Geology, Geophysics & Environment 2025, vol. 51 (3): 255–270

# Non-history-based DEM model for predictions of numerical earthquakes

Piotr Klejment

*Institute of Geophysics, Polish Academy of Sciences, Warsaw, Poland, e-mail: pklejment@igf.edu.pl, ORCID ID: 0000-0003-3018-3816*



Received: 30 March 2025; accepted: 10 July 2025; first published online: 15 September 2025

***Abstract:*** Stick-slip phenomena roughly describe the behavior of a tectonic fault. A simplified model of stick-slip events is often assumed in laboratory experiments and numerical simulations of laboratory earthquakes. This work proposes a more advanced approach. The Discrete Element Method (DEM) was used to generate a numerical model for simulating the laboratory earthquakes in which the granular layer was taken into account. The proposed model takes into account an irregular, random pattern of stress increase and decrease in such a system. At 5,000 selected, regularly spaced time points, the so-called "checkpoints", 25 parameters were measured, describing the average state of all particles forming the numerical fault at a given moment. The created dataset was used to train the Random Forest algorithm, and then, as part of the tests, this algorithm was used to predict subsequent stick-slip events. The algorithm made predictions solely on the basis of information about the current parameters of the particles. Importantly, the predictions made did not use the history of previous stick-slip events. Feature Importance and SHapley Additive exPlanations (SHAP) were used to assess the contribution of individual particle physical parameters to the prediction results.

***Keywords:*** granular fault gouge, supervised machine learning, Discrete Element Method

## INTRODUCTION

According to current knowledge, earthquakes are unpredictable. At least, there is no general rule that can predict the approach of an earthquake with a high degree of probability (the time and magnitude of an event), or detect its precursors (Beroza et al. 2021, Senatorski 2023, Kubo et al. 2024). It is known that shear stress accumulates on a fault and increases until a critical point is reached, leading to a sudden failure. This pattern repeats itself cyclically and is similar to the stick-slip phenomenon, where the earthquake is identified with the slip phase (Rouet-Leduc et al. 2017). On the basis of this periodicity, attempts were made to predict the time and magnitude of subsequent earthquakes, based on past measurements of a given fault.

An example of this approach was the prediction of the Parkfield earthquake in California (Rouet-Leduc et al. 2017). Based on seismic data from 1857 to 1966, it was calculated that the average time between subsequent quakes is 21.9 ±3.1 years, and on this basis, the next major quake was predicted for 1988–1993. Ultimately, it occurred in 2004. This shows that this methodology can sometimes work and sometimes not because there is too much randomness in earthquakes. This not only concerns the complex structure and physics of the faults, which are still not fully explained, but also the influence of other faults. Therefore, earthquakes can only describe

https://doi.org/10.7494/geol.2025.51.3.255 ISSN 2299-8004 | e-ISSN 2353-0790 | Wydawnictwa AGH

stick-slip cycles in which the time and magnitude of subsequent shocks are irregular (they take into account the element of randomness).

In recent years, machine learning has shown great promise for new quality and progress in earthquake research. Machine learning algorithms are great at dealing with large datasets and can find hidden and non-obvious (linear and non-linear) relationships and connections between data. One of the most pioneering and innovative works in this field was the article published by Rouet-Leduc et al. (2017). They performed experimental work on laboratory earthquake models and showed that, by using supervised machine learning models based on the acoustic signal emitted during stick-slip cycles, it is possible to predict, with high accuracy, the moment of occurrence of subsequent slip events. An extremely important achievement of this work was predicting movements within the fault not based on the history of previous stick-slip events, but solely on the basis of signals currently recorded in the model.

Machine learning is appearing more and more often in such applications. For example, Bolton et al. (2020) showed that machine learning can predict the timing and magnitude of laboratory earthquakes using acoustic emissions. The evolution of acoustic energy is critical for earthquake prediction, but the relationship between acoustic energy and the fault zone leading to failure is still poorly understood. In addition to physical laboratory experiments, numerical simulations of laboratory-scale shear fault experiments have also been performed in recent years to study earthquake physics. Ren et al. (2019) used a rigid particle-based Discrete Element Method (DEM) to predict the macroscopic friction of the system. Their predictions used signals from single particles. Mollon et al. (2023) showed that laboratory rock shear experiments exhibiting the stick-slip phenomenon can be similar in nature to the dynamics of natural earthquakes. As for other examples of research on similar topics, in the work of Zhou et al. (2018) the process of stick-slip shear failure was investigated. In Li and Zhou (2021) attempts were made to predict the magnitude, location, and time of the laboratory earthquake of granite. The prediction model of stress drop was established based on a backpropagation neural network considering different system parameters. In the work of Zhou et al. (2021) it was shown how the tectonic movement rate, in situ stresses (e.g., normal stress) and fault roughness may change during the long evolutionary histories of seismogenic faults. In the work of Li and Zhou (2022), the problem of understanding the mechanism of stick-slip instability of faults, which is of great significance for earthquake forecast and seismic fortification, was addressed. In the work of Zhou et al. (2023) mechanoluminescent material was developed to measure the full-interface shear stress of the fault. The experimental and numerical results showed that the fault interface can be divided into shear stress-strengthening zone and shear stress-weakening zone.

