Application of artificial neural networks to solution of variational problems in hydrodynamics

E Kornaeva¹*, A Kornaev² and S Egorov³

¹Department of Information Systems, Orel State University named after I.S. Turgenev, 95 Komsomolskaya Street, Orel 302026, Russian Federation
²Department of Mechatronics, Mechanics and Robotics, Orel State University named after I.S. Turgenev, 95 Komsomolskaya Street, Orel 302026, Russian Federation
³Department of Computer Integrated Systems in Mechanical Engineering, Tambov State Technical University, 106 Sovetskaya Street, Tambov 392000, Russian Federation

*E-mail: lenoks_box@mail.ru

Abstract. The paper deals with a combined approach to approximation of velocity fields and minimization the objective functional in solving viscous fluids flow problems. The mathematical formulation of the problem is presented in the form of a generalized Lagrange functional. The flow function is designed using a feed forward artificial neural network with one hidden layer and with logistic activation function. The boundary values of the flow function are determined using the fluid flow rate. Thus, the problem of determination the velocity field is reduced to the problem of finding the network weights by minimizing the generalized Lagrange functional.

1. Introduction

A number of physics laws are brought to a statement that some value in the process under study has to reach its minimum or maximum. In such a formulation these laws are called variational principles [1, 2]. In hydromechanics the variational approach connected with finding the extremum of the objective functional is an alternative to the classical approach connected with solution the boundary value problems [3, 4]. Variational statements of viscous fluid flow problems are usually used in studying the motion of fluids without taking into account the inertia forces, but including non-Newtonian properties of medium [5] and mass forces [2]. The generalized Lagrange variational principle was proved in [5, 6] and used to solve quasistationary and non-Newtonian fluid flow variational problems.

Ritz method was frequently used direct method in variational calculus [1,2]. In numerical implementation the variational problem is reduced to determination of the minimum of the multiple variables function. Accuracy of the solution is largely determined by successful choice of unknown functions set. Alternatively, direct methods in variational calculus can be implemented using artificial neural networks (ANNs). It is a well known fact that feed forward ANNs with one or more hidden layers perform well in solving fitting problems [7].

In work [8] the author demonstrates applications of ANNs for variational problems in mathematics and engineering, including the geodesic problem, the brachistochrone problem, the catenary problem and others. The boundary value problems are presented as inverse problems with objective functionals in the form of cost functions. In should be noted that the cost functions don’t have any physical meaning. In general, the solving approach is represented in three steps. Step one is choose a suitable parametrized function set. Step two is choose an appropriate objective functional. Step three is design and train an
ANN [8]. A hybrid method for the solution of initial-boundary value problems based on the specific trial solutions and the corresponding adjustable parameters is proposed in [9]. Method is based on Kolmogorov and Cybenko theorems and the functional minimization technique. The applications of ANNs in hydrodynamics are not limited with variational calculus. Deep learning methods and convolutional neural networks (CNNs) have applications to hydrodynamics. The CNNs allow to extrapolate the fluid flow in time using images of the fluid flow process [10, 11].

The limits of variational methods applications can be associated with the complexity of solving non-stationary problems and with the complexity of defining the set of unknown. This work deals with overcoming these limitations due to a new quasistationary formulation of the variational problem of hydrodynamics and the use of artificial neural networks as approximators of unknown functions.

2. Mathematical model

Suppose that the fluid completely fills the domain $\Omega$ with surface $S$, which is characterized by a unit outer normal vector $n$. Supposed that a velocity field $v$ is the vorticity of some vector field $a$: $v = \nabla \times a$ and its components $a_i$ are unknown functions in the variational problem. The velocity field is solenoidal $\nabla \cdot v = 0$. The functions $a$ and $v$ are given at the surface $S$. This is equivalent to the static and kinematic boundary conditions specification. The simplest flow domain between two rigid parallel plates is represented in figure 1.

![Figure 1. Flow domain between two parallel plates (a) and the boundary conditions (b).](image)

It is demonstrated in [5] that the following generalized Lagrange variational principle:

$$ J_L^* (a_m) = \int_{\Omega} \Pi_v \, d\Omega, \quad \tag{1} $$

is connected with viscoplastic potential $\Pi_v$ and takes minimal value on a real vector field $a$ compared to any other field $a'$ with the same boundary conditions.

