Formal Methods for Mesh Protocols

Peter Höfner

18/01/2012

Australian Government
Department of Broadband, Communications and the Digital Economy
Australian Research Council

NICTA Funding and Supporting Members and Partners

Project Structure

- Formal Methods for Routing Protocols of Wireless Mesh Networks (WMNs)

- Part of "**Adaptive and Verified Wireless Mesh Protocols (Mesh Protocols)**"
- Across research groups
  - close cooperation with Network Research Group
- Across research labs
  - NRL, QRL

- start November 2010

Project Team

- Formal Methods for WMNs @ SSRG
  - Rob van Glabbeek (Leader, 70%)
  - Peter Höfner (100%)
  - Ansgar Fehnker (until end 2011, 20%)
- Networks Research Group
  - Annabelle McIver
  - Marius Portmann
  - Wee Lum Tan



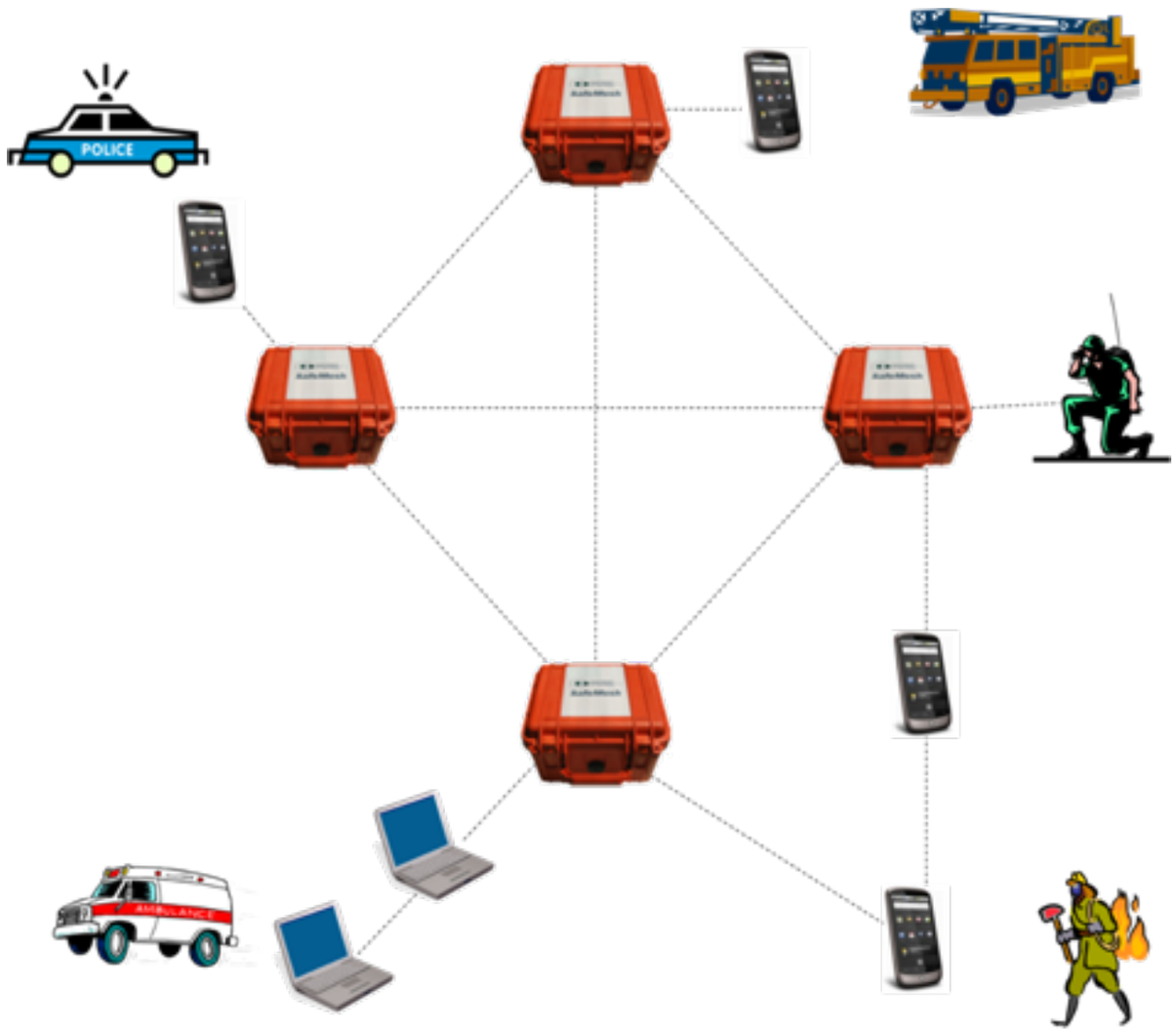



- ~2 FTEs (7 for entire project)

Background

- Wireless Mesh Networks (WMNs) / Mobile Ad hoc Networks (MANETs)
  - key features: mobility, dynamic topology, wireless multi-hop backhaul
  - quick and low cost deployment

- Applications
  - public safety (e.g. CCTV)
  - emergency response, disaster recovery
  - mining
  - smart grid
  - ...



Problems

- Mesh routing protocols
  - challenging environment
    (mobility, dynamic topology, link quality, communication...)
  - design is difficult
    - complex
    - error prone
  - limited reliability and performance (confirmed by industry and end users)

- Standards (IETF RFCs) are not precise
  - written in English
  - ambiguous
  - no formal specification or reasoning

- Traditional evaluation techniques: simulation, test-bed experiments
  - expensive, time-consuming
  - limited to (a small number of) specific scenarios
  - protocol errors still found even after years of intensive evaluation

Internet Engineering Task Force (IETF)

