EGYPTE

Application-Level Multicast Transmission Techniques Over The Internet

Ayman EL-SAYED

Projet Planète; INRIA Rhône-Alpes

Supervisor

Dr. Vincent ROCA

Director of thesis

Prof. Andrzej DUDA

March 8th, 2004

Outline of the presentation

1. Introduction
2. Our proposal: Host Based Multicast (HBM)
3. Evaluation and Improvements
    1. List of items addressed
    2. Improving the robustness
    3. An example of use: VPRN
4. Discussion, Conclusion, and Future Work

Part 1

Introduction

Introduction to application-level multicast

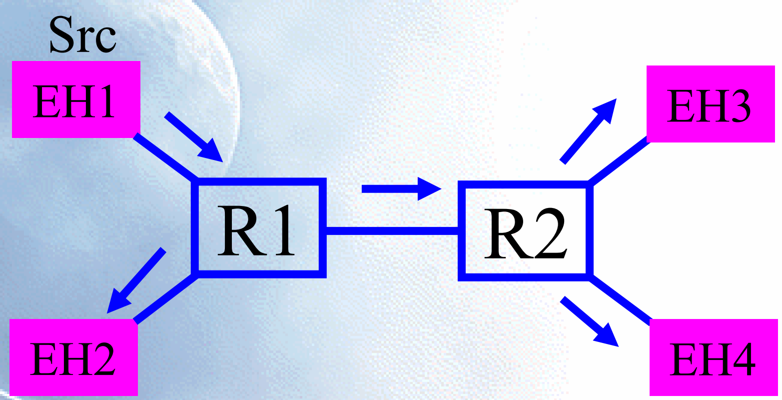
- Motivations
  - multicast routing is not available everywhere
- Application-Level Multicast
  - shifts the multicast support from core routers to end-systems
  - automatic creation of an overlay topology
    - **use unicast between two end-systems**
    - **the underlying physical topology is hidden**
    - **try to find an ``optimal'' overlay topology (e.g. a spanning tree with minimal global cost)**

Introduction … (cont')

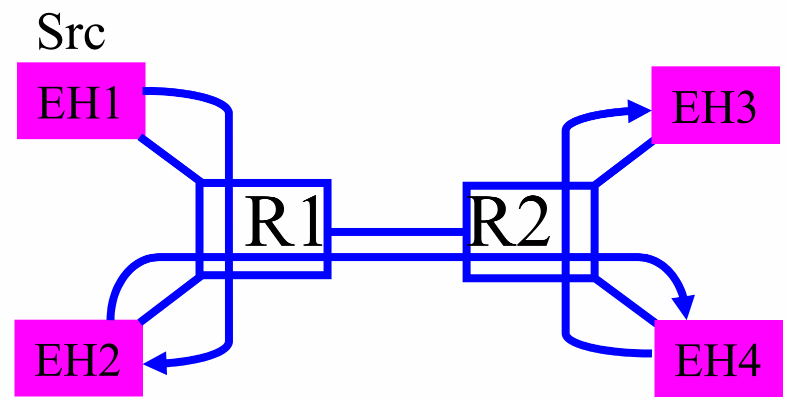
- Application-Level Multicast (cont')
  - Requires a dynamic overlay topology update
    - **because the network conditions dynamically change**
      - try to stay as close as possible to an optimal overlay topology
      - can be regarded as "static QoS routing"
    - **because the group is dynamic, the topology quickly becomes sub-optimal**
      - after a node departure/failure, a quick and dirty local solution is found to avoid topology partition
      - when a node arrives, he joins the current topology as a leaf to create as little perturbation as possible
  - We need to periodically update the whole topology!

Introduction … (cont')

- Application-Level Multicast (cont')
  - Example



With multicast routing



With Application-level multicast

  - Topology building algorithm can be
    - **Centralized (HBM, ALMI …)**
    - **Distributed (NARADA, Overcast, Nice, TBCP …)**

Part 2

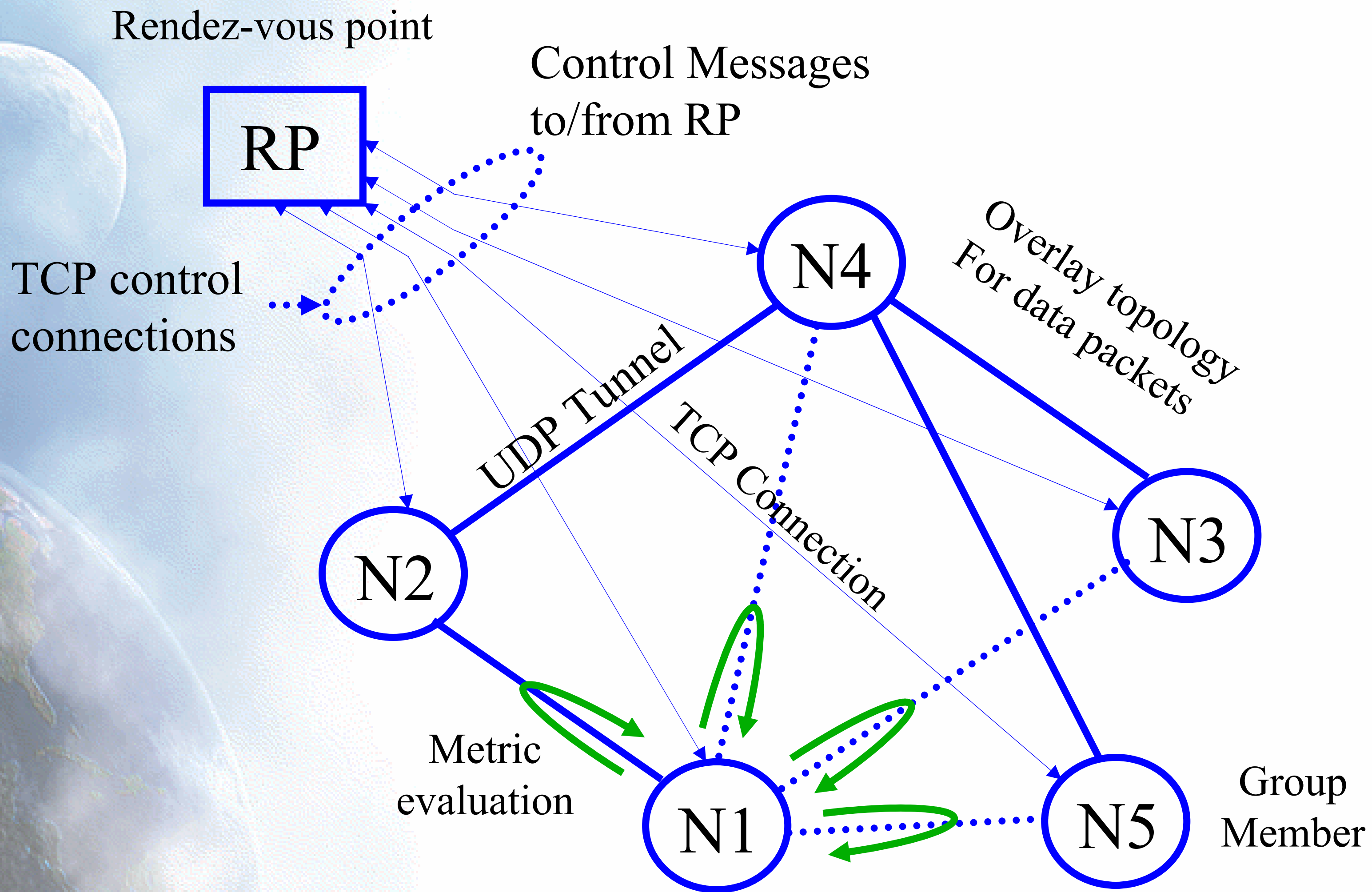
Our proposal:
Host Based Multicast
(HBM)

Our HBM Proposal

- Centralized approach: everything is under control by RP
- The RP has a complete knowledge of group membership/communication costs.
- Take into account several metrics (RTT, loss, …) when creating the virtual topology
- Data flows on the virtual topology (no RP implication)
- Each node periodically evaluates metrics between itself and other nodes and informs the RP
- Likewise RP periodically refresh the topology and inform all nodes

