高性能数值计算大作业

苏智龙　2018Z8017761055

（1）整个区域可以分为 $N_x * N_y$ 个单元，一共有（N_x+1）*（N_y+1）个点需要计算，令 h_x=1/n，h_y=1/m 为步长。采用均匀网格做离散得：

$$\frac{2u_{i,j}-u_{i+1,j}-u_{i-1,j}}{h_x^2}+\frac{2u_{i,j}-u_{i,j+1}-u_{i,j-1}}{h_y^2}-\frac{\omega_{i,j+1}-\omega_{i,j-1}}{2h_y}=0$$

$$\frac{2v_{i,j}-v_{i+1,j}-v_{i-1,j}}{h_x^2}+\frac{2v_{i,j}-v_{i,j+1}-v_{i,j-1}}{h_y^2}-\frac{\omega_{i+1,j}-\omega_{i-1,j}}{2h_x}=0$$

$$\frac{2\omega_{i,j}-\omega_{i+1,j}-\omega_{i-1,j}}{h_x^2}+\frac{2\omega_{i,j}-\omega_{i,j+1}-\omega_{i,j-1}}{h_y^2}+DIW(\omega)-G_r\frac{T_{i+1,j}-T_{i-1,j}}{2h_x}=0$$

$$\frac{2T_{i,j}-T_{i+1,j}-T_{i-1,j}}{h_x^2}+\frac{2T_{i,j}-T_{i,j+1}-T_{i,j-1}}{h_y^2}+P_r*DIV(T)=0$$

$$DIV(C)=u_{i,j}*\frac{(C_{i+1,j}-C_{i-1,j})}{2h_x}+v_{i,j}*\frac{(C_{i,j+1}-C_{i,j-1})}{2h_y}$$

$$C=\omega，T$$

$$h_x=\frac{1}{n}，\ h_y=\frac{1}{m}\ \text{I, j=0,1, …,n;}$$

$$n=N_x$$

$$m=N_y$$

X=（u，v，ω，T），写成向量的形式可得：$F(X)=0$

（2）编译程序 ex19.c：

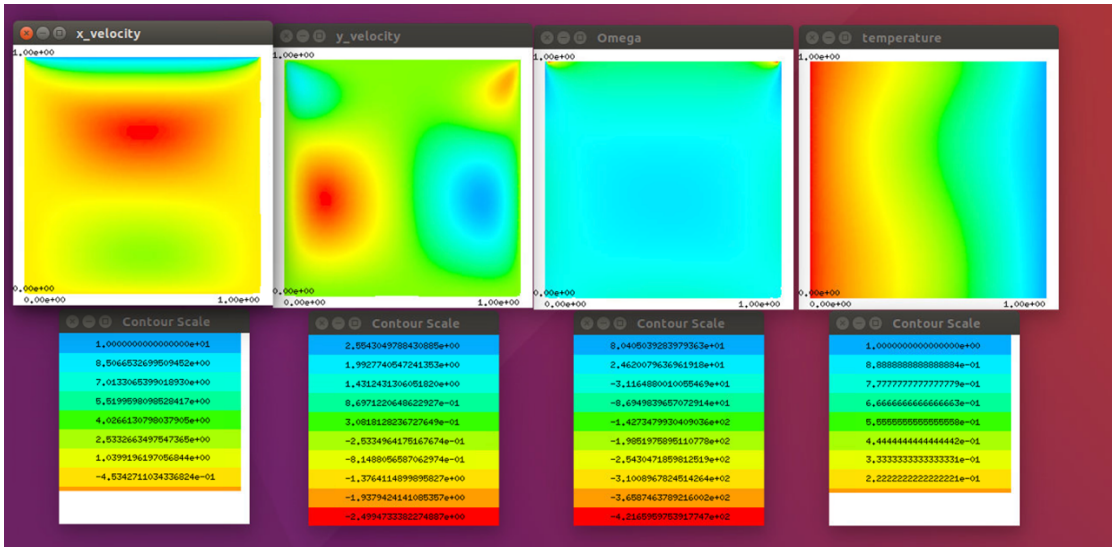
```
make PETSC_DIR=~/code/petsc ex19
```

运行程序：

```
mpirun -np 4 ./ex19 -da_grid_x 64 -da_grid_y 64 -snes_monitor -
snes_converged_reason -lidvelocity 10. -prandtl 1. -grashof 1000. -contours 1 -
draw_pause -2
```

