基于奇异值分解更新的多元在线异常检测方法

钱叶魁[①②] 陈 鸣[①]
[①](解放军理工大学指挥自动化学院 南京 210007)
[②](解放军防空兵指挥学院 郑州 450052)

摘 要：网络异常检测对于保证网络稳定高效运行极为重要。基于主成分分析的全网络异常检测算法虽然具有很好的检测性能，但无法满足在线检测的要求。为了解决此问题，该文引入流量矩阵模型，提出了一种基于奇异值分解更新的多元在线异常检测算法MOADA-SVDU，该算法以增量的方式构建正常子空间和异常子空间，并实现网络流量异常的在线检测。理论分析表明与主成分分析算法相比，该算法具有更低的存储和计算开销。因特网实测的流量矩阵数据集以及模拟试验数据分析表明，该算法不仅实现了网络异常的在线检测，而且取得了很好的检测性能。

关键词：网络异常检测；在线算法；奇异值分解；多元分析；增量学习
中图分类号：TP393 **文献标识码**：A **文章编号**：1009-5896(2010)10-2404-06
DOI: 10.3724/SP.J.1146.2009.01342

A Multivariate Online Anomaly Detection Algorithm Based on SVD Updating

Qian Ye-kui[①②] Chen Ming[①]
[①](*Institute of Command Automation, PLA University of Science & Technology, Nanjing* 210007, *China*)
[②](*Air Defence Forces Command Academy of PLA, Zhengzhou* 450052, *China*)

Abstract: Network anomaly detection is critical to guarantee stabilized and effective network operation. Although PCA-based network-wide anomaly detection algorithm has good detection performance, it can not satisfy demands of online detection. In order to solve the problem, the traffic matrix model is introduced and a Multivariate Online Anomaly Detection Algorithm based on Singular Value Decomposition Updating named MOADA-SVDU is proposed. The algorithm constructs normal subspace and abnormal subspace incrementally and implements online detection of network traffic anomalies. Theoretic analysis shows that MOADA-SVDU has lower storage and less computing overhead compared with PCA. Analyses for traffic matrix datasets from Internet and simulation experiments show that MOADA-SVDU algorithm not only achieves online detection of network anomaly but also has very good detection performance.

Key words: Network anomaly detection; Online algorithm; Singular Value Decomposition (SVD); Multivariate analysis; Incremental learning

1 引言

所谓网络异常是指网络运行偏离正常状态的情况。导致网络异常的原因很多，包括网络设备误配置、网络故障、网络安全事件(分布式拒绝服务攻击、蠕虫传播等)以及不寻常的用户行为等。异常检测对于保证网络的正常运行具有重要意义，一直是网络研究的热点问题。

Lakhina 等人[1, 2]提出的基于 PCA 的异常检测算法奠定了多元异常检测的基础。Jin 等人[3]运用流量活动图分解的方法对骨干网流量进行了分析，揭示了各种网络异常行为；Torres 等人[4]运用流量聚类的分析方法推断 P2P 网络的异常行为。这些方法都是一种离线的分析方法，无法适用于在线分析和检测异常的需要。而现有的网络异常在线检测算法[5]大都仅仅关注单条链路的流量异常或单条端到端路径的性能异常。国内有关网络流量异常检测的文献很多，但是大多是针对某种特定异常的离线检测方法[6,7]。因此，网络管理员亟需一种既能够在线地检测网络异常，又能够取得很好的检测性能的算法。

本文正是以此作为研究目标，提出一种基于奇异值分解更新的多元在线异常检测算法(MOADA-SVDU)。主要创新点包括以下两个方面：(1)利用奇异值分解的更新算法，提出了一种全网络异常的在

2009-10-15 收到，2010-02-16 改回
国家自然科学基金重大研究计划(90304016)，国家 863 计划项目(2007AA01Z418)和江苏省自然科学基金(BK2009058)资助课题
通信作者：钱叶魁 qyk1129@hotmail.com

线检测算法 MOADA-SVDU，并对该算法的复杂度进行了分析；(2)通过分析因特网实测的流量矩阵数据集及模拟试验，证实了 MOADA-SVDU 算法的有效性。

2　MOADA-SVDU 算法

本节首先描述能够刻画网络流量完整视图的流量矩阵模型，然后利用奇异值分解更新的方法以增量的方式建立能够刻画网络流量变化规律的常态模型，由此提出一种增量的多元在线异常检测算法 MOADA-SVDU，最后对该算法的存储开销和时间复杂度进行理论分析。

