Survivability in Wireless Networks: A Case for Overhead Reduction

Axel Krings
Computer Science Department
University of Idaho

krings@cs.uidaho.edu
http://www.cs.uidaho.edu/~krings/

Outline

- Introduction
- Background and Definitions
- Wireless Network Model
- Increasing Path Reliability
- Overload Scheduling
- Reliability Analysis
- Conclusions

Introduction

- Wireless Networks have gained great popularity
- Special focus
    - Ad hoc networks, MANETs
    - Sensor networks
- Wireless has many potential problems w.r.t.
    - Security
    - Reliability
    - Mobility

Introduction

- Problems include
    - Security
        - broadcast, "everybody can see"
        - nodes may be captured/impersonated/... many flavors
    - Reliability
        - nodes may be mobile
        - links and nodes have reliability/availability constraints
        - external interference, benign - malicious

Fault Models

- What are the assumptions about faults?
  - crash faults, omission faults, etc.
  - independence of faults
  - dependence of faults => common mode fault
  - recovery differs greatly depending on the fault model

Recovery needs Redundancy

- Time redundancy
- Information redundancy
- Spatial redundancy

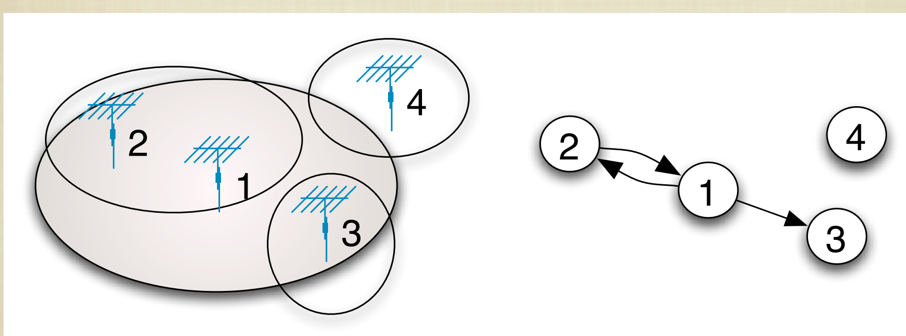
  e.g. if one considers *s* symmetric and *b* benign faults, then one needs $N > 2s + b$ redundant units to mask the faults

Fault Assumptions

- Faults are seen only in the context of their definition within the fault model under consideration
- Many mechanisms from security & fault-tolerance
  - e.g. encryption, authentication, ...
- BUT in the end, their impact on the faults they can produce is what really counts

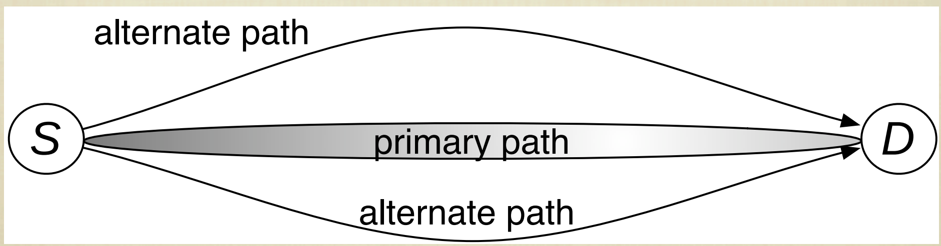
Network Graph

- Network Graph G is a digraph



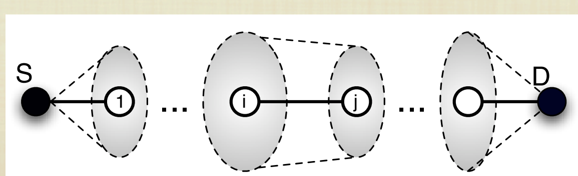
Network Graph

- General Communication Model
  - Reliability considerations:
    - increase path reliability/security
    - utilize multipath approach



