Communication Complexity in the Field: New Questions from Practice

Qin Zhang

Indiana University Bloomington

BIRS Workshop
March 20, 2017

This talk

Not on a particular problem

Try to present a few new questions that I have encountered when trying to apply comm. complexity in various settings

Agenda

I will talk about

1. Number-in-hand CC with input sharing
   – Distributed computation of graph problems
2. Primitive problems overlap; direct-sum does not apply
   – Distributed joins
3. Higher LB in simultaneous comm. than one-way comm.?
   – Sketching edit distance

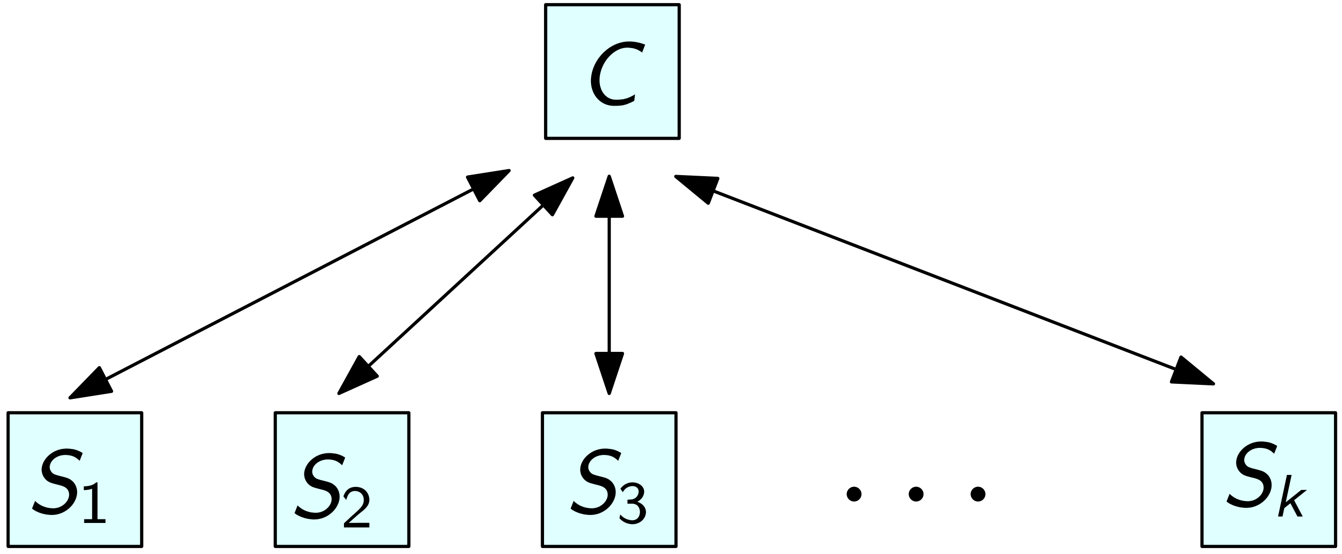
Distributed graph computation

Real world systems: Pregel, Giraph, GPS, GraphLab, etc.

The coordinator model

The coordinator model: We have k machines (sites) and one central server (coordinator).

– Each site has a 2-way comm. channel with the coordinator.
– Each site has a piece of data x_i.
– **Task**: compute $f(x_1, \ldots, x_k)$ together via comm., for some f. Coordinator outputs the answer.
– **Goal**: minimize total communication



Distributed graph computation

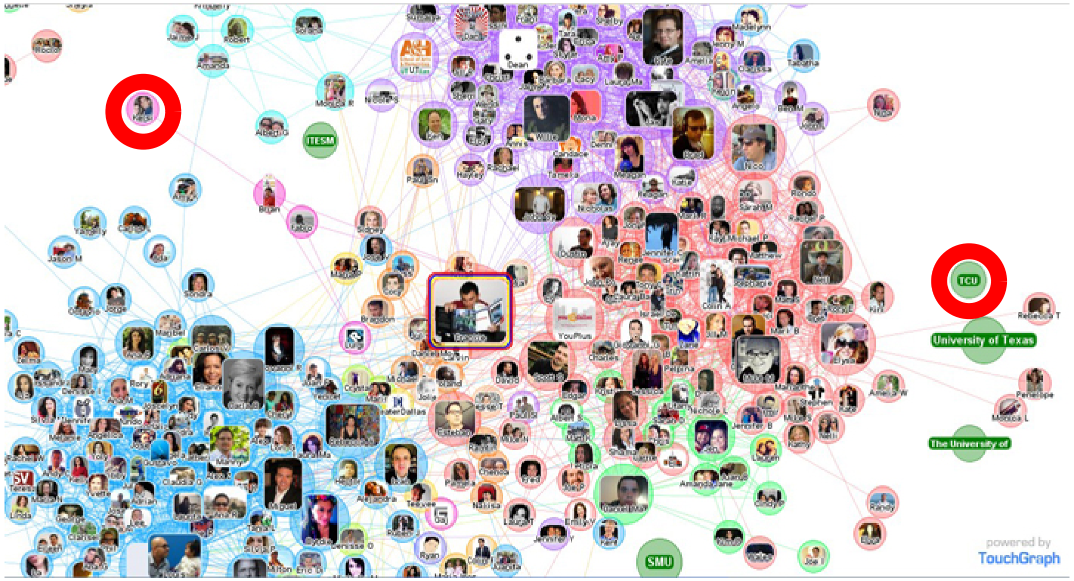
Let's think about the **graph connectivity** problem:

k sites each holds a portion of a graph.
Goal: compute whether the graph is connected.

