Core-idling on MPI Intra-node Communication Channels for Energy Efficiency

Keon-Woo Kim
Konkuk University, Seoul, Korea
tmffka7@konkuk.ac.kr

Hyun-Wook Jin
Konkuk University, Seoul, Korea
jinh@konkuk.ac.kr

Eun-Kyu Byun
KISTI, Daejeon, Korea
ekbyun@kisti.re.kr

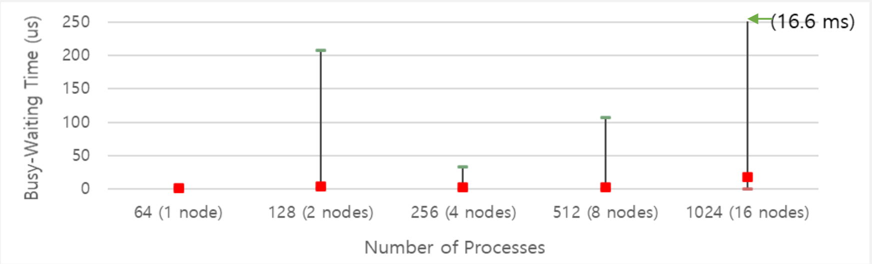
SC21 St. Louis, MO | science & beyond

System Software Laboratory — KU KONKUK UNIVERSITY — KISTI Korea Institute of Science and Technology Information www.kisti.re.kr

Introduction and Motivation

Busy-Waiting and Energy Efficiency

- Parallel programming libraries are optimized on the assumption that a parallel process occupies an entire CPU core
- Existing MPI libraries perform busy-waiting to check the completion of outstanding communications
- The longer the busy-waiting time, the higher the energy consumption



Studies on Energy-Efficient MPI

- Dynamic Voltage and Frequency Scaling (DVFS)
  - It provides different levels of voltage and frequency for operation processors (e.g., P-states)
  - Lowering voltage and frequency for busy-waiting time can enhance energy efficiency
- Core-idling
  - It turns off hardware components of idle cores (e.g., C-States)
  - Suspending a process instead of busy-waiting can enhance energy efficiency
- Prediction schemes
  - Both DVFS and core-idling cause additional latency of using energy-saving modes
  - Researchers have focused on prediction schemes that determine when and which energy-saving mode to enter

Motivation

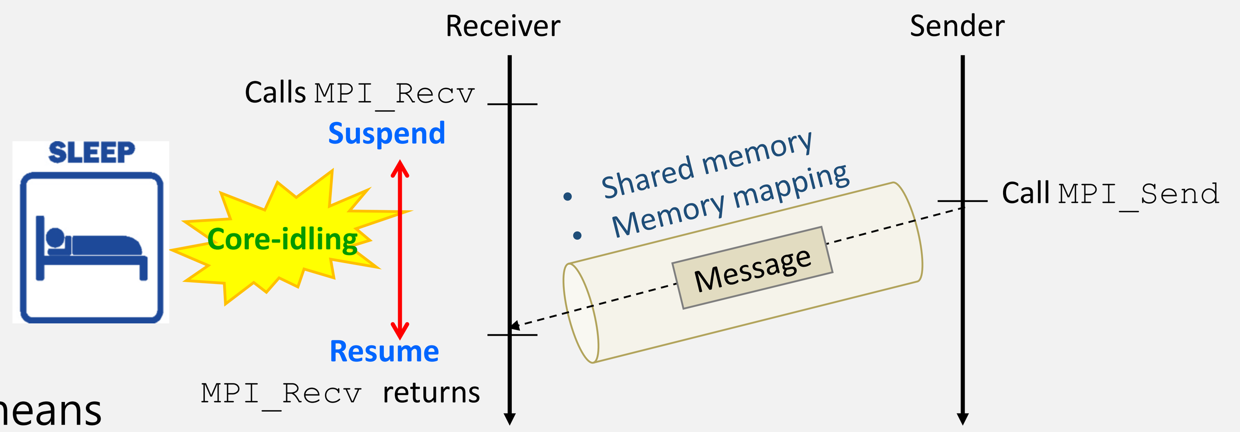
- If the prediction can be done inside of the MPI library, it can gain much insight by utilizing internal information concealed beneath application programming interfaces

A redesign of the MPI library for high energy efficiency over multiple communication channels has been paid less attention

