Frédéric Lassiaille June 2010

VALIDATION OF DARK MATTER EXPLANATION

This paper is following the article [6]: <<A solution for the "dark matter mystery" based on Euclidean relativity>>. It validates the solution for dark matter mystery explained in the above article.

1. INTRODUCTION

The article [6] presents a dark matter explanation. Confrontation with experiments is done only through an estimated law of matter density, and for a supposed galaxy. Therefore we hope to execute those speed profiles calculations by using, this time, real matter density values.

The program below is calculating the curve for the star's velocities around the center of the NGC 3310, NGC 1068, NGC 157, NGC 7541, and NGC 7331 galaxies.

The result is the following :

- same shape between measured curves and theoretical curves,

- calculated rotational speeds are close to measured values: 10 % precision for NGC3310 and 63 % precision for NGC1068,

- explanations for "out of range" measured values of NGC 3310, by dynamic effects generated by a theoretical strong gravitational wave.

This calculation is based on the corrected Newton's law explained in [6]. It uses information coming from [1,2,3,4], for the NGC 3310, NGC 1068, NGC 157, NGC 7541, and NGC 7331 galaxies.

F.Lassiaille, may 2010

2. CALCULATIONS

1. CALCULATION OF THE GRAVITATIONAL FORCE

In the program below i have used the following formula :

$$e = \sqrt{(8R/x)} \; (\sqrt{r} + \sqrt{rho0}) / (\sqrt{r} + \sqrt{rho}).$$

It is a reformulation of : $e = \sqrt{(8R/x)} / (k + g(x))$, which is the equation given in [6], chapter 6.

We have also $g(x) = 1/x$ in the article, but here it is not the case because $g(x)$ is taken now from the NGC 3310 figure.

In those equations :

- <<rho0>> is the solar system matter density in our milky-way galaxy. I used rho0 = 003 * M0 / (YL**3), where YL stands for 1 year-light, and M0 is the sun mass.

- <<rho>> is the matter density value in NGC 3310, function of the distance, <<x>>, from the galactic center.

- <<R>> is the Schwarzschild ray of the galactic center for NGC 3310. R = MG/c², with M, the galaxy center mass : M = 0.07 * 10**23 * M0. This value of M is estimated with the help of the integral mass profile of NGC 3310 given in the article ("Fig 5 : Integral mass profile …").

- <<r>> stands for the square of the extra-galactic luminous contributions. It is calculated with the help of rho0 and G'/G (gravitational constant outside and inside the galaxies, respectively). See below.

- the term <<$\sqrt{r} + \sqrt{rho0}$>> is coming from the fact that we must have the classical Newton's law when we are in same case as our solar system. For getting this term i suppose that the evolution of matter density is weak, in the solar system, in front of rho0 (not only rho0, but also its variation, must be used when calculating the corrected Newton's law).

I calculated the <<r>> value this way.
G'/G = 10, where G' is supposed to be the extragalactic gravitational constant for the studied galaxy (supposed to be the same, just outside the studied galaxy, and just outside the milky-way).

Hence, $\sqrt{(rho0/r)} = Si/So = \sqrt{(G'/G)} - 1 = 2.16$. For this equation, refer to [6], chapter 7.
Then :
$$r = (So/Si)^2 \, rho0 = [\sqrt{(G'/G)} - 1](-2) \; rho0$$

F.Lassiaille, may 2010

2. CALCULATION OF THE ROTATIONAL SPEED

The issue with the calculated force profile above is that there are as much positive (attractive) as negative (repulsive) values. Hence, the most natural way to calculate the rotational speed of the stars is the following. It is to calculate it only for positive values of the gravitational force F, using the centrifugal force formula: $<<v = \sqrt{(F\ x/m)}>>$.

This is done when running the program below with the ProgMode parameter set to 0.

The global shape of the resulting curves is not very interesting since the negative (repulsing) values of the force are forced to 0. (Of course this is mandatory in order to calculate something correct with the centrifugal force equation above). Hence the corresponding values for the speed are also forced to 0 in this case.

But the interesting result with these curves is the calculation of the absolute speed values when the force is positive (attractive). This calculation is noticeably precise for distances corresponding to the maximums of the force profiles. Of course, those calculations can be done only when each parameters are known. This is the case only with the NGC 3310 and the NGC 1068 galaxies.

The results of these calculations are described on chapter 4 ("comments on galaxy curves").

3. CALCULATION OF THE ROTATIONAL SPEED USING A SMOOTHED FORCE

For NGC 3310, for example, the first calculated curve for the speed v is "moving" very fast along the y axis. That is because the value of F, the calculated gravitational force, is changing very fast and strongly along the x axis. In fact the speed cannot change so fast because of inertial reasons. There are, roughly, as much positive (attractive) values as negative values. This is why the following usual formula, $v^2 = F\ x\ /\ m$, is not correct in this case, because this formula supposes a positive (attractive) value of F.

That is why i simulated roughly the somewhat complicated dynamics here. For that i simply smoothed the Force profile, with a "FSmoothFactor" (same name in the MAPLE program). Also i raised it up with a "F_pull" constant value, in order to get only positive values for F.

The FSmoothFactor value has been fitted in order to get the best resulting compared speed values for the 2 maximum of the NGC 3310 galaxy speed curve.

After that the speed v is calculated with : $<<v = \sqrt{(F\ x/m)}>>$.
Since the gravitational force was raised up with a value F_pull, i substract to v the term $<<sqrt(F\_pull\ x)>>$ after calculating the speed v ($<<x>>$ is he distance from the galactic center).

The exact "smoothed" calculation of v from F is :

F.Lassiaille, may 2010

F_mean = mean(F)   arithmetic mean of F in plotting range of x.
Fs = (F-F_mean) FSmoothFactor   "smoothed" force.
Fup = F_pull – F_mean FSmoothFactor   raising up constant.
$v = \sqrt{((Fs + F\_pull) x / m)} - \sqrt{(Fup\ x / m)}$ if Fup >0
$v = \sqrt{((Fs + F\_pull) x / m)} + \sqrt{(-Fup\ x / m)}$ if Fup <0

with F_pull a constant calculated with the help of the FSmoothFactor parameter, and max, min and mean of F.

This is to be compared with the non smoothed calculation :

$v = \sqrt{(\max(F, 0)\ x / m)}$ negative values forced to 0

I note that this "smoothing" of the force works well, noticeably for NGC 7541. Hence for this galaxy the force is always repulsive, for each value of x, the distance from the galactic center. But repulsive forces are the worst case for this "smoothing".

Hence, a star which is located in a place where exists a strong repulsive force, will be "pushed" away from the galactic center. This trajectory modification will decrease the rotational speed of this star. On the contrary, a low repulsive force will modify slowly the rotational speed of the star.

Therefore, sure this calculation is strongly unperfect. But this artificial calculation is justified, roughtly, qualitatively. Moreover, we note that the galaxy curves proves that it works well for 5 galaxies upon 5.

I execute exactly the same program but with the NGC 1068, NGC 157 NGC 7541, and NGC 7331 galaxies. Again i obtain nearly the same relative values and exactly the same shapes.

This calculation of the speed profile is done when running the program with the ProgMode parameter set to 10.

4. CALCULATION OF THE ROTATIONAL SPEED USING SIMULATION

A more correct calculation of the speed, done without modifying the gravitational force, is executed by running the program below with ProgMode parameter set to 24.

In this case the program executes a simulation of the trajectories of the stars. Each simulated star is starting on the same "Ox" axis and at the same speed. This axis is a straight line which encounters the origin: (x,y) = (0,0). This origin is located on the galactic center. For each simulated star, the program stops the star trajectory simulation as soon as it reaches an "Or" axis, and calculates the rotational speed of this star at this

F.Lassiaille, may 2010

instant. The "Or" axis is a straight line starting from the origin, and which makes an "a" angle with the Ox initial axis.

This "a" angle, the initial speed, and some more parameters are set depending of the galaxy the program is working on.

F.Lassiaille, may 2010

3. RESULTS

The theoretical speed profiles below are calculated with a "smoothed" gravitational force (refer to the paragraph 3 of chapter 2 above, ProgMode set to 10).
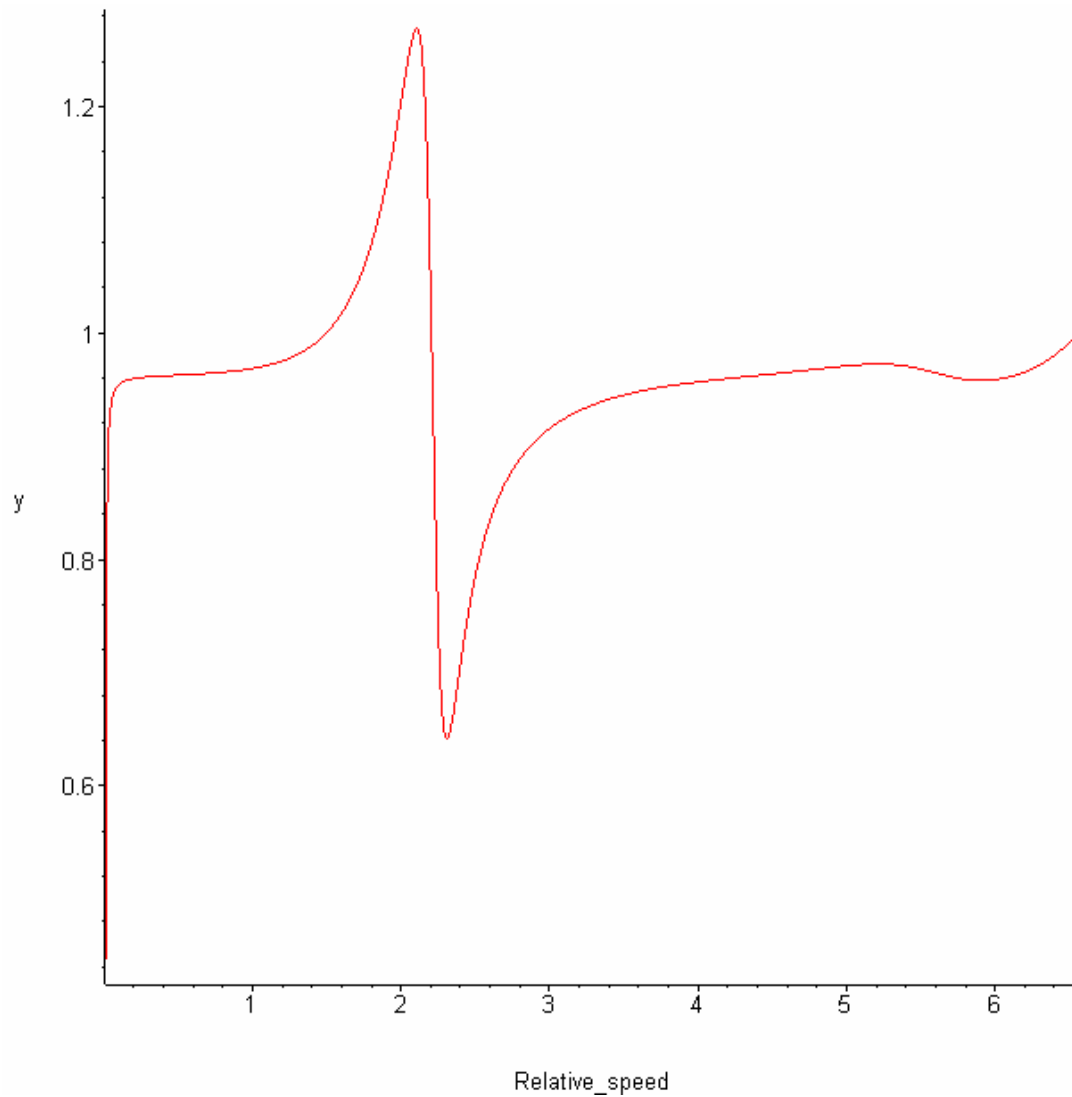
1. NGC 3310

Figure 1 : Theoretical speed curve of the NGC 3310 galaxy.

The y ordinate on the figure above is a relative speed. The unit of the x ordinate is kpc.

F.Lassiaille, may 2010

Figure 2 : Measured speed curve of the NGC 3310 galaxy.

2. NGC 1068

Figure 3 : Theoretical speed curve of the NGC 1068 galaxy.

The y ordinate on the figure above is a relative speed. The unit of the x ordinate is kpc.
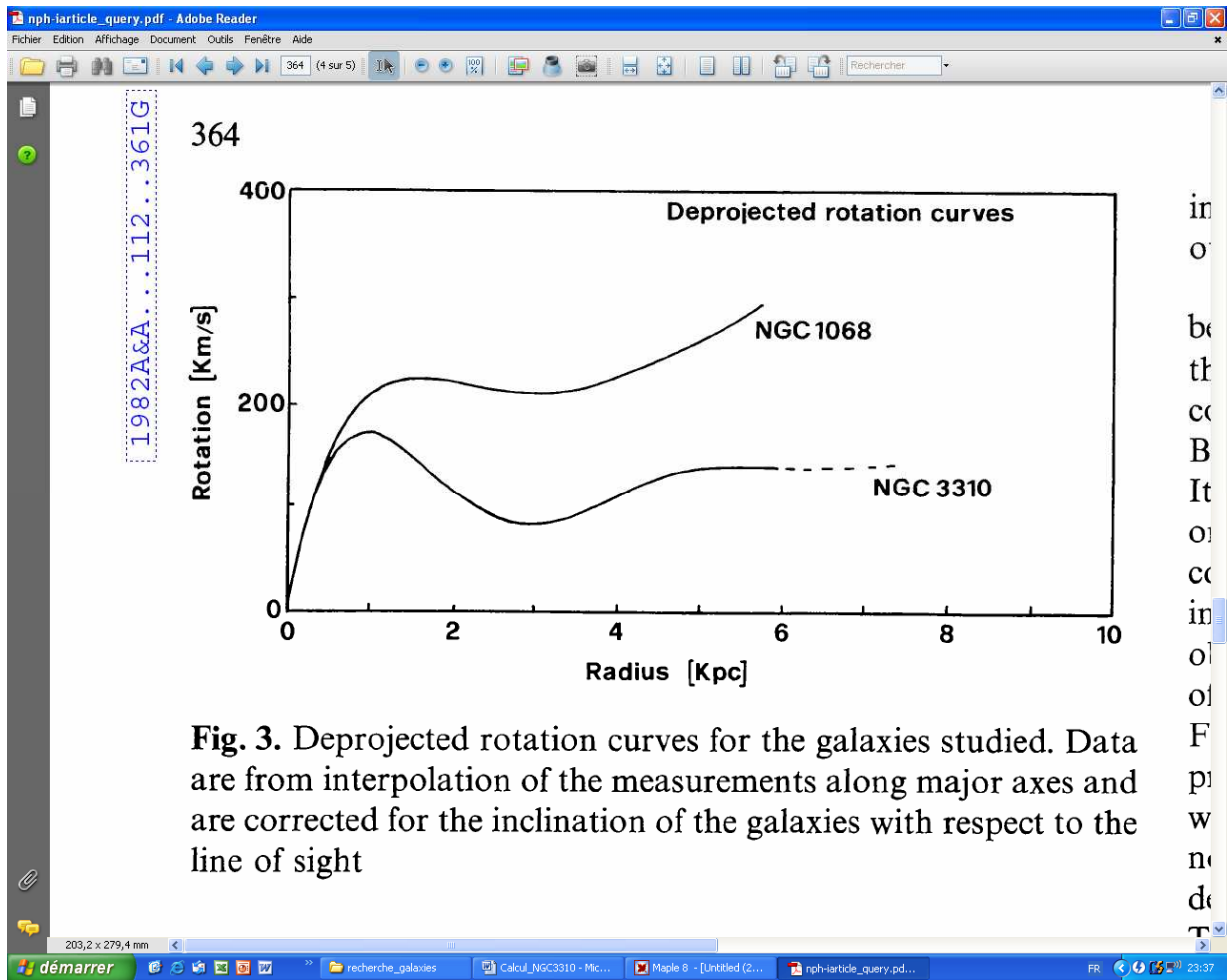
F.Lassiaille, may 2010

Figure 4 : Measured speed curve of the NGC 1068 galaxy.

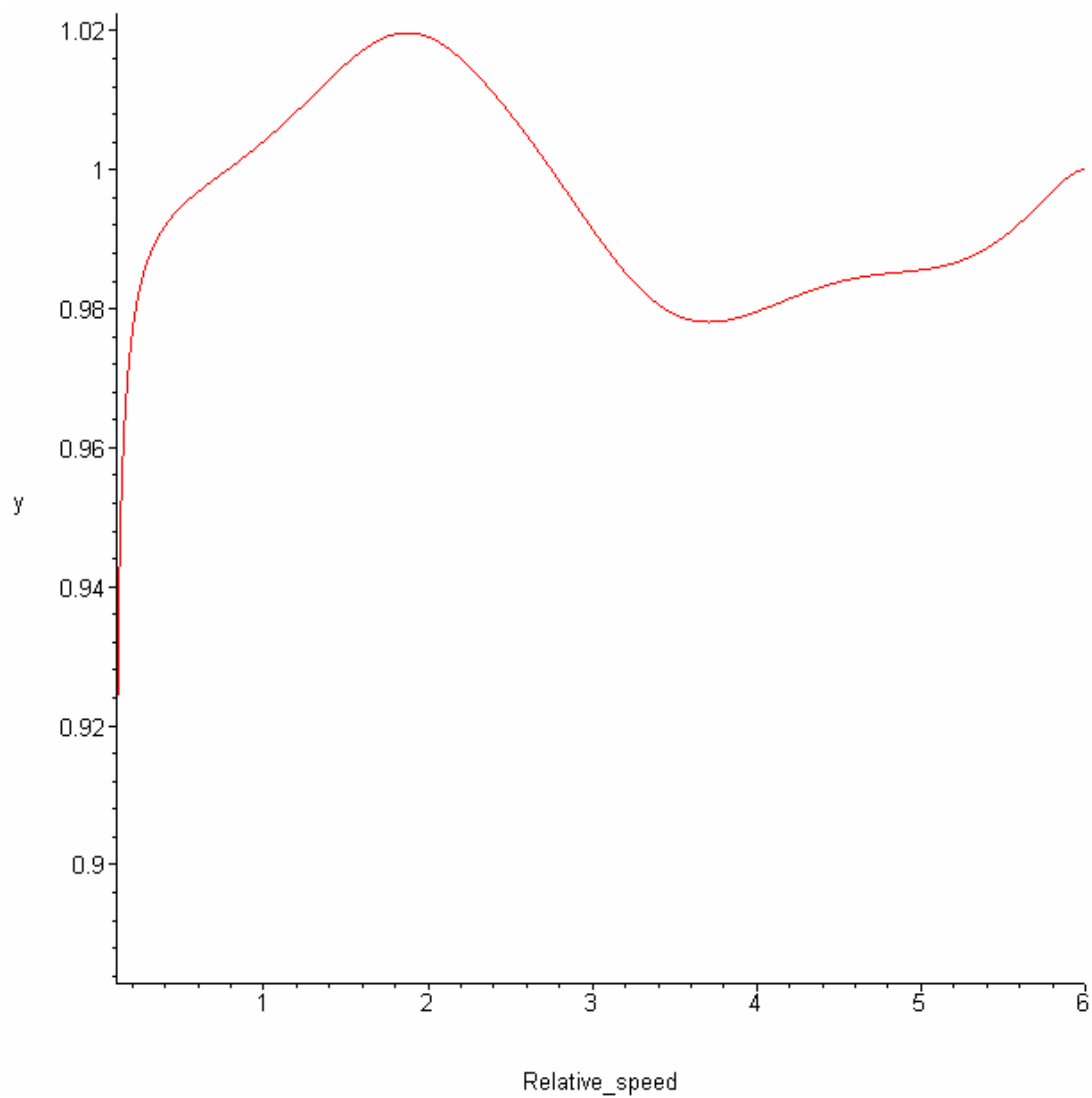F.Lassiaille, may 2010

3. NGC 157

Figure 5 : Theoretical speed curve of the NGC 157 galaxy.

