

## 基于余弦函数局部特征的时间衰变模式

樊海宽<sup>1,2,3</sup>, 刘奇志<sup>1,2\*</sup>

(1. 南京大学 软件新技术国家重点实验室, 南京 210093; 2. 南京大学 计算机科学与技术系, 南京 210093;

3. 国防科学技术大学 计算机学院, 长沙 410073)

(\* 通信作者电子邮箱 lqz@nju.edu.cn)

**摘要:** 数据流具有无限增长的特征, 目前的计算系统无法在线处理整个数据集, 只能在有限空间内对部分数据进行处理。为了能够得到尽可能合理的结果, 数据流系统常常采用单调递减函数由数据的时间戳来确定数据的权值, 根据权值选择数据。广泛使用的单调函数是指数函数和多项式函数, 但它们存在衰变速度太快或太慢等问题。提出一种新的时间衰变模式——使用余弦函数的局部衰变速度介于指数和多项式之间的特征来确定数据的权值。实验结果显示相对于指数和多项式衰变, 局部余弦衰变具有衰变速度合理、参数易于确定、适用于乱序数据流等优势。

**关键词:** 数据流; 时间衰变模式; 余弦函数; 乱序数据流; 前向衰变模型

**中图分类号:** TP311.13 **文献标志码:** A

### Time decay mode based on degressive cosine ramp

FAN Hai-kuan<sup>1,2,3</sup>, LIU Qi-zhi<sup>1,2\*</sup>

(1. State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing Jiangsu 210093, China;

2. Department of Computer Science, Nanjing University, Nanjing Jiangsu 210093, China;

3. College of Computer, National University of Defense Technology, Changsha Hunan 410073, China)

**Abstract:** Unlimited growth is one of the main characteristics of data stream. Current computing systems can only process a portion of data instead of the full data set online because of the limited memory and space. In order to obtain reasonable results, decay functions are often used in data stream systems to map the weights of data from timestamps. Monotonic decay functions such as exponential and polynomial functions are widely used, but they decay too fast or too slowly. In this paper, a new decay mode based on cosine function whose decay speed is between exponential function and polynomial function was proposed. The experimental results show that compared to exponential and polynomial decay modes, the degressive cosine ramp decays more reasonably and it is easy to appoint the parameter but also applicable to out-of-order data stream.

**Key words:** data stream; time decay model; cosine function; out-of-order data stream; forward decay model

## 0 引言

在处理带有时间信息的数据流时, 常常根据数据项的生存年龄赋予数据相应的权值, 并进行动态刷新, 以表征数据在不同时刻的重要程度不同, 数据流中同一数据的权值在时间轴上呈衰减趋势<sup>[1-4]</sup>。这种处理方式在直观上反映最近的数据和查询结果最为相关, 而老数据的重要性则相对较低, 甚至可以被完全忽略。

研究人员通常用指数函数或多项式函数作为时间衰变函数, 将数据项的生存年龄作为衰变函数的自变量以计算数据项的权值。这种时间衰变概念在数据仓库、传感器网络和其他分布式监测系统都被广泛采纳。实际应用时通过对具体应用系统的定性分析可以选择合适的衰变函数, 通过实验方法可以确定其中的参数。但是许多数据流实际应用系统中, 最新一组数据一般都很重要, 其重要程度往往难分伯仲, 而“老数据”一般不重要, 并且区分它们的不重要程度意义不大。这就好比在搜索引擎的返回结果中, 排在前面的权重要比排在后面的权重大, 而排在首页的结果之间的权重差别要远小于首页与后面其他页面结果之间的权重差别, 至于以后页面的结果, 没有必要严格区分它们的权重。也就是说, 权重衰变

速度通常呈现“慢—快—慢”的趋势, 然而已有常用衰变函数难以表征上述特征<sup>[5]</sup>。比如多项式衰变速度的趋势比较慢, 指数衰变速度的趋势是“快—慢”。

本文基于余弦函数的局部特征提出利用局部余弦函数的时间衰变模式, 其衰变速度合理, 衰变趋势与数据流应用特征相吻合, 此外还具有参数易于确定、更适合处理乱序数据流, 以及在前向衰变模型<sup>[6]</sup>下函数形式不变、不降低小权值数据精度等优势, 可以提高数据流系统对数据质量<sup>[7]</sup>的控制能力。

