

# Efficient Partial-Pairs SimRank Search on Large Networks

Weiren Yu and Julie McCann


Department of Computing  
Imperial College London

## ➔ Overview


- Partial-Pairs SimRank search
  - Motivation
  - “High iteration coupling” problem
- Our solutions
  - “Seed germination” model
  - Backward pruning method
  - Extension to partial-pairs SimRank\*
- Experimental Evaluation
- Conclusions

# Overview


- SimRank in real-world applications:




Customers Who Bought This Item Also Bought




Nikon COOLPIX P510 16.1 MP CMOS Digital Camera with 42x Zoom NIKKOR ED Glass ...  
★★★★★ (418)  
\$299.00



Canon SX40 HS 12.1MP Digital Camera with 35x Wide Angle Optical Image Stabilized Zoom and ...  
★★★★★ (389)  
\$319.76




Sony Cyber-shot DSC-HX200V 18.2 MP Exmor R CMOS Digital Camera with 30x ...  
★★★★★ (216)  
\$348.00



Canon PowerShot SX500 IS 16.0 MP Digital Camera with 30x Wide-Angle Optical ...  
★★★★★ (86)  
\$249.00

## Recommender System



Hub | [ScienceDirect](#) | [Scopus](#) | [Applications](#)

Home | [Publications](#) | [Search](#) | [My settings](#) | [My alerts](#)

Articles  All fields  Author

Images  Journal/Book title  Volume  Issue  Page

8 [Evolution of trust networks in social web applications using supervised learning](#) Original Research Article  
*Procedia Computer Science*, Volume 3, 2011, Pages 833-839  
 Kiyana Zolfaghar, Abdollah Aghaie

[Show preview](#) | [PDF \(417 K\)](#) | [Related articles](#) | [Related reference](#)

## Citation Graph



## Collaboration Network

# SimRank Overview

- SimRank

- An appealing similarity measure based on graph structure
- Central idea:

Two nodes are similar if they are pointed to by similar nodes. *(recursion)*

Each node is most similar to itself. *(base case)*

- Two formulations of SimRank

- Jeh and Widom's form *(SIGKDD'02)*

$$s(a, b) = \begin{cases} 1 & (a = b) \\ \gamma \cdot \frac{\sum_{(i,j) \in N_a \times N_b} s(i,j)}{|N_a| |N_b|} & (a \neq b) \end{cases}$$

similarity btw. nodes  $a$  and  $b$

damping factor

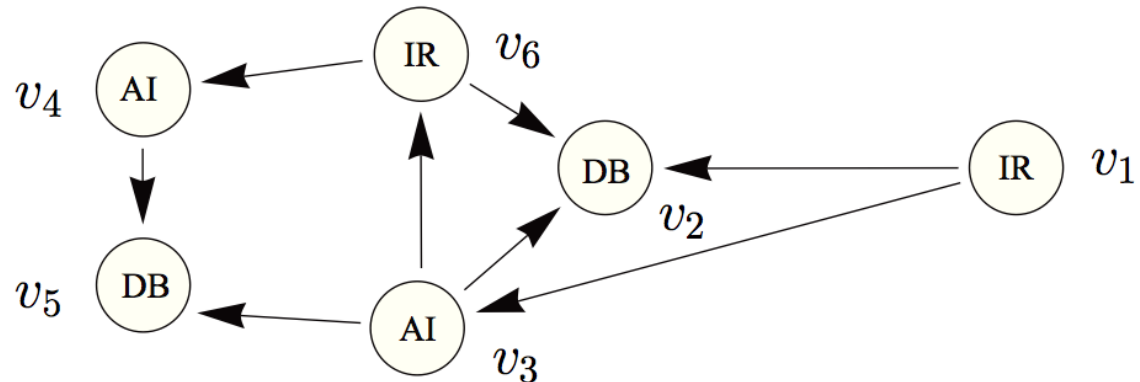
in-neighbor set of node  $b$

- Kusumoto et al.'s form *(SIGMOD'14)*

$$\mathbf{S} = \mathbf{C} \cdot \mathbf{W}^T \cdot \mathbf{S} \cdot \mathbf{W} + \mathbf{D}$$

# Motivation

- Example:



*Can we evaluate only the similarities of the papers between DB and AI areas?*

- Partial-Pairs SimRank Problem:

- **Given** a graph  $G(V,E)$ , a decay factor  $C$ , and two collections of nodes  $A$  and  $B$  in  $G$
- **Retrieve** partial-pairs scores  $\{s(x,y)\} \forall x \in A, \forall y \in B$

# Existing “high iteration coupling” barrier

$$S_{k+1} = C \cdot W^T \cdot S_k \cdot W + D$$

( $a \neq b$ )

The diagram illustrates the high iteration coupling barrier. It shows the calculation of a single element  $S_{k+1}(a,b)$  from the previous iteration's matrix  $S_k$ . The diagram consists of four blue boxes connected by an equals sign. The first box shows a matrix with a small blue square at row  $a$  and column  $b$ , labeled  $S_{k+1}(a,b)$ . The second box shows a matrix with a blue horizontal bar at row  $a$ , labeled  $W^T(a,*)$ . The third box is a solid blue square labeled  $S_k(*,*)$ . The fourth box shows a matrix with a blue vertical bar at column  $b$ , labeled  $W(*,b)$ . Dashed orange lines indicate the row  $a$  and column  $b$  across the matrices.

