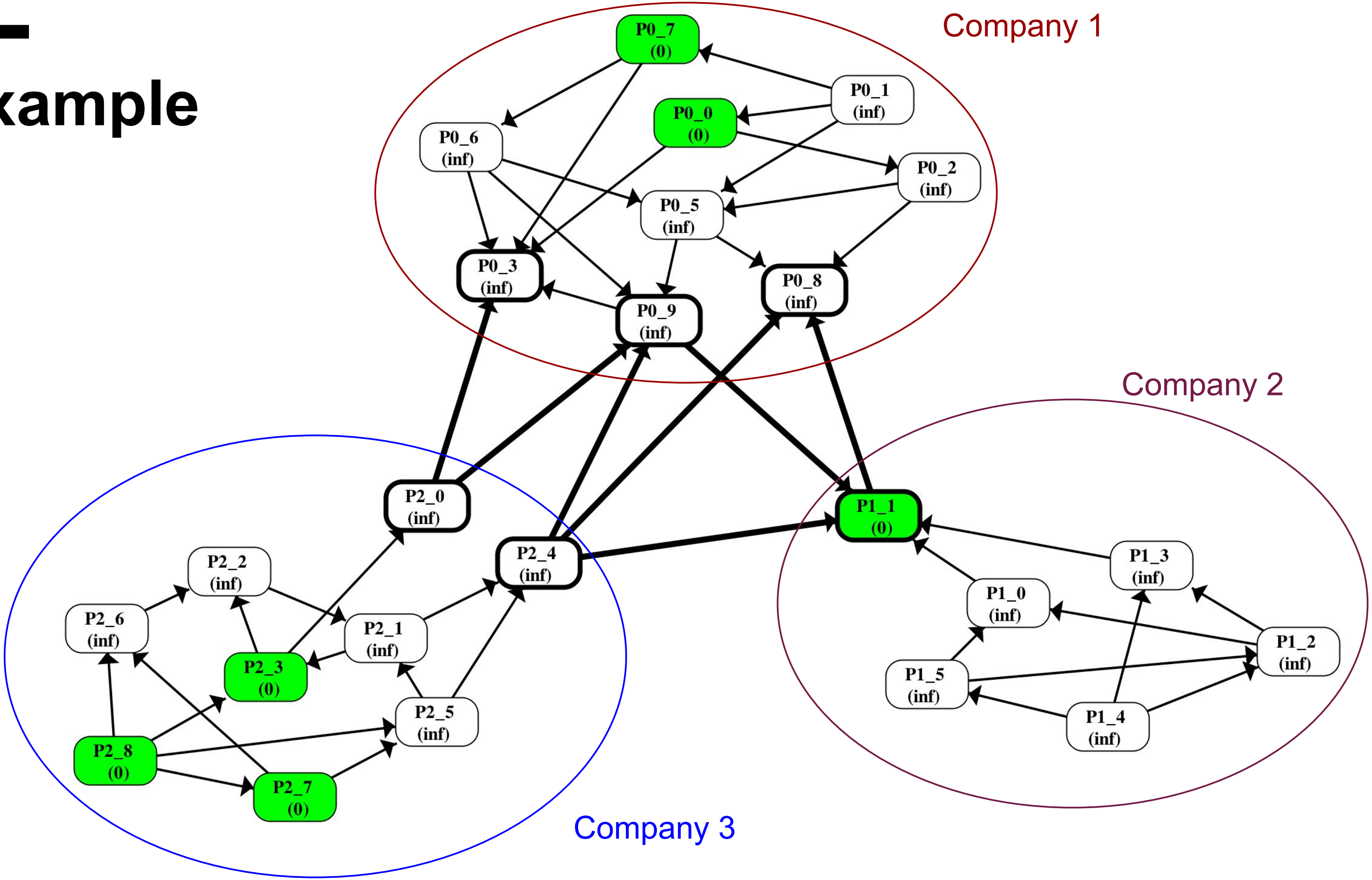
Network Distance

Fast Iterative MPC by Symbolic Optimization of Unrolled Loops.

Network Distance

- Large network with nodes belonging to different companies.
- Two kinds of nodes: Vulnerable, and Safe.
- Compute minimum distance to a vulnerable node for all nodes.
- Vulnerable nodes have distance 0
- Safe nodes have distance infinity.
- Computation consists of alternating rounds of addition and min.

Example



Company 1
P0_7
(0)
P0_1
(inf)
P0_0
(0)
P0_6
(inf)
P0_2
(inf)
P0_5
(inf)
P0_3
(inf)
P0_8
(inf)
P0_9
(inf)
Company 2
P2_0
(inf)
P1_1
(0)
P2_4
(inf)
P1_3
(inf)
P2_2
(inf)
P1_0
(inf)
P2_6
(inf)
P2_1
(inf)
P1_2
(inf)
P2_3
(0)
P1_5
(inf)
P2_5
(inf)
P1_4
(inf)
P2_8
(0)
P2_7
(0)
Company 3

Security

- Passive Security.
- Hide internal topology of a company’s network.
- Final distance is known but not any intermediate values.
- Final distance is known but not the causing vulnerable nodes.
- Gateways graph is public.
- Each company learns only the outputs of its own nodes.

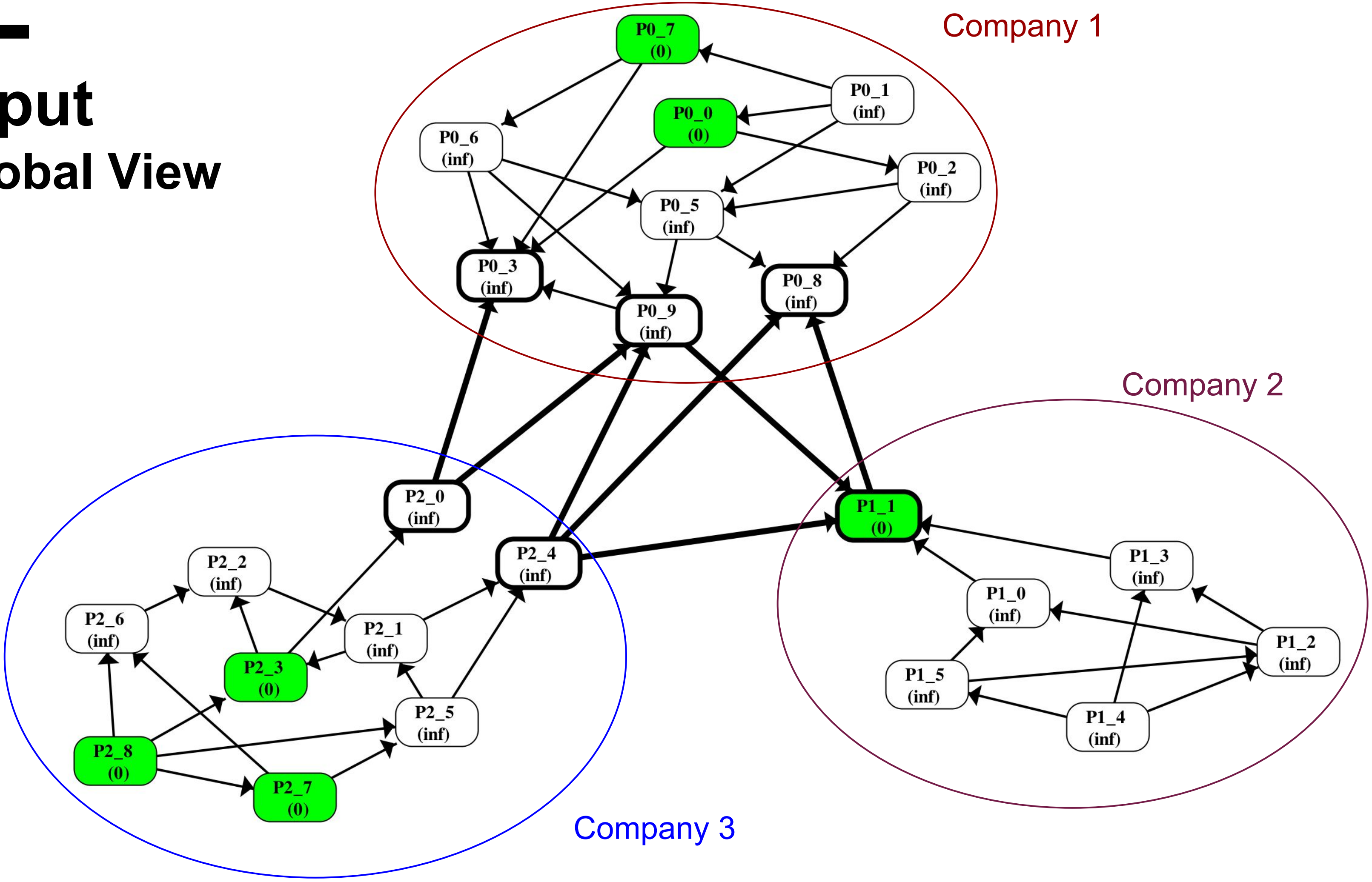
Problem

- Graph can be very large
  - Cannot perform MPC on the entire graph.

