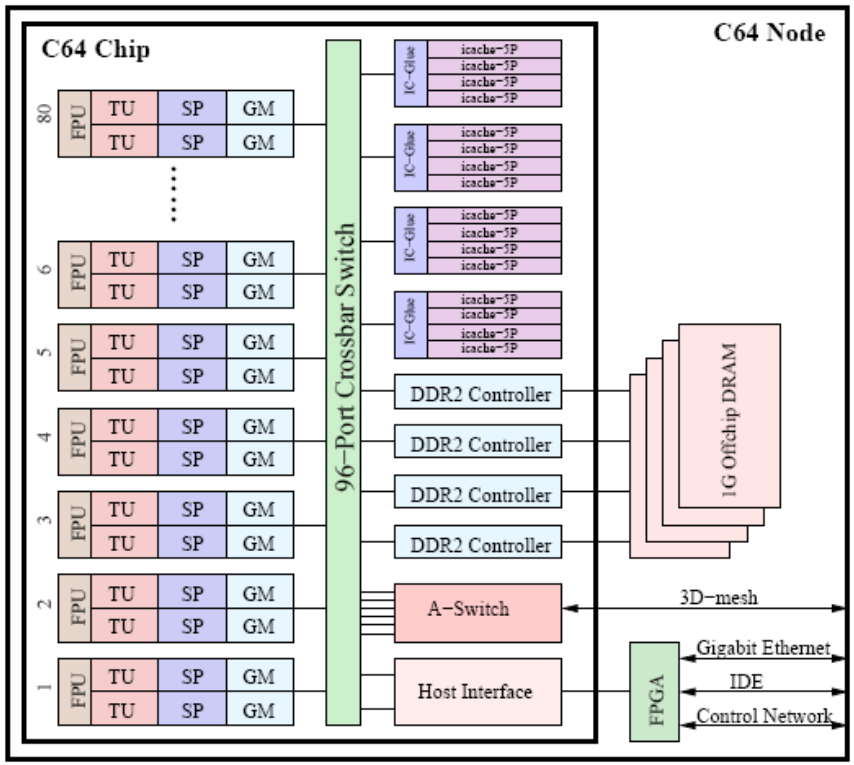
A Multithreaded Implementation of a Network Communication Protocol on IBM Cyclops-64 Multithreaded Architecture

Gan, Ziang Hu, Juan Cuvillo, Guang R. Gao
University of Delaware
{gan,hu,jcuvillo,ggao@capsl.udel.edu}
Dept. of Electrical and Computer Engineering

Outline

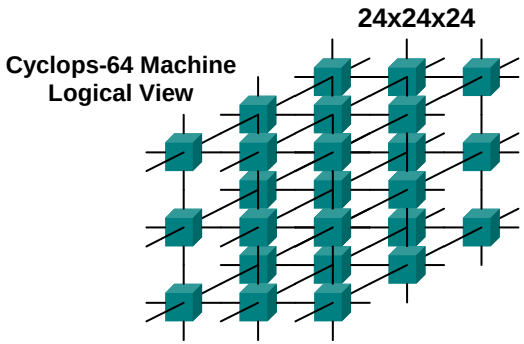
- Introduction to IBM Cyclops-64 multi-core processor;
- Problem formulation;
- Cyclops Datagram Protocol (CDP) Design;
- A multithreaded implementation;
- Performance evaluation;

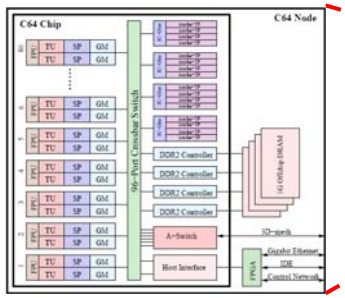
Cyclops-64 Multi-core Processor



- 80 single issue RISC cores per chip, clocked at 500MHz.
- 2 thread units + 1 floating point unit per core;
- On-chip SRAM (2x32KB) instead of data cache;
- 32KB i-cache shared among 5 cores;
- 96-port on-chip crossbar switch;
- A-switch enable inter-node comm.;
- Software controlled explicit memory hierarchy;
- Thread is non-preemptive;
- No memory virtualization;