The y ordinate on the figure above is a relative speed. The unit of the x ordinate is kpc.

F.Lassiaille, may 2010
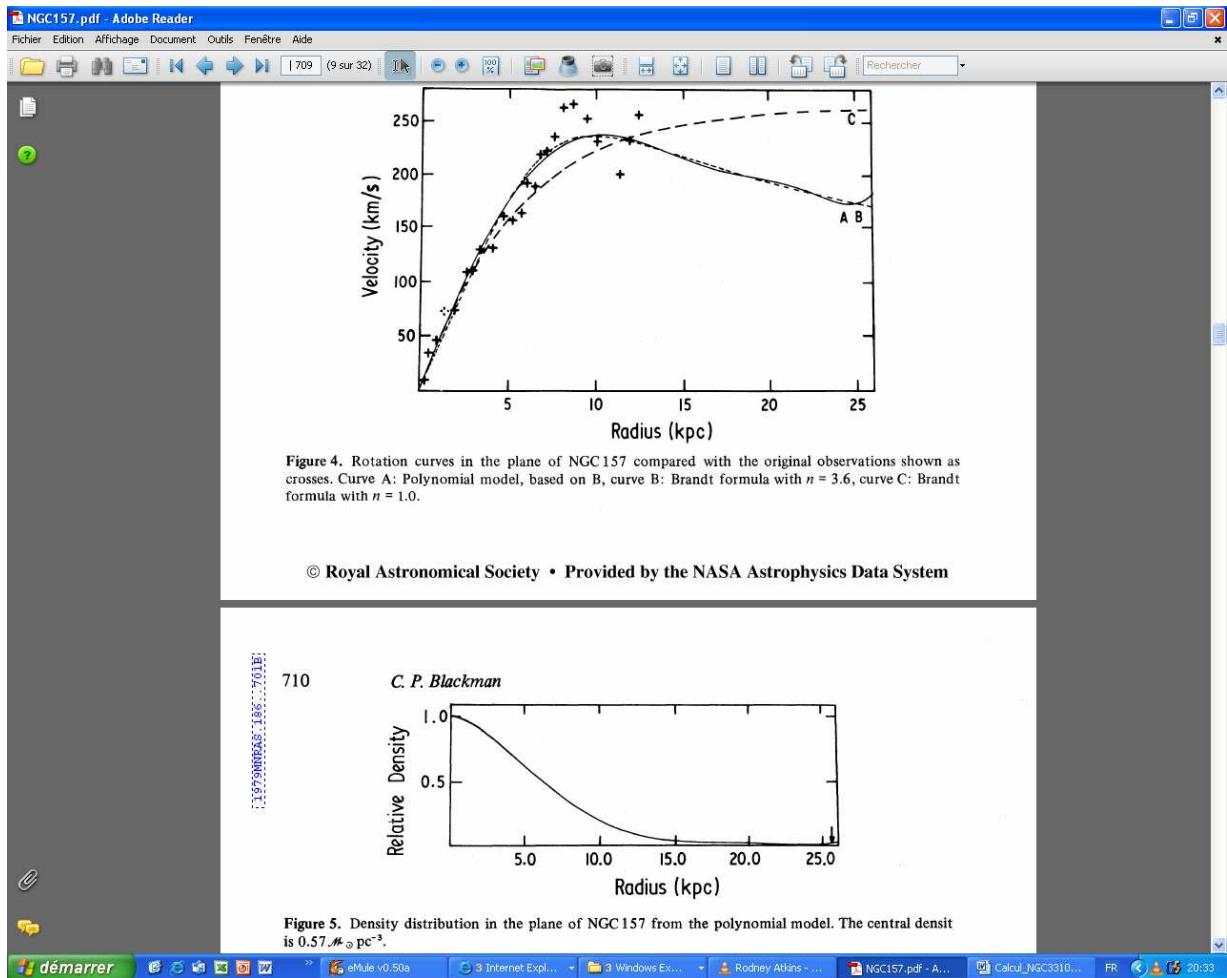
Figure 6 : Measured speed curve for NGC 157 (above in plain line).

F.Lassiaille, may 2010

4. NGC 7541

Figure 7 : Theoretical speed curve of the NGC 7541 galaxy.

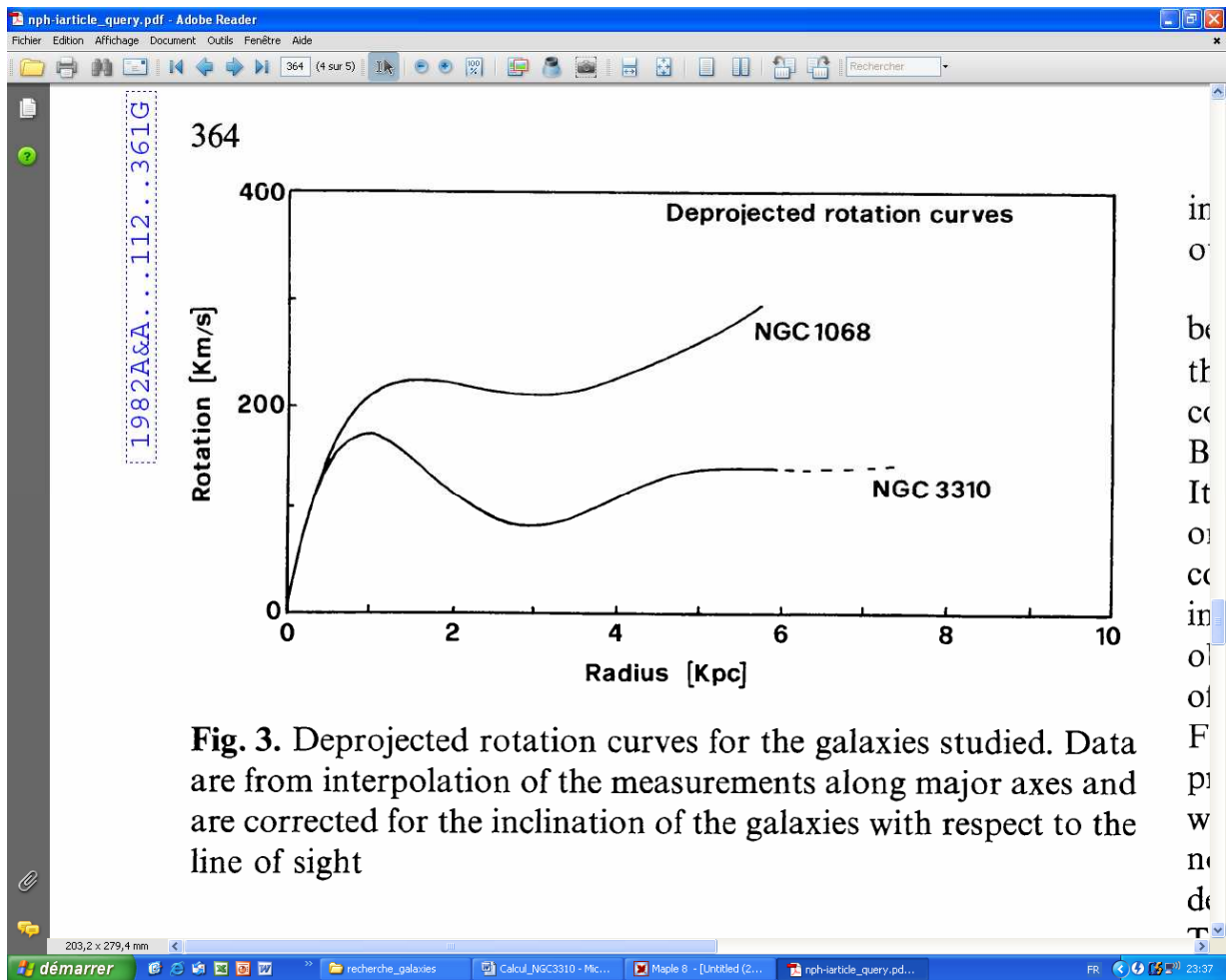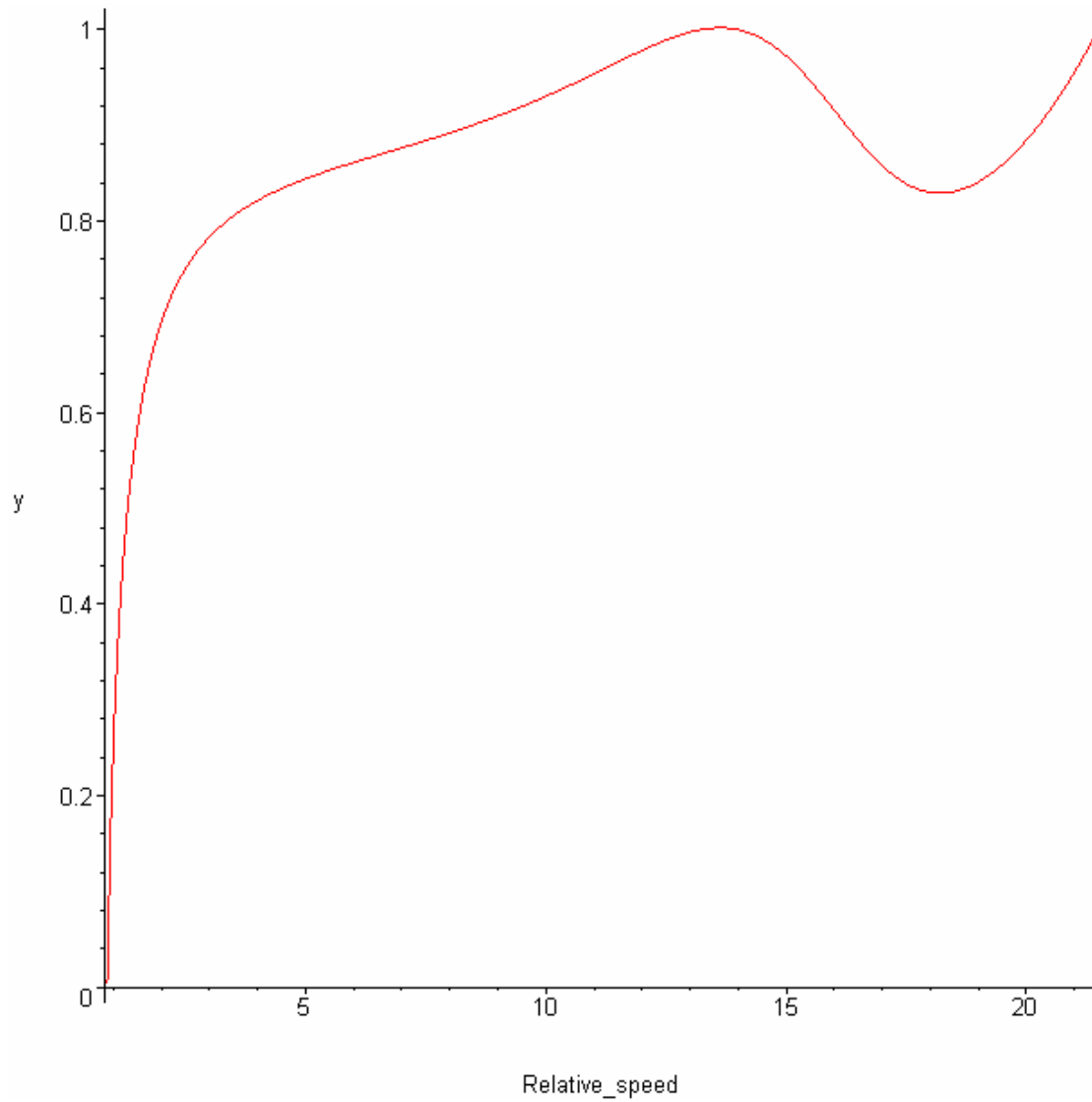The y ordinate on the figure above is a relative speed. The unit of the x ordinate is kpc.

F.Lassiaille, may 2010

FIG. 3. Circles, observed rotation curve of NGC 7541; solid line, mo[
rotation curve; dashed line, surface-density profile.

Figure 8 : Measured speed curve for NGC 7541.

5. NGC 7331

Figure 9 : Theoretical speed curve of the NGC 7331 galaxy.

The y ordinate on the figure above is a relative speed. The unit of the x ordinate is kpc.
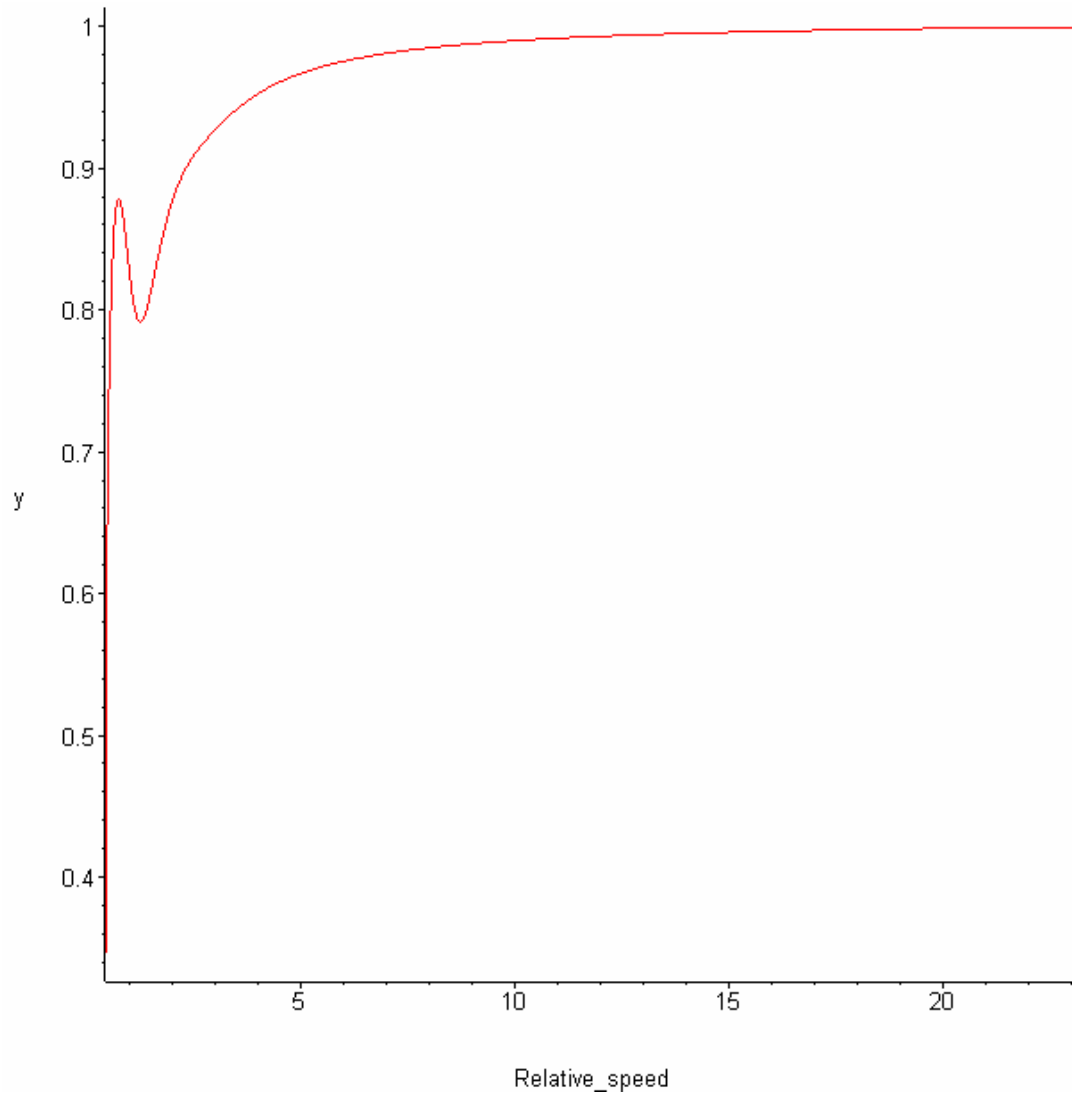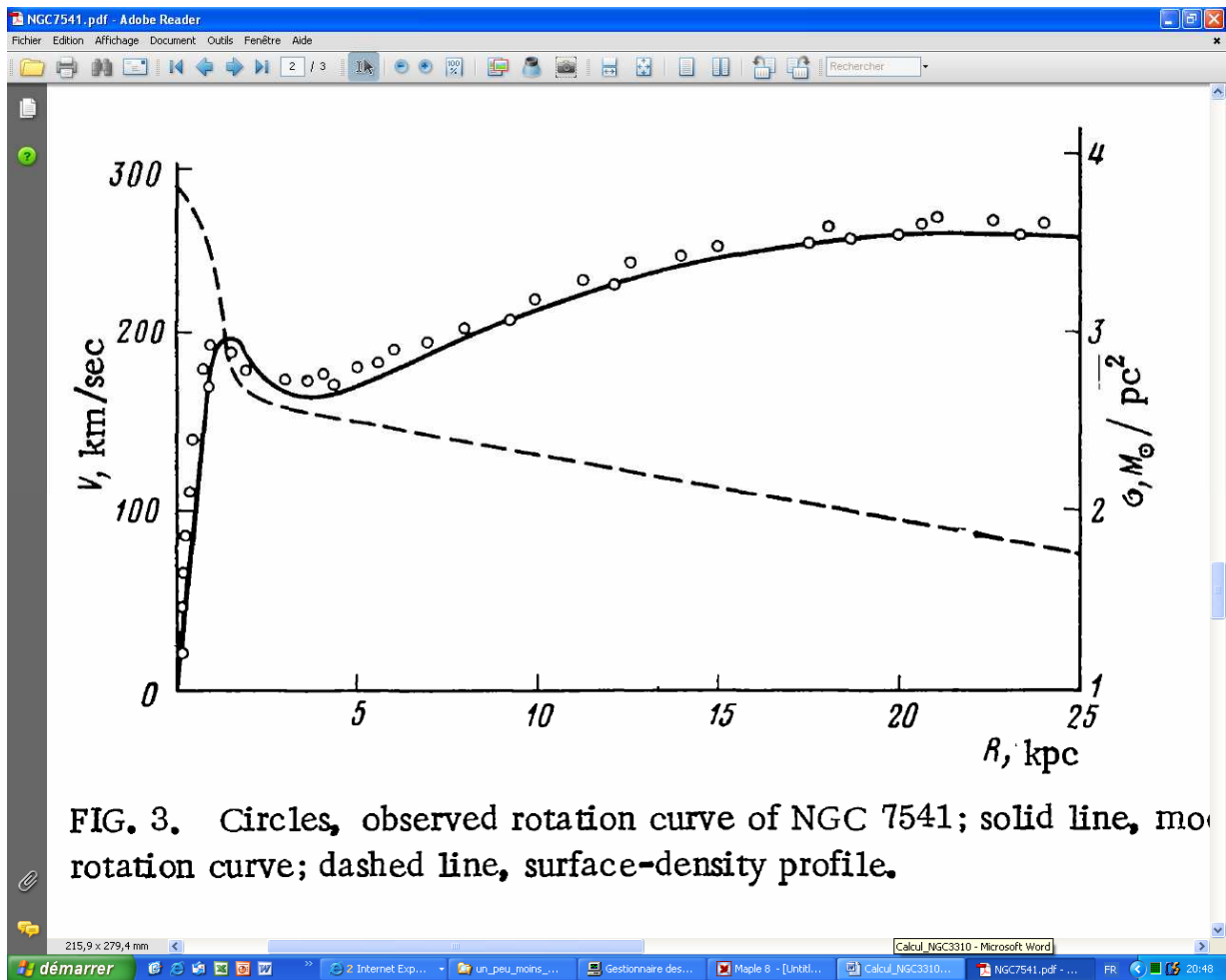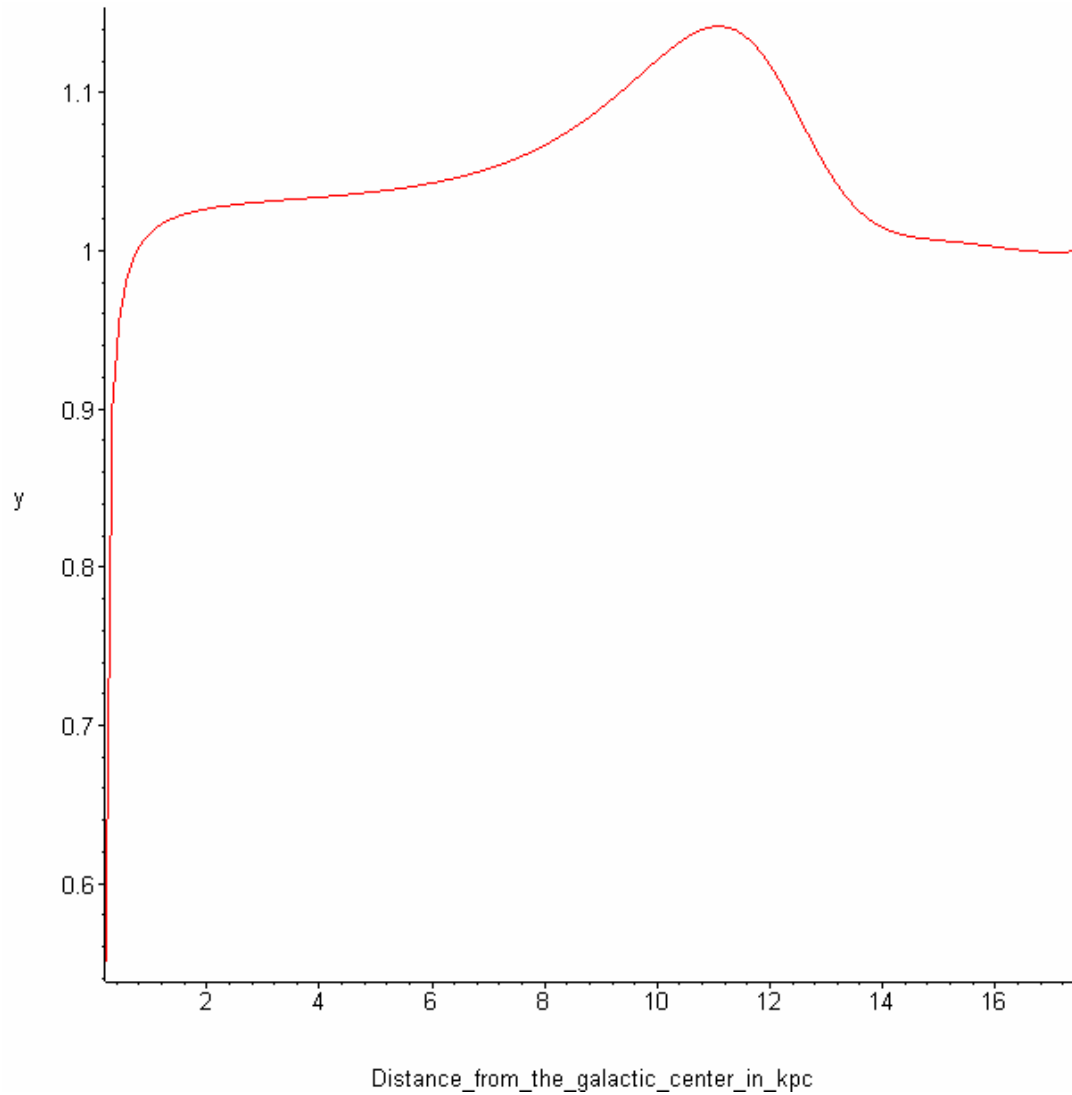
F.Lassiaille, may 2010
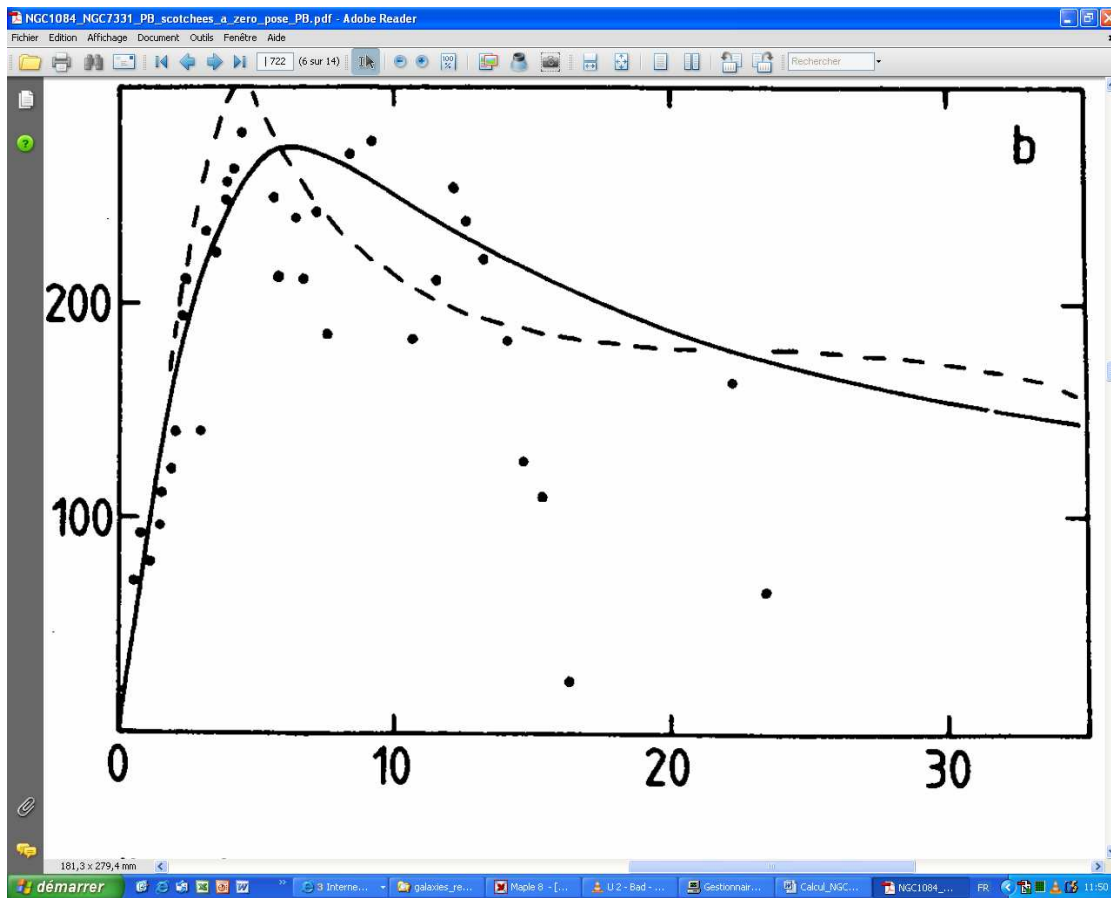
Figure 10 : Measured speed curve for NGC 7331.