2.1 流量矩阵模型

定义 1　流量矩阵　假设某自治系统(Autonomous System, AS)有 n 个边界路由器，以一定的时间间隔(周期)连续地被动测量任意一对边界路由器之间的流量，然后将这些测量值排列成一个 $p\times T$ 的矩阵 $\boldsymbol{X}$，它表示所有这些流量测量值的时间序列。其中，T 表示测量的周期数，p 表示每个周期内测量获得的流量测量值的个数，即 $p=n\times n$；第 i 列表示在第 i 个周期内流量测量值向量，通常用 $\boldsymbol{X}_i$ 表示，第 j 行表示第 j 个路由器对之间流量测量值的时间序列。矩阵 $\boldsymbol{X}$ 称为 AS 的路由级流量矩阵，简称为流量矩阵。根据选择的流量测度，可以定义基于不同测度的流量矩阵：分组数矩阵、流数矩阵、字节数矩阵等。

2.2 MOADA-SVDU 算法

流量异常检测的前提是建立网络流量的常态模型，然后根据这个常态模型来分析获得的测量样本，从而确定它们是正常的还是异常的。在离线异常检测中，通常以批处理的方式建立常态模型，从而确定在这批测量样本中哪些属于异常；而在在线异常检测中，通常要以增量的方式建立常态模型，每步只处理单个测量样本，判断该测量样本是否属于异常，并利用该测量样本更新常态模型。

许多规模较大的 AS(如 Abilene)通常具有十几个边界路由器。如果将 AS 所有边界路由器间的流量测量值看作一个输入向量 $\boldsymbol{X}_i$，则该向量 $\boldsymbol{X}_i$ 是存在于高维空间 $\mathbb{R}^p$ 中的一个多元变量，流量矩阵 $\boldsymbol{X}$ 可以看作高维空间 $\mathbb{R}^p$ 中多元变量的时间序列 $\{\boldsymbol{X}_i\}_{i=1}^T$，其中 T 表示测量的周期数。而 PCA 是处理类似流量矩阵高维数据的一种最有效的方法，它能够通过降低维度的方法实现最小均方意义下原始数据的重构。

PCA 是对归一化的输入向量 $\boldsymbol{X}_i\in\mathbb{R}^p, i=1,\cdots,T$ 构成的协方差矩阵 $\boldsymbol{C}$ 进行谱分解而实现：

$$\boldsymbol{C}=\boldsymbol{X}\boldsymbol{X}^{\mathrm{T}}=\frac{1}{T}\sum_{i=1}^{T}(\boldsymbol{X}_i-\mu)(\boldsymbol{X}_i-\mu)^{\mathrm{T}} \tag{1}$$

$$\boldsymbol{C}=\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^{\mathrm{T}} \tag{2}$$

其中 $\mu=\dfrac{1}{T}\sum_{i=1}^{T}\boldsymbol{X}_i$ 表示输入向量 $\boldsymbol{X}_i$ 的均值向量；$\boldsymbol{U}$ 表示特征向量 $\boldsymbol{u}_i$ 构成的矩阵，称为特征向量矩阵；$\boldsymbol{\Lambda}$ 表示对角线元素为特征值 λ_i 的矩阵，称为特征值矩阵。

奇异值分解(Singular Value Decomposition, SVD)是实现 PCA 的另一种重要方法。直接对归一化的输入向量 $\boldsymbol{X}_i\in\mathbb{R}^p, i=1,\cdots,T$ 进行 SVD 如下：

$$\boldsymbol{X}=\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}} \tag{3}$$

因此

$$\boldsymbol{X}\boldsymbol{X}^{\mathrm{T}}=\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}}\boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{U}^{\mathrm{T}}=\boldsymbol{U}\boldsymbol{\Sigma}^2\boldsymbol{U}^{\mathrm{T}} \tag{4}$$

由式(1)–式(4)可以看出：对 $\boldsymbol{X}$ 的协方差矩阵 $\boldsymbol{C}$ 进行谱分解得到的特征向量矩阵与直接对 $\boldsymbol{X}$ 进行 SVD 得到的特征向量矩阵完全相等；而对 $\boldsymbol{X}$ 的协方差矩阵 $\boldsymbol{C}$ 进行谱分解得到的特征值矩阵是直接对 $\boldsymbol{X}$ 进行 SVD 得到的奇异值矩阵 $\boldsymbol{\Sigma}$ 的平方，即 $\boldsymbol{\Lambda}=\boldsymbol{\Sigma}^2$。因此，可以直接对 $\boldsymbol{X}$ 进行 SVD，求取协方差矩阵 $\boldsymbol{C}$ 对应的特征值矩阵和特征向量矩阵，从而实现 PCA。

