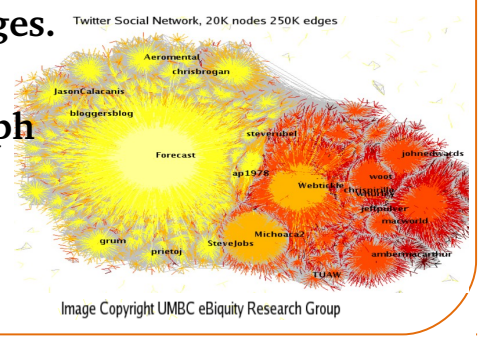
Automatic Parallelization of Graph Queries with MapReduce

Tao Zan[1] Yu Liu[1] Zhenjiang Hu[2]
[1]The Graduate University for Advanced Studies [2]National Institute of Informatics

Problem

♦ Graphs in real world can be quite huge with millions of nodes and edges.



♦ How to parallelize graph queries efficiently is an important issue.

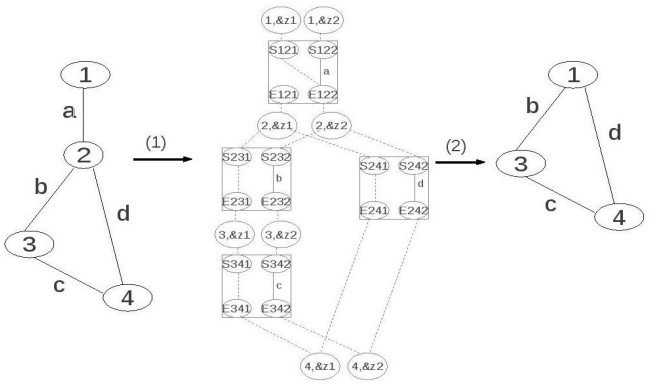
Goal

We aim at a new way of processing graph queries with MapReduce by translating UnQL query to structural recursion, then parallelizing structural recursion. When achieving this goal, thus can we use UnQL for large scale data processing by MapReduce.

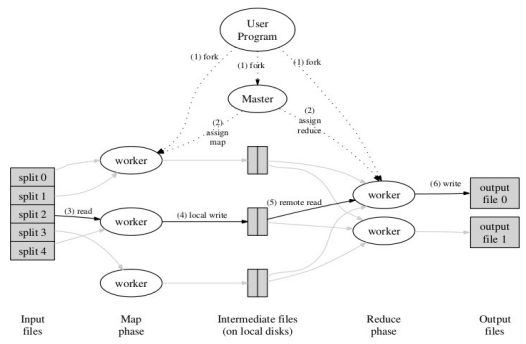
Background

♦ UnQL is a language that was designed for querying unstructured data.

An example:
Q1 = select t
where ∗.a.t in $db



♦ MapReduce was introduced for processing huge datasets on clusters of computers.



Methods, Results and Future Work

♦ Methods

● Properties for Parallelization

All join-free queries can be highly parallelized in MapReduce based on three decomposable properties for structural recursion.

rec(e)({$l : $g}) = e @ rec(e)($g)
rec(e)($g1 @ $g2) = rec(e)($g1) @ rec(e)($g2)
rec(e)(cycle($g)) = cycle(rec(e)($g))

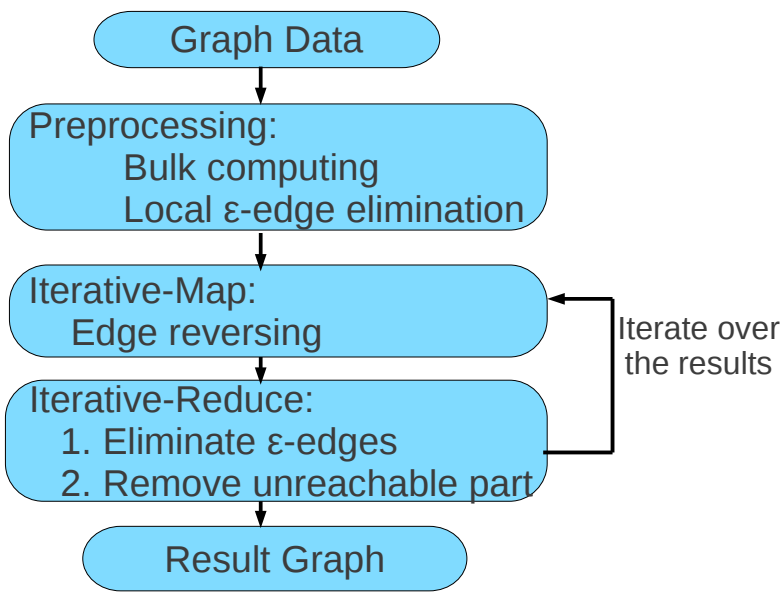
A graph can be expressed as:

g = &x @ cycle(g1 ⊕ g2 ⊕ ... ⊕ gn)

By induction:

rec(e)(g)
= rec(e)(&x) @ rec(e)(cycle(g1 ⊕ g2 ⊕ ... ⊕ gn))
= rec(e)(&x) @ cycle(rec(e)(g1 ⊕ g2 ⊕ ... ⊕ gn))
= rec(e)(&x) @ cycle(rec(e)(g1) ⊕ rec(e)(g2) ⊕ ... ⊕ rec(e)(gn))

● Parallel Computation with MapReduce



♦ Results

We evaluated in Hadoop cluster constructed with 8 virtual machines (VM). Each VM has 3 GB memory and one single-core 2.8 GHz CPU.

Q1 = select {a : t} where ∗.a.t in $db
Q2 = select {a : {select b where ∗.b.t2 in $t1 }} where ∗.a.t1 in $db
Q3 = select {a : {select {b : {select c where c.t3 in $t2 }} where b.t2 in $t1 }} where ∗.a.t1 in $db

Graph Queries on Hadoop

Queries	1+3 million	10+30 million	20+60 million
Q_1	104 sec.	545 sec.	1091 sec.
Q_2	138 sec.	773 sec.	1429 sec.
Q_3	251 sec.	1073 sec.	1683 sec.

(x+y)million : a graph with x million vertexes and y million edges

♦ Future work

● Parallelization of graph query on cyclic graphs.
● A combinational approach for parallelizing graph query UnQL.

NII
連絡先：:胡振江（Zhenjiang HU）／ 国立情報学研究所 アーキテクチャ科学研究系 教授
TEL：03-4212-2530 FAX：03-4212-2533 Email：hu@nii.ac.jp