Revista Mexicana de Astronomía y Astrofísica, **58**, 321–332 (2022)
© 2022: Instituto de Astronomía, Universidad Nacional Autónoma de México
https://doi.org/10.22201/ia.01851101p.2022.58.02.13

ANN AND ANALYTICAL SOLUTIONS TO RELATIVISTIC ISOTHERMAL GAS SPHERES

Mohamed I. Nouh[1], Yosry A. Azzam[1], Emad A-B Abdel-Salam[2], Farag I. Elnagahy[1], and Tarek M. Kamel[1]

Received May 19 2022; accepted July 7 2022

ABSTRACT

Relativistic isothermal gas spheres are a powerful tool to model many astronomical objects, like compact stars and clusters of galaxies. In the present paper, we introduce an artificial neural network (ANN) algorithm and Taylor series to model the relativistic gas spheres using Tolman-Oppenheimer-Volkoff differential equations (TOV). Comparing the analytical solutions with the numerical ones revealed good agreement with maximum relative errors of 10^{-3}. The ANN algorithm implements a three-layer feed-forward neural network built using a back-propagation learning technique that is based on the gradient descent rule. We analyzed the mass-radius relations and the density profiles of the relativistic isothermal gas spheres against different relativistic parameters and compared the ANN solutions with the analytical ones. The comparison between the two solutions reflects the efficiency of using the ANN to solve TOV equations.

RESUMEN

Las esferas isotérmicas relativistas son una herramienta poderosa para modelar objetos astronómicos, tales como estrellas compactas y cúmulos de galaxias. En este trabajo presentamos un algoritmo basado en una red neuronal artificial (ANN) y series de Taylor para modelar esferas de gas relativistas usando las ecuaciones diferenciales de Tolman-Oppenheimer-Volkoff (TOV). La comparación de las soluciones analíticas con las numéricas muestra una buena concordancia, con errores relativos máximos de 10^{-3}. El algoritmo ANN implementa una red neuronal de tres niveles con pro-alimentación, construída usando una técnica de aprendizaje con retro-propagación basada en la regla del gradiente descendente. Analizamos las relaciones masa-radio y los perfiles de densidad de las esferas relativistas isotérmicas y comparamos las soluciones ANN con las analíticas. Esta comparación muestra la eficiencia de ANN para resolver las ecuaciones TOV.

Key Words: methods: analytical — methods: miscellaneous — relativistic processes — stars: interiors

1. INTRODUCTION

Many astrophysical issues, particularly those involving star structure and galactic dynamics, benefit from the use of isothermal models (Binney & Tremaine 1987; Chandrasekhar 1939). In terms of stellar structure and evolution theory, isothermal self-gravitating spheres may be used to calculate the behaviour of physical variables.

There are a lot of studies concerning the isothermal spheres in the framework of Newtonian mechanics (non-relativistic) (Milgrom 1984; Liu 1996; Natarajan & Lynden-Bell 1997; Roxburgh & Stockman 1999; Hunter 2001; Raga et al. 2013). Several works have conducted numerical investigations on relativistic isothermal spheres within the framework of general relativity theory. Chavanis (2002) explored the structure and stability of

[1] Astronomy Department, National Research Institute of Astronomy and Geophysics (NRIAG), Cairo, Egypt.
[2] Department of Mathematics, Faculty of Science, New Valley University, Egypt.

isothermal gas spheres using a linear equation of state in the context of general relativity. Sharma (1990) used the Pade approximation technique to provide straightforward and accessible approximate analytical solutions to the TOV equation of hydrostatic equilibrium, and his results indicate that general relativity isothermal arrangements have a limited extent. Saad (2017) proposed a novel approximate analytical solution to the TOV equation using a combination of the Euler-Abel transformation and the Pade approximation. Besides the numerical and analytical solutions of similar nonlinear differential equations similar to TOV equations, artificial intelligence approaches presented reliable results in many problems arising in astrophysics. In this context, Morawski & Bejger (2020) proposed a unique method based on ANN algorithms for reconstructing the neutron star equation of state from the observed mass-radius relationship. Nouh et al. (2020) presented a neural network-based computational approach for solving fractional Lane-Emden differential equation problems. Azzam et al. (2021) used an artificial neural network (ANN) approach and simulate the conformable fractional isothermal gas spheres and compared them with the results of the analytical solution deduced using the Taylor series. Abdel-Salam et al. (2021) presented the neural network (NN) mathematical model and developed a neural network approach for simulating the helium burning network using a feed-forward mechanism.

In the present article, we solve TOV equations by the ANN and analytically by an accelerated Taylor series. We employ an ordinary feed-forward neural network to estimate the solution of TOV equations for the ANN simulation, which has been shown to outperform competing computational approaches. A three-layer feed-forward neural network is used, which was trained using a back-propagation learning approach based on the gradient descent rule. The following is the paper's structure: The relativistic isothermal polytrope is discussed in § 2. The analytical solution to the TOV equation is found in § 3. The mathematical modelling of the ANN is covered in § 4. § 5 summarizes the results obtained, and § 6 presents the conclusion.

