

Grid Simulator with Production Scheduling Algorithms

Miroslav Ruda

Institute of Computer Science, Masaryk University
Brno, Czech Republic



CoreGRID Institute on Resource Management and Scheduling



- 1 Motivation
- 2 Simulator with Virtual Machines
- 3 Preemption using virtual machines
- 4 Experimental results
- 5 Conclusion

Scheduling Algorithms

Algorithms in Grid simulators

- SimGrid, GridSim, GSSIM , Alea
- development and testing of new algorithms
- comparison of algorithms

Algorithms in production systems

- PBSPro, SGE, Maui, Moab
- in simulators: approximated with FIFO (with backfilling)
- hard to reimplement
 - many rules, features, bugs
 - closed source, algorithms not published

Example

www.excludus.com

- Information about Excludus "Grid Optimizer"
 - uses innovative real-time scheduling algorithm
 - dynamic adaptive scheduling beyond traditional workload managers



Grid Simulator
with
Production
Scheduling
Algorithms

Miroslav Ruda

Motivation

Simulator with
Virtual
Machines

Preemption
using virtual
machines

Experimental
results

Conclusion

Scheduling Algorithms

Algorithms in Grid simulators

- SimGrid, GridSim, GSSIM , Alea
- development and testing of new algorithms
- comparison of algorithms

Algorithms in production systems

- PBSPro, SGE, Maui, Moab
- in simulators: approximated with FIFO (with backfilling)
- hard to reimplement
 - many rules, features, bugs
 - closed source, algorithms not published

Example

www.excludus.com

- Information about Excludus "Grid Optimizer"
 - uses innovative real-time scheduling algorithm
 - dynamic adaptive scheduling beyond traditional workload managers

Scheduling Algorithms

Algorithms in Grid simulators

- SimGrid, GridSim, GSSIM , Alea
- development and testing of new algorithms
- comparison of algorithms

Algorithms in production systems

- PBSPro, SGE, Maui, Moab
- in simulators: approximated with FIFO (with backfilling)
- hard to reimplement
 - many rules, features, bugs
 - closed source, algorithms not published

Example

www.excludus.com

- **Information about Excludus "Grid Optimizer"**
 - uses innovative real-time scheduling algorithm
 - dynamic adaptive scheduling beyond traditional workload managers

Simulator with production resource management system



Grid Simulator
with
Production
Scheduling
Algorithms

Miroslav Ruda

Motivation

Simulator with
Virtual
Machines

Preemption
using virtual
machines

Experimental
results

Conclusion

Experiments with PBSPro

- different setup of PBS scheduler
 - queues, priorities, backfilling, . . .
- different setup of worker nodes
 - number of nodes per queue, . . .
- modifications of PBS scheduler
- inclusion of virtual machines into PBS

Future: new scheduler

Simulator with Virtual Machines



Grid Simulator
with
Production
Scheduling
Algorithms

Miroslav Ruda

Motivation

Simulator with
Virtual
Machines

Preemption
using virtual
machines

Experimental
results

Conclusion

- Worker nodes represented by virtual machines
- Standard PBS Server and Scheduler
 - running on dedicated server
- Standard PBS Mom
 - running within each virtual machine
- Sleep jobs
 - no CPU/memory consumption

Workloads

- Real workloads
- Czech Grid *META Centrum*
- Extracted from PBS accounting
- 2005-2008

- Jobs submitted with the same requirements
 - on worker nodes
 - to the same queues
 - with original owners, ...
- Time reduction
 - configurable reduce factor (600)
 - expected and real wall-clock time
 - job arrival time



Vserver based virtual machines

- One kernel space - **very lightweight**
- Similar to
 - Linux chroot or Solaris containers, with better protection
- Access limits
 - standard: filesystem
 - added: processes, network devices ...
- No hardware emulation, no paravirtualisation
 - no performance penalty
- **Copy On Write filesystem**
 - one RO root filesystem, with RW **overlay** filesystem
- System daemons
 - running only once in hosting environment



Experimental Testbed

- **Current workloads (year 2007)**

- January 4.700 jobs, March 14.000 jobs, Jan-August 70.000 jobs

- **150 Vserver domains**

- 16 core AMD machine can use more physical machines
- represents 300 nodes can be extended

- **COW filesystem**

- 300 MB one system instalation
- 12 GB used to represent 150 virtual machines

- **Virtual machine: only PBS Mom + sshd**

- **Submission of all jobs**

- without any sleep takes less then 10 minutes

- **Reduce factor 600**

- 1 month -> 1.5 hours, 1 year -> \leq 1 day
- reasonably small simulation overhead



