

Numerical simulations with
Ricci flow,
an overview and cosmological
applications.

Dumitru Vulcanov
The West University of Timisoara
2010

Introduction

In differential geometry, the Ricci flow is a process which deforms the metric of a Riemannian manifold in a manner formally analogous to the diffusion of heat, smoothing out irregularities in the metric.

It plays an important role in the apparent proof of the Poincaré conjecture, one of the seven Millennium Prize Problems for which the Clay Mathematics Institute offers a 1,000,000 USD prize for a correct solution.



R. S. Hamilton

G. Perelman



Introduction

Poincare conjecture says that :

Every simply connected, closed 3-manifold is homeomorphic to the 3-sphere.

In particular, Perelman proved Thurston's geometrization conjecture. This solved in the affirmative the Poincaré conjecture, posed in 1904, which before its solution was viewed as one of the most important and difficult open problems in topology.

G. Perelman



Introduction

Given a Riemannian manifold with metric tensor g_{ij} the Ricci flow may be defined by the geometric evolution equation :

$$\frac{\partial g_{ij}}{\partial t} = -2R_{ij} \quad (1)$$

Where R_{ij} is the Ricci tensor and t the "time"

The normalised version of (1) is

$$\frac{\partial g_{ij}}{\partial t} = -2R_{ij} + \frac{2}{n} r g_{ij} \quad (2)$$

Examples (trivial ones)

- Euclidean space, or more generally (Ricci) flat; Ricci flow has no effect.
- Sphere ; Ricci flow collapses the manifold to a point in finite time.
- Einstein manifolds (Ricci = constant \times metric) ; Ricci flow will collapse it to a point if it has positive curvature, leave it invariant if it has zero curvature, and expand it if it has negative curvature.

In particular, this shows that in general the Ricci flow cannot be continued for all time, but will produce singularities. For a 3 dimensional manifold, Perelman showed how to continue past the singularities using surgery on the manifold.

Introduction

Examples (non-trivial ones)

- **Cigar soliton or Witten Black-Hole** having a metric

$$g = \frac{dx^2 + dy^2}{1 + x^2 + y^2} = \frac{dr^2 + r^2 d\theta^2}{1 + r^2} = ds^2 + \tanh^2 s d\theta^2$$

Where : $s = \operatorname{arcsinh} r = \log(1 + \sqrt{1 + r^2})$

- **This is a stationary solution to the Ricci flow (steady soliton)**
- **Has positive curvature**
- **Asymptotic to a cylinder**
- **Curvature decays exponentially**
- **Is rotationally symmetric**

Introduction

Many versions of the Ricci flow have also been studied:

- **Various curvature flows defined using either an extrinsic curvature, which describes how a curve or surface is embedded in a higher dimensional flat space, or an intrinsic curvature, which describes the internal geometry of some Riemannian manifold,**
- **Various flows which extremalize some quantity mathematically analogous to an energy or entropy,**
- **Various flows controlled by a p.d.e. which is a higher order analog of a nonlinear diffusion equation....**

Numerical simulations

The problem of numerical simulations and visualizations with Ricci flow of 2 or 3 dimensional manifolds comes naturally as Ricci flow is geometric by nature.

Ricci flow acts directly on the metric of the surface, tending not to preserve the embeddedness. A number of interesting results have been obtained by restricting to classes of metrics of revolution, since such symmetries are preserved under Ricci flow and the metric depends on considerably fewer parameters in such cases.

These surfaces tend to remain embedded in \mathbb{R}^3 making direct visualization possible.

Numerical simulations

Since the Ricci flow equations (1) or (2) are only weakly parabolic and since numerical evolutions appear to be very unstable, we need some solutions to avoid this. Mainly there are in the literature two solutions (see below), namely :

- filtering and reparametrization method (inspired by spectral methods)
- using of other flow equations derived from (1), the deTurck flow is such an example.

However an explicit finite-difference scheme for (1) is still usable with very small time-steps and high number digits precision (several hundred) - removing short wavelength instabilities.

Numerical simulations

The main goal of numerical simulations with Ricci flow is to exhibit the formation of singularities and/or neck pinching phenomenon, which occurs naturally for metrics of revolution.

Numerical simulations in Ricci flow are very similar to those in Numerical relativity, thus some methods and experience can be... imported here – even codes.

There are not very many results in this direction reported till now. We can mention only two articles, namely :

- 1) Garfinkle & Isenberg - math.DG/0306129 (GI)**
- 2) Rubinstein & Sinclair - math.DH/0406189 (RS)**

2-surfaces

A two-dimensional surface of revolution of genus zero embedded in \mathbb{R}^3 can be defined in a polar representation with coordinates $x^1 = \rho$, $x^2 = \theta \in [0, 2\pi[$ by a metric of the form :

$$[g_{\mu\nu}] = \begin{bmatrix} h(\rho) & 0 \\ 0 & m(\rho) \end{bmatrix}$$

where $\sqrt{m(\rho)}$ has the direct physical interpretation as the radius from the axis of rotation. For a closed surface, we have

$$m(\rho_{pole}) = 0$$

Choosing $\rho = 0$ at the North Pole and $\rho = \pi$ at the South Pole

2-surfaces

To avoid the numerical instabilities in an implicit scheme for (1) and inspired by spectral methods, RS introduced a filter which consists of transforming to Fourier space (DFT), dropping shorter wavelength terms, and then transforming back.

