Science & Technology Development Journal – Engineering and Technology 2023, 6(3):1946-1954

Open Access Full Text Article

Research article

Accelerating the nonlinear analysis of hyper-elastic behavior by GMDH-assisted Newton-Raphson scheme

Minh. N. Nguyen[1,2,*]

[1]*Duy Tan Research Institute for Computational Engineering (DTRICE) – Duy Tan University, 6 Tran Nhat Duat, District 1, Ho Chi Minh city, Vietnam*

[2]*Faculty of Civil Engineering, Duy Tan University, Da Nang, Vietnam*

Correspondence

Minh. N. Nguyen, Duy Tan Research Institute for Computational Engineering (DTRICE) – Duy Tan University, 6 Tran Nhat Duat, District 1, Ho Chi Minh city, Vietnam

Faculty of Civil Engineering, Duy Tan University, Da Nang, Vietnam

Email:
nguyennngocminh6@duytan.edu.vn

History

- Received: 09-8-2022
- Accepted: 29-9-2023
- Published Online: 30-9-2023

DOI :

https://doi.org/10.32508/stdjet.v6i3.1034

ABSTRACT

This paper presents an accelerated iterative scheme for nonlinear problems. Commonly, analysis of nonlinear behavior is conducted by the Newton-Raphson (NR) method. It is well-known that the number of iterations required depends on the deviation between the ``starting point'' and the converged solution. In practice, the solution of previous load step is taken as the ``starting point'', while the converged solution of the current load step is not known beforehand. Therefore, difficulties or even non-convergence may occur. Recently, it is suggested that a neural network is employed to predict the solution of the current load step. This prediction is then used as the ``starting point'' for NR scheme. It is expected, that the true converged solution (of current step) is closer to the prediction by neural network than to the solution of previous load step. As a result, the scheme becomes faster due to less iterations. Obviously, any techniques for time-series forecasting can be used. Here, the Group Method of Data Handling (GMDH) is proposed. Loosely speaking, GMDH is a feedforward neural network without backpropagation. Practically, the GMDH-assisted NR scheme should not take longer time than conventional NR scheme. The advantage of GMDH is fast computation; however, the accuracy may be not as high as a network that has backpropagation. Therefore, careful consideration on the construction of GMDH network is needed. In the current work, the performance of GMDH-assisted NR scheme is investigated in analysis of hyper-elastic behavior, which involves both geometrical and material nonlinearity. A study on the influence of activation function on the accuracy is presented. Also, it is found that prediction for incremental displacement (between the current load step and the previous load step) could be better than prediction of displacement of the current load step.

Key words: Time-series forecasting, GMDH network, Newton-Raphson scheme, Accelerated nonlinear analysis

INTRODUCTION

Nonlinear behaviors are usually encountered in engineering problems. Therefore, an efficient algorithm for nonlinear analysis is always needed. In practice, the iterative Newton-Raphson (NR) scheme is the most commonly used, due to simple implementation and quadratic convergence rate. The method has been widely applied in analyses of unsaturated flow[1], plastic deformation[2,3], geometrical nonlinearity[4–6], hyper-elastic behavior[7–9], temperature-dependent heat transfer[10–12] and many other types of problem.

It is common knowledge if the deviation between the converged solution and the "initial guess" or "starting point" is large, difficulties may occur[13–15], being reflected in the large number of iterations. Non-convergence may even be encountered. Practically, the *converged solution of previous load step* is used as the "starting point" for the current load step. Usually, small step size is applied, with the hope to increase the possibility of convergence. However, the process would be very time-consuming. In 2001, Kim and Kim introduced the employment of neural network to predict the starting point[13]. A pre-analysis is required to compute parameters that are characteristic to the pattern of the three previous converged solutions. The neural network is then trained to learn the pattern and estimate a starting point for iterations of the current load step. Recently, Nguyen T. N. et al.[14] directly exploits the time-series forecasting of GMDH-type neural network to predict the converged solution of the current load step. The predicted values are then used as the starting point for NR scheme. Ref.[15] further discussed that a careful selection of parameters for GMDH-network could reduce the number of iterations to one iteration. Nevertheless, Refs.[14,15] were limited to geometrically nonlinear analysis of shell structures. Furthermore, from a practical point of view, a neural-network-assisted NR scheme should take less time than the conventional NR scheme. Probably this is the reason that there were not much publications on this area, since it is difficult

Cite this article : Nguyen M N. **Accelerating the nonlinear analysis of hyper-elastic behavior by GMDH-assisted Newton-Raphson scheme**. *Sci. Tech. Dev. J. – Engineering and Technology* 2023; 6(3):1946-1954.

to select a network that satisfies both accuracy and speed.

For analysis of hyper-elastic behaviors of neo-Hookean type materials, which include both geometrical and material nonlinearities, there were some attempts to accelerate the computational process, for e.g. by POD-DEIM[16], or by the reduced-basis technique namely Combined Approximations (CA)[8]. However, the purpose of Refs.[8,16] is saving time in each iteration, rather than reduction of number of iterations as in Refs.[14,15]. Being inspired from the Refs.[14,15], in this paper the GMDH-assisted NR scheme is further investigated for analysis of hyper-elastic solids.