Distributed graph computation

Let's think about the **graph connectivity** problem:

k sites each holds a portion of a graph.
Goal: compute whether the graph is connected.

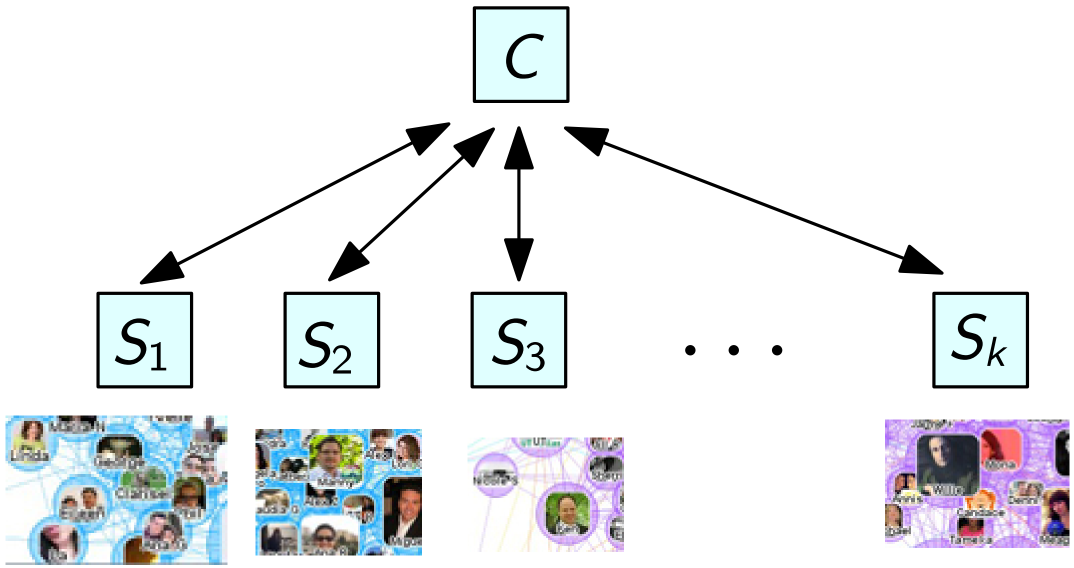


Distributed graph computation

Let's think about the **graph connectivity** problem:

k sites each holds a portion of a graph.

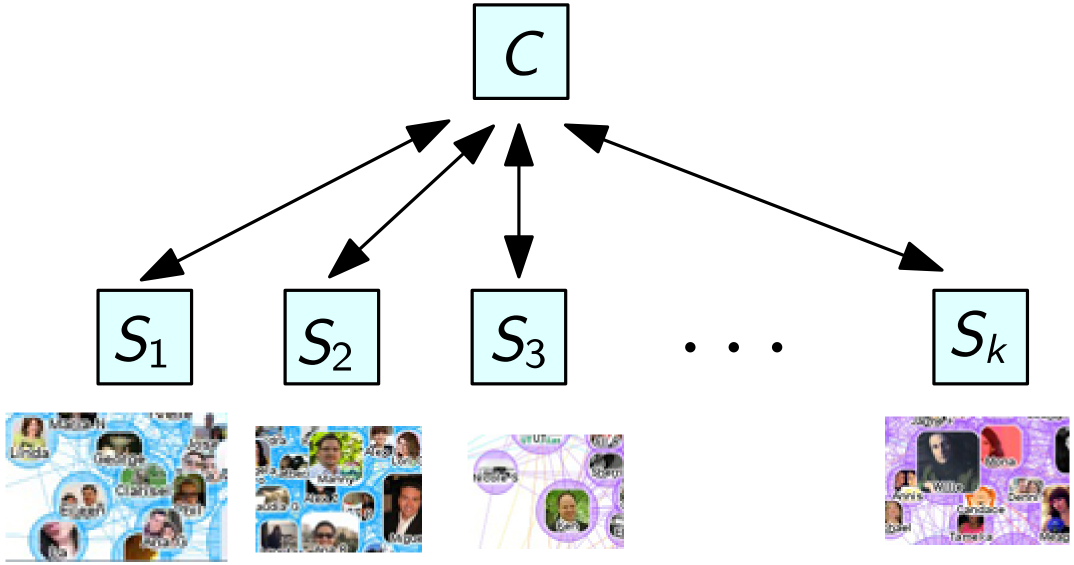
Goal: compute whether the graph is connected.



Distributed graph computation

Let's think about the **graph connectivity** problem:

k sites each holds a portion of a graph.
Goal: compute whether the graph is connected.