F.Lassiaille, may 2010

4. COMMENTS ON GALAXY CURVES

1. NGC 3310

Figure 2 represents the measured curve for NGC 3310, to be compared with my theoretical curve on figure 1. The 2 curves have the same shape, except for distances around 2 kpc.

But this theoretical curve is calculated when supposing no change in the speed profile as time goes by (static profile). But in reality there is a dynamic profile. That is to say that the curve of the speed of the stars do not stay the same. The speed profile is a function of time and the axis angle. The simulation of the trajectories of the stars retrieves clearly this behaviour. This simulated speed profile is given on figure 11.

NGC_3310_theoretical_speed_curve
Figure 11 : Theoretical speed curve for NGC 7331 (simulation)

F.Lassiaille, may 2010

Hereafter are the parameter values which has been used in the program for this simulation (ProgMode = 24).

```
NBSTEP_NGC3310 := 75 :        # Nb of steps for star simulation.
NBTRAJ_NGC3310 := 3000 :      # Nb of stars for star simulation.
V0_NGC3310 := 150*10**3/kpc:  # Initial speed for star simul.
DY_NGC3310 := 1/50 :          # Sampling space period in kpc.
AXANG_NGC3310 := 0.33 :       # Angle of axis for profile in rad.
```
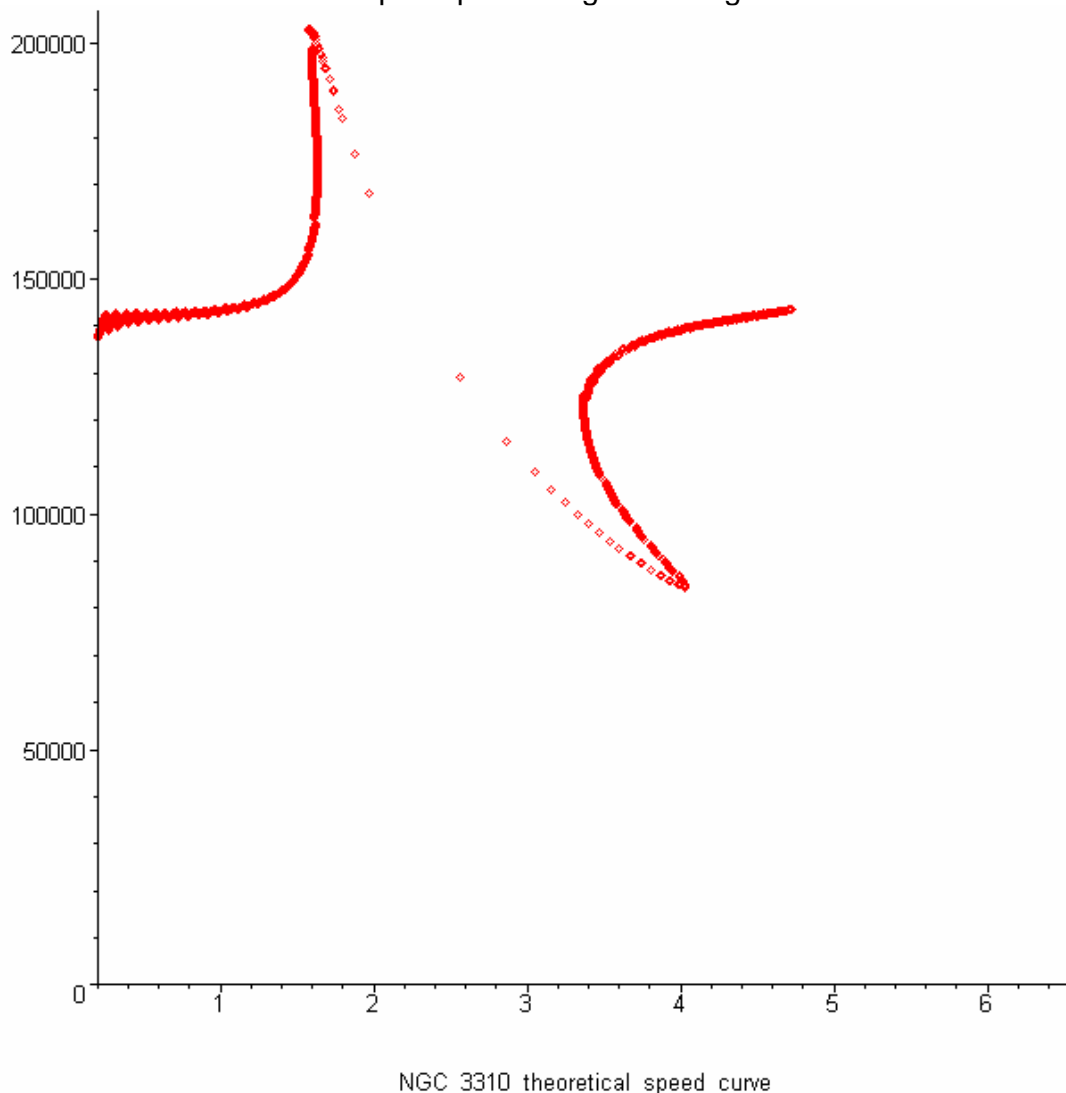
The maximum and minimum speed values, for 1 and 4 kpc, respectively, are also retrieved on the measured curve of [1], figure 2 b. But they are not all taken in account for the polynomial interpolation of [1].
On figure 2 b of [1], the triangles in parenthesis, for the maximum, (25 arcsec, 130 km/s) corresponds to measured speed values which has not been used for the final curve of the measured speed profile. But on the theoretical curve of figure 11, those values are clearly noticed. (130 km/s on figure 2 b of [1] corresponds roughtly to 175 km/s on figure 3 of [1], and corresponds roughly to this maximum value of figure 11 above). Moreover, the explanation for those "out of range" values is now clear. They are coming from the strong gravitational wave around 2 kpc. This wave can be seen on the figure below.

F.Lassiaille, may 2010
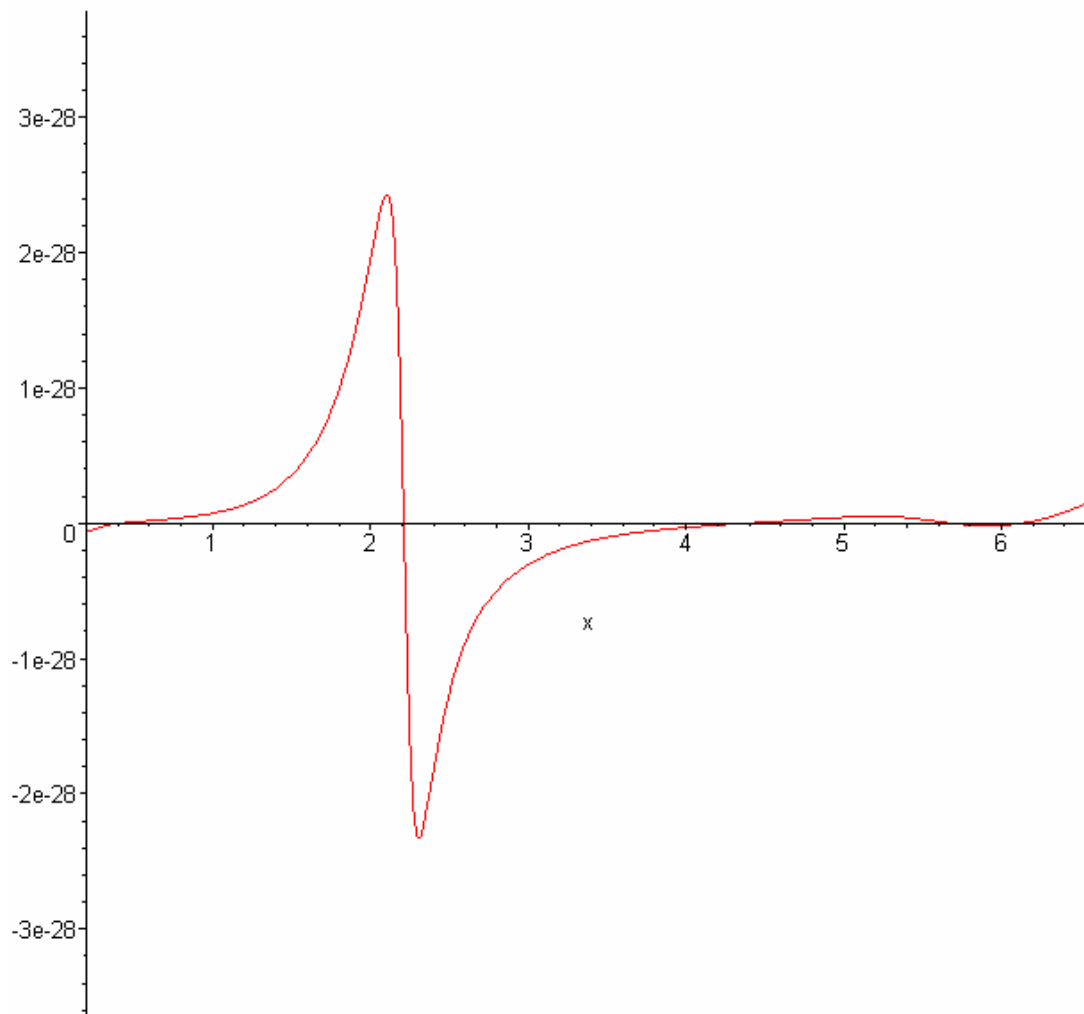
Figure 12 : Gravitationnal force curve for NGC 7331

On figure 12, the force is positive when the galactic center is attractive.

The theoretical absolute speed values may be calculated for NGC 3310, as well as for NGC 1068. Hence, i have for those galaxies an estimated value of the galactic center mass. This estimation is done using [1] (Figure 5 "Integral mass profile…"). Using this curve, I can estimated a value of 0.7 10^10 sun mass.

Those speed values are calculated when running the program with the option "ProgMode" set to 0, which means that no smoothing is done on the gravitational force curve. The only modification done, on the gravitational force, in this case, is setting to 0 the negative values.

The speed curve which is obtained in this case is shown below for the NGC 3310.

F.Lassiaille, may 2010
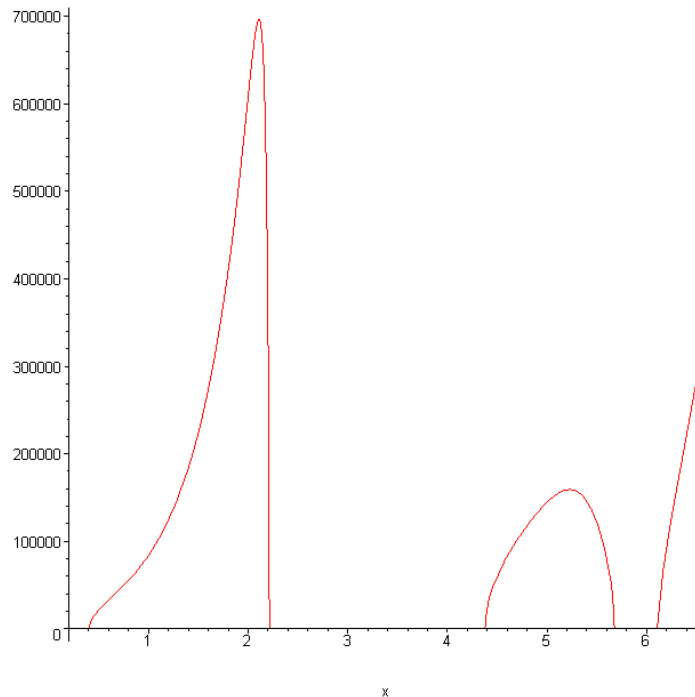
Figure 13 : Theoretical speed curve of NGC 3310 without smoothing

The theoretical speed values are precise only for the maximums of the speed profile. Hence, this calculation is done using the equation v = √[F x / m]. But this formula supposes that the speed profile do not change as time goes by. This is the case, roughly, for the maximum of the speed profile.

For NGC3310, there are 2 maximums :

> - for the first maximum (on the left of the figure), the speed value is in the range [20-700] km/s. The corresponding measured value is 175 km/s.

> - for the second maximum (on the right of the figure), the speed value is equal to 150 km/s. The corresponding measured value is 137 km/s, which is 10% of precision.

2. NGC 1068

For this galaxy, and only for this galaxy, i used extrapolated values (for x greater than 5.5 kpc).

If i only use the values available on the NGC 1068 article, without extrapolated values, then i get only the first "maximum" on the left of the measured speed curve, and the speed stays quite constant as a function of <<x>>.

F.Lassiaille, may 2010

The extrapolated values supposes that the matter density is decreasing after 5.5 kpc. With those extrapolated values i obtain the figure 3.