This report is organized as follows. Right after the Introduction, a brief review on hyper-elastic behavior is presented in Section 2. The time-series forecasting GMDH network is described in Section 3. Section 4 is reserved for numerical results and discussion. Finally, concluding remarks are drawn in the last section.

HYPER-ELASTIC BEHAVIOR

Let us consider a two-dimensional (2D) solid domain Ω, bounded by Γ, see Figure 1. It is subject to a body force **b**, while traction **t** is applied on the part Γ_t of the boundary, and displacement constraints takes place on Γ_u.

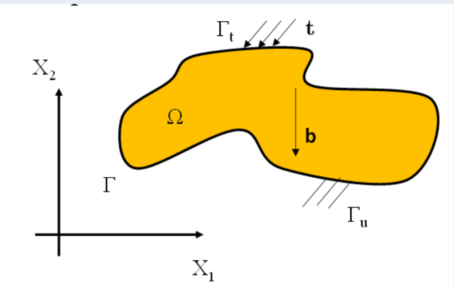

Figure 1: Sketch of a 2D solid domain

By denoting **x** the current position of a point in Ω, and **X** the position of that point in the initial configuration of the body, the displacement vector is **u** = **x** – **X**. Using finite element analysis, the governing equation for equilibrium problem can be written using the initial configuration as follows

$$G(u) = F_{int} - F_{ext} = 0 \tag{1}$$

where the internal force, $\mathbf{F}_{int}$, and the external force, $\mathbf{F}_{ext}$, are given by

$$F_{int} = \int_{\Omega}^{\square} B_1^T S d\Omega \tag{2}$$

$$F_{ext} = \int_{\Omega}^{\square} N^T b d\Omega + \int_{\Gamma_t}^{\square} N^T t d\Gamma \tag{3}$$

Here, **S** is the second Piola-Kirchhoff stress tensor, **N** is the vector of shape functions and $\mathbf{B}_1$ is the derivative operator. Applying the total Lagrangian approach, linearization of Equation (1) for Newton-Raphson iterative scheme takes the following form[8,9]

$$K_{\tan \delta u = -G,} \tag{4}$$

where $\delta\mathbf{u}$ is the change of displacement between two consecutive iterations and $\mathbf{K}_{tan}$ is the tangent stiffness

$$K_{\tan = K_1 + K_2,} \tag{5}$$

$$K_1 = \int_{\Omega}^{\square} B_1^T D B_1 d\Omega, \tag{6}$$

$$K_2 = \int_{\Omega}^{\square} B_2^T \widetilde{S} B_2 d\Omega, \tag{7}$$

In Equation (6), **D** is the fourth-order constitutive tensor, which relates strain and stress components. The matrix $\mathbf{B}_1$ and $\mathbf{B}_2$ are calculated by

$$B_1 = \begin{bmatrix} B_1^1 & B_1^2 & \ldots & B_1^n \end{bmatrix}, \tag{8}$$

$$B_2 = \begin{bmatrix} B_2^1 & B_2^2 & \ldots & B_2^n \end{bmatrix}, \tag{9}$$

where *n* is the number of node within an element. For each local node *i* (*i = 1, 2, …, n*), we have

$$B_1^i = \begin{bmatrix} F_{11}N_{i,1} & F_{21}N_{i,1} \\ F_{12}N_{i,2} & F_{22}N_{i,2} \\ F_{11}N_{i,2} + F_{12}N_{i,1} & F_{21}N_{i,2} + F_{22}N_{i,1} \end{bmatrix}, \tag{10}$$

$$B_2^i = \begin{bmatrix} N_{i,1} & 0 \\ N_{i,2} & 0 \\ 0 & N_{i,1} \\ 0 & N_{i,2} \end{bmatrix}. \tag{11}$$

In Equation (10), F_{IJ} are components of the deformation gradient tensor **F**, which are given by

$$F = \frac{\partial x}{\partial X} = 1 + \frac{\partial u}{\partial X}, \tag{12}$$

where **1** is second-order identity tensor. The "comma" sign (,) in Equation (10) and (11) denotes spatial derivative; for e.g., $N_{i,1}$ is the derivative of N_i with respect to X_1.

