Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Linear systems of equations

Michael Quinn

Parallel Programming with MPI and OpenMP

material do autor

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Outline

- Terminology
- Back substitution
- Gaussian elimination

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Problem

- System of linear equations
  - Solve $Ax = b$ for x

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Back Substitution

- Used to solve upper triangular system
  $Tx = b$ for x
- Methodology: one element of x can be immediately computed
- Use this value to simplify system, revealing another element that can be immediately computed
- Repeat

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Back Substitution

$$
\begin{array}{rrrrcr}
1x_0 & +1x_1 & -1x_2 & +4x_3 & = & 8 \\
 & -2x_1 & -3x_2 & +1x_3 & = & 5 \\
 & & 2x_2 & -3x_3 & = & 0 \\
 & & & 2x_3 & = & 4
\end{array}
$$

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Back Substitution

$$
\begin{array}{ccccccc}
1x_0 & +1x_1 & -1x_2 & +4x_3 & = & 8 \\
 & -2x_1 & -3x_2 & +1x_3 & = & 5 \\
 & & 2x_2 & -3x_3 & = & 0 \\
 & & & 2x_3 & = & 4
\end{array}
$$

$$x_3 = 2$$

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Back Substitution

$$
\begin{array}{rrrrcr}
1x_0 & +1x_1 & -1x_2 & & = & 0 \\
 & -2x_1 & -3x_2 & & = & 3 \\
 & & 2x_2 & & = & 6 \\
 & & & 2x_3 & = & 4
\end{array}
$$

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Back Substitution

$$
\begin{array}{ccccc}
1x_0 & +1x_1 & -1x_2 & & = & 0 \\
 & -2x_1 & -3x_2 & & = & 3 \\
 & & 2x_2 & & = & 6 \\
 & & & 2x_3 & = & 4
\end{array}
$$

$$x_2 = 3$$

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Back Substitution

$$
\begin{array}{cccccc}
1x_0 & +1x_1 & & & = & 3 \\
 & -2x_1 & & & = & 12 \\
 & & 2x_2 & & = & 6 \\
 & & & 2x_3 & = & 4
\end{array}
$$

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Back Substitution

$$
\begin{array}{lllll}
1x_0 & +1x_1 & & & = 3 \\
 & -2x_1 & & & = 12 \\
 & & 2x_2 & & = 6 \\
 & & & 2x_3 & = 4
\end{array}
$$

$$x_1 = -6$$

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Back Substitution

$$
\begin{aligned}
1x_0 & & & & = 9 \\
& -2x_1 & & & = 12 \\
& & 2x_2 & & = 6 \\
& & & 2x_3 & = 4
\end{aligned}
$$

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Back Substitution

$$\begin{array}{rrrrcl} 1x_0 & & & & = & 9 \\ & -2x_1 & & & = & 12 \\ & & 2x_2 & & = & 6 \\ & & & 2x_3 & = & 4 \end{array}$$

$$x_0 = 9$$

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

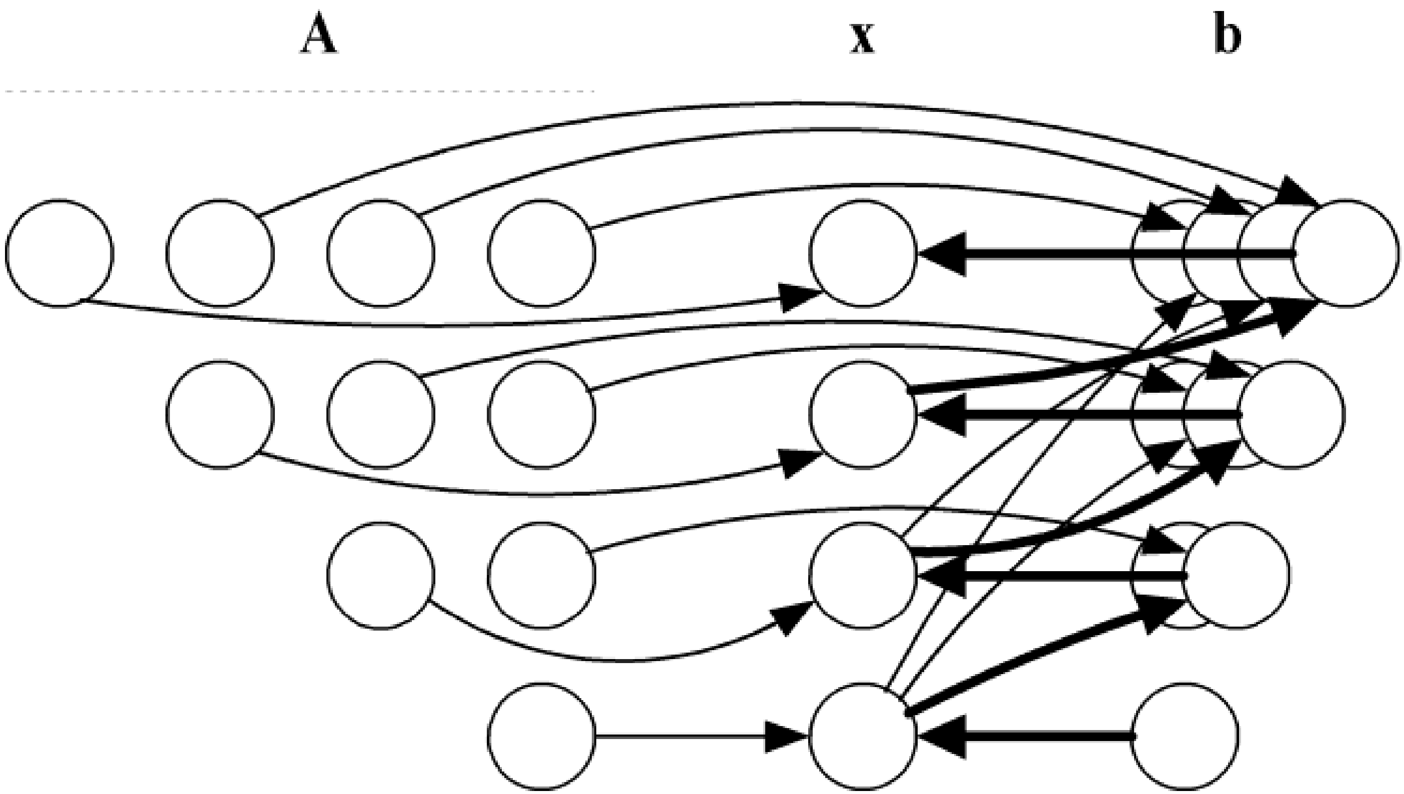
Pseudocode

for $i \leftarrow n - 1$ down to 1 do
    $x[i] \leftarrow b[i] / a[i, i]$
    for $j \leftarrow 0$ to $i - 1$ do
        $b[j] \leftarrow b[j] - x[i] \times a[j, i]$
    endfor
endfor

Time complexity: $\Theta(n^2)$

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Data Dependence Diagram



We cannot execute the outer loop in parallel.
We can execute the inner loop in parallel.