运行结果及图像如下：

```
su@ssss:~/code/petsc/src/snes/examples/tutorials$ mpirun -np 4 ./ex19 -da_grid_x 64 -da_grid_y 64 -snes_monitor -snes_converged_reason -lidvelocity 10. -prandtl 1. -grashof 1000. -contours 1
lid velocity = 10., prandtl # = 1., grashof # = 1000.
lid velocity = 10., prandtl # = 1., grashof # = 1000.
lid velocity = 10., prandtl # = 1., grashof # = 1000.
  0 SNES Function norm 8.027463868964e+01
  0 SNES Function norm 8.027463868964e+01
lid velocity = 10., prandtl # = 1., grashof # = 1000.
  0 SNES Function norm 8.027463868964e+01
  0 SNES Function norm 8.027463868964e+01
  1 SNES Function norm 1.788396671509e+01
  1 SNES Function norm 1.788396671509e+01
  1 SNES Function norm 1.788396671509e+01
  1 SNES Function norm 1.788396671509e+01
  2 SNES Function norm 4.743612590174e-03
  2 SNES Function norm 4.743612590174e-03
  2 SNES Function norm 4.743612590174e-03
  2 SNES Function norm 4.743612590174e-03
  3 SNES Function norm 1.595522140204e-06
  3 SNES Function norm 1.595522140204e-06
  3 SNES Function norm 1.595522140204e-06
  3 SNES Function norm 1.595522140204e-06
  4 SNES Function norm 5.064026281545e-12
Nonlinear solve converged due to CONVERGED_FNORM_RELATIVE iterations 4
Number of SNES iterations = 4
  4 SNES Function norm 5.064026281545e-12
Nonlinear solve converged due to CONVERGED_FNORM_RELATIVE iterations 4
Number of SNES iterations = 4
  4 SNES Function norm 5.064026281545e-12
Nonlinear solve converged due to CONVERGED_FNORM_RELATIVE iterations 4
Number of SNES iterations = 4
  4 SNES Function norm 5.064026281545e-12
Nonlinear solve converged due to CONVERGED_FNORM_RELATIVE iterations 4
Number of SNES iterations = 4
```



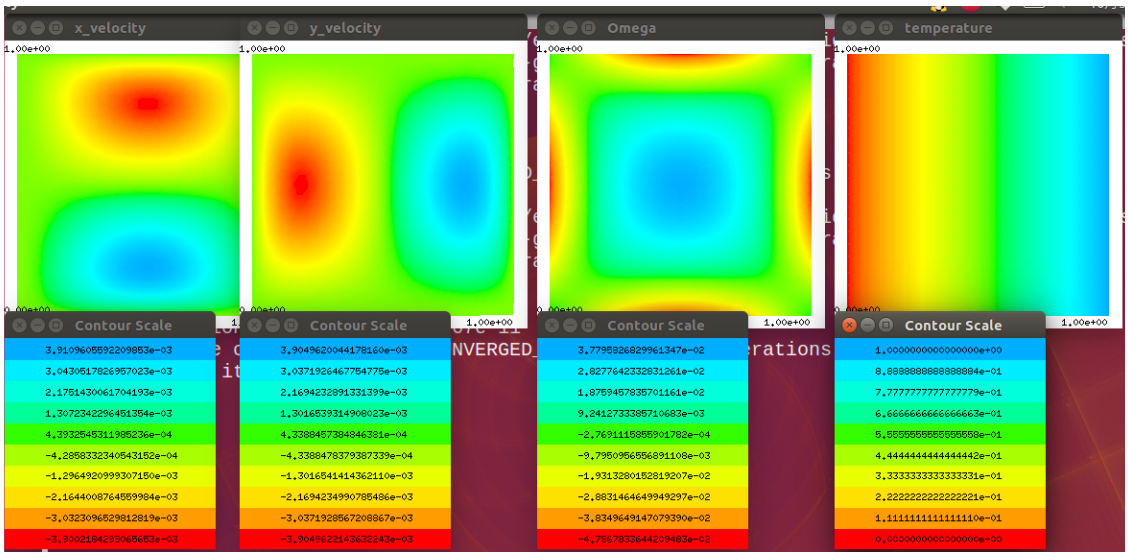
(3) 基于 64 × 64 的网络，通过 -lidvelocity，-prandtl 和-grashof 三个选项改变参数的值，研究三个无量纲参数对于计算的影响。建议选择 lid velocity V_lid=10^-4, 0.1, 10;Pr=0.1, 1, 10;Gr=1, 100, 1000，选定参数后，配合下面的选项运行程序