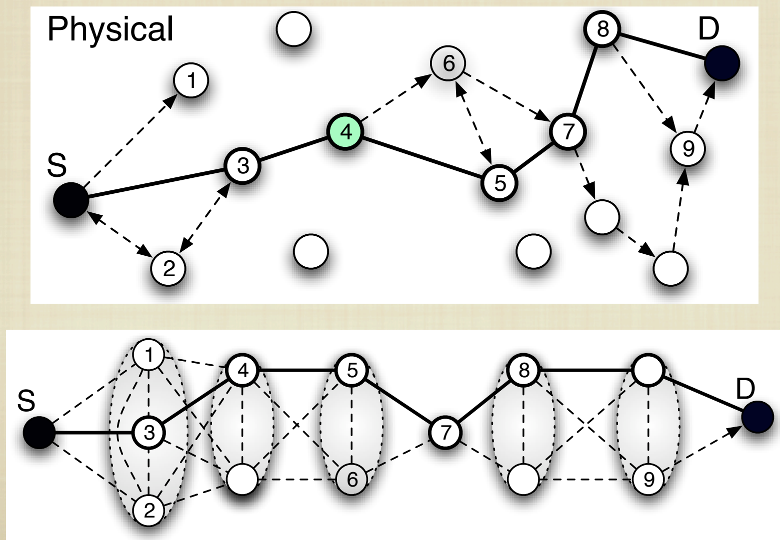
Increasing Path Reliability

- Two dimensional watchdog approach
  - Krings Axel and Zhanshan Ma, "Fault-Models in Wireless Communication: Towards Survivable Ad Hoc Networks", MILCOM 2006, 23-25 October, 7 pages, 2006.
  - Use neighborhood induced by general join graph (GJG)



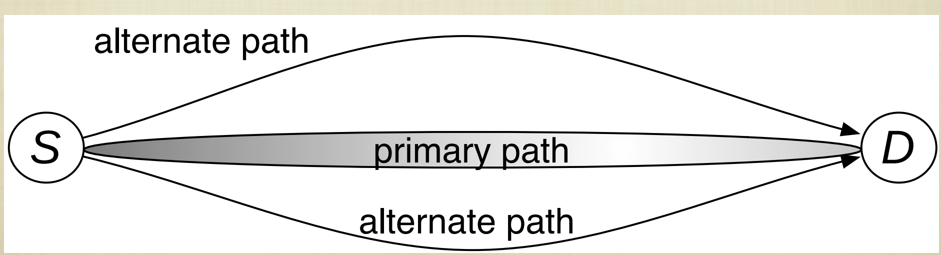
Example

- Assume nodes are moved to implement the GJG below



Multi-Path Approach

- Increased Reliability through Multi-path Routing
  - single path (even if GJG) may be subject to local disturbance
  - alternate paths can serve as multi-path option
    - multi-path is not a new concept, but this is different
    - what about the overhead....?

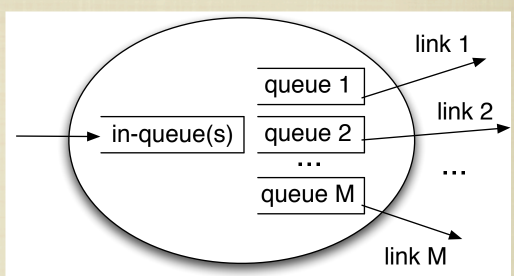


Simple Overlay Scheduling

- Used in Real-time Multi-processor Systems
  - Ghosh [1994], Tsuchiya [1995], Ghosh [1997], Manimaran [1998], Al-Omari [2004],...
- Primary-backup scheduling
  - overhead is negligibly small in the fault-free case
  - non-preemptive task consists of primary and backup
  - accept new task into system if feasibility test guaranteed that task can be scheduled to meet it deadline
  - uses backup overloading to avoid unnecessary overhead

Conceptual Network Node

- Node is viewed as having
  - input queue(s)
  - output queues/links



- This makes sense in fixed network, but what about wireless nodes?
  - MIMO
  - CDMA
  - TDMA

Packet Attributes

- A Packet P_j is scheduled on link L_i
- Packet attributes

a_j	arrival time
r_j	ready time
s_j	start time (of transmission)
l_j	transmission time (depends on length and line speed)
f_j	finish time
d_j	deadline

Primary-Backup

- A packet P_i consists of two parts
  - Primary Pr_i
  - Backup copy Bk_i
    - Bk_i serves as backup if primary fails
    - If Pr_i is delivered successfully, Bk_i is "unscheduled"

Primary-Backup

- Acknowledge time $ack(Pr_i)$
  - constitutes the maximum time up to which one can wait for an acknowledge
- Actual acknowledge time $t_{ack}(Pr_i)$
  - actual time when Pr_i is acknowledged

$$ack(Pr_i) = s(Pr_i) + \alpha t_a$$

    - alpha is a constant affecting how sensitive the fault detection is
    - t_a is the expected time to acknowledge Pr_i