## 1 相关工作

### 1.1 问题的定义

**定义 1** 数据流  $S = \langle \dots, s_i(t_i, v_i), \dots, s_j(t_j, v_j), \dots \rangle$ , 其中:  $v_i(v_j)$  为第  $i(j)$  个数据项(元组)的属性值, 且  $v_i, v_j \in \mathbf{R}$ ;  $t_i, t_j$  为时间戳。

**定义 2** 时间衰变函数以数据生存年龄为自变量返回数据项的权值, 记作  $w(i, t_i)$  或  $w(a)$ ,  $a$  为数据项的年龄。时间衰变函数满足如下性质:

1) 当  $t_i = t, w(i, t_i) = 1$ ; 当  $t_i \leq t, 0 \leq w(i, t_i) \leq 1$  ( $t$  为当前时刻)。或当  $a = 0, w(a) = 1$ ; 当  $a > 0, 0 \leq w(a) < 1$ 。

收稿日期: 2011-06-17; 修回日期: 2011-07-28。

**作者简介:** 樊海宽(1988-), 男, 江苏徐州人, 硕士研究生, 主要研究方向: 无线网络; 刘奇志(1971-), 女, 安徽芜湖人, 副教授, 博士, CCF 会员, 主要研究方向: 高性能数据管理。

网络出版时间 2011-10-13 16:48。网络出版地址 <http://www.cnki.net/kcms/detail/51.1307.TP.20111013.1648.001.html>。

2) 若  $t - t_i \geq t - t_j$ , 则  $w(i, t_i) \leq w(i, t_j)$ ; 或若  $a > a'$ , 则  $w(a) \leq w(a')$ , 即单调非递增。

### 1.2 常用权值函数

在数据流应用中, 用来计算权值的函数通常包括以下四类。

1) 常函数:  $w(a) = 1$ 。即任意年龄数据项的权值相同。

2) 阶跃函数: 给定一个窗口<sup>[8-9]</sup>宽度  $W$ , 当  $a < W$  时,  $w(a) = 1$ ;  $a \geq W$  时,  $w(a) = 0$ 。用阶跃函数作时间衰变函数意味着只有年龄小于  $W$  的数据项才被考虑。

3) 指数函数:  $w(a) = \exp(-\lambda a)$  ( $\lambda > 0$ ), 当  $a = 0$  时,  $w(a) = 1$ 。数据项的权值随年龄的增长急剧下降。

4) 多项式函数:  $w(a) = (a + 1)^{-\alpha}$  ( $\alpha > 0$ ), 当  $a = 0$  时,  $w(a) = 1$ 。数据项的权值随年龄的增长逐渐下降。

在实际数据流应用中, 人们对不同时期产生的数据关注程度不同, 所以一般不用常函数计算权值; 通常对产生时间久远的数据关注程度较弱, 不是一点都不关注, 所以阶跃函数也不常用; 对近期产生的数据往往更为关注, “近期”指的是一段时期, 而不是某个时刻, 所以指数函数所描述的“近期”时间段过短, 即其开始衰变速度过快; 而多项式函数衰变速度又过慢。可以采用指数与多项式结合的形式来获得一些新的衰变函数, 如超指数函数  $w(a) = \exp(-\lambda a^2)$ , 次多项式函数  $w(a) = (1 + \ln(1 + a))^2$  等, 但其中参数的确定比较困难。

### 1.3 前向衰变模型

动态刷新权值时, 一般根据数据的生存年龄(当前时刻与数据的时间戳的差值)来计算数据的权值, 即在时间坐标轴上从当前时刻向后(左)度量生存年龄, 生存年龄越大, 权值越小, 此谓后向衰变模型; 也可以把数据的时间戳与界标时间戳的差值作为权值计算函数的自变量, 该差值越大, 权值越大, 这是前向衰变模型<sup>[5]</sup>。

## 2 局部余弦衰变函数

本文根据余弦函数局部特征设计时间衰变函数。

### 2.1 局部余弦衰变函数一般特征

局部余弦衰变函数  $w(a) = \cos^2(\lambda a)$  ( $\lambda > 0$ ), 如图1所示, 其衰变速度呈“慢—快—慢”趋势。观察余弦函数曲线, 容易发现在递减的第一个半周期内, 变化趋势是先慢后快再慢, 这与数据流应用中人们对数据的关注程度趋势吻合。距离当前时刻较近的数据都比较重要, 所以权值衰变速度不应太快, 而距离当前时刻较远时, 无需严格区分各自不重要的程度。