3.1. lid velocity = 0.0001, prandtl = 0.1, grashof = 1

```
mpirun -np 4 ./ex19 -da_grid_x 64 -da_grid_y 64 -snes_monitor -snes_converged_reason -lidvelocity 0.0001 -prandtl 0.1 -grashof 1 -contours 1 -draw_pause -2
```

运行结果及图像如下：

迭代次数为 2 次



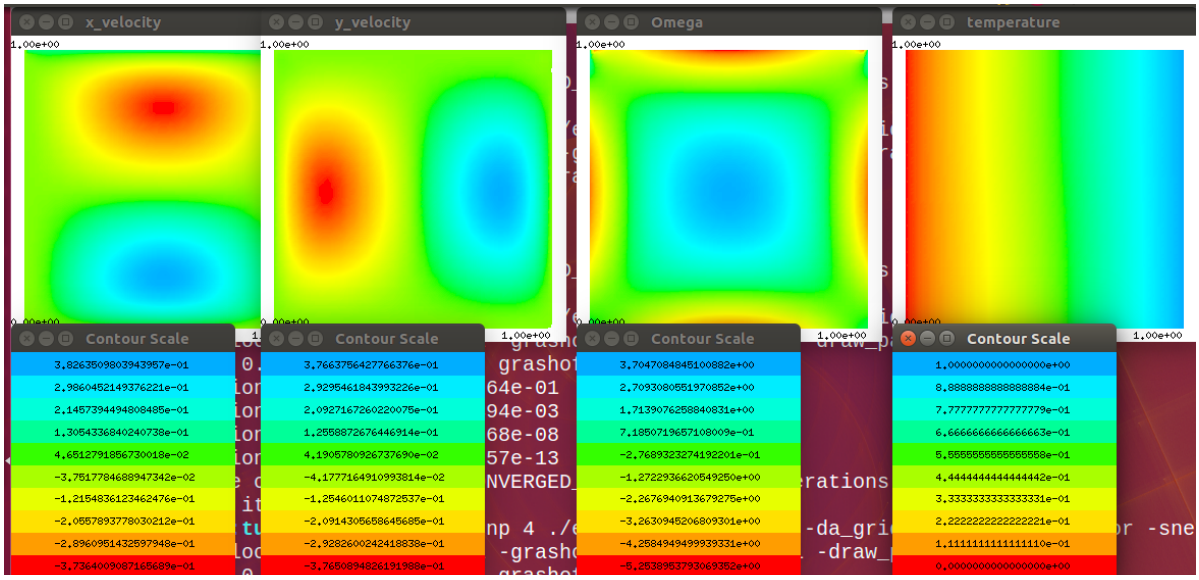
3.2. lid velocity = 0.1, prandtl = 1, grashof = 100

```
mpirun -np 4 ./ex19 -da_grid_x 64 -da_grid_y 64 -snes_monitor -snes_converged_reason -lidvelocity 0.1 -prandtl 1. -grashof 100. -contours 1 -draw_pause -2
```

运行结果及图像如下：

```
su@ssss:~/code/petsc/src/snes/examples/tutorials$ mpirun -np 4 ./ex19 -da_grid_x 64 -da_grid_y 64 -snes_monitor -snes_converged_reason -lidvelocity 0.1 -prandtl 1 -grashof 100 -contours 1 -draw_pause -1
lid velocity = 0.1, prandtl # = 1., grashof # = 100.
lid velocity = 0.1, prandtl # = 1., grashof # = 100.
  0 SNES Function norm 1.749335921862e+00
lid velocity = 0.1, prandtl # = 1., grashof # = 100.
  0 SNES Function norm 1.749335921862e+00
lid velocity = 0.1, prandtl # = 1., grashof # = 100.
  0 SNES Function norm 1.749335921862e+00
  0 SNES Function norm 1.749335921862e+00
  1 SNES Function norm 1.299668799759e-02
  1 SNES Function norm 1.299668799759e-02
  1 SNES Function norm 1.299668799759e-02
  1 SNES Function norm 1.299668799759e-02
  2 SNES Function norm 2.884466564615e-07
  2 SNES Function norm 2.884466564615e-07
  2 SNES Function norm 2.884466564615e-07
  2 SNES Function norm 2.884466564615e-07
  3 SNES Function norm 2.596772629691e-12
Nonlinear solve converged due to CONVERGED_FNORM_RELATIVE iterations 3
Number of SNES iterations = 3
  3 SNES Function norm 2.596772629691e-12
Nonlinear solve converged due to CONVERGED_FNORM_RELATIVE iterations 3
Number of SNES iterations = 3
  3 SNES Function norm 2.596772629691e-12
Nonlinear solve converged due to CONVERGED_FNORM_RELATIVE iterations 3
Number of SNES iterations = 3
  3 SNES Function norm 2.596772629691e-12
Nonlinear solve converged due to CONVERGED_FNORM_RELATIVE iterations 3
Number of SNES iterations = 3
```