Evaluation Criteria

Grid Simulator
with
Production
Scheduling
Algorithms

Miroslav Ruda

Motivation

Simulator with
Virtual
Machines

Preemption
using virtual
machines

Experimental
results

Conclusion

Standard monitoring during simulation run

- number of running/waiting/done jobs
- number of used nodes

Analysis of accounting data

- Weighted Response Time (WRT)
- Weighted SlowDown Time (WSD)
- Weighted Wait Time (WWT)
- metrics per user, queue
- also structured by number of nodes used by job

Standard submit script in simulator

```
#!/bin/bash
reduce=600 #reduce factor

sleep $(( $SIMSLEEP / $reduce )) #gap in workload

sudo $SIMUSER qsub -q $SIMQUEUE
    #the same node requirements
    -l nodes=$SIMNODESL
    -l walltime=$(( $SIMREQ / $reduce )) <<EOF

#sleep instead of real job
sleep $(( $SIMWALL / $reduce ))
EOF
```



Jobs with Preemption

- Motivation: better support of parallel jobs or priorities
- Two virtual machines running on physical machine
 - first machine: standard jobs
 - second machine: privileged/parallel jobs
- Magrathea allows
 - several VMs running on a single computer
 - jobs submitted directly to VMs
- When job is started in privileged domain, Magrathea
 - suspends job in standard domain (if needed)
 - almost all CPU/memory resources are given to privileged domain (but standard is still running)
- Support of simulator
 - Magrathea installed on simulated machines too
 - sleep jobs must respect preemption



Preemption in simulator

```
reduce=600
sleep $((($SIMSLEEP/$reduce))
sudo $SIMUSER qsub -q sim$SIMQUEUE
    -l nodes=$SIMNODESL
    -l walltime=$((($SIMREQ/$reduce))<<EOF

sleeptime = $((($SIMWALL/$reduce))
while ($sleeptime >0) do
    sleep $sleeptime
    #check long how job has been preempted
    sleeptime='magrathea-preempted-time';
done
EOF
```

Experiment motivation – ITI cluster

- cluster owned by Institute of Theoretical Informatics
- all grid users may submit short jobs on this cluster
- ITI users must have higher priority
 - PBSPPro prioritization working for sequential jobs only
 - neither PBSPPro starvation support is useful
 - not possible to define that only ITI jobs are starving
 - starving jobs blocking the whole *META Centre*
- experiment with 500 jobs submitted to ITI cluster (extracted from *META Centre* workload)
- 82 parallel jobs owned submitted by ITI



Experiment description

- all jobs to one queue
 - starvation
 - strict fifo
- parallel jobs to high priority queue
 - with original starvation
 - with starvation used only for parallel jobs
 - backfilling
- modified starvation support
 - reservation for starving jobs, not blocking jobs submitted to non-reserved nodes
 - experiments with number of running or starving jobs
- preemption using virtual machines/Magrathea
 - experiments with number of running parallel jobs

Grid Simulator
with
Production
Scheduling
Algorithms

Miroslav Ruda

Motivation

Simulator with
Virtual
Machines

Preemption
using virtual
machines

Experimental
results

Conclusion

Experimental results

experiment	time	average waiting time						
		1	4	8	12	16	28	40
one queue	26:11	215	226	354	581	661	856	1069
+starving	36:48	562	446	552	612	1202	1536	920
parallel queue	26:08	218	258	290	668	659	844	1080
+limit 10 jobs	26:42	262	178	241	641	657	874	1092
+starvation	29:12	954	235	196	234	372	599	453
+backfilling	28:41	970	196	155	200	353	599	414
+parallel only	28:25	1018	125	81	89	266	470	342
preemption	23:47	620	61	74	59	185	465	678
+ only 5 jobs	23:28	307	179	178	192	358	655	876
reservations	30:30	1177	141	119	123	362	585	404
+only 5 jobs	27:00	638	265	243	304	481	684	489
+3 reserv.	28:03	466	235	288	350	692	947	637



Experimental results

experiment	time	average waiting time						
		1	4	8	12	16	28	40
one queue	26:11	215	226	354	581	661	856	1069
+starving	36:48	562	446	552	612	1202	1536	920
parallel queue	26:08	218	258	290	668	659	844	1080
+limit 10 jobs	26:42	262	178	241	641	657	874	1092
+starvation	29:12	954	235	196	234	372	599	453
+backfilling	28:41	970	196	155	200	353	599	414
+parallel only	28:25	1018	125	81	89	266	470	342
preemption	23:47	620	61	74	59	185	465	678
+ only 5 jobs	23:28	307	179	178	192	358	655	876
reservations	30:30	1177	141	119	123	362	585	404
+only 5 jobs	27:00	638	265	243	304	481	684	489
+3 reserv.	28:03	466	235	288	350	692	947	637

