Physical Algorithms
Spot the Differences
Too Many!
Spot the Differences
Still Many!
Spot the Differences
Better Screen
Bigger Disk
More RAM
Cooler Design

…
Better Screen
Bigger Disk
More RAM
Cooler Design

...
Clock speed flattening sharply

Transistor count still rising

Advent of multi-core processors!
Why Should I Care?
Computer Science ➔ Washing Machine Science

[Roger Boyle, Maurice Herlihy]
Algorithms
simple and robust model
comparable results
complexity theory
Algorithm vs. Output
The Future of Computing?
Talk Overview

Introduction & Motivation

Examples for Physical Algorithms in the Context of Sensor Networks

What are Physical Algorithms?
Clock Synchronization
Clock Synchronization in Networks

Global Positioning System (GPS)

Radio Clock Signal

AC-power line radiation

Synchronization messages
Clock Synchronization in Networks

Global Positioning System (GPS)

Radio Clock Signal

AC-power line radiation

Synchronization messages
Problem: Physical Reality

Clock rate:

\[ \frac{1}{1-\epsilon} \leq \frac{1+\epsilon}{1} \leq 1 \]

Message delay:

Relative frequency vs. delay distribution.
Clock Synchronization in Theory?

Given a communication network

1. Each node equipped with hardware clock with drift
2. Message delays with jitter

Goal: Synchronize Clocks (“Logical Clocks”)

• Both global and local synchronization!
Time Must Behave!

- Time (logical clocks) should **not** be allowed to **stand still** or **jump**
• Time (logical clocks) should **not** be allowed to stand still or jump

• Let’s be more careful (and ambitious):
  • Logical clocks should **always move forward**
    • Sometimes faster, sometimes slower is OK.
    • But there should be a minimum and a maximum speed.
    • As close to correct time as possible!
Local Skew

Tree-based Algorithms
e.g. FTSP

Neighborhood Algorithms
e.g. GTSP

Bad local skew
Synchronization Algorithms: An Example ("A^\text{max}")

- **Question:** How to update the logical clock based on the messages from the neighbors?

- **Idea:** Minimizing the skew to the fastest neighbor
  - Set clock to maximum clock value you know, forward new values immediately

- **First all messages are slow (1), then suddenly all messages are fast (0)!**
Everybody's expectation, 10 years ago ("solved")

Lower bound of $\log D / \log \log D$
[Fan & Lynch, PODC 2004]

Blocking algorithm

All natural algorithms
[Locher et al., DISC 2006]

Tight lower bound
[Lenzen et al., FOCS 2008]

Dynamic Networks!
[Lenzen et al., PODC 2009]

Dynamic Networks!
[Kuhn et al., SPAA 2009]

Together
[JACM 2010]
Experimental Results for Global Skew

FTSP

PulseSync

[Lenzen, Sommer, W, SenSys 2009]
Experimental Results for Global Skew

[Lenzen, Sommer, W, SenSys 2009]
Clock Synchronization vs. Car Coordination

- In the future cars may travel at high speed despite a tiny safety distance, thanks to advanced sensors and communication.
Clock Synchronization vs. Car Coordination

• In the future cars may travel at high speed despite a tiny safety distance, thanks to advanced sensors and communication.

• How fast & close can you drive?

• Answer possibly related to clock synchronization
  – clock drift ↔ cars cannot control speed perfectly
  – message jitter ↔ sensors or communication between cars not perfect
Wireless Communication
Wireless Communication
- EE, Physics
- Maxwell Equations
- Simulation, Testing
- ‘Scaling Laws’

Network Algorithms
- CS, Applied Math
- [Geometric] Graphs
- Worst-Case Analysis
- Any-Case Analysis
CS Models: e.g. Disk Model (Protocol Model)
EE Models: e.g. SINR Model (Physical Model)
Signal-To-Interference-Plus-Noise Ratio (SINR) Formula

\[
\frac{P_u}{d(u,v)\alpha} + \sum_{w \in V \setminus \{u\}} \frac{P_w}{d(w,v)\alpha} + N \geq \beta
\]
Example: Protocol vs. Physical Model

Assume a single frequency (and no fancy decoding techniques!)

