OPEN PROBLEMS IN MASSIVELY PARALLEL QUERY PROCESSING

Paris Koutris
Venice, ICDT 2017

University of Wisconsin-Madison
What is the parallel complexity of computing a join query $q$ with a massively parallel cluster of $p$ machines?
MPC = **Massively Parallel Communication** [BKS - PODS’13]

- The data of size $M$ is distributed evenly among the $p$ machines
- The computation proceeds in rounds: each round performs **local computation** followed by **synchronized communication**
- In the end the output is the union of the output of the $p$ machines

We measure the complexity using two parameters

- $r$: the number of rounds
- $L$: the maximum amount of data received by any machine at any round

\[
\frac{M}{p} \leq L \leq M
\]

In general, $L = M/p^{1-\varepsilon}$ for some parameter $0 \leq \varepsilon < 1$
The MPC Model

MPC = Massively Parallel Communication [BKS - PODS’13]

- The data of size $M$ is distributed evenly among the $p$ machines
- The computation proceeds in rounds: each round performs local computation followed by synchronized communication
- In the end the output is the union of the output of the $p$ machines

We measure the complexity using two parameters

- $r$: the number of rounds
- $L$: the maximum amount of data received by any machine at any round

\[
\frac{M}{p} \leq L \leq M
\]

In general, $L = M/p^{1-\varepsilon}$ for some parameter $0 \leq \varepsilon < 1$
A fractional edge packing $u$ for a CQ $q$ assigns a weight $u_j$ to each relation $R_j$, such that for every variable $x$, the sum of the weights of the relations that contain $x$ is $\leq 1$.

**Theorem (BKS - PODS'14)**

Any MPC algorithm that computes a full CQ $q$ in one round must have load

$$L \geq \max_u \left( \frac{\prod_{j=1}^{\ell} M_j^{u_j}}{p} \right)^{1/\sum_j u_j}$$

For data without skew, the HyperCube algorithm can achieve the above optimal load.

When all sizes are at most $M$, the bound becomes $L \geq M/p^{1/\tau^*}$, where $\tau^*$ is the maximum edge packing.
A **matching database** is a database where each value of the domain appears exactly once (degree \(d = 1\))

**Theorem (BKS - PODS'13)**

The query \(L_k = R_1(x_1, x_2), R_2(x_2, x_3), \ldots, R_k(x_k, x_{k+1})\) can be computed on matching databases in \(r\) rounds with load

\[
L = O\left(\frac{M}{p^{1/\lceil k^1/r/2\rceil}}\right)
\]

The algorithm is (almost) optimal

**OPEN PROBLEM #1:** What is the optimal load for a full CQ \(q\) in \(r > 1\) rounds on data without skew?

Not even known for matching databases!
A **matching database** is a database where each value of the domain appears exactly once (degree $d = 1$)

**Theorem (BKS - PODS’13)**

The query $L_k = R_1(x_1, x_2), R_2(x_2, x_3), \ldots, R_k(x_k, x_{k+1})$ can be computed on matching databases in $r$ rounds with load

$$L = O\left(\frac{M}{p^{1/\left\lceil k^{1/r}/2\right\rceil}}\right)$$

The algorithm is (almost) optimal

**OPEN PROBLEM #1**: What is the optimal load for a full CQ $q$ in $r > 1$ rounds on data without skew?

Not even known for matching databases!
A fractional edge cover of a CQ $q$ assigns a weight $v_j$ to each relation $R_j$ such that for every variable $x$, the sum of the weights of the relations that contain $x$ is $\geq 1$

- $\rho^*(q) =$ minimum edge cover
- AGM Bound: If each relation has size $\leq M$, the maximum output can be at most $M\rho^*(q)$ [AGM - FOCS’08]

**Theorem (KBS - ICDT’16)**

*For any full CQ $q$, there exists a family of instances of size at most $M$ such that every MPC algorithm that computes $q$ with $p$ machines using a constant number of rounds requires load $\Omega(M/p^{1/\rho^*(q)})$*

**OPEN PROBLEM #2:** What is the lower bound when we restrict the relations to have different sizes $M_j$?
A fractional edge cover of a CQ $q$ assigns a weight $v_j$ to each relation $R_j$ such that for every variable $x$, the sum of the weights of the relations that contain $x$ is $\geq 1$

