Sandbox

Ben Johnson Weitang Liu Agnieszka Łupińska Muhammad Osama John D. Owens Yuechao Pan Leyuan Wang Xiaoyun Wang Carl Yang UC Davis

Contents

1 Sandbox for playing with pandoc/slate

Chapter 1

Sandbox for playing with pandoc/slate

Normal table

Operation	Self	Peer	Host	All
Regular read	448.59	14.01	444.74	12.17
Regular write	442.98	16.21	16.18	12.17
Regular update	248.80	11.71	0.0028	6.00
Random read	6.78	1.43	2.39	4.04
Random write	6.63	1.14	3.47E-5	3.82
Random update	3.44	0.83	1.92E-5	2.08

Pretty wide table

Parts	Comp. cost	Comm.	Comp. to comm.	Scalal	oi ldy mory usage
Modularity optimization	10(E + V) /p	20V bytes	E/p : 2V	Okay	88E/p + 12V bytes
Graph contraction	5E / p + E'	8E' bytes	5E/p + E': 8E'	Hard	16E' bytes
Louvain	10(E + V) / p	20V bytes	E/p: 2V	Okay	88E/p + 12V + 16E' bytes

Pretty wide table, with math

Parts	Comp. cost	Comm.	Comp. to comm.	Scalab	oi My mory usage
Modularity optimization Graph	10(E+V)/p $5E/p+E'$	20V bytes $8E'$	E/p: 2V $5E/p + E':$	Okay Hard	88E/p + 12V bytes $16E'$ bytes
contraction Louvain	10(E+V)/p	bytes $20V$ bytes	8E' E/p : 2V	Okay	88E/p + 12V + 16E' bytes

JDO hacked version of above

Parts	Comp cost	Comm cost	Comp/comm ratio	Scalability	Memory usage (B)
Modularity optim. Graph contraction Louvain	10(E + V) /p	20V bytes	E/p : 2V	Okay	88E/p + 12V
	5E / p + E'	8E' bytes	5E/p + E' : 8E'	Hard	16E'
	10(E + V) / p	20V bytes	E/p : 2V	Okay	88E/p + 12V + 16E'

and with math!

Parts	Comp cost	Comm cost	Comp/comm ratio	Scalability	Memory usage (B)
Modularity optim. Graph contraction Louvain	10(E+V)/p $5E/p+E'$ $10(E+V)/p$	20V bytes $8E'$ bytes $20V$ bytes	E/p : 2V 5E/p + E' : 8E' E/p : 2V	Okay Hard Okay	88E/p + 12V 16E' 88E/p + 12V + 16E'

Even wider table

Parts	Comp.	Comm.	Comp. to comm. ratio	Scalability	Memory usage
Wedge generation	dE/p				

	Comp.	Comm.	Comp. to	~	Memory
Parts	cost	cost	comm. ratio	Scalability	usage
Wedge	0	$aE/p \ge 12$			
communication		bytes			
Wedge checking	aE/p x				
	$\log(d)$				
AllReduce	2V	$2V \times 4$			
		bytes			
Triangle	(d + a x)	$aE/p \times 12$	\sim (d + a x	Okay	
Counting	log(d))E/p + 2V	0 + 8V bytes	log(d)): 12a		
Scan Statistics	(d + a x)	12aE/p +	$\sim (d + a x)$	Okay	
(wedge checks)	+2V+	8V bytes	log(d)) : 12a		
	V/p				
Scan Statistics	Vdd +	8V bytes	dd:8	Perfect	
(intersection)	V/p				

Even wider table, with math!

