

Positron acceleration and electron filament energy recovery

How to get HiPACE++ running on Glenbox

Useful documentation on building/installing HiPACE++: <https://hipace.readthedocs.io/en/latest/building/building.html>

1. ssh into Glen's PC (from my laptop, typically I would ssh into centos7 and then ssh into pc95258).
- a. I set up an ssh hop proxy for ease of access (in `.ssh/config`):

```
Host centos7-proxy
  HostName centos7.slac.stanford.edu
  User mvarvera
  ControlMaster auto
  ControlPath ~/.ssh/master-%r@%h:%p

Host pc95258-proxy
  HostName pc95258
  User mvarvera
  ControlMaster auto
  ControlPath ~/.ssh/master-%r@%h:%p
  ProxyJump centos7-proxy
```

2. Use Spack to create an environment for HiPACE++

```
spack env
create hipace-
dev
spack env
activate
hipace-dev
spack add
ccache %
gcc@11.3.0
spack add
cmake % gcc@11.
3.0
spack add fftw
% gcc@11.3.0
spack add hdf5
% gcc@11.3.0
spack add mpi
% gcc@11.3.0
spack add
pkgconfig %
gcc@11.3.0
spack add cuda
% gcc@11.3.0
spack install
% gcc@11.3.0
```

It will take a while to install everything...(As per the HiPACE++ documentation: in new terminals, re-activate the environment with `spack env activate hipace-dev` again)

Then edit the `spack config` and change `unify: true` to `unify: when_possible`

spack config edit

3. Configure the compiler (or do the meatier compilation in step 5 right off the bat)

```
export CC=$(which gcc-11)
export CXX=$(which g++-11)
export CUDACXX=$(which nvcc)
export CUDAHOSTCXX=$(which g++-11)

export GPUS_PER_SOCKET=1
export GPUS_PER_NODE=2
export AMREX_CUDA_ARCH=7.0 # use 8.0 for A100 or 7.0 for V100
```

For some reason g++ versions above 8 are unsupported with the version of CUDA I used, so you have to manually set CUDAHOSTCXX to your g++-7 path (version 8 is not on this machine I guess but 7 seems to work for my purposes)

4. Clone (or pull if already cloned an older version) the HiPACE++ GitHub repo:

```
`git clone https://github.com/Hi-PACE/hipace.git $HOME/src/hipace # or choose your preferred path`
```

(or pull if already cloned an older version using `git pull`)

5. configure the program (run this in \$HOME/src/hipace directory):

```
cmake -S . -B build -DHiPACE_COMPUTE=CUDA
```

(Or do this all at once)

```
cmake -S . -B build -DHiPACE_COMPUTE=CUDA \  
-DCMAKE_C_COMPILER=$(which gcc-11) \  
-DCMAKE_CXX_COMPILER=$(which g++-11) \  
-DCMAKE_CUDA_COMPILER=$(which nvcc) \  
-DCMAKE_CUDA_HOST_COMPILER=$(which g++-11) \  
-DAMReX_CUDA_ARCH=Volta \  
-DGPUS_PER_SOCKET=1 \  
-DGPUS_PER_NODE=2
```

6. build using n threads (replace n with an integer, 4 for example):

```
cmake --build build -j <n>
```

7. To run a simulation, you can execute:

```
<path>/<to>/hipace/build/bin/hipace <input_file_name>
```

How to copy insitu & hdf5 files from Glenbox over to local machine for data analysis simultaneously:

There's probably a better way, but this is what I've been using for now. This is setup with my ssh proxy hop thing. Run command in the local directory where you want files to be copied to.

```
scp -r -oControlPath=~/.ssh/master-%r@%h:%p pc95258-proxy:'/home/mvarvera/HiPACE++/<path to insitu data>/' \  
<desired local insitu path> && \  
scp -r -oControlPath=~/.ssh/master-%r@%h:%p pc95258-proxy:'/home/mvarvera/HiPACE++/<path to hdf5 data>/' \  
<desired local hdf5 path>
```

Useful links

GitHub with relevant code: <https://github.com/MaxVarverakis/PositronPWFA>

EAAC 2023 Proceedings Paper: <https://doi.org/10.48550/arXiv.2311.07087>

Summer internship overview slides: https://docs.google.com/presentation/d/11A7qIPXztXu0Elt8Cbj2m_hojxxPuYZjqP9pSMLbxg4/edit?usp=sharing

Slides from meeting with Severin Diederichs, Carl Schroeder, Spencer Gessner, Robert Holtzapfel: <https://docs.google.com/presentation/d/1VmR9LG82rfL51h9zfaRNk5IMi9Xj31UNgLKcDrwNI9o/edit?usp=sharing>

Linear regime analysis notes/comparisons to simulations: [Linear_WFA_Notes.pdf](#)