Asymmetry in Large-Scale Graph Analysis, Explained

Vasiliki Kalavri[1], Stephan Ewen[2], Kostas Tzoumas[3],
Vladimir Vlassov[4], Volker Markl[5], Seif Haridi[6]

[1, 4, 6] KTH Royal Institute of Technology, [2, 3, 5] Technical University of Berlin
[1, 4, 6]{kalavri, vladv, haridi}@kth.se
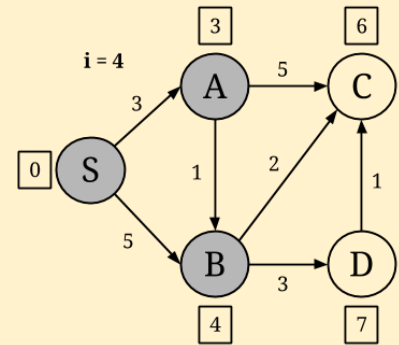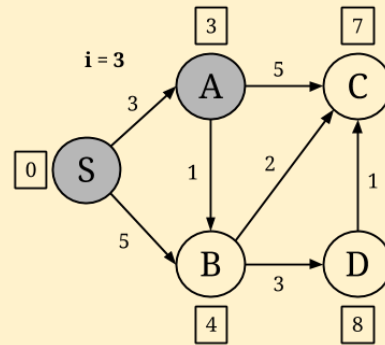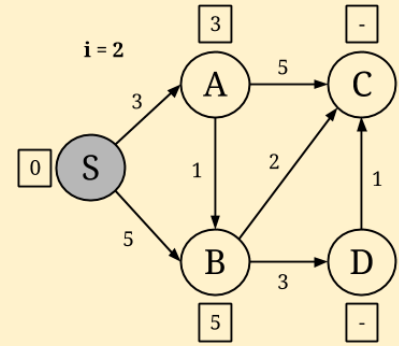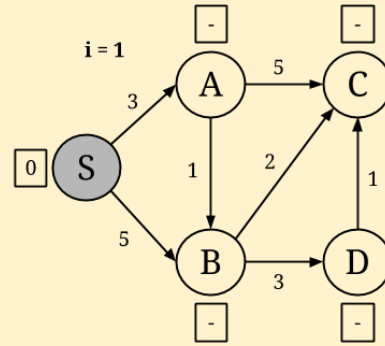[2, 3, 5]{firstname.lastname}@tu-berlin.de

Motivation

- Many of large-scale data processing applications include *fixed point* iterations
  - social network analysis
  - web graph analysis
  - machine learning

Asymmetrical Convergence

- Often, in fixed point iterations, some elements converge faster than others
- Not all elements require an update in every iteration

Can we detect the elements that require recomputation and avoid redundant computations?

Contributions

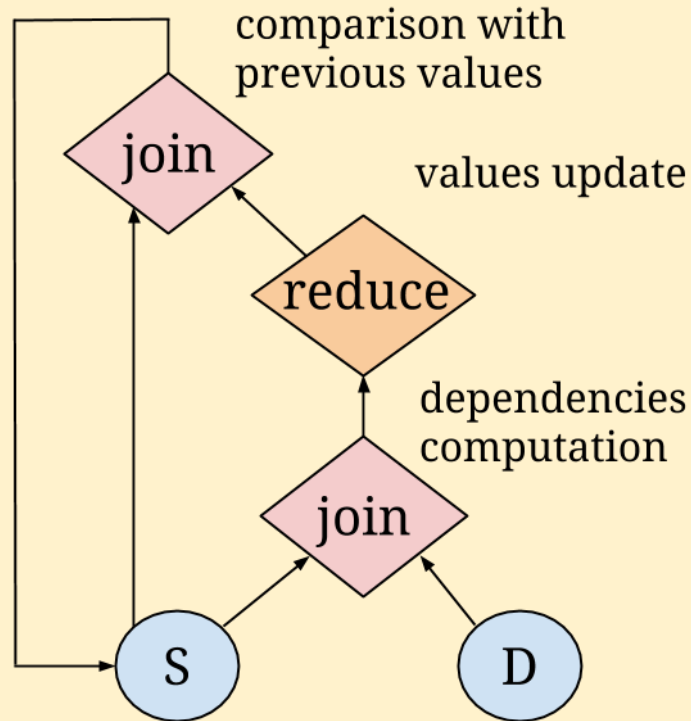- A categorization of optimizations for fixed point iterative graph processing
- Necessary conditions under which, it is safe to apply optimizations
- A mapping of existing techniques to graph processing abstractions
- An implementation of template execution plans

