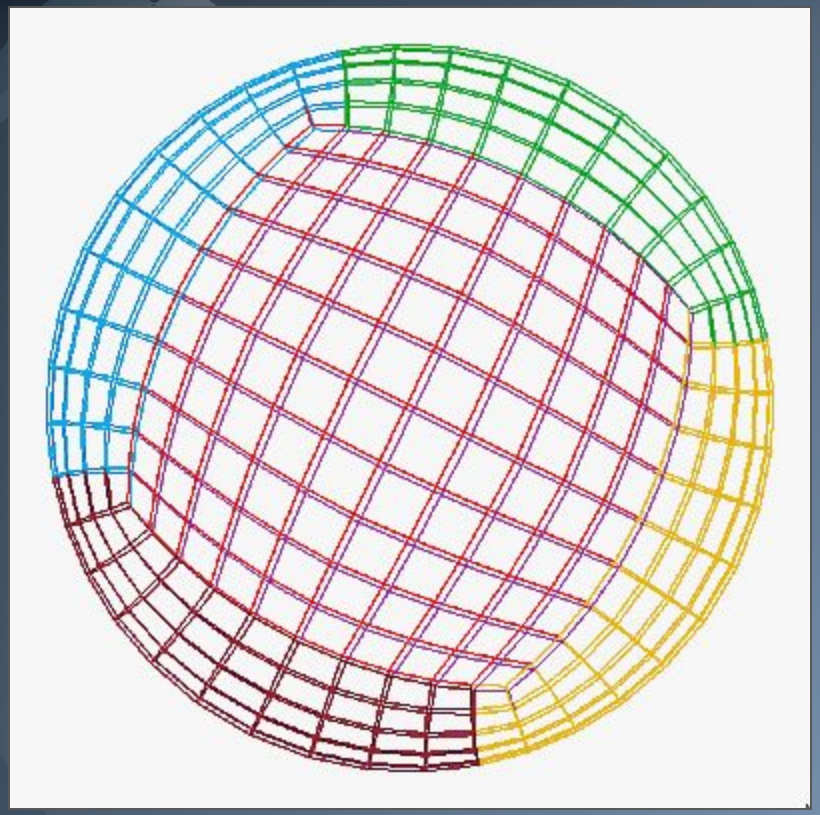
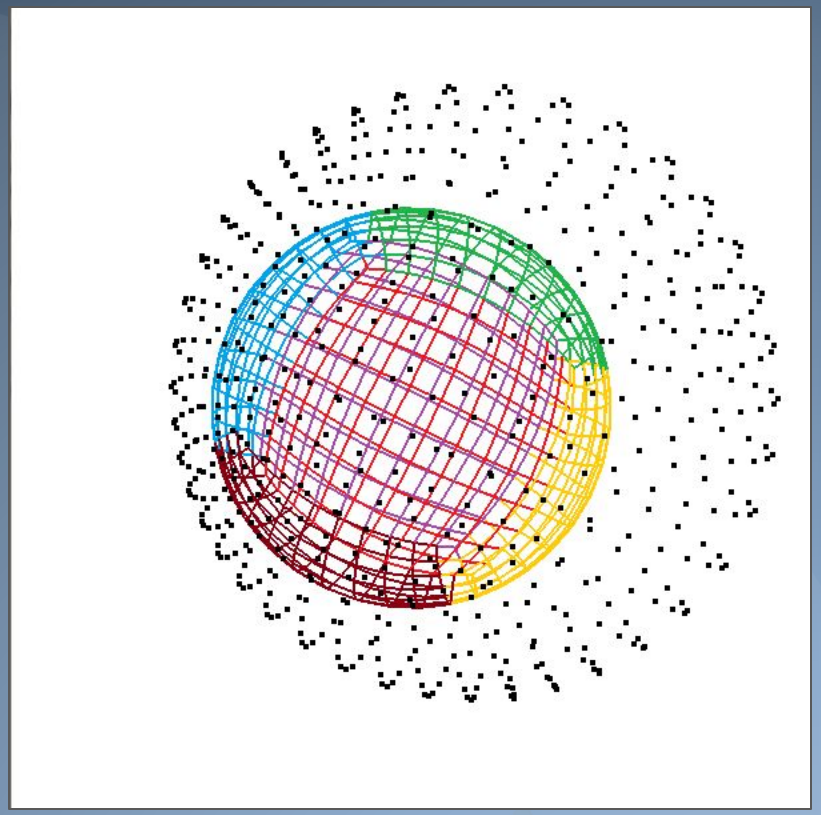
Cubed Sphere Simulator

Natalia Nebulishvili
Giorgi Rukhaia
Department of Mathematics
Tbilisi State University

Mariam Doliashvili
Department of Computer Science
Luka Tarielashvili
Tbilisi State University