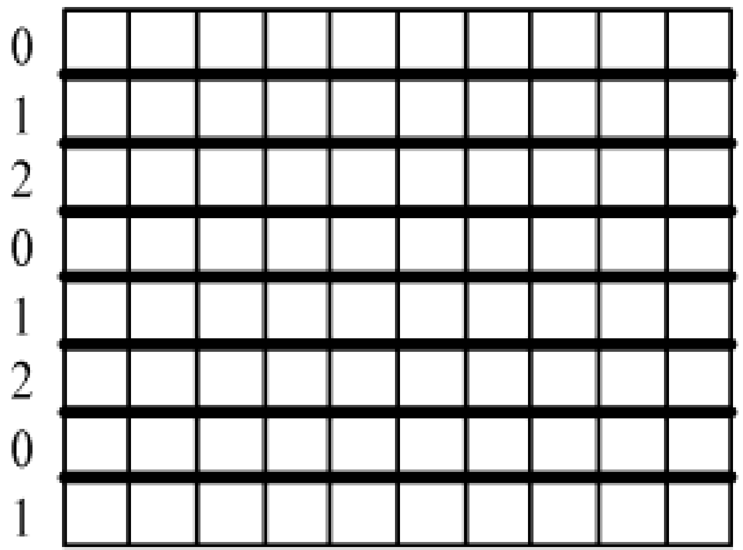
Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Row-oriented Algorithm

- Associate primitive task with each row of *A* and corresponding elements of *x* and *b*
- During iteration *i* task associated with row *j* computes new value of b_j
- Task *i* must compute x_i and broadcast its value
- Agglomerate using rowwise interleaved striped decomposition

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Interleaved Decomposition



Rowwise interleaved
striped decomposition

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Complexity Analysis

- Each process performs about n / (2p) iterations of loop j in all
- A total of n -1 iterations in all
- Computational complexity: $\Theta(n^2/p)$
- One broadcast per iteration
- Communication complexity: $\Theta(n \log p)$

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Gaussian Elimination

- Used to solve $Ax = b$ when A is dense
- Reduces $Ax = b$ to upper triangular system $Tx = c$
- Back substitution can then solve $Tx = c$ for x

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Gaussian Elimination

$$
\begin{array}{rrrrcr}
4x_0 & +6x_1 & +2x_2 & -2x_3 & = & 8 \\
2x_0 & & +5x_2 & -2x_3 & = & 4 \\
-4x_0 & -3x_1 & -5x_2 & +4x_3 & = & 1 \\
8x_0 & +18x_1 & -2x_2 & +3x_3 & = & 40
\end{array}
$$

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Gaussian Elimination

$$
\begin{array}{ccccccc}
4x_0 & +6x_1 & +2x_2 & -2x_3 & = & 8 \\
 & -3x_1 & +4x_2 & -1x_3 & = & 0 \\
 & +3x_1 & -3x_2 & +2x_3 & = & 9 \\
 & +6x_1 & -6x_2 & +7x_3 & = & 24
\end{array}
$$

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Gaussian Elimination

$$
\begin{array}{rrrrcr}
4x_0 & +6x_1 & +2x_2 & -2x_3 & = & 8 \\
 & -3x_1 & +4x_2 & -1x_3 & = & 0 \\
 & & 1x_2 & +1x_3 & = & 9 \\
 & & 2x_2 & +5x_3 & = & 24
\end{array}
$$

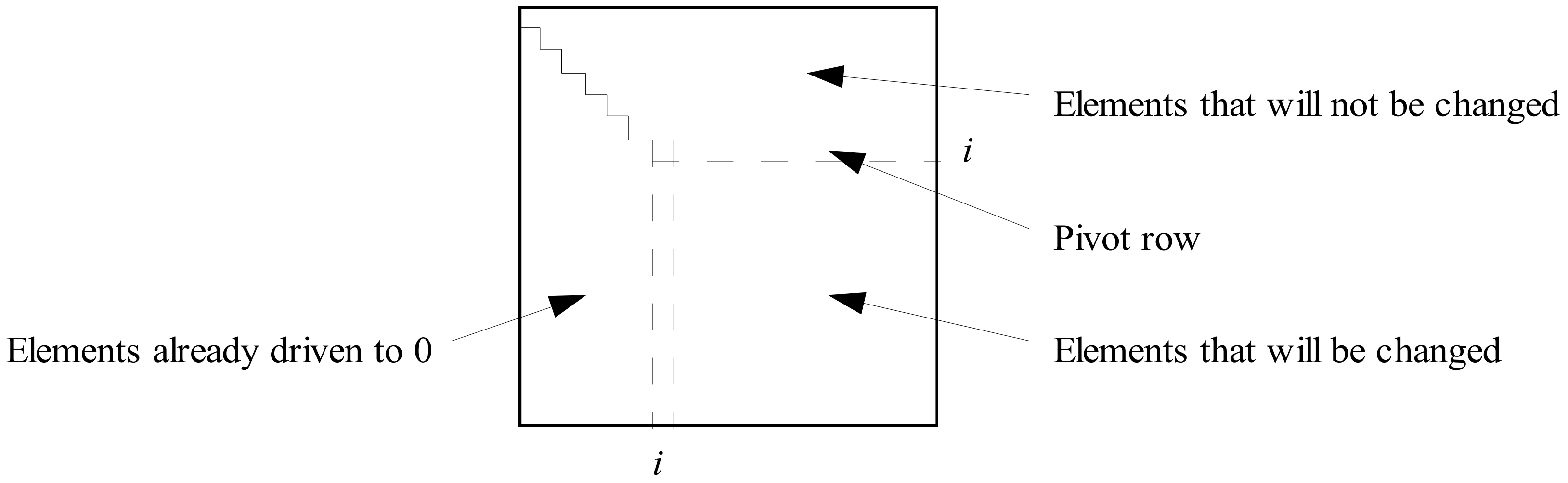
Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Gaussian Elimination

$$
\begin{array}{rrrrcr}
4x_0 & +6x_1 & +2x_2 & -2x_3 & = & 8 \\
 & -3x_1 & +4x_2 & -1x_3 & = & 0 \\
 & & 1x_2 & +1x_3 & = & 9 \\
 & & & 3x_3 & = & 6
\end{array}
$$

