cdot (p \cdot t \cdot \mathbf{dup})^*$$



NetKAT framework in a Nutshell

Syntax & (Big-Step Denotational) Semantics

$$Pr ::= \mathbf{0} \mid \mathbf{1} \mid f = n \mid Pr + Pr \mid Pr \cdot Pr \mid \neg Pr$$
$$N ::= Pr \mid f \leftarrow n \mid N + N \mid N \cdot N \mid N^* \mid \mathbf{dup}$$

$$\begin{aligned} \mathit{Fields} &::= f_1 \mid \ldots \mid f_k \\ \mathit{Packets}\ \sigma &\triangleq \{f_1 = n_1, \ldots, f_k = n_k\} \\ \mathit{Histories}\ h &\triangleq \sigma_1 :: \sigma_2 :: \ldots :: \sigma_k :: \langle\rangle \end{aligned}$$

$$[\![-]\!] : N \to (H \to \mathcal{P}(H))$$

$$\begin{aligned} [\![\mathbf{1}]\!](h) &\triangleq \{h\} \\ [\![\mathbf{0}]\!](h) &\triangleq \{\} \\ [\![f = n]\!]\,(\sigma::h) &\triangleq \begin{cases} \{\sigma::h\} & \text{if } \sigma(f) = n \\ \{\} & \text{otherwise} \end{cases} \\ [\![\neg a]\!]\,(h) &\triangleq \{h\} \setminus [\![a]\!]\,(h) \\ [\![f \leftarrow n]\!]\,(\sigma::h) &\triangleq \{\sigma[f := n]::h\} \\ [\![p + q]\!]\,(h) &\triangleq [\![p]\!]\,(h) \cup [\![q]\!]\,(h) \end{aligned}$$

$$\begin{aligned} [\![p \cdot q]\!]\,(h) &\triangleq ([\![p]\!] \bullet [\![q]\!])\,(h) \\ [\![p^*]\!]\,(h) &\triangleq \bigcup_{i \in N} F^i\,(h) \\ F^0\,(h) &\triangleq \{h\} \\ F^{i+1}\,(h) &\triangleq ([\![p]\!] \bullet F^i)\,(h) \\ (f \bullet g)(x) &\triangleq \bigcup\{g(y) \mid y \in f(x)\} \\ [\![\mathbf{dup}]\!]\,(\sigma::h) &\triangleq \{\sigma::(\sigma::h)\} \end{aligned}$$

NetKAT framework in a Nutshell

Syntax & (Big-Step Denotational) Semantics

$$Pr ::= \mathbf{0} \mid \mathbf{1} \mid f = n \mid Pr + Pr \mid Pr \cdot Pr \mid \neg Pr$$
$$N ::= Pr \mid f \leftarrow n \mid N + N \mid N \cdot N \mid N^* \mid \mathbf{dup}$$

$$\begin{aligned} Fields &::= f_1 \mid \ldots \mid f_k \\ Packets\ \sigma &\triangleq \{f_1 = n_1, \ldots, f_k = n_k\} \\ Histories\ h &\triangleq \sigma_1 :: \sigma_2 :: \ldots :: \sigma_k :: \langle\rangle \end{aligned}$$

$$[\![-]\!] : N \to (H \to \mathcal{P}(H))$$

$$\begin{aligned} [\![\mathbf{1}]\!](h) &\triangleq \{h\} \\ [\![\mathbf{0}]\!](h) &\triangleq \{\} \\ [\![f = n]\!]\ (\sigma::h) &\triangleq \begin{cases} \{\sigma::h\} & \text{if } \sigma(f) = n \\ \{\} & \text{otherwise} \end{cases} \\ [\![\neg a]\!]\ (h) &\triangleq \{h\} \setminus [\![a]\!]\ (h) \\ [\![f \leftarrow n]\!]\ (\sigma::h) &\triangleq \{\sigma[f := n]::h\} \\ [\![p + q]\!]\ (h) &\triangleq [\![p]\!]\ (h) \cup [\![q]\!]\ (h) \end{aligned}$$

$$\begin{aligned} [\![p \cdot q]\!]\ (h) &\triangleq ([\![p]\!] \bullet [\![q]\!])\ (h) \\ [\![p^*]\!]\ (h) &\triangleq \bigcup_{i \in N} F^i\ (h) \\ F^0\ (h) &\triangleq \{h\} \\ F^{i+1}\ (h) &\triangleq ([\![p]\!] \bullet F^i)\ (h) \\ (f \bullet g)(x) &\triangleq \bigcup \{g(y) \mid y \in f(x)\} \\ [\![\mathbf{dup}]\!]\ (\sigma::h) &\triangleq \{\sigma::(\sigma::h)\} \end{aligned}$$

Sound & Complete Axiomatisation (ENK)

$$[\![p]\!] = [\![q]\!]\ \textit{iff}\ E_{NK} \vdash p \equiv q$$

NetKAT

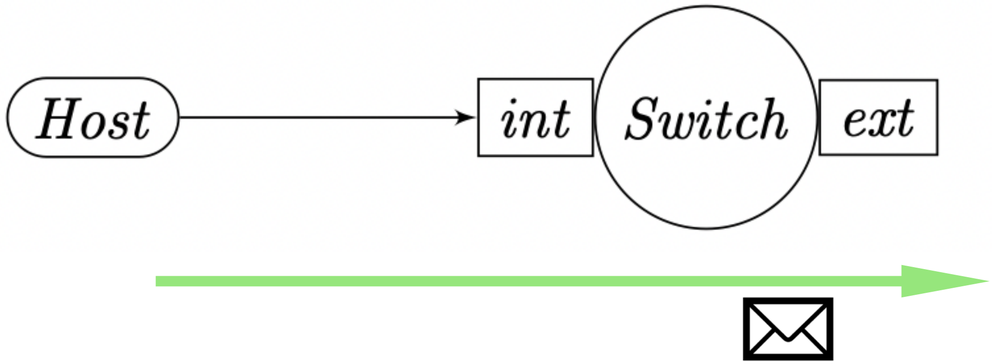
Extensions & Limitations

- Extensions
  - Congestion control [Foster & al., ESOP'16]
  - History-based routing, Stateful network updates [Beckett & al., PLDI'16] [McClurg & al., PLDI'16]
  - Higher order functions [Vandenbroucke & al., POPL'20]
  - Concurrent extensions [Wagemaker & al., CONCUR'20] [Wagemaker & al., ESOP'22]
- Limitations
  - Dynamic network reconfigurations / flow table updates
  - Interaction between control plane and data plane

Dynamic NetKAT (DyNetKAT)

Supports…

- Dynamic & stateful behaviour
- Synchronisation of
  - Controllers trigger network reconfigurations
  - Switches accept reconfigurations & update flow tables

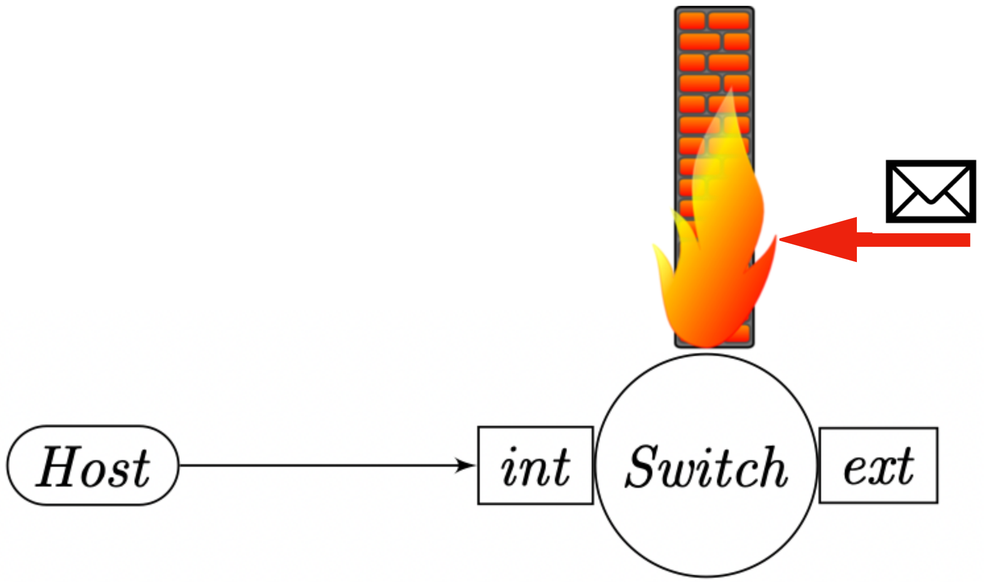


Stateful Firewall

Dynamic NetKAT (DyNetKAT)