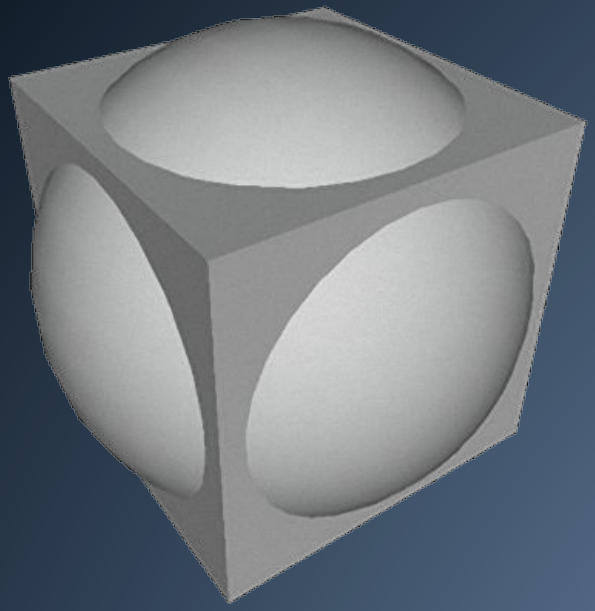
5th Georgian-German School and Workshop in Basic Science



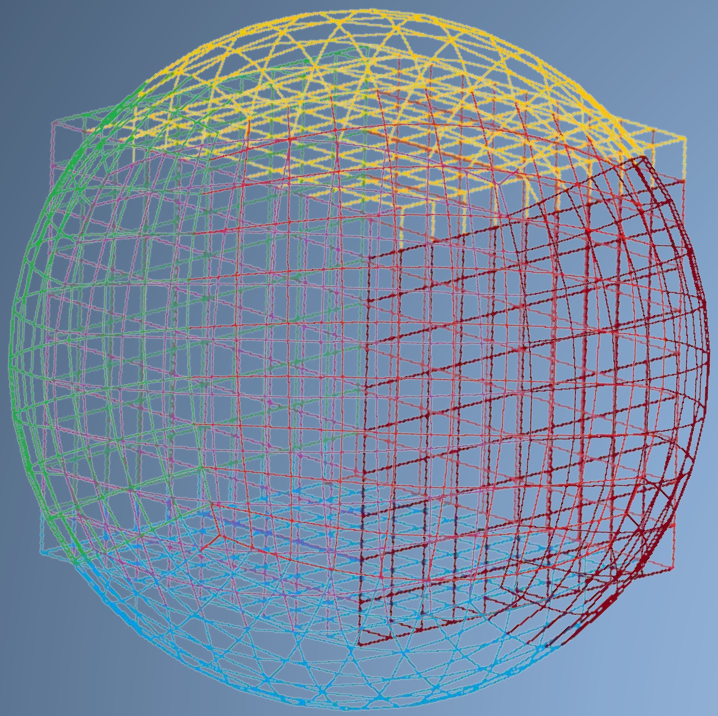


Main Goal

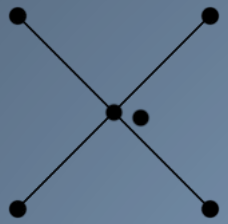
The environment for solving differential equations on the sphere