This paper presents the application of supervised machine learning to predict laboratory earthquakes, but using the DEM numerical model. In general, the existing numerical and laboratory works assume a simplified model of stress accumulation on a fault. Here, an improved DEM model was proposed to take into account the random nature of stress changes in the fault and was applied to the prediction of stick-slip phenomena using supervised machine learning, using only continuous measurements of the model state (changes in energy, velocities, displacements and others) to train the algorithm.

## DATA COLLECTION AND METHODOLOGY

### Discrete Element Method

Among the many numerical methods used to describe the evolution of granular systems, the Discrete Element Method (DEM) is one of the most popular due to its ability to describe material behavior at the particle level. DEM models allow the analysis of phenomena in a way that is not possible in the laboratory, including access to data about every particle inside the model at any time during the simulation.

In general, the flowchart of each DEM simulation (Fig. 1) starts by defining the model geometry, and then the main program loop begins. At each step, contacts between particles are checked, the forces acting and the resulting displacements are calculated, and then the geometry is updated. This pattern is repeated until the end of the simulation.

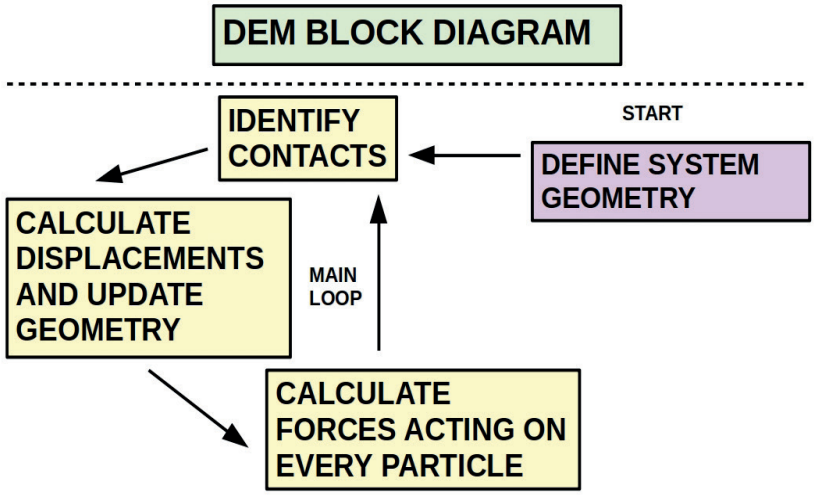



***Fig. 1.*** *General scheme of DEM simulation (O'Sullivan 2011)*

The DEM is a numerical method for simulating brittle and granular materials, in particular rock physics and nonlinear earthquake dynamics (Cundall & Strack 1979). A unique feature of this method is the consideration of all particles in the system individually, depending on needs, along with their linear and rotational movements. One of the most popular DEM implementations is the open-source software ESyS-Particle (Wang et al. 2006, Wang 2009, Weatherley et al. 2010, Abe et al. 2014), used in this work.

## Applied DEM model

The proposed model had a size of 10 mm × 10 mm × 1 mm and consisted of 2,465 particles that were subjected to the force of friction. The model included horizontal layers with different properties, which were a schematic representation (Ferdowsi 2014) of the structure of a real fault (Fig. 2). The upper and lower extreme layers represented undeformed bedrock and consisted of particles with a radius of 0.2 mm.

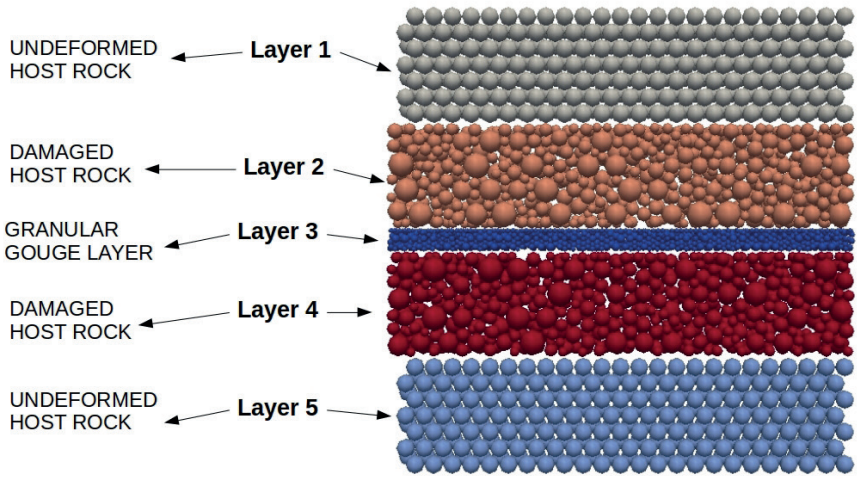



***Fig. 2.*** *Schematic representation of a DEM model of a numerical fault*

The two intermediate layers represented damaged host rock and consisted of particles with radii from 0.1 to 0.3 mm. In the center of the model, there was a granular layer with particles ranging from 0.05 to 0.1 mm. Mature faults in the Earth contain granular fault gouges created by the friction and fragmentation of bedrock at the core of the fault zone. Interactions in the granular fault zone play an important role in the dynamics of the fault, the rate of stress build-up, the recurrence time between slip instabilities, stress drop, and energy release (Ferdowsi 2014), therefore, the presence of the granular zone was taken into account in the model.