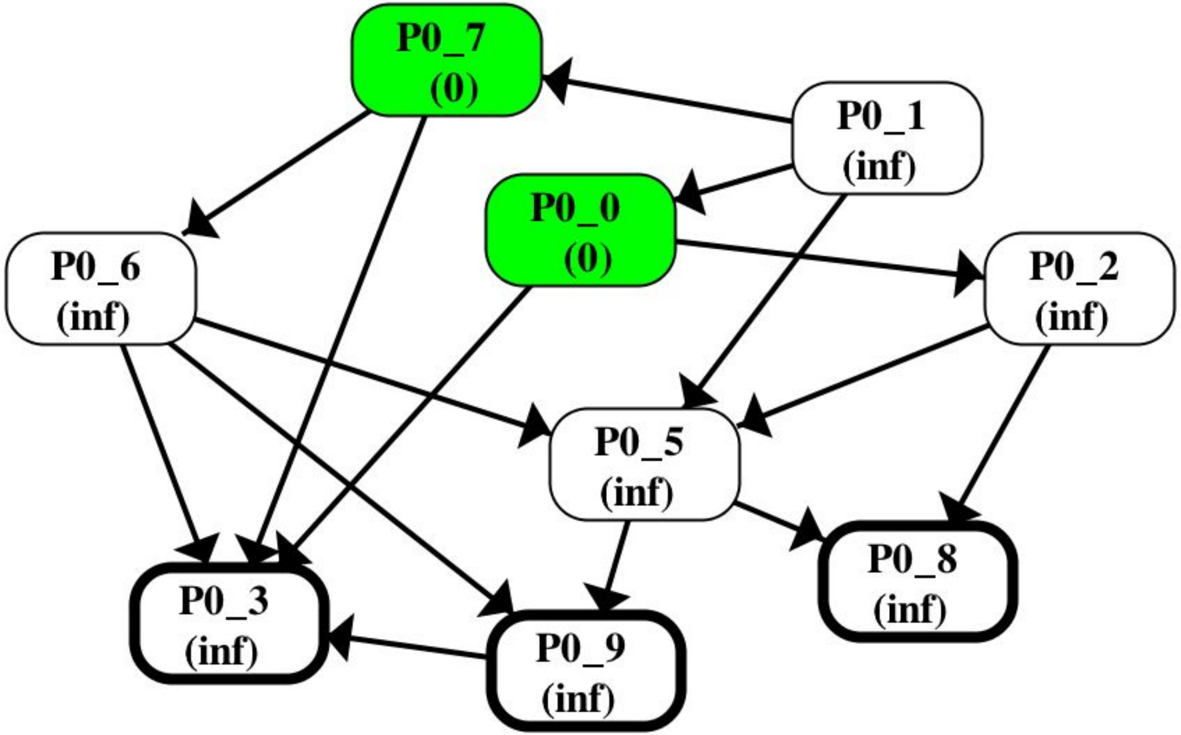
Solution

- Companies perform direct computation on their private network.
- MPC computation on the subgraph containing only gateways of companies.
- Direct computation to propagate gateways results back into private network.

