

Examen Multiprocesadores

Otoño 2005

Preguntas/Problemas (sin apuntes 40')

Responder razonadamente las siguientes cuestiones:

1. Cual es la diferencia esencial entre una arquitectura de memoria compartida y una de memoria distribuida.
2. Cual es la posible ventaja de gang scheduling frente a FIFO?
3. Describir en que situaciones se usaría la primitiva `irecv` de MPI
4. Describir el proceso de “read merge” en un sistema basado en bus de ciclo partido.
5. En un sistema de coherencia basado en directorio:
 - Quien establece el orden de los accesos a una variable?
 - Que condiciones es necesario imponer para mantener consistencia secuencial?
 - En que situaciones se usaría mensajes de tipo NACK.
6. Que es el problema de “back to back barriers”? Como se resuelve?

Problema: Difusion

Tenemos el siguiente código que se ejecuta dentro de un bucle iterativo externo que lo ejecuta 1000 veces.

```
Double A[N+2][N+2];

for (i=1;i<N;i++) {
    for (j=1;j<N;j++) {
        tmp = A[i][j];
        A[i][j] = 0.2*(A[i][j]
                      +A[i][j-1]+A[i-1][j]
                      +A[i][j+1]+A[i+1][j]);
    }
}
```

1. Que dependencias tiene el bucle? Indicar el grado de paralelismo existente en el algoritmo. Es constante?
2. Paralelizar el bucle con MPI partiendo la matriz por columnas y preservando todas las dependencias.
3. Que topología de red de interconexión sería adecuada para ejecutar este código?

4. Hacer un modelo del tiempo de ejecución en función del número de procesadores suponiendo que el procesador fuera de 1 MFLOP, las llamadas MPI tuvieran un overhead de 10 microsegundos y el ancho de banda de los links de la red de interconexión 100 MB/s.
5. Y si el procesador fuera de velocidad infinita, el overhead MPI de 0 microsegundos y el ancho de banda de 1MB/s ?
6. Tenemos un sistema con topología de interconexión malla 1D con 8 procesadores. Describir cualitativamente cómo impactarían en el rendimiento distintos mapeos?
7. Cual sería el peor mapeo posible?

Encapsulamos la siguiente variante del bucle

```
process(int a) {
    for (i=1;i<N;i++) {
        start = (i+a)%2+1;
        for (j=start;j<N;j+=2) {
            tmp = A[i][j];
            A[i][j] = 0.2*(A[i][j]
                          +A[i][j-1]+A[i-1][j]
                          +A[i][j+1]+A[i+1][j]);
        }
    }
}
```

8. Que dependencias tiene el bucle?
9. Como se paralelizaría con OpenMP?
10. Si N es 1024, cuantos fallos de cache se producirían en cada procesador si la cache es de 1MB y se ejecuta con 4 procesadores? Y si se ejecuta con 16?

El bucle iterativo hace ahora dos invocaciones de process

```
process(1);
process(0);
```

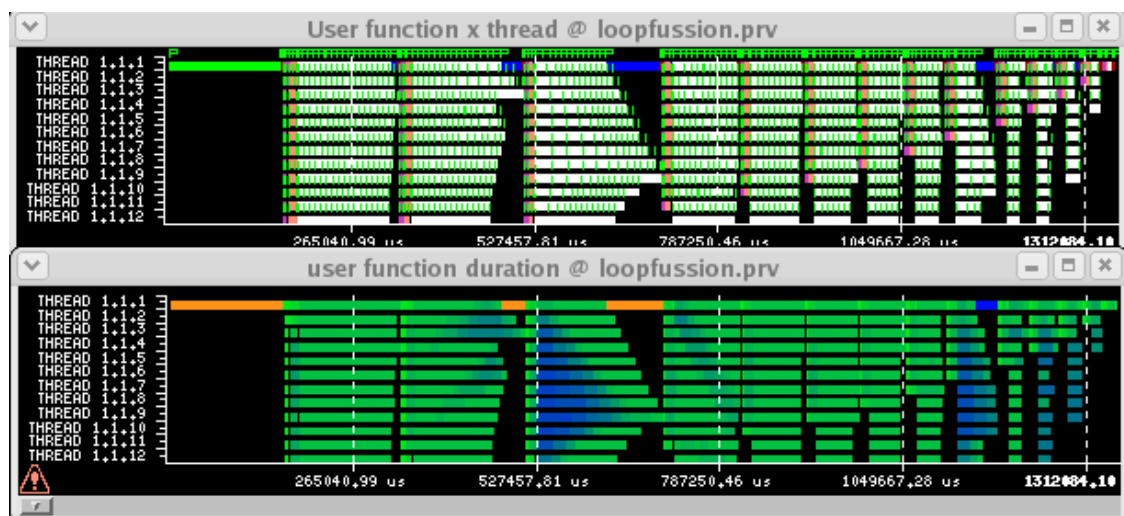
11. Pueden invocarse en paralelo usando la directiva “sections”. Explicar el motivo.
12. Usamos la versión de process paralelizada con OpenMP y los mismos tamaños de problema y máquina que en el punto 10. Cuantas invalidaciones se producirán.
13. Cual es el impacto esperado de esas invalidaciones en el rendimiento para cada caso?

