**An Empirical Study on the Impact of Deep Parameters on Mobile App Energy Usage**

Qi Qiang Xu, James C. Davis, and Y. Charlie Hu  
Purdue University

**Motivation**
- Parameter tuning in conventional software is shown to improve the performance.
- Mobile apps have hundreds of deep parameters that are scattered around the source code.
- Little is known about the energy impacts of deep parameters in mobile apps.
- Prior works only studied deep parameters in specific modules or libraries, not systematically.

**Deep Parameter**
A constant in app source code that can be changed by app developers, while the app still functions properly with minimal impact on user experience.

- buffer size, timeout, UI element size, ...
- socket.setSocketTimeout(0);
- error code, loop initializer, ...
- for (int i = 0; ...)

**Developer Perspectives**
25 Android developers, on average 6-10 years of software development experience, 3-5 years of Android development experience, 19 questions.

For what proportion of the parameters in your app are you confident about the energy impact of changing them?

<table>
<thead>
<tr>
<th>None</th>
<th>A Few</th>
<th>A Lot</th>
<th>Unsure</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>3</td>
<td>9</td>
<td>12</td>
</tr>
<tr>
<td>15</td>
<td>18</td>
<td>21</td>
<td>24</td>
</tr>
</tbody>
</table>

Few developers (12%) are confident about the energy impact of parameters. Possibly due to the lack of handy tools for energy tuning.

**Deep Parameter Testing Framework**

- Parameter Extraction
- Deep Parameters
- Parameter Mutation
- Automated Test
- Energy Reductions
- Manual Validation
- Suboptimal Parameterizations

**Experimental Results**
16 popular open-source apps from 16 categories, one 30-60s test scenario for each app.

1644 deep parameters tested, 28 observed energy reduction, 2 manually validated to reduce energy drain without breaking app functionality.

**True positives**
- Prefetching size in Reddit client, optimal value depends on the scenario.
- Ping interval in P2P browser to multiple peers; reducing from every 1s to every 8s saves 12% energy.

**False positives**
- Broken app functionality (13/26)
- Test stochasticity (12/26)

**Conclusion**
- **Survey:** developers are uncertain about, and largely ignore the energy impact of deep parameters.
- **Experiment:** single-parameter–induced energy inefficiency is uncommon.
- **Takeaway:** developers can safely ignore the energy impact when choosing deep parameter values for now.
- **Future work:** interactions between deep parameters.

**Survey framework, results:**

**Survey framework, results:** https://doi.org/10.5281/zenodo.5823364

**An Empirical Study on the Impact of Deep Parameters on Mobile App Energy Usage**

Qi Qiang Xu, James C. Davis, and Y. Charlie Hu  
Purdue University

**Motivation**
- Parameter tuning in conventional software is shown to improve the performance.
- Mobile apps have hundreds of deep parameters that are scattered around the source code.
- Little is known about the energy impacts of deep parameters in mobile apps.
- Prior works only studied deep parameters in specific modules or libraries, not systematically.

**Deep Parameter**
A constant in app source code that can be changed by app developers, while the app still functions properly with minimal impact on user experience.

- buffer size, timeout, UI element size, ...
- socket.setSocketTimeout(0);
- error code, loop initializer, ...
- for (int i = 0; ...)

**Developer Perspectives**
25 Android developers, on average 6-10 years of software development experience, 3-5 years of Android development experience, 19 questions.

For what proportion of the parameters in your app are you confident about the energy impact of changing them?

<table>
<thead>
<tr>
<th>None</th>
<th>A Few</th>
<th>A Lot</th>
<th>Unsure</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>3</td>
<td>9</td>
<td>12</td>
</tr>
<tr>
<td>15</td>
<td>18</td>
<td>21</td>
<td>24</td>
</tr>
</tbody>
</table>

Few developers (12%) are confident about the energy impact of parameters. Possibly due to the lack of handy tools for energy tuning.

**Deep Parameter Testing Framework**

- Parameter Extraction
- Deep Parameters
- Parameter Mutation
- Automated Test
- Energy Reductions
- Manual Validation
- Suboptimal Parameterizations

**Experimental Results**
16 popular open-source apps from 16 categories, one 30-60s test scenario for each app.

1644 deep parameters tested, 28 observed energy reduction, 2 manually validated to reduce energy drain without breaking app functionality.

**True positives**
- Prefetching size in Reddit client, optimal value depends on the scenario.
- Ping interval in P2P browser to multiple peers; reducing from every 1s to every 8s saves 12% energy.

**False positives**
- Broken app functionality (13/26)
- Test stochasticity (12/26)

**Conclusion**
- **Survey:** developers are uncertain about, and largely ignore the energy impact of deep parameters.
- **Experiment:** single-parameter–induced energy inefficiency is uncommon.
- **Takeaway:** developers can safely ignore the energy impact when choosing deep parameter values for now.
- **Future work:** interactions between deep parameters.

**Survey framework, results:**

**Survey framework, results:** https://doi.org/10.5281/zenodo.5823364