



ΠΑΡΟΥΣΙΑΣΗ ΔΙΔΑΚΤΟΡΙΚΗΣ ΔΙΑΤΡΙΒΗΣ

ΗΜΕΡΟΜΗΝΙΑ: Δευτέρα, 16 Ιουλίου 2018

ΩΡΑ: 12:00

ΑΙΘΟΥΣΑ: Αίθουσα Σεμιναρίων
Κτήριο Τμήματος Μηχανικών Η/Υ & Πληροφορικής

ΟΜΙΛΗΤΗΣ: Κωνσταντίνος Σεμερτζίδης

Θ έ μ α

«Storage, Processing and Analysis of Large Evolving Graphs»

Επταμελής Εξεταστική Επιτροπή:

1. **Ευαγγελία Πιτουρά**, Καθηγήτρια του Τμήματος Μηχανικών Η/Υ & Πληροφορικής του Πανεπιστημίου Ιωαννίνων (επιβλέπουσα).
2. **Παναγιώτης Βασιλειάδης**, Αναπληρωτής Καθηγητής του Τμήματος Μηχανικών Η/Υ & Πληροφορικής του Πανεπιστημίου Ιωαννίνων.
3. **Παναγιώτης Τσαπάρας**, Αναπληρωτής Καθηγητής του Τμήματος Μηχανικών Η/Υ & Πληροφορικής του Πανεπιστημίου Ιωαννίνων.
4. **Νικόλαος Μαμουλής**, Αναπληρωτής Καθηγητής του Τμήματος Μηχανικών Η/Υ & Πληροφορικής του Πανεπιστημίου Ιωαννίνων.
5. **Γεωργία Κολωνιάρη**, Επίκουρη Καθηγήτρια του Τμήματος Εφαρμοσμένης Πληροφορικής του Πανεπιστημίου Μακεδονίας.
6. **Ιωάννης Κωτίδης**, Αναπληρωτής Καθηγητής του Τμήματος Πληροφορικής του Οικονομικού Πανεπιστημίου Αθηνών.
7. **Εβημαρία Τερζή**, Αναπληρώτρια Καθηγήτρια, Department of Computer Science, Boston University, USA.



Περίληψη

In recent years, increasing amounts of graph structured data are being made available from a variety of sources, such as social, citation, computer and hyperlink networks. Their continuous evolution is becoming a subject that attracts considerable attention and finds a wide spectrum of applications ranging from social network marketing to virus propagation and digital forensics. Although the analysis of the graph evolution is important of our understanding of the network, the main focus of research has been on efficiently storing and retrieving the graph snapshots. Furthermore, processing graph data through a variety of graph queries including reachability, distance and pattern-based ones, is limited to static graphs, leaving query processing on evolving graphs unexplored.

This dissertation focus on managing and querying the full history of a graph as it evolves. We introduce a compact representation of an evolving graph, where each graph element i.e., node or edge is annotated with the set of time intervals that refer to the existence of each element through the graph evolution.

We then include studies of different ways of extracting information from the evolving graph by posing different queries on a sequence of graph snapshots. We call such queries *historical queries*. In particular, we first introduce historical graph traversal queries that consider paths that existed in a sufficient number of graph snapshots. We exploit variants of two types of historical traversal queries, reachability and shortest paths. Historical reachability queries ask whether two nodes are connected in some time instance, in all time instances, or in a sufficient number of time instances whereas historical shortest path queries ask for the shortest path between two nodes posing requirements on the lifespan of such paths. We provide efficient algorithms for supporting historical graph traversal queries. We propose effective implementations of our algorithms based on time index structures for in-memory and graph database systems and provide an extensive experimental evaluation of various aspects of our approach.

We also formalize a new problem that of finding the top- k most durable matches of an input graph pattern query, that is the matches that exist for the longest period of time. Locating durable matches in the evolution of large graphs is important for our understanding of the network, and it may be crucial for many applications. Applying previous approaches to pattern matching problem at each snapshot and aggregating the results for large networks and long sequences, it is computationally expensive, since all matches have to be generated at each snapshot, including those appearing only once. Thus, we propose a new efficient and effective approach that uses appropriate time indexes to prune the search space and strategies to estimate the duration of the seeking matches in large evolving graphs.

Furthermore, we systematically study density in evolving graphs, and provide definitions for density over multiple graph snapshots, that capture different



semantics of connectedness over time. We study the complexity of the different variants of the problem and we propose a generic algorithmic framework for solving our problems, that works in linear time. Our experimental evaluation shows both the efficiency of our algorithms and the usefulness of the problems.

Finally, we introduce three approaches of modeling evolving graphs in native graph databases, as well as, algorithms for processing historical queries that use these approaches.