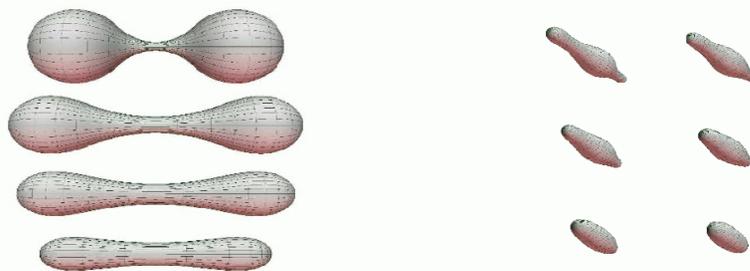
Rubnstein and Sinclair used their own code, called Ricci_rot (in C) for numerical simulations and special codes for visualisations (in OpenGL)

No convergence and stability analysis was done – at least not reported, for the explicit direct finite differencing schema used. These produced several troubles, especially in the 3D case.

2-surfaces

The Ricci flow forces some parts of a surface to contract while others are inflating creating further sources of numerical instability. The solution to this problem is to reparametrize the metric, the most pleasant one let $h(\rho)$ be a constant.

Here are some results reported by RS



**Dumbbell shape 2-surface
under Ricci flow –
Reproduced from [1]
Courtesy of H. Rubinstein**

Numerical simulations with Maple

Advantages :

- Storing the analytical and numerical results in separate Maple libraries, for later use. Example : step (1) above produce the main equations stored in libraries loaded later in separate Maple worksheets (bisurface_eqs.m is for 2 surfaces of revolution – see below)
- as a result steps (2) and (3) or (4) above can be done separately in special worksheets
- Separately doing the numerical evolution and visualization after analysis steps
- Everything done under the same language and environment !
- Easy to use results for anybody familiar with Maple and not only !

A special mention for the step (1) . Here we used the GrTensorII package for algebraic computing of the main equations and geometrical objects (as the Ricci tensor and scalar). GrTensorII is specially designed for differential geometry calculations in riemannian geometry

Numerical simulations with Maple

For the 2-surface described earlier (the metric (3)) the Ricci tensor has only two non-vanishing components, and the Ricci flow corresponding equations are :

$$\frac{\partial}{\partial t} h(\rho, t) + \frac{1}{2} \frac{\left(\frac{\partial}{\partial \rho} m(\rho, t)\right)^2}{(m(\rho, t))^2} - \frac{\frac{\partial^2}{\partial \rho^2} m(\rho, t)}{m(\rho, t)} + \frac{1}{2} \frac{\frac{\partial}{\partial \rho} h(\rho, t) \frac{\partial}{\partial \rho} m(\rho, t)}{h(\rho, t) m(\rho, t)} = 0$$
$$\frac{\partial}{\partial t} m(\rho, t) + \frac{1}{2} \frac{\left(\frac{\partial}{\partial \rho} m(\rho, t)\right)^2}{(h(\rho, t) m(\rho, t))} - \frac{\frac{\partial^2}{\partial \rho^2} m(\rho, t)}{h(\rho, t)} + \frac{1}{2} \frac{\frac{\partial}{\partial \rho} h(\rho, t) \frac{\partial}{\partial \rho} m(\rho, t)}{h(\rho, t)^2} = 0$$

These and other geometrical objects are then stored in a Maple library (*.m extension) for later use and load. For the 2 surfaces these are the main equations we used for numerical simulations.

Numerical simulations with Maple

In our investigations we used as initial the function $m(\rho)$ as

$$m_0(\rho) = \frac{(\sin(\rho) + c_3 \sin(3\rho) + c_5 \sin(5\rho))^2}{(1 + 3c_3 + 5c_5)^2}$$