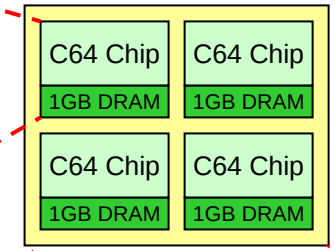
Cyclops-64 System Overview

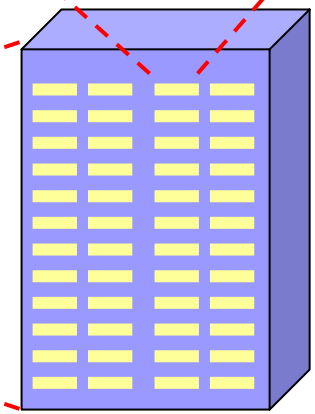




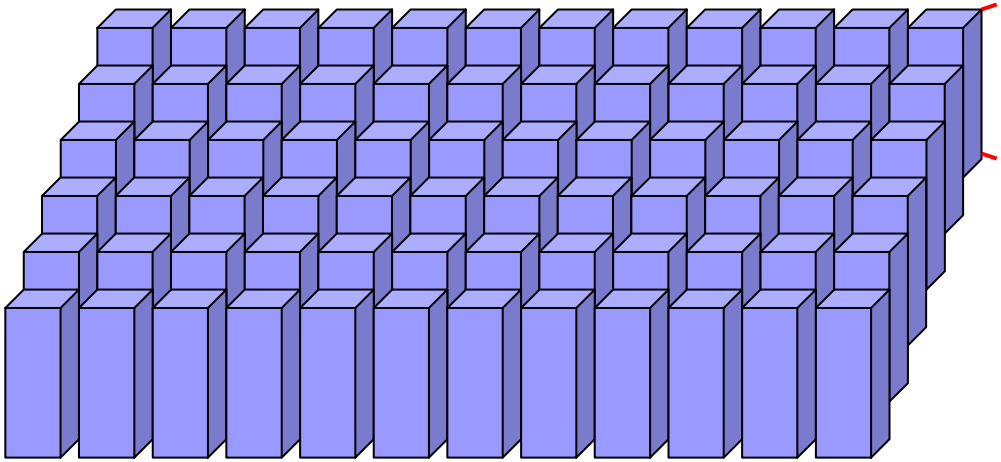
1 Cyclops-64 Chip

1 Board
(4 Cyclops-64 Chips)





1 Rack (48 Boards)

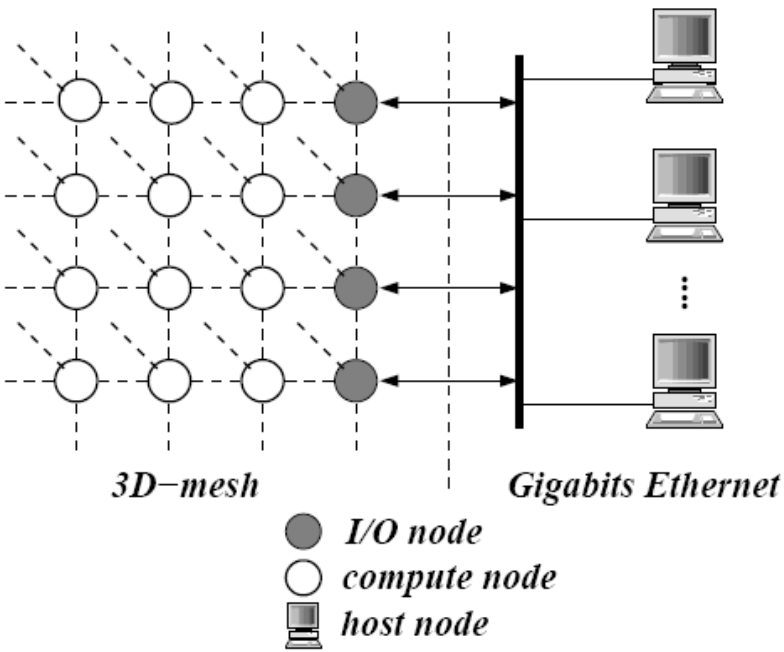


1 Cyclops-64 Machine (72 Racks)

- **Chip Number**
  4x48x72 = 13824
- **Peak Perf.**
  13824x80x2x500M = 1.106 PetaFLOPS
- **2006 Top1 (BlueGene/L) Perk Perf.**
  0.367 PetaFLOPS