Experimental results

experiment	time	average waiting time						
		1	4	8	12	16	28	40
one queue	26:11	215	226	354	581	661	856	1069
+starving	36:48	562	446	552	612	1202	1536	920
parallel queue	26:08	218	258	290	668	659	844	1080
+limit 10 jobs	26:42	262	178	241	641	657	874	1092
+starvation	29:12	954	235	196	234	372	599	453
+backfilling	28:41	970	196	155	200	353	599	414
+parallel only	28:25	1018	125	81	89	266	470	342
preemption	23:47	620	61	74	59	185	465	678
+ only 5 jobs	23:28	307	179	178	192	358	655	876
reservations	30:30	1177	141	119	123	362	585	404
+only 5 jobs	27:00	638	265	243	304	481	684	489
+3 reserv.	28:03	466	235	288	350	692	947	637

Experimental results

experiment	time	average waiting time						
		1	4	8	12	16	28	40
one queue	26:11	215	226	354	581	661	856	1069
+starving	36:48	562	446	552	612	1202	1536	920
parallel queue	26:08	218	258	290	668	659	844	1080
+limit 10 jobs	26:42	262	178	241	641	657	874	1092
+starvation	29:12	954	235	196	234	372	599	453
+backfilling	28:41	970	196	155	200	353	599	414
+parallel only	28:25	1018	125	81	89	266	470	342
preemption	23:47	620	61	74	59	185	465	678
+ only 5 jobs	23:28	307	179	178	192	358	655	876
reservations	30:30	1177	141	119	123	362	585	404
+only 5 jobs	27:00	638	265	243	304	481	684	489
+3 reserv.	28:03	466	235	288	350	692	947	637



Experimental results

experiment	time	average waiting time						
		1	4	8	12	16	28	40
one queue	26:11	215	226	354	581	661	856	1069
+starving	36:48	562	446	552	612	1202	1536	920
parallel queue	26:08	218	258	290	668	659	844	1080
+limit 10 jobs	26:42	262	178	241	641	657	874	1092
+starvation	29:12	954	235	196	234	372	599	453
+backfilling	28:41	970	196	155	200	353	599	414
+parallel only	28:25	1018	125	81	89	266	470	342
preemption	23:47	620	61	74	59	185	465	678
+ only 5 jobs	23:28	307	179	178	192	358	655	876
reservations	30:30	1177	141	119	123	362	585	404
+only 5 jobs	27:00	638	265	243	304	481	684	489
+3 reserv.	28:03	466	235	288	350	692	947	637

Experimental results

experiment	time	average waiting time						
		1	4	8	12	16	28	40
one queue	26:11	215	226	354	581	661	856	1069
+starving	36:48	562	446	552	612	1202	1536	920
parallel queue	26:08	218	258	290	668	659	844	1080
+limit 10 jobs	26:42	262	178	241	641	657	874	1092
+starvation	29:12	954	235	196	234	372	599	453
+backfilling	28:41	970	196	155	200	353	599	414
+parallel only	28:25	1018	125	81	89	266	470	342
preemption	23:47	620	61	74	59	185	465	678
+ only 5 jobs	23:28	307	179	178	192	358	655	876
reservations	30:30	1177	141	119	123	362	585	404
+only 5 jobs	27:00	638	265	243	304	481	684	489
+3 reserv.	28:03	466	235	288	350	692	947	637

Experimental results

experiment	time	average waiting time						
		1	4	8	12	16	28	40
one queue	26:11	215	226	354	581	661	856	1069
+starving	36:48	562	446	552	612	1202	1536	920
parallel queue	26:08	218	258	290	668	659	844	1080
+limit 10 jobs	26:42	262	178	241	641	657	874	1092
+starvation	29:12	954	235	196	234	372	599	453
+backfilling	28:41	970	196	155	200	353	599	414
+parallel only	28:25	1018	125	81	89	266	470	342
preemption	23:47	620	61	74	59	185	465	678
+ only 5 jobs	23:28	307	179	178	192	358	655	876
reservations	30:30	1177	141	119	123	362	585	404
+only 5 jobs	27:00	638	265	243	304	481	684	489
+3 reserv.	28:03	466	235	288	350	692	947	637

Experimental results

experiment	time	average waiting time						
		1	4	8	12	16	28	40
one queue	26:11	215	226	354	581	661	856	1069
+starving	36:48	562	446	552	612	1202	1536	920
parallel queue	26:08	218	258	290	668	659	844	1080
+limit 10 jobs	26:42	262	178	241	641	657	874	1092
+starvation	29:12	954	235	196	234	372	599	453
+backfilling	28:41	970	196	155	200	353	599	414
+parallel only	28:25	1018	125	81	89	266	470	342
preemption	23:47	620	61	74	59	185	465	678
+ only 5 jobs	23:28	307	179	178	192	358	655	876
reservations	30:30	1177	141	119	123	362	585	404
+only 5 jobs	27:00	638	265	243	304	481	684	489
+3 reserv.	28:03	466	235	288	350	692	947	637