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Iteration of Gaussian Elimination



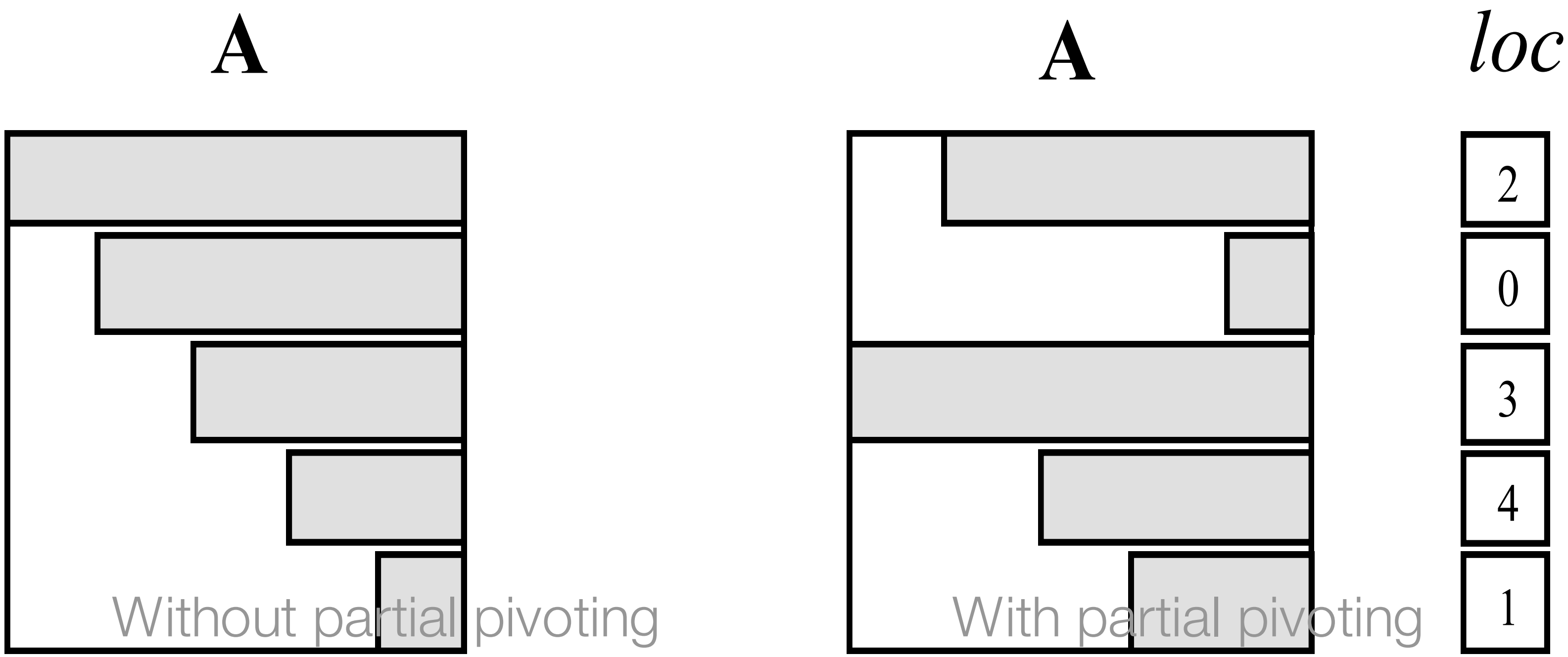
Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Numerical Stability Issues

- If pivot element close to zero, significant roundoff errors can result
- Gaussian elimination with partial pivoting eliminates this problem
- In step *i* we search rows *i* through *n*-1 for the row whose column *i* element has the largest absolute value
- Swap (pivot) this row with row *i*

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Implementing Partial Pivoting



- loc[i] indica onde encontrar a linha i da matriz triangular

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Row-oriented Parallel Algorithm

- Associate primitive task with each row of *A* and corresponding elements of *x* and *b*
- A kind of reduction needed to find the identity of the pivot row
- Tournament: want to determine identity of row with largest value, rather than largest value itself
- Could be done with two all-reductions
- MPI provides a simpler, faster mechanism

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

MPI_MAXLOC, MPI_MINLOC

- MPI provides reduction operators MPI_MAXLOC, MPI_MINLOC
- Provide datatype representing a (value, index) pair