- “Formal languages are useful tools for specifying parts of protocols. However, as of today, there exists no well-known language that is able to capture the full syntax and semantics of reasonably rich IETF protocols.“
  [IETF]

- IETF’s requirements (for formal languages)
  - relatively easy to extract code
  - complete specification
  - implementation independent

Research Aims

- Provide complete and practical formal methods for mesh protocols
  - expressive power
    (mobility, dynamic topology, types of communication, link failures...)
  - usable / intuitive
  - description language + proof methodology

- Specification, verification and analysis of mesh protocols
  - formalise relevant standard protocols
  - analyse the protocols w.r.t. key requirements, e.g. loop freedom

- Development of improved protocol(s)
  - assured protocol correctness
  - improve reliability
  - improve performance

Challenges

- Grand-challenges it contributes to
  - trustworthiness:
    fundamental properties have to be guaranteed
  - real systems:
    wireless mesh networks and protocols are widely used by industry and end users

- Technical research challenges
  - formal description language for WMN routing protocols
  - proof methodology

Research Approach

- Unique interdisciplinary collaboration
  - across research groups, across research labs (networking & formal methods)
  - helps to stick to reality (no over-abstraction, no toy examples)

- Formal method approaches
  - process algebra (process calculus)
    - formal language for concurrent systems including communication
    - basic theory well established, but not adapted to WMNs
  - model checking
    - automated method to analyse concurrent systems
    - tool support, but restricted to specific scenarios
  - routing algebra
    - use (linear) algebra to model behaviour
    - easy to automate, but not adapted to WMNs and relatively new

Research Outcomes (Process Algebra)

