AxGAMES: Towards Crowdsourcing Quality Target Determination in Approximate Computing

Jongse Park, Emmanuel Amaro, Divya Mahajan, Bradley Thwaites, Hadi Esmaeilzadeh

Alternative Computing Technologies (ACT) Lab
Georgia Institute of Technology

ASPLOS 2016
Approximate computing
Embracing imprecision

Relax the abstraction of “near perfect” accuracy in

Accept imprecision to improve performance energy efficiency
Tradeoff b/w quality and benefits

Approximation

Benefits

Lowest quality (random)

Output Quality

Perfect quality (non-approximated)
Tradeoff b/w quality and benefits

Approximation

Benefits

Unacceptable Quality

Acceptable Quality

Lowest quality (random)

Output Quality

Perfect quality (non-approximated)
Tradeoff b/w quality and benefits

Approximation

Benefits

Unacceptable Quality

Quality Target

Acceptable Quality

Lowest quality (random)

Output Quality

Perfect quality (non-approximated)
Acceptable quality is

- Subjective
- Input-data dependent
- Application specific
- Approximation technique specific
Acceptable quality
Subjective

100% 90% 80% 70%

High Low

Output Quality
Acceptable quality
Subjective

Which has the lowest acceptable quality?
Acceptable quality
Subjective

Which has the lowest acceptable quality?
Acceptable quality
Subjective

Which has the lowest acceptable quality?
Acceptable quality
Input data dependent

Leopard

Orca

High | 100% | 90% | 80% | 70% | Low

Output Quality
Acceptable quality
Application specific

sobel

jpeg

High 100% 90% 80% 70% Low

Output Quality
Acceptable quality
Approximation technique specific

Technique A

Technique B

Output Quality

High 100% 90% 80% 70% Low
AxGAMES
Transforming the tradeoff in approximate computing

Approximation Benefits

Output Quality

AxGAMES

Approximation Benefits

% Users Satisfied
AXGAMES: systematic and general framework

Subjective
Input data dependent
Application specific
Approx. technique specific

Approximate Program
Approximation knob
AXGAMES

Approximation
Benefits

% Users Satisfied
AXGAMES
A systematic solution for quality target determination
AXGAMES
A systematic solution for quality target determination
AxGAMES
A systematic solution for quality target determination

Approximate Program
Approximation Knob

Preprocessing Engine → Approximated Output DB → Games

Polluce Verso
Winabatt
Qna

Statistics Collection Engine → Crowd Response Analyzer

Input datasets

Crowd
AXGAMES
A systematic solution for quality target determination

Approximate Program
Approximation Knob

Preprocessing Engine
Approximated Output DB

Games
POLLICE VERSO
WINABATT
QNA

Statistics Collection Engine
Crowd Response Analyzer

Input datasets
Crowd
AXGAMES
A systematic solution for quality target determination

Approximate Program
Approximation Knob

Preprocessing Engine → Approximated Output DB

Games
POLLICE VERSO
WINABATT
QNA

Statistics Collection Engine → Crowd Response Analyzer

Crowd
Input datasets
Three games
Research questions to be answered
Three games
Research questions to be answered

1. How much quality loss would the crowd accept?
Three games
Research questions to be answered

1. How much quality loss would the crowd accept?

2. How much quality loss would the crowd accept when quality-cost tradeoff is considered?
Three games
Research questions to be answered

1. How much quality loss would the crowd accept?

2. How much quality loss would the crowd accept when quality-cost tradeoff is considered?

3. How much quality loss would the crowd accept when quality-cost tradeoff and context of application are considered?
Let’s play!
AXGAMES
A systematic solution for quality target determination

Approximate Program
Preprocessing Engine
Approximated Output DB
Games
POLLICE VERSO WINABATT QNA
Statistics Collection Engine
Crowd Response Analyzer
Approximation Knob
Input datasets
Crowd
Crowd response analyzer
Statistical analysis

Quality \( q \) (Lowest < \( q < \) Highest)

Statistical Analysis

% Users Satisfied
Crowd response analyzer
Statistical analysis

Binomial Proportion Confidence Interval (Clopper-Pearson Exact Method)

\[
(n_{\text{trials}}, n_{\text{success}}) \rightarrow \frac{1}{1 + \frac{(n_{\text{trials}} - n_{\text{success}} + 1)}{n_{\text{success}} \times F[1 - \alpha; 2n_{\text{success}}, 2(n_{\text{trials}} - n_{\text{success}} + 1)]}} < \text{SuccessRate}
\]

with a confidence level
Crowd response analyzer
Statistical analysis

Binomial Proportion Confidence Interval (Clopper-Pearson Exact Method)

\[
(n_{\text{trials}}, n_{\text{success}}) \quad \rightarrow \quad r < \text{SuccessRate}
\]

with a confidence level
Crowd response analyzer

Statistical analysis

Binomial Proportion Confidence Interval (Clopper-Pearson Exact Method)

\[(n_{\text{trials}}, n_{\text{success}}) \rightarrow r < \text{SuccessRate} \text{ with a confidence level}\]

E.g., (100, 80) \[\rightarrow\] 72.28% < \text{SuccessRate} \text{ with 95% confidence level}\]
Crowd response analyzer
Statistical analysis

