Mining for Statistical Models of Availability in Large-Scale Distributed Systems: An Empirical Study of SETI@home

Bahman Javadi[1], Derrick Kondo[1], Jean-Marc Vincent[1,2], David P. Anderson[3]

[1]Laboratoire d'Informatique de Grenoble, MESCAL team, INRIA, France
[2]University of Joseph Fourier, France
[3]UC Berkeley, USA

IEEE/ACM International Symposium on Modelling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS 2009)

- P2P, Grid, Cloud, and Volunteer computing systems

- P2P, Grid, Cloud, and Volunteer computing systems
- Main Features:
  - Tens or hundreds of thousands of unreliable and heterogeneous hosts

- P2P, Grid, Cloud, and Volunteer computing systems
- Main Features:
  - Tens or hundreds of thousands of unreliable and heterogeneous hosts
  - Uncertainty of host availability

- P2P, Grid, Cloud, and Volunteer computing systems
- Main Features:
  - Tens or hundreds of thousands of unreliable and heterogeneous hosts
  - Uncertainty of host availability

- P2P, Grid, Cloud, and Volunteer computing systems
- Main Features:
  - Tens or hundreds of thousands of unreliable and heterogeneous hosts
  - Uncertainty of host availability

Main Motivation

Effective Resource Selection for Stochastic Scheduling Algorithms

- P2P, Grid, Cloud, and Volunteer computing systems
- Main Features:
  - Tens or hundreds of thousands of unreliable and heterogeneous hosts
  - Uncertainty of host availability

Main Motivation

Effective Resource Selection for Stochastic Scheduling Algorithms

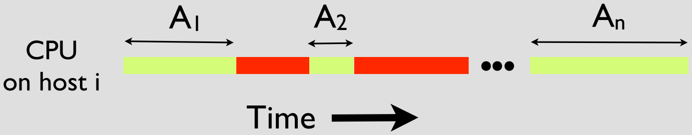
Goal

Model of host availability
(i.e., subset of hosts with the same availability distribution)

Outline

Define Availability

CPU availability on each host



Length of Availability Intervals: A1, A2, ..., An

Measurement Method

BOINC

- Middleware for volunteer computing systems
- Underlying software infrastructure for projects such as SETI@home

Measurement Method

BOINC

- Middleware for volunteer computing systems
- Underlying software infrastructure for projects such as SETI@home

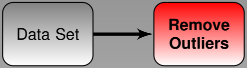
We instrumented the BOINC client to collect CPU availability traces:

- Total number of host traces: 226,208
- Collection period: April 1, 2007 - Jan 1, 2009
- Total CPU time: 57,800 years
- Number of intervals: 102,416,434
- Assume 100% or 0% availability

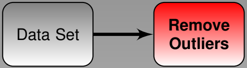
Outline

1. Introduction and Motivation
2. Measurement
   - Remove outliers
3. Modelling Process
   - Randomness Tests
   - Clustering
   - Model fitting
4. Discussions
   - Significance of Clustering Criteria
   - Scheduling Implications
5. Related Work
6. Conclusion and Future Work

Outliers

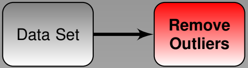


Outliers

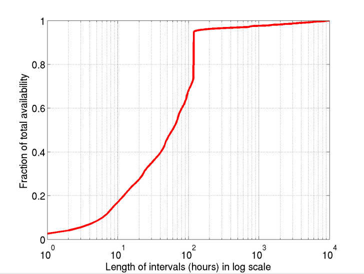


Check for outliers: Artifacts resulted from a benchmark run periodically every five days

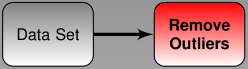
Outliers



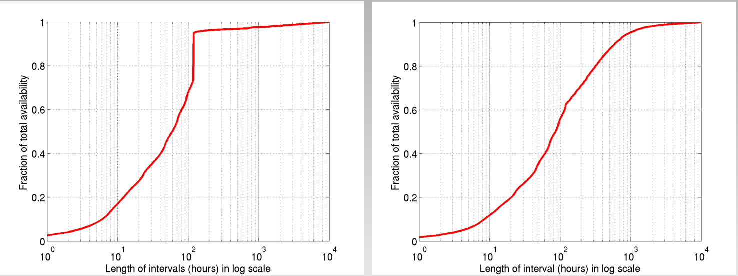
Check for outliers: Artifacts resulted from a benchmark run periodically every five days



Outliers

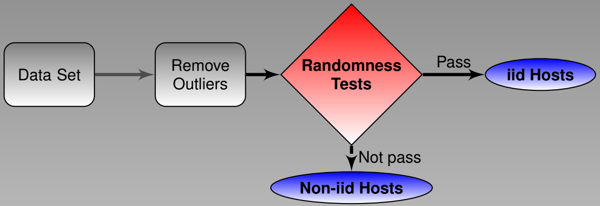


Check for outliers: Artifacts resulted from a benchmark run periodically every five days



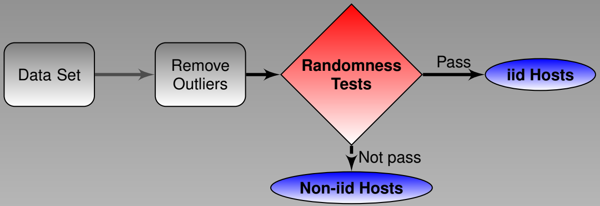
Outline

Randomness Tests



To determine which hosts have truly random availability intervals

Randomness Tests

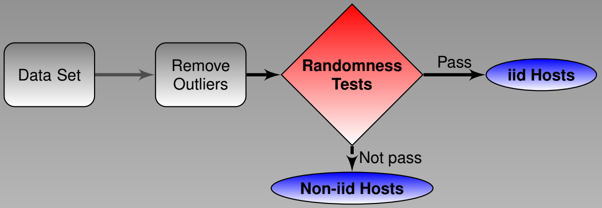


To determine which hosts have truly random availability intervals

Four well-known non-parametric tests:

- Runs test
- Runs up/down test
- Mann-Kendall test
- Autocorrelation function test (ACF)

Randomness Tests



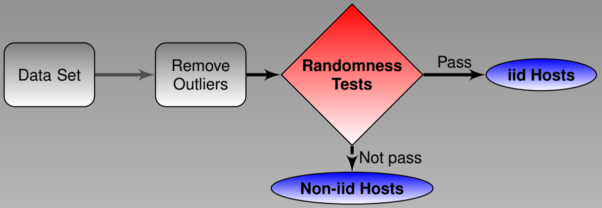
To determine which hosts have truly random availability intervals

Four well-known non-parametric tests:

- Runs test
- Runs up/down test
- Mann-Kendall test
- Autocorrelation function test (ACF)

Test	Runs std	Runs up/down	ACF	Kendall	All
# of hosts	101649	144656	109138	101462	**57757**
Fraction	0.602	0.857	0.647	0.601	**0.342**

Randomness Tests



To determine which hosts have truly random availability intervals

Four well-known non-parametric tests:

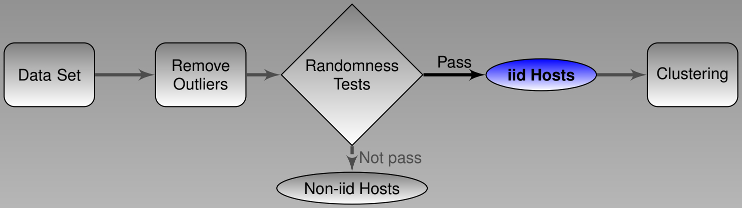
- Runs test
- Runs up/down test
- Mann-Kendall test
- Autocorrelation function test (ACF)

Test	Runs std	Runs up/down	ACF	Kendall	All
# of hosts	101649	144656	109138	101462	**57757**
Fraction	0.602	0.857	0.647	0.601	**0.342**

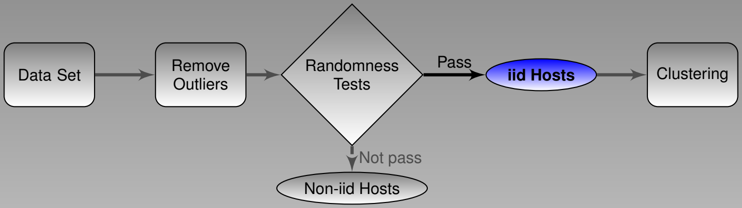
Result: 34% are i.i.d. hosts (2.2 PetaFLOPS)

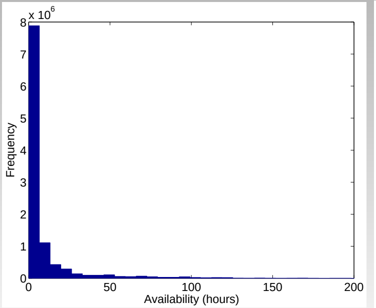
Outline

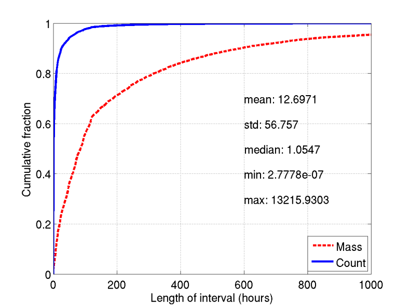
Distribution of Availability Intervals



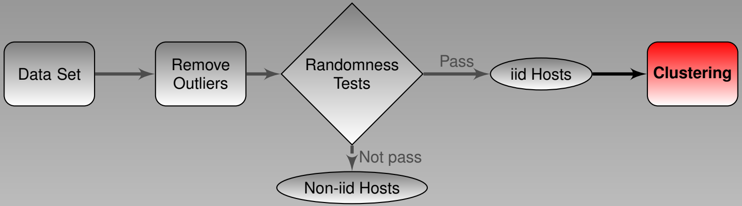
Distribution of Availability Intervals





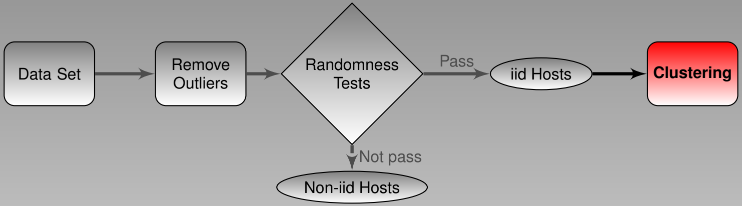


Clustering Method



Generate a few clusters based on availability distribution function

Clustering Method

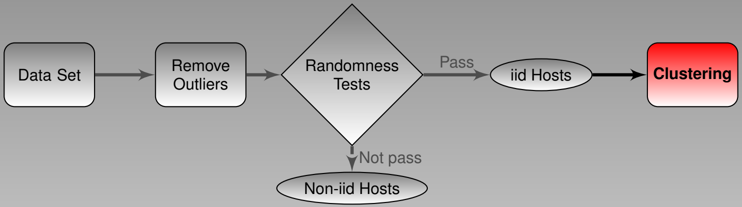


Generate a few clusters based on availability distribution function

Method:

- Hierarchical

Clustering Method

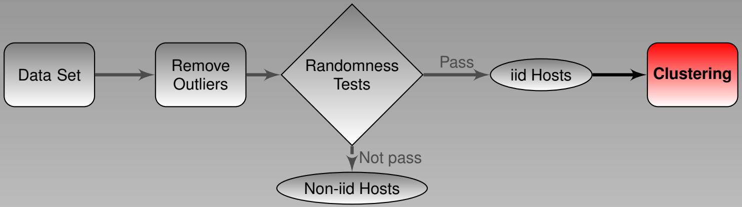


Generate a few clusters based on availability distribution function

Method:

- Hierarchical
  - Compute all permutations

Clustering Method

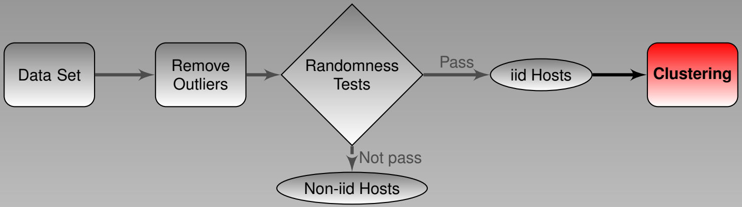


Generate a few clusters based on availability distribution function

Method:

- Hierarchical
  - Compute all permutations
  - Memory intensive

Clustering Method

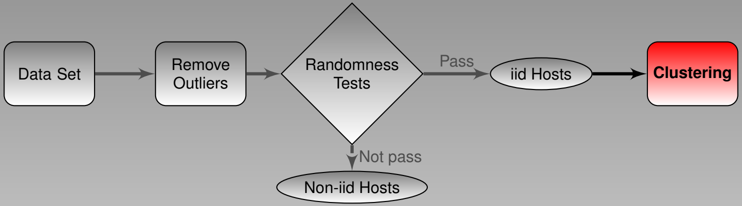


Generate a few clusters based on availability distribution function

Method:

- Hierarchical
  - Compute all permutations
  - Memory intensive
- K-means (fast K-means)

Clustering Method

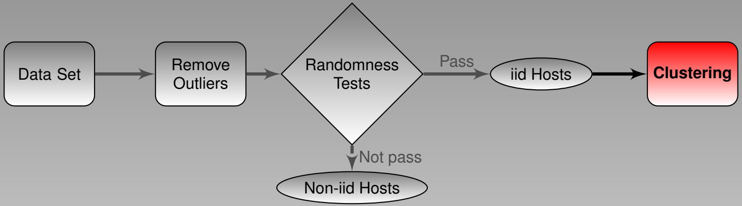


Generate a few clusters based on availability distribution function

Method:

- Hierarchical
  - Compute all permutations
  - Memory intensive
- K-means (fast K-means)
  - Fast convergence

Clustering Method



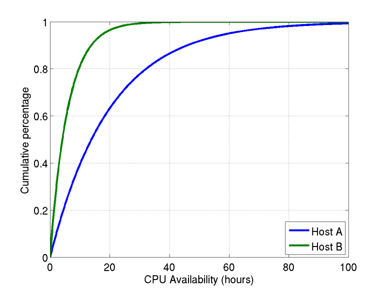
Generate a few clusters based on availability distribution function

Method:

- Hierarchical
  - Compute all permutations
  - Memory intensive
- K-means (fast K-means)
  - Fast convergence
  - Dependent on initial centroids

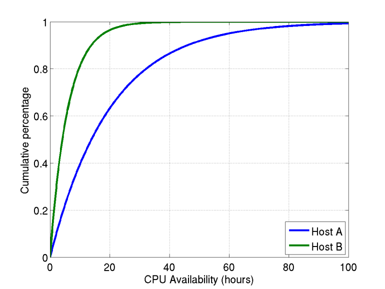
Distance Metrics

Distance between CDF of two hosts



Distance Metrics

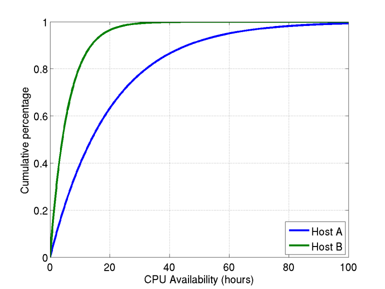
Distance between CDF of two hosts



- Kolmogorov-Smirnov: Maximum difference between two CDFs

Distance Metrics

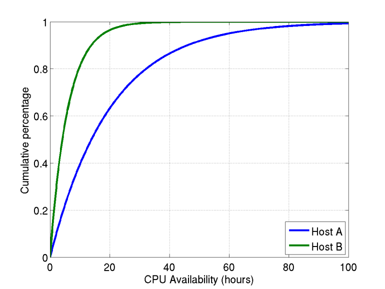
Distance between CDF of two hosts



- Kolmogorov-Smirnov: Maximum difference between two CDFs
- Kuiper: Maximum deviation above and below of two CDFs

Distance Metrics

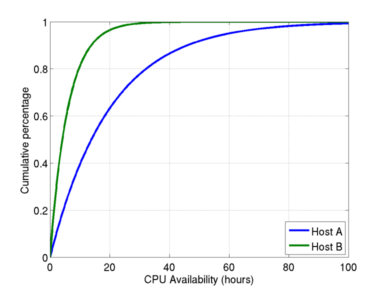
Distance between CDF of two hosts



- Kolmogorov-Smirnov: Maximum difference between two CDFs
- Kuiper: Maximum deviation above and below of two CDFs
- Cramer-von Mises: Area between two CDFs

