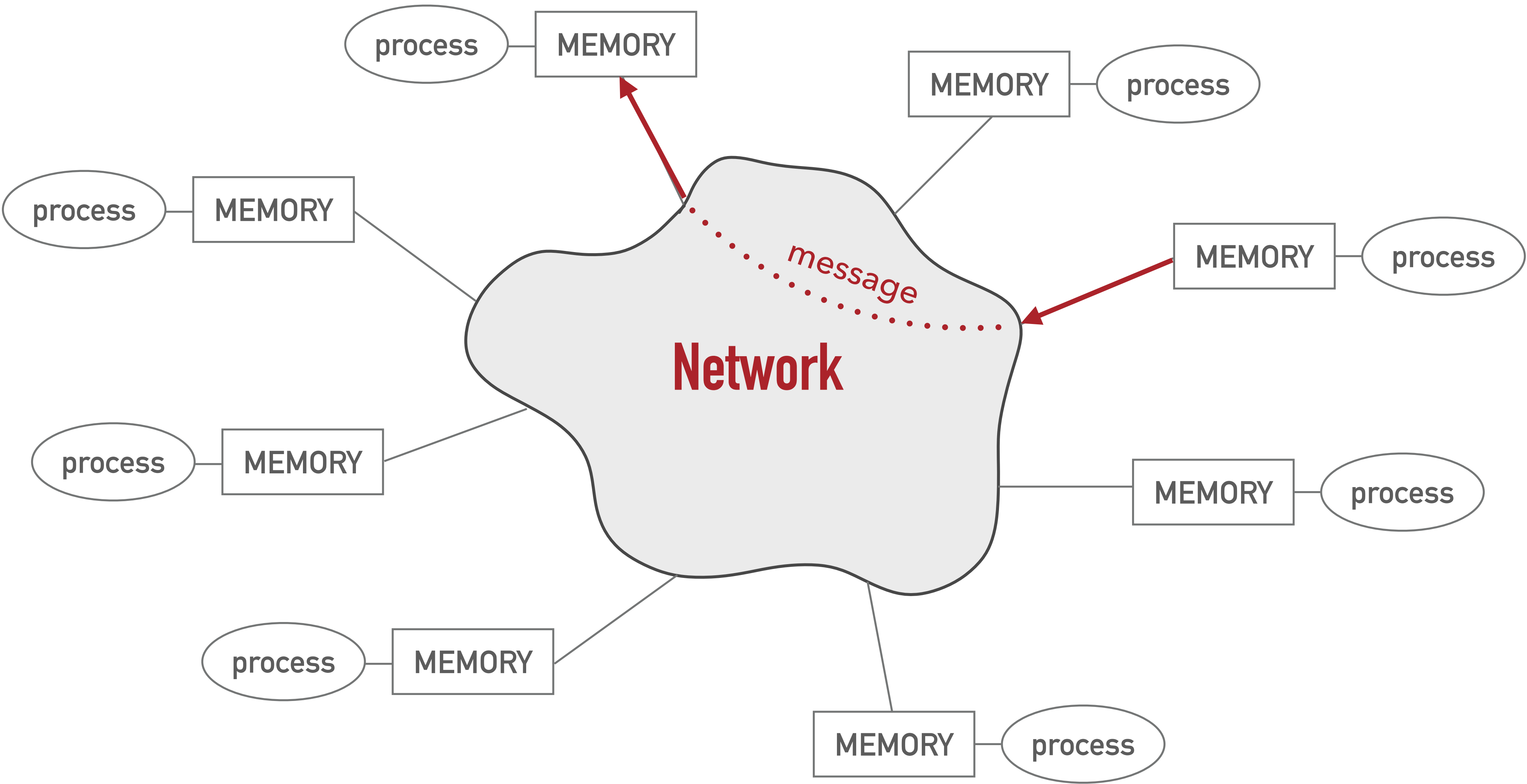




ME 471/571

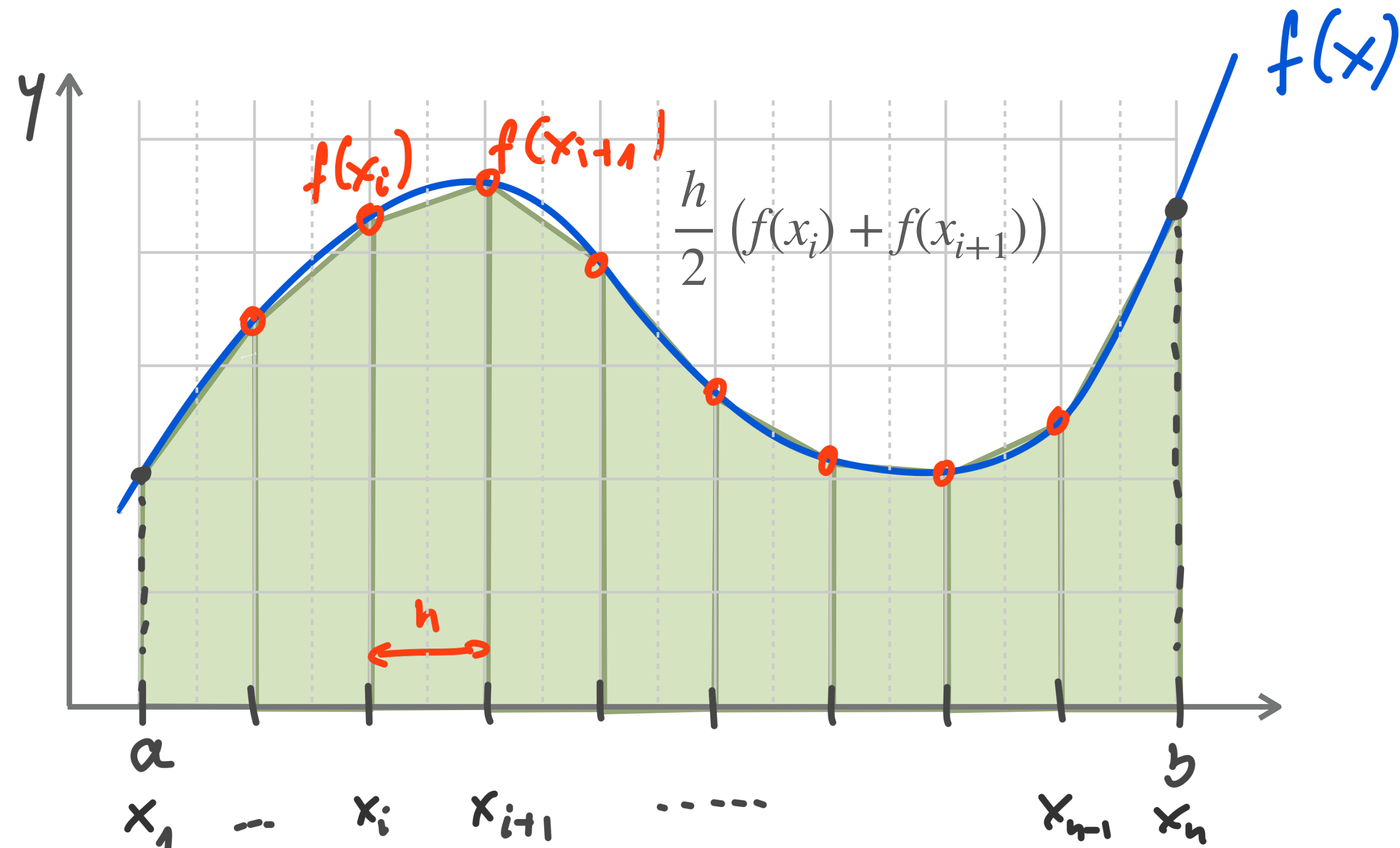
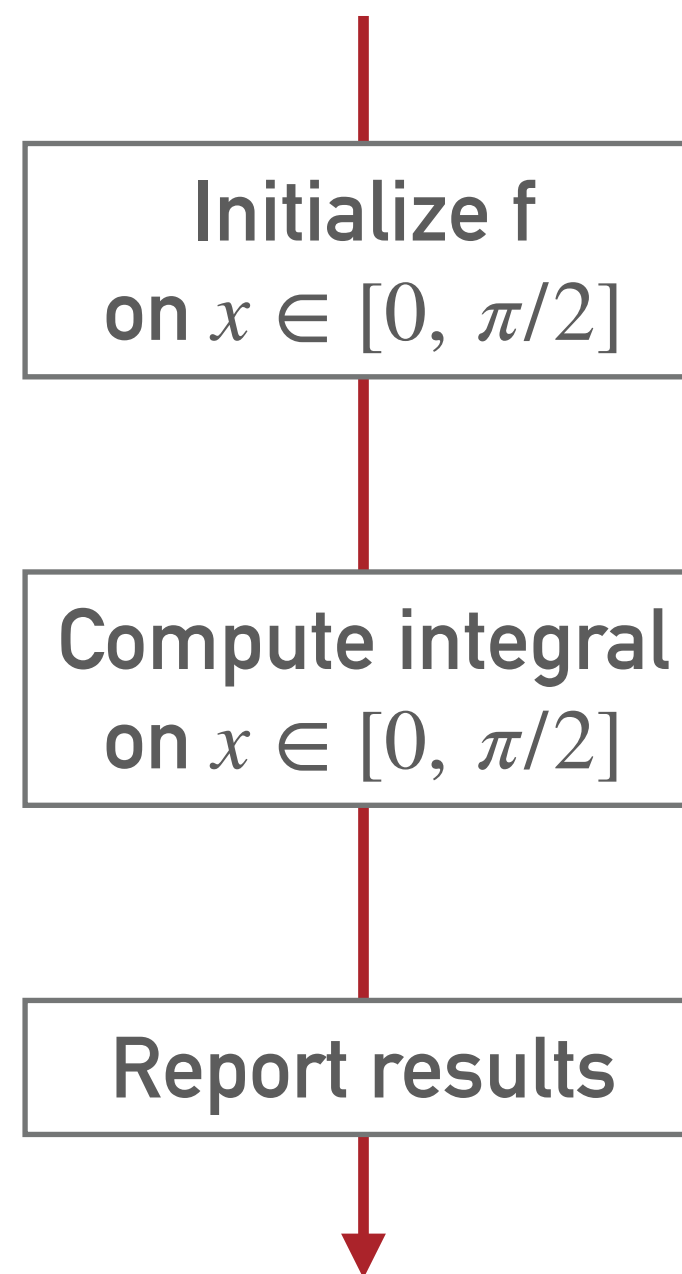
Week 4 - Collective Communication

