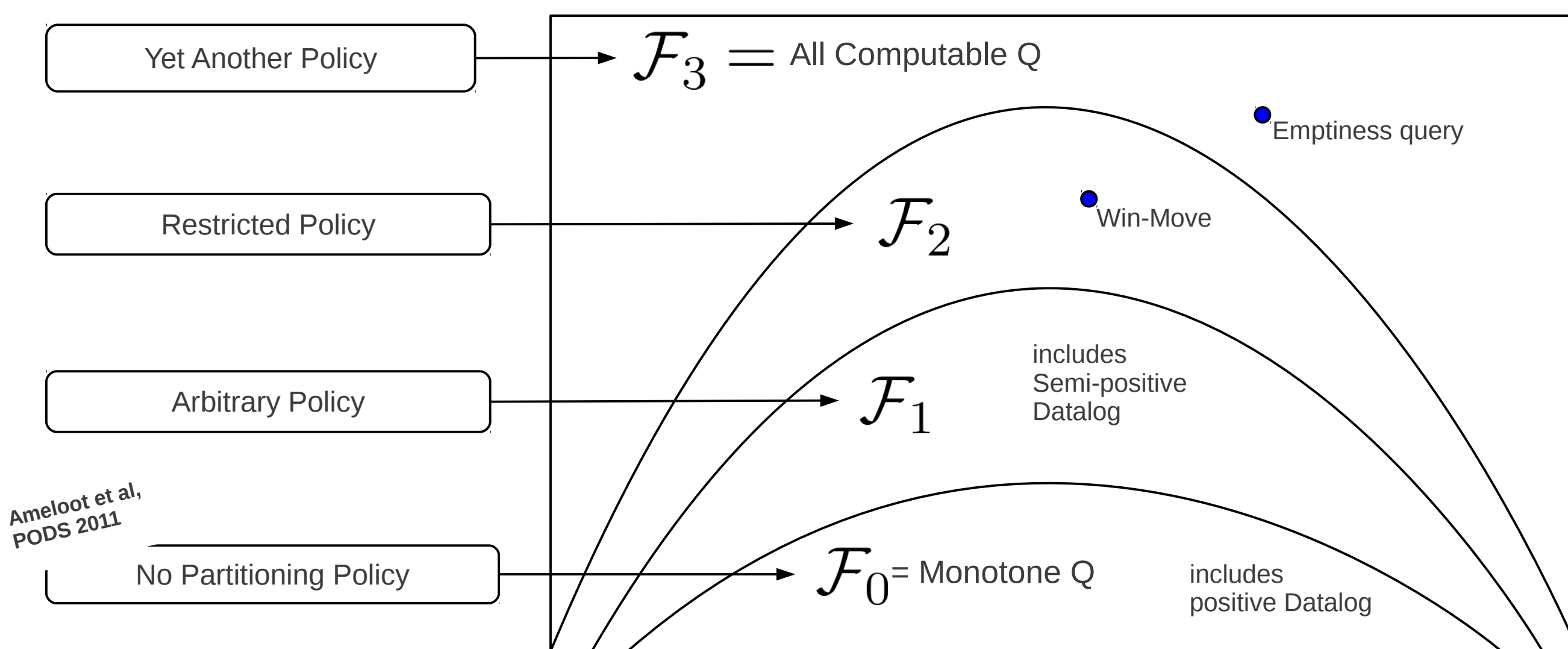


Main Contributions

- Parameterized notions of coordination-freeness
- Corresponding query classes form strict hierarchy



Coordination-Freeness and Weakened Forms of Monotonicity

Q is monotone iff $Q(I) \subseteq Q(I \cup \{x\})$ for all I and x .	\mathcal{F}_0
Q is adom-monotone iff $Q(I) \subseteq Q(I \cup \{x\})$ for all I and x with $adom(x) \cap adom(I) \neq \emptyset$.	\mathcal{F}_1
Q is weak-adom-monotone iff $Q(I) \subseteq Q(I \cup \{x\})$ for all I and x for which for all $i: a_i \notin adom(I)$ with $x = R(a_1, \dots, a_n)$.	\mathcal{F}_2

Datalog Stratum Boundaries = Coordination Barriers?

Stratum-Boundary $A \leftarrow B, \neg C.$ Coordination-Barrier?!

