# 1、运行suntans模型

应用triangule程序生成三角网格：

mpiexec sun -t -g -vvv --datadir=./rundata

网格分区命令：

mpiexec -n 4 sun -g -vvv --datadir=./rundata

-v 显示信息，最多4个v

运行模型：

mpiexec -n 4 sun -s -vvv --datadir=./rundata

-s 运行suntans

-w 输出可能导致计算错误的警告，依赖于-v

-r 重启suntans（热启动）

# 2、创建或读取三角网格

## 2.1使用triangle库

Triangle库从简单的planar straight line graph (pslg)生成三角网格。

# 3、设置初始条件和边界条件

通过编辑initialization.c和boundaries.c文件设置初始条件和边界条件。

## 3.1 初始条件

有5个函数定义初始水深、自由水位、盐度、温度和流速场。需要编辑函数，基于xyz坐标返回希望设置的初始条件。

（1）初始水深

suntans.dat中，IntDepth==0，使用程序中的函数；否则，IntDepth=1, 由depth文件插值得到（3列数），depth为负数，elevation为整数。

（3）初始盐度和初始温度

beta=0: 不计算盐度输移；gamma=0: 不计算温度输移

## 3.2 边界条件

流速场的边界条件可以在pslg文件中设置标记，标记为：

1-封闭边界；2-开边界或指定流速；3-开边界指定水位；4-no slip边界

## 3.3 设置热通量

设置温度方程中的源项，需要编辑sources.c中的HeatSource()函数。suntans模型假设水面和河床均绝热，所有热通量以源项在温度方程中的形式如下：



# 4、热启动

suntans输出热启动文件在StoreFile定义

# 5、运行例子

# 附录1 SUNTANS代码结构

主程序：

main(){

StartMpi() // 启动MPI

ParseFlags() // 读取命令参数 -t -g -s -vvv

if(GRID){

GetGrid() // -t 使用Triangle库生成三角网格

}else{

ReadGrid() //读入Triangle或Gambit生成的三角网格

}

if(SOLVE){ // -s

ReadProperties() / / 从suntans.dat读取参数到求解器

InitializeVerticalGrid //初始化垂向分层，dzf, dzz, dzzold

AllocatePhysicalVariables() //分配物理变量的内存空间

AllocateTransferArrays() //分配MPI通信的数组内存空间

OpenFiles() // 打开计算结果输出文件 fopen

if(RESTART){ // -r 热启动

ReadPhysicalVariables() // 读取热启动的中间结果文件

}else{

InitializePhysicalVariables() //初始化物理变量的值

Solve() //核心计算程序，包含时间迭代

}

}

EndMpi() //结束MPI

}

子程序：

Solve()

{

ComputeConservatives() // 确保动量和质量守恒

MemoryStats() // -vv 屏幕输出使用的内存量 (Mb)

// 设置边界流速、开边界通量、开边界标量

// 初始化气象条件输出数据文件

//。。。。。。

for(n=nstart;n<nsteps;n++){ // 时间层迭代

StoreVariables() // 存储旧时刻的变量值

SetFluxHeight() // 获取自由水位波动带来的通量值

HorizontalSource() // 计算水平源项（旧时刻的静水压力梯度、Coriolis力

UPredictor() // 计算预测的水平流速（静水压力）

blowup = CheckDZ // 保证垂向分层间距>=0，且当wetdry!=0时自由水位不穿过单元

Continuity() //求解连续方程，Eq.82

EddyViscosity() //计算涡粘性系数

UpdateAge() // if calcage>0, 更新age(passive)标量浓度

if(gamma){ // gamma>0，更新计算温度场

getTsurf() // 获取水面温度

HeatSource() //计算热源

UpdateScalars() //更新计算标量

getchangeT() // 计算水面温度的变化

}

updateAirSeaFluxes() // 更新大气-海面通量

if(beta) { } // 计算盐度，跟温度计算差不多，考虑了大气通量

if(computeSediments) { } // 计算泥沙输移

if(nonhydrostatic && !blowup){ // 计算垂向动量和非静水压力

WPredictor() // 计算预测的垂向流速: phys->w

ComputeQSource() // 计算Poisson压力方程的源项, physc->stmp

CGSolveQ() // 求解非静水压力

Corrector() // 使用非静水压力修正非静水压力场下的速度场

}

Continuity() // 代入垂向流速，检验是否满足连续方程

BoundaryScalars() // 设置新时刻(n), 标量和风应力的边界值

WindStress() // 风应力

SetDragCoefficients() // 设置阻力系数

if(gamma || beta) SetDensity() //如果计算了温度或盐度，就重新计算密度

ComputeUC() // 计算新时刻所有网格边上的流速: uc, vc

if(newcells) NewCells() //考虑水位波动后表层单元的动量变化

if(calcaverage) UpdateAverageVariables(); UpdateAverageScalars()