## High Iteration Coupling

- To retrieve a single-pair  $S_{k+1}(a,b)$ , all pairs of  $S_k(*,*)$  at the previous iteration need to be determined beforehand.



# Kusumoto et al.'s linearization

- Linearized SimRank model: (*SIGMOD'14*)

$$\mathbf{S} = C \times \mathbf{W}^T \mathbf{S} \mathbf{W} + \mathbf{D}$$

$$\Leftrightarrow \mathbf{S}(a,b) = \mathbf{e}_a^T \mathbf{D} \mathbf{e}_b + C (\mathbf{W} \mathbf{e}_a)^T \mathbf{D} (\mathbf{W} \mathbf{e}_b) + C^2 (\mathbf{W}^2 \mathbf{e}_a)^T \mathbf{D} (\mathbf{W}^2 \mathbf{e}_b) + \dots$$

## Complexity of Single-Pair SimRank

- Computing  $S_k(a,b)$  needs  $O(m)$  space and  $O(k^2m)$  time.  
( $m = |E|$ ,  $k = \#$  of iterations)



*If straightforwardly extended to partial-pairs case*

- $O(m)$  space and  $O(k^2|A||B|m)$  time would be required to compute  $\{S_k(x,y) \mid \forall x \in A, \forall b \in B\}$

**Can we do it better?**



# “Seed germination” model

$$S(*, j) = \mathbf{D}\mathbf{e}_j + C\mathbf{W}^T\mathbf{D}(\mathbf{W}\mathbf{e}_j) + C^2(\mathbf{W}^2)^T\mathbf{D}(\mathbf{W}^2\mathbf{e}_j) + C^3(\mathbf{W}^3)^T\mathbf{D}(\mathbf{W}^3\mathbf{e}_j) + \dots$$

- 3<sup>rd</sup> term:  $\mathbf{W}^T\mathbf{W}^T\mathbf{D}\mathbf{W}\mathbf{W}_{*,j}$
- 4<sup>th</sup> term:  $\mathbf{W}^T\mathbf{W}^T\mathbf{W}^T\mathbf{D}\mathbf{W}\mathbf{W}_{*,j}$

$$\mathbf{W}^T\mathbf{W}^T\mathbf{D}\mathbf{W}\mathbf{W}_{*,j}$$

$$\mathbf{W}^T\mathbf{W}^T\mathbf{W}^T\mathbf{D}\mathbf{W}\mathbf{W}_{*,j}$$

## Question

- How to keep “right-to-left association” while reducing duplicate computations across summands?

## Our solution

- “Seed germination” model to compute  $\{s(x,y)\}_{\forall x \in A, \forall y \in B}$   
 $O(km \min\{|A|, |B|\})$  time ;  $O(m+kn)$  memory

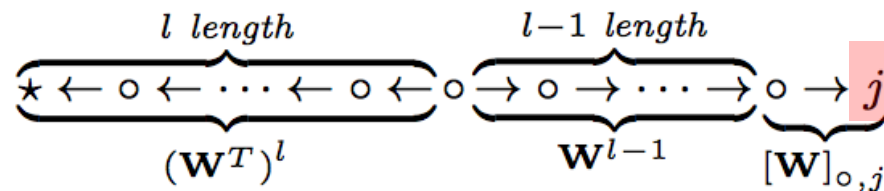
Existing:  $O(k^2m |A||B|)$  time ;  $O(m)$  memory



# Main Idea

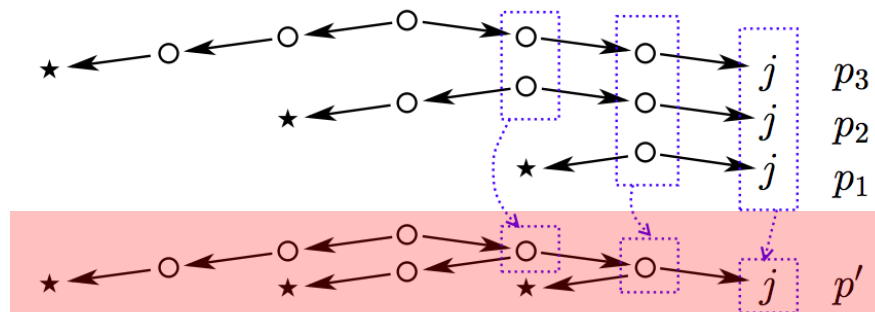
$$\mathbf{S}(*, j) = \mathbf{D}\mathbf{e}_j + C\mathbf{W}^T \mathbf{D}(\mathbf{W}\mathbf{e}_j) + C^2 (\mathbf{W}^2)^T \mathbf{D}(\mathbf{W}^2\mathbf{e}_j) + \dots$$

- $l$ -th term,  $(\mathbf{W}^l)^T \mathbf{D}(\mathbf{W}^l \mathbf{e}_j)$ , can tally the paths starting from node  $j$ :