MESSAGE PASSING MODEL



EXAMPLE – INTEGRATE A FUNCTION

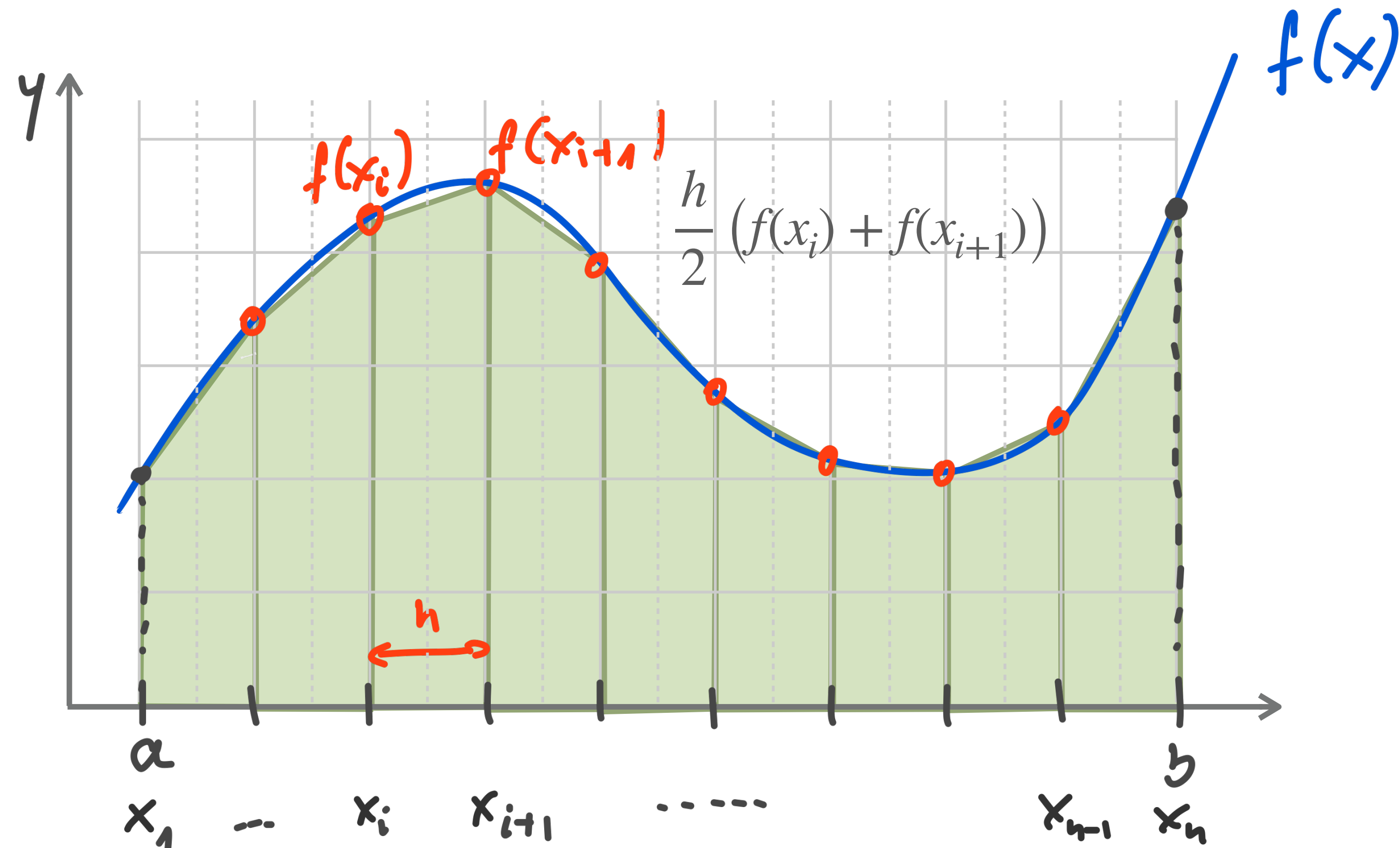
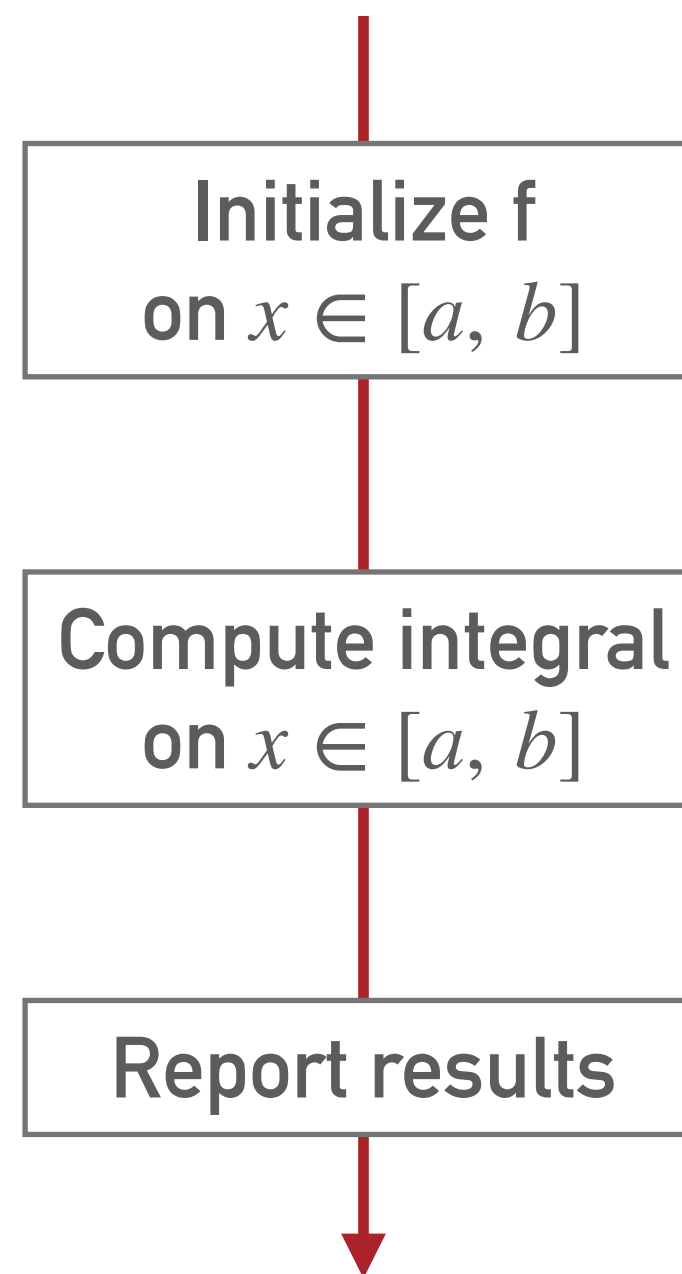
Integrate $f(x) = \sin(x)$ on $x \in [0, 2\pi]$ using trapezoidal rule:

$$\int_a^b f(x) \, dx \approx \sum_{i=0}^{N-1} \frac{1}{2} h (f(x_i) + f(x_{i+1}))$$