Check() // 检查程序是否blowup

OutputPhysicalVariables() or WriteOutputNcmerge() // 输出结果(二进制格式或netCDF格式：二进制为合并的结果，nc格式可合并也可单独输出)

if(calcaverage) WriteAverageNCmerge() //平均后的数组输出(只有nc格式)

InterpData() // 先确定各进程上给定输入点的最近点，然后以一定的输出频率(suntans.dat)插值到输出结果的数组上

Progress() // 输出计算进度信息

} //时间层迭代

}

计算泥沙输移和河床演变的模块，被phys.c调用

ComputeSediments()

{

if(n==1+nstart){ // 第一个时间步

sediments = (sedimentsT \*)SunMalloc(); //泥沙结构体

ReadSediProperties(); // 读取泥沙属性数据

OpenSediFiles(); // 打开存储泥沙计算结果文件

} // 泥沙初始化

CalculateErosion(); // 计算侵蚀量

for(k=0;k<size;k++){ // 分组计算悬沙

SedimentSource(); // 计算悬移质泥沙浓度场

SedimentVerticalVelocity(); //计算悬沙垂向运动速度： w + ws

CalculateSediDiffusivity(); // 计算悬沙的扩散系数

UpdateScalars(); //跟其他标量一样，更新浓度值

SedimentVerticalVelocity(); //计算悬沙垂向运动速度：w - ws

}

if(WSconstant==0) SettlingVelocity(); // 计算絮凝条件下的泥沙沉速

if(bedInterval) BedChange(); // 河床演变计算

BoundarySediment() ; // 获取下一时刻悬沙的边界条件值

OutputSediment(); // 输出泥沙计算结果到文件

}

CalculateErosion(){

// 分为soft erosion hard erosion

}

CalculateDeposition(){

// 基于泥沙沉速和悬沙浓度

}

# 附录2 VS2013编译suntans遇到的一些问题及解决

（1）error4703

在VS2012创建项目时，会有一个勾选项，叫做“安全开发生命周期（SDL）检查”，这个东西是微软在VS2012新推出的东西，为了是能更好的监管开发者的代码安全，如果勾选上这一项，那么他将严格按照SDL的规则编译代码，会有一些以前常用的函数无法通过编译，比如在VS2010中的scanf是warning那么在VS2012中就是error了。

查看MSDN中对SDL的描述，SDL检查会视以下告警为错误：

从这些功能来看，有些功能还是不错的，不但方便了程序员使用也增强了程序的健壮性，但是对老版本程序的兼容就会出现问题，以前程序中编译通过的内容也许在VS2012中就没有办法通过。

问题解决方法：

有一个选择打开和关闭SDL检查的位置就是：项目属性->配置属性->C/C++->SDL检查，选测是或者否。

此时我们选择”否”，则上面的那个错误就不会出现了

（2）警告关闭，谨慎使用。

在suntans.h中使用：

#pragma warning(disable:4018 4101 4244) //LJ for VS2013

（3）Visual studio中没有strptime()函数

VS中关于时间的函数time()。使用田开发的代码解决这个问题。

（4）

MSVCRTD.lib(crtexew.obj) : error LNK2019: unresolved external symbol[\_WinMain@16](http://wxiaolei.blog.163.com/blog/static/103876012009101791613792/_WinMain@16) referenced in function \_\_\_tmainCRTStartup

在网上搜了一下，做如下修改：  
Project -> xxxx Properties -> Configuration Properties -> Linker -> System ->Console ，原来是/subsystem:windows，这里改为(/SUBSYSTEM:CONSOLE)。  
修改后， 该问题解决。

（5）在VS2008工程中，经常出现warning LNK4099: 未找到 PDB“vc90.pdb等问题，解决办法如下：

1. 在项目属性—C/C++—输出文件—程序数据库文件名后面填写$(IntDir)\vc80.pdb。

2. 在项目属性—连接器—调试—生成调试信息后面改成“否”。（但是之后就不能进行调试了！）

（6）添加triangle库后，运行中断。

可以使用gambit生成网格，读入，不需要使用triangle库。因此，使用triangulate-notriangle.c