## Phase 1

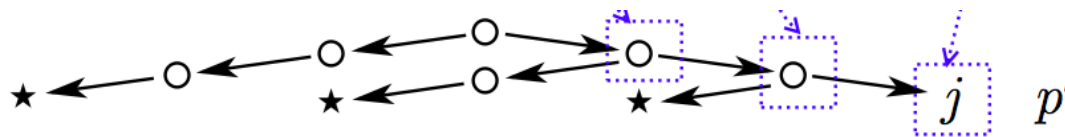
- merging the paths that are counted by  $l$ -th summands into a compact tree:



# Main idea (Cont.)

## Phase 2

- “Seed germination” search over the compact tree:



Paths Talled via “Seed Germination”	Step	Associated with Iterations
<div style="display: flex; align-items: center;"> <div style="border: 1px dashed red; border-radius: 50%; width: 20px; height: 20px; margin-right: 10px;"></div> <div style="margin-right: 10px;">“seed” nodes</div> </div> <div style="display: flex; align-items: center;"> <div style="border: 1px dashed blue; border-radius: 50%; width: 20px; height: 20px; margin-right: 10px;"></div> <div style="margin-right: 10px;">new “bud” nodes</div> </div> <div style="display: flex; align-items: center;"> <div style="border: 1px dashed green; border-radius: 50%; width: 20px; height: 20px; margin-right: 10px;"></div> <div style="margin-right: 10px;">old “germinated” nodes</div> </div>		$u_0 := e_j$ $u_1 := W \cdot e_j$
	2	$u_2 := W \cdot u_1 = W^2 \cdot e_j$
	3	$u_3 := W \cdot u_2 = W^3 \cdot e_j$ $v_0 := u_3$
	4	$v_1 := C \cdot W^T \cdot v_0 + u_2$ $= C \cdot W^T \cdot W^3 \cdot e_j + W^2 \cdot e_j$
	5	$v_2 := C \cdot W^T \cdot v_1 + u_1$ $= C \cdot (W^T)^2 \cdot W^3 \cdot e_j + W^T \cdot W^2 \cdot e_j + W \cdot e_j$
	6	$v_3 := C \cdot W^T \cdot v_2 + u_0$ $= C \cdot \sum_{l=0}^3 (W^T)^l \cdot W^l \cdot e_j$

# Separated Position

- Given two collections of nodes A and B, the term

$$[\mathbf{W}^T]_{A,\star} \cdot (\mathbf{W}^T)^{l-1} \cdot \mathbf{W}^{l-1} \cdot [\mathbf{W}]_{\star,B}$$

can be computed efficiently by grouping all the multiplications

- a) from “left- to-right” if  $|A| < |B|$ ;

$$\underbrace{\left( \left( \left( \left( [\mathbf{W}^T]_{A,\star} \cdot \mathbf{W}^T \right) \cdot \mathbf{W}^T \right) \dots \mathbf{W}^T \right) \cdot \mathbf{W} \right) \dots \mathbf{W}}_{l-1} \cdot [\mathbf{W}]_{\star,B}$$

- b) from “right-to-left” if  $|A| \geq |B|$ .

$$[\mathbf{W}^T]_{A,\star} \cdot \underbrace{\left( \mathbf{W}^T \dots \left( \mathbf{W}^T \cdot \left( \mathbf{W} \dots \left( \mathbf{W} \cdot \left( \mathbf{W} \cdot [\mathbf{W}]_{\star,B} \right) \right) \right) \right) \right)}_{l-1}$$

Minimum cost is attained when the “separated position”  $p$  is at end points

$$\underbrace{\left( \left( \left( [\mathbf{W}^T]_{A,\star} \cdot \mathbf{W}^T \right) \cdot \mathbf{W}^T \right) \dots \right)}_{p \text{ terms}} \cdot \underbrace{\left( \dots \left( \mathbf{W} \cdot \left( \mathbf{W} \cdot [\mathbf{W}]_{\star,B} \right) \right) \right)}_{(2l-p) \text{ terms}}$$

# Partial-Pairs Iteration Model

## Partial-Pairs SimRank Iteration

- Given two subsets A and B of nodes in V (assume  $|A| > |B|$ ), the partial-pairs SimRank at iteration k can be computed as

$$[\mathbf{S}_k]_{A,B} = C \times [\mathbf{W}^T]_{A,*} \mathbf{V}_{k-1} + \mathbf{I}_{A,B}$$

