Argonne NATIONAL LABORATORY

Non-Collective Communicator Creation

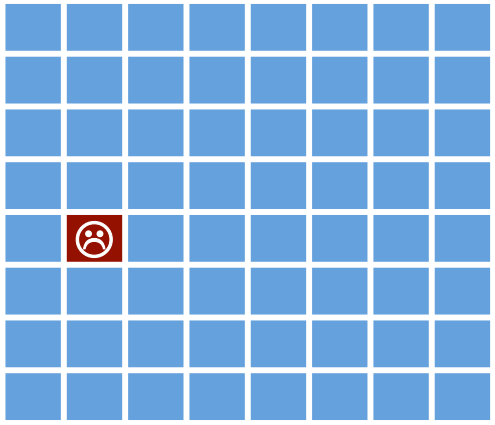
Ticket #286

int MPI_Comm_create_group(MPI_Comm comm,
MPI_Group group, int tag, MPI_Comm *newcomm)

Non-Collective Communicator Creation in MPI. Dinan, et al., Euro MPI '11.

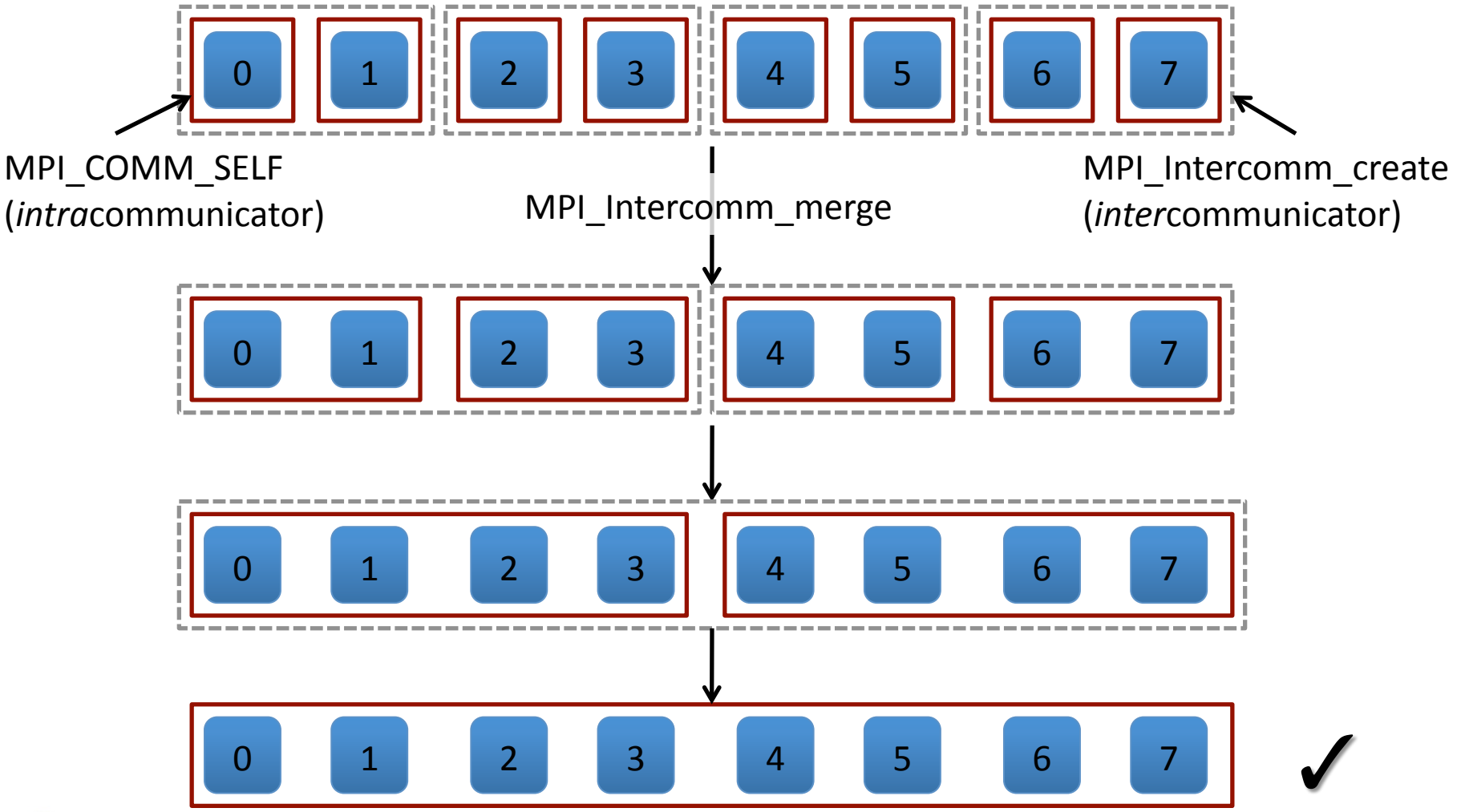
Non-Collective Communicator Creation

- Currently: Collective operation
- Non-Collective: Create communicator collectively only on new members