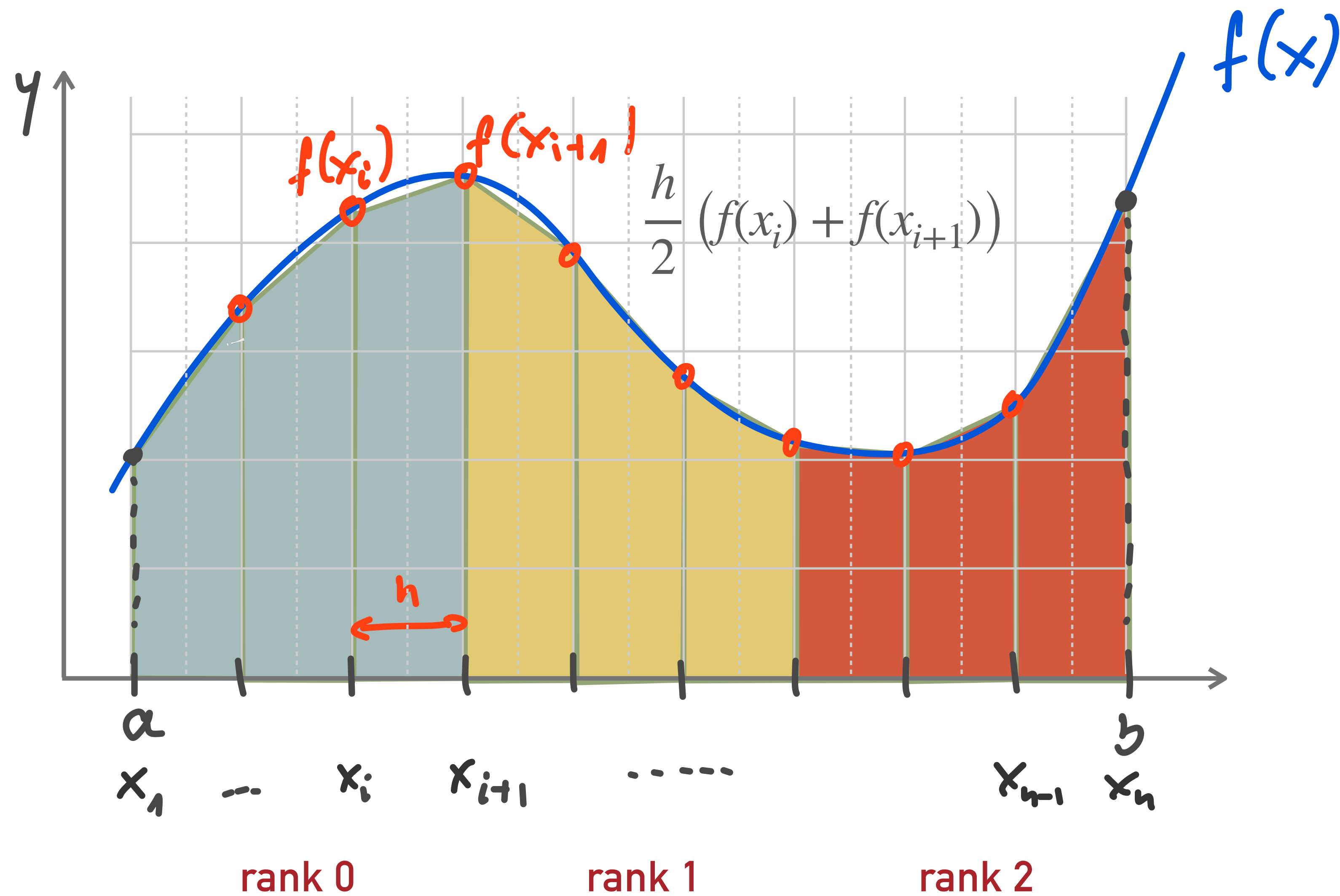
EXAMPLE – INTEGRATE A FUNCTION

Integrate $f(x)$ using trapezoidal rule:

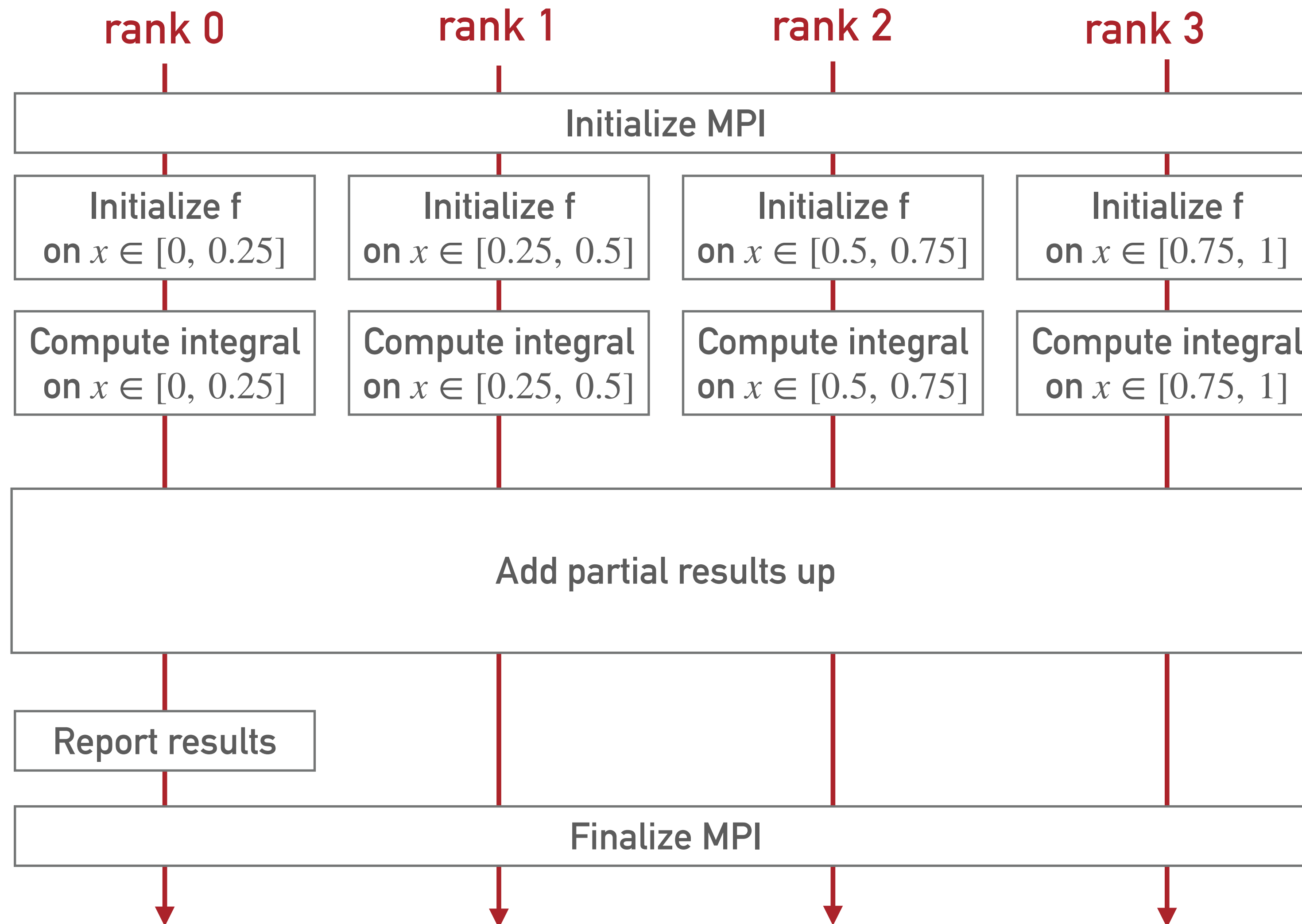
$$\int_a^b f(x) \, dx \approx \sum_{i=0}^{N-1} \frac{1}{2} h (f(x_i) + f(x_{i+1}))$$