The presented DEM model was too simplified to directly refer to real tectonic faults. This model only schematically reflects the layered structure of the real fault. However, it was characterized by a similar grain size distribution in the inner layers to the distribution of particle sizes within fault zones (Fig. 3). In fault zones, grain size distribution typically follows a power law distribution, where a small number of large particles are found alongside a large number of small particles, which was shown, for example, by Muto et al. (2015). The particle size distribution in the inner layers of the model is presented in Figure 3 for particles with radius smaller than 0.2 mm.

As already mentioned, the stick-slip cycle roughly describes the phenomena occurring on a tectonic fault, leading to a sudden failure. In such an assumption (Fig. 4A), stress slowly increases in the fault until it reaches critical strength. Then, a sudden failure occurs and the accumulated energy is released, which is equivalent to an earthquake. Then this process is repeated cyclically. However, a real earthquake is a random phenomenon whose occurrence is influenced by a huge number of factors. Therefore, a better description of this phenomenon is provided by an irregular cycle (Fig. 4B), in which all variables and values change over time.

Here, a compressive force was applied to the DEM model in the direction of the *y*-axis, and a shear force was applied in the direction of the *x*-axis. Periodic boundary conditions were applied. The simulation time step was selected according to the Courant formula (O'Sullivan & Bray 2004, O'Sullivan 2011) to ensure numerical stability. It is known that predicting stick-slip cycles (laboratory earthquakes) is possible in the case of stick-slip cycles appearing in laboratory experiments, such as (Rouet-Leduc et al. 2017), or in numerical experiments (Ren et al. 2019). However, in such cases, a simplified model (with constant confining force and constant shear velocity) of the system is generally adopted (Fig. 4A). Here, the simulation followed a novel applied force scheme. In the model prepared for the purposes of this work, the random nature of forces acting on the system was achieved (Fig. 4C). This was done by randomizing the critical strength and time to failure after each slip event. Therefore, each simulation performed with this model was characterized by a unique cycle of the force and stress changes in the model.

