Toward Autonomic Grids: Analyzing the Job Flow with Affinity Streaming

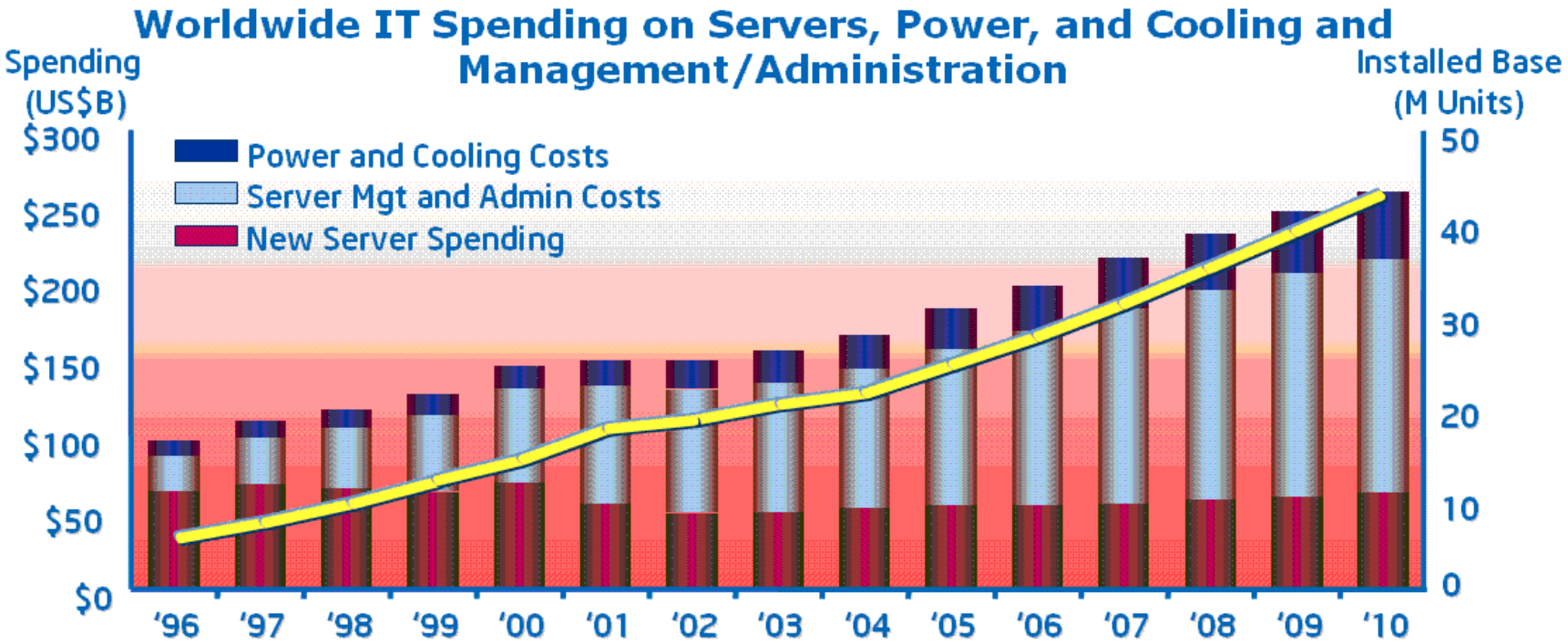
Xiangliang Zhang, Cyril Furtlehner
Julien Perez, Cécile Germain-Renaud, Michèle Sebag

TAO team – INRIA CNRS
Université de Paris-Sud, F-91405 Orsay Cedex, France

29 June 2009

Contents

Motivations of Autonomic Computing



Rate of Server Management and Power/Cooling Cost Increase

Source: IDC

Goals of Autonomic Computing

AUTONOMIC VISION & MANIFESTO
http://www.research.ibm.com/autonomic/manifesto/

Self-managing system with the ability of

- Self-healing: detect, diagnose and repair problems
- Self-configuring: automatically incorporate and configure components
- Self-optimizing: ensure the optimal functioning wrt defined requirements
- Self-protecting: anticipate and defend against security breaches

Data Mining for Autonomic Computing

Autonomic Grid Computing System

EGEE gird

68,000 CPU

Flow of jobs

300,000/day

G-StrAP:
Multi-scale Job Stream monitoring

Summarized
Outputs

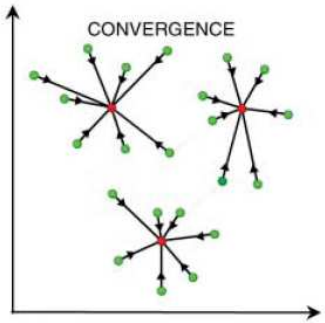
System
Administrator

Contents

G-StrAP: relies on Affinity Propagation (AP)

