Energy Costs for Computing

Energy for computing is an important problem

- Mobile/wireless devices
  - Trend toward smaller, lighter, more compact devices with significant computing resources
  - Limited improvements in battery life have not kept up

- Thermal issues
  - Constraining hardware development
  - Heat production affects cooling requirements and fan noise

- Demand for energy resources
  - Electricity costs
  - Environmental impact
Growing Electricity Use for Computing: Data for Servers and Data Centers

Source: EPA Report to Congress, Aug 07

The Dalles, OR

$4.5B
Growing Electricity Use for Computing: Data for Servers and Data Centers

Potential to do less harm

Source: EPA Report to Congress, Aug 07
Residential Electricity Consumption

Figure 1. Percent of Total Electricity Consumption in U.S. Housing Units, 2001

Water Heating 9.1
Space Heating 10.1
Air-Conditioning 16
Lighting 8.8
Other 42.2
All Others

Home computers 2
Color TVs 2.9
Furnace Fans 3.3
Freezers 3.5
Clothes Dryers 5.8

Potential of computing to contribute to energy efficiency for these other uses?
Electricity Consumption $\Rightarrow$ GHG Emissions

Source: Energy Information Admin, eia.doe.gov

Source: Environment Canada, ec.gc.ca
Understanding the Impacts

Environmental sensor networks

Computational support for science of global climate change
Outline

• Motivation for focus on reducing energy demand

• Three Dimensions
  – Do less harm: Energy efficiency for computing
    • Example: Milly Watt project and ECOSystem
  – Provide a benefit: Context-aware computing in residential energy management
    • Example: Smart House applications
  – Enable greater understanding: Sensor networks for environmental science
    • Example: Data-directed sensing in Duke forest

• Vision of a bigger role for computing research
Energy Management for Computing

<table>
<thead>
<tr>
<th>Unmodified Apps</th>
<th>Energy-Aware Apps</th>
</tr>
</thead>
<tbody>
<tr>
<td>OS &amp; System Software for Energy / Thermal Management</td>
<td></td>
</tr>
<tr>
<td>Power-Aware Computer Architecture</td>
<td></td>
</tr>
<tr>
<td>Low Power Circuit Design</td>
<td></td>
</tr>
</tbody>
</table>
## Energy Management for Computing

<table>
<thead>
<tr>
<th>Unmodified Apps</th>
<th>Energy-Aware Apps</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voltage Frequency Scheduling</td>
<td>Voltage Scaling</td>
</tr>
<tr>
<td>Caching Prefetching</td>
<td>DRAM power states</td>
</tr>
<tr>
<td>Page placement</td>
<td>Disk spin down</td>
</tr>
<tr>
<td>Power down policies</td>
<td>Power save radios</td>
</tr>
</tbody>
</table>

### Low Power Circuit Design
Milly Watt Project

<table>
<thead>
<tr>
<th>Unmodified Apps</th>
<th>FaceOff Display Mgt [HotOS03]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Energy Centric Operating System (ECOSystem) [ASPLOS02, USENIX03, IEEE Pervasive Computing05]</td>
<td></td>
</tr>
<tr>
<td>Power-Aware Memory &amp; DVS [ASPLOS00, ISLPED01, PACS02, PACS03]</td>
<td></td>
</tr>
<tr>
<td>Low Power Circuit Design</td>
<td></td>
</tr>
</tbody>
</table>
Prerequisites to Effective Energy Research

1) Choosing an appropriate metric
   - Energy (Joules)
   - Power (Watts)
   - Battery lifetime
   - MFLOPS/Watt MB/Watt transactions/Watt
   - Energy*delay
   - Energy, subject to QoS constraints (meeting deadlines)

   Does the goal include a justification for impact on performance?

2) Understanding how energy is being used – power model.

