CloudPin: A Root Cause Localization Framework of Shared Bandwidth Package Traffic Anomalies in Public Cloud Networks

Shize Zhang, Yunfeng Zhao, Jianyuan Lu, Biao Lyu, Shunmin Zhu, Zhiliang Wang, Jiahai Yang, Lin He, Jianping Wu

Tsinghua University, China
Alibaba Group, China

zsz16@mails.tsinghua.edu.cn

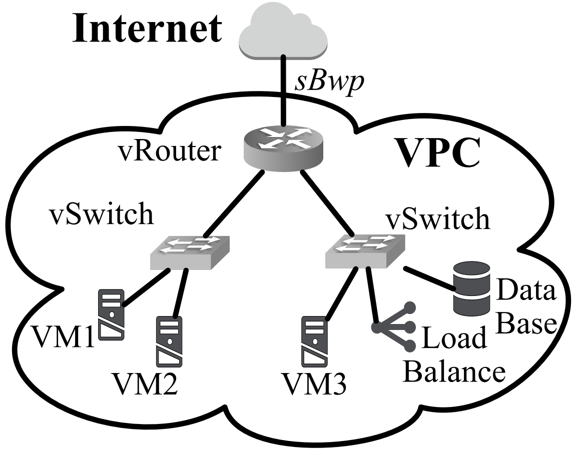
ISSRE 2021

BACKGROUND

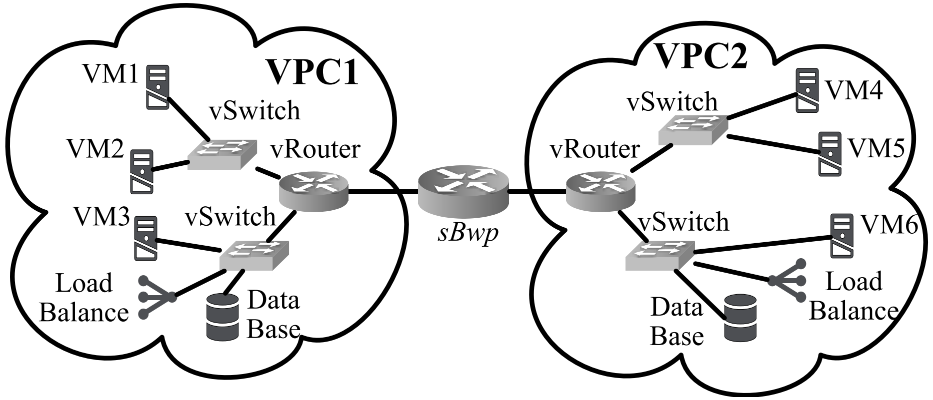
- According to the latest statistics, the worldwide cloud market grew 32% to $39.9 billion in the fourth quarter of 2020.
- With the ever-increasing number of users and services, the operation and maintenance of public clouds are facing more challenges.
- One of the most common services in public cloud networks is sharing of resources to achieve convenient management and low cost.
- For cloud users, the common sharing of resources service is the shared bandwidth package (sBwp).

BACKGROUND

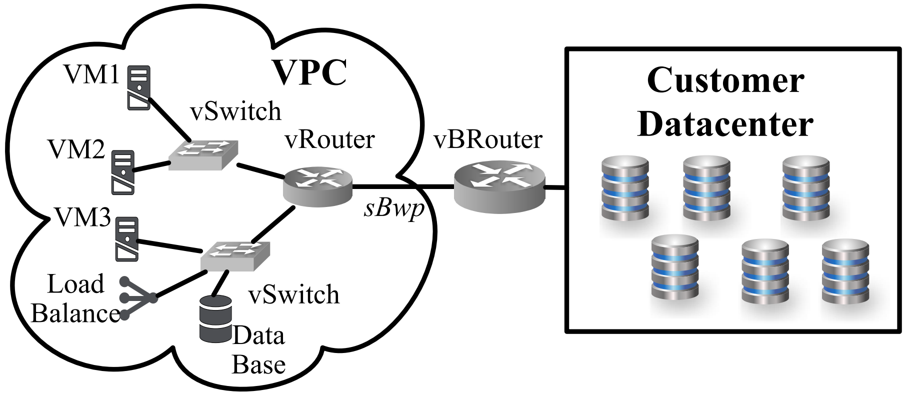
- sBwp is a user-friendly traffic management service.
- Specifically, users with a large number of VMs can choose to use sBwp service to purchase aggregate bandwidth for all VMs, rather than purchasing bandwidth for each VM individually.