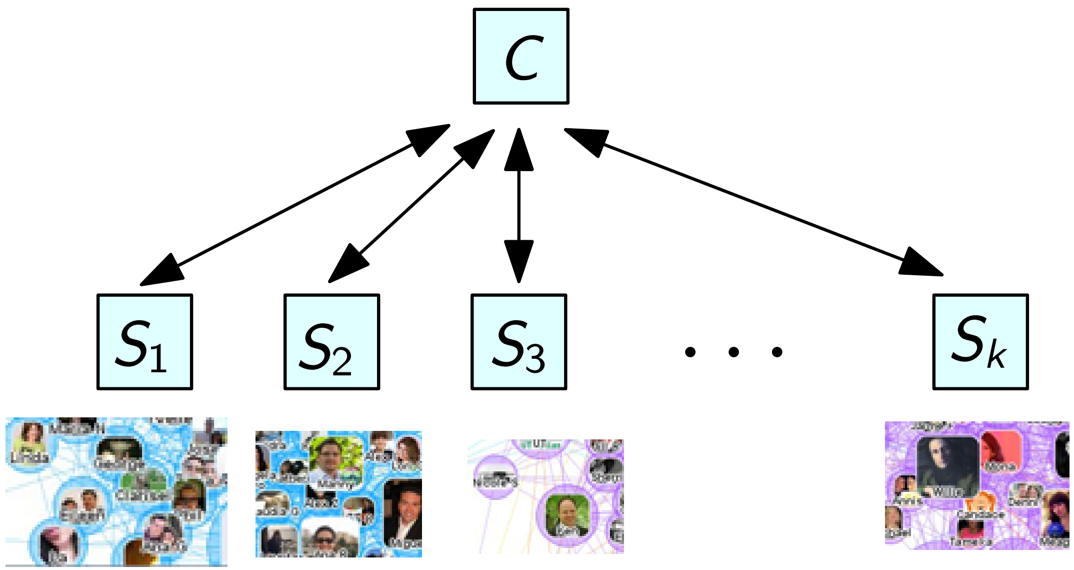
A trivial solution:
each S_i sends a local spanning forest to C. Cost $O(kn \log n)$ bits.

n: # nodes of the graph

Distributed graph computation

Let's think about the **graph connectivity** problem:

k sites each holds a portion of a graph.
Goal: compute whether the graph is connected.



A trivial solution:
each S_i sends a local spanning forest to C. Cost $O(kn \log n)$ bits.

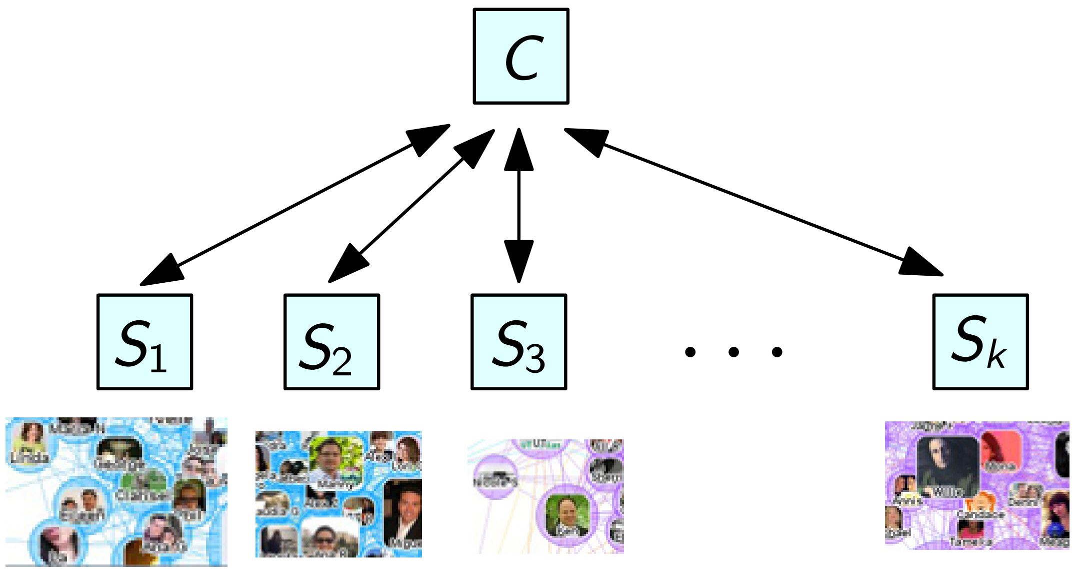
n: # nodes of the graph

Can we do better, e.g., $o(kn)$ bits of comm. in total?

Distributed graph computation

Let's think about the **graph connectivity** problem:

k sites each holds a portion of a graph.
Goal: compute whether the graph is connected.



A trivial solution:
each S_i sends a local spanning forest to C. Cost $O(kn \log n)$ bits.