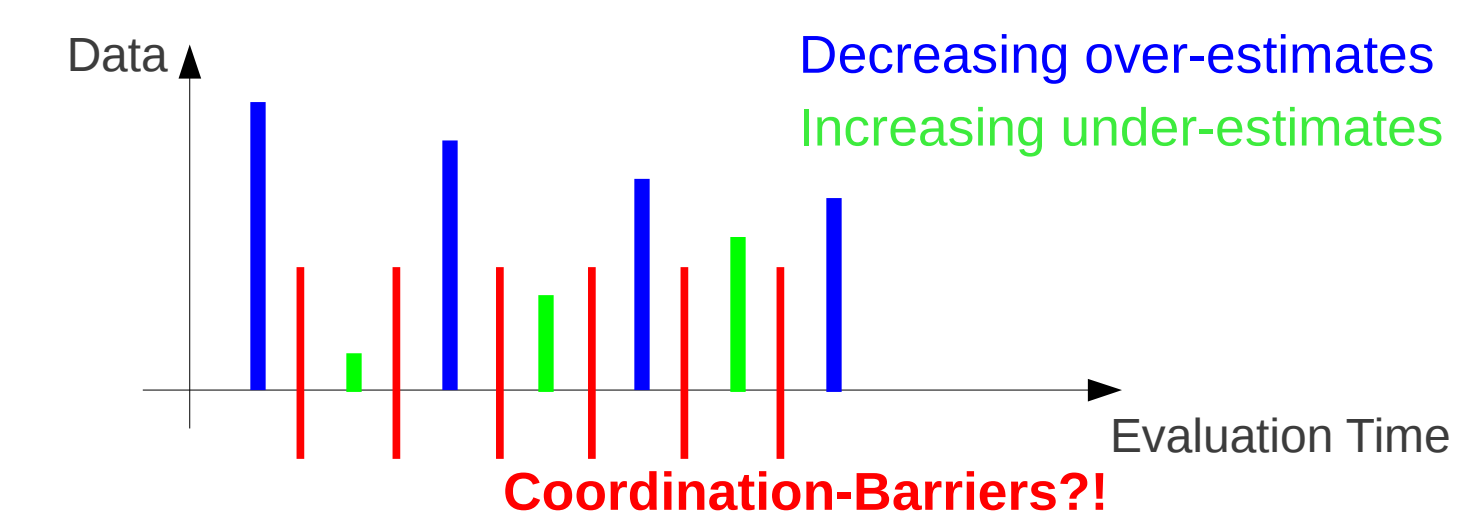
Stratum-Boundary $C \leftarrow D, \neg E.$ Coordination-Barrier?!

$E \leftarrow F, G.$

Number of strata = # coordination barriers?!

$win(X) \leftarrow move(X, Y), \neg win(Y).$ Non-stratified

Alternating fixpoint procedure:



Non-stratified => unbounded # boundaries?!