2. THE RELATIVISTIC ISOTHERMAL GAS SPHERE (RIGS)

In the polytropic equation of state ($P = K\rho^{1+1/n}$, where P is the pressure, ρ is the density, and K is the polytropic constant), the polytropic index n ranges from 0 to ∞. When n approaches or equals ∞, the isothermal equation of state $P = K\rho$ emerges. By combining the isothermal equation of state with the equation of the hydrostatic equilibrium in the frame of the general relativity, the TOV equation of the isothermal gas sphere could be given as (Sharma 1990),

$$\frac{dP}{dr} = -\left(\frac{GM_r}{r^2}\right)\frac{\rho + (P/c^2)(1 + (4\pi P r^3/M_r c^2))}{(1 - (2GM_r/rc^2))}, \tag{1}$$

where M_r is the total mass energy or 'effective mass' of the star of radius r including its gravitational field is given by

$$M_r = \frac{4\pi}{c^2}\int\limits_0^r \rho c^2 r^2 dr. \tag{2}$$

Define the variables, χ, ν, u, and the relativistic parameter s, by

$$\begin{aligned} &\chi = rA; \\ &A^2 \ = 4\pi G\rho_c/sc^2; \\ &s = \tfrac{P_c}{\rho_c c^2}, \end{aligned} \tag{3}$$

where ρ_c and P_c are the central density and central pressure of the star respectively, u(x) and $\nu(x)$ are the Emden and is the mass functions of radius, and c is the speed of light. Equations (1) and (2) can be transformed into the dimensionless forms in the (x, u) plane as

$$\frac{(1 - 2s\nu(x)/x)}{1+\sigma} + x^2\frac{du}{dx} - \nu(x) - se^{-u}x^3 = 0 \tag{4}$$

$$\frac{d\nu}{dx} = x^2 e^{-u}, \tag{5}$$

which satisfy the initial conditions

$$u(0) = 0; \frac{du(0)}{dx} = 0; \nu(0) = 0. \tag{6}$$

If the pressure is substantially lower than the energy density at the center of a star (i.e., goes to zero), as in the non-relativistic case, the system of equations (4) and (5) simplifies to the Newtonian isothermal structure equations

$$\frac{1}{x^2}\frac{d}{dx}(x^2\frac{du}{dx}) = e^{-u}. \tag{7}$$

One can calculate physical parameters of the stellar models like radius (R), density (ρ) , and mass M_r using the following equations

$$\begin{aligned} &R = xc\sqrt{\tfrac{s}{4\pi G\rho_c}} \\ &\rho = \rho_c e^{-u}; \\ &M_r = \tfrac{4\pi\rho_c}{A^2}\nu(x). \end{aligned} \tag{8}$$

3. ANALYTICAL SOLUTION TO EQUATION (4)

Rearrange equations (4) and (5) as

$$\begin{aligned} &x^2\tfrac{du}{dx} - 2svx\tfrac{du}{dx} - \nu - \nu s - sx^3e^{-u} - s^2x^3e^{-u} = 0, \\ &\tfrac{d\nu}{dx} = x^2e^{-u}, \end{aligned} \tag{9}$$

where the initial conditions are

$$u(0) = 0; \quad \frac{du(0)}{dx} = 0; \quad \upsilon(0) = 0.$$

Write the Taylor series for the function $u(x)$ as

$$u(x) = u(0) + D_xu(0)x + \frac{D_x^2u(0)}{2!}x^2 + \frac{D_x^3u(0)}{3!}x^3 + \frac{D_x^4u(0)}{4!}x^4 + \frac{D_x^5u(0)}{5!}x^5 + ..., \tag{10}$$

at $x = 0$ the first derivative is given by

$$\nu_0' = (0)^2e^{-u(0)} = 0,$$

and the second derivative is

$$\nu_0'' = 2(0)e^{-u(0)} - (0)^2e^{-u(0)}u_0' = 0,$$

differentiating another time, we have

$$\nu''' = 2e^{-u} - 4xe^{-u}u' + x^2e^{-u}u'^2 - x^2e^{-u}u''; \tag{11}$$

when $x = 0$ we have

$$\nu_0''' = 2e^{-u(0)} - 4(0)e^{-u(0)}u'(0) + (0)^2e^{-u(0)}u'(0)^2 - (0)^2e^{-u(0)}u''(0) = 2(1) = 2;$$

the fourth derivative is given by

$$\nu^{(4)} = -6e^{-u}u' + 6xe^{-u}u'^2 - 6xe^{-u}u'' - x^2e^{-u}u'^3 + 3x^2e^{-u}u'u'' - x^2e^{-u}u''', \tag{12}$$

when $x = 0$ we have $\nu_0^{(4)} = 0$;
the fifth derivative is given by

$$\begin{aligned} \nu^{(5)} = {}& 6e^{-u}u'^2 - 6e^{-u}u'' + 6e^{-u}u'^2 - 6xe^{-u}u'^3 + 12xe^{-u}u'u'' - 6e^{-u}u'' + 6xe^{-u}u'u'' \\ & -6xe^{-u}u''' - 2xe^{-u}u'^3 + x^2e^{-u}u'^4 - 3x^2e^{-u}u'^2u'' + 6xe^{-u}u'u'' - 3x^2e^{-u}u'^2u'' \\ & +3x^2e^{-u}u''^2 + 3x^2e^{-u}u'u''' - 2xe^{-u}u''' + x^2e^{-u}u'u''' - x^2e^{-u}u^{(4)}. \end{aligned}$$

Substituting $x = 0$ we have

$$\nu_0^{(5)} = 12e^{-u_0}(u_0')^2 - 12e^{-u_0}u_0'' = -12u_0'',$$

and so on. Now write the first TOV equation of equation (9) as

$$x^2u' - 2s\upsilon xu' - \upsilon - \nu s - sx^3e^{-u} - s^2x^3e^{-u} = 0. \tag{13}$$