n: # nodes of the graph

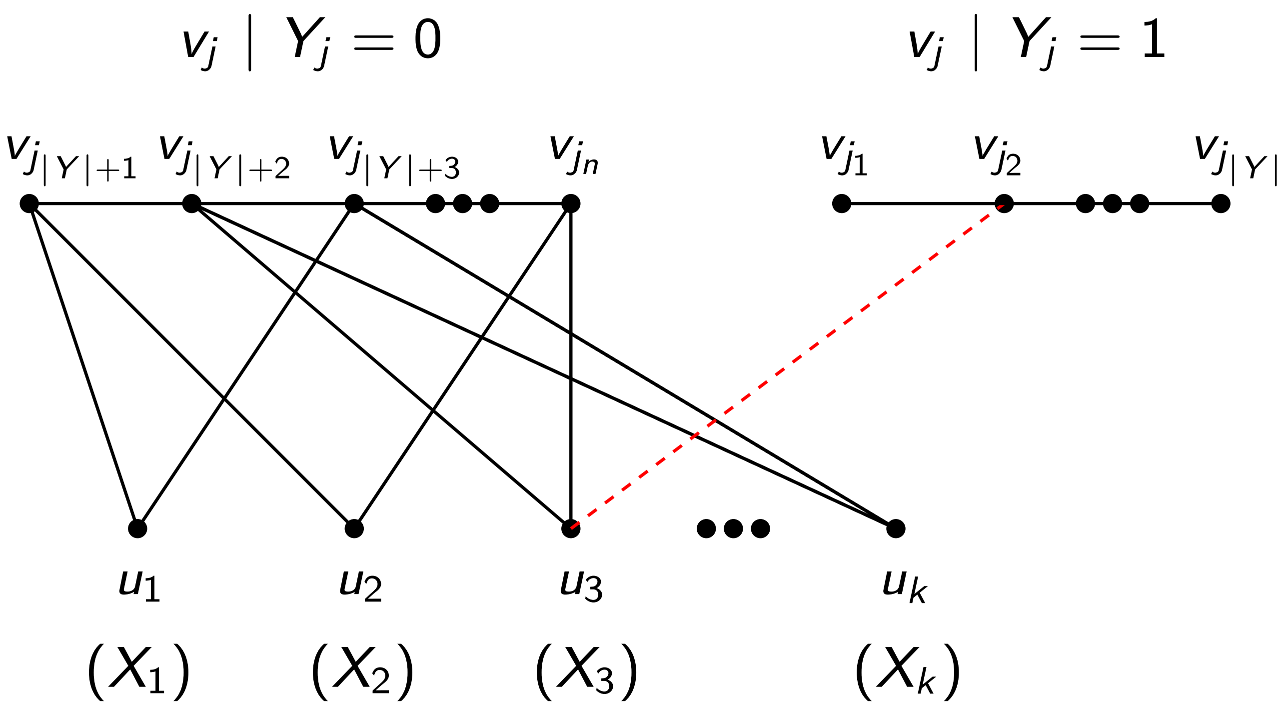
Can we do better, e.g., $o(kn)$ bits of comm. in total?

If graph is edge partitioned among k sites, $\Omega(kn)$

[Woodruff, Z. '13]

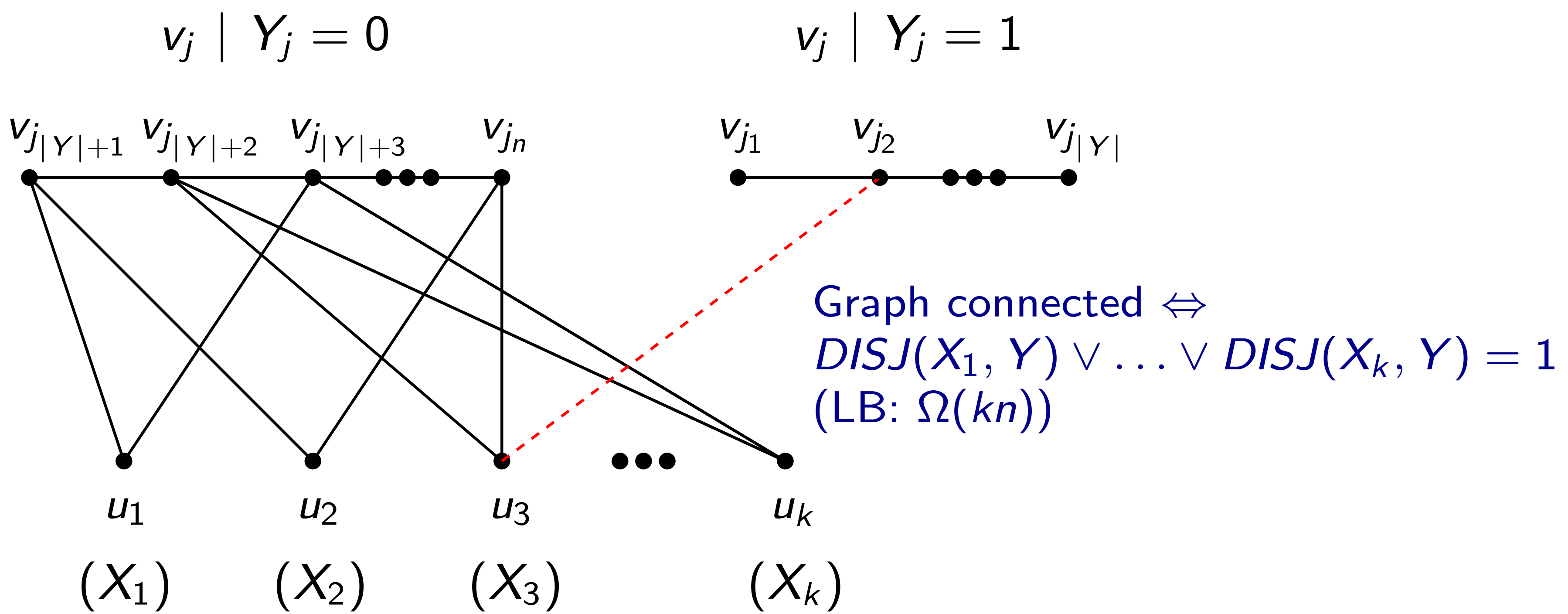
LB graph for edge partition:

For each $i \in [k]$, $(X_i, Y) \sim \mu$ which is a hard input distribution for set-disjointness. Each site S_i holding $X_i = \{X_{i,1}, \ldots, X_{i,n}\}$ creates an edge (u_i, v_j) for each $X_{i,j} = 1$. The coordinator holding $Y = \{Y_1, \ldots, Y_n\}$ creates a path containing $\{v_j \mid Y_j = 1\}$ and a path containing $\{v_j \mid Y_j = 0\}$.