Distance Metrics

Distance between CDF of two hosts



- Kolmogorov-Smirnov: Maximum difference between two CDFs
- Kuiper: Maximum deviation above and below of two CDFs
- Cramer-von Mises: Area between two CDFs
- Anderson-Darling: Area between two CDFs, more weight on the tail

Distance Metrics

Important Challenge:

Number of samples in each CDF

- Few samples –> not enough confidence on the result

Distance Metrics

Important Challenge:

Number of samples in each CDF

- Few samples –> not enough confidence on the result
- Too much samples –> the metric will be too sensitive

Distance Metrics

Important Challenge:

Number of samples in each CDF

- Few samples –> not enough confidence on the result
- Too much samples –> the metric will be too sensitive

- Data Set: different hosts have different number of samples

Distance Metrics

Important Challenge:

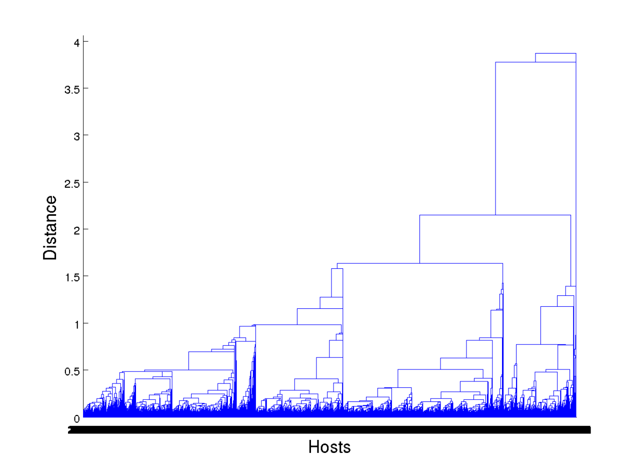
Number of samples in each CDF

- Few samples –> not enough confidence on the result
- Too much samples –> the metric will be too sensitive

- Data Set: different hosts have different number of samples
- Our solution: randomly select a fixed number of intervals from each host (i.e., 30 samples)

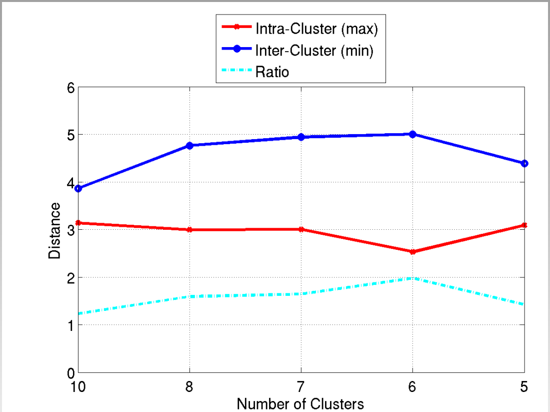
Clustering Results

Dendrogram of hierarchical clustering: 5-10 distinct groups (bootstrap)

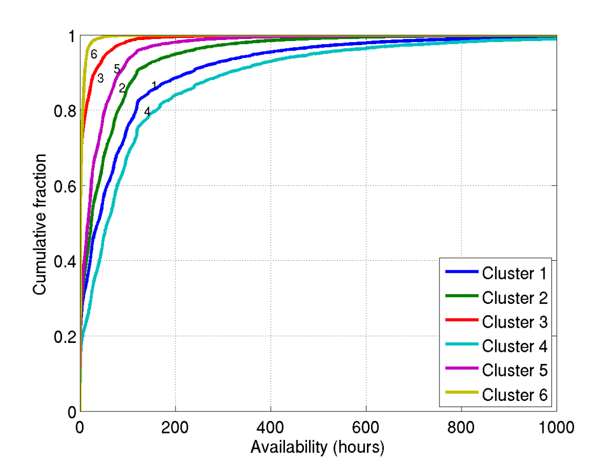


Clustering Results

Comparison of distances in clusters (k-means for all iid hosts):

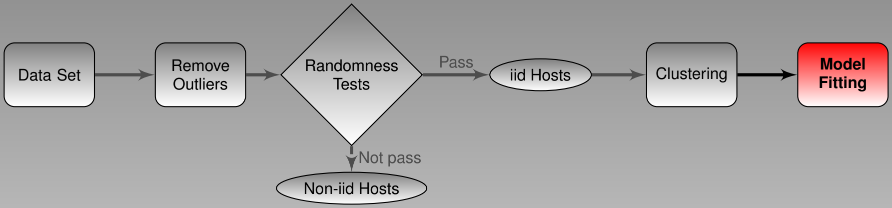


EDF of clusters

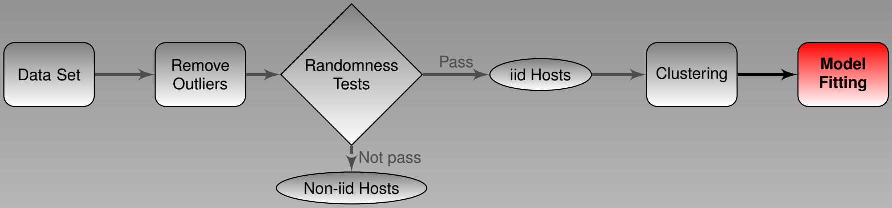


Outline

Methods


Data Set
Remove Outliers
Randomness Tests
Pass
iid Hosts
Clustering
Model Fitting
Not pass
Non-iid Hosts