EXAMPLE – INTEGRATE A FUNCTION

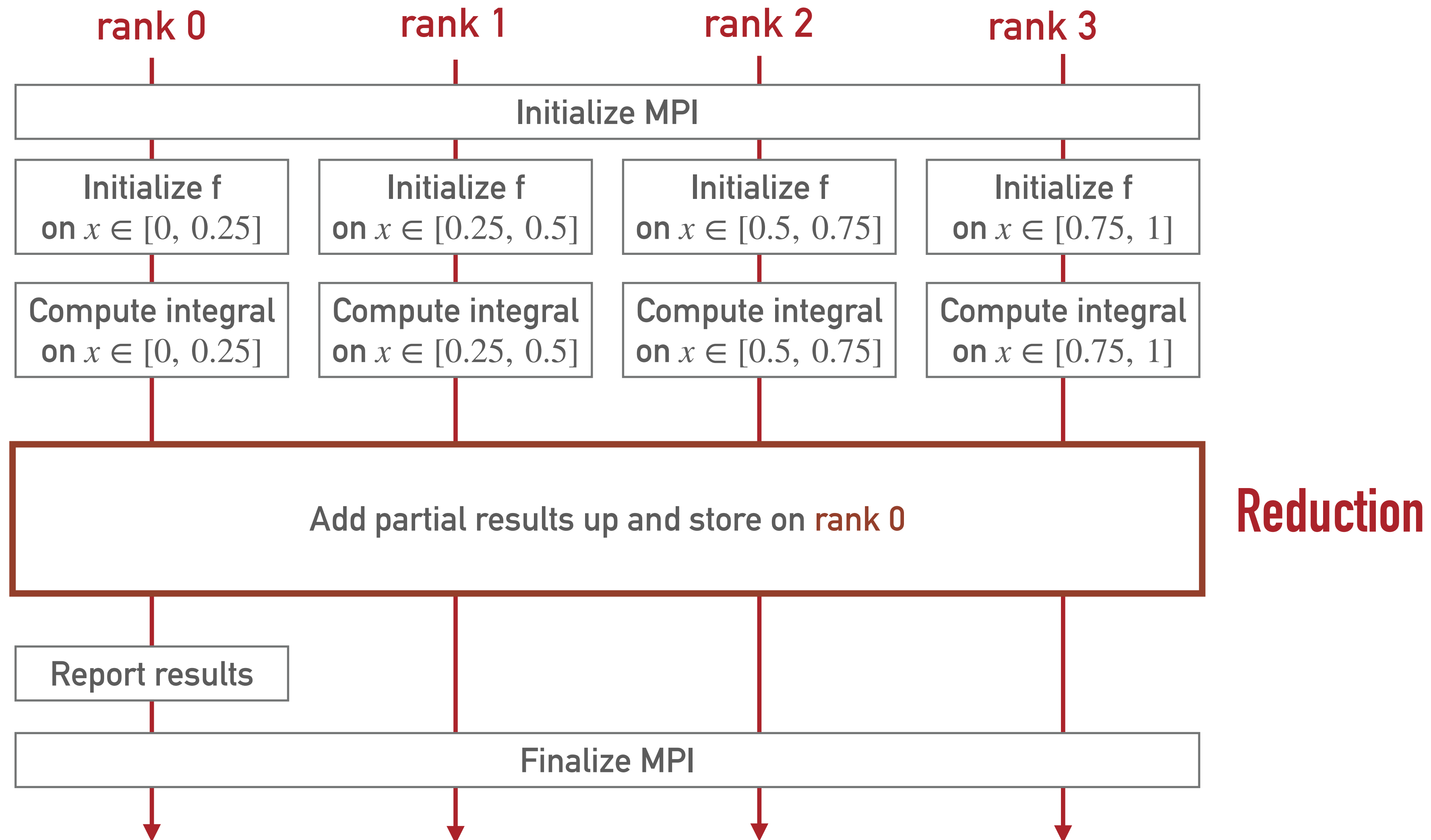


TRAPEZOIDAL RULE EXAMPLE

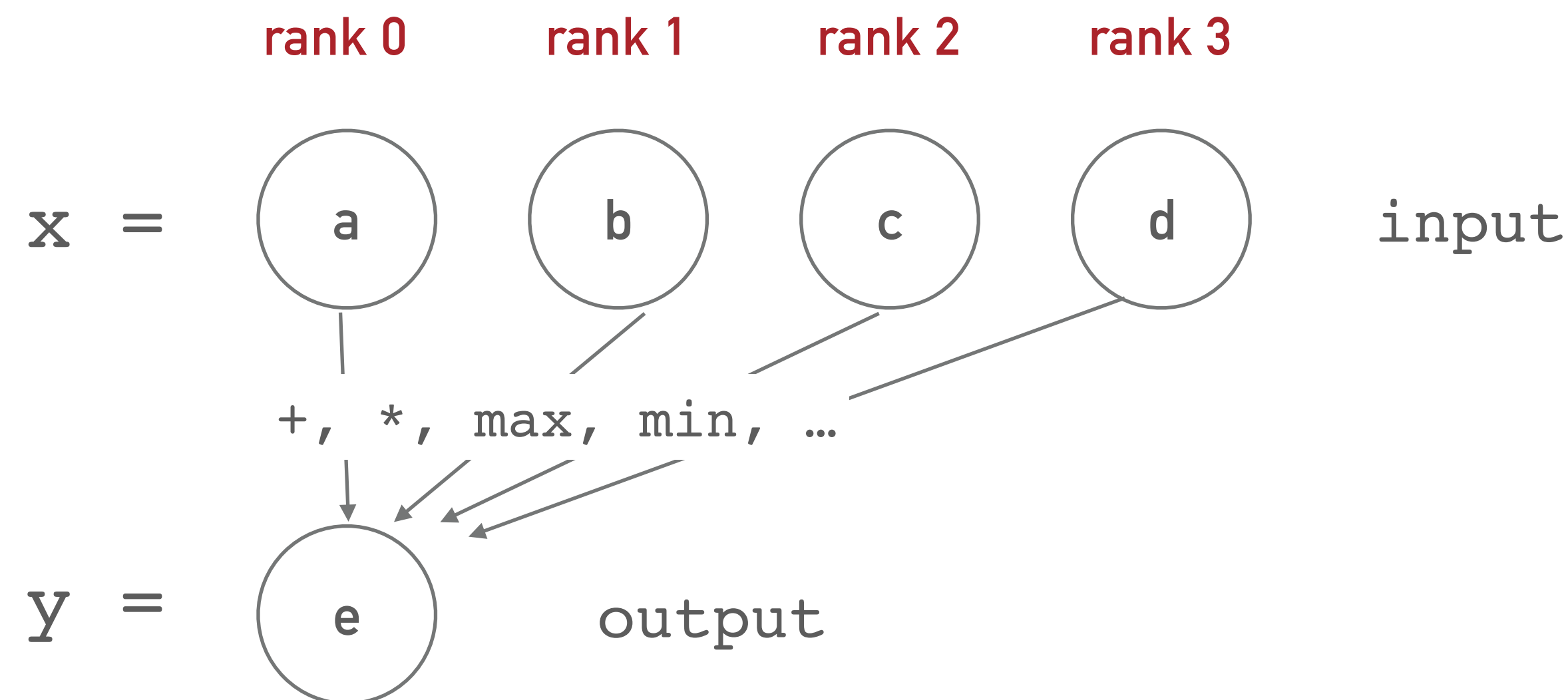


$$f(x) = \frac{4}{1 + x^2}$$
$$x \in [0, 1]$$

TRAPEZOIDAL RULE EXAMPLE



COLLECTIVE COMMUNICATION PATTERNS



Reduction

```
MPI_Reduce(input, output, count, datatype, operation, root, communicator);
```

*which variable
to reduce*

*variable for
the result*

*how many data elements
to send per rank*

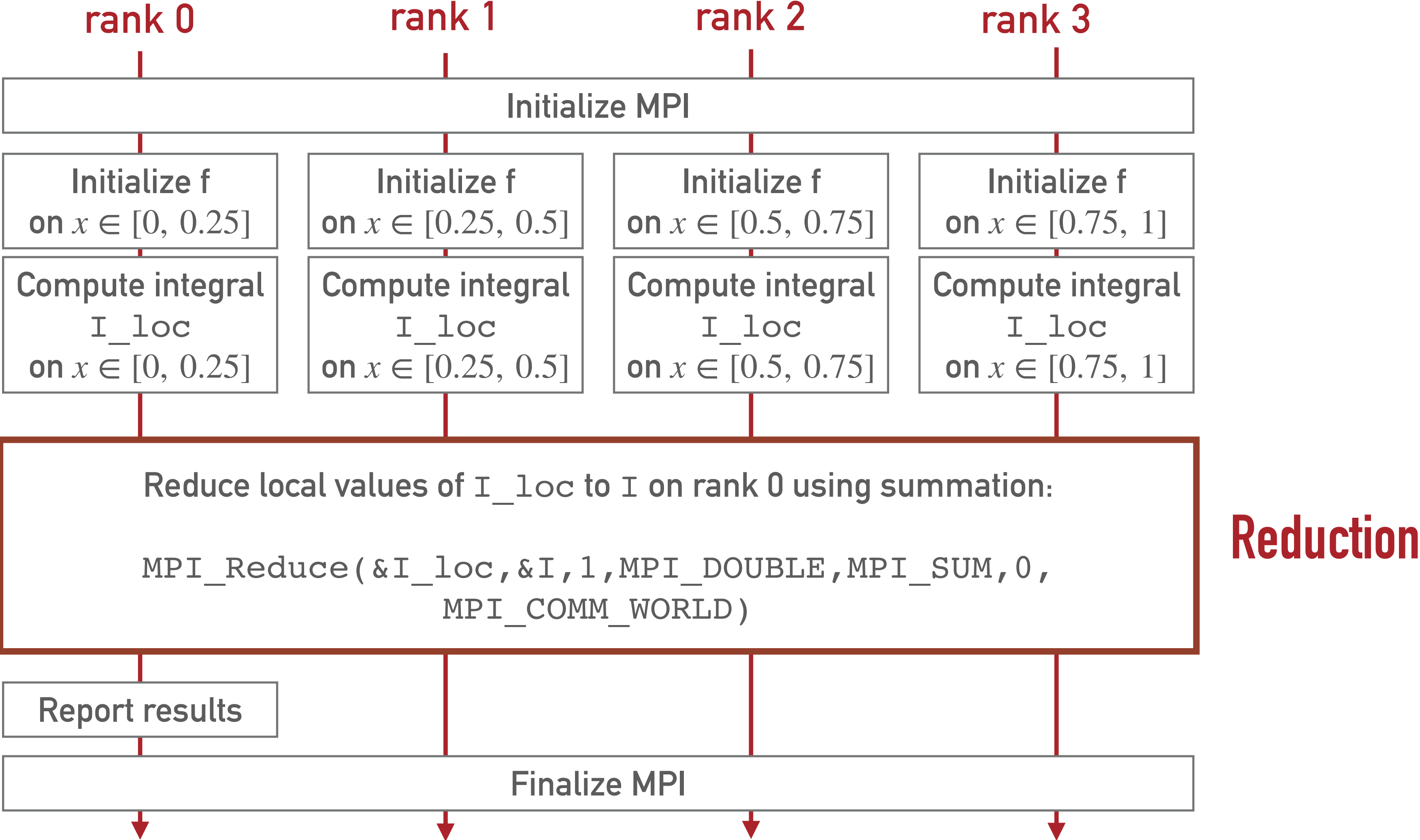
what type?

- MPI_SUM
- MPI_PROD
- MPI_MAX
- MPI_MIN
- ...