where

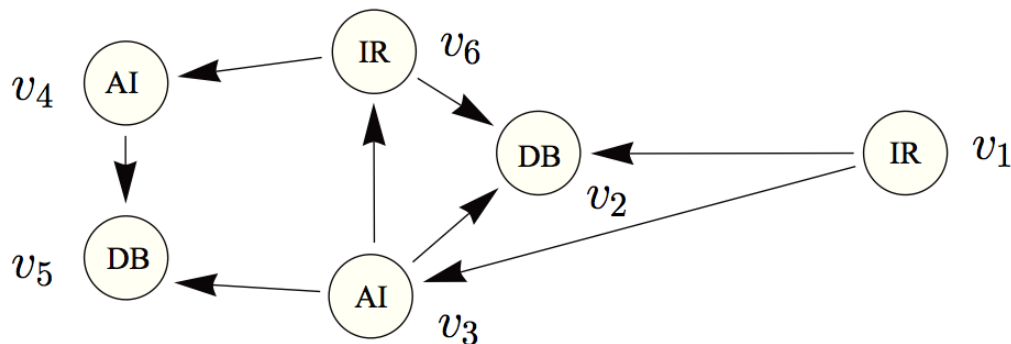
$$\begin{cases} \mathbf{V}_0 = \mathbf{D}\mathbf{U}_{k-l} \\ \mathbf{V}_l = C \cdot \mathbf{W}^T \mathbf{V}_{l-1} + \mathbf{U}_{k-l} \end{cases} \quad \text{and} \quad \begin{cases} \mathbf{U}_0 = \mathbf{I}_{*,B} \\ \mathbf{U}_l = \mathbf{W}\mathbf{U}_{l-1} \end{cases}$$

## Convergence

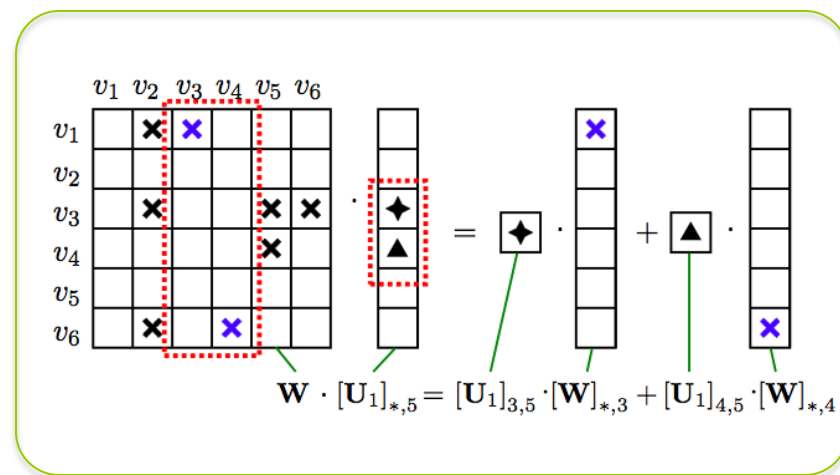
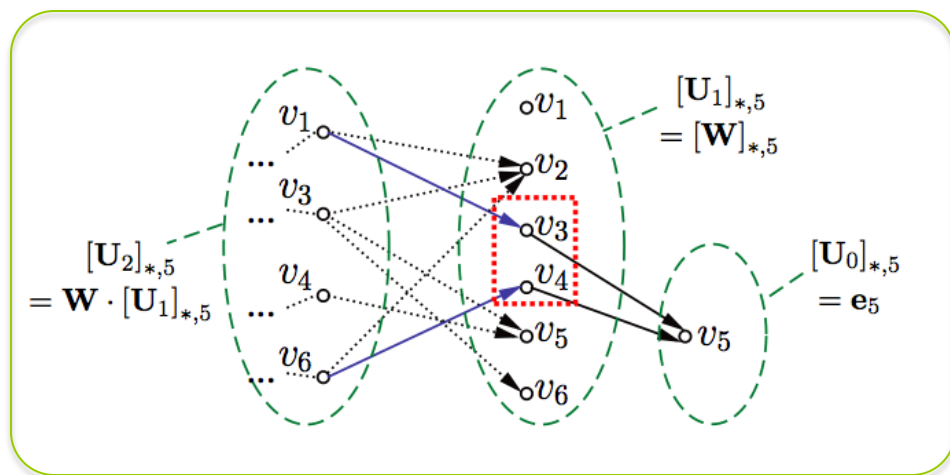
- For every iteration  $k=0,1,2, \dots$ ,

$$\|[\mathbf{S}_k]_{A,B} - [\mathbf{S}]_{A,B}\|_{\max} \leq C^{k+1}$$

# Eliminating Redundant Edge Access



- Unnecessary edge access in SpMxM can be pruned further.



**Edge Access: 8 → 2**

Time Complexity:

$$O(m \min\{|A|, |B|\}), \text{ with } m \leq \min\{k|E|, \Delta^{2k}\}$$

# Partial-pairs SimRank\*

$$\begin{aligned}
 [\tilde{\mathbf{S}}_3]_{*,j} &= (1 - C) \cdot (\mathbf{W}^T \cdot \mathbf{v}_2 + \mathbf{e}_j) \\
 &= (1 - C) \cdot \sum_{l=0}^3 \left(\frac{C}{2}\right)^l \cdot \sum_{\alpha=0}^l \binom{l}{\alpha} \cdot (\mathbf{W}^T)^{l-\alpha} \cdot \mathbf{W}^\alpha \cdot \mathbf{e}_j
 \end{aligned}$$