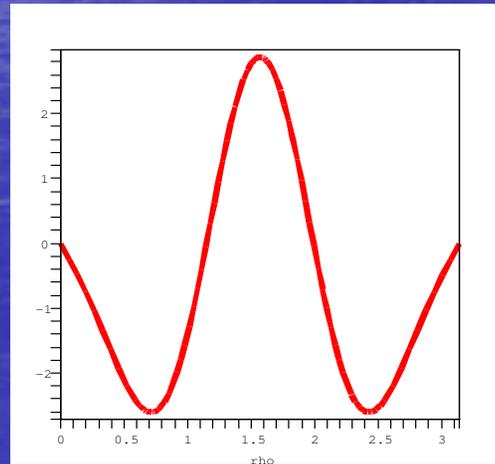
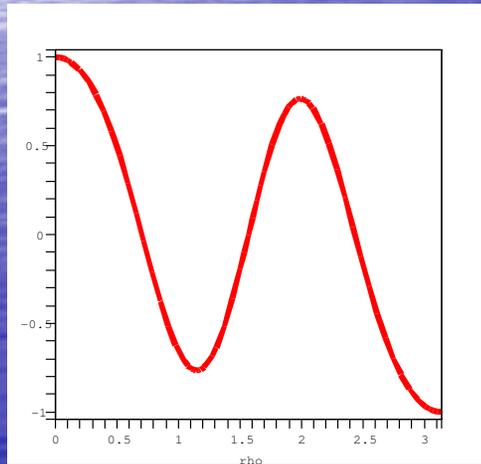
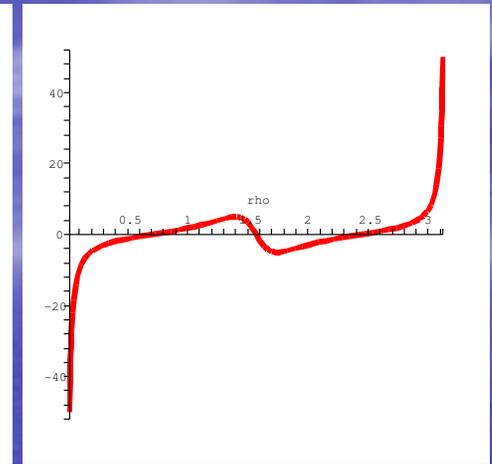
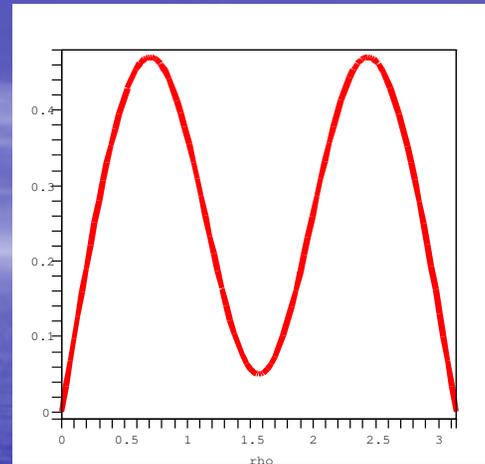
with appropriate values for the coefficients c_3 and c_5 controlling the shape of the surface - for example for $c_3 = c_5 = 0$ the surface is a sphere.

For initial value of the $h(\rho)$ functions we also used $h_0(\rho) = 1$.

Before using these initial data, we investigate the behaviour and shape of these functions and their derivatives in order to reveal some tricky points, if any.

Numerical simulations with Maple

Thus for $c3 = 0.766$ and $c5 = -0.0091$ we have the shape of the $m_0()$ function (left panel) and of the first main equation above with $m(\rho, t)$ replaced with m_0 (right panel).



The first and the second spatial derivatives of the m_0 function are plotted in revealing some turning points where we need special care with the future numerical simulations.

Numerical simulations with Maple

With these established, we composed a Maple program for finite differencing the above equations.

First we denoted the two unknown functions $m(\rho, t)$ and $h(\rho, t)$ with two matrices, $mm[i,j]$ and $hh[i,j]$ respectively.

We also denoted the spatial interval between the points on the ρ axis with $\delta = \Delta\rho$ and the time step with $dete = \Delta t$ variables.

Separately we denoted also the parts from the two above equations containing only spatial derivatives with two matrices, namely $eno1[i,j]$ and $eno2[i,j]$.

For the time derivatives we used the forward Euler method (it has the advantage that one is able to calculate quantities at timestep $j + 1$ in terms of only quantities known at timestep j) and for the spatial derivatives we can use a second-order representation still using only quantities known at timestep j .

Numerical simulations with Maple

Thus the "eno" matrices have the shape as :

$$\text{eno1}_{i,j} = -\frac{(\text{mm}(i+1, j) - \text{mm}(i-1, j))^2}{8\text{mm}(i, j)^2 \delta^2} + \frac{\text{mm}(i+1, j) - 2\text{mm}(i, j) + \text{mm}(i-1, j)}{\text{mm}(i, j)\delta^2}$$
$$\frac{(\text{hh}(i+1, j) - \text{hh}(i-1, j))(\text{mm}(i+1, j) - \text{mm}(i-1, j))}{8\text{mm}(i, j)\text{hh}(i, j)\delta^2}$$
$$\text{eno2}_{i,j} = -\frac{(\text{mm}(i+1, j) - \text{mm}(i-1, j))^2}{8\text{mm}(i, j)\text{hh}(i, j)\delta^2} + \frac{\text{mm}(i+1, j) - 2\text{mm}(i, j) + \text{mm}(i-1, j)}{\text{hh}(i, j)\delta^2}$$
$$\frac{(\text{hh}(i+1, j) - \text{hh}(i-1, j))(\text{mm}(i+1, j) - \text{mm}(i-1, j))}{8\text{hh}(i, j)^2 \delta^2}$$

Now the time integration of the two finite-differenced equations is straightforward as :

$$\text{hh}(i, j+1) = \text{hh}(i, j) + \text{dete} * \text{eno1}(i, j)$$

$$\text{mm}(i, j+1) = \text{mm}(i, j) + \text{dete} * \text{eno2}(i, j)$$

for any time $j > 0$ if the initial values for $\text{hh}[i,0]$ and $\text{mm}[i,0]$ are provided.

Numerical simulations with Maple

With the above established theoretical steps we proceeded to compose a special Maple program (worksheet) for accomplishing the numerical tasks.