迭代次数为 3 次



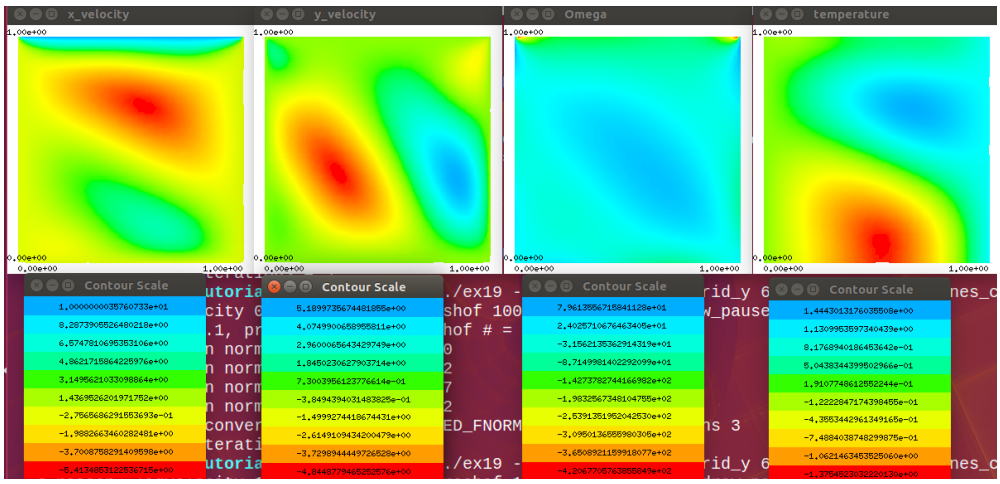
3.3. lid velocity = 10, prandtl = 10, grashof = 1000

```
mpirun -np 4 ./ex19 -da_grid_x 64 -da_grid_y 64 -snes_monitor -snes_converged_reason -lidvelocity 10. -prandtl 10. -grashof 1000. -contours 1 -draw_pause -2
```

运行结果及图像如下：

```
su@ssss:~/code/petsc/src/snes/examples/tutorials$ mpirun -np 4 ./ex19 -da_grid_x 64 -da_grid_y 64 -snes_monitor -snes_converged_reason -lidvelocity 10. -prandtl 10. -grashof 1000. -contours 1 -draw_pause
-1
lid velocity = 10., prandtl # = 10., grashof # = 1000.
lid velocity = 10., prandtl # = 10., grashof # = 1000.
  0 SNES Function norm 8.027463868964e+01
lid velocity = 10., prandtl # = 10., grashof # = 1000.
lid velocity = 10., prandtl # = 10., grashof # = 1000.
  0 SNES Function norm 8.027463868964e+01
  0 SNES Function norm 8.027463868964e+01
  0 SNES Function norm 8.027463868964e+01
  1 SNES Function norm 1.823282308169e+01
  1 SNES Function norm 1.823282308169e+01
  1 SNES Function norm 1.823282308169e+01
  1 SNES Function norm 1.823282308169e+01
  2 SNES Function norm 9.306307526623e-01
  2 SNES Function norm 9.306307526623e-01
  2 SNES Function norm 9.306307526623e-01
  2 SNES Function norm 9.306307526623e-01
  3 SNES Function norm 2.481616555550e-01
  3 SNES Function norm 2.481616555550e-01
  3 SNES Function norm 2.481616555550e-01
Nonlinear solve did not converge due to DIVERGED_LINEAR_SOLVE iterations 3
Number of SNES iterations = 3
Nonlinear solve did not converge due to DIVERGED_LINEAR_SOLVE iterations 3
Number of SNES iterations = 3
Nonlinear solve did not converge due to DIVERGED_LINEAR_SOLVE iterations 3
Number of SNES iterations = 3
Nonlinear solve did not converge due to DIVERGED_LINEAR_SOLVE iterations 3
Number of SNES iterations = 3
```

迭代次数为 3 次



（4）

4.1 需要修改的部分：

4.1.1 结构体需要增加 semp、arandtl、brashof，如下图：

```
/*
   User-defined routines and data structures
*/
typedef struct {
  PetscScalar u,v,omega,temp,semp;
} Field;

PetscErrorCode FormFunctionLocal(DMDALocalInfo*,Field**,Field**,void*);

typedef struct {
  PetscReal   lidvelocity,prandtl,arandtl,grashof,brashof;  /* physical parameters */
  PetscBool   draw_contours;                /* flag - 1 indicates drawing contours */
} AppCtx;
```