$\alpha$	$l$	update $\{\mathbf{u}_{2+\alpha-l}\}_{0 < \alpha < l < 2}$
0	0	$\mathbf{u}_2 := \mathbf{u}_2 + \mathbf{W} \cdot \mathbf{u}_3 = \left( \left(\frac{C}{2}\right)^2 \mathbf{I} + \left(\frac{C}{2}\right)^3 \mathbf{W} \right) \mathbf{e}_j$
	1	$\mathbf{u}_1 := \mathbf{u}_1 + \mathbf{W} \cdot \mathbf{u}_2 = \left( \left(\frac{C}{2}\right) \mathbf{I} + \left(\frac{C}{2}\right)^2 \mathbf{W} + \left(\frac{C}{2}\right)^3 \mathbf{W}^2 \right) \mathbf{e}_j$
	2	$\mathbf{u}_0 := \mathbf{u}_0 + \mathbf{W} \cdot \mathbf{u}_1 = \left( \mathbf{I} + \left(\frac{C}{2}\right) \mathbf{W} + \left(\frac{C}{2}\right)^2 \mathbf{W}^2 + \left(\frac{C}{2}\right)^3 \mathbf{W}^3 \right) \mathbf{e}_j$
1	1	$\mathbf{u}_2 := \mathbf{u}_2 + \mathbf{W} \cdot \mathbf{u}_3 = \left( \left(\frac{C}{2}\right)^2 \mathbf{I} + 2\left(\frac{C}{2}\right)^3 \mathbf{W} \right) \mathbf{e}_j$
	2	$\mathbf{u}_1 := \mathbf{u}_1 + \mathbf{W} \cdot \mathbf{u}_2 = \left( \left(\frac{C}{2}\right) \mathbf{I} + 2\left(\frac{C}{2}\right)^2 \mathbf{W} + 3\left(\frac{C}{2}\right)^3 \mathbf{W}^2 \right) \mathbf{e}_j$
2	2	$\mathbf{u}_2 := \mathbf{u}_2 + \mathbf{W} \cdot \mathbf{u}_3 = \left( \left(\frac{C}{2}\right)^2 \mathbf{I} + 3\left(\frac{C}{2}\right)^3 \mathbf{W} \right) \mathbf{e}_j$

$l$	update $\{\mathbf{v}_l\}_{0 < l < 2}$
0	$\mathbf{v}_0 := \mathbf{u}_3 = \left(\frac{C}{2}\right)^3 \mathbf{e}_j$
1	$\mathbf{v}_1 := \mathbf{W}^T \cdot \mathbf{v}_0 + \mathbf{u}_2 = \left( \left(\frac{C}{2}\right)^3 \mathbf{W}^T + 3\left(\frac{C}{2}\right)^3 \mathbf{W} + \left(\frac{C}{2}\right)^2 \mathbf{I} \right) \mathbf{e}_j$
2	$\mathbf{v}_2 := \mathbf{W}^T \cdot \mathbf{v}_1 + \mathbf{u}_1 = \left( \left(\frac{C}{2}\right)^3 (\mathbf{W}^T)^2 + 3\left(\frac{C}{2}\right)^3 \mathbf{W}^T \mathbf{W} + 3\left(\frac{C}{2}\right)^3 \mathbf{W}^2 + \left(\frac{C}{2}\right)^2 \mathbf{W}^T + 2\left(\frac{C}{2}\right)^2 \mathbf{W} + \left(\frac{C}{2}\right) \mathbf{I} \right) \mathbf{e}_j$

# Experimental Settings

- Datasets

- Real-life Data:

Data	$ G $ ( $ V $ , $ E $ )	$d$	Data	$ G $ ( $ V $ , $ E $ )	$d$
P2P	27.1K (6.3K, 20.8K)	3.3	AM	3.8M (403K, 3.4M)	8.4
DBLP	49.5K (13.2K, 36.3K)	2.7	CitP	20.3M (3.8M, 16.5M)	4.4
WebS	2.6M (282K, 2.3M)	8.2	SocL	73.8M (4.8M, 69.0M)	14.2

