# 摘要

图数据结构能够很好的表达数据之间的关联性，因此在社交分析、商品推荐、舆论监测和欺诈检测等应用中被广泛使用。随着互联网的发展，现实社会和生产环境中的图数据越来越呈现海量和动态特性。

目前发展较为成熟的分布式图处理框架Google Pregel、Spark GraphX和GraphLab等，所处理的图数据都是静态稳定的图数据。针对动态变化图数据的处理，大多集中在算法研究层面上，仅有的KineoGraph和IncGraph等系统是以串行的方式进行增量式更新，无法充分利用多机并行更新的优势，此外SpecGraph提出的基于推测机制的并发更新模型，虽然提高了系统的并行性，但其模型的出发点是状态更新只与顶点接收消息有关而与原始状态无关，这个约束又使得模型的表达能力有限。

基于现有工作的不足，本文提出了一种基于状态更新的流式图计算模型GraphFlow，它将连续不断的图数据流抽象成一系列的事件流，将用户关心的图计算结果抽象成图的状态，用户只需要定义图状态如何根据到达的事件增量式地进行状态转换，就能够完成事件流到状态流的映射，提供实时反馈中间计算结果的能力。状态是用户进行定义，直接反应了用户所关注的信息，使得系统无需存储全部的图数据，而只需存储用户关心的数据，从而减少了存储开销；通过采用增量更新和变化传播的方式，使得增量数据对全图的影响范围更小，迭代收敛的速度更快；通过分析典型的图算法特征，抽象出两种常见的状态类型：独立状态和关联状态；通过对独立状态的分布式存储和并发更新策略，以及对关联状态的细粒度分布式锁的更新策略，能够有效解决关联状态下更新冲突的问题，从而提高了系统的并行性和正确率。

试验结果表明，相比较传统的批处理图计算系统，GraphFlow能够实时计算并反馈结果，90%的图数据更新请求都能在12ms内得到响应；相比较动态图数据的估计模型，GraphFlow的准确率较高，计算偏差在5%以内；而采用细粒度锁的方式进行并发更新时，更新冲突的概率在3%以内；系统的准确率高，实时性好，符合流式图计算的要求。

**关键词**: 图处理系统；实时计算；分布式系统；状态更新；增量图计算；分区并行更新；细粒度锁

# Abstract

Graph perfectly reflects the association between data, thus it is widely used in applications like social analytics, commodity recommendation, public opinion monitoring and fraud detection. With the development of the Internet, the graph data becomes more huge and increase more rapidly.

While the well-developed distributed graph computing framework like Google Pregel、Spark GraphX and GraphLab focus on the stable graph data, as for dynamic graph data, the main research focus on specific algorithms rather than system or framework. Only KineoGraph and IncGraph apply incremental updating in a serial way instead of parallel updating. In addition, SpecGraph realizes the parallel updating model based on speculative mechanism, but its model assumes the state update is only related to the receiving message and has nothing to do with the original state, this strong assume constraints expressive capability of this model.

Due to the disadvantages of current studies as mentioned above, this paper proposes a streaming graph computing model GraphFlow which is based on state updates, this model abstract the continuous change of the graph in a series of events, and abstract the computing result of the graph into graph state. Users only need to define graph state and how graph state can be transformed according to the current state and arrival event. By this way, the system can complete the event flow to the state flow mapping, and provide real-time feedback of the intermediate state of the streaming graph computing. The graph state which the user define directly reflects the user’s concern, making the system only need to store the user defined data instead of the whole graph data, thereby reducing the storage overhead; by using the strategy of incremental updates and Change propagation to transform the state of the graph, the sphere of influence of the incremental data is limited within a certain range and the iteration of the graph computing converges faster; by analyzing the characteristics of the typical graph algorithm, two kinds of common graph state types are abstracted: independent state and association state; by using distributed storage and concurrent update strategy for independent state and fine-grained distributed lock strategy for associated state, the updating conflict problem is solved effectively, thus improving the system parallelism and correct rate.

The experimental result show that GraphFlow can calculate and feedback the result in real time compared with the traditional batch graph computing system, 90% graph update request can get the response within 12ms; Compared with the estimated model of the streaming graph computing, the accuracy of GraphFlow is higher, the computing deviation is within 5%; the probability of concurrent update conflict is within 3% by using the fine-grained distributed lock; thus GraphFlow has high accuracy and good real-time performance, which meets the requirements of the streaming graph computing.