

CS528

**Dynamic Resource Allocation Using
Virtual Machines for Cloud
Computing Environment**

A Sahu

Dept of CSE, IIT Guwahati

Outline

- Zhen Xiao, Weijia Song, and Qi Chen “**Dynamic Resource Allocation Using Virtual Machines for Cloud Computing Environment**”, IEEE Trans. on Parallel & Dist. Computing., JUNE 2013

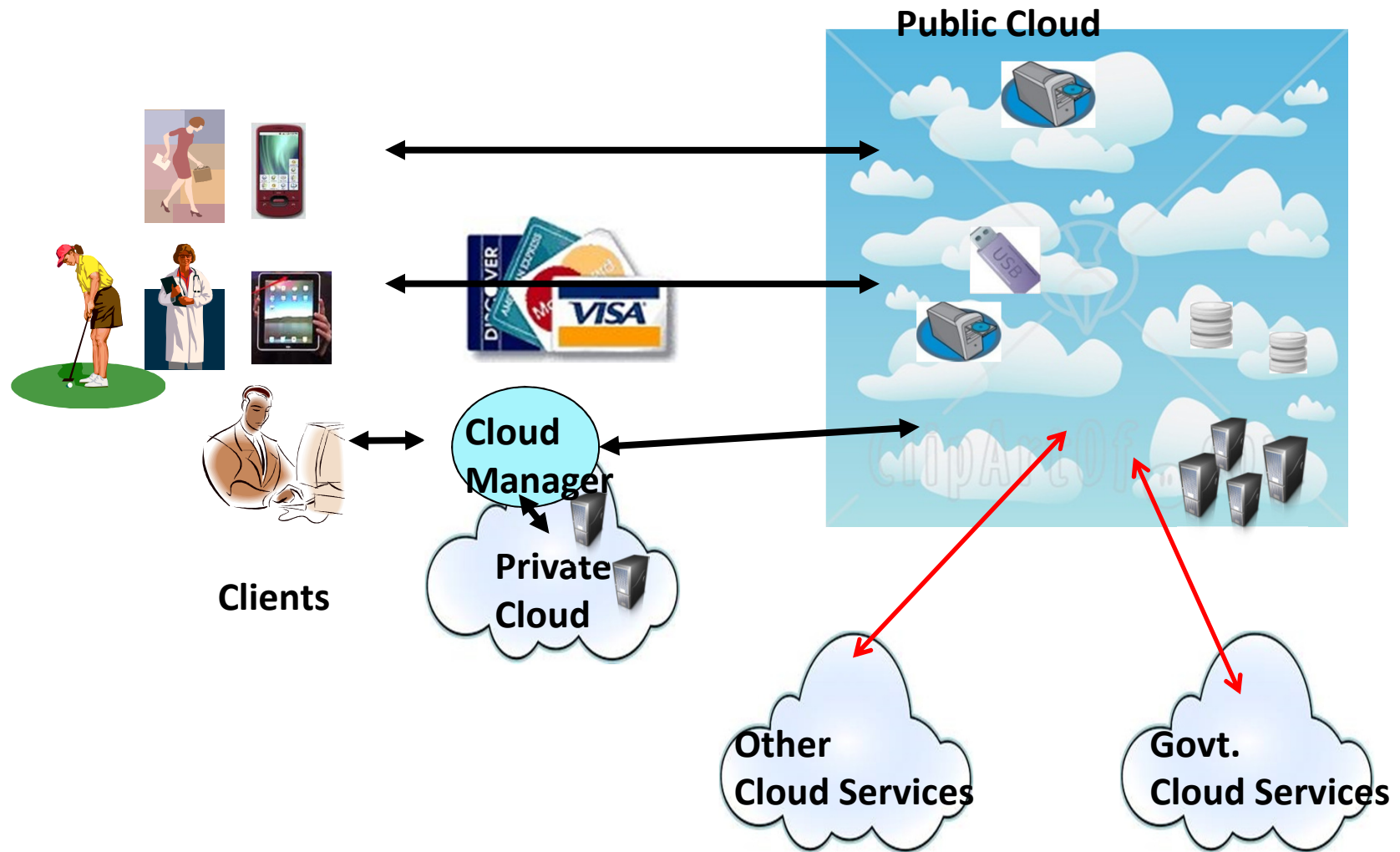
Outline

- Work load Prediction
 - EWMA, FUSD
- Dynamic Resource Management
 - Scale up and Scale Down
- SLAV

Reference

Zhen Xiao, Weijia Song, and Qi Chen “**Dynamic Resource Allocation Using Virtual Machines for Cloud Computing Environment**”, IEEE Trans. on Parallel & Dist. Computing., JUNE 2013