Problema: LUBb

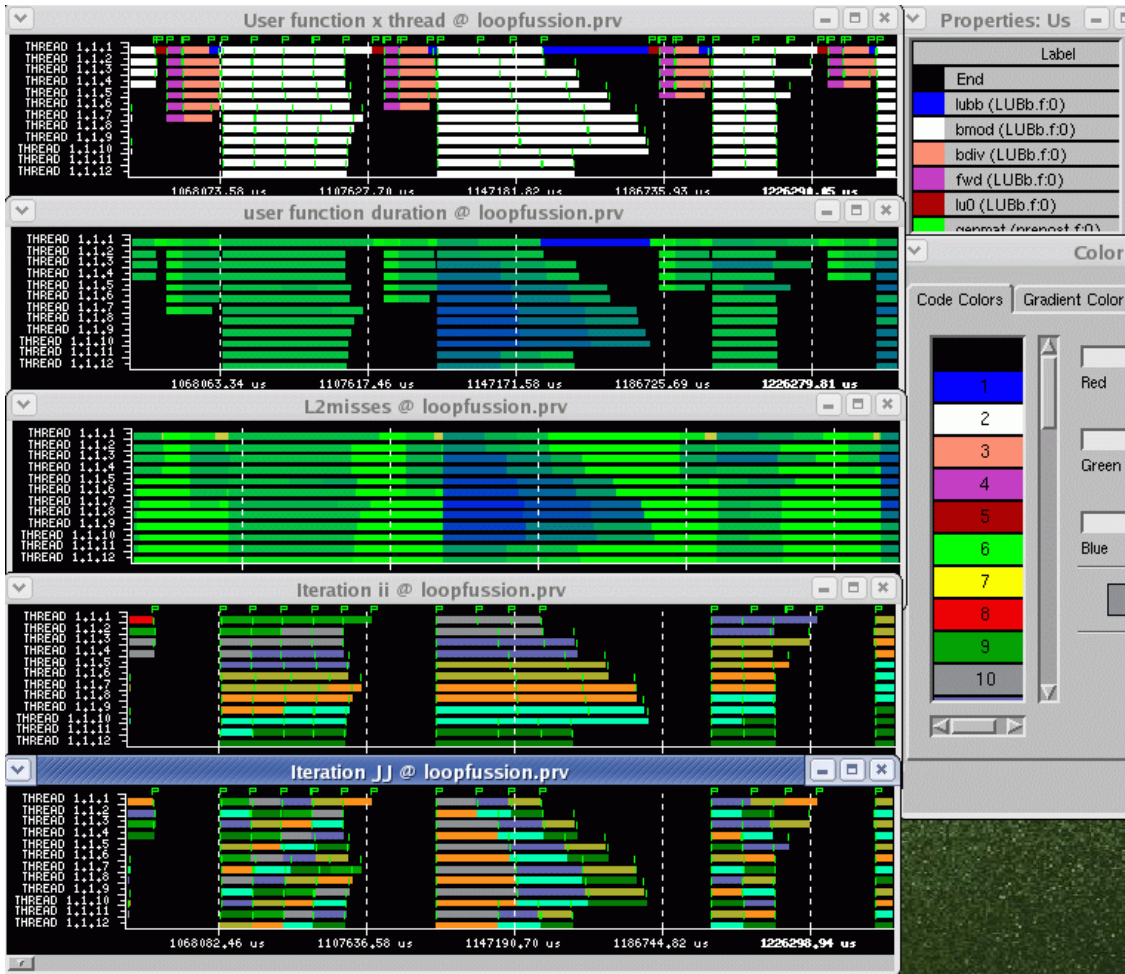
Comentar la paralelización siguiente del programa LUBb de la práctica (recordar que el tamaño de la matriz es de 1500x1500 (doubles – 8bytes), el tamaño de bloque 100x100, fortran almacena por columnas)

```
do kk=1,NB
  call lu0(A,kk)
c$omp parallel do private (jj)
  do jj=kk+1,NB
    call fwd(A,kk,jj)
  enddo
c$omp parallel do private (ii)
  do ii=kk+1,NB
    call bdiv (A,ii,kk)
  enddo
  nblocks=NB-ii
c$omp parallel do private (ii,jj,IJ)
  do IJ=0, nblocks*nblocks-1
    ii = IJ/nblocks+ii
    jj= MOD(IJ,nblocks)+ii
    call bmod(A,ii,jj,kk)
  enddo
enddo
```

