SFU

Dependency-Driven Disk-based Graph Processing

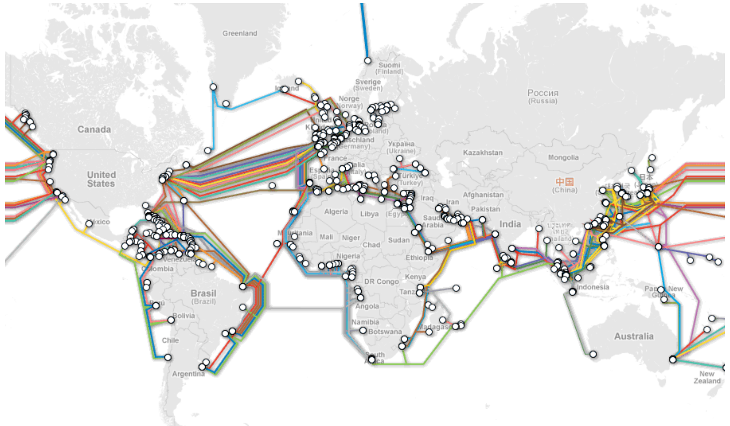
Keval Vora
Simon Fraser University

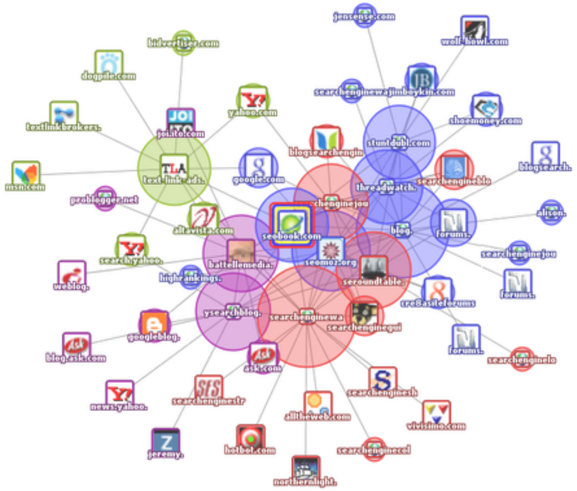
USENIX ATC'19 - Renton, WA

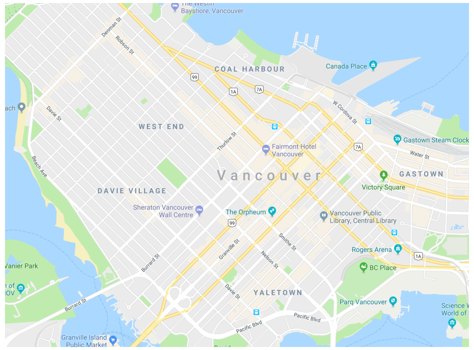
July 11, 2019

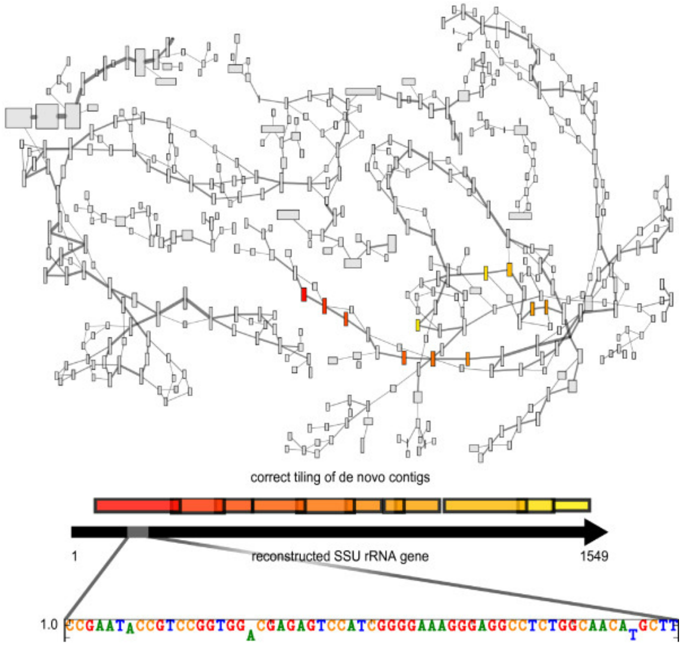
https://github.com/pdclab/lumos

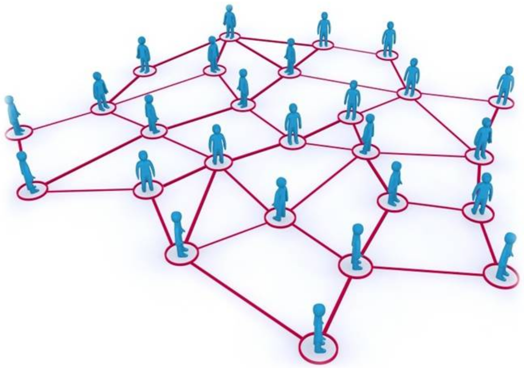
Graph Processing

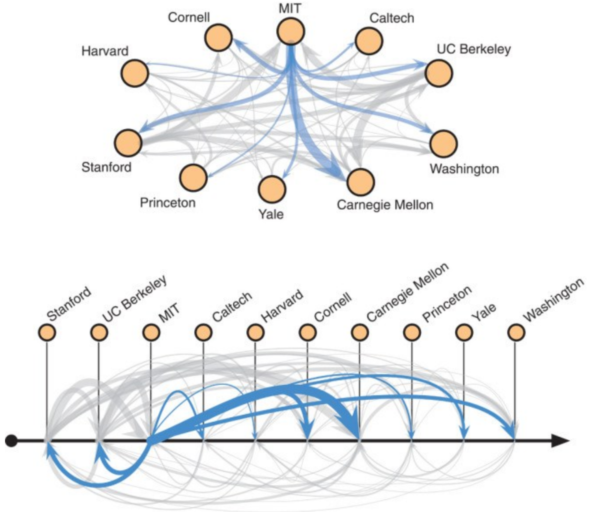












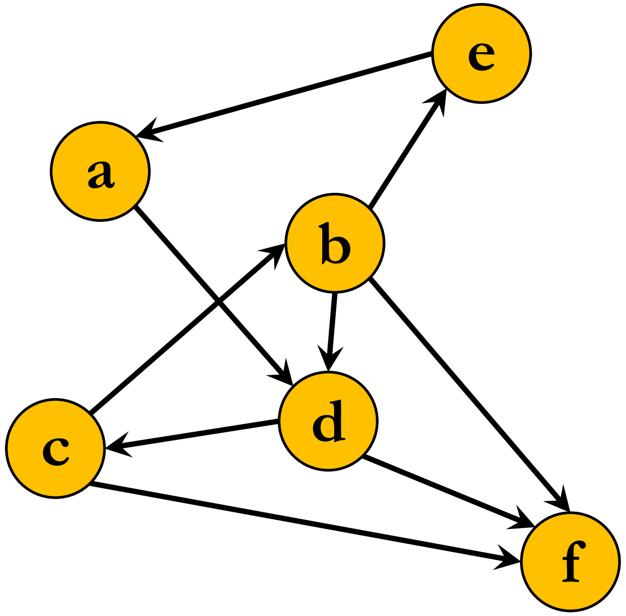
Iterative Graph Processing

PageRank computation

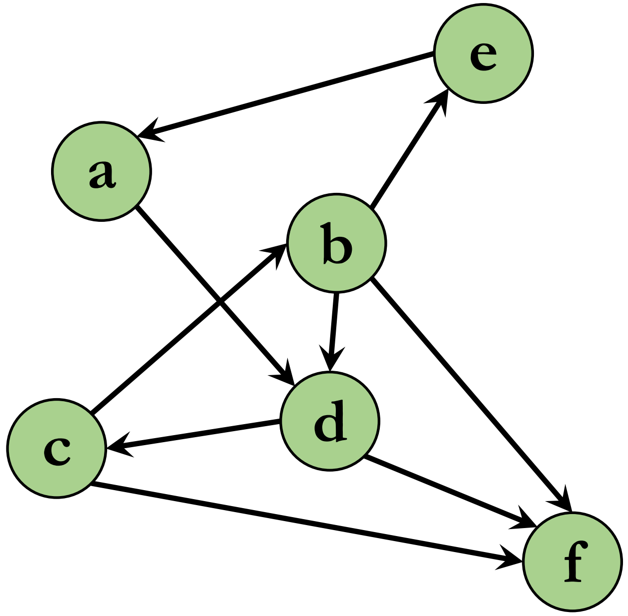
$$\mathrm{PR}(v^t) = 1 - d \; + \; d \times \sum_{\mathrm{inedges}(v)} \mathrm{PR}(u^{t-1}) \;/\; |\mathrm{outedges}(u)|$$

Iterative Graph Processing

$$\mathrm{PR}(v^t) = 1 - d \;+\; d \times \sum_{\mathrm{inedges}(v)} \mathrm{PR}(u^{t-1}) \;/\; |\mathrm{outedges}(u)|$$