Thanks Fei Chen for the figures

Network Topology

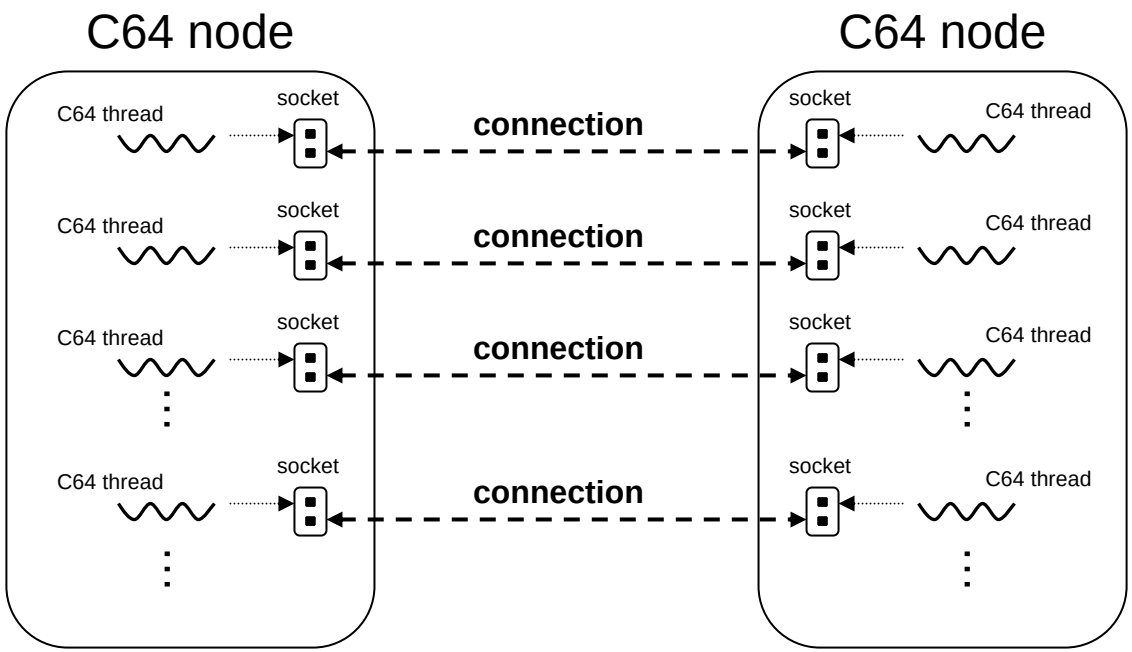


- 3D-mesh network on backend;
- 3D-mesh is error-free;
- Gigabit Ethernet on front end;
- C64 I/O nodes bridge host machine to compute engine;
- Host machines are responsible for system admin, job scheduling, file system, software development environment;

Problem Formulation

- Is it possible to design and implement a network communication protocol in a way such that it can utilize the massive thread-level parallelism effectively and therefore achieve good performance scalability?
- Is the communication protocol developed for C64 processor an efficient one compared with other network protocols?

Schema of Solution



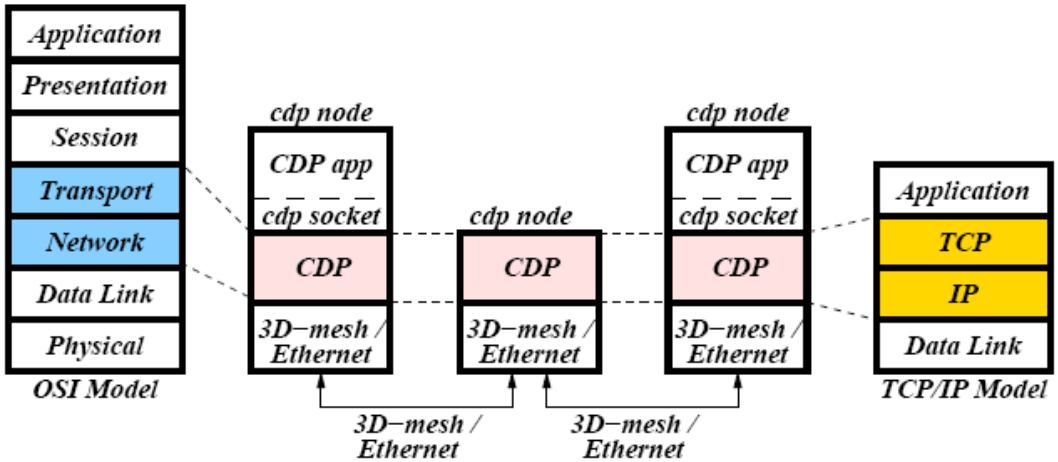
Protocol Design Goals

- No lost data gram;
- In order;
- No duplicate;
- Error free;
- Efficient;
- Scalable;