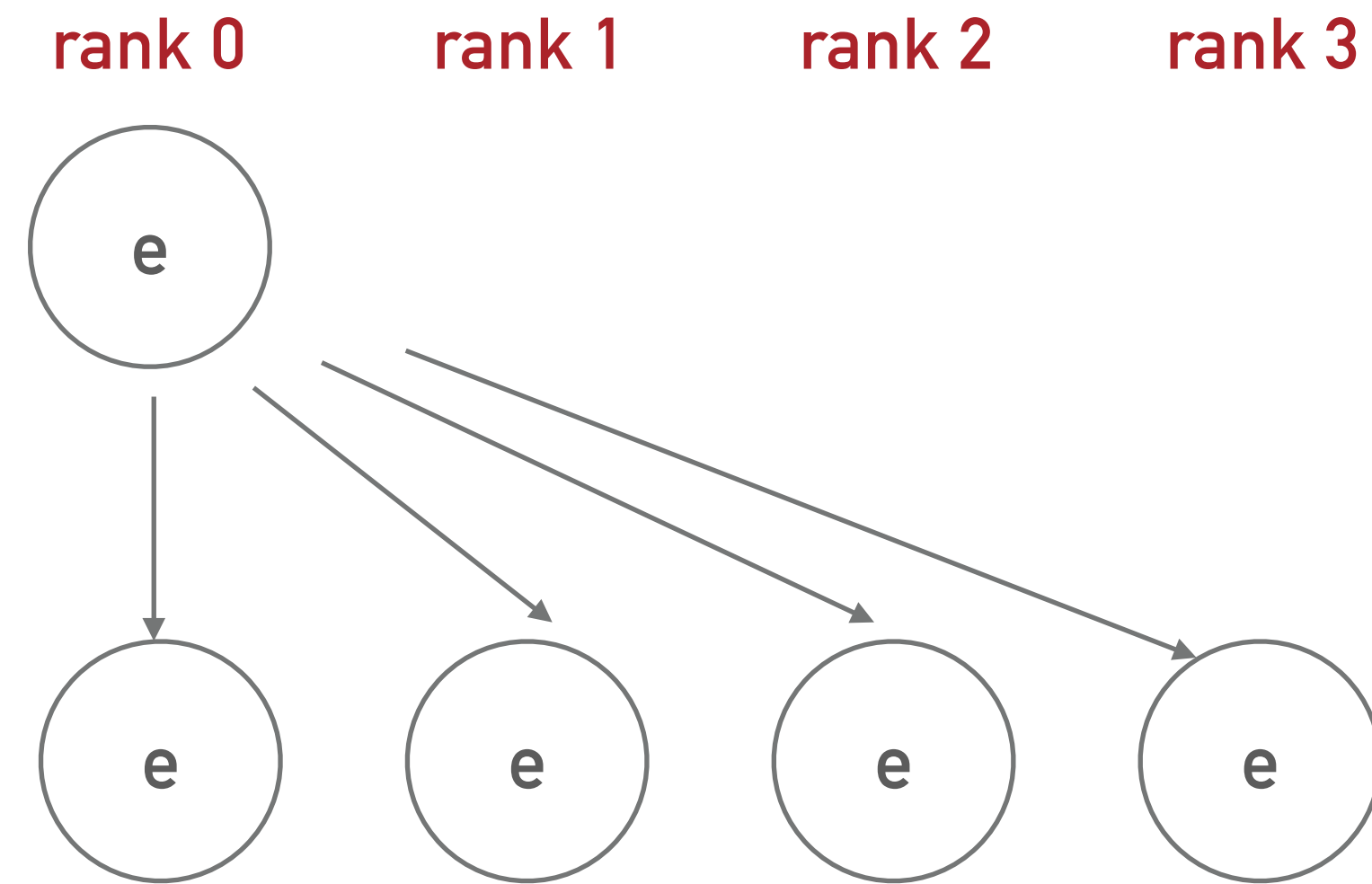
*which rank
is receiving?*

*which
communicat
or to use?*

TRAPEZOIDAL RULE EXAMPLE



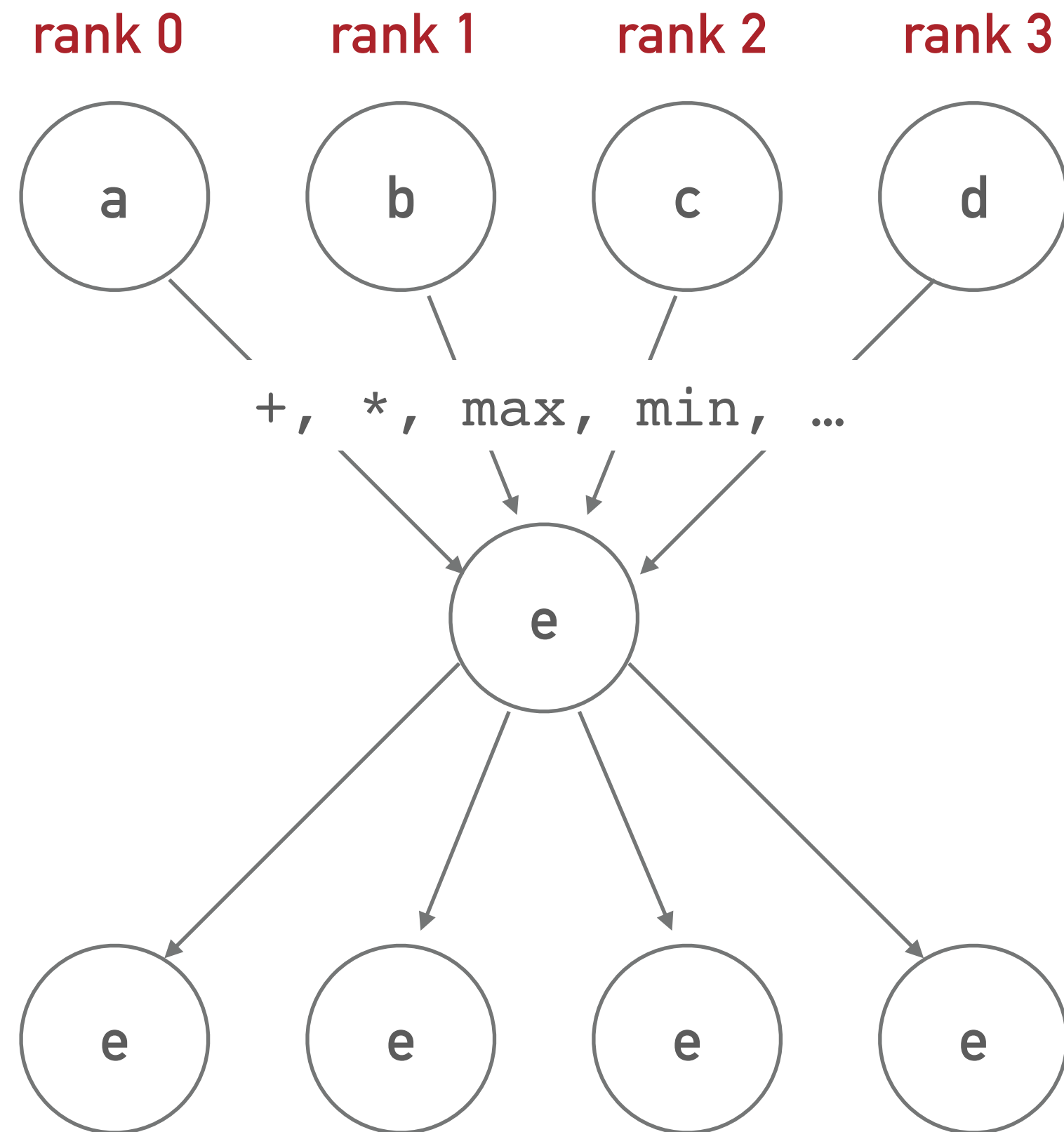
COLLECTIVE COMMUNICATION PATTERNS



Broadcast

```
MPI_Bcast(data, count, datatype, root, communicator);
```

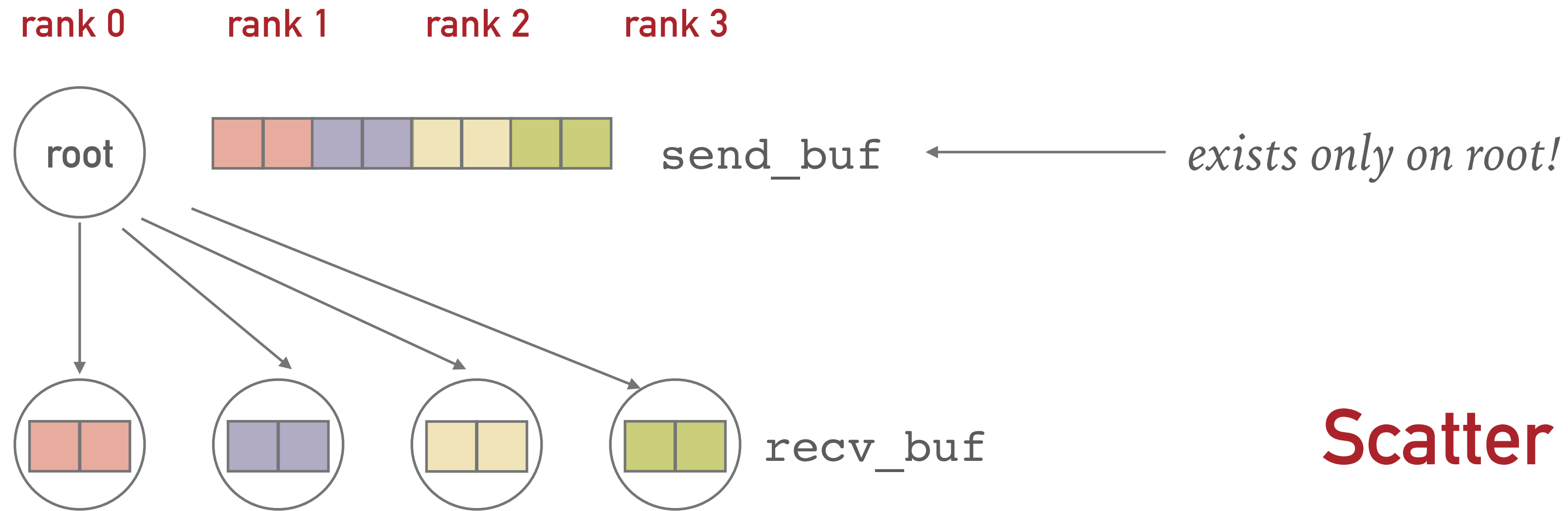
COLLECTIVE COMMUNICATION PATTERNS



Reduction + Broadcast = Allreduce

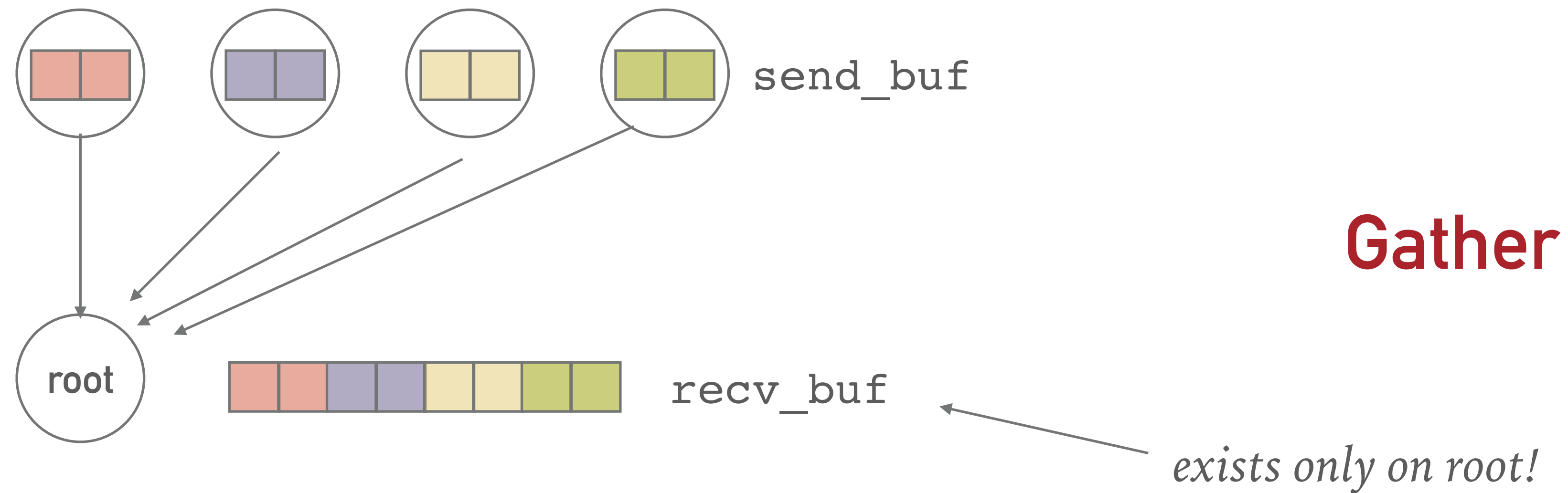
```
MPI_Allreduce(input, output, count, datatype, operator, communicator);
```

COLLECTIVE COMMUNICATION PATTERNS



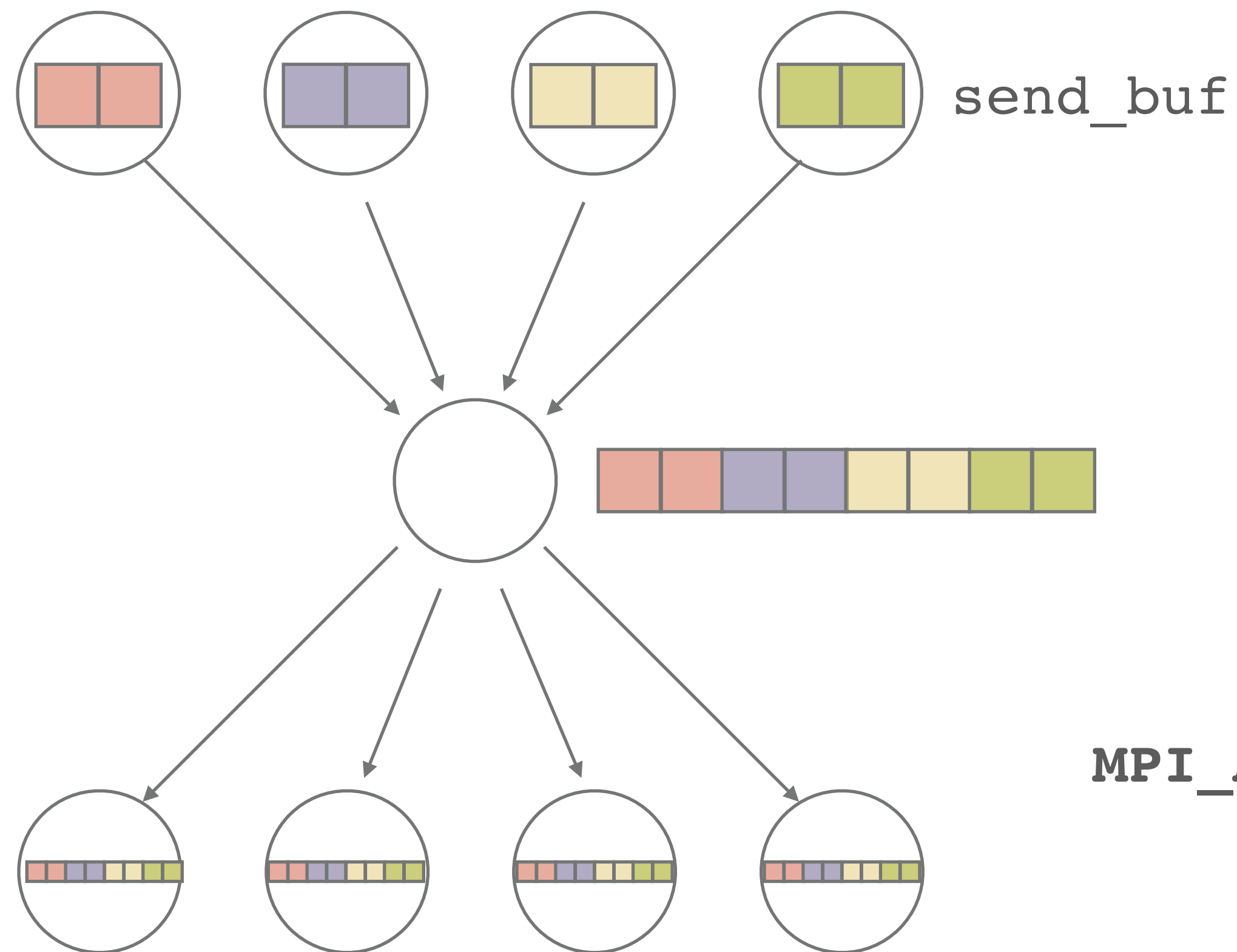
```
MPI_Scatter(send_buf, send_count, send_type,...  
            recv_buf, recv_count, recv_type,...  
            root, communicator);