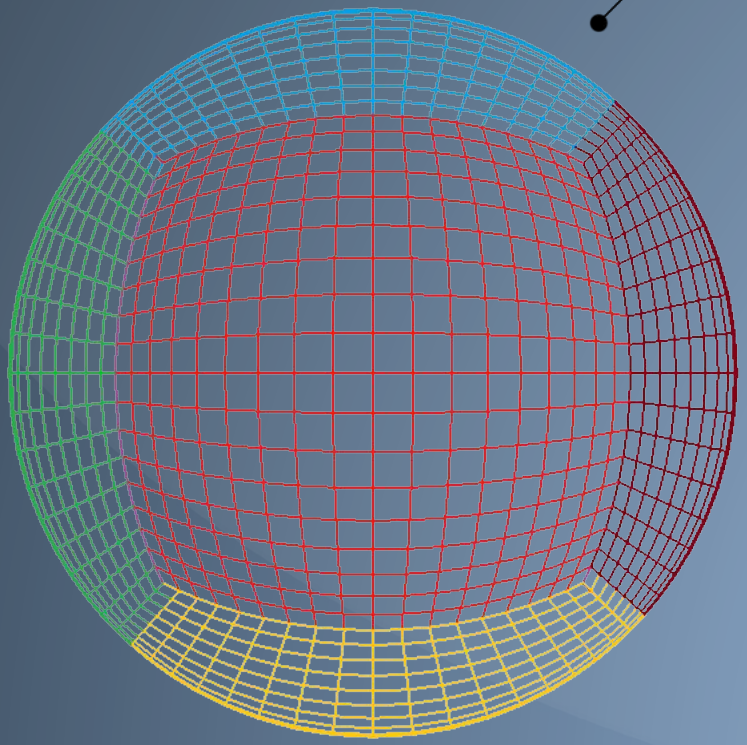


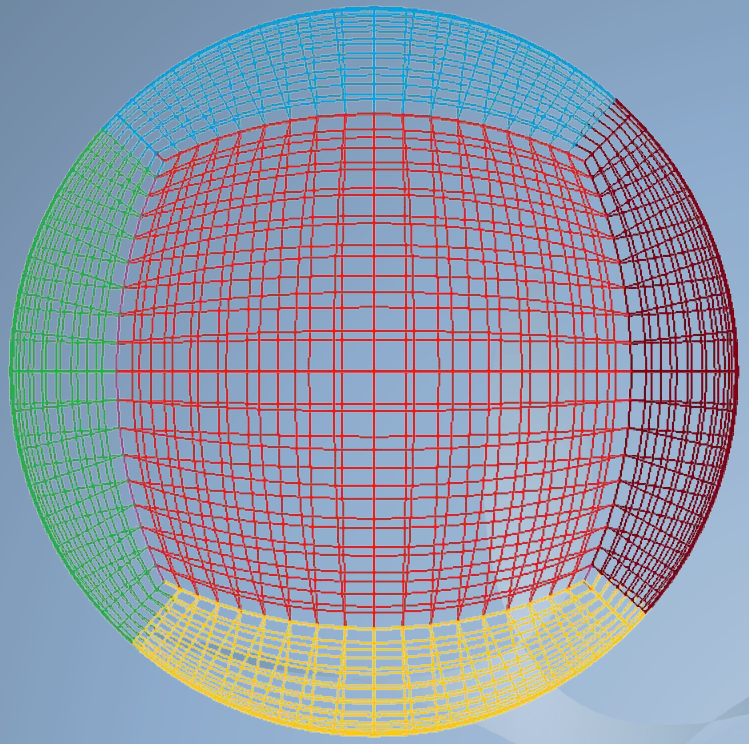
Creating a Grid



Smoothing the Grid







Non-uniform

Caushy Problem for Differential Equations

$$\frac{du_{ij}(t)}{dt} = \sum_{k \in S_{ij}} \alpha_{ijk} u_{ijk}(t) \qquad t \in (0, T]$$

S_{ij} -set of "neighbours" of ij point

$$u_{ij}(0) = u_{ij}^{0}$$

Stencils


u_{ij2}
u_{ij1} u_{ij0} u_{ij3}
u_{ij4}
u_{ij1}
u_{ij2}
u_{ij3} u_{ij4} u_{ij0} u_{ij5} u_{ij6}
u_{ij7}
u_{ij8}


u_{ij1} u_{ij2} u_{ij3}
u_{ij4} u_{ij0} u_{ij5}
u_{ij6} u_{ij7} u_{ij8}


u_{ij1} u_{ij2}
u_{ij0}
u_{ij3} u_{ij4}
u_{ij1} u_{ij3}
u_{ij2} u_{ij4}
u_{ij0}
u_{ij5} u_{ij6}
u_{ij7} u_{ij8}

Euler Explicit Scheme

$$u_{ij}^{m+1} = u_{ij}^{m} + \tau \sum_{k \in S_{ij}} \alpha_{ijk} u_{ijk}^{m}$$

τ - time step

$$u_{i,j,m} \approx u_{ij}(t_m)$$

Locally One Directional Decomposition

$$\frac{du_{ij}(t)}{dt} = \sum_{k \in S_{ij}} \alpha_{ijk} u_{ijk}(t) = f = f_{ij}^1 + f_{ij}^2 + f_{ij}^3 \qquad t \in (0, T]$$

$$\frac{du_{ij}^{(1)}(t)}{dt} = f_{ij}^1 \qquad \frac{du_{ij}^{(2)}(t)}{dt} = f_{ij}^2 \qquad \frac{du_{ij}^{(3)}(t)}{dt} = f_{ij}^3$$

$$u_{ij}^{(1)}(t_m) = u_{ij}(t_m) \qquad u_{ij}^{(2)}(t_m) = u_{ij}^{(1)}(t_{m+1}) \qquad u_{ij}^{(3)}(t_m) = u_{ij}^{(2)}(t_{m+1})$$