Supports…

- Dynamic & stateful behaviour
- Synchronisation of
  - Controllers trigger network reconfigurations
  - Switches accept reconfigurations & update flow tables

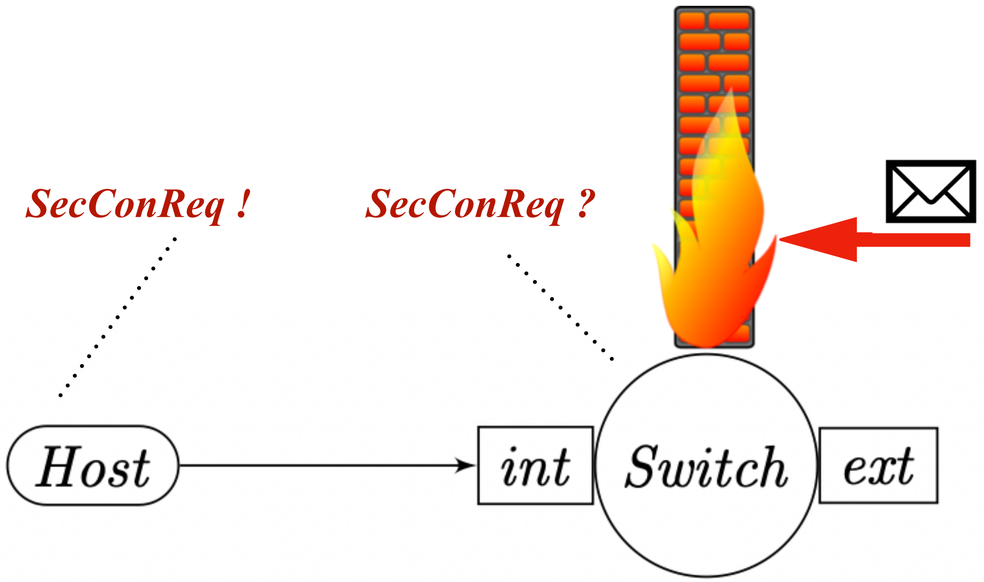


Stateful Firewall

Dynamic NetKAT (DyNetKAT)

Supports…

- Dynamic & stateful behaviour
- Synchronisation of
  - Controllers trigger network reconfigurations
  - Switches accept reconfigurations & update flow tables

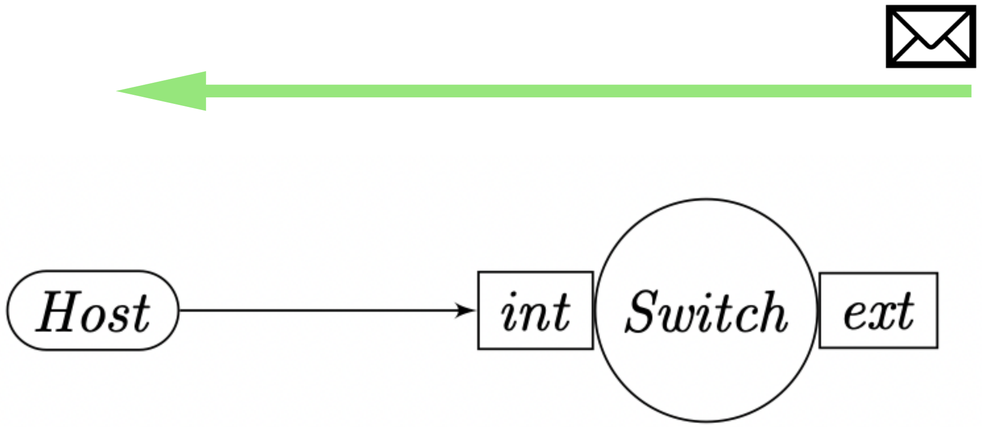


Stateful Firewall

Dynamic NetKAT (DyNetKAT)

Supports…

- Dynamic & stateful behaviour
- Synchronisation of
  - Controllers trigger network reconfigurations
  - Switches accept reconfigurations & update flow tables



Stateful Firewall

DyNetKAT
Overview

- Conservative extension of NetKAT
- Minimalist language
- Multi-packet behaviour
  - Every “in-flight” packet observes a single set of flow tables
    - Inspired by abstractions in the domain [Reitblatt & al., SIGCOMM’12]

DyNetKAT Language

A Process Algebraic Approach

- Syntax

$$N ::= \mathrm{NetKAT}^{-\mathbf{dup}}$$

$$D ::= \bot \mid N\,;D \mid x?N\,;D \mid x!N\,;D \mid D \,||\, D \mid D \oplus D \mid X$$
$$X \triangleq D$$

DyNetKAT Language
A Process Algebraic Approach

- Syntax

$$N ::= \mathrm{NetKAT}^{-\mathbf{dup}}$$

$$D ::= \bot \mid N\,;D \mid x?N\,;D \mid x!N\,;D \mid D \,||\, D \mid D \oplus D \mid X$$
$$X \triangleq D$$

- (Small-Step Operational) Semantics $(p, H_0, H_1) \xrightarrow{\gamma} (p', H_0', H_1')$

DyNetKAT Language
A Process Algebraic Approach

- Syntax

$$N ::= \mathrm{NetKAT}^{-\mathbf{dup}}$$

$$D ::= \bot \mid N\,;D \mid x?N\,;D \mid x!N\,;D \mid D \,||\, D \mid D \oplus D \mid X$$
$$X \triangleq D$$

- (Small-Step Operational) Semantics $(p, H_0, H_1) \xrightarrow{\gamma} (p', H_0', H_1')$

$$\frac{\sigma' \in [\![p]\!](\sigma :: \langle\rangle)}{(p; q, \sigma :: H, H') \xrightarrow{(\sigma,\sigma')} (q, H, \sigma' :: H')}$$

DyNetKAT Language
A Process Algebraic Approach

- Syntax

$$N ::= \mathrm{NetKAT}^{-\mathbf{dup}}$$

$$D ::= \bot \mid N\,;D \mid x?N\,;D \mid x!N\,;D \mid D \,||\, D \mid D \oplus D \mid X$$
$$X \triangleq D$$

- (Small-Step Operational) Semantics

$$(p, H_0, H_1) \xrightarrow{\gamma} (p', H_0', H_1')$$

$$\frac{\sigma' \in [\![p]\!](\sigma :: \langle\rangle)}{(p; q, \sigma :: H, H') \xrightarrow{(\sigma,\sigma')} (q, H, \sigma' :: H')}$$

$$\frac{(p, H_0, H_0') \xrightarrow{\gamma} (p', H_1, H_1')}{(p \oplus q, H_0, H_0') \xrightarrow{\gamma} (p', H_1, H_1')}$$

DyNetKAT Language
A Process Algebraic Approach

- Syntax

$$N ::= \mathrm{NetKAT}^{-\mathbf{dup}}$$

$$D ::= \bot \mid N\,;D \mid x?N\,;D \mid x!N\,;D \mid D \,||\, D \mid D \oplus D \mid X$$
$$X \triangleq D$$