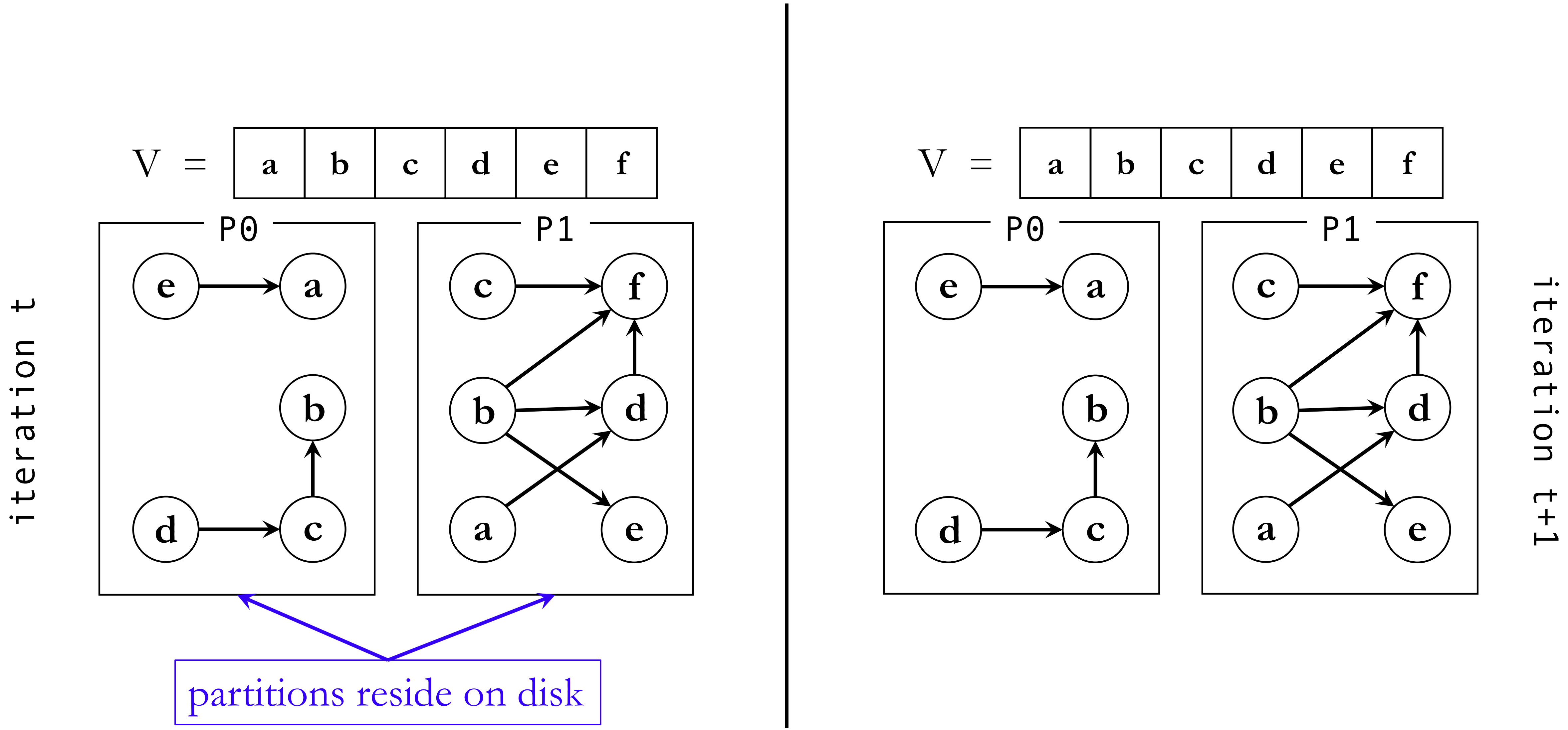


iteration t

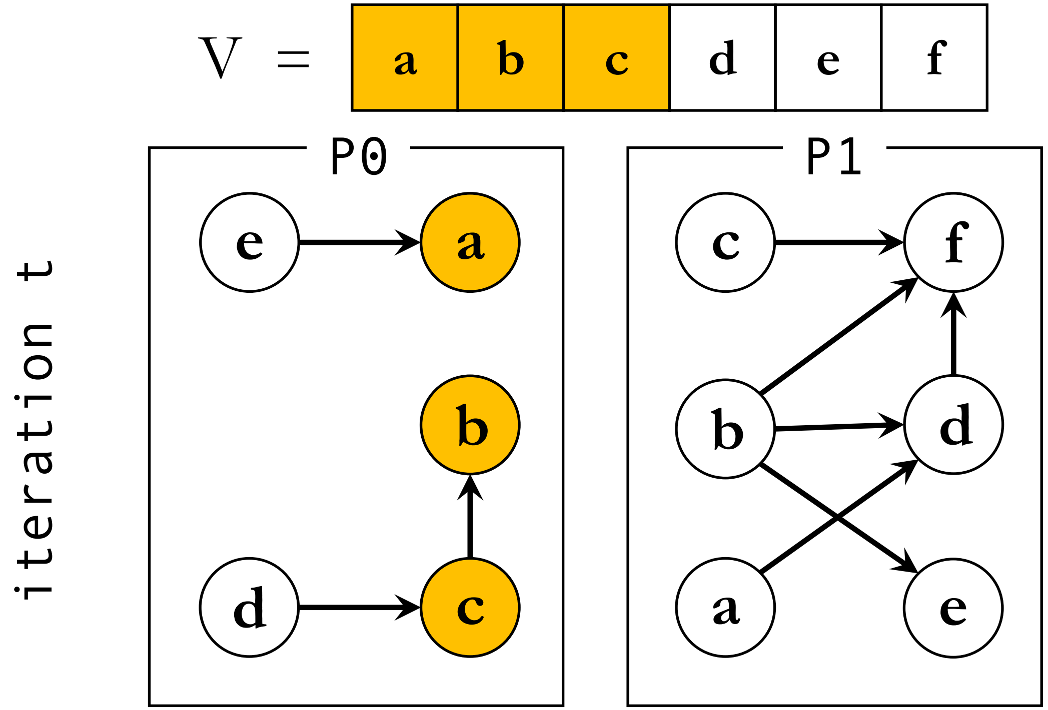


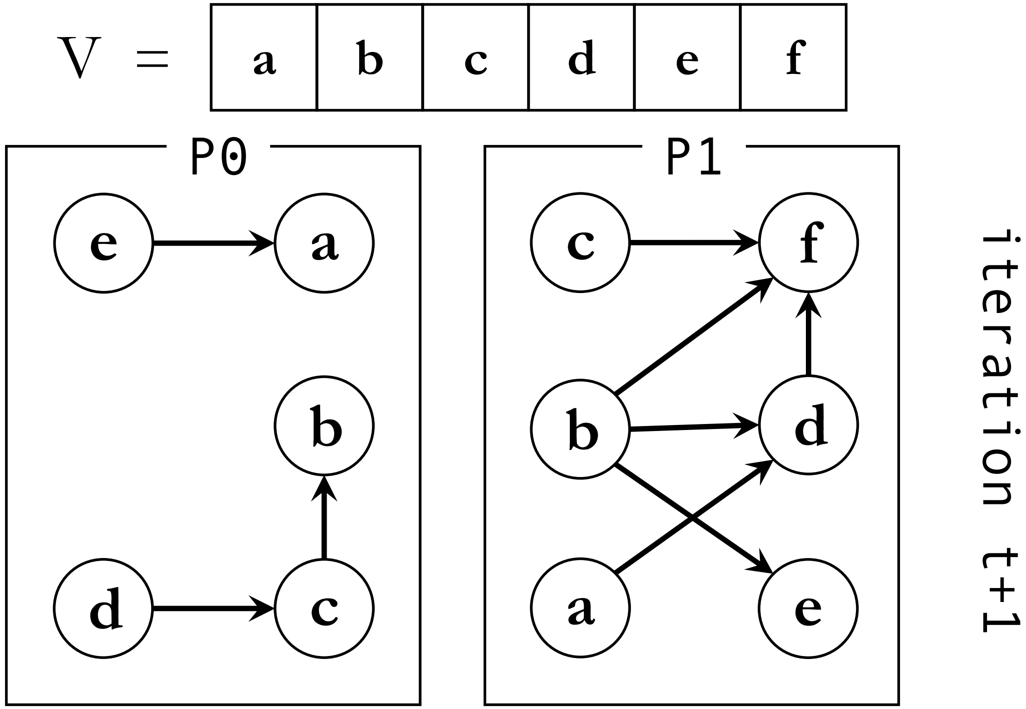
iteration t+1

Out-of-Core Graph Processing

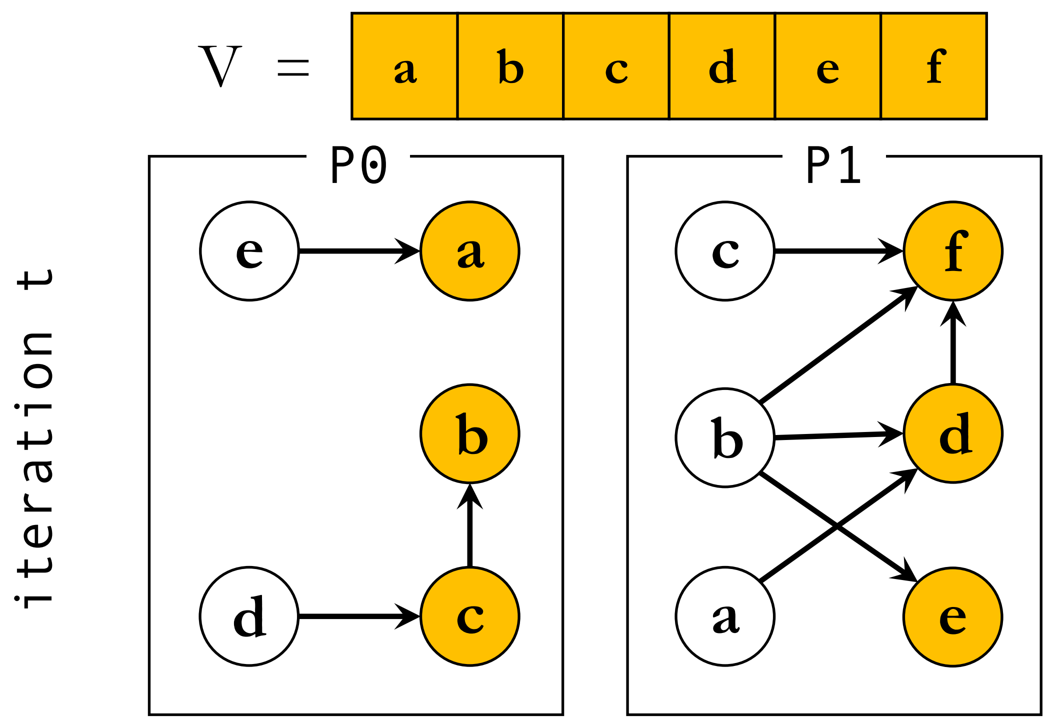


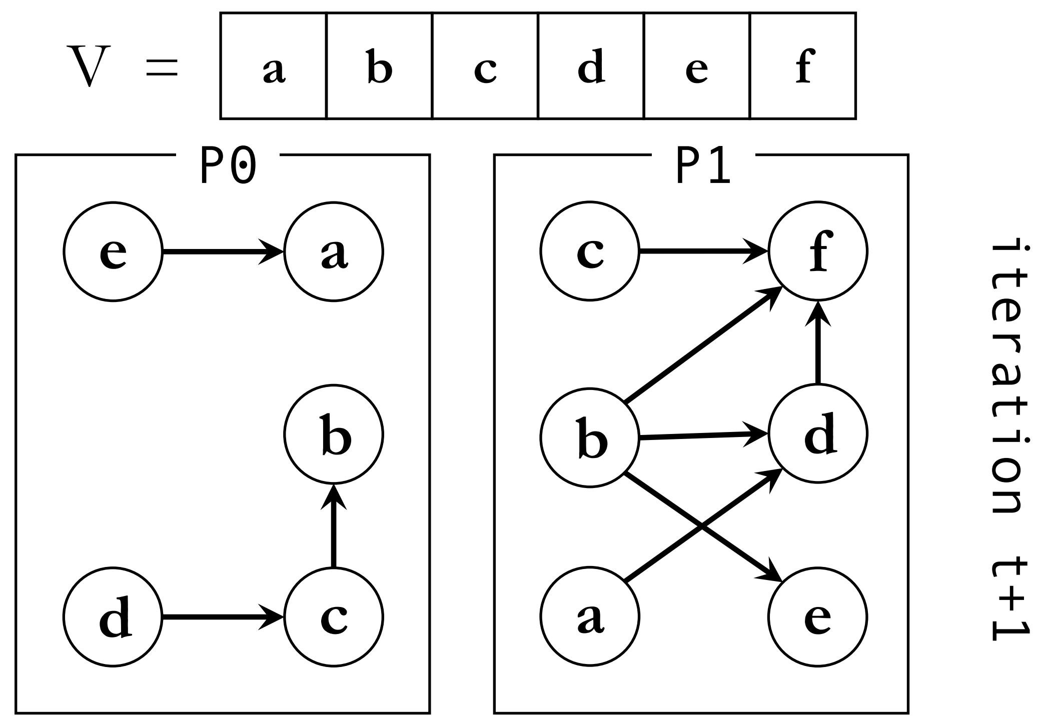
Out-of-Core Graph Processing



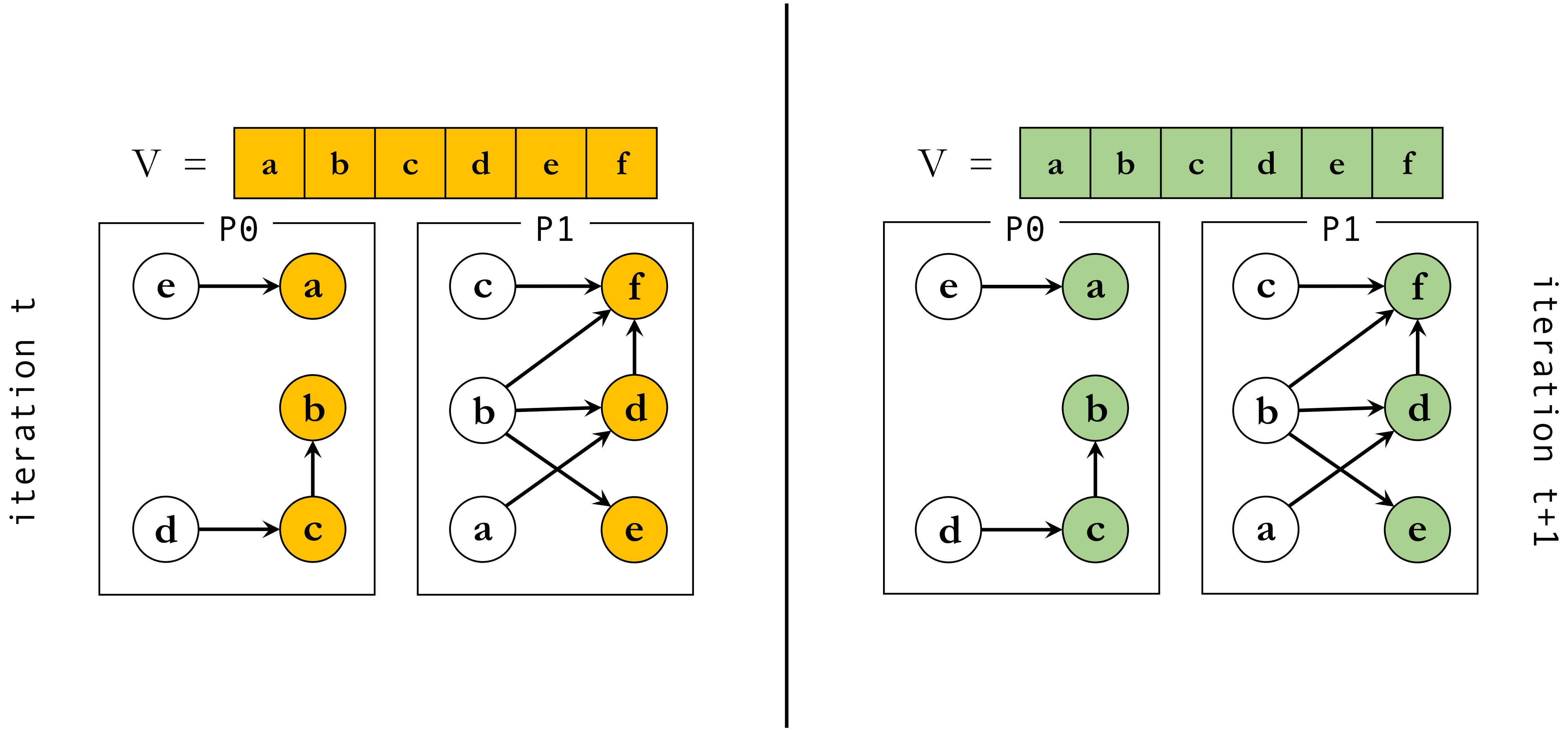


Out-of-Core Graph Processing

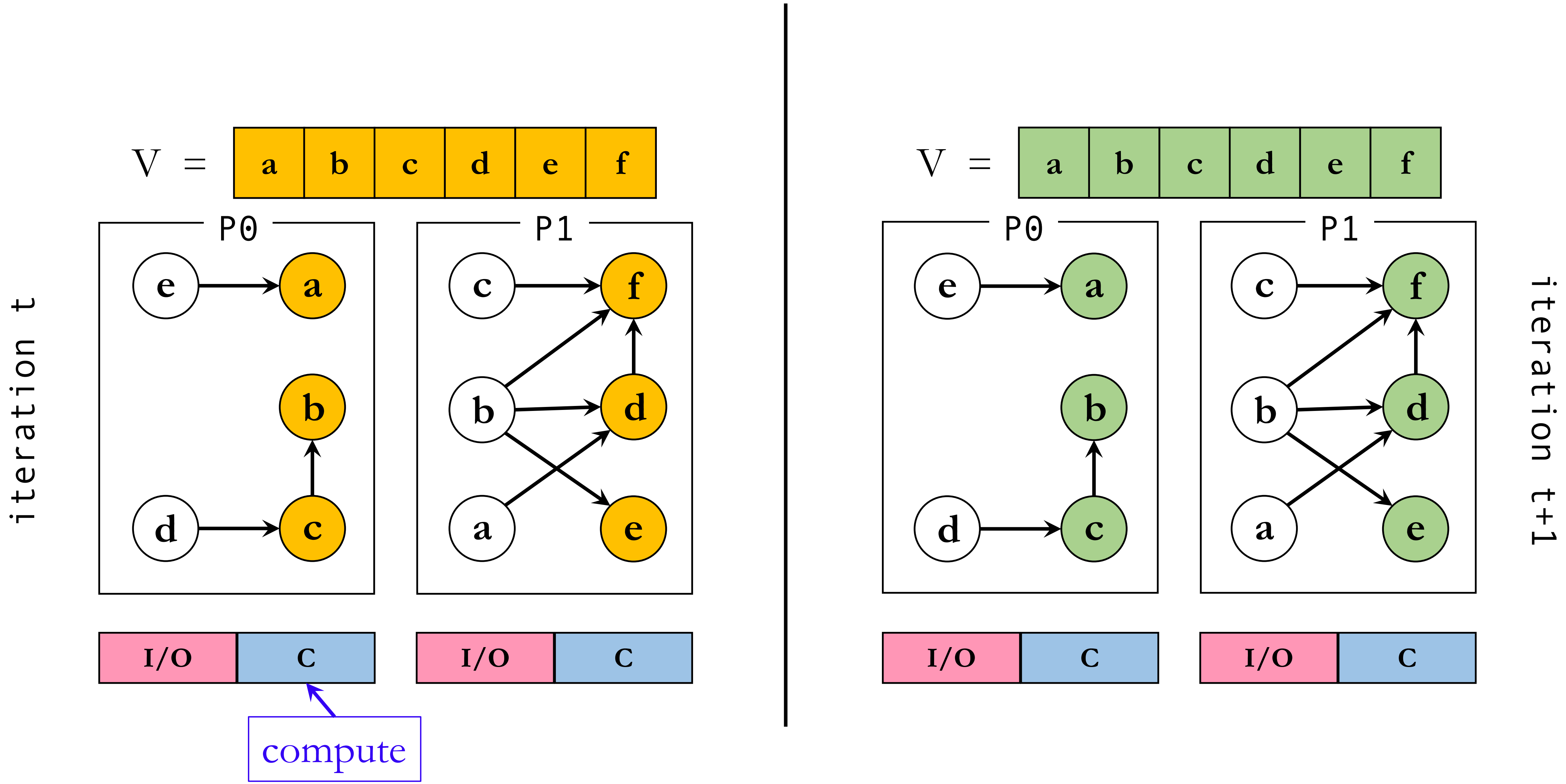




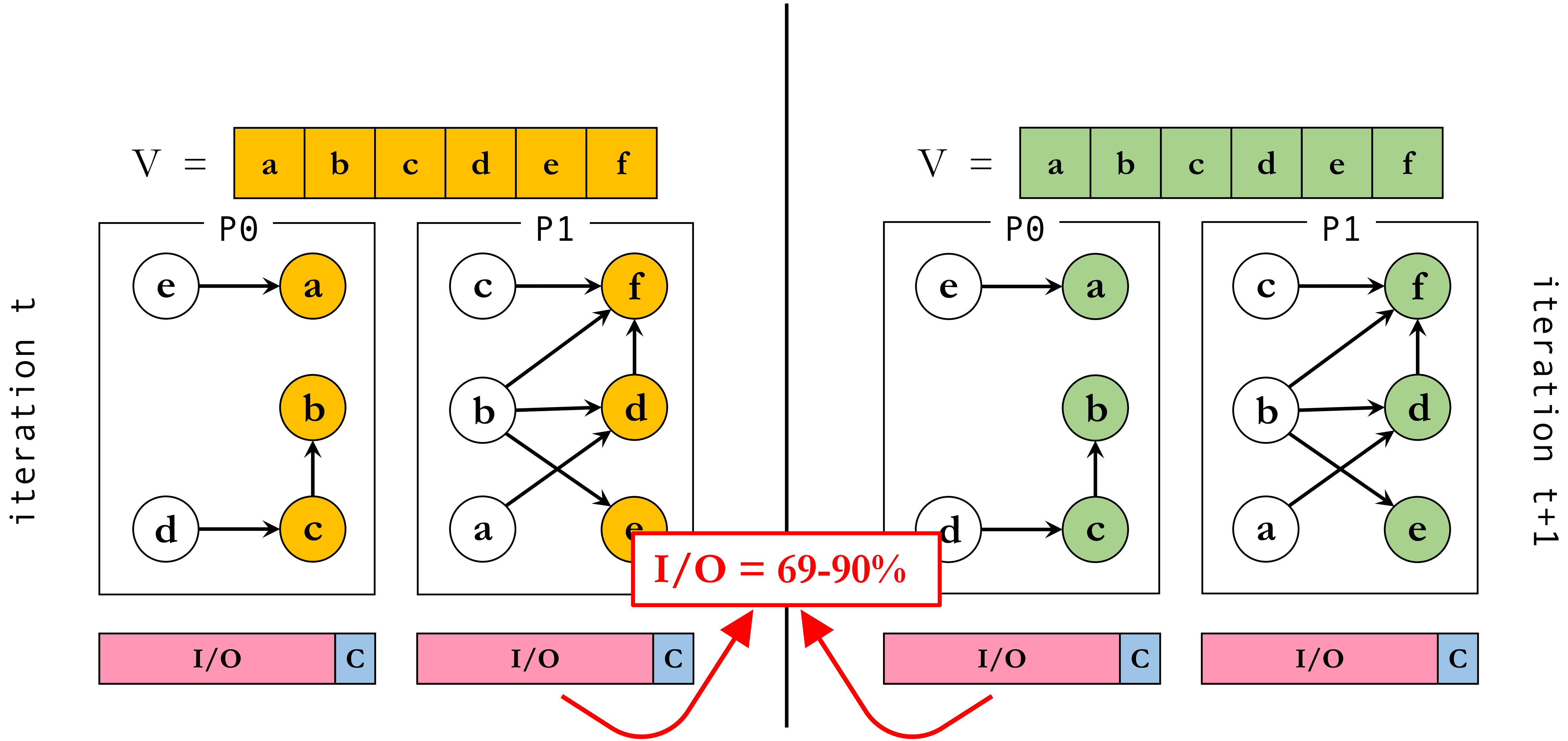
Out-of-Core Graph Processing



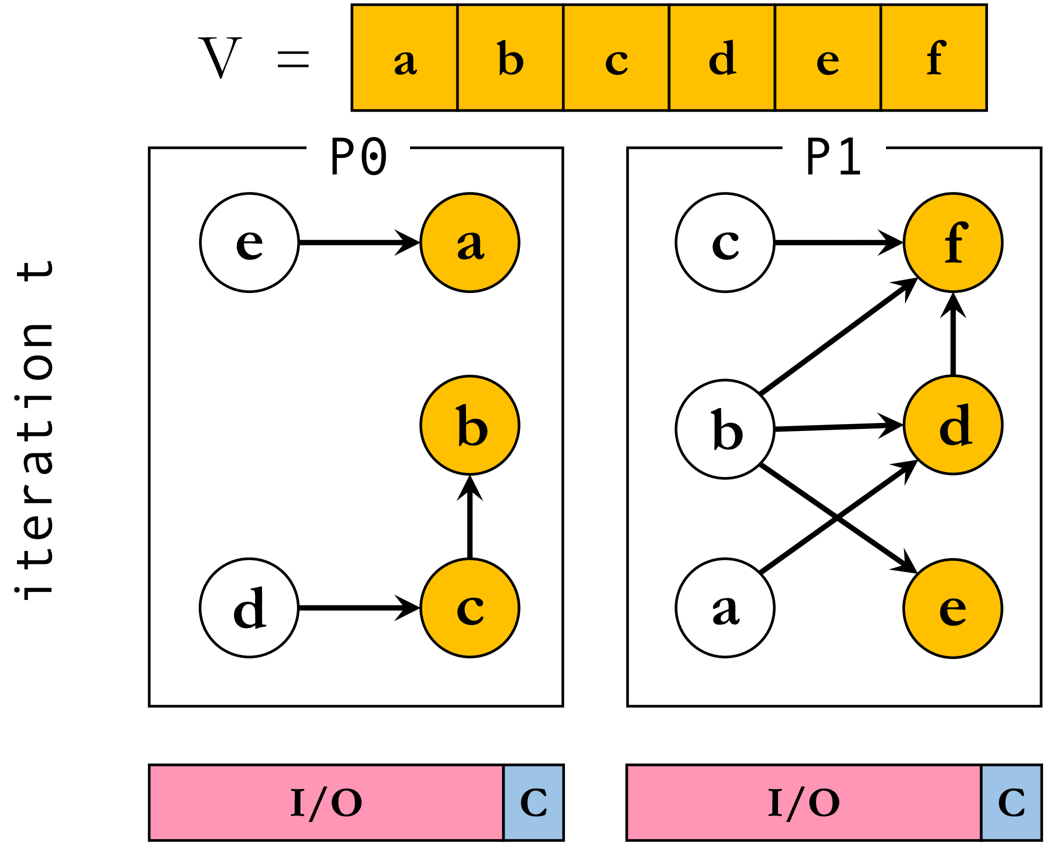
Out-of-Core Graph Processing



Out-of-Core Graph Processing

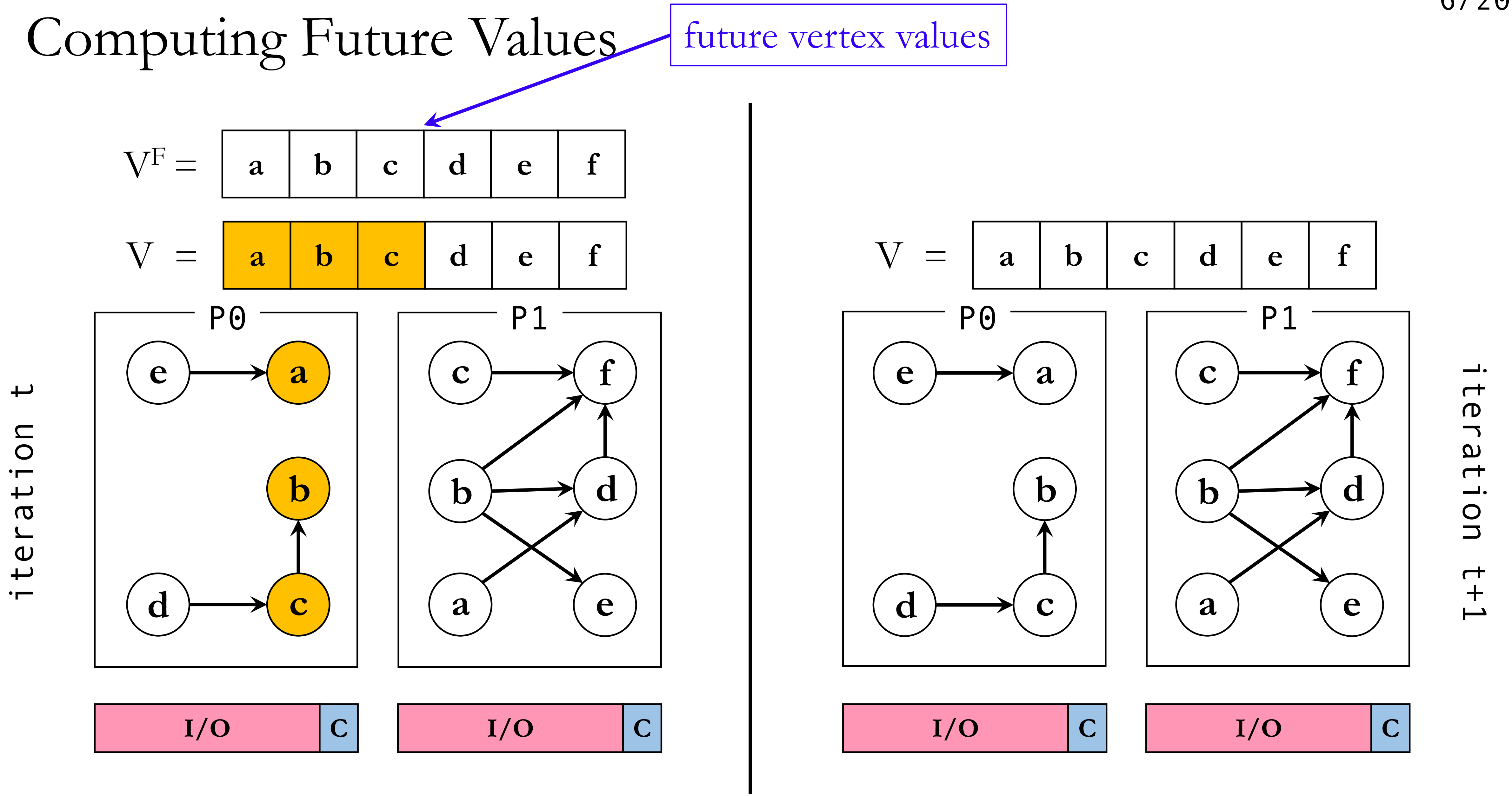


Lumos