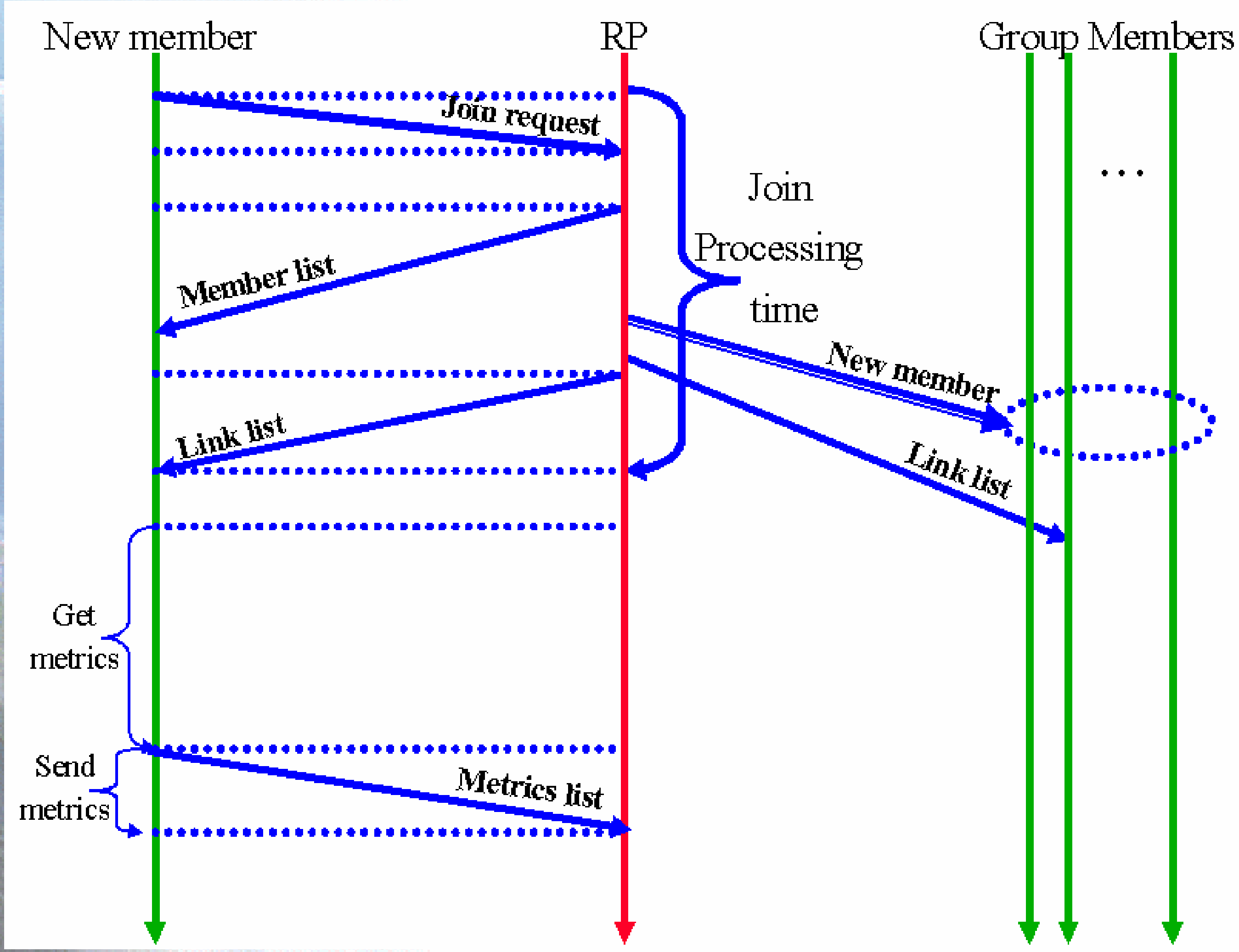
Our HBM Proposal … (cont')

- HBM Control Connections



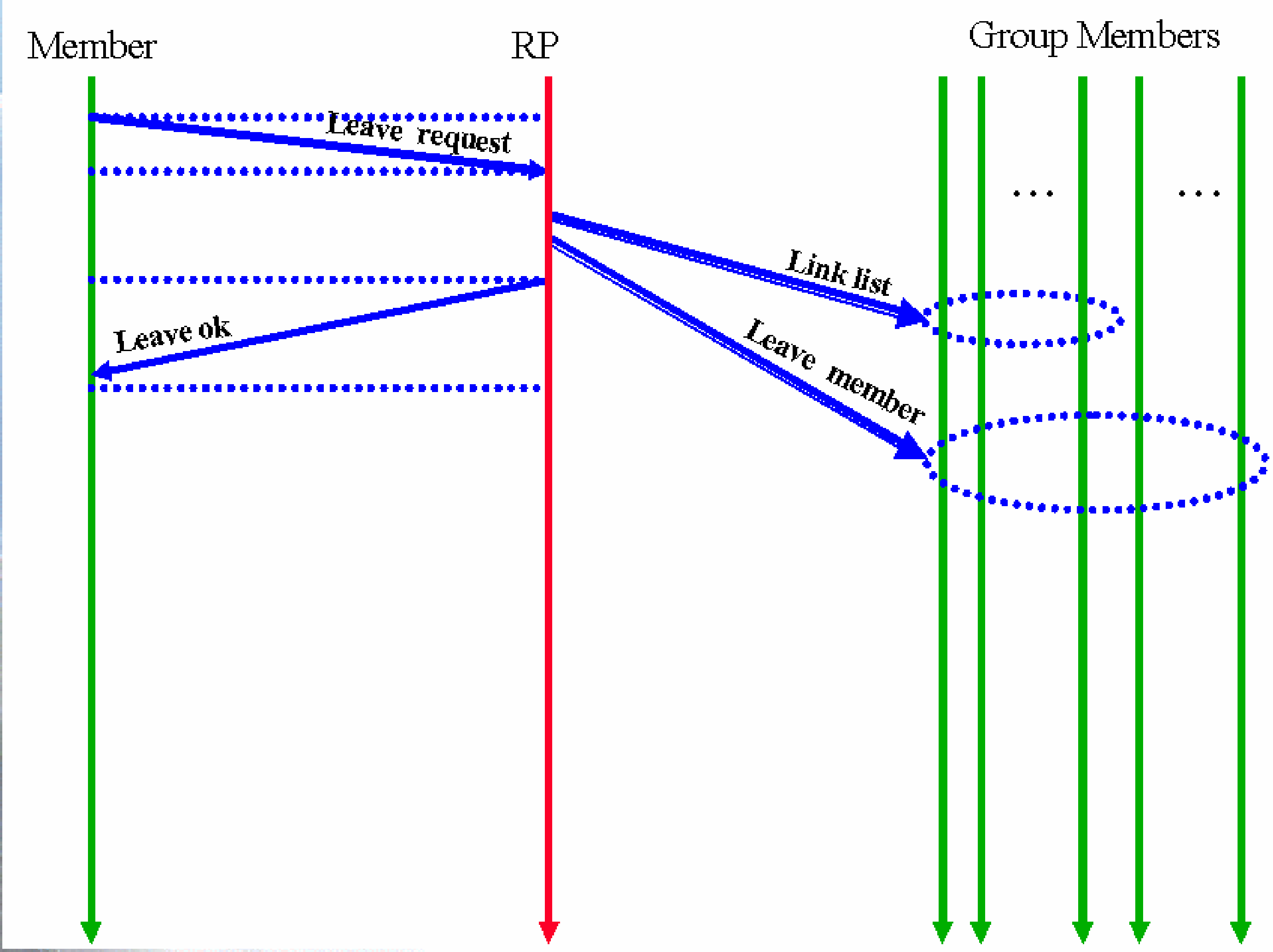
Our HBM Proposal …(cont')