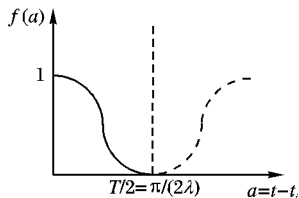


图1 局部余弦衰变函数

衰变函数的参数选择对于计算出的权值分布均匀程度至关重要。若所有数据项的权值在  $[0, 1]$  区间均匀分布, 则说明利用该衰变函数可以有效区分不同数据的重要程度; 若权值集中在偏0的一边, 则说明该函数低估了部分新数据的重要性; 若权值集中在偏1的一边, 则说明该函数高估了部分老数据的重要性。对于指数函数和多项式函数而言, 参数只能采用穷举法通过测试来确定, 直到选择到最优参数为止。而对于局部余弦函数, 只需要通过对当前时刻的简单计算就可以避免

大规模的穷举测试。

在后向衰变模型下, 由于局部余弦衰变只使用函数  $w(a) = \cos^2(\lambda a)$  的第一个半周期, 即  $0 < \lambda < \pi/(2a)$ , 所以根据最大生存年龄  $a$  算出  $\pi/(2a)$ , 只要  $\lambda$  略小于该值就可以达到最优效果。

### 2.2 乱序数据流中局部余弦衰变的特征

在实际应用系统中, 数据流并不会精确地按照时间戳的先后顺序得到处理, 数据延迟到达或合并多股数据流时都有可能乱序数据流<sup>[10-11]</sup>。上述计算权值过程中, 默认当前的查询时刻  $t$  比目前所有数据项的  $t_i$  大, 而在乱序数据流中, 这一前提条件有时并不成立, 也就是说, 存在数据项的时间戳  $t_i$  大于  $t$  的情况, 此时同样应该遵循如下规则: 距离  $t$  的时间差越小, 其重要性越大, 权值也越大。然而通过指数函数或多项式函数计算的权值会违背这一规则。比如, 设查询时刻  $t = 200$ , 待查询数据项的时间戳  $t_i = 220$ , 在后向衰变模型下, 数据生存年龄  $a = t - t_i = -20$ , 经指数函数计算的权值  $w(a) > 1$ , 经多项式函数计算的权值  $w(a) < 0$ , 不符合时间衰变函数的定义。而余弦函数关于  $x = 0$  对称, 所以  $w(a)$  仍然在  $[0, 1]$  区间, 且符合“距离当前时刻  $t$  越近的数据权值越大, 越远的数据权值越小”这一规则。

### 2.3 前向衰变模型下局部余弦衰变的特征

前向衰变模型基于一个小于所有时间戳的固定标志时刻沿着时间坐标轴向前(右)看, 以衡量数据的新旧程度。与后向衰变模型相比, 前向衰变模型具有计算量小等优点<sup>[5]</sup>。

定义3 给定一个正的单调非递减函数  $g$ , 一个标志时刻  $L$ , 到达时刻为  $t_i$  ( $> L$ ) 的数据项在时刻  $t$  的权值为:  $w(i, t) = g(t_i - L)/g(t - L)$ 。

由定义3可知, 当  $t = t_i$  时权值为1。随着时刻  $t$  的增长, 由于函数  $g$  的单调非递减性质, 权值永远不会增长, 而是一直保持在  $[0, 1]$  区间。一般情况下, 把标志时刻  $L$  设为当前查询数据流中的最小时间戳。

在前向衰变模型下, 常函数、阶跃函数的形式与后向衰变模型下相同。指数函数需变形为  $g(a) = \exp(\lambda a)$  ( $\lambda > 0$ )。多项式函数需变形为  $g(a) = (a + 1)^\alpha$  ( $\alpha > 0$ )。

局部余弦函数形式与后向衰变模型下相同:  $w(a) = \cos^2(\lambda a)$  ( $\lambda > 0$ ), 不过需要将标志时刻  $L$  固定为一个虚拟时刻  $-nT/2$  ( $n$  为奇数, 一般设为1) 处, 以保证数据项时间戳和查询时刻的时间差落在局部余弦函数的递增部分。

此外, 在前向衰变模型下, 局部余弦函数在权值计算精度上也更有优势。假设计算一组较老数据的权值(并且在这部分老数据到达之后一直没有新数据到达), 即  $t \gg t_i$ , 则根据定义3, 计算出的这部分老数据的权值会非常小, 大分母  $g(t - L)$  会使权值更小, 由于存储空间限制等缘故, 会损失数值精度。而局部余弦函数下, 分母  $g(t - L)$  的值始终小于1, 不会使权值进一步大幅度减小。

