

H27年度
SX-ACE 並列プログラミング入門(MPI)
(演習補足資料)

2015年 6月25日
大阪大学サイバーメディアセンター
日本電気株式会社

演習問題の構成

■ ディレクトリ構成

MPI/

- | - - practice_1 演習問題 1
- | - - practice_2 演習問題 2
- | - - practice_3 演習問題 3
- | - - practice_4 演習問題 4
- | - - practice_5 演習問題 5
- | - - practice_6 演習問題 6
- | - - sample テキスト内のsampleX.fとして掲載しているプログラム
- | - - etc その他, テキスト内のetcX.fとして掲載しているプログラム

3 . 演習問題1 - 1 (practice_1)

■ P16のプログラム(sample2.f)をpractice1.fとしてコピーし、
コンパイル・実行してください

- ・ ファイルのコピー

```
% cd MPI/practice_1
```

```
% cp ../sample/sample2.f practice1.f
```

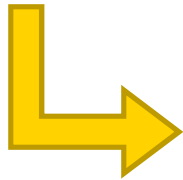
- ・ コンパイル方法

```
% sxmpif90 practice1.f
```

3 . 演習問題1 - 1 (practice_1) つづき

- ・ 実行スクリプトの確認

% **cat run.sh**



```
#!/bin/csh
#PBS -q ACE
#PBS -l cpunum_job=4,memsz_job=60GB,elapstim_req=0:05:00
#PBS -T mpisx
#PBS -b 1
setenv MPIPROGINF DETAIL
cd $PBS_O_WORKDIR
mpirun -np 4 ./a.out
```

- ・ ジョブの投入(実行)

% **qsub run.sh**

- ・ 実行結果の確認

% **cat p1-practice.oXXXX**

(XXXXはシステムにより付与されるジョブID)

3 . 演習問題1 - 2 (practice_1)

■ 演習問題1 - 1で使ったMPIプログラム「Hello World」の結果をランク0のみが出力するように書き換えてください

・ ファイルのコピー

```
% cp practice1.f practice1-2.f
```

・ プログラムの編集

```
% vi practice1-2.f
```

・ コンパイル

```
% sxmpif90 practice1-2.f
```

・ 実行

```
% qsub run.sh
```

4 . 演習問題2 (practice_2)

■ MPIプログラミング入門テキスト P19の1から1000の総和を求める
逐次プログラムを4分割してMPI並列で実行し,各部分和を各ランク
から出力してください.

◆ ヒント:プログラムの流れは下記のとおり

MPIの初期化処理

プロセス数と自プロセスのランク番号の取得

分割時の始点と終点を求める

部分和に初期値(=0)を与える
部分和を求めるループの実行

部分和の出力

MPIの終了化処理

4 . 演習問題2 (practice_2) つづき

- ・ ディレクトリの移動

```
% cd MPI/practice_2
```

- ・ プログラムの編集

逐次プログラムは ディレクトリ practice_2/ にあります .

```
% vi practice2.f
```

- ・ コンパイル

```
% sxmpif90 practice2.f
```

- ・ 実行

```
% qsub run.sh
```

5 . 演習問題3 (practice_3)

■ 演習問題2のプログラムの各ランクの部分和をランク0に集めて、
総和を計算し出力してください

◆ ヒント: 転送処理は以下

ランク1,2,3(0以外)

```
call MPI_SEND(isum,1,MPI_INTEGER,0,  
& itag,MPI_COMM_WORLD,ierr)
```