For the fluids described by the Herschel-Bulkley model [5, 12] the functional (1) in Cartesian coordinates takes the form of equation (2):

$$ J_L^* (a_m) = \int_{\Omega} \left( q_0 \xi_{ij} \xi_{ji} + q_1 \xi_{ij} \xi_{ji} \right) \, d\Omega, \quad \tag{2} $$

where, $q_0, q_1, z$ are the Herschel-Bulkley model parameters [12] and $\xi_{ij}$ are the components of strain rate tensor [5, 13].

Supposed that the boundary conditions are as follow: the flow rate is given at the thresholds of the flow domain and the velocity is given on the plates (figure 1). The flow rate can be calculated as the difference of the flow function values at the plates: $Q = \Psi |_{S_4} - \Psi |_{S_2}$. Supposed that $\Psi |_{S_2} = 0$, the boundary conditions can be represented in by formula (3):
Thus, the formulation of the fluid flow problem is presented in the form of the variational problem (2), (3), taking into account the fact that the components of the strain rate tensor are determined using the stream function fixed at the boundary.

3. Simulation model

Objective functional (2) can be represented in the form of the finite series:

$$J^*_L(\alpha_m) = \sum_{k=1}^{m} \left(q_0 \xi^k_{ij} \xi^k_{ji} + \frac{q_1}{z+1} \left(2 \xi^k_{ij} \xi^k_{ji}\right)^{z+1} \right) \Delta x_k,$$

where, $\xi^k_{ij}$ are values of $\xi_{ij}$ in specific point $k$ on the discrete flow domain and $m$ is the total number of points.

Bellow, for simplicity of notation the variable $x_2$ is replaced by $x$. It is proposed to determine the unknown components of the strain rate tensor $\xi_{ij}$ by means of the flow function $\Psi(x)$. Then the nonzero components of the strain rate tensor are defined as:

$$\xi_{12} = \xi_{21} = \frac{1}{2} \frac{d^2 \Psi}{dx^2}.$$  

The flow function is determined using an artificial neural network with one hidden layer taking into account the boundary conditions (3).

It should be noted that the classical fitting problem in machine learning [14] with an error function (so-called cost function or loss function) as the objective functional is not solved in this work. The objective functional is the generalized Lagrange functional (1). The functional (1) has specific physical meaning and variational problem is connected with minimization of internal forces power. The neural network represents the set of flow functions parameterized with networks weights. Weights are determined as the result of solution of the problem, and hence the form of the unknown flow function.

The ANN architecture is presented in figure 2. In the hidden layer, the logistic function [15] of the following form is used as the activation function:

$$H(x) = (1 + \exp(-\theta_j x^j))^{-1},$$

where, $\theta_j x^j$ is a polynomial of degree $d$, $j = 0, d$. The Einstein’s summation notation and the Lourie’s exception are used.

For further calculations, it is convenient to present this formula in matrix form:

$$H(\chi) = \left(1 + \exp\left(-\chi(\theta^{(1)})^T\right)\right)^{-1},$$

where, $H_{[m \times N_{hid}]}$ is a matrix with the components $H_{pk}$ for the hidden layer neurons, $\chi_{m \times (d+1)}$ is a matrix with the components $x_{pj} = x_j^p$ equal to the values of the polynomial terms at each calculation point, $\theta^{(1)}_{N_{hid} \times (d+1)}$ is a matrix of the weights for the hidden layer neurons with the components $\theta_{kj}^{(1)}$. 

The input layer contains $N_{inp} = d + 1$ neurons with the values correspond to the terms $\theta_j x^j$ of a polynomial of degree $d$ (a hyperparameter). The hidden layer contains $N_{hid}$ neurons. According to the boundary conditions (3) the number of hidden layers is more or equal to 4. The optimal number of neurons can be determined using the exact solution of the problem.

A linear activation function is used in the output layer. Then the stream function takes the form:

$$\Psi(\chi) = H^{(2)}_H^T,$$

where, $\Psi(\chi)$ is a matrix with the components equal to the value of stream function at each calculation point, $H_{(m \times (N_{hid}+1))}$ is a matrix with the components $H_{pk}$ for the hidden layer neurons taking into account the unit value neuron, $\theta^{(2)}_{1 \times (N_{hid}+1)}$ is a matrix of the weights for the output layer with the components $\theta^{(2)}_p$.

