

Jun 28 2006 17:58

rala_csa.f90

Page 1

```

! -----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
! Matrices ralas: formato (i, j, a_ij).
! Producto y=Ax con scatter emulado
! -----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
! INTEL:
! ifort -vms -c m_ctes.f90
! ifort -vms -c m_temps.f90
! ifort -vms -c rala_csa.f90
! ifort -vms -o rala_csa.exe1 *.o
!
! GNU:
! g95 -w -c m_ctes.f90 --free-form
! g95 -w -c m_tools.f90 --free-form
! g95 -w -c rala_csa.f90 --free-form
! g95 -w -o rala_csa.exe2 *.o
!
! -----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
program rala_csa
  use m_ctes
  use m_tools
  implicit none
  integer (iin), parameter :: n = 4, p = 7
  integer (iin), dimension (n) :: x, y
  integer (iin), dimension (p) :: a, b
  integer (iin), dimension (p) :: i, j
!
  write (*,*)
  write (*,*) "Matrices ralas: formato (i,j,a_ij) "
  write (*,*) "producto y=Ax con scatter emulado"
!
! caso n = 4, p = 7
  i = [ 1, 2, 2, 3, 3, 4, 4 ]
  j = [ 2, 1, 3, 1, 3, 2, 4 ]
  b = [ 5, 6, 7, 8, 9, 1, 2 ]
  x = [ 4, 5, 6, 7 ]
!
! caso n = 7, p = 17
! i = [ 1, 4, 6, 2, 3, 1, 3, 5, 4, 2, 5, 7, 3, 6, 7, 5, 7 ]
! j = [ 1, 1, 1, 2, 2, 3, 3, 3, 4, 5, 5, 5, 6, 6, 6, 7, 7 ]
! b = [ 1, 2, 3, 4, 5, 6, 7, 8, 9, 1, 2, 3, 4, 5, 6, 7, 8 ]
! x = [ 1, 2, 3, 4, 5, 6, 7 ]
  a = b * x (j)
! y (1:n) = scatter_add ( a (1:p), y (1:n), i (1:p) )
  call scatter_add ( y (1:n), a (1:p), i (1:p) )
  write (*,*)
  write (*,100) " i          : ", i
  write (*,100) " j          : ", j
  write (*,100) " a          : ", a
  write (*,100) " b          : ", b
  write (*,*)
  write (*,100) " x          : ", x
  write (*,100) " a * x(j)   : ", x (j)
  write (*,100) " a * x(j)   : ", b
  write (*,100) " y          : ", y
  write (*,*)
  100 format (a, 10 (1x,i4))
end program
! -----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

Jun 28 2006 16:34

rala_csa.hpf

Page 1

```

! -----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
! Matrices ralas: formato (i, j, a_ij).
! Producto y=Ax con scatter HPF
! -----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
! INTEL:
! ifort -vms -c m_ctes.f90
! ifort -vms -c m_temps.f90
! ifort -vms -c rala_csa.hpf
! ifort -vms -o rala_csa.exe1 *.o
!
! GNU:
! g95 -w -c m_ctes.f90 --free-form
! g95 -w -c m_tools.f90 --free-form
! g95 -w -c rala_csa.hpf --free-form
! g95 -w -o rala_csa.exe2 *.o
!
! ADAPTOR (HPF)
! compilation for Distributed Memory Machines using MPI:
! adaptor -hpf -dm -free -c m_ctes.f90
! adaptor -hpf -dm -free -c m_tools.f90
! adaptor -hpf -dm -free -c rala_csa.hpf
! adaptor -hpf -dm -o rala_csa.exe3 *.o
!
! Running with MPICH:
! mpdboot -n 4 -f ~/mpd.hosts
! mpdtrace
! mpdringtest
! mpiexec -machinefile ~/machi.dat -np 2 rala_csa.exe3
! mpiexec -machinefile ~/machi.dat -n 4 rala_csa.exe3
! mpdallexit
!
! -----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
program rala_csa
use hpf_library
implicit none
integer, parameter :: n = 4, p = 7
integer, dimension (n) :: x, y
integer, dimension (p) :: a, b
integer, dimension (p) :: i, j
!hpf$ distribute (block) :: a
!hpf$ align (:) with a (:) :: b
!hpf$ align (:) with a (:) :: i, j
!hpf$ align (:) with a (:) :: x, y
!
write (*,*)
write (*,*) "Matrices ralas: formato (i,j,a_ij) "
write (*,*) "producto y=Ax con scatter HPF"
!
! caso n = 4, p = 7
i = [ 1, 2, 2, 3, 3, 4, 4 ]
j = [ 2, 1, 3, 1, 3, 2, 4 ]
a = [ 5, 6, 7, 8, 9, 1, 2 ]
x = [ 4, 5, 6, 7 ]
!
! caso n = 7, p = 17
i = [ 1, 4, 6, 2, 3, 1, 3, 5, 4, 2, 5, 7, 3, 6, 7, 5, 7 ]
j = [ 1, 1, 1, 2, 2, 3, 3, 3, 4, 5, 5, 5, 6, 6, 6, 7, 7 ]
a = [ 1, 2, 3, 4, 5, 6, 7, 8, 9, 1, 2, 3, 4, 5, 6, 7, 8 ]
x = [ 1, 2, 3, 4, 5, 6, 7 ]
b = a * x (j)
y (1:n) = sum_scatter (b (1:p), y (1:n), i (1:p) )
write (*,*)
write (*,100) " i          : ", i
write (*,100) " j          : ", j
write (*,100) " a          : ", a
write (*,100) " b          : ", b
write (*,*)
write (*,100) " x          : ", x
write (*,100) "      x(j)   : ", x (j)

```

Jun 28 2006 16:34

rala_csa.hpf

Page 2