Input
Global View

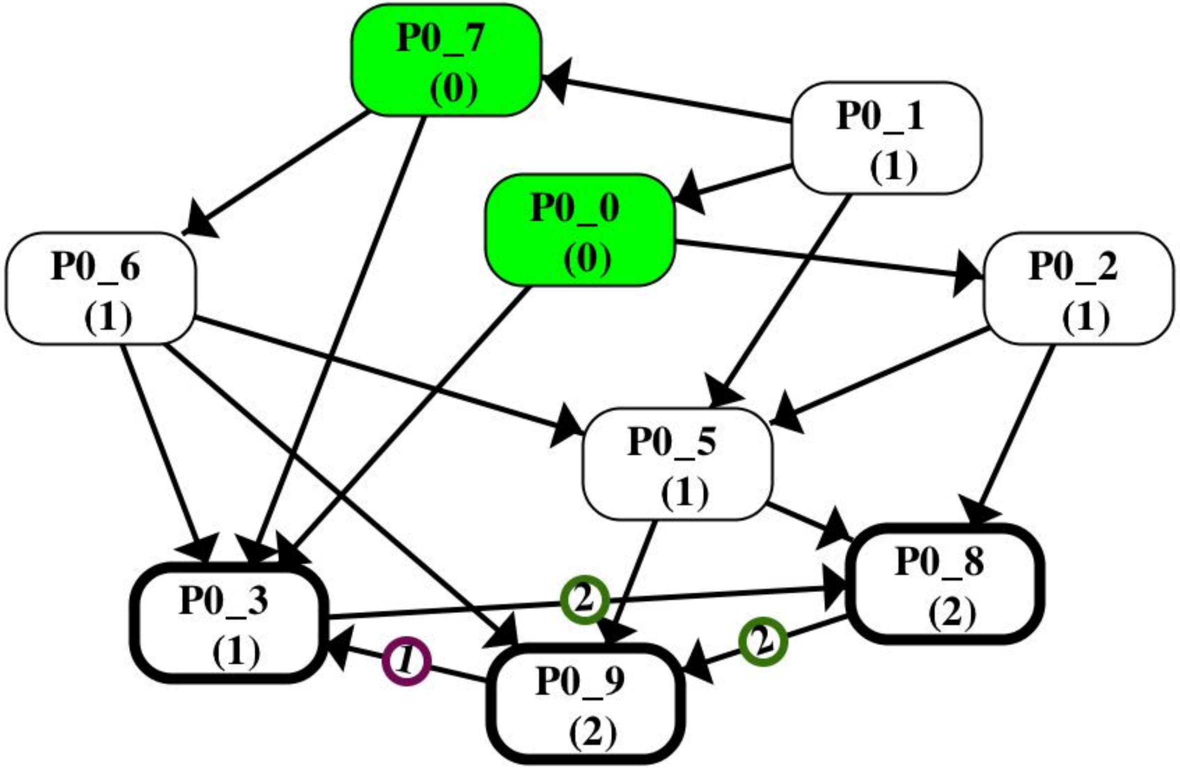


Local Network
Company 1

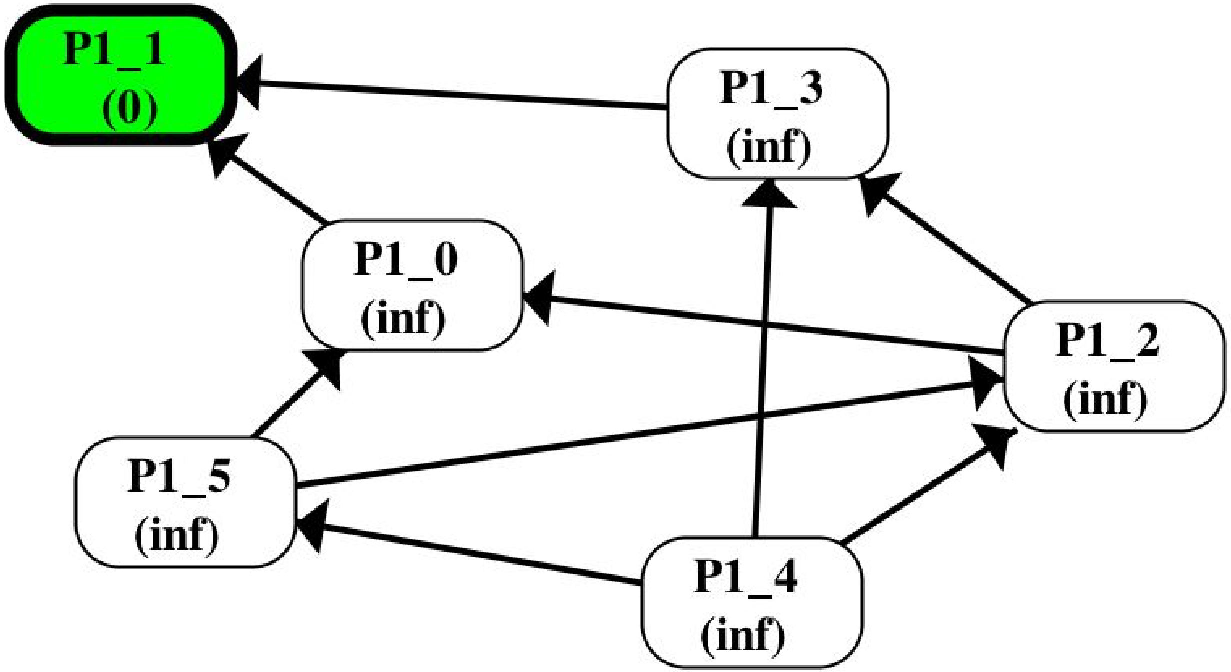


Local Computation Output

Company 1

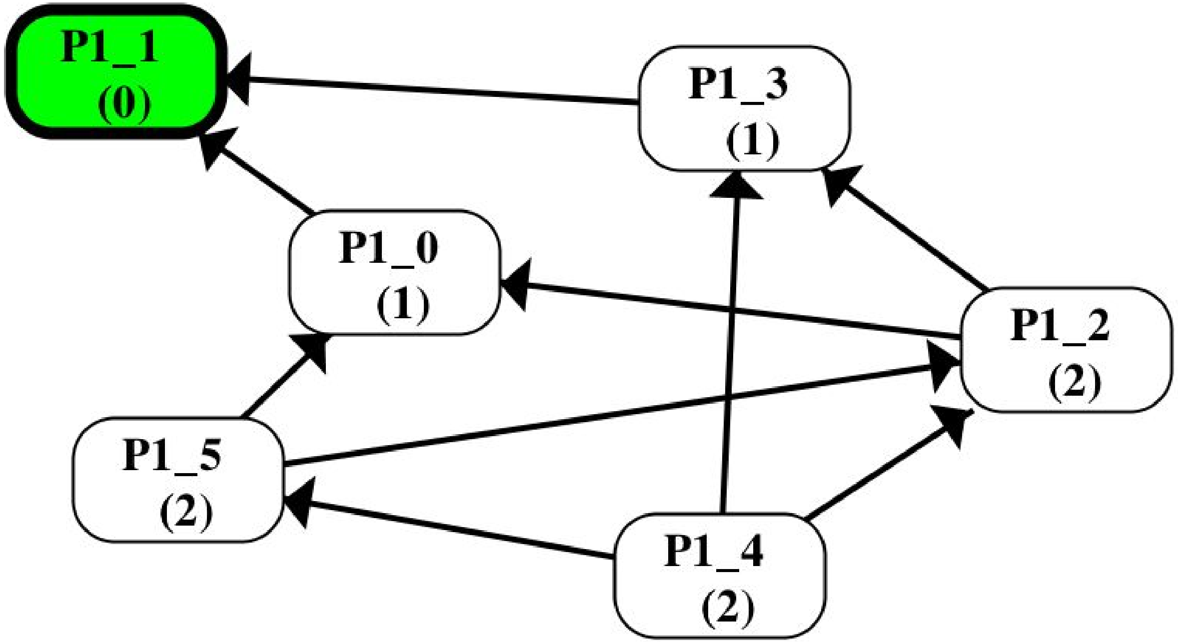


Local Network
Company 2

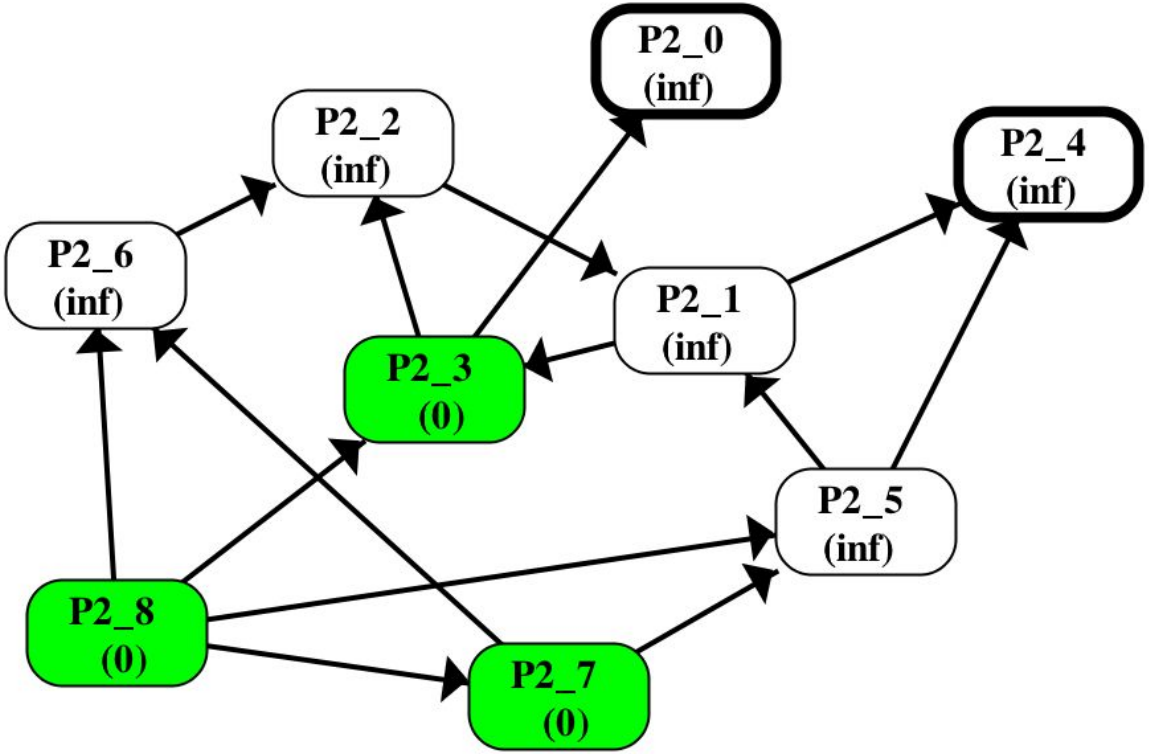


Local Computation Output

Company 2

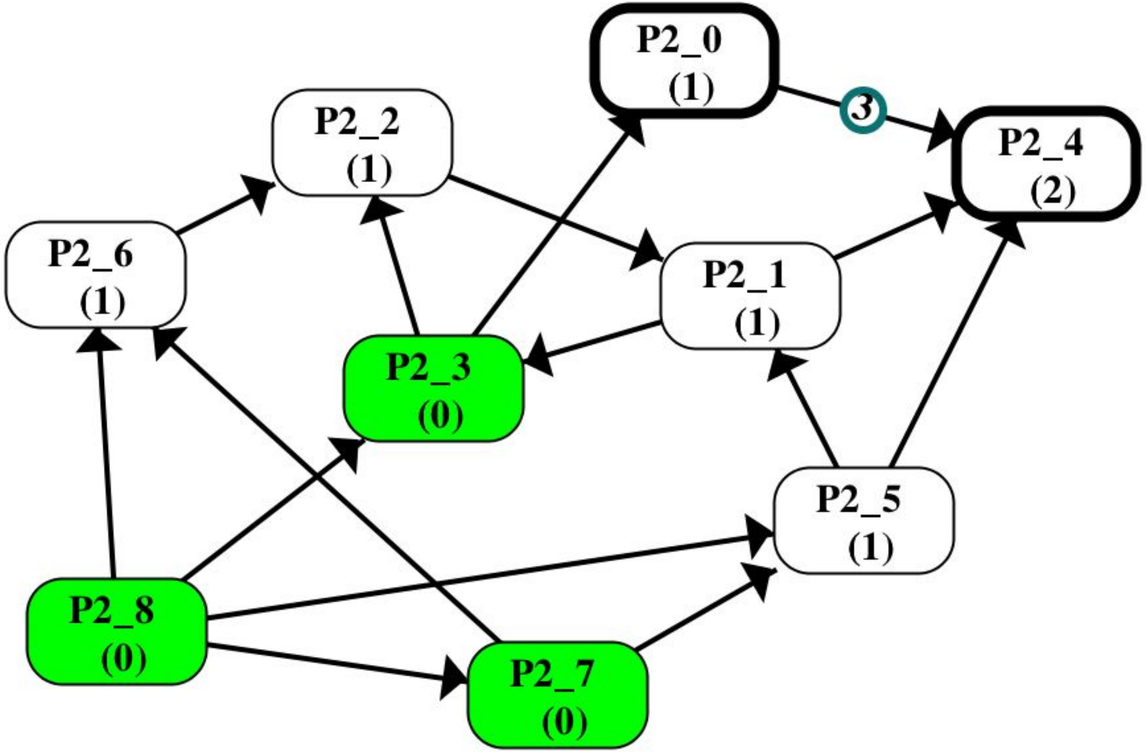


Local Network
Company 3