- (Small-Step Operational) Semantics

$$(p, H_0, H_1) \xrightarrow{\gamma} (p', H_0', H_1')$$

$$\frac{\sigma' \in [\![p]\!](\sigma :: \langle\rangle)}{(p; q, \sigma :: H, H') \xrightarrow{(\sigma,\sigma')} (q, H, \sigma' :: H')}$$

$$\frac{(p, H_0, H_0') \xrightarrow{\gamma} (p', H_1, H_1')}{(p \oplus q, H_0, H_0') \xrightarrow{\gamma} (p', H_1, H_1')}$$

$$\frac{(q, H, H') \xrightarrow{x\,\clubsuit\,p} (q', H, H') \quad (s, H, H') \xrightarrow{x\,\spadesuit\,p} (s', H, H')}{(q||s, H, H') \xrightarrow{\mathbf{rcfg(x,p)}} (q'||s', H, H')}$$

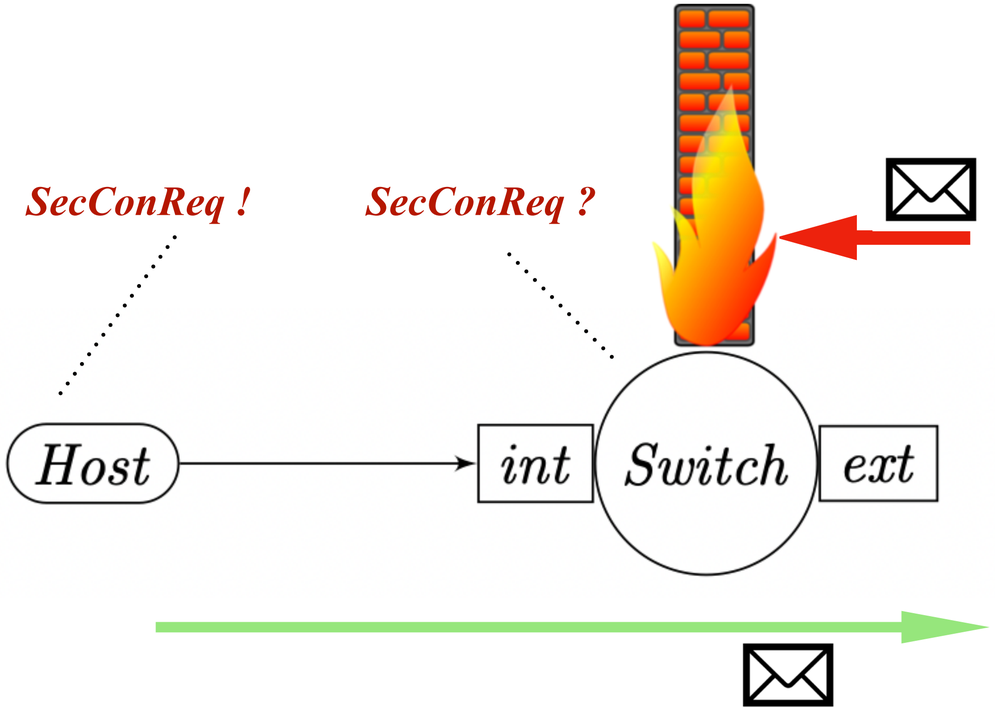
$\clubsuit =?$ $\spadesuit =!$
or
$\clubsuit =!$ $\spadesuit =?$

Example

$$Switch \triangleq ((port = int) \cdot (port \leftarrow ext)) \,;\, Switch \oplus$$
$$((port = ext) \cdot \mathbf{0}) \,;\, Switch \oplus$$
$$secConReq?\mathbf{1} \,;\, Switch'$$
$$Switch' \triangleq ((port = int) \cdot (port \leftarrow ext)) \,;\, Switch' \oplus$$
$$((port = ext) \cdot (port \leftarrow int)) \,;\, Switch' \oplus$$
$$secConEnd?\mathbf{1} \,;\, Switch$$

$$Host \triangleq secConReq!\mathbf{1} \,;\, Host \oplus$$
$$secConEnd!\mathbf{1} \,;\, Host$$

$$Init \triangleq Host \,||\, Switch$$



Stateful Firewall

DyNetKAT Equivalence

Bisimilarity $(\sim)$**.** A symmetric relation R over DyNetKAT is a *bisimulation* whenever for $(p, q) \in R$ the following holds:
If $(p, H_0, H_1) \xrightarrow{\gamma} (p', H_0', H_1')$ then exists q' s.t. $(q, H_0, H_1) \xrightarrow{\gamma} (q', H_0', H_1')$ and $(p', q') \in R$, with $\gamma ::= (\sigma, \sigma') \mid x?r \mid x!r \mid \mathbf{rcfg}(\mathbf{x}, \mathbf{r})$.

DyNetKAT Equivalence

Semantic Layering. Let p and q be NetKAT$^{-\mathbf{dup}}$ policies. The following holds: $[\![p]\!] = [\![q]\!]$ iff $(p; d) \sim (q; d)$ for any DyNetKAT policy d.

- *Conservative extension of the NetKAT semantics*
- *Re-usability of the NetKAT tools (e.g., axiomatisation)*
- *Modular extension to dynamic behaviour - DyNetKAT*

DyNetKAT Equivalence

Sound & Ground Complete Axiomatisation (E_{DNK}) - ACP style

for $p, q, r \in \text{DyNetKAT}$ and $z, y \in \text{NetKAT}^{-\mathbf{dup}}$

for $a ::= z \mid x?z \mid x!z \mid \mathbf{rcfg}_{x,z}$

$$\mathbf{0}\,;p \equiv \bot \qquad (A0)$$
$$(z+y)\,;p \equiv z\,;p \oplus y\,;p \qquad (A1)$$
$$p \oplus q \equiv q \oplus p \qquad (A2)$$
$$(p \oplus q) \oplus r \equiv p \oplus (q \oplus r) \qquad (A3)$$
$$p \oplus p \equiv p \qquad (A4)$$
$$p \oplus \bot \equiv p \qquad (A5)$$
$$p \,||\, q \equiv q \,||\, p \qquad (A6)$$
$$p \,||\, \bot \equiv p \qquad (A7)$$
$$p \,||\, q \equiv p \between q \oplus q \between p \oplus p \mid q \qquad (A8)$$
$$\bot \between p \equiv \bot \qquad (A9)$$
$$(a\,;p) \between q \equiv a\,;(p \,||\, q) \qquad (A10)$$
$$(p \oplus q) \between r \equiv (p \between r) \oplus (q \between r) \qquad (A11)$$
$$(x?z\,;p) \mid (x!z\,;q) \equiv \mathbf{rcfg}_{x,z}\,;(p \,||\, q) \qquad (A12)$$
$$(p \oplus q) \mid r \equiv (p \mid r) \oplus (q \mid r) \qquad (A13)$$
$$p \mid q \equiv q \mid p \qquad (A14)$$
$$p \mid q \equiv \bot \; [owise] \qquad (A15)$$

for $at ::= \alpha \cdot \pi \mid x?z \mid x!z \mid \mathbf{rcfg}_{x,z}$:

$$\delta_{\mathcal{L}}(\bot) \equiv \bot \qquad (\delta_\bot)$$
$$\delta_{\mathcal{L}}(at\,;p) \equiv at\,;\delta_{\mathcal{L}}(p) \text{ if } at \notin \mathcal{L} \qquad (\delta_;)$$
$$\delta_{\mathcal{L}}(at\,;p) \equiv \bot \text{ if } at \in \mathcal{L} \qquad (\delta_;^\bot)$$
$$\delta_{\mathcal{L}}(p \oplus q) \equiv \delta_{\mathcal{L}}(p) \oplus \delta_{\mathcal{L}}(q) \qquad (\delta_\oplus)$$

for $n \in \mathbb{N}$:

$$\pi_0(p) \equiv \bot \qquad (\Pi_0)$$
$$\pi_n(\bot) \equiv \bot \qquad (\Pi_\bot)$$
$$\pi_{n+1}(at\,;p) \equiv at\,;\pi_n(p) \qquad (\Pi_;)$$
$$\pi_n(p \oplus q) \equiv \pi_n(p) \oplus \pi_n(q) \qquad (\Pi_\oplus)$$

$$p \equiv q \text{ if } \forall n \in \mathbb{N} : \pi_n(p) \equiv \pi_n(q) \; (AIP)$$