Let $\alpha=3$, $\beta=3$, and $N=10nW$

Transmission powers: $P_B = -15$ dBm and $P_A = 1$ dBm

**SINR of A at D:**
$$\frac{1.26mW/(7m)^3}{0.01\mu W + 31.6\mu W/(3m)^3} \approx 3.11 \geq \beta$$

**SINR of B at C:**
$$\frac{31.6\mu W/(1m)^3}{0.01\mu W + 1.26mW/(5m)^3} \approx 3.13 \geq \beta$$
This works in practice!

... even with very simple hardware (sensor nodes)

Time for transmitting 20’000 packets:

<table>
<thead>
<tr>
<th>Node</th>
<th>Time required (standard MAC)</th>
<th>“SINR-MAC”</th>
</tr>
</thead>
<tbody>
<tr>
<td>u₁</td>
<td>721s</td>
<td>267s</td>
</tr>
<tr>
<td>u₂</td>
<td>778s</td>
<td>268s</td>
</tr>
<tr>
<td>u₃</td>
<td>780s</td>
<td>270s</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Messages received</th>
<th>standard MAC</th>
<th>“SINR-MAC”</th>
</tr>
</thead>
<tbody>
<tr>
<td>u₄</td>
<td>19999</td>
<td>19773</td>
</tr>
<tr>
<td>u₅</td>
<td>18784</td>
<td>18488</td>
</tr>
<tr>
<td>u₆</td>
<td>16519</td>
<td>19498</td>
</tr>
</tbody>
</table>

Speed-up is almost a factor 3

The Capacity of a Network
(How many concurrent wireless transmissions can you have)
... is a well-studied problem in Wireless Communication

The Capacity of Wireless Networks
Gupta, Kumar, 2000

[Toumpis, TWC’03] [Gamal et al, INFOCOM’04]
[Liu et al, INFOCOM’03] [Kyasanur et al, MOBICOM’05]
[Kodialam et al, MOBICOM’05] [Gastpar et al, INFOCOM’02]
[Li et al, MOBICOM’01] [Mitra et al, IPSN’04] [Zhang et al, INFOCOM’05]
[Bansal et al, INFOCOM’03] [Dousse et al, INFOCOM’04]
[Yi et al, MOBIHOC’03] [Perevalov et al, INFOCOM’03]

etc...
Network Topology?

- All these capacity studies make very strong assumptions on node deployment, topologies
  - randomly, uniformly distributed nodes
  - nodes placed on a grid
  - etc.