Cyclops Datagram Protocol (CDP)

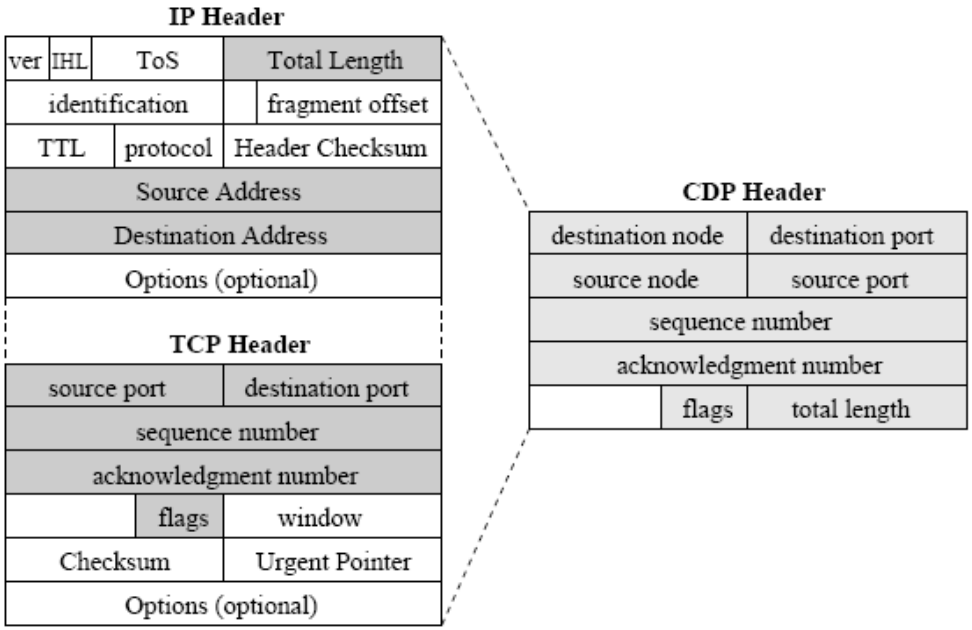
- Datagram-based, connection-oriented;
- Reliable and supports timeout retransmission;
- Sliding-window based flow control mechanism to avoid traffic congestion;
- Full-duplex transmission services;
- Standard BSD socket programming interfaces;

CDP Protocol Stack



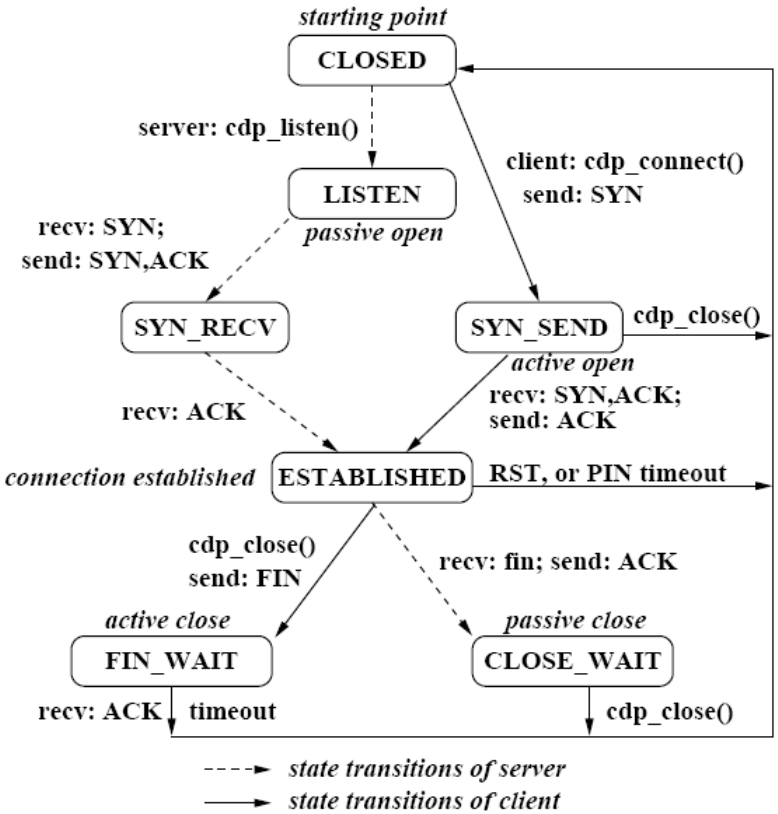
- Equal to Transport layer + Network Layer;
- Need a mechanism for addressing;
- Currently support two different sub-networks;
- Require a simple routing mechanism;
- The underlying sub-networks are robust;