Differentiating equation (13) for x we have

$$2xu' + x^2u'' - 2s\nu u' - 2s\nu' xu' - 2s\nu xu'' - \nu' - s\nu' - 3sx^2e^{-u} + sx^3e^{-u}u' = 0; \tag{14}$$

when $x = 0$ all terms are equal to zero.
Differentiating equation (14) for x we have

$$\begin{aligned}&2u' + 2xu'' + 2xu'' + x^2u''' - 2s\nu'u' - 2s\nu u'' - 2s\nu'u' - 2s\nu''xu' - 2s\nu'xu''\\ &-2s\nu u'' - 2s\nu'xu'' - 2s\zeta xu''' - \nu'' - s\nu'' - 6sxe^{-u} + 3sx^2e^{-u}u' + 3sx^2e^{-u}u'\\ &-sx^3e^{-u}u'^2 + sx^3e^{-u}u'' = 0;\end{aligned}$$

which can be simplified into

$$\begin{aligned}&2u' + 4xu'' + x^2u''' - 4s\nu'u' - 4s\upsilon u'' - 2s\nu''xu' - 4s\nu'xu'' - 2s\nu xu'''\\ &-\nu'' - s\nu'' - 6sxe^{-u} + 6sx^2e^{-u}u' - sx^3e^{-u}u'^2 + sx^3e^{-u}u'' = 0.\end{aligned} \tag{15}$$

Putting $x = 0$ we have

$$\begin{aligned}&2u_0' + 4(0)u_0'' + (0)^2u_0''' - 4s\nu_0'u_0' - 4s\upsilon_0u_0'' - 2s(0)\nu_0''u_0' - 4s(0)\nu_0'u_0''\\ &-2s(0)\nu_0u_0''' - \nu_0'' - s\nu_0'' - 6s(0)e^{-u_0} + 6s(0)^2e^{-u_0}u_0' - s(0)^3e^{-u_0}(u_0')^2 + s(0)^3e^{-u_0}u_0'' = 0.\end{aligned}$$

Since

$$u_0 = 0,\ \nu_0 = 0,\ \nu_0' = 0,\ \nu_0'' = 0,$$

this gives

$$u_0' = 0.$$

By differentiating equation (15) and putting $x = 0$ this gives

$$u_0'' = \frac{1}{3} + \frac{1}{3}s + s + s^2 = \frac{1}{3}(1 + s)(1 + 3s).$$

Following the same procedure mentioned above, we obtained the values of $u_0''', u_0^{(4)}, u_0^{(5)}, \nu_0^{(6)}, u_0^{(6)}$ and so on. Substituting these values into the Taylor series, equation (10), the Emden function is given by

$$u(x) = \frac{1}{6}(1 + s)(1 + 3s)x^2 + ... \tag{16}$$

4. THE ANN ALGORITHM

4.1. *Simulation of the RIGS*

The proposed ANN simulation scheme of the RIGS is plotted in Figure 1. First, we assume $u_t = (x, p)$ to be the neural network's approximate solution to equation (4), which may be expressed as follows (Yadav et al. 2015):

$$u_t(x, p) = f(x, N(x, p)) + A(x), \tag{17}$$

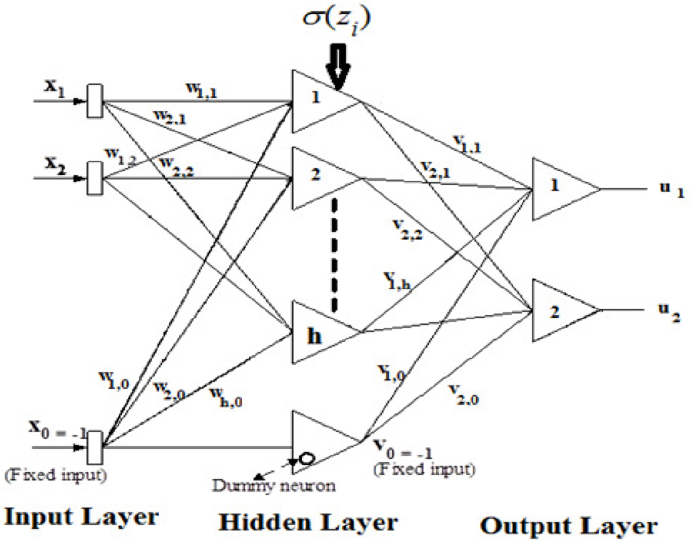

Fig. 1. Proposed ANN architecture for simulating the isothermal gas sphere.

where the first term represents a feed-forward neural network with input vector x and p is the corresponding vector of adjustable weight parameters, and the second term represents the boundary or initial value.
The ANN output, $N(x, p)$, is provided by

$$N(x,p) = \sum\nolimits_{i=1}^{H} \nu_i \sigma(z_i), \tag{18}$$

where $z_j = \sum_{i=1}^{n} w_{ij} x_j + \beta_i$, and w_{ij} represents the weight from the input unit j to the hidden unit i, ν_i represents the weight from the hidden unit i to the output, β_i is the bias of the i^{th} hidden unit, and $\sigma(z_i)$ is the sigmoid activation function that has the form $\sigma(x) = \frac{1}{1+e^{-x}}$.

