# Automated clustering of VMs for scalable cloud monitoring and management


C. Canali
R. Lancellotti

*University of Modena and Reggio Emilia*

# *Cloud computing challenges*

- **Large datacenters → can have > 10^5 VMs**
- **Scalability problems:**
  - VMs monitoring
  - VMs management (migration, packing, ...)
- **Current approach reduce amount of data in a uniform way:**
  - Reduce sampling frequency (e.g., only 2 samples per day)
  - Reduce number of metrics considered (e.g., consider only CPU, disregard network)
- **→ Reduced monitoring effectiveness**
  - Less information available to take management decision

# Exploiting VM similarity

- **No information on VM behavior is used to improve scalability**
  - Consistent with IaaS vision
  - → Room for improvement

- **Improving scalability of monitoring and management**
  - Cluster VM with similar behavior
  - Exploit a two step approach to monitoring and management

# *Improving monitoring scalability*

- **Group similar VM together**
- **Elect a few (e.g., 3) cluster representatives**
- **Detailed monitoring of cluster rep.**
- **Reduced monitoring of other VMs**
- **Data collected can be reduced by 1 OoM**
- **Numeric example:**
  - 110 VMs, 11 metrics, sampling freq. 5 min. → ~2 M samples/day
  - 2 classes, 3 representative per class → 100K samples/day
  - Data reduced to ~1/20

# Contribution

- **Proposal: Methodology for automated clustering of VMs**
- **Two steps:**
    1. Extraction of a quantitative model of VM behavior
    2. Clustering of VMs
- **Exploit data about each VM for a short period of time** (initial dataset used for clustering)

# Methodology details

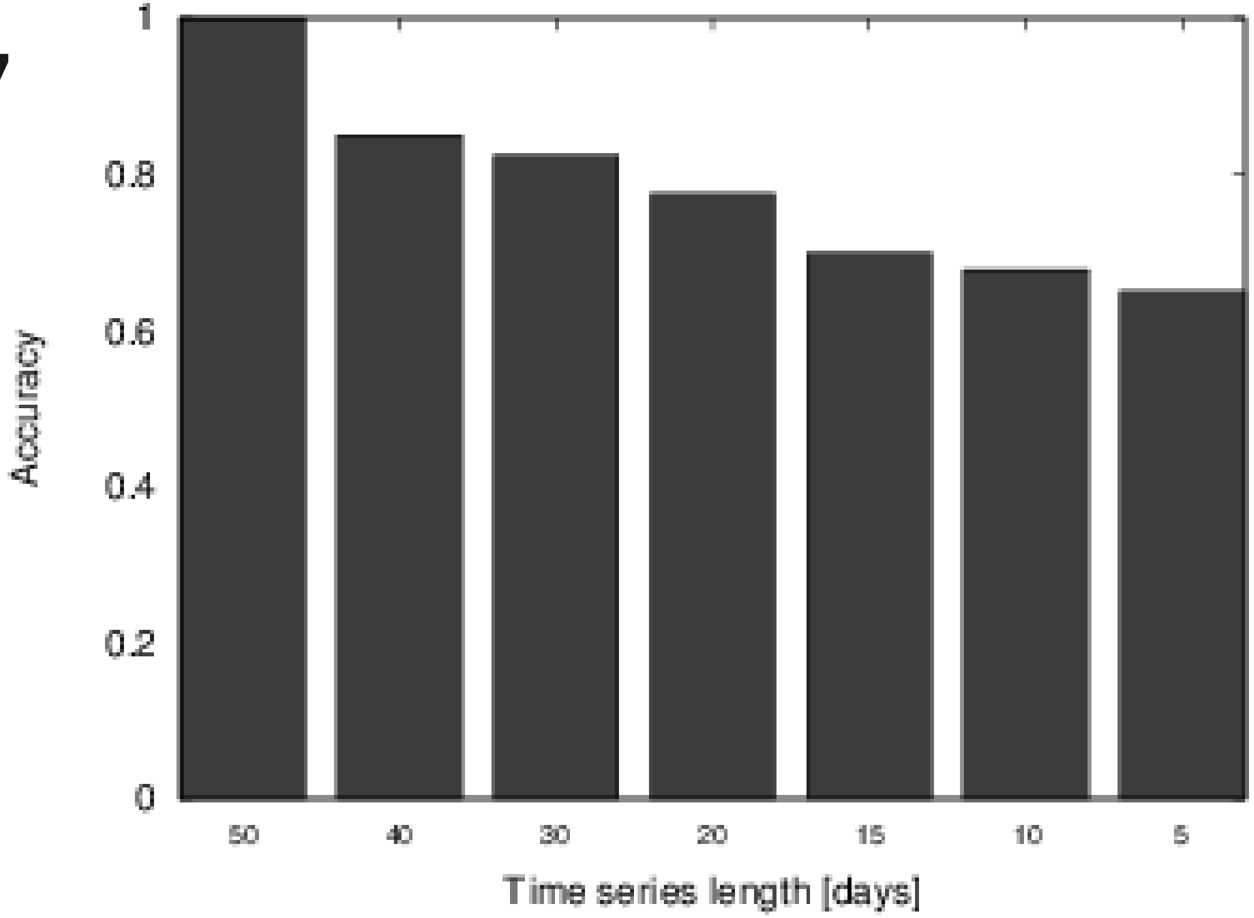
- **Extraction of a quantitative model of VM behavior**
  - *Input:* time series of metrics describing VM i behavior (X1, ... ,Xm)
  - Compute correlation matrix Si for each VM i
  - *Output:* feature vectors Vi obtained form Si
- **Clustering of VMs**
  - *Input:* feature vector Vi
  - Clustering based on k-means algorithm
  - *Output:* clustering solution