但是，SVD 同样属于批处理算法，它要求流量矩阵 $\boldsymbol{X}$ 必须提前给定。为了以增量的方式计算特征值矩阵和特征向量矩阵，本文引入一种 SVD 更新算法[8]。

假定某时刻流量矩阵 $\boldsymbol{A}\in\mathbb{R}^{p\times n}$, $\boldsymbol{A}_{p\times n}=\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}}$ 则 $\boldsymbol{A}_{p\times n}$ 近似为 $\widehat{\boldsymbol{A}}_{p\times n}=\boldsymbol{U}_k\boldsymbol{\Sigma}_k\boldsymbol{V}_k{}^{\mathrm{T}}$，其中 $\boldsymbol{U}_k$ 和 $\boldsymbol{V}_k$ 分别由 $\boldsymbol{U}$ 和 $\boldsymbol{V}$ 的前 k 列构成；$\boldsymbol{\Sigma}_k$ 由 $\boldsymbol{\Sigma}$ 的前 k 个特征值构成的子对角矩阵；$\widehat{\boldsymbol{A}}_{p\times n}$ 称作 $\boldsymbol{A}_{p\times n}$ 的秩-k 近似矩阵，记作 $\mathrm{best}_k(\boldsymbol{A})$。SVD 更新算法的目的是以一种有效的方式执行矩阵 $[\boldsymbol{A}_{p\times n},\boldsymbol{B}_{p\times r}]$ 的 SVD，其中 $\boldsymbol{B}_{p\times r}$ 是由新增加的 r 个列向量构成的 $p\times r$ 的矩阵。具体步骤如下：

$$(\boldsymbol{I}-\boldsymbol{U}_k\boldsymbol{U}_k^{\mathrm{T}})\boldsymbol{B}=\boldsymbol{Q}\boldsymbol{R} \tag{5}$$

$$\begin{bmatrix}\boldsymbol{\Sigma}_k & \boldsymbol{U}_k^{\mathrm{T}}\boldsymbol{B}\\ 0 & \boldsymbol{R}\end{bmatrix}=\widehat{\boldsymbol{U}}\widehat{\boldsymbol{\Sigma}}\widehat{\boldsymbol{V}}^{\mathrm{T}} \tag{6}$$

$$\mathrm{best}_k([\boldsymbol{A},\boldsymbol{B}])=([\boldsymbol{U}_k,\boldsymbol{Q}]\widehat{\boldsymbol{U}})\widehat{\boldsymbol{\Sigma}}\left(\begin{bmatrix}\boldsymbol{V}_k & 0\\ 0 & \boldsymbol{I}\end{bmatrix}\widehat{\boldsymbol{V}}\right)^{\mathrm{T}} \tag{7}$$

其中式(5)是对矩阵 $(\boldsymbol{I}-\boldsymbol{U}_k\boldsymbol{U}_k^{\mathrm{T}})\boldsymbol{B}$ 执行 QR 分解；式(6)是对矩阵 $[\boldsymbol{\Sigma}_k,\boldsymbol{U}_k^{\mathrm{T}}\boldsymbol{B};0,\boldsymbol{R}]$ 执行 SVD；式(7)是根据式(5)，式(6)计算得到的有关参数计算矩阵 $[\boldsymbol{A}_{p\times n},\boldsymbol{B}_{p\times r}]$ 的秩-k 近似矩阵。

特征向量的集合 $\{\boldsymbol{u}_i\}_{i=1}^p$ 相互垂直正交，且张成一个新的空间，称为特征空间，每个特征向量称为特征空间的主轴。流量矩阵 $\boldsymbol{X}$ 在特征空间中每个主轴上的投影称为流量矩阵的主成分。选择 k 个最大

的特征值对应的特征向量张成 k 维子空间 S，而选择剩余的 $p-k$ 个特征值对应的特征向量张成 $p-k$ 维子空间 $\tilde{S}$。由于流量矩阵 $\boldsymbol{X}$ 在 S 中的所有主成分均呈现正常的变化趋势，而异常行为均出现在 $\tilde{S}$ 的主成分中[1]，因此，把 S 称为正常子空间，而把 $\tilde{S}$ 称为异常子空间。