Write the derivative of networks output N for the input vector x_j as

$$\begin{aligned} &D_{xj} N(x,p) = D_{xj} (\textstyle\sum_{i=1}^{H} \nu_i \sigma (z_j = \sum_{i=1}^{n} w_{ij} x_j + \beta_i,)) \\ &= \textstyle\sum_{i=1}^{h} \nu_i w_{ij} \sigma^1, \sigma^1 = D_x \sigma(x) \end{aligned} \tag{19}$$

and the n^{th} derivative of N, equation (18), is

$$D_{xj}^{(n)} N(x,p) = \sum\nolimits_{i=1}^{n} \nu_i P_i \sigma_i^{(n)}, \tag{20}$$

where

$$P_i = \prod\nolimits_{k=1}^{n} w_{ik}, \sigma_i = \sigma(z_i).$$

As a result, the proposed solution for the TOV equations is given by

$$u_t(x,p) = x^2 N(x,p), \tag{21}$$

$$\nu_t(x,p) = x^3 M(x,p), \tag{22}$$

this fulfills the initial conditions as

$$u_t(0,p) = 0 \cdot N(0,p) = 0,\ \nu_t(0,p) = (0)^3 M(0,p) = 0.$$

And

$$\begin{aligned} D_x u_t(x,p) &= 2xN(x,p) + x^2 D_x N(x,p), \\ D_x \nu_t(x,p) &= 3x^2 M(x,p) + x^3 D_x M(x,p), \end{aligned} \tag{23}$$

$$\begin{aligned} D_x u_t(0,p) &= 2(0)N(0,p) + 0.D_x N(0,p), \\ D_x \nu_t(0,p) &= 3(0)^2 M(0,p) + (0)^3 D_x M(0,p). \end{aligned} \tag{24}$$

4.2. *ANN Gradient Computations and Updating the Parameters*

When we examine the solution presented by equations (21), we transform the problem to an unconstrained optimization one, and the error to be minimised is given by (Yadav et al. 2015).

$$\begin{aligned} E(x) = \sum_i & \{x^2 D_x u_t(x_i,p) - 2\sigma x \nu_t(x_i,p) D_x u_t(x_i,p) - \nu_t(x_i,p) \\ & -\sigma \nu_t(x_i,p) u_t(x_i,p) - \sigma x^3 e^{-u_t(x_i,p)} - \sigma^2 x^3 e^{-u_t(x_i,p)}\}^2 \\ & + \sum_i \{D_x \nu_t(x_i,p) - x^2 e^{-u_t(x_i,p)}\}^2, \end{aligned} \tag{25}$$

where

$$\begin{aligned} D_x u_t(x,p) &= 2xN(x,p) + x^2 D_x^\alpha N(x,p), \\ D_x \nu_t(x,p) &= 3x^2 M(x,p) + x^3 D_x M(x,p). \end{aligned} \tag{26}$$

To upgrade network parameters, we calculate the NN derivative for both input and the parameters of the network and train the NN for the optimum parameter value. Once the network has been trained, one can optimize the network parameters and compute $u_t(x,p) u_t(x,p) = x^2 N(x,p)$.

The derivative of any of its inputs is analogous to the feed-forward neural network N with one hidden layer, the same weights w_{ij}, and thresholds β_i, and each weight ν_i replaced with $\nu_i P_i$ where $P_i = \prod_{k=1}^n w_{ik}^{\alpha k}$. Moreover, the sigmoid function's nth order derivative is substituted for the transfer function of each hidden unit. As a result, the gradient with respect to the original network parameters may be determined as follows:

$$\begin{aligned} & D_{\nu i} N = P_i \sigma_i^{(n)}, \\ & D_{\beta i} N = \nu_i P_i \sigma_i^{(n+1)}, \\ & D_{wij} N = x_i \nu_i P_i \sigma_i^{(n+1)} + \nu_i (\textstyle\prod_{k=1, k\neq j} w_{ik}) \sigma_i^{(n)}. \end{aligned} \tag{27}$$

The updating rule of the network parameters will be given as follows

$$\nu_i(x+1) = \nu_i(x) + a D_{\nu i} N, \tag{28}$$

$$\beta_i(x+1) = \beta_i(x) + b D_{\beta i} N, \tag{29}$$

$$w_{ij}(x+1) = w_{ij}(x) + c D_{wij} N, \tag{30}$$

where a, b, c are learning rates, $i = 1, 2, ..., n$, and $j = 1, 2, .., h$.

An ANN's main processing unit is the neuron, which is capable of carrying out local information and processing local memory. The net input (z) of each neuron is calculated by adding the weights it receives to form a weighted sum of such inputs and then adding them with a bias (β). The net input (z) is then processed by an activation function, which is likely to result in neuron output (shown in Figure 1).

