Parallel Evaluation of Multi-Semi-Joins

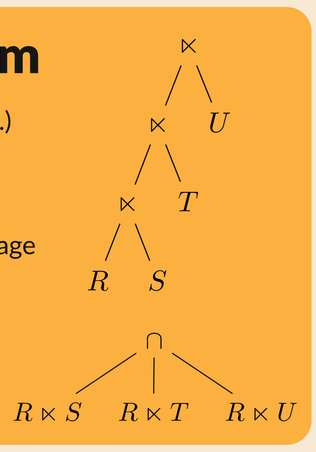
Jonny Daenen — Hasselt University
Frank Neven — Hasselt University
Tony Tan — National Taiwan University
Stijn Vansummeren — Université Libre de Bruxelles

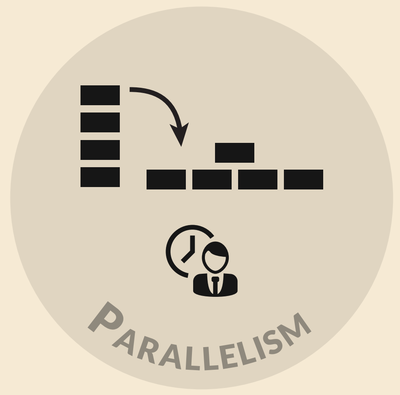
Problem

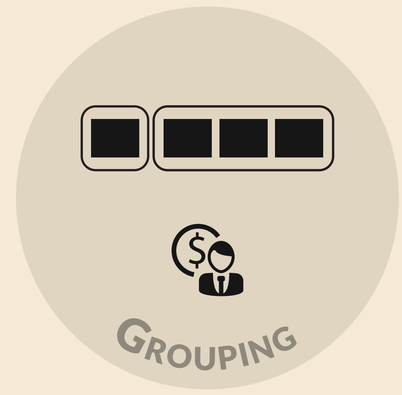
Cloud (AWS, GCP, etc.)
Pay-as-you-go

Keep low time
Minimize resource usage

SGF Queries
Parallel queries
Minimize total time







Semi-Joins

$$R \ltimes S = \pi_{R.*}(R \bowtie S)$$

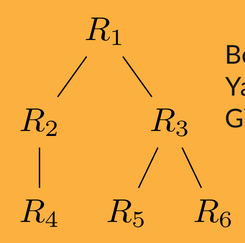
$$R \bowtie S = (R \ltimes S) \bowtie (S \ltimes R)$$

0	1
1	2
8	1

1	2
1	4
3	8

Semi-Join Reducers

Reduce Communication Overhead



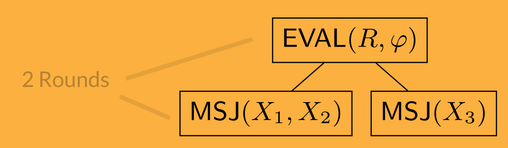
Bernstein et al. 1981
Yannakakis 1981
GYM (Afrati et al. 2014)

BSGF Queries

```
SELECT x,y
FROM R(x,y,z)
WHERE
      S(x,y) AND S(y,x) OR NOT T(x,w)
```

1 Guard Atom
Boolean Combination
Conditional Atoms

```
X1 = SELECT x,y,z FROM R(x,y,z) WHERE S(x,y)
X2 = SELECT x,y,z FROM R(x,y,z) WHERE S(y,x)
X3 = SELECT x,y,z FROM R(x,y,z) WHERE T(x,w)
```



SGF Queries

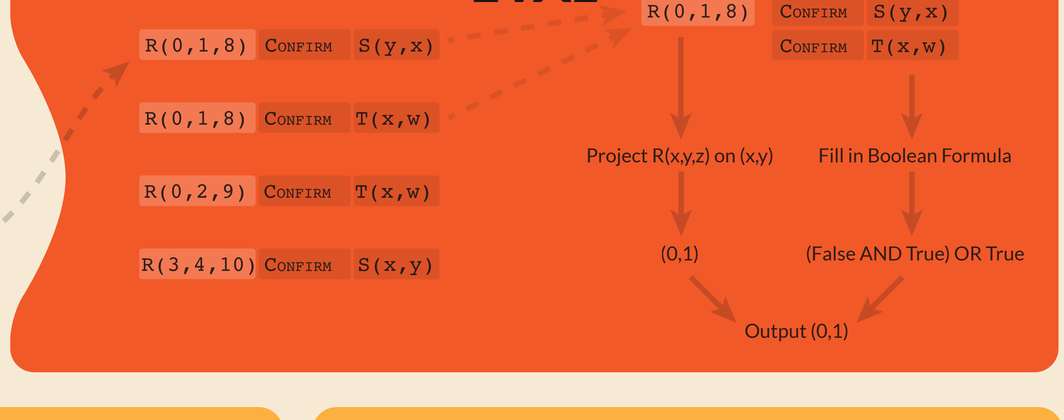
Strictly Guarded Fragment
Nesting
GF ~ Semi-Join Algebra
Stricly = no top-level Boolean combinations

```
Z1 = SELECT aut FROM Amazon(title,aut,'bad')
     WHERE BN(title,aut,'bad') AND
           BD(title,aut,'bad')

Z2 = SELECT newttl FROM Upcoming(newttl,aut)
     WHERE NOT Z1(aut)
```