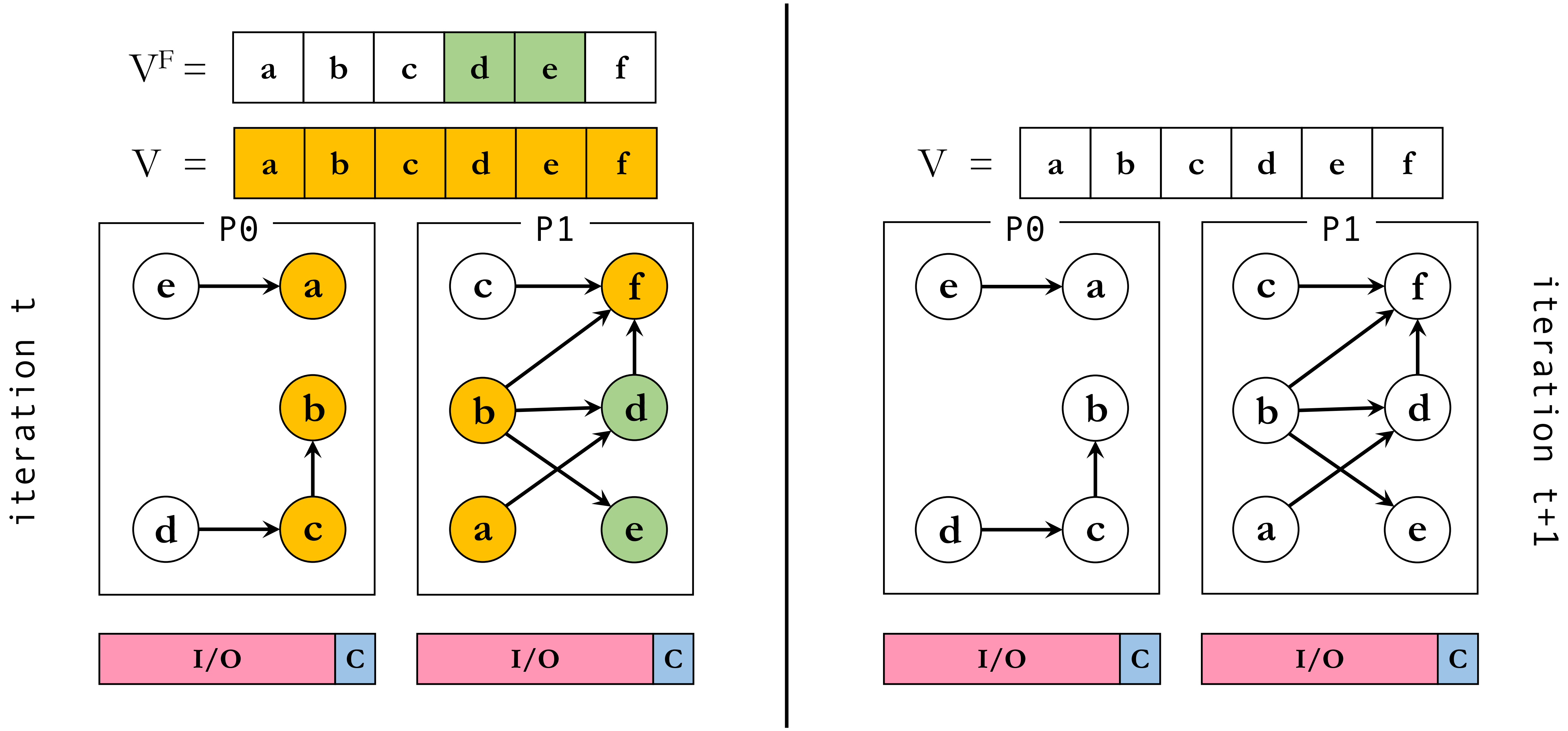


- **Out-of-Order** processing
  - Amortize I/O across multiple iterations
- Guarantee **Bulk Synchronous Parallel** Semantics
- **Dependency-Driven Cross-Iteration** value propagation
  - ***Safely*** compute future values

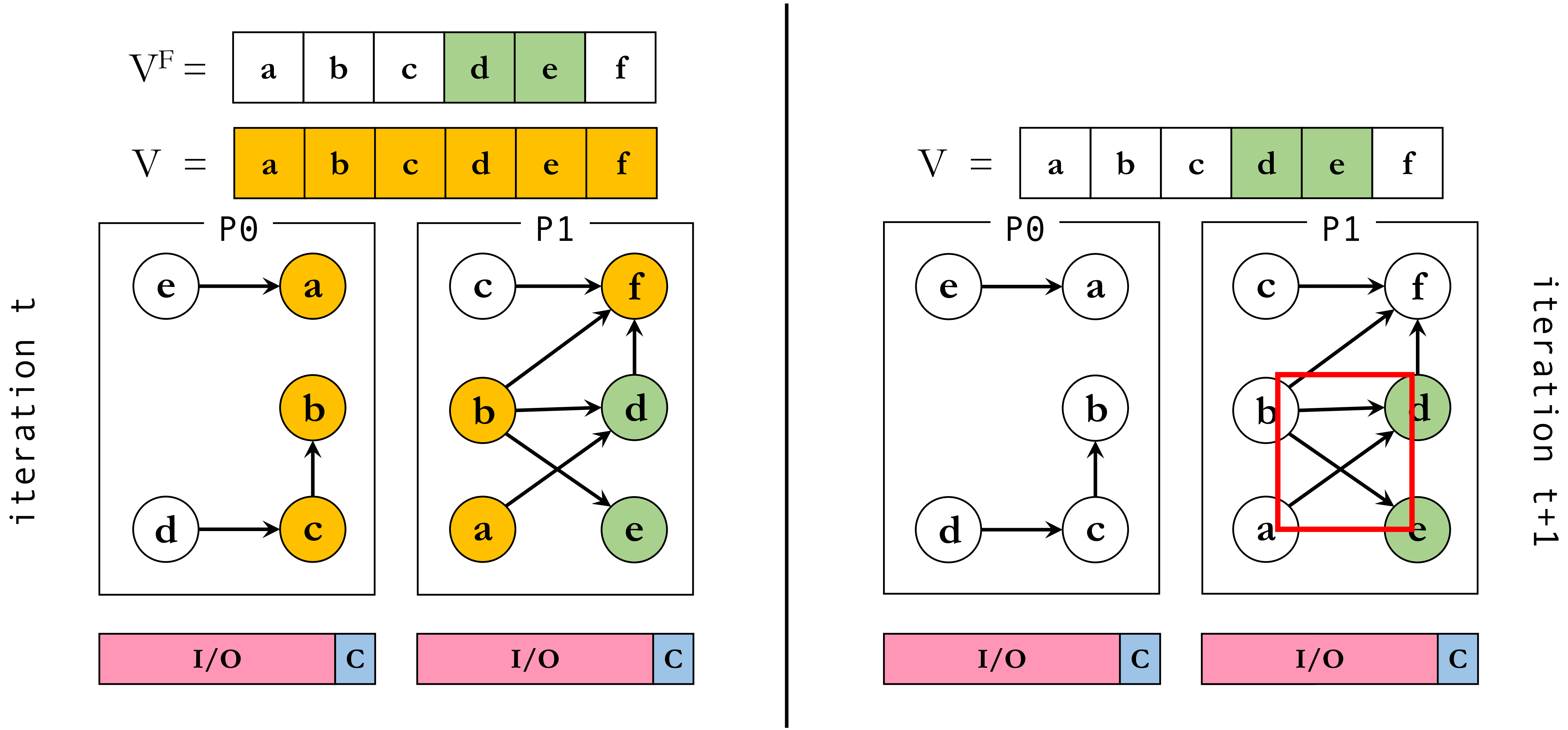
Computing Future Values



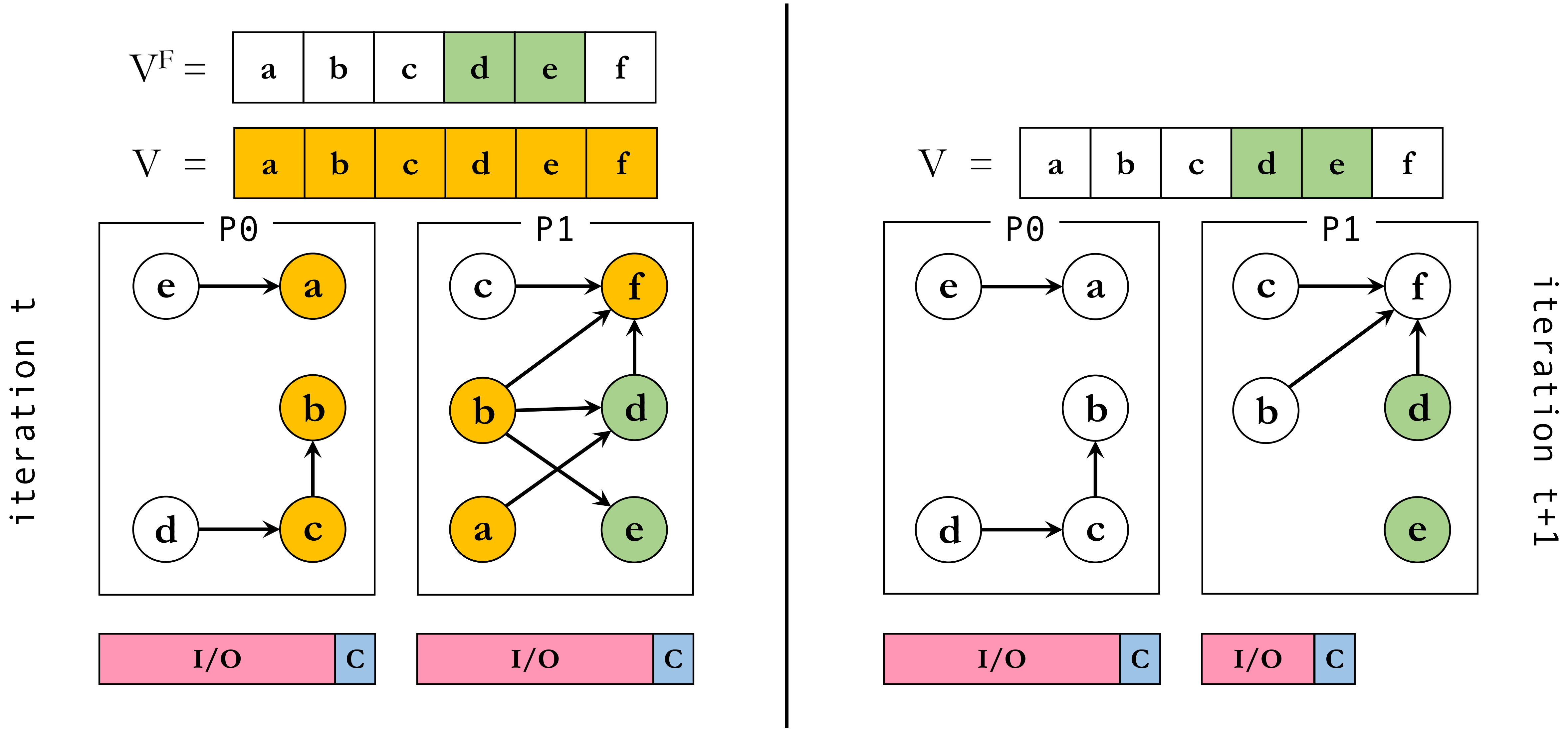
Computing Future Values



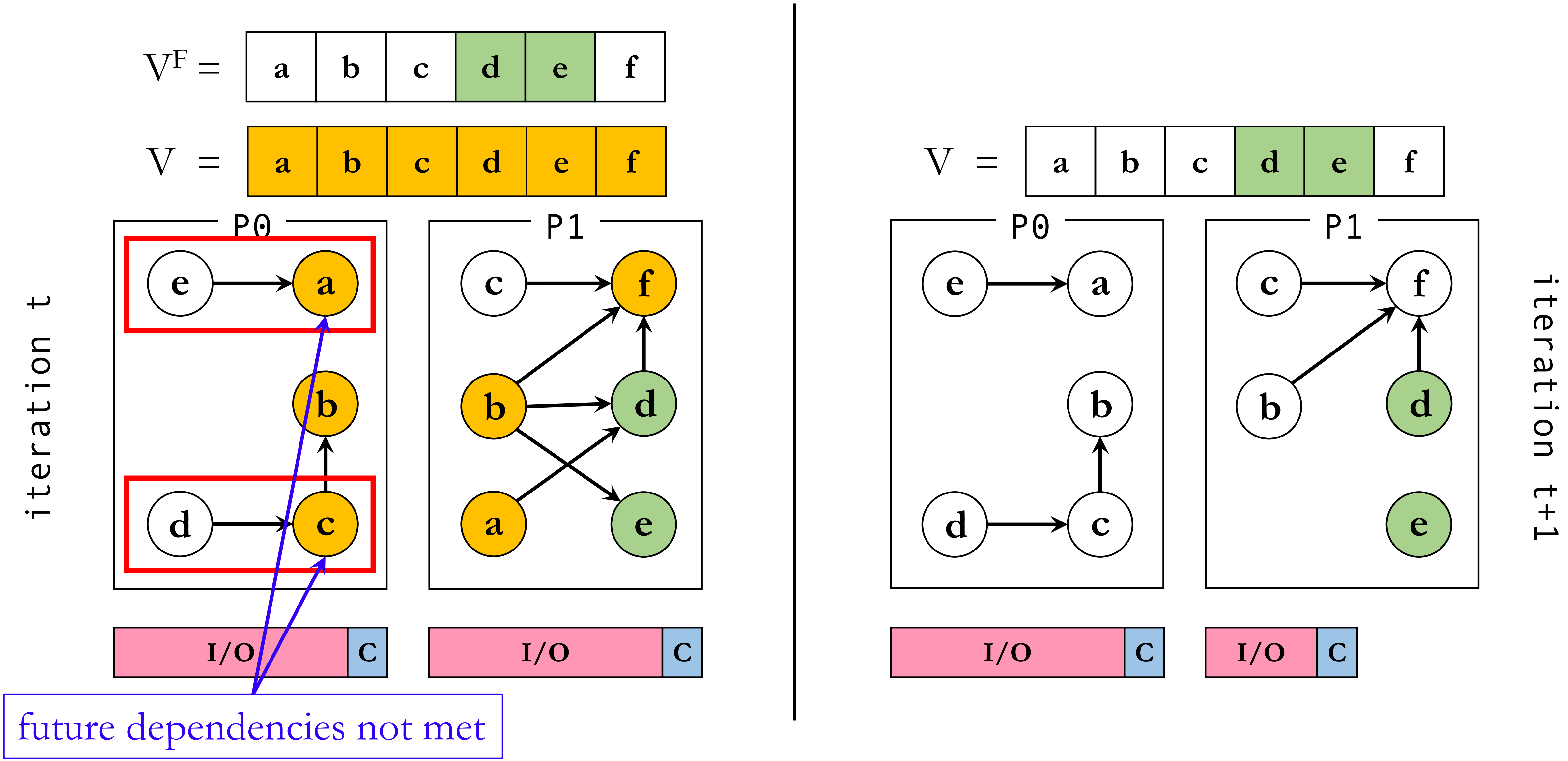
Computing Future Values



Computing Future Values

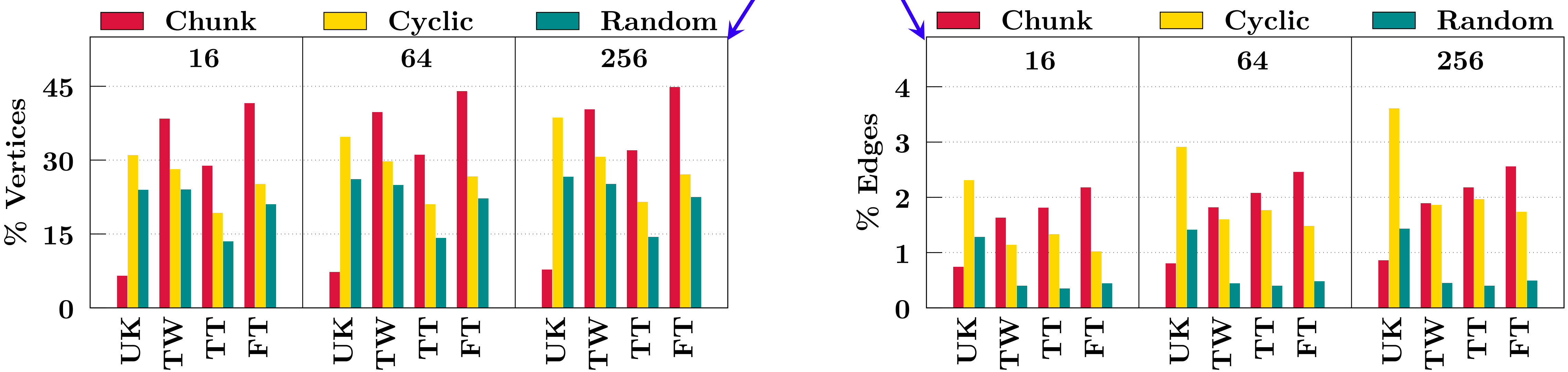


Computing Future Values



Computing Future Values

30-45% vertices compute future values, and they contribute for less than 4% edges