```

write (*,100) " a * x(j)   : ", b
write (*,100) " y          : ", y
write (*,*)
100 format (a, 10 (1x,i4))
end program
! -----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

Jun 28 2006 16:35

rala_csc.hpf

Page 1

```

! -----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
! Matrices ralas: formato CSC (Compressed Sparse Column)
! Producto y=Ax con scatter HPF
! -----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
! INTEL:
! ifort -vms -c m_ctes.f90
! ifort -vms -c m_temps.f90
! ifort -vms -c rala_csc.hpf
! ifort -vms -o rala_csc.exe1 *.o
!
! GNU:
! g95 -w -c m_ctes.f90 --free-form
! g95 -w -c m_tools.f90 --free-form
! g95 -w -c rala_csc.hpf --free-form
! g95 -w -o rala_csc.exe2 *.o
!
! ADAPTOR (HPF)
! compilation for Distributed Memory Machines using MPI:
! adaptor -hpf -dm -free -c m_ctes.f90
! adaptor -hpf -dm -free -c m_tools.f90
! adaptor -hpf -dm -free -c rala_csc.hpf
! adaptor -hpf -dm -o rala_csc.exe3 *.o
!
! Running with MPICH:
! mpdboot -n 4 -f ~/mpd.hosts
! mpdtrace
! mpdringtest
! mpiexec -machinefile ~/machi.dat -np 2 rala_csc.exe3
! mpiexec -machinefile ~/machi.dat -n 4 rala_csc.exe3
! mpdallexit
!
! -----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
program rala_csc
  use hpf_library
  implicit none
  integer, parameter :: n = 4, p = 7
  integer, dimension (n) :: x, y
  integer, dimension (n+1) :: j
  integer, dimension (p) :: i
  integer, dimension (p) :: a, b, c, d
  logical, dimension (p) :: s
  !hpfs distribute (block) :: a
  !hpfs align (:) with a (:) :: b, c, d
  !hpfs align (:) with a (:) :: i, s
  !hpfs align (:) with a (:) :: x, y
  integer :: k, l
  logical :: tira
  !
  write (*,*)
  write (*,*) "Matrices ralas: formato CSC (Compressed Sparse Column)"
  write (*,*) "producto y=Ax con scatter HPF"
  !
  ! caso n = 4, p = 7
  j = [ 1, 3, 5, 7, 8 ]
  i = [ 2, 3, 1, 4, 2, 3, 4 ]
  a = [ 6, 8, 5, 1, 7, 9, 2 ]
  x = [ 4, 5, 6, 7 ]
  !
  ! caso n = 7, p = 17
  j = [ 1, 4, 6, 9, 10, 13, 16, 18 ]
  i = [ 1, 4, 6, 2, 3, 1, 3, 5, 4, 2, 5, 7, 3, 6, 7, 5, 7 ]
  a = [ 1, 2, 3, 4, 5, 6, 7, 8, 9, 1, 2, 3, 4, 5, 6, 7, 8 ]
  x = [ 1, 2, 3, 4, 5, 6, 7 ]
  !
  ! construye segmentos booleanos para identificar cada columna
  tira = .true.
  do k = 1, n
    do l = j (k), j (k + 1) - 1
      s (l) = tira
    end do
  end do