## 3 实验分析与评估

实验的硬件环境为: Intel Pentium Dual E2180, 2.0 GHz CPU, 1 GB 内存; 操作系统为 Windows XP Professional。为了便于观察实验效果, 使用一幅灰度数字图像(图2)的像素值模拟数据流, 对本文提出的衰变函数进行实验。图像中每个像素点的灰度值代表数据流中一个数据项的时间戳, 像素值越大, 时间戳越大, 数据越新; 反之像素值越小, 时间戳越小,

数据越旧。



图2 模拟数据流的数字图像

### 3.1 后向衰变下各衰变函数性能评估

首先对各衰变函数衰变速度进行实验。在实验测得的最优参数情况下,不难发现局部余弦函数(cos)的衰变速度介于指数函数(exp)和多项式函数(pol)之间,如图3所示。

其次,对各衰变函数计算得到的权值分布情况进行实验。图4为通过指数函数、多项式函数和局部余弦函数计算所得的权值分布图。图4(c)、(f)、(i)分别为近似最优参数下的分布。对于指数函数,实验测得近似最优参数为0.005(图4

(c)),减小(图4(b))或增大(图4(a))这个参数值,分布均匀性变差。对于多项式函数,实验测得近似最优参数为50(图4(f)),改变这个参数值,分布均匀性变差(图4(d)、(e))。对于局部余弦函数,由于图像中的灰度值在0~255,所以 $\lambda$ 取0.005(略小于 $\pi/(2 \times 255)$ )时近似最优(图4(i)),减小这个参数值,分布均匀性变差(图4(g)、(h))。比较图4(c)、(f)、(i)发现,最优参数下局部余弦函数区分新旧数据重要性的效果最好。

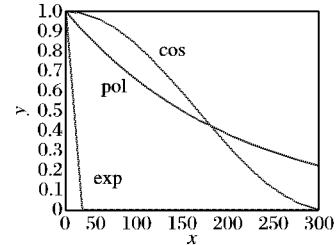


图3 衰变函数衰变速度

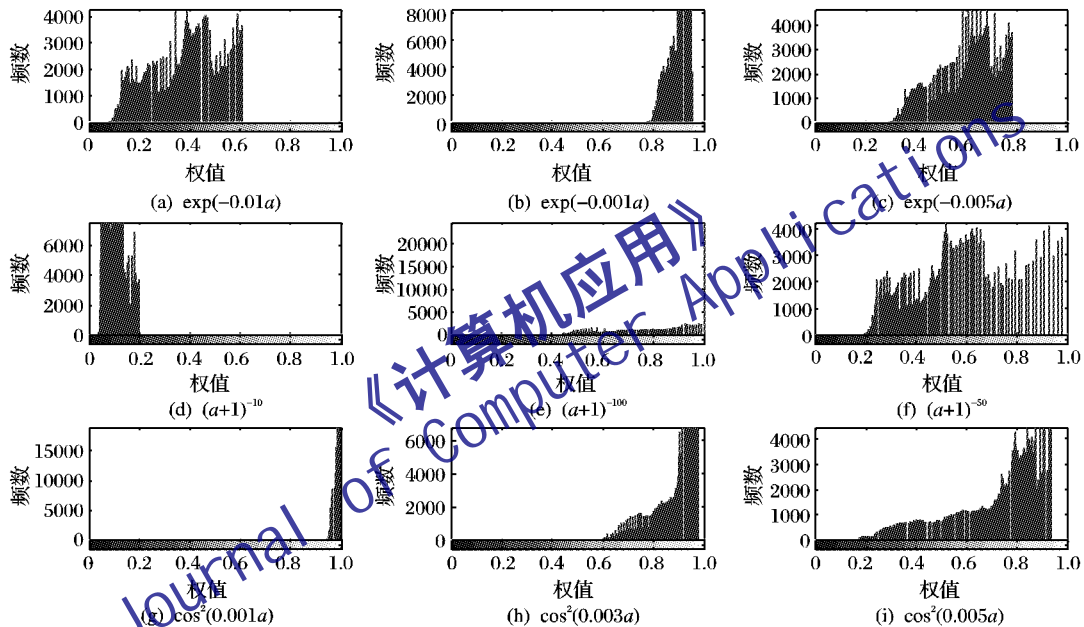


图4 后向衰变模型下衰变函数对应权值分布对比

接着,测试各衰变函数对乱序数据流的处理性能。在乱序情况下,用指数函数、多项式函数、局部余弦函数计算出的权值分别如图5所示。实验测得经指数函数计算后的部分权值超过1(图5(a));经多项式函数计算后的权值出现在正负数之间摇摆的现象(图5(b));而经局部余弦函数计算后的权值能够保持均匀的分布(图5(c))。

