Energy-Efficient Resource Management for Cloud Computing Infrastructures

Marco Guazzone, Cosimo Anglano, Massimo Canonico

University of Piemonte Orientale
Alessandria, Italy

CloudCom’2011
Athens, Greece

November 30th, 2011
Outline

1. Motivation & Goal
2. Our Contribution
3. Experimental Evaluation
4. Conclusions
5. References
## Research Goals & Challenges

### Problem

Minimize **TCO** subject to **SLA** constraints

### Challenges

- Conflicting objectives
- Physical Resources and Applications heterogeneity
- Workload dynamics

### Goals

To **automatically** manage computing resources in order to

- **Satisfy** SLAs (as more as possible)
- **Reduce** TCO (in terms of energy consumption)
- **Adapt** to *dynamic, conflicting and distributed* environment
SLO metric: application-level response time
Shared physical resource: CPU
• **Goal**: to find the *best* tier CPU shares for the monitored application...
  - According to current operating conditions
  - In order to achieve SLOs
• One manager for each application
Adaptive Feedback Control with **Self-Tuning Regulation (STR)** scheme [1]
• **Goal**: *to model application dynamics*

• **We use** **black-box** models
  
  • Relationships between system **input** and system **output**
    
    • System input: **CPU shares**
    
    • System output: **tier mean residence times**

• **Discrete-time MIMO ARX model** [8]
Application Manager: System Parameters Estimation

Controller Design (LQR) → System Parameters Estimation (RLS) → System Model (ARX)

Specifications → Controller Design (LQR) → Controller

Reference Input → Controller Design (LQR) → Controller

Error → Controller Design (LQR) → Controller

Transduced Output → Transducer (EWMA) → Measured Output
• **Goal:** to identify ARX model structure and parameters

• We use **online system identification** to cope with time-varying workload
  - **Recursive Least-Squares (RLS)** algorithm to estimate system parameters at each control interval
  - We evaluated several variants and chose
    - RLS with **variable forgetting factor** [10]
Application Manager: Transducer

Controller Design (LQR)

System Parameters

System Parameters Estimation (RLS)

Controller

System Model (ARX)

Specifications

Reference Input

Error

Transduced Output

Control Input

Output

Energy-Efficient Resource Management for Cloud Computing Infrastructures

Guazzone - Anglano - Canonico
• **Goal:** *to filter system output* in order to
  - Remember past system behavior
  - Mitigate the effect of short peaks
  - Predict outputs during idle periods
• We use the **Exponentially Weighted Moving Average (EWMA)** filter

\[ S_k = \alpha X_k + (1 - \alpha) S_{k-1}, \quad 0 \leq \alpha \leq 1 \]

• Exponential decay of the weight of past outputs
• Smooth increments for short peaks
• Smooth decrements for idle periods
Goal: to design controller parameters
We use optimal control by means of the infinite-horizon discrete-time Linear Quadratic (LQ) control design
For each control interval, find the optimal state-feedback gain matrix which minimizes the cost function:

\[ J = J_s + J_c \]

where:
- \( J_s \): cost to keep system output near to its SLO value
- \( J_c \): cost to improve controller stability

We evaluated several variants and chose
- Linear Quadratic Regulator with Output Weighting [7]
Goal: to compute optimal tier CPU shares to achieve application SLOs

State-feedback control which computes the optimal control sequence from the LQ control design
• **Goal**: to arbitrate among conflicting CPU share demands.
  - Application Managers work independently from each other
  - The aggregated CPU share demand coming to each physical machine may exceed the maximum available
  - CPU shares are adjusted according to a given policy
  - One manager for each physical machine
• **Proportional** policy: for each control interval $k > 0$:
  - Let $n$ be the number of VMs hosted on a specific physical machine,
  - Let $D$, for $0 < D \leq 1$, be the maximum CPU share,
    - CPU shares are bounded in the $(0, D]$ real interval
  - Let $d_1(k), \ldots, d_n(k)$ be the incoming CPU share demands for the $n$ VMs,
  - The adjusted CPU share demands $\hat{d}_1(k), \ldots, \hat{d}_n(k)$ is computed as:
    $$\hat{d}_i(k) = \frac{d_i(k)}{\sum_{j=1}^{n} d_j(k)} D$$
Experimental Evaluation: Setup