LB graph for edge partition:

For each $i \in [k]$, $(X_i, Y) \sim \mu$ which is a hard input distribution for set-disjointness. Each site S_i holding $X_i = \{X_{i,1}, \ldots, X_{i,n}\}$ creates an edge (u_i, v_j) for each $X_{i,j} = 1$. The coordinator holding $Y = \{Y_1, \ldots, Y_n\}$ creates a path containing $\{v_j \mid Y_j = 1\}$ and a path containing $\{v_j \mid Y_j = 0\}$.



What if the graph is node partitioned?

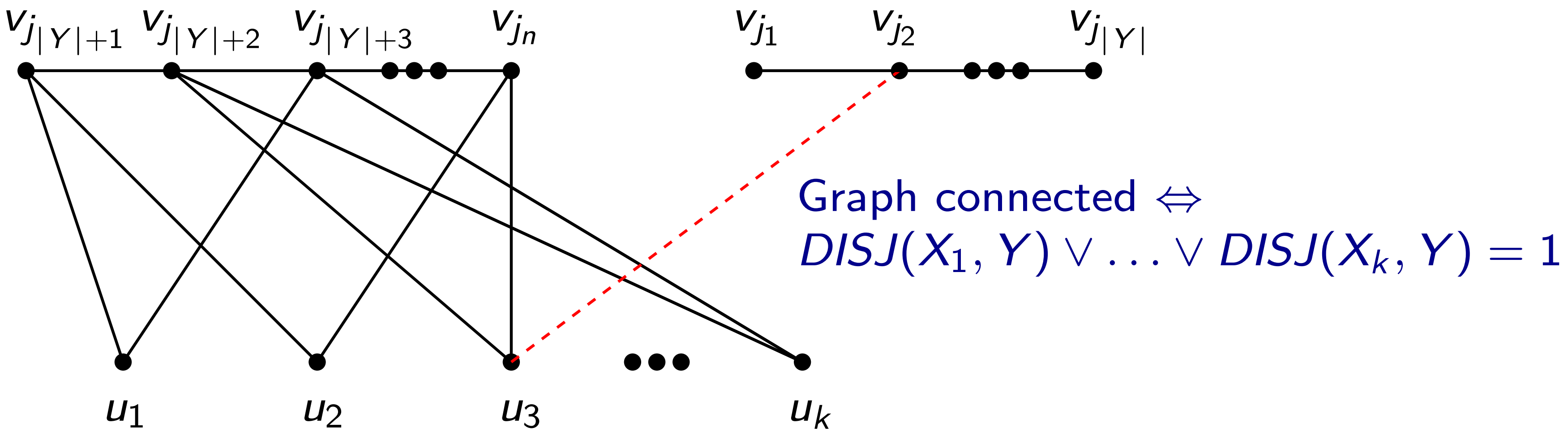
In most practical systems, graph is node partitioned.

Can we prove a similar LB?

What if the graph is node partitioned?

In most practical systems, graph is node partitioned.

Can we prove a similar LB?

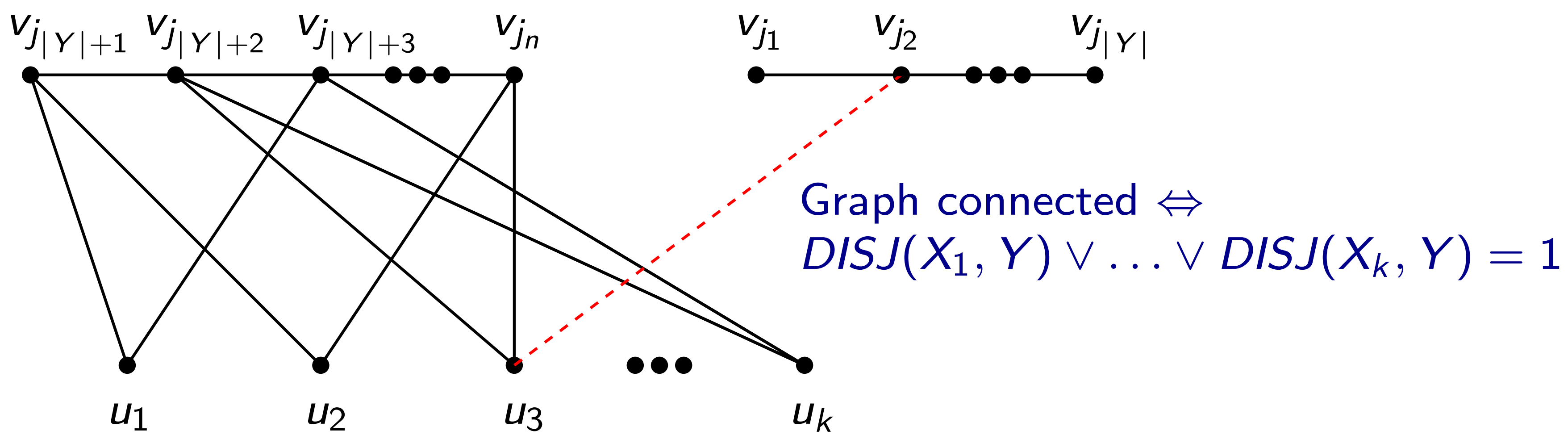


Basically, only bottom nodes (and their adjacent edges) are partitioned

What if the graph is node partitioned?