![Neural network architecture.](image)

**Figure 2.** Neural network architecture.

Taking into account equations (5) and (7) the objective functional (4) is transformed into a function with unknown weights $\theta^{(1)}_{jk}$ and $\theta^{(2)}_p$. Then the flow function calculation is reduced to the problem of unconstrained optimization (4). Given the boundary conditions (3) four unknown coefficients for the output layer are unambiguously expressed from the following equations:

$$\begin{align*}
\theta^{(2)}_p H_{1p} &= 0, p = 0, N_{hid}, \\
\theta^{(2)}_p H_{mp} &= Q, p = 0, N_{hid}, \\
\theta^{(2)}_j \frac{dH}{dx} |_{ij} &= 0, j = 1, N_{hid}, \\
\theta^{(2)}_j \frac{dH}{dx} |_{mj} &= 0, j = 1, N_{hid},
\end{align*}$$

where, $\frac{dH}{dx} |_{ij} = \left( H_{ij} \left( 1 - H_{ij} \right) \right) k \theta^{(1)}_{jk} \chi_{ik-1}$,  \( i = 1, m \)  \( k = 1, d \).

The search for weight coefficients is performed using the gradient descent method implemented in the engineering programming environment [16] in the form of built-in “fminunc” function. The initial values of the weights $\theta^{(1)}_{jk}, \theta^{(2)}_p$ are calculated as the values of a uniformly distributed random variable in the unit interval $(0;1)$. The convergence process for the functional (4) is shown in figure 3. On average, the iterative process converges for 200-300 iterations.
After the problem (4) is solved taking into account (5) and (8), the one-component velocity field is calculated as follows:

\[ V_i = \theta_j^{(1)} \frac{dH}{dx} \delta_{ij}, \quad i = 1, m \quad j = 1, N_{hid}. \]  

(9)

In figure 4a the velocity field (9) obtained for different values of the parameter that determines the degree of the polynomial in the activation function (6) are compared with the analytical solution [6]:

\[ V(x) = -\left(\frac{\Delta p}{q_1l_1}\right)^{1/2} \frac{1}{1+1/z} \left(x^{1+1/z} - L_2^{1+1/z}\right), \]  

(10)

where, \(\Delta p\) is a pressure drop in fluid flow direction; \(L_1\) is the length of the channel; \(L_2\) – is the one-half part of the width of the channel.

**Figure 3.** Iterative process convergence.

**Figure 4.** Comparison of the numerical and the analytical solutions: with the various values of the parameter ‘d’ of the activation function (6) (a) and with the various values of the pressure drop (b).

The validation procedure and the selection of hyperparameters \(m, d\), as well as the number of input and hidden neurons selection, was carried out by choosing the best solution obtained using the described method. An analogue of the labeled sample for the validation procedure is the analytical solution for Newtonian fluid obtained in the cited paper [6].
Validating the numerical algorithm in the form of a program for calculating the flow velocity field (figure 1) made it possible to establish that the error between the numerical solution and the analytical does not exceed 2% with the specific parameter values $d$ is equal to two or three. Despite the fact that with an increase of $d$ the error continues to decrease, a large degree of the polynomial (6) should not be specified in order to avoid the high variance effect [15]. Upon further program testing for determination the number of neurons at the input ($d + 1$) and the number of hidden layer neurons (corresponds to the number of terms in (6), but not less than 5, which is determined by the presence of boundary conditions (3)), it was established that the accuracy of the solution varies slightly with $m$ not less than twenty and $N_{hid}$ not less than ten. A further computational experiment was carried out with the following values of the model parameters: $d$ is equal to two, $m$ is equal to twenty, $N_{hid}$ is equal to ten. In figure 4b the numerical and analytical one-dimensional velocity fields obtained are compared at various pressure drops.

Testing of the model with the selected parameters was carried out on three samples which were the results of exact solutions for various values of the pressure drop along the channel length. The results are presented in figure 4b. The hyperparameters were determined on the basis of comparison the numerical solution with the exact solution. It is sufficient to use 3 neurons in the input layer. Then, the logistic activation function has a second-degree polynomial as an argument. It is sufficient to use 10 neurons in the hidden layer. Discretization of the flow domain is represented by a relatively small number of points $m$ equal to twenty. As the result, the error does not exceed 2%.

Summary
The paper presents the new numerical approach based on a combination of variational principle and the apparatus of artificial neural networks to solve boundary value problems in hydrodynamics. The formulation of the problem is kinematic. That is, as a result of the solution, the velocity field is determined. It is proposed to use the generalized Lagrange functional as the objective functional. This functional has a strict physical meaning connected with the value of inner power. A feed forward artificial neural network is used for minimization of the objective functional. The artificial neural network determines a set of functions parametrized by its weights, on which a solution is sought. Comparison of the numerical solution with the analytical solution for the simplest test problem demonstrated high accuracy of the numerical method.