Joe Hellerstein, PODS 2010 Keynote

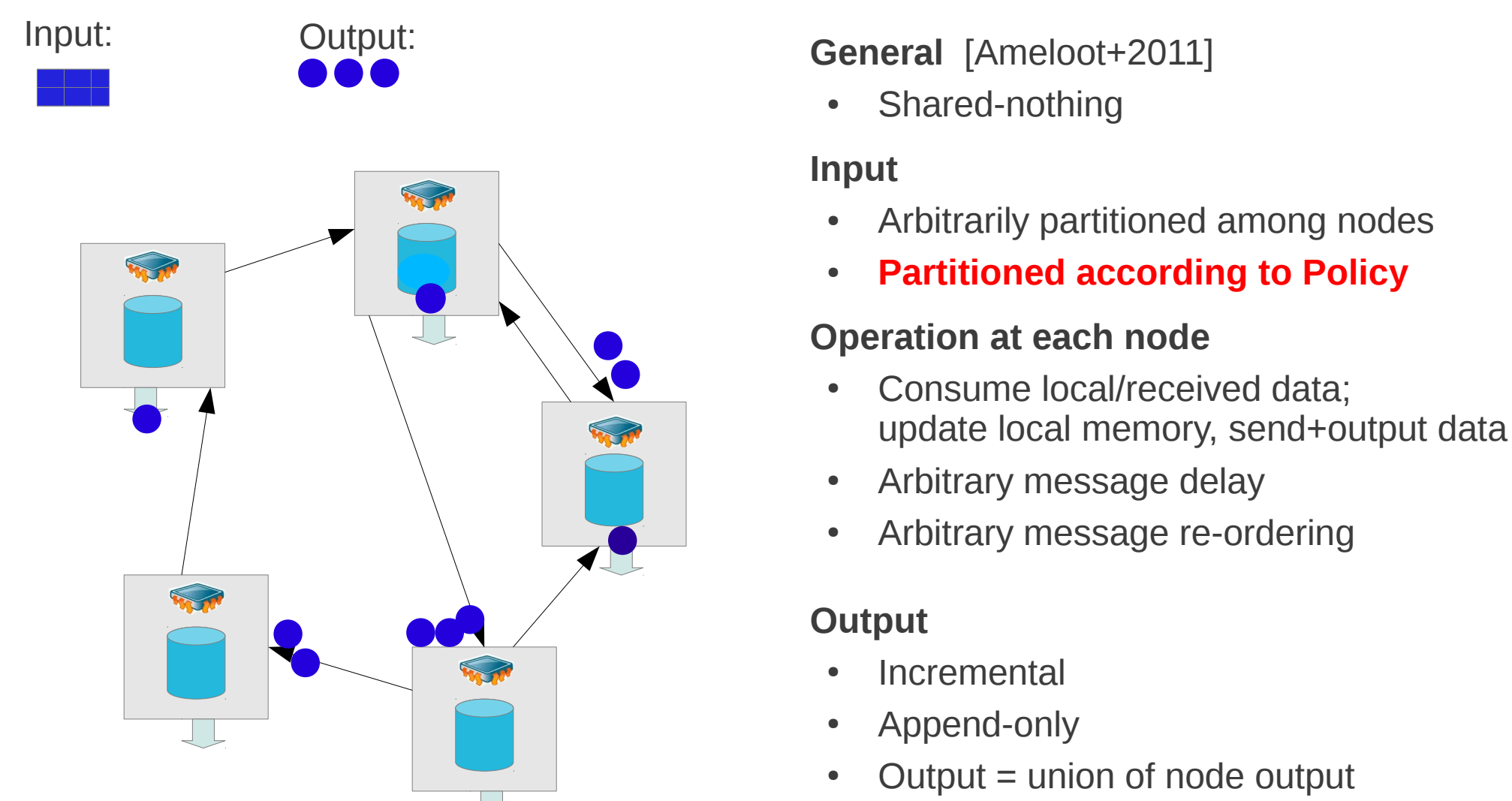
CONJECTURE 1. Consistency And Logical Monotonicity (CALM).
A program has an eventually consistent, coordination-free evaluation strategy iff it is expressible in (monotonic) Datalog.

Ameloot et al, PODS 2011

Corollary 13 (CALM Property). The following are equivalent for any query Q:

- Q can be distributedly computed by a transducer that is coordination-free.
- Q can be distributedly computed by a transducer that is oblivious.
- Q is monotone.

Distributed System Model: Networks of Relational Transducers



Require: Eventually correct result despite non-determinism.

A Parametrized Definition of Coordination-Freeness

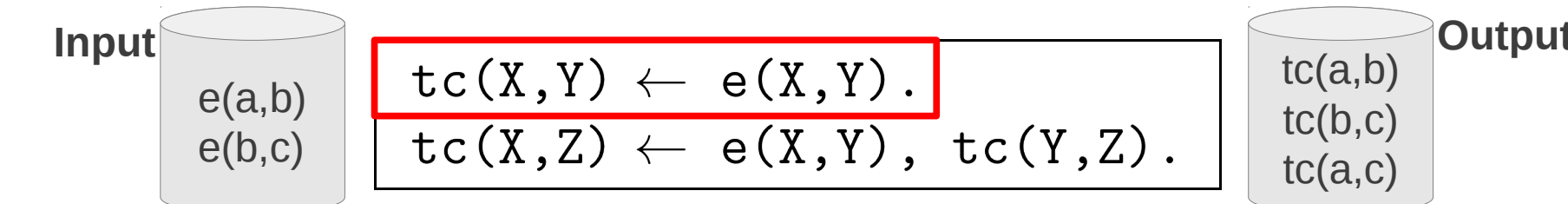
Ameloot et al, PODS 2011

Call a relational transducer **coordination-free** if for every network and input, can produce *complete answer* for *some partitioning** of input data, even when all network communication is blocked.

Why not just "partition all data to one node"? Node does not know it has all data (needs communication to find out).