Interaction with the Internet.



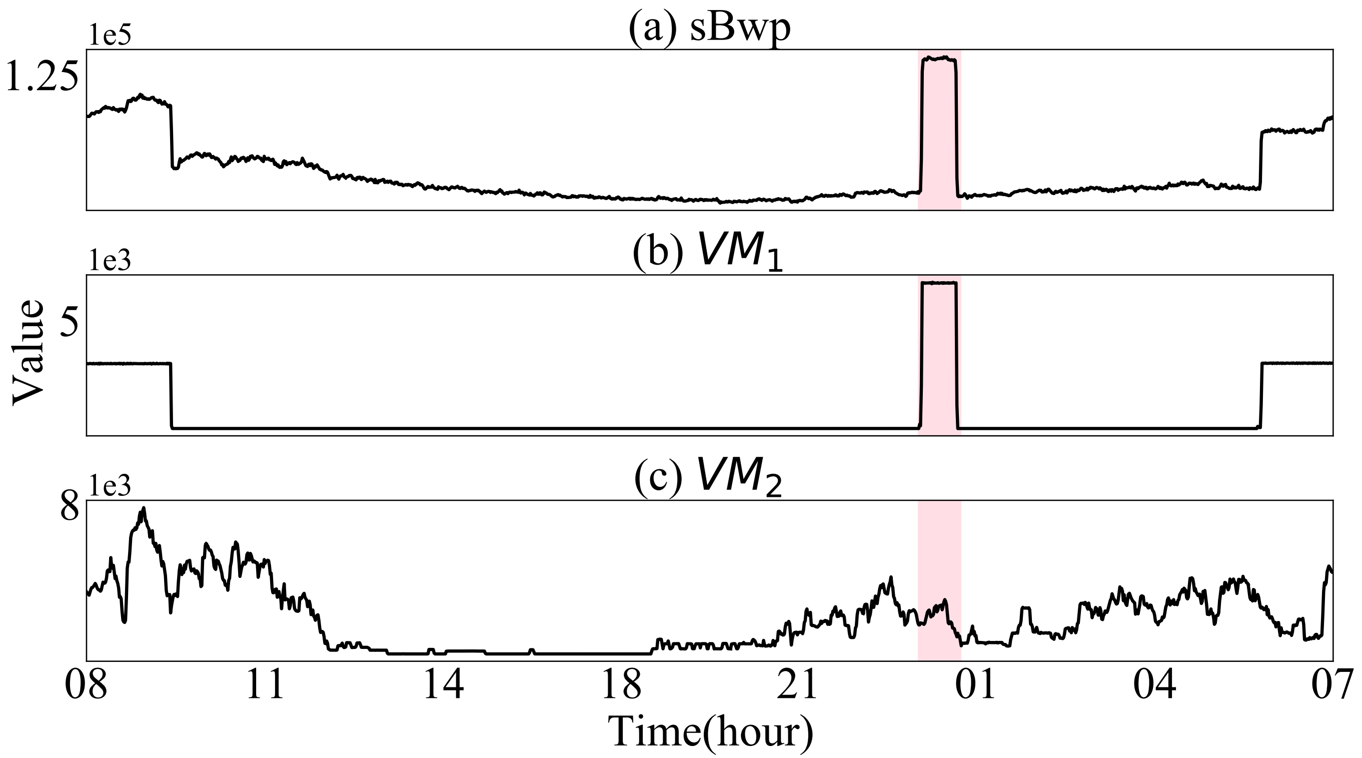
Interaction between VPCs.



Interaction with the customer Datacenter.