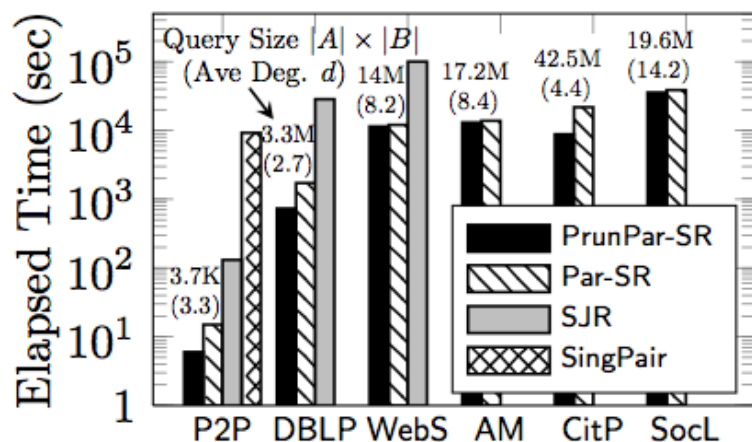
- Synthetic Data: GraphGen generator

- Compared Algorithms

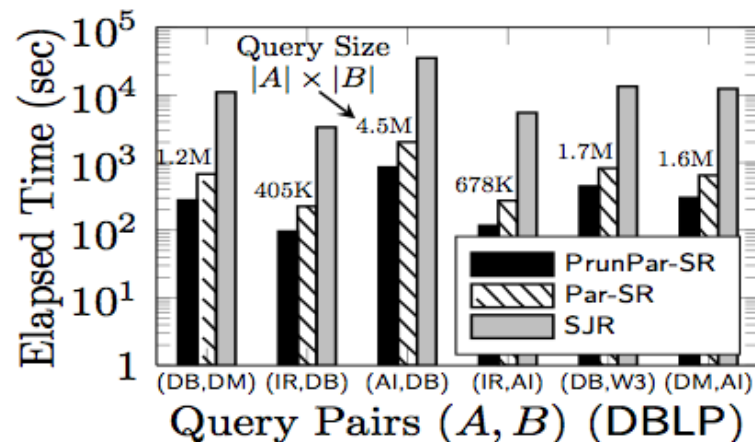
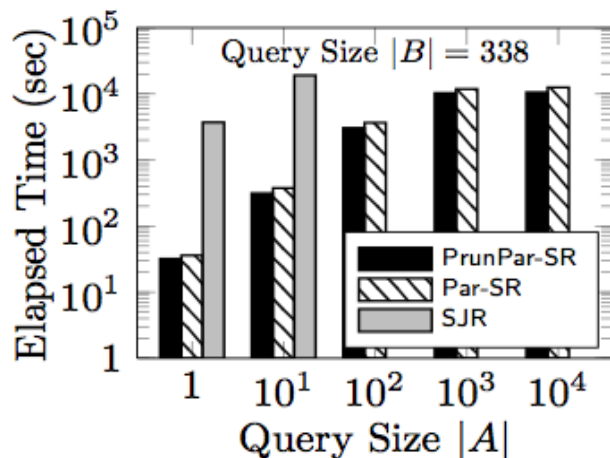
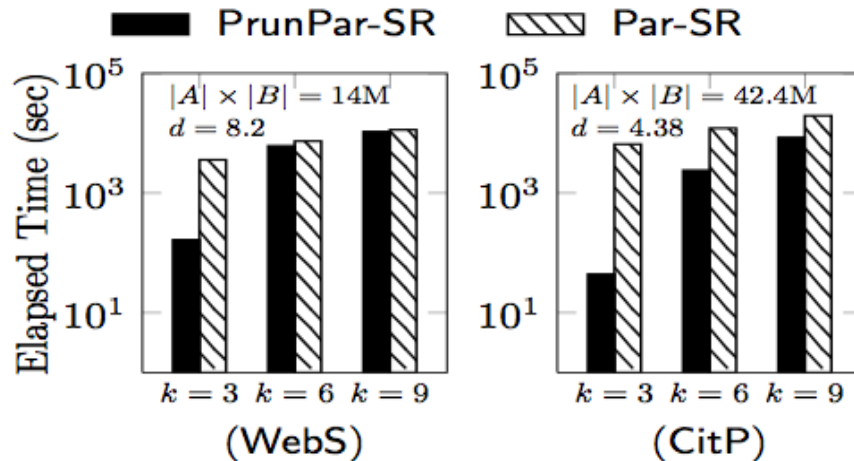
Algorithm	Description	Type
PrunPar-SR	our algorithm in Sect. 3.2, with pruning	partial pairs
Par-SR	our algorithm in Sect. 3.1, without pruning	
PrunPar-SR*	variation of PrunPar-SR ported to SimRank*	
SJR	SimRank-based similarity join [20]	
TopSim-SM	top-K random walk based SimRank [10]	single source
SimMat	top-K matrix-based SimRank [4]	
Psum	partial sum memoization SimRank [13]	all pairs
OIP	fine-grained memoization SimRank [16]	
Psum-SR*	partial sum memoization SimRank* [18]	
Memo-SR*	edge concentration SimRank* [18]	



