Arash Deshmeh, Jacob Machina, and Angela C. Sodan
University of Windsor, Canada

ADEPT Scalability Predictor in Support of Adaptive Resource Allocation

IPDPS 2010
Outline

- Background: Adaptive Resource Allocation
- Related Work
- Downey Runtime/Speedup Model
- The ADEPT Predictor
- Experimental Results
- Anomaly Detection
- Automated Reliability Judgment
- Summary and Conclusion
Background: Adaptive Resource Allocation

- Adaptive resource allocation:
  - Up to 70% improvement in avg. response times by
    - Reducing fragmentation
    - Adapting to current load (low/high)
  - 98% of applications said to be moldable

Requires knowing jobs’ scalability / efficiency
  but not practically available yet
  In fact, it is a response-time function in dependence on CPU/core resources (Burton Smith)
Illustration of Adaptive Resource Allocation

Fragmentation reduction

Adaptation to current load

- Run at higher efficiency with smaller sizes if high load
- Run at lower efficiency with larger sizes of low load
More Background

- **Benefits for user:**
  - Help in choosing job sizes tactically
  - Determine maximum meaningful job sizes
    → our data about real applications

- **Relevance for resource allocation in:**
  - Clusters (MPI jobs)
  - SMPs (OpenMP or MPI jobs)
  - Virtual-machine resource provisioning
Related Work

- Most approaches are white-box (detailed model)
  - Require tools: code instrumentation, compiler/OS support, analysis of memory-access behavior, etc.
  - Complex and computationally expensive
  - Unsuitable for large-scale use in HPC centers
  - Valuable for cross-site or new-platform performance projection

- Black-box approaches (few observ. points, simple model)
  - Easy-to-use and cheap
  - Suffer from anomalies or non-uniform scalability patterns
Goals of ADEPT Scalability Predictor

- **Goals of ADEPT**
  - Achieve high prediction accuracy
  - Provide computationally efficient approach
  - Detect and automatically correct individual anomalies
  - Detect and model non-uniform patterns (multi-phase)
  - Perform reliability judgment with potential advice for outcome improvement

- Apply black-box prediction
- Based on Downey runtime/speedup model
## Downey Model

<table>
<thead>
<tr>
<th>Mode</th>
<th>$n$ range</th>
<th>$S(n)$</th>
<th>$T(n)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low variance</td>
<td>$1 \leq n \leq A$</td>
<td>$An / (A+(\sigma/2)(n-1))$</td>
<td>$(A-\sigma/2)/n + \sigma/2$</td>
</tr>
<tr>
<td></td>
<td>$A \leq n \leq 2A-1$</td>
<td>$An / (\sigma(A-1/2+n(1-\sigma/2))$</td>
<td>$\sigma(A-1/2)/n + 1 - \sigma/2$</td>
</tr>
<tr>
<td></td>
<td>$2A-1 \leq n$</td>
<td>$A$</td>
<td>$1$</td>
</tr>
<tr>
<td>High variance</td>
<td>$1 \leq n \leq A+A\sigma-\sigma$</td>
<td>$nA(\sigma+1) / (\sigma(n+A-1)+A)$</td>
<td>$\sigma + (A+A\sigma-\sigma)/n$</td>
</tr>
<tr>
<td></td>
<td>$A+A\sigma-\sigma \leq n$</td>
<td>$A$</td>
<td>$\sigma + 1$</td>
</tr>
</tbody>
</table>

- Simple (only $A$ and $\sigma$ to be learned)
- Needs few observation points
ADEPT Predictor

1. Anomaly detection and scalability-pattern identification
2. Envelope derivation
3. Curve fitting
4. Reliability judgment

Core of ADEPT
Core: Envelope Derivation

- Derives constraints from observations
- Calculates closed-form solutions (within certain percentage of deviation) from pairs of observations
- Use lowest and highest bounds as overall envelope
Core: Curve Fitting

- Prediction per target point, biased to closest observations
- Weighted least-squared relative errors
- Two-step
  1. Closest point fixed
  2. Extending variation by certain percentage within envelope
- Constraints from envelope and two-step curve fitting make ADEPT both accurate and fast
Experimental Set-Up

- Experiments with MPI and OpenMP
- NAS benchmarks BT, CG, FT, LU, SP
- 7 real anonymous applications
  (from administrator scalability tests)
- Both interpolation and extrapolation
- 3 to 4 input observation points
- Prediction of $T(n)$ and $S(n)$
- $T(1)$ not always available
Experimental Results: Speedup

- Applied fitting approach better than non-weighted
- Both interpolation and extrapolation work well
- Most extrapolation still good on twice the number of nodes
- Accuracy higher for closer extrapolation
Experimental Results: Runtime

Both interpolation and extrapolation work well
Whether $T(1)$ available or not did not make any difference
Some predictions perfect match (App_A, App_C, App_G)
Accuracy higher for closer extrapolation
ADEPT Predictor

1. Anomaly detection and scalability-pattern identification
2. Envelope derivation
3. Curve fitting
4. Reliability judgment

\{ \text{Core of ADEPT} \}
Anomaly Detection

- Serious deviations from model can be detected  
  (Application never fully conforms to model)
- Approach: fluctuation metric $R$
  $$R_i = ((t_i * n_i/n_{i+1})/t_{i+1})*(1+(n_{i+1}-n_i)/n_{i+1})$$  
  (idea is relative speedup, normalized to distance)  
  Check whether $R_{i+1} > (1+\epsilon)R_i$  
  with $\epsilon$ being sensitivity factor  
  both $R_{i+1}$ and $R_i$ are anomaly candidates
Individual Anomalous Points

- Minimum of 4 input points required
- Check $R$ curve after removal of anomaly candidate
- If improvement, classify as anomaly point and reduce its weight for curve fitting
Currently considered:

- Stepwise scalability (minimum of 5 points required)
  → Model instance per phase
- Specially optimized for certain numbers of nodes, e.g. powers of two (minimum of 9 points required), regular anomalous points
  → Omit other points from curve fitting
  → Report suitable allocations
ADEPT Predictor

1. Anomaly detection and scalability-pattern identification
2. Envelope derivation
3. Curve fitting

4. Reliability judgment

Core of ADEPT
Automated Reliability Judgment

- All input points in linear section
  → More input points needed ($n \geq A$)

- High fitting error, not explainable as anomaly
  → Report problem

- Runner-up problem (two or more model instances with greatly different $A$ match)
  → More input points needed (beyond current range)
All 3 cases (linear, high-fitting error, runner-up) successfully detected
Summary and Conclusion

- ADEPT is accurate and efficient
  - For both interpolation and extrapolation (if not too far away)
  - Works well without serial time $T(1)$
  - Performance similar to that reported in literature for white-box approaches

- Employs envelope derivation technique to constrain search during model fitting

- Biased model fitting with efficient two-level approach

- Anomaly detection based on fluctuation metric and automatic correction

- Warnings by reliability judgment if prediction uncertain

- Suitable for production environments
  - Extrapolative scalability prediction as feedback to users
  - Adaptive resource allocation