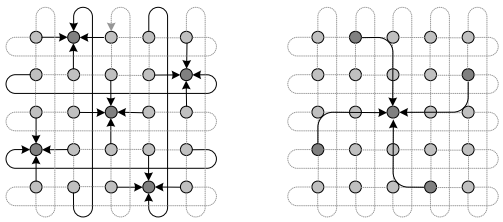
Hacettepe Üniversitesi
Bilgisayar Mühendisliği Bölümü
BİL431 Koşut İşlem
Genel Sınav 13-01-2012

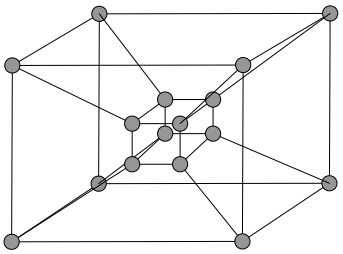
Soru 1. Şekil 1'deki gibi 5x5 boyutundaki "torus" üzerinde "gather" (hepsinden birine verileri gönderme) işlemi gerçekleştiriliyor. Aynı anda gönderilebilen iletilerle iki adımda işlem tamamlanıyor. Gönderilen verilerin 10 birim uzunluğunda olduğunu varsayınız.

a) Bu algoritmanın iletişim maliyetini "Store-and-forward" iletim yöntemi için hesaplayınız. Bir verinin bir zaman biriminde aktarıldığını varsayınız.
b) Bu algoritmanın iletişim maliyetini "Wormhole" iletim yöntemi için hesaplayınız. Bir birim verinin bir zaman biriminde aktarıldığını varsayınız.



Şekil 1

Soru 2."MPI_Bcast" (Broadcast) işlevinin gerçekleştirim adımlarını, 16 işlemcili "hypercube" topolojili koşut bilgisayar için kısaca anlatınız. Kaç ileti süresinde işlem tamamlanır?



Şekil 2

Soru 3. 4x4 boyutundaki "torus" üzerinde "Cannon" matris çarpma algoritmasını kullanarak 100x100 boyutundaki iki matris çarpılmaktadır. Bir verinin bir link üzerinden bir zaman biriminde aktarıldığını ve iki sayıyı çarpıp başka bir sayıya ekleme işleminin $\frac{1}{10}$ zaman biriminde yapıldığını varsayınız.

a) İşletim zamanını hesaplayınız.
b) Hızlanmayı hesaplayınız.
c) Etkinliği hesaplayınız.

Soru 4. İki boyutlu "torus" topolojisindeki bir koşut bilgisayar için aşağıda tanımlanan kod kesimlerini MPI kitaplığını kullanarak kodlayınız.

a) "Her görev en yakın dört komşusu ve kendisindeki verinin en küçüğünü bulur."

b) "Her görev en yakın sekiz komşusu ve kendisindeki verinin en küçüğünü bulur."

Kullanabileceğiniz MPI işlevlerinden bazıları:

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

int MPI_Finalize()

int MPI_Comm_size(MPI_Comm comm, int *size)

int MPI_Comm_rank(MPI_Comm comm, int *rank)


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

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

int MPI_Sendrecv(void *sendbuf, int sendcount,MPI_Datatype senddatatype,
                int dest, int sendtag,void *recvbuf, int recvcount,
                MPI_Datatype recvdatatype,int source, int recvtag,
                MPI_Comm comm,MPI_Status *status)

int MPI_Get_count(MPI_Status *status, MPI_Datatype datatype,int *count)

int MPI_Cart_create(MPI_Comm comm_old, int ndims, int *dims, int
                *periods, int reorder, MPI_Comm *comm_cart)

int MPI_Cart_rank(MPI_Comm comm_cart, int *coords, int *rank)

int MPI_Cart_coord(MPI_Comm comm_cart, int rank, int maxdims,  int
                *coords)

int MPI_Cart_shift(MPI_Comm comm_cart, int dir, int s_step, int
                *rank_source, int *rank_dest)
```

Kullanabileceğiniz değişmezler: **MPI_COMM_WORLD,MPI_ANY_SOURCE,MPI_ANY_TAG**