- We have implemented a discrete-event simulator in C++
- Output analysis by means of the Independent Replications method
  - Performance indices:
    - Response Time
    - % SLO Violations
    - Energy Consumption
  - Replication length: at least $10^6$ succeeded requests
  - Number of replications: 95% confidence interval half length $\leq 4\%$
Experimental Evaluation: Setup (cont’d)

- Three **3-tier** applications
  - SLO metric: **0.99th quantile** of response time distribution
  - Per tier request service time:
    
    | App1  | [Det(0.060), Det(0.060), Det(0.060)] |
    |-------|--------------------------------------|
    | App2  | [Det(0.030), Det(0.060), Det(0.030)] |
    | App3  | [Det(0.015), Det(0.030), Det(0.060)] |

- Five **homogeneous** physical machines
  - CPU capacity: 2000
  - Energy model (Watt): \( E(u) = 143 + 258.2u + 117.2u^{0.355} \)

- VMs initial placement: **Best-fit**
  - Place each VM in the physical machine which leave the least amount of residual space
Four scenarios based on the type of the arrival process:

- **Behavioral pattern**: Deterministic Modulated Poisson Process (DMPP)
- **Self-similarity**: Pareto Modulated Poisson Process (PMPP)
- **Temporal burstiness**: Markov Modulated Poisson Process (MMPP)
- **Mix**: a mixture of the three types above

Three resource management approaches:

- **STATIC-SLO**: SLO-conserving approach
- **STATIC-ENERGY**: energy-conserving approach
- **OUR-APPROACH**: our solution
  - **No Migration Manager**
### Experimental Evaluation: Results

#### DMPP Scenario

<table>
<thead>
<tr>
<th>Approach</th>
<th>Application #1 % SLO violations</th>
<th>Application #2 % SLO violations</th>
<th>Application #3 % SLO violations</th>
<th>Power Consumption Watt</th>
<th>% Wasted Joules</th>
</tr>
</thead>
<tbody>
<tr>
<td>STATIC-SLO</td>
<td>0.67%</td>
<td>0.75%</td>
<td>0.78%</td>
<td>1043.59</td>
<td>0.73%</td>
</tr>
<tr>
<td>STATIC-ENERGY</td>
<td>19.19%</td>
<td>14.05%</td>
<td>19.40%</td>
<td>1013.04</td>
<td>17.68%</td>
</tr>
<tr>
<td>OUR-APPROACH</td>
<td>0.36%</td>
<td>0.49%</td>
<td>0.49%</td>
<td>1037.69</td>
<td>0.44%</td>
</tr>
</tbody>
</table>

#### PMPP Scenario

<table>
<thead>
<tr>
<th>Approach</th>
<th>Application #1 % SLO violations</th>
<th>Application #2 % SLO violations</th>
<th>Application #3 % SLO violations</th>
<th>Power Consumption Watt</th>
<th>% Wasted Joules</th>
</tr>
</thead>
<tbody>
<tr>
<td>STATIC-SLO</td>
<td>0.86%</td>
<td>0.78%</td>
<td>0.69%</td>
<td>1158.37</td>
<td>0.78%</td>
</tr>
<tr>
<td>STATIC-ENERGY</td>
<td>21.91%</td>
<td>17.41%</td>
<td>15.58%</td>
<td>1083.23</td>
<td>18.28%</td>
</tr>
<tr>
<td>OUR-APPROACH</td>
<td>0.88%</td>
<td>0.75%</td>
<td>0.59%</td>
<td>1150.11</td>
<td>0.75%</td>
</tr>
</tbody>
</table>
### MMPP Scenario