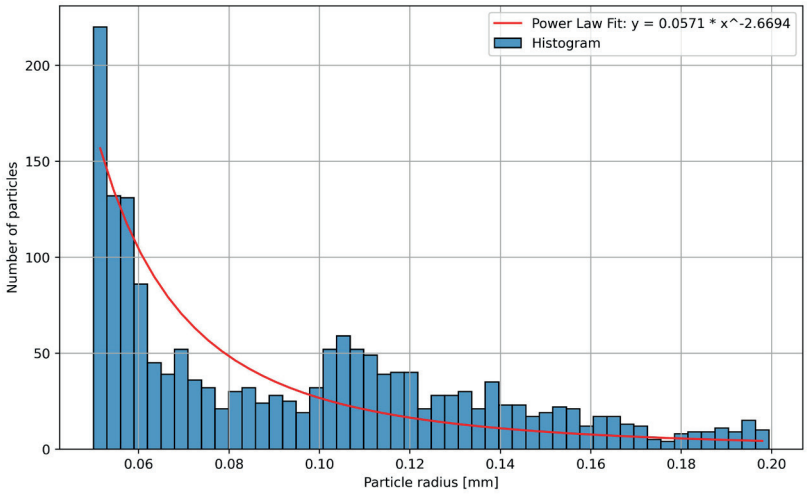



***Fig. 3.*** *Particle size distribution in the model follows a power law distribution*

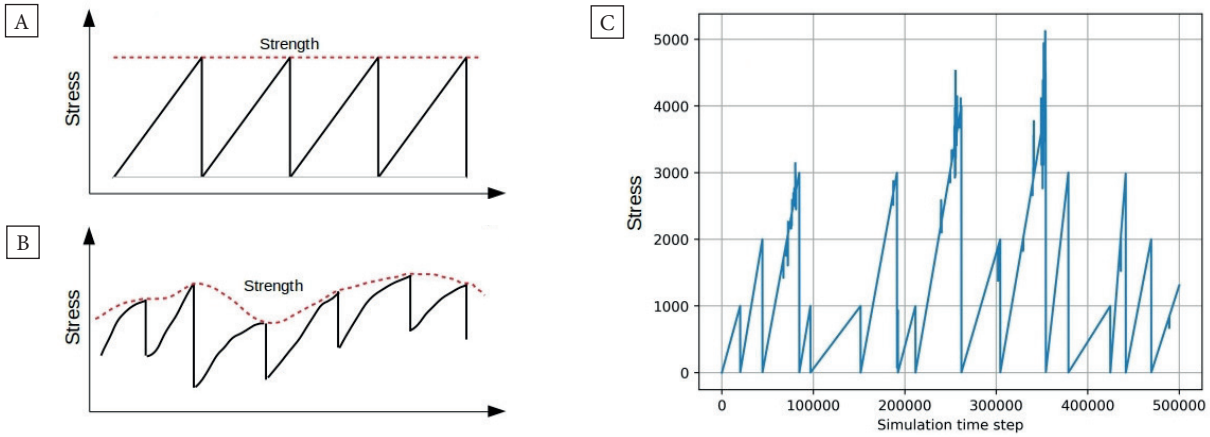



***Fig. 4.*** *The stick-slip cycle roughly describes the increases and decreases in stress on a tectonic fault (Ferdowsi 2014). In the simplest assumption, such a cycle is considered regular (A), but in reality, it is a largely random and irregular phenomenon (B). In this work, a numerical fault model was created in which the irregular and random nature of stress changes (C) was recreated*

For comparison with other scientific studies, in the work by Rouet-Leduc et al. (2017), the laboratory system was a two-fault configuration that contained fault gouge material submitted to double direct shear. In Ren et al. (2019), a DEM model was used, and the constant confining stress was applied to the model from above and the shearing constant velocity was applied from below. In the work of Ma et al. (2022) the stick-slip behavior of a slowly sheared granular system was examined using Discrete Element Method simulations. The granular gouge was confined by two rough particle walls used to apply the shear loading and normal pressure. The normal pressure was applied to the top wall. The granular gouge was sheared by moving the bottom wall in the $x$ direction with a constant velocity.

Here, in the presented model, a constant shearing velocity was applied to the model from the bottom, and a compressive force from the top. The innovation introduced here was that the force was not constant, but time-varying, which provided additional variability of forces and stresses inside the model. At the beginning of the simulation, the maximum value of the force $F$ was randomly selected from an arbitrary range ($F_1$ = 1,000, $F_2$ = 5,000) and the time of action of the force $ft$ was randomly selected from an arbitrary range ($ft_1$ = 1,000, $ft_2$ = 60,000) of time steps. Then the force was increased linearly to a value of $F$ over time $ft$. A uniform distribution was used for the sampling. Then the randomization procedures were repeated in the same way until the end of the simulation time. Regardless of the changes in the compressive force applied at the top of the sample, a constant shear rate was applied to the bottom of the sample at all times, which did not change. The proposed mechanism introduced additional, random changes in the forces and stresses in the model, because, in reality, over long time, real faults are subject to variable rather than constant forces.

Due to the cost of calculations, the model was a small equivalent of a real fault. However, it was shown that stick-slip phenomena in both experimental and numerical models obey Gutenberg Richter's law and show similar $b$-values (Dahmen et al. 2011, Dorostkar 2018). Additionally, Kanamori and Anderson (1975), Prieto et al. (2004) showed that earthquakes have a fractal nature. This means that the phenomenon of the millimeter or centimeter scale is similar to the phenomenon of the meter or kilometer scale (Rivière et al. 2018). This justifies conducting research on small laboratory and numerical models. Here, in the presented model, if a large number of time steps is considered (like 5,000,000 time steps in the Figure 5A), the number of events (understood as decreases in forces, and therefore stresses in the system) resulting from stick-slip cycles obey the Gutenberg–Richter law (Fig. 5B).