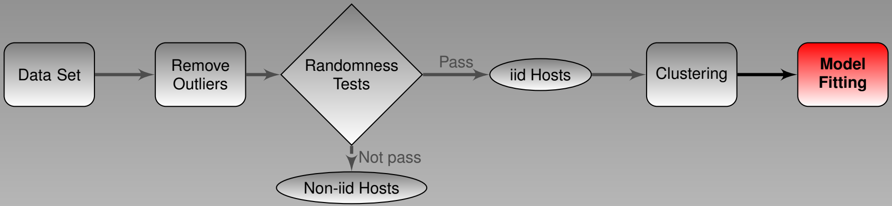
Methods



Method:

- Maximum Likelihood Estimation (MLE)
- Moment Matching (MM)

Methods



Method:

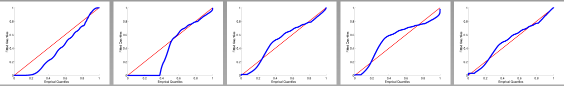
- Maximum Likelihood Estimation (MLE)
- Moment Matching (MM)

Target Distributions:

- Exponential
- Pareto
- Weibull
- Log-normal
- Gamma

Graphical Test

PP-plots: Exponential, Pareto, Weibull, Log-normal, Gamma

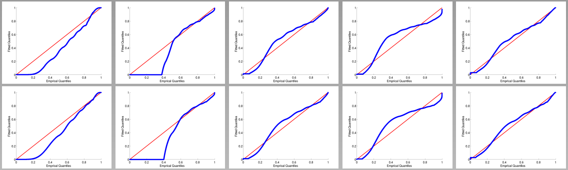
Cluster 1: 

Graphical Test

PP-plots: Exponential, Pareto, Weibull, Log-normal, Gamma

Cluster 1:

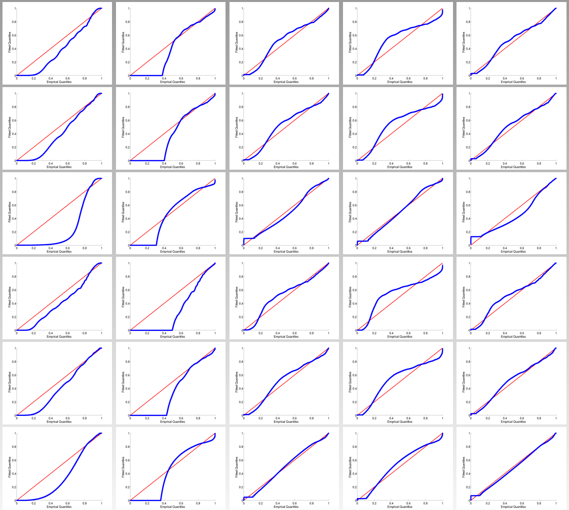
Cluster 2:



Graphical Test

PP-plots: Exponential, Pareto, Weibull, Log-normal, Gamma

Cluster 1:
Cluster 2:
Cluster 3:
Cluster 4:
Cluster 5:
Cluster 6:



Goodness Of Fit Tests

Generate p-values by two GOF tests (average over 1000 runs):

- Kolmogorov-Smirnov (KS) test
- Anderson-Darling (AD) test

Goodness Of Fit Tests

Generate p-values by two GOF tests (average over 1000 runs):

- Kolmogorov-Smirnov (KS) test
- Anderson-Darling (AD) test

Data sets	Exponential		Pareto		Weibull		Log-Normal		Gamma	
	AD	*KS*	*AD*	*KS*	*AD*	*KS*	*AD*	*KS*	*AD*	*KS*
All iid hosts	0.004	0.000	0.061	0.013	**0.581**	**0.494**	0.568	0.397	0.431	0.359
Cluster 1	0.155	0.071	0.029	0.008	0.466	0.243	0.275	0.116	**0.548**	**0.336**
Cluster 2	0.188	0.091	0.020	0.004	0.471	0.259	0.299	0.128	**0.565**	**0.384**
Cluster 3	0.002	0.000	0.068	0.023	0.485	0.380	**0.556**	**0.409**	0.372	0.241
Cluster 4	0.264	0.163	0.002	0.000	0.484	0.242	0.224	0.075	**0.514**	**0.276**
Cluster 5	0.204	0.098	0.013	0.002	0.498	0.296	0.314	0.153	**0.563**	**0.389**
Cluster 6	0.059	0.016	0.033	0.009	**0.570**	**0.439**	0.485	0.328	0.538	0.467

Some properties of clusters

Clusters	# of hosts	% of total avail.	mean (hrs)	Best fit	Parameters	
					shape	*scale*
All iid hosts	57757	1.0	12.697	Weibull	0.3787	3.0932
Cluster 1	3606	0.16	90.780	Gamma	0.3131	289.9017
Cluster 2	9321	0.35	54.563	Gamma	0.3372	161.8350
Cluster 3	13256	0.22	11.168	Log-Normal	-0.8937	3.2098
Cluster 4	275	0.01	123.263	Gamma	0.3739	329.6922
Cluster 5	1753	0.05	34.676	Gamma	0.3624	95.6827
Cluster 6	29546	0.20	4.138	Weibull	0.4651	1.8461