ランク0

```
call MPI_RECV(isum2,1,MPI_INTEGER,1,  
& itag,MPI_COMM_WORLD,status,ierr)  
call MPI_RECV(isum2,1,MPI_INTEGER,2,  
& itag,MPI_COMM_WORLD,status,ierr)  
call MPI_RECV(isum2,1,MPI_INTEGER,3,  
& itag,MPI_COMM_WORLD,status,ierr)
```

isumで受信するとランク0の部分和が上書きされてしまう

5 . 演習問題3 (practice_3) つづき

- ・ ディレクトリの移動

```
% cd MPI/practice_3
```

- ・ プログラムの編集

演習問題 2の回答例を practice3.f として用意しています

```
% vi practice3.f
```

- ・ コンパイル

```
% sxmpif90 practice3.f
```

- ・ 実行

```
% qsub run.sh
```

6 . 演習問題4 (practice_4)

■ 演習問題3のプログラムで、各ランクの部分和をMPI_REDUCEを使用してランク0に集計して、ランク0から結果を出力してください

・ディレクトリの移動

```
% cd MPI/practice_4
```

・プログラムの編集

演習問題3の回答例を practice4.f として用意しています

```
% vi practice4.f
```

・コンパイル

```
% sxmpif90 practice4.f
```

・実行

```
% qsub run.sh
```

8 . 演習問題 5 (practice_5)

■ P59のetc4.fをP57の「代表プロセス入力 + メモリ削減」の例のように、各プロセスに必要な領域だけ確保するように修正してください。

◆ ヒント:

senddata,recvdataを動的に確保するようにallocatable宣言する

各プロセスが確保する領域(ist,iend)を求める

各プロセスで必要なsenddataの領域を確保する(allocate)

ランク0でrecvdataの領域を確保する(allocate)

8 . 演習問題 5 (practice_5) つづき

- ・ ディレクトリの移動

```
% cd MPI/practice_5
```

- ・ プログラムの編集

practice5.f を用意しています

```
% vi practice5.f
```

- ・ コンパイル

```
% sxmpif90 practice5.f
```

- ・ 実行

```
% qsub run.sh
```

9 . 演習問題6 (practice_6)

行列積プログラムをMPIで並列化してください

```
implicit real(8)(a-h,o-z)
parameter ( n=12000 )
real(8) a(n,n),b(n,n),c(n,n)
real(4) etime,cp1(2),cp2(2),t1,t2,t3
do j = 1,n
  do i = 1,n
    a(i,j) = 0.0d0
    b(i,j) = n+1-max(i,j)
    c(i,j) = n+1-max(i,j)
  enddo
enddo
write(6,50) ' Matrix Size = ',n
50 format(1x,a,i5)
t1=etime(cp1)
do j=1,n
  do k=1,n
    do i=1,n
      a(i,j)=a(i,j)+b(i,k)*c(k,j)
    end do
  end do
end do
t2=etime(cp2)
t3=cp2(1)-cp1(1)
write(6,60) ' Execution Time = ',t2,' sec', ' A(n,n) = ',a(n,n)
60 format(1x,a,f10.3,a,1x,a,d24.15)
stop
end
```

- ◆ 左記に行列積を行うプログラムをMPI化して4プロセスで実行してください。出力はプロセス0で行ってください。

9 . 演習問題6 (practice_6) つづき

◆ ヒント: プログラムの流れは下記のとおり

MPIの初期化処理

プロセス数と自プロセスのランク番号の取得

分割時の始点と終点を求める

解を格納する配列aの初期化
行列bとcの値の設定

各プロセスが担当する範囲の行列積を計算

解を格納する配列aをランク0に集める

ランク0が結果を出力

MPIの終了化処理

時間計測はMPI_Wtimeを使用する

時間を格納する変数はreal*8で定義する

```
real*8 t1,t2
```

測定する区間の始まりと終わりの時間を計測する

```
call MPI_BARRIER(MPI_COMM_WORLD,IERR)  
t1=MPI_WTIME()
```

[測定区間]