The non smoothed curve for NGC 1068 galaxy is the following.
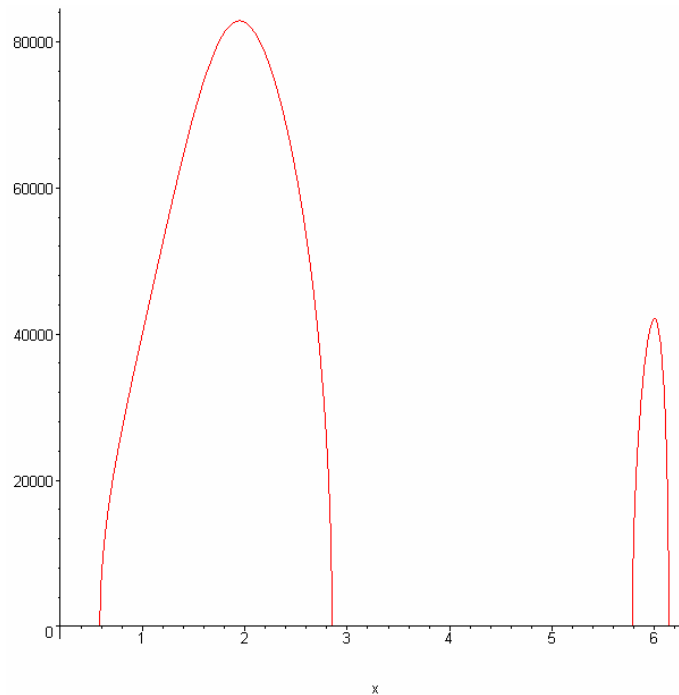
Figure 14 : Theoretical speed curve of NGC 1068 without smoothing

The maximum value above is 82 km/s, compared with the measured value of 220 km/s, that is 63%.

The second maximum on the right cannot be compared to the measured values, because it comes from the extrapolated values.

We can modify the galaxy parameters in order to retrieve the exact measured speed value. With a galactic center mass for NGC 1068 equal to 7.5 10^10 M0 (in place of the estimated value from [1], which is 10^10 M0), we retrieve the estimated speed value of 230 km/s for the first maximum.

The simulation of the speed of the stars, with this galaxy, gives the following result.

F.Lassiaille, may 2010

NGC_1068_theoretical_speed_curve

<u>Figure 15 : Theoretical speed curve of NGC 1068 with simulation</u>

Hereafter are the parameter values which has been used in the program for this simulation (ProgMode = 24).

```
NBSTEP_NGC1068 := 480 :        # Nb of steps for star simulation.
NBTRAJ_NGC1068 := 3000 :       # Nb of stars for star simulation.
V0_NGC1068 := 80*10**3/kpc:     # Initial speed for star simul.
DY_NGC1068 := 1/400 :           # Sampling space period in kpc.
AXANG_NGC1068 := 0.16 :        # Angle of axis for profile in rad.
```

Once again, the profile has the same shape as the measured one. The speed values on the left, before 1 kpc, are too high, compared to the measured values. But this simulation is done using a constant initial value. Of course this is not the case in reality.

3. NGC 157

For this galaxy, i could not have any estimated nor observed value for the galactic center mass. Hence, the galactic center mass of NGC 3310 is used, and a matter density with similar values to the NGC 3310 ones.

The global shape is retrieved. The figures 5 and 6 shows the same global shape.

But as for the NGC 3310, the speed "wave" is much thinner on the theoretical curve than on the measured curve. As for the NGC 3310, the explanation is found with the simulated curve which reveals the dynamic nature of the real speed profile.

F.Lassiaille, may 2010

This is given on the figure below, which represents the speed curve, calculated from the simulated trajectories of the stars.
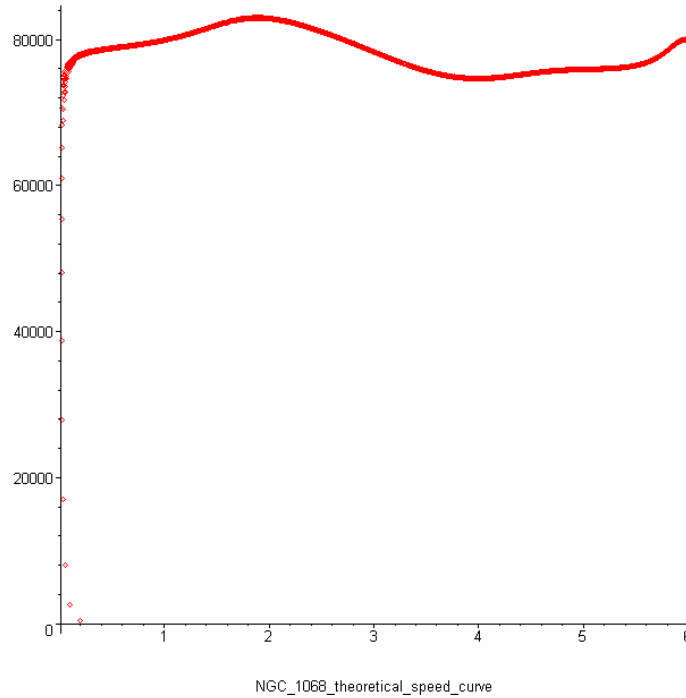

NGC_157_theoretical_speed_curve
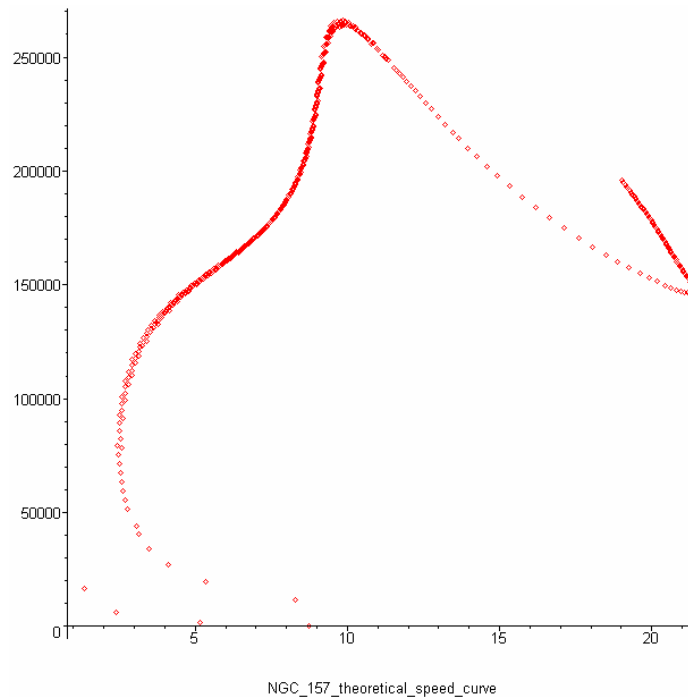
Figure 16 : Theoretical speed curve of NGC 157 with simulation

Hereafter are the parameter values which has been used in the program for this simulation of the speed of the stars (ProgMode = 24).

```
NBSTEP_NGC157 := 200 :        # Nb of steps for star simulation.
NBTRAJ_NGC157 := 500 :        # Nb of stars for star simulation.
V0_NGC157 := 180*10**3/kpc:    # Initial speed for star simul.
DY_NGC157 := 1/20 :            # Sampling space period in kpc.
AXANG_NGC157 := 0.4 :         # Angle of axis for profile in rad.
```

Once again, the length of the measured wave is retrieved. Hence, the length of the wave on figure 13 is equal to (22 – 10) = 12 kpc, and the measured value of figure 6 is equal to. (25 – 10) = 15 kpc.

Moreover, as for NGC 3310, we see that the simulated curve above retrieves, exactly, the "out of range" measured values around the maximum. Hence, for an x distance around 10 kpc, the measured points are in fact above the plain line measured curve. One can see that on figure 6. But those points are clearly noticed on the simulated theoretical curve, on the figure above.

F.Lassiaille, may 2010

4. NGC 7541

For this galaxy, i have no matter density curve, but only a surface matter density. i have also no estimated value for the mass of the galactic center.

Once again, the theoretical curve shape is very close to the measured one.

The gravitational force is always repulsive, for available <<x>> range of [0, 25 kpc]. (For other galaxies, the force is repulsive and attractive depending on the distance <<x>> from the galactic center).

It should be interesting to check if this is possible or not, with the help of what we actually know about the galaxies.

Anyway the "smoothing" of the force, which results with a positive force, followed by using the equation <<v = √(F x / m)>>, works well in this case.
Hence, a star which is located in a place where exists a strong repulsive force, will be "pushed" away from the galactic center. This trajectory modification will decrease the rotational speed of this star. On the contrary, a low repulsive force will modify slowly the rotational speed of the star.

The simulation of the speed of the stars, with this galaxy, gives the following result.

NGC_7541_theoretical_speed_curve
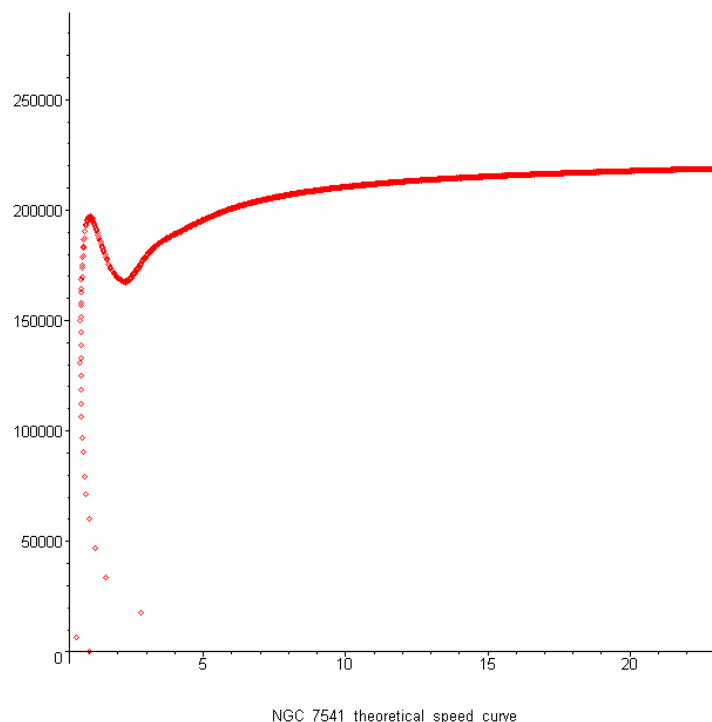
Figure 17 : Theoretical speed curve of NGC 7541 with simulation

　　　　　　　　F.Lassiaille, may 2010

Hereafter are the parameter values which has been used in the program for this simulation (ProgMode = 24).

```
NBSTEP_NGC7541 := 4200 :      # Nb of steps for star simulation.
NBTRAJ_NGC7541 := 2000 :      # Nb of stars for star simulation.
V0_NGC7541 := 287*10**3/kpc:   # Initial speed for star simul.
DY_NGC7541 := 1/78 :           # Sampling space period in kpc.
AXANG_NGC7541 := 0.94 :       # Angle of axis for profile in rad.
```

Once again, the profile has the same shape as the measured one. The speed values on the right, around 30 kpc, are too low, compared to the measured values. But this simulation is done using a constant initial value. Of course this is not the case in reality.

5. NGC 7331

The theoretical curve shape is close to the measured one, for the available distance range.

Here, i have a matter density curve, but no galactic center mass.

The x range has been reduced from the initial one of [0-33] kpc, to the [1-17.5] kpc range. That's because of the difficulties to interpolate the matter density points with the polynomial interpolation. And this comes from the difficulty to have precise high sampled values of matter density curve. The "paper" method is too imprecise.

In general, it should be much better to use the exact data coming from the galaxy study articles.

For x = 2 kpc, the theoretical values are too low, compared to original observations. This might come from the uncorrect calculation of the speed, with this "smoothed" method.

But is seems, in [4], that for x around 14 kpc (and y around 100 km/s) , that the original observations, plotted with points, fit better with the theoretical curve of figure 9. For these special x and y values, this fitting is better than with the measured curve extrapolated with Brandt formula (solid line of figure 10) Hence, for 14 kpc, a step is clearly seen on the theoretical curve. This step is noticed on the original observations, but not on the solid lined measured curve calculated with Brandt formula.

The simulation of the speed of the stars, with this galaxy, gives the following result.

F.Lassiaille, may 2010
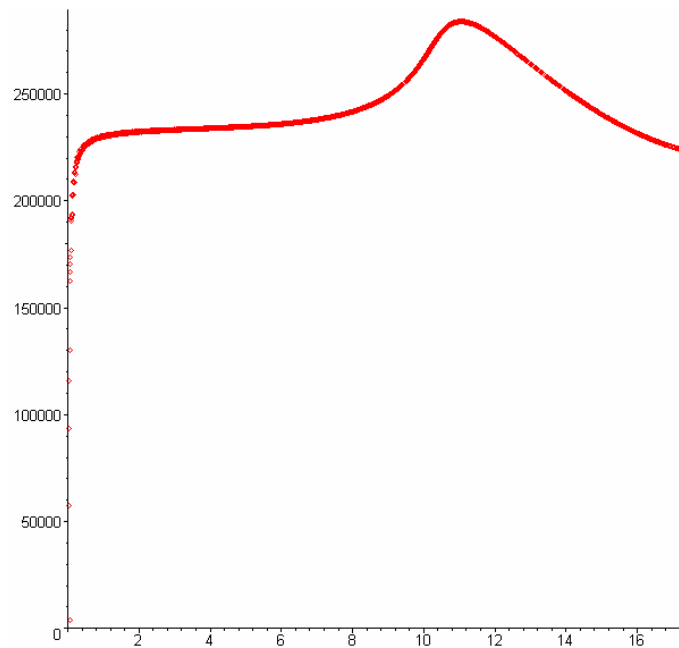
NGC_7331_theoretical_speed_curve

Figure 18 : Theoretical speed curve of NGC 7331 with simulation

Hereafter are the parameter values which has been used in the program for this simulation (ProgMode = 24).

```
NBSTEP_NGC7331 := 3600 :      # Nb of steps for star simulation.
NBTRAJ_NGC7331 := 3000 :      # Nb of stars for star simulation.
V0_NGC7331 := 280*10**3/kpc:  # Initial speed for star simul.
DY_NGC7331 := 1/200 :         # Sampling space period in kpc.
AXANG_NGC7331 := 0.72 :       # Angle of axis for profile in rad.
```

The similarity with the plain line measured curve of figure 10 is not clear. But, when we compare this simulated speed profile with the measured points of figure 10, then the decreasing of the speed values between 12 and 16 kpc is noticed, once again.
For this galaxy, it should be interesting to plot the theoretical curves on a broader range of distance values. This is not the case yet because it is difficult to get a correct interpolation of the matter density curve for this galaxy.

6. GENERAL CONSIDERATIONS ON THE CURVES

Thinner "waves"
For NGC 3310, NGC 1068, and NGC 157, we can see that the "waves" formed by the curves are thinner in the theoretical curves.

That comes simply from the fact that, if 2 stars A and B are located respectively at xa and xb distances from the galactic center, if xa<xb, if A is subject to a more important attracting

F.Lassiaille, may 2010

force than its near neighbour B, then A will be attracted more by the galactic center than B. It results immediately an increase of the distance xb-xa between the 2 stars.

More generally there is a dynamic link between the gravitational force and the rotational speed of the stars. Once again, the classical equation $F = m\,v^2/x$ is partially false here. It supposes that the behaviour of the stars is not changing with time (static, not dynamic). Once again, it should be interesting to calculate a more realistic dynamic relation in this case, than $F = m\,v^2/x$.

Greater global increase

Another remark on the curves is that, for great values of x, the theoretical speed increase less than the measured one. This is noticed for the "smoothed" theoretical curves ("ProgMode" equal to 10 or 11 in the program).

This comes probably from the fact that the calculation of v is not perfect in this case. Let's make some calculation about it. We have :

$$v = \sqrt{(F\,x\,/\,m)} \qquad\qquad\qquad \text{if } F > 0$$

$$w = \sqrt{((F - F_{mean})\,FSmoothFactor + F\_pull)\,x\,/\,m}$$
$$\qquad\qquad - \sqrt{((F\_pull - F_{mean}\,FSmoothFactor)\,x\,/\,m)} \qquad \text{in all cases}$$

where <<v>> is the real speed, at the t instant, and <<w>> is the theoretical "smoothed" speed calculated by the MAPLE program below (with Smooth parameter set to 1).

<<F>> is the theoretical calculated gravitational force. <<F_pull>> is the constant used to raise up each F values (after the first smoothing). <<m>> is the mass of the star. <<x>> is the distance of the star from the galactic center.

Then we have, after calculation :

$$w - v \approx F/(2\sqrt{Fu})\ \sqrt{x} \qquad\qquad \text{with } Fu = F\_pull - F_{mean}\,FSmoothFactor$$

(supposing that F<<Fu which is the case with the 5 galaxies).

$$\approx \tfrac{1}{2}\,\sqrt{(F/Fu)}\ v \qquad\qquad \text{after calculation}$$

For example, if we supposes

$$v = k\,\sqrt{x}, \qquad\qquad \text{then we have :}$$

$$v = \sqrt{(F\,x\,/\,m)} = k\,\sqrt{x} \ \Rightarrow\ F = k^2\,m \ \Rightarrow\ w-v \approx \tfrac{1}{2}\,\sqrt{(k^2 m/Fu)}\ k\,\sqrt{x} \text{ which gives :}$$

$$w-v \approx \tfrac{1}{2}k^2\sqrt{(m/Fu)}\ \sqrt{x}$$

hence w-v is increasing with x with a $\sqrt{x}$ law, as v.

F.Lassiaille, may 2010

That's for F values being positive.

For negative F values, then i use the paragraph "Greater variations" above.
It explained that the decreasing of v between 2 maximum of v (maximums usually obtained for F>0) is much greater than the real one.

F.Lassiaille, may 2010

5. COLLECTING MATTER DENSITY VALUES

The method i used for retrieving the matter density curves for those galaxies is the following.

I printed on a printer the matter density curves, and i measured the values by hand. That's why the initial matter density values in my program are expressed in mm.

For example, below is the matter density curve for NGC 3310 (and NGC 1068) which i printed on a paper and measured by hands for obtaining the matter density values used by my program below.
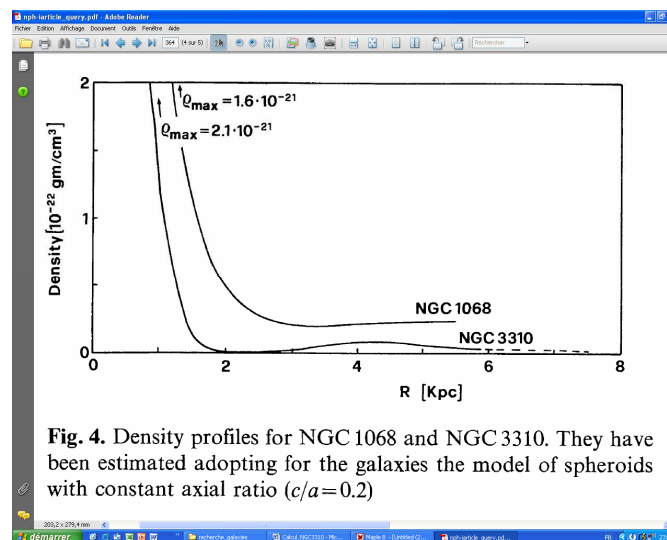
Figure 19 : Matter density curves for NGC 3310 and 1068 galaxies.

I had to transform those values in my program with a polynomial interpolation.

That's because the values i measured on my paper are not precise at all. It is also because my theoretical algorithm is very sensitive to low variations.

One may check that the resulting values after interpolation are close enough to the one of the galaxies articles. For this checking, just run the program with the option "ProgMode" set to 40.

F.Lassiaille, may 2010

6. CONCLUSION

The theoretical speed curves are close to the measured one, and this for each of the 5 galaxies : NGC 3310, NGC 1068, NGC 157, NGC 7541, and NGC 7331.

The calculated gravitational force is attractive and repulsive depending on the distance from the galactic center. For NGC 7541, it is always repulsive for the available range of distances. For calculating the speed with repulsive values of the force, the usual equation $v = \sqrt{[F \, x \, / \, m]}$ is not correct. That's why the program raises the values of the force with a constant value Fup, and smoothes the force. After that, the resulting force is positive and the classical equation above is used for v. Finally, the program subtract $\sqrt{(Fup \, x \, / \, m)}$ from the obtained value of v.