# Case study

- **Datacenter supporting a Web application**
  - Web server and DBMS
  - 110 VMs
  - 11 metrics for each VM,
  - Sampling frequency: 5 min
- **Goal: separate Web servers and DBMS**
  - Main metric: Accuracy of identification
- **Three types of analyses**
  - Impact of time series length
  - Impact of filtering techniques
  - Impact of number of nodes

# Impact of time series length

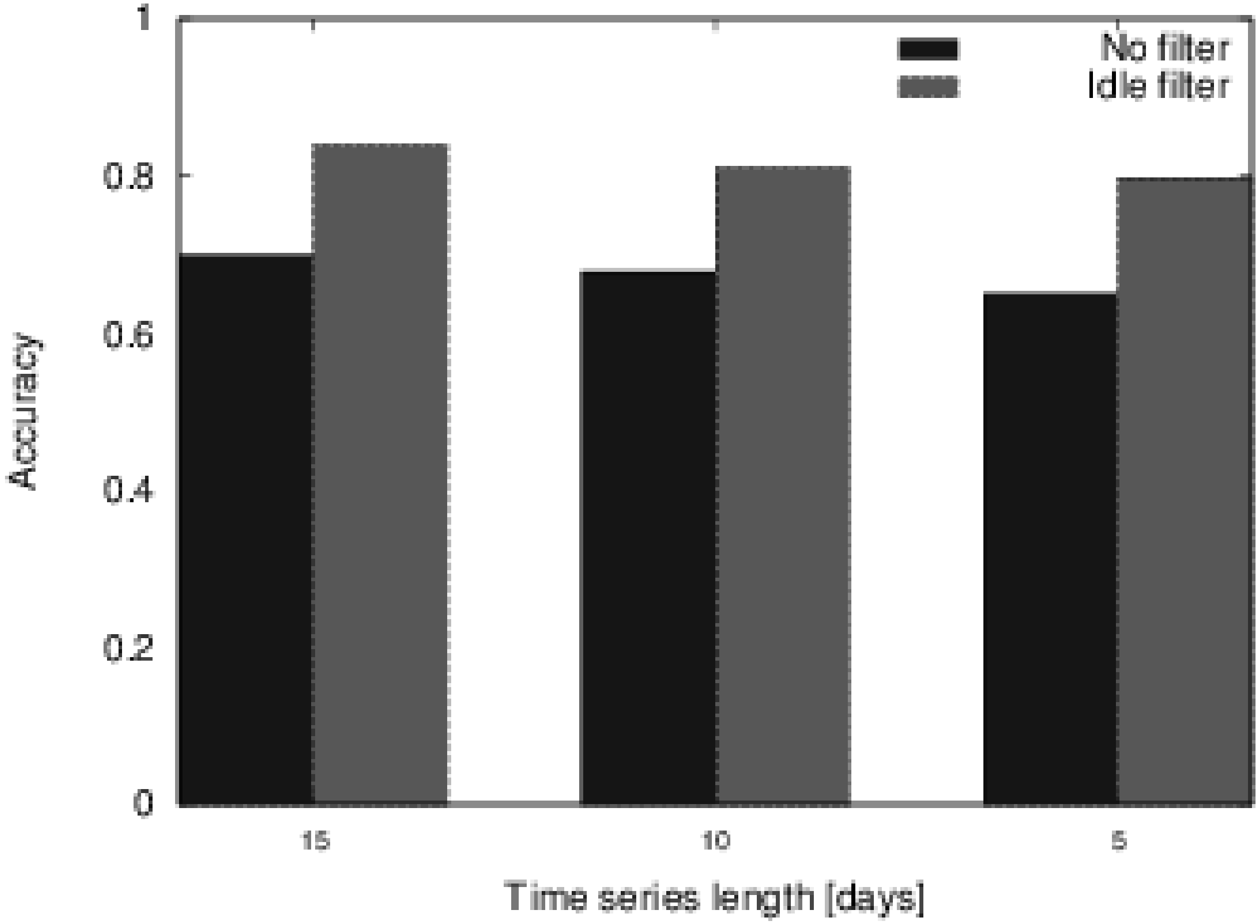
- **Reduction of available data → reduction in the accuracy of clustering**
- **Accuracy > 0.7 for time series > 20 dd**



# Impact of filtering techniques

- **Application of data filtering:**
  - Remove idle periods in time series

- **Data filtering improves performance**
  - Removal of periods providing limited information
- Accuracy >0.8 even for 5 days time series



# Impact of number of nodes

| Number of VMs | Accuracy | Clustering time [s] |
|---|---|---|
| 10 | 1 | 49.7 |
| 30 | 0.86 | 59.5 |
| 50 | 0.84 | 68.6 |
| 70 | 0.84 | 78.0 |
| 90 | 0.83 | 88.3 |
| 110 | 0.84 | 95.3 |

- **Accuracy is not adversely affected by # of VM**
  - **Accuracy ~ 0.85 for [30-110] VMs**
- **Clustering time grows linearly with # of VM**
- **We expect clustering time to remain acceptable even for large data centers**

# *Conclusion and future work*

- **Scalability in cloud systems is an open issue**
- **Proposal of novel methodology to improve scalability through clustering of similar VMs**
- **First experimental results are encouraging**
  - Accuracy >0.8 even for very short time series
- **Future research directions:**
  - Validation with more data set *(Help!)*
  - Performance improvement
  - Other approaches to model VM behavior (e.g., Bhattacharyya distance)
  - Other clustering algorithms (e.g., spectral clustering)

# Automated clustering of VMs for scalable cloud monitoring and management


C. Canali

R. Lancellotti

*University of Modena and Reggio Emilia*