CDP Packet Header Format



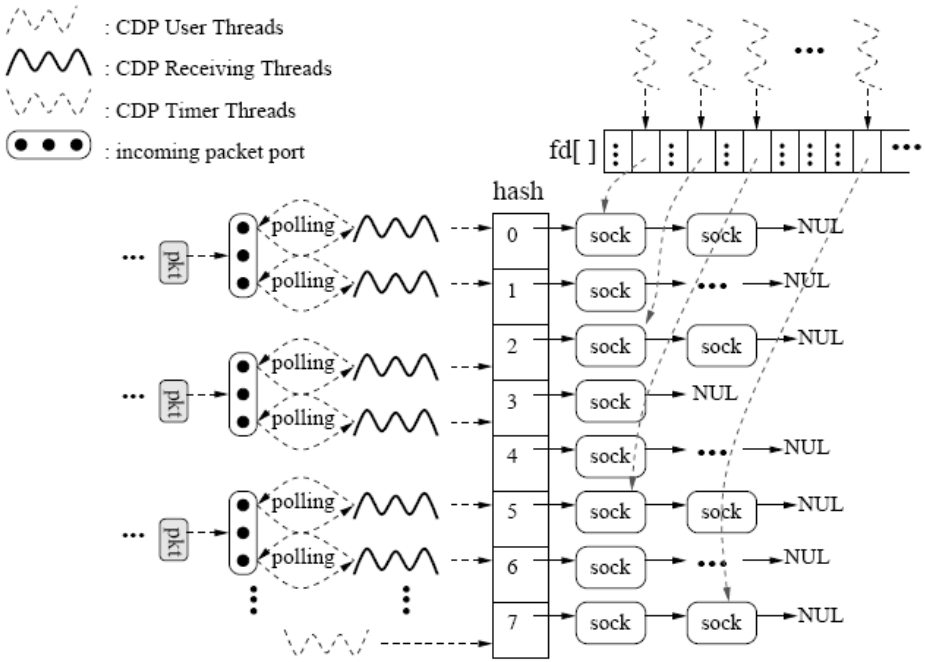
- Merging TCP header into IP header;
- Similar addressing mechanism;
- Similar control bits in flags fields;
- Sequence number identifies a datagram;
- Don’t support Fragment and De-fragment;
- No Checksum;

