Parallel Programming

Prof. **Paolo Bientinesi**

pauldj@aices.rwth-aachen.de

WS 16/17

Collective Communication

Barrier

Broadcast $\leftrightarrow$ Reduce

Scatter $\leftrightarrow$ Gather

Allgather $\leftrightarrow$ Reduce-scatter

Allreduce

Alltoall

$\vdots$

`int MPI_BCast(...)`

Before:

Node_0	Node_1	Node_2	Node_3
		α	

After:

Node_0	Node_1	Node_2	Node_3
α	α	α	α

`int MPI_BCast(...)`

Before:

Node_0	Node_1	Node_2	Node_3
		α	

After:

Node_0	Node_1	Node_2	Node_3
α	α	α	α

```
MPI_Bcast( Buffer, count, type, root, communicator );
```

`int MPI_Reduce(...)`

Before:

Node_0	Node_1	Node_2	Node_3
δ_0	δ_1	δ_2	δ_3

After:

Node_0	Node_1	Node_2	Node_3
		$\text{Op}_{i=0}^{p-1} \delta_i$	

`int MPI_Reduce(...)`

Before:

Node_0	Node_1	Node_2	Node_3
δ_0	δ_1	δ_2	δ_3

After:

Node_0	Node_1	Node_2	Node_3
		$\text{Op}_{i=0}^{p-1} \delta_i$	

```
MPI_Reduce( sendBuffer, recvBuffer, count, type,
            operation, root, communicator );
```

`int MPI_Reduce(...)`

Before:

Node_0	Node_1	Node_2	Node_3
δ_0	δ_1	δ_2	δ_3

After:

Node_0	Node_1	Node_2	Node_3
		$\text{Op}_{i=0}^{p-1} \delta_i$	

```
MPI_Reduce( sendBuffer, recvBuffer, count, type,
            operation, root, communicator );
```

`MPI_Op_create`

`int MPI_Scatter(...)`

Before:

$Node_0$	$Node_1$	$Node_2$	$Node_3$
		v[0] v[1] v[2] v[3]	

After:

$Node_0$	$Node_1$	$Node_2$	$Node_3$
v[0]			
	v[1]		
		v[2]	
			v[3]

`int MPI_Scatter(...)`

Before:

$Node_0$	$Node_1$	$Node_2$	$Node_3$
		v[0] v[1] v[2] v[3]	

After:

$Node_0$	$Node_1$	$Node_2$	$Node_3$
v[0]			
	v[1]		
		v[2]	
			v[3]

```
MPI_Scatter( sendBuffer, sendCount, sendType,
             recvBuffer, recvCount, recvType,
             root, communicator );
```

`int MPI_Gather(...)`

Before:

$Node_0$	$Node_1$	$Node_2$	$Node_3$
v[0]	v[1]	v[2]	v[3]

After:

$Node_0$	$Node_1$	$Node_2$	$Node_3$
		v[0] v[1] v[2] v[3]	

int MPI_Gather(...)

Before:

$Node_0$	$Node_1$	$Node_2$	$Node_3$
v[0]	v[1]	v[2]	v[3]

After:

$Node_0$	$Node_1$	$Node_2$	$Node_3$
		v[0] v[1] v[2] v[3]	

```
MPI_Gather( sendBuffer, sendCount, sendType,
            recvBuffer, recvCount, recvType,
            root, communicator );
```

`int MPI_Allgather(...)`

Before:

$Node_0$	$Node_1$	$Node_2$	$Node_3$
v[0]			
	v[1]		
		v[2]	
			v[3]

After:

$Node_0$	$Node_1$	$Node_2$	$Node_3$
v[0]	v[0]	v[0]	v[0]
v[1]	v[1]	v[1]	v[1]
v[2]	v[2]	v[2]	v[2]
v[3]	v[3]	v[3]	v[3]

`int MPI_Allgather(...)`

Before:

$Node_0$	$Node_1$	$Node_2$	$Node_3$
v[0]	v[1]	v[2]	v[3]

After:

$Node_0$	$Node_1$	$Node_2$	$Node_3$
v[0]	v[0]	v[0]	v[0]
v[1]	v[1]	v[1]	v[1]
v[2]	v[2]	v[2]	v[2]
v[3]	v[3]	v[3]	v[3]

```
MPI_Allgather( sendBuffer, sendCount, sendType,
               recvBuffer, recvCount, recvType,
               communicator );
```

`int MPI_Reduce_scatter(...)`

Before:

$Node_0$	$Node_1$	$Node_2$	$Node_3$
$v_0[0]$	$v_1[0]$	$v_2[0]$	$v_3[0]$
$v_0[1]$	$v_1[1]$	$v_2[1]$	$v_3[1]$
$v_0[2]$	$v_1[2]$	$v_2[2]$	$v_3[2]$
$v_0[3]$	$v_1[3]$	$v_2[3]$	$v_3[3]$

After:

$Node_0$	$Node_1$	$Node_2$	$Node_3$
$\underset{i}{\text{Op}}\ v_i[0]$	$\underset{i}{\text{Op}}\ v_i[1]$	$\underset{i}{\text{Op}}\ v_i[2]$	$\underset{i}{\text{Op}}\ v_i[3]$

int MPI_Reduce_scatter(...)

