Numerical Study of Membrane Dynamics in String Theory

CPSC 520 Term Project

Benjamin Gutierrez
Department of Physics and Astronomy
University of British Columbia
November 28th, 2005

http://laplace.physics.ubc.ca/People/benjamin/projects/timelike/

Outline

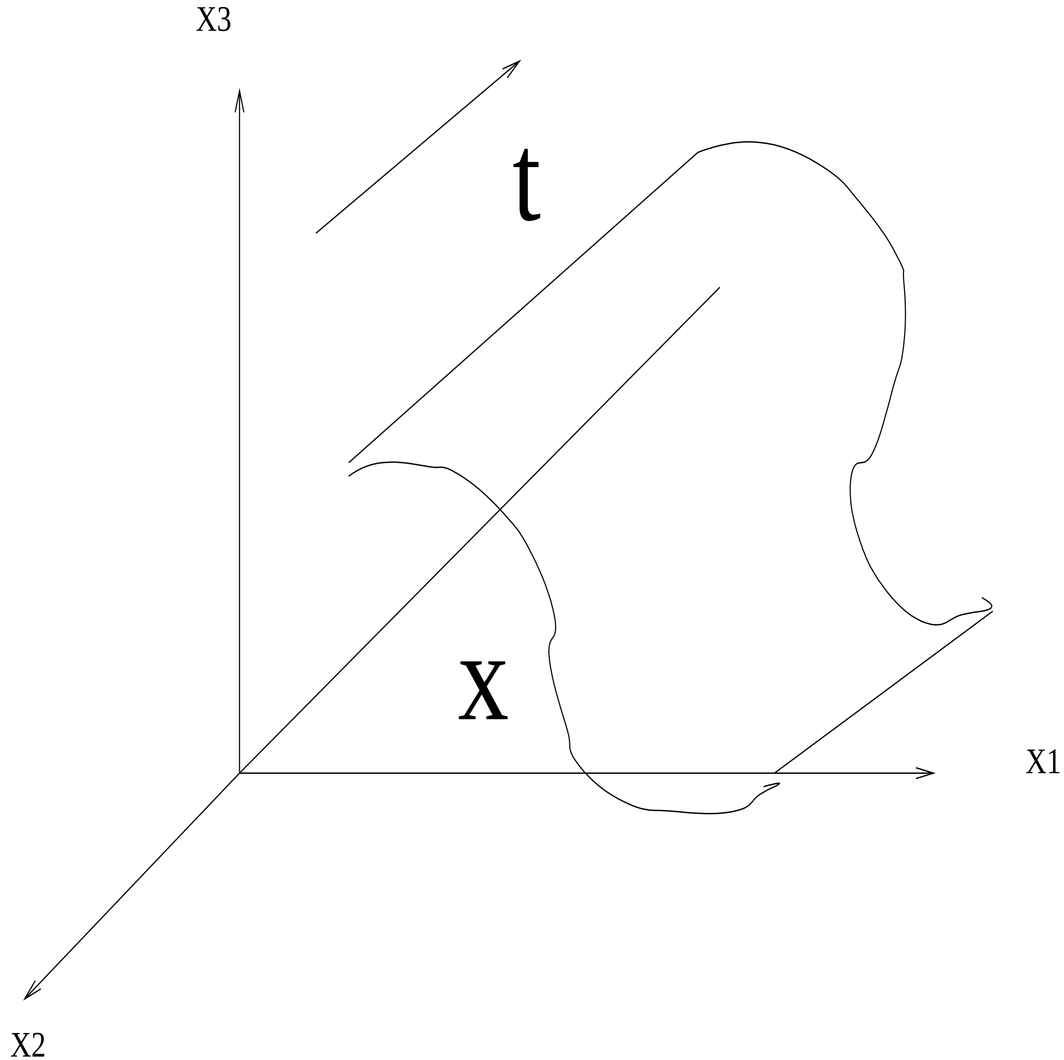
- Motivation: String Theory in a Nutshell
- The Model: Timelike Minimal Surfaces
- The Equations of Motion
- The Numerical Scheme
- Preliminary Results
- To Do List