Table 1 Excerpt of AWN spec. for AODV

$\texttt{AODV(ip,sn,rt,rreqs,store)} \stackrel{def}{=}$

1. /*depending on the message on top of the message queue, the node calls different processes*/
2. ...
3. $[\ \texttt{msg} = \texttt{rreq(hops, rreqid, dip, dsn, oip, osn, sip)} \wedge \texttt{(oip, rreqid)} \in \texttt{rreqs}\ ]$
4. /*silently ignore RREQ, i.e. do nothing, except update the entry for the sender*/
5. $[\![\texttt{rt} := \texttt{update(rt, (sip, 0, val, 1, sip))}]\!]$. /*update the route to sip*/
6. AODV(ip,sn,rt,rreqs,store)
7. $+\ [\ \texttt{msg} = \texttt{rreq(hops, rreqid, dip, dsn, oip, osn, sip)} \wedge \texttt{(oip, rreqid)} \notin \texttt{rreqs}) \wedge \texttt{dip} = \texttt{ip}\ ]$
8. /*answer the RREQ with a RREP*/
9. $[\![\texttt{rt} := \texttt{update(rt, (oip, osn, val, hops} + 1, \texttt{sip}))]\!]$ /*update the routing table*/
10. $[\![\texttt{rreqs} := \texttt{rreqs} \cup \{\texttt{(oip, rreqid)}\}]\!]$ /*update the array of already seen RREQ*/
11. $[\![\texttt{sn} := \max\texttt{(sn, dsn)}]\!]$ /*update the sqn of ip*/
12. $[\![\texttt{rt} := \texttt{update(rt, (sip, 0, val, 1, sip))}]\!]$ /*update the route to sip*/
13. **unicast**(nhop(rt,oip),rrep(0,dip,sn,oip,ip)) .
14. AODV(ip,sn,rt,rreqs,store)
15. $+\ [\ \texttt{msg} = \texttt{rreq(hops, rreqid, dip, dsn, oip, osn, sip)} \wedge \texttt{(oip, rreqid)} \notin \texttt{rreqs}) \wedge \texttt{dip} \neq \texttt{ip} \wedge$
    $(\texttt{dip} \notin \texttt{vD(rt)} \vee \texttt{sqn(rt,dip)} < \texttt{dsn} \vee \texttt{sqnf(rt,dip)} = \texttt{unk})\ ]$
16. /*forward RREQ*/
17. $[\![\texttt{rt} := \texttt{update(rt, (oip, osn, val, hops} + 1, \texttt{sip}))]\!]$ /*update routing table*/
18. $[\![\texttt{rreqs} := \texttt{rreqs} \cup \{\texttt{(oip, rreqid)}\}]\!]$ /*update the array of already seen RREQ*/
19. $[\![\texttt{rt} := \texttt{update(rt, (sip, 0, val, 1, sip))}]\!]$ /*update the route to the sender*/
20. **broadcast**(rreq(hops + 1,rreqid,dip,max(sqn(rt, dip), dsn),oip,osn,ip)) .
21. AODV(ip,sn,rt,rreqs,store)
22. $+\ [\ \texttt{rreq(hops, rreqid, dip, dsn, oip, osn, sip)} \wedge \ldots\ ]$
23. ...

Research Outcomes (Process Algebra)

- Algebra for Wireless Networks (AWN)
  - novel treatment of data structures, conditional unicast und local broadcast
  - formalisation and (dis)proof of key aspects of routing protocols, e.g. loop freedom, packet delivery
- Case study
  - Ad-hoc On Demand Distance Vector Protocol (AODV)
    - model the standard
    - **first formal and complete proof of loop freedom**
    - analysed more key properties such as packet delivery
  - Analysed variants/interpretations of AODV
    - all reasonable interpretations of the standard (RFC) analysed