$$u_{ij}(t_{m+1}) = u_{ij}^{(3)}(t_{m+1})$$

Euler Implicit Scheme

$$u_{ij}^{(l),m+1} = u_{ij}^{(l),m} + \tau f_{ij}^{l,m+1} \quad l = 1,2,3$$

τ - time step

$$u_{i,j,m}^{(l)} \approx u_{ij}^{(l)}(t_m)$$

Classical Runge-Kutta Method

$$\frac{du_{ij}(t)}{dt} = \sum_{k \in S_{ij}} \alpha_{ijk} u_{ijk}(t) = f$$

$$u_{ij}(0) = u_{ij}^0$$

$$k_1 = f(u_{ij}^m)$$

τ - time step

$$k_2 = f(u_{ij}^m + \frac{1}{2}\tau k_1)$$

$$k_3 = f(u_{ij}^m + \frac{1}{2}\tau k_2)$$

$$k_4 = f(u_{ij}^m + \tau k_3)$$

$$u_{ij}^{m+1} = u_{ij}^m + \frac{\tau}{6}(k_1 + k_2 + k_3 + k_4)$$

$$u_{i,j,m} \approx u_{ij}(t_m)$$

Fourth-Order SSP Schemes

0					
C_2	a_{21}				
C_3	a_{31}	a_{32}			
- - - - - - - - - - - - - - - - - - - -					
Cs	a_{s1}	a_{s2}	. . . . .	$a_{s,s-1}$	
	b_1	b_2	. . . . .	b_{s-1}	b_s

$$\alpha = \begin{pmatrix} \alpha_{10} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ a_{s,0} & \cdots & a_{s,s-1} \end{pmatrix}$$

$$\widehat{U}^0 = U^m$$

$$\widehat{U}^i = \sum_{k=0}^{i-1} \left[\alpha_{ik} \widehat{U}^k + \tau \beta_{ik} F(\widehat{U}^k) \right], \quad i = \overline{1,s}$$

$$U^{n+1} = \widehat{U}^s$$

$$\beta = \begin{pmatrix} \beta_{10} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ \beta_{s,0} & \cdots & \beta_{s,s-1} \end{pmatrix}$$

$$K_{ik} = \beta_{ik} + \sum_{j=k+1}^{i-1} \alpha_{ik} K_{jk}$$

τ - time step

$$a_{ik} = K_{i-1,k-1,} \; k = \overline{1,i-1}, \quad i = \overline{1,s-1}$$

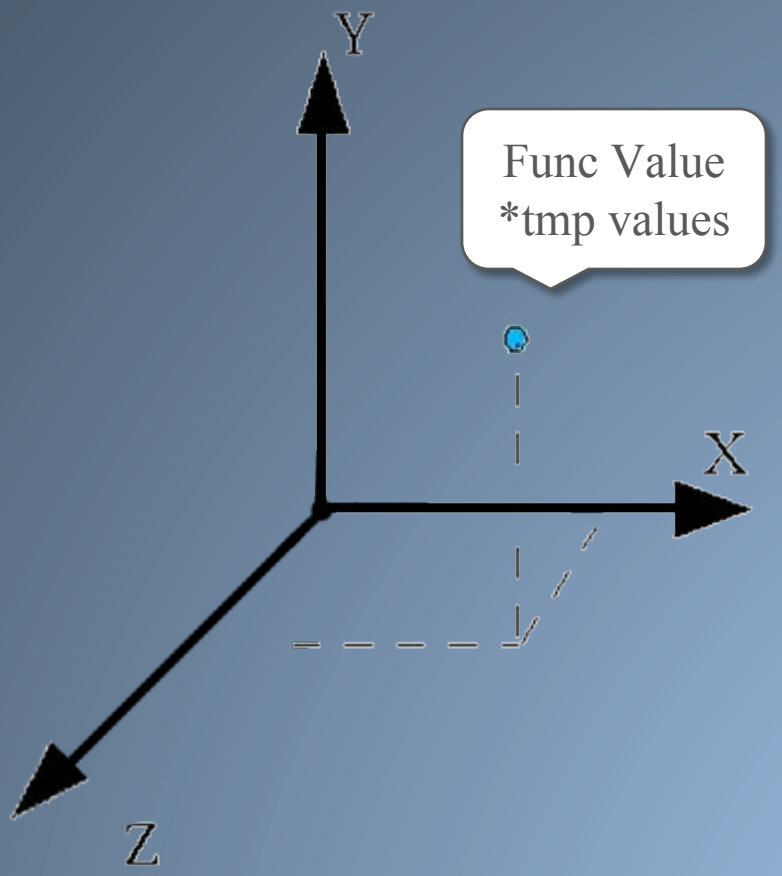
$$b_k = K_{s,k-1}, \quad k = \overline{1,s}$$