In most practical systems, graph is node partitioned.

Can we prove a similar LB?



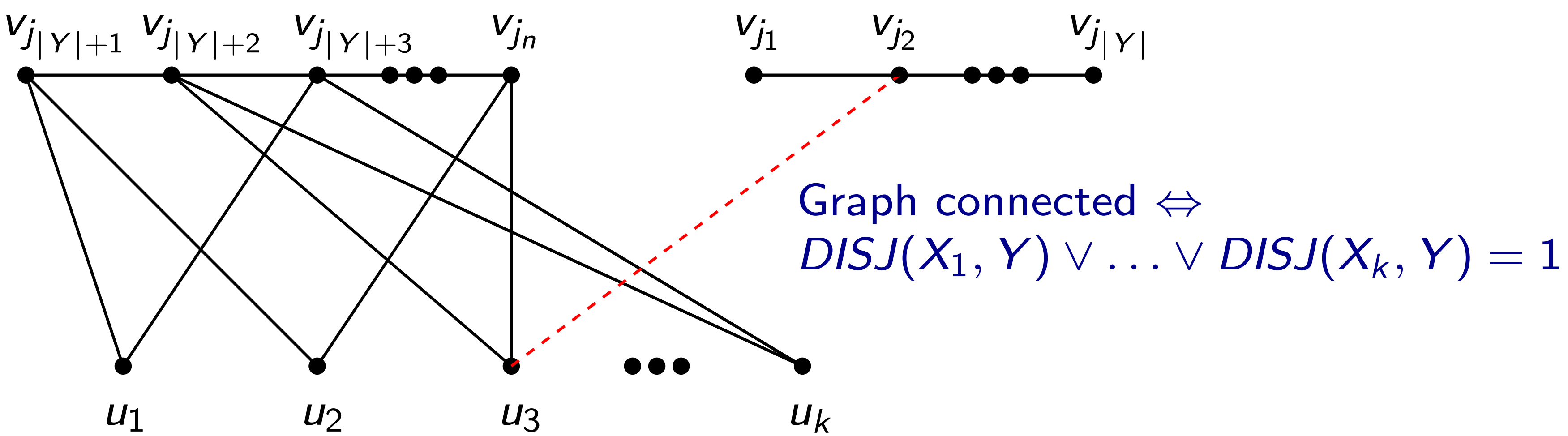
Basically, only bottom nodes (and their adjacent edges) are partitioned

If we also partition the top nodes (and their adjacent edges), then the $\Omega(kn)$ LB does not hold.

What if the graph is node partitioned?

In most practical systems, graph is node partitioned.

Can we prove a similar LB?



Basically, only bottom nodes (and their adjacent edges) are partitioned

If we also partition the top nodes (and their adjacent edges), then the $\Omega(kn)$ LB does not hold.

Not a surprise. If a graph is node partitioned, $\tilde{O}(n)$ suffices.

[Ahn, Guha, McGregor '12]

Input sharing

Input sharing

To prove LB in the node partition model, one needs to deal with input sharing: each edge may be stored in two sites.

Need new techniques?

Input sharing

Input sharing

To prove LB in the node partition model, one needs to deal with input sharing: each edge may be stored in two sites.

Need new techniques?

A concrete problem: **Breadth First Search Tree**

Given a node u, the parties want to jointly compute a BSF tree rooted at u. The coordinator outputs the final BFS tree.

What is the comm. complexity?

Distributed joins

Set-intersection join

$A_1, \ldots, A_m \subseteq [n] = \{1, 2, \ldots, n\}$, and $B_1, \ldots, B_m \subseteq [n]$

$$A = \begin{bmatrix} A_1 \\ \vdots \\ A_m \end{bmatrix}$$

e.g., skills of applicants

$$\begin{bmatrix} B_1 & \cdots & B_m \end{bmatrix} = B$$

e.g., skills required by a job positions

Set-Intersection Join (cardinality version)

$SIJ(A, B) = |\{(i, j) \text{ for which } C_{i,j} > 0, \text{ where } C = A \cdot B\}|$

An important operation in databases

Set-intersection join (cont.)

The problem: estimate $SIJ(A, B)$ up to a $(1 + \epsilon)$ factor.
Useful e.g. in query planning.

Set-intersection join (cont.)

The problem: estimate $SIJ(A, B)$ up to a $(1 + \epsilon)$ factor.
Useful e.g. in query planning.

Current LB $\Omega(n/\epsilon^{2/3})$: (Van Gucht, Williams, Woodruff, Z. '15)

Set-intersection join (cont.)

The problem: estimate $SIJ(A, B)$ up to a $(1 + \epsilon)$ factor.
Useful e.g. in query planning.