在构造正常子空间 S 和异常子空间 $\tilde{S}$ 以后，就可以将流量矩阵 $\boldsymbol{X}$ 向这两个子空间进行投影。如果把任意时刻的流量测量值向量(即流量矩阵 $\boldsymbol{X}$ 的某一列)记作 $\boldsymbol{x}$，则

$$\boldsymbol{x}=\hat{\boldsymbol{x}}+\tilde{\boldsymbol{x}} \tag{8}$$

其中 $\hat{\boldsymbol{x}}$ 表示 $\boldsymbol{x}$ 在正常子空间中的投影向量，被称为模型流量(modeled traffic)；$\tilde{\boldsymbol{x}}$ 表示 $\boldsymbol{x}$ 在异常子空间中的投影向量，被称为残余流量(residual traffic)。把对应正常子空间的主轴的集合 $\{\boldsymbol{u}_i\}_{i=1}^{k}$ 按序排列成 $p\times r$ 的矩阵 $\boldsymbol{U}_k$，则

$$\hat{\boldsymbol{x}}=\boldsymbol{U}_k\boldsymbol{U}_k^{\mathrm{T}}\boldsymbol{x} \tag{9}$$

$$\tilde{\boldsymbol{x}}=(\boldsymbol{I}-\boldsymbol{U}_k\boldsymbol{U}_k^{\mathrm{T}})\boldsymbol{x} \tag{10}$$

若 $\|\tilde{\boldsymbol{x}}\|_2^2$ 超过某一阈值则认为出现异常，通常阈值采用 Q 统计量[1]。

本文在 4.1 节将通过实测数据分析表明 MOADA-SVDU 算法得到的残余向量与 PCA 算法得到的残余向量基本相同，而且对于同样的流量测度，相邻时期的流量矩阵对应的 Q 统计量阈值非常接近。因此，可以将前一阶段的流量矩阵作为训练集，并在该训练集上执行 PCA 算法以获得 Q 统计量阈值，然后以此作为下一阶段 MOADA-SVDU 算法的异常检测阈值。

基于以上基本原理，本文提出一种基于奇异值分解更新的多元在线异常检测算法 MOADA-SVDU，算法流程见表 1。

表 1 MOADA-SVDU 算法

输入：流量矩阵 $\boldsymbol{X}$ 和参数 k
输出：$\|\tilde{\boldsymbol{x}}\|_2^2$ 和警报信息
用部分输入向量初始化 $\boldsymbol{X}_1=\boldsymbol{X}(:,1:n)$；
对 $\boldsymbol{X}_1$ 执行中心标准化；
对 $\boldsymbol{X}_1$ 执行 SVD: $\boldsymbol{X}_1=\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}}$；
求取 $\mathrm{best}_k(\boldsymbol{X}_1)=\boldsymbol{U}_k\boldsymbol{\Sigma}_k\boldsymbol{V}_k^{\mathrm{T}}$；
for $t=n+1,\cdots$ do
  获取新增的测量样本: $\boldsymbol{x}=\boldsymbol{X}(:,t)$ ；
  对 $\boldsymbol{x}$ 执行中心标准化；
  按照，式(5)-式(7)计算矩阵 $[\boldsymbol{X}_1,\boldsymbol{x}]$ 的特征值矩阵和特征向量矩阵；
  更新 $\boldsymbol{U}_k,\boldsymbol{\Sigma}_k,\boldsymbol{V}_k$；
  按照式(10)计算残余流量 $\tilde{\boldsymbol{x}}$；
  if $\|\tilde{\boldsymbol{x}}\|_2^2\geq$ Q 统计量
    发出红色警报；
  $\boldsymbol{X}_1=[\boldsymbol{X}_1,\boldsymbol{x}]$；
end for

对于在线异常检测算法来说，存储开销和时间复杂度是至关重要的两个指标。MOADA-SVDU 算法可以分为两个阶段：在初始化阶段，需要存储 $\boldsymbol{X}_1$；在增量检测阶段，需要存储 $\boldsymbol{U}_k$、$\boldsymbol{\Sigma}_k$、$\boldsymbol{V}_k$ 以及 $\boldsymbol{x}$。因此，MOADA-SVDU 算法总的存储开销为 $p\times n+p\times k+k\times k+n\times k+p\times 1$，而 PCA 算法的存储开销为 $p\times T$ 的流量矩阵。由于 $n\ll T$,$k\ll p$，所以 MOADA-SVDU 算法的存储开销远小于 PCA。