Clouds offer Subscription-Oriented IT Services: {compute, apps, data,..} as a Service (..aaS)



Basic Approaches

- Cloud computing allows business customers
 - To scale up and scale down
 - Their resource usage based on needs
- Allocate data center resources dynamically
 - based on application demands
- Supporting green computing
 - by optimizing the number of servers in use
- “skewness” to measure
 - unevenness in the multidimensional resource utilization of a server

Service Level Agreement

- Service Level Agreement
 - Agreement between user and provider
 - Cost, resources, QoS, timely manner
- Agreement in Cloud User and Provider
 - Want to execute a task with time deadline D for E time and require C cpu, M memory unit
 - Cost Payable to Cloud provider by User
 - If Violate, provider needs to penalty

Service Level Agreement

- Real life Example : SLA is every where
- Giving Party Food contract to Caterer
 - Caterer A is good one, what ever he say he do
 - Caterer B is not good: he say some thing but do not provide what he says, always he provide less.
- Transport service : Taking pickup/drop/tour service from Travel agent/Airlines
 - Travel agent TA1: Gives proper services, always service in times, no late, all service are excellent
 - Travel agent TA2: Don't gives proper services, always service in, late, service are bellow expectation based on price.

Resource Allocation Approaches

- **Service Level Agreement**

- Agreement between user and provider
- Cost, resources, QoS, timely manner

- **Overload Avoidance**

- Capacity of a PM should be sufficient
- to satisfy the resource needs of all VMs running on it.
- Otherwise, the PM is overloaded and can lead to degraded performance of its VMs.
- Hot threshold: Utilization of a host $U_h < U_{\text{hotthreshold}}$

Resource Allocation Approaches

- Green Computing
 - Number of PMs used should be minimized
 - as long as they can still satisfy the needs of all VMs.
 - Idle PMs can be turned off to save energy.
- Term used to do
 - **Cold Threshold**: Under utilized Server
 - **Green Computing Threshold** : Average Utilization of all active PM is below this → Some server are under utilized

Resource Allocation Approaches

- Assume 10 servers are there
- Average utilization is 40%
- Can we say some server are underutilized $U_h < 25\%$
 - Yes, may be some server have
- If a server **H** have $U_h < 25\%$
 - Migrate VMs of **H** to other servers
 - Switch off **H** to save power
- Migrate to VMs of **H** to other server **S**
 - where $U_s < 90\%$ after migration
 - Should not be overloaded

Resource Prediction

- Given Catering to Caterer A :100 Plates of food
 - Caterer will estimates 80% of guest will come
 - Average amount food (F_{avg}) per person consumes A knows in general
 - He (A) will prepare : $80 * F_{avg}$ food for the party.
- Train services
 - During holiday period people go for travels
 - During Holi/Diwali/Dushehara require special train
- Customer at shop
 - Night time customer will be less
 - Evening time is peak time ==>prepare for that

Resource Prediction

- General prediction of online series
- May be more complex
- Most people use based on history, last one/n time slots
- Re Call : CPU burst time prediction by OS for an application
 - Application have series Compute and I/O requests