PROBLEM DEFINITION

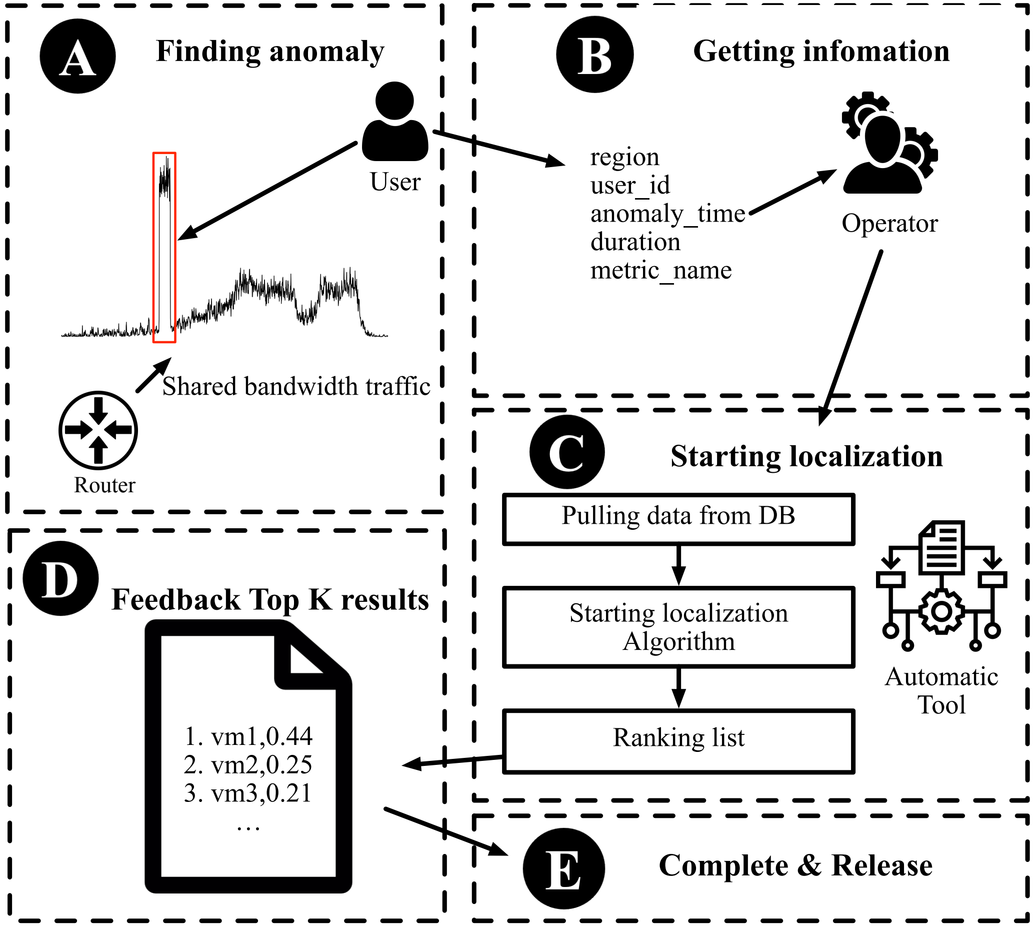
- **Problem Statement**
  - When users find anomalies in sBwp traffic, they want to locate the specific VM(s) in sBwp associated with the anomaly.
- **Problem Formulation:**
  - Traffic time series of sBwp: $y(t)$
  - Set of traffic time series of the VMs:
    $X = \{x_1(t), x_2(t), \dots, x_n(t)\}$
  - Since all $x(t)$ in X together produce the sBwp traffic:
    $$y(t) = x_1(t) + x_2(t) + \dots + x_n(t)$$
  - Anomaly time interval: (t_s, t_e)
  - The goal of the localization algorithm: find a subset of X that is the anomaly root cause of $y(t)$ in (t_s, t_e).



A specific anomaly example.

WORKING PROCESS

- **Finding Anomaly**
  - ➢ The user discovers an anomaly in the sBwp traffic time series.
- **Getting Information**
  - ➢ The user reports the anomaly information to the cloud network.
- **Starting Localization**
  - ➢ Pulling data from DB.
  - ➢ Starting localization algorithm.
  - ➢ Getting ranking list.
- **Feedback Top K Results**
  - ➢ The operator feeds top k back to the user.
- **Complete & Release**