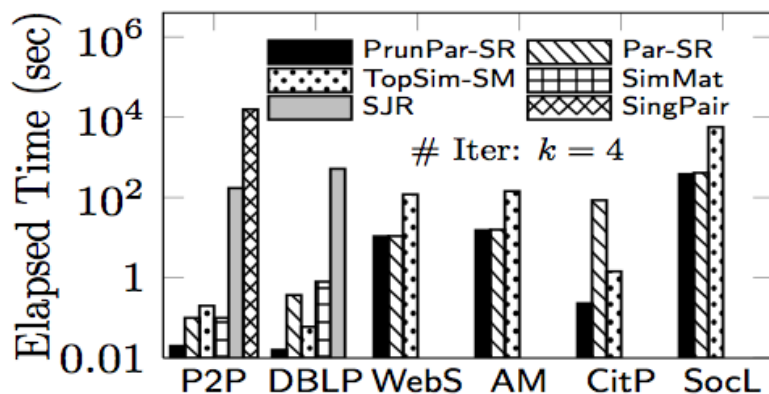
# Exp-1 Computational Time



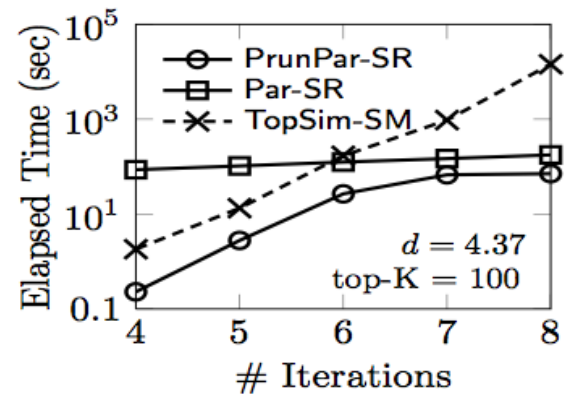
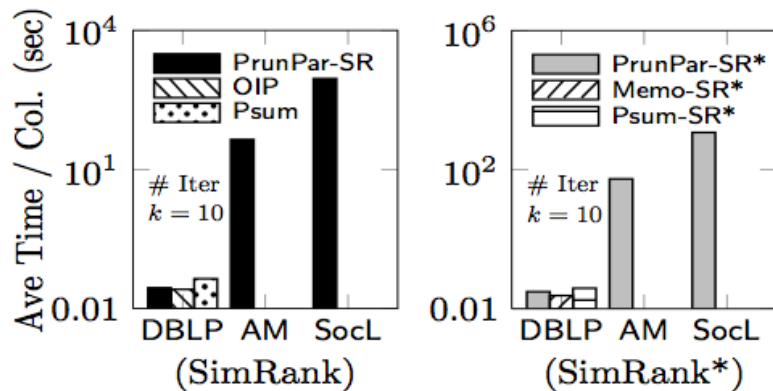
(a) Time on Real Data

(b) Vary  $(A, B)$  on DBLP(c) Vary  $|A|$  on WebS(d) Vary  $k$  on WebS and CitP

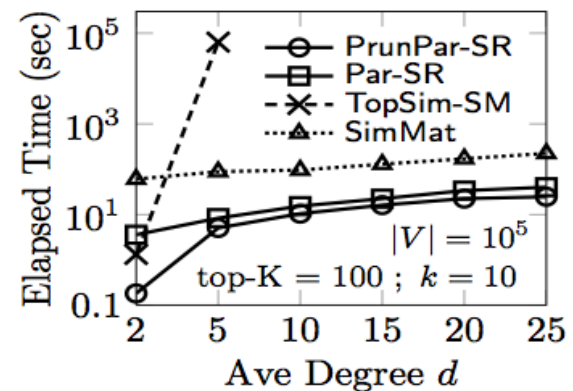
# Exp-2 Time w.r.t. query size, $k$ , $d$



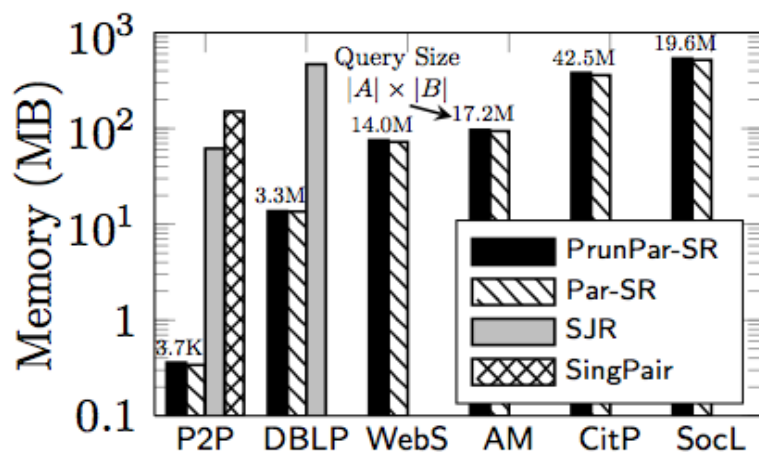
(e) Time for SS (Single Source)

(f) Vary  $k$  on CitP for SS

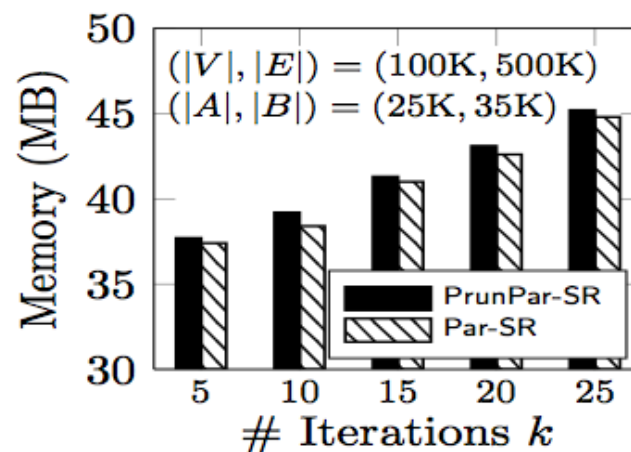
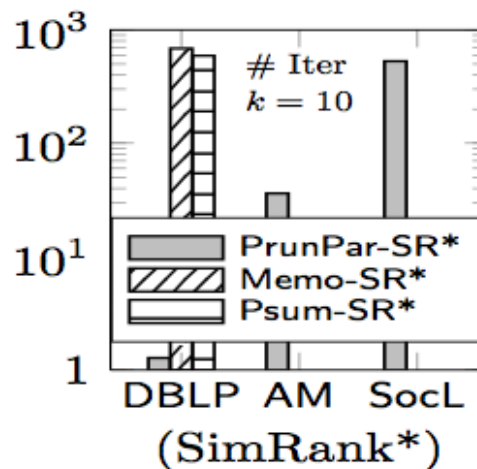
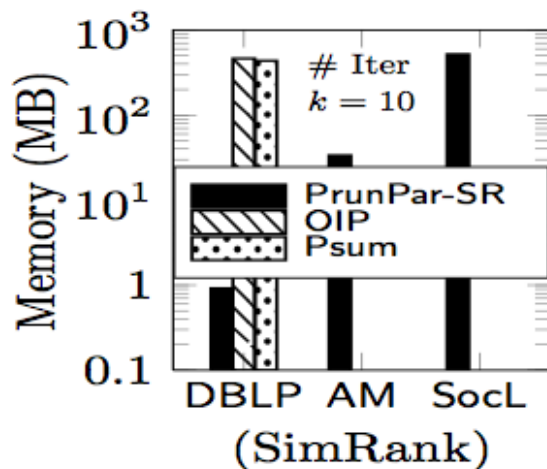
(g) Ave Time per Col for All Pairs