First the program starts with the necessary loadings, namely the library for the bisurface case equations (bisurface eq.m) previously saved and then the finite-differences discretization of the main equations :

```
> restart;
> read("bisurface_eqs.m");
> eno1[i,j]:= ((subs(m(rho,t)=mm[i,j],subs(h(rho,t)=hh[i,j],
  subs(diff(m(rho,t),rho)=(mm[i+1,j]-mm[i-1,j])/2/delta,
  subs(diff(h(rho,t),rho)=(hh[i+1,j]-hh[i-1,j])/2/delta,
  (subs(diff(m(rho,t),rho,rho)=(mm[i+1,j]-2*mm[i,j]+
    mm[i-1,j])/delta/delta,eno1))))))):
```

And similarly for the eno2 object.

Numerical simulations with Maple

The next lines simply introduce of the initial data (as the function $m_0(\rho)$) and some checkings of it's values in different points of the grid. Then comes the establishing of the grid values, the number of the points on the axis (n), the values of the values for the c_3, c_5 constants and of the spatial interval on the axis (δ):

```
> m0(rho):=((sin(rho)+c3*sin(3*rho)+c5*sin(5*rho))/  
            (1+3*c3+5*c5))**2;  
> plot(subs(c3=0.766,c5=-0.091,m0(rho)),rho=0..10);  
> n:=50;dete:=0.0018;  
> c3:=0.766;c5:=-0.091;delta:=evalf(Pi/n);  
> courfac:=dete/delta/delta/2;
```

After a series of commands establishing the initial values of main matrices, comes the most important part of the program, were we integrate the two equations :

```
> tmax:=80 ; for v from 1 to tmax do; for k from 1 to n-v-1 do  
> hhini:=eval(subs(i=k,j=v-1,hh[i,j])); mmini:=eval(subs(i=k,j=v-1,mm[i,j]));  
> coco1:=eval(subs(i=k,j=v-1,eno1[i,j]));coco2:=eval(subs(i=k,j=v-1,eno2[i,j]));  
> hh[k,v]:=eval(hhini+dete*coco1);  
> mm[k,v]:=eval(mmini+dete*coco2);end do:end do;
```

and this is all, as for the numerical calculations !!!! Well, almost !

Numerical visualisations with Maple

The next lines of the program are just a series of plotting commands in order to visualise the shape and values of the unknown functions, at all the times processed or at different times separately.

```
> for j from 0 to tmax do  
> bebe[j]:=plot([[r[p],mm[p,j]]$p=0..n],t) od;  
> display(seq(bebe[m],m=0..tmax));  
> plot([[r[p],mm[p,0]]$p=0..n],t);  
> plot([[r[p],mm[p,1]]$p=0..n],t);.....  
> plot([[r[p],mm[p,tmax]]$p=0..n],t);
```

Here we need plots package and of course the program contains similar lines for plotting the values for $h(\rho; t)$

The rest of the program is mainly dedicated to visualisations of the results, several plottings were done, including some series of pictures for animation movies !

Numerical visualisations with Maple

For more striking view of the results we used a separate sequence of commands, where we plotted in 3D figures the time evolution of the two functions $m(\rho:t)$ and $h(\rho:t)$, namely

```
> ccc:=[seq([ seq([p,k, evalf(subs(i=p,j=k,mm[i,j]))]),  
p=tmax+1..n-tmax-1]), k=0..tmax): surfdata(ccc,axes=BOXED);  
> ddd:=[seq([ seq([p,k, evalf(subs(i=p,j=k,hh[i,j]))]),  
p=tmax+1..n-tmax-1]), k=0..tmax): surfdata(ddd,axes=BOXED);
```

The time evolution of both functions are plotted as a surface in a figure having the time as one of the axes. Some examples will be plotted in the next slides.

For the stability analysis, we used, in a separate Maple worksheet the von Neumann analysis – other methods are also possible. For our case of a 2-surface of revolution we established the Courant factor as :

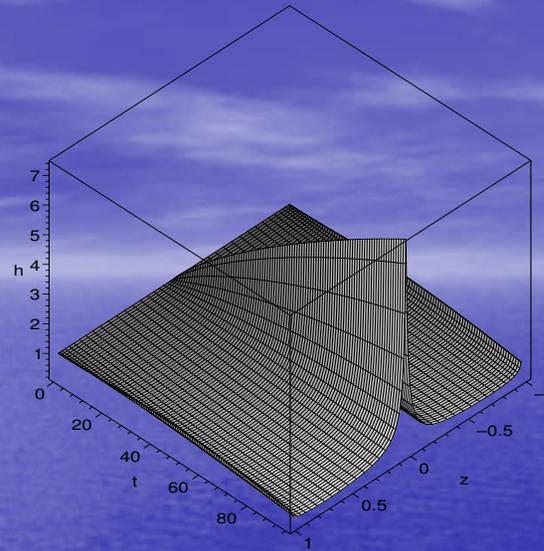
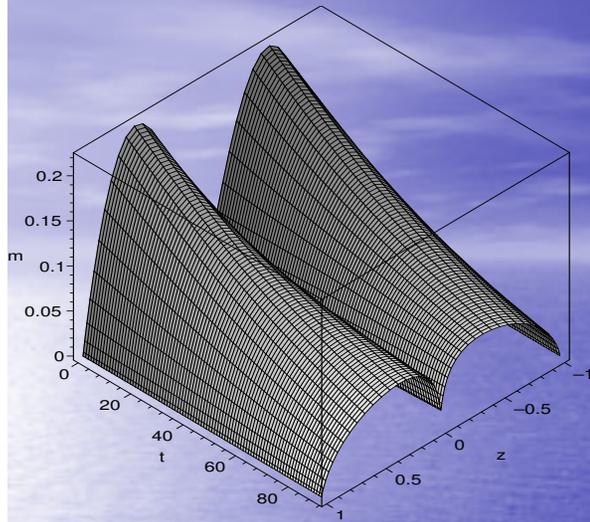
$$\frac{\Delta t}{2(\Delta \rho)^2} \leq 1 \quad \text{actually we used} \quad 0.2533029590$$