E_{NK}

Soundness & Completeness:
$p \sim q$ *iff* $E_{DNK} \vdash p \equiv q$

DyNetKAT Equivalence

Sound & Ground Complete Axiomatisation (E_{DNK}) - ACP style

for $p, q, r \in \mathrm{DyNetKAT}$ and $z, y \in \mathrm{NetKAT}^{-\mathbf{dup}}$

for $a ::= z \mid x?z \mid x!z \mid \mathbf{rcfg}_{x,z}$

$$\mathbf{0}\,;p \equiv \bot \qquad (A0)$$
$$(z+y)\,;p \equiv z\,;p \oplus y\,;p \qquad (A1)$$
$$p \oplus q \equiv q \oplus p \qquad (A2)$$
$$(p \oplus q) \oplus r \equiv p \oplus (q \oplus r) \qquad (A3)$$
$$p \oplus p \equiv p \qquad (A4)$$
$$p \oplus \bot \equiv p \qquad (A5)$$
$$p \,||\, q \equiv q \,||\, p \qquad (A6)$$
$$p \,||\, \bot \equiv p \qquad (A7)$$
$$p \,||\, q \equiv p \between q \oplus q \between p \oplus p \mid q \qquad (A8)$$
$$\bot \between p \equiv \bot \qquad (A9)$$
$$(a\,;p) \between q \equiv a\,;(p \,||\, q) \qquad (A10)$$
$$(p \oplus q) \between r \equiv (p \between r) \oplus (q \between r) \qquad (A11)$$
$$(x?z\,;p) \mid (x!z\,;q) \equiv \mathbf{rcfg}_{x,z}\,;(p \,||\, q) \qquad (A12)$$
$$(p \oplus q) \mid r \equiv (p \mid r) \oplus (q \mid r) \qquad (A13)$$
$$p \mid q \equiv q \mid p \qquad (A14)$$
$$p \mid q \equiv \bot \ [owise] \qquad (A15)$$

for $at ::= \alpha \cdot \pi \mid x?z \mid x!z \mid \mathbf{rcfg}_{x,z}$:

$$\delta_{\mathcal{L}}(\bot) \equiv \bot \qquad (\delta_\bot)$$
$$\delta_{\mathcal{L}}(at\,;p) \equiv at\,;\delta_{\mathcal{L}}(p) \text{ if}$$
$$\delta_{\mathcal{L}}(at\,;p) \equiv \bot \text{ if } at \in \mathcal{L}$$
$$\delta_{\mathcal{L}}(p \oplus q) \equiv \delta_{\mathcal{L}}(p) \oplus \delta_{\mathcal{L}}($$

$$(z+y)\,;p \equiv z\,;p \oplus y\,;p \ (A1)$$

for $n \in \mathbb{N}$:

$$\pi_0(p) \equiv \bot \qquad (\Pi_0)$$
$$\pi_n(\bot) \equiv \bot \qquad (\Pi_\bot)$$
$$\pi_{n+1}(at\,;p) \equiv at\,;\pi_n(p) \qquad (\Pi_;)$$
$$\pi_n(p \oplus q) \equiv \pi_n(p) \oplus \pi_n(q) \qquad (\Pi_\oplus)$$

$$p \equiv q \text{ if } \forall n \in \mathbb{N} : \pi_n(p) \equiv \pi_n(q) \ (AIP)$$

E_{NK}

Soundness & Completeness:
$p \sim q$ *iff* $E_{DNK} \vdash p \equiv q$

DyNetKAT Equivalence

Sound & Ground Complete Axiomatisation (E_{DNK}) - ACP style

for $p, q, r \in \mathrm{DyNetKAT}$ and $z, y \in \mathrm{NetKAT}^{-\mathbf{dup}}$

for $a ::= z \mid x?z \mid x!z \mid \mathbf{rcfg}_{x,z}$

$$\mathbf{0}\,;p \equiv \bot \qquad (A0)$$
$$(z+y)\,;p \equiv z\,;p \oplus y\,;p \qquad (A1)$$
$$p \oplus q \equiv q \oplus p \qquad (A2)$$
$$(p \oplus q) \oplus r \equiv p \oplus (q \oplus r) \qquad (A3)$$
$$p \oplus p \equiv p \qquad (A4)$$
$$p \oplus \bot \equiv p \qquad (A5)$$
$$p \,||\, q \equiv q \,||\, p \qquad (A6)$$
$$p \,||\, \bot \equiv p \qquad (A7)$$
$$p \,||\, q \equiv p \between q \oplus q \between p \oplus p \mid q \qquad (A8)$$
$$\bot \between p \equiv \bot \qquad (A9)$$
$$(a\,;p) \between q \equiv a\,;(p \,||\, q) \qquad (A10)$$
$$(p \oplus q) \between r \equiv (p \between r) \oplus (q \between r) \qquad (A11)$$
$$(x?z\,;p) \mid (x!z\,;q) \equiv \mathbf{rcfg}_{x,z}\,;(p \,||\, q) \qquad (A12)$$
$$(p \oplus q) \mid r \equiv (p \mid r) \oplus (q \mid r) \qquad (A13)$$
$$p \mid q \equiv q \mid p \qquad (A14)$$
$$p \mid q \equiv \bot \ [owise] \qquad (A15)$$

for $at ::= \alpha \cdot \pi \mid x?z \mid x!z \mid \mathbf{rcfg}_{x,z}$:

$$\delta_{\mathcal{L}}(\bot) \equiv \bot \qquad (\delta_\bot)$$
$$\delta_{\mathcal{L}}(at\,;p) \equiv at\,;\delta_{\mathcal{L}}(p) \text{ if } at \notin \mathcal{L} \qquad (\delta_;)$$
$$\delta_{\mathcal{L}}(at\,;p) \equiv \bot \text{ if } at \in \mathcal{L} \qquad (\delta_;^\bot)$$
$$\delta_{\mathcal{L}}(p \oplus q) \equiv \delta_{\mathcal{L}}(p) \oplus \delta_{\mathcal{L}}(q) \qquad (\delta_\oplus)$$

for $n \in \mathbb{N}$:

$$\pi_0(p) \equiv \bot \qquad (\Pi_0)$$
$$\pi_n(\bot) \equiv \bot \qquad (\Pi_\bot)$$
$$\pi_{n+1}(at\,;p) \equiv at\,;\pi_n(p) \qquad (\Pi_;)$$
$$\pi_n(p \oplus q) \equiv \pi_n(p) \oplus \pi_n(q) \qquad (\Pi_\oplus)$$

$$p \equiv q \text{ if } \forall n \in \mathbb{N} : \pi_n(p) \equiv \pi_n(q) \ (AIP)$$