	Edges	Vertices
UK	1B	39.5M
TW	1.5B	41.7M
TT	2B	52.6M
FT	2.5B	68.3M

Graph Computation

$$\mathrm{PR}(v^t) = \underbrace{1 - d \ + \ d \times}_{f} \overbrace{\sum_{\mathrm{inedges}(v)}}^{\bigoplus} \mathrm{PR}(u^{t-1}) \ / \ |\mathrm{outedges}(u)|$$

$$v^t = f\Big(\bigoplus_{\mathrm{inedges}(v)} (u^{t-1})\Big)$$

Partial Aggregation

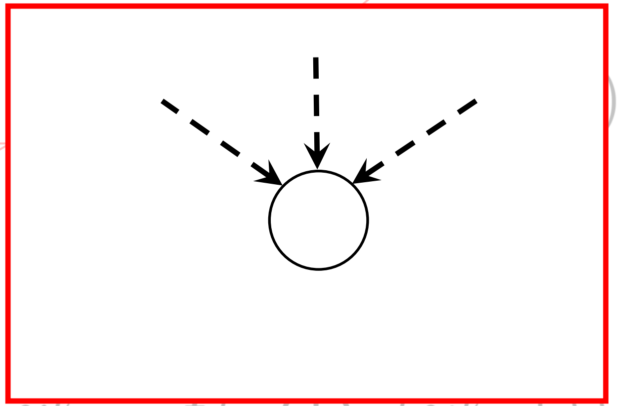
$$\mathrm{inedges}(v) = \mathrm{X} \cup \mathrm{Y} \implies \bigoplus_{\mathrm{inedges}(v)} (u^t) = \bigoplus \Big(\bigoplus_{\mathrm{X}}(u^t) \ , \ \bigoplus_{\mathrm{Y}}(u^t) \ \Big)$$

Graph Computation

$\bigoplus$

$\mathrm{PR}(v^t) = 1 - d$ … $)\ /\ |\mathrm{outedges}(u)|$

f



$v^t = f(\bigoplus_{\mathrm{inedges}(v)} (u^{t-1}))$

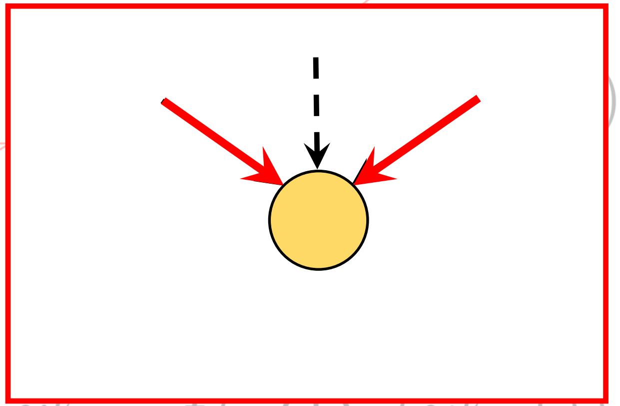
Partial Aggregation

$$\mathrm{inedges}(v) = \mathrm{X} \cup \mathrm{Y} \implies \bigoplus_{\mathrm{inedges}(v)} (u^t) = \bigoplus \Big(\bigoplus_{\mathrm{X}} (u^t)\ ,\ \bigoplus_{\mathrm{Y}} (u^t) \Big)$$

Graph Computation

$$\mathrm{PR}(v^t) = 1 - d \quad) \;/\; |\mathrm{outedges}(u)|$$

f



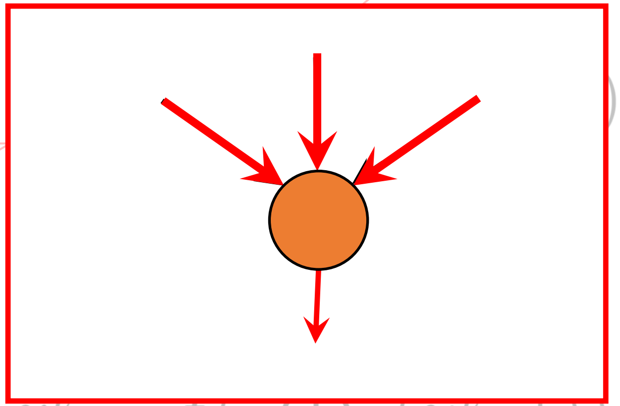
$$v^t = f(\bigoplus_{\mathrm{inedges}(v)} (u^{t-1}))$$

Partial Aggregation

$$\mathrm{inedges}(v) = \mathrm{X} \cup \mathrm{Y} \implies \bigoplus_{\mathrm{inedges}(v)} (u^t) = \bigoplus\left(\bigoplus_{\mathrm{X}}(u^t) \;,\; \bigoplus_{\mathrm{Y}}(u^t) \right)$$

Graph Computation

$\bigoplus$

$\mathrm{PR}(v^t) = 1 - d$  $) \ / \ |\mathrm{outedges}(u)|$

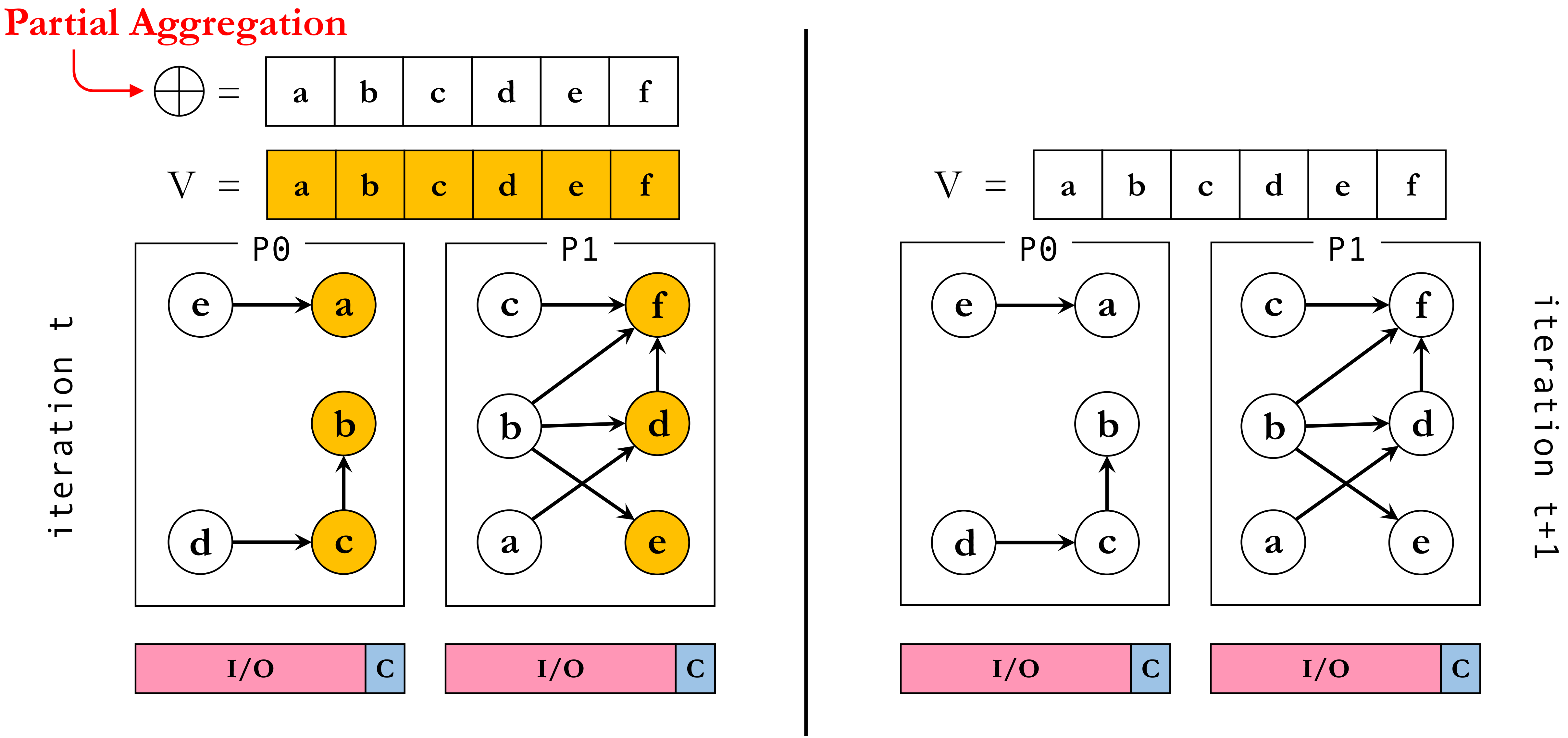
f

$v^t = f(\bigoplus_{\mathrm{inedges}(v)} (u^{t-1}))$

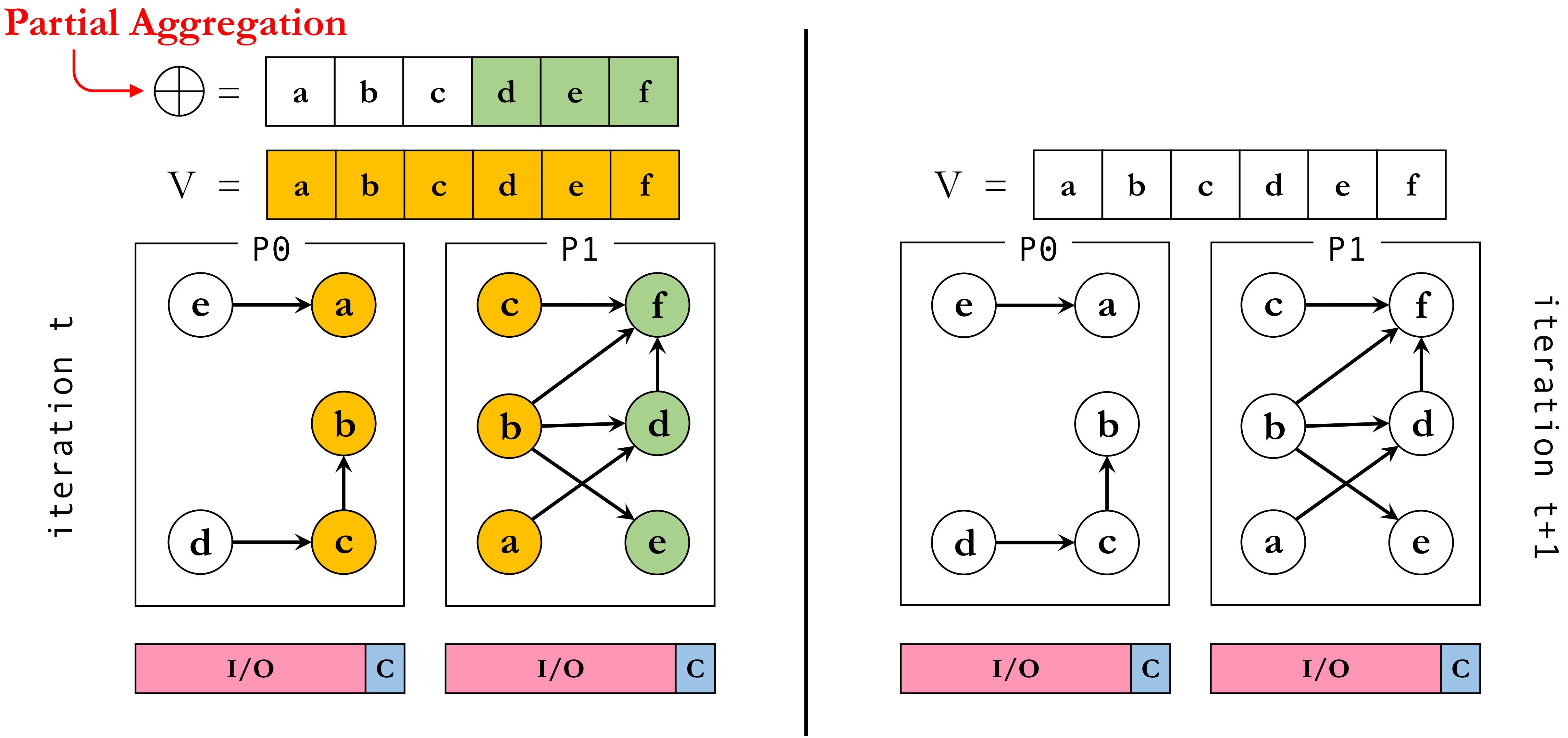
Partial Aggregation

$$\mathrm{inedges}(v) = \mathrm{X} \cup \mathrm{Y} \implies \bigoplus_{\mathrm{inedges}(v)} (u^t) = \bigoplus \left(\bigoplus_{\mathrm{X}} (u^t) \ , \ \bigoplus_{\mathrm{Y}} (u^t) \right)$$