- Joining a group



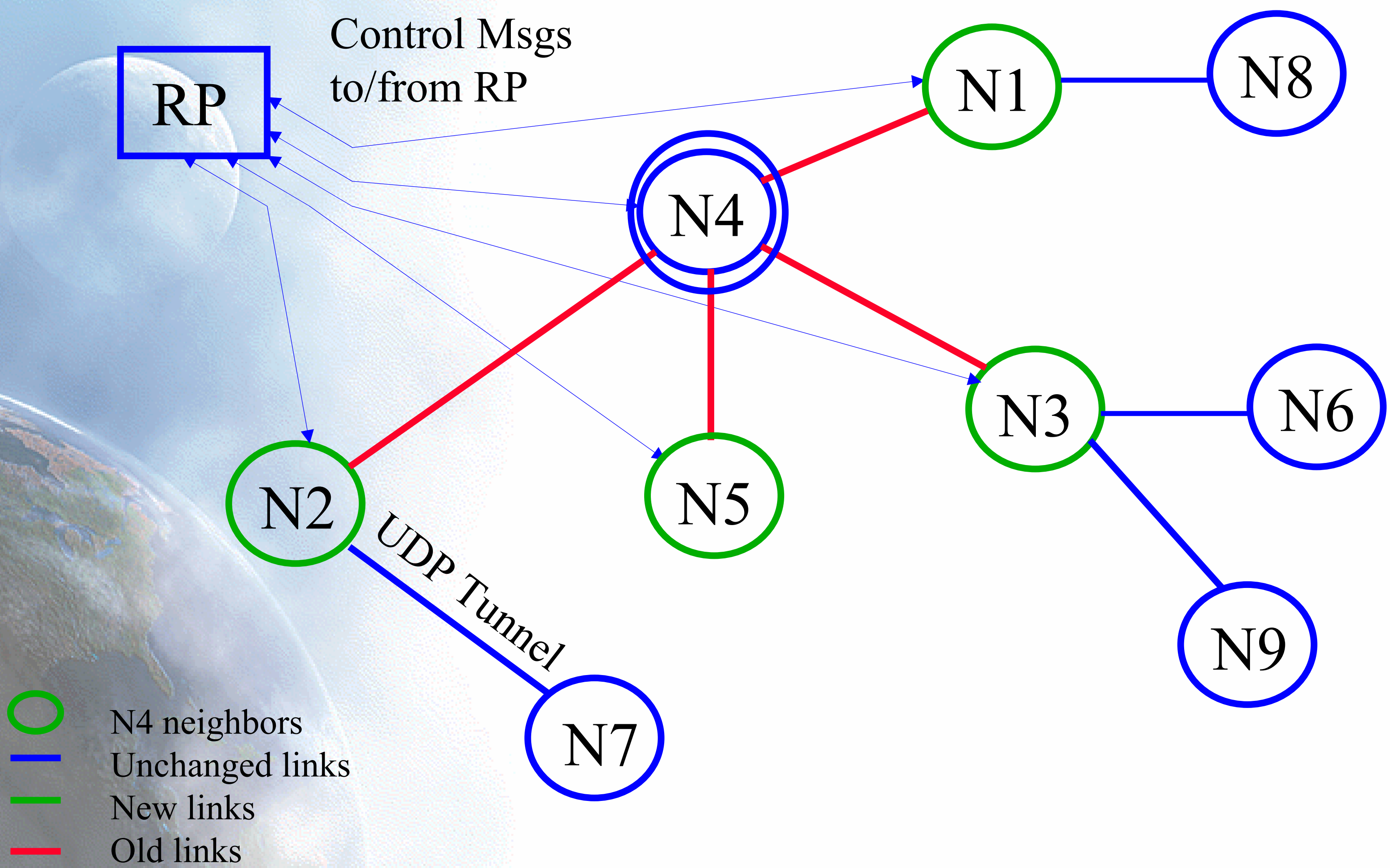
Our HBM Proposal …(cont')

- Leaving a group



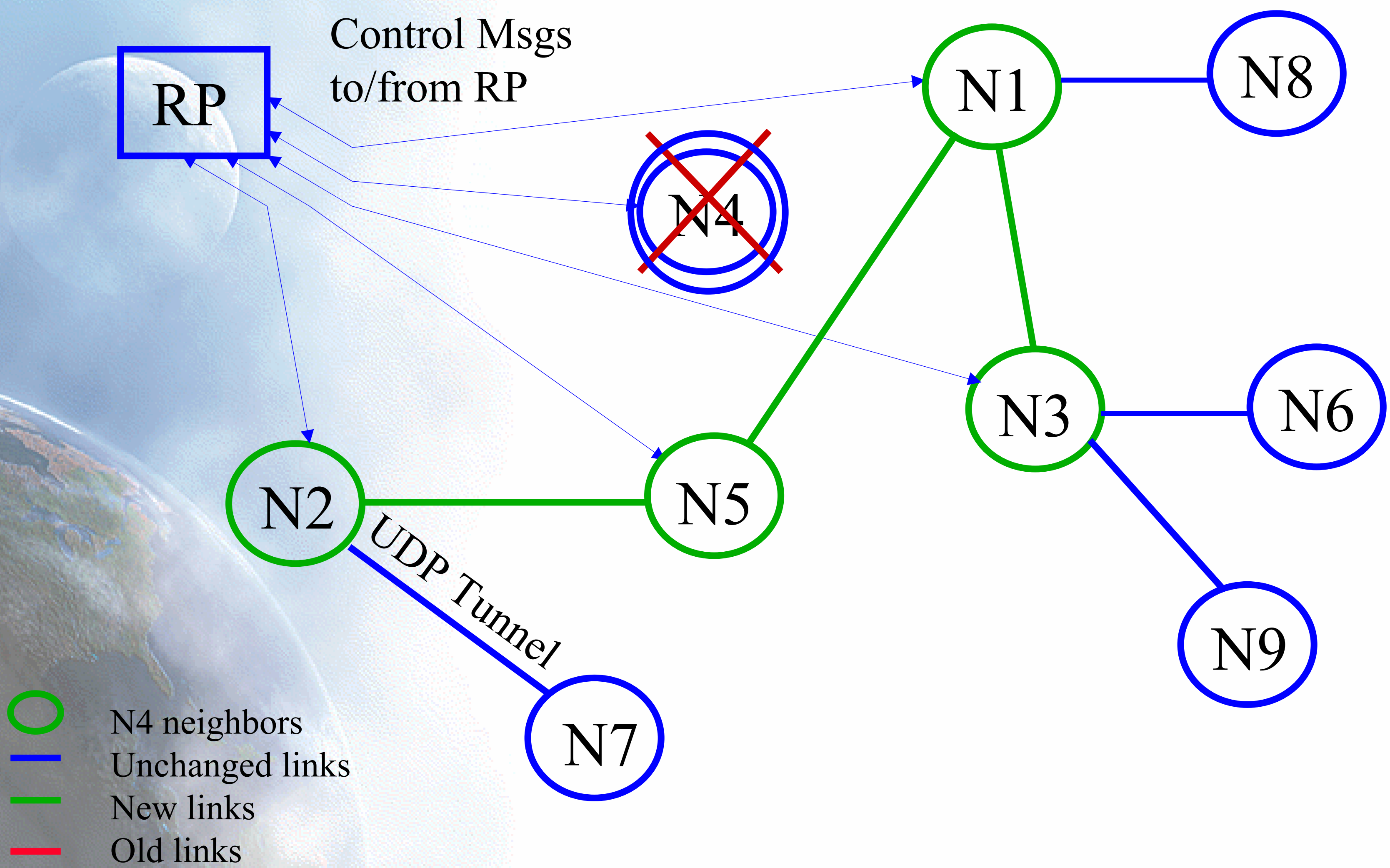
Our HBM … (cont')

- Example: node N4 leaves the group



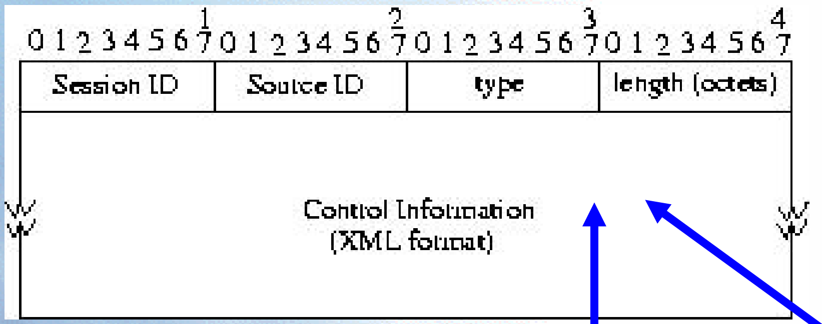
Our HBM … (cont')