E_{NK}

E_{NK}

Soundness & Completeness:
$p \sim q$ *iff* $E_{DNK} \vdash p \equiv q$

DyNetKAT Equivalence

Sound & Ground Complete Axiomatisation (E_{DNK}) - ACP style

for $p, q, r \in \mathrm{DyNetKAT}$ and $z, y \in \mathrm{NetKAT}^{-\mathbf{dup}}$
for $a ::= z \mid x?z \mid x!z \mid \mathbf{rcfg}_{x,z}$

$$\mathbf{0}\,;p \equiv \bot \qquad (A0)$$
$$(z+y)\,;p \equiv z\,;p \oplus y\,;p \qquad (A1)$$
$$p \oplus q \equiv q \oplus p \qquad (A2)$$
$$(p \oplus q) \oplus r \equiv p \oplus (q \oplus r) \qquad (A3)$$
$$p \oplus p \equiv p \qquad (A4)$$
$$p \oplus \bot \equiv p \qquad (A5)$$
$$p \,||\, q \equiv q \,||\, p \qquad (A6)$$
$$p \,||\, \bot \equiv p \qquad (A7)$$
$$p \,||\, q \equiv p \between q \oplus q \between p \oplus p \mid q \qquad (A8)$$
$$\bot \between p \equiv \bot \qquad (A9)$$
$$(a\,;p) \between q \equiv a\,;(p \,||\, q) \qquad (A10)$$
$$(p \oplus q) \between r \equiv (p \between r) \oplus (q \between r) \qquad (A11)$$
$$(x?z\,;p) \mid (x!z\,;q) \equiv \mathbf{rcfg}_{x,z}\,;(p \,||\, q) \qquad (A12)$$
$$(p \oplus q) \mid r \equiv (p \mid r) \oplus (q \mid r) \qquad (A13)$$
$$p \mid q \equiv q \mid p \qquad (A14)$$
$$p \mid q \equiv \bot \ [owise] \qquad (A15)$$

for $at ::= \alpha \cdot \pi \mid x?z \mid x!z \mid \mathbf{rcfg}_{x,z}$:

$$\delta_{\mathcal{L}}(\bot) \equiv \bot \qquad (\delta_\bot)$$
$$\delta_{\mathcal{L}}(at\,;p) \equiv at\,;\delta_{\mathcal{L}}(p) \text{ if } at \notin \mathcal{L} \qquad (\delta_;)$$
$$\delta_{\mathcal{L}}(at\,;p) \equiv \bot \text{ if } at \in \mathcal{L} \qquad (\delta_;^\bot)$$
$$\delta_{\mathcal{L}}(p \oplus q) \equiv \delta_{\mathcal{L}}(p) \oplus \delta_{\mathcal{L}}(q) \qquad (\delta_\oplus)$$

for $n \in \mathbb{N}$:

$$\pi_0(p) \equiv \bot \qquad (\Pi_0)$$
$$\pi_n(\bot) \equiv \bot \qquad (\Pi_\bot)$$
$$\pi_{n+1}(at\,;p) \equiv at\,;\pi_n(p) \qquad (\Pi_;)$$
$$\pi_n(p \oplus q) \equiv \pi_n(p) \oplus \pi_n(q) \qquad (\Pi_\oplus)$$

$$p \equiv q \text{ if } \forall n \in \mathbb{N} : \pi_n(p) \equiv \pi_n(q) \ (AIP)$$

E_{NK}

$$p \,||\, \bot \equiv p \ (A7)$$

Soundness & Completeness:
$p \sim q$ *iff* $E_{DNK} \vdash p \equiv q$

Safety / Reachability in DyNetKAT

Framework for Safety

- Specification language for safety properties

$$
\begin{aligned}
act &::= \alpha \cdot \pi \mid \mathbf{rcfg}_{x,p} \quad (\alpha \cdot \pi,\ \mathbf{rcfg}_{x,p} \in \mathcal{A}) \\
regexp &::= true \mid act \mid \neg act \mid regexp + regexp \mid regexp \cdot regexp \mid \\
&\quad\ (regexp)^n \quad (\text{with } n \geq 1) \\
prop &::= [regexp]false
\end{aligned}
$$

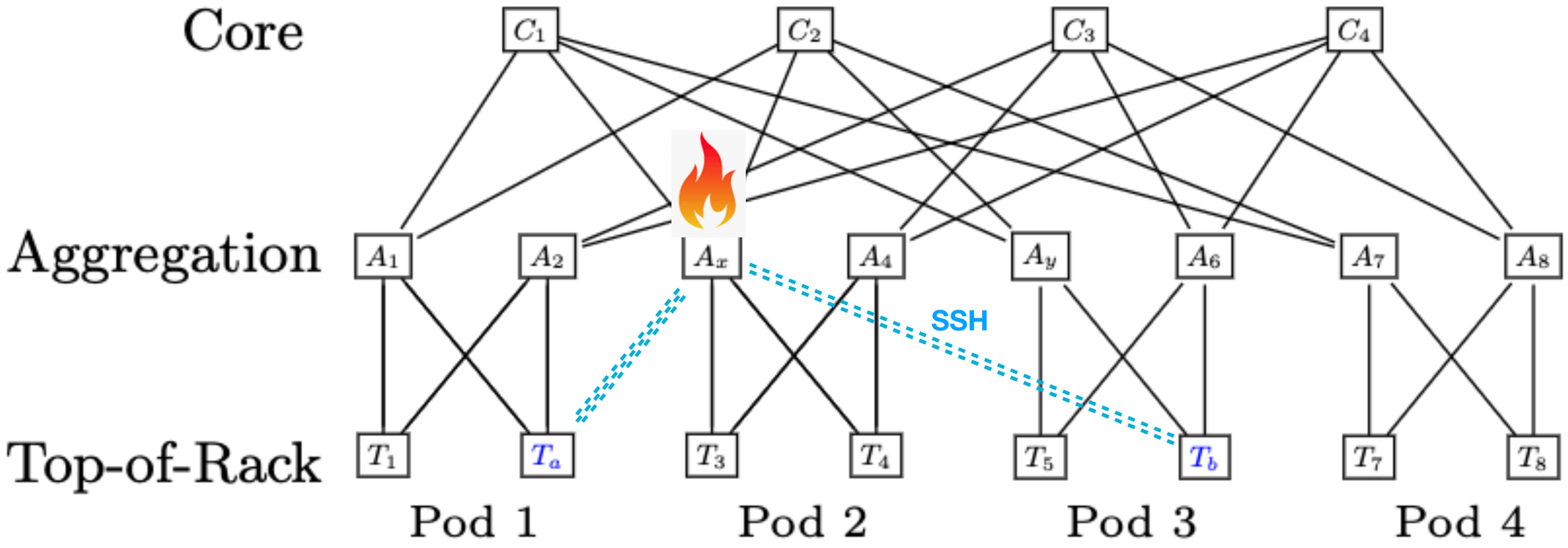
$$
[\![-]\!] : Prop \to \mathrm{DyNetKAT}
$$

- E.g., Stateful Firewall: “No traffic from Internet to intranet without secure connection”, equationally

$$
\begin{aligned}
spec_n &\triangleq [(\neg \mathbf{rcfg}_{secConReq,1})^n \cdot (port = ext) \cdot (port \leftarrow int)]false \\
E^{tr}_{DNK} &\vdash [\![spec_n]\!] \oplus Init \equiv [\![spec_n]\!]
\end{aligned}
$$