This uncorrect fashion of calculating the speed profile yields information about the repulsive values of the gravitational force, and works well for retrieving the whole speed profiles.

The only big difference between these theoretical and the measured curves appears for the lengths of some "waves" in the speed profiles.
When those differences occurs (NGC 3310 and NGC157), the explanation is clearly given by the dynamics of the stars inside the galaxies. Each time, this explanation is obvious on the speed profiles calculated using a simulation of the trajectories of the stars.

This shows that the speed profiles are a consequence of static but also dynamics consequences of the gravitational force. This is not a new information in general, but now we can calculate the exact dynamics of the stars inside the galaxies. This possibility yields actually some good results in the understanding of the NGC 3310 and NGC 157 dynamics.
In a more general way, this new information could help the studying of the galaxies.

More than the speed profiles, the values of the speed themselves can be calculated. This is possible when we have values of matter density curves and galactic center masses.
This is the case with NGC 3310 and NGC 1068. The theoretical values are 10 % and 63 % close to the measured ones, respectively for NGC 3310 and NGC 1068.
The exact values are retrieved when modifying the galactic center masses. For example, for NGC 1068 the exact value of 220 km/s is retrieved when using the value of 7.5 * 10^10 sun mass in place of 10^10 sun mass.

In the actual state this paper is a validation of [6].
Let's remind that this article explains also the second dark matter mystery, which concerns the speed of the galaxies themselves.

F.Lassiaille, may 2010

Since this article is totally compatible with the three element theory [5], this paper is also a validation of this unifying physics theory.

F.Lassiaille, may 2010

7. APPENDIX : SMOOTHED SPEED PROFILES WITH GREATER CENTER MASSES

The MAPLE program below has given the red figures above.

But there are some more figures which can be printed by this program. It depends of the value of the ProgMod parameter.

The first type of figure is the calculation of the "smoothed" speed profiles using a stronger value for the galactic center masses. This calculation is done when the ProgMode parameter is set to 11. It means: the same as ProgMode set to 10, but with greater galactic center masses.

The result of this modification is to lower the effect of the matter density distribution. It leads interesting speed profile, noticeably for NGC 3310 galaxy. Of course, this speed profile is uncorrect. But it reveals exactly the same shape for the first maximum (on the left) as the one of the NGC 3310 speed profile. This is shown on the figure below.
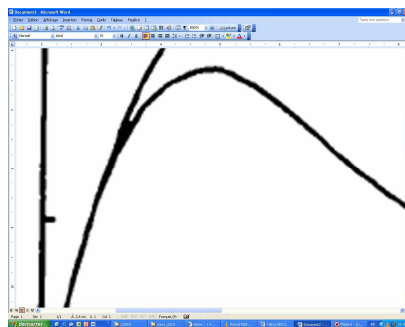
Figure 20 : Zoom on measured speed profile of NGC 3310: first maximum.

It must be compared with the first maximum of **Figure 22 : Theoretical smoothed speed profile for NGC 3310 with a greater center mass.** This zoom is shown just below.
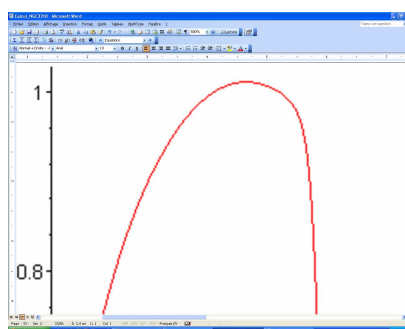
Figure 21 : Zoom on theoretical speed profile of NGC 3310: first maximum.

F.Lassiaille, may 2010

The "right border" of these 2 maximums shows exactly the same global variation of the curvature. This variation is a strong acceleration of the curvature, followed by a nearly straight line.

The figures below shows the result of these calculations, done with ProgMod set to 11.
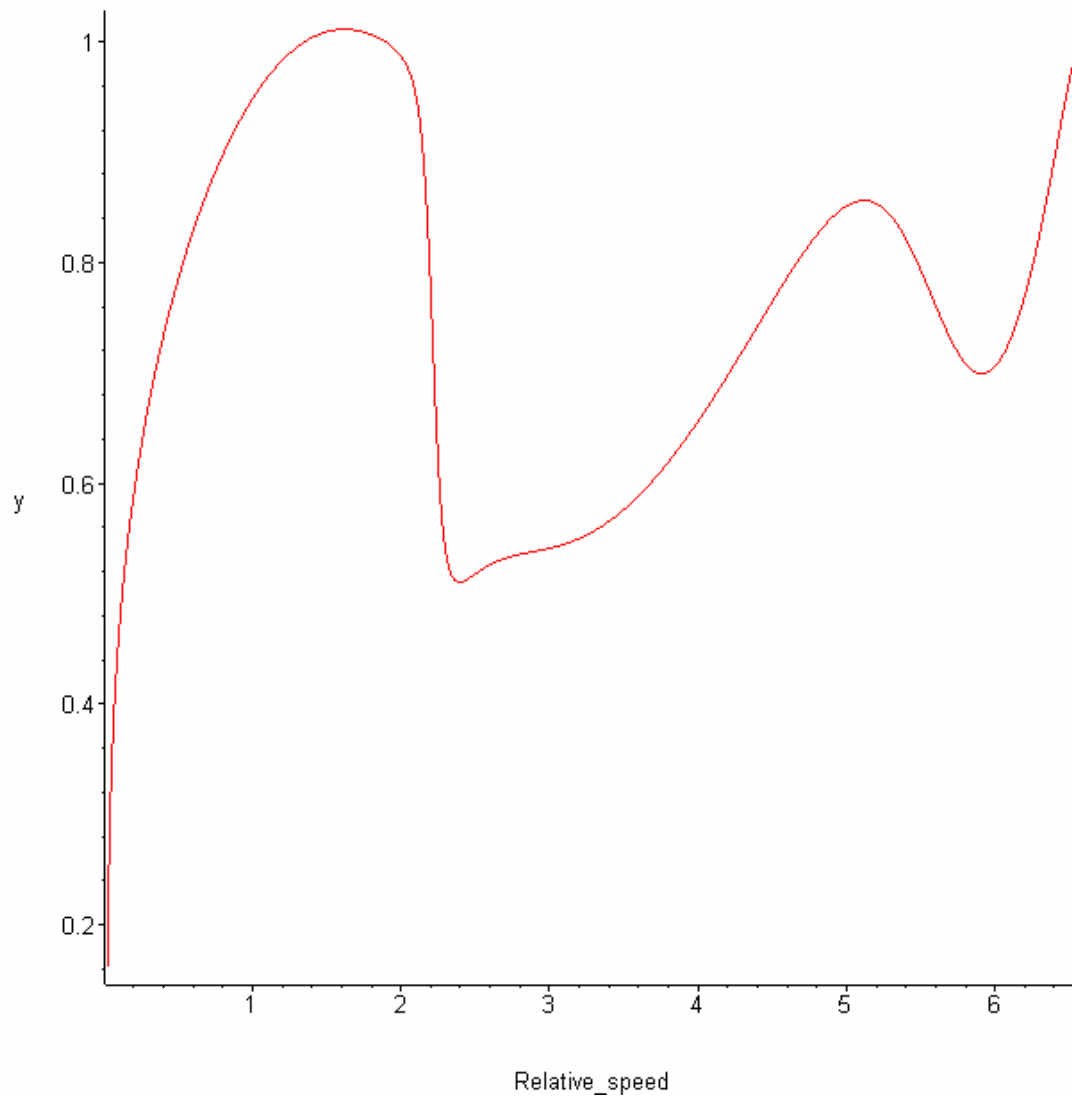
Figure 22 : Theoretical smoothed speed profile for NGC 3310 with a greater center mass.

F.Lassiaille, may 2010
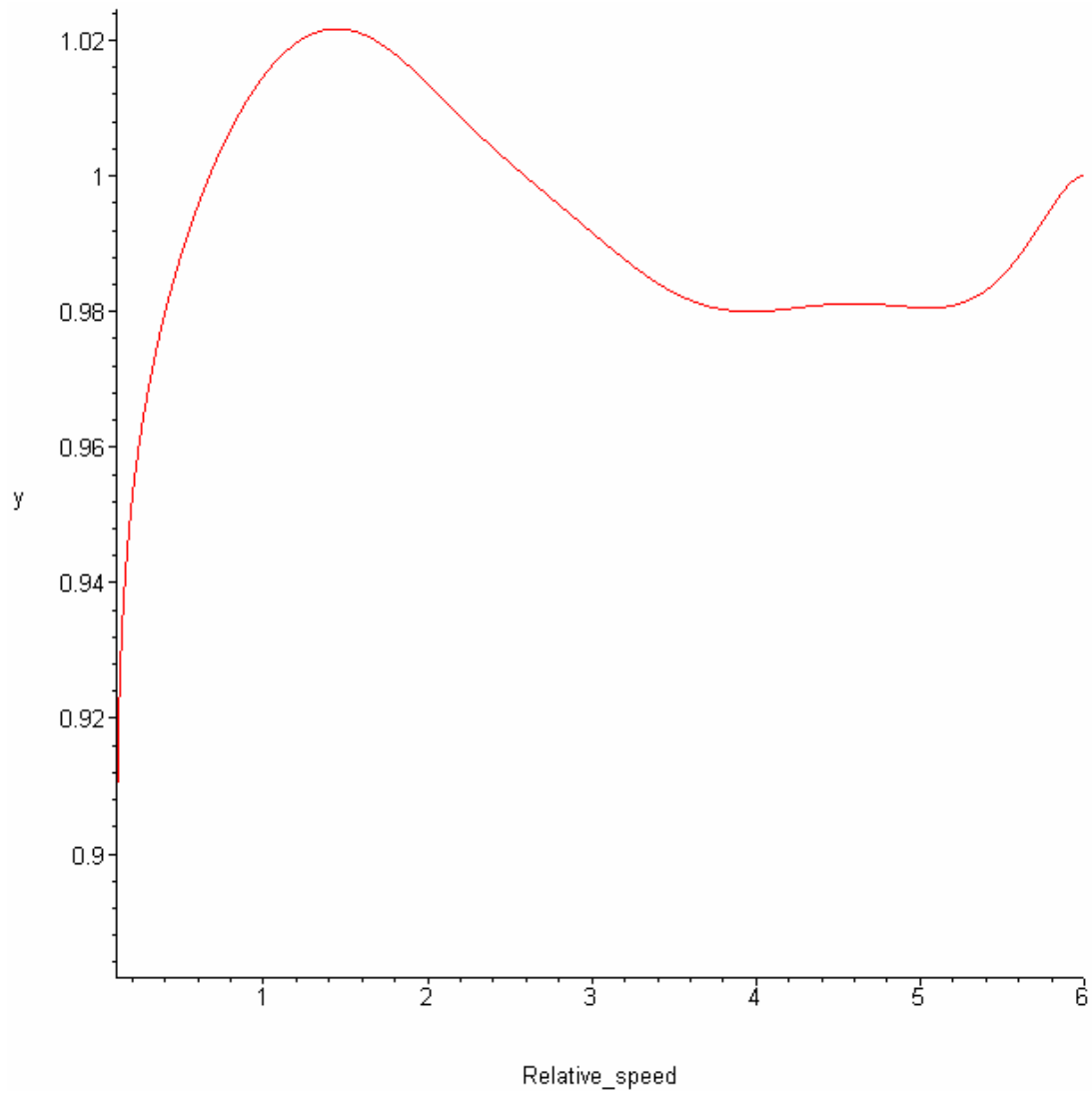
Figure 23 : Theoretical smoothed speed profile for NGC 1068 with a greater center mass.

F.Lassiaille, may 2010
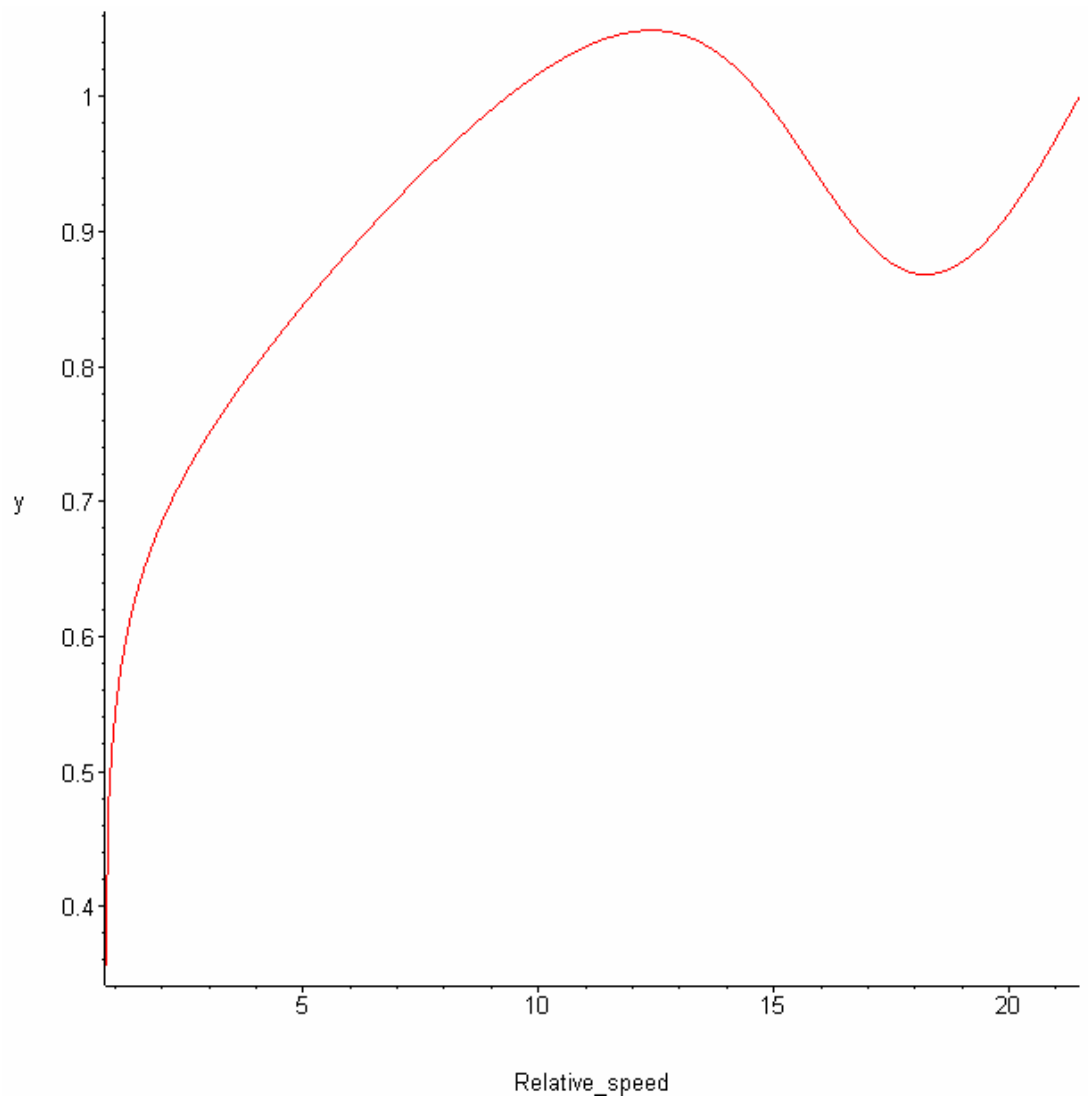
Figure 24 : Theoretical smoothed speed profile for NGC 157 with a greater center mass.

F.Lassiaille, may 2010
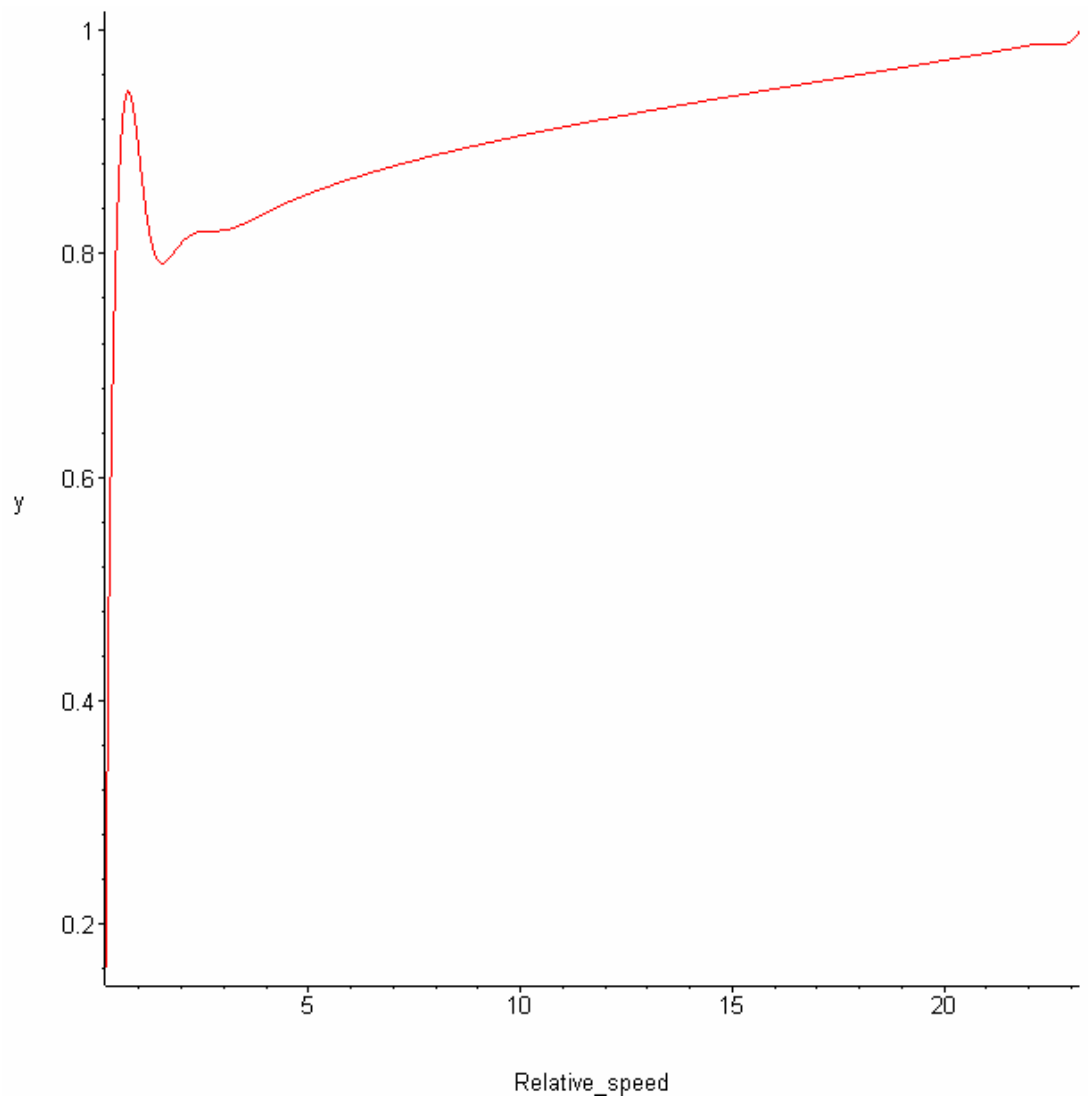
Figure 25 : Theoretical smoothed speed profile for NGC 7541 with a greater center mass.