CDP State Transition Diagram



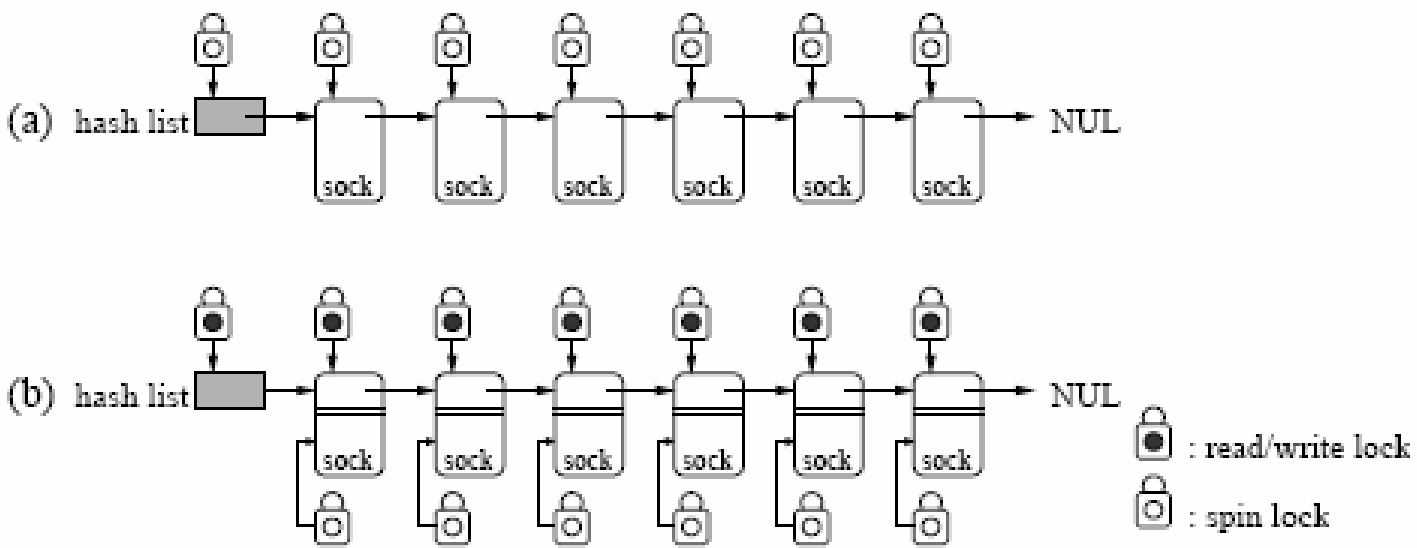
- 3-way handshake connection establishment protocol;
- 2-way handshake connection closing protocol. Do not support “half-close” connection;
- Client-Server model;
- Ping-pong packet to differentiate “silent” client and “dead” client;

Implementation Internal



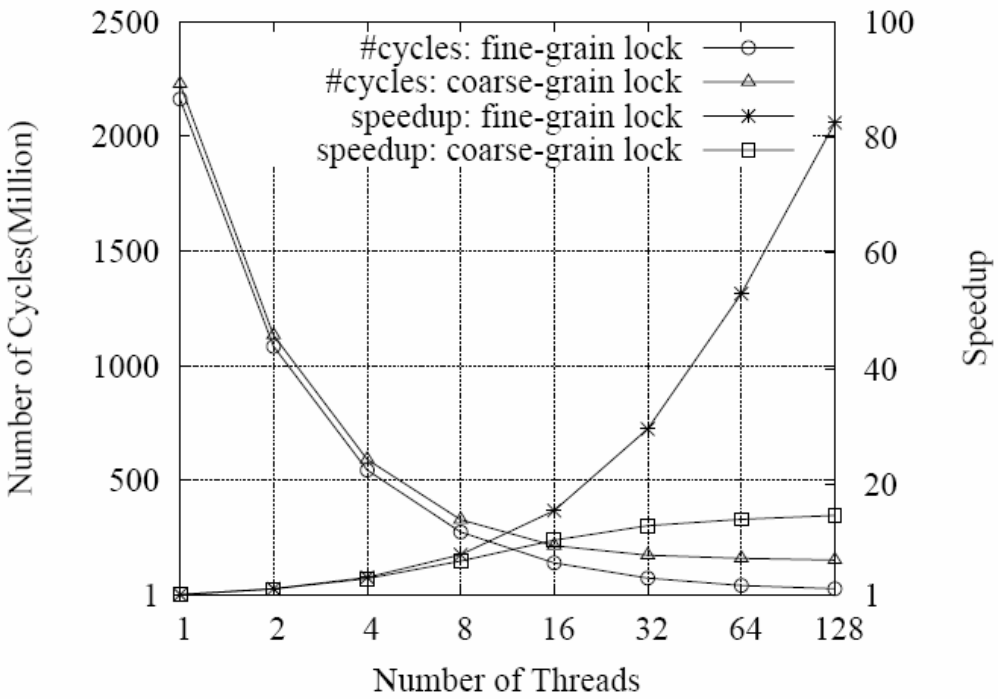
- CDP socket represents a connection;
- CDP user threads implement the semantics of CDP programming interfaces;
- CDP receiving threads poll on incoming packet port;
- CDP receiving threads can process multiple connections at the same time;
- CDP timer thread is responsible for timeout retransmission;