```

COLLECTIVE COMMUNICATION PATTERNS



```
MPI_Gather(send_buf, send_count, send_type,...  
           recv_buf, recv_count, recv_type,...  
           root, communicator);
```

COLLECTIVE COMMUNICATION PATTERNS



Gather + Scatter = Allgather

```
MPI_Allgather(send_buf, send_count, send_type,...  
               recv_buf, recv_count, recv_type,...  
               communicator);
```


CLASSROOM EXAMPLE

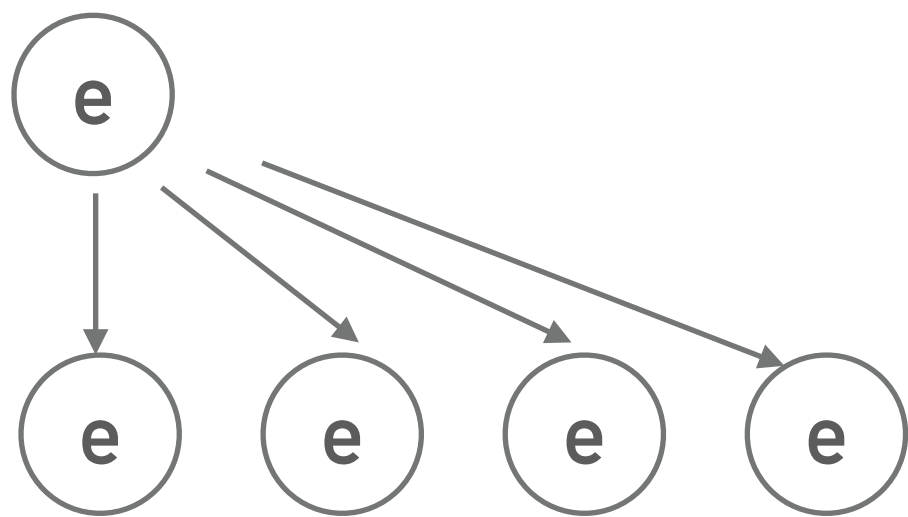
- 1. *An instructor is announcing the date of the test.*
- 2. *The students write the test and the instructor is collecting their individual papers.*
- 3. *The instructor gives the papers back to students.*
- 4. *The instructor computes the average score.*

