AMD

ROCm

Chapter 6.1: Running RCCL On ROCm

ROCm Tutorial | AMD 2020

AMD
ROCm

Table of Contents

Chapter 6.1: Running RCCL On ROCm

This hands-on tutorial shows how we can calculate the sum of arrays that are located on multiple GPUs and distribute the data to each GPU. We will be using the RCCL library with AllReduce as the main operation of this application.

Preparation

1. Ensure you have a multi-GPU system to run this example (the example uses 2)

2. Ensure ROCm and HIP are installed correctly by following our installation manuals

3. RCCL needs to be installed before proceeding.

4. You also need to install RCCL separately.

   ○ https://github.com/ROCmSoftwarePlatform/rccl contains instructions on installing RCCL.

   ○ Recommended and easy way is to use the script “install.sh”

   ○ If you have root access run “./install.sh -i” which install to /opt/rocm

   ○ If you do not have root access run “./install.sh --prefix YOUR_PATH” where “YOUR_PATH” refers to the location that you want to install RCCL.

   ○ Also ensure your LD_LIBRARY_PATH points to this location which contains librccl.so

Application Logic

1. The RCCL communicator is first created

2. The application then initializes the send and receive buffer for each GPU using the regular HIP allocation APIs as well as creates HIP streams.

3. The buffers basically contain the number 1 repeated as many times as the number of elements in the buffer.

4. RCCL communicator is then initialized

5. This is followed by running the RCCL "AllReduce" operation which basically does the device-wide reduction and broadcasts the results to the different GPUs.

6. In the example code, each element in input buffer on the device has the value of 1 and we have four devices. So, doing an AllReduce, will result in a output value of 4.
   - For demonstration we are printing a few elements from the output buffer of device 0

7. The independent streams on the different devices are then synchronized to ensure the operation has been completed on all devices

8. The GPU buffers are then freed

9. Finally, the communicator is destroyed

Compiling and Executing

1. Check the number of GPUs available

   - Run /opt/rocm/bin/rocminfo

2. Update the number of GPUs and the device list to fit the platform.

3. Compile the program:

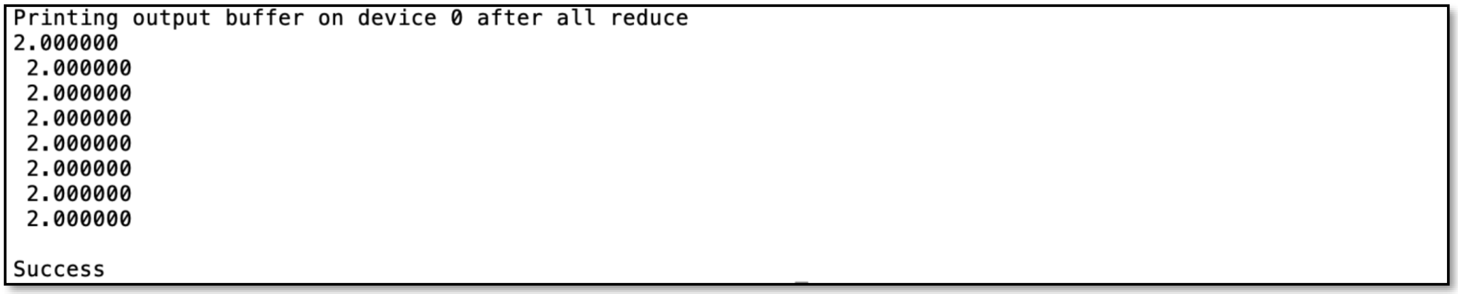
   - hipcc multigpu_rccl.cc -I/opt/rocm/rccl/include -L/opt/rocm/rccl/lib/ -lrccl -o multigpu_rccl

   - If RCCL is installed in some other path, modify the include flags(-I) and linker flags(-L) accordingly

4. Run the program

   - ./multigpu_rccl

5. You will see the output as shown below:



```
Printing output buffer on device 0 after all reduce
2.000000
 2.000000
 2.000000
 2.000000
 2.000000
 2.000000
 2.000000
 2.000000

Success
```

Figure 1: Output of Multi-GPU RCCL example