Affinity Propagation (AP) [Frey2007]

- statistic physics algorithm for clustering
  (based on *messaging passing*)
- a cluster = an exemplar
  (akin k-centers)
- the model = set of {exemplar, frequency}

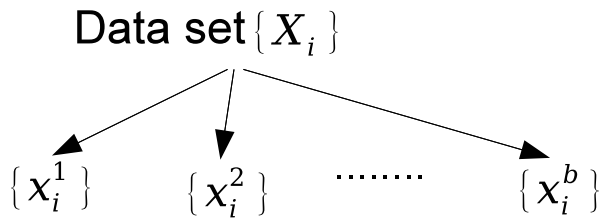


Why AP ?

- Traceability: real jobs as exemplars
  because of categorical variables, e.g., userid, queue name etc
- No prior knowledge of K, number of clusters
- quasi optimality wrt. information loss
  —> stability [Meila2006]

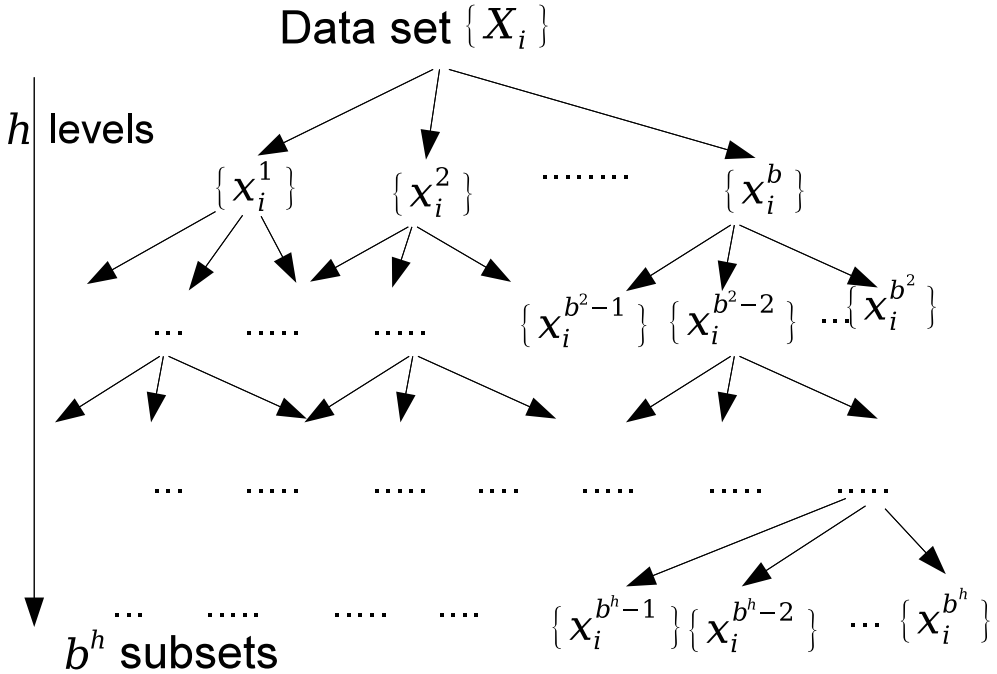
1 SCALABILITY : from $\mathcal{O}(N^2 \log N)$ to $\mathcal{O}(N^{\frac{h+2}{h+1}})$

Hierarchical Affinity Propagation



1. subset clustering → sub-exemplars
2. sub-exemplars aggregation
3. cluster the aggregation –> exemplars