Restrictions on Primaries

- Lemma 1
  - The primary and backup of P_i cannot be scheduled on the same link

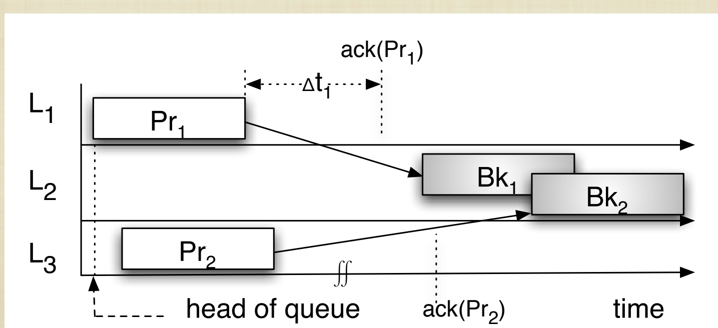
$$L(Pr_i) \neq L(Bk_i)$$

Restrictions on Primaries

- **Lemma 2** *Given Lemma 1, if two backups Bk_i and Bk_j are overlapping on a link, i.e. $S(Bk_i) \cap S(Bk_j) \neq \Phi$, then Pr_i and Pr_j must be scheduled on different links, i.e. $L(Pr_i) \neq L(Pr_j)$. Conversely, if Pr_i and Pr_j are scheduled on the same link, then their backups must not overload.*

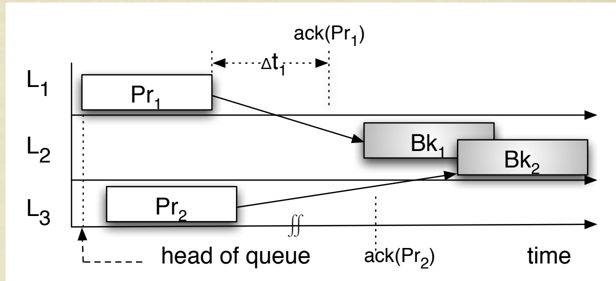
Backup Overloading

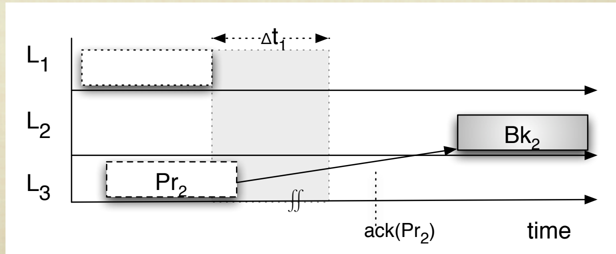
- Lemma 2
  - If two backups Bk_i and Bk_j are overlapping on link L_k, then Pr_i and Pr_j must be scheduled on different links, i.e.,



No-Fault Scenario

- If acknowledgment $t_{ack}(Pr_1)$ arrives in Δt_1 then Bk_1 is unscheduled
- Note: at $t_{ack}(Pr_1)$ packet Pr_2 may or may not have been sent out, but acknowledgment may not arrive until $ack(Pr_2)$





Unscheduling

- **Lemma 3** *Given packet P_i, backup Bk_i can be deleted only if Pr_i is delivered successfully at $t_{ack}(Pr_i) \leq ack(Pr_i)$.*

Time-To-Second-Fault

- Link 1 experiences a permanent fault

$$\text{TTSF}(L_2) = t_{ack}(Bk_1) \leq ack(Bk_1)$$
$$\text{TTSF}(L_3) = t_{ack}(Pr_2) \leq ack(Pr_2)$$
$$\text{TTSF} = \max\{\text{TTSF}(L_2), \text{TTSF}(L_3)\}$$