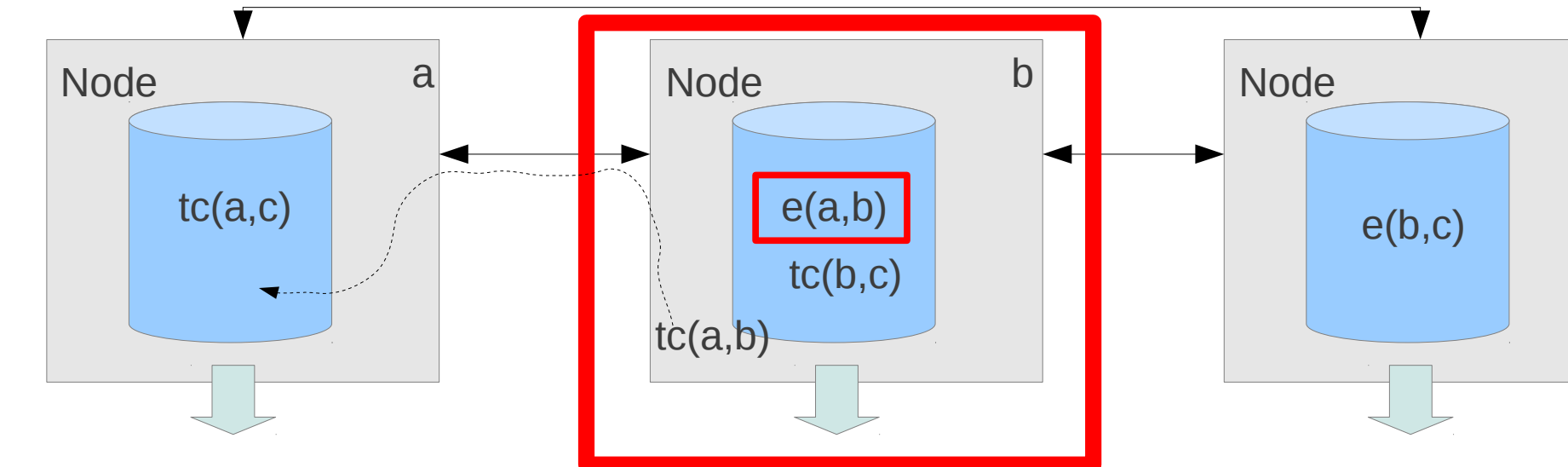
* knob: Distribution according to a Policy? What kind of Policy?

Warmup: Distributed Evaluation of Positive Datalog



Data Distribution – Partitioning Policy

- Partition e on second attribute
- Partition tc on first attribute

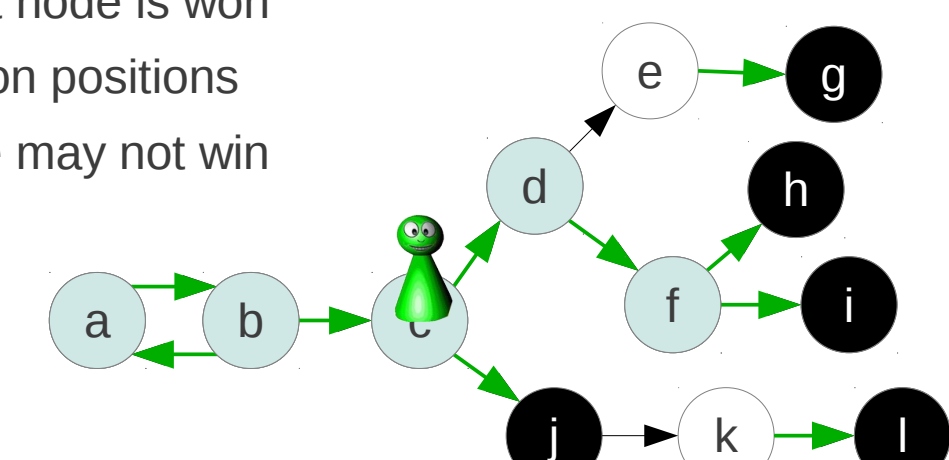


Disorderly Evaluation of Win-Move



Keep track of good moves and "may-win" nodes

- Initially, any move is good, and any node with move out may win
- Node with good move to lost node is won
- Remove "good" moves to won positions
- A node without a good move may not win



Via Datalog?

$won(X) \leftarrow move(X, Y), \neg may_win(Y).$
 $\neg good_move(X, Y) \leftarrow move(X, Y), won(Y).$
 $\neg may_win(X) \leftarrow \forall Y \neg good_move(X, Y).$

Disorderly OK here!

