Distributed Partial Simulation for Graph Pattern Matching

AISSAM AOUAR[1], SAÏD YAHIAOUI[2], LAMIA SADEG[1], NADIA NOUALI-TABOUDJEMAT[2] KADDA BEGHDAD BEY[1]

[1]Ecole Millitaire Polytechnique, BP 17, Bordj el Bahri, Algiers, 16111, Algeria
[2]CERIST Research Center on Scientific and Technical Information, Ben Aknoun, Algiers, 16030, Algeria
Email: aissam.a@gmail.com

Pattern matching in big graphs is important for different modern applications. Recently, this problem was defined in terms of multiple extensions of *graph simulation*, to reduce complexity and capture more meaningful results. These results were achieved through the relaxation of commonly used constraint in *subgraph isomorphism* pattern matching. Nevertheless, these graph simulation variant models are still too strict to provide results in many cases, especially when analyzed graphs contain anomalies and incomplete information. To deal with this issue, we introduce a new graph pattern matching method, called *partial simulation*, capable of retrieving matches despite missing parts of the pattern graph, such as vertices and/or edges. Furthermore, considering the number and inequality of the outputs, we define a relevance function to compute a value expressing how each match vertex respects the pattern graph. Similarly, we define *partial dual simulation* graph pattern matching that returns vertices that satisfy a part of the *dual simulation* constraints and assigns a relevance value to them. Additionally, we provide distributed scalable algorithms to evaluate the proposed partial simulation methods based on the distributed vertex-centric programming paradigm. Finally, our experiments on real-world data graphs demonstrate the effectiveness of the proposed models and the efficiency of their associated algorithms.

Keywords: Big graphs; Graph pattern matching; Graph simulation; Distributed computing; Vertex-centric programming

Received 28 March 2022; revised 28 March 2022

1. INTRODUCTION

Nowadays, graphs are used intuitively to model the relationships between data entities in a wide range of modern applications, including social networks, homeland security, biology, cyber-security and computer networks. These graphs are huge, including billions of vertices and edges, and are distributed across multiple data centers around the globe. For instance, according to Meta's second-quarter 2022 results report, Facebook has approximately 2.93 billion monthly active users [1]. Processing this massive number of linked entities to extract valuable information is a significant challenge in this context, where traditional graph frameworks fall short and new concepts for big graph analysis are needed. Graph Pattern Matching (GPM) is one of the frequent analysis techniques in graph processing. It consists in finding matched subgraphs for a given pattern (query) graph Q in a data graph G and it is generally defined in terms of subgraph isomorphism. The subgraph isomorphism matching model returns the strictest results for GPM in terms of topology [2], but this method comes with an NP-complete complexity [3] which is impractical for big graphs. Moreover, big graphs are often littered with missing or incorrect data and the strictness of an exact match is too rigid to return results. Likewise, information retrieval is an interactive and iterative process and user needs to reformulate the initial query because it is over- or under-specified or simply contains errors [4]. The task is even more complex when a user does not have sufficient knowledge about the data graph [5]. In that case, the exact GPM that can not tolerate query design errors is inappropriate. To address the computational complexity of exact matching, the uncertainty associated with data graphs, and the potential of mistakes in pattern graphs, researchers propose *inexact GPM* approaches. They relax matching conditions in order to find an approximate solution in a reasonable time, owing to the fact that their matched subgraph is less sensitive