4.1.2 参数 4,1,0,0 改为 5,1,0,0，给 user.arandtl、user.brashof 赋值

```
/*
     Create distributed array object to manage parallel grid and vectors
     for principal unknowns (x) and governing residuals (f)
*/
ierr = DMDACreate2d(PETSC_COMM_WORLD,
                    DM_BOUNDARY_NONE,
                    DM_BOUNDARY_NONE,
                    DMDA_STENCIL_STAR,4,4,
                    PETSC_DECIDE,PETSC_DECIDE,
                    5,1,0,0,
                    &da);
                    CHKERRQ(ierr);
```

```
/*
   Problem parameters (velocity of lid, prandtl, and grashof numbers)
*/
user.lidvelocity = 1.0/(mx*my);
user.prandtl     = 1.0;
user.grashof     = 1.0;
user.arandtl     = 1.0;
user.brashof     = 1.0;


ierr = PetscOptionsGetReal(NULL,NULL,"-lidvelocity",&user.lidvelocity,NULL);CHKERRQ(ierr);
ierr = PetscOptionsGetReal(NULL,NULL,"-prandtl",&user.prandtl,NULL);CHKERRQ(ierr);
ierr = PetscOptionsGetReal(NULL,NULL,"-grashof",&user.grashof,NULL);CHKERRQ(ierr);
ierr = PetscOptionsGetReal(NULL,NULL,"-arandtl",&user.arandtl,NULL);CHKERRQ(ierr);
ierr = PetscOptionsGetReal(NULL,NULL,"-brashof",&user.brashof,NULL);CHKERRQ(ierr);
ierr = PetscOptionsHasName(NULL,NULL,"-contours",&user.draw_contours);CHKERRQ(ierr);


ierr = DMDASetFieldName(da,0,"x_velocity");CHKERRQ(ierr);
ierr = DMDASetFieldName(da,1,"y_velocity");CHKERRQ(ierr);
ierr = DMDASetFieldName(da,2,"Omega");CHKERRQ(ierr);
ierr = DMDASetFieldName(da,3,"temperature");CHKERRQ(ierr);
ierr = DMDASetFieldName(da,4,"semperature");CHKERRQ(ierr);
```

4.1.3 修改边界

```
if (yints == 0) {
  j     = 0;
  yints = yints + 1;
  /* bottom edge */
  for (i=info->xs; i<info->xs+info->xm; i++) {
    f[j][i].u     = x[j][i].u;
    f[j][i].v     = x[j][i].v;
    f[j][i].omega = x[j][i].omega + (x[j+1][i].u - x[j][i].u)*dhy;
    f[j][i].temp  = x[j][i].temp-x[j+1][i].temp;
    f[j][i].semp  = x[j][i].semp;
  }
}

/* Test whether we are on the top edge of the global array */
if (yinte == info->my) {
  j     = info->my - 1;
  yinte = yinte - 1;
  /* top edge */
  for (i=info->xs; i<info->xs+info->xm; i++) {
    f[j][i].u     = x[j][i].u - lid;
    f[j][i].v     = x[j][i].v;
    f[j][i].omega = x[j][i].omega + (x[j][i].u - x[j-1][i].u)*dhy;
    f[j][i].temp  = x[j][i].temp-x[j-1][i].temp;
    f[j][i].semp  = x[j][i].semp + 1;
  }
}

/* Test whether we are on the left edge of the global array */
if (xints == 0) {
  i     = 0;
  xints = xints + 1;
  /* left edge */
  for (j=info->ys; j<info->ys+info->ym; j++) {
    f[j][i].u     = x[j][i].u;
    f[j][i].v     = x[j][i].v;
    f[j][i].omega = x[j][i].omega - (x[j][i+1].v - x[j][i].v)*dhx;
    f[j][i].temp  = x[j][i].temp;
    f[j][i].semp  = x[j][i].semp - x[j][i+1].semp;
  }
}

/* Test whether we are on the right edge of the global array */
if (xinte == info->mx) {
  i     = info->mx - 1;
  xinte = xinte - 1;
  /* right edge */
  for (j=info->ys; j<info->ys+info->ym; j++) {
    f[j][i].u     = x[j][i].u;
    f[j][i].v     = x[j][i].v;
    f[j][i].omega = x[j][i].omega - (x[j][i].v - x[j][i-1].v)*dhx;
    f[j][i].temp  = x[j][i].temp - (PetscReal)(grashof>0);
    f[j][i].semp  = x[j][i].semp - x[j][i-1].semp;
  }
}
```