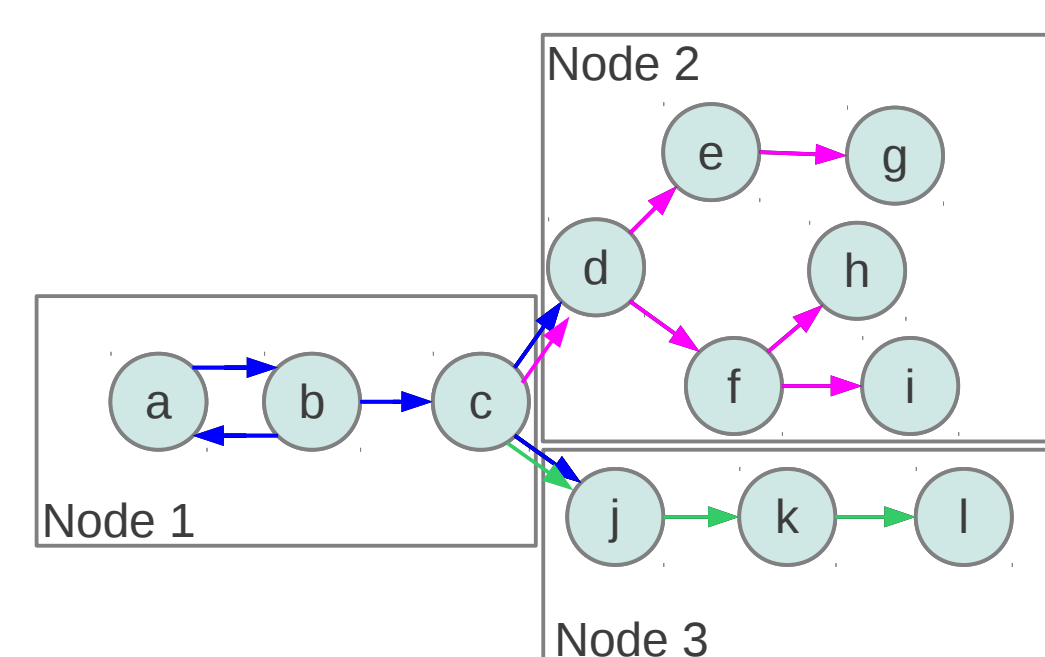
Semi-Monotone Datalog

Def. P is **semi-monotone** if relation names in $idb(P)$ occur only positively, and relation names in $eidb(P)$ occur only negatively.

$won(X) \leftarrow move(X, Y), \neg may_win(Y).$
 $\neg good_move(X, Y) \leftarrow move(X, Y), won(Y).$
 $\neg may_win(X) \leftarrow \forall Y \neg good_move(X, Y).$

Thm. Semi-monotone programs are eventually deterministic.

Element-Determined Partitioning Policies



- Policy = arbitrary mapping from **domain elements** to nodes
- Input facts distributed according containing constants
- Transducer can locally answer: "Am I responsible for constant X?"

Node 1

move(a,b)
move(b,a)
move(b,c)
move(c,d)
move(c,j)

Node 2

move(c,d)
move(d,e)
move(e,g)
move(d,f)
move(f,h)
move(f,h)

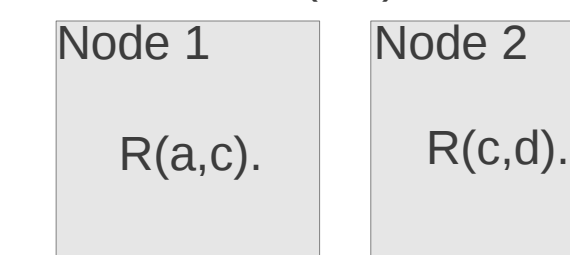
Node 3

move(c,j)
move(j,k)
move(k,l)

Arbitrary Partitioning Policies

- Policy = arbitrary mapping from **fact in Herbrand base** to nodes
- Transducer can locally answer: "Am I responsible for a potential input fact?"

Input:
 $R(a,c).$
 $R(c,d).$



Non-monotone Query:

$answer \leftarrow \neg R(a, b).$

Coordination-free Transducer:

$answer \leftarrow \neg R(a, b), responsible_R(a, b).$

Semi-positive Datalog is \mathcal{F}_1 coordination-free.

Preliminary Simulation Results

Cost Model of Distributed System

- Fully switched network
- Unit-cost for sending data over cable
- Unit-cost for deriving new fact
- "Instant" fixpoint detection for Alt-FP

Move Graph

- 5k edges, endpoints exponentially distributed
- Diameter: around 13

Result

- Coord-free better parallel efficiency
- Coord-free faster for larger networks