The matrix $\widetilde{S}$ in Equation (7) stores the components of the second Piola-Kirchhoff stress

$$\widetilde{S} = \begin{bmatrix} S_{11} & S_{12} & 0 & 0 \\ S_{12} & S_{22} & 0 & 0 \\ 0 & 0 & S_{11} & S_{12} \\ 0 & 0 & S_{12} & S_{22} \end{bmatrix}. \tag{13}$$

The second Piola-Kirchhoff stress tensor **S** (see Equation (2)) is symmetric and can be calculated by

$$S = 2\frac{\partial W}{\partial C} = 2\Sigma_{k=1}^{3}\frac{\partial W}{\partial I_k}\frac{\partial I_k}{\partial C} \tag{14}$$

where **C** = $\mathbf{F}^T$ **F** is the right Cauchy-strain tensor. I_1, I_2, I_3 are the three invariants of tensor **C**:

$$I_1 = trace\left(C\right), \tag{15}$$

$$I_2 = \frac{1}{2}\left[\left(trace\left(C\right)\right)^2 - trace\left(C^2\right)\right], \tag{16}$$

$$I_3 = det\left(C\right). \tag{17}$$

The strain energy function W is characteristic to each type of hyper-elastic material. Under the assumption of neo-Hookean materials, W is given by

$$W = \frac{\kappa}{2}\left(J-1\right)^2 + \frac{\mu}{2}\left(I_1 - 2\log J - 3\right), \tag{18}$$

where κ and μ are the bulk and shear moduli, respectively, and J = det(**F**). From Equation (14), the second Piola-Kirchhoff stress tensor S is determined by

$$S = \kappa C^{-1}\log J + \mu\left(1 - C^{-1}\right) \tag{19}$$

The constitutive tensor **D** is obtained by

$$D = 2\frac{\partial S}{\partial C} = 4\frac{\partial^2 W}{\partial C \partial C} \tag{20}$$

METHODOLOGY: GMDH-TYPE NEURAL NETWORK

The group method of data handling (GMDH)[17] is a self-organizing deep learning technique. GMDH is quite similar to a deep neural network, however the number of hidden layers and the number of neurons in each hidden layer is determined by the network itself during the training stage. Another requirement is that output layer of GMDH has only one neuron.

Construction of the first hidden layer

Each neuron of the first hidden layer is equipped by one activation function and one transfer function. The activation function is usually a polynomial that takes k inputs. For example, the activation function in form of a bi-variate (*two inputs*) quadratic polynomials is written as follows

$$\begin{aligned} z\left(x_i, x_j\right) &= a_0 + a_1 x_i + a_2 x_j \\ &+ a_3 x_i^2 + a_4 x_j^2 + a_5 x_i x_j, \end{aligned} \tag{21}$$

where x_i, x_j are the two arbitrary values taken from the input layer. The neuron then produces one output value by the transfer function, which can be chosen as identity function or sigmoid function

Identity function:

$$g\left(z\right) = z, \tag{22}$$

Sigmoid function:

$$g\left(z\right) = \frac{1}{1 + exp\left(-z\right)} \tag{23}$$

Because each neuron takes two inputs, the number of neurons in the first hidden layer can be determined by

$$m = C_2^n = \frac{n^2 - n}{2}, \tag{24}$$

where n is the number of inputs.

The neuron is applied to all samples of the training dataset. Assuming that there are s samples in the *training dataset*, the following equations are obtained for each neuron

$$\begin{aligned} & z_{(1)} = a_0 + a_1 x_{i,(1)} + a_2 x_{j,(1)} + \\ & a_3 x_{i,(1)}^2 + a_4 x_{j,(1)}^2 + a_5 x_{i,(1)} x_{j,(1)} \\ & z_{(2)} = a_0 + a_1 x_{i,(2)} + a_2 x_{j,(2)} + \\ & a_3 x_{i,(2)}^2 + a_4 x_{j,(2)}^2 + a_5 x_{i,(2)} x_{j,(2)} \\ & z_{(s)} = a_0 + a_1 x_{i,(s)} + a_2 x_{j,(s)} + \\ & a_3 x_{i,(s)}^2 + a_4 x_{j,(s)}^2 + a_5 x_{i,(s)} x_{j,(s)} \end{aligned}$$

Or in the matrix form

$$Z = XA, \; and \; X^T Z = X^T XA \tag{25}$$

where **X** (size s-by-6) contains the input values and **A** (size 6-by-1) stores the coefficients. Requiring that produced values from transfer function are equal to the true output of the dataset, we have

$$Y = g\left(Z\right), \; thus \; Z = g^{-1}\left(Y\right). \tag{26}$$

The vector of coefficients, **A**, is then calculated by

$$A = \left(X^T X\right)^{-1} X^T Z \tag{27}$$

Once all the neurons of the layer have been constructed, the mean squared error (MSE) between the values produced by each neuron with the targeted output (from the *validation dataset*) can be evaluated. The neurons are then sorted by MSE (in ascending order). A threshold can be applied here to remove neurons that have MSE higher than threshold.

Construction of other hidden layers

The construction of other hidden layers is similar to that of the first layer, except that the inputs for the r^{th} hidden layer are taken from the produced values from the previous layer, i.e. the $(r\text{-}1)^{th}$ hidden layer.

The construction stops if: (i) the maximum number of hidden layers (specified by the user) is reached, or (ii) the MSE of the best neuron of the current hidden layer is not lower than the MSE of the best neuron of the previous layer. Case (ii) indicates that the result cannot be improved. Therefore, the current layer will be removed and the construction process is terminated.