![Breakdown of Average Power Consumption under Loads](chart.png)

- Other
- Power Supply Loss
- WLAN
- Memory
- Graphics
- Display/backlight
- HDD
- DVD
- CPU
Explicit Energy Management by the Operating System

Energy is not *just another* resource
- Energy has a impact on every other resource of a computing system - a *first-class* resource

A focus on explicit energy management provides an opportunity to *rethink* OS design
- Affects every aspect of OS services and structure:
  - Interfaces needed by applications that want to affect power consumption
  - Internal organization and algorithms
  - Resource management policies and mechanisms
Initial Research Statement

How to
- to achieve energy-related goals (e.g. achieving a target battery lifetime)
- by the OS with off-the-shelf hardware
- without requiring applications to change
- with a whole-system perspective

⇒ Energy Centric Operating System
A Concrete Energy Goal: Battery Lifetime

1. Explicitly manage energy use to reach a target battery lifetime.
   • Coast-to-coast flight with your laptop
   • Sensors that need to operate through the night and recharge when the sun comes up

2. If that requires reducing workload demand, use energy in proportion to task’s importance.
   Scenario:
   • Revising and rehearsing a PowerPoint presentation
   • Spelling and grammar checking threads
   • Listening to MP3s in background
A Concrete Energy Goal: Battery Lifetime

3. Deliver good performance given constraints on energy availability
   - Fully utilize the battery capacity within the target battery lifetime with little leftover capacity – no lost opportunities.
   - Encourage efficiency in performing desired work.
   - Address observed performance problems (e.g. energy-based priority inversions).
Energy Centric Operating System (ECOSystem)

1. Energy can serve as a unifying concept for managing a diverse set of resources.
   - We introduce the currency abstraction to represent the energy resource (and its global impact on the system).

2. A framework is needed for explicit monitoring and management of energy.
   - We develop mechanisms for currency accounting, currency allocation, and scheduling of currency use.

3. We need policies to achieve energy goals.
   - Need to arbitrate among competing demands and reduce demand when energy is limited (and when applications themselves don’t know how).
Unified Currentcy Model

Energy accounting and allocation are expressed in a common currentcy. Abstraction for

1. Characterizing power costs of accessing resources
2. Controlling overall energy consumption
3. Sharing among competing tasks
Mechanisms in the Framework

Currentcy Allocation
• Epoch-based allocation – periodically distribute currentcy “allowance”

Currentcy Accounting
• Monitor system-wide energy behavior
• Attribute energy use to correct task
• Break down power costs by device

Basic policy
• Pay as you go for resource use
• No more currentcy ⇒ no more service.
Currentcy Flow

1. Determine *overall* amount of currentcy available per energy epoch.
2. Distribute available currentcy proportionally among tasks.
3. Deduct currentcy from task’s account for resource use.
ECOSystem Prototype

Modified Linux kernel 2.4.0-test9

- Interface for specifying input parameters (target lifetime, task proportions)
- New kernel thread for currentcy allocation
- Simple implementation of resource containers
- Simple policies for allocation, scheduling, and accounting.
ECOSystem Prototype

IBM Thinkpad T20 laptop platform
- Power model – calibrated by measurements
- 650MHz PIII CPU: 15.5W active
- Orinoco 802.11b PC card:
  - doze 0.045W, receive 0.925W, transmit 1.425W
- IBM Travelstar hard disk
- Base power consumption of 13W captures everything else and inactive states of above
# Hard Disk Power Model

<table>
<thead>
<tr>
<th>State</th>
<th>Cost</th>
<th>Timeout</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access</td>
<td>1.65 mJ</td>
<td></td>
</tr>
<tr>
<td>Idle1</td>
<td>1600 mW</td>
<td>0.5 s</td>
</tr>
<tr>
<td>Idle2</td>
<td>650 mW</td>
<td>2 s</td>
</tr>
<tr>
<td>Idle3</td>
<td>400 mW</td>
<td>27.5 s</td>
</tr>
<tr>
<td>Standby</td>
<td>0 mW</td>
<td></td>
</tr>
<tr>
<td>Spinup</td>
<td>6000 mJ</td>
<td></td>
</tr>
<tr>
<td>Spindown</td>
<td>6000 mJ</td>
<td></td>
</tr>
</tbody>
</table>
The Policy Space

