Adaptive Query Scheduling for Mixed Database Workloads with Multiple Objectives

Stefan Krompass\textsuperscript{TUM}, Harumi Kuno\textsuperscript{HPL}, Kevin Wilkinson\textsuperscript{HPL}, Umeshwar Dayal\textsuperscript{HPL}, Alfons Kemper\textsuperscript{TUM}

\textsuperscript{TUM}Technische Universität München  
Munich, Germany

\textsuperscript{HPL}Hewlett-Packard Laboratories  
Palo Alto, CA, USA
Problem statement

- $n$ service classes (i.e., a set of queries)
- $n \cdot m$ objectives (multiple objectives per service class)
- $n \cdot k$ control knobs (to control service per class, e.g., MPL)

Search problem

Find control knobs settings to achieve objectives for all service classes
Problem statement

- $n$ service classes (i.e., a set of queries)
- $n \cdot m$ objectives (multiple objectives per service class)
- $n \cdot k$ control knobs (to control service per class, e.g., MPL)

Search problem

Find control knobs settings to achieve objectives for all service classes
Difficulties

- **Large search space**
- Queries have different characteristics (resource requirements, variance in resource requirements)
- Service classes have different characteristics (start time, arrival rate, objectives)
- Contention among the queries unknown
- Non-linear relationships between objectives and the control parameters
Difficulties

- **Large search space**
- Queries have different characteristics (resource requirements, variance in resource requirements)
- Service classes have different characteristics (start time, arrival rate, objective)
- Contention among the queries unknown
- Non-linear relationships between objectives and the control parameters

In this presentation: Present framework and experiments with algorithm to tackle the search problem
Solution approach

- Base: algorithm devised by Niu et. al: “Adapting Mixed Workloads to Meet SLOs in Autonomic DBMSs”
  ⇒ Multi-class, single objective
- Extension: assume relationship between objectives is known in order to solve our problem
Workload Adaptation - Maximize Single Objective

- Goal: maximize overall utility (measure to quantify how well the system meets the objectives)
- Service classes $s_1, \ldots, s_n$, each with single objective
- Idea: assign system resources to service classes by controlling the number of queries a service class may run
- Service class $s_i$ has control knob $x_i$
- Assumption: $\exists$ “system cost limit” $X$ where performance is maximized
Workload Adaptation - Maximize Single Objective

\[
\text{maximize } u_1 \left( h_1 (x_1) \right) + \cdots + u_n \left( h_n (x_n) \right)
\]

subject to \( x_1 + \cdots + x_n = X \)

- **Estimation model:** control knob setting \( (x_i) \rightarrow \) estimated performance
- **Utility function:** performance value \( \rightarrow \) utility (positive if performance > objective, negative otherwise; utility decrease faster for lower performance, utility increase slower with better performance)
Dominance

Definition

Objective $o$ is *dominant* for a service class if a set of conditions satisfying $o$ implies that the other objectives of this service class are satisfied as well.

Note

- Dominance holds only for a specified range of control knob settings
- Dominance applies to objectives of a single service class only

Example

If average response time requirement is satisfied, throughput is also
Framework

Workload objectives

Policy control loop

Policy controller

Workload manager
- Admission controller
- Scheduler
- Execution controller

DBMS

Workload
- Queries
- Service level objectives
Framework

- Workload objectives
- Workload
  - Queries
  - Service level objectives
- Policy control loop
  - Policy controller
    - Workload manager
      - Admission controller
      - Scheduler
      - Execution controller
  - Simulated DBMS
Why a simulator?

- Deterministic
- Repeatable results
- Experiment with varying workloads with varying characteristics
- Easily change system configuration
- Speedup
Experiments

Purpose of the experiments

- Two service classes, each with throughput and average response time objectives
- Control knobs: vary MPL for each service class
- Goal: find MPL settings where each objective is met
Experiments

Experimental setup

- Database engine models a parallel, shared-nothing architecture; four nodes with eight disks
- Data is partitioned across the disks
- More details on simulated engine in the paper
- Multiple streams per service class, each stream sends queries one after the other with no wait time between two queries
- OLTP-style queries; a query accesses data on a single partition only
Experiment 1

<table>
<thead>
<tr>
<th></th>
<th>Service class 1</th>
<th>Service class 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average response time (sec)</td>
<td>0.25</td>
<td>1.0</td>
</tr>
<tr>
<td>Throughput (q/sec)</td>
<td>130</td>
<td>80</td>
</tr>
<tr>
<td>Dominant objective</td>
<td>throughput</td>
<td>throughput</td>
</tr>
<tr>
<td>Algorithm optimizes for</td>
<td>throughput</td>
<td>throughput</td>
</tr>
</tbody>
</table>
Results

overall search space
Results

overall search space

Objectives of class 2 met, objectives of class 1 violated

Objectives of class 1 met, objectives of class 2 violated

Objectives of classes 1 and 2 met
Results

search space considered by workload adaptation-MSO
Results
workload adaptation-MSO
Results

workload adaptation-MSO
# Experiment 2

<table>
<thead>
<tr>
<th></th>
<th>Service class 1</th>
<th>Service class 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average response time (sec)</td>
<td>0.25</td>
<td>0.6</td>
</tr>
<tr>
<td>Throughput ($q/sec$)</td>
<td>100</td>
<td>80</td>
</tr>
<tr>
<td>Dominant objective</td>
<td>average response time</td>
<td>throughput</td>
</tr>
<tr>
<td>Algorithm optimizes for</td>
<td>throughput</td>
<td>throughput</td>
</tr>
</tbody>
</table>
Results

naïve
Results

search space considered by workload adaptation-MSO
Results

search space considered by workload adaptation-MSO

Cannot find setting in operating envelope with MPL sum $= 32$

$\Rightarrow$ increase to 48
## Results

Search space considered by workload adaptation-MSO

<table>
<thead>
<tr>
<th>MPL service class 1</th>
<th>2</th>
<th>4</th>
<th>6</th>
<th>8</th>
<th>10</th>
<th>12</th>
<th>14</th>
<th>16</th>
<th>18</th>
<th>20</th>
<th>22</th>
<th>24</th>
<th>26</th>
<th>28</th>
<th>30</th>
<th>32</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPL service class 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>2</th>
<th>4</th>
<th>6</th>
<th>8</th>
<th>10</th>
<th>12</th>
<th>14</th>
<th>16</th>
<th>18</th>
<th>20</th>
<th>22</th>
<th>24</th>
<th>26</th>
<th>28</th>
<th>30</th>
<th>32</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Results

workload adaptation-MSO

![Graph showing MPL over time with two distinct lines for MPL1 and MPL2, indicating changes in throughput and average response time over a range of time intervals.]
Results

workload adaptation-MSO
Results

workload adaptation-MSO

Solution exists in the search space but algorithm does NOT find it
Results

workload adaptation-MSO

throughput okay
Results
workload adaptation-MSO

average response time of service class $s_2$ violated
Conclusion and ongoing work

- Presentation of test framework
- Comprehensive search solves the search problem, and gives additional information: Does a solution exist? How many settings satisfy the constraints? $\rightarrow$ prohibitively expensive
  $\Rightarrow$ Need heuristic approach
- Solutions found by *workload adaptation-MSO* are “fragile”
  $\Rightarrow$ Need different set of algorithms to solve the search problem