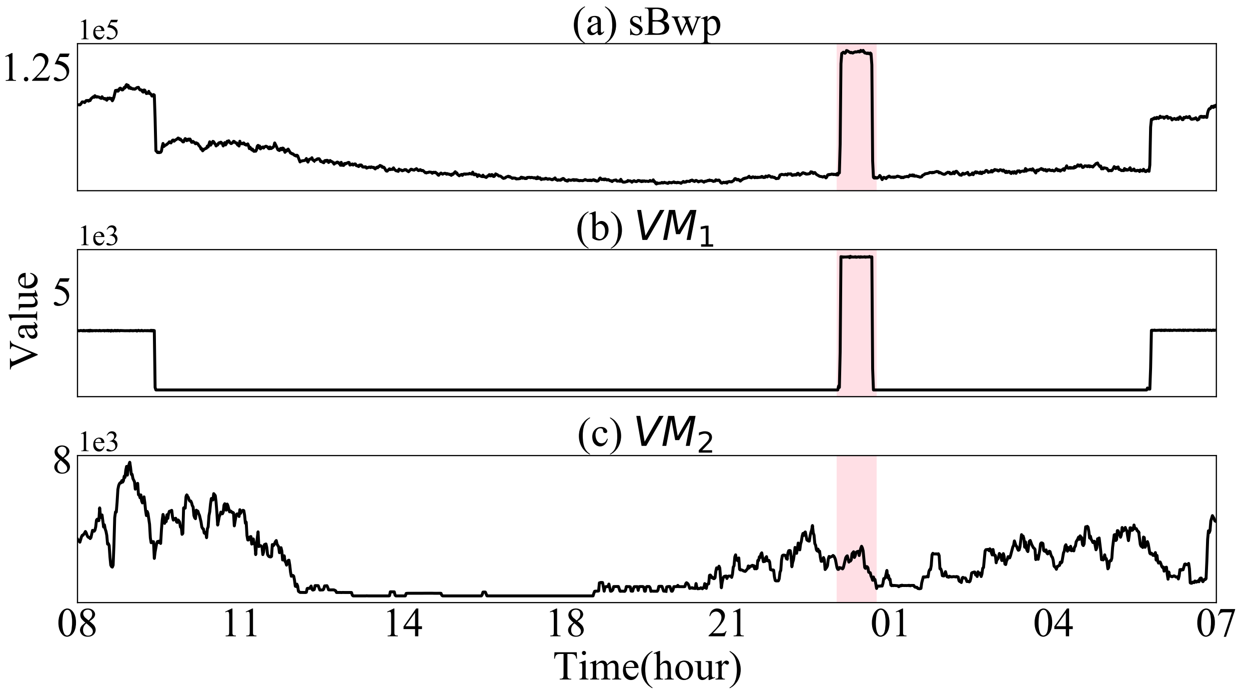
Working process of the anomaly localization.

CHALLENGES

- **Large scale.**
  - The number of users using shared bandwidth package services in the public cloud is vast, with virtual machines deployed worldwide.
- **Diversity of traffic anomalies.**
  - Due to the different service applications deployed by users in the cloud network, traffic anomalies exhibit diversity.
- **Low overhead and real-time.**
  - Since cloud service providers need to provide root cause localization services to a multitude of users, the localization algorithm is constrained in computational and storage resources.
- **Dynamicity.**
  - Users frequently change the deployment of virtual machines and network devices, benefitting from the powerful dynamic tuning capabilities of cloud networks.

