

ARIA: Automated Resource Inference and Allocation for MapReduce Environments*

Motivation

- MapReduce applications used for processing petabytes of data across the enterprise
- Controlling the allocation of resources in shared MapReduce environments is a key challenge
- Many users require job completion time guarantees
- Existing schedulers do not support SLOs
- In order to achieve SLOs, we need to answer:
 - How long will the job take given x% resources?
 - How much resources are required to complete the job within a given deadline?

Job Profiles

- Job Profiles compactly summarize performance metrics of different stages collected from logs that are
 - independent of job execution style
 - independent of application's input dataset size



ARIA Design





executions with varied inputs and number of allocated slots. Workload: WikiTrends, WordCount, Sort, Bayes, TF-IDF, Twitter



Abhishek Verma University of Illinois verma7@illinois.edu

Ludmila Cherkasova HP Labs, Palo Alto

Problem Definition and Approach

- For a given MapReduce application and number of map/reduce slots, what is the completion time?
- Inverse question: Given a deadline, how many map/reduce slots need to be allocated to finish within the deadline?
- Hadoop uses greedy task assignment for scheduling Our approach: Bounds-based Makespan model for greedy task assignment
- Let *n* tasks be processed by *k* slots. Let *avg* and *max* be the average and maximum duration of the *n* tasks resp.
- The job makespan is at least $T_{low} = n.avg/k$
- The job makespan at most $T_{un} = (n-1).avg/k + max$

Solving the Inverse Problem

v 192 160 ¥ 128

Meeting SLOs

	SL	0	∎ Lo	ower b	oound
	1600				
sec	1400	-	_		
(in	1200	+			
ne	1000	+		•	
tir	800	-			
ion	600	_			
olet	400	-			
<u> </u>	200	_			
00	0	+			
Job		0		2	
-					

Load threshold for arrival (%)	SLO exceeded utility (%)	# of jobs with missed SLOs	Average Load (%)
105	7.62	1	29.27
100	4.58	1	27.34
95	0	0	26.46
90	0	0	25.63







Job completion times are within 8 of their deadlines

Deadlines are missed only under high load