Current LB $\Omega(n/\epsilon^{2/3})$: (Van Gucht, Williams, Woodruff, Z. '15)

For each $i \in [m]$, choose $(A_i, B_i) \sim \mu$ where μ is a hard input distribution for set-disjointness.
Define $SUM(A, B) = \sum_{i \in [m]} DISJ(A_i, B_i)$. W.h.p.

$$SIJ(A, B) = SUM(A, B) + m(m - 1).$$

Using basically a direct-sum (Gap-hamming + DISJ), any rand. algo. that computes $SUM(A, B)$ w.pr. 0.99 up to an additive error $\sqrt{m/2}$ needs $\Omega(mn)$ comm.

Set $m = 1/\epsilon^{2/3}$ to get $\Omega(n/\epsilon^{2/3})$ LB

Set-intersection join (cont.)

The current best UB: $\tilde{O}(m/\epsilon^2)$
using F_0-sketch, and is one-way

Can we prove an $\Omega(n/\epsilon^2)$ LB?

Not enough to apply a direct-sum type argument on $(A_1, B_1), \ldots, (A_m, B_m)$, since each A_i is going to join each B_j. In other words,
the primitive problems overlap.

Need new techniques?

Sketching threshold edit distance

Edit Distance

Definition: Given two strings $s, t \in \Sigma^n$:

$ed(s, t) =$ minimum number of character operations (insertion/deletion/substitution) that transform s to t.

Edit Distance

Definition: Given two strings $s, t \in \Sigma^n$:

$ed(s, t)$ = minimum number of character operations (insertion/deletion/substitution) that transform s to t.

ed(banana ,
ananas) = 2

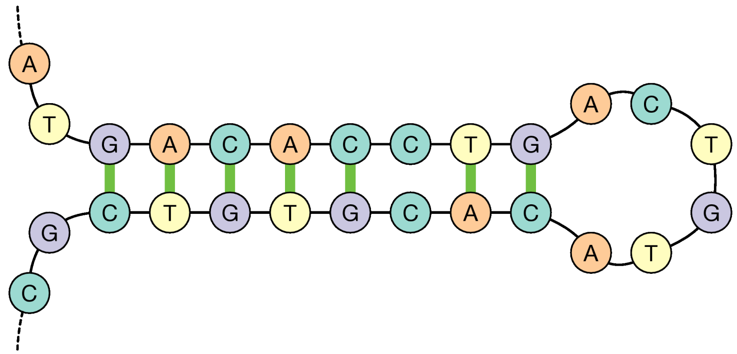
Edit Distance

Definition: Given two strings $s, t \in \Sigma^n$:

$ed(s,t)$ = minimum number of character operations (insertion/deletion/substitution) that transform s to t.

ed(banana ,
ananas) = 2

Applications: numerous. E.g.,



bioinformatics (measuring similarity between DNA seq.

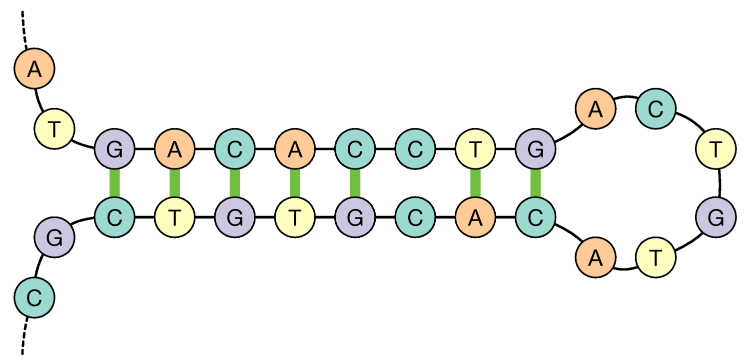
Edit Distance

Definition: Given two strings $s, t \in \Sigma^n$:

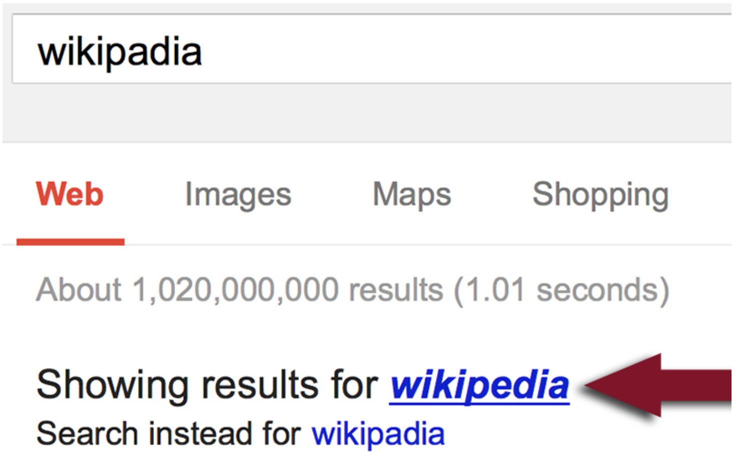
$ed(s,t)$ = minimum number of character operations (insertion/deletion/substitution) that transform s to t.

ed(banana ,
ananas) = 2

Applications: numerous. E.g.,



bioinformatics (measuring similarity between DNA seq.