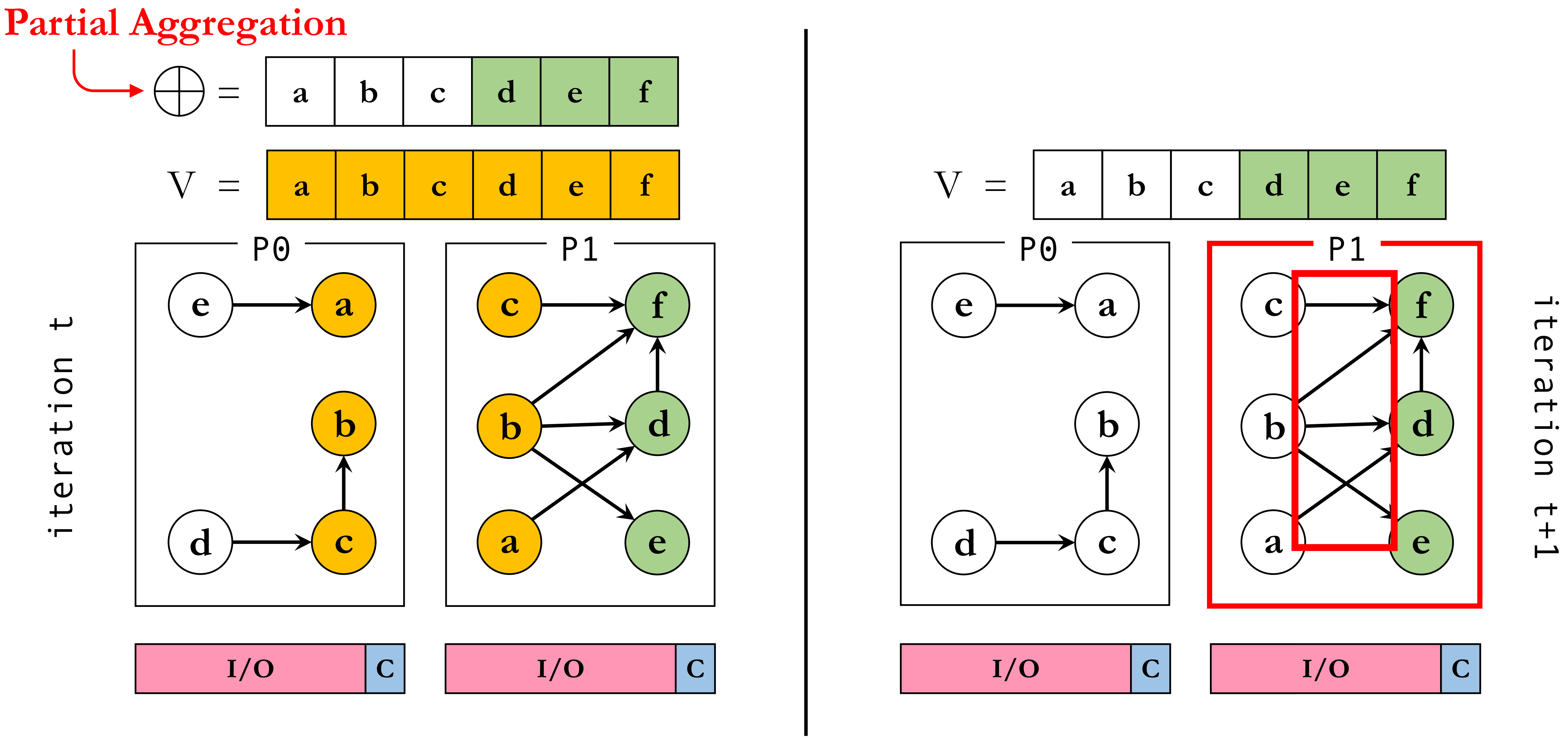
Cross-Iteration Value Propagation



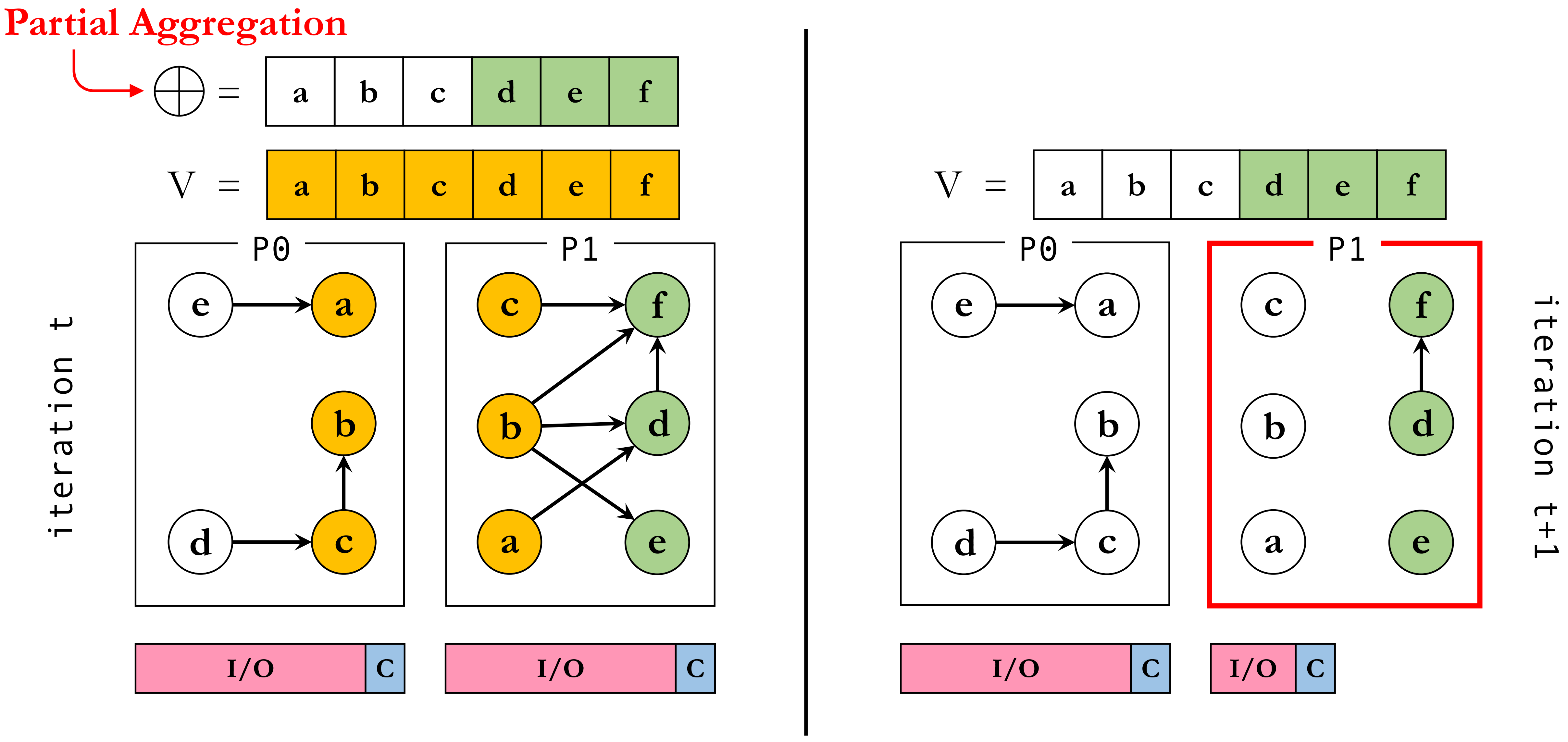
Cross-Iteration Value Propagation



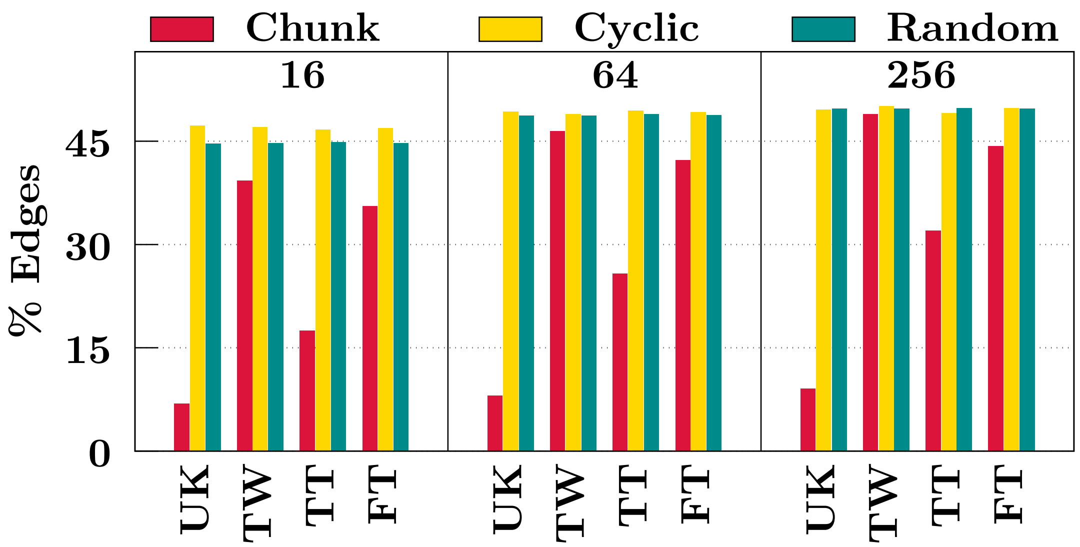
Cross-Iteration Value Propagation



Cross-Iteration Value Propagation

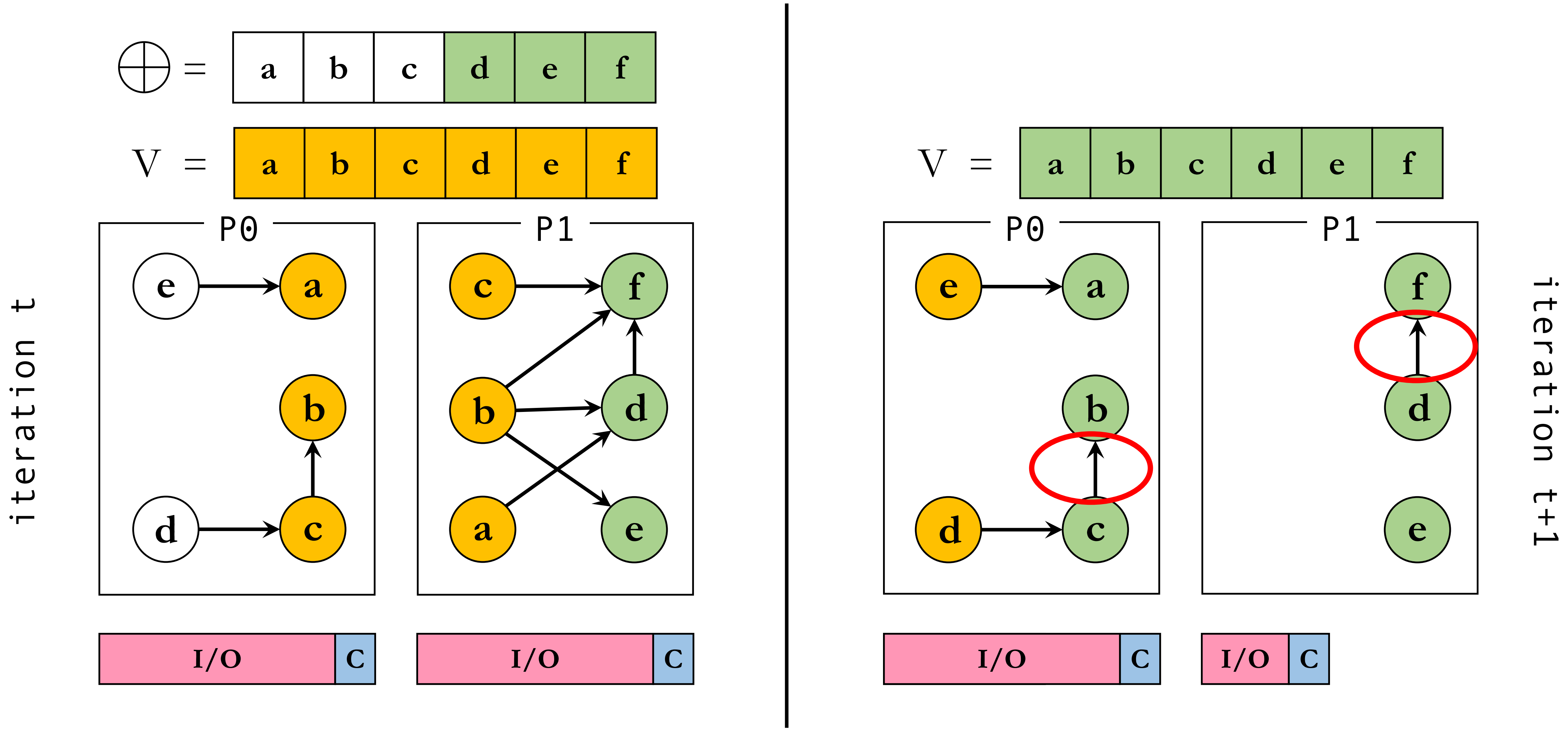


Cross-Iteration Value Propagation

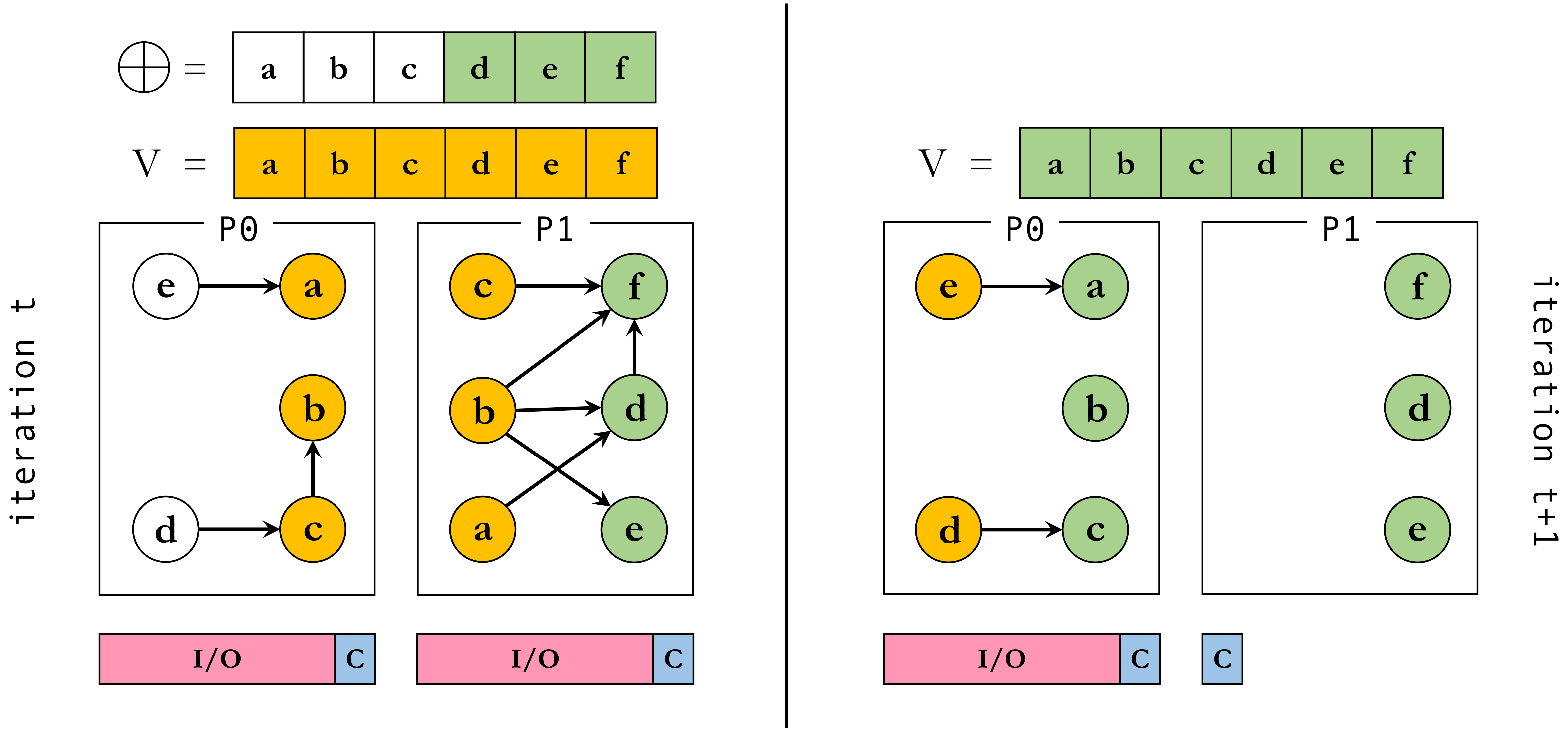


future values computed across >45% edges

Intra-Partition Propagation



Intra-Partition Propagation



Lumos: Cross-Iteration Value Propagation

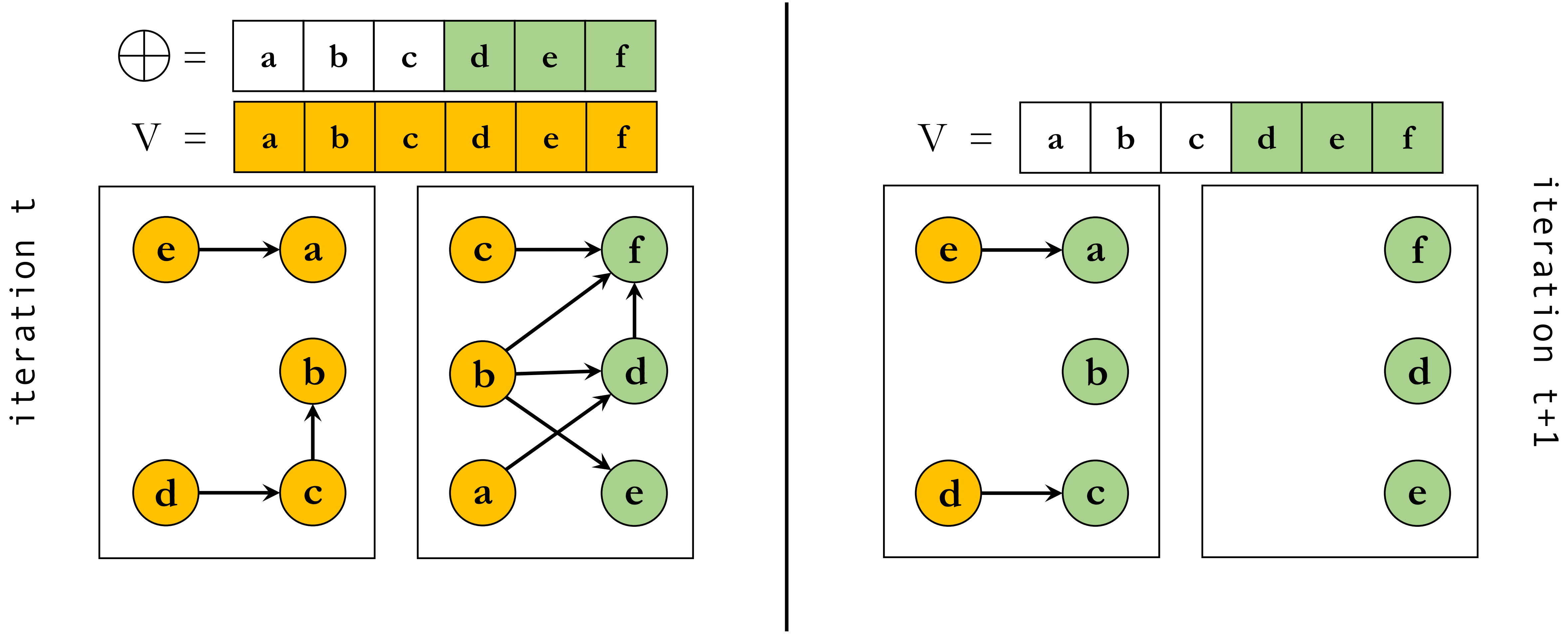
- Propagation across **multiple future iterations**
- Synergy with **Dynamic Shards** [ATC'16]
- Detailed **comparison with out-of-order** techniques
- Generalized **graph layout** & **partitioning strategies**
- Cross-iteration propagation across **different partition sizes**
- Interplay with **selective scheduling**