For both finite differences equations !

Special care also we had for the boundary values where we used specific method in every case – this is to be reported elsewhere !

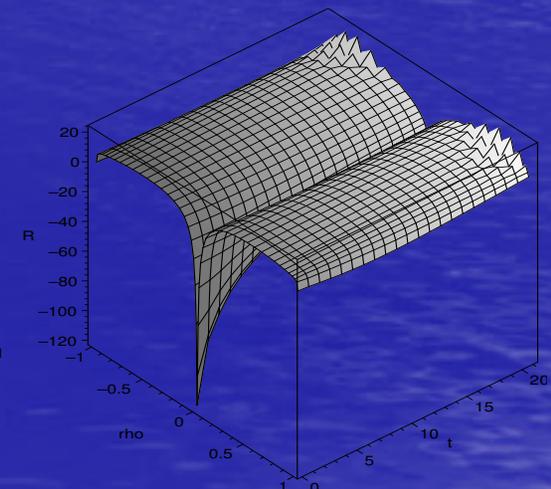
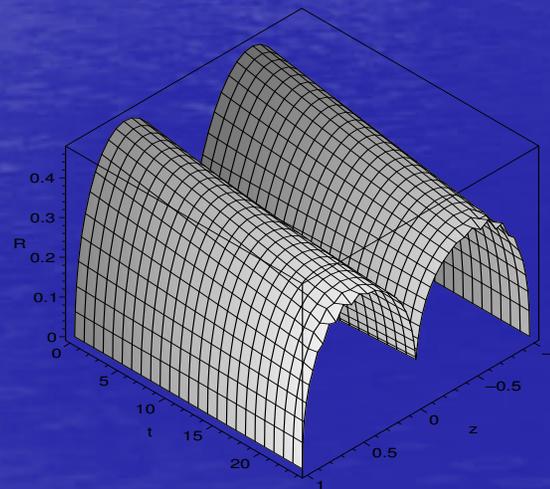
Numerical visualisations with Maple

$C3=0,766$ $C5= -0.091$
 $N=50$ $\Delta t=0.002$
 $\delta = 0.06283185308$



Time evolution of the
← functions $m(t)$ (left)
and $h(t)$ (right)

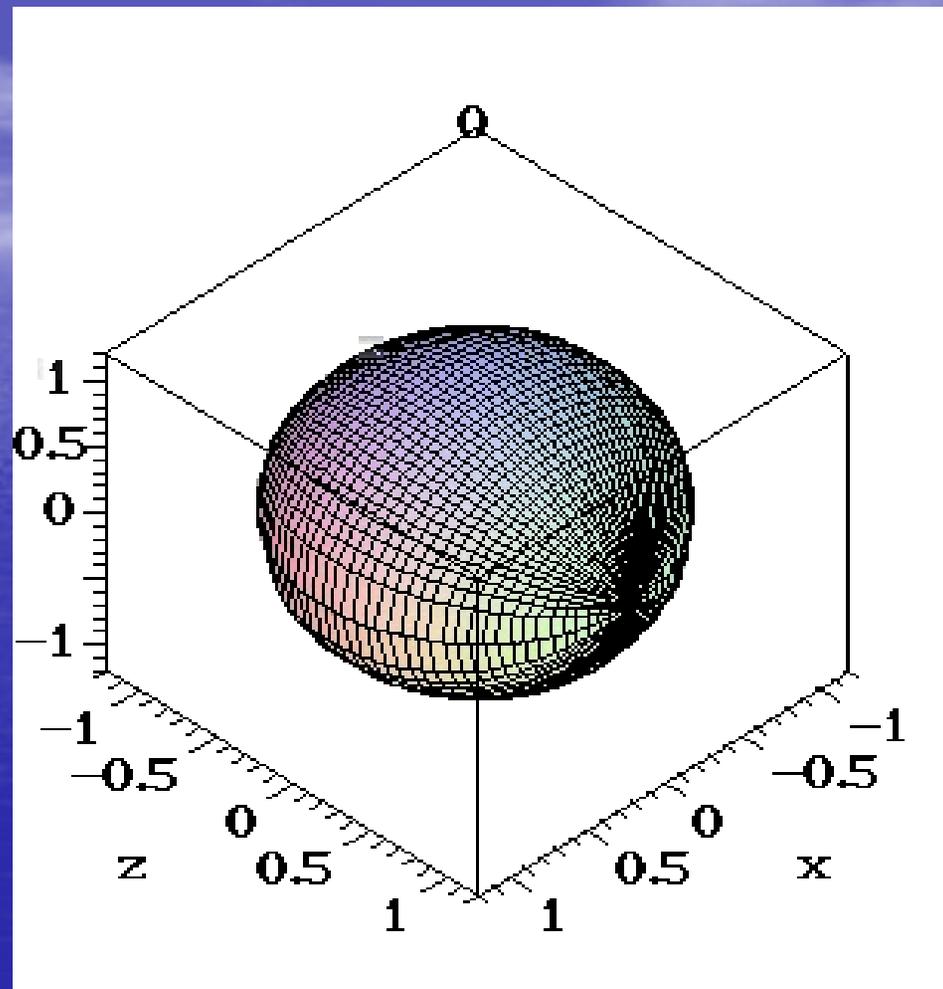
Time evolution of the
radius $m(t)^{1/2}$ (left) and
of the Ricci scalar $R(t)$
(right) →