The final output of the GMDH network is the output value of the best neuron of the very last layer.

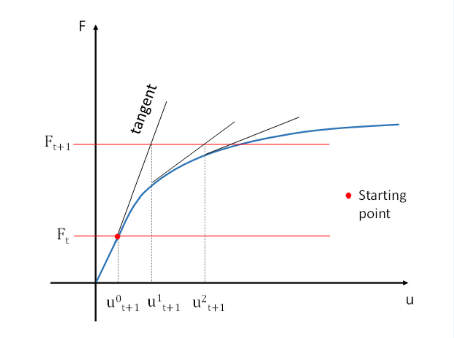

Figure 2: Illustration of Newton-Raphson scheme

GMDH-assisted Newton-Raphson scheme

Figure 2 presents an illustration for Newton-Rapshson (NR) algorithm. Typically, in the beginning of the current load step, namely load step (t+1), the converged solution of the previous step, i.e. load step (t), is taken as the “starting point” for iterations (the solid dot in Figure 2). Here, following Refs. [14,15], the GMDH network for time-series forecasting is employed to predict the converged solution of the current step. That predicted value is then used as the “starting point” for NR scheme. It is expected that the predicted value will be closer to the true converged solution and thus, reduction in the number of iterations can be achieved.

Assuming that M solutions from load step (t-M) to load step (t) have been known. Given the number of delays (i.e. the number of inputs in each sample), the data can be arranged into input sets and target sets. Figure 3 is an illustration of data preparation for $M = 8$ and delays = 2, where there are 7 samples of data (each sample has 2 inputs and 1 targeted output). These data are sent to GMDH to train and predict the solution of load step (t+1). A portion of samples are used for training (training dataset) and the rest are for validation (validation dataset).

It is noted that similar to any other deep learning technique, normalization of data could be necessary. Furthermore, a GMDH network is needed for each unknown degrees of freedom.

In this paper, the performance of GMDH-assisted NR scheme for analysis of hyper-elastic behavior is investigated. Unlike in previous works [14,15], where only geometrical nonlinearity is considered, both geometrical and material nonlinearities are involved in behavior of hyper-elastic solids. Therefore, the problem is more complicated.

It is noted that the components of tensor **D** (see Eq. (6) and Eq. (20)) at an arbitrary point are dependent on the strain values at that point, indicating material nonlinearity. In Refs. [14,15], where material nonlinearity is not considered, tensor **D** would be constant. Here, there are two sources of nonlinearities, meaning that complexity is increased. As a result, it is more difficult for the neural network to learn and provide prediction with high accuracy.

RESULTS AND DISCUSSION

The proposed model is applied to study behavior of a curved beam, as sketched in Figure 4. The beam is subject to an inclined uniform load (45^o) at one end, while the other end is fixed. The material is assumed to be neo-Hookean type with bulk modulus κ = 120.291 MPa and shear modulus μ = 80.194 MPa. The uniform load q = 0.5 N/mm^2. The problem domain is uniformly discretized by 9 x 100 8-node quadrilateral elements (9 elements along the radial direction and 100 elements along the circumferential direction). There are 2919 nodes in total.

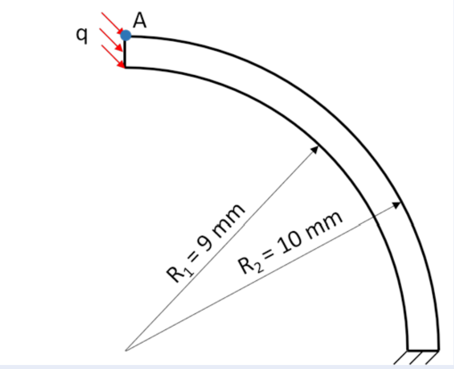

Figure 4: Sketch of the curved beam problem

The load is gradually increased by 40 steps, i.e. incremental size Δq = 0.0125 N/mm^2. The conventional