- Lumos for **Asynchronous Algorithms**

Lumos for Asynchronous Algorithms

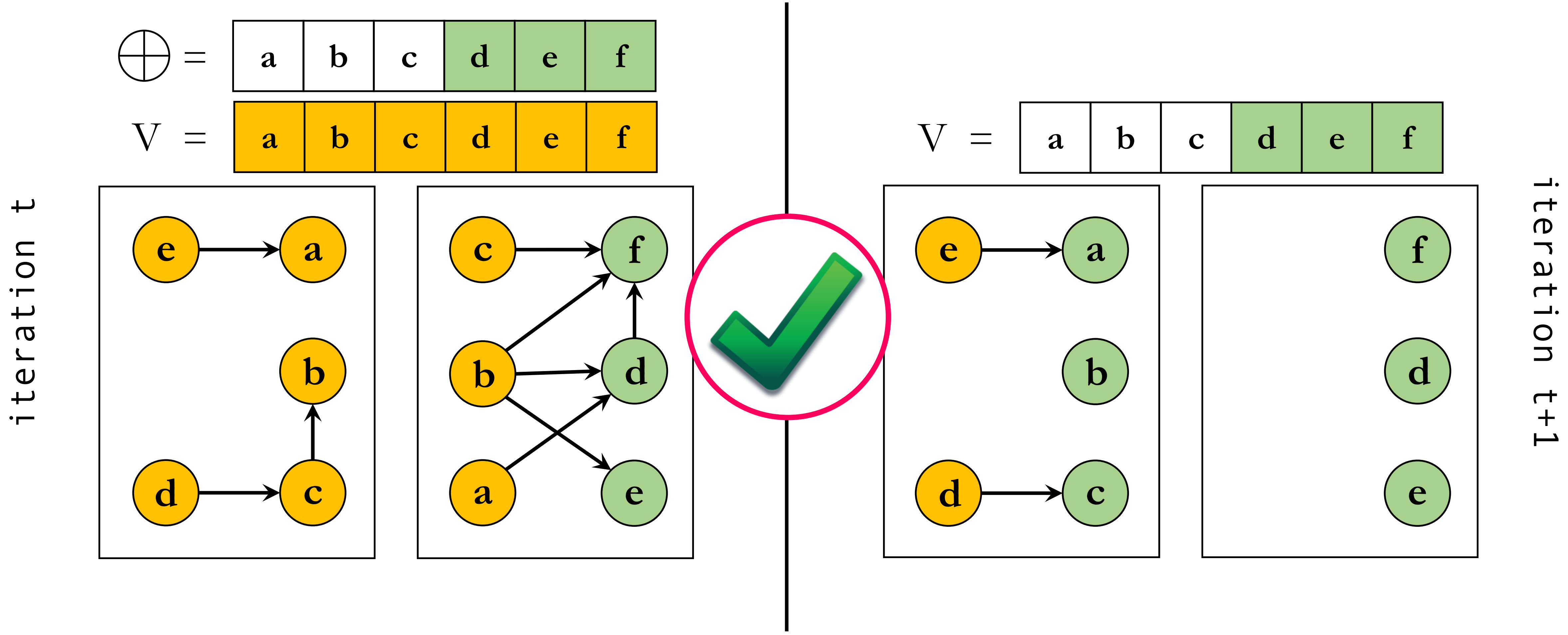
shortest paths computation

$$\mathrm{SSSP}(v^{t+1}) = \min_{\mathrm{inedges}(v)} \mathrm{SSSP}(u^t) + \mathrm{weight}(u, v)$$



Lumos for Asynchronous Algorithms

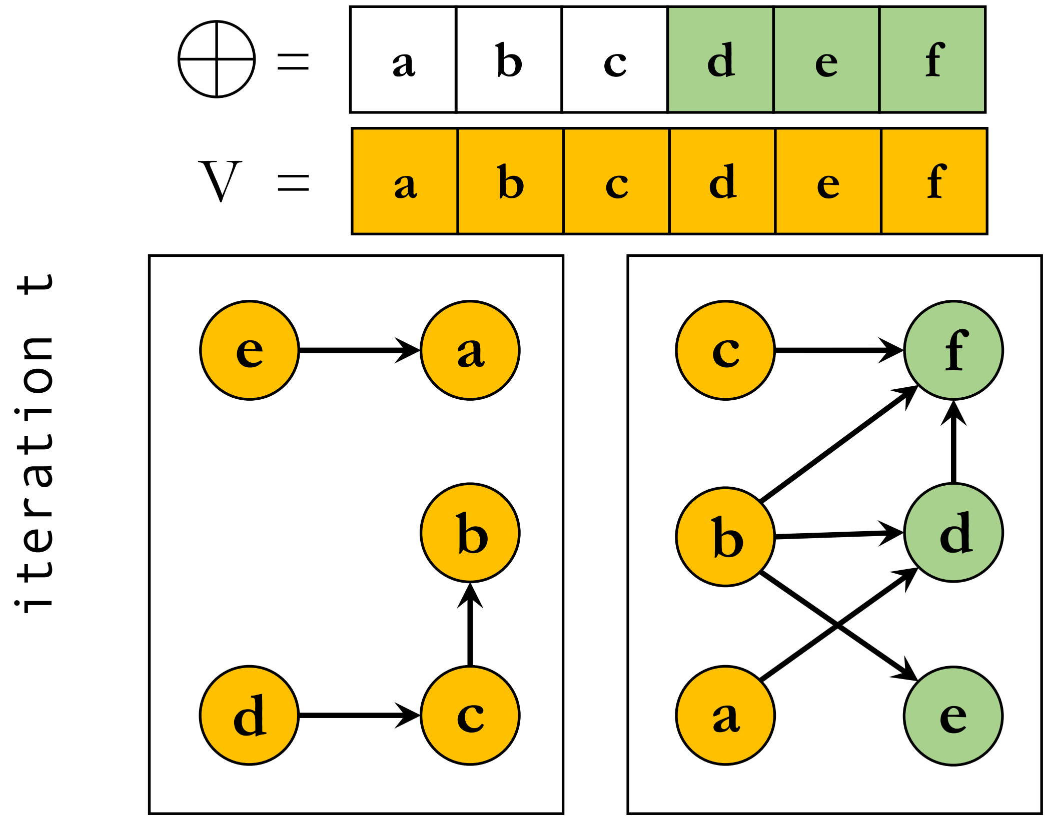
$$\mathrm{SSSP}(v^{t+1}) = \min_{\mathrm{inedges}(v)} \mathrm{SSSP}(u^t) + \mathrm{weight}(u, v)$$



Lumos for Asynchronous Algorithms

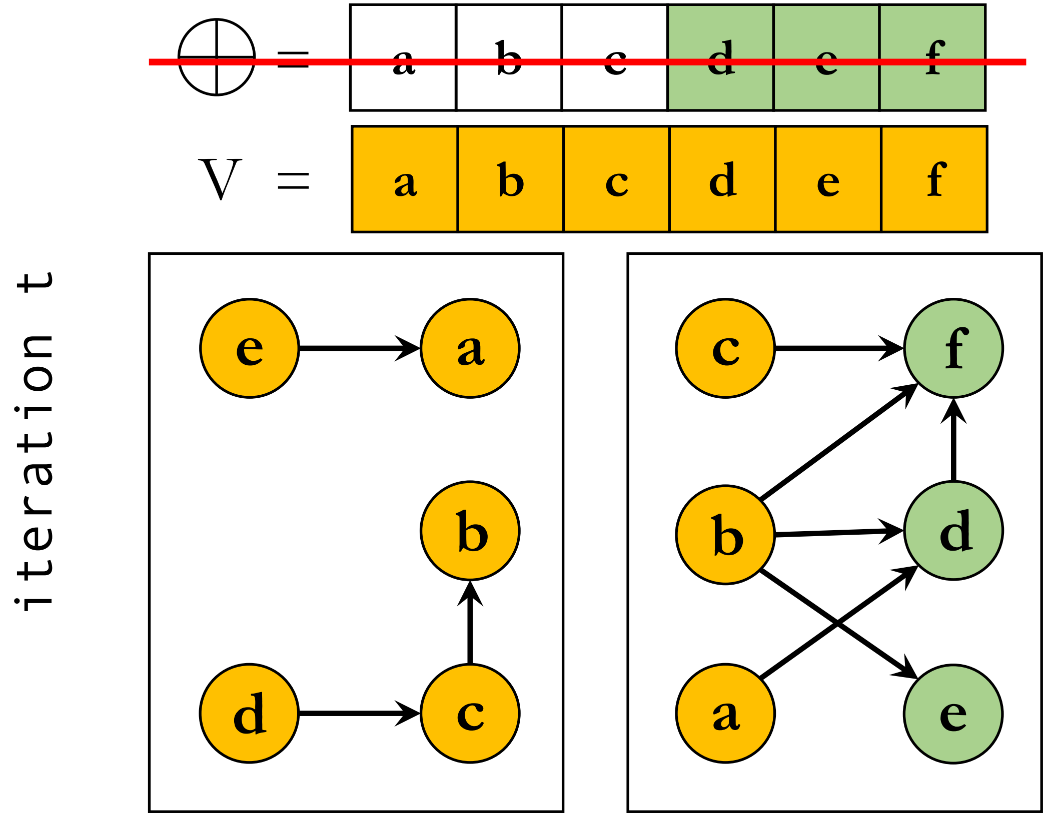
$$\mathrm{SSSP}(v^{t+1}) = \min_{\mathrm{inedges}(v)} \mathrm{SSSP}(u^t) + \mathrm{weight}(u, v)$$

- Monotonic selection (KickStarter [ASPLOS'17])



Lumos for Asynchronous Algorithms

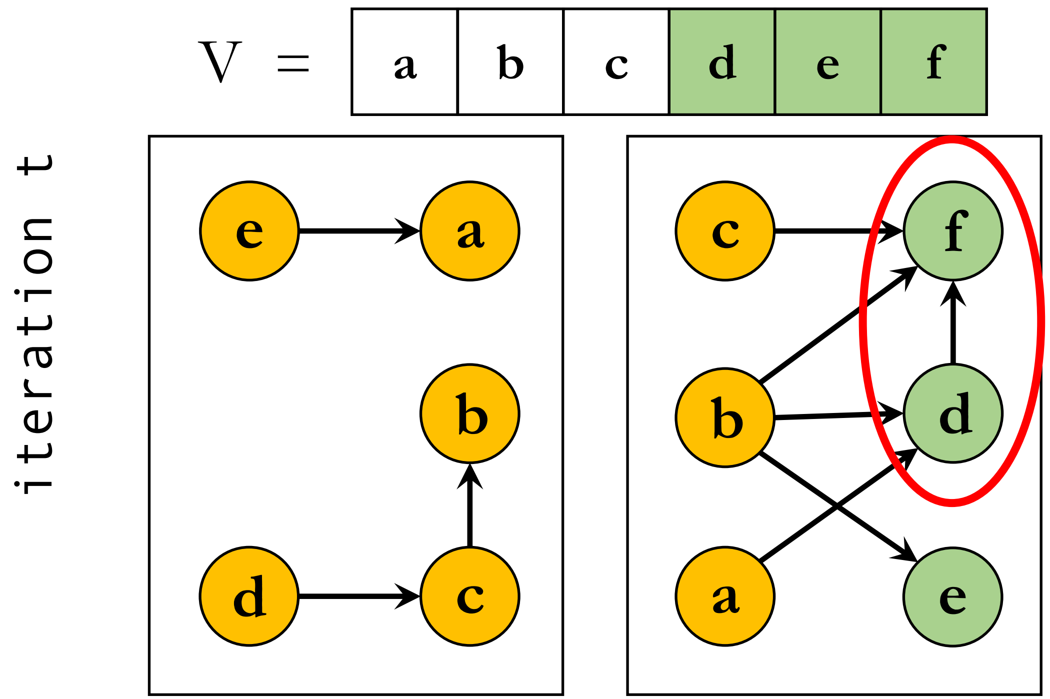
$$\mathrm{SSSP}(v^{\cancel{t+1}}) = \cancel{\min}_{\mathrm{inedges}(v)} \mathrm{SSSP}(u^{\cancel{t}}) + \mathrm{weight}(u, v)$$



- Monotonic selection (KickStarter [ASPLOS'17])
- Partial aggregations merged directly

Lumos for Asynchronous Algorithms

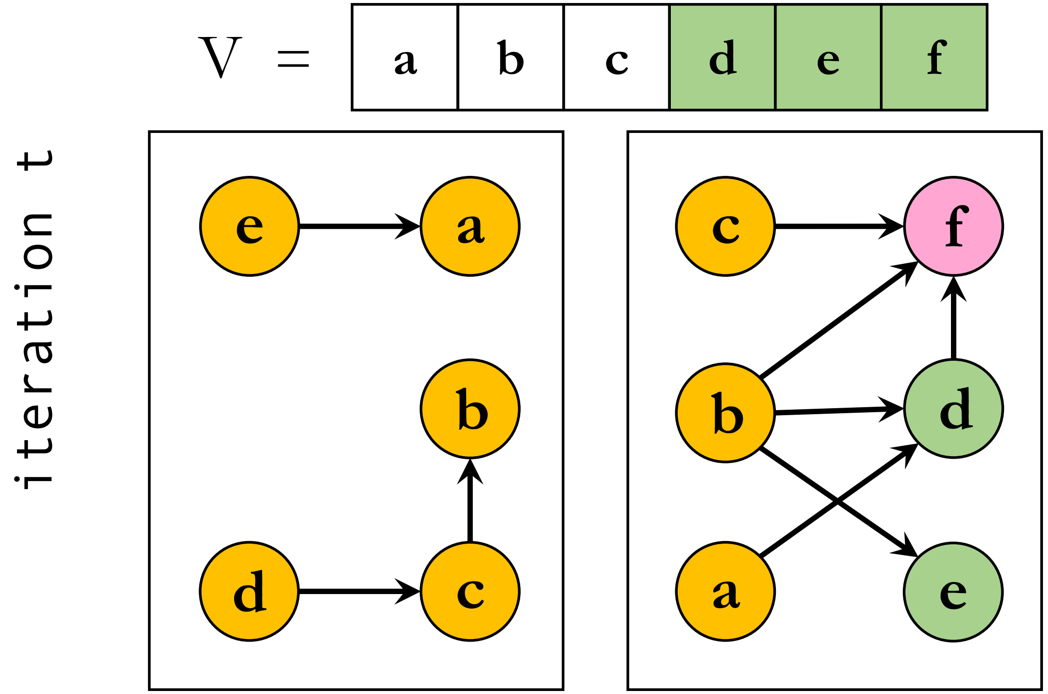
$$\mathrm{SSSP}(v) = \min_{\mathrm{inedges}(v)} \mathrm{SSSP}(u) + \mathrm{weight}(u, v)$$



- Monotonic selection (KickStarter [ASPLOS'17])
- Partial aggregations merged directly
- Partial values safe to be propagated
    - Any change can be propagated immediately

Lumos for Asynchronous Algorithms

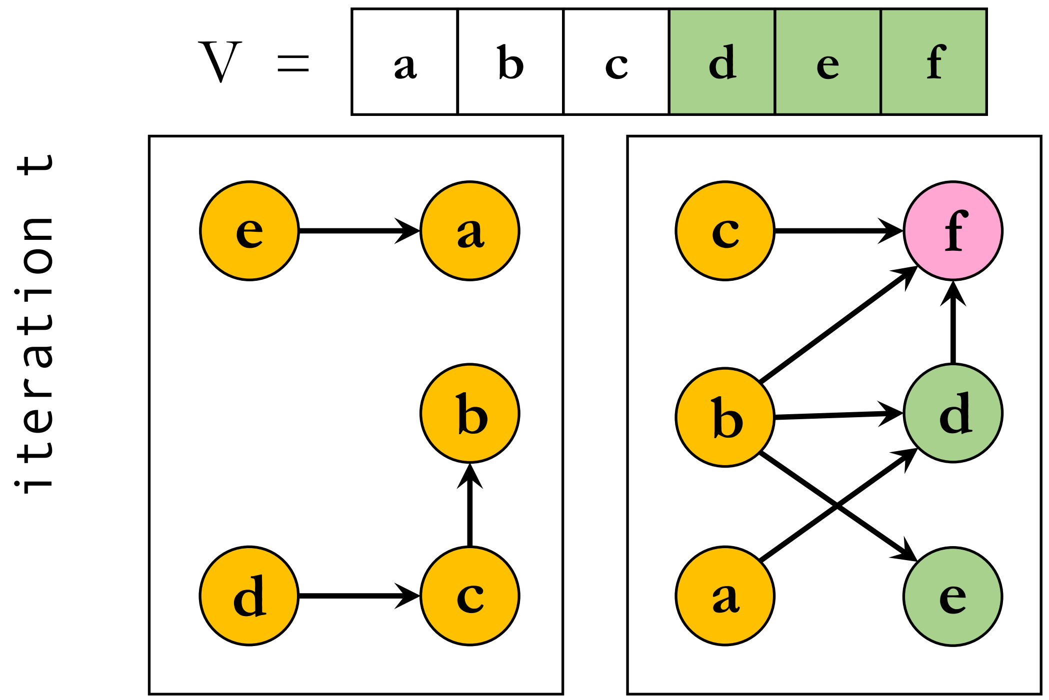
$$\mathrm{SSSP}(v) \;=\; \min_{\mathrm{inedges}(v)} \mathrm{SSSP}(u) + \mathrm{weight}(u, v)$$



- Monotonic selection (KickStarter [ASPLOS'17])
- Partial aggregations merged directly
- Partial values safe to be propagated
  - Any change can be propagated immediately

Lumos for Asynchronous Algorithms

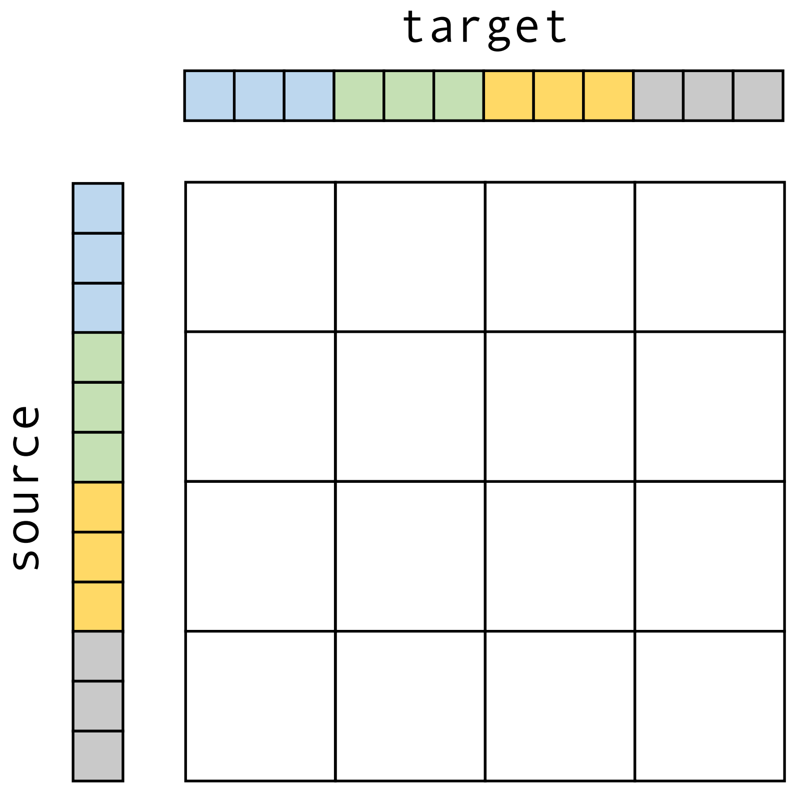
$$\text{SSSP}(v) \;=\; \min_{\text{inedges}(v)} \text{SSSP}(u) + \text{weight}(u, v)$$