The following advantages of the proposed numerical method were demonstrated in the work or obviously follow from the results obtained:
- a numerical solution is determined in the entire flow domain in the form of a continuous function that can be expressed analytically, therefore, the resulting solution can be easily analyzed;
- a numerical solution is sought on a set of functions that allow to describe more complex forms of velocity fields than coordinate functions in the Ritz method;
- a set of the unknown functions can be complicated by the use of multilayer neural networks with various activation functions;
- the back-propagation algorithm allows to increase the gradient descent method calculation rate and reduces the simulation time.

Further studies are related to solving problems of three-dimensional Non-Newtonian fluids flows. And with a comparison of the accuracy and simulation rate of the proposed method with the well-known methods of computational fluid dynamics, which are based on the finite element method and the control volume method.

Acknowledgments
This research was funded by Russian Science Foundation, grant number 16-19-0186. The authors gratefully acknowledge this support. The authors also express graduate to organizing committee of the conference “Virtual simulation, prototyping and industrial design -2019” for the possibility to publish this article in the Journal of Physics: Conference Series.
Authors Contribution
Elena Kornaeva developed a new numerical approach based on application of artificial neural networks to find minimum of generalized Lagrange functional. Alexey Kornaev and Elena Kornaeva developed mathematical and simulation models. Elena Kornaeva and Sergey Egorov performed the simulation experiment.

References
[1] Serrin J 1959 Mathematical Principles of Classical Fluid Mechanics 8 (Berlin: Springer Verlag) p 256
[2] Schechter R 1959 The Variational Method in Engineering (New York: McGraw-Hill Book Company) p 287
[3] Roache P 1998 Fundamentals of Computational Fluid Dynamics (New Mexico: Hermosa Pub) p 276
[4] Hori Y 2006 Hydrodynamic Lubrication (Tokyo: Yokendo Ltd) p 239
[5] Savin L, Kornaev A and Kornaeva E 2015 Variational principle in hydrodynamic lubrication theory International Journal of Mathematical Models and Methods in Applied Sciences 9 114
[6] Kornaev A 2019 On proof of the generalized lagrange variational principle Lec. N. Mech. Eng. Proc. 4th Int. Conf. on Industrial Engineering (New York: Springer Science + Business Media) 1071 DOI: 10.1007/978-3-319-95630-5_112
[7] Neural Net Fitting. MathWorks Help Center. Available online: https://www.mathworks.com/help/deeplearning/ref/neuralnetfitting-app.html
[8] Gonzales R 2008 Neural networks for variational problems in engineering, PhD Thesis, Technical University of Catalonia
[9] Beidokhti R and Malek A 2009 Solving initial-boundary value problems for systems of partial differential equations using neural networks and optimization techniques Journal of the Franklin Institute 346 898 https://doi.org/10.1016/j.jfranklin.2009.05.003
[10] Huan H, Xiao B, Xiong H, Wu Z, Mu Y and Song H 2019 Applications of deep learning to relativistic hydrodynamics Nucl. Phys. A 982 927-930 DOI: 10.1016/j.nuclphysa.2018.11.004
[11] Mehta P, Bukov M, Wang Ch-H, Day A G R, Richardson C, Fisher C K and Schwab D J 2019 A high-bias, low-variance introduction to machine learning for physicists Phys. Rep. 810 1 DOI: 10.1016/j.physrep.2019.03.001
[12] Herman F and Qureshi F 2013 Newton’s law of viscosity, Newtonian and non-Newtonian fluids Encyclopedia of tribology ed Q Wang and Y-W Chung (New York: Springer Science + Business Media) pp 2416-2420 DOI: 10.1007/978-0-387-92897-5_143
[13] Milne-Thomson L 1962 Theoretical Hydrodynamics (London: Macmillan & Co. Ltd.) p 660
[14] Murphy K 2012 Machine Learning: A Probabilistic Perspective (London: MIT Press Cambridge, Massachusetts) 1067
[15] Haykin S 2008 Neural Network Theory (Berlin: Springer-Verlag Berlin Heidelberg) p 939
[16] GNU Octave URL: http://www.gnu.org/software/octave (accessed 30 Sept 2019)