Motivation

- String Theory in a Nutshell
  - String theory is a model of fundamental physics whose building blocks are one-dimensional extended objects (strings) rather than the zero-dimensional points (particles) that are the basis of the Standard Model of particle physics.
  - We also have branes, membranes and higher-dimensional objects.
- Extra Dimensions?
  - One intriguing feature of string theory is that it predicts the number of dimensions which the universe should possess.
  - string theory allows one to compute the number of spacetime dimensions from first principles (Lorentz invariance)
  - Nowdays this field is very active due to the possible experimental searches for signatures of extradimensions in particle accelerator experiments and tests of Newton's square law at short distances.

The Model: Timelike Minimal Surfaces

- Problem: Embedding of a higher dimensional bulk spacetime into a target space
- Membrane located inside the bulk, where we have an induced metric.
- The membrane is then a minimal (timelike) surface that describes the evolution of the string in time.
- This is relevant for string and M-theory, because these backgrounds are explicitly time dependent.
- In terms of the physics, the timelike minimal surface problem comes up in both "regular" string theory and in so-called m-theory. In the m-theory case, it is the equation governing membrane motion.
- The equations of motion are difficult so we need to solve them numerically
- Baby Model: From the d-brane action we obtain the minimal surface functional by turning off all the physics tensors. .


X3
t
X
X1
X2

Specific Model

- What is a Metric function?

- Consider an embedding of R^{1+n} into Minkowski spacetime R^{1+n+q} given by the functions (fields) f^I, $I = 1, \dots, q$.

- Following a suggestion by Jim Isenberg[1], I intend to solve the equations of motion for this problem which are the EOM's for a scalar field in the background geometry:

$$h_{\alpha\beta} = \eta_{\alpha\beta} + f^I_\alpha f^J_\beta \delta_{IJ} \tag{1}$$

where $f^I_\alpha = \partial_\alpha f^I$. We will solve the problem in $1+1$ dimensions so we have a 2-dimensional "target" surface embedded in a 3-dimensional "bulk" space, so $q = 1$ i.e. $\alpha, \beta = 1, 2 = t, r$ and $I, J = 1$, then $\eta = diag(-1, 1)$. The extra field induces perturbations in the $2d$ metric.

[1] Allen, P., Andersson, Lars and Isenberg, James; Time-like Minimal Surfaces of General Co-dimension in Minkowski Spacetime, private communication.

The Equations of Motion

- Timelike Minimal Surfaces (branes, membranes)

- The equations of motion are the Klein-Gordon equation in this background geometry,

$$\partial_\mu \left[\sqrt{deth} \ \ h^{\mu\nu} \partial_\nu f^I \right] = 0 \tag{2}$$

- We need to calculate

I compute the metric components:

$$\begin{aligned} h_{11} &= \eta_{11} + f_1^I f_1^J \delta_{IJ} && (3) \\ h_{22} &= \eta_{22} + f_2^I f_2^J \delta_{IJ} && (4) \\ h_{12} &= f_1^I f_2^J \delta_{IJ} && (5) \\ h_{21} &= f_2^I f_1^J \delta_{IJ} && (6) \end{aligned}$$

- So we have the metric

$$h_{\mu\nu} := \begin{bmatrix} -1 + f11^2 & f11\, f21 \\ f11\, f21 & 1 + f21^2 \end{bmatrix}$$

- The inverse is

$$h^{\mu\nu} := \frac{1}{det(h_{\mu\nu})} \begin{bmatrix} 1 + f21^2 & -f11\, f21 \\ -f11\, f21 & -1 + f11^2 \end{bmatrix}$$

- with the determinant of the metric

$$det(h_{\mu\nu}) = (-1 + f11^2)(1 + f22^2) - (f11^2\, f21^2) \tag{7}$$

The equations of Motion (Cont)