$$u_{i,j,m} \approx u_{ij}(t_m)$$

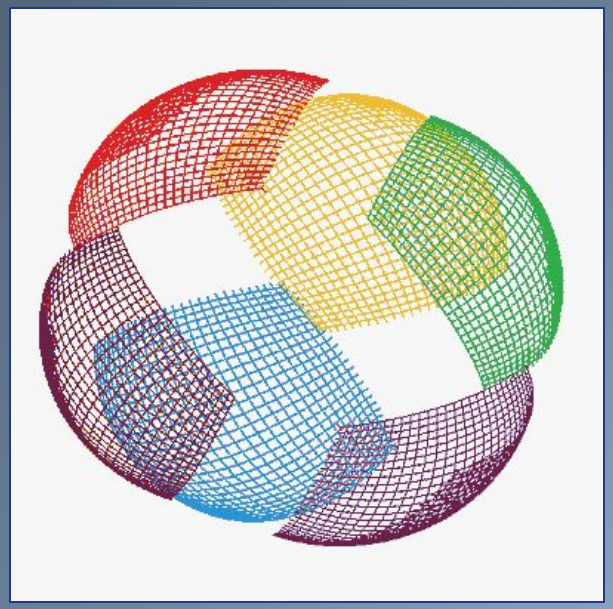
Programming language:

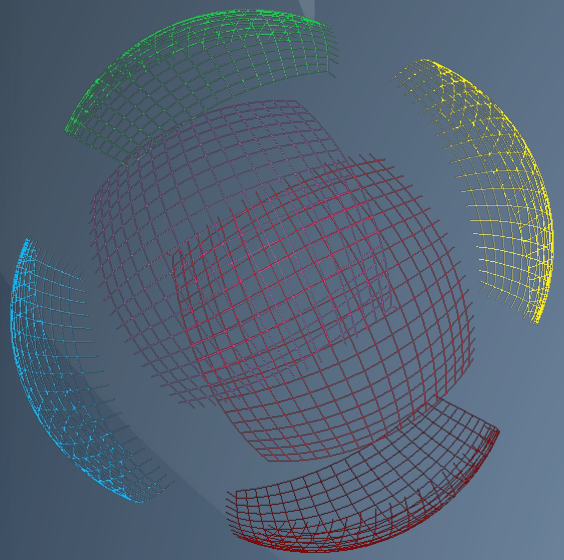
```
#pragma once
namespace CubedSphere{

class Point
{
private:
    double x, y, z;
    double value, tmpValue, k1, k2;
    double *k;
public:
    Point();
    double getX();
    double getY();
    double getZ();
    double getAlpha(i
    double getValue();
```