MOTIVATIONS

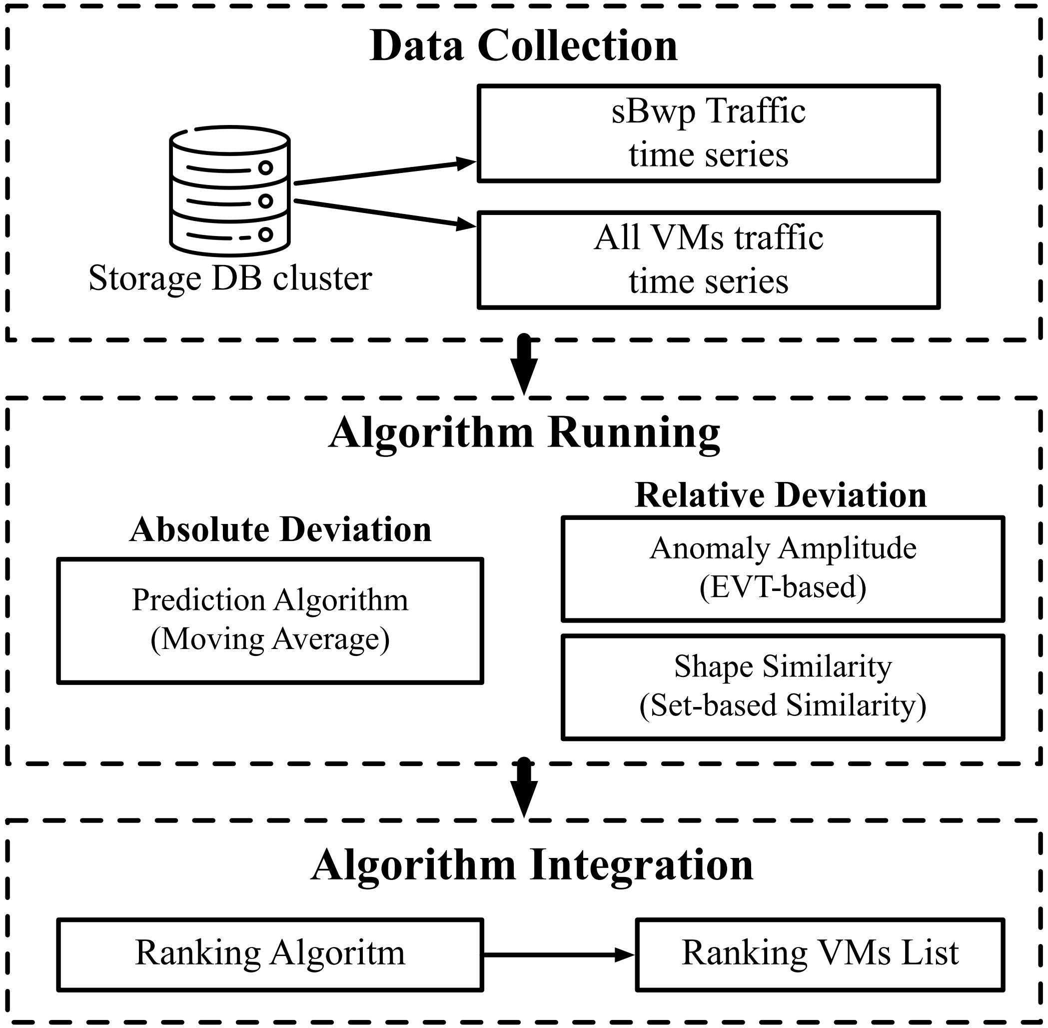
- **Intuitive Idea**:
  - Building a prediction model to calculate the deviation between the prediction time series and the real time series.
- **Limitations**:
  - **Deep learning-based methods**, high accuracy, but requiring offline training.
  - **Traditional statistic-based methods**, easy to implement, but low accuracy.
- **Solutions:**
  - Based on traditional statistic-based methods.
  - Adding new dimensions to improve accuracy.



An example of motivation.

SYSTEM DESIGN -- OVERVIEW

- **Data Collection**
  - Obtain the traffic data of the sBwp and all VMs using the sBwp service.
- **Algorithm Running**
  - Absolute Deviation
    - Prediction Algorithm
  - Relative Deviation
    - Anomaly Amplitude
    - Shape Similarity
- **Algorithm Integration**
  - Ranking Algorithm



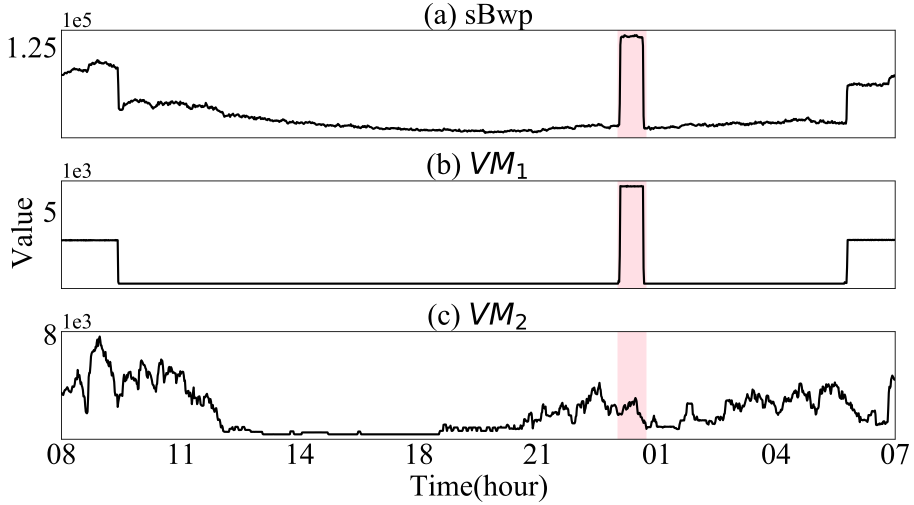
Structure of localization system.