1. Avoid bulk synchronization
   - Load balancing: Reassign processes from idle groups to active groups
2. Overhead reduction
   - Multi-level parallelism, small communicators
3. Recovery from failures
   - Not all ranks in parent can participate
4. Compatibility with Global Arrays
   - Past: collectives using MPI Send/Recv

Non-Collective Communicator Creation Algorithm



Non-Collective Algorithm in Detail

INPUT: $group$, $comm$, tag
OUTPUT: $comm'$
REQUIRE: $group$ is ordered by desired rank in $comm'$ and is identical on all callers
LET: $grp\_pids[0..|group|-1] = \mathbb{N}$ and $pids[\,]$ be arrays of length $|group|$

MPI_Comm_rank($comm$, &$rank$)
MPI_Group_rank($group$, &grp_rank), MPI_Group_size($group$, &grp_size)
MPI_Comm_dup(MPI_COMM_SELF, &$comm'$)

MPI_Comm_group($comm$, &$parent\_grp$)
MPI_Group_translate_ranks($group$, grp_size, grp_pids, $parent\_grp$, $pids$)
MPI_Group_free(&$parent\_grp$)

Translate group ranks to ordered list of ranks on parent communicator

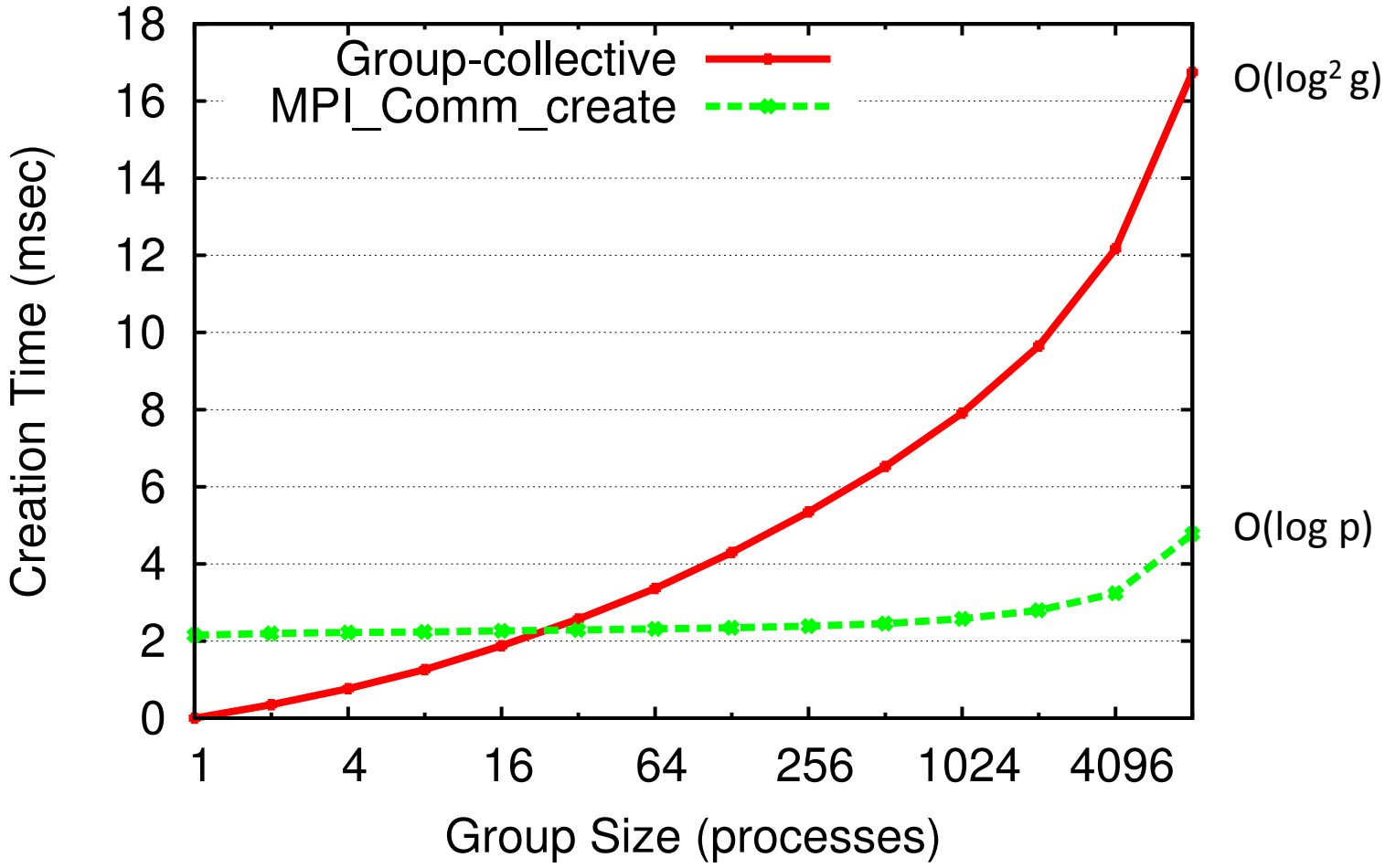
for ($merge\_sz \leftarrow 1$; $merge\_sz < grp\_size$; $merge\_sz \leftarrow merge\_sz \cdot 2$) **do**
  $gid \leftarrow grp\_rank / merge\_sz$, $comm\_old \leftarrow comm'$
  if $gid \mod 2 = 0$ **then**
    if $((gid+1) \cdot merge\_sz < grp\_size$ **then**
      MPI_Intercomm_create($comm'$, 0, $comm$, $pids[(gid+1) \cdot merge\_sz]$, tag, &ic)
      MPI_Intercomm_merge(ic, 0 /* LOW */, &$comm'$)
    end if
  else
    MPI_Intercomm_create($comm'$, 0, $comm$, $pids[(gid-1) \cdot merge\_sz]$, tag, &ic)
    MPI_Intercomm_merge(ic, 1 /* HIGH */, &$comm'$)
  end if
  if $comm' \neq comm\_old$ **then**
    MPI_Comm_free(&ic)
    MPI_Comm_free(&$comm\_old$)
  end if
end for