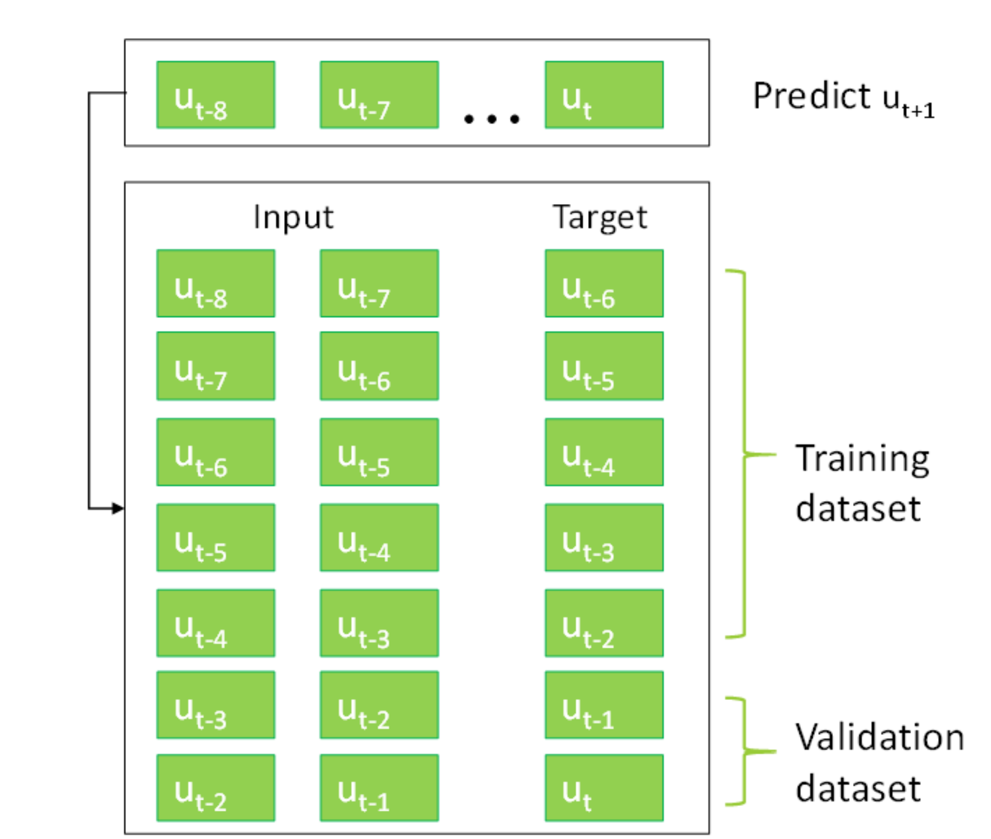
u_{t-8}
u_{t-7}
...
u_t
Predict u_{t+1}
Input
Target
u_{t-8} u_{t-7} u_{t-6}
u_{t-7} u_{t-6} u_{t-5}
u_{t-6} u_{t-5} u_{t-4}
u_{t-5} u_{t-4} u_{t-3}
u_{t-4} u_{t-3} u_{t-2}
u_{t-3} u_{t-2} u_{t-1}
u_{t-2} u_{t-1} u_t
Training dataset
Validation dataset

Figure 3: Illustration of data preparation for GMDH network

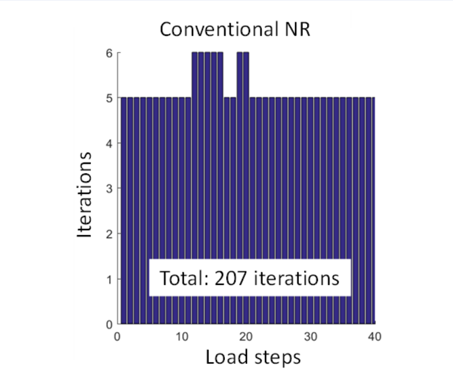

Figure 5: Number of iterations required by Conventional NR for the curved beam problem

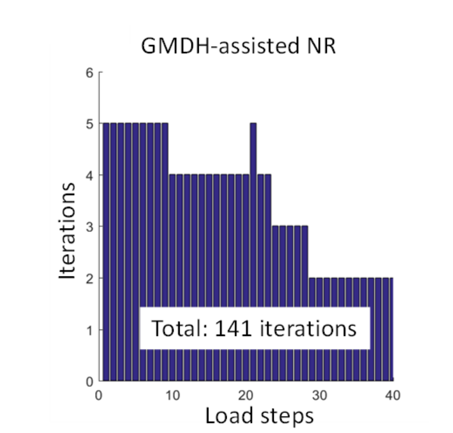

Figure 6: Number of iterations required by GMDH-assisted NR for the curved beam problem

NR scheme is conducted to solve for the first $M = 9$ load steps. The rest of load steps are aided by GMDH to estimate the starting point. By default, identity function is used as transfer function and number of delays is 3.

Comparison between conventional NR and GMDH-assisted NR

Figure 5 and Figure 6 respectively present the number of iterations in each load step, for conventional NR and GMDH-assisted NR using tri-variate, quadratic polynomials (in short "3-quadratic") as activation function. It is observed inFigure 6 that from load step 10 to load step 40, with the aid of GMDH, the number of iterations in each load step is generally reduced, resulting a total of 141 iterations, which is much less than 207 iterations needed by the conventional NR.

The reduction in number of iteration indeed boosts the computational speed, as elapsed time in GMDH-assisted NR (The time for running GMDH is already included) is much less than that in conventional NR, see Table 1. Roughly 30% of total time can be saved. In Ref.[15], even one iteration for each load step can be achieved, when only large deformation (geometrical nonlinearity) is considered. The hyper-elastic behavior is more complicated in nature, since the nonlinearities come from both large deformation and the stress-strain relation.