	Comp.	Comm.	Comp. to	~	Memory
Parts	cost	cost	comm. ratio	Scalability	usage
Wedge	dE/p				
generation					
Wedge	0	$aE/p \times 12$			
communication		bytes			
Wedge checking	$aE/px\log$	$\zeta(d)$			
AllReduce	2V	$2V \times 4$			
		bytes			
Triangle	(d +	aE/px12 +	$(d+a\cdot\log(d))$:	Okay	
Counting	$ax \log(d)$	E V +bytes	12a		
	2V				
Scan Statistics	(d +	12aE/p +	$(d + a \cdot \log(d))$:	Okay	
(wedge checks)	$ax \log(d)$	E V +bytes	12a		
	2V +				
	V/p				
Scan Statistics	Vdd +	8V bytes	dd:8	Perfect	
(intersection)	V/p				

JDO hacked version of above

	Comp.	Comm. cost		
Parts	cost	(B)	Comp/comm ratio	Scalability
Wedge generation	dE/p			
Wedge communication	0	$aE/p \times 12$		
Wedge checking	aE/p x $log(d)$			
AllReduce	$2\overline{\mathrm{V}}$	$2V \times 4$		
Triangle Counting	$\begin{array}{l} (d + a x \\ log(d))E/p \\ + 2V \end{array}$	/ 1		Okay
Scan Statistics	$(d + a x \log(d))E/p$	12aE/p + 8V	\sim (d + a x log(d)) : 12a	Okay
(with wedge checks)	+ 2V + V/p			
Scan Statistics	Vdd + V/p	8V	dd: 8	Perfect
(with intersection)	•			

Table with line breaks

Application	Computation to communication ratio	Scalability	Implementation difficulty
Louvain	E/p : 2V	Okay	Hard
Graph SAGE	$\sim CF : min(C, 2p)x4$	Good	Easy
Random walk	Duplicated graph: infinity	Perfect	Trivial
	Distributed graph: 1:24	Very poor	Easy
Graph search: Uniform	1:24	very poor	Easy
Graph search: Greedy	Straight forward: d : 24 Pre-visit: 1:24	Poor very poor	Easy Easy
Graph search:	Straight forward: d: 24	Poor very	Easy Easy
Stochastic greedy	Pre-visit: $log(d)$: 24	poor	
Geo location	Explicit movement: 25E/p:	Okay	Easy Easy
	4V UVM or peer access: 25:	Good	
Vertex nomination	$E: 8V \times min(d, p)$	Okay	Easy
Scan statistics	Duplicated graph: infinity	Perfect	Trivial
	Distributed graph: \sim (d + a * log(d)) : 12	Okay	Easy
Sparse fused lasso	~ a:8	Less than okay	Hard
Graph projection	Duplicated graph : infinity Distributed graph : $dE/p + E' : 6E'$	Perfect Okay	Easy Easy
Local graph clustering	(6 + d)/p : 4	Good	Easy

Application	Computation to communication ratio	Scalability	Implementation difficulty
Seeded graph matching			
Application			
classification			

Table with line breaks, laid out as a grid table

	Computation to		Implementation
Application	communication ratio	Scalability	difficulty
Louvain	E/p:2V	Okay	Hard
Graph SAGE	$\sim CF: \min(C,2p)\cdot 4$	Good	Easy
Random walk	Duplicated graph: infinity	Perfect	Trivial
	Distributed graph: 1 : 24	Very poor	Easy
Graph search: Uniform	1:24	Very poor	Easy
Graph search:	Straightforward: d : 24	Poor	Easy
Graph search. Greedy	Pre-visit: 1:24	Very poor	Easy
Graph search:	Straightforward: d : 24	Poor	Easy
Stochastic greedy	Pre-visit: $\log(d)$: 24	Very poor	Easy
Geolocation	Explicit movement:	Okav	Easy
Geolocation	25E/p:4V	Good	Easy
	UVM or peer access: $25:1$	Good	Базу
Vertex nomination	$E: 8V \cdot \min(d, p)$	Okav	Easy
Scan statistics	Duplicated graph: infinity	Perfect	Trivial
	Distributed graph: $\sim (d + a \cdot \log(d)) : 12$	Okay	Easy
Sparse fused lasso	$\sim a:8$	Less than okay	Hard
Graph projection	Duplicated graph: infinity	Perfect	Easy
	Distributed graph : $dE/p + E' : 6E'$	Okay	Easy
Local graph clustering Seeded graph matching Application classification	(6+d)/p:4	Good	Easy

