**PIP: A Connection-Oriented, Multi-Hop, Multi-Channel TDMA-based MAC for High Throughput Bulk Transfer**

Bhaskaran Raman  
br@cse.iitb.ac.in

Kameswari Chebrolu  
chebrolu@cse.iitb.ac.in

Sagar Bijwe  
sag.bijwe@gmail.com

**Vijay Gabale**  
vijaygabale@cse.iitb.ac.in

**SyNerG: Systems and Networks Group**  
Department of CSE, IIT Bombay, India

**SenSys 2010**
Other applications: Volcanic activity monitoring (50KB/sensor), bulk data collection in sensor network

Goal: Effective and efficient delivery of sensed data over a multi-hop wireless sensor network.
**Goal:** Effective and efficient delivery of sensed data over a multi-hop wireless sensor network.

**Requirements and Challenges**

**Quick data transfer/High Throughput**
- Event detection ability is minimally affected.
- Energy savings after data collection.

**Reliability**
- Delivers important fragments of data.

**Challenges**
- Mitigating inter-path, intra-path interference?
- Using multiple channels for spatial efficiency?
- Handling wireless packet errors?
Focus of This Work

**A TDMA-based MAC** for multi-hop bulk data transfer

- What is the **capacity upper bound** for bulk transfer on an 802.15.4-based multi-hop wireless network?
- What is the **efficacy of a time synchronization mechanism** for a multi-hop wireless network?

PIP: A TDMA-based MAC for high throughput bulk transfer
Outline of Talk

- PIP Design Choices
- A Transport Protocol using PIP MAC
- PIP Implementation
- PIP Evaluation
- Related Work
- Conclusion
PIP MAC: Design Choices
**Goal:** Effective and efficient delivery of sensed data over a multi-hop wireless sensor network.

1. **Channel Access: TDMA**
2. **Access Control: Centralized**
3. **Multi-Channel Operation**
4. **Connection-oriented MAC**

**Mitigating inter-path*, intra-path interference**

**Using multiple channels for spatial efficiency**

**Handling wireless packet errors**

*We assume one flow at a time, and thus do not consider inter-path interference much like in Flush, Fetch.*
Goal: Effective and efficient delivery of sensed data over a multi-hop wireless sensor network.

1. Channel Access: TDMA

- Time divided into a frame.
- A frame divided into slots.
- Interfering links on different slots.

<table>
<thead>
<tr>
<th>Slots</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frame</td>
<td></td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>

What is the efficacy of time sync in practice? Later in Evaluation.
**Goal:** Effective and efficient delivery of sensed data over a multi-hop wireless sensor network.

1. **Channel Access: TDMA**
2. **Access Control: Centralized**
3. **Multi-Channel Operation**

**Mitigating intra-path interference**

**Using multiple channels for spatial efficiency**

**Handling wireless packet errors**

**Interfering links on different channels**

**50% throughput improvement**
Goal: Effective and efficient delivery of sensed data over a multi-hop wireless sensor network.

PIP: Design Choices

1. Channel Access: TDMA
2. Access Control: Centralized
3. Multi-Channel Operation

Mitigating intra-path interference
Using multiple channels for spatial efficiency
Handling wireless packet errors

Reliability
Loss of time-sync

Per-hop ack
In-band time sync
Goal: Effective and efficient delivery of sensed data over a multi-hop wireless sensor network.

1. Channel Access: TDMA
2. Access Control: Centralized
3. Multi-Channel Operation
4. Connection-oriented

Mitigating intra-path interference
Using multiple channels for spatial efficiency
Handling wireless packet errors

Connection-state for a node: (slot, channel, sender/receiver).
Transport protocol is simplified using PIP MAC module.

3 phases: Schedule dissemination, data transfer, tear down
Sink

Has routing path, computes schedule

Source

Bulk data
PIP: MAC Operation

- Extract schedule info
- Forward request
- Start MAC level timer
- Tune to reception data channel

Schedule + Conn Request
In-band time sync
- Data packets carry time sync info
- Nodes synchronized to source
PIP: MAC Operation

- **Sink**
- Schedule + Conn Request
- Data + Ack
- EOF
- SNACK
- Source

- **EOF**: End Of File
- **SNACK**: Selective Negative Ack

SMC
PIP: MAC Operation

1. **Sink**
   - Schedule + Conn Request
   - Data + Ack
   - Tear Down

2. **Source**
   - EOF
   - SnACK

3. **SMC**

4. **Odd (slot 1)**
   - CH1
   - CH2

5. **Even (slot 2)**
   - CH1
   - CH2

6. **PIP: “Packets In Pipeline”** along the path
Combination of design choices is unique to achieve multi-hop bulk transfer.