F.Lassiaille, may 2010
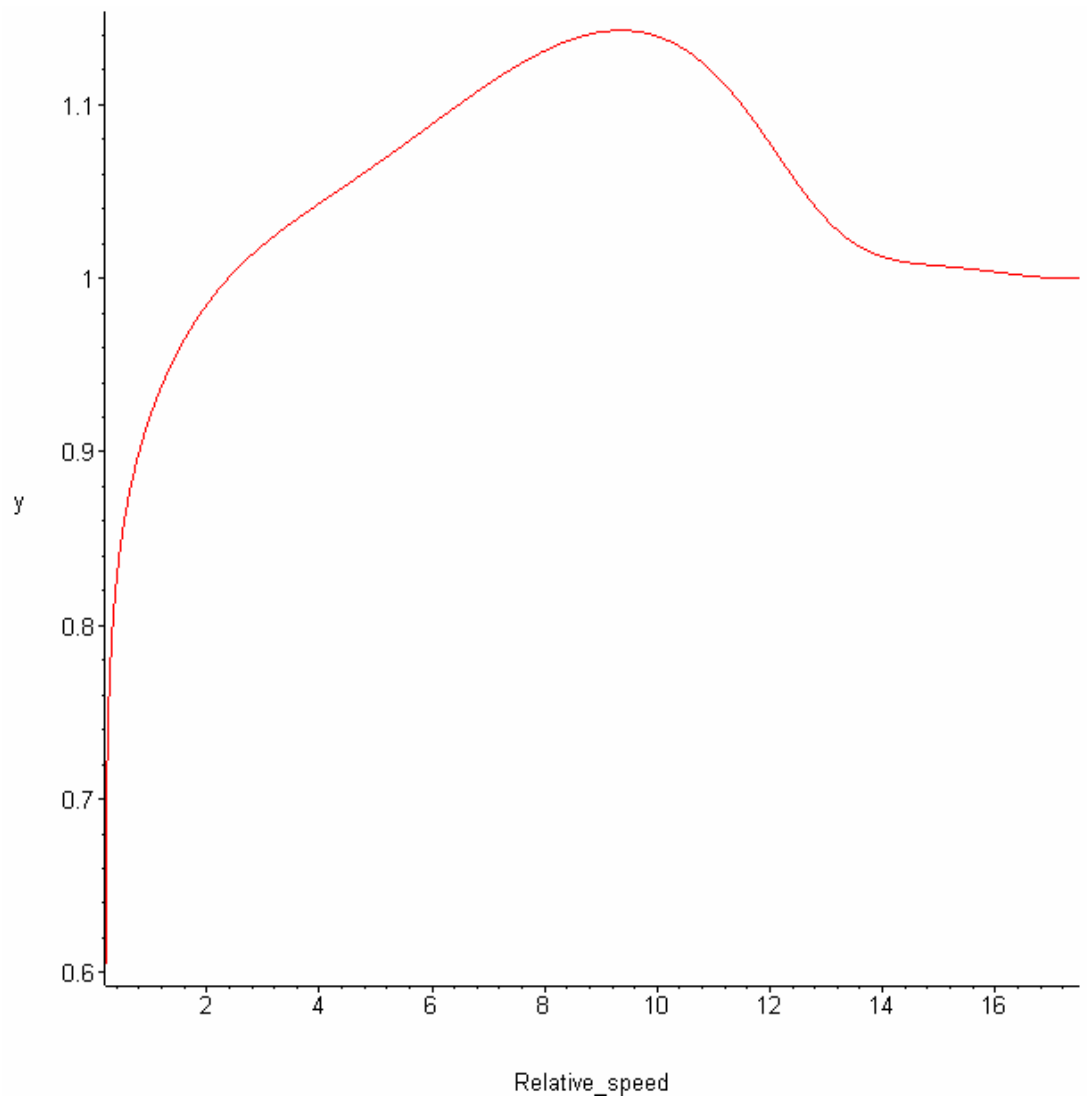
Figure 26 : Theoretical smoothed speed profile for NGC 7331 with a greater center mass.

F.Lassiaille, may 2010

8. APPENDIX : SIMULATED TRAJECTORIES

The following figure is showing the simulated trajectories of the stars in the case of the NGC 3310 galaxy. They are calculated when running the program with the ProgMode parameter set to 30.
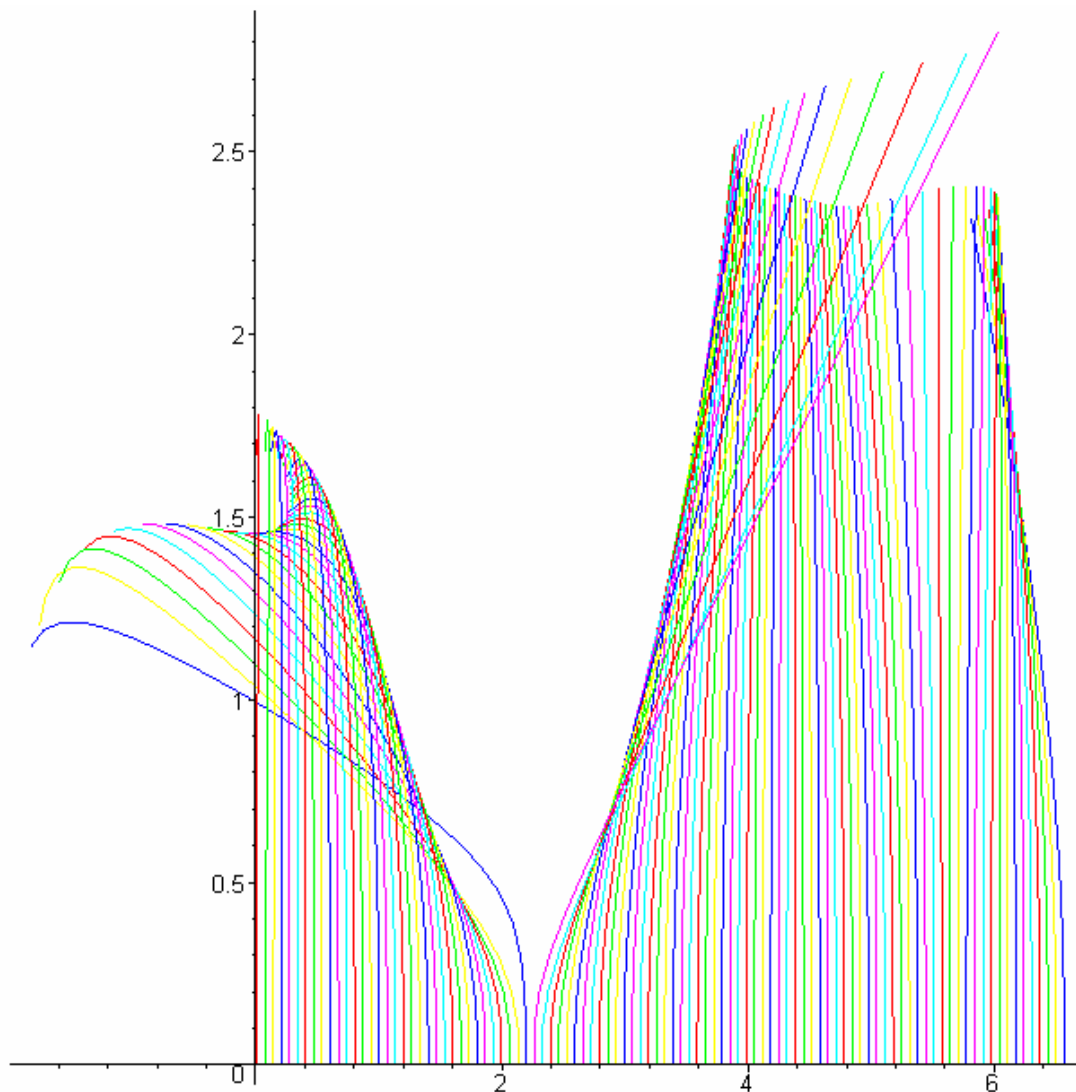
Figure 27 : Simulated trajectories of the stars for NGC 3310.

The parameter values for the simulation above are the following.

```
NBSTEP_NGC3310 := 25 :        # Nb of steps for star simulation.
NBTRAJ_NGC3310 := 100 :       # Nb of stars for star simulation.
V0_NGC3310 := 150*10**3/kpc:     # Init speed for star simul.
DY_NGC3310 := 1/10 :             # Sampling space period in kpc.
AXANG_NGC3310 := 0.33 :       # Angle of axis for profile in rad.
```

F.Lassiaille, may 2010
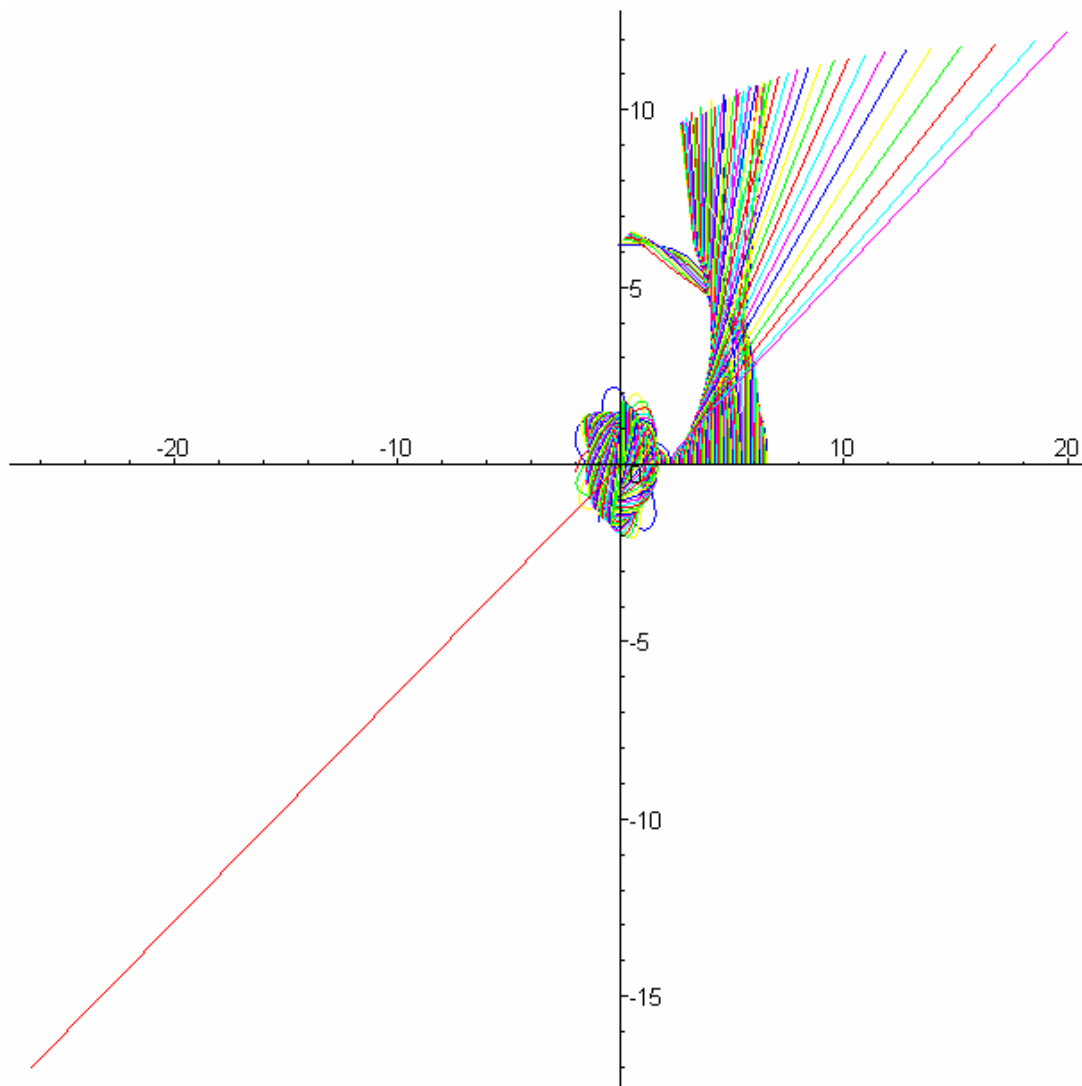
Figure 28 : Simulated trajectories of the stars for NGC 3310 (more steps).

The parameter values for the simulation above are the following.

```
NBSTEP_NGC3310 := 100 :        # Nb of steps for star simulation.
NBTRAJ_NGC3310 := 100 :        # Nb of stars for star simulation.
V0_NGC3310 := 150*10**3/kpc:       # Init speed for star simul.
DY_NGC3310 := 1/10 :               # Sampling space period in kpc.
AXANG_NGC3310 := 0.33 :        # Angle of axis for profile in rad.
```

F.Lassiaille, may 2010
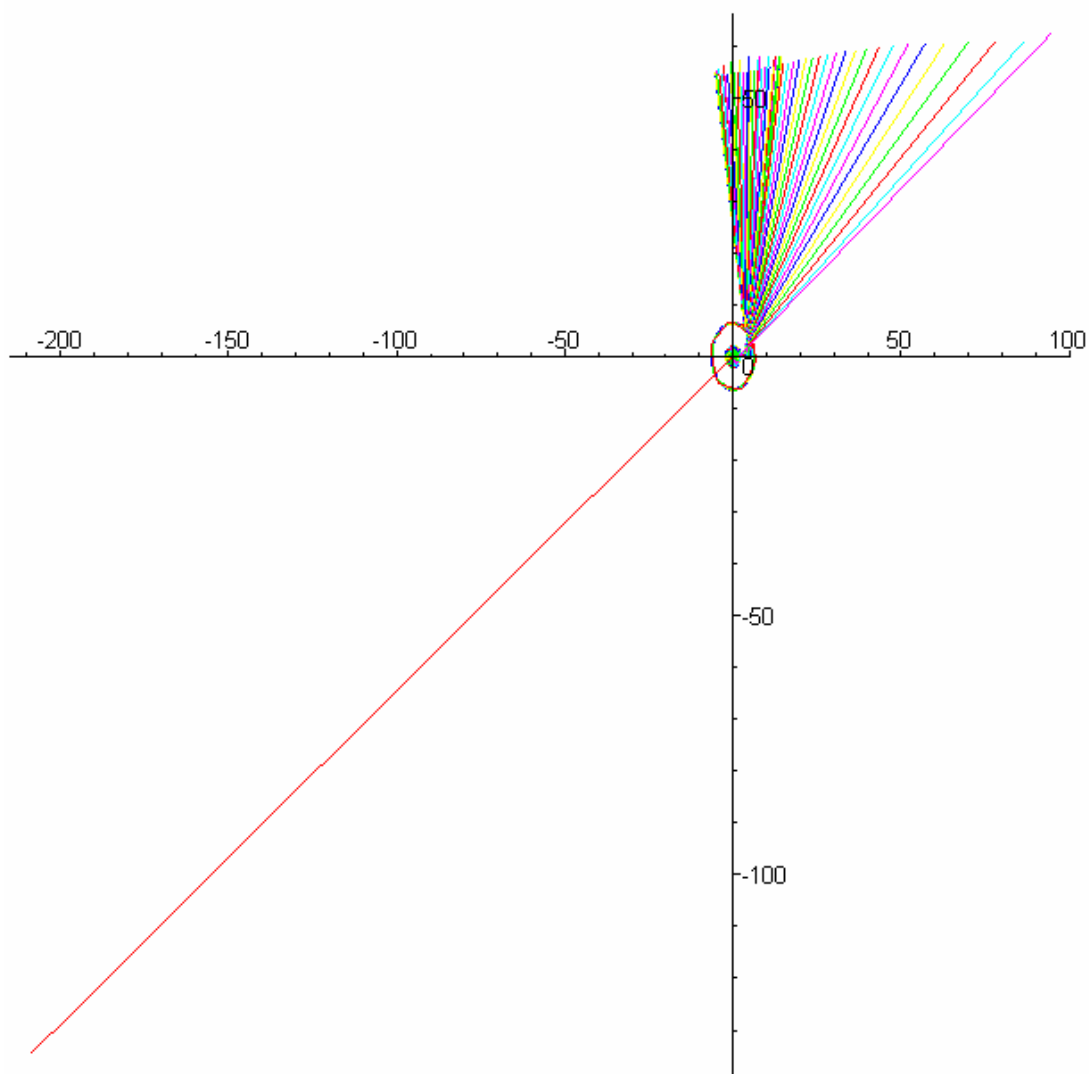
Figure 29 : Simulated trajectories of the stars for NGC 3310 (again more steps).

The parameter values for the simulation above are the following.

```
NBSTEP_NGC3310 := 500 :        # Nb of steps for star simulation.
NBTRAJ_NGC3310 := 100 :        # Nb of stars for star simulation.
V0_NGC3310 := 150*10**3/kpc:   # Initial speed for star simul.
DY_NGC3310 := 1/10 :           # Sampling space period in kpc.
AXANG_NGC3310 := 0.33 :        # Angle of axis for profile in rad.
```