Energy-Efficient MPI Intra-node Communication

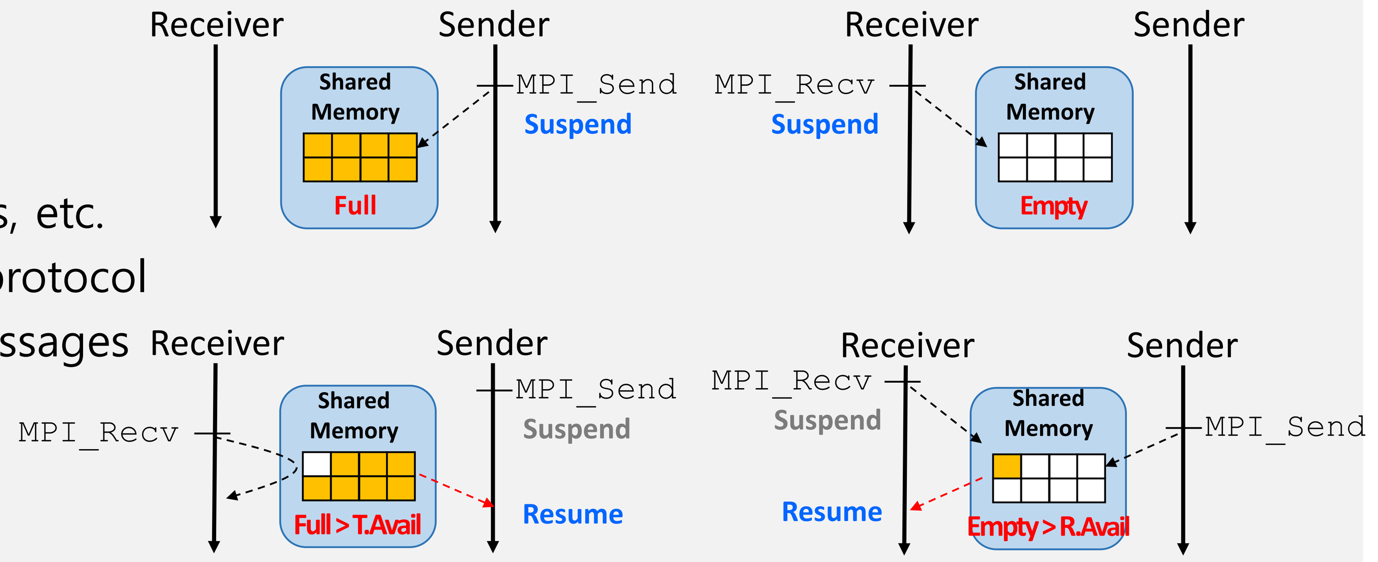
MPI Communication Channels

- Inter-node communication channels
  - InfiniBand, Omni-Path, Ethernet, etc.
- Intra-node communication channels
  - Shared memory: moves messages from source to destination via a shared memory region and is used for small messages (Eager protocol)
  - Memory mapping: directly moves messages from source to destination by means of a kernel-level support and is used for large messages (Rendezvous protocol)



Signaling-based Core-idling

- Identification of suspending and resuming points in MPI library
  - Suspending in shared memory: no available shared buffer, no messages, etc.
  - Suspending in memory mapping: no control messages of rendezvous protocol
  - Resuming in shared memory: shared buffer becomes available, new messages
  - Resuming in memory mapping: new (control) messages
- Implementation methodologies
  - Assembly instructions (e.g., mwait): CPU dependent (X)
  - Timers: only for coarse-grained controls (X)
  - Semaphores: deadlock-prone (X)
  - Signals (O)



Performance Evaluation

- Experiment system
  - Intel Xeon Ivy Bridge 2.8 GHz processors (Deca-core x 2)
  - NAS Parallel Benchmarks (MG, CG, IS, and FT) Class C
  - Package-level energy consumption measured by RAPL
- Measurement results
  - Undersubscribed case (Fig. 1(a)): core-idling could save energy consumption up to 12% with a marginal (less than 2%) impact on execution time
  - Oversubscribed case (Fig. 1(b)): both energy consumption and execution time were improved up to 91% in CG
  - CPU idle time increased significantly with core-idling (Fig. 2), which was the main reason for observing a higher energy saving

