Sampling Methods

ESESC Tutorial

Speaker: Sina Hassani
You will learn:
- Why we need sampling in simulation
- How time-based sampling works
- Sampling configurations in ESESC
- How to use Gstats to add a new statistics counter
• Sampling Overview
• ESESC sampling parameters
• Demo single threaded sampling
• Demo multithreaded sampling
• Collecting statistics in ESESC
• Demo adding a new statistics counter
Sampling Overview
Outline

• Sampling overview
• ESESC sampling parameters
• Demo single threaded sampling
• Demo multithreaded sampling
• Collecting statistics in ESESC
• Demo adding a new statistics counter
ESESC Sampling Modes

- Rabbit
  - Emulation only

- Warmup
  - Update cache

- Detail
  - Full simulation but discard statistics

- Timing
  - Full simulation and keep statistics

Rabbit | Warmup | Detail | Timing
--- | --- | --- | ---
R | W | D | T
R | W | D | T
R | W | D | T
R | W | D | T

Sampling Methods
Sina Hassani
### ESESC Sampling Parameters

- **type**
  - inst, time

- **nInstSkip**
  - Number of instructions main thread skips

- **nInstSkipThreads**
  - Number of instructions spawned threads skip

- **maxnsTime**
  - Maximum simulation time in ns

- **nInstMax**
  - Maximum number of instructions to simulate

- **nInstRabbit**
  - Emulation only

- **nInstWarmup**
  - Emulation plus cache

- **nInstDetail**
  - Detailed warm-up of full pipeline

- **nInstTiming**
  - Collect statistics

---

```plaintext
# Sampling mode
samplerSel = "TASS"
samplerSel = "TBS"
samplerSel = "SkipAll"

[TBS]
type = "time"
nInstSkip = 1
nInstSkipThreads = 1
maxnsTime = 3e7
nInstMax = 1e7
nInstRabbit = 2097593
nInstWarmup = 3276509
nInstDetail = 37199
nInstTiming = 110503
PowPredictionHist = 5
doPowPrediction = 1
TempToPerfRatio = 1.0
```
Outline

• Sampling overview
• ESESC sampling parameters
• Demo single threaded sampling
• Demo multithreaded sampling
• Collecting statistics in ESESC
• Demo adding a new statistics counter
Demo 1: Single-threaded with TBS

• Run crafty with TBS
• Single core configured
### Sampler 0 (Procs 0)

<table>
<thead>
<tr>
<th></th>
<th>Rabbit</th>
<th>Warmup</th>
<th>Detail</th>
<th>Timing</th>
<th>Total</th>
<th>KIPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>KIPS</td>
<td>6417</td>
<td>18282</td>
<td>651</td>
<td>642</td>
<td>8330</td>
<td></td>
</tr>
<tr>
<td>Inst</td>
<td>0.0%</td>
<td>95.6%</td>
<td>1.1%</td>
<td>3.3%</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Instruction percentage and speed in each mode
Outline

• Sampling overview
• ESESC sampling parameters
• Demo single threaded sampling
• Demo multithreaded sampling
• Collecting statistics in ESESC
• Demo adding a new statistics counter
ESESC Multicore Configuration

- Homogenous and heterogeneous multicore configuration

  Homogenous
  - cpusimu[0:NUM_CORES-1]

  Heterogeneous
  - cpusimu[0] = ‘coreType1’
  - cpusimu[1] = ‘coreType2’

```
#Single Core
cpuemu[0] = 'QEMUSectionCPU'
cpusimu[0] = "$(coreType)"

#Homogenous Multicore
cpuemu[0:1] = 'QEMUSectionCPU'
cpusimu[0:1] = "$(coreType)"

#Heterogenous Multicore
cpuemu[0:1] = 'QEMUSectionCPU'
cpusimu[0] = "$(coreType1)"
cpusimu[1] = "$(coreType2)"
```
Demo 2: Multithreaded Sampling

• Run FFT with TBS
• Multicore configuration
Outline

- Sampling overview
- ESESC sampling parameters
- Demo single threaded sampling
- Demo multithreaded sampling
- Collecting statistics in ESESC
- Demo adding a new statistics counter
• Unified statistics management with GStats classes
• Raw GStats output is processed by report.pl script
• Only collect statistics during timing simulation
Statistics with Sampling

- Use GStats class for simulation statistics
- Each GStat must have a unique name
  - All stats stored in a hash map
  - Need to use state of instruction when updating counter
- GStatsCntr
  - Counter that supports
    - add – add specified amount to counter
    - inc – increment by 1
    - dec – decrement by 1
- GStatsAvg
  - Average value
- GStatsMax
  - Number of samples and max value
Example GStat: nCommitted

- GStatsCntr nCommitted counts number of committed instructions
  - Defined in GProcessor.h as part of GProcessor class
- Used in OoOProcessor which inherits from GProcessor
- Reads state of dinst->getStatsFlag()

```c
448 449 450 451 452
if (!flushing) {
    nCommitted.inc(dinst->getStatsFlag());
}
```
Outline

• Sampling overview
• ESESC sampling parameters
• Demo single threaded sampling
• Demo multithreaded sampling
• Collecting statistics in ESESC
• Demo adding a new statistics counter
Demo 3: Add a Counter to ESESC

- Add counter to ESESC
- Run and explain GStats output
Summary

• ESESC uses sampling for speed-up
• Sampling has minimum effect in accuracy
• Time-based sampling supports single-threaded and multi-threaded simulation
• Sampling mode and parameters can be easily changed
• ESESC uses Gstats for gathering statistics
• You can easily add other statistics counters
Sampling Methods

Sina Hassani

ESESC

Sampling References

• SMARTS

• Thermal

• Multithreaded (TBS)
  • ESESC: A Fast Multicore Simulator Using Time-Based Sampling, Ardestani, et al., HPCA 2013
Questions?