- Example: node N4 leaves the group

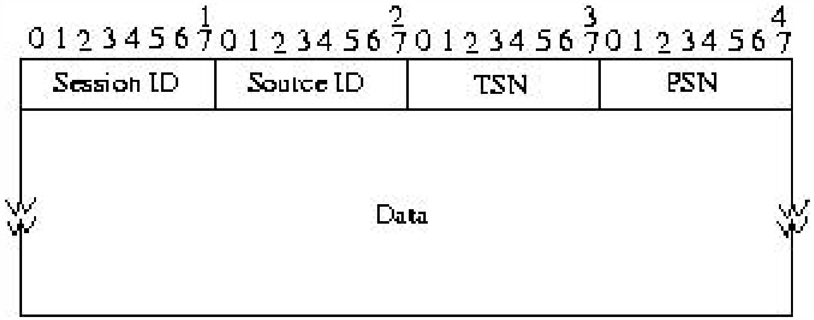


Our HBM Proposal …(cont')

- The Message/Packet Format



(TCP/IP) control message



Forwarded data Packet Format (UDP/IP).

```
<metricupdate>2                          # MU report form node id=2
 records=4                               # Number of metrics
<record>2, 1                             # metric between 2 and 1
<metric>2.50, 0.12</metric>              # RTT=10.10ms, loss=0.12%
</record>
<record>2, 3
<metric>30.10, 0.010</metric>
</record>
<record>2, 4
<metric>1.60, 0.195</metric>
</record>
<record>2, 5
<metric>10.10, 0.001</metric>
</record>
</metricupdate>                          # Message end
```

MU Control Info.

```
<topologyupdate>2                        # Message start for node id =2
 records=2                               # Number of links =2
<record>2, 4                             # link between 2 and 4
 type=1                                  # type of link = 1: tree link
groups=1                                 # Number of groups in this link =1
<group>16843232, 1111</group>            # group ip=224.1.1.1, port = 1111
</record>
<record>2, 1
 type=1
groups=2
<group>16843232, 1111</group>            # group ip=224.1.1.1, port = 1111
</record>
</topologyupdate>                        # Message end
```

TU Control Info.

Our HBM Proposal …(cont')

- Node characteristics are taken into account when creating the topology
  - Node stability
  - Node type of connection to the Internet
  - Node needs

- Distinguish
  - Core Member (CM) can be transit node
  - Non-core Member (nonCM) are always leaves

Part 3

- **Evaluation and Improvements**

  1. List of items addressed ←

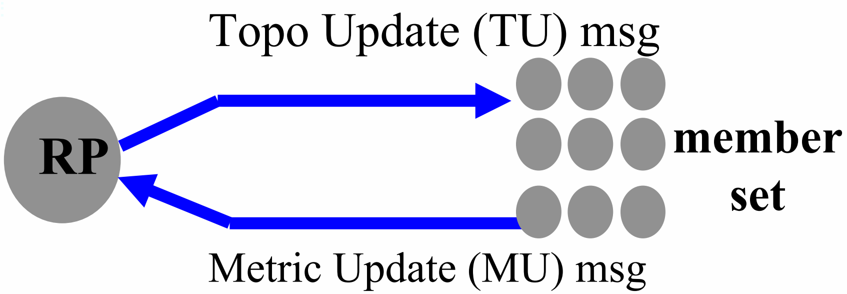
  2. Improving the robustness
     - in front of node failure
     - during a topology update

  3. An example of use: VPRN

List of items addressed

- Overlay topology creation

- Improving the scalability



  - Limit the control overhead
  - Found a strategy that has an appropriate compromise for that

We won’t detail them, we only focus on:

- Improving the robustness
- An example of use: VPRN

Part 3

- **Evaluation and Improvements**

1. List of items addressed
2. Improving the robustness
   - in front of node failure ←
   - during a topology update
3. An example of use: VPRN

Robustness In front of node failures

- Application-level partition is possible when a node fails
- Goal:

  reduce the partition probability
- Solution:

  Add Redundant Virtual Links (RVL)

- But:
  - How many RVL?
  - Between which nodes?
  - Source dependent or not?

Robustness In front of node failures…(cont')

- Adding RVL strategy I:
  - *Add a RVL between the farthest two nodes,*
  - *Split group into two subgroups,*
  - *Repeat for each sub-group which has at least 3 nodes.*

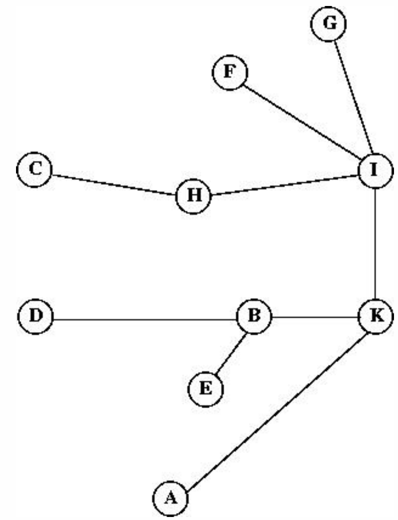
- Other possibilities: choose the farthest two nodes in the group where:
  - Strategy II : a leaf node can have at most one RVL
  - Strategy III: RVL between two leaf nodes are forbiden
  - Strategy IV: RVL between transit nodes only
  - Strategy V : RVL between each leaf node and its farthest transit node

Robustness In front of node failures…(cont')