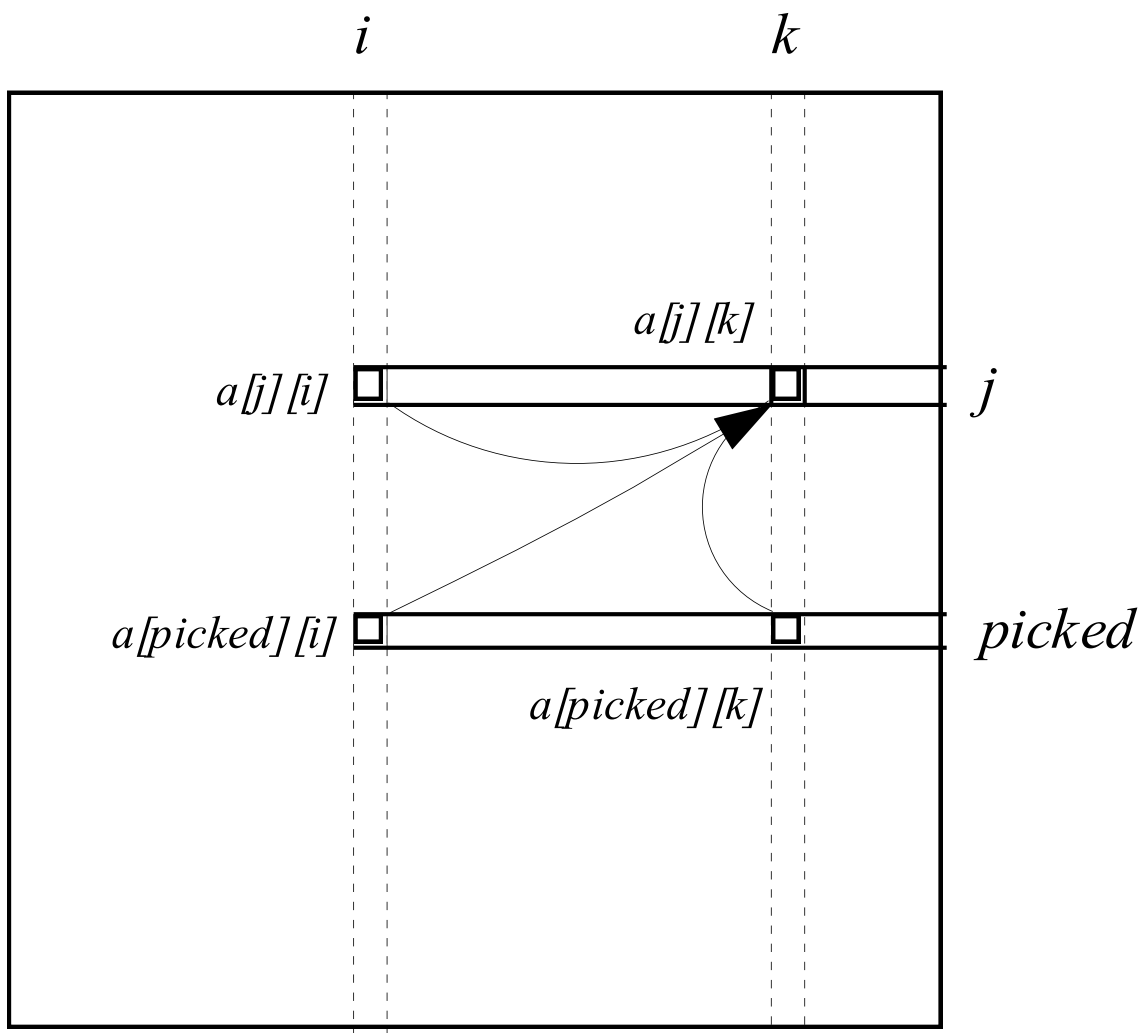
Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Example Use of MPI_MAXLOC

```
struct {
   double value;
   int    index;
} local, global;
...
local.value = fabs(a[j][i]);
local.index = j;
...
MPI_Allreduce (&local, &global, 1,
   MPI_DOUBLE_INT, MPI_MAXLOC,
   MPI_COMM_WORLD);
```

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Second Communication per Iteration



Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Communication Complexity

- Complexity of tournament: $\Theta(\log p)$
- Complexity of broadcasting pivot row: $\Theta(n \log p)$
- A total of $n - 1$ iterations
- Overall communication complexity: $\Theta(n^2 \log p)$

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Problems

- Algorithm breaks parallel execution into computation and communication phases
- Processes not performing computations during the broadcast steps
- Time spent doing broadcasts is large enough to ensure poor scalability

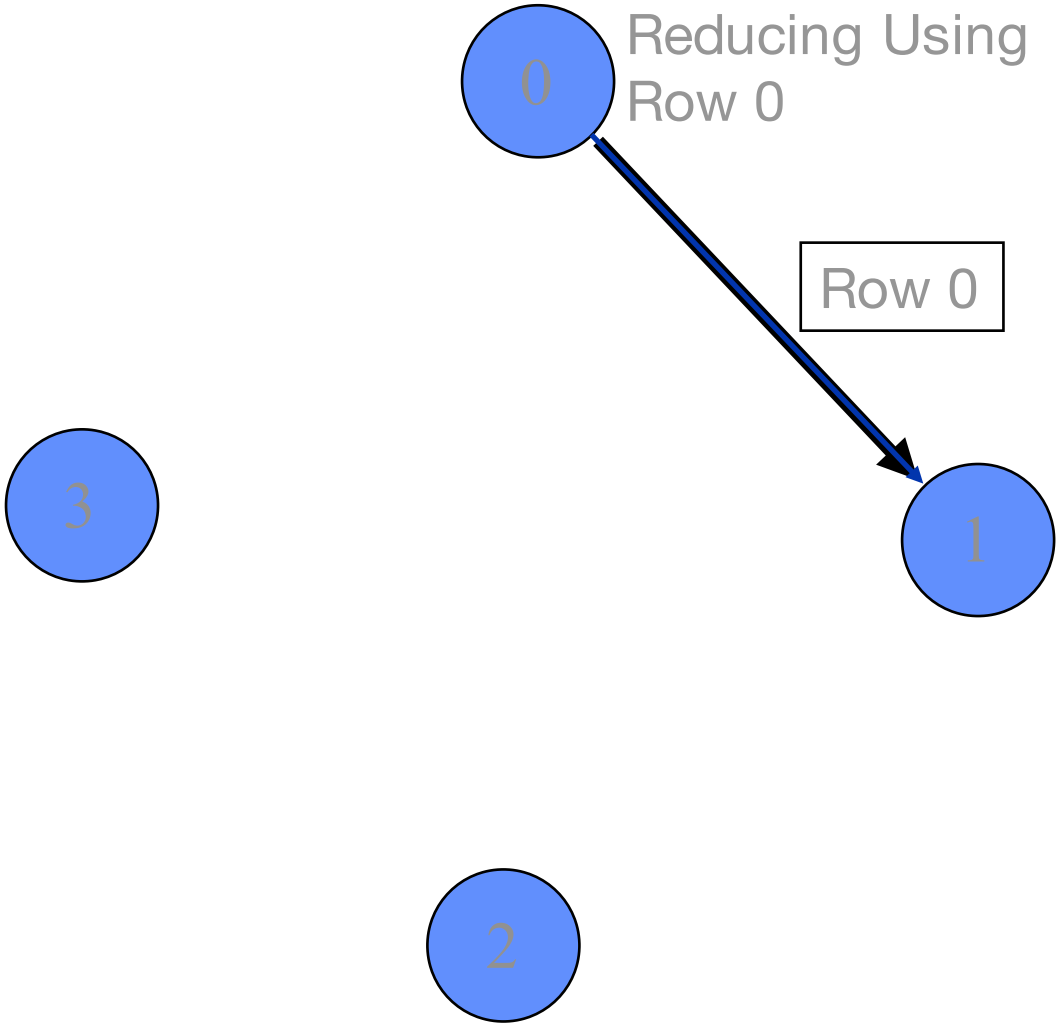
Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