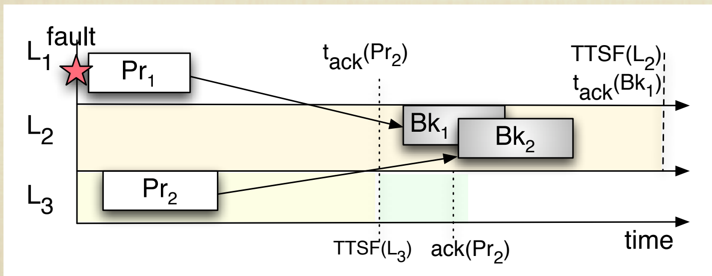
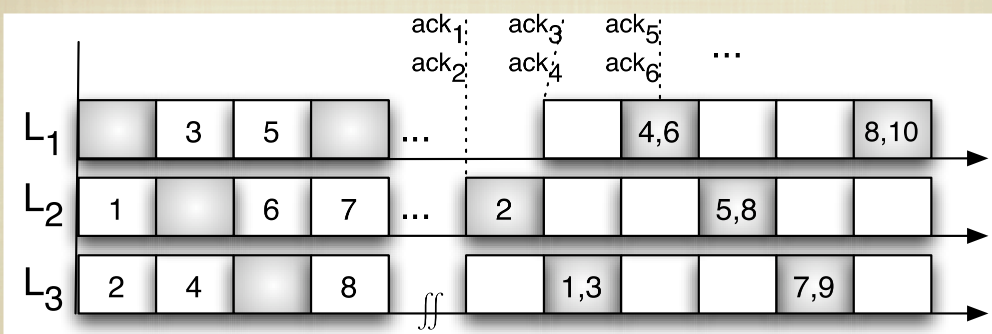


Figure 9. TTSF after link fault

Fixed Packet Link Allocation

- Backup slots are striped

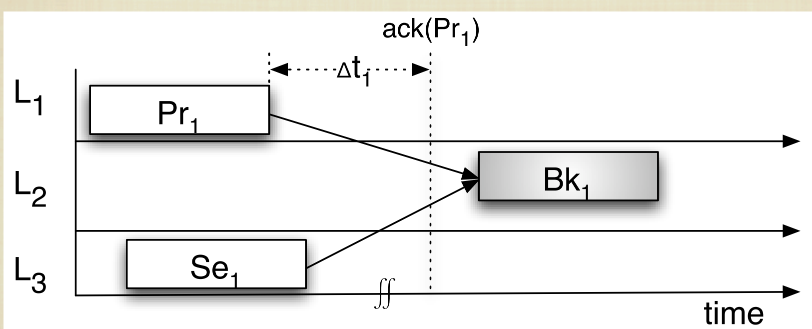


Overlay Scheduling for Hybrid Fault Models

- The concept can be extended to include extensions, analogous to the alternatives in FERTstones
  - [Bondavalli, Stankovic, Strigini 1993]
  - TMR, hybrid-selfchecking-TMR, k-of-N

Hybrid-selfchecking-TMR

- The concept is essentially equivalent to



Permanent Value Fault

Lemma 4 *Assume there is a source for permanent value faults. To avoid packet loss, the primary, secondary and backup of P_i must be scheduled on different links, i.e.* $L(Pr_i) \neq L(Se_i) \neq L(Bk_i)$.

Assume Value Fault

Theorem 2 *Assume that packets P_i are scheduled using backup overloading under a hybrid-SCP-TMR strategy. Furthermore, assume that at time t link L_k experiences permanent value faults. Then another fault can be tolerated at time* $t' = \max\{t_1, t_2, t_3\}$, *where*

$$t_1 = \max\{t_{ack}(Bk_i), \ \forall Pr_i : L(Pr_i) = L_k\}$$

$$t_2 = \max\{t_{ack}(Bk_i), \ \forall Se_i : L(Se_i) = L_k\}$$

$$t_3 = \max\{t_{ack}(Pr_i), t_{ack}(Se_i), \ \forall Pr_i, Se_i : L(Bk_i) = L_k\}$$

If the exact time of $t_{ack}(Pr_i) \leq ack(Pr_i)$ *is not known,* $t_{ack}(Pr_i) = ack(Pr_i)$ *must be assumed. The same holds for* Se_i *and* Bk_i.

Reliability of PB Scheduling

- Consider again previous example
- Four scheduling approaches
  - Single Path
  - PB Scheduling
  - Hybrid SCP-TMR Scheduling (for value faults)
  - Hybrid with benign faults only

Analytical Model

- Unreliabilities

Communication scenario	Unreliability $F(t) = 1 - R(t)$
Single Path	$F(t) = 1 - e^{-\lambda t}$
PB	$F(t) = 1 - 2e^{-\lambda t} + e^{-2\lambda t}$
Hybrid SCP-TMR	$F(t) = 1 - 3e^{-2\lambda t} + 2e^{-3\lambda t}$
Hybrid with Benigns	$F(t) = 1 - 3e^{-\lambda t} + 3e^{-2\lambda t} - e^{-3\lambda t}$