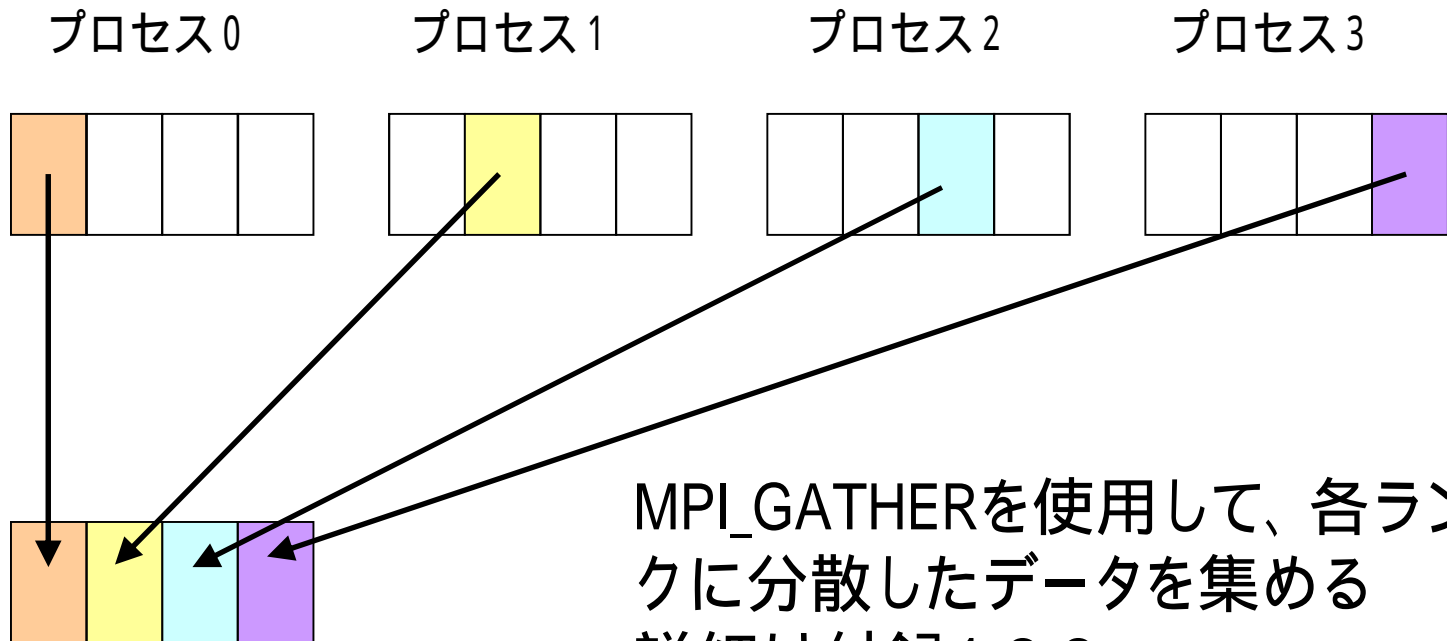
```
call MPI_BARRIER(MPI_COMM_WORLD,IERR)  
t2=MPI_WTIME()
```

t2 - t1が計測区間の時間となる

9 . 演習問題6 (practice_6) つづき

データの転送方法(行列 - ベクトル積)

- プロセス0はプロセス1, 2, 3から計算結果を格納した配列xを受け取る(下図)



9 . 演習問題6 (practice_6) つづき

- ・ ディレクトリの移動

```
% cd MPI/practice_6
```

- ・ プログラムの編集

MPIプログラム入門テキストP90のsample6.fをpractice6.fとして用意しています

```
% vi practice6.f
```

- ・ コンパイル

```
% sxmpif90 practice6.f
```

- ・ 実行

```
% qsub run.sh
```

演習問題解答例

3 . 演習問題1 - 2 (practice_1) 解答例

```
program example1
  include 'mpif.h'
  integer ierr,myrank
  call MPI_INIT(ierr)
  call MPI_COMM_RANK(MPI_COMM_WORLD,myrank,ierr)
  if(myrank.eq.0) print *, "Hello World",myrank
  call MPI_FINALIZE(ierr)
  stop
end
```

```
% sxmpif90 practice1.f
% qsub run.sh
% cat p1-practice.oXXXX
  Hello World 0
```