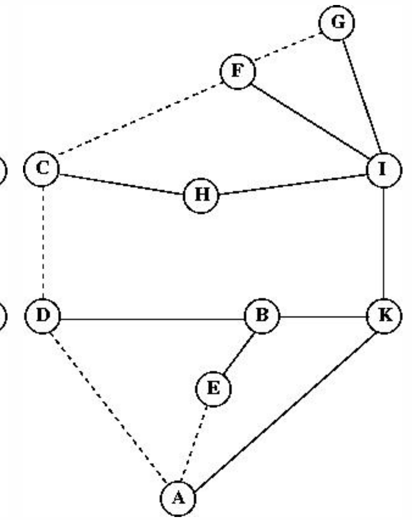
•An example: 10 nodes

Dotted line : RVL links

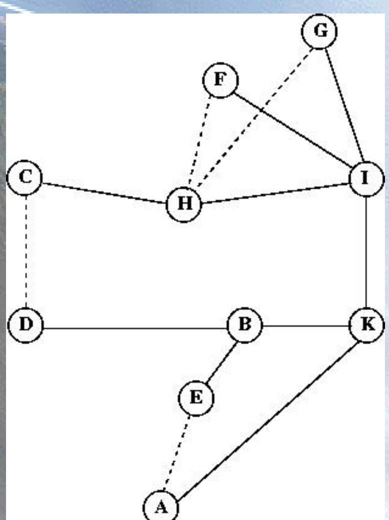
Bold line : Overlay links



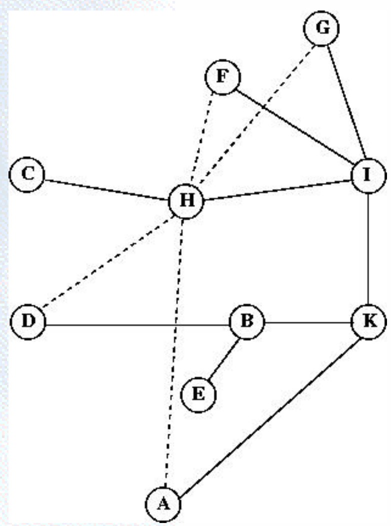
Initial Overlay



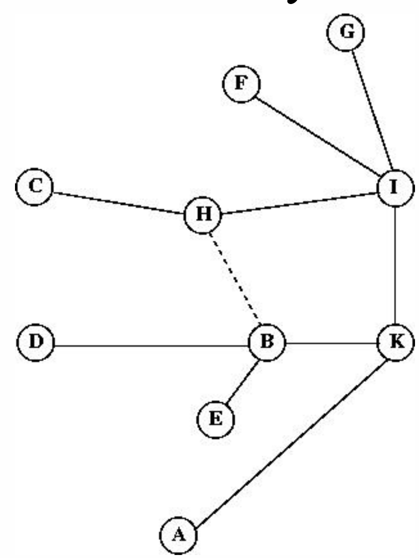
Strategy I



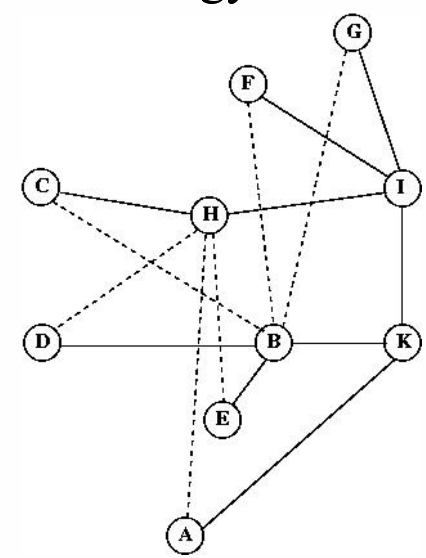
Strategy II



Strategy III



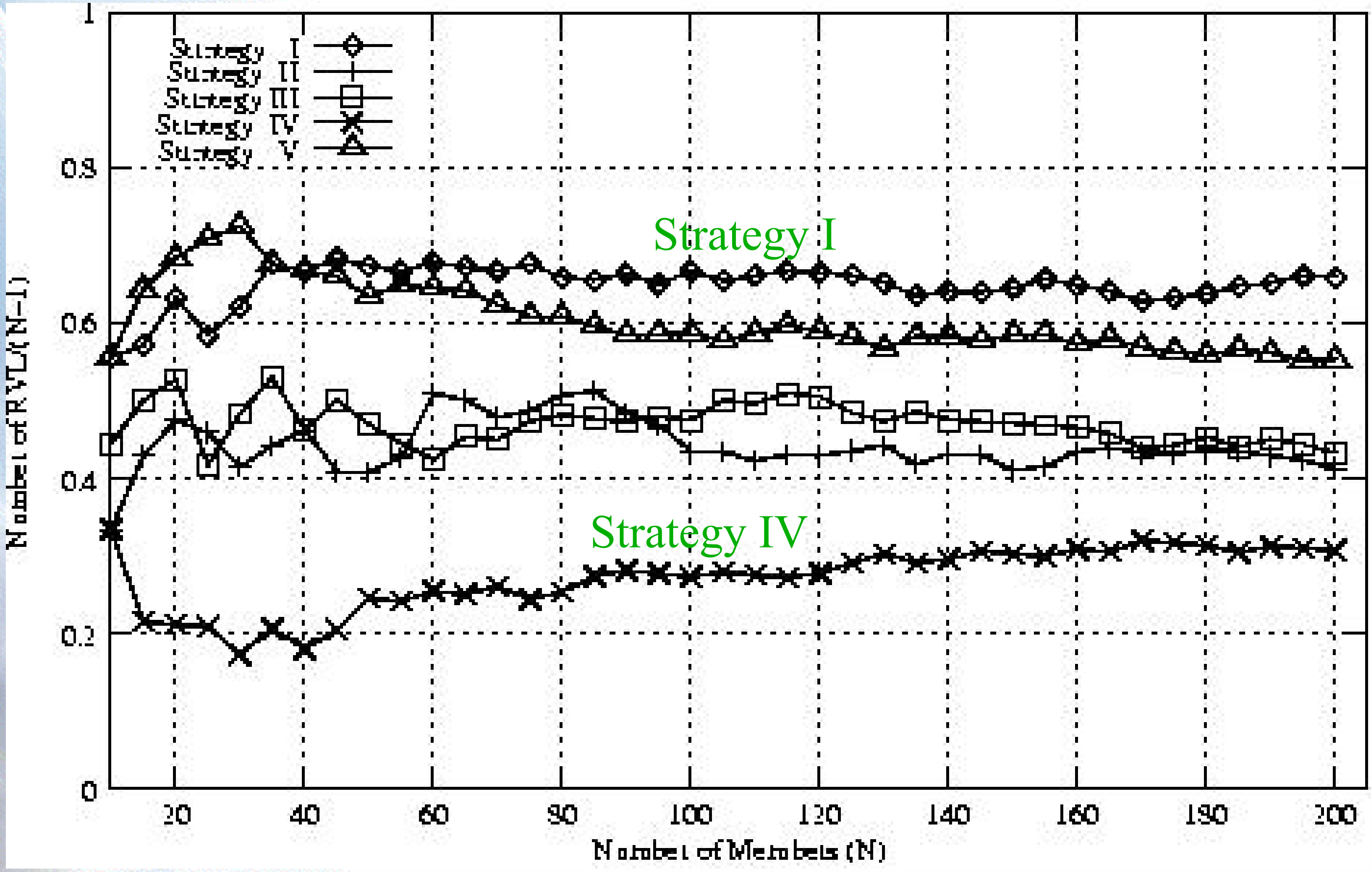
Strategy IV



Strategy V

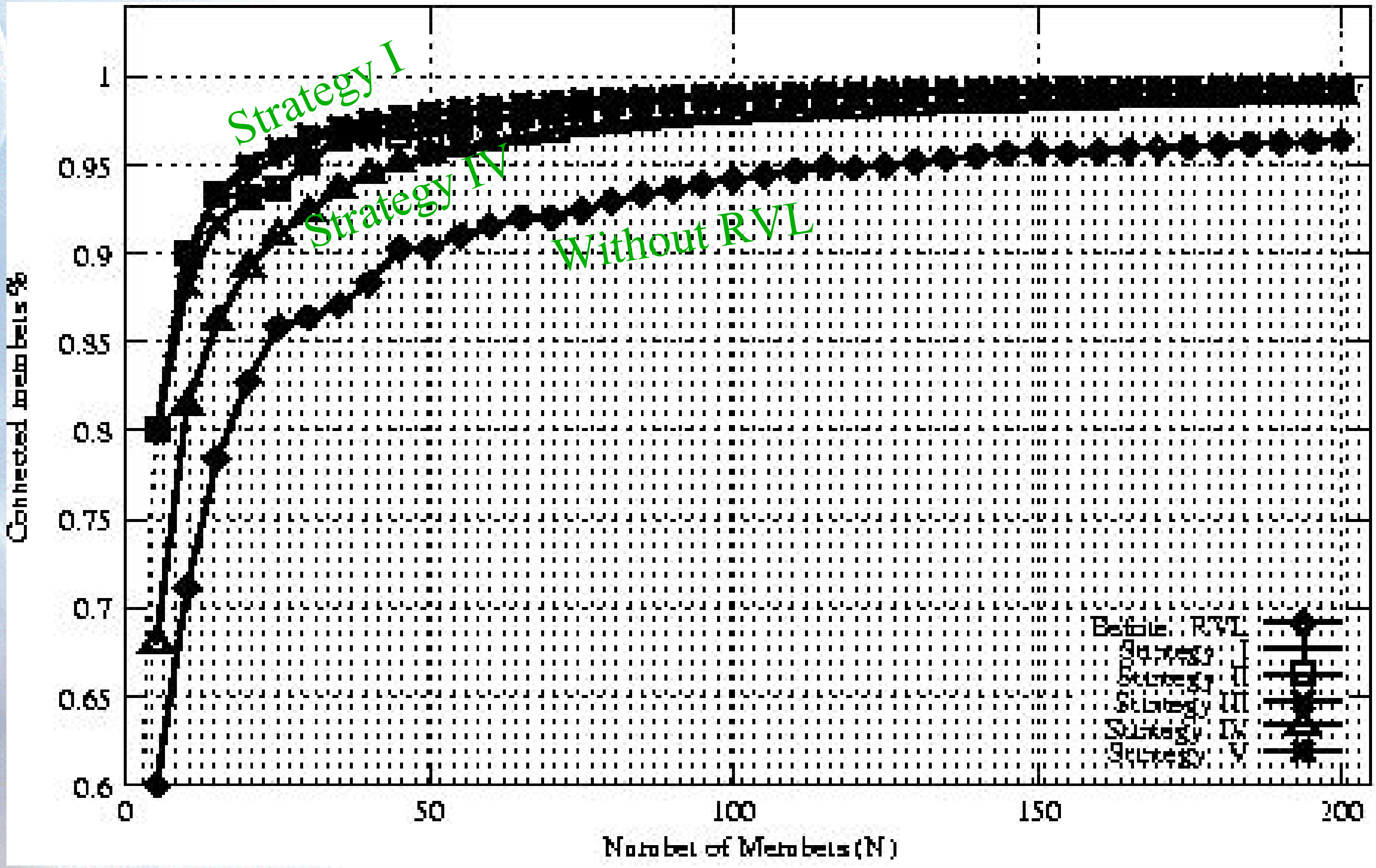
Robustness In front of node failures…(cont')