• Base policies
  – CPU: hybrid of sampling and task switch accounting
  – Disk: tasks directly pay for file accesses, sharing of spinup & spindown costs.
  – Network: source or destination task pays based on length of data transferred

• Adv. policies dealing with
  – Mismatches between user-supplied specifications and actual needs of the task
  – Schedules not offering opportunities to fully spend allocation
  – I/O devices and other activity causing a form of inversion
Experimental Evaluation of Base Policies

- Validated the embedded energy model.
- Achieves a target battery lifetime.
- Achieves proportional energy usage among multiple tasks.
- Assess the performance impact of limiting energy availability.
  - Performance of compute bound task (e.g. ijpeg) scales proportionally with currentcy allocation
  - Some applications (e.g. netscape) don’t gracefully degrade with drastically reduced currentcy allocations
Problem: Mismatch between Shares and Needs

1. To fully utilize available battery capacity within the desired battery lifetime with little or no leftover (residual) capacity.

⇒ Devise an allocation policy that balances supply and demand among tasks. Currentcy conserving allocation.
Symptom: Residual Energy

Allocations do not reflect actual consumption needs
Symptom: Residual Energy

A task’s unspent currency (above a “cap”) is being thrown away to maintain steady battery discharge. ⇒ Leftover energy capacity at end of lifetime.
Currentcy Conserving Allocation

Two-step policy. Each epoch:
1. Adjust per-task caps to reflect observed need
   • Weighted average of currentcy used in previous epochs.
Currentcy Conserving Allocation

2. Redistribute overflow currentcy
Currentcy Conserving Allocation Experiment

Workload:

- Computationally intensive ijpeg – image encoder
- Image viewer, gqview, with think time of 10 seconds and images from disk
  - Performance levels out at 6500mW allocation.
- Total allocation of 12W, shares of 8W for gqview (too generous) and 4W for ijpeg (capable of 15.5W).
Currentcy Conserving Allocation Results

6.7% remaining

<1% remaining
Currentcy Conserving Allocation Results

<1% remaining capacity
Problem: Scheduler as a Gatekeeper

2. To produce more robust proportional sharing by ensuring adequate spending opportunities.

⇒ Develop CPU scheduling that considers energy expenditures on non-CPU resources.

**Currentcy-aware scheduling.**

⇒ Develop currentcy-aware scheduling for other devices.
Problem: Payday Syndrome

3. To reduce response time variability when energy is limited.

⇒ Design a scheduling policy that controls the pace of currentcy consumption.
Problem: Fostering Cooperation

4. To encourage greater energy efficiency (lower average cost) for I/O accesses on power-managed disks.

⇒ Amortize spinup and spindown costs over multiple disk requests by shaping request patterns to be more bursty.

Buffer management and prefetching strategies.
Discussion

• ECOSYSysterm is a powerful framework for managing energy explicitly as a first-class OS resource.
• Currentcy model is capable of formulating non-trivial energy goals and serving as the basis for solutions
• Lack of knowledge about applications limits possible energy goals and appropriate share settings (e.g. capturing “need”)
• Battery lifetime is not always the “right” goal
Outline

• Motivation for focus on reducing energy demand

• Three Dimensions
  – Do less harm: Energy efficiency for computing
    • Example: Milly Watt project and ECOSystem
  – Provide a benefit: Context-aware computing in residential energy management
    • Example: Smart House applications
  – Enable greater understanding: Sensor networks for environmental science
    • Example: Data-directed sensing in Duke forest

• Vision of a bigger role for computing research
Duke / Home Depot
Smart Home

• “Green” live-in lab (dorm) open to any Duke student who will advocate for sustainability and play with the new technologies.