Zhang et. al ECML08



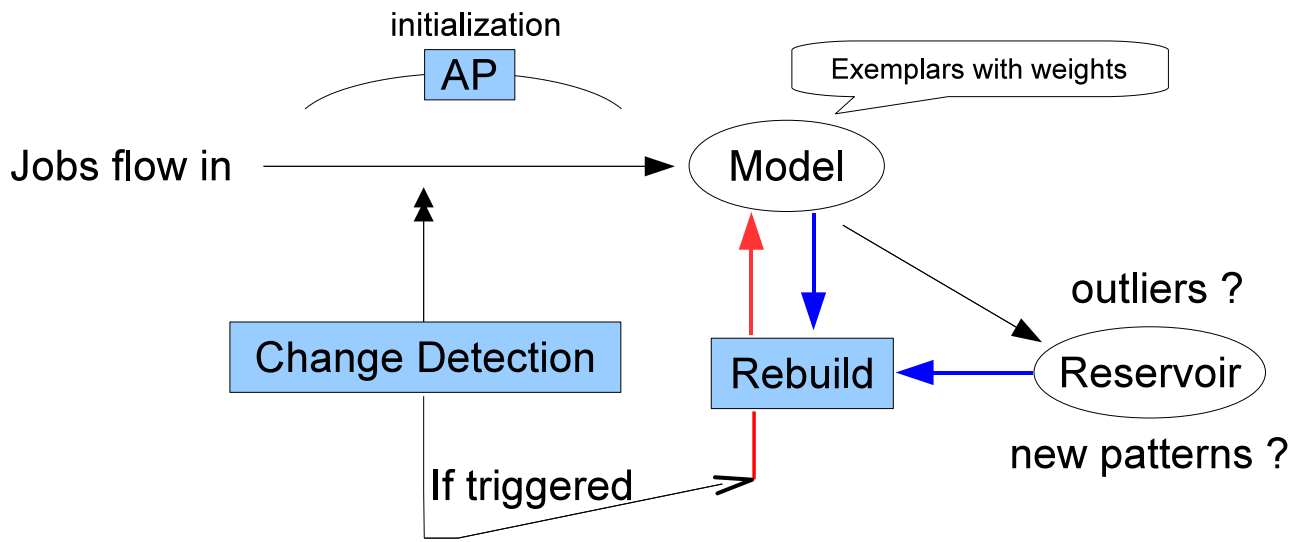
- b: branching factor
- h: number of levels

- negligible infromation loss (proof in the paper)

2 Non stationary distribution

- various Virtual Organization
- number and expertise of users

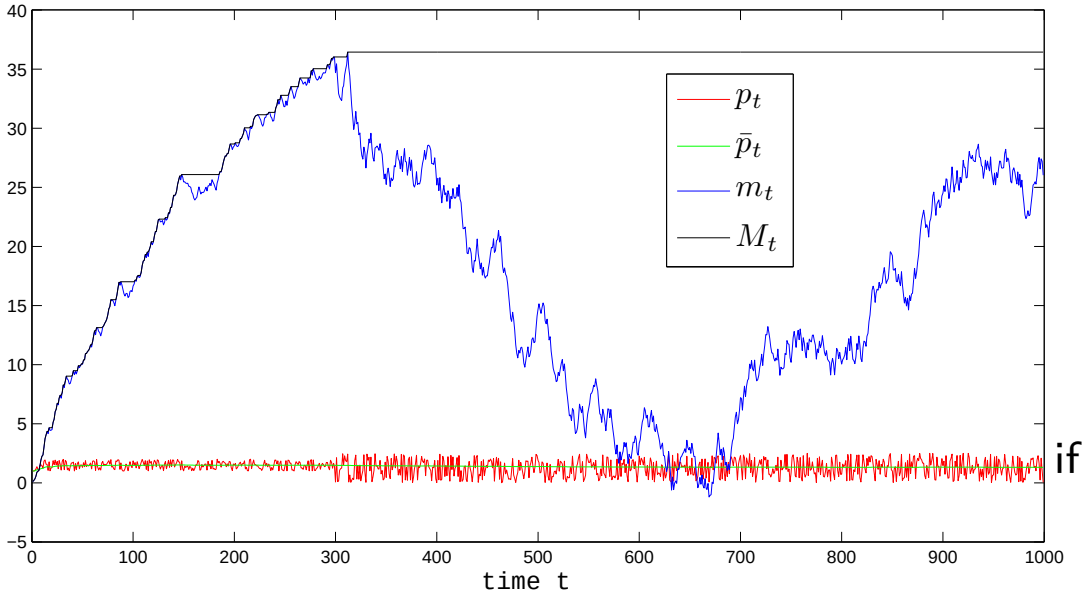
Streaming AP (StrAP)



Non stationary distribution, continue

Page-Hinkley statistic
(Cumulative-Sum-like test)

[Page54,Hinkley70]



p_t changing distribution

$$\bar{p}_t = \frac{1}{t} \sum_{\ell=1}^{t} p_\ell$$

$$m_t = \sum_{\ell=1}^{t} (p_\ell - \bar{p}_\ell + \delta)$$

$$M_t = max\{m_\ell\}$$

$$PH_t = M_t - m_t$$

if $PH_t > \lambda$, changed detected

How to set λ ???