<table>
<thead>
<tr>
<th>Approach</th>
<th>Application #1</th>
<th>Application #2</th>
<th>Application #3</th>
<th>Power Consumption</th>
<th>% Wasted Joules</th>
</tr>
</thead>
<tbody>
<tr>
<td>STATIC-SLO</td>
<td>0.68%</td>
<td>0.76%</td>
<td>0.77%</td>
<td>1064.90</td>
<td>0.74%</td>
</tr>
<tr>
<td>STATIC-ENERGY</td>
<td>n/a</td>
<td>n/a</td>
<td>n/a</td>
<td>n/a</td>
<td>n/a</td>
</tr>
<tr>
<td>OUR-APPROACH</td>
<td>0.81%</td>
<td>0.77%</td>
<td>0.66%</td>
<td>1064.12</td>
<td>0.75%</td>
</tr>
</tbody>
</table>

### Mix Scenario

<table>
<thead>
<tr>
<th>Approach</th>
<th>Application #1</th>
<th>Application #2</th>
<th>Application #3</th>
<th>Power Consumption</th>
<th>% Wasted Joules</th>
</tr>
</thead>
<tbody>
<tr>
<td>STATIC-SLO</td>
<td>0.77%</td>
<td>0.77%</td>
<td>0.53%</td>
<td>1029.71</td>
<td>0.68%</td>
</tr>
<tr>
<td>STATIC-ENERGY</td>
<td>n/a</td>
<td>n/a</td>
<td>n/a</td>
<td>n/a</td>
<td>n/a</td>
</tr>
<tr>
<td>OUR-APPROACH</td>
<td>0.77%</td>
<td>0.76%</td>
<td>0.64%</td>
<td>1036.31</td>
<td>0.72%</td>
</tr>
</tbody>
</table>
Summary

• Effective administration of Cloud Infrastructure resources is challenging
• Our goal is to design an automatic and adaptive resource management for SLO satisfaction and TCO reduction
• Our approach is based on control-theoretic techniques
• Preliminary results show that by implementing smart resource management strategies it is possible to achieve good results in terms of both energy consumption and SLO preservation
Future Works

• Application Manager
  • Consider *minimum-variance* controllers
  • Evaluate other type of control design (e.g., PID)
  • Evaluate other type of system models (e.g., ARMAX)

• Migration Manager (WiP)
  • Optimization techniques
  • Approximated algorithms
  • Incremental VMs placement
Thank You!!
References

*Adaptive Control*.  

Exponential convergence of a modified directional forgetting identification algorithm.  

*Feedback Control of Computing Systems*.  

*Linear Systems Theory*.  
Designing controllable computer systems.

Tracking of slowly varying parameters by directional forgetting.

Linear Optimal Control Systems.
*System Identification: Theory for the User.*  

*Process Modelling, Identification, and Control.*  

Fast tracking rls algorithm using novel variable forgetting factor with unity zone.  

A comparison of high-level full-system power models.  
Additional Slides...
• Input-output relationship at control interval $k > 0$:
  • System input: CPU shares $s_i(k)$
  • System output: tier mean residence times $p_i(k)$
    • Application mean response time $p(k)$ is simply $p(k) = \sum_i p_i(k)$

• Issues:
  ① Nonlinear relationship between $p_i(k)$ and $s_i(k)$ [3]
  ② Different order of magnitude of $p_i(k)$ and $s_i(k)$ [5]

• Solution:
  ① Local linearization around an equilibrium point [4]
  ② Normalization
Relative deviations with respect to the equilibrium point \((\bar{s}_i, \bar{p}_i)\)

\[
\Delta \tilde{p}_i(k) = \frac{p_i(k) - \bar{p}_i}{\bar{p}_i}, \quad \text{(controlled variable)}
\]

\[
\Delta \tilde{s}_i(k) = \frac{s_i(k) - \bar{s}_i}{\bar{s}_i}, \quad \text{(control variable)}
\]
Discrete-time MIMO ARX model with structure \((n_a, n_b, n_k)\):

\[
\Delta \tilde{p}(k) + \sum_{j=1}^{n_a} A_j \Delta \tilde{p}(k - j) = \sum_{j=1}^{n_b} B_j \Delta \tilde{s}(k - j - n_k) + e(k)
\]

where:

- \(\Delta \tilde{p}(k) \in \mathbb{R}^m\) is the column vector of output relative deviations, at control interval \(k\)
- \(\Delta \tilde{s}(k) \in \mathbb{R}^m\) is the column vector of input relative deviations, at control interval \(k\)
- \(n_a, n_b, n_k\) are the number of poles, the number of zeros plus one, and the input delay, respectively
- \(A_1, \ldots, A_{n_a}\) and \(B_1, \ldots, B_{n_b}\) are the system parameters matrices with dimension \(\mathbb{R}^{m \times m}\)
- \(e(k) \in \mathbb{R}^m\) is the white noise column vector, at control interval \(k\)
Application Manager: System Parameters Estimation

- Identification of:
  - ARX model structure \((n_a, n_b, n_k)\)
  - ARX parameters \(A_1, \ldots, A_{n_a}\) and \(B_1, \ldots, B_{n_b}\)

- Offline system identification
  - Only used to infer the model structure
  - Inadequate to estimate system parameters
    - Unable to find a reasonable low-order model with a good fit

- Online system identification
  - Recursive Least-Squares (RLS) algorithm to estimate system parameters at each control interval
    - Exponential Forgetting [8]
    - Direction Forgetting (DF) [6]
    - DF + Bittanti’s correction [2]
    - Exponentially Weighted RLS (EWRLS) [10]
System output $\hat{p}_i(k)$ (at control interval $k$) is filtered by an Exponentially Weighted Moving Average (EWMA) filter:

$$p_i(k) = \alpha \hat{p}_i(k) + (1 - \alpha)p_i(k - 1)$$

- Smooth increments for short peaks
- Smooth decrements for idle periods
Application Manager: Controller Design

- Optimal control by means of the infinite-horizon discrete-time Linear Quadratic (LQ) control design:
  - Find the optimal state-feedback gain matrix $L$ which minimizes the cost function:
    \[
    J(u) = \sum_{k=0}^{\infty} \left( x^T(k)Qx(k) + u^T(k)Ru(k) + 2x^T(k)Nu(k) \right)
    \]

- Variants:
  - Linear Quadratic Regulator (LQR)
  - Linear Quadratic Regulator with Output Weighting (LQRY)
  - Linear Quadratic control with Integral Action (LQI)

**Note**

Need a state-space representation of the target system
From MIMO ARX to state-space MISO system representation:

\[ x(k + 1) = Ax(k) + Bu(k) \]
\[ y(k) = Cx(k) + Du(k) \]

such that:

\[
x(k) = \begin{pmatrix} \Delta \hat{p}(k - n_a + 1) \\ \vdots \\ \Delta \hat{p}(k) \end{pmatrix}, \quad u(k) = \begin{pmatrix} \Delta \hat{s}(k - n_b - n_k + 1) \\ \vdots \\ \Delta \hat{s}(k - n_k) \end{pmatrix}
\]

\[
A = \begin{pmatrix} Z & I & Z & \cdots & Z \\ Z & Z & I & \cdots & Z \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ Z & Z & \cdots & \cdots & \vdots \\ -A_{n_a} & -A_{n_a-1} & -A_{n_a-2} & \cdots & -A_1 \end{pmatrix}, \quad B = \begin{pmatrix} Z & \cdots & Z \\ \vdots & \vdots & \vdots \\ Z & \cdots & Z \\ B_{n_b} & \cdots & B_1 \end{pmatrix}
\]

\[
C = \begin{pmatrix} 0^T \\ \vdots \\ 0^T \\ 1^T \end{pmatrix}, \quad D = 0^T
\]
• State-feedback control which computes the optimal control sequence

\[ u^*(k) = -Lx(k) \]

which minimizes the LQ cost function

• The feedback gain matrix $L$ is obtained during the LQ design from the solution of the associated DARE