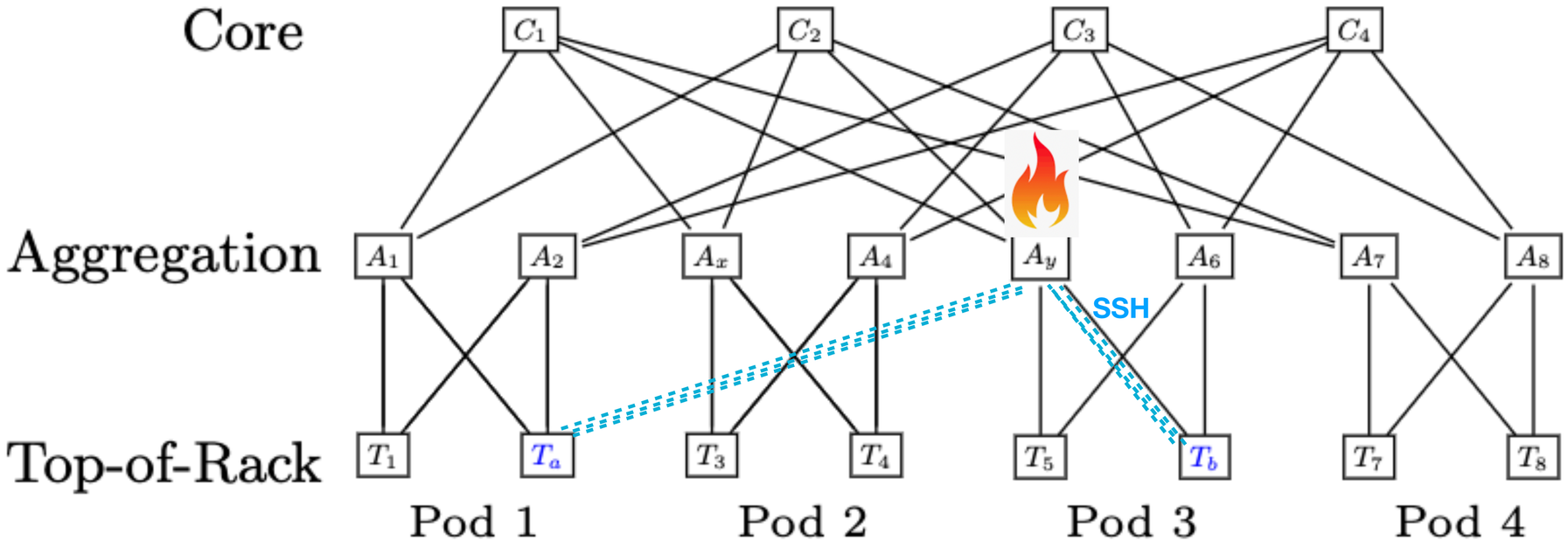
Safety / Reachability in DyNetKAT

FatTree Topologies



Safety / Reachability in DyNetKAT

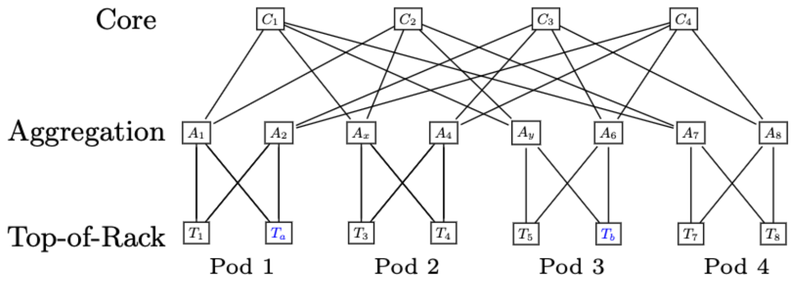
FatTree Topologies

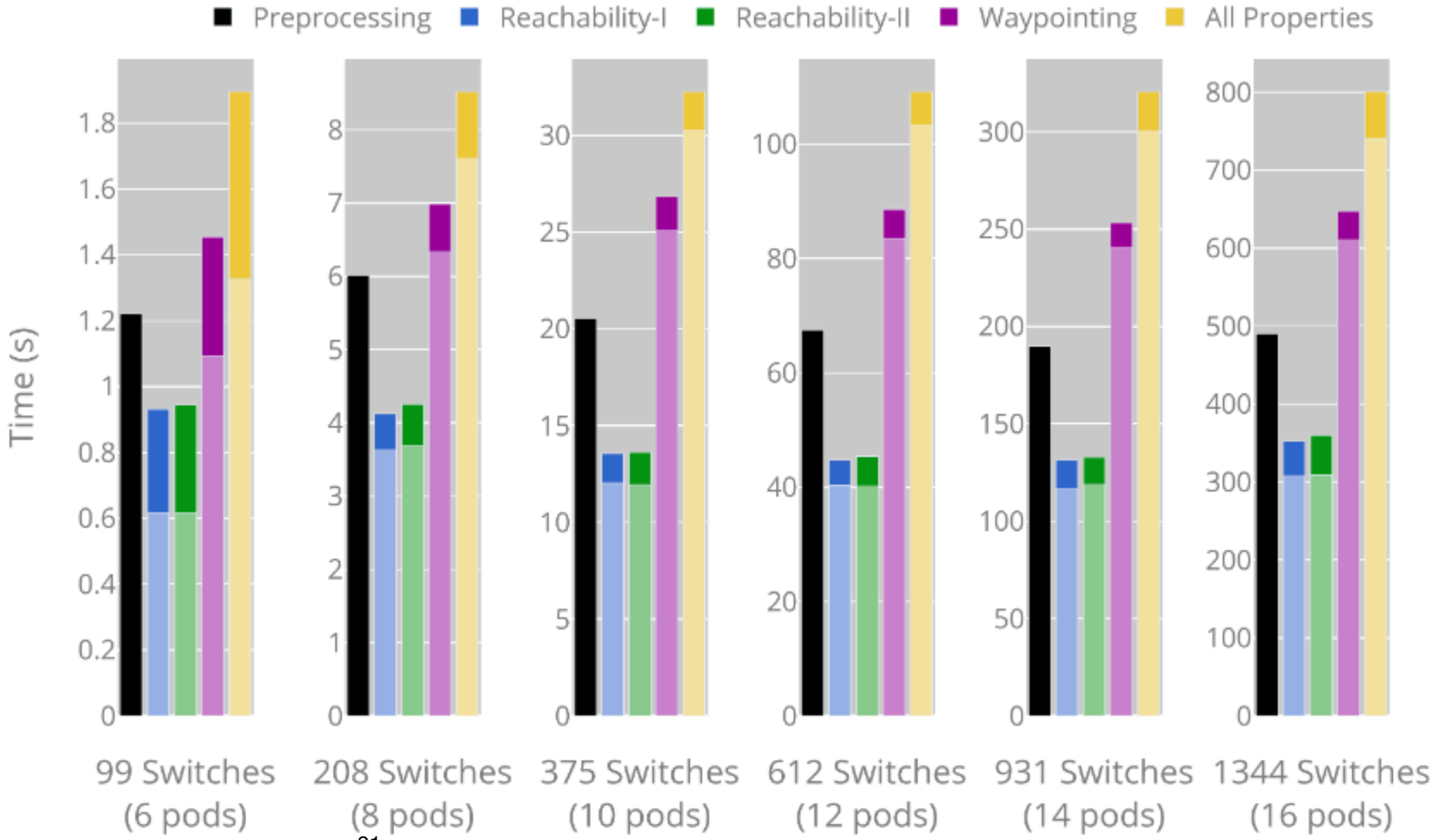


- Reachability I: While controller updates, allow non-SSH from Ta to Tb
- Reachability II: While controller updates, block SSH from Ta to Tb
- Waypointing: After controller updates, Ay is a waypoint between Ta and Tb

Safety / Reachability in DyNetKAT

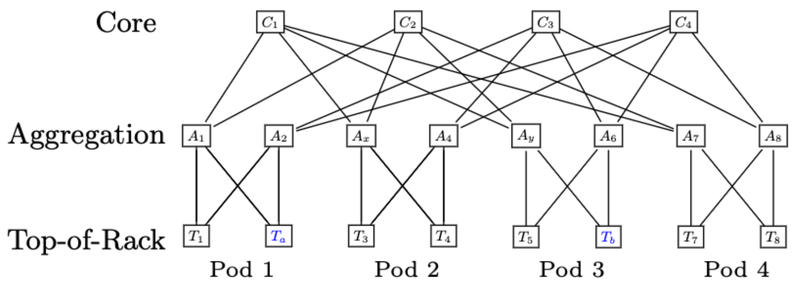
- Safety Framework Implemented in Maude: github.com/hcantunc/DyNetiKAT