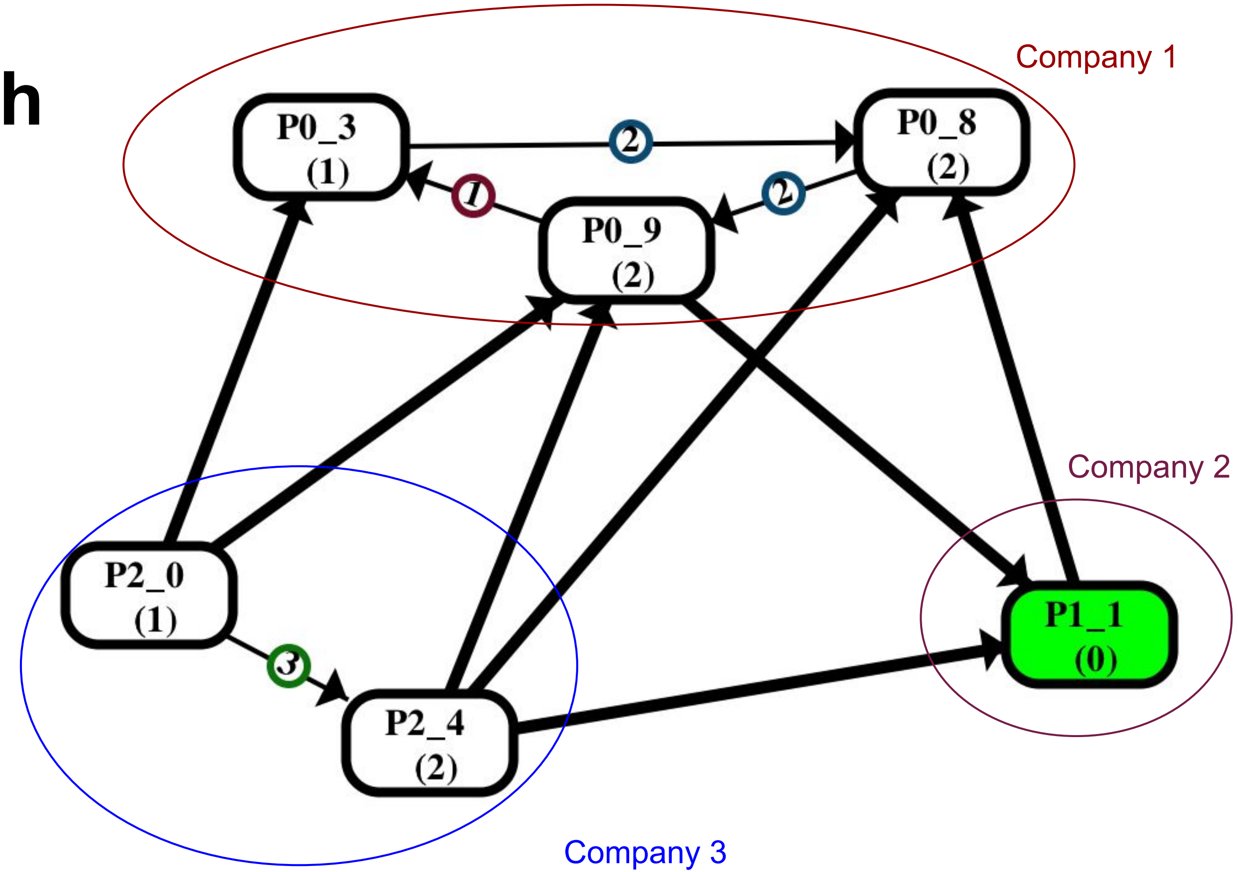


Local Computation Output

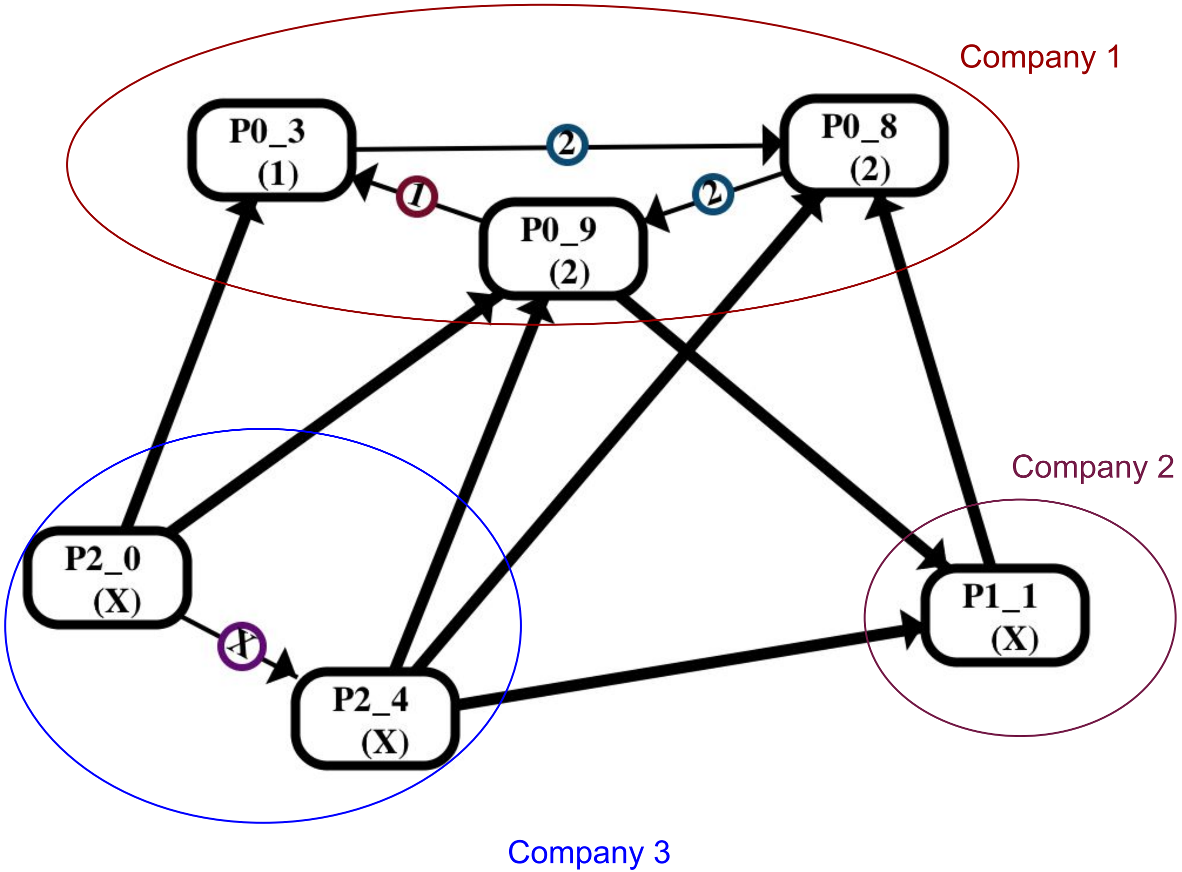
Company 3



Gateways Graph
Global View

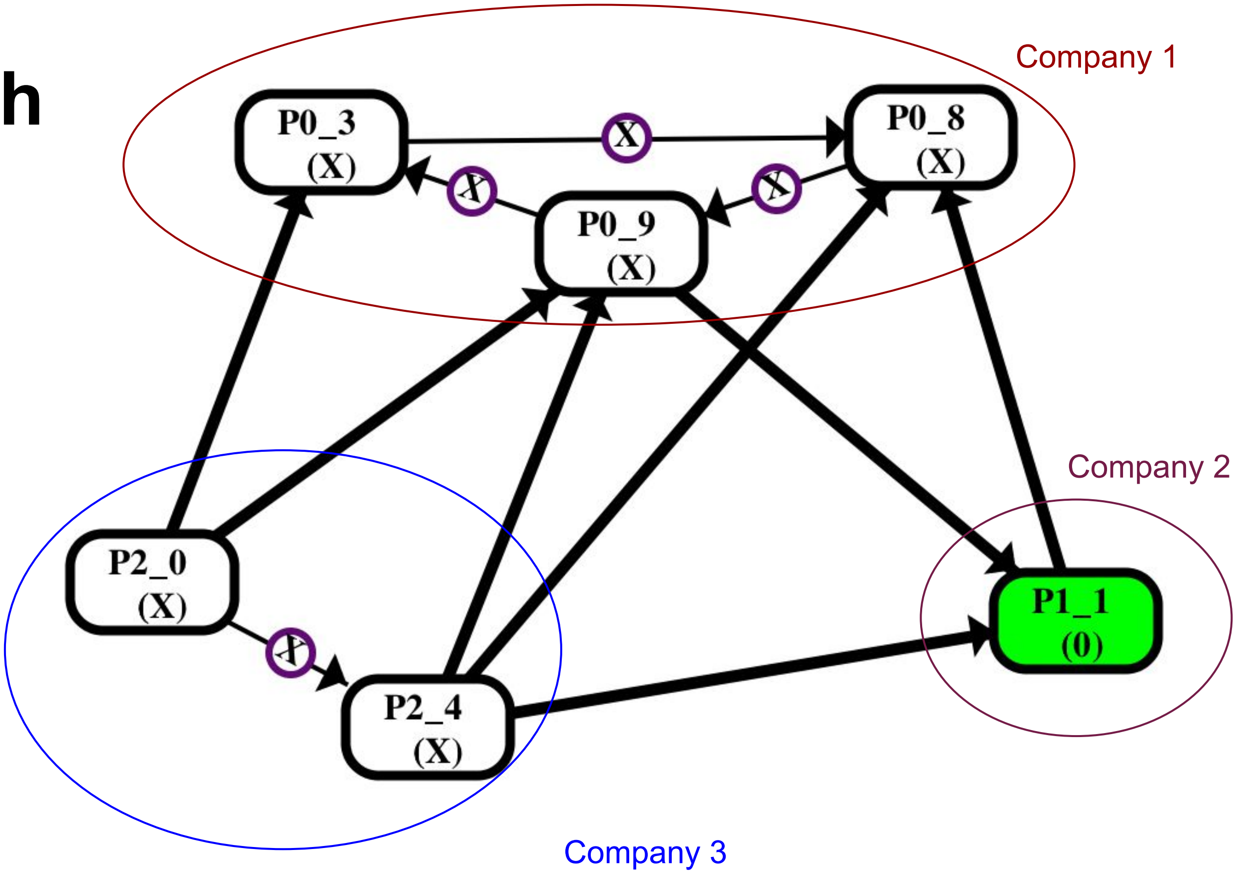


Gateways Graph
Company 1



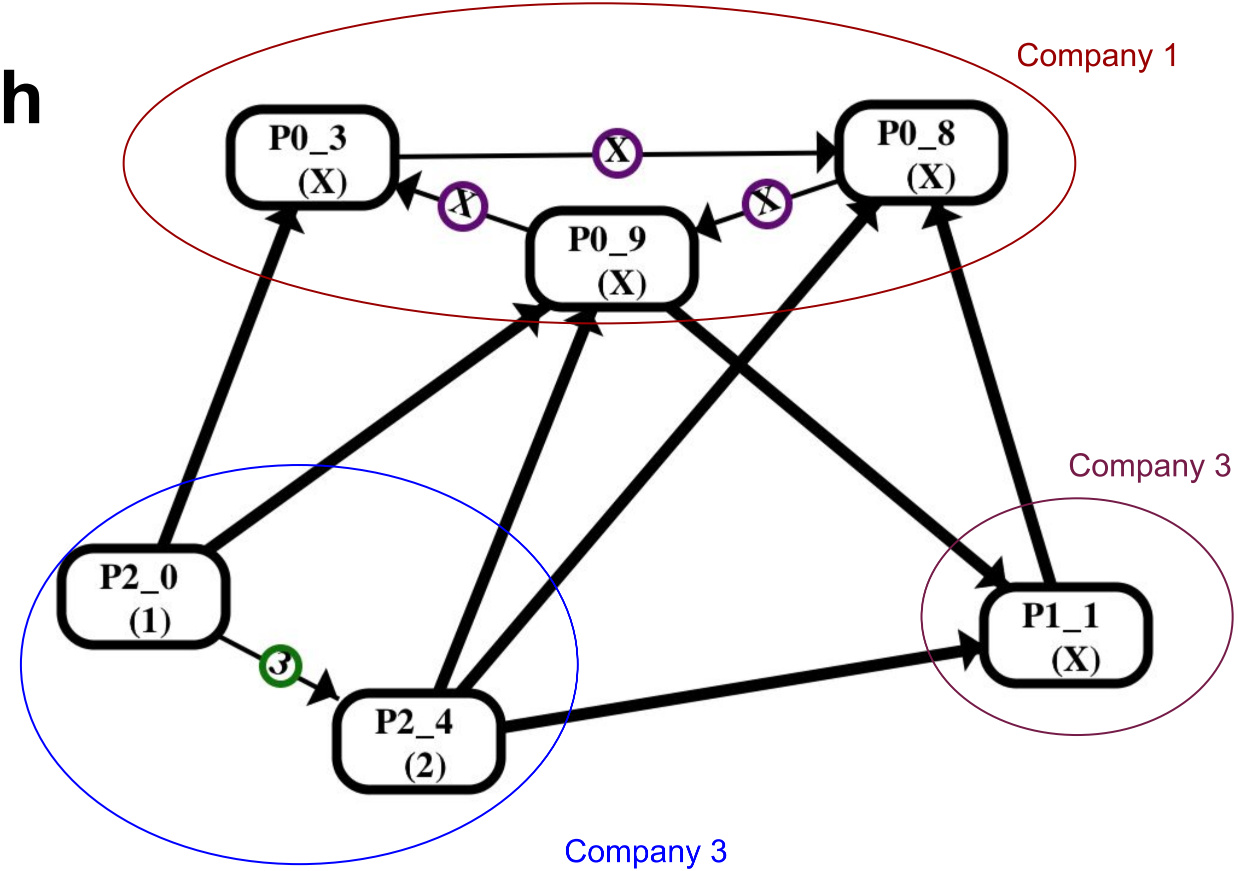
Gateways Graph

Company 2

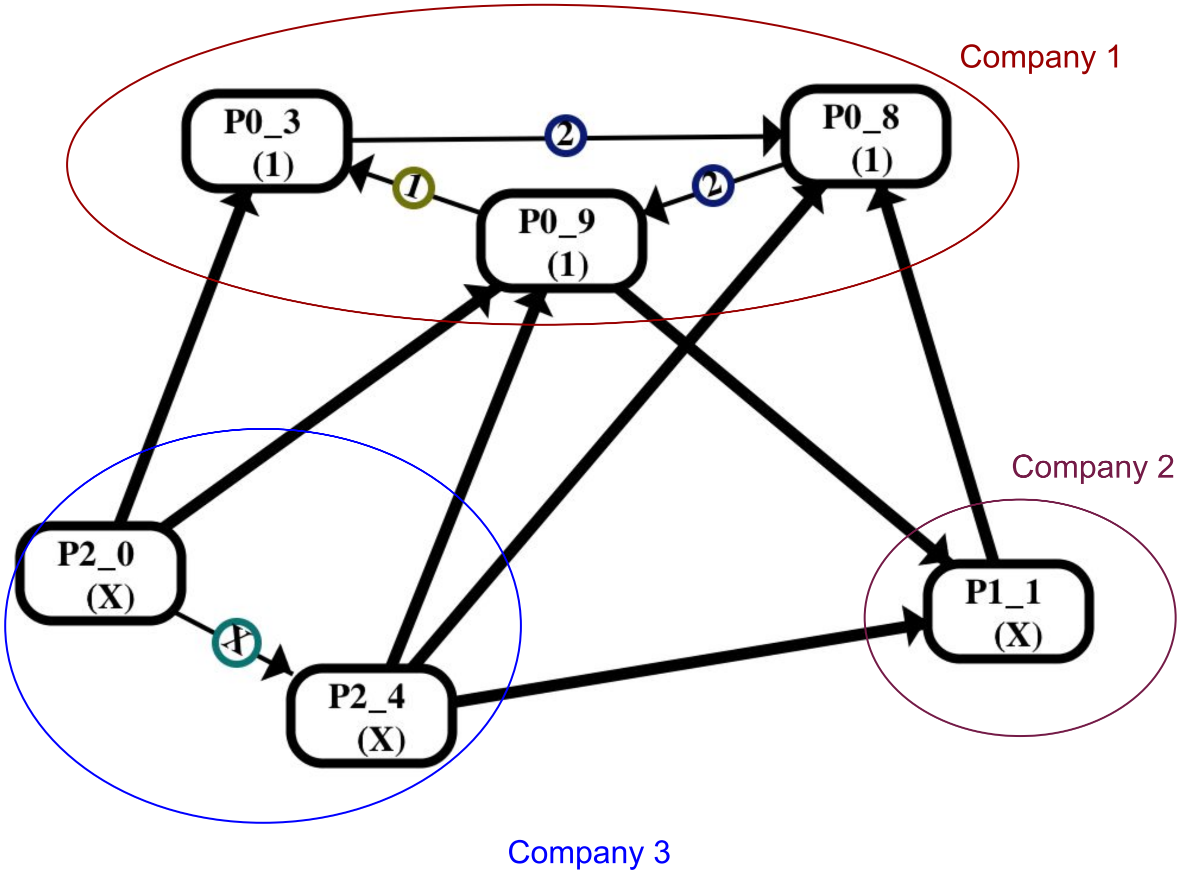


Gateways Graph

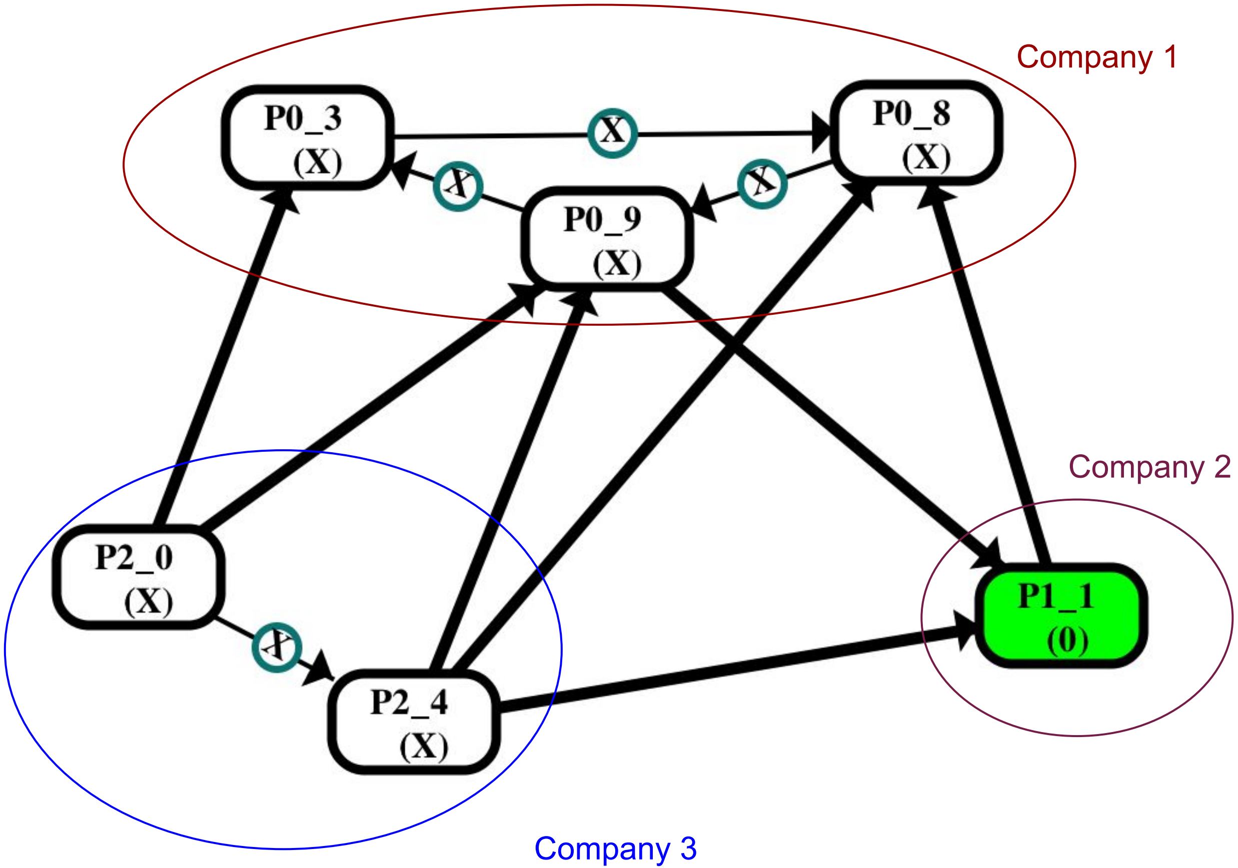
Company 3



MPC Output