- Publications

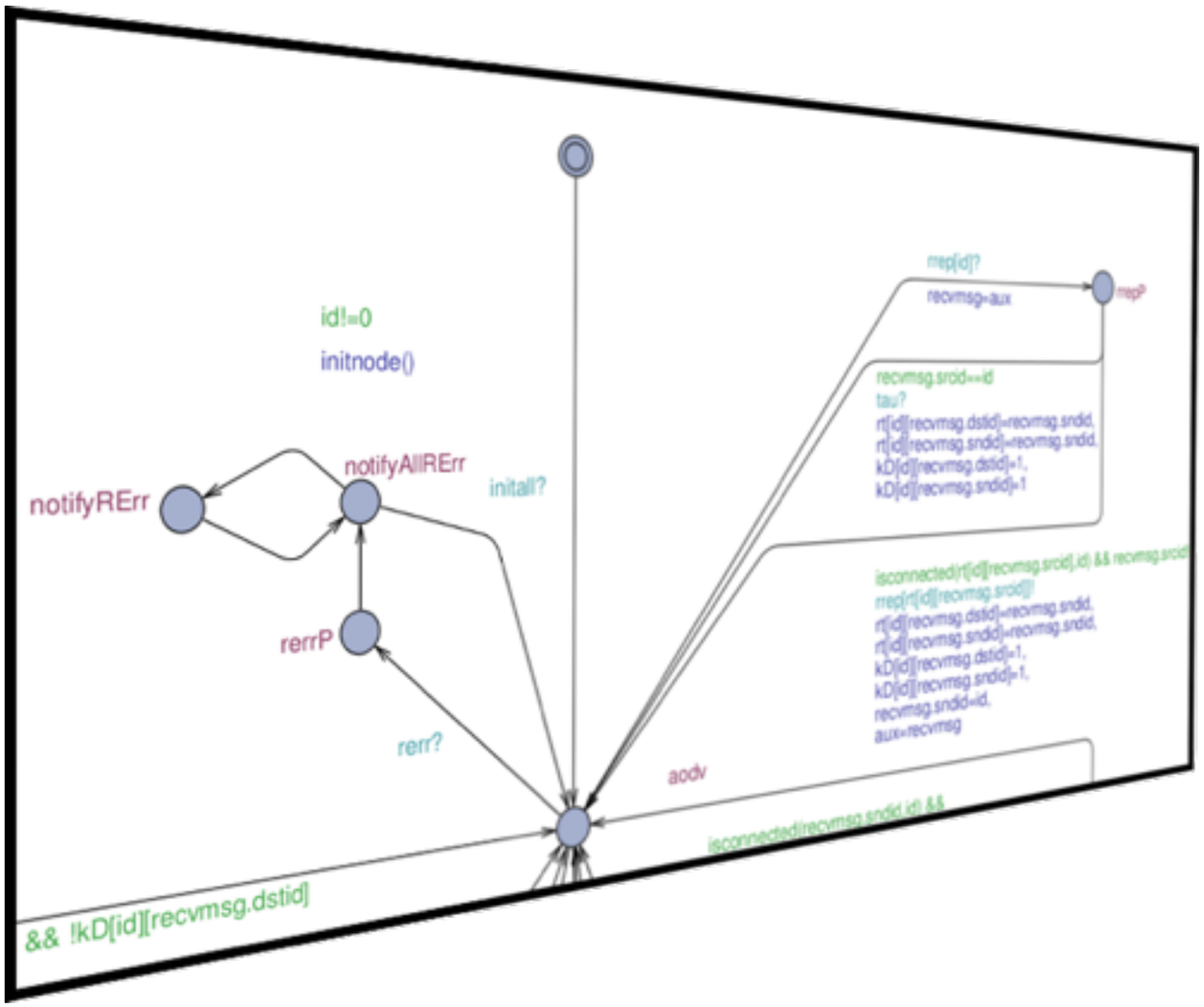
[1] A Process Algebra for Wireless Mesh Networks. In European Symposium on Programming (ESOP 2012), Lecture Notes in Computer Science, Springer, 2012. (to appear)
[2] A Process Algebra for Wireless Mesh Networks used for Modelling, Verifying and Analysing AODV. Technical report 5513, NICTA, 2012

