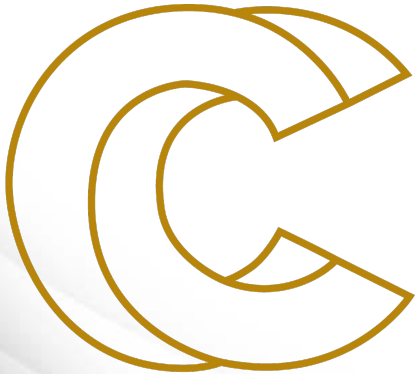

EURO

MPI

Kuramsal Bilgiler

Giriş

MPI’a Genel Bakış

- Varolan dili kullanarak
  - C/C++, Fortran
- Sıradan dilbilgisi ile
  - memcpy eşleniği işlevler
- Dağıtık bellek alanında
  - bağımsız, ayrı bilgisayarlarda barınabilen programlar
- Genellikle işlem/iletişim döngüsünde
  - …, parçayı işle, sonucu paylaş, parçayı işle, sonucu paylaş, …
- Kütüphane ve çalışma ortamları kullanarak
  - OpenMPI, MPICH, …

MPI'ın Geçmişi

- Ön çalışmalar
  - 1991
- MPI 1
  - taslak (1992)
  - taslağın Supercomputing toplantısında sunulması (1993)
  - geri bildirimlere dayanarak yayınlanması (1994)
  - *Message Passing* odaklı
- MPI 2
  - Parallel I/O, RDMA, …
- MPI 3
  - toplu iletişim işlevlerinin *non-blocking* sürümleri

Temel İşlevler

İşlevler

```
int MPI_Init(
    int *argc,
    char ***argv
    )

int MPI_Finalize(
    void
    )
```

İşlevler

```
int MPI_Comm_size(
    MPI_Comm comm,
    int *size
    )

int MPI_Comm_rank(
    MPI_Comm comm,
    int *rank
    )
```

Kavramlar

- communicator
  - bir işlem grubunu belirten değişken
  - MPI_Comm comm
  - MPI_COMM_WORLD
- rank
  - bir işlemin, verilen gruptaki sıra numarası

Örnek

```
#include <mpi.h>
#include <stdio.h>

int main(int argc, char** argv) {
    int rank, size;

    MPI_Init(&argc, &argv);
    MPI_Comm_size(MPI_COMM_WORLD, &size);
    MPI_Comm_rank(MPI_COMM_WORLD, &rank);
    printf("%4d / %4d\n", rank, size);

    MPI_Finalize();
}
```

1-1 İletişim

İşlevler

```
int MPI_Send(
    const void *buf,
    int count,
    MPI_Datatype datatype,
    int dest,
    int tag,
    MPI_Comm comm
    )
```

Kavramlar

- buf / count / datatype
  - buf: gönderilecek verinin başlangıç konumu
  - count: değişken sayısı
  - datatype: değişken tipi
    - MPI_CHAR
    - MPI_INT
    - MPI_FLOAT
    - MPI_DOUBLE
- tag
  - iletileri ayırt etmek için verilen sıra numarası

İşlevler

```
int MPI_Recv(
    const void *buf,
    int count,
    MPI_Datatype datatype,
    int source,
    int tag,
    MPI_Comm comm,
    MPI_Status *status
    )
```

Kavramlar

- status
  - gönderici belirtilmemiş olabilir
    - source: MPI_ANY_SOURCE
    - tag: MPI_ANY_TAG
  - bu durumda ek bilgilere status ile ulaşılabilir

Örnek

```
if( myrank == 0 ) {

    MPI_Recv( b, 10, MPI_INT, 1, 19, MPI_COMM_WORLD, &status );
    MPI_Send( a, 10, MPI_INT, 1, 17, MPI_COMM_WORLD );

} else if( myrank == 1 ) {

    MPI_Recv( b, 10, MPI_INT, 0, 17, MPI_COMM_WORLD, &status );
    MPI_Send( a, 10, MPI_INT, 0, 19, MPI_COMM_WORLD );

}
```

Kavramlar

- deadlock
  - döngüsel gereksinimler nedeni ile tıkanma

Örnek

```
if( myrank == 0 ) {

    MPI_Send( a, 10, MPI_INT, 1, 17, MPI_COMM_WORLD );
    MPI_Recv( b, 10, MPI_INT, 1, 19, MPI_COMM_WORLD, &status );

} else if( myrank == 1 ) {

    MPI_Send( a, 10, MPI_INT, 0, 19, MPI_COMM_WORLD );
    MPI_Recv( b, 10, MPI_INT, 0, 17, MPI_COMM_WORLD, &status );

}
```