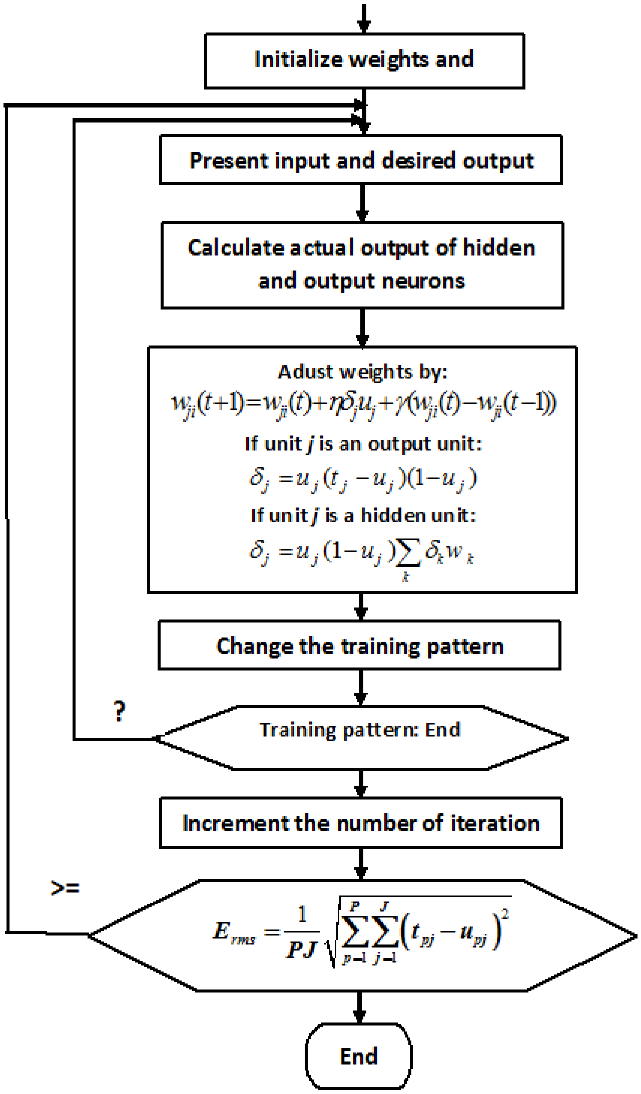
Initialize weights and
Present input and desired output
Calculate actual output of hidden and output neurons
Adust weights by:
$w_{ji}(t+1)=w_{ji}(t)+\eta\delta_j u_j+\gamma(w_{ji}(t)-w_{ji}(t-1))$
If unit *j* is an output unit:
$\delta_j = u_j(t_j - u_j)(1-u_j)$
If unit *j* is a hidden unit:
$\delta_j = u_j(1-u_j)\sum_k \delta_k w_k$
Change the training pattern
?
Training pattern: End
Increment the number of iteration
>=
$E_{rms} = \frac{1}{PJ}\sqrt{\sum_{p=1}^{P}\sum_{j=1}^{J}(t_{pj}-u_{pj})^2}$
End

Fig. 2. Proposed ANN architecture for simulating the isothermal gas sphere.

4.3. *Back-Propagation Learning Algorithm*

Back-propagation (BP) is a gradient technique that seeks to minimize the mean square error between the actual and predicted outputs of a feed-forward net. It demands continuously differentiable non-linearity. Figure 2 depicts a flow chart of a back-propagation offline learning method (Leshno et al. 1993). We used a recursive technique that started with the output units and moved back to the first concealed layer. To compare the output u_j at the output layer to the desired output t_j, an error function of the following kind is used:

$$\delta_j = u_j(t_j - u_j)(1 - u_j). \tag{31}$$

For the hidden layer, the error function takes the form:

$$\delta_j = u_j(1 - u_j)\sum\nolimits_k \delta_k w_k. \tag{32}$$

where δ_j is the error term in the output layer and w_k is the weight between the hidden and output layers. The error is replicated backward from the output layer to the input layer as follows to modify the weight of each connection:

$$w_{ji}(t + 1) = w_{ji}(t) + \eta\delta_j u_j + \gamma(w_{ji}(t) - w_{ji}(t - 1)). \tag{33}$$

The rate of learning η must be chosen so that it is neither too small, resulting in a slow convergence, nor too large, resulting in misleading results. The momentum term in equation (33) is added with the momentum constant γ to accelerate the convergence of the back-propagation learning algorithm error and to aid in stomping the changes over local increases in the energy function and trying to push the weights to follow the overall downhill direction (Leshno et al. 1993). This term adds a percentage of the most recent weight adjustment to the current weight adjustments. At the start of the training phase, both η and γ terms are allocated and determine the network stability and speed (Elminir et al. 2007; Basheer & Hajmeer 2000; El-Mallawany et al. 2014).

The procedure is repeated for each input pattern until the network output error falls below a preset threshold. The final weights are frozen during the test session and utilized to calculate the precise values of both the density profile and mass-radius relation. To evaluate the training's success and quality, the error is computed for the entire batch of training patterns using the root-mean-square normalised error, which is defined as:

$$E_{rms} = \frac{1}{PJ}\sqrt{\sum\nolimits_{p=1}^{P}\sum\nolimits_{j=1}^{J}\left(t_{pj}-u_{pj}\right)^2}, \tag{34}$$

where P is the number of training patterns, J is the number of output units, t_{pj} is the target output at unit j, and u_{pj} is the actual output at the same unit j. Zero error would indicate that all of the ANN's output patterns completely match the predicted values and that the ANN is completely trained. Internal unit thresholds are also modified by assuming they are connection weights on links derived from an auxiliary constant-valued input.

5. RESULTS AND DISCUSSIONS

5.1. *Training Data Preparation*

To prepare the data for the ANN simulation of the problem, we elaborated a Mathematica code to derive the Emden and mass functions of the relativistic isothermal gas spheres using the Taylor series. We performed the calculations for the range of the dimensionless parameter $x = 0 - 35$; this upper limit of x corresponds to an isothermal sphere on the verge of gravothermal collapse (Antonov 1985). To obtain an accurate result for these analytical physical parameters, we accelerated both the series expansions of the Emden function (u) and the mass function (ν) by the accelerated scheme suggested by Nouh (2004); Saad et al. (2021). The radius, density, and mass of the gas sphere are calculated using equations (8).

We calculated 50 gas models for the range of the relativistic parameter $\sigma = 0 - 0.5$ with step 0.01. The radius, density, and mass of the star are calculated for the typical neutron star physical parameters: mass $M_* = 1.5M_\odot$, central density $\rho_c = 5.75 \times 10^{14}\ \mathrm{g\,cm^{-3}}$, central pressure $P_c = 2 \times 10^{33}$ par, and radius $R_* = 1.4 \times 10^6$ cm. We plotted in Figure 3 the density profiles and the mass-radius relations of the relativistic isothermal gas spheres. Because the effect of the relativistic parameter on the Emden function is smaller than that on the mass function (the relation between the density and the Emden function is given by $\rho = \rho_c e^{-u}$), we plotted the density profiles for the relativistic parameter values $\sigma = 0.1$, 0.2, 0.3, and 0.4 only.