- Cluster sizes are different and often significant

Some properties of clusters

Clusters	# of hosts	% of total avail.	mean (hrs)	Best fit	Parameters	
					shape	*scale*
All iid hosts	57757	1.0	12.697	Weibull	0.3787	3.0932
Cluster 1	3606	0.16	90.780	Gamma	0.3131	289.9017
Cluster 2	9321	0.35	54.563	Gamma	0.3372	161.8350
Cluster 3	13256	0.22	11.168	Log-Normal	-0.8937	3.2098
Cluster 4	275	0.01	123.263	Gamma	0.3739	329.6922
Cluster 5	1753	0.05	34.676	Gamma	0.3624	95.6827
Cluster 6	29546	0.20	4.138	Weibull	0.4651	1.8461

- Cluster sizes are different and often significant
- Heterogeneity in distribution parameters (different *scale* parameters)

Some properties of clusters

Clusters	# of hosts	% of total avail.	mean (hrs)	Best fit	Parameters	
					shape	*scale*
All iid hosts	57757	1.0	12.697	Weibull	0.3787	3.0932
Cluster 1	3606	0.16	90.780	Gamma	0.3131	289.9017
Cluster 2	9321	0.35	54.563	Gamma	0.3372	161.8350
Cluster 3	13256	0.22	11.168	Log-Normal	-0.8937	3.2098
Cluster 4	275	0.01	123.263	Gamma	0.3739	329.6922
Cluster 5	1753	0.05	34.676	Gamma	0.3624	95.6827
Cluster 6	29546	0.20	4.138	Weibull	0.4651	1.8461

- Cluster sizes are different and often significant
- Heterogeneity in distribution parameters (different *scale* parameters)
- Decreasing hazard rate

Outline

Significance of Clustering Criteria

Could the same clusters have been found using some other static criteria?

Significance of Clustering Criteria

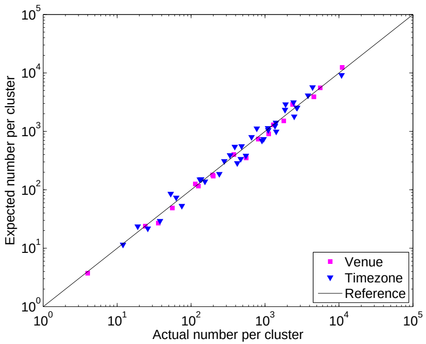
Could the same clusters have been found using some other static criteria?

- Cluster by venue: Work, Home, School
- Cluster by Time zone: 6 different time zones

Significance of Clustering Criteria

Could the same clusters have been found using some other static criteria?

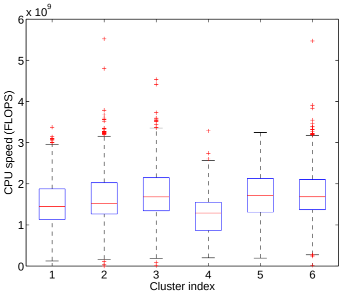
- Cluster by venue: Work, Home, School
- Cluster by Time zone: 6 different time zones



Significance of Clustering Criteria

Could the same clusters have been found using some other static criteria?

- Cluster by venue: Work, Home, School
- Cluster by Time zone: 6 different time zones
- Cluster by CPU speed



Outline

Scheduling Implications

Scheduling accuracy

Global model vs. Individual cluster model

Scheduling Implications

Scheduling accuracy

Global model vs. Individual cluster model

Ex: Completion probability of a 24-hour task:

Scheduling Implications

Scheduling accuracy

Global model vs. Individual cluster model

Ex: Completion probability of a 24-hour task:

- Global model: <20%
- Cluster 4: 70%

Scheduling Implications

Scheduling accuracy

Global model vs. Individual cluster model

Ex: Completion probability of a 24-hour task:

- Global model: <20%
- Cluster 4: 70%

Resource Selection/Replication

Scheduling Implications

Scheduling accuracy

Global model vs. Individual cluster model

Ex: Completion probability of a 24-hour task:

- Global model: <20%
- Cluster 4: 70%

Resource Selection/Replication

- Single job: Prediction of task failure

Scheduling Implications

Scheduling accuracy

Global model vs. Individual cluster model

Ex: Completion probability of a 24-hour task:

- Global model: <20%
- Cluster 4: 70%

Resource Selection/Replication

- Single job: Prediction of task failure
- Multi-job: How the task size distribution follows the availability distribution

Related Work

Different from other research

- Measurement

Related Work

Different from other research

- Measurement
  - Resource type: home, work, and school

Related Work

Different from other research

- Measurement
  - Resource type: home, work, and school
  - Scale: 200,000 hosts

Related Work

Different from other research

- Measurement
    - Resource type: home, work, and school
    - Scale: 200,000 hosts
    - Duration: 1.5 years

Related Work

Different from other research

- Measurement
  - Resource type: home, work, and school
  - Scale: 200,000 hosts
  - Duration: 1.5 years
  - Availability : CPU availability

Related Work

Different from other research

- Measurement
  - Resource type: home, work, and school
  - Scale: 200,000 hosts
  - Duration: 1.5 years
  - Availability : CPU availability
- Modelling

Related Work

Different from other research

- Measurement
  - Resource type: home, work, and school
  - Scale: 200,000 hosts
  - Duration: 1.5 years
  - Availability : CPU availability
- Modelling
  - Classification according to randomness tests

Related Work

Different from other research

- Measurement
  - Resource type: home, work, and school
  - Scale: 200,000 hosts
  - Duration: 1.5 years
  - Availability : CPU availability
- Modelling
  - Classification according to randomness tests
  - Cluster-based Model vs Global Model

