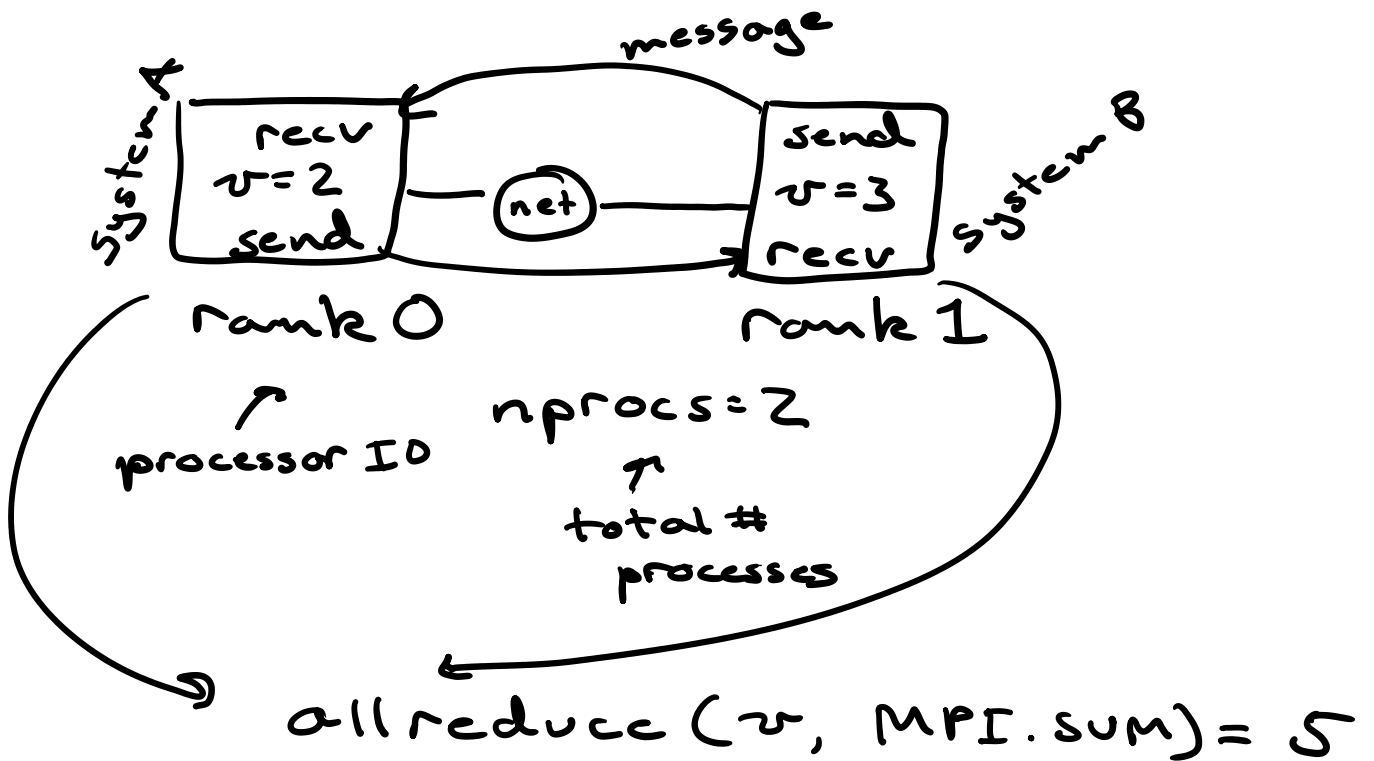


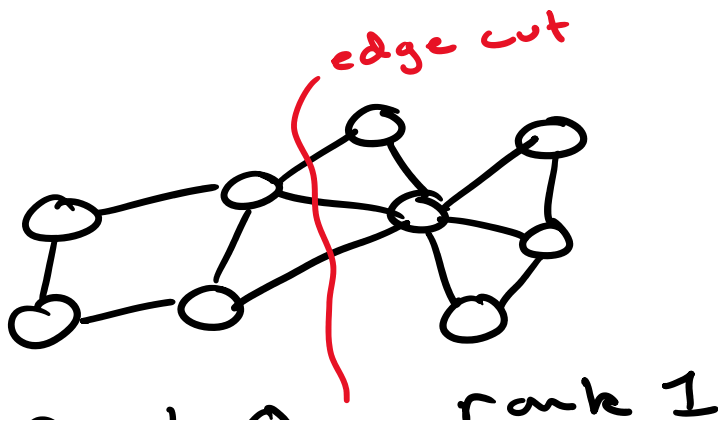
Recall: MPI

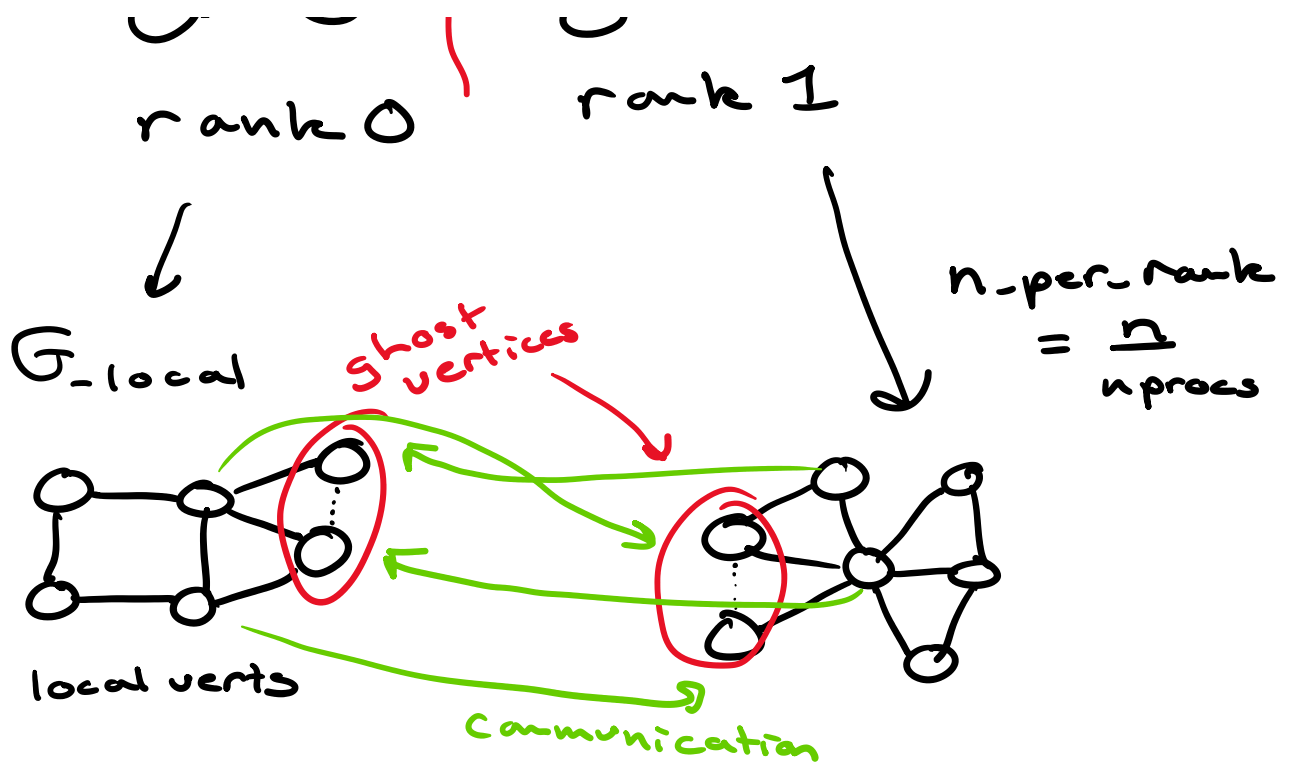
(message passing)

for distributed processing



Distributed Graph G





→ We compute updates on local vertices and exchange messages via ghost vertices

Baseline BFS

For all $v \in V(G)$

level $[v] = -1$

$Q \leftarrow \text{root}$

cur_level = 0

level [root] = 0

While $Q \neq \emptyset$

← we'll be using a local graph

← need to know for ghost vertices if we've already discovered

← needs to consider a global queue

While $Q \neq \emptyset$ ← needs to consider a global queue

cur_level += 1

For all $v \in Q$:

For all $u \in N(v)$

if level[u] == -1

level[u] = cur_level

$Q_{next} \leftarrow u$

$Q = Q_{next}$

$Q_{next} = \emptyset$

↑
we need to consider ghosts

↑
we need to exchange discovered ghost vertices

Distributed BFS

For all $v \in V(G_x)$ ← local graph

level[v] = -2

if root $\in V_{local}$

$Q \leftarrow root$

cur_level = 0

allreduce(len(Q), global_sum, MPI.SUM) ← send value ↓ reduced value

allreduce(len(Q), global_sum, MPI_SUM)

while global_sum > 0

For v in Q

level[v] = cur_level

For u in N(v)

if level[u] == -2

level[u] = -1

Q_{next} ← u

Q = ∅

For i in (0, nprocs)

For j in (0, nprocs)

if i == procid

sendbuf = [v in Q_{next} |
where rank(v)
equals j]

owner
rank of v

if i == procid & j == procid

Q ← sendbuf

else if i == procid

MPI.send(sendbuf, j)

else if j == procid

else if $j == \text{procid}$ ~
MPI.recv(Q, i)

allreduce(len(Q), global_sum,
Qnext = \emptyset MPI.SUM)