4 . 演習問題2 (practice_2) 解答例

```
program example2
  include 'mpif.h'
  integer ierr,myrank,nprocs,ist,ied
  parameter(n=1000)
  integer isum
  call MPI_INIT(ierr)
  call MPI_COMM_SIZE(MPI_COMM_WORLD,nprocs,ierr)
  call MPI_COMM_RANK(MPI_COMM_WORLD,myrank,ierr)
  ist=((n-1)/nprocs+1)* myrank+1
  ied=((n-1)/nprocs+1)*(myrank+1)
  isum=0
  do i=ist,ied
    isum=isum+i
  enddo
  write(6,6000) myrank,isum
6000 format("Total of Rank:",i2,i10)
  call MPI_FINALIZE(ierr)
  stop
end
```

```
% sxmpif90 practice2.f
% qsub run.sh
% cat p2-practice.oXXXX
  Total of Rank: 0    31375
  Total of Rank: 2   156375
  Total of Rank: 3   218875
  Total of Rank: 1    93875
```

5 . 演習問題3 (practice_3) 解答例

```
program example3
include 'mpif.h'
integer ierr,myrank,nprocs,ist,ied
integer status(MPI_STATUS_SIZE)
parameter(n=1000)
integer isum, isum2
call MPI_INIT(ierr)
call MPI_COMM_SIZE(MPI_COMM_WORLD,nprocs,ierr)
call MPI_COMM_RANK(MPI_COMM_WORLD,myrank,ierr)
ist=((n-1)/nprocs+1)*myrank+1
ied=((n-1)/nprocs+1)*(myrank+1)
isum=0
do i=ist,ied
    isum=isum+i
enddo
```

5 . 演習問題3 (practice_3) 解答例(つづき)

```
    itag=1
    if(myrank.ne.0) then
        call MPI_SEND(isum,1,MPI_INTEGER,0,
&                itag,MPI_COMM_WORLD,ierr)
    else
        call MPI_RECV(isum2,1,MPI_INTEGER,1,
&                itag,MPI_COMM_WORLD,status,ierr)
        isum=isum+isum2
        call MPI_RECV(isum2,1,MPI_INTEGER,2,
&                itag,MPI_COMM_WORLD,status,ierr)
        isum=isum+isum2
        call MPI_RECV(isum2,1,MPI_INTEGER,3,
&                itag,MPI_COMM_WORLD,status,ierr)
        isum=isum+isum2
        write(6,6000) isum
6000    format("Total Sum = ",i10)
    endif
    call MPI_FINALIZE(ierr)
    stop
end
```

```
% sxmpif90 practice3.f
% qsub run.sh
% cat p3-practice.oXXXX
    Total Sum =    500500
```

6 . 演習問題4 (practice_4) 解答例

```
program example4
include 'mpif.h'
integer ierr,myrank,nprocs,ist,ied
parameter(n=1000)
integer isum,isum2
call MPI_INIT(ierr)
call MPI_COMM_SIZE(MPI_COMM_WORLD,nprocs,ierr)
call MPI_COMM_RANK(MPI_COMM_WORLD,myrank,ierr)
ist=((n-1)/nprocs+1)*myrank+1
ied=((n-1)/nprocs+1)*(myrank+1)
isum=0
do i=ist,ied
    isum=isum+i
enddo
```

6 . 演習問題4 (practice_4) 解答例 (つづき)

```
call MPI_REDUCE(isum, isum2, 1, MPI_INTEGER, MPI_SUM, 0,  
& MPI_COMM_WORLD, ierr)  
if(myrank.eq.0) write(6,6000) isum2  
6000 format("Total Sum = ",i10)  
call MPI_FINALIZE(ierr)  
stop  
end
```

```
% sxmpif90 practice4.f  
% qsub run.sh  
% cat p4-practice.oXXXX  
Total Sum = 500500
```

MPI_REDUCEでは送信するデータと受信するデータの領域に重なりがあってはならない. isumとisum2に分けて使用.