Optimized algorithms yield *order of magnitude* gains!
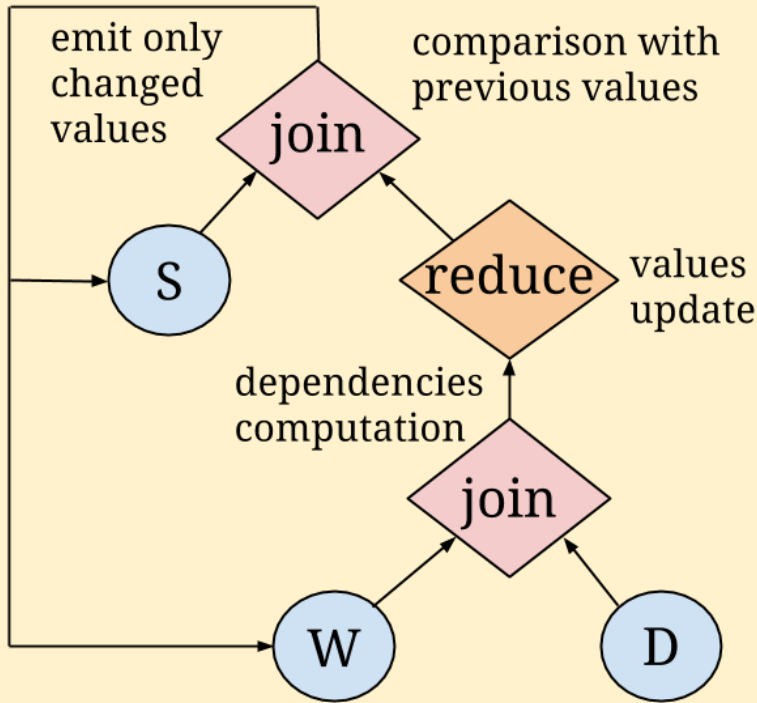
Asymmetry in Connected Components

Updated Elements

Connected Components - Livejournal
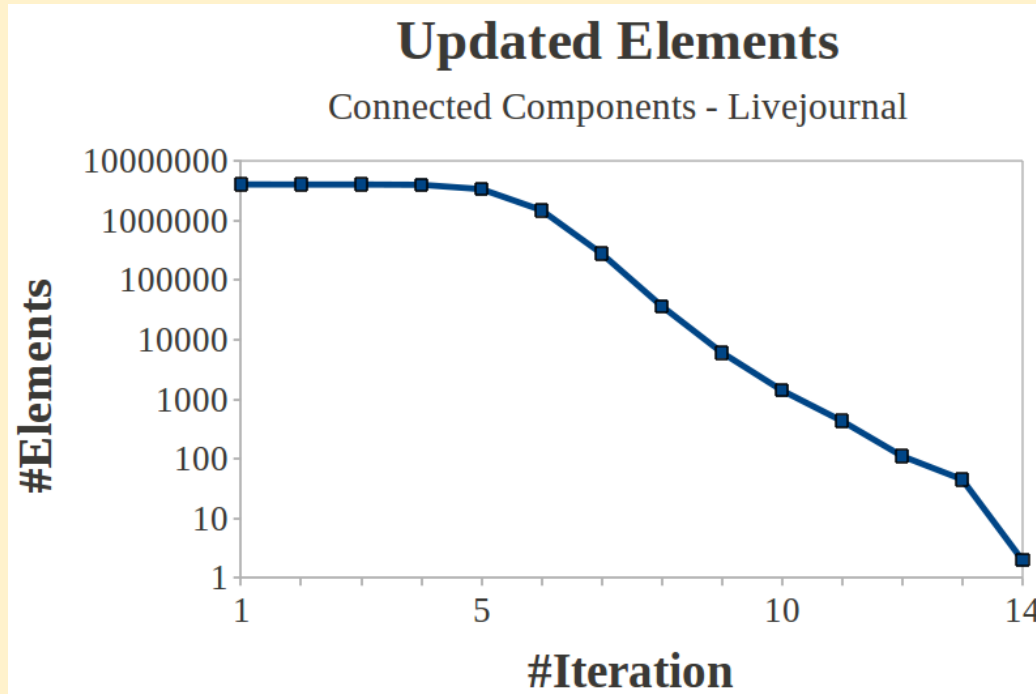
Iterative Plans - Bulk

- In each iteration, **all elements** are computed
- Always applicable

Iterative Plans - Dependency

bulk plan

join

reduce

join — candidates' dependencies

join

S

remove duplicates — reduce

candidates for recomputation — join

W   D

- In each iteration, only elements whose **at least one neighbor has changed** are computed
- The state is computed using the values of **all neighbors**
- Always applicable

