Distributed Loopy Belief Propagation on Real-World Graphs

Saehan Jo
Seoul National Univ.
naheas@snu.ac.kr

Jaemin Yoo
Seoul National Univ.
jaeminyoo@snu.ac.kr

U Kang
Seoul National Univ.
ukang@snu.ac.kr

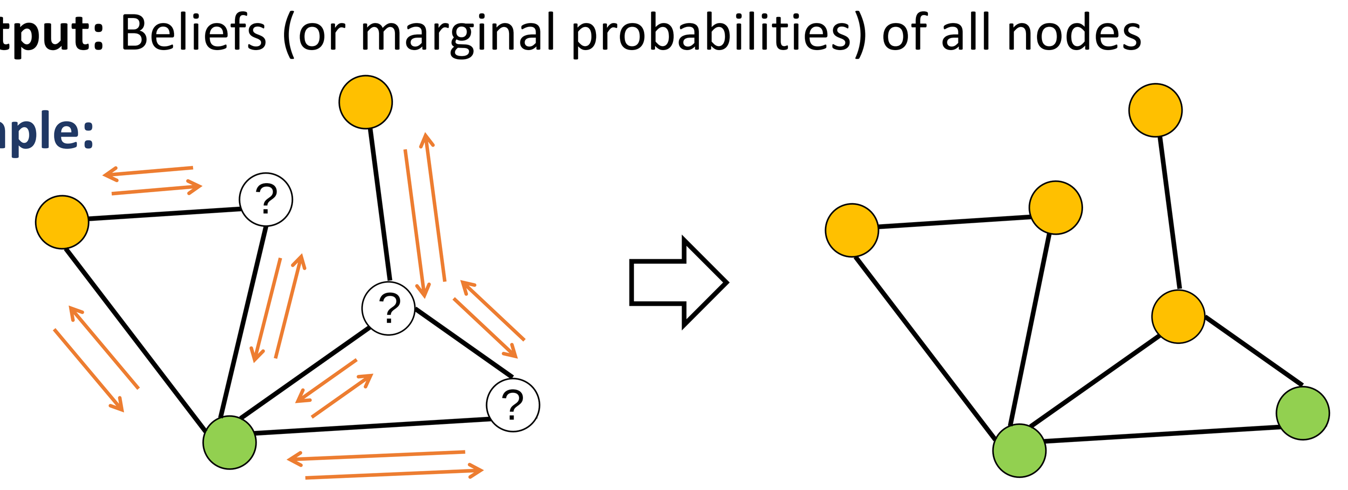
Summary

- **Problem:** Given large real-world graphs that do not fit in a single machine, how can we efficiently make inference on unobserved vertices?
- **Base algorithm:** Loopy Belief Propagation (LBP)
- **Challenges:**
  - Power-law degree distribution of real-world graphs
  - Burdensome iterative computations
  - High communication overhead
- **Our Method:** Distributed Loopy Belief Propagation (DLBP)
  - Utilize correct convergence criterion for real-world graphs
  - Carefully schedule the iterations to minimize data communication
- **Experimental Results:**
  - Mostly identical accuracies (less than 0.14% difference)
  - Up to 10.7x faster than standard distributed LBP
- **Codes and datasets:** *https://datalab.snu.ac.kr/dlbp*

Base Algorithm

Loopy Belief Propagation

- **Problem:** Inference on probabilistic graphical models
- **Algorithm:** Propagates information by iterative message passing
  - **Input:** Priors and edge potentials
  - **Output:** Beliefs (or marginal probabilities) of all nodes
- **Example:**



Our Contributions

- **Algorithm: DLBP** is a novel algorithm for LBP on a distributed environment, which solves the challenges associated with the power-law degree distribution of real-world graphs
- **Analysis:**
  - We provide a theoretical analysis of two different convergence criteria of LBP (message and belief)
  - We analyze **DLBP** in terms of time complexity, space complexity, and the amount of shuffled data
- **Experiment:**
  - **DLBP** demonstrates up to 10.7x speed up on real-world graphs compared to standard distributed LBP
  - **DLBP** shows near-linear scalability with the number of machines