• Computing to support energy conservation efforts in residential buildings.
  – Context-aware systems inferring user behaviors
  – Interfaces exposing energy use to users
Example: **Weakly Identifying System** for **Doorway Monitoring** (WISDOM)

- **Goal:** To count the number and direction of people passing through a doorway
  - Using low cost, unobtrusive sensors
  - People walking side-by-side and passing in opposite directions
- **Not intended for identification** – possibly verification of RFID badges (#tags = #people)
- **However,** it has been disconcerting how well individuals can be identified by body characteristics. Raises privacy concerns or unobtrusive tracking.
WISDOM Prototype

All measures in meters

- Sharp GP2Y0A02YK
  - Long Range (1.6m) Infrared Rangefinder
- Devantech SRF08
  - Ultrasonic Rangefinder
- Visonic Ltd. Clip-4
  - Motion Detector
- Sharp GP2D12
  - Short Range (0.8m) Infrared Rangefinder
- Optex AX-100S
  - Infrared Beam Detector
- Tapeswitch
  - Pressure Mat

Origin + Vertical Through + Lateral
Results of User Study: Accuracy of Count and Direction

- Subjects were asked to walk through doorway alone or as pair, with different paces, directions, and configurations.
- Sensing width of one or two people walking through doorway – measured distance from each side.

<table>
<thead>
<tr>
<th>Experiment</th>
<th>Passes</th>
<th>% Correct Direction</th>
<th>% Correct Count</th>
<th>% Missed</th>
<th>% Wrong</th>
</tr>
</thead>
<tbody>
<tr>
<td>One Slow Pace</td>
<td>64</td>
<td>100</td>
<td>100</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>One Normal Pace</td>
<td>68</td>
<td>100</td>
<td>98.5</td>
<td>0</td>
<td>1.5</td>
</tr>
<tr>
<td>One Fast Pace</td>
<td>72</td>
<td>100</td>
<td>100</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>One Left</td>
<td>68</td>
<td>100</td>
<td>100</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>One Right</td>
<td>70</td>
<td>100</td>
<td>98.6</td>
<td>0</td>
<td>1.4</td>
</tr>
<tr>
<td>Two Serial</td>
<td>32</td>
<td>100</td>
<td>100</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Two Abreast</td>
<td>94</td>
<td>100</td>
<td>100</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Two Opposite</td>
<td>62</td>
<td>100</td>
<td>100</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Total</td>
<td>530</td>
<td>100</td>
<td>99.6</td>
<td>0</td>
<td>0.4</td>
</tr>
</tbody>
</table>

Using 8 side infrared rangefinders only
Possible Applications

• Knowing how many warm bodies are in a conference room for more intelligent thermostat settings

• Automating “Last one out turn off the lights.”
My Own Smart House
(“Future work”)

Technology transfer from my recent research projects

• Centralized household monitoring & control center
• Context-aware geothermal radiant floor heating
• Soil moisture-based release of collected stormwater
Outline

• Motivation for focus on reducing energy demand
• Three Dimensions
  – Do less harm: Energy efficiency for computing
    • Example: Milly Watt project and ECOSystem
  – Provide a benefit: Context-aware computing in residential energy management
    • Example: Smart House applications
  – Enable greater understanding: Sensor networks for environmental science
    • Example: Data-directed sensing in Duke forest
• Vision of a bigger role for computing research
Example: Duke Forest Sensor Network

Goal is to predict biodiversity response to changes in climate, disturbance.
Forest Biodiversity and Tree Growth
Multiple Processes at Multiple Scales

Spatial scales in water transport

- **Xylem** water transport
- **Ecosystem** biosphere-atmosphere exchange
- **Landscape-atmosphere exchange**

**Scales**
- $10^0 \text{ m}^2$
- $10^1 \text{ m}^2$
- Tree 'global'

**Minutes to years**
(5-13 yrs)
WiSARD (Wireless Sensing and Relay Device)

Built-in probe interfaces
- 4 temperature channels - thermocouple
- 4 light intensity channels - photodiode
- 2 general purpose probe channels
  - Soil moisture - Decagon Ech2oprobe
  - Sap flow

