

GFortran と Co-array Fortran

Yuma Osada

Feb 3, 2022

目次

1	最初に失敗話	1
2	実行環境	2
3	Co-array Fortran	3
4	gfortran で Co-array Fortran を使う方法	3
4.1	apt を使う方法	3
4.2	ソースからビルドする方法	5
5	実行	6
5.1	apt から	7
5.2	caf	7
5.3	速度の比較	8
6	結論	8
7	参考	8

1 最初に失敗話

私の PC では最初に Co-array Fortran を使っても高速化しなかった. おそらく, gfortran や mpirun などの PATH がめちゃくちゃだったせい.(再現ができないので原

因が分からない...) 環境の整理をしたら直った.

2 実行環境

- Ubuntu20.04
- Ubuntu20.04 on VirtualBox6.1.14

```
1 echo ${PATH} | sed -s "s=${HOME}=~/g"
```

```
~/ .nvm/versions/node/v16.3.0/bin:~/ .cabal/bin:/usr/local/gcc-11.2.0/bin:/opt/bin:~
```

```
1 which gfortran
```

```
/usr/local/gcc-11.2.0/bin/gfortran
```

```
1 gfortran --version
```

```
GNU Fortran (GCC) 11.2.0
```

```
Copyright (C) 2021 Free Software Foundation, Inc.
```

```
This is free software; see the source for copying conditions.  There is NO  
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```

```
1 which mpirun
```

```
/usr/local/gcc-11.2.0/bin/mpirun
```

```
1 mpirun --version
```

```
mpirun (Open MPI) 4.1.2
```

```
Report bugs to http://www.open-mpi.org/community/help/
```

3 Co-array Fortran

Co-array Fortran は Fortran の言語仕様の一部に組込まれているが, gfortran のみでは使えない. この記事は Co-array で書かれたプログラムを gfortran により実行するためのメモである. Ubuntu20.04 では `apt` を使う方法とソースからビルドする方法がある. `apt` を使う方法はインストールが楽だがコンパイルが面倒. ソースからビルドするとラッパースクリプトの `caf` と `cafrun` が手に入るためコンパイルは楽になる.

4 gfortran で Co-array Fortran を使う方法

Co-array Fortran のコンパイルのテスト用ファイル.

```
1 program coarrays_test
2   implicit none
3   integer :: my_image, n_images
4   my_image = this_image()
5   n_images = num_images()
6   print*, "I'm ", my_image, "/", n_images
7 end program coarrays_test
```

4.1 apt を使う方法

まずは `apt` で検索をかけてみる.

```
1 apt search coarrays
```

ソート中...

全文検索...

libcaf-mpich-3/focal 2.8.0-1 amd64

Co-Array Fortran libraries (MPICH)

libcaf-openmpi-3/focal,now 2.8.0-1 amd64 [インストール済み、自動]

Co-Array Fortran libraries (OpenMPI)

libcoarrays-dev/focal,now 2.8.0-1 amd64 [インストール済み]

Co-Array Fortran libraries

libcoarrays-mpich-dev/focal 2.8.0-1 amd64

Co-Array Fortran libraries for - development files (MPICH)

libcoarrays-openmpi-dev/focal,now 2.8.0-1 amd64 [インストール済み]

Co-Array Fortran libraries - development files (OpenMPI)

mpich と openmpi がある. openmpi を使うならば.

```
1 sudo apt install -y libcoarrays-openmpi-dev
```

実行するには必要なライブラリをリンクする必要があるが, pkg-config を使えば良い.
インストールされた .pc ファイルを検索すると

```
1 dpkg -L libcoarrays-openmpi-dev | grep -e '\.pc$'
```

/usr/lib/x86_64-linux-gnu/pkgconfig/caf-openmpi.pc

/usr/lib/x86_64-linux-gnu/open-coarrays/openmpi/pkgconfig/caf-openmpi.pc

/usr/lib/x86_64-linux-gnu/open-coarrays/openmpi/pkgconfig/caf.pc

OpenMPI を使う場合は, caf-openmpi.pc を利用する.

```
1 gfortran -o coarrays_test.out -fcoarray=lib coarrays_test.f90 $(pkg-config --libs  
  ↪ --cflags caf-openmpi)  
2 mpirun -np 2 ./coarrays_test.out
```

```
I'm          1 /          2  
I'm          2 /          2
```

おそらく, `-lcaf_openmpi` だけで十分.