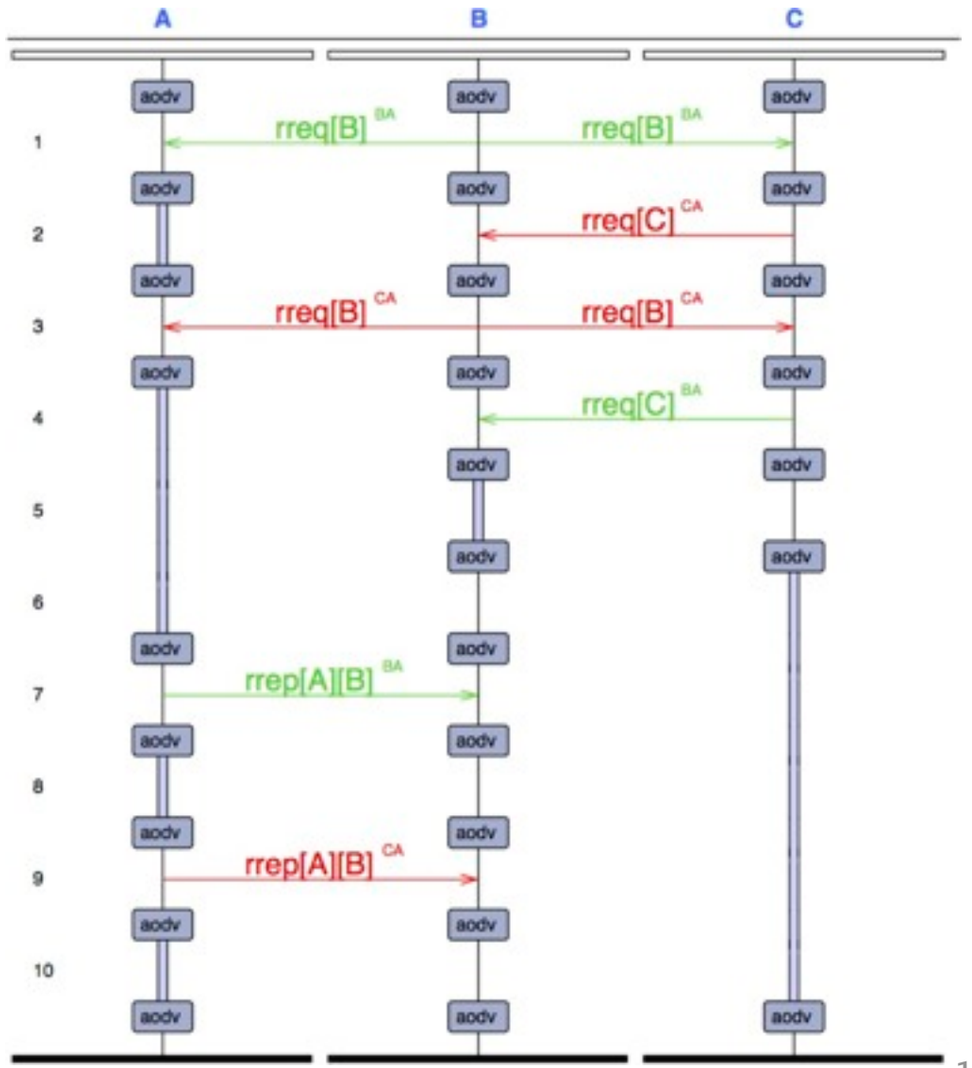
Research Outcomes (Process Algebra)

- Advantages
  - language supports key primitives for WMN routing protocols
    - mobility, dynamic topology, broadcast, unicast, ...
  - proof methodology
    - reasoning about key protocol properties
  - applicable to complex and rich protocols
  - easy to read, close to programming languages

- On-going and future work
  - support for time and probability
  - proof automatisation with Isabelle/HOL

Research Outcomes (Model Checking)





Research Outcomes (Model Checking)