```

Jun 28 2006 16:35

rala_csc.hpf

Page 2

```

  end do
  tira = .not. tira
end do
write (*,*)
write (*,100) " i           : ", i
write (*,100) " j           : ", j
write (*,100) " a           : ", a
write (*,110) " s           : ", s
!
b (j (1:n)) = x
c = copy_prefix (b, segment = s)
d = a * c
y = 0
y (1:n) = sum_scatter (d (1:p), y (1:n), i (1:p) )
write (*,*)
write (*,100) " b(j(1:n)) : ", b
write (*,100) " copy_prefix: ", c
write (*,100) " a * b       : ", d
write (*,100) " x           : ", x
write (*,100) " y           : ", y
write (*,*)
100 format (a, 20 (1x,i4))
110 format (a, 20 (1x,l4))
end program
! -----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

Jun 28 2006 16:37

rala_csr.hpf

Page 1

```

! -----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
! Matrices ralas: formato CSR (Compressed Sparse Row)
! Producto y=Ax con scatter HPF
! -----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
! INTEL:
! ifort -vms -c m_ctes.f90
! ifort -vms -c m_temps.f90
! ifort -vms -c rala_csr.hpf
! ifort -vms -o rala_csr.exe1 *.o
!
! GNU:
! g95 -w -c m_ctes.f90 --free-form
! g95 -w -c m_tools.f90 --free-form
! g95 -w -c rala_csr.hpf --free-form
! g95 -w -o rala_csr.exe2 *.o
!
! ADAPTOR (HPF)
! compilation for Distributed Memory Machines using MPI:
! adaptor -hpf -dm -free -c m_ctes.f90
! adaptor -hpf -dm -free -c m_tools.f90
! adaptor -hpf -dm -free -c rala_csr.hpf
! adaptor -hpf -dm -o rala_csr.exe3 *.o
!
! Running with MPICH:
! mpdboot -n 4 -f ~/mpd.hosts
! mpdtrace
! mpdringtest
! mpiexec -machinefile ~/machi.dat -np 2 rala_csr.exe3
! mpiexec -machinefile ~/machi.dat -n 4 rala_csr.exe3
! mpdallexit
!
! -----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
program rala_csr
use hpf_library
implicit none
integer, parameter :: n = 4, p = 7
integer, dimension (n) :: x, y
integer, dimension (n+1) :: i
integer, dimension (p) :: j
integer, dimension (p) :: a, b, c, d
logical, dimension (p) :: s
!hpf$ distribute (block) :: a
!hpf$ align (:) with a (:) :: b, c, d
!hpf$ align (:) with a (:) :: i, s
!hpf$ align (:) with a (:) :: x, y
integer :: k, l
logical :: tira
!
write (*,*)
write (*,*) "Matrices ralas: formato CSR (Compressed Sparse Row)"
write (*,*) "producto y=Ax con scatter HPF"
!
! caso n = 4, p = 7
i = [ 1, 2, 4, 6, 8 ]
j = [ 2, 1, 3, 1, 3, 2, 4 ]
a = [ 5, 6, 7, 8, 9, 1, 2 ]
x = [ 4, 5, 6, 7 ]
!
! caso n = 7, p = 17
i = [ 1, 3, 5, 8, 10, 13, 15, 18 ]
j = [ 1, 3, 2, 5, 2, 3, 6, 1, 4, 3, 5, 7, 1, 6, 5, 6, 7 ]
a = [ 1, 6, 4, 1, 5, 7, 4, 2, 9, 8, 2, 7, 3, 5, 3, 6, 8 ]
x = [ 1, 2, 3, 4, 5, 6, 7 ]
!
! construye segmentos booleanos para identificar cada fila
tira = .true.
do k = 1, n
do l = i (k), i (k + 1) - 1
s (l) = tira

```

Jun 28 2006 16:37

rala_csr.hpf

Page 2

```

end do
tira = .not. tira
end do
write (*,*)
write (*,100) " i           : ", i
write (*,100) " j           : ", j
write (*,100) " a           : ", a
write (*,110) " s           : ", s
!
b = x (j (1:p))
c = a * b
d (1:p) = sum_suffix (c (1:p), segment = s)
y = d (i (1:n))
write (*,*)
write (*,100) " x(j(1:p)) : ", b
write (*,100) " a * b      : ", c
write (*,100) " sum_suffix : ", d
write (*,100) " x          : ", x
write (*,100) " y          : ", y
write (*,*)
100 format (a, 20 (1x,i4))
110 format (a, 20 (1x,i4))
end program
! -----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```