Protocol is simple to implement on resource constrained platforms.

In-built and on-demand time synchronization.

Periodic data packets carry timing info, no separate channel required for time sync.
PIP: Prototype Implementation
**Software/Hardware Platform:**

- **Telosb Platform**
  - **TI MSP 430 MCU**  
    (8MHz, 10K RAM, 48K Flash)
  - **CC2420 802.15.4 Radio**  
    (250Kbps, TX/RX Buffer = 128 B)
  - **Onboard Antenna**  
    (50m Indoors)
  - **TinyOS 2.1.0**

- **SFD based Time Sync:**
  - Hardware level timestamping (e.g. TSMP)
  - No software latency
  - Clock skew ~2 ticks (1 tick = 30.5 us)

- **All functions of the MAC implemented in prototype.**
- **Data of 103 B.**
- **Logging module to collect logs at each node**
# PIP: Guard Band, Frame Duration

<table>
<thead>
<tr>
<th>Component</th>
<th>Measured value (1 tick = 30.5 micro-sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time sync error</td>
<td>+/- 1 tick per hop</td>
</tr>
<tr>
<td>(a) right after sync</td>
<td></td>
</tr>
<tr>
<td>(b) drift error</td>
<td>&lt; 1.5 ticks/sec</td>
</tr>
<tr>
<td>Inaccuracy in timer fire</td>
<td>1-2 ticks</td>
</tr>
<tr>
<td>Processing jitter</td>
<td>1-2 ticks</td>
</tr>
<tr>
<td>Channel switching time</td>
<td>~ 10 ticks</td>
</tr>
</tbody>
</table>

- **Guard Time** = 15 ticks.
- **Slot Time** = 200 ticks.
  - Transmission + ACK
- **Frame Duration** = \((200 + 15) \times 2 = 430\) ticks.

**Time Sync is required only w.r.t. neighboring nodes.**
PIP: Pipelining Inside a Node

**No Pipelining**

1. **Critical Path:**
   - Step 1: RX BUF
   - Step 2: TX BUF
   - Step 3: MCU
   - Step 4: CC2420

**With Pipelining**

1. **Critical Path:**
   - Step 1: RX BUF
   - Step 2: TX BUF
   - Step 3: MCU
   - Step 4: CC2420

**Tmote Sky platform**

RX BUF
TX BUF
PIP: Pipelining Inside a Node

No Pipelining

With Pipelining

Tmote Sky platform

Critical Path:
**PIP: Evaluation and Comparison**

- 10 node (9 hop) implementation results
- Used Implementation parameters for PIP simulator
- Modeling a node's queue as DTMC and analytical results
- Comparison with Flush
Simulation, analytical and implementation results match.

~ 63Kbps throughput, 12 X improvement over Flush.
- PIP throughput decreases only slightly with number of nodes.
- PIP throughput is robust to error rates.

Throughput decreases slightly.
Robust to wireless packet errors.
Radio capacity of 250 Kbps = 138 ticks/128 B packet

Single radio forwarding, 2 slots, capacity reduced to 125 Kbps = 276 ticks/128 B packet

Ack packet reception 37 ticks
Software overhead (e.g. interrupts, logging) 25 ticks
Guard band 15 ticks

total ticks = 276 + 2 * (37+25 +15) = 430 (frame duration)

Effective throughput = 103 * 8 / 430 =~ 63 Kbps
**How would it have helped had PIP been used for the Volcano Monitoring application** *(thought experiment)*?

<table>
<thead>
<tr>
<th>No. of Hops</th>
<th>No. of Nodes</th>
<th>Fetch</th>
<th>Flush</th>
<th>PIP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>4.4</td>
<td>94</td>
<td>94</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>2.6</td>
<td>157</td>
<td>942</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>1.9</td>
<td>220</td>
<td>1100</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>1.4</td>
<td>283</td>
<td>566</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>1.2</td>
<td>346</td>
<td>346</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>1</td>
<td>409</td>
<td>409</td>
</tr>
</tbody>
</table>

**Effectiveness due to the multi-channel TDMA operation**

**Time required/node to transfer data is ~7 seconds.**

**Node’s ability to sense the events is minimally affected!**
For 10% error rate, PIP gives 50Kbps whereas Flush gives 8.5Kbps, a factor of 5.6 improvement over Flush.
Question: *Performance of PIP in the presence of external interference* (e.g. WiFi in 2.4 Ghz frequency range)?
Node follows a frequency sequence in circular fashion
Sequence conveyed during connection set up.