- Explicitly we have,

$$\partial_t \left[\sqrt{-deth}\;\; h^{t\nu}\partial_\nu f^I\right] + \partial_r \left[\sqrt{deth}\;\; h^{r\nu}\partial_\nu f^I\right] = 0 \tag{8}$$

$$\partial_t \left[\sqrt{-deth}(h^{tt}\partial_t f^I + h^{tr}\partial_r f^I)\right] + \partial_r \left[\sqrt{-deth}(h^{rt}\partial_t f^I + h^{rr}\partial_r f^I)\right] = 0 \tag{9}$$

After performing the calculations in a maple worksheet (see website) we obtain the following partial differential equation for the field:

$$-\frac{\partial^2 f}{\partial t^2} + \frac{\partial^2 f}{\partial x^2} = (\frac{\partial f}{\partial t})^2(\frac{\partial^2 f}{\partial x^2}) - 2\frac{\partial f}{\partial t}\frac{\partial f}{\partial x}\frac{\partial^2 f}{\partial t \partial x} + \frac{\partial^2 f}{\partial t^2}(\frac{\partial f}{\partial x})^2 \tag{10}$$

Numerical Scheme

- Discretization: First i get a first order set of equations with the following changes of variables:

$$\Pi = \frac{df}{dt} \quad \Phi = \frac{df}{dx} \tag{11}$$

Then equation (9) takes the form:

$$\Pi^2\Phi' - 2\Pi\Phi\Pi' + \dot{\Pi}\Phi^2 + \dot{\Pi} - \Phi' = 0 \tag{12}$$

where tilde indicates the derivative with respect to x and the dot wrt to t, as usual.

- So we end up with the following set of equations:

$$\dot{\Pi} = -\frac{(\Pi^2 - 1)\Phi'}{\Phi^2 + 1} + \frac{2\Pi\Phi\Pi'}{\Phi^2 + 1} \tag{13}$$

$$\dot{\Phi} = \Pi' \tag{14}$$

Where the last equation follows from the Schwarz theorem.

Discretization (Cont)

- The I apply a Crank-Nicholson scheme, From eqn (13) we have:

$$\left(\frac{\Pi_j^{n+1}-\Pi_j^n}{dt}\right)=\frac{1}{2}(op^{n+1}+op^n) \tag{15}$$

where each operator is: (space derivatives are centered, 2nd order)

$$\begin{aligned} op^{n+1} &= \left[-\frac{(\Pi^2)_j^{n+1}-1}{(\Phi^2)_j^{n+1}+1}\left(\frac{\Phi_{j-1}^{n+1}-\Phi_{j+1}^{n+1}}{2dx}\right)+\frac{2\Pi_j^{n+1}\Phi_j^{n+1}}{(\Phi^2)_j^{n+1}+1}\left(\frac{\Pi_{j+1}^{n+1}-\Pi_{j-1}^{n+1}}{2dx}\right)\right] \\ op^n &= \left[-\frac{(\Pi^2)_j^n-1}{(\Phi^2)_j^n+1}\left(\frac{\Phi_{j-1}^n-\Phi_{j+1}^n}{2dx}\right)+\frac{2\Pi_j^n\Phi_j^n}{(\Phi^2)_j^n+1}\left(\frac{\Pi_{j+1}^n-\Pi_{j-1}^n}{2dx}\right)\right] \end{aligned}$$

and from eqn (14) we have:

$$\left(\frac{\Phi_j^{n+1}-\Phi_j^n}{dt}\right)=\frac{1}{2}\left[\left(\frac{\Pi_{j-1}^{n+1}-\Pi_{j+1}^{n+1}}{2dx}\right)+\left(\frac{\Pi_{j-1}^n-\Pi_{j+1}^n}{2dx}\right)\right] \tag{16}$$