Self-adaptive change detection test

Self-adapt λ $\equiv$ An optimization problem

BIC: $\mathcal{F}_\lambda = \frac{1}{|C|}\sum_{i=1}^{|C|}\left(\frac{1}{n_i}\sum_{e_j \in C_i} d(e_j, e_i^*)\right) + \varphi\frac{\rho}{2}\log N + \eta O_t$

$\propto$ loss + size of model + percentage of outlier

OPTIMIZATION:

- ϵ-greedy search from a finite set of λ values

$$\lambda = argmin\{\mathbf{E}(F_\lambda\}),$$

λ_1	λ_2	λ_3	λ_4	...
$\mathbf{E}(F_{\lambda_1})$	$\mathbf{E}(F_{\lambda_2})$	$\mathbf{E}(F_{\lambda_3})$	$\mathbf{E}(F_{\lambda_4})$	...

- Gaussian Process Regression based on $\{\lambda_i, F_{\lambda_i}\}$ continuous value of λ is generated

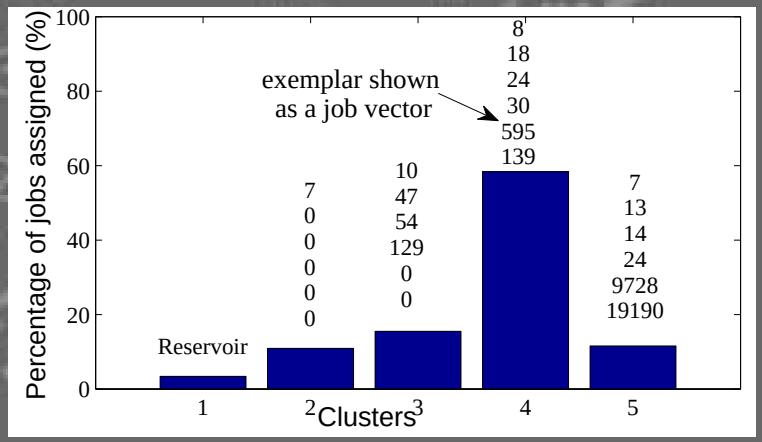
Contents

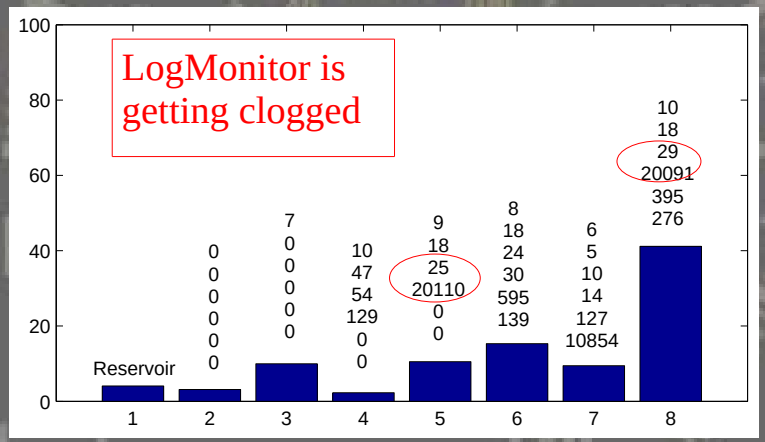
G-StrAP : Multi-scale Realtime Monitor

Dashboard for Monitoring

G-StrAP Dashboard for Grid Monitoring

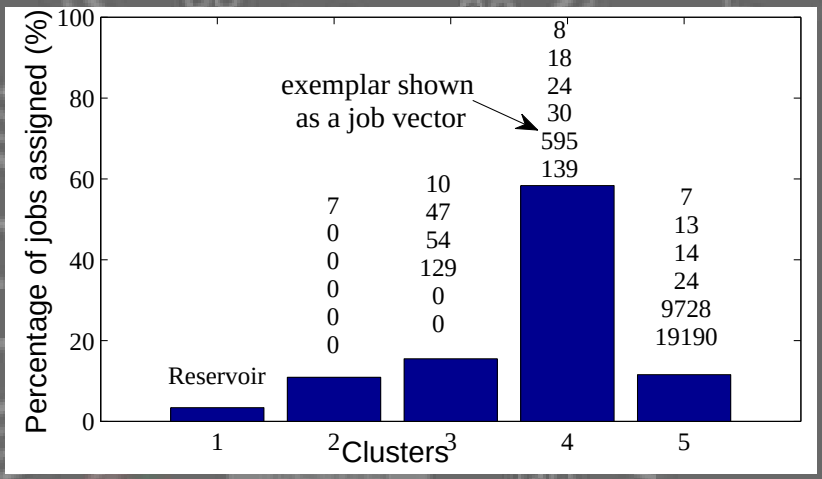
Online Monitoring

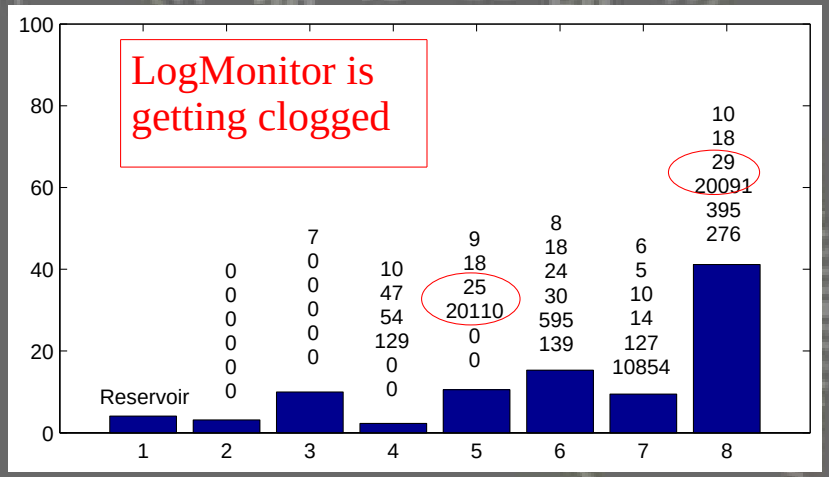




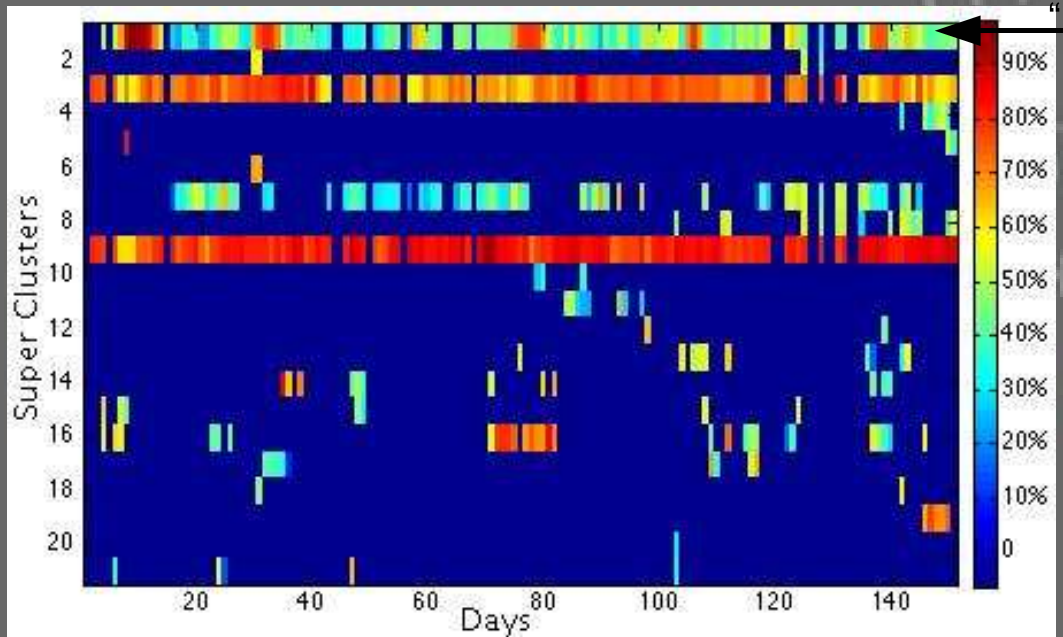
G-StrAP Dashboard for Grid Monitoring

Online Monitoring





Off-line Analysis



"Early stopped error"
When and Who ?

- no prior knowledge about failure patterns
- summarizing Terabyte data

Contents

Discussion and Conclusion

G-StrAP:

- linear computational complexity $\mathcal{O}(N^{1+\epsilon})$ with bounded information loss
- dealing with non stationary distribution data with self-adaptive change detection test

G-StrAP for Grid Monitoring

- Traceability of failures
- providing multi-scale models to describing the status of Grid
  - online description of different type of job patterns
  - offline globally analysis
- good quality clustering wrt. supervised learning

Perspectives

Theoretical Perspectives:

- limitation 1: AP single parameter $s*$
  —> self-adapt $s*$
- limitation 2: better coping with flowing patterns
  —> Hierarchical G-StrAP, absorb new patterns from reservoir more efficiently

Applicative Perspectives:

- profiling users
  —> a user = a histogram of job exemplars
  - customized interface
  - characterize evolution of users or virtual organizations
- build crisis scenari to test Middleware

Xiangliang ZHANG

xlzhang@lri.fr

http://www.lri.fr/∼xlzhang