- Single failure, phys. topo. generated by GT-ITM, 600 routers
- We measure RVL Ratio = $\dfrac{Num\_Of\_RVL}{N-1}$



Robustness In front of node failures…(cont')

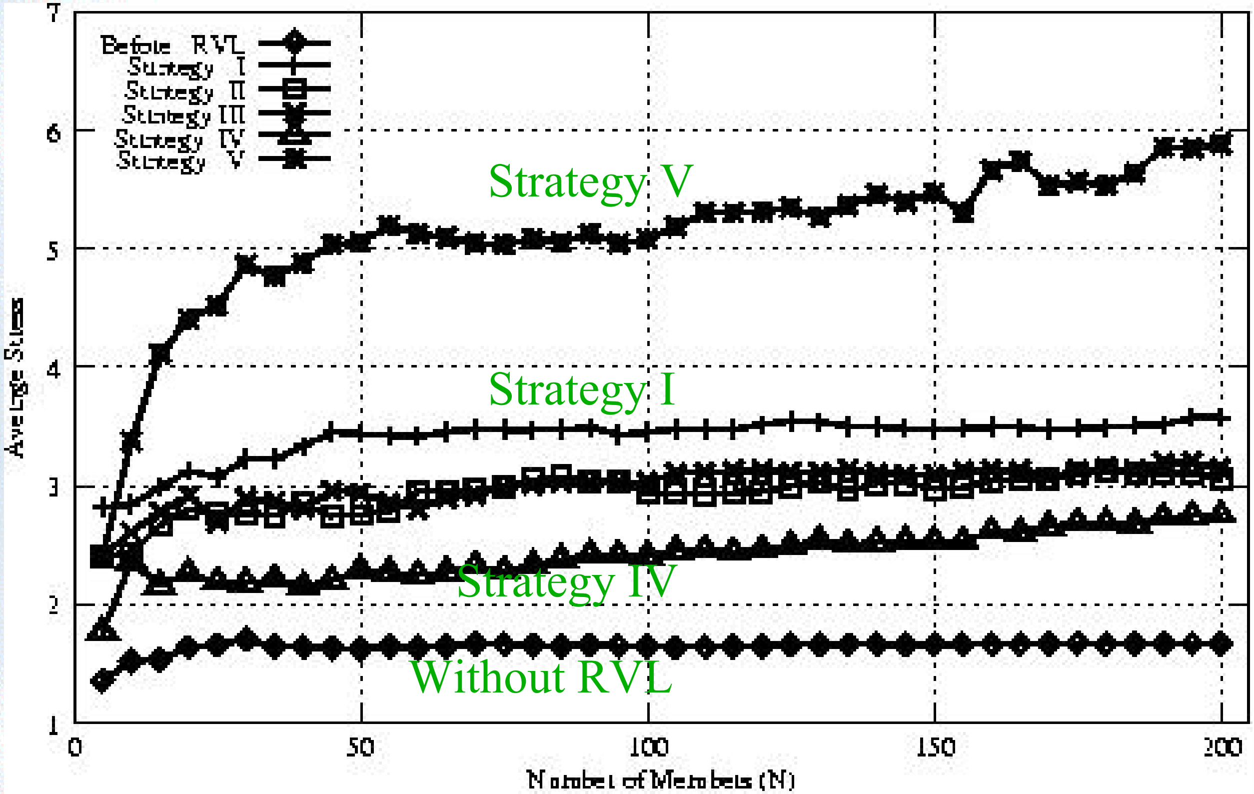
- Single failure, phys. topo. generated by GT-ITM, 600 routers
- We measure Ratio of connected nodes $= \dfrac{Num\_Of\_Connected\_Node}{N}$



Robustness In front of node failures…(cont')

•Single failure, phys. topo. generated by GT-ITM, 600 routers

•We measure Link stress:number of identical copies of packets carried by a physical link



Average link stress with/without strategies

Robustness In front of node failures…(cont')

- Conclusions
  - strategy 4 offers a good balance between the robustness and the additional traffic generated
  - they offer also some protection for two or more node failures

Part 3

- **Evaluation and Improvements**

1. List of items addressed
2. Improving the robustness
   - in front of node failure
   - during a topology update ←
3. An example of use: VPRN

Robustness during a topology update

- Application-level packet in transit can be lost during a topology update.

- Goal:

  reduce the packet loss probability

- Solution:

  Nodes remember several overlay topologies.

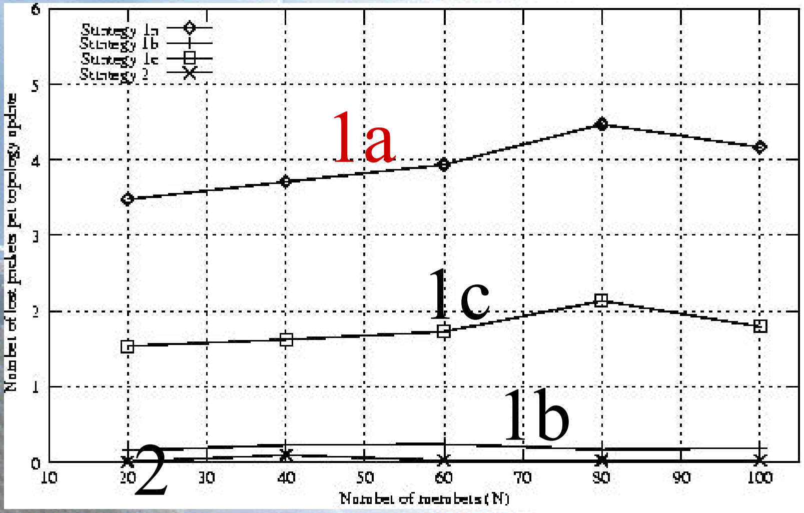
  Topologies are identified by a TSN which is included in the packet header.

Robustness during a topology update…(cont')

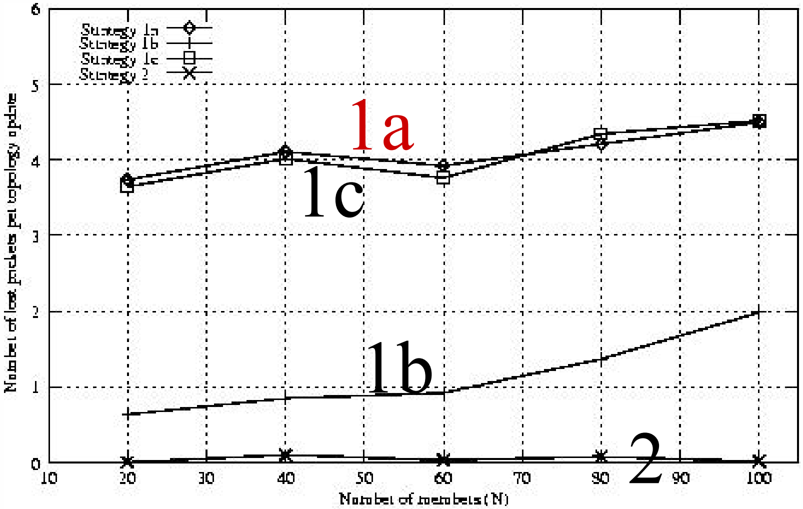
- Strategies for reducing packet losses
  - Strategy 1: remember the current topology only, if a packet is received via another topology:
    A. **drop this packet. → the reference**
    B. **if it has never been received before, forward over the current overlay**
    C. **If it is received from a link on current topology, forward it, otherwise drop it.**
  - Strategy 2: remember two topologies (previous and current). Forward the packets appropriately or drop.