(h) Vary  $d$  on SYN for SS

# Exp-3 Memory Usage

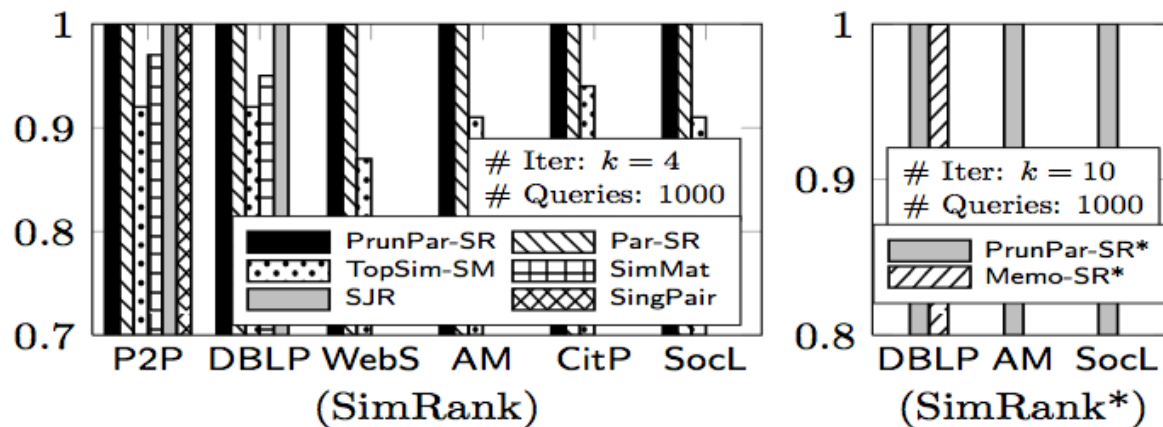


(j) Memory on Real Data

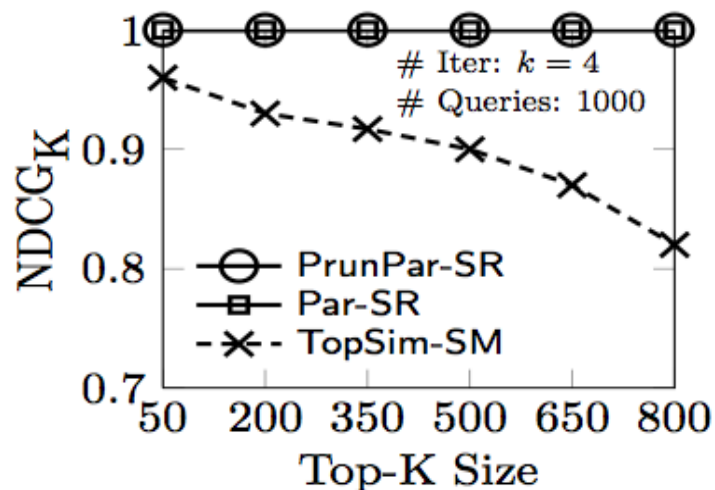
(k) Memory *vs.*  $k$  on SYN

(m) Memory on Real Data for All Pairs

# Exp-4 Accuracy & Exactness



(n) Accuracy on Real Data



(o) Accuracy *vs.* Top-K Size

# In Conclusion

- We have proposed efficient techniques for partial-pairs SimRank evaluation:
  - Design a “seed germination” model that can achieve  $O(k|E| \min\{|A|, |B|\})$  time and  $O(|E|+k|V|)$  memory
  - Devise an effective backward pruning method to speed up the time to  $O(m \min\{|A|, |B|\})$ , with  $m \leq \min\{k|E|, \Delta^{2k}\}$
  - Extend our method to other similarity measures to evaluate their partial-pairs scores



A stage with blue curtains and a wooden floor. The text "Thank you!" and "Q/A" is displayed in the center.

***Thank you!***

***Q/A***