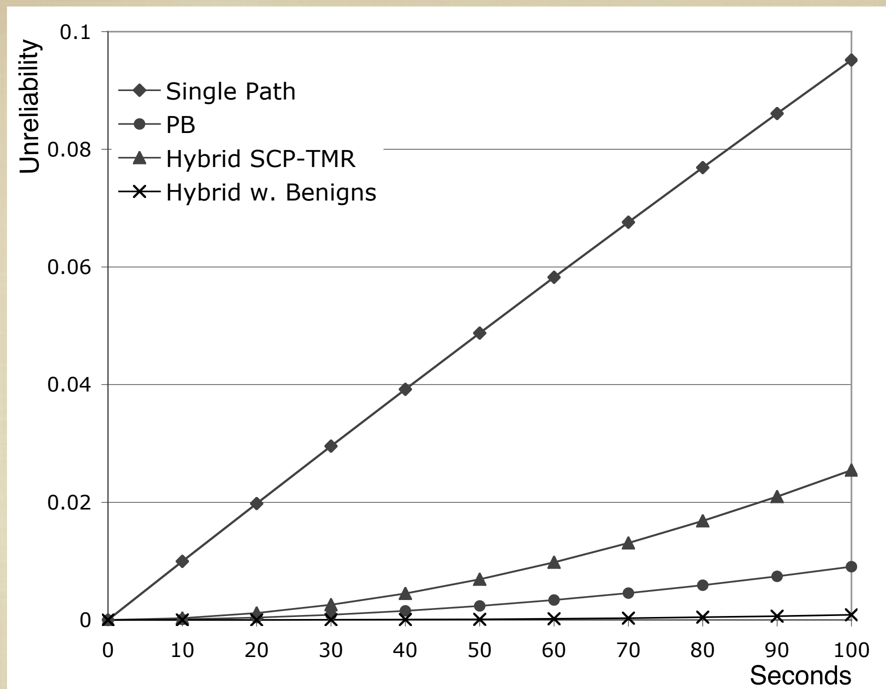


Figure 7: Communication link Unreliability - 100 seconds

fail rate 1/1000s

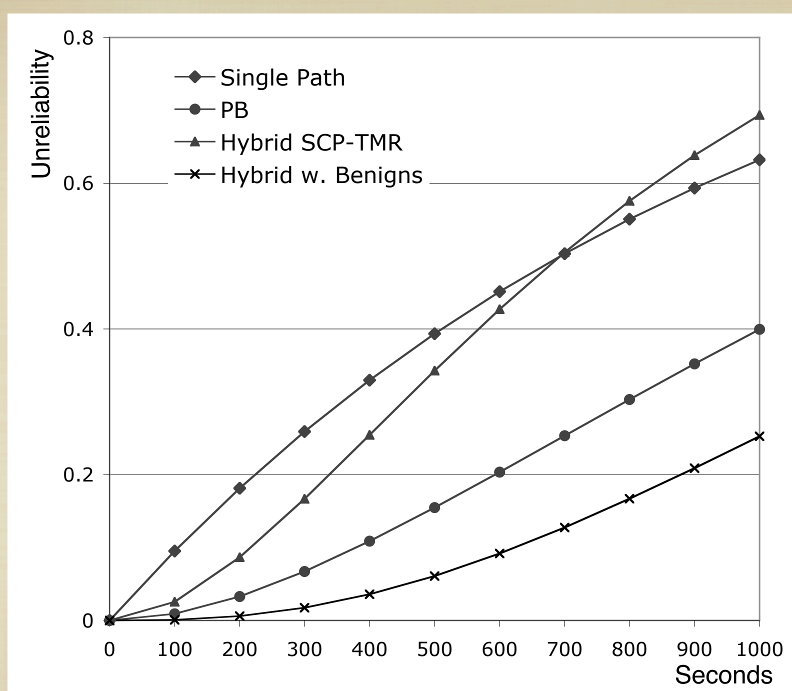


Figure 8: Communication link Unreliability - 1000 seconds

fail rate 1/1000s

Conclusions

- Reliability and survivability of wireless networks can be greatly improved by using cross-monitoring, i.e. GJG

- PB scheduling reduces overhead, increases network reliability and has potential to drastically reduce delays

  e.g. RTO (Retransmission Timeout period) in TCP

- Can be used to adapt network to the required level of reliability