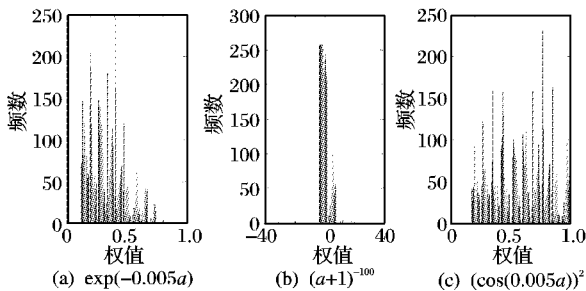


图5 后向衰变模型下乱序数据流的权值分布

为了更形象地比较各函数的优劣,将图2中图像的像素值(0~255)作为不同衰变函数的自变量,用计算结果(0~1)与像素值做点乘,恢复图像。所得图像在原来的基础上变灰,不同的像素点变化不同,像素值较小的点变得更黑,像素值较

大的点变黑幅度小。因此一个好的衰变函数会使恢复的图像更清晰。使用最优参数的各衰变函数,按上述方法测试所得结果如图6所示。可以看出,局部余弦衰变使图像更清晰(图6(d))。

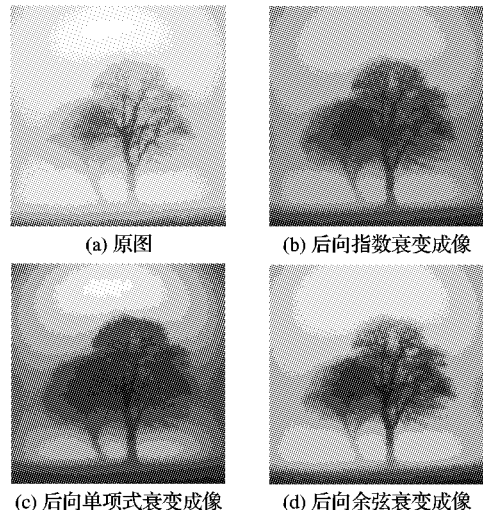


图6 后向衰变模型下各衰变函数性能的形象化比较



### 3.2 前向衰变下各衰变函数性能评估

在前向衰变模型下,实验测得指数函数和多项式函数的近似最优参数分别为0.005和2,局部余弦函数最优参数选择方法与后向衰变模型下的选择方法类似。各函数计算权值在近似最优参数下的分布如图7(c)、(f)、(i)所示,增大或减小参数值所得权值分布情况如图7(a)、(b)、(d)、(e)、(g)、(h)所示。不难看出局部余弦函数(图7(g)、(h)、(i))算得的权值在[0,1]区间分布最为均匀。

在前向衰变模型下,对乱序数据流的处理,用指数函数、多项式函数、局部余弦函数计算权值。实验测得经指数函数

和多项式计算后的部分权值超过1;而经局部余弦函数计算后的权值能够保持均匀的分布。这是因为局部余弦函数具有对称性。

在前向衰变模型下,用不同衰变函数恢复图2中图像的效果,局部余弦函数的效果最佳。

### 3.3 各衰变函数对小权值处理的精度实验

在数字图像的模拟数据流中,分别选择最优参数下的三种衰变函数测试小权值计算精度,结果发现指数和多项式函数计算得到的权值太小,以至于无法显示,而经局部余弦函数计算的权值仍在可接受范围之内。