```
1 gfortran -o coarrays_test.out -fcoarray=lib coarrays_test.f90 -lcaf_openmpi
2 mpirun -np 2 ./coarrays_test.out
```

```
I'm          1 /          2
I'm          2 /          2
```

4.2 ソースからビルドする方法

<http://www.opencoarrays.org/> の <https://github.com/sourceryinstitute/OpenCoarrays/tree/master> からソースコードをクローンする.

```
1 git clone https://github.com/sourceryinstitute/OpenCoarrays.git
2 cd OpenCoarrays/
```

`cmake` を使ってビルドとインストールをする.

```
1 FC=/usr/local/gcc-11.2.0/bin/gfortran cmake -B _build -DBUILD_TYPE=Release
   ↪ -DCMAKE_INSTALL_PREFIX=/usr/local/gcc-11.2.0
2 cmake --build _build
3 cmake --build _build --target install
```

これでラッパースクリプトの `caf` と `cafrun` が `/usr/local/gcc-11.2.0` にインストールされる. `FC` や `PREFIX` を変えれば好きな場所にインストールできる. `caf` が何をラップしているかを知るためには

```
1 caf -w
```

`caf wraps /usr/local/gcc-11.2.0/bin/gfortran-11.2.0`

`caf` の version を知るためには

```
1  caf -v
```

OpenCoarrays Coarray Fortran Compiler Wrapper (caf version 2.9.2-13-g235167d)

Copyright (C) 2015-2020 Sourcery Institute

Copyright (C) 2015-2020 Sourcery, Inc.

OpenCoarrays comes with NO WARRANTY, to the extent permitted by law. You may redistribute copies of OpenCoarrays under the terms of the BSD 3-Clause License. For more information about these matters, see the file named LICENSE that is distributed with OpenCoarrays.

を実行すればよい.

```
1  caf -o coarrays_test_caf.out coarrays_test.f90
2  cafrun -np 2 ./coarrays_test_caf.out
```

```
I'm          1 /          2
I'm          2 /          2
```

5 実行

ソースコードは NAG のチュートリアルからダウンロード https://www.nag-j.co.jp/fortran/coarray/coarrayTutorial_3.html.

```
1  [ -f quad1.f90 ] || wget https://www.nag-j.co.jp/fortran/coarray/code/quad1.f90
2  [ -f coquad1.f90 ] || wget https://www.nag-j.co.jp/fortran/coarray/code/coquad1.f90
3  [ -f coarrays.f90 ] || wget https://www.nag-j.co.jp/fortran/coarray/code/coarrays.f90
4  gfortran -fcoarray=lib -o coarrays.out quad1.f90 coquad1.f90 coarrays.f90 -lcaf_openmpi
5  caf -o coarrays_caf.out quad1.f90 coquad1.f90 coarrays.f90
```

5.1 apt から

```
1 mpirun -np 1 ./coarrays.out
```

Calculated value: 0.900000000000031166
True value (approx): 0.90000000000000002
Relative error 3.4626622556920158E-013
Time taken 9.620 seconds by 1 images, = 9.62 computing power

```
1 mpirun -np 4 ./coarrays.out
```

Calculated value: 0.900000000000022018
True value (approx): 0.90000000000000002
Relative error 2.4461913975907617E-013
Time taken 2.742 seconds by 4 images, = 10.97 computing power

5.2 caf

```
1 cafrun -np 1 ./coarrays_caf.out
```

Calculated value: 0.900000000000031166
True value (approx): 0.90000000000000002
Relative error 3.4626622556920158E-013
Time taken 9.626 seconds by 1 images, = 9.63 computing power

```
1 cafrun -np 4 ./coarrays_caf.out
```

Calculated value: 0.900000000000022018
True value (approx): 0.90000000000000002
Relative error 2.4461913975907617E-013

Time taken 2.644 seconds by 4 images, = 10.58 computing power

5.3 速度の比較

	time	speedup
mpirun -np 1	9.620	1
mpirun -np 4	2.742	3.5083880
cafrun -np 1	9.626	1
cafrun -np 4	2.644	3.6406959

6 結論

どちらの方法でも速度は大して変わらない. おおよそコア数分のスピードアップをしている.

ビルドの手間とコンパイルの手間を天秤にかけると, コンパイルが楽な `caf` を使った方が良いだろう.

7 参考

- OpenCoarrays
<http://www.opencoarrays.org/>
<https://github.com/sourceryinstitute/OpenCoarrays/tree/master>
- NAG のチュートリアル
https://www.nag-j.co.jp/fortran/coarray/coarrayTutorial_3.html