broadcast

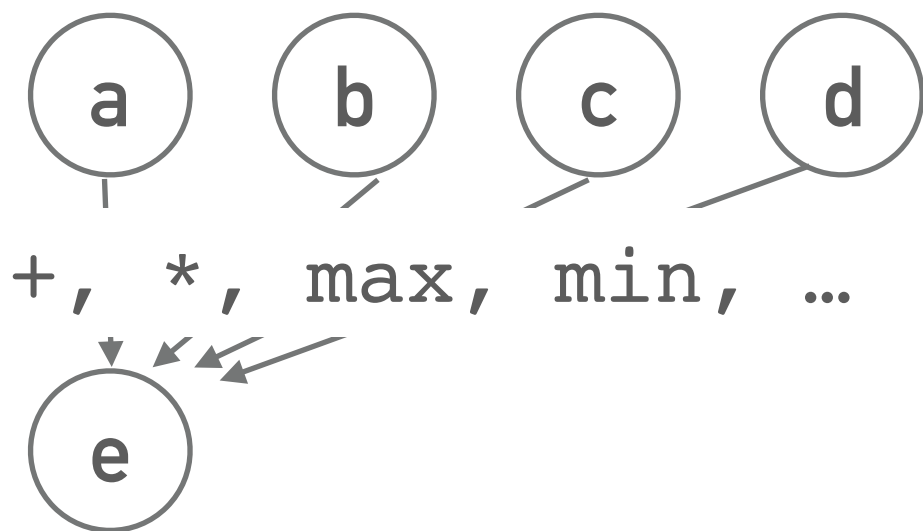
gather

scatter

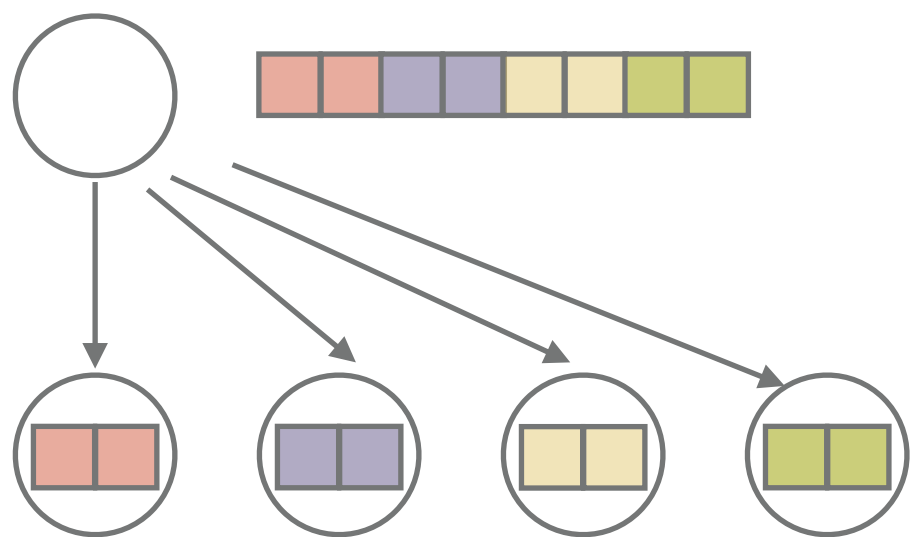
reduction



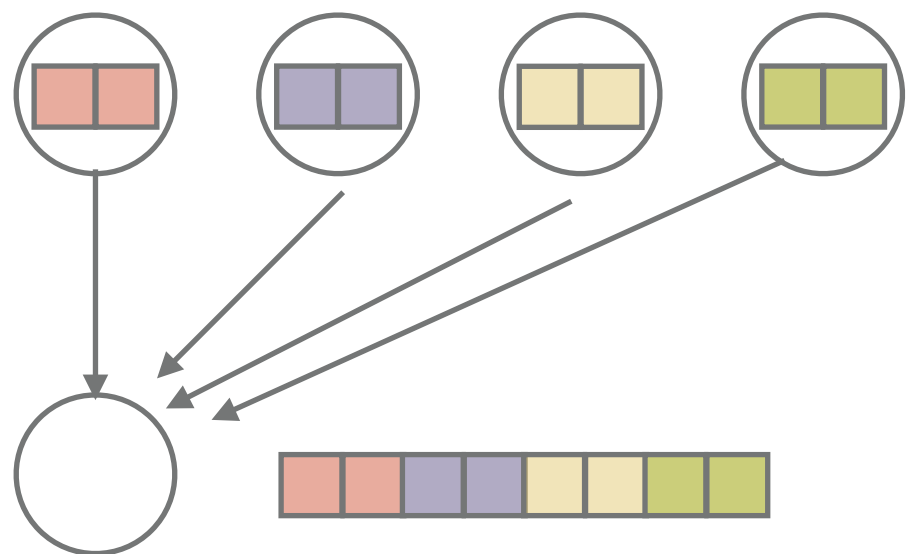
Broadcast



Reduction

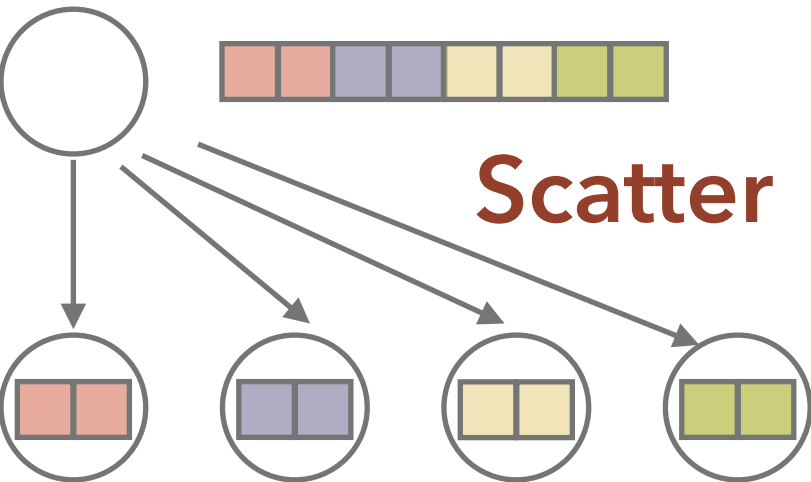
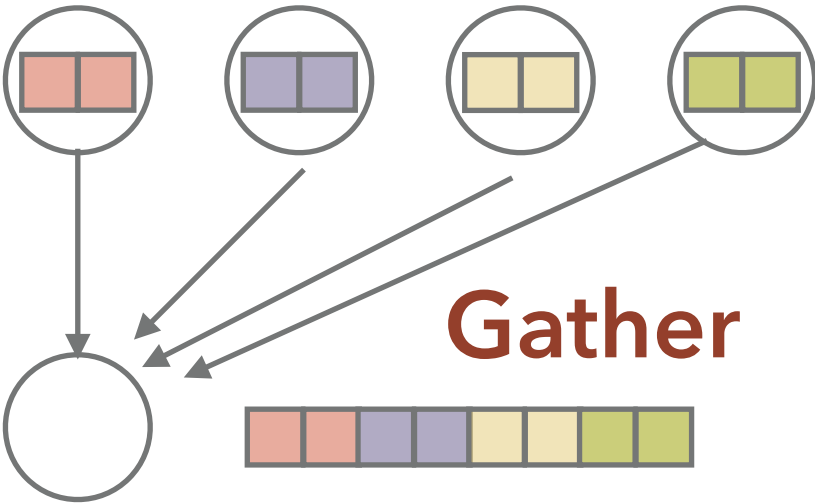
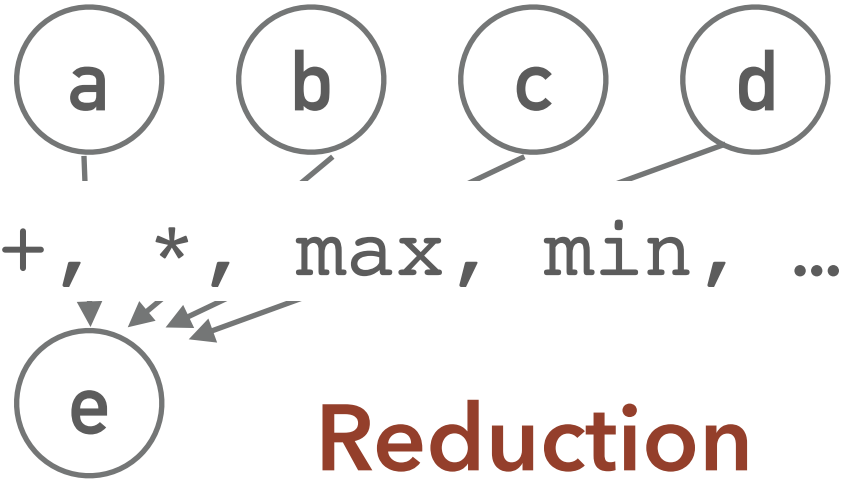
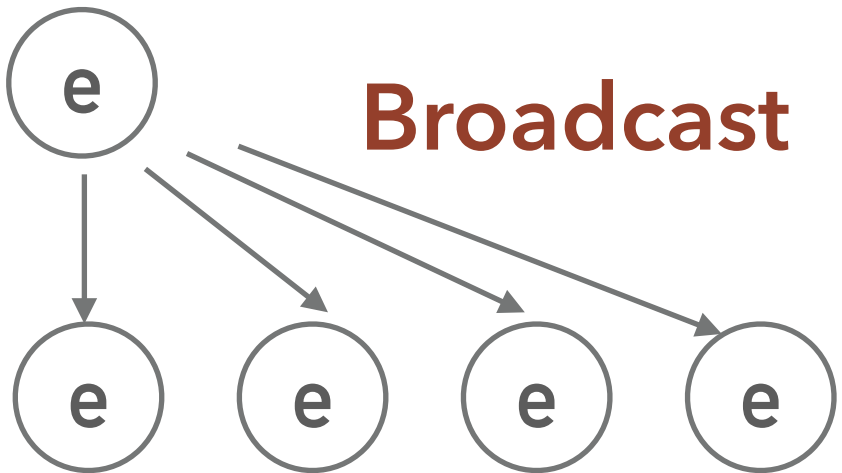
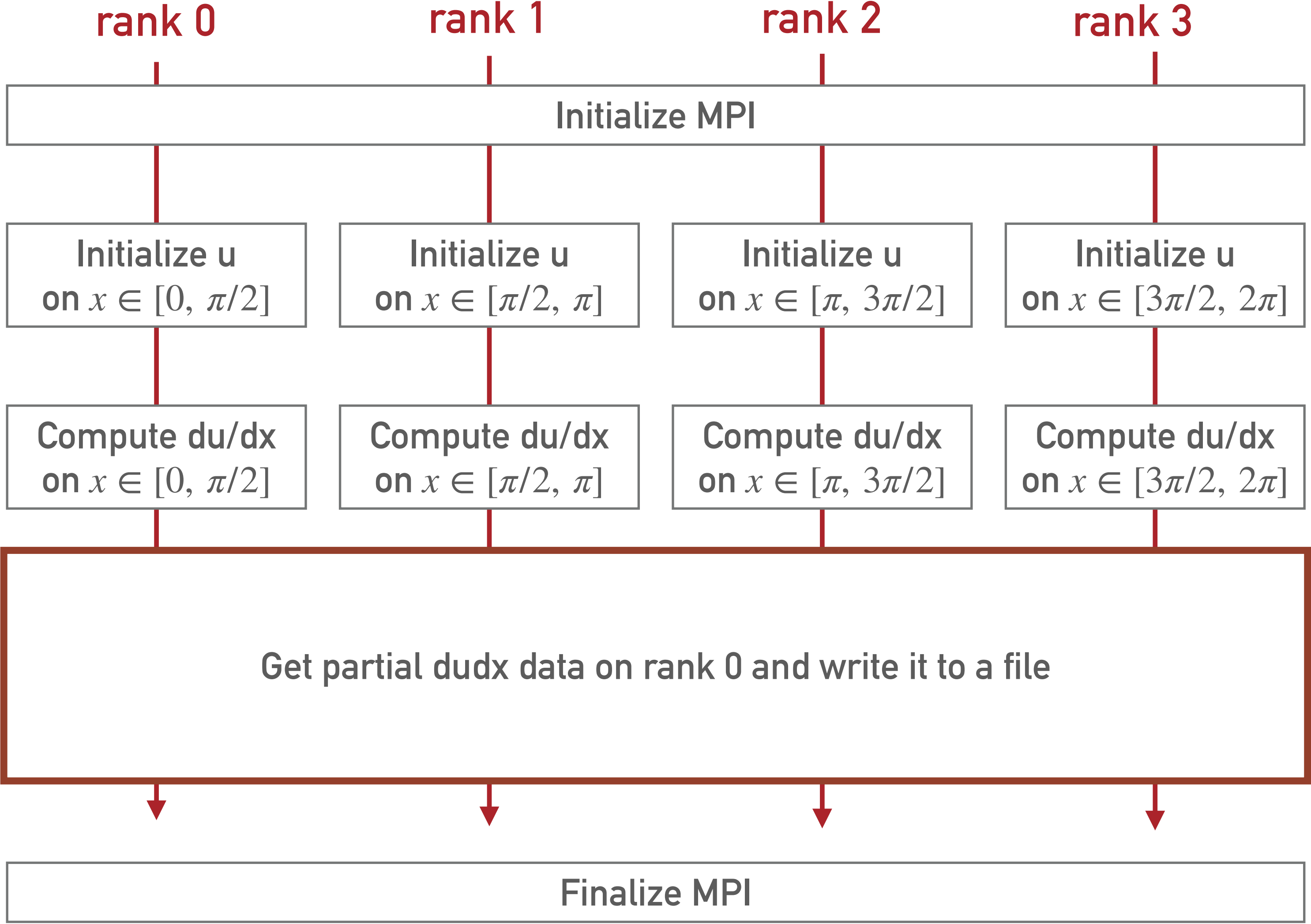


Scatter