5.2. *Network Training*

The data calculated in the previous section have been used to train the NN we used to simulate the Emden and mass functions of the relativistic isothermal gas spheres. We trained the ANN that has the architecture previously shown in Figure 1 for such a purpose. The network is composed of three layers which are the input layer, the hidden layer, and the output layer. The input layer has two inputs which symbolize the relativistic parameter (σ) and the ratio (R/R_*), where R_* is the radius of the typical neutron star which is equal to $R_* = 1.4 \times 10^6$ cm. This ratio takes values from 0 to 1 in steps of 0.02. The output layer has also two outputs which represent the Emden function (density profile) and the mass-radius relation of the relativistic isothermal gas spheres. We checked three configurations for the number of hidden layer neurons by testing 20, 50, and 100 neurons in that layer to find which of them was the best. The data values for the relativistic function (σ) that have been used for the training and testing of the adopted neural network are shown in Table 1.

After checking the three configurations of the hidden layer neurons, we concluded that the best number of those neurons was 50. This number of the hidden layer neurons gave the least RMS error of 0.000156 in almost

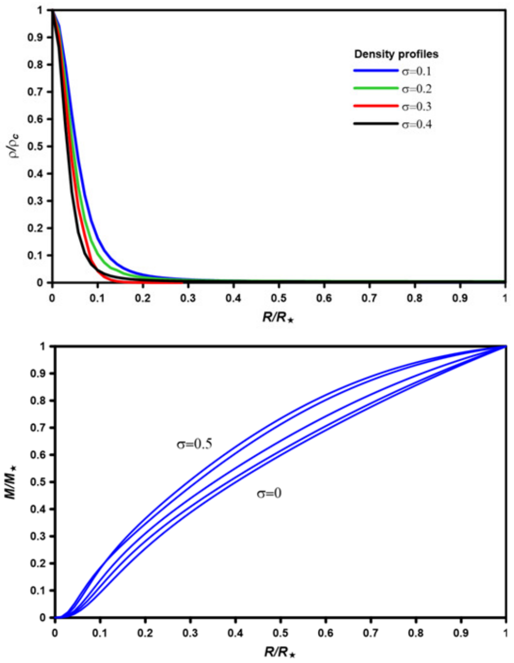

Fig. 3. The density profiles (upper panel) and the mass-radius relations (lower panel) of the relativistic isothermal gas spheres. The color figure can be viewed online.

TABLE 1

TRAINING AND TESTING DATA FOR THE NN ADOPTED TO SIMULATE THE EMDEN AND MASS FUNCTIONS OF THE RELATIVISTIC ISOTHERMAL GAS SPHERES

Training Data	Testing Data
σ	σ
0.01 0.02 0.03 0.05 0.06	0.04 0.08 0.12 0.16 0.20
0.07 0.09 0.010 0.11 0.13	0.24 0.28 0.32 0.36 0.40
0.14 0.15 0.17 0.18 0.19	
0.21 0.22 0.23 0.25 0.26	
0.27 0.29 0.30 0.31 0.34	
0.35 0.37 0.38 0.39 0.41	
0.42 0.43 0.45 0.46 0.49	
0.50	

the same number of training iterations. So, the configuration of the NN adopted to simulate the Emden and mass functions of the relativistic isothermal gas spheres was 2-50-2.

While in the training phase of the NN, we used a value of the momentum ($\gamma = 0.5$) and a value for the learning rate ($\eta = 0.03$).Those values of γ and η were found to speed up the convergence of the back-propagation training algorithm without exceeding the solution. The convergence and stability behaviors of the input layer weights, bias, and output layer weights (w_i, β_i, $and\,\nu_i$) were studied during the training phase which is shown in Figure 4. The weight values were initialised to random values, which can be seen in such figures, and after a number of iterations, they converged to stable values.

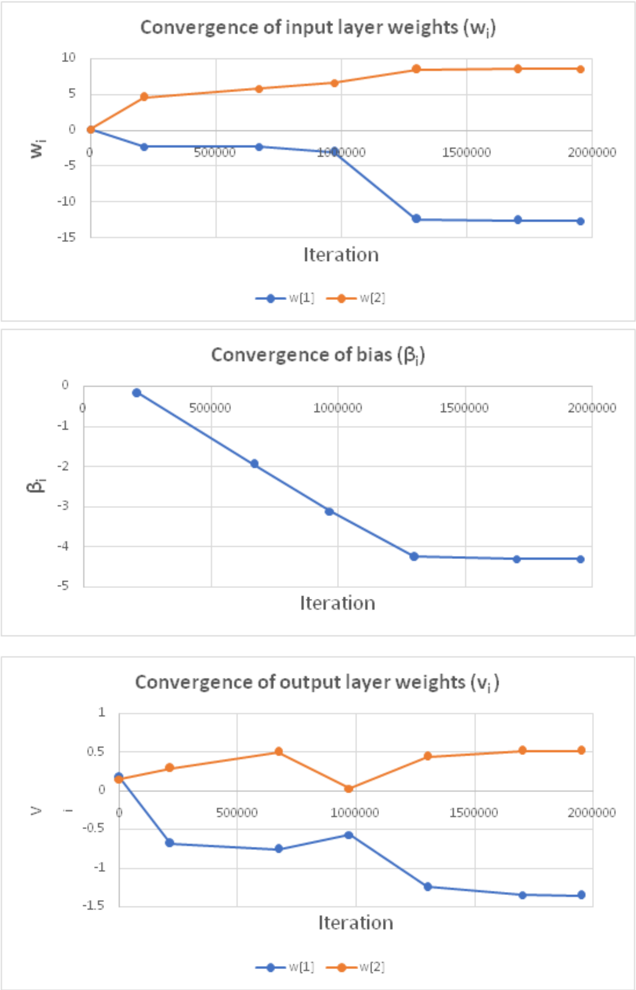

Fig. 4. The density profiles (upper panel) and the mass-radius relations (lower panel) of the relativistic isothermal gas spheres. The color figure can be viewed online.