Company 1

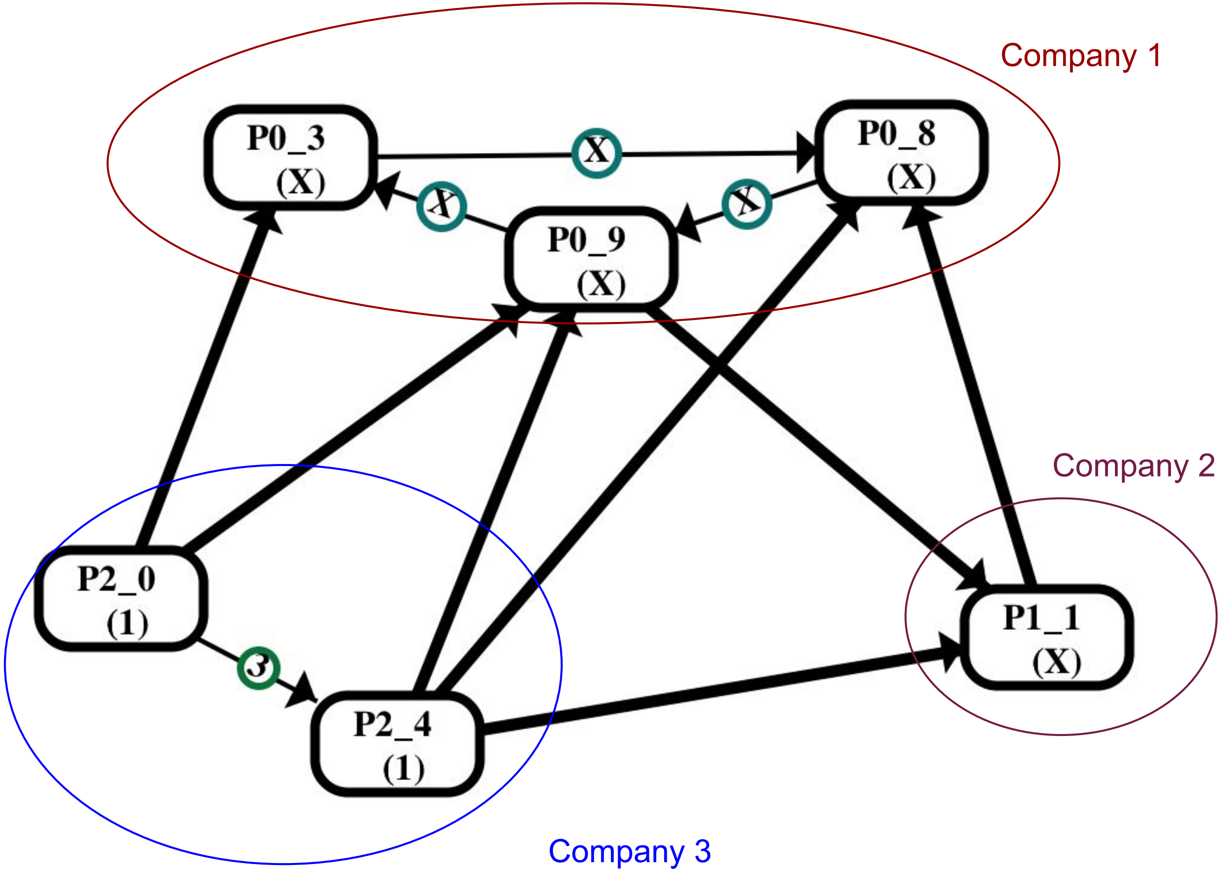


MPC Output
Company 2



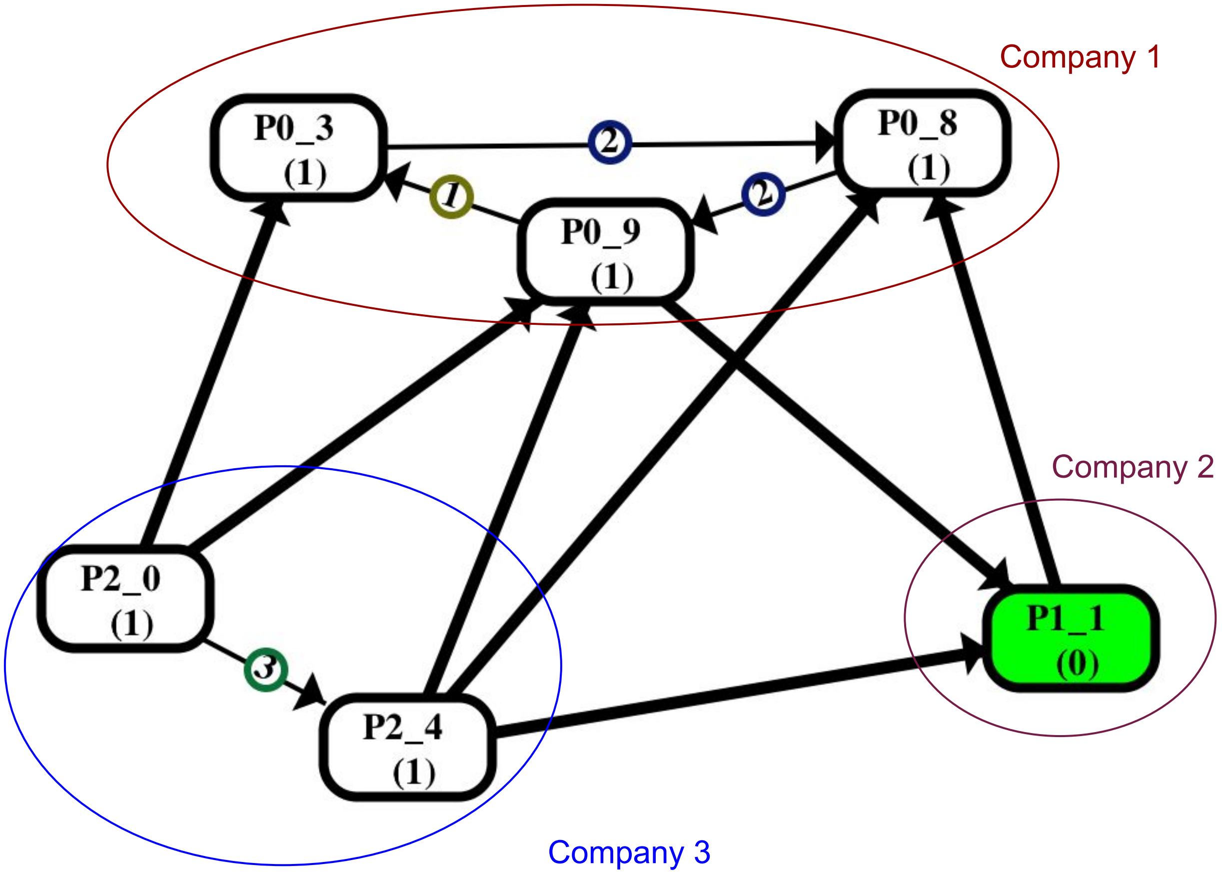
MPC Output

Company 3

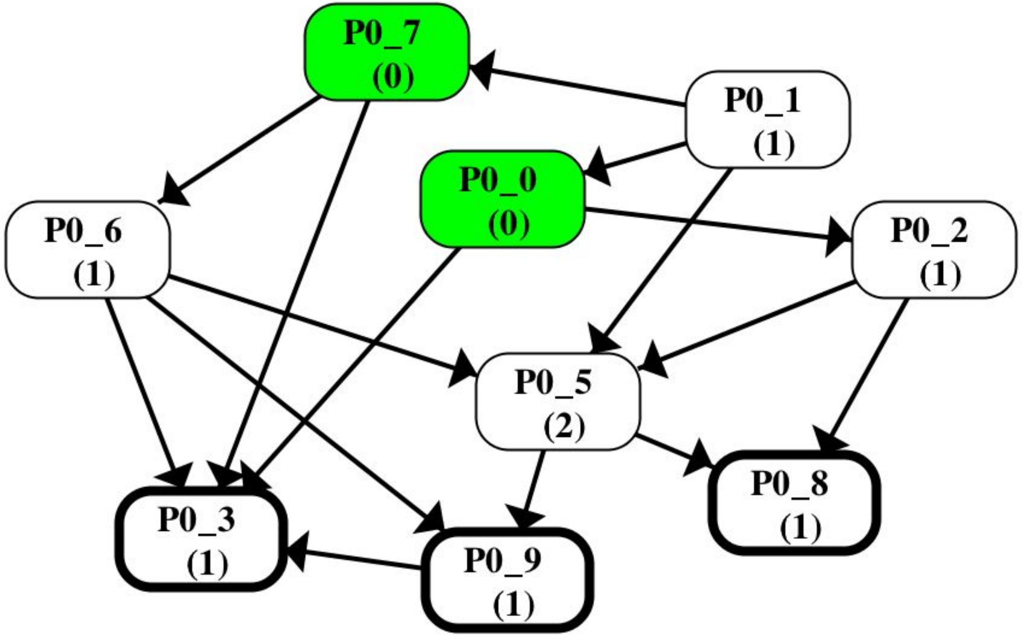


MPC Output

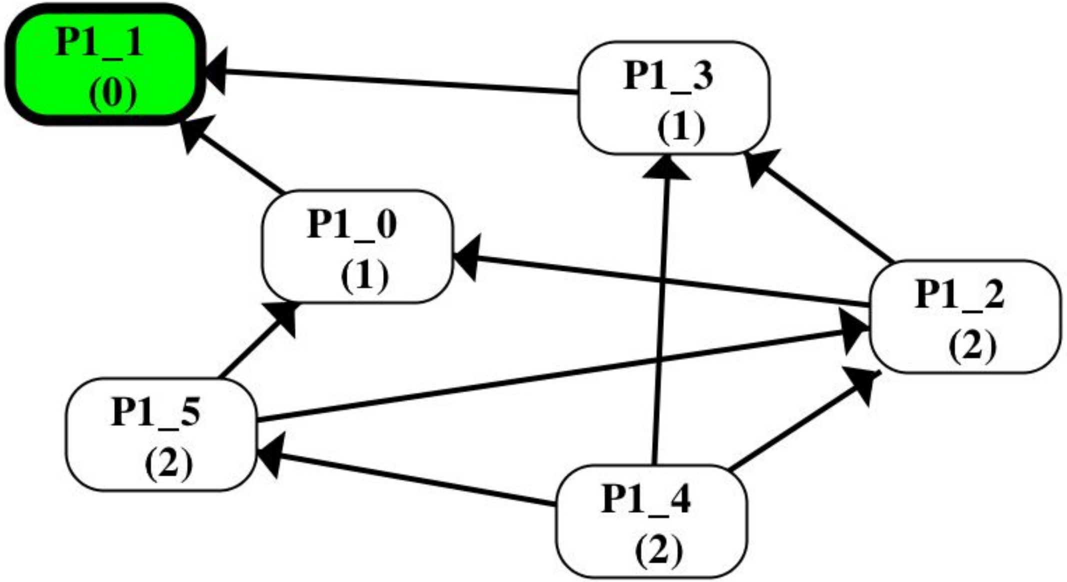
Global View



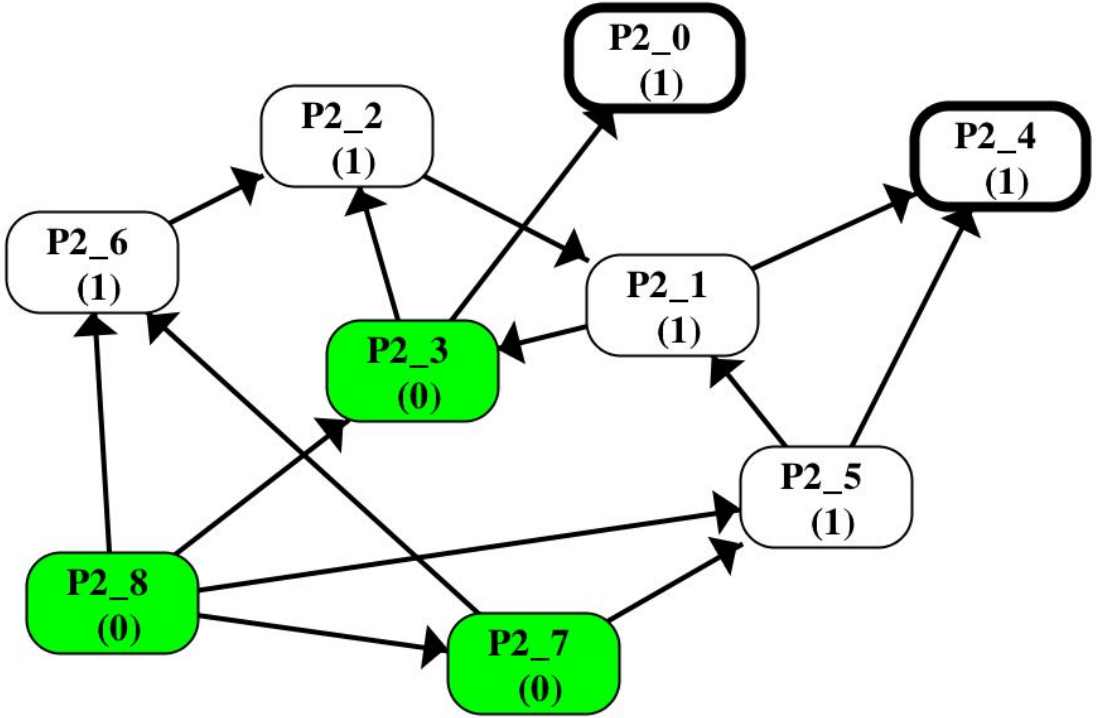
Final Output
Company 1



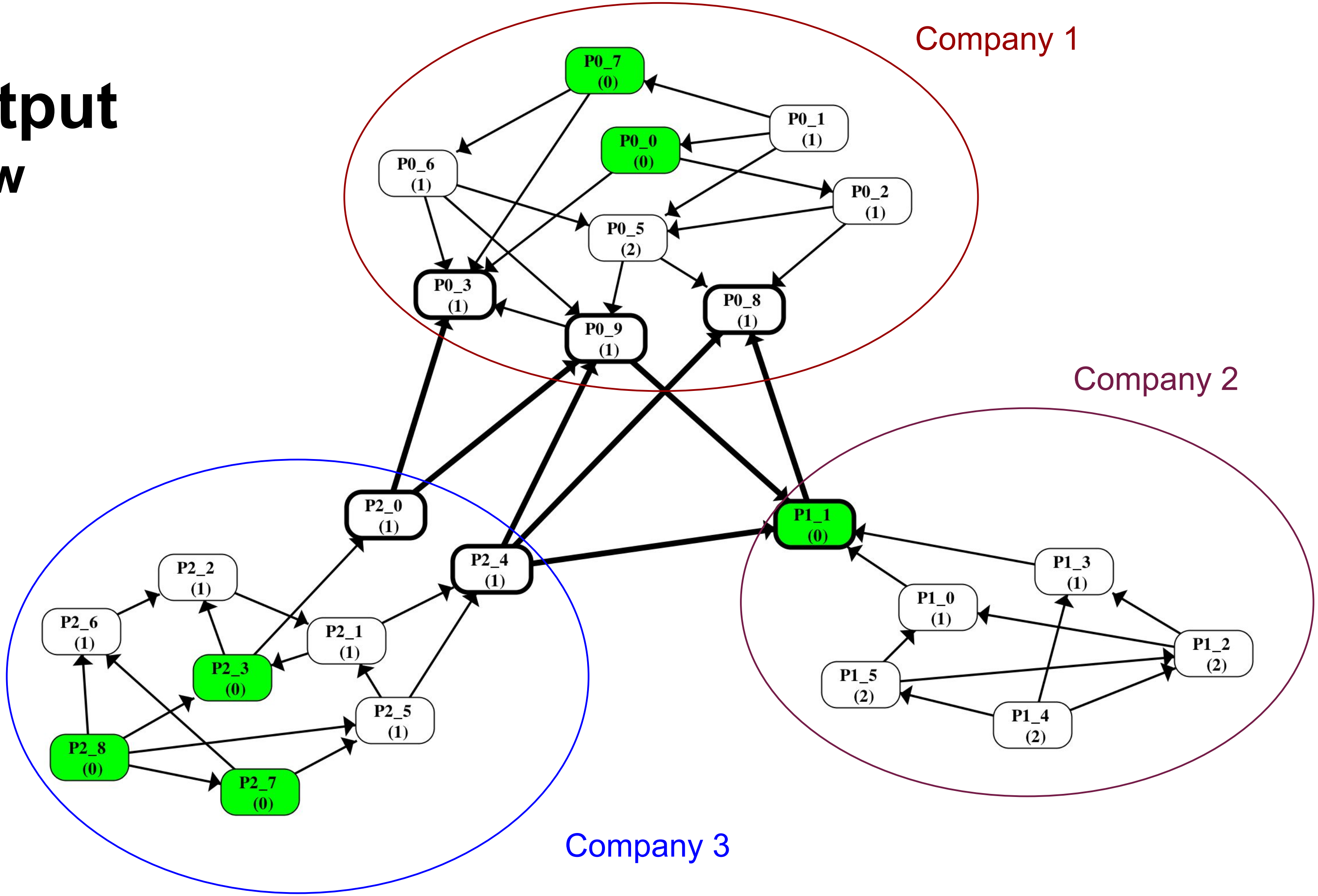
Final Output
Company 2



Final Output