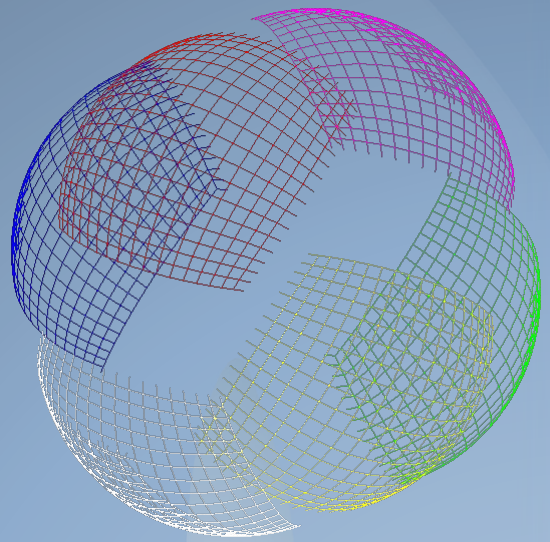
C++

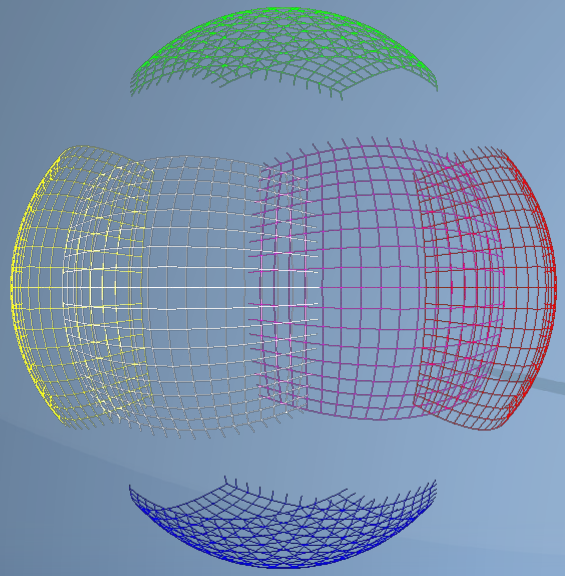

Y
Func Value
*tmp values
X
Z

Splitting the Sphere

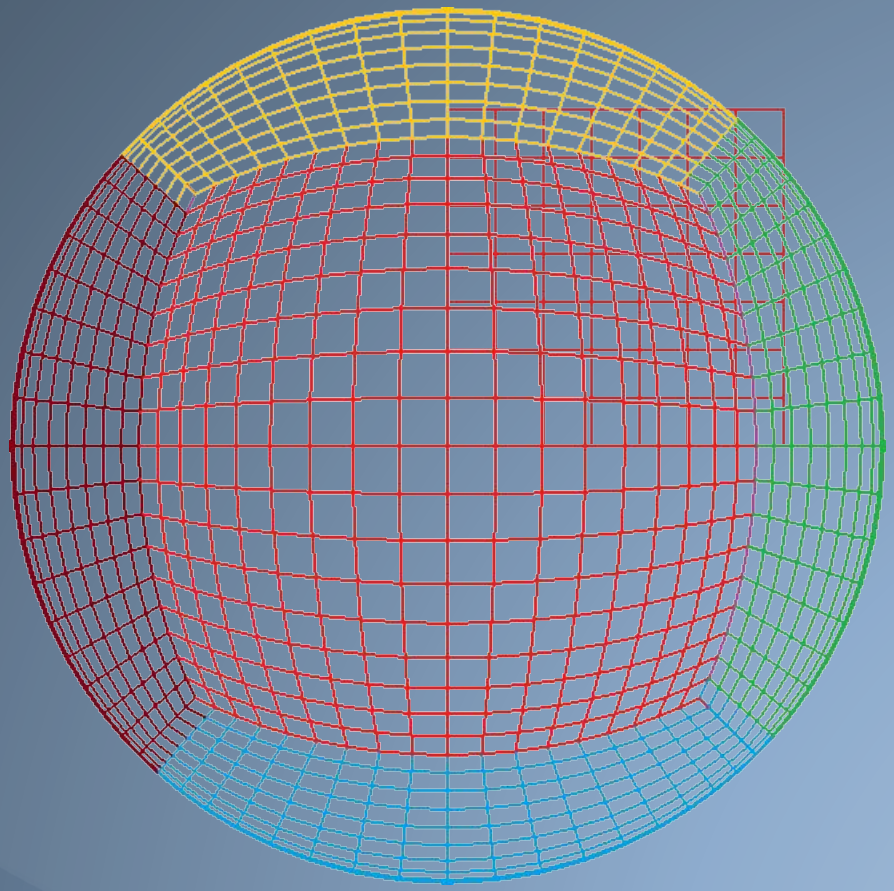






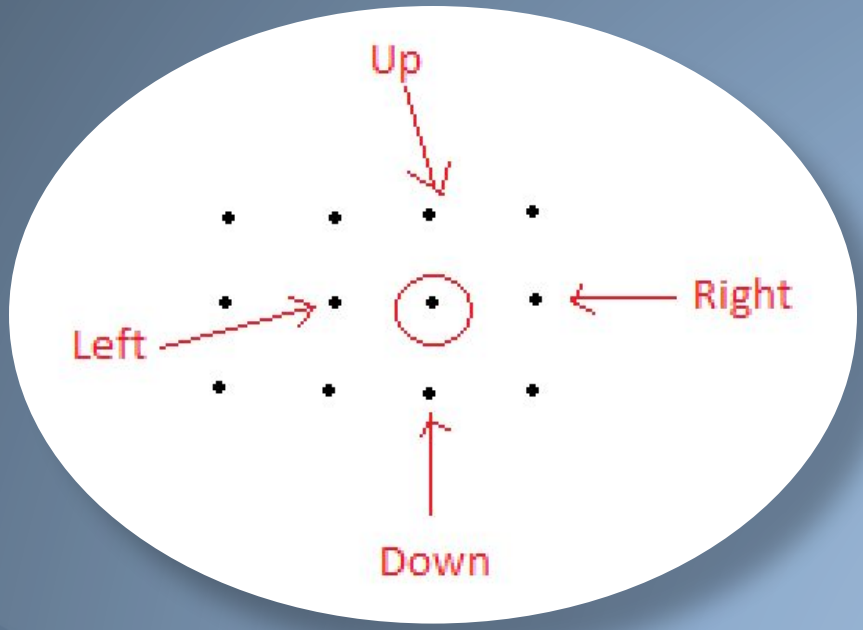