Before:

$Node_0$	$Node_1$	$Node_2$	$Node_3$
$v_0[0]$	$v_1[0]$	$v_2[0]$	$v_3[0]$
$v_0[1]$	$v_1[1]$	$v_2[1]$	$v_3[1]$
$v_0[2]$	$v_1[2]$	$v_2[2]$	$v_3[2]$
$v_0[3]$	$v_1[3]$	$v_2[3]$	$v_3[3]$

After:

$Node_0$	$Node_1$	$Node_2$	$Node_3$
$\underset{i}{\text{Op}}\ v_i[0]$	$\underset{i}{\text{Op}}\ v_i[1]$	$\underset{i}{\text{Op}}\ v_i[2]$	$\underset{i}{\text{Op}}\ v_i[3]$

```
MPI_Reduce_scatter( sendBuffer,   recvBuffer,
                    recvCounts[], type, operation,
                    communicator );
```

`int MPI_Allreduce(...)`

Before:

Node_0	Node_1	Node_2	Node_3
δ_0	δ_1	δ_2	δ_3

After:

Node_0	Node_1	Node_2	Node_3
$\text{Op}_{i=0}^{p-1}\ \delta_i$	$\text{Op}_{i=0}^{p-1}\ \delta_i$	$\text{Op}_{i=0}^{p-1}\ \delta_i$	$\text{Op}_{i=0}^{p-1}\ \delta_i$

`int MPI_Allreduce(...)`

Before:

Node_0	Node_1	Node_2	Node_3
δ_0	δ_1	δ_2	δ_3

After:

Node_0	Node_1	Node_2	Node_3
$\text{Op}_{i=0}^{p-1} \delta_i$	$\text{Op}_{i=0}^{p-1} \delta_i$	$\text{Op}_{i=0}^{p-1} \delta_i$	$\text{Op}_{i=0}^{p-1} \delta_i$

```
MPI_Allreduce( sendBuffer,  recvBuffer,
               count, type, operation,
               communicator );
```

`int MPI_Alltoall(...)`

Before:

$Node_0$	$Node_1$	$Node_2$	$Node_3$
$v_0[0]$	$v_1[0]$	$v_2[0]$	$v_3[0]$
$v_0[1]$	$v_1[1]$	$v_2[1]$	$v_3[1]$
$v_0[2]$	$v_1[2]$	$v_2[2]$	$v_3[2]$
$v_0[3]$	$v_1[3]$	$v_2[3]$	$v_3[3]$

After:

$Node_0$	$Node_1$	$Node_2$	$Node_3$
$v_0[0]$	$v_0[1]$	$v_0[2]$	$v_0[3]$
$v_1[0]$	$v_1[1]$	$v_1[2]$	$v_1[3]$
$v_2[0]$	$v_2[1]$	$v_2[2]$	$v_2[3]$
$v_3[0]$	$v_3[1]$	$v_3[2]$	$v_3[3]$

`int MPI_Alltoall(...)`

Before:

$Node_0$	$Node_1$	$Node_2$	$Node_3$
$v_0[0]$	$v_1[0]$	$v_2[0]$	$v_3[0]$
$v_0[1]$	$v_1[1]$	$v_2[1]$	$v_3[1]$
$v_0[2]$	$v_1[2]$	$v_2[2]$	$v_3[2]$
$v_0[3]$	$v_1[3]$	$v_2[3]$	$v_3[3]$

After:

$Node_0$	$Node_1$	$Node_2$	$Node_3$
$v_0[0]$	$v_0[1]$	$v_0[2]$	$v_0[3]$
$v_1[0]$	$v_1[1]$	$v_1[2]$	$v_1[3]$
$v_2[0]$	$v_2[1]$	$v_2[2]$	$v_2[3]$
$v_3[0]$	$v_3[1]$	$v_3[2]$	$v_3[3]$

```
MPI_Alltoall( sendBuffer, sendCount, sendType,
              recvBuffer, recvCount, recvType,
              communicator );
```

More Collectives

Variable length

- `MPI_Scatterv`
- `MPI_Gatherv`
- `MPI_Allgatherv`
- `MPI_Alltoallv`

More Collectives

Variable length

- `MPI_Scatterv`
- `MPI_Gatherv`
- `MPI_Allgatherv`
- `MPI_Alltoallv`

```
MPI_Scatter(
    sendBuffer, sendCount,
                sendType,
    recvBuffer, recvCount,
                recvType,
    root, communicator );
```

```
MPI_Scatterv(
    sendBuffer, sendCounts[],
    displs[],   sendType,
    recvBuffer, recvCount,
                recvType,
    root, communicator )
```

Even more Collectives

Partial reduction

- `MPI_Scan`

Non-blocking collectives

- `MPI_I*`