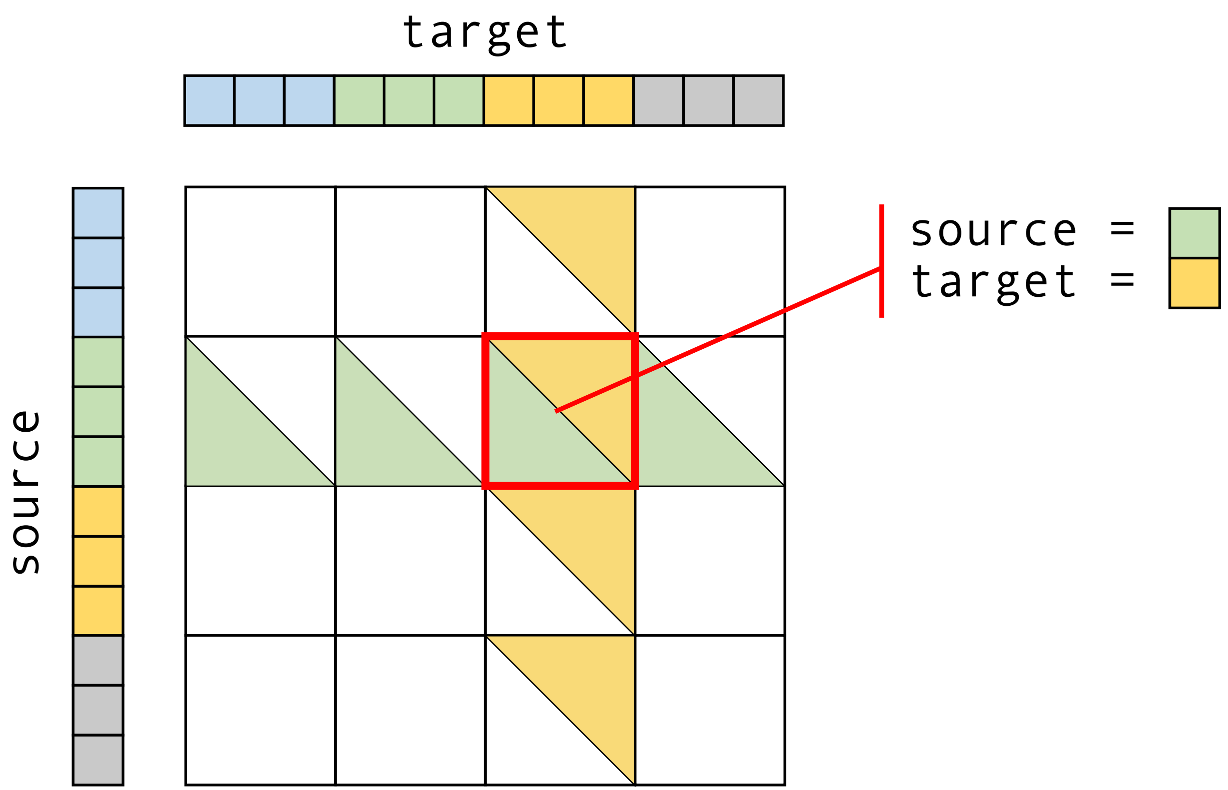
- Monotonic selection (KickStarter [ASPLOS'17])
- Partial aggregations merged directly
- Partial values safe to be propagated
  - Any change can be propagated immediately
- Intra-partition propagation based on degree of asynchrony (ASPIRE [OOPSLA'14])

Lumos System

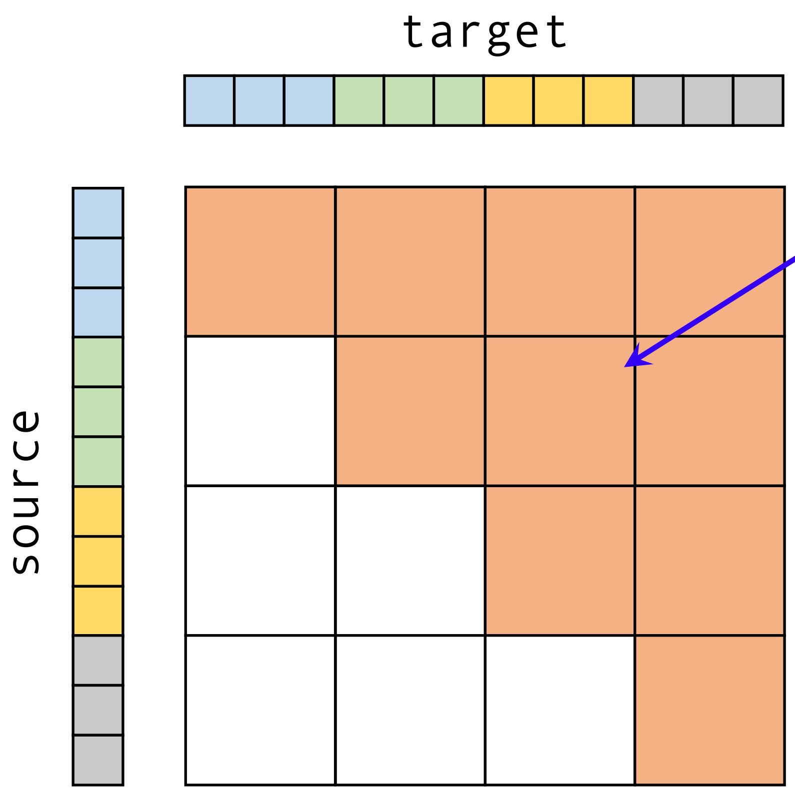
- Grid layout based on GridGraph [ATC'15]



Lumos System

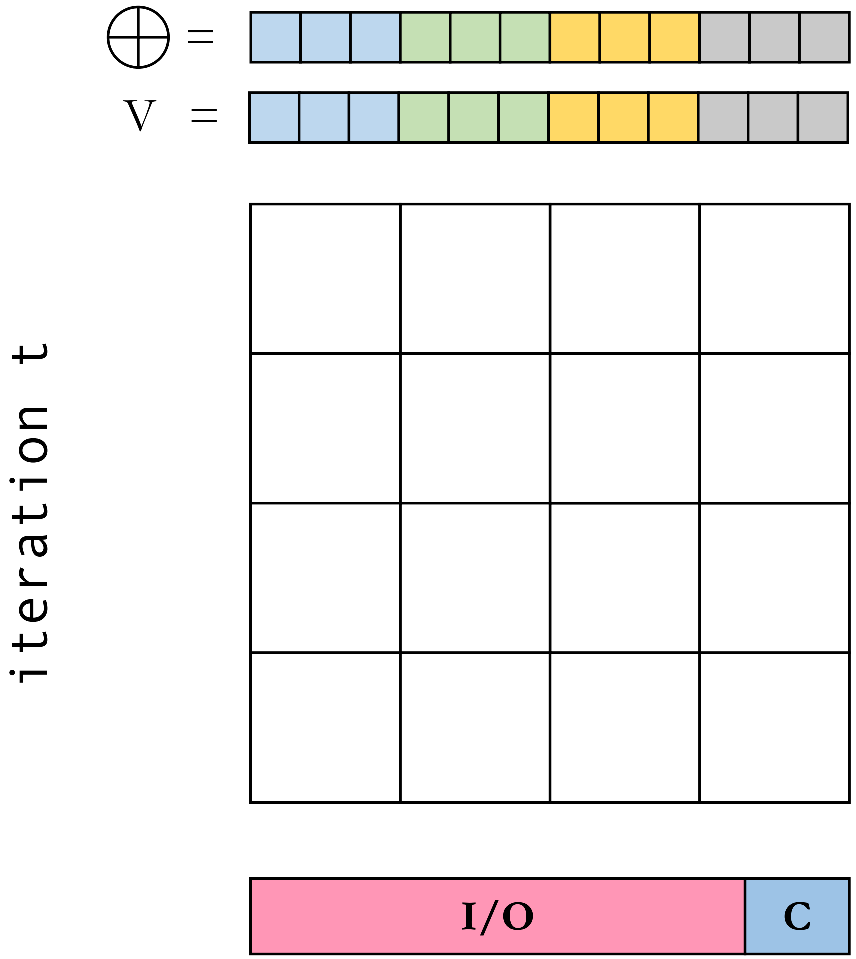


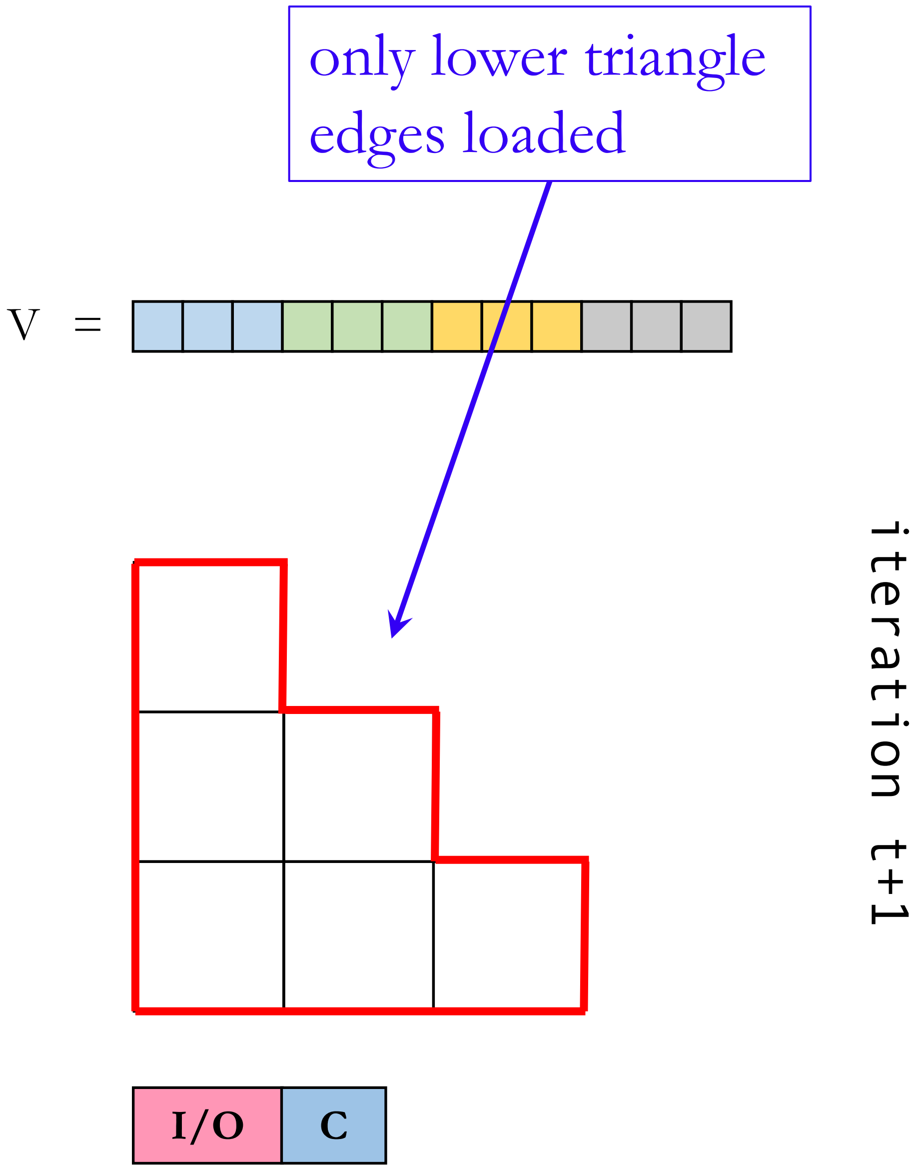
Lumos System



future values propagated for upper triangle + diagonal edges

Lumos System

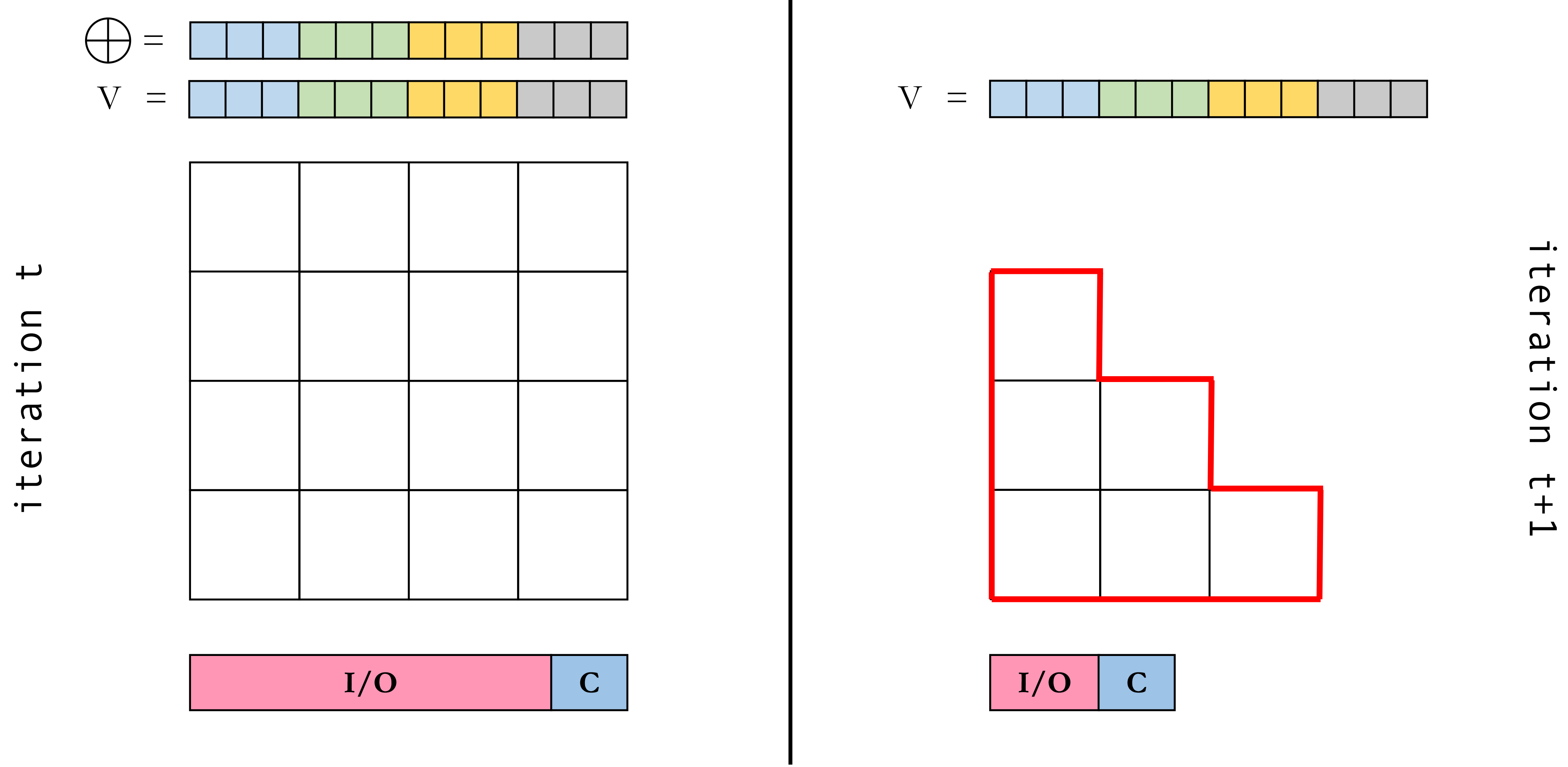




Lumos System

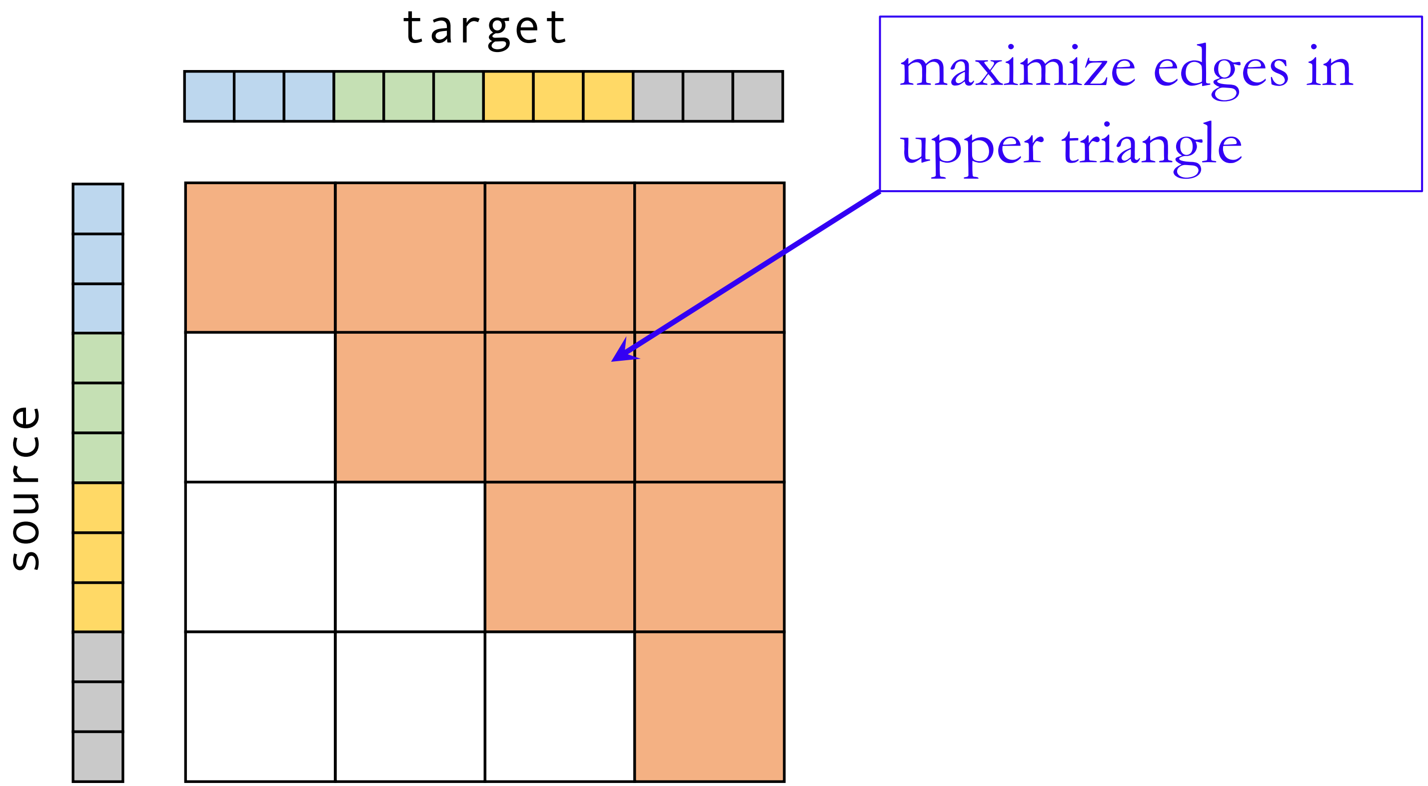
function processPrimary(propagate, crossPropagate, compute)

function processSecondary(propagate, compute)