Coarse-Grain Lock vs Fine-Grain Lock



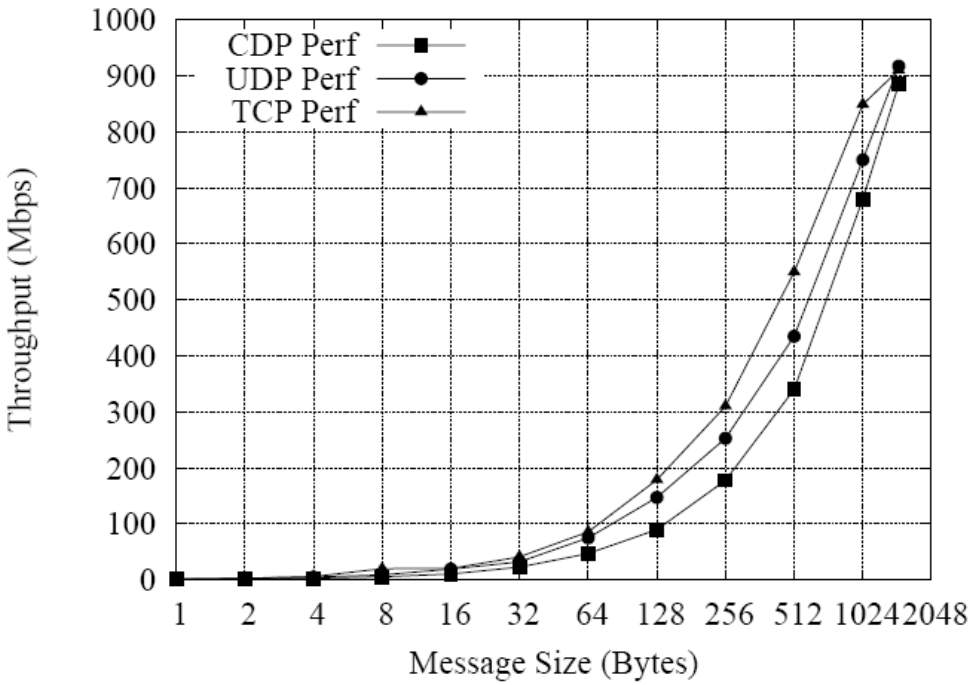
(a) Coarse-Grain Lock (b) Fine-Grain Lock

Performance Evaluation: Scalability



- Using 1 to 128 Cyclops-64 Tiny thread library;
- Running on Cyclops-64 FAST simulator;
- 128 connections dynamically generated;
- Process 256,000 packets;
- Payload per datagram: 1472 bytes;

Performance Evaluation: Throughput

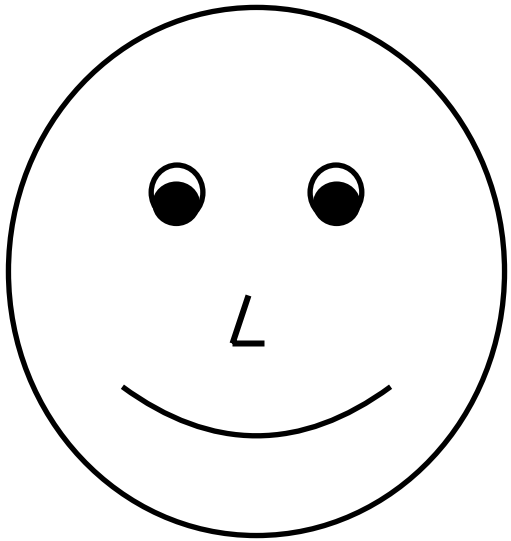


- Pthread version;
- Sequential execution;
- Linux user mode program;
- Peak throughput number: 884Mbps
- TCP: 927Mbps
- UDP: 920Mbps
- Within 95%

Conclusion

- The Tiny thread implementation of CDP demonstrate good scalability. It can sustain reasonable number of CDP connections scaling.
- As a communication protocol, CDP is comparable with TCP and UDP. Its pthread version running in user mode achieves 884Mbps peak throughput on Gigabit Ethernet.

Q & A:



Thank You!