SYSTEM DESIGN -- ABSOLUTE DEVIATION

- **Prediction Model Formulation:**
  - Traffic time series of sBwp: $y(t)$
  - Set of traffic time series of the VMs:

    $X = \{x_1(t), x_2(t), ..., x_n(t)\}$
  - Prediction model: H ; $H(y(t_c))$ represents the predicted value of $y(t)$ at time t_c .
  - Absolute deviation: $d_f(y(t_c)) = |y(t_c) - H(y(t_c))|$
  - Total absolute deviation of $y(t)$ in the (t_s, t_e) :

    $d_f^{sum}(y(t)) = d_f(y(t_s)) + d_f(y(t_s+1)) + ... + d_f(y(t_e))$
  - Similarly, the cumulative absolute deviation in (t_s, t_e) for each x in X: $d_f^{sum}(x_1(t)), d_f^{sum}(x_2(t)), \ldots, d_f^{sum}(x_n(t))$



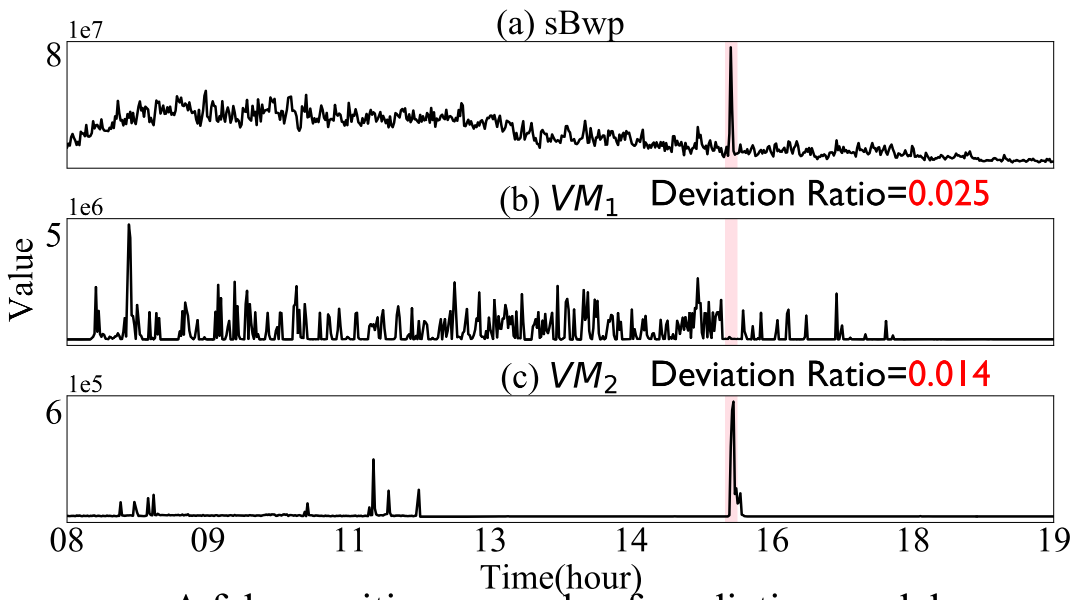
A specific anomaly example

- Ratio of each VM to the sbwp: $d_f^p(x(t)) = \frac{d_f^{sum}(x(t))}{d_f^{sum}(y(t))}$
- The higher the value of $d_f^p(x(t))$, the more likely it is the root cause.
- Select Moving Average algorithm as H.

SYSTEM DESIGN -- RELATIVE DEVIATION & RANKING ALGORITHM

- **Anomaly Amplitude**
  - To cover high-frequency jitter traffic time series
  - Considers the **anomaly significance** of each VM's traffic time series at the anomaly time compared with its own normal time.
  - The EVT algorithm based on extreme value theory
- **Shape Similarity**
  - Consider the **shape similarity** between VM and sBwp traffic sequence in the anomaly time interval.
  - Set-based similarity method
- **Ranking Algorithm**

$$In(x(t)) = d_f^p(x(t)) * (\omega_\alpha * \alpha^{max}(x(t)) + \omega_s * S(x(t)))$$



A false positive example of prediction model.

$d_f^p(x(t))$: Result of Prediction Model

$\alpha^{max}(x(t))$: Result of **Anomaly Amplitude**

$S(x(t))$: Result of **Shape Similarity**

ω_α, ω_s :Weight

EVALUATION

- **Dataset**
  - 3 data centers from Alibaba Cloud.
  - A total of 183 anomaly cases from 2020-11-30 to 2021-04-01.
- **Evaluation metric**
  - Precison@top k.
  - k=5, according the experience

DETAILS OF THE DATA.