Optimised Algorithm for Grid Creation

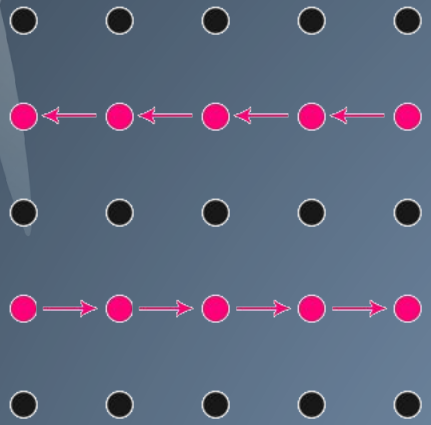


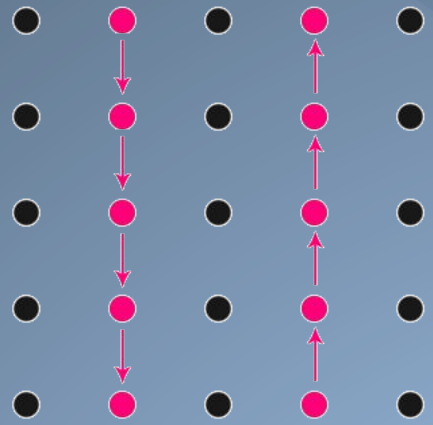
Sphere

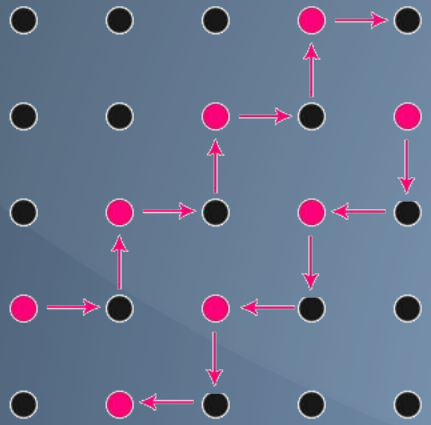
1) “Neighbours”