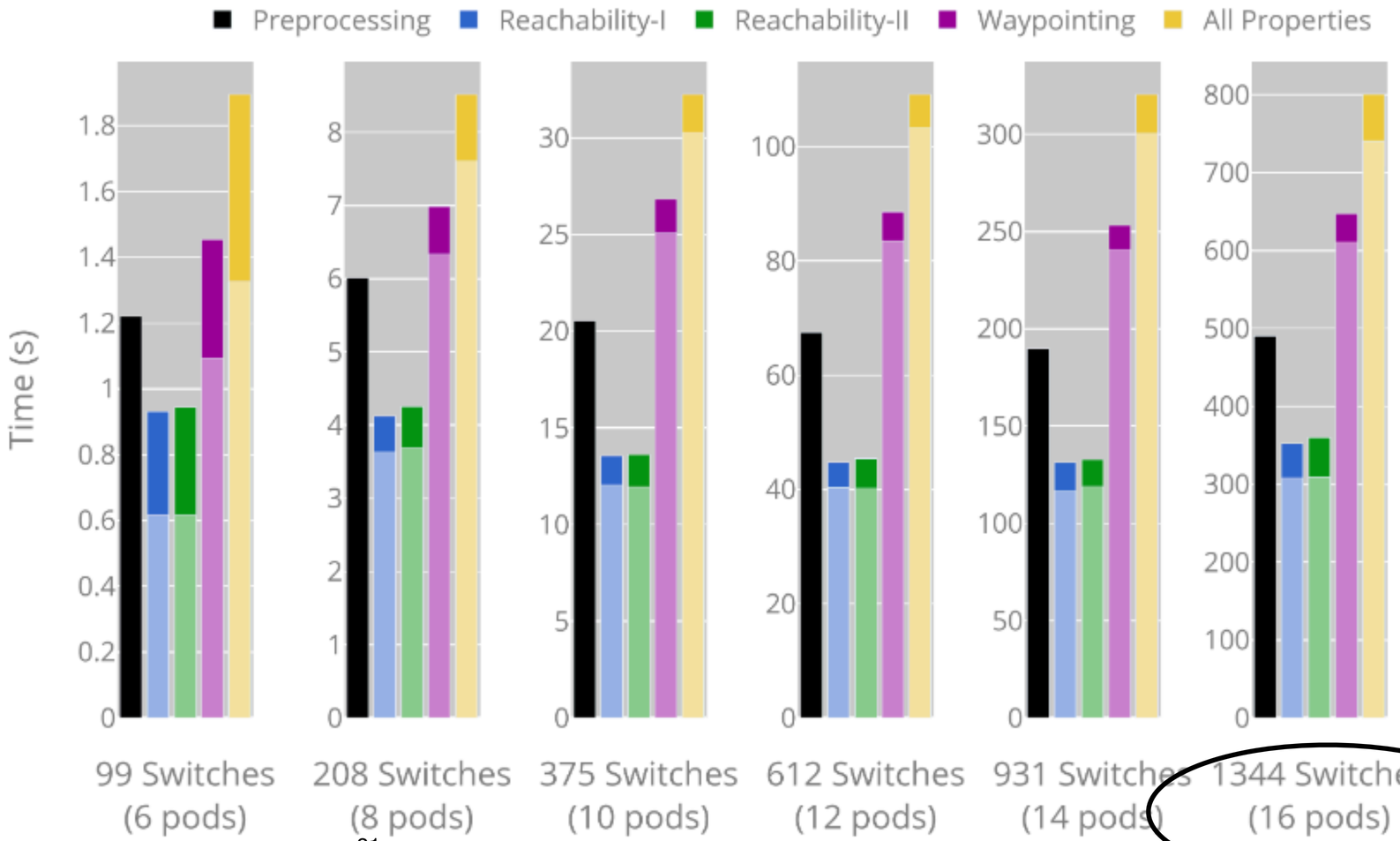




Safety / Reachability in DyNetKAT

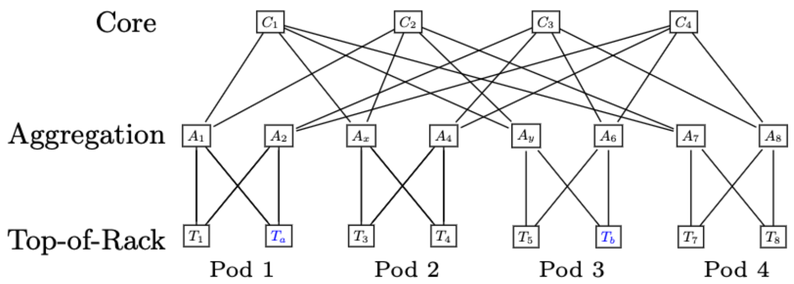
- Safety Framework Implemented in Maude: github.com/hcantunc/DyNetiKAT

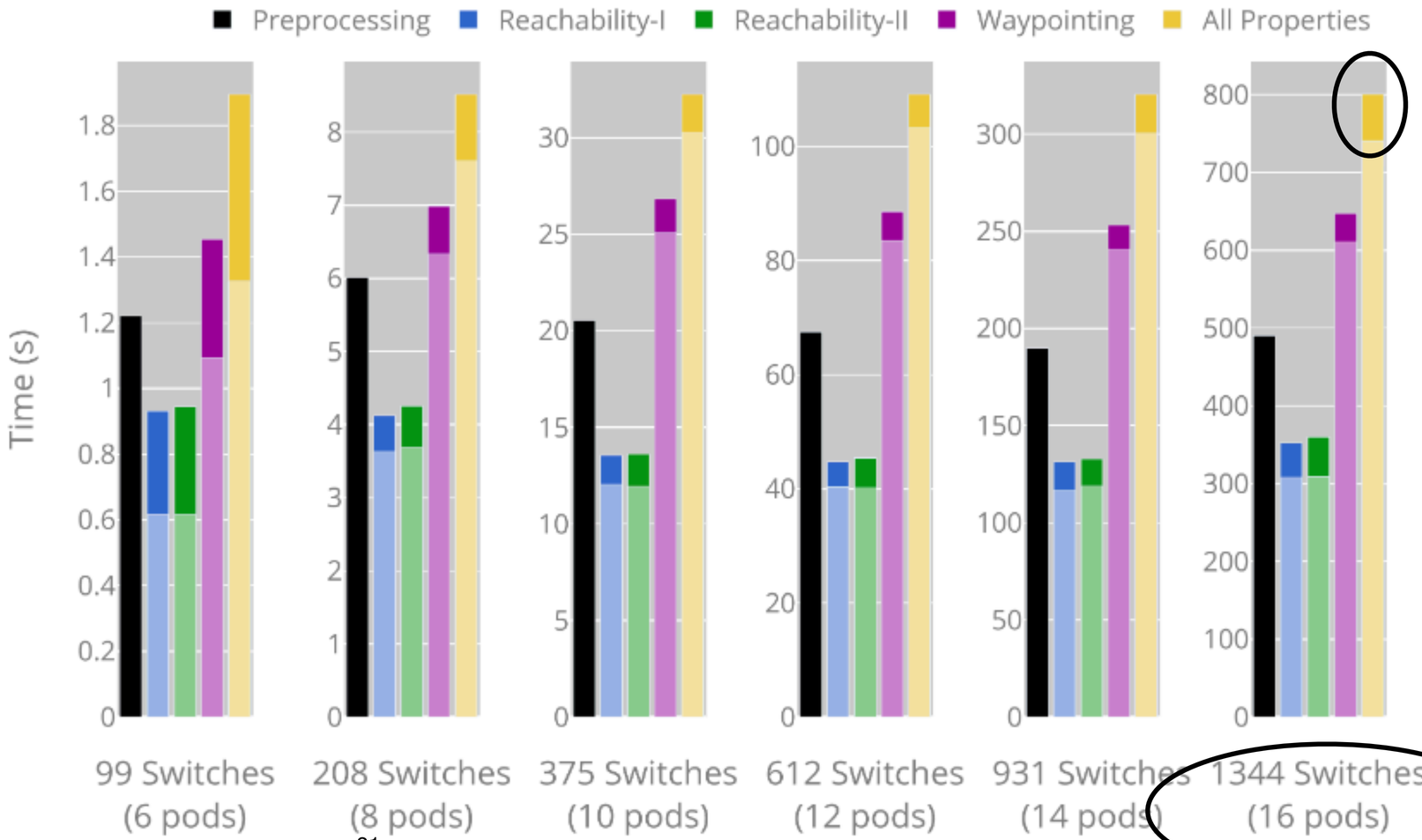




Safety / Reachability in DyNetKAT

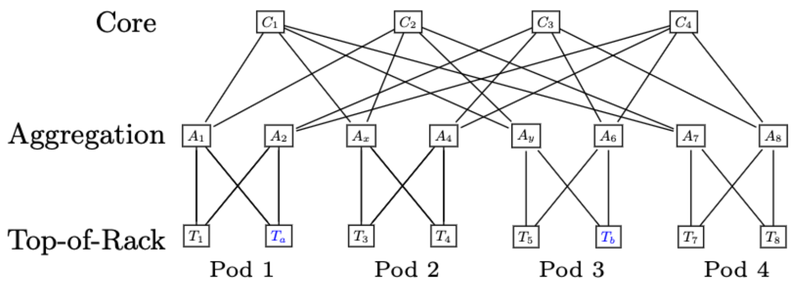
- Safety Framework Implemented in Maude: github.com/hcantunc/DyNetiKAT