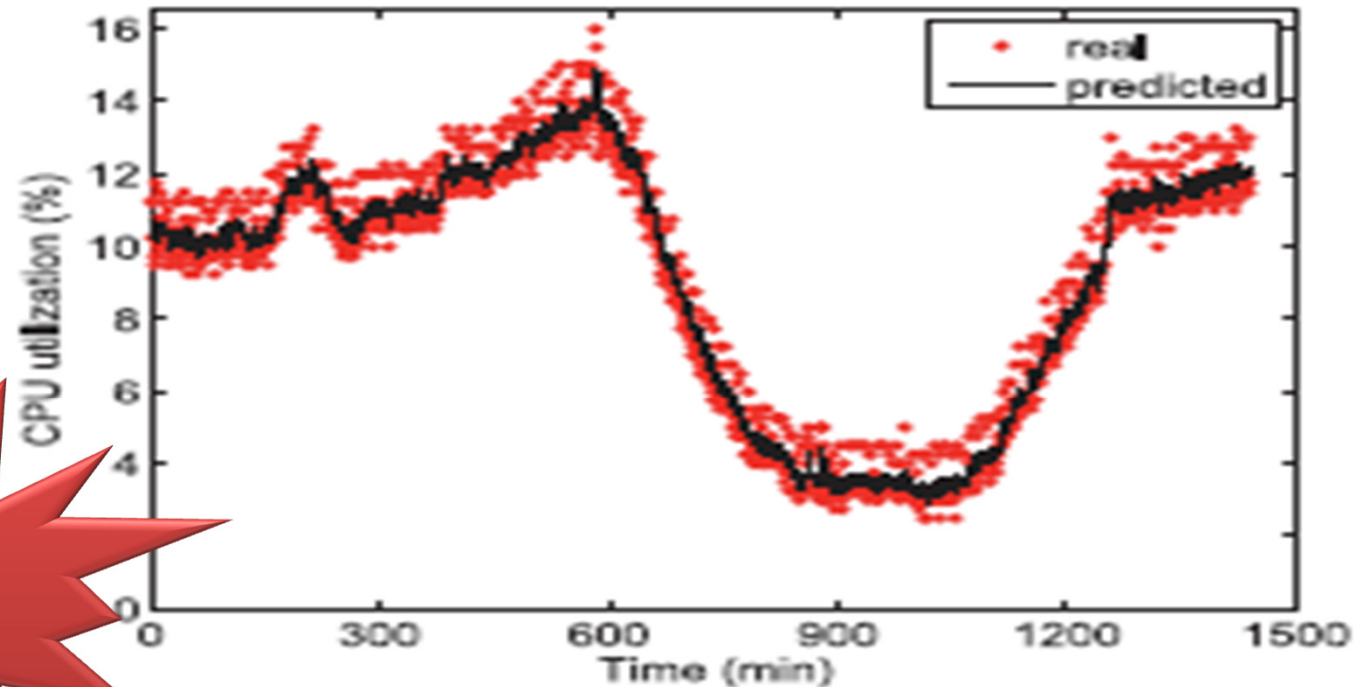
- OS schedule task on processor by estimating the CPU time of task for next CPU burst

Resource Prediction

- Exponential Weighted Moving Average (EWMA)

$$E(t) = \alpha * E(t-1) + (1-\alpha) * O(t)$$

$E(t)$ and $O(t)$: Estimated and Observed load at time t



$E(t) < O(t)$

(a) EWMA: $\alpha = 0.7$, $W = 1$

Resource Prediction Approaches

- If $E(t) < O(t)$, estimated is less than observed
 - Party meals: Estimated 80 but came 90: 10 will not get any thing
 - **Safe side: Estimate should be a bit higher to maintain the agreement (SLA)**
- **When observed usage is going down**
 - **be conservative in reducing our estimation**
- Solution to Problem of EWMA
$$E(t) = -|\alpha| E(t-1) + (1+|\alpha|)O(t)$$
$$= O(t) + |\alpha| (O(t) - E(t-1))$$
- When $O(t) > E(t)$ use α^u
- When $O(t) < E(t)$ use α^d