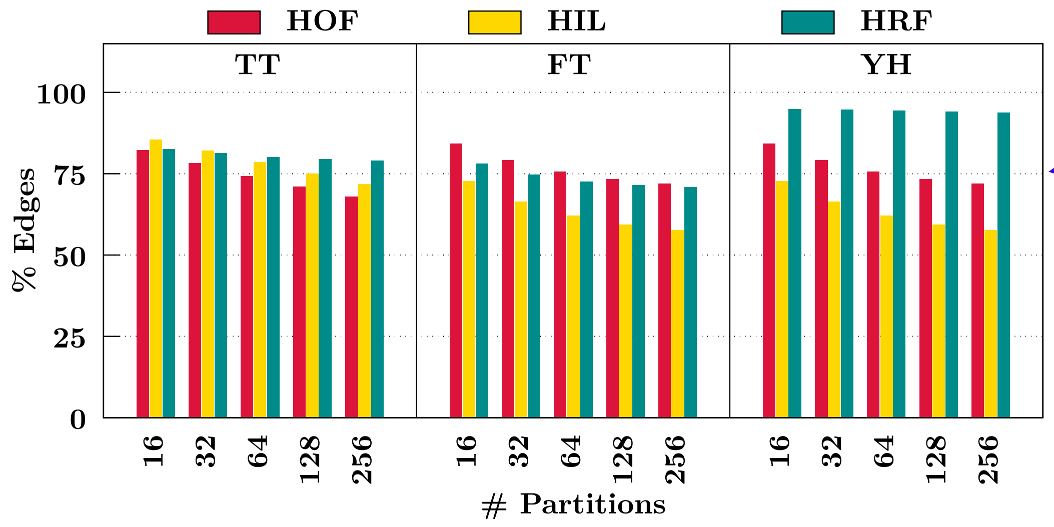
Light-weight Degree-Aware Partitioning

HOF: Highest Out-Degree First
HIL: Highest In-Degree Last
HRF: Highest Out-Degree to In-Degree Ratio First



Light-weight Degree-Aware Partitioning

HOF: Highest Out-Degree First
HIL: Highest In-Degree Last
HRF: Highest Out-Degree to In-Degree Ratio First



cross-iteration propagation across 72-93% edges

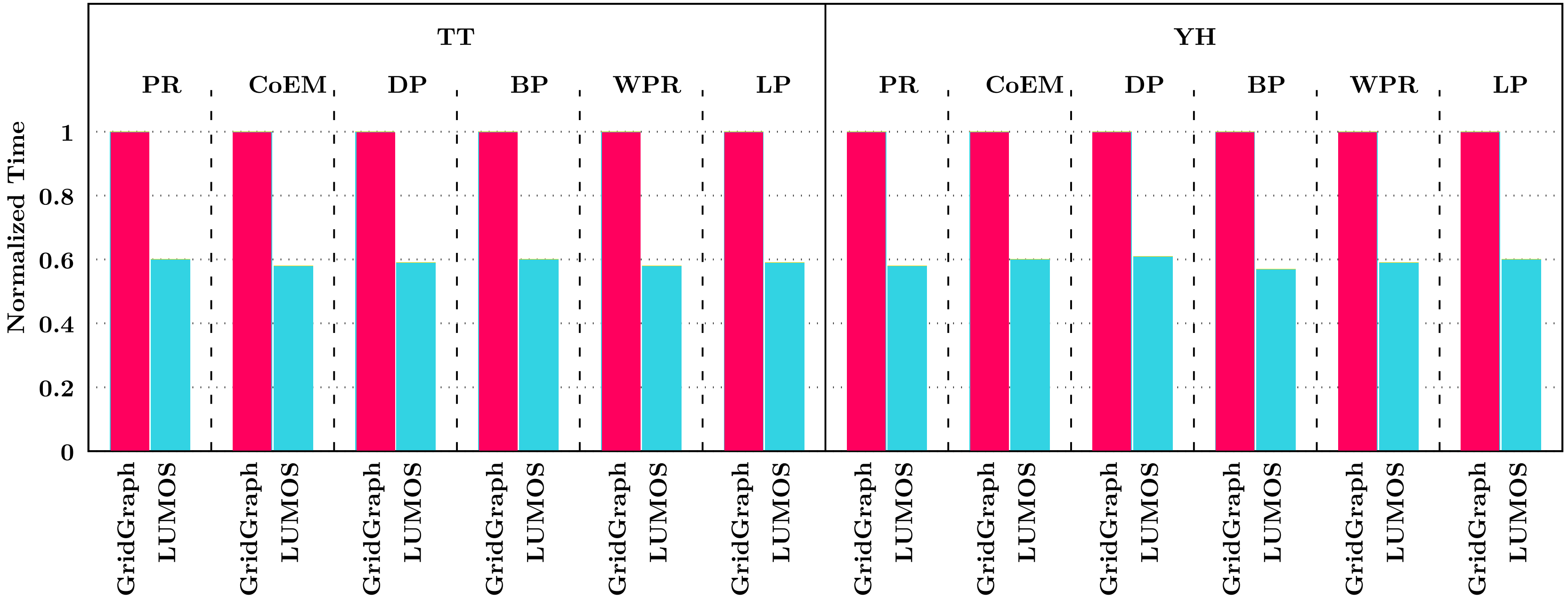
	Edges	Vertices
TT	2B	52.6M
FT	2.5B	68.3M
YH	6.6B	1.4B

Experimental Setup

- Graph algorithms
  - PageRank (weighted and unweighted), Belief Propagation, Co-Training Expectation Maximization, Dispersion, Label Propagation
- Performance on single disk
  - h1.2xlarge: 278MB/sec HDD read bandwidth
- Scaling I/O
  - d2.4xlarge: 195-768MB/sec read bandwidth over 1-4 HDDs
  - i3.8xlarge: 1.2-4.1GB/sec read bandwidth over 1-4 SSDs

Performance

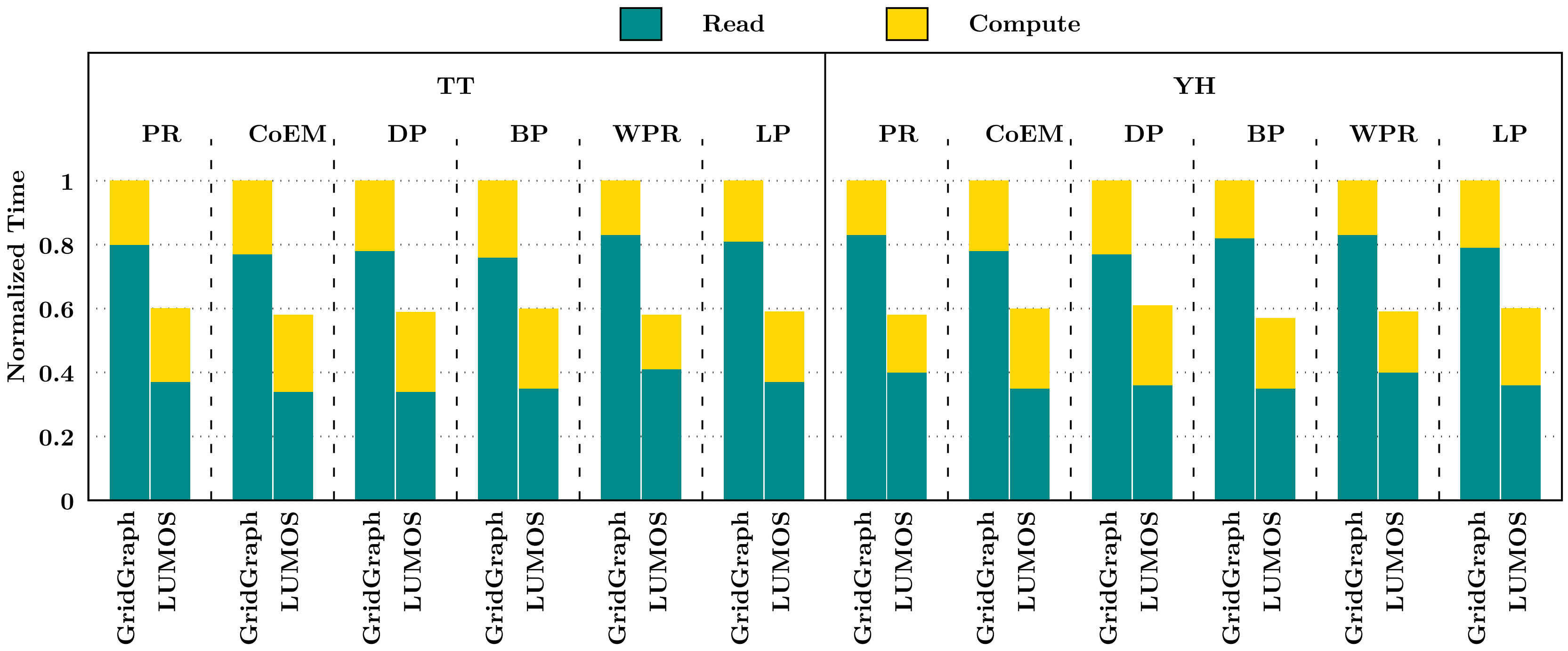
h1.2xlarge: 278MB/sec HDD read bandwidth



	Edges	Vertices
TT	2B	52.6M
YH	6.6B	1.4B

Performance

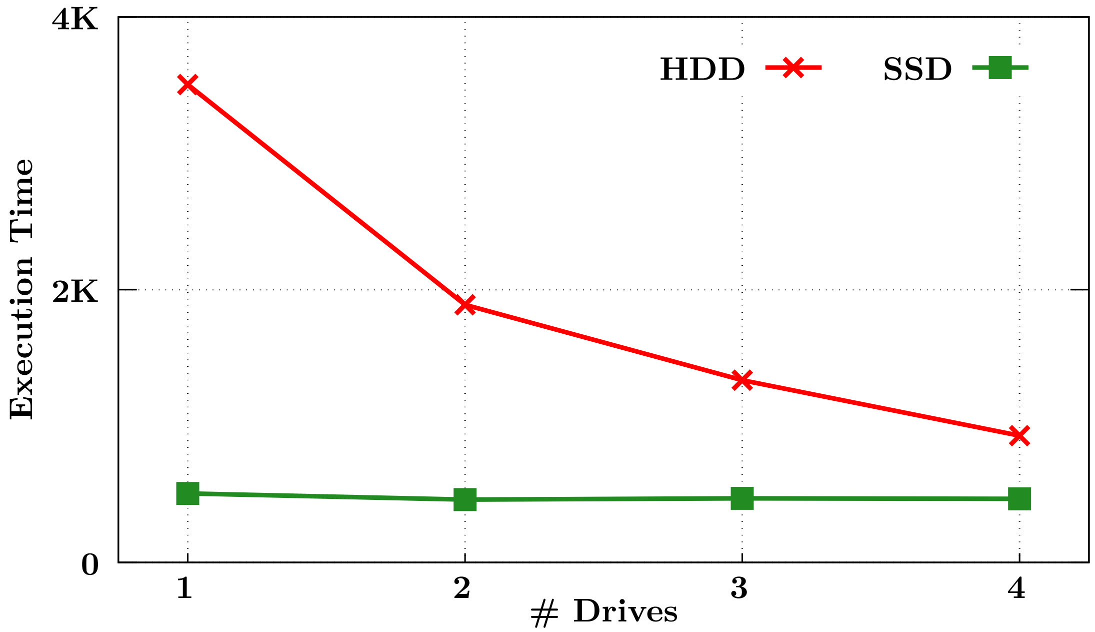
h1.2xlarge: 278MB/sec HDD read bandwidth



	Edges	Vertices
TT	2B	52.6M
YH	6.6B	1.4B

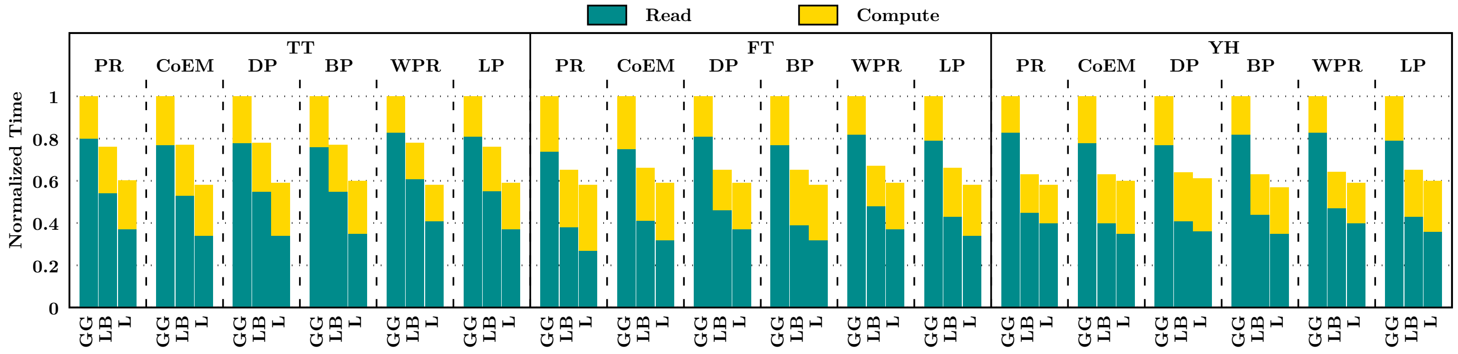
Scaling I/O

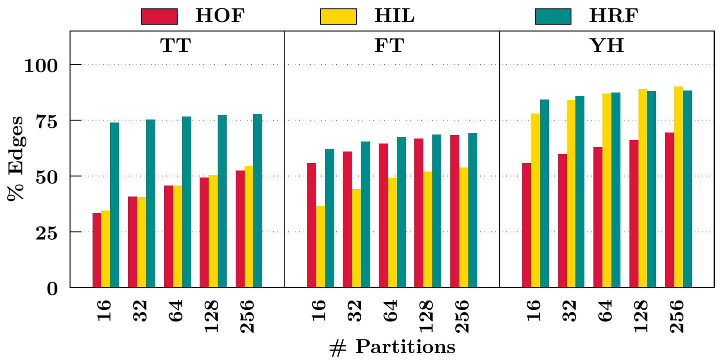
	Edges	Vertices
RMAT29	8.6B	537M

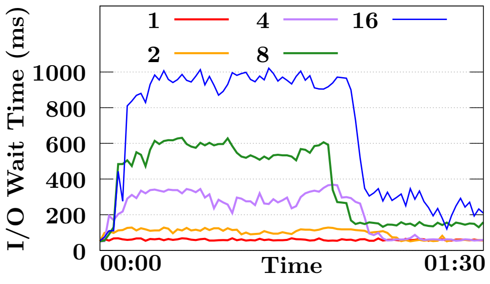


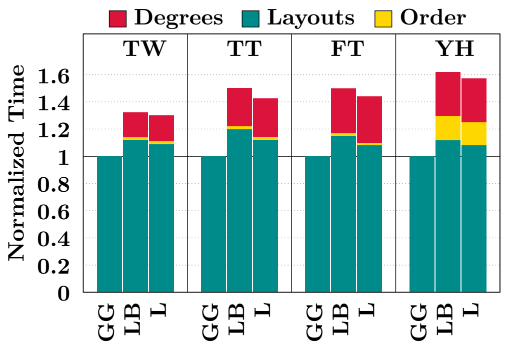
	Device Type	Single Drive	RAID-0 with k drives		
			k = 2	k = 3	k = 4
d2.4xlarge	HDD	195MB/s	368MB/s	590MB/s	768MB/s
i3.8xlarge	SSD	1.2GB/s	3.8GB/s	4.1GB/s	3.9GB/s

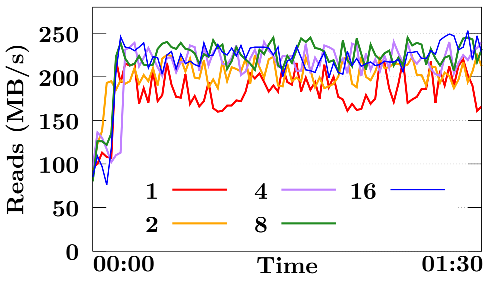
Detailed Evaluation of Lumos

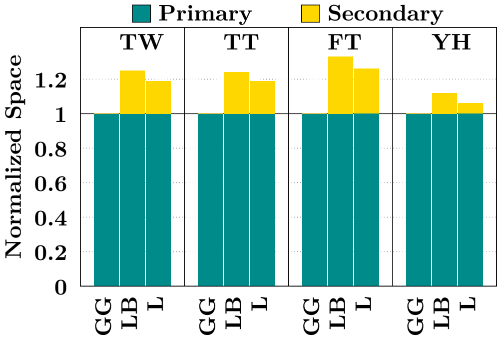












	Version	TT	FT	YH
PR	GridGraph	737	1008	3223
	Lumos-Base	563	659	2027
	Lumos	439	583	1885
	× Lumos	**1.68×**	**1.73×**	**1.71×**
CoEM	GridGraph	1119	1554	5082
	Lumos-Base	861	1029	3216
	Lumos	651	914	3043
	× Lumos	**1.72×**	**1.70×**	**1.67×**
DP	GridGraph	846	1032	3484
	Lumos-Base	656	675	2219
	Lumos	498	611	2111
	× Lumos	**1.70×**	**1.69×**	**1.65×**
BP	GridGraph	2498	3782	13769
	Lumos-Base	1921	2456	8660
	Lumos	1487	2212	7913
	× Lumos	**1.68×**	**1.71×**	**1.74×**
WPR	GridGraph	984	1302	4330
	Lumos-Base	769	874	2758
	Lumos	569	770	2547
	× Lumos	**1.73×**	**1.69×**	**1.70×**
LP	GridGraph	1054	1421	4583
	Lumos-Base	805	935	2976
	Lumos	624	826	2728
	× Lumos	**1.69×**	**1.72×**	**1.68×**

Conclusion

github.com/pdclab/lumos

- **Dependency-Driven Cross-iteration** value propagation
    - **Out-of-Order** processing to reduce I/O
    - Guarantees **Bulk Synchronous Parallel** semantics
- Generic technique that fundamentally eliminates the performance barrier