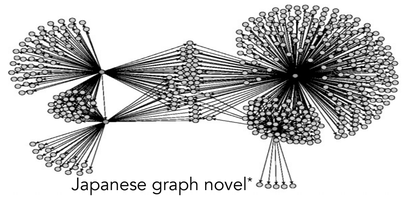
HBMax: Compress-to-Compute for Influence Maximization

Xinyu Chen (xinyu.chen1@wsu.edu), Marco Minutoli, Jiannan Tian, Mahantesh Halappanavar, Ananth Kalyanaraman, Dingwen Tao

WASHINGTON STATE UNIVERSITY

INTRODUCTION & MOTIVATION

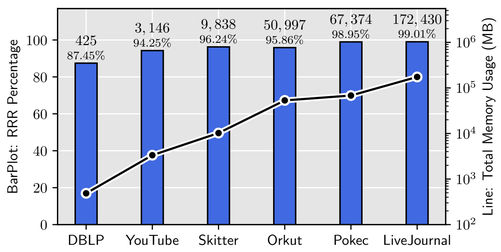
Influence Maximization studies the word-of-mouth effects in the viral marketing, politics, public health, bioinformatics and sensor networks.



Japanese graph novel*

It is an **NP-hard** optimization problem to activate the top-k vertices that can get maximal expected influence in graph *G*.

HBMax Parallel Influence Maximization algorithm using Huffman coding and Bitmap coding to address the memory inflation challenge.



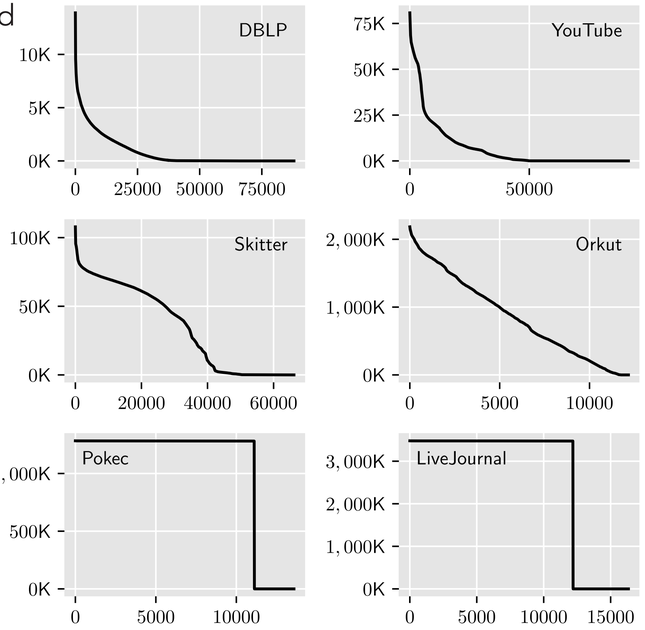
CONTRIBUTIONS

- **Characterize** memory footprints
- **Identify** various shapes of intermediate RRRs
- **Compress** with Huffman or Bitmap coding
- **Query** partially decoded or compressed data.
- **Reduces** memory usage up to 82.1%
- **Speedups** 6.3% (in average) than Ripples.

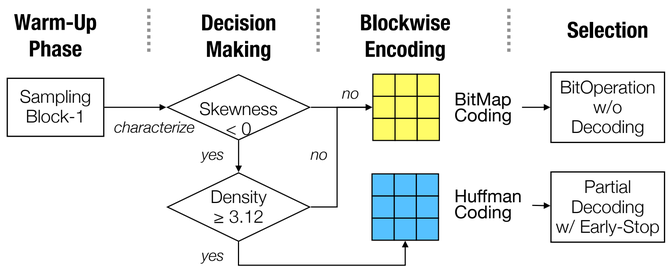
* The dynamics of viral marketing, Jure *et al.* 2007

PROFILING

- Skewness is from flat-headed (-1) to skewed(12).
- Density is from sparse (0.26%) to dense (53.3%)