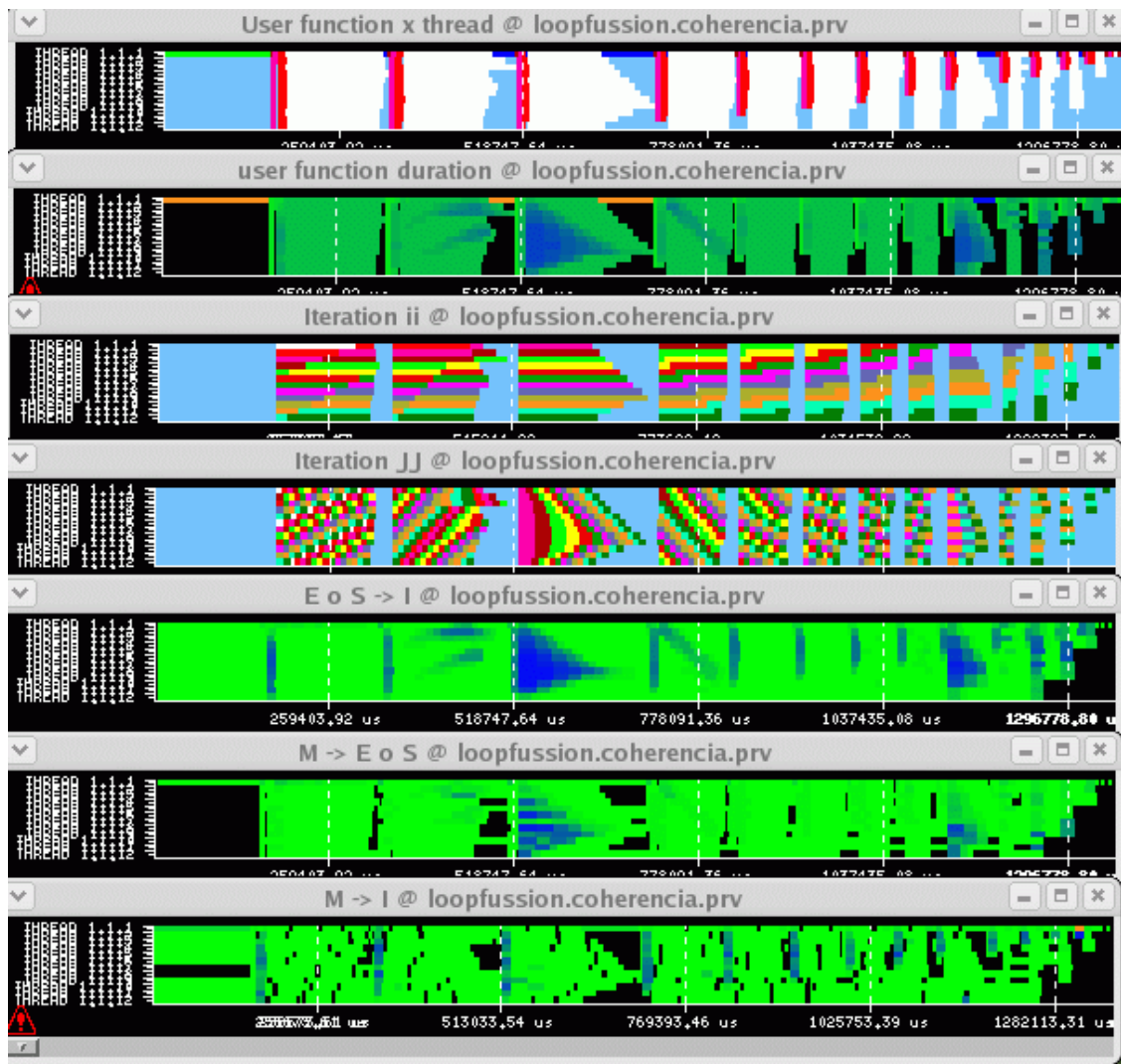
1. Que ventajas/inconvenientes (en cuanto a grado de paralelismo explotado y balanceo de carga) tiene con respecto a las primeras paralelizaciones de grano grueso y fino hechas en el laboratorio.
2. Se ha insertado en el fuente llamadas a ompitrace_event para indicar los valores de ii, jj y nblocks. La siguiente figura muestra las funciones de usuario, su duración. Comentarlos



- La siguiente figura es un zoom del área de tres iteraciones kk tales que nblocks vale 7, 8 y 9 respectivamente? Se muestran también las iteraciones que le toca a cada thread y los fallos de cache. Cual es la posible causa de que ciertos valores de kk tengan problemas de rendimiento?



- La siguiente figura da las transiciones de estado para la misma zona. (la escala ara las transiciones M->I es de 0 a 200, para las otras dos de 0 a 25000) Explicar el comportamiento.



5. Cómo podría modificarse el programa para mejorar el rendimiento el comportamiento de memoria del bucle IJ.
6. Podría extraerse mayor paralelismo de alguna otra parte del bucle kk? Escribir el programa.