- $\rho^*(q) = \text{minimum edge cover}$
- AGM Bound: If each relation has size $\leq M$, the maximum output can be at most $M^{\rho^*(q)}$ [AGM - FOCS'08]

**Theorem (KBS - ICDT'16)**

*For any full CQ $q$, there exists a family of instances of size at most $M$ such that every MPC algorithm that computes $q$ with $p$ machines using a constant number of rounds requires load $\Omega(M/p^{1/\rho^*(q)})$*

**OPEN PROBLEM #2:** What is the lower bound when we restrict the relations to have different sizes $M_j$?
The $\Omega(M/p^{1/p^*(q)})$ lower bound is optimal for:

- CQs with binary signature [KS - PODS’17]
- Loomis-Whitney joins, Star joins [KBS - ICDT’16]

OPEN PROBLEM #3: Is it possible to compute any full CQ using a constant number of rounds with load $O(M/p^{1/p^*(q)})$?

OPEN PROBLEM #4: How many rounds are actually necessary to achieve the optimal load?
The $\Omega(M/p^{1/p^*(q)})$ lower bound is optimal for:

- CQs with binary signature [KS - PODS’17]
- Loomis-Whitney joins, Star joins [KBS - ICDT’16]

**OPEN PROBLEM #3:** Is it possible to compute any full CQ using a constant number of rounds with load $O(M/p^{1/p^*(q)})$?

**OPEN PROBLEM #4:** How many rounds are actually necessary to achieve the optimal load?
The $\Omega(M/p^{1/p^*(q)})$ lower bound is optimal for:

- CQs with binary signature [KS - PODS’17]
- Loomis-Whitney joins, Star joins [KBS - ICDT’16]

**OPEN PROBLEM #3:** Is it possible to compute any full CQ using a constant number of rounds with load $O(M/p^{1/p^*(q)})$?

**OPEN PROBLEM #4:** How many rounds are actually necessary to achieve the optimal load?
Consider the join query

\[ Q = R(x,z), S(y,z) \]

- **No Skew**: the 1-round load is \( O(M/p) \)
- **Worst-case**: the 1-round load is \( O(M/p^{1/2}) \)
- If we know that the degrees of a \( z \)-value \( h \) are \( m_R(h) \) and \( m_S(h) \), we can design an 1-round algorithm with load

\[
\max \left( \frac{M}{p}, \sqrt{\sum_h m_R(h) \cdot m_S(h)} \right)
\]

**OPEN PROBLEM #5**: What are the upper and lower bounds for the load when the degrees of values are known?
Consider the join query

\[ Q = R(x, z), S(y, z) \]

- **No Skew**: the 1-round load is \( O(M/p) \)
- **Worst-case**: the 1-round load is \( O(M/p^{1/2}) \)
- If we know that the degrees of a \( z \)-value \( h \) are \( m_R(h) \) and \( m_S(h) \), we can design an 1-round algorithm with load

\[
\max \left( \frac{M}{p}, \sqrt{\frac{\sum_h m_R(h) \cdot m_S(h)}{p}} \right)
\]

**OPEN PROBLEM #5**: What are the upper and lower bounds for the load when the degrees of values are known?
Connections With the External Memory Model

The External Memory Model:

- **internal** memory of $W$ words + **external** memory of unbounded size
- data can move between the memories in **blocks** of consecutive $B$ words
- the **I/O complexity** of an algorithm is the number of blocks that are moved in and out of the internal memory

We can simulate an MPC algorithm that computes a full CQ $q$ to obtain an EM algorithm that: this gives I/O optimal algorithms for certain CQs

**OPEN PROBLEM #6**: Does every simulation of an optimal MPC algorithm lead to an I/O worst-case optimal algorithm?
The External Memory Model:

- **internal** memory of $W$ words + **external** memory of unbounded size
- data can move between the memories in **blocks** of consecutive $B$ words
- the **I/O complexity** of an algorithm is the number of blocks that are moved in and out of the internal memory

We can simulate an MPC algorithm that computes a full CQ $q$ to obtain an EM algorithm that: this gives I/O optimal algorithms for certain CQs

**OPEN PROBLEM #6**: Does every simulation of an optimal MPC algorithm lead to an I/O worst-case optimal algorithm?