Kreiss-Oliger Dissipation

- Most FD schemes can not propagate acurrately high-frequency components of the solution
- Dissipation is a low-pass filter applied to the grid function, supressing high-frequency components in the numerical solution
- High frequency means wavelenghts on the order of the mesh spacing h
- Moreover, it is the high-frequency components that tend to exhibit the fastest growth in an unstable scheme
- A popular dissipation technique is the Kreiss-Oliger method, whereby a term of the form
$$D_{k_0}\hat{u}_i = \frac{\epsilon}{16}\left(\hat{u}_{i-2} - 4\hat{u}_{i-1} + 6\hat{u}_i - 4\hat{u}_{i+1} + \hat{u}_{i+2}\right) \tag{17}$$
is added to the differential equation. ϵ is a positive, adjustable parameter controlling the amount of dissipation added, and must be less than 1 for stability.
- The taylor expansion of this operator about x_i shows that adding it to a second-order accurate FDA should not affect the convergence properties of the

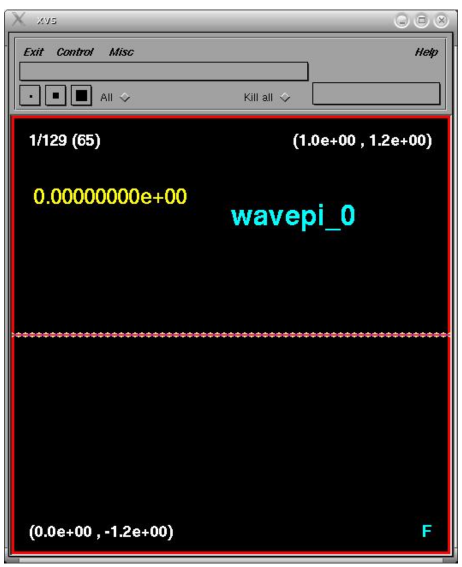
scheme (its 4th order).

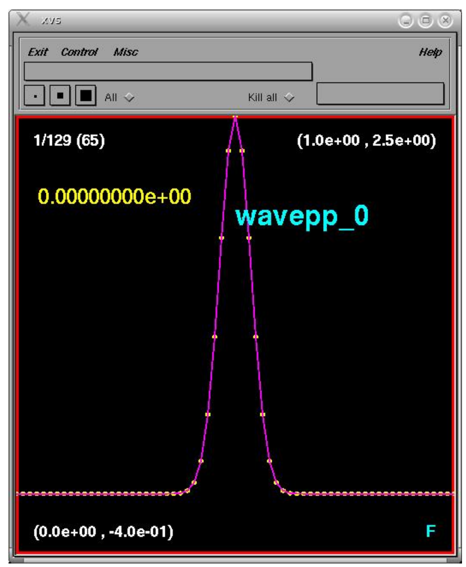
Preliminary Results

- The solution of this equation tells us how the string, or the brane if we add more dimensions, propagates in time i.e. the dynamics of this geometry.

- For a low amplitude limit of the initial profile, say $A = 0.0001$, I expect a wavelike solution behaviour

- Its reasonable to expect a solitionic solution. For large amplitudes the non-linearities dominate





Wave Equation

Final Comments

- Work to do:
  - Perform an independent residual evaluation
  - Explore different initial profiles and different amplitudes
  - Determine if the asymmetry between right and left that I see is a numerical bug
  - Try a curved geometry and maybe two spatial dimensions (lots of algebra)
- All the material in this talk is located in
  `http://laplace.physics.ubc.ca/People/benjamin/projects/timelike`
- The RNPL website
  `http://laplace.physics.ubc.ca/People/matt/Rnpl/index.html`

RNPL

- Now I code the equations in RNPL (Rapid Numerical Prototyping Language
  - This is a high purpose language developed by Matthew Choptuik and Robert Marsa, fort at UT Austin and then here at UBC.
  - The objective is to have a rapid prototyping tool for time dependent systems of PDEs
  - Focused on numerical relativity but flexible to solve most types of PDEs
  - The RNPL uses a parser (lex) to generate code in Fortran or C
  - Then using symbolic manipulation for point-wise Newton-Gauss-Seidel relaxation to generate update functions, for explicit and implicit schemes
  - The user defines the operators, the coordinates, the gri functions, the initial profile and RNPL generates and compiles the code for the update and residual functions

How RNPL code looks like