MSJ

EVAL



Cost Model

$$\underbrace{\underbrace{\text{read} \to \text{map} \to \text{sort} \to \text{merge}}_{\text{Map Phase}} \to \underbrace{\text{trans.} \to \text{merge} \to \text{reduce} \to \text{write}}_{\text{Reduce Phase}}}_{\text{Input} \qquad \text{Intermediate Data} \qquad \text{Output}}$$

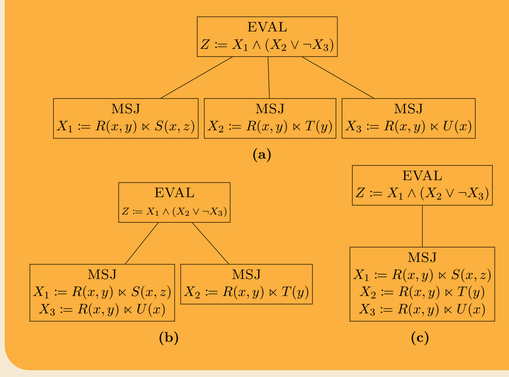
$$cost_{map}(N_i, M_i) = h_r N_i + \text{merge}_{map}(M_i) + l_w M_i,$$

$$cost_{red}(M, K) = tM + \text{merge}_{red}(M) + h_w K,$$

$$cost_h + \sum_{i=1}^{k} cost_{map}(N_i, M_i) + cost_{red}(M, K),$$

Non-uniform input contribution

BSGF-OPT

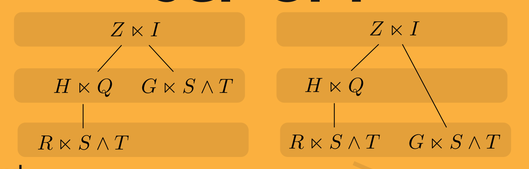


NP-hard
(Nykiel et al. 2010)

Greedy:

	X1	X2	X3
X1	0	+200	+300
X2		0	-100
X3			0

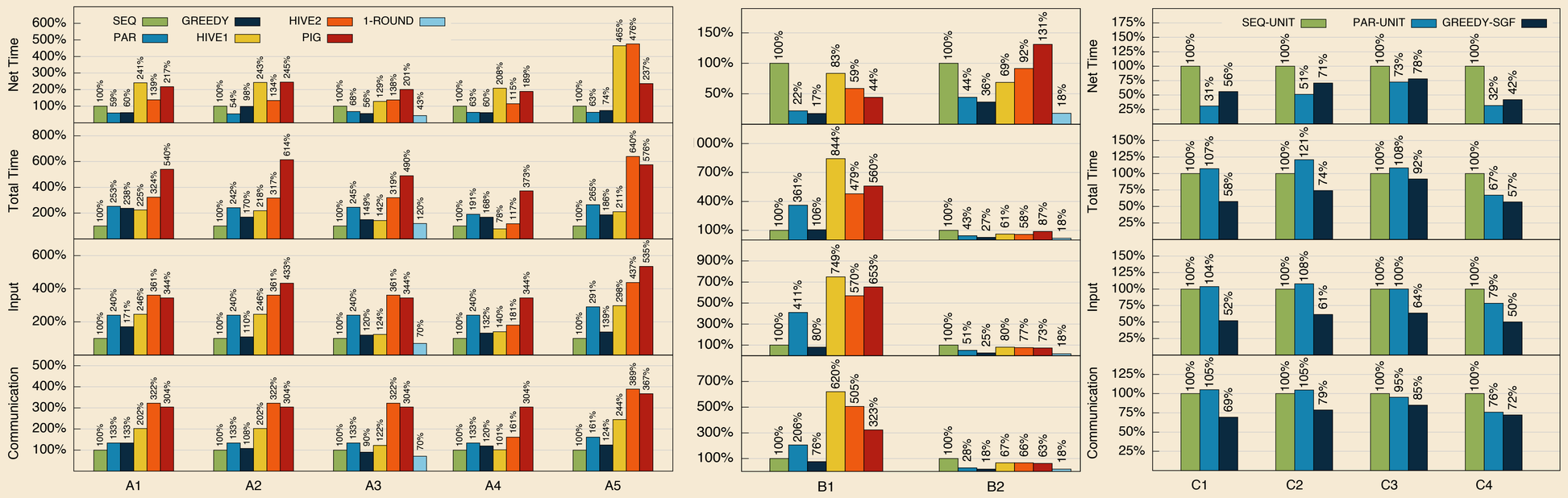
SGF-OPT



NP-hard
Greedy:

```
for each leaf
  find group with highest overlap +dependencies
if no overlap, add one query to new group
else add a max. overlap query to group
remove leaf
```

