Numerically Generating Novikov Coordinates

March 7, 1996

Consider a set of timelike observers each with a different value of the Schwarzschild radial coordinate r_m at $\tau = 0$. At $\tau = 0$ each observer has a value,

$$R = \sqrt{\frac{r_m}{2M} - 1}, \tag{1}$$

of the Novikov radial coordinate, R. They retain this value of R as they plung in towards the singularity. The set (τ, R) are the Novikov coordinates which are related to the standard Schwarzschild coordinates parametrically.

$$r = M(R^2 + 1)(1 + \cos\eta), \tag{2}$$

$$\tau = M(R^2 + 1)^{3/2}(\eta + \sin\eta) \tag{3}$$

and

$$t = 2M \ln\left|\frac{R + \tan\frac{\eta}{2}}{R - \tan\frac{\eta}{2}}\right| + 2MR\{\eta + \frac{1}{2}(R^2 + 1)(\eta + \sin\eta)\} \tag{4}$$

To generate $r = r(\tau, R)$ and $t = t(\tau, R)$ then requires that we solve for $\eta(\tau, R)$. Rewriting equation (3) we have that

$$\eta + \sin\eta = \frac{\tau}{M(R^2 + 1)^{3/2}}. \tag{5}$$

Solving this equation for η for a given τ and R will give us back $r(R, \tau)$ and $t(R, \tau)$. We use Newtons method to solve (5). The algorithm proceeds as follows: Define

$$F[\eta] \equiv \eta + \sin\eta - \frac{\tau}{M(R^2 + 1)^{3/2}} \tag{6}$$

and its derivative with respect to η is

$$dF[\eta] \equiv 1 + \cos\eta. \tag{7}$$

Given R,τ and M we would like to obtain an η that satisfies $F[\eta] = 0$. The algorithm used to solve for η is as follows:

Start with some initial guess η_0.

Begin newton iterations

Evaluate equation (6) or $F[\eta]$ with given values of R,τ and M.

Evaluate equation (7) or $dF[\eta]$.

Calculate $\delta\eta = -F[\eta]/dF[\eta]$.

If $||\delta\eta|| >$ Stopping Criterion then

Update $\eta_0 = \eta_0 + \delta\eta$.

If $||\delta\eta|| <$ Stopping Criterion then break out of loop.

Set $\eta = \eta_0$.

Our approach then is to set up a 2D grid in τ and R and for each set of values (τ, R) compute η using the algorithm just given.

To generate tables for η, r and t the following routine is used. It uses the Newton algorithm to solve for η and evaluates $r = r(\tau, R)$ and $t = t(\tau, R)$.

```
gentable(Rstar, tau, eta,r,t,dR, dtau, Rmin, Rmax, tmin, tmax, stp,
itmax, mass, NT,NR)
```

Where the variables are declared as:

```
integer NR, NT, itmax
real Rstar(NR), tau(NT)
real r(NT,NR), t(NT,NR), eta(NT, NR)
real mass, dR, dtau, stp
real Rmin, Rmax, tmin, tmax
```

An example is provided in `driver.f` in the links that follow.