**Remote Execution to Save Local Energy**

Compute locally on battery power

vs

Transmit to wired compute server +
Idle in low power mode + Receive results

- Why and when it works?
- How? The mechanisms required.

**Effectiveness?**

(Rudenko et al)

- Identical, dedicated laptops
- Wireless WaveLAN send 3W, receive 1.48 W, sleep .18W, 2MB/s
- Li ion battery power, measured by APM metric
- No interference
- Display and disk timeouts of 1 min

**Compilation**

- Both have replicas of source code to be compiled
- Sends back stripped executables
- Send modifications only

**Compilation Results**

% Battery consumed

0 125 250 375 500

KB of changed code
Gaussian Solution of System of Equations

- Little I/O, little VM paging
- Entire matrix shipped
- Solution vector returned

Gaussian Results

<table>
<thead>
<tr>
<th>Size of Matrix</th>
<th>500x500</th>
<th>700x700</th>
<th>800x800</th>
<th>900x900</th>
<th>1000x1000</th>
</tr>
</thead>
<tbody>
<tr>
<td>% Battery consumed</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>local</td>
<td>0.5</td>
<td>0.6</td>
<td>0.7</td>
<td>0.8</td>
<td>0.9</td>
</tr>
<tr>
<td>remote</td>
<td>0.3</td>
<td>0.4</td>
<td>0.5</td>
<td>0.6</td>
<td>0.7</td>
</tr>
</tbody>
</table>

Latex

- Both have replicas of source to be formatted
- Sends back formatted document

Latex Results

<table>
<thead>
<tr>
<th>KB of altered text</th>
<th>36</th>
<th>45</th>
<th>127</th>
<th>439</th>
<th>845</th>
</tr>
</thead>
<tbody>
<tr>
<td>% Battery consumed</td>
<td>0.1</td>
<td>0.2</td>
<td>0.3</td>
<td>0.4</td>
<td>0.5</td>
</tr>
</tbody>
</table>
With Noise

A second pair of machines with saturated socket noise resulting in significant backoff and retransmit.

CRA-W Summer Project ‘99

Transmission: 150mW (Palm) + 387 mW (modem) 14.4Kbps

Computation: ave. 130mW

- D’Agents server
- Fast server machine (but interpretation of Tcl)
- Tcl agents “parked” on the Palm; PalmOS C program of similar functionality
- PilotLauncher - to ship code and data

Results (constant message size)

Results (data shipped)
Energy Aware Adaptation
(Flinn and Satya - SOSP99)

- Odyssey - system for adaptation
- Fidelity - the degree to which delivered data matches the reference copy at the server – type-specific notion
- Question: Can lowering fidelity be used as a mechanism for energy-aware adaptation? Enough savings possible?

Odyssey Architecture

- Monitors resource availability
- Type specific

Odyssey Speech Recognizer

- Local, remote, & hybrid (compressed source)
- Fidelity: reduced vocabulary database – tradeoff: fewer words but fewer mistakes

Energy of Speech Recognition

- Baseline - local, full fidelity, no HW power mgt.
- HW-only - display off, disk spun-down, network off
- Remote - most of the energy spent in idle mode on portable
Conclusions

- Significant variation in effectiveness of fidelity reduction across data objects
- Significant variation in effectiveness of fidelity reduction across applications
- Reducing fidelity can enhance effectiveness of HW power management (by lowering utilization of HW)

How?

- Mixed results as to the effectiveness of remote execution for energy savings. How can “right” choices be made?
- What aspects could benefit from more energy efficient implementation of the support systems themselves?
  - D’agents system used interpretation on the server vs. direction execution on portable.
  - Implementation of “Idle”
- What are the basic building blocks?

Infrastructure Requirements (Rudenko et al)

- Ability to do remote execution, shipping arguments and results
  - Energy efficient listening on portable
- Replication mechanisms and synchronization for data and code.
- Providing consistent execution environment
- Decision strategy for choosing between local and remote (UI or automagically?)