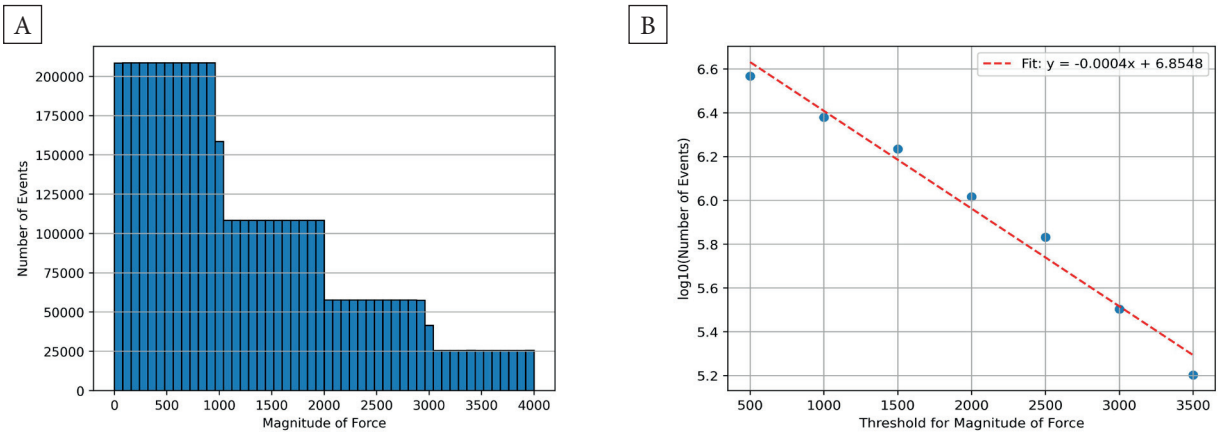



***Fig. 5.*** *Distribution of the number of events in the DEM model*

## Methodology for applying machine learning

Supervised machine learning, which is ideal for analyzing structured datasets, was used. The Random Forest (RF) machine learning algorithm was used, which has already proven its worth in working with this type of problems (Rouet-Leduc et al. 2017). The dataset created as a result of collecting information about particles creating a numerical fault during the simulation was divided into a training part and a testing part. In the training part, the algorithm compares the input data (independent variables) and the predicted output (dependent variable) in order to learn linear and non-linear relationships. In the testing dataset, the algorithm receives only independent variables and must predict the final variable itself (in this case, time to failure). The algorithm's performance was assessed using a metric specific to regression (Géron 2019, Raschka & Mirjalili 2019, Liu 2020): coefficient of determination $R_2$ (1). $R_2$ is the proportion of the variance in the dependent variable predicted from the independent variable, with the best possible result being 1.0 (100%) and the worst possible being 0 (0%). $R_2$ can be also negative when the chosen model does not follow the trend of the data, so fits worse than a horizontal line:

$$R^2 = 1 - \frac{\Sigma\left(y_i - y_{\text{pred}}\right)^2}{\Sigma\left(y_i - y_{\text{mean}}\right)^2} \quad (1)$$

where $y_i$ is the simulation results, and $y_{\text{pred}}$ is the predictions of machine learning algorithms.

The influence of individual parameters on the dependent parameter was determined using two techniques: Feature (Gini) Importance and SHapley Additive exPlanations (SHAP) (2). In Feature (Gini) Importance, the higher the result, the greater the impact of a given microparameter on the prediction results. Feature Importance is calculated as the (normalized) total criterion reduction resulting from this feature. SHAP comes from game theory and permits the explanation of the influence of input parameters on the output of a machine learning algorithm:

$$\Phi_i = \sum_{S \subseteq \{1,\ldots,p\}\{i\}} \frac{|S|!\left(p - |S| - 1\right)!}{p!}\left[val\left(S \cup \{i\}\right) - val\left(S\right)\right] \quad (2)$$

where $S$ is a subset of the features used in the model, $p$ is the number of features, $val(S)$ is the prediction for feature values in set $S$ that are marginalized over features that are not included in set $S$, and the SHAP value for feature value $j$ is the value of the $j$-th feature contributed $\Phi_i$ to the prediction of this particular instance compared to the average prediction for the dataset (Štrumbelj & Kononenko 2014).

Numerical simulations were performed using ESyS-Particle, GenGeo and Paraview. Analysis and visualization of results and application of the machine learning algorithm were performed using Python3 and its libraries (Matplotlib, Seaborn, Pandas, Scikit-learn and others).

# RESULTS

## Recording of the signal from particles

The primary goal of the simulations was to try to predict irregular stick-slip cycles (time to failure). Here, the time to failure is the time between successive stress (force) drops in the created numerical system DEM described above. Slip events were to be predicted not on the basis of the history of previous cycles (which is impossible due to the randomness implemented in the model), but on the basis of continuous measurements of the current state of the particles creating the numerical slip. Twenty-five (25) features describing the average state of particles were measured (Table 1). These features were related to the total displacement from the beginning of the simulation, the displacement from the previous time step, velocity, force and kinetic energy.