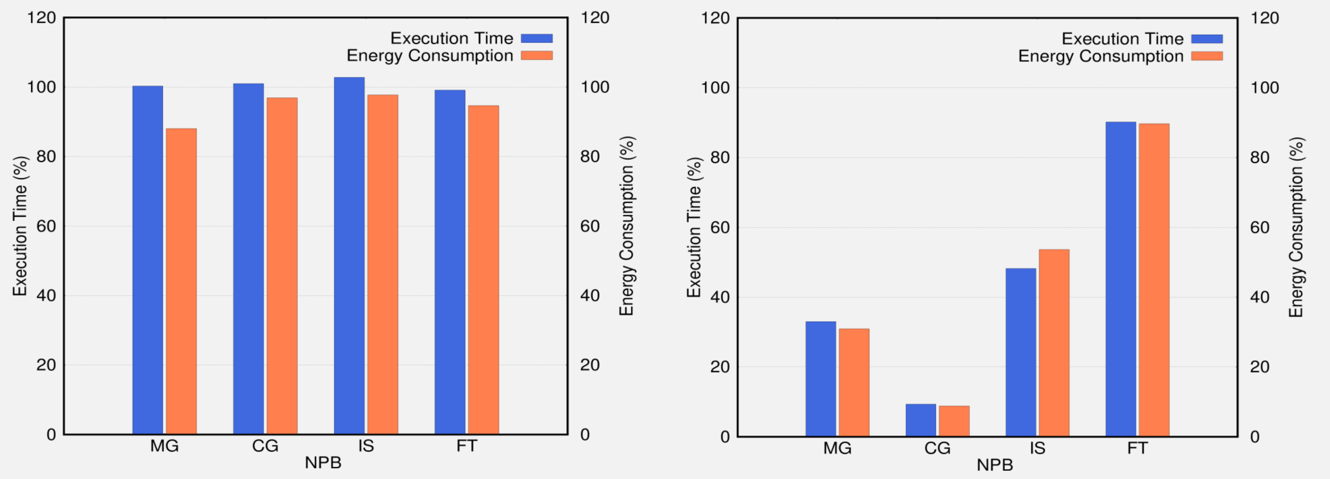


Fig1. Comparisons of execution time and energy consumption: (a) 16 processes (undersubscribed case); (b) 32 processes (oversubscribed case)

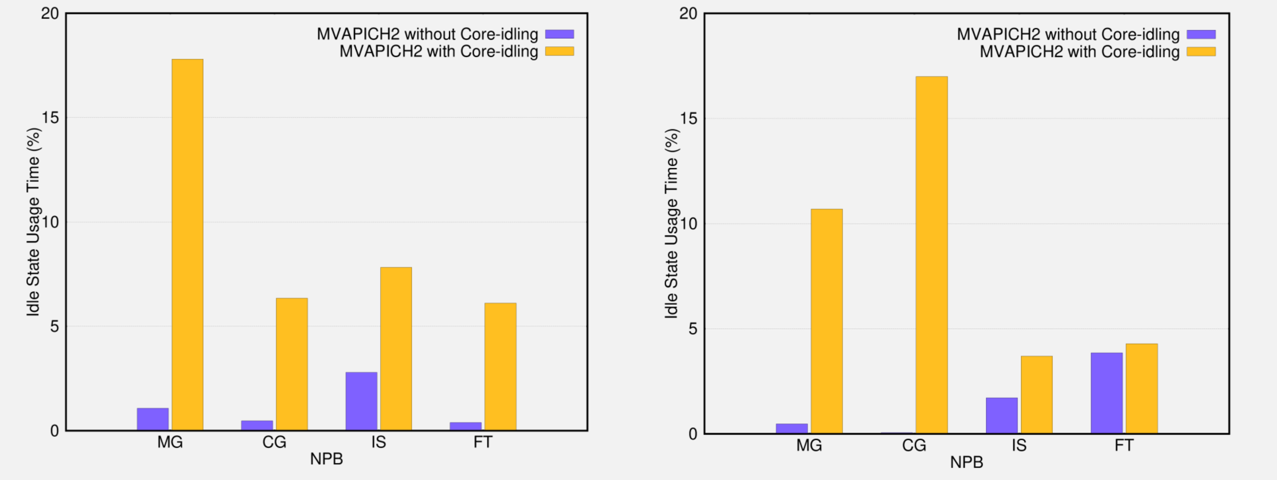


Fig2. Average time spent in the deepest energy-saving mode: (a) 16 processes (undersubscribed case); (b) 32 processes (oversubscribed case)

As a first step, we focus on the MPI intra-node communications that utilize a shared memory channel for small messages and a memory mapping channel for large messages

Our signaling-based core-idling could reduce the energy consumption of NPB up to 12% and 91% with 16 process and 32 process, respectively, on a NUMA-based system

Conclusions and Future Work

Conclusions

- We suggested signaling-based core-idling for energy efficient MPI over multiple communication channels
  - Identified when a process should be suspended and resumed on the shared memory and memory mapping channels for intra-node communications
- Our preliminary implementation in MVAPICH2 could reduce the energy consumption of NPB MG by 12% for undersubscribed case and that of NPB CG by 91% for oversubscribed case

Future Work

- We plan to measure the energy efficiency of signaling-based core-idling on various CPU architectures
- We will expand our implementation to support the inter-node communication channel

This research was supported by the Supercomputer Development Leading Program of the National Research Foundation of Korea (NRF) funded by the Korean government (Ministry of Science and ICT (MSIT)) (No. 2020M3H6A1084853)

Signaling-based core-idling will be flexible enough to support the inter-node communication channel and able to leverage existing prediction schemes used in DVFS and core-idling approaches

http://sslab.konkuk.ac.kr