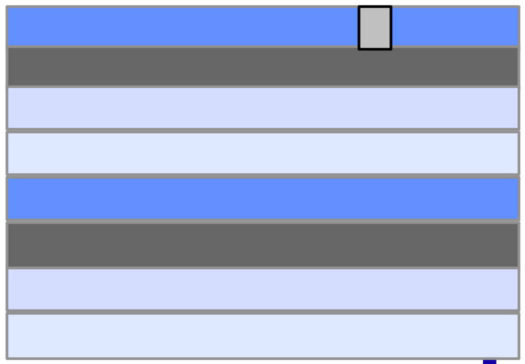
Pipelined, Row-Oriented Algorithm

- Want to overlap communication time with computation time
- We could do this if we knew in advance the row used to reduce all the other rows.
- Let's pivot columns instead of rows!
- In iteration *i* we can use row *i* to reduce the other rows.

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

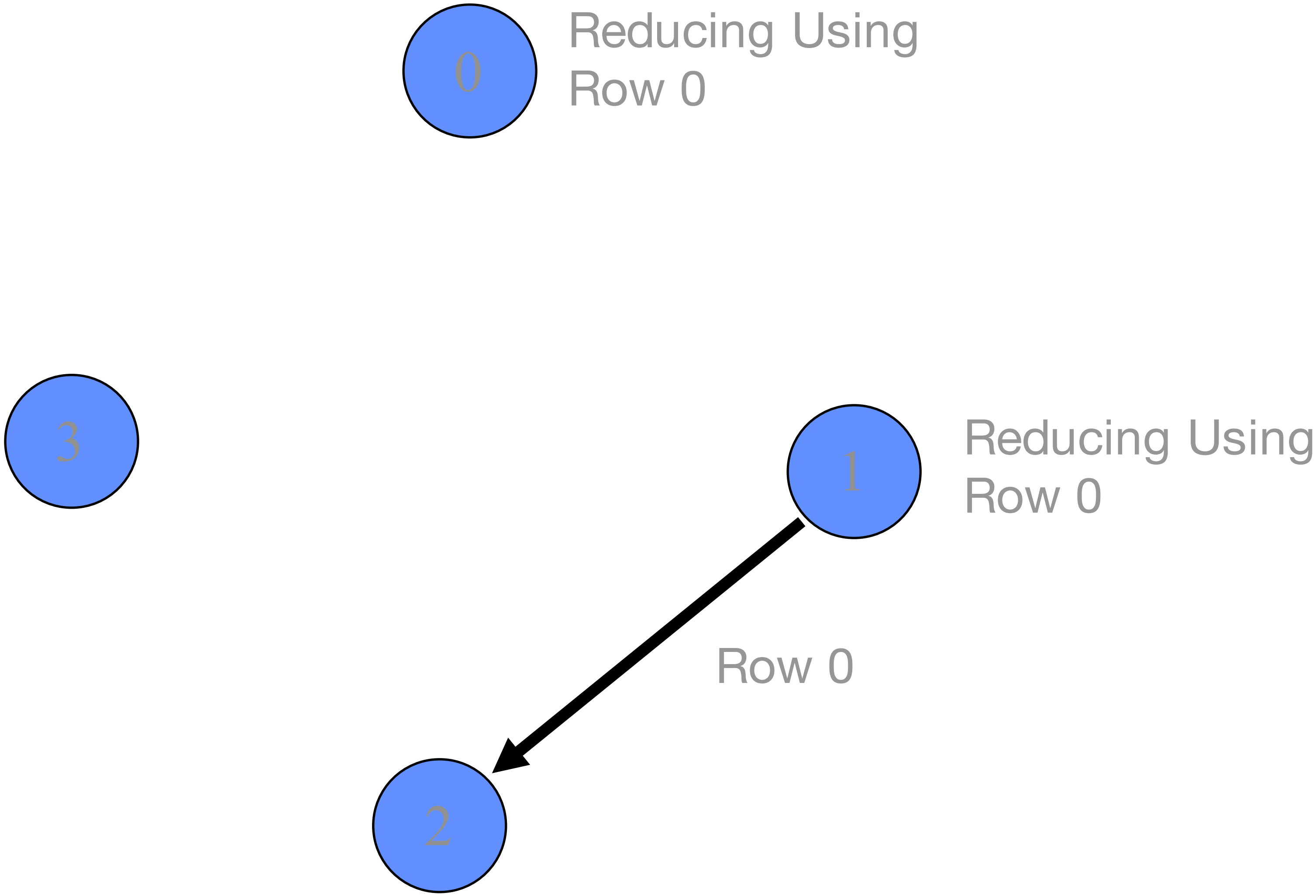
Communication Pattern





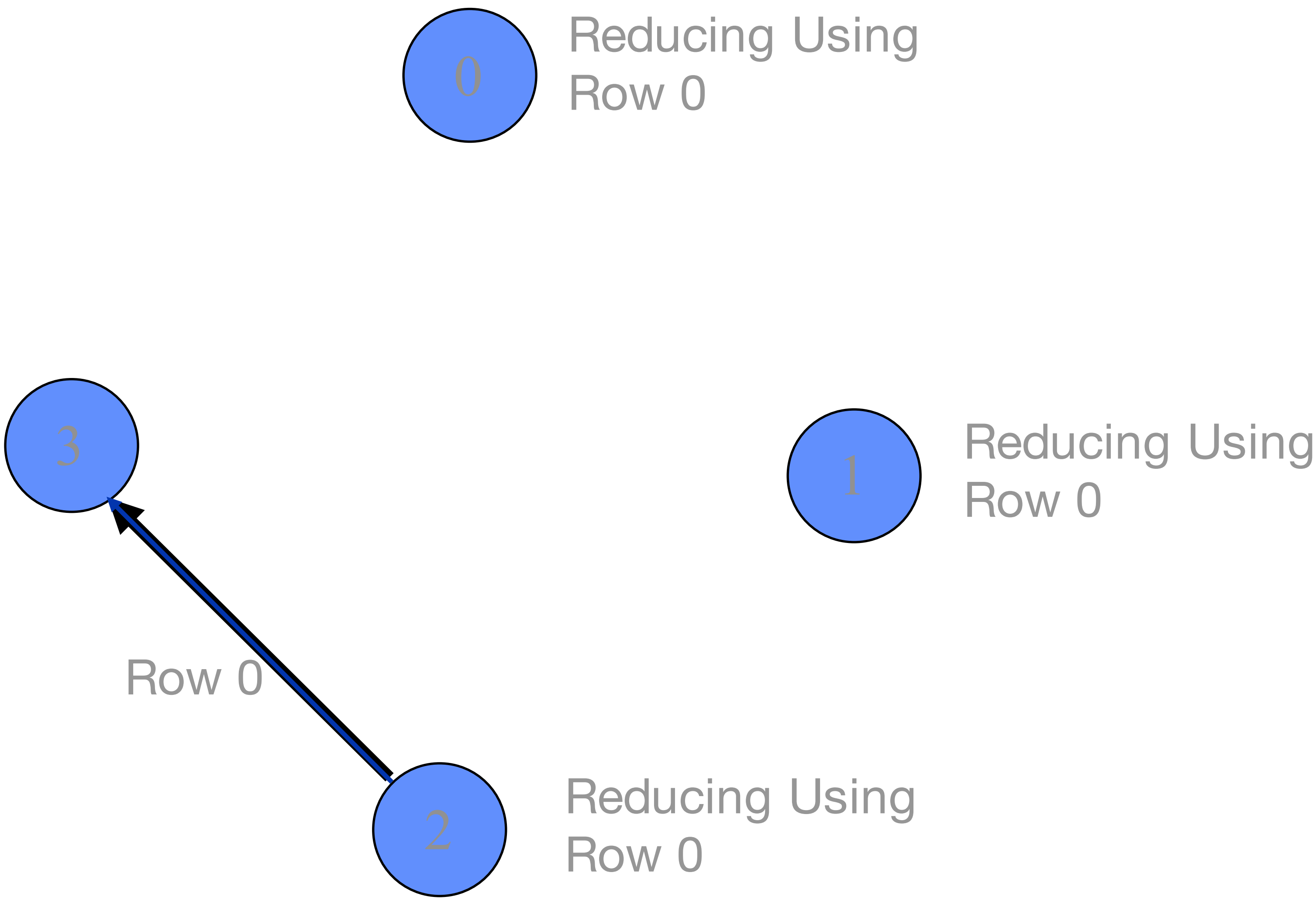
Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Communication Pattern



Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Communication Pattern



Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

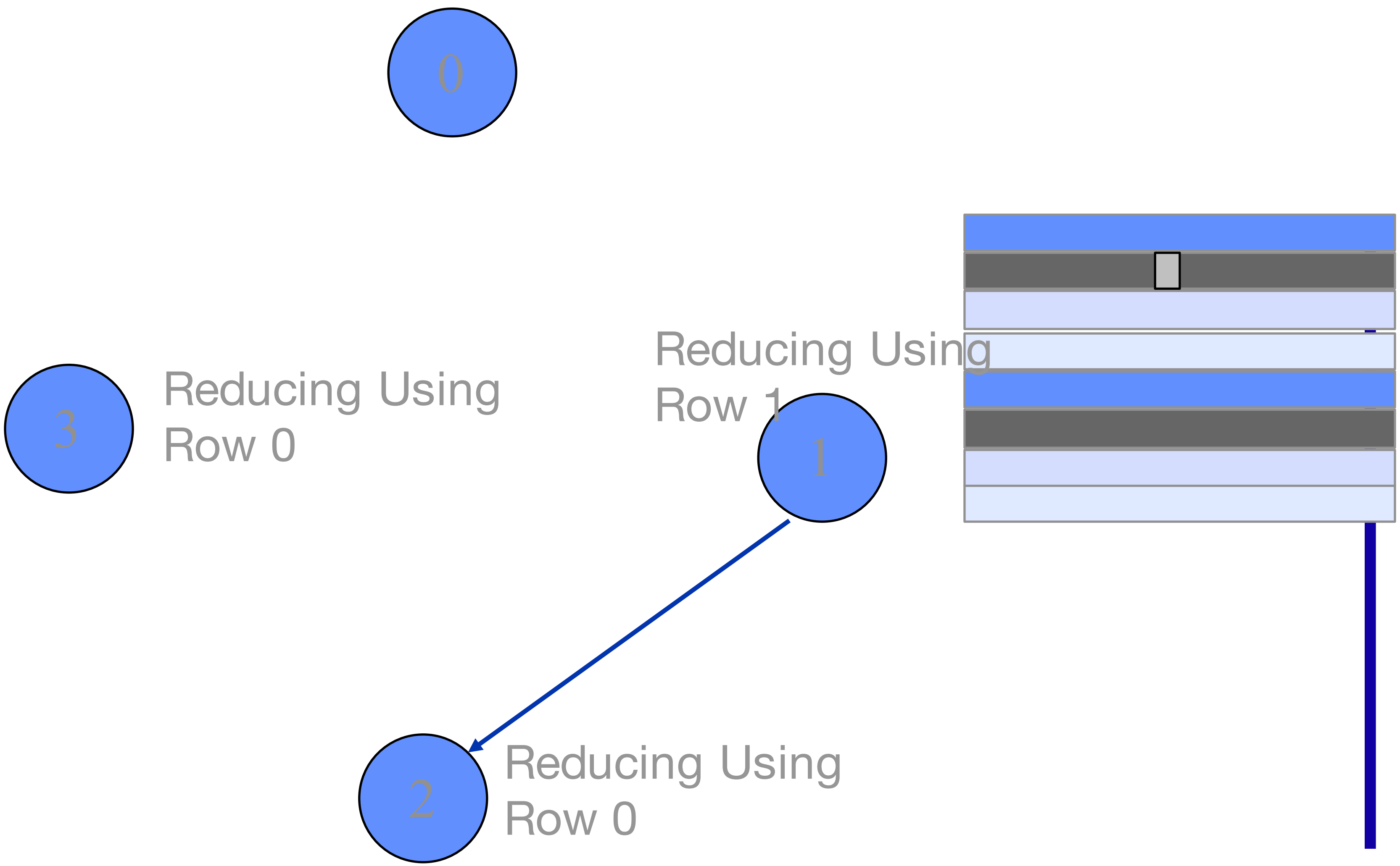
Communication Pattern

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Communication Pattern