Figure 7 depicts the curves of vertical reaction force and vertical displacement at point A (see Figure 4). The results by GMDH-assisted NR and Conventional NR is in good agreement. This is as expected, because the overall procedure of NR is not altered. The role of GMDH is simply providing a better starting point for iterations.

Effects of activation function

Next, the effect of activation function is investigated. Four types of polynomials are taken into account: bi-variate quadratic (2-quadratic), tri-variate quadratic (3-quadratic), bi-variate cubic (2-cubic) and tri-variate cubic (3-cubic). Among the four variations, 3-quadratic is computationally the most efficient, as reported in Table 2. On the other hand, non-convergence occurs in 2-cubic.

Prediction of incremental displacement

Here, we predict the incremental displacement between load step (t) and load step (t+1), $\Delta\ \mathbf{u} = \mathbf{u}_{t+1}$ - $\mathbf{u}_t$, instead of directly predict the displacement at load step (t+1), $\mathbf{u}_{t+1}$. The same four variations as in subsection 4.2 are considered.

The performance of GMDH in Table 3 is in general better than that in Table 2, in terms of number of iterations. Convergence is achieved by all four variations. In both Table 2 and Table 3, the tri-variate polynomials are better choice for activation function than the two-variate polynomials.

CONCLUSIONS

The GMDH-assisted NR has been successfully further extended for analysis of hyper-elastic behavior of two-dimensional solids. The prediction by GMDH provides a better starting point for NR iterative scheme, such that the number of iterations can be reduced. As a result, a large amount of computational time is saved.

The efficiency of the proposed scheme comes from the quick process of GMDH. This is important, because the training is online, i.e. it is conducted when the problem is solved. There is no pre-training. Furthermore, several GMDH networks are needed every load steps. The number of GMDH networks is equal to that of unknown degrees of freedom. Therefore, the necessary to have fast computation in each individual network is more pronounced.

It is aware that the accuracy of prediction by GMDH is crucial. The choice of transfer function and activation function would have influence on the accuracy. A comparative study on activation function, while identity function is selected as transfer function, has shown that tri-variate polynomials would result in better performance than two-variate polynomials. Currently, only polynomials are considered for activation function. The possibility of other types of activation function should also be studied in future works. Even the GMDH could be replaced by any other time-series forecasting network. From practical point of view, an accelerated NR scheme is only beneficial if it is faster than conventional NR scheme. Therefore, any attempts to improve accuracy of prediction should always pay attention to the elapsed time. Furthermore, it is found that prediction for incremental displacement could be possibly more robust than direct prediction for the converged value of displacement.

Last but not least, by a good estimation of starting point, GMDH could help to reduce the number of iterations, but it does not have any role in the computational time of each iteration. On the other hand, the reduced basis approach[8] is efficient to accelerate each iteration but cannot reduce the number of iterations. Therefore, combination of two techniques is promising for future works.

LIST OF ABBREVIATIONS

GMDH: Group method of data handling
NR: Newton-Raphson
MSE: mean squared error

CONFLICT OF INTEREST

There is no conflict of interest.

Table 1: Comparison of elapsed time between Conventional NR and GMDH-assisted NR

	Iterations	Time
Conventional NR	207	~ 701 s
GMDH-assisted NR	141	~ 485 s

Table 2: Comparison of elapsed time variations of GMDH-assisted NR

	Iterations	Time
2-quadratic	174	~ 595 s
2-cubic	N/A	N/A
3-quadratic	141	~ 485 s
3-cubic	179	~ 614 s

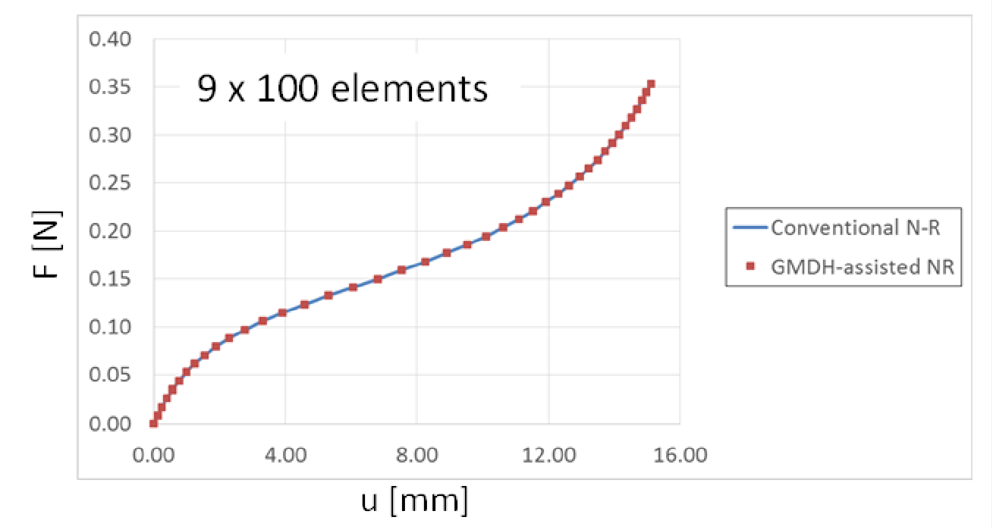

Figure 7: The curve of vertical reaction force and vertical displacement at point A

Table 3: Comparison of elapsed time variations of GMDH-assisted NR, in which the incremental displacements are predicted

	Iterations	Time
2-quadratic	154	~ 556 s
2-cubic	149	~ 544 s
3-quadratic	143	~ 525 s
3-cubic	140	~ 516 s

CONTRIBUTION OF EACH AUTHOR

Minh N. Nguyen contributes every aspect of the manuscript: ideas brainstorming, data checking, writing, proofreading and editing of the manuscript.