Multihop Wireless Network

Power Management
- Monitor power status
- Report battery voltage
- Budget
  - Sending .052A
  - Listening/Receiving .033A
  - Processing .007A
Weather Station Peripheral

- Vaisala smart probe for above-canopy data
  - Wind Speed and Direction
    - ultrasonic vector anemometer
  - Liquid/Hail Precipitation
    - piezoelectric impact detector
  - Relative Humidity/VPD
  - Temperature
  - Barometric Pressure

- Solar Panel
Multiple Environmental Variables Affecting Tree Growth
Ecologist Goals / Computer Scientist Goals

• Networks / data sets will serve many users

• Value of an observation varies in space, time, & among models
  
  ⇒ Collect all data (to within some precision)

• Costs of acquiring and delivering an observation affects the lifetime of the nodes/network or the amount of maintenance required (battery replacement)

  ⇒ Adaptively sample and send (informed by models)
Example: Soil Moisture Data Observations

winter summer

precipitation runoff

transpiration wilting
Ecologist Goals / Computer Scientist Goals

• Soil moisture model to fit the data
  \[
  \frac{dW}{dt} = P - T(\text{light}, \text{VPD}, W) - Dr(W, \text{topo})
  \]
  = Precip - Evapotrans - Drainage
  
  – transpiration response vs soil moisture available to a tree

• In-network control - adapting the capture & delivery of data stream to save battery lifetime.
  
  – Which data to suppress?

Models can’t substitute for actual readings when we still learning about the physical process being studied.

The correctness of data collection should not depend on correctness of models.
Example: Soil Moisture Data Observations

winter summer

precipitation runoff
transpiration wilting
Data-driven Suppression

Exploit correlation in data & put smarts in network

Base station

Model \( p(X^{(t)}|o^{(t-1)}, o^{(t-2)}, \ldots) \)

Values transmitted at time \( t-1 \)

Sensor network

Model \( p(X^{(t)}|o^{(t-1)}, o^{(t-2)}, \ldots) \)

Transmit \( o^{(t)} \) such that

\[
|X^{(t)} - E(X^{(t)}|o^{(t)}, o^{(t-1)}, \ldots)| \leq \epsilon
\]

Compare actual reading \( x^{(t)} \) with model prediction \( E(X^{(t)}|o^{(t-1)}, o^{(t-2)}, \ldots) \)

Differ by more than \( \epsilon \)?

Regardless of model quality, base station knows \( x^{(t)} \) to within \( \epsilon \)

Better model \( \Rightarrow \) fewer transmissions
Temporal Suppression

• Suppress transmission if $|\text{current reading} - \text{last transmitted reading}| \leq \varepsilon$
  – Model: $X(t) = x(t-1)$

Effective when readings change slowly
Failure and Suppression

• Message failure common in sensor networks
  – Interference, obstacles, congestion, etc.

• Is a non-report due to suppression or failure?
  – Without additional information/assumption, base station has to treat every non-report as plain “missing”—no accuracy bounds!
BaySail (Bayesian Analysis of Suppression and Failure)

- Inference with redundancy and knowledge of suppression scheme [Silberstein et al., VLDB 2007]
- At app level, piggyback redundancy on each report
  - **Counter**: number of reports to base station thus far
  - **Timestamps**: last $r$ timesteps when node reported
  - **Timestamps+Direction Bits**: in addition to the last $r$ reporting timesteps, bits indicating whether each report is caused by (actual – predicted > $\varepsilon$) or (predicted – actual > $\varepsilon$)
Suppression-aware inference

Redundancy + knowledge of suppression scheme $\Rightarrow$ hard constraints on $X_{\text{mis}}$