Experimental results

experiment	time	average waiting time						
		1	4	8	12	16	28	40
one queue	26:11	215	226	354	581	661	856	1069
+starving	36:48	562	446	552	612	1202	1536	920
parallel queue	26:08	218	258	290	668	659	844	1080
+limit 10 jobs	26:42	262	178	241	641	657	874	1092
+starvation	29:12	954	235	196	234	372	599	453
+backfilling	28:41	970	196	155	200	353	599	414
+parallel only	28:25	1018	125	81	89	266	470	342
preemption	23:47	620	61	74	59	185	465	678
+ only 5 jobs	23:28	307	179	178	192	358	655	876
reservations	30:30	1177	141	119	123	362	585	404
+only 5 jobs	27:00	638	265	243	304	481	684	489
+3 reserv.	28:03	466	235	288	350	692	947	637

Experimental results

experiment	time	average waiting time						
		1	4	8	12	16	28	40
one queue	26:11	215	226	354	581	661	856	1069
+starving	36:48	562	446	552	612	1202	1536	920
parallel queue	26:08	218	258	290	668	659	844	1080
+limit 10 jobs	26:42	262	178	241	641	657	874	1092
+starvation	29:12	954	235	196	234	372	599	453
+backfilling	28:41	970	196	155	200	353	599	414
+parallel only	28:25	1018	125	81	89	266	470	342
preemption	23:47	620	61	74	59	185	465	678
+ only 5 jobs	23:28	307	179	178	192	358	655	876
reservations	30:30	1177	141	119	123	362	585	404
+only 5 jobs	27:00	638	265	243	304	481	684	489
+3 reserv.	28:03	466	235	288	350	692	947	637

Experimental results

experiment	time	average waiting time						
		1	4	8	12	16	28	40
one queue	26:11	215	226	354	581	661	856	1069
+starving	36:48	562	446	552	612	1202	1536	920
parallel queue	26:08	218	258	290	668	659	844	1080
+limit 10 jobs	26:42	262	178	241	641	657	874	1092
+starvation	29:12	954	235	196	234	372	599	453
+backfilling	28:41	970	196	155	200	353	599	414
+parallel only	28:25	1018	125	81	89	266	470	342
preemption	23:47	620	61	74	59	185	465	678
+ only 5 jobs	23:28	307	179	178	192	358	655	876
reservations	30:30	1177	141	119	123	362	585	404
+only 5 jobs	27:00	638	265	243	304	481	684	489
+3 reserv.	28:03	466	235	288	350	692	947	637



Experimental results

experiment	time	average waiting time						
		1	4	8	12	16	28	40
one queue	26:11	215	226	354	581	661	856	1069
+starving	36:48	562	446	552	612	1202	1536	920
parallel queue	26:08	218	258	290	668	659	844	1080
+limit 10 jobs	26:42	262	178	241	641	657	874	1092
+starvation	29:12	954	235	196	234	372	599	453
+backfilling	28:41	970	196	155	200	353	599	414
+parallel only	28:25	1018	125	81	89	266	470	342
preemption	23:47	620	61	74	59	185	465	678
+ only 5 jobs	23:28	307	179	178	192	358	655	876
reservations	30:30	1177	141	119	123	362	585	404
+only 5 jobs	27:00	638	265	243	304	481	684	489
+3 reserv.	28:03	466	235	288	350	692	947	637

Conclusion & Future Work

New Grid simulator

- Inclusion of production resource management system
- New experiments: PBSPro (and other algorithms)
- Novel proposal with virtual machines
- New experiments: scheduling with Magrathea
- usable also for
 - 1 million of jobs test for EGEE L&B and Provenance
 - stability/race conditions test of PBSPro and Magrathea

Future work

- Study: limits of the simulator
 - efficient scheduling algorithms needed
 - cannot use actual load on machines
 - network usage/overloading not supported
 - missing host availability data
 - monitoring issues
 - workload tail
- New scheduler

Weighted Response Time, SlowDown, and Wait Time

Grid Simulator
with
Production
Scheduling
Algorithms

Miroslav Ruda

Motivation

Simulator with
Virtual
Machines

Preemption
using virtual
machines

Experimental
results

Conclusion

$$SA_j = reqResources_j \times (endTime_j - startTime_j)$$

$$TotalSA = \sum_{j \in Jobs} SA_j$$

$$SD = \frac{(endTime_j - submitTime_j)}{runtime_j}$$

$$WRT = \frac{\sum_{j \in Jobs} (SA_j (endTime_j - submitTime_j))}{TotalSA}$$

$$WSD = \frac{\sum_{j \in Jobs} SA_j \times SD_j}{TotalSA}$$

$$WWT = \frac{\sum_{j \in Jobs} SA_j \times (startTime_j - submitTime_j)}{TotalSA}$$