The sphere

$$c_3 = 0$$

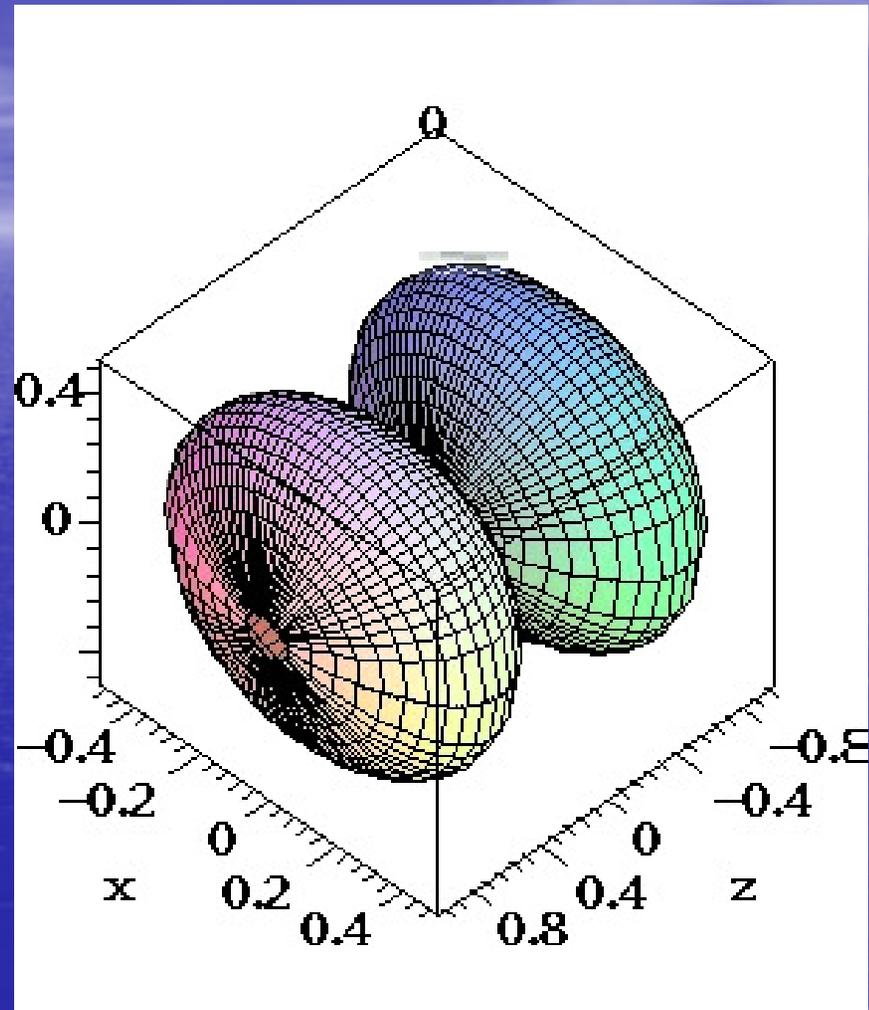
$$c_5 = 0$$



Two folded surface

$$c_3 = 0.766$$

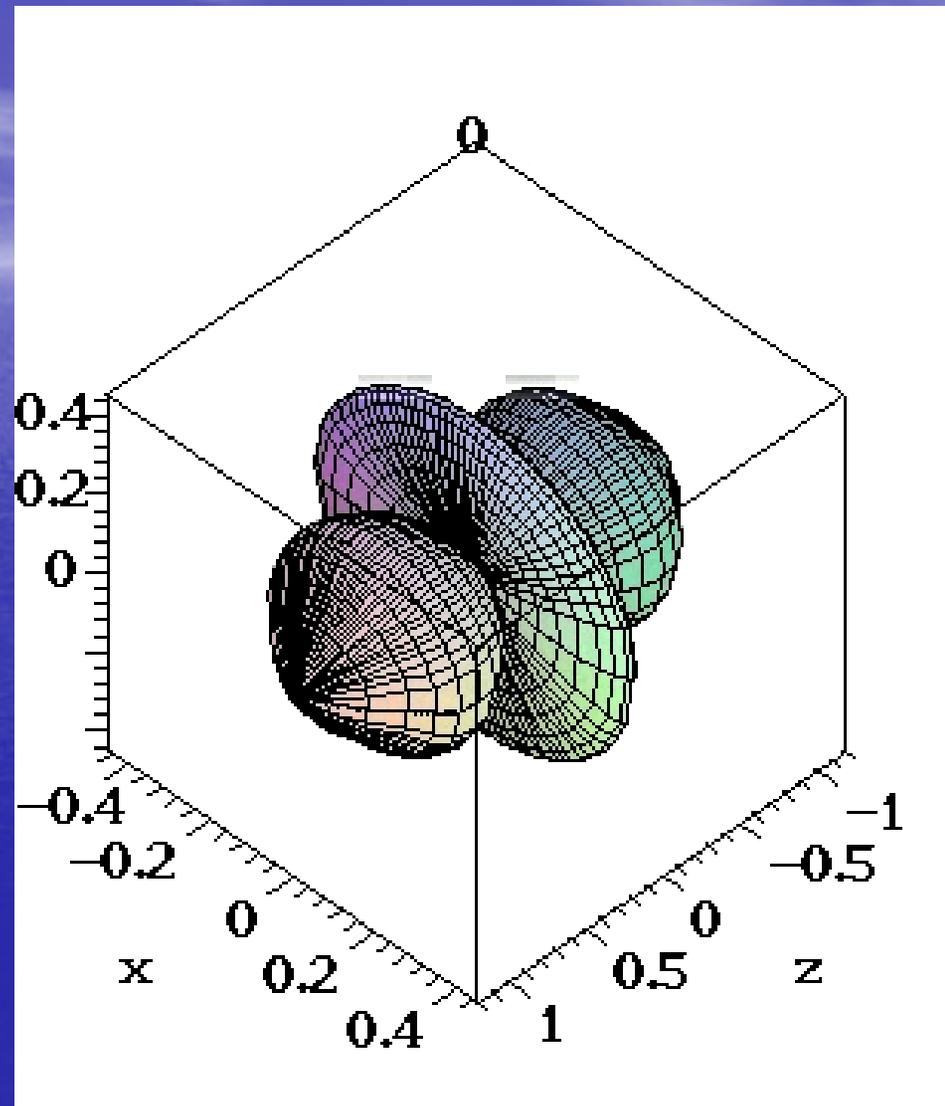
$$c_5 = -0.091$$



Three folded surface

$$c_3 = 0.021$$

$$c_5 = 0.598$$



3-surfaces

For the 3-dimensional surface of revolution we investigated (as RS) the metric as

$$[g_{\mu\nu}] = \begin{bmatrix} h(\rho) & 0 & 0 \\ 0 & m(\rho) & 0 \\ 0 & 0 & m(\rho) \cos^2(\sqrt{K_2} \theta) \end{bmatrix}$$

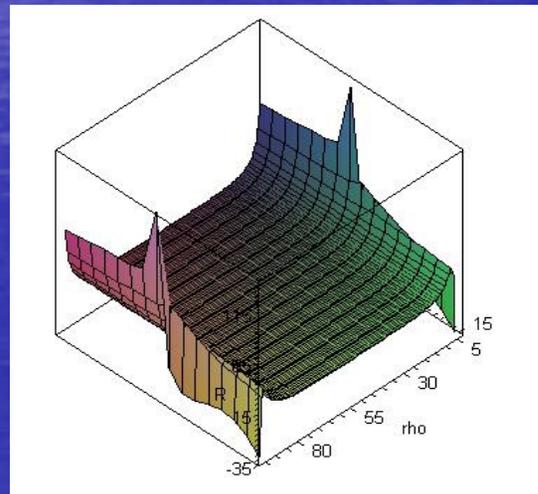
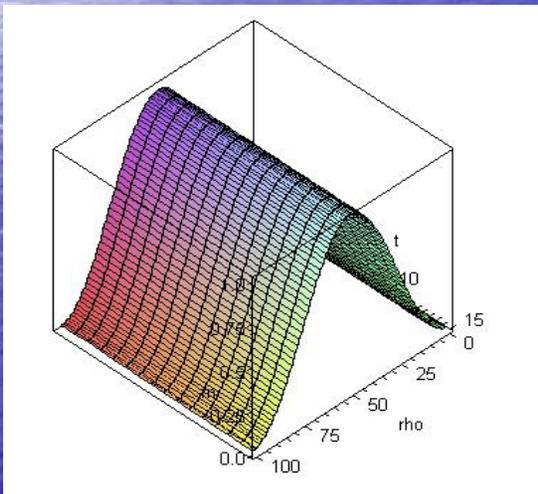
where $\rho \equiv x^1$ plays the role of a latitude and $\theta \equiv x^2$ the role of a latitude on the abstract Riemannian surface of revolution and K_2 is a constant.

The pinching behavior of this surface under unnormalized Ricci flow (1) was studied by Rubinstein and Sinclair starting at $t = 0$. Also this time an explicit finite-differences schema was used. The instabilities forced to restrict to fairly large time steps and they were forced to run their code only to a restricted (even not at equal times) number of steps. These results were called by the authors as "qualitatively correct" !!!