Calculate my group ID

Left group

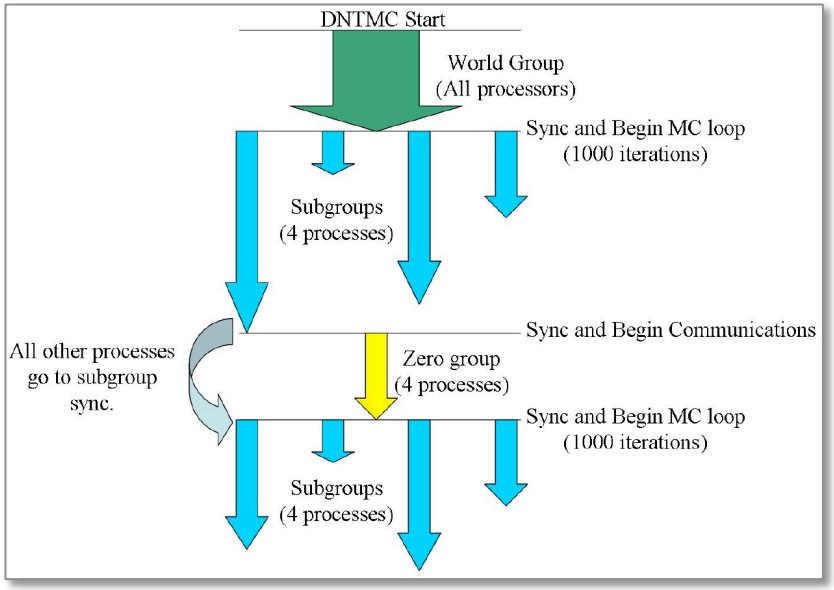
Right group

Evaluation: Microbenchmark



Case Study: Markov Chain Monte Carlo

- Dynamical nucleation theory Monte Carlo (DNTMC)
  - Part of NWChem
  - Markov chain Monte Carlo
- Multiple levels of parallelism
  - Multi-node "Walker" groups
  - Walker: *N* random samples



T L Windus *et al* 2008 *J. Phys.: Conf. Ser.* **125** 012017

- Load imbalance
  - Sample computation (energy calculation) is irregular, can be rejected
- Regroup idle processes into active group
  - Preliminary results: 30% decrease in total application execution time

- Santorini straw vote to proceed with a formal reading:

  Yes (16), No (0), Abstain (1)