MOADA-SVDU 算法在增量检测阶段的计算瓶颈是式(5)的 QR 分解和式(6)的 SVD，其时间复杂度分别为 $O(p)$ 和 $O(p\times k)$；而 PCA 算法的计算瓶颈是对整个流量矩阵执行谱分解，其时间复杂度为 $O(T\times p^2)$。显然，MOADA-SVDU 算法的时间复杂度远低于 PCA。

3 算法评价

为评价 MOADA-SVDU 算法的检测性能，本文采用了两种方法：对因特网实测数据的分析以及对模拟试验数据的分析，并在同等条件下与 PCA 算法进行比较。

3.1 因特网实测数据的分析

(1)数据集　利用从 Abilene 实测得到的流量矩阵数据集[1,2,9]来评价 MOADA-SVDU 算法的检测性能，数据集的具体描述见表 2。

(2)检测结果　选择表 2 中 4 种不同流量测度的数据集：Dataset1，Dataset2，Dataset4 和 Dataset5，分别画出输入向量 2-范数的平方、PCA 算法计算获得的残余向量 2-范数的平方以及 MOADA-SVDU 算法计算获得的残余向量 2-范数的平方，如图 1(a)-1(d)所示。可以看出，MOADA-SVDU 算法不仅实现了网络流量异常的在线检测，而且在大部分情况下都具有与 PCA 算法类似的检测性能。

异常检测阈值是影响 MOADA-SVDU 算法检测性能的重要参数，为了验证以 Q 统计量作为阈值的可行性，计算并画出表 2 中数据集 Dataset1~

表 2 Abilene 流量矩阵

持续时间	间隔时间 (min)	测度	矩阵形式	数据集
2003.12.15-12.21	5	分组数	121×2010	Dataset1
2003.12.15-12.21	5	流数	121×2010	Dataset2
2003.12.15-12.21	5	字节数	121×2010	Dataset3
2004.03.01-03.07	5	字节数	144×2016	Dataset4
2004.03.08-03.14	5	字节数	144×2016	Dataset5
2004.03.15-03.21	5	字节数	144×2016	Dataset6