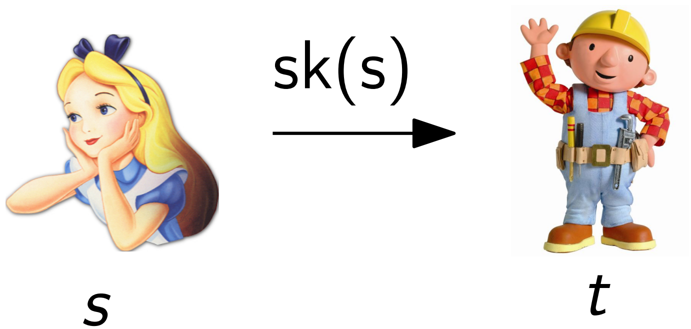
automatic spelling correction

Problems

The threshold version of ED: Given two strings $s, t \in \{0,1\}^n$ and a threhold K, output all the edits if $ed(s,t) \le K$, output "`Error`" otherwise.

Problems

The threshold version of ED: Given two strings $s, t \in \{0,1\}^n$ and a threhold K, output all the edits if $ed(s,t) \le K$, output "`Error`" otherwise.



document exchange

App: remote file sync;

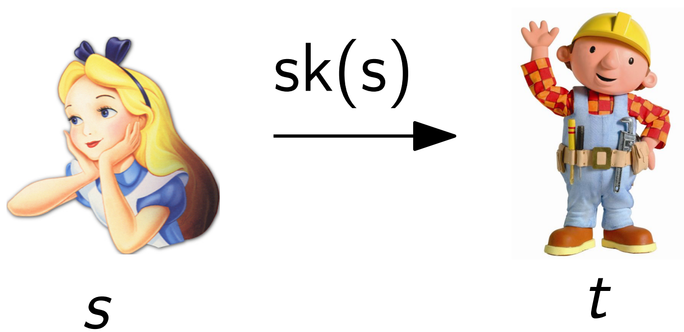
file transmission through

a noisy channel

One-way comm.

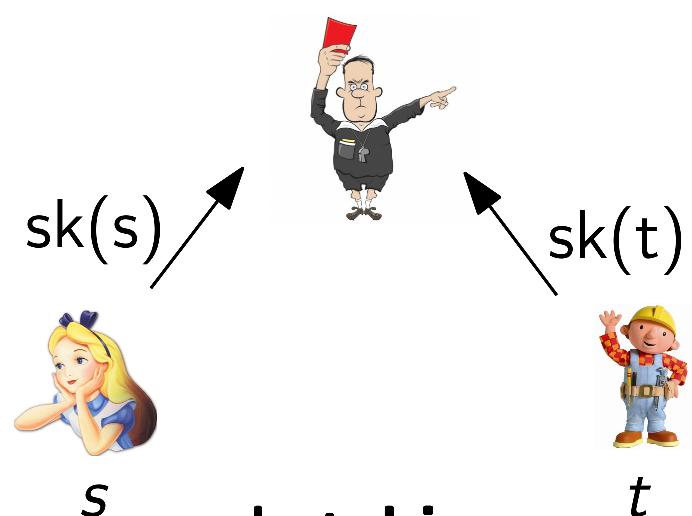
Problems

The threshold version of ED: Given two strings $s, t \in \{0,1\}^n$ and a threhold K, output all the edits if $ed(s,t) \le K$, output "`Error`" otherwise.



document exchange

App: remote file sync; file transmission through a noisy channel



sketching

App: distributed similarity join

One-way comm.

Simultaneous comm.

What we have known

problem	comm. / size / space (bits)	running time	rand. or det.	ref.
document-exchange	$O(K \log n)$	$n^{O(K)}$	D	[23]
	$O(K \log(n/K) \log n)$	$\tilde{O}(n)$	R	[18]
	$O(K \log^2 n \log^* n)$	$\tilde{O}(n)$	R	[19]
	$O(K^2 + K \log^2 n)$	$\tilde{O}(n)$	D	[5]
	$O(K^2 \log n)$	$\tilde{O}(n)$	R	[8]
	$O(K(\log^2 K + \log n))$	$\tilde{O}(n)$	R	**new**
sketching	$O(K^8 \log^5 n)$	$\tilde{O}(K^2 n)$ (enc.), poly$(K \log n)$ (dec.)	R	**new**
streaming	$O(K^8 \log^5 n)$	$\tilde{O}(K^2 n)$	R	**new**
simultaneous-streaming	$O(K^6 \log n)$	$\tilde{O}(n)$	R	[8]
	$O(K \log n)$	$O(n)$	D	**new**

New: results from [Belazzougui, Z. '16]. For simplicity, assuming $K < n^{0.1}$

The one-way CC of K-threshold ED is $\Theta(K \log n)$.

The simultaneous CC of K-threshold ED is $O(K^8 \log^5 n)$.
Should be able to improve it to $K^4 \cdot \text{poly}\log(n)$ or $K^3 \cdot \text{poly}\log(n)$.
But I am not sure if we can do it in $o(K^2) \cdot \text{poly}\log(n)$. **LB**?

A possible hard distribution

Conjecture: the following may be a hard distribution for K-threshold ED, i.e., any algo needs $\Omega(K^2)$ comm.

A possible hard distribution

Conjecture: the following may be a hard distribution for K-threshold ED, i.e., any algo needs $\Omega(K^2)$ comm.

W.pr. $1/2$, the K edits are randomly located in s and t;

W.pr. $1/2$, the K edits are located in a random group of adjacent positions.

The general question

Can we prove higher LB in the simultaneous comm. model than in the one-way comm. model for natural problems?

If you know any example/result, please let me know. Thanks.

Thank you!
Questions?