What if a network looks differently...?
```

"Convergecast Capacity" in Wireless Sensor Networks

<table>
<thead>
<tr>
<th>Networks Model/Power</th>
<th>Worst-Case Capacity</th>
<th>Best-Case Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Protocol Model</td>
<td>Max. rate in arbitrary, worst-case deployment</td>
<td>Max. rate in random, uniform deployment</td>
</tr>
<tr>
<td></td>
<td>$\Theta(1/n)$</td>
<td>$\Theta(1/\log n)$</td>
</tr>
<tr>
<td>Physical Model</td>
<td>$\Omega(1/\log n)$</td>
<td>$\Omega(1/\log n)$</td>
</tr>
</tbody>
</table>

Exponential gap between protocol and physical model!

The Price of Worst-Case Node Placement:
- Exponential in protocol model
- Polylogarithmic in physical model (almost no worst-case penalty!)

[Convergecast Capacity](Moscibroda, W, 2006)
[Halldorsson, Mitra, 2012]

[Best-Case Capacity](Giridhar, Kumar, 2005)```
Physical Algorithms

Real Capacity
How much information can be transmitted in any network?

“Classic” Capacity
How much information can be transmitted in nice networks?

Worst-Case Capacity
How much information can be transmitted in nasty networks?
Wireless Communication

EE, Physics
Maxwell Equations
Simulation, Testing
‘Scaling Laws’

Network Algorithms

CS, Applied Math
[Geometric] Graphs
Worst-Case Analysis
Any-Case Analysis
... is really taking off right now!

The Complexity of Connectivity ...
Moscibroda et al., Infocom 2006

[Fanghaenel et al, PODC’09]
[Lotker et al., INFOCOM’11]  [Kesselheim, SODA’11]  [Fanghaenel et al, ICALP’09]
[Kesselheim et al, DISC’10]  [Halldorsson et al, SODA’11]  [Goussevskaia et al, INFOCOM’09]
[Avin et al, PODC’11]  [Kantor et al, STOC’11]  [Lebhar et al, IPDPS’09]
[Halldorsson, ESA’09]  [Hua et al, TCS’09]  etc...
Possible Application – Hotspots in WLAN
Possible Application – Hotspots in WLAN
Physical Algorithms?
Physical Algorithms

no seq. input/output       beyond laws of physics
Some Unifying Theory?
Distributed Algorithms is a tool to understand physical algorithms.

- **Self-Assembling Robots**
- **Self-Stabilization**
- **Dynamics**
- **Applications e.g. Multicore**
- **Distributed Algorithms**
- **Sublinear Estimators**
Distributed Algorithms: A Simple Example
How Many Nodes in Network?
How Many Nodes in Network?
How Many Nodes in Network?
How Many Nodes in Network?
How Many Nodes in Network?
How Many Nodes in Network?
How Many Nodes in Network?
With a simple flooding/echo process, a network can find the number of nodes in time $O(D)$, where $D$ is the diameter (size) of the network.
Diameter (Size) of Network?

- Distance between two nodes = Number of hops of shortest path
Diameter (Size) of Network?

- **Distance** between two nodes = Number of hops of shortest path
Diameter (Size) of Network?

- **Distance** between two nodes = Number of hops of shortest path
- **Diameter** of network = Maximum distance, between any two nodes
Networks Cannot Compute Their Diameter in Sublinear Time!

(even if diameter is just a small constant)
Networks Cannot Compute Their Diameter in Sublinear Time!

(even if diameter is just a small constant)
Networks Cannot Compute Their Diameter in Sublinear Time!

(even if diameter is just a small constant)
Networks Cannot Compute Their Diameter in Sublinear Time!

(even if diameter is just a small constant)
Networks Cannot Compute Their Diameter in Sublinear Time!

(even if diameter is just a small constant)
Networks Cannot Compute Their Diameter in Sublinear Time!

(even if diameter is just a small constant)
Networks Cannot Compute Their Diameter in Sublinear Time!

(even if diameter is just a small constant)
Networks Cannot Compute Their Diameter in Sublinear Time!

(even if diameter is just a small constant)
Networks Cannot Compute Their Diameter in Sublinear Time!

(even if diameter is just a small constant)

Pair of nodes not connected on both sides? We have $\Theta(n^2)$ information that has to be transmitted over $O(n)$ edges, which takes $\Omega(n)$ time!

[Frischknecht, Holzer, W, 2012]
Summary

Self-Stabilization Distributed Algorithms Dynamics Self-Assembling Robots Applications e.g. Multicore

Sublinear Estimators

Distributed Algorithms

Robots

Mobile Networks

Networks

Parallelism

Self-Organization

Game Theory

BAR Games

Crypto

Agents

Network Mobility
Thank You!
Questions & Comments?

Thanks to my co-authors
Silvio Frischknecht
Stephan Holzer
Christoph Lenzen
Thomas Moscibroda
Thomas Locher
Philipp Sommer

www.disco.ethz.ch