- Model Checking allows
  - evaluation of WMNs routing protocols
  - finding of problematic and undesirable behaviour
  - exhaustive search over different scenarios
  - adaptation to variants
- Translation from AWN to UPPAAL
  - AWN: process algebra for wireless networks
  - UPPAAL: off-the-shelf model checker from Uppsala university
  - combines both approaches
- Case study
  - Ad-hoc On Demand Distance Vector Protocol (AODV)
  - checked 17400 models (topologies)
  - Analysed protocol limitations
- Publications

[1] Automated Analysis of AODV in UPPAAL. In Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2012), Lecture Notes in Computer Science, Springer, 2012. (to appear)
[2] Modelling and Analysis of AODV in UPPAAL, Workshop on Rigorous Protocol Engineering (W-RiPE 2011).

Research Outcomes (Model Checking)

- Advantages
  - model generated from process algebra
  - inherited model guarantees accuracy/correctness
  - able to find problems in specification before proof attempts
  - quick automatic exhaustive search

- On-going and future work
  - time, probability

Research Outcomes (Routing Algebra)

$$
\begin{array}{c} \\ A \\ B \\ C \\ D \\ \vdots \end{array}
\begin{array}{c}
\begin{array}{cccc} A & B & C & D \end{array} \\
\left(\begin{array}{cccc}
(\_,0) & (B,1) & (B,2) & (\_,\infty) \\
(A,1) & (\_,0) & (C,1) & (\_,\infty) \\
(\_,\infty) & (B,1) & (\_,0) & (\_,\infty) \\
(\_,\infty) & (\_,\infty) & (\_,\infty) & (\_,0)
\end{array}\right) \\
\begin{array}{cc} \vdots & \end{array}
\end{array}
\begin{array}{l} \\ \\ \\ \\ \\ \text{routing table of } A \end{array}
$$

$$
\begin{array}{c} \\ \\ \\ \\ \\ \text{“routes” to } B \end{array}
\begin{array}{c}
\begin{array}{c} \\ \\ \\ \\ \vdots \end{array} \\
\begin{array}{c} \\ \\ \\ \\ \end{array}
\end{array}
\begin{array}{c}
\begin{array}{c} \\ \\ \\ \\ \end{array} \\
\begin{array}{c} \\ \\ \\ \\ \ddots \end{array}
\end{array}
$$

$$
\left(\begin{array}{cccc}
(\_,0) & (B,1) & (C,1) & (\_,\infty) \\
(A,1) & (\_,0) & (\_,\infty) & (D,1) \\
(A,1) & (\_,\infty) & (\_,0) & (D,1) \\
(\_,\infty) & (B,1) & (C,1) & (\_,0)
\end{array}\right)
\bullet
\left(\begin{array}{cccc}
(\_,0) & (\_,\infty) & (\_,\infty) & (\_,\infty) \\
(\_,\infty) & (\_,\infty) & (\_,\infty) & (\_,\infty) \\
(\_,\infty) & (\_,\infty) & (\_,\infty) & (\_,\infty) \\
(\_,\infty) & (\_,\infty) & (\_,\infty) & (\_,\infty)
\end{array}\right)
\bullet
\left(\begin{array}{cccc}
(\_,0) & (B,1) & (\_,\infty) & (\_,\infty) \\
(\mathbf{D},\mathbf{3}) & (\_,0) & (\_,\infty) & (\_,\infty) \\
(A,1) & (\_,\infty) & (\_,0) & (D,1) \\
(C,2) & (\_,\infty) & (C,1) & (\_,0)
\end{array}\right)
$$

topology sender routing table

$$
= \left(\begin{array}{cccc}
(\_,0) & (B,1) & (\_,\infty) & (\_,\infty) \\
(\mathbf{A},\mathbf{1}) & (\_,0) & (\_,\infty) & (\_,\infty) \\
(A,1) & (\_,\infty) & (\_,0) & (D,1) \\
(C,2) & (\_,\infty) & (C,1) & (\_,0)
\end{array}\right)
$$

updated routing table

Research Outcomes (Routing Algebra)

- New (high-level) approach
- Algebra for routing protocols (not necessarily limited to WMNs)
  - use algebraic structures, such as matrices
  - similar approaches used for other protocols
    - Dijkstra's shortest path, BGP...
    - cross-reasoning
      understanding similarities of different types of protocols
- Model main aspects of WMN protocols such as message sending
  - can be easily used for automatic verification and simulation

- Publications

[1] Towards an Algebra of Routing Tables. In Relational and Algebraic Methods in Computer Science (RAMICS 2011), Lecture Notes in Computer Science 6663, 212-229, Springer, 2011.