8 . 演習問題5 (practice_5) 解答例

```
include 'mpif.h'
integer,parameter :: numdat=100
integer,allocatable :: senddata(:),recvdata(:)
call MPI_INIT(ierr)
call MPI_COMM_RANK(MPI_COMM_WORLD,myrank,ierr)
call MPI_COMM_SIZE(MPI_COMM_WORLD,nprocs,ierr)
ist = ((numdat-1)/nprocs+1)*myrank+1
ied = ((numdat-1)/nprocs+1)*(myrank+1)
allocate(senddata(ist:ied))
if(myrank.eq.0) allocate(recvdata(numdat))
icount=(numdat-1)/nprocs+1
do i=1,icount
    senddata(icount*myrank+i)=icount*myrank+i
enddo
```


8 . 演習問題5 (practice_5) 解答例(つづき)

```
call MPI_GATHER(senddata( icount*myrank+1),  
&                icount, MPI_INTEGER, recvdata,  
&                icount, MPI_INTEGER, 0, MPI_COMM_WORLD,  
&                ierr)  
if(myrank.eq.0) then  
  open(60, file='fort.60')  
  write(60, '(10I8)') recvdata  
endif  
call MPI_FINALIZE(ierr)  
stop  
end
```

```
% sxmpif90 practice5.f
```

```
% qsub run.sh
```

```
% cat fort.60
```

```
1 2 3 4 5 ...
```

9 . 演習問題6 (practice_6) 解答例

```
program example6
implicit real(8)(a-h,o-z)
include 'mpif.h'
integer ierr,myrank,nprocs,ist,ied
parameter ( n=12000 )
real(8) a(n,n),b(n,n),c(n,n)
real(8) d(n,n)
real(8) t1,t2
call MPI_INIT(ierr)
call MPI_COMM_SIZE(MPI_COMM_WORLD,nprocs,ierr)
call MPI_COMM_RANK(MPI_COMM_WORLD,myrank,ierr)
ist=((n-1)/nprocs+1)*myrank+1
ied=((n-1)/nprocs+1)*(myrank+1)
n2=n/nprocs
```

9 . 演習問題6 (practice_6) 解答例 (つづき)

```
do j = 1,n
  do i = 1,n
    a(i,j) = 0.0d0
    b(i,j) = n+1-max(i,j)
    c(i,j) = n+1-max(i,j)
  enddo
enddo
if(myrank.eq.0) then
write(6,50) ' Matrix Size = ',n
endif
50 format(1x,a,i5)
```

9 . 演習問題6 (practice_6) 解答例 (つづき)

```
call MPI_BARRIER(MPI_COMM_WORLD, ierr)
t1=MPI_WTIME()
do j=ist,ied
  do k=1,n
    do i=1,n
      a(i,j)=a(i,j)+b(i,k)*c(k,j)
    end do
  end do
end do
call MPI_GATHER(a(1,ist),n*n2,MPI_REAL8,d,n*n2
&               ,MPI_REAL8,0,MPI_COMM_WORLD, ierr)
call MPI_BARRIER(MPI_COMM_WORLD, ierr)
t2=MPI_WTIME()
if(myrank.eq.0) then
write(6,60) ' Execution Time = ',t2-t1,' sec', ' A(n,n) = ',d(n,n)
endif
60 format(1x,a, f10.3,a, 1x,a,d24.15)
call MPI_FINALIZE(ierr)
stop
end
```

```
% sxmpif90 practice6.f
% qsub run.sh
% cat p6-practice.oXXXX
Matrix Size = 12000
Execution Time = 13.957 sec
A(n,n) = 0.12000000000000000D+05
```