Kavramlar

- buffered communication
  - verilerin aktarma öncesi üçüncü bir konuma taşınması
- blocking / non-blocking communication
  - blocking: aktarım sona erene kadar bekler
  - non-blocking: aktarım bitmeden sonraki adıma geçer
- iletişim türünü gönderici belirler

Kavramlar

- synchronous send
  - bittiğinde, alıcının ama işlemine başladığı kesindir
- ready send
  - başlamadan önce alıcının alma isteği göndermesi gerekir
- durum sorgulama işlevleri
  - MPI_Test
  - MPI_Wait

Örnek

```
if( myrank == 0 ) {

    MPI_Recv( b, 10, MPI_INT, 1, 19, MPI_COMM_WORLD, &status );
    MPI_Send( a, 10, MPI_INT, 1, 17, MPI_COMM_WORLD );

} else if( myrank == 1 ) {

    MPI_Send( a, 10, MPI_INT, 0, 19, MPI_COMM_WORLD );
    MPI_Recv( b, 10, MPI_INT, 0, 17, MPI_COMM_WORLD, &status );

}
```

Toplu İletişim

Kavramlar

- 1-N
  - MPI_Bcast
  - MPI_Scatter
- N-1
  - MPI_Reduce
  - MPI_Gather
- N-N
  - MPI_Allreduce
  - MPI_Allgather
  - MPI_Barrier

İşlevler

```
int MPI_Barrier(
    MPI_Comm comm
    )
```

İşlevler

```
int MPI_Bcast(
    void *buffer,
    int count,
    MPI_Datatype datatype,
    int root,
    MPI_Comm comm
    )
```

İşlevler

```
int MPI_Reduce(
    const void *sendbuf,
    void *recvbuf,          // sadece root için önemli
    int count,
    MPI_Datatype datatype,
    MPI_Op op,
    int root,
    MPI_Comm comm
    )
```

Kavramlar

- MPI_Op
  - MPI_MAX, MPI_MIN
  - MPI_SUM, MPI_PROD
  - MPI_LAND, MPI_BAND
  - MPI_LOR, MPI_BOR
  - MPI_LXOR, MPI_BXOR
  - MPI_MINLOC, MPI_MAXLOC
- kullanıcı da tanımlayabilir

İşlevler

```
int MPI_Allreduce(
    const void *sendbuf,
    void *recvbuf,
    int count,
    MPI_Datatype datatype,
    MPI_Op op,
    MPI_Comm comm
    )
```

İşlevler

```
int MPI_Gather(
    const void *sendbuf,
    int sendcount,
    MPI_Datatype sendtype,
    void *recvbuf,              // sadece root için önemli
    int recvcount,              // sadece root için önemli
    MPI_Datatype recvtype,      // sadece root için önemli
    int root,
    MPI_Comm comm
    )
```

İşlevler

```
int MPI_Allgather(
    const void *sendbuf,
    int sendcount,
    MPI_Datatype sendtype,
    void *recvbuf,
    int recvcount,
    MPI_Datatype recvtype,
    MPI_Comm comm
    )
```

İşlevler

```
int MPI_Scatter(
    const void *sendbuf,      // sadece root için önemli
    int sendcount,            // sadece root için önemli
    MPI_Datatype sendtype,    // sadece root için önemli
    void *recvbuf,
    int recvcount,
    MPI_Datatype recvtype,
    int root,
    MPI_Comm comm
    )
```

İleri Konular

Kavramlar

- *Virtual Topologies*
- Sıradışı MPI_Send ve MPI_Recv seçenekleri
- MPI_Sendrecv
- Kullanıcı tanımlı veri türleri
- Kullanıcı tanımlı indirgeme işlemleri
- *Parallel I/O*

Sorular ve Tartışma

EURO

Thanks!

This project has received funding from the European High-Performance Computing Joint Undertaking (JU) under grant agreement No 951732. The JU receives support from the European Union's Horizon 2020 research and innovation programme and Germany, Bulgaria, Austria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Greece, Hungary, Ireland, Italy, Lithuania, Latvia, Poland, Portugal, Romania, Slovenia, Spain, Sweden, United Kingdom, France, Netherlands, Belgium, Luxembourg, Slovakia, Norway, Switzerland, Turkey, Republic of North Macedonia, Iceland, Montenegro