WiFi source (channel occupancy of ~2.3 ms)

Frequency/channel hopping across slots

5 Mhz channel of 802.15.4

20 Mhz spread of 802.11 channel

Question: Performance of PIP in the presence of external interference (e.g. WiFi in 2.4 Ghz frequency range)?
PIP: External Interference, Channel Hopping

Frequency hopping effective in mitigating external interference.

1.6 to 1.8 X improvement
PIP: External Interference, Channel Hopping

Frequency hopping effective in mitigating external interference.
## Related Work

<table>
<thead>
<tr>
<th></th>
<th>Centralized</th>
<th>Distributed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uncordinated (CSMA)</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Coordinated (TDMA)</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>RT-Link</strong></td>
<td><strong>Wimax, TSMP, PIP</strong></td>
</tr>
</tbody>
</table>

**PIP vs. WiMAX:** PIP - implementation-based evaluation for a bulk transfer application.

**PIP vs. TSMP/FPS:** PIP - optimized for bulk data transfer applications, uses *data packets themselves for synchronization*.

**PIP vs. Flush:**
*Time synchronized* slots to clock the packets.
*Multiple channels* to increase the spatial reuse.
*Hardware optimization* to move packet copy off the critical path.
Goal: Achieve high throughput bulk data transfer

**PIP: An efficient centralized multi-channel TDMA system**
- Effective time sync mechanism
- *Non-requirement of flow control*

**Feasibility & effectiveness via implementation & evaluation**
- Enables fast data transfer
- Robust to wireless packet errors, external interference
- *Low memory requirement: queue size of 10 packets*
Thanks! Questions???

PIP publication/presentation/source code at: http://www.cse.iitb.ac.in/~vijaygabale/pip/pip.php

Vijay Gabale
vijaygabale@cse.iitb.ac.in

SyNerG: Systems and Networks Group
http://www.cse.iitb.ac.in/synerg

SenSys 2010
**PIP: Pipelining Inside a Node and Slot Size**

- **TX (R) done**
  - Timer fired, start TX(R)
  - E_SFD detected, start ACK timer
  - FIFOP detected, cancel ACK timer
  - FIFOP high

- **RX (R) done**
  - S_SFD detected, start RX(S)
  - Packet received
  - Switch to RX channel, set next timer

**Slot size**

\[
\text{Slot size} = \max (\text{TX(R)}, \text{RX(R)}, \text{TX(S)}, \text{RX(S)}) = \text{TX(R)}
\]

- **TX(R)**
  - \(T\) (MAC payload 103 B + Time stamp 4 B + PIP header 3 B + 802.15.4 header 12 B) + \(T\) (ACK 4 B) + Software Overhead = 138 + 37 + 25 = 200 ticks
Guard Band: (1) No transmission (2) Continue reception
Queue requirement: To account for temporary variations in error rates of the incoming versus the outgoing wireless hop.

We choose queue size of 10 for our implementation.
Flow control: In simulator, a node “magically” knows queue state of downstream node and refrains from transmitting.

PIP performs well despite no flow control.
**PIP: Time Sync Mechanism**

Node 1

- Capture S_SFD, insert global timestamp (T1)

Node 2

- Capture S_SFD, record local time (T2)

**Offset of Node 2: T1 - T2.**

- Hardware level timestamping, no software latency.
- Clock skew of ~2 ticks (1 tick ~ 30.5 us).
- Periodic data packets keep clock correction to minimal.
External interference in wireless networks

Performance of PIP in the presence of WiFi in 2.4 Ghz range?

**Frequency/channel hopping in slots of the frame:**
- Node follows a frequency sequence of \{k, k+1, \ldots, n, 1, \ldots\} in circular fashion
- Sequence conveyed during connection setup.

**Experimental set up:**
- WiFi source on channel 6 (center frequency of 2.426 Ghz)
- Interferences with five 5Mhz 802.15.4 channels
- WiFi source generates 1777 B at 6Mbps (channel occupancy of \sim 2.3ms) for varying the inter-packet time.
Frequency hopping effective in mitigating external interference.