5.3. *Comparison with the Analytical Model*

After the training phase is completed, we can use the code with its frozen weight values to compute the Emden and mass functions of the RIGS. To compare the analytical models and the ANN ones, we calculated the isothermal gas models for the relativistic parameters shown in Column 2 of Table 1. The results of these comparisons are displayed in Figure 5, which shows the identical overlap between the density profiles and their relevant ANN models. The figures also show the limited discrepancies between the ANN model and their relevant mass-radius relations. In general, we obtained good agreement with a maximum absolute error for the density profiles and mass-radius relations of 1% and 5%, respectively.

According to Nouh et al. (2020), the large discrepancy in the curve of the mass-radius relations can be attributed to numerical instability caused by the mass function's (ν) acceleration process.

6. CONCLUSIONS

In the present paper, the Taylor series and an artificial neural network (ANN) approach were used to solve the relativistic isothermal gas sphere. We wrote a Mathematica code to derive the symbolic expressions for the Emden and the relativistic functions. Then, the radii, densities, and masses of the gas spheres were calculated and the convergence of the analytical models was accelerated using Euler-Abel and Pade transformations. A

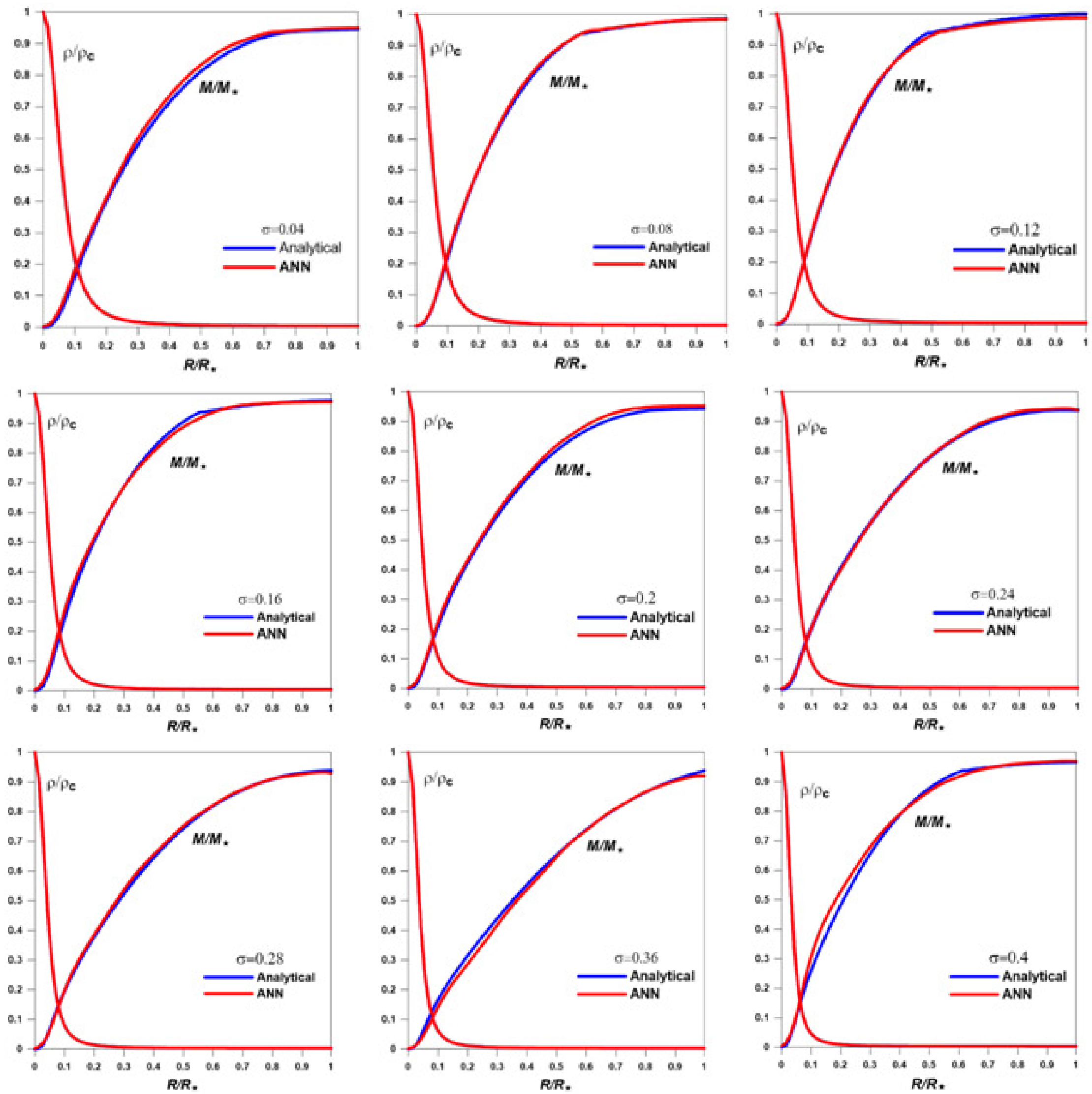

Fig. 5. The mass-radius relation and density profile were calculated for the relativistic parameter range σ=0.04, 0,08, 0,12, 0.16, 0.2, 0.24, 0.28, 0.36, and 0.4. The red lines represent the accelerated Taylor series solutions, while the blue lines represent the ANN solutions. The color figure can be viewed online.

total sum of 50 analytical gas models was calculated for the range of the relativistic parameter σ =0 - 0.5 with step 0.01 and compared to the numerical ones which indicated that the maximum relative errors are 1% and 3% for the density profiles and mass-radius relations, respectively.