***Table 1***
*Presentation of 25 features, independent parameters, used to train the RF algorithm*

| Total displacement | One step displacement | Velocity | Force | Energy |
|---|---|---|---|---|
| **Feature1**<br>**tmx_m**<br>mean of particles' total displacement in the X direction<br><br>**Feature2**<br>**tmx_s**<br>standard deviation of particles' total displacement in the X direction<br><br>**Feature3**<br>**tmy_m**<br>mean of particles' total displacement in the Y direction<br><br>**Feature4**<br>**tmy_s**<br>standard deviation of particles' total displacement in the Y direction<br><br>**Feature5**<br>**tmz_m**<br>mean of particles' total displacement in the Z direction<br><br>**Feature6**<br>**tmz_s**<br>standard deviation of particles' total displacement in the Z direction | **Feature7**<br>**omx_m**<br>mean of particles' one step displacement in the X direction<br><br>**Feature8**<br>**omx_s**<br>standard deviation of particles' one step displacement in the X direction<br><br>**Feature9**<br>**omy_m**<br>mean of particles' one step displacement in the Y direction<br><br>**Feature10**<br>**omy_s**<br>standard deviation of particles' one step displacement in the Y direction<br><br>**Feature11**<br>**omz_m**<br>mean of particles one step displacement in the Z direction<br><br>**Feature12**<br>**omz_s**<br>standard deviation of particles' one step movement in the Z direction | **Feature13**<br>**vx_m**<br>mean of the X component of the particles' velocity<br><br>**Feature14**<br>**vx_s**<br>standard deviation of the X component of the particles' velocity<br><br>**Feature15**<br>**vy_m**<br>mean of the Y component of the particles' velocity<br><br>**Feature16**<br>**vy_s**<br>standard deviation of the Y component of the particles' velocity<br><br>**Feature17**<br>**vz_m**<br>mean of the Z component of the particles' velocity<br><br>**Feature18**<br>**vz_s**<br>standard deviation of the Z component of the particles' velocity | **Feature19**<br>**fx_m**<br>mean of the X component of the net force acting on the particles<br><br>**Feature20**<br>**fx_s**<br>standard deviation of the X component of the net force acting on the particles<br><br>**Feature21**<br>**fy_m**<br>mean of the Y component of the net force acting on the particles X<br><br>**Feature22**<br>**fy_s**<br>standard deviation of the Y component of the net force acting on the particles<br><br>**Feature23**<br>**fz_m**<br>mean of the Z component of the net force acting on the particles<br><br>**Feature24**<br>**fz_s**<br>standard deviation of the Z component of the net force acting on the particles | **Feature25**<br>ek<br>particles' average kinetic energy |

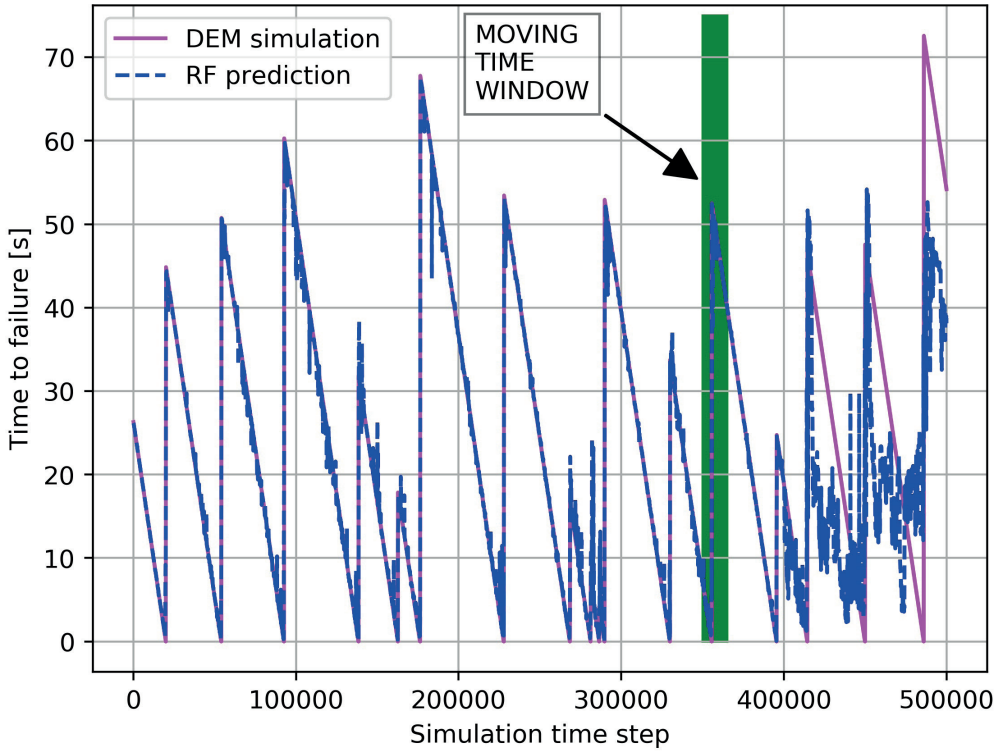



***Fig. 6.*** *Information about 25 features describing the state of the system at a given moment was collected in 5,000 checkpoints regularly selected during the simulation, using the concept of a moving time window. The collected data was used to train the RF algorithm to predict the time to subsequent slip events in the stick-slip cycle*

During the simulation, a so-called checkpoint was established at every 1,000th time step. At each checkpoint, the features described above were measured. This is the concept of the so-called sliding time window (Fig. 6). The obtained data was used to train the RF supervised machine learning algorithm, and on the testing dataset, the algorithm used these features to predict the time remaining until the next event. Tests of the algorithm were performed on a sequence of stick-slip events not seen by the model, and predictions were made only on the basis of information from individual time windows. This means that predictions based on the physical characteristics of the model did not use the history of previous events.

## Time to failure predictions in the separate experiments

Eight separate simulations were performed: Experiment 1, Experiment 2, Experiment 3, Experiment 4, Experiment 5, Experiment 6, Experiment 7, and Experiment 8. Each was characterized by an individual course of stick-slip cycles due to the random external force implemented in the model – random magnitude, and random duration. The time course from cycles in the form of time remaining until the next slip events in each experiment is presented in Figure 7.