Region	#cases	#machines	virtual machines distribution					
			> 10	> 50	> 100	> 500	> 1000	> 10000
A	77	24549	76	54	25	8	8	0
B	49	50815	47	41	23	11	10	1
C	57	9287	57	53	25	2	1	0

EVALUATION

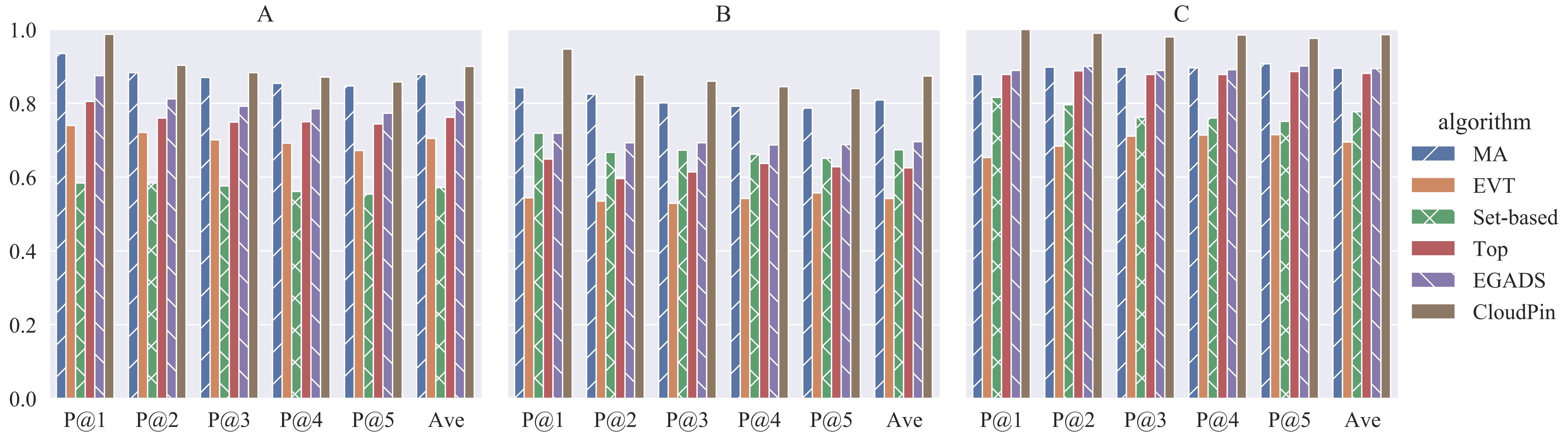
- Overall Evaluation
  - ***CloudPin*** outperforms multiple baseline models.
  - ***CloudPin*** can reach precison of 97.8%, when k=1.

OVERALL PERFORMANCE EVALUATIONS .

Type	Algorithm name	Precision@1	Precision@2	Precision@3	Precision@4	Precision@5	Average precision
Prediction Model	Linear Regression	0.587	0.570	0.579	0.579	0.589	0.581
	Olympic	0.847	0.847	0.839	0.837	0.833	0.841
	Polynomial	0.579	0.552	0.560	0.563	0.559	0.563
	Difference	0.885	0.869	0.852	0.839	0.834	0.856
	Moving Median	0.568	0.568	0.564	0.565	0.570	0.567
	Moving Average	0.891	0.869	0.856	0.846	0.844	0.861
	Simple Exponential Smoothing	0.770	0.779	0.779	0.776	0.774	0.776
Anomaly Amplitude	KDE	0.508	0.481	0.484	0.483	0.473	0.486
	EVT	0.656	0.653	0.650	0.651	0.648	0.652
Shape Similarity	Set-based	0.689	0.667	0.656	0.646	0.637	0.659
	Spearman	0.650	0.626	0.612	0.599	0.598	0.617
Integrated Model	Top	0.776	0.743	0.741	0.749	0.746	0.751
	EGADS	0.825	0.795	0.784	0.780	0.779	0.793
	CloudPin	**0.978**	**0.918**	**0.902**	**0.893**	**0.884**	**0.915**

