A Page Fault Equation for Dynamic Heap Sizing

Y.C. Tay
National University of Singapore

X.R. Zong
Duke University
Java, C#, Ruby ... require garbage collection (GC) for the heap

heap size $H$ determines \#GC

**issue:** how to tune $H$?
For a garbage-collected (GC) application, how does execution time vary with heap size \( H \) ?

RAM allocation \( M \) changes dynamically. How should \( H \) vary with \( M \)?
How should $H$ vary with $M$?

**Our answer:** Heap Sizing Rule

$H = \frac{M - b}{a}$

Where does $H = \frac{M - b}{a}$ come from?
Where does $H = \frac{M - b}{a}$ come from?

How does \#pagefaults $n$ vary with $M$?

Page Fault Equation [TZ]:

$$n = \frac{1}{2} (K + \sqrt{K^2 - 4})(n^* + n_0) - n_0$$

where $K = 1 + \frac{M^* - M_0}{M - M_0}$

universal: works for Linux, Windows, compute/IO/memory-intensive workloads, garbage-collected applications

different heap sizes:

Page Fault Equation [TZ]:

$$n = \frac{1}{2} (K + \sqrt{K^2 - 4})(n^* + n_0) - n_0$$

where $K = 1 + \frac{M^* - M_0}{M - M_0}$

universal: works for Linux, Windows, compute/IO/memory-intensive workloads, garbage-collected applications

different mutators:

different garbage collectors:
Interpretation for $n_0$?

$n_0$ measures memory taken off freelist during GC

How does $H$ affect $n_0$?

Page Fault Equation [TZ]:

$$n = \frac{1}{2} \left( K + \sqrt{K^2 - 4} \right) (n^* + n_0) - n_0$$

where $K = 1 + \frac{M^* - M_0}{M - M_0}$
How does $H$ affect $M^*$?

Page Fault Equation [TZ]:

$$n = \frac{1}{2} \left( K + \sqrt{K^2 - 4} \right) (n^* + n_0) - n_0$$

where $K = 1 + \frac{M^* - M_0}{M - M_0}$
How should $H$ vary with $M$?

Our answer: Heap Sizing Rule

$$H = \frac{M - b}{a}$$

Experiment: static $M$

JikesRVM dynamic heap sizing (varies $H$ during execution according to heap utilization)

Graph showing elapsed time vs. memory allocation for static $H$ according to Heap Sizing Rule.
How should $H$ vary with $M$?

Our answer: Heap Sizing Rule

$$H = \frac{M - b}{a}$$

<table>
<thead>
<tr>
<th>experiment: dynamic $M$</th>
</tr>
</thead>
<tbody>
<tr>
<td>dynamic $H$ according to Heap Sizing Rule ($H$ adjusted during GC only)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>page faults</th>
<th>MarkSweep pmd</th>
<th>SemiSpace pmd</th>
<th>MarkSweep xalan</th>
</tr>
</thead>
<tbody>
<tr>
<td>RVM</td>
<td>425828</td>
<td>680575</td>
<td>352338</td>
</tr>
<tr>
<td>Rule</td>
<td>36228</td>
<td>36470</td>
<td>64580</td>
</tr>
<tr>
<td>execution</td>
<td>RVM</td>
<td>4762</td>
<td>8362</td>
</tr>
<tr>
<td>time (sec)</td>
<td>Rule</td>
<td>419</td>
<td>404</td>
</tr>
</tbody>
</table>
summary

• pagefault modeling is difficult for GC applications  
  --- reference pattern changes with $H$

• our paper presents a heap-aware pagefault equation

• this equation can be used for dynamic heap sizing

future work

using the equation for heap partitioning