Scalable in-memory graph databases

Highlights
Developing a representation for massive graphs based upon compact data structures
Creating functional in-memory graph database
Focussing on the representing undirected graphs and path query operations

Overview
We live in a networked world: the relationship between entities are even more meaningful than the entities themselves. An obvious example is Facebook, where each entity represents an individual or an organization, and two entities are related if they are “friends” on Facebook. It is possible to find out a lot about an individual from their set of “friends” and how they interact with them (do they react often to their posts?). The logical representation of this information is called a graph, and using the terminology of graphs we would call the individuals “vertices” and the relationships as “edges”. Edges can be labelled with information e.g. the kind of relationship (friend, follower) and some way of quantifying how “strong” this relationship is. The Internet is another real-world object that is easily modelled as a graph. Other real-life examples of graphs are complex protein molecules, metabolic networks, transport and financial networks etc.

Storing, querying and mining massive graphs is essential for a huge variety of “intelligent” systems. Although graph data can be represented in relational databases (the standard workhorse of data storage and manipulation), it is unwieldy to do so and relational databases are not optimized for answering graphical queries. As a result, their performance is poor on the kind of queries that are most necessary to mine the structure of the network.

This project will aim to work on developing a representation for massive graphs that will be based upon compact data structures. Compact data structures help represent data in reduced space while allowing it to be queried, navigated, and operated in compressed form. They also reduce the resources needed in distributed deployments and make better use of the limited memory in low-end devices. The field has developed rapidly, reaching a level of maturity that allows practitioners and researchers in application areas to benefit from the use of compact data structures.

Representing massive networks such as the internet in computer memory. Image from the Opte Project ([www.opte.org](http://www.opte.org))
Compact data structures are highly optimized to support a precisely specified set of desired operations on the data. Thus, it is important to have a standardized set of operations: fortunately, such a standard (G-CORE) has just been published.

The project will aim to create of a functional in-memory graph database that supports at least a substantial subset of the G-CORE language. It is proposed to focus on the key problem of representing undirected graphs with simple annotations on the edges, and to focus on a set of path query operations on the labels of paths.

Methodology, Critical Skills and Training and Development

The basic methodology for this work is reasonably clear. We would study the G-CORE language to elicit requirements, and identify key building blocks. We would develop algorithms and representations for these building blocks, analyse them mathematically and then eventually write code based upon the developed algorithms. This “overall” methodology has been used to success in the SiXML project https://www.cs.le.ac.uk/SiXML/ and related developments in other places including U. Chile.

As stated below, we would be looking for students with a strong background in computer science and programming. A number of specialized skills can be identified:

- algorithm analysis (if student lacks this background – module offered at UoL)
- data compression (as above)
- performance analysis and source code optimization (over the years I and my students have built up a suite of “microbenchmarks” that demonstrate the key (low-level) drivers of performance.
- Succinct and compressed data structures. A PhD course on this topic was offered this summer at University of la Coruna; it is hoped that similar courses will be offered elsewhere during the project.
  https://www.udc.es/gl/iss/courses/courses_2018/Compact_data_structures/

In addition, I have had discussions with Prof. Marcelo Arenas (Pontifical University of Chile), a leading researcher in Database research, who has expressed interest in the above project. Indeed, he was planning to discuss this with Prof. Navarro (U. Chile, proposed additional supervisor), who works in much the same area as I do. We would also draw in relevant expertise from Bell Labs Dublin (Nicholson) and University of Helsinki (Puglisi). Each of these sites has relevant expertise and can contribute development and training resources (in the form of RAs or PhD students).

Further Reading

1. Renzo Angles, Marcelo Arenas, Pablo Barceló, Peter A. Boncz, George H. L. Fletcher, Claudio Gutierrez, Tobias Lindaaker, Marcus Paradies, Stefan Plantikow, Juan F. Sequeda, Oskar van Rest, Hannes Voigt: G-CORE: A Core for Future Graph Query Languages. SIGMOD Conference 2018: 1421-1432

Additional Entry Requirements

- Master’s degree in Computer Science or a related subject
- Good programming skills in Java or C++
- Experience with algorithm analysis and computational complexity theory
Experience with analysing and optimizing the performance of programs.

Funding
This research project is one of a number of projects in the College. It is in competition for funding with one or more of these projects. Usually the project which receives the best applicant will be awarded the funding.

Home/EU Applicants
This project is eligible for a fully funded EPSRC studentship which includes:

- A full UK/EU fee waiver for 3.5 years
- An annual tax free stipend of £14,777 (2018/19)
- Research Training Support Grant (RTSG)

Studentships are available to UK/EU applicants who meet the EPSRC Residency Criteria; if you have been ordinarily resident in the UK for three years you will normally be entitled to apply for a full studentship.

If you are an EU student and do not meet the residency criteria, please contact csepgr@le.ac.uk for more information on the funding options available.

International Applicants
- Unfortunately, there is no funding for international students on this project.

Application Instructions
The online application and supporting documents are due by Monday 21st January 2019.

Any applications submitted after the deadline will not be accepted for the studentship scheme.

References should arrive no later than Monday 28th January 2019.

Applicants are advised to apply well in advance of the deadline, so that we can let you know if anything is missing from your application.

Required Materials
1. Online application form
2. Two academic references
3. Transcripts
4. Degree certificate/s (if awarded)
5. Curriculum Vitae
6. EPSRC Studentship Form
7. English language qualification

Applications which are not complete by the deadline will not be considered for the studentship scheme. It is the responsibility of the applicant to ensure the application form and documents are received by the relevant deadlines.
All applications must be submitted online, along with the supporting documents as per the instructions on the website.

Please ensure that all email addresses, for yourself and your referees, are correct on the application form.

For more information, please visit our website at:

https://www2.le.ac.uk/colleges/scieng/research/postgraduate-opportunities/epsrc-2019/instructions