Conclusion and Future Work

Discovering availability models for host subsets from a distributed system

Conclusion and Future Work

Discovering availability models for host subsets from a distributed system

Conclusion

- Methodology

Conclusion and Future Work

Discovering availability models for host subsets from a distributed system

Conclusion

- Methodology
  - Remove outliers

Conclusion and Future Work

Discovering availability models for host subsets from a distributed system

Conclusion

- Methodology
  - Remove outliers
  - Classification based on the randomness tests (iid vs non-iid)

Conclusion and Future Work

Discovering availability models for host subsets from a distributed system

Conclusion

- Methodology
  - Remove outliers
  - Classification based on the randomness tests (iid vs non-iid)
  - Partitioning hosts into subsets by their availability distribution

Conclusion and Future Work

Discovering availability models for host subsets from a distributed system

Conclusion

- Methodology
  - Remove outliers
  - Classification based on the randomness tests (iid vs non-iid)
  - Partitioning hosts into subsets by their availability distribution
- Modelling (Apply the methodology for the SETI@home)

Conclusion and Future Work

Discovering availability models for host subsets from a distributed system

Conclusion

- Methodology
  - Remove outliers
  - Classification based on the randomness tests (iid vs non-iid)
  - Partitioning hosts into subsets by their availability distribution
- Modelling (Apply the methodology for the SETI@home)
  - 34% of hosts have truly random availability intervals

Conclusion and Future Work

Discovering availability models for host subsets from a distributed system

Conclusion

- Methodology
  - Remove outliers
  - Classification based on the randomness tests (iid vs non-iid)
  - Partitioning hosts into subsets by their availability distribution
- Modelling (Apply the methodology for the SETI@home)
  - 34% of hosts have truly random availability intervals
  - Six clusters with three different distributions: Gamma, Weibull, and Log-normal

Conclusion and Future Work

Discovering availability models for host subsets from a distributed system

Conclusion

- Methodology
  - Remove outliers
  - Classification based on the randomness tests (iid vs non-iid)
  - Partitioning hosts into subsets by their availability distribution
- Modelling (Apply the methodology for the SETI@home)
  - 34% of hosts have truly random availability intervals
  - Six clusters with three different distributions: Gamma, Weibull, and Log-normal

Conclusion and Future Work

Discovering availability models for host subsets from a distributed system

Conclusion

- Methodology
  - Remove outliers
  - Classification based on the randomness tests (iid vs non-iid)
  - Partitioning hosts into subsets by their availability distribution
- Modelling (Apply the methodology for the SETI@home)
  - 34% of hosts have truly random availability intervals
  - Six clusters with three different distributions: Gamma, Weibull, and Log-normal

Future Work

- Apply the result for improving makespan of DAG-applications
- Explore ability of clustering dynamically while the system is on-line

Failure Trace Archive

http://fta.inria.fr

- Repository of availability traces of parallel and distributed systems, and tools for analysis
- Facilitate design, validation and comparison of fault-tolerance algorithms and models
- 15 data sets including SETI@home data set

Failure Trace Archive

http://fta.inria.fr

- Repository of availability traces of parallel and distributed systems, and tools for analysis
- Facilitate design, validation and comparison of fault-tolerance algorithms and models
- 15 data sets including SETI@home data set

More Details

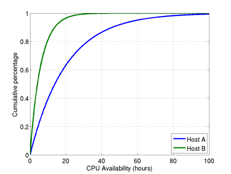
- Poster Session at MASCOTS 2009 (Today 19:00-21:00)
- Website: http://fta.inria.fr

Thank You

Questions?

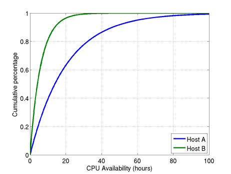
Distance Metrics

Distance between CDF of two hosts



Distance Metrics

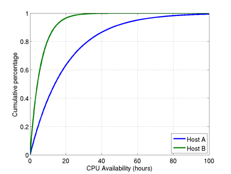
Distance between CDF of two hosts



- Kolmogorov-Smirnov: $D_{n,m} = sup \mid F_n(x) - G_m(x) \mid$

Distance Metrics

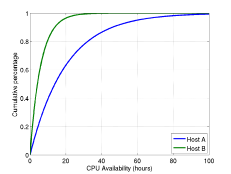
Distance between CDF of two hosts



- Kolmogorov-Smirnov: $D_{n,m} = sup \mid F_n(x) - G_m(x) \mid$
- Kuiper: $V_{n,m} = sup \mid F_n(x) - G_m(x) \mid + sup \mid G_m(x) - F_n(x) \mid$

Distance Metrics

Distance between CDF of two hosts

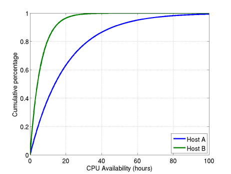


- Kolmogorov-Smirnov: $D_{n,m} = sup \mid F_n(x) - G_m(x) \mid$
- Kuiper: $V_{n,m} = sup \mid F_n(x) - G_m(x) \mid + sup \mid G_m(x) - F_n(x) \mid$
- Cramer-von Mises:

$$T_{n,m} = \frac{nm}{(n+m)^2} \left\{ \sum_{i=1}^{n} [F_n(x_i) - G_m(x_i)]^2 + \sum_{j=1}^{m} [F_n(y_j) - G_m(y_j)]^2 \right\}$$