Iterative Plans - Incremental

- In each iteration, only elements whose **at least one neighbor has changed** are computed
- The state is computed using only the **values of updated neighbors**
- Applicable when the update function is **idempotent and weakly monotonic** (e.g. min)

Iterative Plans - Delta
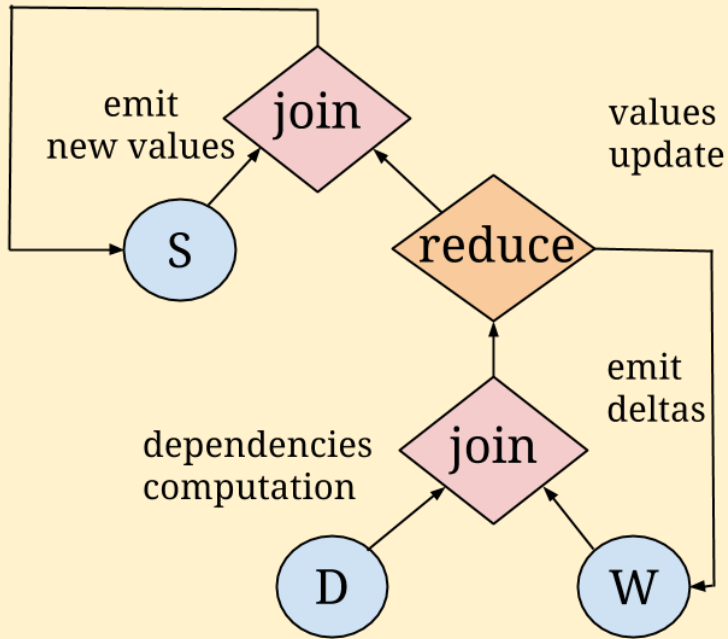
- In each iteration, only elements whose **at least one neighbor has changed** are computed
- The state is computed using only the *deltas* **of updated neighbors**
- Applicable when the update function is **linear over the composition operator** (e.g. sum)

Iteration Techniques Support in Graph Processing Systems

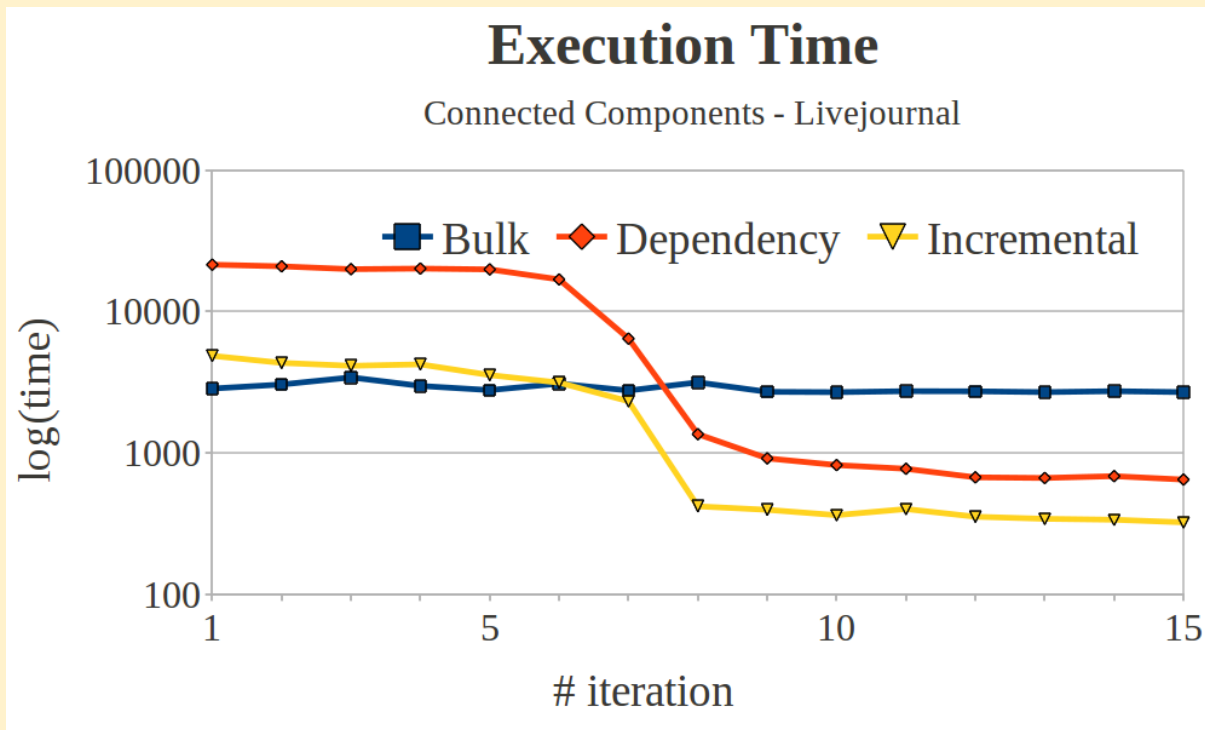| System	Bulk	Dependency	Incremental	Delta
Pregel	X	X	X	X
GraphLab	X	X	X	X
GraphX	X	X	X	X
Powergraph	X	X	X	X
Stratosphere	X	X	X	X

