MATRIX and Distributed Job Launch Hands-on Tutorial

Ke Wang
Data-Intensive Laboratory
Illinois Institute of Technology
Sept. 6th, 2013, CS554 Teaching

MATRIX

- **Requirement and Dependencies**
  - Linux
  - GCC
  - Google Protocol Buffer
    - Version 2.4.1: https://protobuf.googlecode.com/files/protobuf-2.4.1.tar.gz
  - Google Protocol Buffer C-bindings
    - Version 0.15: https://code.google.com/p/protobuf-c/downloads/detail?name=protobuf-c-0.15.tar.gz

MATRIX Compile Steps

- 1. Install Google Protocol Buffers
  - Refer to ZHT tutorial
- 2. Install Google Protocol Buffers C-binding
  - Refer to ZHT tutorial
- 3. Download MATRIX code
  - https://github.com/anupammr89/matrix
- 4. Unzip MATRIX code
- 5. cd to “matrix-master” folder
- 6. read the “README” file for details first
- 7. do “make” to compile

Running a simple MATRIX Benchmark

- 1. Running the “server”
  - ./server portNo neighbor zht.cfg TCP userName logging max_tasks_per_package num_tasks prefix shared
  - Arguments Description
    - portNo is the MATRIX server port number (50000)
    - neighbo is the membership list file: 126.47.142.97 50000
    - Zht.cfg is to configure other parameters in ZHT (use default)
    - TCP is to use TCP protocol, instead, UDP is used
    - userName is auxiliary parameter used by ZHT: “kwang”
    - logging is the flag to turn ON/OFF logging information: 0/1
    - max_tasks_per_package is the number of tasks to bundle in single package before submitting to MATRIX server: 8
    - num_tasks is the total number of tasks submitted by the client: 8
    - prefix is the path to logging directory to be used by every instance of MATRIX server, it is private to every instance: /tmp
    - shared is the path to logging directory that is shared by all MATRIX servers: ~/Downloads/

Running a simple MATRIX Benchmark

- 1. Running the “client”
  - ./client per_client_task neighbor zht.cfg TCP sleep_time logging max_tasks_per_package client_index num_tasks prefix shared DAG_choice
  - Arguments Description:
    - per_client_task is the number of tasks to be submitted by this client; useful when there are multiple clients trying to submit tasks to MATRIX
    - sleep_time is the duration of sleep task that is submitted: 0
    - client_index is the index of this client; used when there are multiple clients: 0
    - DAG_choice is the choice of DAG to be used; 0 - Bag of Tasks, 1 - Random DAG, 2 - Pipeline DAG, 3 - Fan In DAG, 4 - Fan Out DAG
    - Rest arguments are the same as mentioned for the server

MATRIX Cluster Deployment

- Copy the binaries i.e. server and client, to every node
- Copy configure files, that is, neighbor, zht.cfg, to every node
- Edit the the file “neighbor” to contain all the ip and port of these nodes, DO NOT USE localhost
- In every node, run as mentioned in Benchmark

Distributed Job Launch

- **Requirement and Dependencies**
  - Linux
  - GCC
  - ZHT
  - SLURM
    - Version 2.5.3

DJL Compile Steps

- 1. download DJL code
  - https://github.com/kwangiit/dist_job_launch
  - Unzip the file
  - Use “distjoblaunch_4.zip” as the code base
- 2. unzip “distjoblaunch_4.zip”
- 3. cd to “slurm-2.5.3” folder
- 4. install SLURM
  - http://www.schedmd.com/slurmdocs/quickstart_admin.html
  - ./configure
  - make
  - sudo make install
- 5. copy “./etc/slurm.conf.example” file to the “/etc” directory in where the SLURM is installed, and name it as “slurm.conf” file
- 6. modify the “slurm.conf” file to match your configuration, see the sample “slurm.conf” file in the “slurm-2.5.3” folder
- 7. cd to “slurm-2.5.3/src/ZHT/src” folder to compile ZHT
- 8. compile ZHT: refer to ZHT tutorial
- 9. cd to “controller” folder “slurm-2.5.3/src/controller”
- 10. compile our controller: make

Running a simple DJL Benchmark

- 1. Running ZHT server
  - cd to “slurm-2.5.3/src/ZHT/src”
  - run ZHT server command: ./zhtserver -z zht.conf –n neighbor.conf
- Running controller
  - cd to “slurm-2.5.3/src/controller”
  - run controller: ./controller numController memList partitionSize workload zht_config_file zht_mem_list
    - numController: number of controllers, e. g. 1
    - memList: controller membership list, e. g. localhost
    - partitionSize: number of slurm daemons, e. g. 1
    - workload: workload file for this controller, e.g. srun –N1 /bin/sleep 0
    - zht.conf: zht configure file, e. g. ../../ZHT/src/zht.conf
    - neighbor.conf: zht membership list, e.g. ../../ZHT/src/neighbor.conf
- Running SLURM daemon
  - slurmd –D -vvvvv

DJL program logic

- Initialize as ZHT client: c_zht_init
- Read the controller membership list
- Read workload
- Create a thread to receive messages
- Create individual thread to launch each job
  - allocate nodes
  - create job
  - create job step
  - launch the job step
- Wait until all jobs are finished

More Information

- More information:
  - http://datasys.cs.iit.edu/~kewang/
- Contact:
  - kwang22@hawk.iit.edu
- Questions?