WORKFLOW

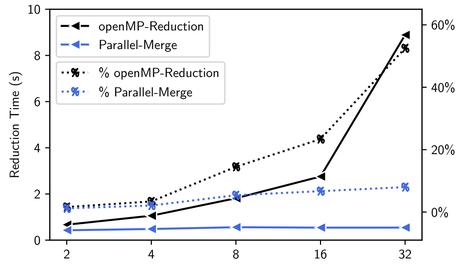


Block-wise workflow: warm-up, sampling-encoding, selection

https://arxiv.org/pdf/2208.00613.pdf

IMPLEMENTATION

- Parallelize by **OpenMP**
- Parallel Merge
- Consider **NUMA** effects
- Leverage **bit operations**



	R_1	R_2	R_3	R_4	R_5	R_6	R_7	$\hat{h}$
v_1	1	0	0	0	1	1	0	3
v_2	0	0	1	0	0	1	0	2
u^*	1	0	1	1	0	1	1	5
v_4	0	1	1	0	0	0	1	3
v_5	0	1	0	0	0	1	0	2

→

	R_1	R_2	R_3	R_4	R_5	R_6	R_7	$\hat{h}$
v_1	1	0	0	0	1	1	0	3
v_2	0	0	1	0	0	1	0	2
u^*	1	0	1	1	0	1	1	5
v_4	0	1	1	0	0	0	1	3
v_5	0	1	0	0	0	1	0	2

EVALUATION

Memory footprint (in GB) and reduction ratio (in parenthesis)

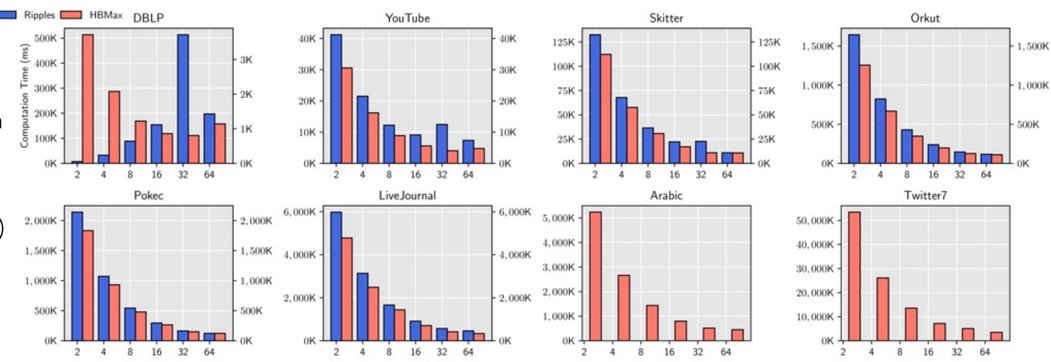
	DBLP	YouTube	Skitter	Orkut	Pokec	LiveJournal	Arabic	Twitter
Ripples	0.4 (1.0)	3.1 (1.0)	9.8 (1.0)	46.5 (1.0)	55.7 (1.0)	163.7 (1.0)	~~348.6 (1.0)~~	~~1193.0 (1.0)~~
HBMax	0.3 (1.3)	1.7 (1.8)	5.3 (1.9)	30.1 (1.5)	10.7 (5.2)	29.3 (5.6)	81.5 (4.3)	200.3 (5.9)

Time-to-Solution (in second) and overhead ratio (in parenthesis)

	DBLP	YouTube	Skitter	Orkut	Pokec	LiveJournal	Arabic	Twitter
Ripples	0.95 (1.0)	6.95 (1.0)	20.5 (1.0)	249.4 (1.0)	262.7 (1.0)	755.6 (1.0)	~~NA~~	~~NA~~
HBMax	1.10 (1.16)	6.31 (0.91)	17.9 (0.88)	235.1 (0.94)	222.6 (0.85)	692.7 (0.89)	1608.5	12098.3

SCALABILITY

- HBMax, Ripples both has strong scalability
- HBMax scales better (high-skewed graphs)
- Overall speedup is 12.98x on 64 cores



ACKNOWLEDGEMENT

The research is supported by the U.S. DOE Exascale Computing Project's (ECP) (17-SC-20-SC) ExaGraph codesign center at Pacific Northwest National Laboratory (PNNL) and by the NSF awards OAC-2034169, OAC-2042084, OAC-1910213, SHF-1919122, and CCF- 1815467 at Washington State University.

https://github.com/hipdac-lab/hbmax-pact

WASHINGTON STATE UNIVERSITY
Pacific Northwest NATIONAL LABORATORY
NSF