Resource prediction Approaches

- FUSD: Fast Up-Slow Down

- Up direction resource increment aggressive
- Down direction : be conservative

if ($O(t-1) > E(t-1)$) // FU $\alpha^u = -0.2$

$$E(t) = \alpha^u * E(t-1) + (1 - \alpha^u) * O(t)$$

else //SD $\alpha^d = 0.7$

$$E(t) = \alpha^d * E(t-1) + (1 - \alpha^d) * O(t)$$

- FU : $E(t) = 0.2E(t-1) + 1.2 O(t) \rightarrow$

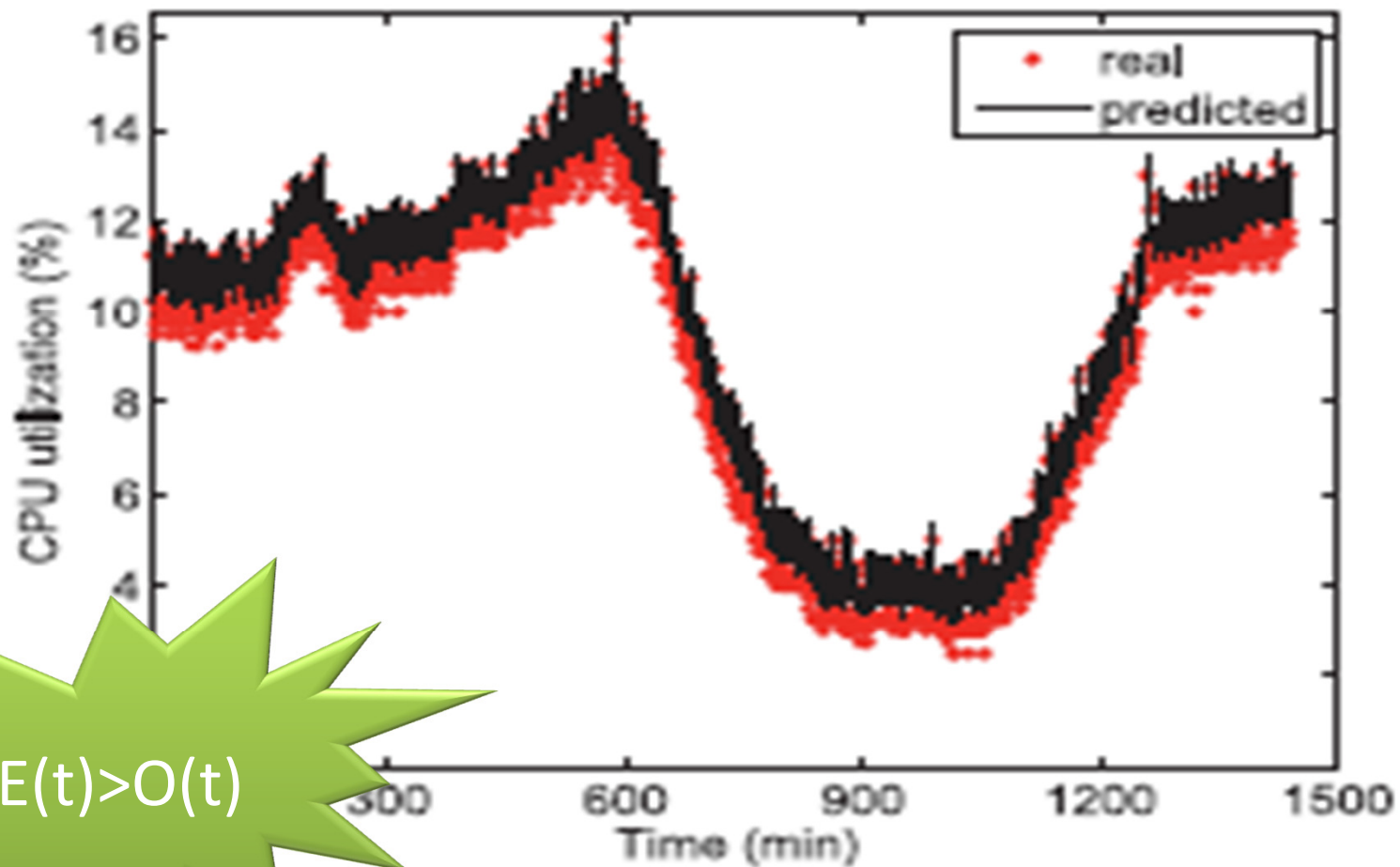
- high weight to $O(t) \rightarrow$ increase aggressively

- SD : $E(t) = 0.7E(t-1) + 0.3 O(t) \rightarrow$

- high weight to $E(t) \rightarrow$ decrease conservatively

Resource Allocation Approaches

- FUSD Example $\alpha^u = -0.2$ $\alpha^d = 0.7$
- $E(t) > O(t)$: No SLA Violation



$E(t) > O(t)$

Skewness

- Quantify unevenness in the utilization of multiple resources on a server
- **Skewness (p) = $\sqrt{\sum_{i=1}^n (\frac{r_i}{\bar{r}} - 1)^2}$**
 - \bar{r} is AVG utilization of the resources for server p
 - r_i is utilization of i_{th} resource
 - Resources are CPU, memory, disk BW, net BW
 - Skewed : CPU Util 90%, memory Util 10%

Hot and Cold Spots

- **Hot spot** : Utilization of **any** of the resources of a server **above** hot threshold
 - Hot threshold say 90% and CPU/mem of server is >90% utilized
- **Cold Spot**: Utilization of **all** of the resources of a server **below** cold threshold
- The term Temp of a server

$$\text{Temp}(p) = \sum_{r \in R} (r - r_t)^2$$

- R is set of overloaded Resources, r_t is hot threshold of r

Hot and Cold Spots

- **Green computing threshold : say 40%**
 - Average Utilization of active server are bellow this
 - Have option is Consolidate
- **Warm threshold : say 65%**
 - Average Utilization above need to start another PM
- **Consolidation limit: Max Num of cold server can be eliminated in each run (say 5%)**
 - Consolidation takes time
 - Consolidation reduce the performance a bit

Typical Values

- Number of Active Physical Machine (APM) to server the work load depend on these values
 - The value of h , g , c , w , l
- Hot Threshold $h = 0.9$
- Cold Threshold $c = 0.25$
- Warm threshold $w = 0.65$
- Green computing threshold $g = 0.4$
- Consolidation limit $l = 0.05$

Impact of threshold on APM

Lower is better (h0.95, g0.5, c0.35)