REFERENCES

1. Nguyen MN, Bui TQ, Yu T, Hirose S. Isogeometric analysis for unsaturated flow problems. Comput Geotech. 2014;62:257-67;Available from: https://doi.org/10.1016/j.compgeo.2014.08.003.
2. Coombs WM, Petit OA, Ghaffari Motlagh YG. NURBS plasticity: yield surface representation and implicit stress integration for isotropic inelasticity. Comput Methods Appl Mech Eng. 2016;304:342-58;Available from: https://doi.org/10.1016/j.cma.2016.02.025.
3. Coombs WM, Ghaffari Motlagh YG. NURBS plasticity: yield surface evolution and implicit stress integration for isotropic hardening. Comput Methods Appl Mech Eng. 2017;324:204-20;Available from: https://doi.org/10.1016/j.cma.2017.05.017.
4. Espath LFR, Braun AL, Awruch AM, Maghous S. NURBS-based three-dimensional analysis of geometrically nolinear elastic structures. Eur J Mech A Solids. 2014;47:373-90;Available from: https://doi.org/10.1016/j.euromechsol.2014.05.005.
5. Nguyen DD, Nguyen MN, Duc ND, Rungamornrat J, Bui TQ. Enhanced nodal gradient finite elements with new numerical integration schemes for 2D and 3D geometrically nonlinear analysis. Appl Math Modell. 2021;93:326-59;Available from: https://doi.org/10.1016/j.apm.2020.11.040.
6. Van Do VNV, Lee C-H. Nonlinear analyses of FGM plates in bending by using a modified radial point interpolation meshfree method. Appl Math Modell. 2018;57:1-20;Available from: https://doi.org/10.1016/j.apm.2017.12.035.
7. Huyssteen D, Reddy BD. A virtual element method for isotropic hyperelasticity. Comput Methods Appl Mech Eng. 2020;367:113134;Available from: https://doi.org/10.1016/j.cma.2020.113134.
8. Nguyen MN, Nguyen NT, Truong TT, Bui TQ. An efficient reduced basis approach using enhanced meshfree and combined approximation for large deformation. Eng Anal Boundary Elem. 2021;133:319-29;Available from: https://doi.org/10.1016/j.enganabound.2021.09.007.
9. Nguyen NT, Nguyen MN, Vu TV, Truong TT, Bui TQ. A meshfree model enhanced by NURBS-based Cartesian transformation method for cracks at finite deformation in hyperelastic solids. Eng Fract Mech. 2022;261:108176;Available from: https://doi.org/10.1016/j.engfracmech.2021.108176.
10. Feng SZ, Cui XY, Li AM, Xie GZ. A face-based smoothed point interpolation method (FS-PIM) for analysis of nonlinear heat conduction in multi-material bodies. Int J Therm Sci. 2016;100:430-7;Available from: https://doi.org/10.1016/j.ijthermalsci.2015.10.017.
11. Feng SZ, Cui XY, Li AM. Fast and efficient analysis of transient nonlinear heat conduction problems using combined approximations (CA) method. Int J Heat Mass Transf. 2016;97:638-44;Available from: https://doi.org/10.1016/j.ijheatmasstransfer.2016.02.061.
12. Nguyen MN, Truong TT, Bui TQ. An enhanced nodal gradient finite element for non-linear heat transfer analysis. Viet J Mech. 2019;41(2):127-39;Available from: https://doi.org/10.15625/0866-7136/12977.
13. Kim JH, Kim YH. A predictor-corrector method for structural nonlinear analysis. Comput Methods Appl Mech Eng. 2001;191(8-10):959-74 ;Available from: https://doi.org/10.1016/S0045-7825(01)00296-1.
14. Nguyen TN, Nguyen-Xuan H, Lee J. A novel data-driven nonlinear solver for solid mechanics using time series forecasting. Finite Elem Anal Des. 2020;171:103377 ;Available from: https://doi.org/10.1016/j.finel.2019.103377.
15. Nguyen TN, Lee J, Dinh-Tien L, Minh Dang LM. Deep-learned one-iteration nonlinear solver for solid mechanics. Int J Numer Methods Eng. 2022;123(8):1841-60;Available from: https://doi.org/10.1002/nme.6918.
16. Radermacher A, Reese S. POD-based model reduction with empirical interpolation applied to nonlinear elasticity. Int J Numer Methods Eng. 2016;107(6):477-95;Available from: https://doi.org/10.1002/nme.5177.
17. Ivakhnenko AG. Polynomial theory of complex systems. IEEE Trans Syst Man Cybern. 1971;SMC-1(4):364-78 ;Available from: https://doi.org/10.1109/TSMC.1971.4308320.