Distance Metrics

Distance between CDF of two hosts



- Kolmogorov-Smirnov: $D_{n,m} = sup \mid F_n(x) - G_m(x) \mid$
- Kuiper: $V_{n,m} = sup \mid F_n(x) - G_m(x) \mid + sup \mid G_m(x) - F_n(x) \mid$
- Cramer-von Mises:

$$T_{n,m} = \frac{nm}{(n+m)^2} \left\{ \sum_{i=1}^{n} [F_n(x_i) - G_m(x_i)]^2 + \sum_{j=1}^{m} [F_n(y_j) - G_m(y_j)]^2 \right\}$$

- Anderson-Darling: $Q_n = \int_{-\infty}^{\infty} [F(x) - F_n(x)]^2 \psi(F(x)) dF$

$\psi(F(x)) = \frac{1}{F(x)(1-F(x))}$

Fitting with Hyper-Exponential

Fitting Method:

- Expectation Maximization (EM) [using EMpht package]
  - Accurate
  - Flexible
  - Slow

Fitting with Hyper-Exponential

Fitting Method:

- Expectation Maximization (EM) [using EMpht package]
  - Accurate
  - Flexible
  - Slow
- Moment Matching (MM)
  - Less accurate
  - Not flexible
  - Very fast

Fitting with Hyper-Exponential

Fitting Method:

- Expectation Maximization (EM) [using EMpht package]
  - Accurate
  - Flexible
  - Slow
- Moment Matching (MM)
  - Less accurate
  - Not flexible
  - Very fast

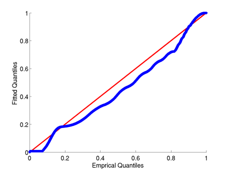
We used MM for 2-phase hyper-exponential by the first two moments as follows:

$$p = \frac{1}{2}\left(1 - \sqrt{\frac{CV^2-1}{CV^2+1}}\right)$$

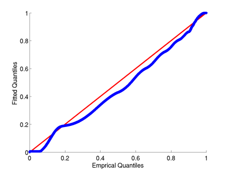
$$\lambda_1 = \frac{2p}{\mu}$$

$$\lambda_2 = \frac{2(1-p)}{\mu}$$

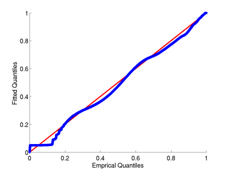
PP-Plots



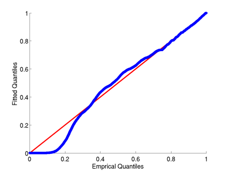
(a) Cluster 1



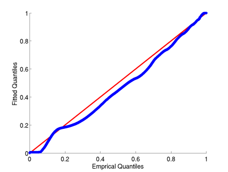
(b) Cluster 2



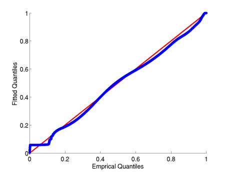
(c) Cluster 3



(d) Cluster 4



(e) Cluster 5



(f) Cluster 6

Goodness of Fit Tests

Data sets	Hyper-Exponential (MM) Parameters	AD	KS	Hyper-Exponential (EM) Parameters	AD	KS
All iid hosts	$p_1 = 0.024$ $\lambda_1 = 0.004$ $p_2 = 0.976$ $\lambda_2 = 0.154$	0.026	0.005	$p_1 = 0.197$ $\lambda_1 = 0.0179$ $p_2 = 0.279$ $\lambda_2 = 29.171$ $p_3 = 0.524$ $\lambda_3 = 0.316$	0.531	0.375
Cluster 1	$p_1 = 0.115$ $\lambda_1 = 0.003$ $p_2 = 0.885$ $\lambda_2 = 0.019$	0.287	0.119	$p_1 = 0.180$ $\lambda_1 = 14.401$ $p_2 = 0.820$ $\lambda_2 = 0.009$	0.450	0.318
Cluster 2	$p_1 = 0.114$ $\lambda_1 = 0.004$ $p_2 = 0.886$ $\lambda_2 = 0.032$	0.275	0.113	$p_1 = 0.183$ $\lambda_1 = 12.338$ $p_2 = 0.817$ $\lambda_2 = 0.015$	0.512	0.403
Cluster 3	$p_1 = 0.030$ $\lambda_1 = 0.005$ $p_2 = 0.970$ $\lambda_2 = 0.174$	0.005	0.000	$p_1 = 0.341$ $\lambda_1 = 0.031$ $p_2 = 0.261$ $\lambda_2 = 71.852$ $p_3 = 0.398$ $\lambda_3 = 1.923$	**0.561**	**0.434**
Cluster 4	$p_1 = 0.136$ $\lambda_1 = 0.002$ $p_2 = 0.864$ $\lambda_2 = 0.014$	0.448	0.273	$p_1 = 0.694$ $\lambda_1 = 0.020$ $p_2 = 0.306$ $\lambda_2 = 0.003$	0.473	0.274
Cluster 5	$p_1 = 0.105$ $\lambda_1 = 0.006$ $p_1 = 0.895$ $\lambda_2 = 0.052$	0.295	0.122	$p_1 = 0.173$ $\lambda_1 = 13.374$ $p_2 = 0.827$ $\lambda_2 = 0.024$	0.523	0.393
Cluster 6	$p_1 = 0.010$ $\lambda_1 = 0.005$ $p_2 = 0.990$ $\lambda_2 = 0.478$	0.114	0.038	$p_1 = 0.516$ $\lambda_1 = 0.131$ $p_2 = 0.150$ $\lambda_2 = 163.771$ $p_3 = 0.334$ $\lambda_3 = 2.411$	**0.572**	**0.470**