X : provided by default
X : can be easily implemented
X : possible, but non-intuitive

Performance - Connected Components

Execution Time

Connected Components - Livejournal

Performance - PageRank

Execution Time

PageRank - Livejournal
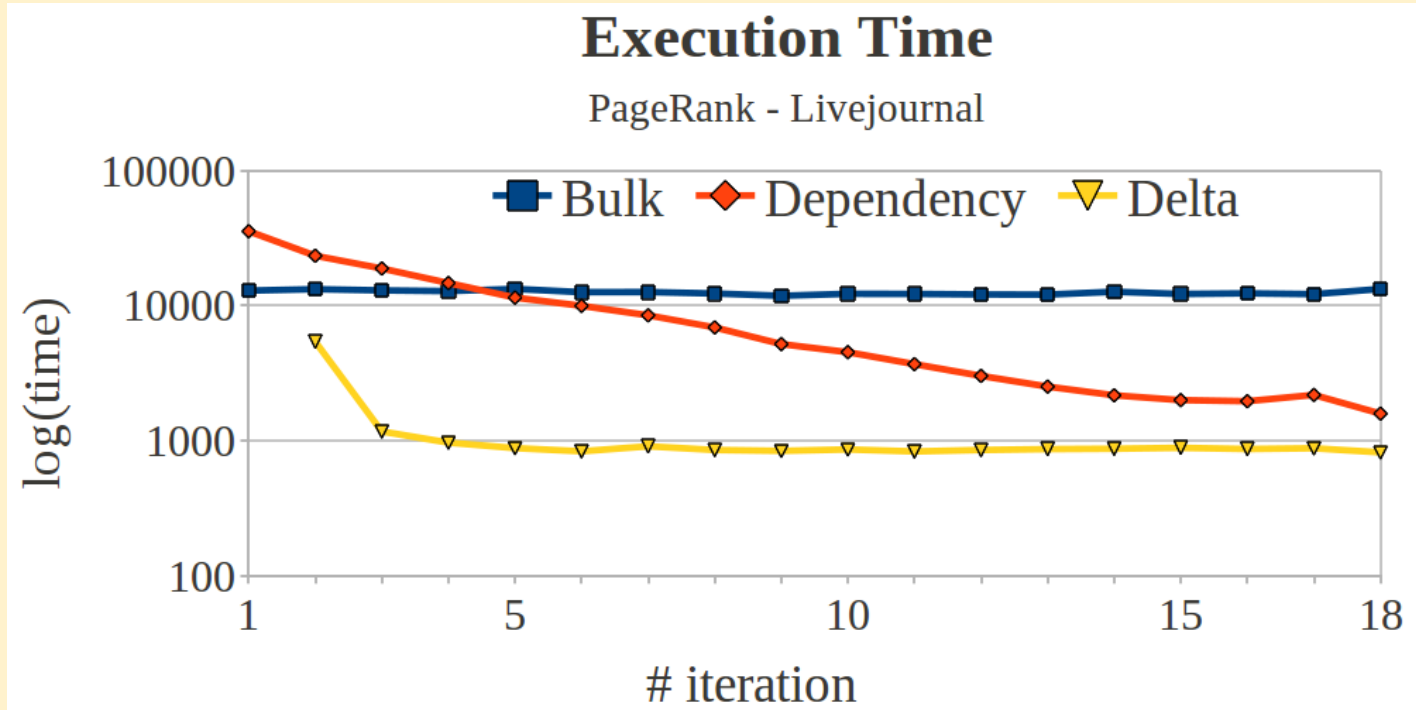
Conclusions & Future Work

- Exploiting asymmetrical convergence can lead to order of magnitude performance gains
- In the future, we plan to
  - Use cost-based optimization, to automatically select the most efficient iterative plan, at runtime.
  - Implement a set of representative applications and compare performance with iterative and graph-processing systems.