ASYMMETRIC MEMORY EXTENSION FOR OPENSHMEM

LATCHESAR IONKOV
GINGER YOUNG
LOS ALAMOS NATIONAL LABORATORY
SYMMETRIC MEMORY

- Shmalloc is global function call.
- Same amount of memory on all PEs.
- Local pointer (plus pointer arithmetic) used to identify remote symmetric region.
- Requires in-step execution.
- Doesn’t work well with heterogeneous, asynchronous code.
ASYMMETRIC MEMORY DESIGN

• Alternatives
  • Register vs. allocate memory.
  • Remote pointer (plus arithmetic) vs. remote handle + offset.

• API
  
  `asym_init()`
  `asym_register(ptr, size)`
  `asym_unregister(ptr)`
  `asym_put(tgt, src, size, pe)`
  `asym_get(tgt, src, size, pe)`
ASYMMETRIC MEMORY IMPLEMENTATION

• UCCS implementation only.

• Asymmetric entries table (in symmetric memory).

• Register/Unregister doesn’t require any communication.

• Remote tables cache (updated on-demand).

• Binary search in the table for each put/get.

• UCCS Limitations:
  • number of registered regions
  • memory size
ASYMMETRIC MEMORY PERFORMANCE

The graph shows the time (in ms) taken for various operations as the number of areas increases. The operations include Asym Put, Asym Get, Sym Put, and Sym Get. The x-axis represents the number of areas, while the y-axis represents the time taken.
ASYMMETRIC MEMORY
FUTURE WORK

- Improve performance
- Support for more Openshmem operations
- Combine symmetric and asymmetric implementations?