JDO hacked version of above

	Computation to		Implementation
Application	communication ratio	Scalability	diff.
Louvain	E/p: 2V	Okay	Hard
Graph SAGE	$\sim CF : min(C, 2p)x4$	Good	Easy
Random walk	Duplicated graph: infinity	Perfect	Trivial
Random walk	Distrib. graph: 1 : 24	Very poor	Easy
Graph search: Uniform	1:24	Very poor	Easy
Graph search: Greedy	Straightforward: d: 24	Poor	Easy
Graph search: Greedy	Pre-visit: 1:24	Very poor	Easy
G.S.: Stochastic greedy	Straightforward: d: 24	Poor	Easy
G.S.: Stochastic greedy	Pre-visit: $log(d)$: 24	Very poor	Easy
Geolocation	Explicit movement: $25E/p$: $4V$	Okay	Easy
Geolocation	UVM or peer access: 25 : 1	Good	Easy
Vertex nomination	$E: 8V \times \min(d, p)$	Okay	Easy
Scan statistics	Duplicated graph: infinity	Perfect	Trivial
Scan statistics	Distrib. graph: \sim (d + a * $\log(d)$) : 12	Okay	Easy
Sparse fused lasso	~ a:8	Less than okay	Hard
Graph projection	Duplicated graph: infinity	Perfect	Easy
Graph projection	Distrib. graph : $dE/p + E'$: $6E'$	Okay	Easy
Local graph clustering	(6 + d)/p : 4	Good	Easy

Really wide table

Parts	Comp. cost	Comm.	Comp. to comm. ratio	Scalability	Memory usage
Feature duplication					
Children selection	BC	8BC bytes	1:8	Poor	
Child-centric comp.	BCF x (2 + L + Wf1.y + Wa1.y)	4B x (F + Wf1.y + Wa1.y) x min(C, 2p) bytes	~ CF : min(C, 2p) x 4	Good	

Parts	Comp.	Comm.	Comp. to comm. ratio	Scalability	Memory usage
Source-centric comp.	B x (CF + (Wf1.y + Wa1.y) x (C + F + Wf2.y +	0 bytes	N.A.	N.A.	5
Graph SAGE	Wa2.y) B x (C + 3CF + 3LCF + (Wf1.y + Wa1.y) x (CF + C + F + Wf2.y + Wa2.y))	8BC + 4B x (F + Wf1.y + Wa1.y) x min(C, 2p) bytes	at least \sim CF : min(C, 2p) x 4	Good	
Direct feature access					
Child-centric comp.	BCF x (2 + L + Wf1.y + Wa1.y)	4B x ((F + Wf1.y + Wa1.y) x min(C, 2p) + CLF) bytes	~ (2 + L + Wf1.y + Wa1.y) : 4L	poor	
Graph SAGE	B x (C + 3CF + 3LCF + (Wf1.y + Wa1.y) x (CF + C + F + Wf2.y + Wa2.y))	8BC + 4B x (F + Wf1.y + Wa1.y) x min(C, 2p) + 4BCFL bytes	$\sim (2 + L + Wf1.y + Wa1.y)$: 4L	poor	

 $Feature\ in \\ UVM$

Parts	Comp.	Comm.	Comp. to comm. ratio	Scalability	Memory usage
Child-centric comp.	(2 + L + Wf1.y +	4B x (F + Wf1.y + Wa1.y) x min(C, 2p) bytes over GPU-GPU + 4BCLF bytes over GPU-CPU	Wf1.y + Wa1.y) : 4L over	very poor	
Graph SAGE	+ 3CF + 3LCF + (Wf1.y + Wa1.y) x (CF + C + F +	8BC + 4B		very	

Really wide table, with math

Parts	Comp.	Comm.	Comp. to comm. ratio	Scalability	Memory usage
Children selection	BC	8BC bytes	1:8	Poor	
Child-centric comp.	$\begin{array}{l}(2+L+\\ Wf1.y+\end{array}$		CF : min $(C, 2p) \cdot 4$	Good	
Source-centric comp.	$B \cdot (CF + (Wf1.y + Wa1.y) \cdot (C + F + Wf2.y + Wa2.y)$	0 bytes	N.A.	N.A.	