Company 3



Final Output
Global View



Problem

- Not revealing any intermediate values
  - Can be done with viff, but with huge cost (min is expensive).

Solution

- Run algorithm with “symbolic” inputs.
- Unwraps iteration/execution yielding an expression.
- Secret share initial distances, evaluate expression in one shot.
- Only final answer is revealed.
- Expression can be optimized. It can be stored and re-used.

Expression

$P2\_G2_1$

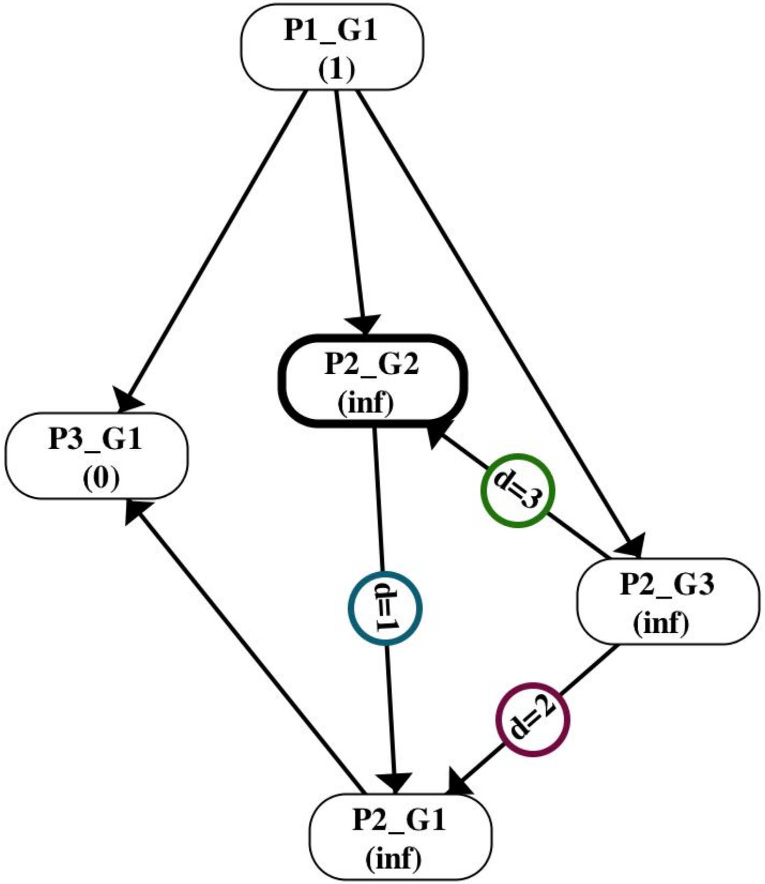
$= \textbf{min}(P2\_G2_0, P1\_G1_0+1, P2\_G3_0+3, P2\_G1_0+1$

$= \textbf{min}(inf, 2, inf, inf)$

$P2\_G2_2$

$= \textbf{min}(P2\_G2_1, P1\_G1_1+1, P2\_G3_1+3, P2\_G1_1+1$

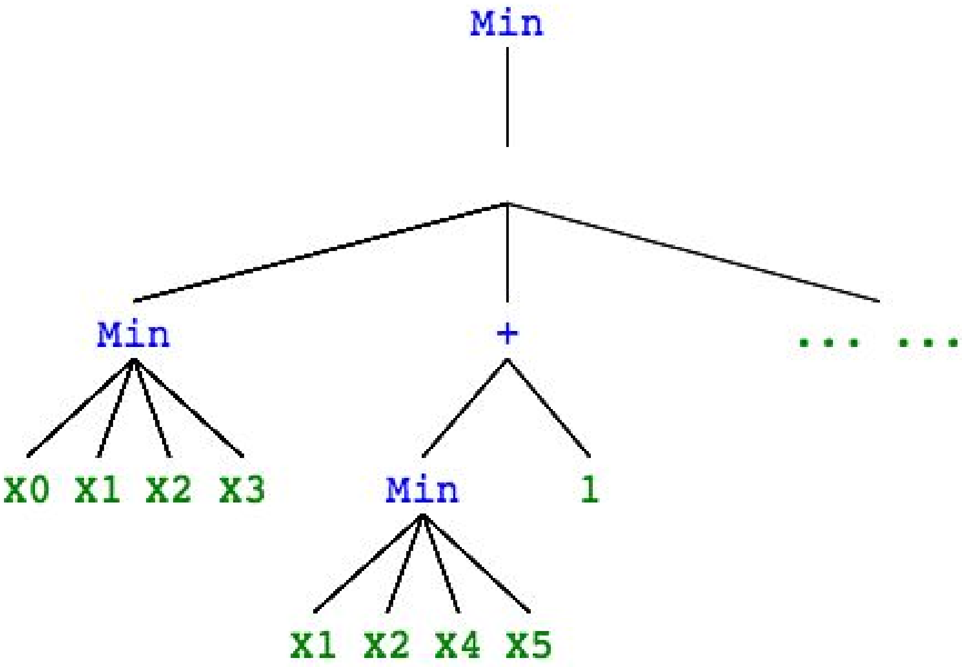
$= \textbf{min}(\ \textbf{min}(inf, 2, inf, inf), \textbf{min}(1, 1, inf, inf)+1, \ \ldots$