For the ANN calculations, we developed a C++ code that implemented the feed-forward back-propagation learning scheme. Training and testing of the ANN algorithm were performed for the density profiles and mass-radius relation of the neutron stars. The efficiency and accuracy of the presented algorithm were evaluated by running it through 10 relativistic isothermal gas models and comparing them to analytical models. The ANN calculations yielded results that were in very good agreement with the analytical results, demonstrating that using the ANN method is effective, accurate, and may outmatch other methods.

Acknowledgment: We are so grateful to the reviewer for his/her many valuable suggestions and comments that significantly improved the paper. This paper is based upon work supported by Science, Technology & Innovation Funding Authority (STDF) Egypt, under Grant Number 37122.

REFERENCES

Abdel-Salam, E. A.-B, Nouh, M. I., Azzam, Y. A., & Jazmati, M. S. 2021, in AdAst2021, 12, https://doi.org/10.1155/2021/6662217

Antonov, V. A. 1985, in Dynamics of star clusters, IAUS 113 (Princeton, NJ: Dordrecht-Reidl), 7, 135

Azzam, Y. A., Abdel-Salam, E. A.-B, & Nouh, M. I. 2021, RMxAA, 57, 189, https://doi.org/10.22201/ia.01851101p.2021.57.01.14

Basheer, I. A. & Hajmeer, M. 2000, Journal Of Microbiological Methods, 43, 3, https://doi.org/10.1016/s0167-7012(00)00201-3

Binney, J. & Tremaine, S. 1987, Galatic Dynamics (Princeton, NJ: PUP)

Chandrasekhar, S. 1939, An Introduction to the study of stellar structure, (Chicago, Ill: UCP)

Chavanis, P. H. 2002, A&A, 381, 709, https://doi.org/10.1051/0004-6361:20011424

El-Mallawany, R., Gaafar, M. S., & Azzam, Y. A. 2014, Chalcogenide Letters, 11, 227

Elminir, H. K., Azzam, Y. A. & Younes, F. I. 2007, Energy, 32, 1513, https://doi.org/10.1016/j.energy.2006.10.010

Hunter, C. 2001, MNRAS, 328, 839, https://doi.org/10.1046/j.1365-8711.2001.04914.x

Leshno, M., Lin, V., Pinkus, A., & Schocken, S. 1993, Neural Networks, 6, 861, https://doi.org/10.1016/s0893-6080(05)80131-5

Liu, F. K. 1996, MNRAS, 281, 1197, https://doi.org/10.1093/mnras/281.4.1197

Milgrom, M. 1984, ApJ, 287, 571, https://doi.org/10.1086/162716

Morawski, F. & Bejger, M. 2020, A&A, 642, 78, https://doi.org/10.1051/0004-6361/202038130

Natarajan, P. & Lynden-Bell, D. 1997, MNRAS, 286, 268, https://doi.org/10.1093/mnras/286.2.268

Nouh, M. I. 2004, NewA, 9, 467, https://doi.org/10.1016/j.newast.2004.02.003

Nouh, M., Abdel-Salam, E., & Azzam, Y. 2022, Artificial Neural Network Approach versus Analytical Solutions for Relativistic Polytropes, Research Square Platform, LCC, https://doi.org/10.21203/rs.3.rs-1340444/v1

Nouh, M. I., Azzam, Y. A., & Abdel-Salam, E. A.-B. 2020, Modeling Fractional Polytropic Gas Spheres Using Artificial Neural Network, arXiv:2006.15445, https://doi.org/10.48550/arXiv.2006.15445

Raga, A. C., Rodríguez-Ramírez, J. C., Villasante, M., Rodríguez-González, A., & Lora, V. 2013, RMxAA, 49, 63

Roxburgh, I. W. & Stockman, L. M. 1999, MNRAS, 303, 466, https://doi.org/10.1046/j.1365-8711.1999.02219.x

Saad, A., Nouh, M. I., Shaker, A. A., & Kamel, T. M. 2017, RMxAA, 53, 247

__________. 2021, RMxAA, 57, 407, https://doi.org/10.22201/ia.01851101p.2021.57.02.13

Sharma, J. P. 1990, Ap&SS, 163, 109

Yadav, N., Yadav, A., & Kumar, M. 2015, An Introduction to Neural Network Methods for Differential Equations, (Springer Dordrecht), https://doi.org/10.1007/978-94-017-9816-7

Yosry A. Azzam, Farag I. Elnagahy, Tarek M. Kamel, and Mohamed I. Nouh: Astronomy Department, National Research Institute of Astronomy and Geophysics (NRIAG), 11421 Helwan, Cairo, Egypt.

Emad A-B Abdel-Salam: Department of Mathematics, Faculty of Science, New Valley University, El-Kharja 72511, Egypt.