D. A	Comp.	Comm.	Comp. to	C 1 1 111	Memory
Parts	cost	cost	comm. ratio	Scalability	usage
Graph SAGE	$B \cdot (C + 3CF + 3LCF + (Wf1.y + Wa1.y) \cdot (CF + C + F + Wf2.y + Wa2.y))$	$8BC + 4B \cdot (F + Wf1.y + Wa1.y) \cdot \min(C, 2p)$ bytes	at least CF : $\min(C, 2p) \cdot 4$	Good	
Direct feature access					
Child-centric comp.	$BCF \cdot (2 + L + Wf1.y + Wa1.y)$	$4B \cdot ((F + Wf1.y + Wa1.y) \cdot \min(C, 2p) + CLF)$ bytes	(2+L+Wf1.y+Wa1.y):4L	poor	
Graph SAGE	$B \cdot (C + \\ 3CF + \\ 3LCF + \\ (Wf1.y + \\ Wa1.y) \cdot \\ (CF + \\ C + F + \\ Wf2.y + \\ Wa2.y))$	$8BC + 4B \cdot (F + Wf1.y + Wa1.y) \cdot \min(C, 2p) + 4BCFL$ bytes	(2+L+Wf1.y+Wa1.y):4L	poor	
Feature in UVM					
Child-centric comp.	$BCF \cdot (2+L+Wf1.y+Wa1.y)$	$4B \cdot (F + Wf1.y + Wa1.y) \cdot min(C, 2p)$ bytes over GPU-GPU + $4BCLF$ bytes over GPU-CPU	(2+L+Wf1.y+Wa1.y): 4L over GPU-CPU	very	

Parts	Comp.	Comm.	Comp. to comm. ratio	Scalability	Memory usage
Graph SAGE	$(Wf1.y+Wa1.y)\cdot (CF+C+F+$	$8BC + 4B \cdot (F + Wf1.y + Wa1.y) \cdot \min(C, 2p)$ bytes over GPU-GPU $+ 4BCFL$ bytes over GPU-CPU	(2+L+Wf1.y+Wa1.y): 4L over GPU-CPU	very poor	

Really wide table, with math, nicely laid out

Parts	Computation cost	Communication cost (Bytes)	Comp. to comm.	Scalability
Feature				
duplication				
Children selection	BC	8BC	1:8	Poor
Child-centric	$BCF \cdot (2 +$	$4B \cdot (F + Wf1.y +$	CF:	Good
comp.	L + Wf1.y +	$\operatorname{Wa}1.y)$ ·	$\min(C, 2p) \cdot 4$	
	Wa1.y)	$\min(C, 2p)$		
Source-centric	$B \cdot (CF +$	0	N.A.	N.A.
comp.	(Wf1.y +			
	Wa1.y) · (C +			
	F + Wf2.y +			
C 1 CACE	Wa2.y)	0 B C + 4 B / E +	.1	C 1
Graph SAGE	$B \cdot (C + 3CF +$	*		Good
	3LCF +	Wf1.y + Wa1.y) ·	$\min(C, 2p) \cdot 4$	
	$(\text{Wf1.}y + \text{Wa1.}y) \cdot$	$\min(C, 2p)$		
	(CF + C +			
	F + Wf2.u +			
	Wa2.y)			
	vv a2.g))			
Direct feature				
access				
Child-centric	$BCF \cdot (2 +$	$4B \cdot ((F +$	(2 + L + Wf1.y +	poor
comp.	L + Wf1.y +	Wf1.y + Wa1.y).	Wa1.y):4L	_
	Wa1.y)	$\min(C, 2p) + CLF)$		

Parts	Computation cost	Communication cost (Bytes)	Comp. to comm.	Scalability
Graph SAGE	$B \cdot (C + 3CF + 3LCF + (Wf1.y + Wa1.y) \cdot (CF + C + F + Wf2.y + Wa2.y))$	$8BC + 4B \cdot (F + Wf1.y + Wa1.y) \cdot min(C, 2p) + 4BCFL$	(2 + L + Wf1.y + Wa1.y) : 4L	poor
Feature in UVM				
Child-centric comp.	$BCF \cdot (2 + L + Wf1.y + Wa1.y)$	$4B \cdot (F + \text{Wf1}.y + \text{Wa1}.y) \cdot \text{min}(C, 2p) \text{ bytes}$ over GPU-GPU + $4BCLF$ bytes over GPU-CPU	$ \begin{aligned} &(2+L+\text{Wf1}.y+\\ &\text{Wa1}.y): 4L \text{ over} \\ &\text{GPU-CPU} \end{aligned} $	very poor
Graph SAGE	$B \cdot (C + 3CF + 3LCF + (Wf1.y + Wa1.y) \cdot (CF + C + F + Wf2.y + Wa2.y))$	$Wf1.y + Wa1.y) \cdot min(C, 2p)$ bytes over GPU-GPU +	$ \begin{aligned} &(2+L+\text{Wf1}.y+\\ &\text{Wa1}.y):4L \text{ over}\\ &\text{GPU-CPU} \end{aligned} $	very poor