Tạp chí Phát triển Khoa học và Công nghệ – Kĩ thuật và Công nghệ 2023, 6(3):1946-1954

Open Access Full Text Article

Tăng tốc phân tích phi tuyến ứng xử siêu đàn hồi bằng giải thuật Newton-Raphson kết hợp GMDH

Nguyễn Ngọc Minh[1,2,*]

TÓM TẮT

Bài báo này trình bày giải thuật lặp được tăng tốc cho bài toán phi tuyến. Thông thường, việc phân tích ứng xử phi tuyến được thực hiện bằng phương pháp Newton-Raphson (NR). Đặc điểm của phương pháp là số lượng bước lặp phụ thuộc vào mức độ sai khác giữa ``dự đoán ban đầu'' và lời giải hội tụ. Do đó, quá trình giải có thể gặp khó khăn, hoặc không hội tụ. Trong thực tế, lời giải hội tụ của bước tải trước sẽ được chọn là ``dự đoán ban đầu'', trong khi lời giải hội tụ của bước tải hiện tại hiển nhiên là không được biết trước. Gần đây, việc sử dụng hệ thần kinh nhân tạo được đề xuất để dự đoán lời giải của bước tải hiện tại. Kết quả dự đoán này sẽ được dùng làm ``dự đoán ban đầu'' cho quá trình lặp NR. Điều kỳ vọng là lời giải hội tụ thực tế (của bước tải hiện tại) sẽ gần với dự đoán của hệ thần kinh nhân tạo, hơn là với lời giải hội tụ của bước tải trước. Hệ quả là quá trình lặp sẽ nhanh hơn, do cần ít bước lặp hơn. Rõ ràng, bất cứ kỹ thuật nào dùng cho dự đoán theo diễn tiến thời gian đều có thể áp dụng. Ở đây, phương pháp Group Method of Data Handling (GMDH) được đề xuất. Có thể nói rằng, GMDH là một hệ thần kinh nhân tạo không có quá trình lan truyền ngược. Trên quan điểm thực tế, việc kết hợp GMDH vào giải thuật NR phải mang lại thời gian tính toán nhanh hơn NR thông thường. Lợi thế của GMDH là tính toán nhanh, nhưng độ chinh xác kém hơn so với hệ có lan truyền ngược. Do đó, cần cân nhắc cẩn thận trong thiết lập hệ GMDH. Trong nghiên cứu hiện tại, hiệu quả của thuật toán NR kết hợp GMDH sẽ được khảo sát khi phân tích ứng xử siêu đàn hồi, trong đó bao gồm cả yếu tố phi tuyến hình học và yếu tố phi tuyến vật liệu. Một khảo sát về ảnh hưởng của hàm kích hoạt đến độ chính xác của dự đoán bởi GMDH được trình bày. Thêm vào đó, nghiên cứu hiện tại cũng cho thấy, dự đoán về số gia chuyển vị có thể dẫn đến hệ hoạt động tốt hơn so với dự đoán trực tiếp giá trị chuyển vị của bước tải hiện tại.

Từ khoá: Dự đoán diễn tiến thời gian, mạng GMDH, Thuật toán Newton-Raphson, Tăng tốc phân tích phi tuyến

[1] *Viện nghiên cứu tính toán kỹ thuật Duy Tân (DTRICE) - Đại học Duy Tân, 06 Trần Nhật Duật, Phường Tân Định, Quận 1, Thành phố Hồ Chí Minh*

[2] *Khoa Xây dựng, Trường Đại học Duy Tân, Đà Nẵng, Việt Nam*

Liên hệ

Nguyễn Ngọc Minh, Viện nghiên cứu tính toán kỹ thuật Duy Tân (DTRICE) - Đại học Duy Tân, 06 Trần Nhật Duật, Phường Tân Định, Quận 1, Thành phố Hồ Chí Minh

Khoa Xây dựng, Trường Đại học Duy Tân, Đà Nẵng, Việt Nam

Email: nguyenngocminh6@duytan.edu.vn

Lịch sử

- Ngày nhận: 09-8-2022
- Ngày chấp nhận: 04-10-2023
- Ngày đăng: 30-9-2023

DOI : https://doi.org/10.32508/stdjet.v6i3.1034

Trích dẫn bài báo này: Minh N N. **Tăng tốc phân tích phi tuyến ứng xử siêu đàn hồi bằng giải thuật Newton-Raphson kết hợp GMDH**. *Sci. Tech. Dev. J. - Eng. Tech.* 2023, 6(3):1946-1954.