EVALUATION

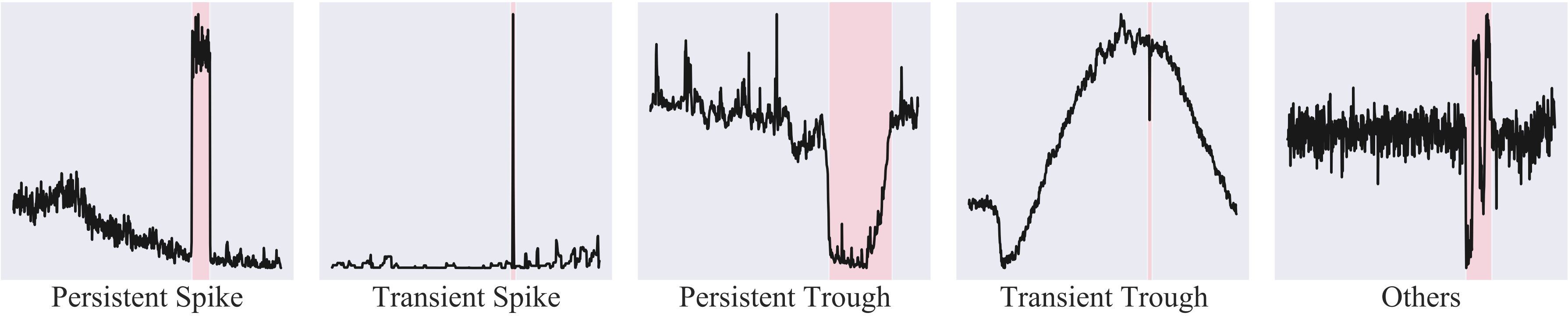
- Data Center Evaluation
  - ***CloudPin*** outperforms other baseline algorithms in different data centers.
  - The overall performance of different algorithms in different data centers is close.



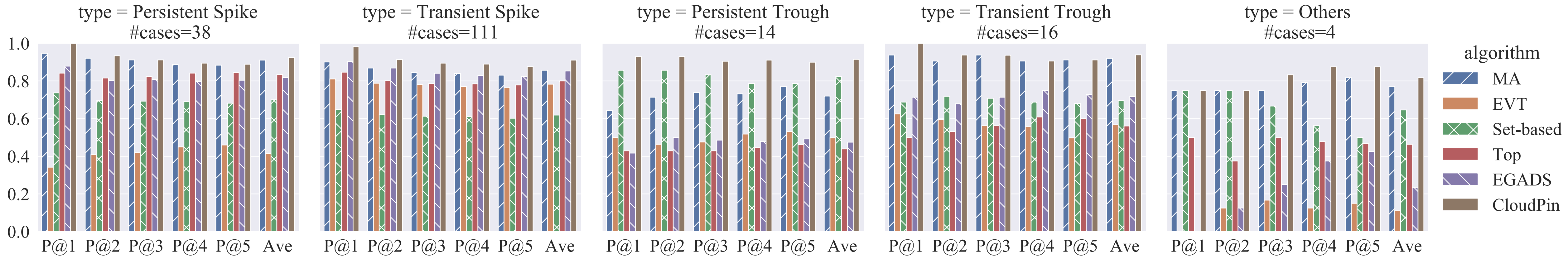
Performance of different regions.

EVALUATION

- Anomaly Type Evaluation
  - ***CloundPin*** can cover different anomaly type.



Examples of different anomaly classification.



Performance evaluations of different anomaly types.

EVALUATION

- **Overhead**
  - Storage
    - The average storage space of each traffic time series does not exceed 200KB
  - Speed
    - An ordinary server (CPU: 16 cores, Memory: 128GB DDR4).

CONSUMPTION TIME EVALUATION .

#VM	#cores	Algorithm name	Time	Speed
1994	1	Moving Average	223s	8.9 vm/s
		EVT	294s	6.8 vm/s
		Set-based	175s	11.4 vm/s
		EGADS	11263s	0.2 vm/s
		CloudPin	**695s**	**2.9 vm/s**
	16	**CloudPin**	**49s**	**40.7 vm/s**

CONCLUSION

- In this paper, we propose ***CloudPin***, a framework based on multi-dimensional analysis to locate root cause of sBwp traffic anomalies in public cloud networks.
- Our algorithm analyzes the three dimensions of **prediction deviation**, **anomaly amplitude**, and **shape similarity** and generates possible root lists through an integrated ranking algorithm.
- Our approach is **cost-effective** and can be **cold-started**.
- We conduct a comprehensive evaluation in a real large-scale cloud network.

THANKS FOR LISTENING

Q & A

For more information, please contact me.

zsz16@mails.tsinghua.edu.cn