# 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。而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。

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.

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。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，以及对关联状态的分区并行更新策略和细粒度锁的更新策略，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。

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

Considering all these disadvantages, a dynamic graph calculation model GraphFlow based on state updating is proposed in this paper. The GraphFlow will abstract the change of graph data in dynamic graph processing into a series of event flows, and abstract graph results concerned by users into the graph states. Users only need to define graph states and how state can be transformed according to the current state and arrival events. 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 dynamic graph computation. State is to define user's data directly from the user's point of view, which not only reduces the storage cost, but also makes the expression of the model stronger; through analyzing the typical graph algorithm characteristics, we (or machine) abstract two kinds of common state types: independent state and associative state. Through the distributed storage and concurrent update strategy for the independent state, and the update policy of the partition and the update strategy of the fine-grained lock for associative state, the problem of updating conflict in associative state can be solved effectively. Furthermore, the parallelism of the system is improved.

The experimental results show that GraphFlow can calculate and feedback the results in real time compared with the traditional batch graph computing system. .90% of the graph data update request time is within 12ms. Compared with the estimation model of dynamic data, the accuracy of GraphFlow is higher, and the calculation error is less than 5%; When using the parallel update of the fine-grained lock, the update conflict is with 3%. Thus, this system has high accuracy and high real-time quality, and fulfills the requirement of fluid graph calculation.

**Key words:** graph processing system; real-time computing; distributed system; state update; incremental graph processing; partition parallelism update; fine-grained lock