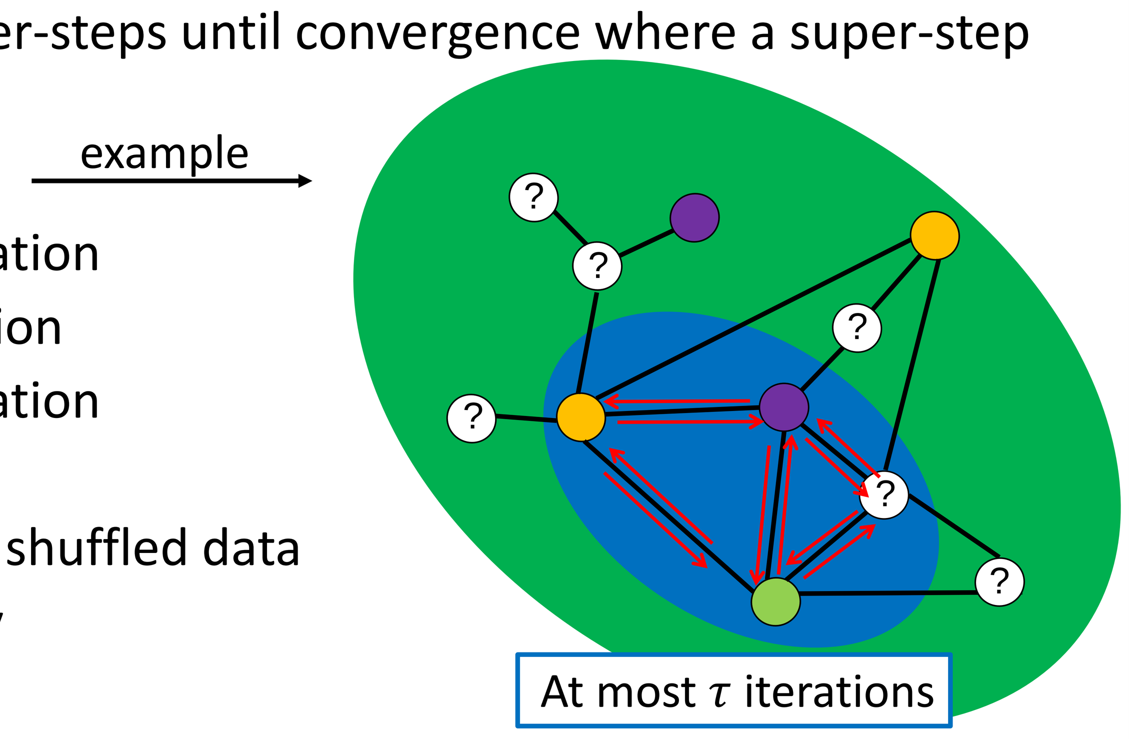
Motivation

- **Question 1: Setting convergence criterion**
  - Is the convergence criterion based on messages appropriate for real-world graphs with high-degree vertices?
- **Question 2: Minimizing numerical computations**
  - How can we minimize the overall computations of messages and the total number of iterations until convergence?
- **Question 3: Minimizing network communication cost**
  - Real-world graphs are known to have highly-skewed degree distribution
  - With this property in mind, how can we partition a real-world graph and arrange the message computations to minimize the data communication between machines?