Robustness during a topology update…(cont')

Results with data rate = 78 packet/sec (512 KbpS)



A small number of links are changed



All the topology links are changed

Robustness during a topology update…(cont')

- Conclusions
  - Strategy 2: remember two overlay topologies
  - Packet losses almost avoided
  - Does not depend on the importance of topology changes

Part 3

- **Evaluation and Improvements**

1. List of items addressed
2. Improving the robustness $\left\{ \begin{array}{l} \text{in front of node failure} \\ \text{during a topology update} \end{array} \right.$
3. An example of use: VPRN

An exampleof use: VPRN

- Application-level the security is not considered yet
- Goal:

  build a secure yet efficient group communication service in a VPN environment

- Solution: Virtual Private Routed Network (VPRN) concept.

An example of use: VPRN …(cont')

What is a VPRN?

«Virtual Private Routed Network»

Secure IP VPN environment for group communication services (IVGMP)

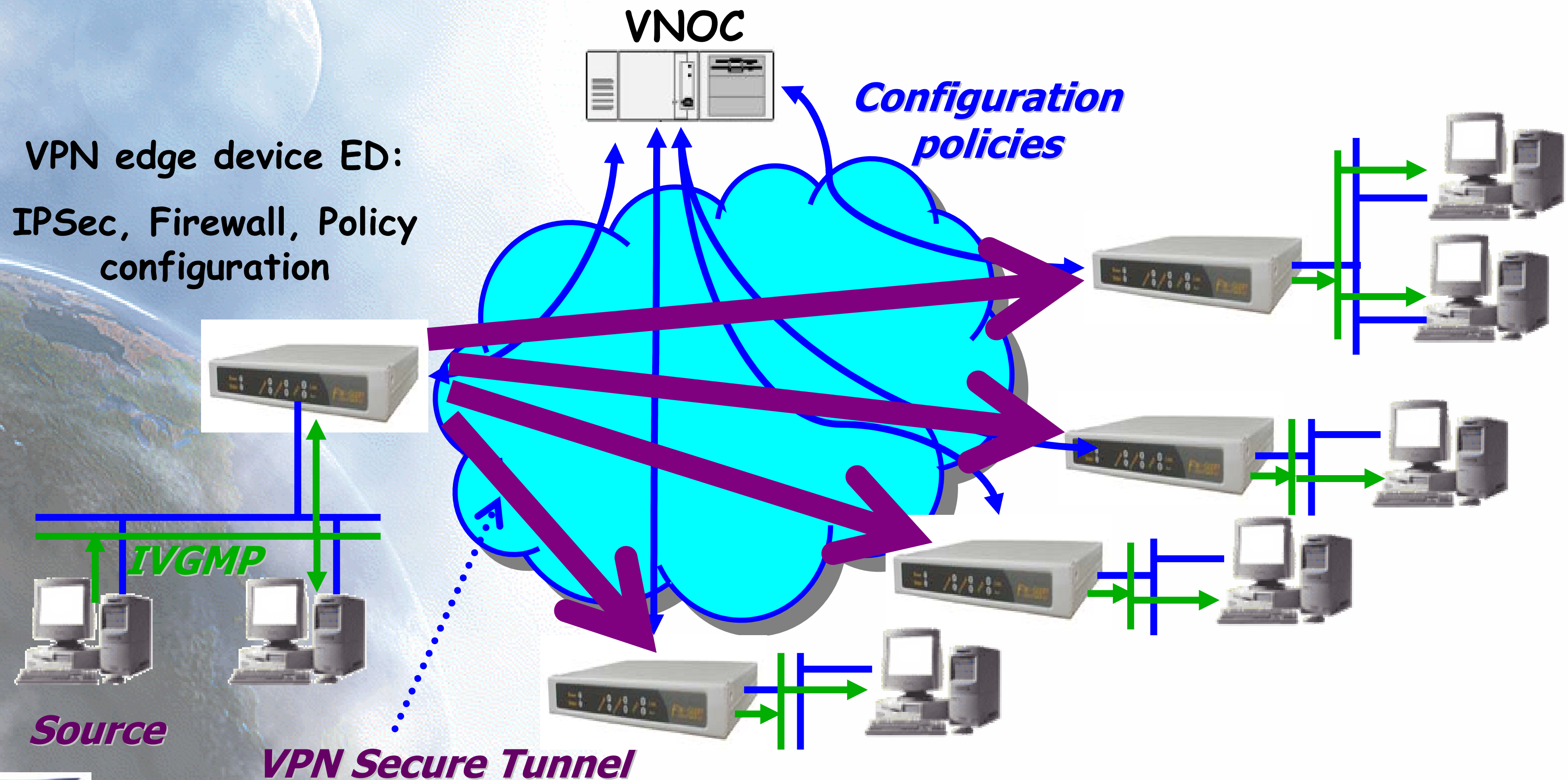
Application-level multicast approach (HBM)

A VPRN solution(or routed VPN) for fully secure yet efficient group communications

An example of use: VPRN …(cont')

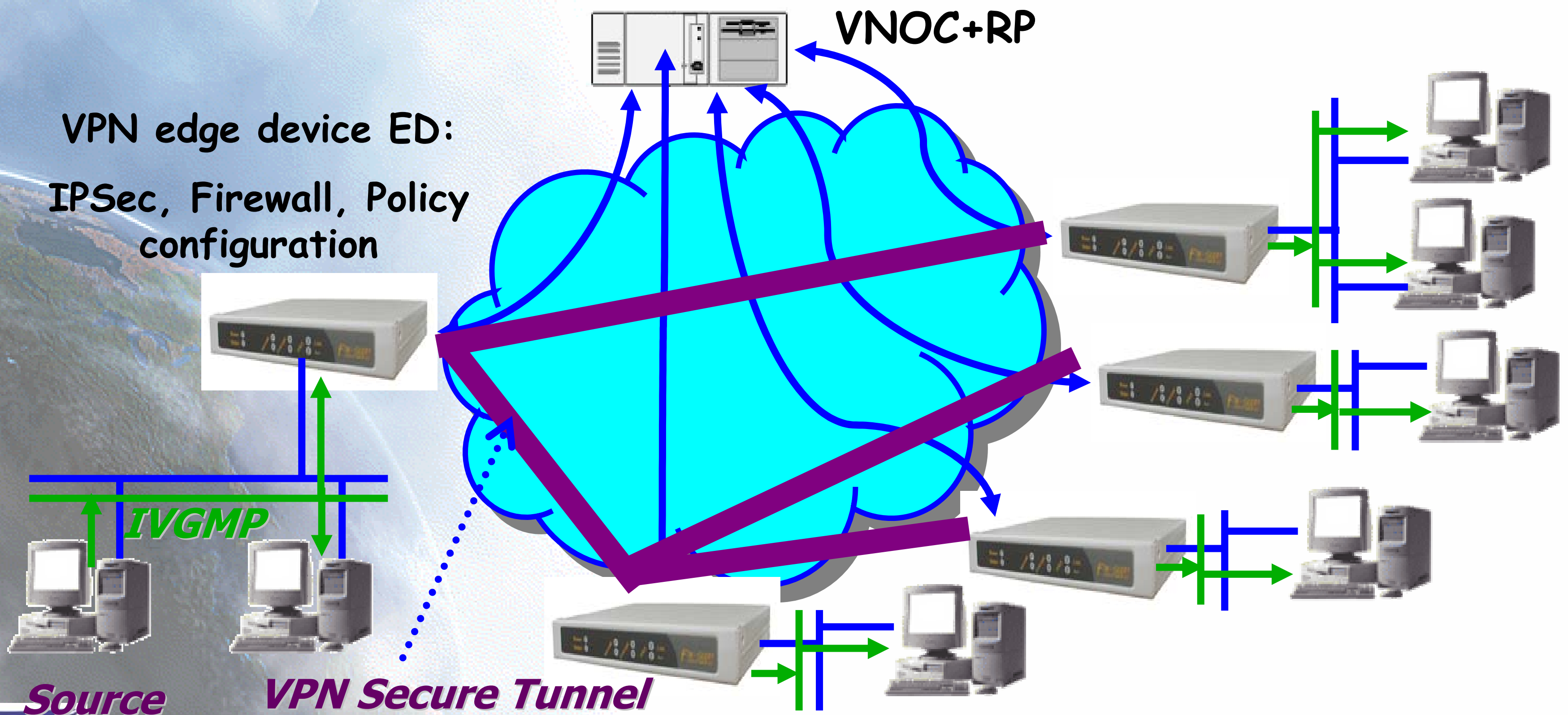
Centralized IP VPN Environment: (Lina Alchaal)