JDO hacked version of above

Parts	Comp. cost
Feature duplication	
Children selection	BC
Child-centric comp.	$BCF \times (2 + L + Wf1.y +$
•	Wa1.y)
Source-centric comp.	$B \times (CF + (Wf1.y +$
•	Wa1.y) x $(C + F + Wf2.y)$
	+ Wa2.y)
Graph SAGE	$B \times (C + 3CF + 3LCF +$
•	$(Wf1.y + Wa1.y) \times (CF +$
	C + F + Wf2.y + Wa2.y)
Direct feature access	
Child-centric comp.	$BCF \times (2 + L + Wf1.v +$
•	Wa1.y)

Parts	Comp. cost
Graph SAGE	B x (C + 3CF + 3LCF + (Wf1.y + Wa1.y) x (CF + C + F + Wf2.y + Wa2.y))
Feature in UVM	
Child-centric comp.	$BCF \times (2 + L + Wf1.y + Wa1.y)$
Graph SAGE	$B \times (C + 3CF + 3LCF + (Wf1.y + Wa1.y) \times (CF + C + F + Wf2.y + Wa2.y))$

Parts	Comm. cost
Feature duplication	
Children selection	8BC bytes
Child-centric comp.	$4B \times (F + Wf1.y + Wa1.y) \times min(C, 2p)$ bytes
Source-centric comp.	0 bytes
Graph SAGE	$8BC + 4B \times (F + Wf1.y +$
	Wa1.y) x min(C, 2p) bytes
Direct feature access	
Child-centric comp.	$4B \times ((F + Wf1.y + Wa1.y) \times min(C, 2p) + CLF)$ bytes
Graph SAGE	$8BC + 4B \times (F + Wf1.y +$
	Wa1.y) x min(C, $2p$) + 4BCFL
	bytes
Feature in UVM	
Child-centric comp.	$4B \times (F + Wf1.y + Wa1.y) \times$
-	min(C, 2p) bytes over
	GPU- GPU
	+ 4BCLF bytes over GPU-CPU
Graph SAGE	$8BC + 4B \times (F + Wf1.y +$
	Wa1.y) x $min(C, 2p)$ bytes over
	GPU-GPU
	+ 4BCFL bytes over GPU-CPU

Parts Comp/comm ratio	Scalability
-----------------------	-------------

Feature duplication

Parts	Comp/comm ratio	Scalability
Children selection	1:8	Poor
Child-centric comp.	$\sim CF : min(C, 2p) \times 4$	Good
Source-centric comp.	N.A.	N.A.
Graph SAGE	at least \sim CF : min(C, 2p) x 4	Good
Direct feature access	(2 + I + W£1 + W ₂ 1) . 4I	
Child-centric comp. Graph SAGE	$\sim (2 + L + Wf1.y + Wa1.y) : 4L$ $\sim (2 + L + Wf1.y + Wa1.y) : 4L$	poor
Graph SAGE	$\sim (2 + L + WII.y + WaI.y)$: 4L	poor
$Feature\ in\ UVM$		
Child-centric comp.	$\sim (2 + L + Wf1.y + Wa1.y) : 4L \text{ over GPU-CPU}$	very poor
Graph SAGE	$\sim (2+L+Wf1.y+Wa1.y):$ 4L over GPU-CPU	very poor

Let's try a grid table with backslashes.

Fruit	Price	Advantages	
Bananas	first line next line	first line next line	
Bananas	first line next line	first line next line	