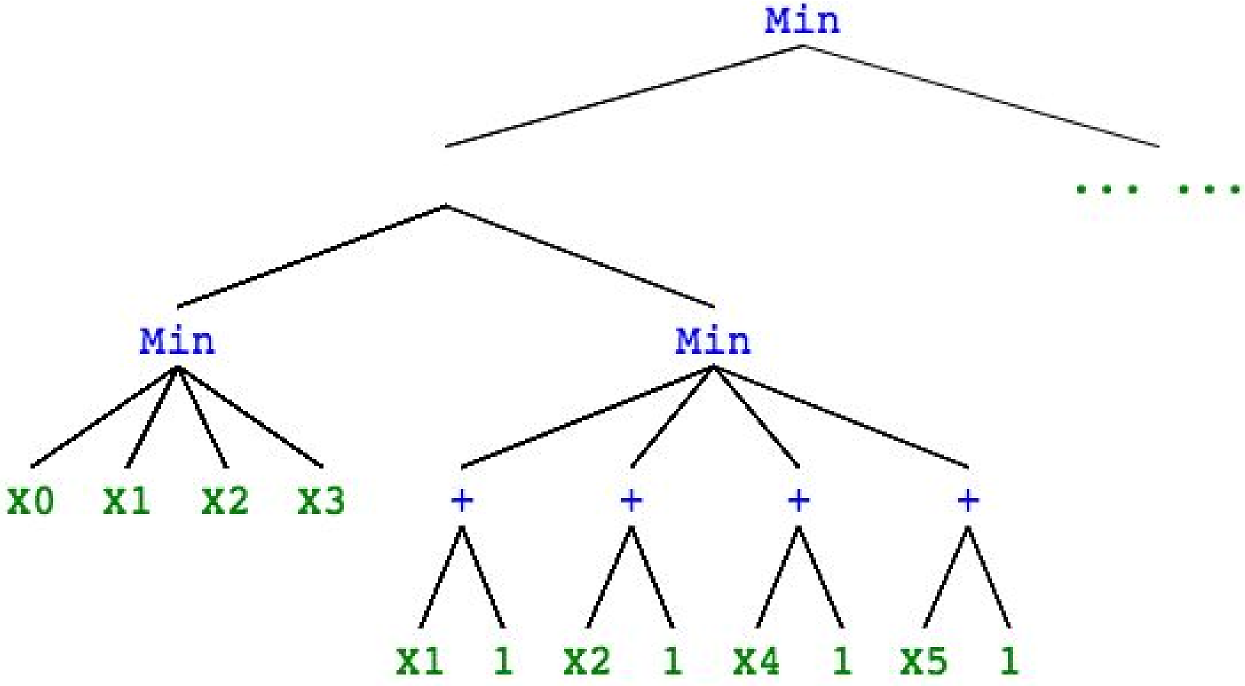
Note that this is a different example than the previously illustrated one

Expression Tree

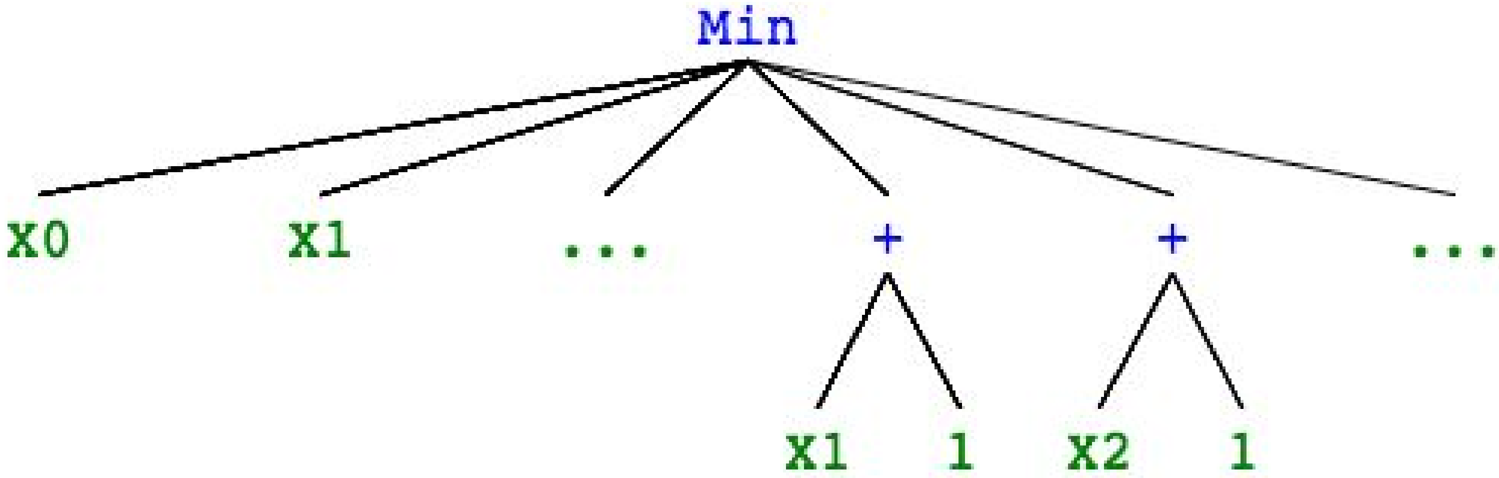
- Expressions as parse trees.
- Levels of + and Min.
- Optimizing the syntax tree is equivalent to simplifying the expression.
- Simplification utilizes distribution properties of + and Min.
- Simplification yields a single Min with compact additions.



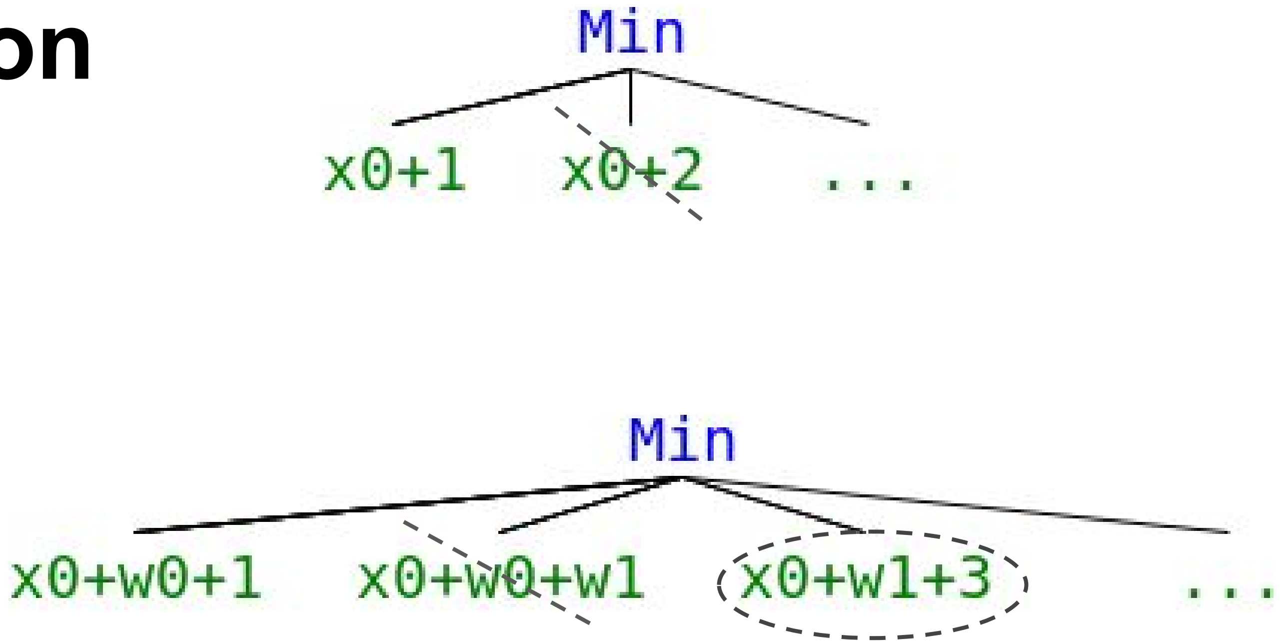
Add-Min Reduction



Min-Min Reduction



Early-Min Reduction



Library Modules

- Implemented as a Python Library.
- Operator overloading to construct Expressions.
- Important Modules
  - Expression : AddExpression, MinExpression, VarExpression …
  - Simplifiers: BaseSimplifier, Reductions, Optimizing Simplifiers …
  - Evaluators: BaseEvaluator (Direct), ViffEvaluator.
- Customize expression, simplifiers, and evaluators (inheritance).
- Code does not reveal MPC, Expressions, or Twisted.
- Example: Simplification: 0.12 (sec). Eval: 16(sec)

  Pure Viff Solution: 24 (sec).

Future Extensions

- Provide builtin support for more expressions and simplifications.
- Apply the library/technique to more problems.
- Include conditionals, loops, and other python statements
- Speed-up evaluation by exploiting similar terms (memoization).
- SPDZ Evaluator ?

Thank You

Questions?