F.Lassiaille, may 2010

9. APPENDIX : MAPLE PROGRAM

```
####################################################################
#                                                                  #
#        CALCULATION OF THE VELOCITY CURVES OF NGC3310             #
#          NGC1068 NGC157 NGC7541 AND NGC7331 GALAXIES            #
#                                                                  #
# This MAPLE program is calculating and plotting the curve of     #
# rotation velocity of the galaxy stars, as a function of         #
# their distance x from the galactic center.                      #
#                                                                  #
# This calculation is done for the NGC3310 NGC1068                #
# NGC157 NGC7541 and NGC7331 galaxies.                            #
#                                                                  #
# It is done using uniquely the <<solution for the                #
# "dark matter mystery" based on Euclidean relativity>>           #
# article. Refer to this article in vixra april 2009              #
# F.Lassiaille : viXra.org, Relativity and Cosmology,             #
# viXra:0912.0048.                                                #
#                                                                  #
# The matter density values are memorized in the "rho" arrays.    #
# They has been printed and measured on the printed papers.       #
# Those values are convert in Kg/AL3, then smoothed with a        #
# polynomial interpolation.                                       #
#                                                                  #
# After this, the calculation is done.                            #
# Equations for calculating the resulting curve is explained      #
# and determined in the article mentioned above.                  #
#                                                                  #
# Determination of the constant <<r>> used in these equations     #
# is made using the informations coming from another "dark        #
# matter" mystery, the velocities of galaxies inside their        #
# groups (see article). It uses also the value of 10 for          #
# G'/G, where G' is the extra galactic gravitational              #
# constant, outside the studied galaxy. G is our                  #
# gravitational constant, inside our milky-way galaxy.            #
#                                                                  #
# The curve for the calculated gravitational force shows          #
# great variations along the x axis. There are, roughly,          #
# as much positive (attractive) values as negative values.        #
# This is why the following usual formula, v² = F x / m, is not #
# correct in this case, because this formula supposes a           #
# positive (attractive) value of F.                               #
#                                                                  #
# For trying to solve simply this problem, i smoothed the         #
# force curve with a FSmoothFactor, and raised it up enough       #
# (+F_pull) in order to get only positive values for F.           #
# The FSmoothFactor value has been fitted in order to get the     #
# best resulting compared speed values for the 2 maximum of the #
# NGC3310 galaxy speed curve.                                     #
#                                                                  #
# After that the speed is calculated with v = sqrt(F x/m).        #
# Since the gravitational force was raised up with a value        #
# F_pull, i substract a value sqrt(F_pull x) after calculating    #
```

```
# the speed.                                                      #
# This is the calculation done with ProgMode set to 10 and 11.   #
#                                                                  #
# The resulting curve has the same shape and the same values as  #
# the measured one.                                                #
#                                                                  #
# A more correct calculation of the speed, using only            #
# the initial curve of the force, is done with ProgMode = 24.    #
# In this case the calculation is a simulation of the            #
# trajectories of the stars. Each simulated star is starting     #
# on the same "Ox" axis and at the same speed.                   #
# For each simulated star, the program stops the star trajectory#
# simulation as soon as this one reach the "Or" axis,            #
# and calculates the rotational speed of this star at this       #
# instant. The "Or" axis is a straight line starting from the    #
# origin (0,0), and makes an "a" angle with "Ox", the initial    #
# axis (starting from the origin also) from where the simulated #
# stars has started.                                              #
# This "a" angle, the initial speed, and some more parameters    #
# are set depending of the galaxy the program is working on.     #
#                                                                  #
# At the end i execute exactly the same program but with the     #
# NGC1068 galaxy. Again i obtain nearly the same values and the  #
# same shape. Also i executed exactly the same program with      #
# the NGC 157 NGC 7541 and NGC 7331 galaxies. But here i dont    #
# get any estimation of galactic center masses. However, the     #
# global shape is retrieved also in this case.                   #
#                                                                  #
####################################################################
####################################################################




####################################################################
####################### Initialisations ####################### #
####################################################################

# -------------------------- MAPLE ------------------------ #
with(CurveFitting):
Digits := 50 :                        # Floating points precision.
MAXVAL := 10**100000000:
unassign('DensPoly'): unassign('n'):
unassign('l'):        unassign('ll'):        unassign('rho'):
unassign('x'):        unassign('xprime'):    unassign('e'):
unassign('tg'):       unassign('cos2'):      unassign('F'):
unassign('v'):

# ----------------- Cosmological values ------------------- #
kpc := 3.08 * 10**19 :        # 1 kilo parsec expressed in meter.
c := 3 * 10**8 / kpc:                      # light speed in m/s.
AL := 365 * 24 * 3600 * c:         # 1 year light in meter.
pc := kpc/1000 :              # 1 parsec expressed in meter.
M0 := 2 * 10**30 :                        # Sun mass in Kg.
G := 6.6742867 * 10**(-11) / kpc**3:    # Gravit constant m3 kg(-1) s(-2)
Gp := 10 * G:         # Gravitational constant outside any galaxy.
rho0 := 0.003 * M0 / (AL**3):  # Matter density near solar system
```

```
# ----------- Corrected Newton's law calculation ------------- #
Si_So := sqrt(Gp/G) - 1: # Intra gal/extra galactic contributions
r := rho0 * Si_So**(-2):   # Square of extragalactic contribution
                    # proportional to the square of extra-galactic
                    # lights contributions. Linked to rho0 value.
                    # Unit : Kg m(-3).
k_factor := sqrt(r) + sqrt(rho0):  # In order to have the normal
              # classical Newton's law in the case of our solar
              # system. This value supposes that the evolution
              # of mass density around the sun is weak
              # and negligeable in front o rho0.


# ---------- Discretising the curve of the force ------------ #
MUL := 10:          # This multiplies the NPoly number to get the
MUL := 50:          # This multiplies the NPoly number to get the
              # number of points for discretising the force curve.



####################################################################
#                         PROGRAM MODE                            #
####################################################################

# ----- WHAT DO YOU WANT TO CALCULATE WITH THIS PROGRAM ? ----- #
ProgMode := 11:    # 0:   Speed profile brute force (neg val = 0).
                   # 10:  Relative speed profile: force F smoothed
                   #      raised up, speed calculated from this F,
                   #      and sqrt(Fup * x) substracted.
                   # 11:  Same as 10 but with a much greater value
                   #      for the mass of the galactic center,
                   #      for weakening the effect of the
                   #      variations of matter density.
                   # 20:  Speed profile: dynamic calculation
                   #      from the star trajectories below.
                   # 24:  Same as 20 but along a specific axis.
                   # 30:  Star trajectories in 2D.
                   # 40:  Plotting the matter density.
                   #      (after interpolation).
                   # 50:  Plotting e: "relativistic ratio".
                   # 60:  Plotting cos2: square of the
                   #      relativistic operator.
                   # 70:  Plotting tg: tan = slope of space line.
                   # 80:  Plotting the gravitationnal force.
                   # 90:  Plotting integral mass profile.
                   # 100: Plotting curve of space inside
                   #      space-time.

# ------------------------ Parameters ----------------------- #
# ------------------ if ProgMode := 10 or 11 ---------------- #
FSmoothFactor := 1.6 * 10**(-4):    # Reduce amplitude of force.
# -------------------- if ProgMode := 11 -------------------- #
MFactor := 0.3 * 10**20:                    # Increasing factor.



####################################################################
################### Calculation procedure ####################
```

```
###############################################################
# M:          Mass of the galactic center, in Kg.
# rho:        Matter density values, in Kg/m3.
# IMin:       First indice of valid values in the rho array.
# IMax:       Last indice of valid values in the rho array.
# xmin:       Distance in m corresponding to imin.
# xmax:       Distance in m corresponding to imax.
# xminpl:     Plotting min distance from the galactic center in m.
# xmaxpl:     Plotting max distance from the galactic center in m.
# NPoly:      Nbpts polynomial smoothing of matter density curve.
# HSpeed:     A constant used for raising or lowing the speed.
#             The shape of the curve in not affected by this value.
#             Only the whole curve is raised up or lowered.
# ProgMode:   Very important : what has to be done.
#             See above all the possible values.
# NbStep:     Only if ProgMode = 20<->30 (simulation of star
#             trajectories). Nb of step of simulation.
# NbTraject: Only if ProgMode = 20<->30. Nb of simulated stars.
# v0:         Only if ProgMode = 20<->30. Initial speed. Kpc/s.
# dy:         Only if ProgMode = 20<->30. Sampling space lgth. Kpc
# AxisAngle: Only if ProgMode = 24. Angle of the axis for the
#             calculation of the speed profile. This axis in the
#             line starting from the galactic center. The angle
#             is between this axis and the axis from which stars
#             are initially starting with v0 speed. In rad.
# Title:      for writing the title of the figure when plotting.

GalaxySpeedCalcul := proc( M, rho, IMin, IMax, xmin,
                           xmax, xminpl, xmaxpl, NPoly,
                           HSpeed, ProgMode, NbStep, NbTraject,
                           v0, dy, AxisAngle, Title )

# This procedure ensures that the calculation is exactly the   #
# same for each galaxy.                                        #

# ---------------- Scaling factor ------------------------- #
xminm := xmin*kpc:
xmaxm := xmax*kpc:
xminplm := xminpl*kpc:
xmaxplm := xmaxpl*kpc:
Delta := (xmaxm-xminm)/(IMax-IMin):  # Distance in m between 2
                            # consecutive measurements on my paper.
DeltaKpc := Delta/kpc:

# ---------------- Cosmological value -------------------- #
if(ProgMode = 11 ) then
  M2 := M * MFactor:        # Weakening effect of matter density.
else:
  M2 := M:                               # No change for M.
end if:
R := M2 * G / c**2:    # Schwarschild ray for considered galaxy.


################ Polynomial interpolation ################
# That's mandatory because calculation of gravitational force
# is very sensitive to low variations. This sensitivity can be
# demonstrated with calculations, and is seen with this program.
```

```
Step := floor((IMax-IMin)/(NPoly-1)):
                        # Nb values skipped between 2 poly pts.
lp := [[xmin + DeltaKpc*nn*Step, rho[IMin + nn*Step]]
                                      $nn=0..(NPoly-1)]:
unassign('x'):
DensPoly := PolynomialInterpolation(lp, x) :

########## Corrected Newton's law calculation ################

# ---------- Calculation of the gravitational force --------- #
e := k_factor * sqrt(8*R/x)/(sqrt(r) + sqrt(DensPoly)):
cos2 := (1 + e) / ((1 + e/2)**2):
tg := sqrt(1/cos2 - 1):     # All this explained in my article.
F := c**2 * diff(tg,x) * tg * (abs(1 - tg**2))**(-3/2) :

unassign('x'):
if( ProgMode = 0 )             # If NO smoothing of the force
then                # plot speed curve with positive values of F.

  ################# Plotting the v curve ###################
  unassign('x'):                          # x is in kpc.
  v := sqrt( abs(max(F,0) * x )) * kpc :   # neg value F => v=0
  plot( v, x=xminpl..xmaxpl );         # v in m/s and x in kpc.

elif( ProgMode = 10 or ProgMode = 11 ) then     # If smoothing
          # gravit force for getting repulsing effect on speed
                                          # of the stars.

  # ---------- Discretising the curve of the force ---------- #
  # in order to be able to smooth it.
  IMinForce := 1:         # Each value between xmaxpl and xminpl
  IMaxForce := NPoly * MUL: # discretised with Npoly*MUL points.
  StepForce := (xmaxpl-xminpl) / (IMaxForce - IMinForce):

  for n from IMinForce to IMaxForce
  do
    x := xminpl + (n - IMinForce) * StepForce:
    Xar[n] := x:                # For performance memorize this.
    F_arr[n] := F(x):                        # Discretisation.
  end do:

  # ---------------- Smoothing the force ------------------- #

  # --- Calculating the min, max and mean of F curve -------- #
  F_mean := 0:
  F_min := MAXVAL:
  F_max := -MAXVAL:
  for n from IMinForce to IMaxForce
  do
    F_mean := F_arr[n]:
    F_min := min(F_arr[n], F_min):
    F_max := max(F_max, F_arr[n]):
  end do:
  F_mean := F_mean/(IMaxForce-IMinForce+1):
  unassign('x'):
```

```
# --- Calculating value for raising up the whole curve ---- #
F_pull := max( F_max-F_mean, F_mean-F_min ) * FSmoothFactor:

# --- Actual smotthing of the curve --------------------- #
unassign('x'):
for n from IMinForce to IMaxForce
do
   FSmooth[n] := F_pull + (F_arr[n]-F_mean) * FSmoothFactor:
end do:

# --- Cutting the edges if negatives values  ------------- #
imin2 := IMinForce:
for n from IMinForce to IMaxForce
do
   if(FSmooth[n] >= 0)
   then
      imin2 := n:
      break:
   end if;
end do:

imax2 := IMaxForce:
for n from IMaxForce to IMinForce by -1
do
   if(FSmooth[n] >= 0)
   then
      imax2 := n:
      break:
   end if;
end do:

# ---------- Calculation of the speed -------------------- #
# using simple smoothing of the gravitational force and    #
# usual equation v² = F x / m.                             #

unassign('x'):
for n from imin2 to imax2
do
   v_arr[n] := sqrt( FSmooth[n] * Xar[n] ):
          # This speed has globally the sqrt(F_pull x) shape.

   # ----- Substracting "raising up force" term ----------- #
   Fup := F_pull - F_mean*FSmoothFactor:
   Fupa := abs(Fup):
   Fups := sign(Fup):
   v_arr[n] := v_arr[n] - Fups * sqrt(Fupa * Xar[n]):
end do:

# --- Calculating boundaries for calculating the min ----- #
imin3 := imin2:
for n from imin2+1 to imax2
do
   if(v_arr[n] < v_arr[n-1])
   then
      imin3 := n-1:
      break:
   end if;
```

```
end do:

imax3 := imax2:
Fold := -MAXVAL:
for n from imax2-1 to imin2 by -1
do
  if(v_arr[n] < v_arr[n+1])
  then
    imax3 := n+1:
    break:
  end if;
end do:

if(imin3 = imax3)                          # No minimum.
then
  imin3 := imin2:                  # Stay with old boundaries.
  imax3 := imin3:
else      # There is a minimum. Boundaries imin 4 and imax 4
end if:         # = maximum on the left and max on the rigth.

# --- Calculating the min, max of v curve --------------- #
vmin := MAXVAL:
vmax := -MAXVAL:
for n from imin3 to imax3
do
  vmin := min( v_arr[n], vmin ):
  vmax := max( v_arr[n], vmax ):
end do:

# --- Calculating value for raising up the whole curve --- #
if(vmin < 0)
then
  # ---- Using the fitting parameter for raising up whole curve
  vup := - vmin + (vmax-vmin) * HSpeed:
else                              # Here no use for raising up.
  vup := 0:
end if:

# --- Raising up curve and cutting edges if neg values  -- #
imin4 := imin2:
for n from imin2 to imax2
do
  if(v_arr[n]+vup >= 0)
  then
    imin4 := n:
    break:
  end if;
end do:

imax4 := imax2:
for n from imax2 to imin2 by -1
do
  if(v_arr[n]+vup >= 0)
  then
    imax4 := n:
    break:
  end if;
```

```
      end do:


      ################ Plotting the curve ###################

      # ---------------------- Speed curve -------------------- #
      ll := [[Xar[nn], (v_arr[nn] + vup)/(v_arr[imax4] + vup)]
                                          $nn=imin4..imax4]:
      plot(ll, Title=xminpl..xmaxpl,
            labels=[Relative_speed, y]);
      #              labels=[Title, y]);

  elif( ProgMode = 20 ) then    # If simulating dynamic in order
                                # to calculate speed of the stars.

      Step := (xmaxpl - xminpl)/(NbTraject-1):
      for k from 1 to NbTraject
      do
        x0 := xminpl + (k-1)*Step:
        x0 := 0.001 + (k-1)*Step:
        dt := dy/v0:
        xx[1] := x0:
        yy[1] := 0:
        xx[2] := xx[1]:
        yy[2] := yy[1] + dy:
        for i from 3 to NbStep
        do
          x := evalf( sqrt(xx[i-1]**2 + yy[i-1]**2) ):
          xx[i] := evalf( - xx[i-2]
                          + xx[i-1] * ( 2 - (dt**2*F(x))/(2*x) )):
          yy[i] := evalf( - yy[i-2]
                          + yy[i-1] * ( 2 - (dt**2*F(x))/(2*x) )):
        end do:

        vx := (xx[NbStep] - xx[NbStep-1])/dt:
        vy := (yy[NbStep] - yy[NbStep-1])/dt:
        rr[k] := evalf( sqrt(xx[NbStep]**2 + yy[NbStep]**2) ):
        if(yy[NbStep] = 0) then
          vv[k] := 0:
        else
          vv[k] := ((xx[NbStep]*vx + yy[NbStep]*vy) / rr[k])
                                        * (xx[NbStep]/yy[NbStep])
                                            - rr[k]*vx/yy[NbStep]:
        end if:
      end do:

      lv := [[rr[kk], vv[kk]*kpc] $kk=1..NbTraject]:
      plot(lv, Title=xminpl..xmaxpl, style=line );

  elif( ProgMode = 24 ) then    # Same as 20 but with calculation
                                # of speed along an axis.

      Step := (xmaxpl - xminpl)/(NbTraject-1):

      l := 1:
      for k from 1 to NbTraject
      do
```

```
    x0 := xminpl + (k-1)*Step:
    x0 := 0.001 + (k-1)*Step:
    dt := dy/v0:
    xx[1] := x0:
    yy[1] := 0:
    xx[2] := xx[1]:
    yy[2] := yy[1] + dy:

    for i from 3 to NbStep
    do
      x := evalf( sqrt(xx[i-1]**2 + yy[i-1]**2) ):
      xx[i] := evalf( - xx[i-2]
                      + xx[i-1] * ( 2 - (dt**2*F(x))/(2*x) )):
      yy[i] := evalf( - yy[i-2]
                      + yy[i-1] * ( 2 - (dt**2*F(x))/(2*x) )):

      if( yy[i]/xx[i] > sin(AxisAngle) ) then
        i := i+1:
        break:
      end if:

    end do:

    if( i = NbStep+1) then
      next:                            # Don't reach the axis : out.
    else:                              # Reached the axis : memorize it.
      vx := (xx[i-1] - xx[i-2])/dt:
      vy := (yy[i-1] - yy[i-2])/dt:
      x := xx[i-1]:
      y := yy[i-1]:
      rr[l] := evalf( sqrt(x**2 + y**2) ):
      if(y = 0) then
        vv[l] := 0:
      else
        vv[l] := ((x*vx+y*vy)/rr[l])*x/y - rr[l]*vx/y:
      end if:
      l := l + 1:
    end if:
  end do:

  lv := [[rr[ll], vv[ll]*kpc] $ll=1..l-1]:
  plot(lv, Title=xminpl..xmaxpl, style=line );
  plot(lv, Title=xminpl..xmaxpl, style=point );

elif( ProgMode = 30 ) then  # Plotting star trajectories in 2D

  Step := (xmaxpl - xminpl)/(NbTraject-1):
  for k from 1 to NbTraject
  do
    x0 := xminpl + (k-1)*Step:
    dt := dy/v0:
    xx[k,1] := x0:
    yy[k,1] := 0:
    xx[k,2] := xx[k,1]:
    yy[k,2] := yy[k,1] + dy:

    for i from 3 to NbStep
```

```
      do
        x := evalf( sqrt(xx[k,i-1]**2 + yy[k,i-1]**2) ):
        xx[k,i] := evalf( - xx[k,i-2]
                          + xx[k,i-1] * ( 2 - (dt**2*F(x))/(2*x) )):
        yy[k,i] := evalf( - yy[k,i-2]
                          + yy[k,i-1] * ( 2 - (dt**2*F(x))/(2*x) )):
      end do:
    end do:

    lxy := [ [[xx[kk,nn], yy[kk,nn]] $nn=1.. NbStep] $kk=1..NbTraject ]:
    plot(lxy, style=line);

  elif( ProgMode = 40 ) then                       # For matter density.
    #plot( DensPoly, x=xminpl..xmaxpl );
    plot( DensPoly, x=1.5..xmaxpl );
  elif( ProgMode = 50 ) then                           # For e.
    plot( e, x=xminpl..xmaxpl );
  elif( ProgMode = 60 ) then                           # For cos2.
    plot( cos2, x=xminpl..xmaxpl );
  elif( ProgMode = 70 ) then                           # For tan.
    plot( tg, x=xminpl..xmaxpl );
  elif( ProgMode = 80 ) then                           # For F.
    plot( F, x=xminpl..xmaxpl );
  elif( ProgMode = 90 ) then          # For integral mass profile.
    IntDensPoly := int(DensPoly, x);
    plot(IntDensPoly, x=xminpl..xmaxpl );         # Does not work.
  elif( ProgMode = 100 ) then          # Space inside space-time.
    # TO BE WRITTEN #
  else
  end if:



end proc:              # end of "GalaxyProgModeul" procedure.



##################################################################
################### NGC3310 and NGC1068 ########################
##################################################################

# ----------------- Matter density values ------------------- #
# Those values are read on a printed paper of the density    #
# matter of the galaxy. Hence their unit is the mm.           #

# ----------------------- NGC3310 ----------------------- #

# Below is the value extrapolated from my paper            #
# supposing that the density matter has a constant slope    #
# between 0 and 1 kpc.                                       #
rho_NGC3310[1] := 700:    rho_NGC3310[2] := 500:
rho_NGC3310[3] := 400:    rho_NGC3310[4] := 300:
rho_NGC3310[5] := 200:

# Below are the values from my paper, plain line curve.     #
rho_NGC3310[6] := 147:
rho_NGC3310[7] := 94:     rho_NGC3310[8] := 63:
rho_NGC3310[9] := 36:     rho_NGC3310[10] := 18:
```

```
rho_NGC3310[11] := 8:      rho_NGC3310[12] := 3.3:
rho_NGC3310[13] := 2.3:    rho_NGC3310[14] := 0.8:
rho_NGC3310[15] := 0.18:   rho_NGC3310[16] := 0.08:
rho_NGC3310[17] := 0.07:   rho_NGC3310[18] := 0.08:
rho_NGC3310[19] := 0.2:    rho_NGC3310[20] := 1.2:
rho_NGC3310[21] := 1.4:    rho_NGC3310[22] := 2.5:
rho_NGC3310[23] := 3:      rho_NGC3310[24] := 4:
rho_NGC3310[25] := 4.5:    rho_NGC3310[26] := 5:
rho_NGC3310[27] := 5.7:    rho_NGC3310[28] := 6.5:
rho_NGC3310[29] := 6.55:   rho_NGC3310[30] := 6.6:
rho_NGC3310[31] := 6.9:    rho_NGC3310[32] := 5.2:
rho_NGC3310[33] := 5.1:    rho_NGC3310[34] := 5:
rho_NGC3310[35] := 4.5:    rho_NGC3310[36] := 4:
rho_NGC3310[37] := 3.85:   rho_NGC3310[38] := 3.7:
rho_NGC3310[39] := 2.9:    rho_NGC3310[40] := 2.1:
rho_NGC3310[41] := 2.05:   rho_NGC3310[42] := 2:

# Below are the values from my paper, plain dashed line.    #
rho_NGC3310[43] := 1.9:    rho_NGC3310[44] := 1.8:
rho_NGC3310[45] := 1.78:   rho_NGC3310[46] := 1.77:
rho_NGC3310[47] := 1.77:   rho_NGC3310[48] := 1.76:
rho_NGC3310[49] := 1.76:   rho_NGC3310[50] := 1.75:
rho_NGC3310[51] := 1.75:   rho_NGC3310[52] := 1.74:

IMIN_NGC3310 := 5:            # First valid indice for rho array.
IMAX_NGC3310 := 52:           # Last valid indice for rho array.
LMIN_NGC3310 := 0.69:        # Distance corresponding to IMin.
LMAX_NGC3310 := 7.25:         # Distance corresponding to Imax.
LMINPL_NGC3310 := 0.01:     # Min value for x, for plot. In Kpc.
LMAXPL_NGC3310 := 6.57:     # Max value for x, for plot. In Kpc.
NPOLY_NGC3310 := 10:  # Nb points for polynomial interpolation.
HSPEED_NGC3310 := 1.02:                    # Fitting value.
M_NGC3310 := 0.07 * 10**11 * M0 :  # NGC3310 center mass in Kg.
NBSTEP_NGC3310 := 75 :       # Nb of steps for star simulation.
NBTRAJ_NGC3310 := 3000 :     # Nb of stars for star simulation.
V0_NGC3310 := 150*10**3/kpc:       # Init speed for star simul.
DY_NGC3310 := 1/50 :          # Sampling space period in kpc.
AXANG_NGC3310 := 0.33 :     # Angle of axis for profile in rad.

# ----------------------- NGC1068 ----------------------- #

# Below is the value extrapolated from my paper          #
# supposing that the density matter has a constant slope   #
# between 0 and 1 kpc.                                     #
rho_NGC1068[1] := 1000:    rho_NGC1068[2] := 800:
rho_NGC1068[3] := 600:     rho_NGC1068[4] := 400:
rho_NGC1068[5] := 350:     rho_NGC1068[6] := 300:
rho_NGC1068[7] := 220:

# Below are the values from my paper, plain line curve.     #
rho_NGC1068[8] := 160:
rho_NGC1068[9] := 124:     rho_NGC1068[10] := 99:
rho_NGC1068[11] := 77:     rho_NGC1068[12] := 62.5:
rho_NGC1068[13] := 48.5:   rho_NGC1068[14] := 40:
rho_NGC1068[15] := 32:     rho_NGC1068[16] := 26.5:
rho_NGC1068[17] := 23:     rho_NGC1068[18] := 20:
rho_NGC1068[19] := 18:     rho_NGC1068[20] := 17:
```

```
rho_NGC1068[21] := 16:    rho_NGC1068[22] := 15.3:
rho_NGC1068[23] := 14.6:  rho_NGC1068[24] := 14.5:
rho_NGC1068[25] := 14.7:  rho_NGC1068[26] := 15:
rho_NGC1068[27] := 15.2:  rho_NGC1068[28] := 15.5:
rho_NGC1068[29] := 15.8:  rho_NGC1068[30] := 16:
rho_NGC1068[31] := 16.5:  rho_NGC1068[32] := 16.8:
rho_NGC1068[33] := 17.05: rho_NGC1068[34] := 17.3:
rho_NGC1068[35] := 17.5:  rho_NGC1068[36] := 17.8:
rho_NGC1068[37] := 18:    rho_NGC1068[38] := 18.1:
rho_NGC1068[39] := 18.1:  rho_NGC1068[40] := 18.2:

# For information only : below are the value extrapolated    #
# from my paper using a simple straight line with the last   #
# right measured slope.                                      #
rho_NGC1068[41] := 18.21:  rho_NGC1068[42] := 18.22:
rho_NGC1068[43] := 18.23:  rho_NGC1068[44] := 18.24:
rho_NGC1068[45] := 18.25:  rho_NGC1068[46] := 18.26:
rho_NGC1068[47] := 18.27:  rho_NGC1068[48] := 18.28:
rho_NGC1068[49] := 18.29:  rho_NGC1068[50] := 18.3:

# ---- Below is the value extrapolated from my paper -------- #
# supposing that the density matter is lowering after 20 kpc. #
# This indice 46 is the only used value by polynomial         #
# interpolation in this extrapolated range.                   #

rho_NGC1068[46] := 10:          # Needed value when ProgMode = 10
                                        # (smoothed force).
rho_NGC1068[46] := 18:          # Most natural extrapolated value
                # when ProMode = 10 (simulated stra trajectories).

IMIN_NGC1068 := 1:          # First valid indice for rho array.
IMAX_NGC1068 := 50:         # Last valid indice for rho array.
LMIN_NGC1068 := 0.1388:        # Distance corresponding to IMIN.
LMAX_NGC1068 := 6.97:         # Distance corresponding to IMAX.
LMINPL_NGC1068 := 0.01:          # Best value for ModProg = 24.
LMINPL_NGC1068 := 0.1:           # Best value for ModProg = 10.
LMAXPL_NGC1068 := 6:          # Max value for x, for plotting.
NPOLY_NGC1068 := 10:    # Nb points for polynomial interpolation.
HSPEED_NGC1068 := 23.5:                         # Fitting value.
M_NGC1068 := 10**10 * M0 :          # NGC1068 center mass in Kg.
#M_NGC1068 := 7.5 * 10**10 * M0 :  # For having measured values.
NBSTEP_NGC1068 := 480 :        # Nb of steps for star simulation.
NBTRAJ_NGC1068 := 3000 :       # Nb of stars for star simulation.
V0_NGC1068 := 80*10**3/kpc:          # Init speed for star simul.
DY_NGC1068 := 1/400:             # Sampling space period in kpc.
AXANG_NGC1068 := 0.16 :       # Angle of axis for profile in rad.

# ------------ Matter density unit conversion --------------- #
alpha := 1.35 * 10**(-24): # Ratio matter density/paper_length
                            # unit : g cm(-3) mm(-1)
alpha2 := 10**3 * kpc**3:  # Ratio for converting matter density
                            # from g cm(-3) to kg kpc(-3).

MatDensUnitConv := proc( rho, IMin, IMax )
  for n from IMin to IMax
  do
    rho[n] := rho[n] * alpha * alpha2:
```

```
   end do:         # converting from mm on paper to Kg/m3 unit.
end proc:

MatDensUnitConv (rho_NGC3310, IMIN_NGC3310, IMAX_NGC3310):
MatDensUnitConv (rho_NGC1068, IMIN_NGC1068, IMAX_NGC1068):

# ----------------- Speed curve calculations ---------------- #
GalaxySpeedCalcul( M_NGC3310, rho_NGC3310, IMIN_NGC3310,
                   IMAX_NGC3310, LMIN_NGC3310, LMAX_NGC3310,
                   LMINPL_NGC3310, LMAXPL_NGC3310,
                   NPOLY_NGC3310, HSPEED_NGC3310, ProgMode,
                   NBSTEP_NGC3310, NBTRAJ_NGC3310,
                   V0_NGC3310, DY_NGC3310, AXANG_NGC3310,
                   NGC_3310_theoretical_speed_curve );
GalaxySpeedCalcul( M_NGC1068, rho_NGC1068, IMIN_NGC1068,
                   IMAX_NGC1068, LMIN_NGC1068, LMAX_NGC1068,
                   LMINPL_NGC1068, LMAXPL_NGC1068, NPOLY_NGC1068,
                   HSPEED_NGC1068, ProgMode,
                   NBSTEP_NGC1068, NBTRAJ_NGC1068,
                   V0_NGC1068, DY_NGC1068, AXANG_NGC1068,
                   NGC_1068_theoretical_speed_curve );




###############################################################
######################### NGC157 ##############################
###############################################################

# ----------------- Matter density values ------------------ #
# Those values are read on a printed paper of the density    #
# matter of the galaxy. Hence their unit is the mm.          #

rho_NGC157[1] := 69.5:    rho_NGC157[2] := 66:
rho_NGC157[3] := 58.5:    rho_NGC157[4] := 50:
rho_NGC157[5] := 41.5:    rho_NGC157[6] := 33.5:
rho_NGC157[7] := 25:      rho_NGC157[8] := 18.5:
rho_NGC157[9] := 12.5:    rho_NGC157[10] := 8.8:
rho_NGC157[11] := 5:      rho_NGC157[12] := 3:
rho_NGC157[13] := 2.1:    rho_NGC157[14] := 1.4:
rho_NGC157[15] := 1.1:    rho_NGC157[16] := 1.05:
rho_NGC157[17] := 1:      rho_NGC157[18] := 0.6:
rho_NGC157[19] := 0.3:    rho_NGC157[20] := 0.3:
rho_NGC157[21] := 0.6:    rho_NGC157[22] := 3:

IMIN_NGC157 := 1:              # First valid indice for rho array.
IMAX_NGC157 := 22:            # Last valid indice for rho array.
LMIN_NGC157 := 0:             # Distance corresponding to IMin.
LMAX_NGC157 := 26.35:         # Distance corresponding to IMax.
LMINPL_NGC157 := 0.8:          # Min value for x, for plotting.
LMAXPL_NGC157 := 21.5:         # Max value for x, for plotting.
NPOLY_NGC157 := 6:      # Nb points for polynomial interpolation.
HSPEED_NGC157 := 4.8:                       # Fitting value.
M_NGC157 := 5 * 10**11 * M0:      # Fitted NGC157 center mass.
                   # No correct available measured value. Kg.
NBSTEP_NGC157 := 200 :       # Nb of steps for star simulation.
NBTRAJ_NGC157 := 500 :       # Nb of stars for star simulation.
V0_NGC157 := 180*10**3/kpc:       # Initial speed for star simul.
```

```
DY_NGC157 := 1/20 :                # Sampling space period in kpc.
AXANG_NGC157 := 0.4 :          # Angle of axis for profile in rad.

# ------------ Matter density unit conversion ---------------- #
paperhigh := 69.5:  # Paper length for rel dens matter = 1 (mm).
alpha := 0.57 * M0 * 10**9 / paperhigh:
                               # Ratio matter density/paper_length
                               # unit : Kg kpc(-3) mm(-1)
for n from IMIN_NGC157 to IMAX_NGC157
do
  rho_NGC157[n] := rho_NGC157[n] * alpha:
end do:             # converting from mm on paper to Kg/m3 unit.
        # Remark : this matter density is quite the same range
        # of values as the one of the NGC3310.

# ------------------ Speed curve calculations ---------------- #
GalaxySpeedCalcul( M_NGC157, rho_NGC157, IMIN_NGC157,
                   IMAX_NGC157, LMIN_NGC157, LMAX_NGC157,
                   LMINPL_NGC157, LMAXPL_NGC157, NPOLY_NGC157,
                   HSPEED_NGC157, ProgMode,
                   NBSTEP_NGC157, NBTRAJ_NGC157,
                   V0_NGC157, DY_NGC157, AXANG_NGC157,
                   NGC_157_theoretical_speed_curve );



##############################################################
####################### NGC7541 #############################
##############################################################

# ----------------- Matter density values ------------------ #
# Those values are read on a printed paper of the density    #
# matter of the galaxy. Hence their unit is the mm.          #

# Below values of matter density on paper for 0 < x < 2 Kpc.  #
rho_NGC7541[1] := 126:    rho_NGC7541[2] := 124.8:
rho_NGC7541[3] := 120.5:  rho_NGC7541[4] := 114:
rho_NGC7541[5] := 104.5:  rho_NGC7541[6] := 86:
rho_NGC7541[7] := 78.5:   rho_NGC7541[8] := 75:
rho_NGC7541[9] := 73.8:   rho_NGC7541[10] := 72.5:
rho_NGC7541[11] := 71.5:  rho_NGC7541[12] := 70.5:
rho_NGC7541[13] := 70:

IMIN_NGC7541 := 1:            # First valid indice for rho array.
IMAX_NGC7541 := 91:          # Last valid indice for rho array.

# Below are the values of matter density on the paper for x   #
# between 2 and 25 kpc. I see a straight line on the paper    #
# for x in this range of values.                             #
for n from 14 to IMAX_NGC7541
do
  rho_NGC7541[n] := (34-70)*(n-13)/(91-13) + 70:
end do:

LMIN_NGC7541 := 0:            # Distance corresponding to IMin.
LMAX_NGC7541 := 25:           # Distance corresponding to IMax.
LMINPL_NGC7541 := 0.3:          # Best value for ModProg = 24.
```

```
LMINPL_NGC7541 := 0.45:              # Best value for ModProg = 10.
LMINPL_NGC7541 := 0.2:               # Best value for ModProg = 11.
LMAXPL_NGC7541 := 23.2:          # Max value for x, for plotting.
NPOLY_NGC7541 := 31:   # Nb points for polynomial interpolation.
HSPEED_NGC7541 := 3.8:                         # Fitting value.
M_NGC7541 := 4.6 * 10**10 * M0:         # NGC7541 center mass !?
            # No available measured value. Kg. =total_mass/10 ?

# Warning: with the values below, this prog must run around
# 20 hours on a dual core 2 G Hz PC (but no parallelization on
# the dual core!).
NBSTEP_NGC7541 := 4200 :       # Nb of steps for star simulation.
NBTRAJ_NGC7541 := 2000 :       # Nb of stars for star simulation.
V0_NGC7541 := 287*10**3/kpc:     # Initial speed for star simul.
DY_NGC7541 := 1/78 :             # Sampling space period in kpc.
AXANG_NGC7541 := 0.94 :        # Angle of axis for profile in rad.

# ------------ Matter density unit conversion --------------- #
alpha := 2 * 10**36:      # Ratio matter density/paper_length
          # FITTED VALUE BECAUSE NO MATTER DENSITY AVALAIBLE HERE
                     # for this galaxy. Unit : Kg m(-3) mm(-1).
for n from IMIN_NGC7541 to IMAX_NGC7541
do
  rho_NGC7541[n] := rho_NGC7541[n] * alpha:
end do:             # converting from mm on paper to Kg/m3 unit.

# ----------------- Speed curve calculations --------------- #
GalaxySpeedCalcul( M_NGC7541, rho_NGC7541, IMIN_NGC7541,
                   IMAX_NGC7541, LMIN_NGC7541, LMAX_NGC7541,
                   LMINPL_NGC7541, LMAXPL_NGC7541, NPOLY_NGC7541,
                   HSPEED_NGC7541, ProgMode,
                   NBSTEP_NGC7541, NBTRAJ_NGC7541,
                   V0_NGC7541, DY_NGC7541, AXANG_NGC7541,
                   NGC_7541_theoretical_speed_curve );



##################################################################
####################### NGC7331 ##################################
##################################################################

# ---------------- Matter density values ------------------ #
# Those values are read on a printed paper of the density   #
# matter of the galaxy. Hence their unit is the mm.         #

# Below values of matter density on paper for 0 < x < 14 Kpc. #
rho_NGC7331[1] := 145:     rho_NGC7331[2] := 128:
rho_NGC7331[3] := 115:     rho_NGC7331[4] := 97:
rho_NGC7331[5] := 85:      rho_NGC7331[6] := 64:
rho_NGC7331[7] := 62.7:    rho_NGC7331[8] := 54:
rho_NGC7331[9] := 48:      rho_NGC7331[10] := 41.5:
rho_NGC7331[11] := 35:     rho_NGC7331[12] := 30:
rho_NGC7331[13] := 25:     rho_NGC7331[14] := 21.5:
rho_NGC7331[15] := 17.5:   rho_NGC7331[16] := 15:
rho_NGC7331[17] := 11.5:   rho_NGC7331[18] := 9.5:
rho_NGC7331[19] := 8:      rho_NGC7331[20] := 6.5:
rho_NGC7331[21] := 4.5:    rho_NGC7331[22] := 4:
```

```
rho_NGC7331[23] := 2.5:    rho_NGC7331[24] := 1.8:
rho_NGC7331[25] := 1.5:    rho_NGC7331[26] := 1.2:
rho_NGC7331[27] := 1:      rho_NGC7331[28] := 0.6:
rho_NGC7331[29] := 0.3:    rho_NGC7331[30] := 0.24:
rho_NGC7331[31] := 0.2:    rho_NGC7331[32] := 0.16:
rho_NGC7331[33] := 0.12:   rho_NGC7331[34] := 0.113:
rho_NGC7331[35] := 0.11:   rho_NGC7331[36] := 0.103:
rho_NGC7331[37] := 0.1:

IMIN_NGC7331 := 1:          # First valid indice for rho array.
IMAX_NGC7331 := 81:          # Last valid indice for rho array.

# Below are the values of matter density on the paper for x    #
# between 15 and 35 kpc. I see a straight line on the paper    #
# for x in this range of values.                               #
for n from 38 to IMAX_NGC7331
do
   rho_NGC7331[n] := (0.01-0.1)*(n-37)/(IMAX_NGC7331-37) + 0.1:
end do:

LMIN_NGC7331 := 0:             # Distance corresponding to IMin.
LMAX_NGC7331 := 33:            # Distance corresponding to IMax.
LMINPL_NGC7331 := 0.01:          # Best value for ModProg = 24.
LMINPL_NGC7331 := 0.1:           # Best value for ModProg = 10.
LMINPL_NGC7331 := 0.2:           # Min value for x, for plotting.
LMAXPL_NGC7331 := 17.5: # No more because pb with polyn interpol
NPOLY_NGC7331 := 14:    # Nb points for polynomial interpolation.
HSPEED_NGC7331 := 7:                          # Fitting value.
M_NGC7331 := 0.43 * 10**11 * M0 :    # NGC7331 center mass in Kg?
                              # No available measured value. Kg.
# With values below: calculation for around 20 hours.
NBSTEP_NGC7331 := 3600 :       # Nb of steps for star simulation.
NBTRAJ_NGC7331 := 3000 :       # Nb of stars for star simulation.
V0_NGC7331 := 280*10**3/kpc:     # Initial speed for star simul.
DY_NGC7331 := 1/200 :          # Sampling space period in kpc.
AXANG_NGC7331 := 0.72 :        # Angle of axis for profile in rad.

# ------------ Matter density unit conversion --------------- #
alpha := 3.6*M0*10**9/145:    # Ratio matter density/paper_lgth
                   # for this galaxy. Unit : Kg kpc(-3) mm(-1).
for n from IMIN_NGC7331 to IMAX_NGC7331
do
   rho_NGC7331[n] := rho_NGC7331[n] * alpha:
end do:               # converting from mm on paper to Kg/m3 unit.

# ----------------- Speed curve calculations --------------- #
GalaxySpeedCalcul( M_NGC7331, rho_NGC7331, IMIN_NGC7331,
                  IMAX_NGC7331, LMIN_NGC7331, LMAX_NGC7331,
                  LMINPL_NGC7331, LMAXPL_NGC7331, NPOLY_NGC7331,
                  HSPEED_NGC7331, ProgMode,
                  NBSTEP_NGC7331, NBTRAJ_NGC7331,
                  V0_NGC7331, DY_NGC7331, AXANG_NGC7331,
                  NGC_7331_theoretical_speed_curve );
# F.Lassiaille may 2010.
```