Our Method

Distributed Loopy Belief Propagation (DLBP)

- **Idea 1: Using "belief" as the convergence criterion**
  - **Message convergence criterion does not guarantee** on high-degree vertices (Lemma 1, 2):
    - the convergence of beliefs
    - the convergence of messages in the next iteration
  - **Belief convergence criterion** guarantees regardless of the degree of a vertex (Lemma 3, 4):
    - the convergence of beliefs
    - the convergence of messages between two converged vertices
- **Idea 2: Skipping of converged vertices**
  - Belief convergence criterion provides better reliability
  - **Advantages:**
    - Omits redundant computations
    - Reduces the number of iterations until convergence
- **Idea 3: Hub-Oriented Scheduling**
  - **Objective:** Minimize the full data shuffling by reducing the number of super-steps until convergence
  - **Preprocessing Stage:** Divide vertices into hubs (high-degree) and spokes (low-degree) using hub ratio k
  - **Main Stage:** Iterate super-steps until convergence where a super-step consists of
    1. Hub-to-Hub Iteration
    2. Hub-to-Spoke Propagation
    3. Spoke-to-Spoke Iteration
    4. Spoke-to-Hub Propagation
  - **Advantages:**
    - Reduce the amount of shuffled data
    - Lower time complexity
    - Lower memory usages

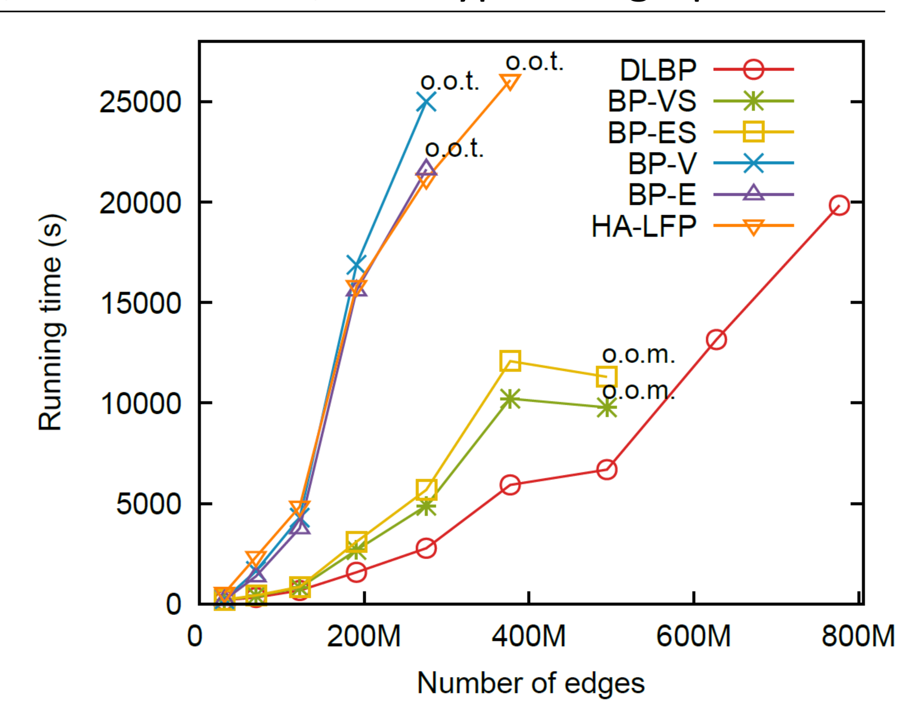


Experiments

- **Real-world datasets**

Name	# of nodes	# of edges	Description
YahooWeb graphs	16,402,838 ~174,577,088	30,403,395 ~776,375,840	Hyperlink graph, Principle submatrices
Campaigns graph	23,191	877,729	Donation graph
PubMed	19,717	88,651	Citation graph
PolBlogs	1,224	16,716	Hyperlink graph

- **Speed**
  - Up to 10.7x faster than standard distributed LBP (BP-E, BP-V)
  - Up to 10.0x faster than distributed LBP on Hadoop (HA-LFP)



- **Label classification accuracy**
  - Less than 0.14% accuracy difference of any two methods

Dataset	BP-E	BP-V	BP-ES	BP-VS	DLBP
Campaigns	89.36%	89.36%	89.31%	89.31%	89.31%
PolBlogs	95.62%	95.62%	95.62%	95.62%	95.62%
PubMed	82.56%	82.56%	82.79%	82.79%	82.56%

- **Machine scalability**
  - Shows near-scalability where DLBP gains 9.46x speed up as the number of machines increases from 1 to 16