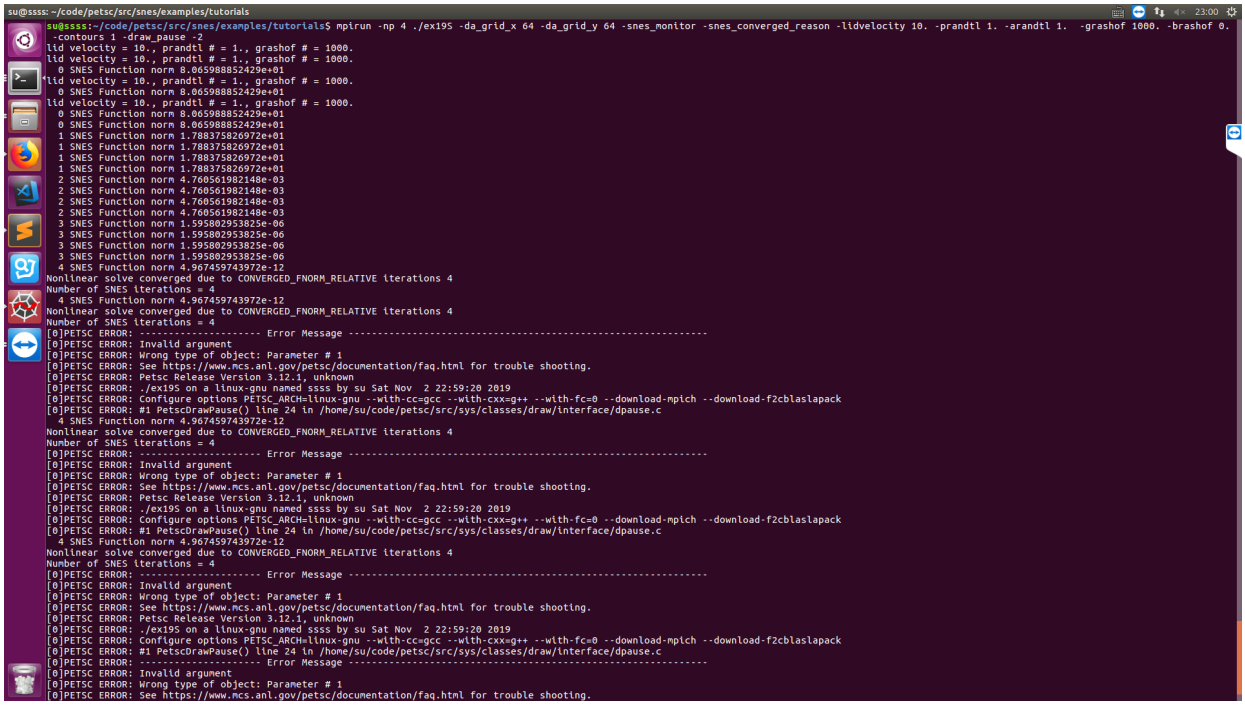
4.1.4 增加 Semperature

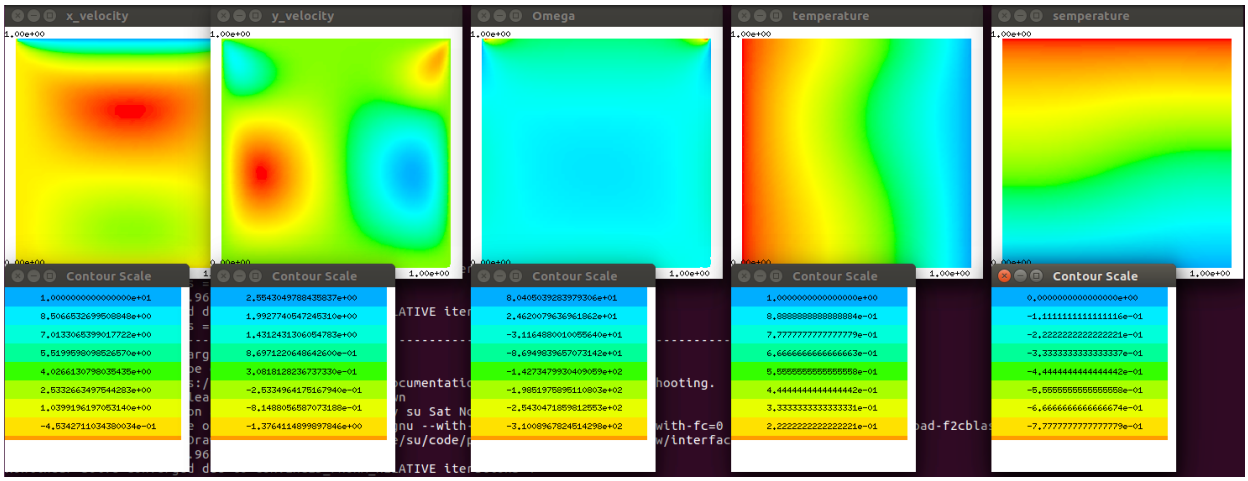
```
/* Omega */
u             = x[j][i].omega;
uxx           = (2.0*u - x[j][i-1].omega - x[j][i+1].omega)*hydhx;
uyy           = (2.0*u - x[j-1][i].omega - x[j+1][i].omega)*hxdhy;
f[j][i].omega = uxx + uyy + (vxp*(u - x[j][i-1].omega) + vxm*(x[j][i+1].omega - u))*hy +
                (vyp*(u - x[j-1][i].omega) + vym*(x[j+1][i].omega - u))*hx -
                .5*grashof*(x[j][i+1].temp - x[j][i-1].temp)*hy;

/* Temperature */
u             = x[j][i].temp;
uxx           = (2.0*u - x[j][i-1].temp - x[j][i+1].temp)*hydhx;
uyy           = (2.0*u - x[j-1][i].temp - x[j+1][i].temp)*hxdhy;
f[j][i].temp =  uxx + uyy  + prandtl*((vxp*(u - x[j][i-1].temp) + vxm*(x[j][i+1].temp - u))*hy +