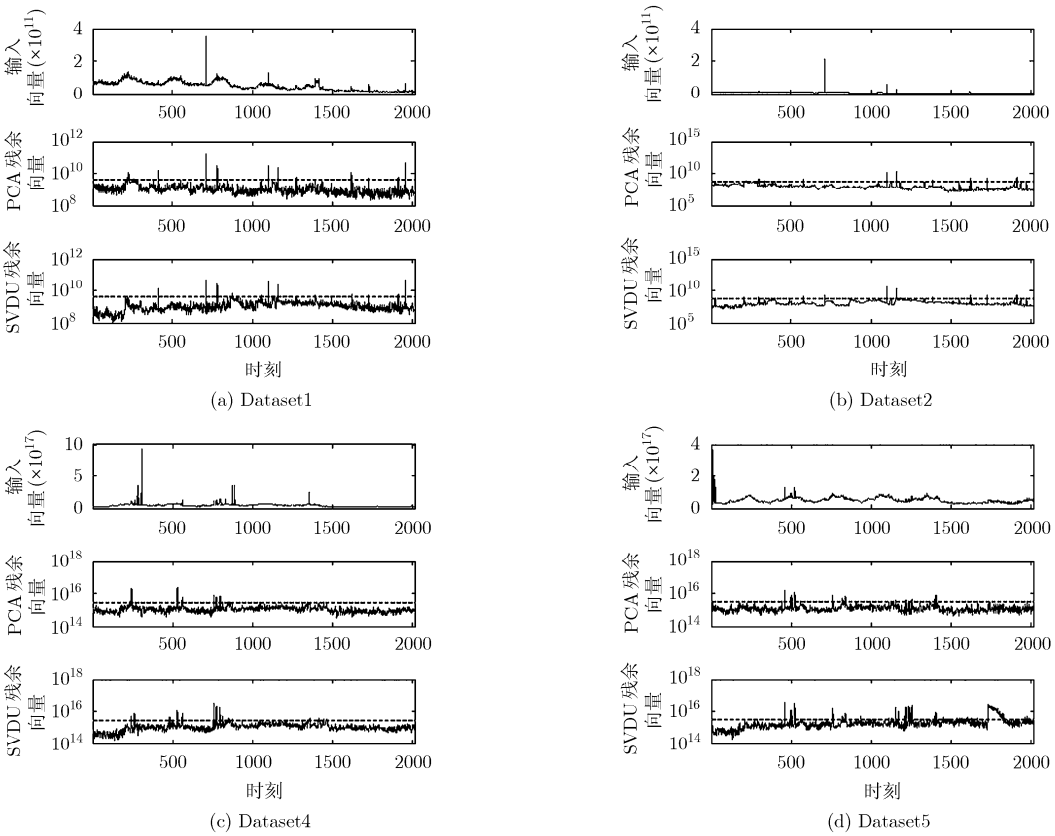


图 1 PCA 和 MOADA-SVDU 算法对实测数据的检测结果

Dataset6 对应的 Q 统计量,其中 Dataset1~Dataset3 对应的 Q 统计量如图 2(a)所示,Dataset4~Dataset6 对应的 Q 统计量如图 2(b)所示。可以看出，虽然不同测度的流量矩阵对应的 Q 统计量差异很大，但相邻时期的流量矩阵对应的 Q 统计量非常接近，这验证了 3.2 节中阈值选取方法的可行性。

3.2 模拟试验数据的分析

(1)模拟方法　考虑到网络流量通常由 3 种成分的流量构成[10]：近似周期性的正常成分、高斯噪声成分和异常成分。因此，我们产生这 3 种成分来并按照适当比例人工合成流量矩阵中每条 OD 流。具体步骤如下：

第 1 步　利用 3 种不同周期的正弦波叠加来模拟正常的 OD 流，并构造基准流量矩阵。

第 2 步　在第 1 步产生的基准流量矩阵的每条 OD 流上加入零均值的高斯噪声，获得不含异常的基准流量矩阵。

第 3 步　在第2步产生的含噪音的基准流量矩阵中以一定的规则加入各种典型异常，异常的合成步骤如图3所示。

由于本文重点关注流量大小异常的检测，所以本文模拟4种最常见的流量异常：阿尔法(alpha)异常、(分布式)拒绝服务攻击(DoS, DDoS)、闪拥(flash crowd)、入口/出口移动(ingress/egress shift)异常。

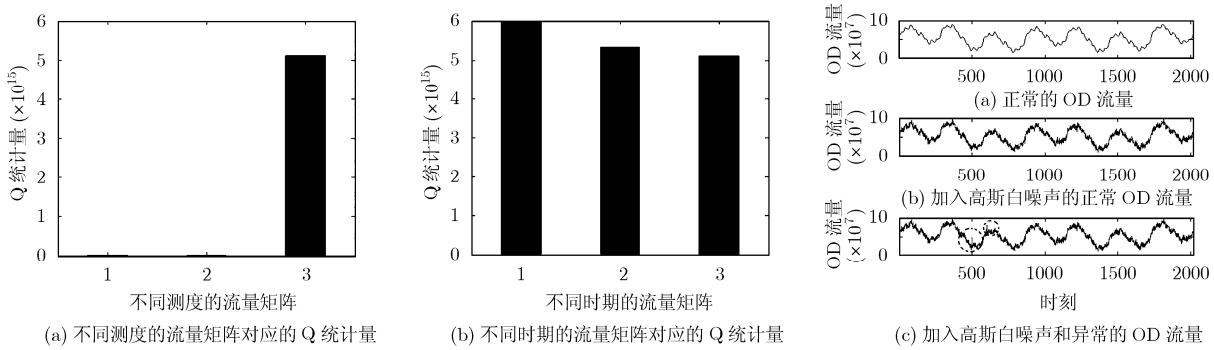


图 2 不同的流量矩阵对应的 Q 统计量

图 3 异常的合成步骤

这4种异常的具体特征见表3。

表3 异常类型及其特征

异常类型	特征
阿尔法	点到点之间不寻常的高速字节传输
(分布式)拒绝服务攻击	单源或多源对单个目的地的洪泛攻击
闪拥	大量客户同时访问某一Web站点
入口/出口移动	BGP策略变化引起流量出口点的变化

可以用 4 个参数来描述这 4 种网络流量异常[11]：持续时间、流量变化大小、源-目的数以及形状函数。当网络异常出现时，可以用两种方式模拟流量大小的变化：一是通过为基准流量矩阵中部分 OD 流乘上一个乘法因子，二是通过为基准流量矩阵中部分 OD 流加上一个常数项。源-目的数是指异常所涉及的 OD 流的数目，记号(1,1)表示异常涉及单个源和单个目的地，这可能是由于拒绝服务攻击或阿尔法事件，(N,1)表示异常涉及 N 个源点和 1 个目的地，这可能是由于出现了分布式拒绝服务攻击或闪拥，(2,2)表示异常涉及 2 个源点和 2 个目的地，这可能是由于入口/出口移动事件引起的。形状函数是用来模拟各种异常的变化行为，这些行为可以用不同的形状函数及其组合来表征。以上 4 个参数的可能取值见表 4。