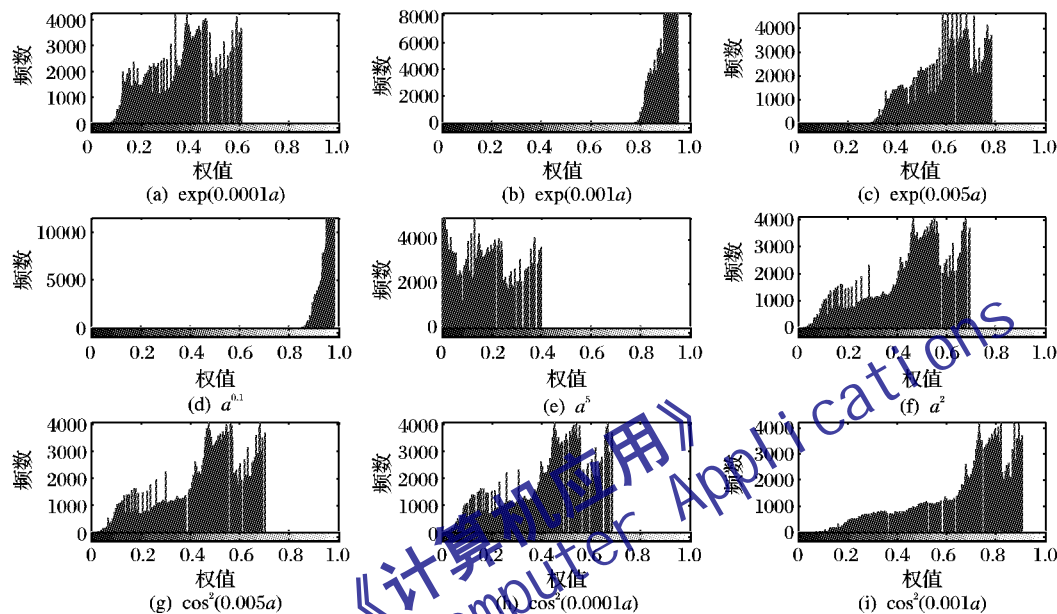


图7 前向衰变模型下衰变函数对应权值分布对比

## 4 结语

本文基于余弦函数的局部性质提出一种新的时间衰变模式。局部余弦衰变函数除衰变速度合理、参数易于确定外,还具有如下优点:更适合处理数据流中乱序数据项的权值;在前向衰变模型下,函数形式不会改变,且不存在小权值数据精度无法表示等问题。

另外,本文采用数字图像的像素值来模拟数据流,实验效果形象直观,但由于软硬件的限制,并不能完全模拟出数据流的无限性、瞬时性和流速不定性等特征,在一定程度上降低了处理数据的难度。进一步工作包括优化权值计算的具体算法和用实验室无线传感器网络采集数据流进行实验。

### 参考文献:

- [1] COHEN E, STRAUSSAT M. Maintaining time-decaying stream aggregates[J]. *Journal of Algorithms*, 2006, 59(1): 19-36.
- [2] CORMODE G, KORN F, TIRTHAPURA S. Exponentially decayed aggregates on data streams[C]// *Proceedings of the 2008 IEEE 24th International Conference on Data Engineering*. Washington, DC: IEEE Computer Society, 2008: 1379-1381.
- [3] CORMODE G, TIRTHAPURA S, XU BOJIAN. Time-decaying sketches for robust aggregation of sensor data[J]. *SIAM Journal on Computing*, 2009, 39(4): 1309-1339.
- [4] CORMODE G, TIRTHAPURA S, XU BOJIAN. Time-decayed correlated aggregates over data streams[J]. *Statistical Analysis and Data Mining*, 2009, 2(5): 294-310.

- [5] BABCOCK B, BABU S, DATAR M, *et al.* Models and issues in data stream systems[C]// *Proceedings of the 21st ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*. New York: ACM, 2002: 1-16.
- [6] CORMODE G, SHKAPENYUK V, SRIVASTAVA D, *et al.* A practical time decay model for streaming systems[EB/OL]. [2011-04-23]. [http://www.research.att.com/people/Cormode\\_Graham/library/publications/CormodeShkapenyukSrivastavaXu09.pdf](http://www.research.att.com/people/Cormode_Graham/library/publications/CormodeShkapenyukSrivastavaXu09.pdf).
- [7] MADNICK S E, WANG R Y, LEE Y W, *et al.* Overview and framework for data and information quality research[J]. *ACM Journal of Data and Information Quality*, 2009, 1(1): 1-22.
- [8] BABCOCK B, DATAR M, MOTWANI R. Sampling from a moving window over streaming data[C]// *Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms*. New York: ACM, 2002: 633-634.
- [9] DATAR M, GIONIS A, INDYK P, *et al.* Maintaining stream statistics over sliding windows[J]. *SIAM Journal on Computing*, 2002, 31(6): 1794-1813.
- [10] CORMODE G, KORN F, TIRTHAPURA S. Time-decaying aggregates in out-of-order streams[C]// *Proceedings of the 27th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*. New York: ACM, 2008: 89-98.
- [11] PHILLIP B, TIRTHAPURA S. Distributed streams algorithms for sliding windows[C]// *Proceedings of the 14th Annual ACM Symposium on Parallel Algorithms and Architectures*. New York: ACM, 2002: 63-72.