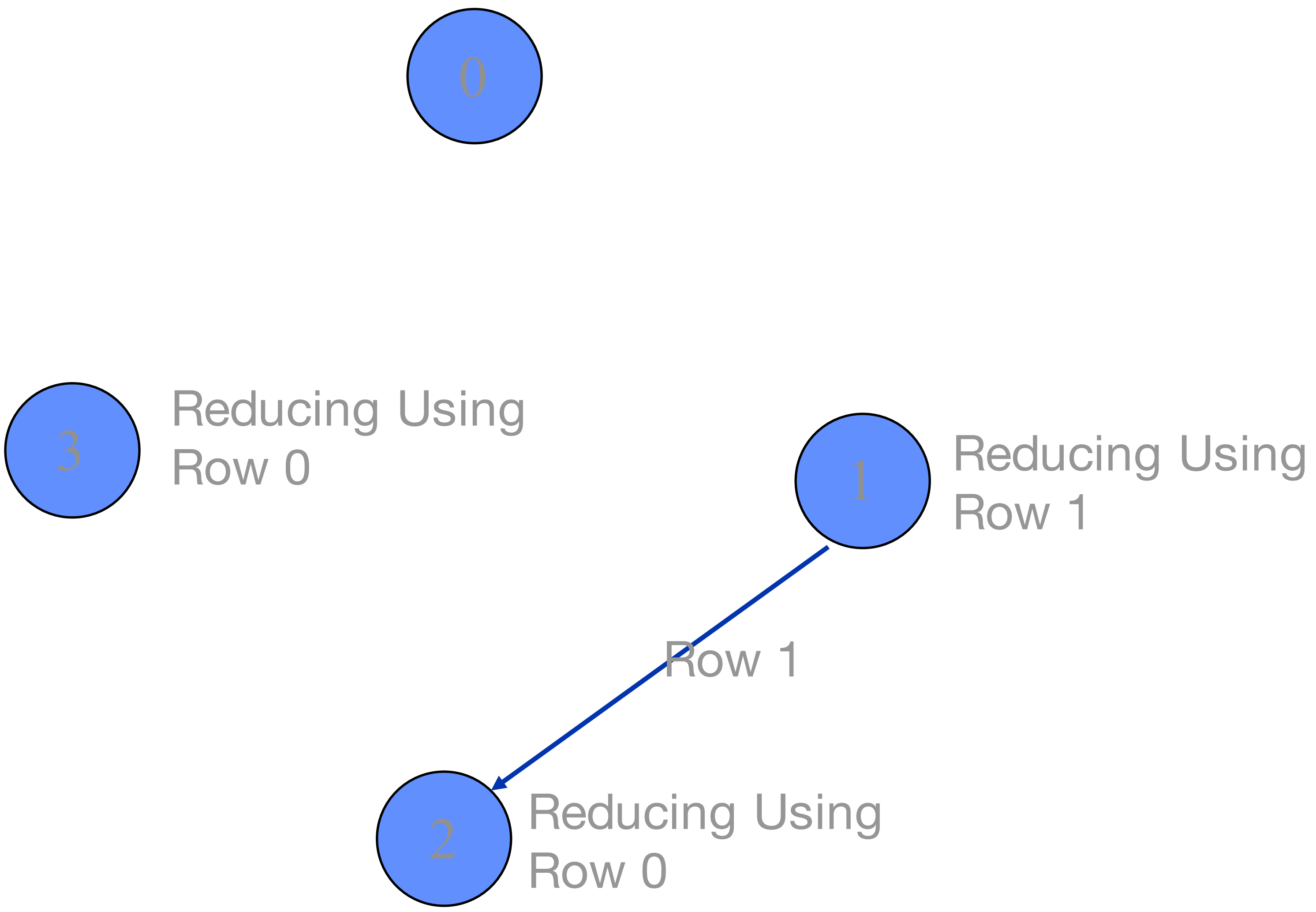
Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Communication Pattern



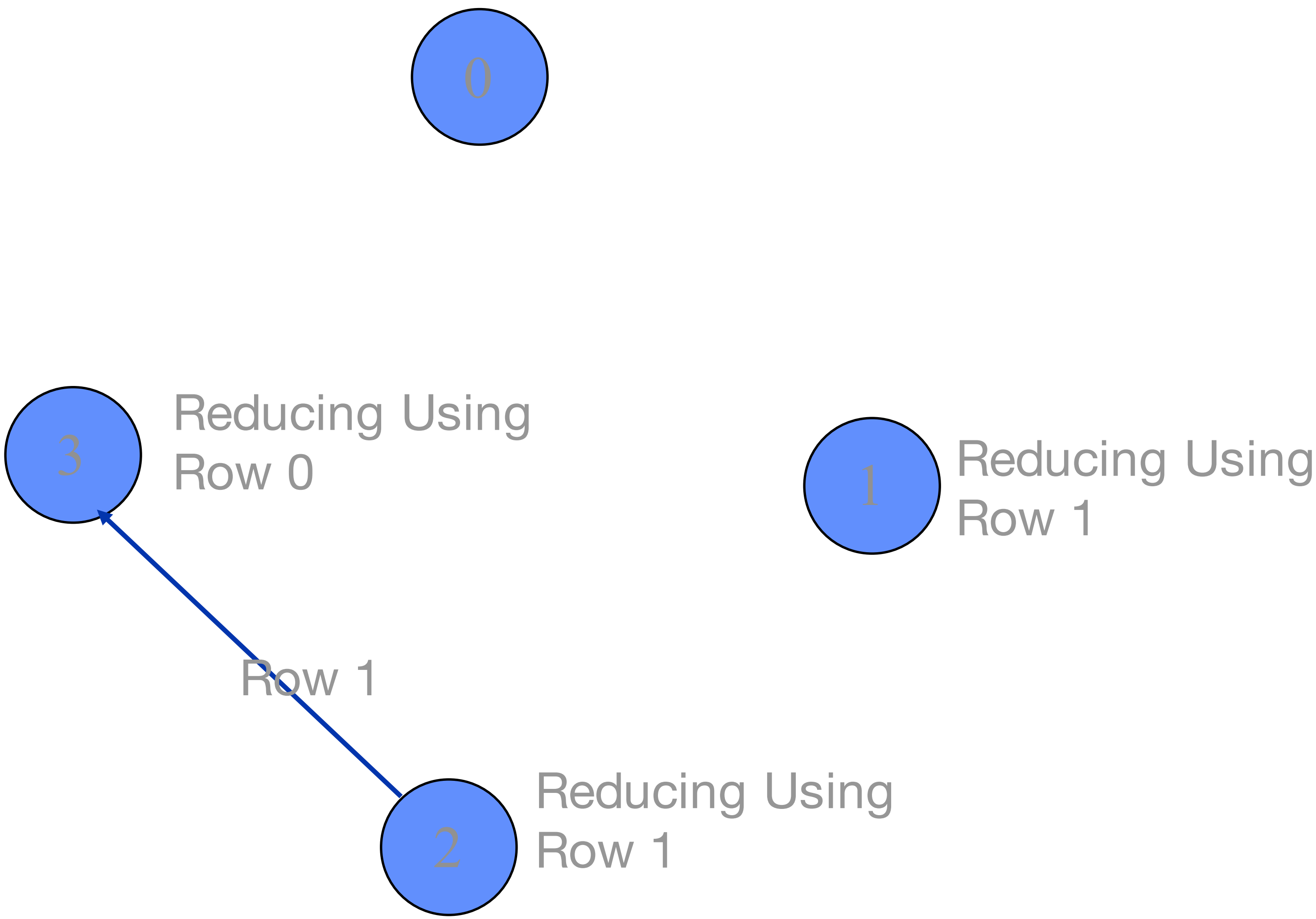
Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Communication Pattern