                                      (vyp*(u - x[j-1][i].temp) + vym*(x[j+1][i].temp - u))*hx);

/* Semperature */
u             = x[j][i].semp;
uxx           = (2.0*u - x[j][i-1].semp - x[j][i+1].semp)*hydhx;
uyy           = (2.0*u - x[j-1][i].semp - x[j+1][i].semp)*hxdhy;
f[j][i].semp =  uxx + uyy  + arandtl*((vxp*(u - x[j][i-1].semp) + vxm*(x[j][i+1].semp - u))*hy +
                                      (vyp*(u - x[j-1][i].semp) + vym*(x[j+1][i].semp - u))*hx);
```

4.2 比较和结论

4.2.1 A(arandtl)=1 B(brashof)=0

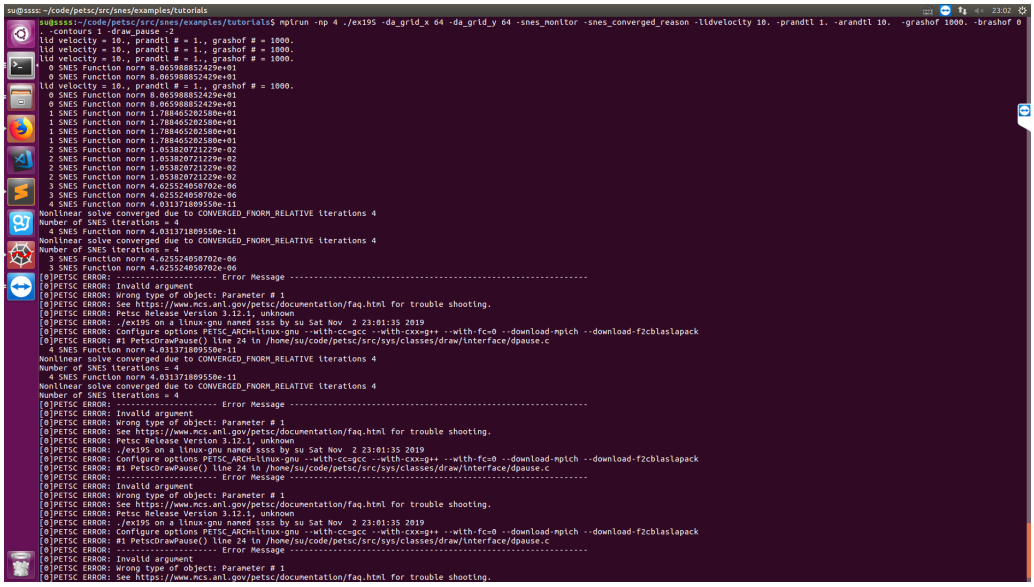
```
mpirun -np 4 ./ex19S -da_grid_x 64 -da_grid_y 64 -snes_monitor -snes_converged_reason
-lidvelocity 10. -prandtl 1. -arandtl 1. -grashof 1000. -brashof 0. -contours 1 -
draw_pause -2
```