(2)检测结果　采用上述方法模拟 4 种不同的网络异常，来评价算法的检测性能。当模拟阿尔法事件或拒绝服务攻击时，从第 1000 时刻开始产生异常，持续时间为 4 个周期，将某条 OD 流的流量大小从增加 20%迅速增加至 80%。当模拟分布式拒绝服务攻击或闪拥事件时，从第 1500 时刻开始产生异常，持续时间为 6 个周期，将 5 条 OD 流的流量大小从增加 10%迅速增加至 50%，然后又逐渐降低至 10%。当模拟入口/出口移动事件时，从第 1700 时刻开始产生异常，持续时间为 40 个周期，将某条 OD 流的流量大小减少 50%，即为某条 OD 流的流量大小乘以 0.5，且将减少的这部分流量加到另一条 OD 流上。检测结果如图 4(a)所示，MOADA-SVDU 算法能够非常准确地检测出这 3 种人造异常，与 PCA 算法的检测结果几乎完全相同。KRLS 算法[9]的检测结果如图 4(b)所示，显然无法通过设置某一阈值来检测异常，其中 δ_t 表示投影误差。因此，MOADA-SVDU 算法的检测性能明显优于KRLS 算法。

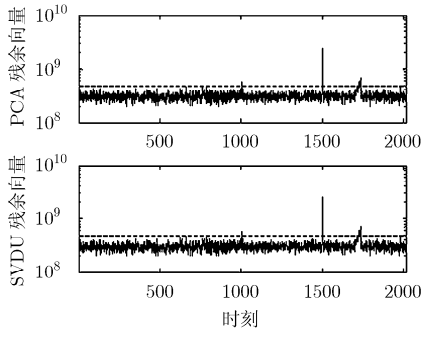
表 4 异常参数及其取值

参数	持续时间	流量变化	源-目的数	形状函数
可能的取值	分钟 小时 天	常数项 乘法因子	(1,1) (N,1) (2,2)	斜坡 指数 台阶

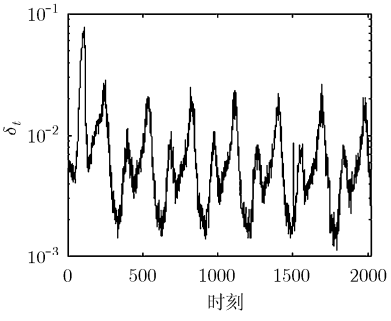
下面进一步研究 MOADA-SVDU 算法的敏感性。从第 1000 时刻开始产生异常，持续时间为 10 个周期，将某条 OD 流的流量大小从 5%逐渐增加至 50%，检测结果如图 5(a)所示，在前 4 个周期，由于异常流量较小，所以无法被检测到，而从第 1004–1009 时刻成功检测到异常，由此可见，异常流量越大，MOADA-SVDU 算法的检测性能越好。在第 1500 时刻，将某条 OD 流的流量大小增加 20%，持续时间为 5 个周期，在第 1550 时刻，将某 10 条 OD 流的流量大小同时增加 20%，持续时间也为 5 个周期，检测结果如图 5(b)所示，异常分布范围越广，算法的检测性能越好，由此可见 MOADA-SVDU 算法尤其适合检测那些局部影响不大但影响范围较广的异常，如 DDoS 攻击。