Research Outcomes (Routing Algebra)

- Advantages
  - well established theory such as linear algebra
  - high level of automatisation
    - numerical mathematical software (e.g. Matlab, Mathematica)
    - theorem proving systems (e.g. Prover9 or Coq)

- On-going and future work
  - model all aspects of WMN protocols
  - comprehensive case study

Key Research Outcomes (Summary)

- New languages and proof methodologies
  - process algebra AWN
  - routing algebra
- Modelling of AODV
  - process algebra: complete and detailed model (no time)
  - model checking: encoding of AWN specification
  - routing algebra: modelled parts of AODV
- Analysing/Verifying AODV
  - process algebra: proof methodology
    first formal proof of loop freedom
  - model checking: automatic finding of problematic behaviour
    no packet delivery
  - analysed variants of AODV

Key Research Outcomes (Future)

- Extend languages
  - process algebra, model checking: time, probability
  - routing algebra: complete expressive power
- Proof automatisation
  - process algebra: Isabelle/HOL
  - routing algebra: Prover9
- Specification vs. Implementation
  - check real implementations against (correct) specification
  - code generation
- Application of developed formal methods to new types of protocols
  - adaptive, modular protocols for WMNs
  - “new generation of protocol”

Links / Engagement

- Within software systems research group
  - proof automatisation for process algebra (Isabelle/HOL)
- Across research groups
  - network research group
  - understand the reality
  - model, analyse and verify real protocols, not toy examples
- Academic cooperation
  - Cambridge, Stanford, Stony Brook ...
- Industry partner
  - Firetide (market leader for WMNs for public safety applications)
  - mostly cooperating with network group.

Global research competitive position

Research Group	Key staff	Scale of effort	Point of difference
NICTA Mesh protocols	Rob van Glabbeek Peter Höfner	2 researchers	rigorous formal methods application to relevant protocols
Cambridge University Metarouting	Timothy G. Griffin	4 researchers and students	focus on analysis of internet protocols (BGP)
AT&T Labs Research	Pamela Zave	numbers vary	focus on higher-level protocols (e.g. SIP)
Stony Brook University	C.R. Ramakrishnan	3 researchers	no close collaboration with network engineers
University of Pennsylvania NetDB@Penn	Boon Thau Loo	2 researchers and 8 PhD students	distributed systems, analysis of BGP, no wireless
Radboud University Model-Based System Develop.,	Frits Vaandrager	4 researchers and students	no focus on networks, no close collaboration with network engineers

Selected Publications

Title	Conference	Year
A Process Algebra for Wireless Mesh Networks	European Symposium on Programming (ESOP 12)	2012
Automated Analysis of AODV using AODV	Tools and Algorithms for the Construction and Analysis of Systems (TACAS 12)	2012
A Process Algebra for Wireless Mesh Networks used for Modelling Verifying and Analysing AODV.	Technical Report, NICTA	2012
Modelling and Analysis of AODV in UPPAAL	Workshop on Rigorous Protocol Engineering (W-Ripe 11)	2011
Towards an Algebra of Routing Tables	Relational and Algebraic Methods in Computer Science (RAMiCS 11)	2011

Path to Impact

- change approach to the development and specification of WMNs protocols
- set up new standards for protocol verification w.r.t. WMNs
- via industry partner (Firetide),

Questions, Comments ?