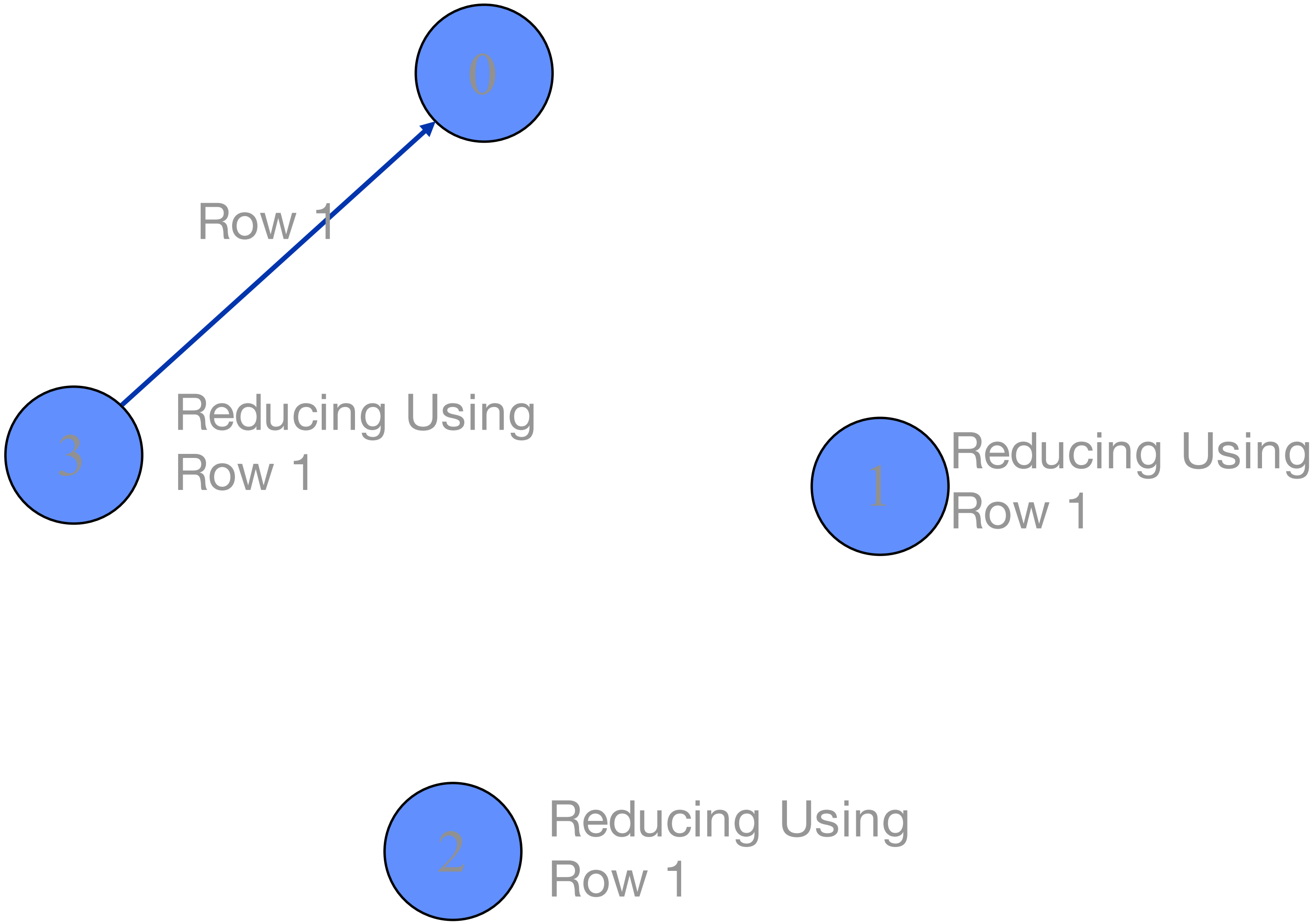
Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Communication Pattern



Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Communication Pattern



Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Analysis (1/2)

- Total computation time: $\Theta(n^3/p)$
- Total message transmission time: $\Theta(n^2)$
- When n large enough, message transmission time completely overlapped by computation time
- Message start-up not overlapped: $\Theta(n)$
- Parallel overhead: $\Theta(np)$

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Sparse Systems

- Gaussian elimination not well-suited for sparse systems
- Coefficient matrix gradually fills with nonzero elements
- Result
  - Increases storage requirements
  - Increases total operation count

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Iterative Methods

- Iterative method: algorithm that generates a series of approximations to solution's value
- Require less storage than direct methods
- Since they avoid computations on zero elements, they can save a lot of computations

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Summary (1/2)

- Solving systems of linear equations
  - Direct methods
  - Iterative methods
- Parallel designs for
  - Back substitution
  - Gaussian elimination
  - Conjugate gradient method

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

Summary (2/2)

- Superiority of one algorithm over another depends on size of problem, number of processors, characteristics of parallel computer
- Overlapping communications with computations can be key to scalability