3-surfaces in Maple

Using our programs in Maple (as described before) for this type of 3surfaces we were able, after a von Neumann stability analysis to establish a Courant factor and to run properly a number of iterations (20..30) using the intinial data as :

$$m(\rho) = \frac{1}{10000} + \sin^2\left(\frac{9\pi\rho}{40}\right) \quad h(\rho) = 1 \quad K_2 = 1$$



←The $m(\rho)$ function (left) and the Ricci scalar (right) for a 3-surface of revolution under Ricci flow

Corseted sphere surfaces

In their study on a critical behaviour of 3-surfaces under Ricci flow, Garfinkle and Isenberg (GI) used a so called corseted sphere geometry having :

$$g = e^{2X} (e^{-2W} d\psi^2 + e^{2W} \sin^2 \psi [d\theta^2 + \sin^2 \theta d\phi^2])$$

Here (ψ, θ, ϕ) are standard angular coordinates on the three sphere. The metric functions X and W are functions only of ψ . $W = X$ and X is choosed so that :

$$4e^{4X} \sin^2 \psi = \sin^2 2\psi \quad \text{for } \cos^2 \psi \geq 1/2$$

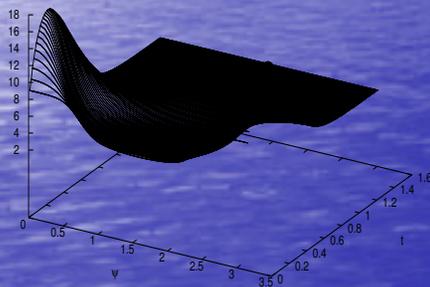
$$4e^{4X} \sin^2 \psi = \sin^2 2\psi + 4\lambda \cos^2 2\psi \quad \text{for } \cos^2 \psi \leq 1/2$$

Here λ is a constant, which parameterizes the degree of corseting for these geometries.

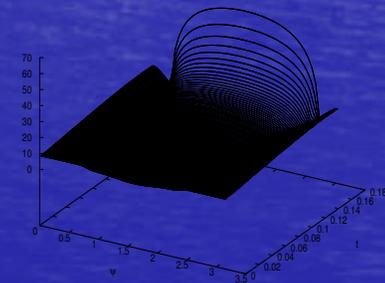
Corseted sphere surfaces

For their numerical investigations, GI used the DeTurck flow, where the PDE equations are strong parabolic and a finite differences schema for the time evolution and centered finite differences for the spatial derivatives. The numerical investigations pointed out a critical value for the λ parameter ($\lambda = 0.1639$) dividing the behavior under the flow into two regimes :

- a subcritical one - for larger λ the flow converges to a round sphere
- a supercritical one - for smaller λ the flow goes to a S^2 neck pinching singularity



Subcritical (left) and supercritical (right) behaviour of R_S^2 under Ricci flow for a corseted sphere (courtesy of David Garfinkle)



At the critical value for λ the flow approaches a “javelin” geometry, marked by curvature singularities at the poles, with roughly uniform curvature between the poles. This javelin geometry corresponds to the “type 3” singularity described by Hamilton and discussed by Chow.

Corseted sphere surfaces

The work of Garfinkle and Isenberg is a typical example of how the experience in numerical relativity can be applied to Ricci flow simulations. The DeTurk “trick” used to make the PDE strongly parabolic is similar to the ADM-BSSN version of numerical relativity.

In spite of this, we still think that the use of initial Ricci flow equations (normalised or not) can be used, as the explicit finite differences schema can be done stable – a Courant factor in certain conditions is possible to exhibit for this schema – as we proved using our Maple programs. This is similar to the “pure” ADM method in NR. We will develop this idea in our next investigations.

But this is an open question from now on, and the results will be reported elsewhere... I hope !

Selected references

- R.S. Hamilton, *J. Diff. Geom.*, **17**, 255 (1982)
- D. Garfinkle, J. Isenberg, *math.DG/0306129* (2003)
- J.H. Rubinstein, R. Sinclair, *Exp.Math.*, **14**, nr.3 (2005),
math.DG/0506189
- B. Chow, *A Survey of Hamilton's Program for the Ricci flow on 3-Manifolds*, *math.DG/0211266* (2002)
- D. DeTurck, *J. Diff. Geom.*, **18**, 157 (1983)
- <http://www.cactuscode.org> for numerical relativity
- <http://grtensor.org> the site for grTensorII package

Selected references

D.N. Vulcanov - Numerical simulations with Ricci flow on surfaces- A review and some recent results, Annals of Univ. of Craiova, vol. 18, 120-129, 2008

- D.N. Vulcanov -Errors and Convergence in Numerical Simulations of Ricci Flow on 2-Dimensional Surfaces, PROCEEDINGS OF THE PHYSICS CONFERENCE TIM-08, Book Series: AIP Conference Proceedings Volume: 1131 Pages: 143-148 Published: 2009

Acknowledgements

Manny worm thanks to prof.dr. Goran Djordjevic and his team for the kind hospitality and support during my visit in Nis under the auspices of SEENET-MTP network

Special thanks go to H. Rubinstein and D. Garfinkle for their permission to reproduce the pictures from their articles I included here !

This is the ... very end of the show !!!