IP VPN: build a secure connection between remote sites across the Internet



An example of use: VPRN …(cont')

- IVGMP/HBM Architecture
  - Add RP functionality to the VNOC
  - Each VPN site can act as a VPRN node
  - Each ED is authenticated by the VNOC
  - VNOC-ED communications are secured with SSL
  - ED-ED communications are secured with IPSec



An example of use: VPRN …(cont')

- Conclusions
  - A new VPRN architecture
  - Fully independent from the ISP
  - Fully dynamic
  - Merge : a VPN group communication architecture + an application-level multicast approach
  - Improved scalability (# of sites) for multicast bulk data distribution

Part 4

Discussion, Conclusion, and Future Work

Discussion, Conclusion, and Future Work

- Ease of Deployment
  - HBM Group Communication Service Library (GCSL) can be:
    - **integrated in applications requiring a group communication service**
    - **a standalone application**
  - GCSL only needs: RP IP address/port number and Group address/port number
  - Future Work:
    - **firewalls→use Application-level gateway to ensure the correct translation of address/port number.**

Discussion, Conclusion, and Future Work

- Robustness
  - Application-level is fragile →partition is possible
  - RP has a global and coherent view of the overlay topology
    - **Robustness improvement is easy**
  - With distributed approach
    - **Robustness improvement is not easy, requires random, less efficient solutions**

Discussion, Conclusion, and Future Work …(Cont')

- Impact of cheats
  - Cheats try to improve their position on the topology:
    - **Directly connected to the source**
    - **No child.**
  - reports minimal distance to the source and huge distance to the rest of the group.

Discussion, Conclusion, and Future Work …(Cont’)

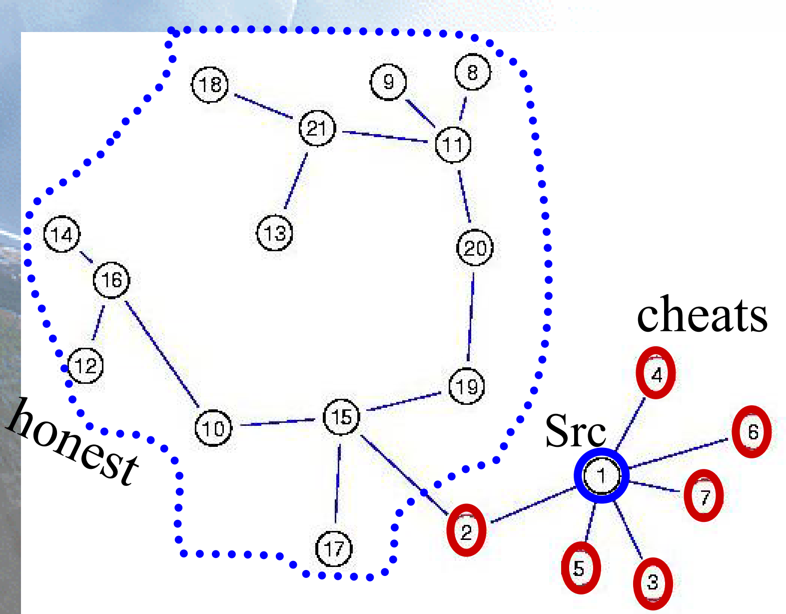
Impact of cheats…(cont’)

- An example: fanout =6

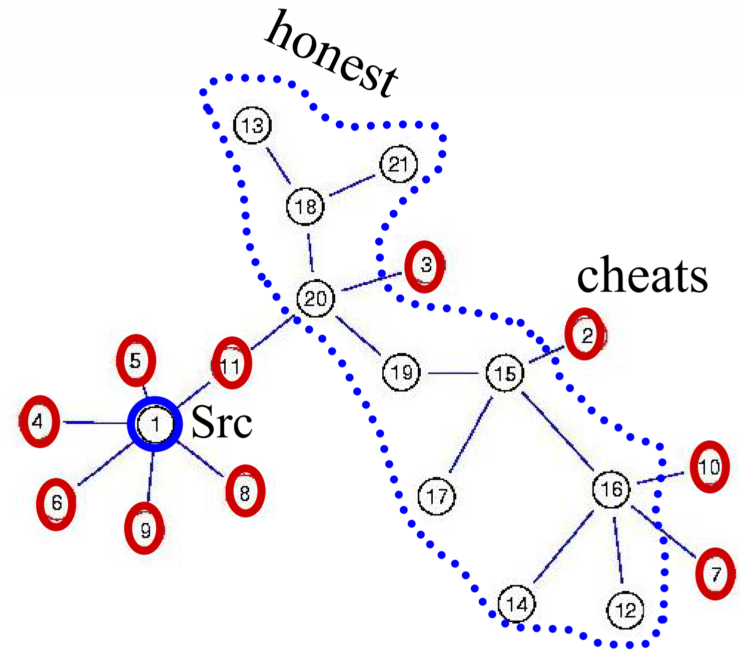
Source-Cheat =0 sec

Cheat-Cheat =RTT+20sec

NonCheat-Cheat=RTT+10sec



Number of cheats = 6



Number of cheats = 10

Discussion, Conclusion, and Future Work …(Cont')

- Impact of cheats…(cont')
  - Conclusion
    - **Cheating is not always efficient**
      - Some cheats are directly connected to the source
      - Other cheats are connected randomly to honest nodes
    - **Cheats lead to sub-optimal overlay topologies**
    - **If cheating is done in a trivial way, detecting them with HBM is possible:**
      - Ex: RTT to source = 0 ➔ it's a cheat
    - **But cheats can be more subtle**

      → Future Works

Discussion, Conclusion, and Future Work …(Cont')

- Security
  - is Neglected in Application-level multicast
    - **Control mechanisms are not secured**
    - **No authorization, authentication, encryption …**
  - But HBM with VPN →VPRN
  - how the authorization, authentication, …etc can be provided by HBM in the future

Discussion, Conclusion, and Future Work …(Cont')

- Performance

  Depends on:

  - Type of topology created
    - **A per-source shortest path tree is more efficient than a single shared tree but has a higher cost**
  - Dynamic topology
    - **Better reflects the dynamic networking conditions**
    - **But the update frequency is low since it creates a high signaling load**
  - Metrics
    - **Tools like ping assume symmetric paths, while in reality paths are often asymmetric**
    - **RTT/loss is not sufficient, other metrics may be more suited depending on the application**

Discussion, Conclusion, and Future Work …(Cont')

- Scalability
  - Not an obligation with Application-Level multicast
    - **Depends on the application.**
  - Other forms of scalability exist
    - **High number of group**
  - Future works
    - **Using a single overlay toplogy for several closely related groups (e.g.. In collaborative work tools).**
    - **One representative per site can distribute traffic locally, using intra-domain multicast routing**

Discussion, Conclusion, and Future Work …(Cont')

- A few more words
  - Many open points
  - « Application requirements » * « problems » is large
  - Our solution addresses only a subset of them !

The End

Merci de m'avoir écouté