Artifact – Getting started

This Guide explains the needed software and hardware requirements to reproduce the paper experiments for the paper “Accelerating Parallel Operation for Compacting Selected Elements on GPUs”. A single Zip file contains all needed source code, run scripts and visualization scripts.
All paths mentioned are relative to the root folder of the zip archive.

To perform the artifact experiments the following minimum requirements must be met:

Operating System: Linux
Hardware: Nvidia GPU, capable of Cuda 11.x

The following software must be installed to be able to compile and execute the artifact:

- Cmake: 3.22.2
- CUDA SDK: 11.2
- NVCC: 11.2.67
- G++ Compiler: 9.3.0-17
- Python 3.10.2

Python Packages to install:

- numpy 1.22.3
- Seaborn 0.11.2
- Pandas 1.4.1
- Matplotlib 3.5.1

Step-by-Step reproducibility instructions

Intro

This guide consists of several step by step instructions that describe how to run the experiments and visualize the results into graphs. These graphs can be compared to the figures in our paper.

Please make sure that all requirements that are mentioned in getting_started.pdf are met.

While we conducted our experiments on 4 different GPUs, we assume that only 1 GPU will be used for the reproducibility experiments, according to the information on the artifact submission website.

If you happen to have access to 4 different GPUs we also provide an option to include all 4 in the reproduction.

The results across different GPUs are very similar with the exception of the A100 Ampere chip. In this case SPACE performs much better compared to CUB across all experiments.

Overview

In this Section we give an overview of the contents of the zip archive. In the root folder the paper and an overview guide can be found.

In the SPACE Folder all content needed to reproduce the papers experiments is present.

Folders in /SPACE:

- src: contains our complete source code used for this paper
- deps: dependencies to the librarys Thrust and CUB which are need for the experiments
- data: CSV data from experiments will be saved in this folder
- vis: empty folder that will contain the visualizations generated from the csv files in /data

Scripts:

- run_full.py conducts all experiments like the paper
- run_small.py reduces the data points by 50%. Instead of increasing the % of selected data by 4%, it is increased by 8% per step.
- run_exp2_only.py: runs only experiment 2
- visualize_results.py: draws 4 graphs which correspond to the 4 graph figures in our paper
- visualize_exp2_only.py: only draws experiment 2

Experiments:

There are a total of 4 different experiments. This table shows how the experiments in the artifact match to figures in the paper.

Experiment 1	Figure 1 (CUB vs best SPACE)
Experiment 2	Figure 5 (4 GPUs all algorithms)
Experiment 3	Figure 6 (low % selected data)
Experiment 4	Figure 7 (influence of datatypes)

Step-by-step guide

Compile the experiment binary from source code

Unpack the archive at a location where you have rights to read, write and execute.

Navigate to the folder /SPACE/ Execute the following commands:

```
./build.sh
```

If you lack the permission to do so execute:

```
sudo chmod 775 build.sh
```

Now a new binary gpu_compressstore2 has been created

Run experiments

Two different python runscripts are provided. run_full.py executes the experiments with the same settings as shown in our paper. The runtime on a PCIE3 RTX Turing 8000 is about 2h 30min. On a more modern PCIE4 platform, the runtime should be about 1 hour.

In case you want to run a smaller set of experiments, you can execute run_small.py. This script reduces the data points by half. The increment of the % of selected data is increased from 4% to 8% to achieve this.

If there are other GPUs available on which the paper can be reproduced, you can run experiment 2 on the other platforms.

Start the experiment:

Option 1 (full run – about 2h 30 on TURING RTX 8000 – PCIE3):

```
nohup python3 run_full.py
```

Option 2(reduced run – about 1h 15 on TURING RTX 8000 – PCIE3):

```
nohup python3 run_small.py
```

Option 3 (additionally run only experiment 2 on other platforms):

Copy over the zip archive to the target platform. Unpack the archive. Compile the binary on the platform. Run only the following script:

```
nohup python3 run_exp2_only.py
```

All results are written as csv files at /data

Visualization

A script that visualizes all 4 experiments is provided. In case of Option 1 or Option 2, just execute the script to generates graphs as png and pdf:

```
python3 visualize_results.py
```

The created graphs can be found at /SPACE/vis.

If you want to perform experiment2 on different platforms, copy over the results into /data and run:

```
python3 visualize_exp2_only.py
```

Comparison to Paper results

Compare the results found in /SPACE/vis with the figures in the paper.