In each experiment, 80% of the initial simulation time was allocated to training the RF algorithm. The $R_2$ metric was used as the prediction quality metric. On the training set, the algorithm learned the applicable rules very well in each of the eight experiments, obtaining $R_2$ equal to 0.99 (Fig. 8). Testing was performed in the final 20% of the simulation duration. Here, the algorithm had to predict completely new data and predict future events that were largely random. The best fit was obtained for Experiment 1 with the metric $R_2 = 0.51$. In this case, it can be said that the algorithm was able to predict the future with a fair degree of accuracy in the stick-slip cycle.

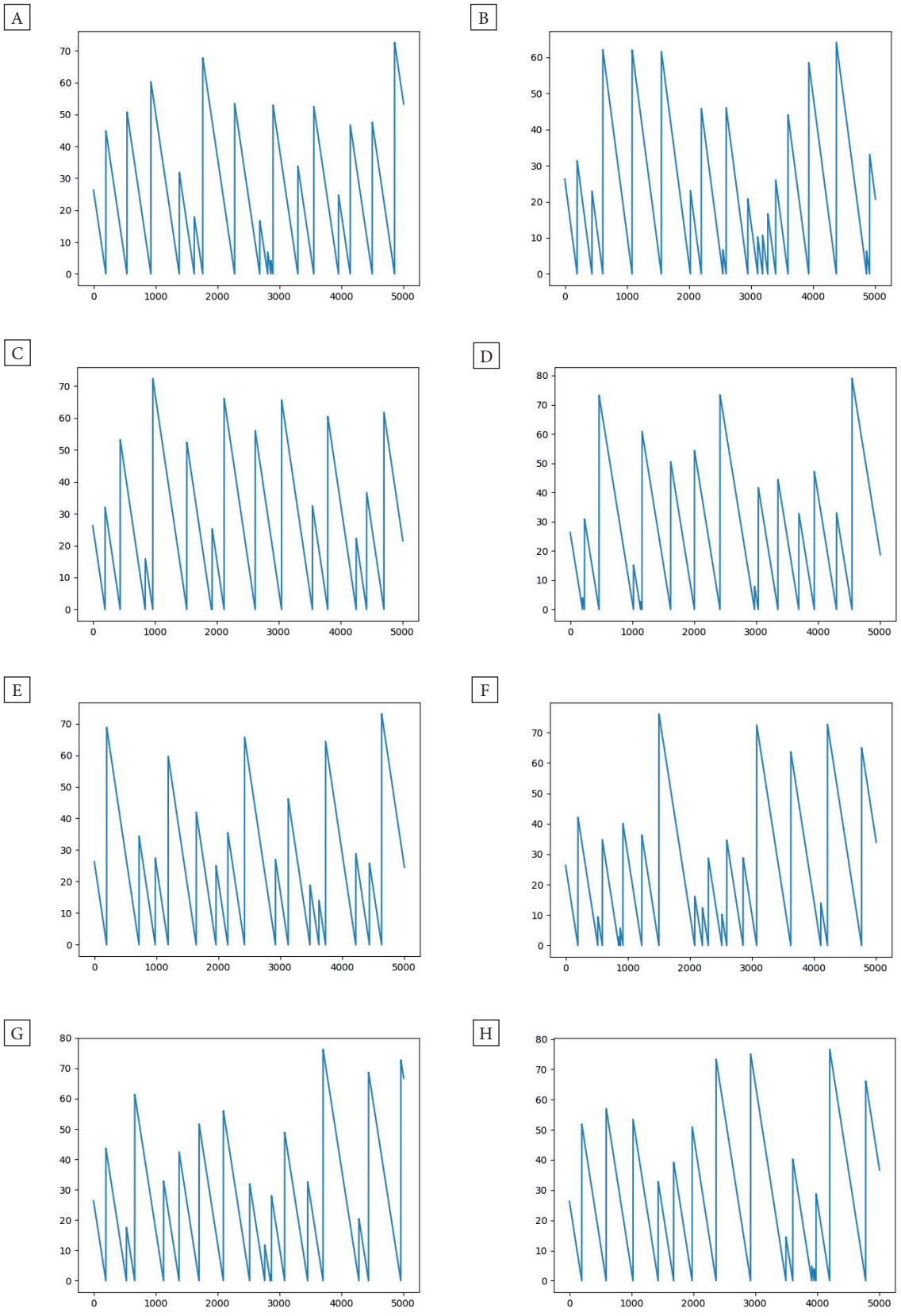



***Fig. 7.*** *The presented DEM model uses a random scheme of external forces applied to the numerical fault, resulting in the unique random nature of stick-slip cycles. Here, the time remaining until the next slip event is presented for each of the eight experiments: 1 (A), 2 (B), 3 (C), 4 (D), 5 (E), 6 (F), 7 (G), and 8 (H)*