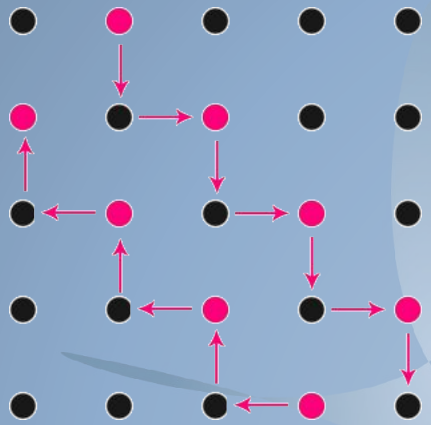


Moving around the Sphere

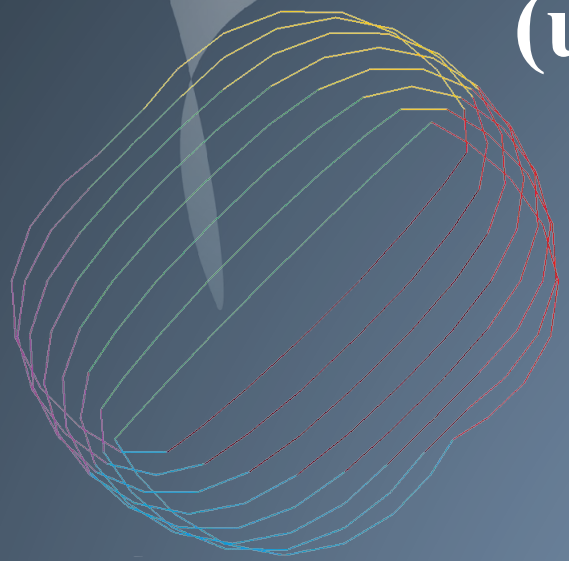


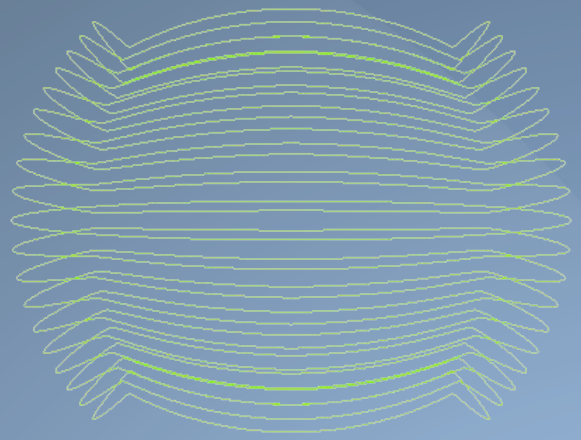


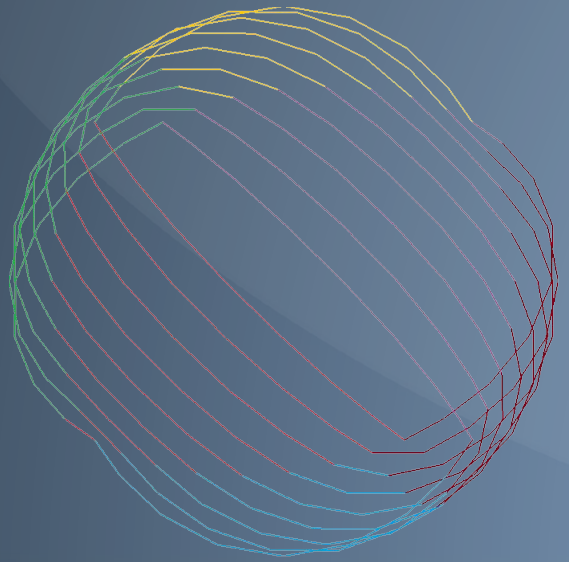


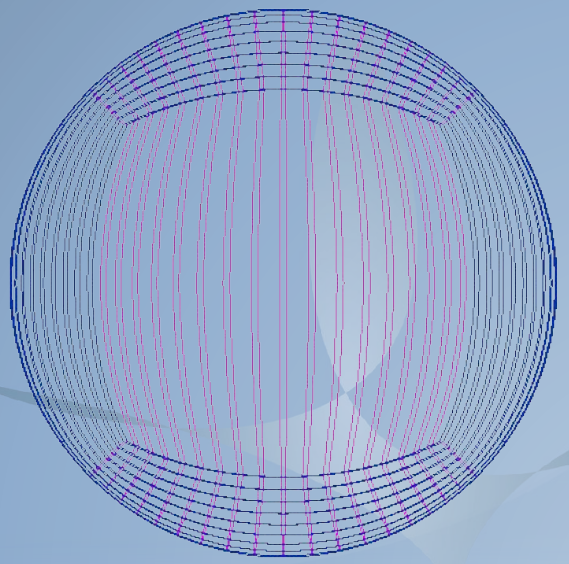


Moving around the Sphere (using streams)

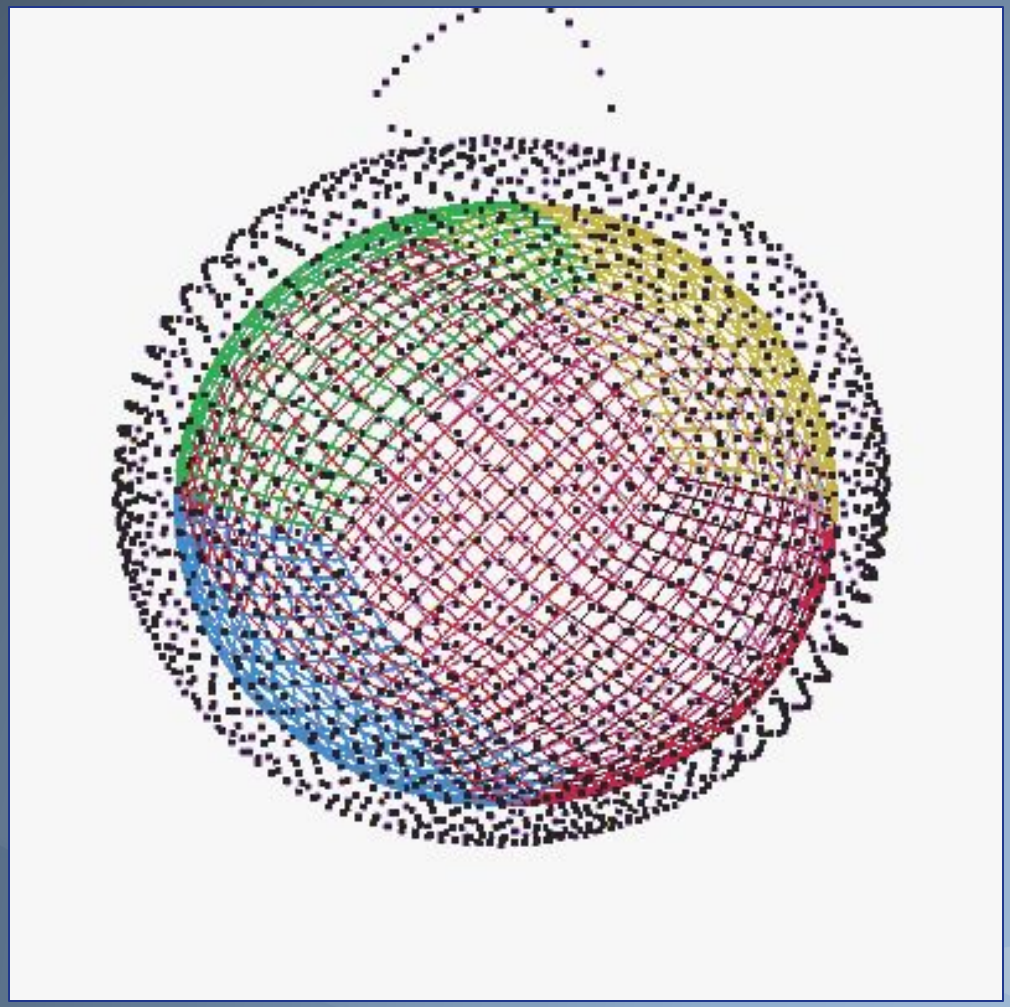








Visualization of Numerical Solution



Future Perspectives

- Local mesh refinement
- Expanding “neighbour “definitions
- Changing structure from sphere to any surface
- Solving real problems
- Further development for testing new schemes

G. Rukhaia L. Tarielashvili M. Doliashvili N. Nebulishvili

გმადლობთ !

Vielen Dank !