QID	Query
A1	$R(x,y,z,w) \ltimes S(x) \wedge T(y) \wedge U(z) \wedge V(w)$
A2	$R(x,y,z,w) \ltimes S(x) \wedge S(y) \wedge S(z) \wedge S(w)$
A3	$R(x,y,z,w) \ltimes S(x) \wedge T(x) \wedge U(x) \wedge V(x)$
A4	$R(x,y,z,w) \ltimes S(x) \wedge T(y) \wedge U(z) \wedge V(w)$ $G(x,y,z,w) \ltimes W(x) \wedge X(y) \wedge Y(z) \wedge Z(w)$
A5	$R(x,y,z,w) \ltimes S(x) \wedge T(y) \wedge U(z) \wedge V(w)$ $G(x,y,z,w) \ltimes S(x) \wedge T(y) \wedge U(z) \wedge V(w)$
B1	$R(x,y,z,w) \ltimes S(x) \wedge T(x) \wedge U(x) \wedge V(x) \wedge S(y) \wedge T(y) \wedge U(y) \wedge V(y) \wedge S(z) \wedge T(z) \wedge U(z) \wedge V(z) \wedge S(w) \wedge T(w) \wedge U(w) \wedge V(w)$
B2	$R(x,y,z,w) \ltimes (S(x) \wedge \neg T(x) \wedge \neg U(x) \wedge \neg V(x)) \vee (\neg S(x) \wedge T(x) \wedge \neg U(x) \wedge \neg V(x)) \vee (S(x) \wedge \neg T(x) \wedge U(x) \wedge \neg V(x)) \vee (\neg S(x) \wedge \neg T(x) \wedge \neg U(x) \wedge V(x))$



Gumbo

Packing

```
0,1 Request S(x,y) R(0,1,8)
0,1 Request T(x,y) R(0,1,8)
        ↓
0,1 Request S(x,y) T(x,y) R(0,1,8)
```

Tuple IDs

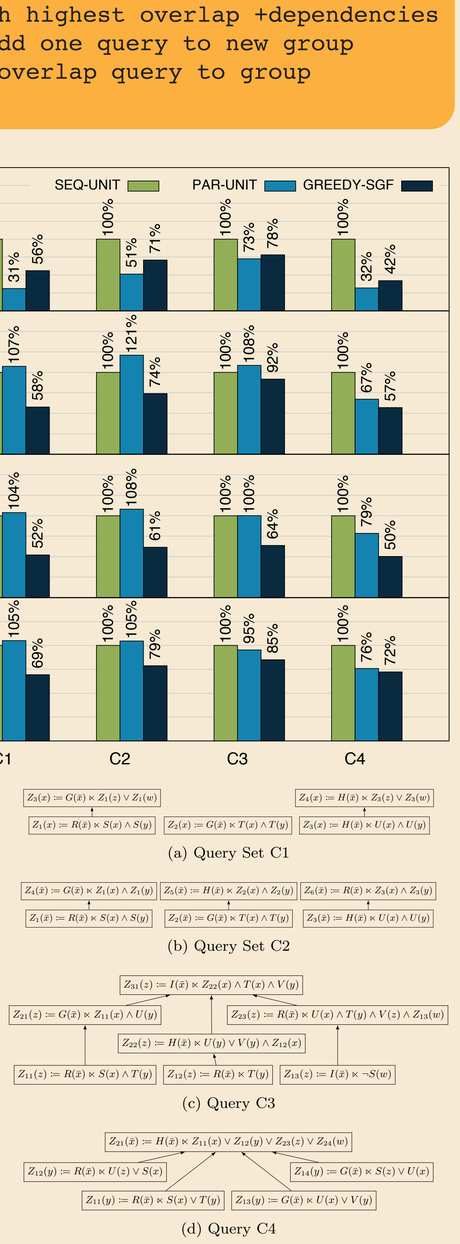
```
0,1 Request S(x,y) R(0,1,8)
        ↓
0,1 T1 A1 #0003
```

github.com/JonnyDaenen/Gumbo

- Map Output Estimation
- MR Cost Model
- 1-Round
- Streaming Reducers
- Reducer Shaping
- Map Shaping
- Multi-Query
- Confirm Reduction

Conclusion

- Parallel MR Query Plan: low net time
- Single Semi-Join: 1 round
- (multi-)BSGF NP-hard
- (multi-)SGF NP-hard
- Greedy Grouping: low total time
- Plug into Pig/Hive/...?
- General MR Optimizations



Poster presentation VLDB 2016, New Delhi, India

universiteit hasselt — KNOWLEDGE IN ACTION
國立臺灣大學 National Taiwan University
ULB UNIVERSITÉ LIBRE DE BRUXELLES