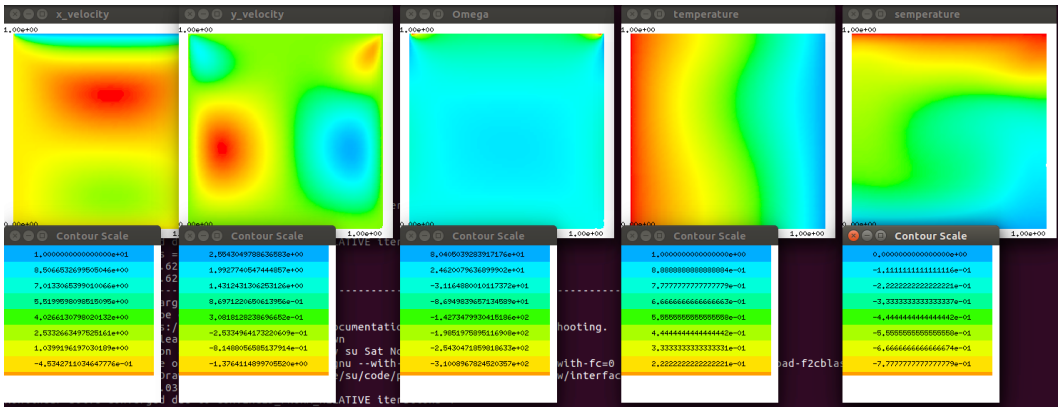




4.2.2 A(arandtl)=10 B(brashof)=0

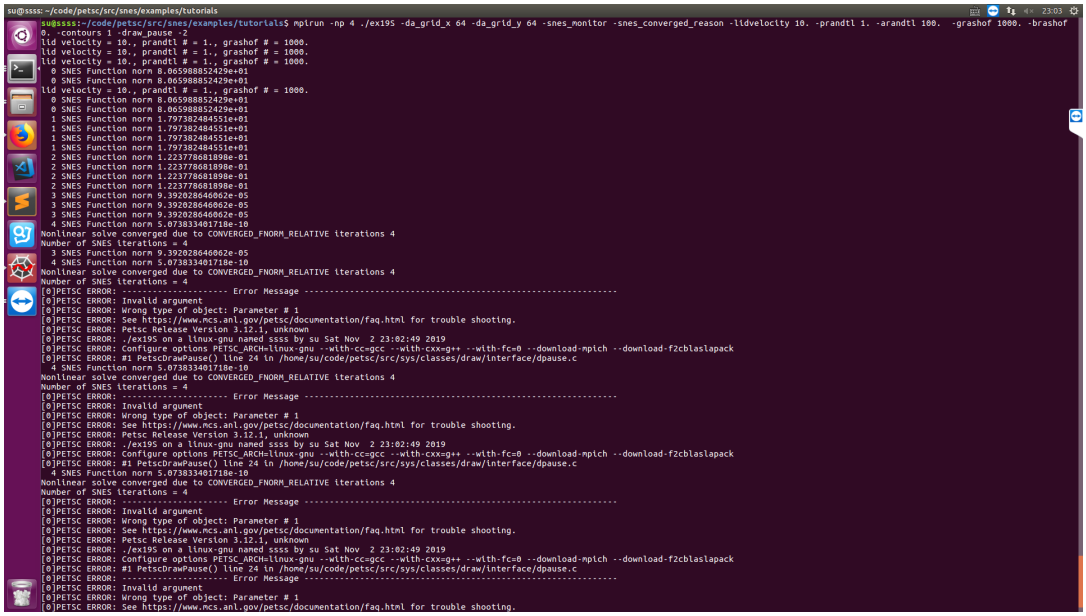
```
mpirun -np 4 ./ex19S -da_grid_x 64 -da_grid_y 64 -snes_monitor -snes_converged_reason
-lidvelocity 10. -prandtl 1. -arandtl 10. -grashof 1000. -brashof 0. -contours 1 -
draw_pause -2
```





4.2.3 A(arandtl)=100 B(brashof)=0

```
mpirun -np 4 ./ex19S -da_grid_x 64 -da_grid_y 64 -snes_monitor -snes_converged_reason -lidvelocity 10. -prandtl 1. -arandtl 100. -grashof 1000. -brashof 0. -contours 1 -draw_pause -2
```



比较以上 A=1，10，100 的 S 的图像可得，S 图像发生了变化。