4 结束语

本文提出了一种基于奇异值分解更新的多元在线异常检测算法 MOADA-SVDU，Abilene 实测的流量矩阵数据和模拟试验数据分析表明该算法不仅

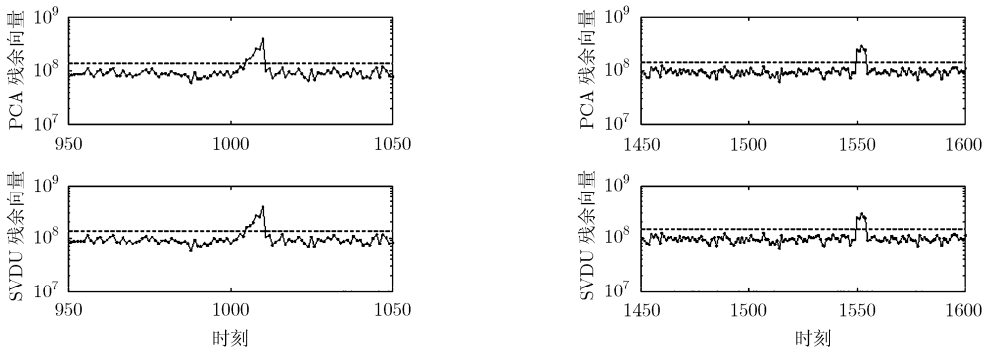


(a) PCA 和 MOADA-SVDU 的检测结果



(b) KRLS 的检测结果

图 4 PCA, MOADA-SVDU 和 KRLS 算法对模拟试验数据的检测结果



(a) 异常流量大小对检测结果的影响　　(b) 异常分布范围对检测结果的影响

图 5 MOADA-SVDU 算法的敏感性分析

实现了网络流量的在线检测，而且取得了与批处理算法 PCA 类似的检测性能，完全能够满足网络管理员在线有效地检测网络异常的需要。近期的研究表明 PCA 异常检测器自身也会遭受方差注入攻击[12]，且流量矩阵可能出现丢失元素值的情况[13]，因此下一步我们将研究更为鲁棒的 PCA 异常检测方法。

参 考 文 献

[1] Lakhina A, Crovella M, and Diot C. Diagnosing network-wide traffic anomalies[C]. SIGCOMM, Portland, Oregon, USA, 2004: 224–235.

[2] Lakhina A, Crovella M, and Diot C. Mining anomalies using traffic feature distributions[C]. SIGCOMM, Philadelphia, Pennsylvania, USA, 2005: 164–175.

[3] Jin Y, Sharafuddin E, and Zhang Z L. Unveiling core network-wide communication patterns through application traffic activity graph decomposition[C]. SIGMETRICS, Seattle, WA, USA, 2009: 86–91.

[4] Torres R, Hajjat M, and Rao S G, *et al*.. Inferring undesirable behavior from P2P traffic analysis[C]. SIGMETRICS, Seattle, WA, USA, 2009: 156–167.

[5] Logg C, Cottrell L, and Navratil J. Experiences in traceroute and available bandwidth change analysis[C]. SIGCOMM Workshop, Portland, Oregon, USA, 2004: 81–90.

[6] 吴志军, 张东. 低速率 DDoS 攻击的仿真和特征提取[J]. 通信学报, 2008, 29(1): 71–76.
Wu Z J and Zhang D. Attack simulation and signature extraction of low-rate DDoS[J]. *Journal on Communications*, 2008, 29(1): 71–76.

[7] 谢逸, 余顺争. 基于 Web 用户浏览行为的统计异常检测[J]. 软件学报, 2007, 18(4): 967–977.
Xie Y and Yu S Z. Anomaly detection based on web users' browsing behaviors[J]. *Journal of Software*, 2007, 18(4): 967–977.

[8] Zhao H, Yuen P C, and Kwok J T. A novel incremental principal component analysis and its application for face recognition[J]. *IEEE Transactions on Systems, Man, and Cybernetics-Part B*: *Cybernetics*, 2006, 36(3): 873–886.

[9] Ahmed T, Coates M, and Lakhina A. Multivariate online anomaly detection using kernel recursive least squares[C]. INFOCOM, Los Angeles, USA, 2007: 387–396.

[10] Lakhina A, Papagiannaki K, and Crovella M, *et al*.. Structural analysis of network traffic flows[C]. SIGMETRICS, New York, NY, USA, 2004: 156–167.

[11] Soule A, Salamatian K, and Taft N. Combining filtering and statistical methods for anomaly detection[C]. IMC, Boston, USA, 2005: 168–179.

[12] Rubinstein B, Nelson B, and Huang L. Stealthy poisoning attacks on PCA-based anomaly detectors[C]. SIGMETRICS, Seattle, WA, USA, 2009: 168–179.

[13] Zhang Y, Roughan M, and Willinger W, *et al*.. Spatio-temporal compressive sensing and Internet traffic matrices[C]. SIGCOMM, Barcelona, Spain, 2009: 110–121.

钱叶魁：　男，1980 年生，博士生，讲师，研究领域为网络测量、网络安全.

陈　鸣：　男，1956 年生，博士，教授，博士生导师，研究方向为网络测量、网络体系结构、网络管理等.