- Temporal suppression with $\epsilon = 0.3$, prediction = last reported
- Actual: $(x_1, x_2, x_3, x_4) = (2.5/\text{sent}, 3.5/\text{sent}, 3.7/\text{suppressed}, 2.7/\text{sent})$
- Base station receives: $(2.5, \text{nothing}, \text{nothing}, 2.7)$
- With *Timestamps* ($r=1$)
  - $(2.5, \text{failed, suppressed}, 2.7)$
  - $|x_2 - 2.5| > 0.3; |x_3 - x_2| \leq 0.3; |2.7 - x_2| > 0.3$
- With *Timestamps+Direction Bits* ($r=1$)
  - $(2.5, \text{failed \& under-predicted, suppressed, 2.7 \& over-predicted})$
  - $x_2 - 2.5 > 0.3; -0.3 \leq x_3 - x_2 \leq 0.3; x_2 - 2.7 > 0.3$
- With *Counter*
  - One suppression and one failure in $x_2$ and $x_3$; not sure which
Discussion

• **Benefit**: how much uncertainty it helps to remove
  – *Counter* can cover long periods, but helps very little in bounding particular values

• **Energy cost**
  – *Counter* < *Timestamps* < *Timestamps* + *Direction Bits*

• **Complexity of in-network implementation**
  – Coding app-level redundancy in TinyOS was much easier than finding the right parameters to tune for ACK/Retransmit!

• **Cost of out-of-network inference**
  – May be significant even with powerful base stations!
Outline

• Motivation for focus on reducing energy demand
• Three Dimensions
  – Do less harm: Energy efficiency for computing
    • Example: Milly Watt project and ECOSystem
  – Provide a benefit: Context-aware computing in residential energy management
    • Example: Smart House applications
  – Enable greater understanding: Sensor networks for environmental science
    • Example: Data-directed sensing in Duke forest
• Vision of a bigger role for computing research
Opportunities!

• Any subdiscipline of computer science can find research topics related to energy efficiency (if only adopting it as a new metric).

• Inter-disciplinary research with scientists & engineers dealing with environmental and energy problems.
Computing Research toward Greener Computing

• Low power / low energy computing systems
  – Broader systems context for energy management
  – Develop energy metrics / measurement expertise / tools
  – Energy-aware applications and algorithms
  – Software engineering to develop energy-aware apps

• Improved lifecycle: reduce/reuse/recycle
  – Incentive systems to encourage energy-motivated resource sharing
    • P2P to achieve more efficient utilization of existing unused capacity
Role for Computing Research in Energy Management

• Managing energy distribution systems
  – Microgrids: peer-to-peer power generation

• Supporting energy conservation efforts in buildings, transportation systems, manufacturing processes, etc.
  – Interfaces exposing energy use to users
  – Pervasive, context-aware systems

• Collaboration applications for more effective teleconferencing / telecommuting
Role for Computing Research in Climate Science

- Deployment of sensor networks designed specifically for environmental monitoring
  - Harvesting energy in the field
- Large-scale scientific modeling and simulations
- Management of huge data sets
- Application-specific tools
- Visualization
Acknowledgements

Milly Watt Group
• Alvy Lebeck
• Amin Vahdat (UCSD)
• Jam Jenkins, Ph.D.
• Rebecca Braynard, Ph.D.
• Angela Dalton, Ph.D.
• Xiaobo Fan, Ph.D.
• Heng Zeng, Ph.D.
• Shobana Ravi, M.S.

DDDAS Group
• Jim Clark (Biology)
• Alan Gelfand (Statistics)
• Pankaj Agarwal, Kamesh Munagala,, Jun Yang (CS)
• Paul Flikkema (NAU)
• Adam Silberstein, Ph.D
• Rebecca Braynard,Ph.D.
• David Bell, Gavino Puggioni (current students)
• Improved lifecycle: reduce/reuse/recycle
  – Incentive systems to encourage energy-motivated resource sharing
  • P2P to achieve more efficient utilization of existing unused capacity

Thank you!