Quality $q$

$(n_{\text{Votes}}, n_{\text{GoodEnough}})$

$r < \% \text{ Users Satisfied}$
Crowd response analyzer
Statistical analysis

Quality $q$

$(n_{Votes}, n_{GoodEnough})$

$70\%$

$(1000, 634)$

$r < \% Users Satisfied$

$60.82\% < \% Users Satisfied$
## Benchmark

<table>
<thead>
<tr>
<th>Benchmarks</th>
<th>Description</th>
<th>Quality Metric</th>
</tr>
</thead>
<tbody>
<tr>
<td>emboss</td>
<td>Embossing filter</td>
<td>Normalized Root Mean Square Error (NRMSE)</td>
</tr>
<tr>
<td>jpeg</td>
<td>Lossy compression</td>
<td></td>
</tr>
<tr>
<td>mean</td>
<td>Blurring filter</td>
<td></td>
</tr>
<tr>
<td>sobel</td>
<td>Edge detection</td>
<td></td>
</tr>
<tr>
<td>audio-enc</td>
<td>Audio encoder</td>
<td></td>
</tr>
<tr>
<td>ocr</td>
<td>Optical character recognition</td>
<td></td>
</tr>
<tr>
<td>speech2txt</td>
<td>Embossing filter</td>
<td></td>
</tr>
</tbody>
</table>
Crowd recruitment

700 Turkers for 7 benchmarks
30 rounds per player
(10 rounds per game)
Acceptable quality loss for applications/games

- 99% of Satisfied Users
- 90% of Satisfied Users
- 80% of Satisfied Users

<table>
<thead>
<tr>
<th>Application</th>
<th>Acceptable Level of Quality Loss</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pollice Verso</td>
<td>0% mean, 0% audio-enc, 0% ocr</td>
</tr>
<tr>
<td>WinABatt</td>
<td>10% mean, 10% audio-enc, 10% ocr</td>
</tr>
<tr>
<td>QnA</td>
<td>20% mean, 20% audio-enc, 20% ocr</td>
</tr>
<tr>
<td>Pollice Verso</td>
<td>30% mean, 30% audio-enc, 30% ocr</td>
</tr>
<tr>
<td>WinABatt</td>
<td>40% mean, 40% audio-enc, 40% ocr</td>
</tr>
<tr>
<td>QnA</td>
<td>50% mean, 50% audio-enc, 50% ocr</td>
</tr>
</tbody>
</table>
Acceptable quality loss for applications/games

- **99% of Satisfied Users**
- **90% of Satisfied Users**
- **80% of Satisfied Users**

Projected Acceptable Level of Quality Loss

<table>
<thead>
<tr>
<th>App</th>
<th>0%</th>
<th>10%</th>
<th>20%</th>
<th>30%</th>
<th>40%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pollice Verso</td>
<td>mean</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WinABatt</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>QnA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pollice Verso</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WinABatt</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>QnA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pollice Verso</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WinABatt</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>QnA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>QnA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

36
Acceptable quality loss for applications/games

- 99% of Satisfied Users
- 90% of Satisfied Users
- 80% of Satisfied Users

Projected Acceptable Level of Quality Loss

<table>
<thead>
<tr>
<th></th>
<th>Pollice Verso</th>
<th>WinABatt</th>
<th>QnA</th>
<th>Pollice Verso</th>
<th>WinABatt</th>
<th>QnA</th>
<th>Pollice Verso</th>
<th>WinABatt</th>
<th>QnA</th>
</tr>
</thead>
<tbody>
<tr>
<td>mean</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>audio-enc</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ocr</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- 2% of Satisfied Users
- 26% of Satisfied Users

37
Different patterns for different domains

Projected fraction of satisfied users with 95% confidence level

- **Image Processing (mean)**
- **Audio Processing (audio-enc)**
- **Text Processing (ocr)**

Output Quality Loss
(statistics collected from the QNA game)
Tradeoff change in approximate computing

Example: mean

Tradeoff change from

quality vs. benefits to user satisfaction vs. benefits

<table>
<thead>
<tr>
<th>Output Quality</th>
<th>Improvement in Energy X Delay</th>
</tr>
</thead>
<tbody>
<tr>
<td>80%</td>
<td>17</td>
</tr>
<tr>
<td>85%</td>
<td>13</td>
</tr>
<tr>
<td>90%</td>
<td>9</td>
</tr>
<tr>
<td>95%</td>
<td>5</td>
</tr>
<tr>
<td>100%</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>% Users Satisfied</th>
<th>Improvement in Energy X Delay</th>
</tr>
</thead>
<tbody>
<tr>
<td>80%</td>
<td>17</td>
</tr>
<tr>
<td>85%</td>
<td>13</td>
</tr>
<tr>
<td>90%</td>
<td>9</td>
</tr>
<tr>
<td>95%</td>
<td>5</td>
</tr>
<tr>
<td>100%</td>
<td>1</td>
</tr>
</tbody>
</table>
AxGAMES

Approximation Benefits
Output Quality

Approximation Benefits
% Users Satisfied

http://act-lab.org/artifacts/axgames/
https://bitbucket.org/act-lab/game.code