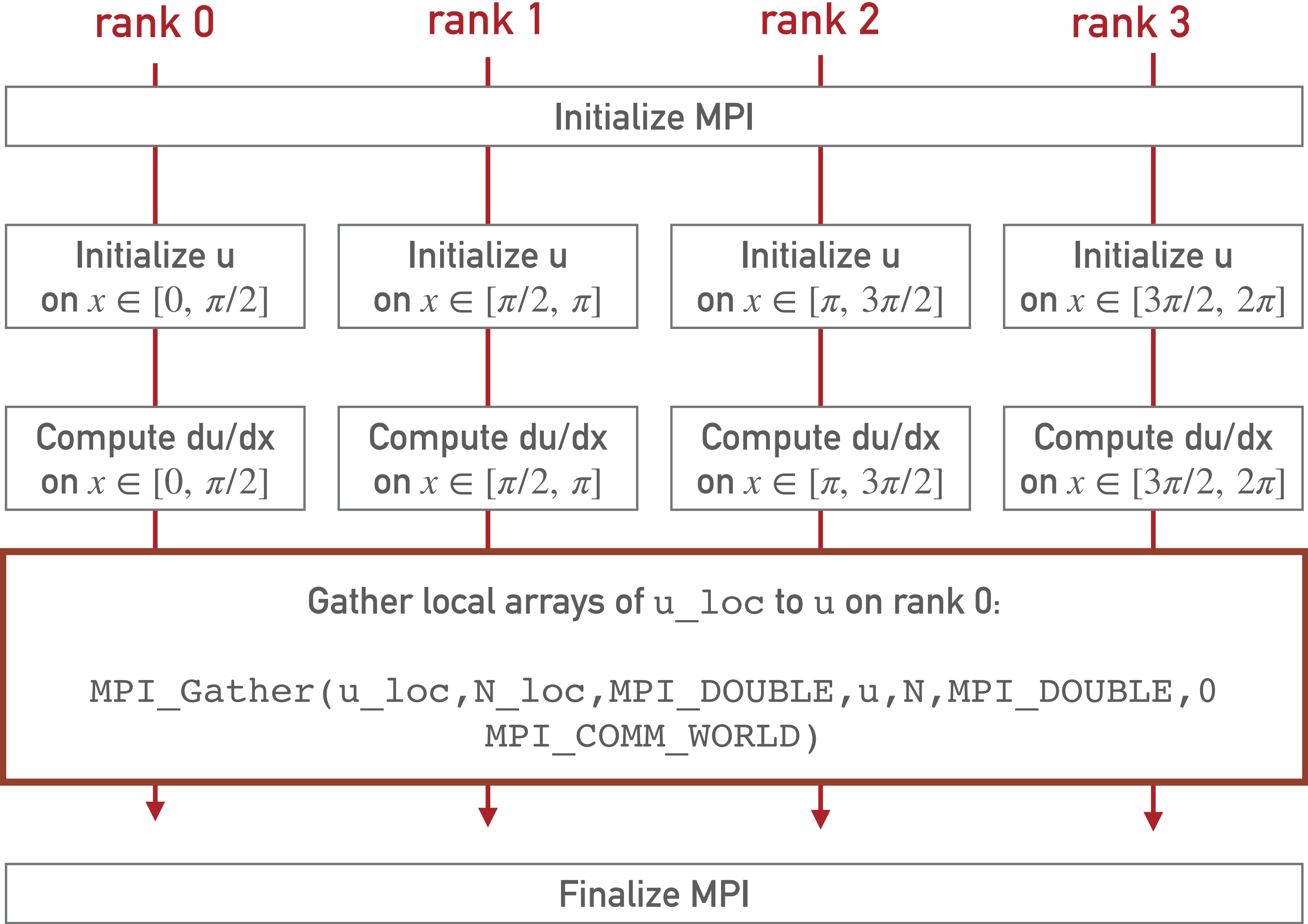


Gather

DERIVATIVE EXAMPLE



DERIVATIVE EXAMPLE



EXAMPLE

*Imagine you need to compute a standard deviation of a large set of numbers saved in a file.
How would you design a parallel algorithm using Reduce, Broadcast, Scatter, Gather, ...
functionalities?*

$$\sigma = \sqrt{\frac{1}{N} \sum_{i=0}^{N-1} (x_i - \mu)^2} \quad \text{where} \quad \mu = \frac{1}{N} \sum_{i=0}^{N-1} x_i$$