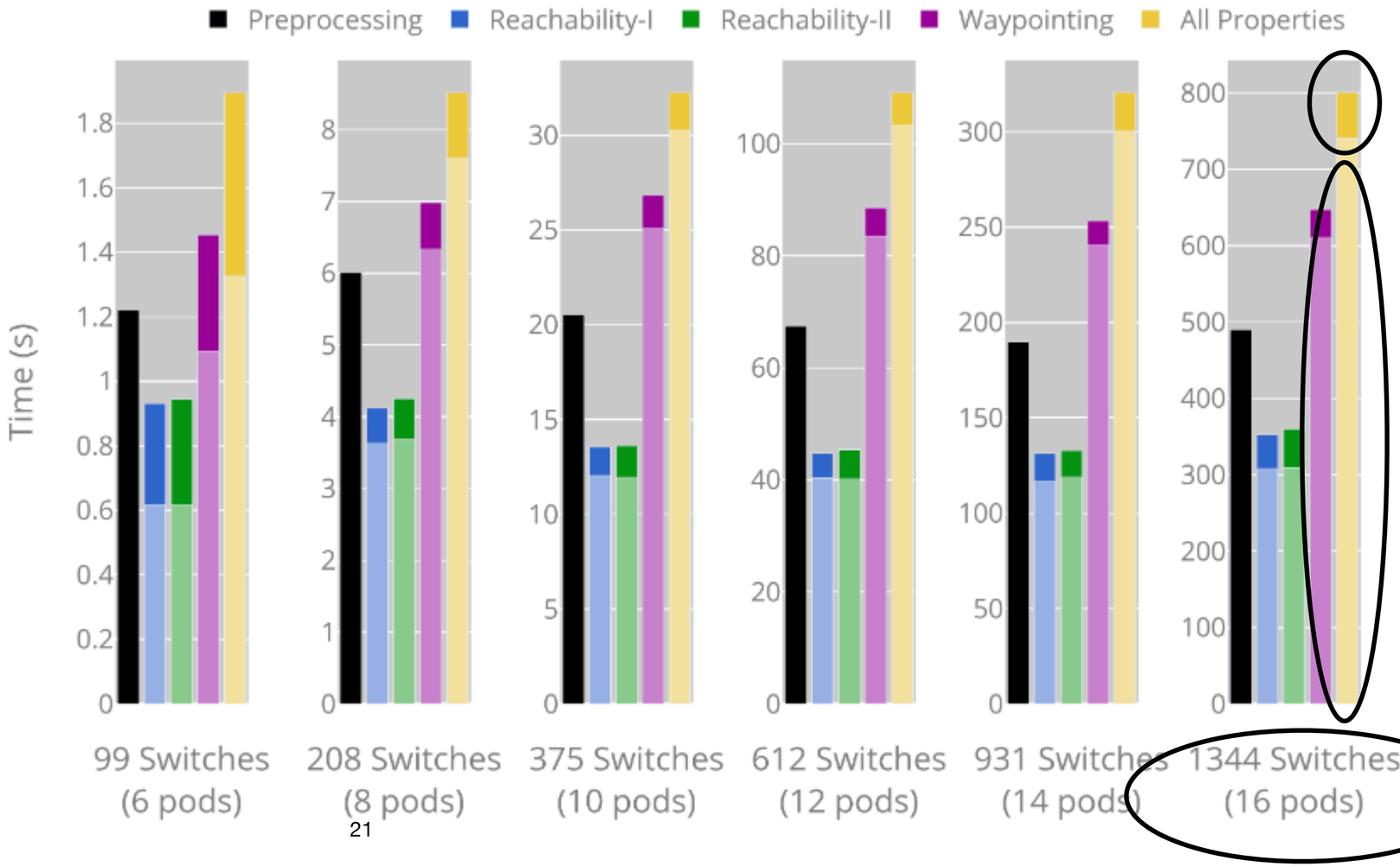




Safety / Reachability in DyNetKAT

- Safety Framework Implemented in Maude: github.com/hcantunc/DyNetiKAT



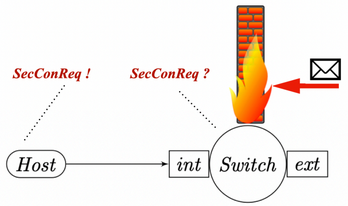


Conclusions

Dynamic NetKAT (DyNetKAT)

Supports…

- Dynamic & stateful behaviour
- Synchronisation of
  - Controllers trigger network reconfigurations
  - Switches accept reconfigurations & update flow tables



Stateful Firewall

DyNetKAT Language

A Process Algebraic Approach

- Syntax

$$N ::= \mathrm{NetKAT}^{-\mathbf{dup}}$$

$$D ::= \bot \mid N ; D \mid x?N ; D \mid x!N ; D \mid D \mid\mid D \mid D \oplus D \mid X$$
$$X \triangleq D$$

- (Small-Step Operational) Semantics $(p, H_0, H_1) \xrightarrow{\gamma} (p', H_0', H_1')$

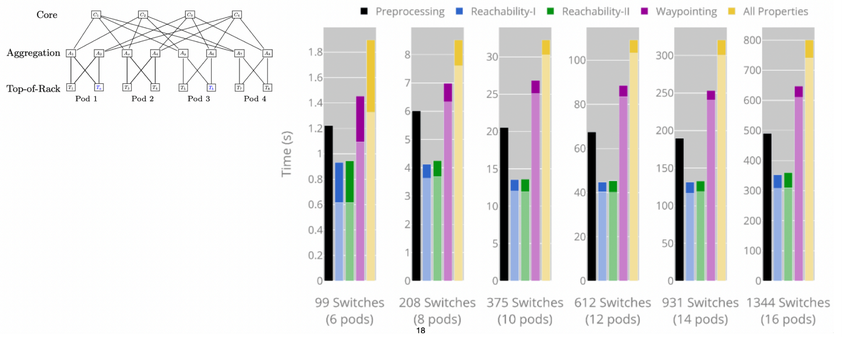
$$\frac{\sigma' \in [\![p]\!](\sigma :: \langle\rangle)}{(p; q, \sigma :: H, H') \xrightarrow{(\sigma,\sigma')} (q, H, \sigma' :: H')} \qquad \frac{(p, H_0, H_0') \xrightarrow{\gamma} (p', H_1, H_1')}{(p \oplus q, H_0, H_0') \xrightarrow{\gamma} (p', H_1, H_1')}$$

$$\frac{(q, H, H') \xrightarrow{x\clubsuit p} (q', H, H') \quad (s, H, H') \xrightarrow{x\spadesuit p} (s', H, H')}{(q||s, H, H') \xrightarrow{\mathbf{rcfg(x,p)}} (q'||s', H, H')} \quad \begin{matrix}\clubsuit =? \ \ \spadesuit =! \\ \text{or} \\ \clubsuit =! \ \ \spadesuit =?\end{matrix}$$

13

Safety / Reachability in DyNetKAT

- Safety Framework Implemented in Maude: github.com/hcantunc/DyNetiKAT



18

- Related: Concurrent NetKAT [Wagemaker & al., ESOP'22]
  - Focus on the actual flow of (multiple, concurrent) network packets

Thanks!