REFERENCES

[1]   G. Galletta, and E.Recillas-Cruz, "The large scale trend of rotation curves in the spiral galaxies NGC 1068 and NGC 3310" (1982), http://adsabs.harvard.edu/full/1982A&A...112..361G.

[2]   C.P.Blackman, "Surface photometry and mass distributions of spiral galaxies – I NGC 157" (1978), http://adsabs.harvard.edu/full/1979MNRAS.186..701B.

[3]   G.A. Kyazumov, "The velocity field of the galaxy NGC 7541" (1980), http://adsabs.harvard.edu/full/1980SvAL....6..220K.

[4]   <<Surface photometry and mass distributions of spiral galaxies – II. NGC 1084 and NGC 7331>>, C.P. Blackman, 1978, http://adsabs.harvard.edu/full/1979MNRAS.186..717B.

[5]   F.Lassiaille, "Three elements theory" (1999), http://lumi.chez-alice.fr/3elt.pdf.

[6]   <<A solution for the "dark matter mystery" based on Euclidean relativity>>, F. Lassiaille, Feb 2010, http://lumi.chez-alice.fr/anglais/mystmass.pdf>>.

TABLE OF CONTENTS

F.Lassiaille, may 2010

TABLE OF FIGURES

F.Lassiaille, may 2010