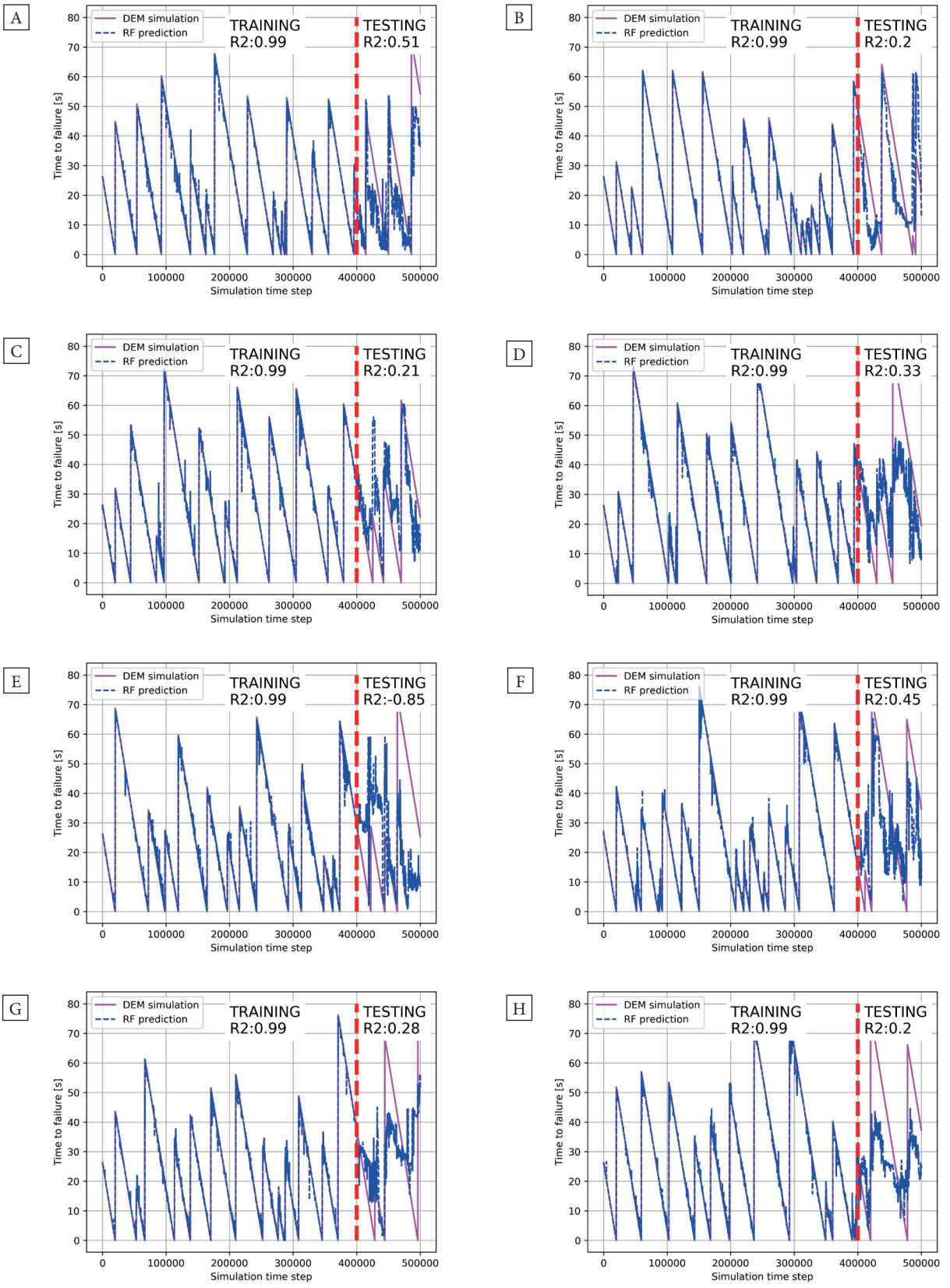



***Fig. 8.*** *Comparison of the prediction results of the RF algorithm with the DEM simulation results for eight experiments: 1 (A), 2 (B), 3 (C), 4 (D), 5 (E), 6 (F), 7 (G), and 8 (H). The algorithm was trained on 80% of the first simulation time steps, and testing was performed on the remaining 20% of time steps*

## An attempt to make universal predictions

The results presented in Figure 8 correspond to the situation where each experiment was treated as a separate event and the RF algorithm was trained separately for each experiment. The next step attempted a more universal approach. Four experiments, 1–4, were devoted exclusively to training the algorithm, and the trained algorithm was tested for experiments 5–8. The results are shown in Figure 9 and are promising. As shown, the shape of the prediction approximately corresponds to the shape of the stick-slip cycles from numerical simulations, and the time gradually decreases as the next event approaches. However, the issue of precision leaves much to be desired. Nevertheless, the results seem interesting, especially since the algorithm makes predictions only on the basis of the current signal recorded among the particles.

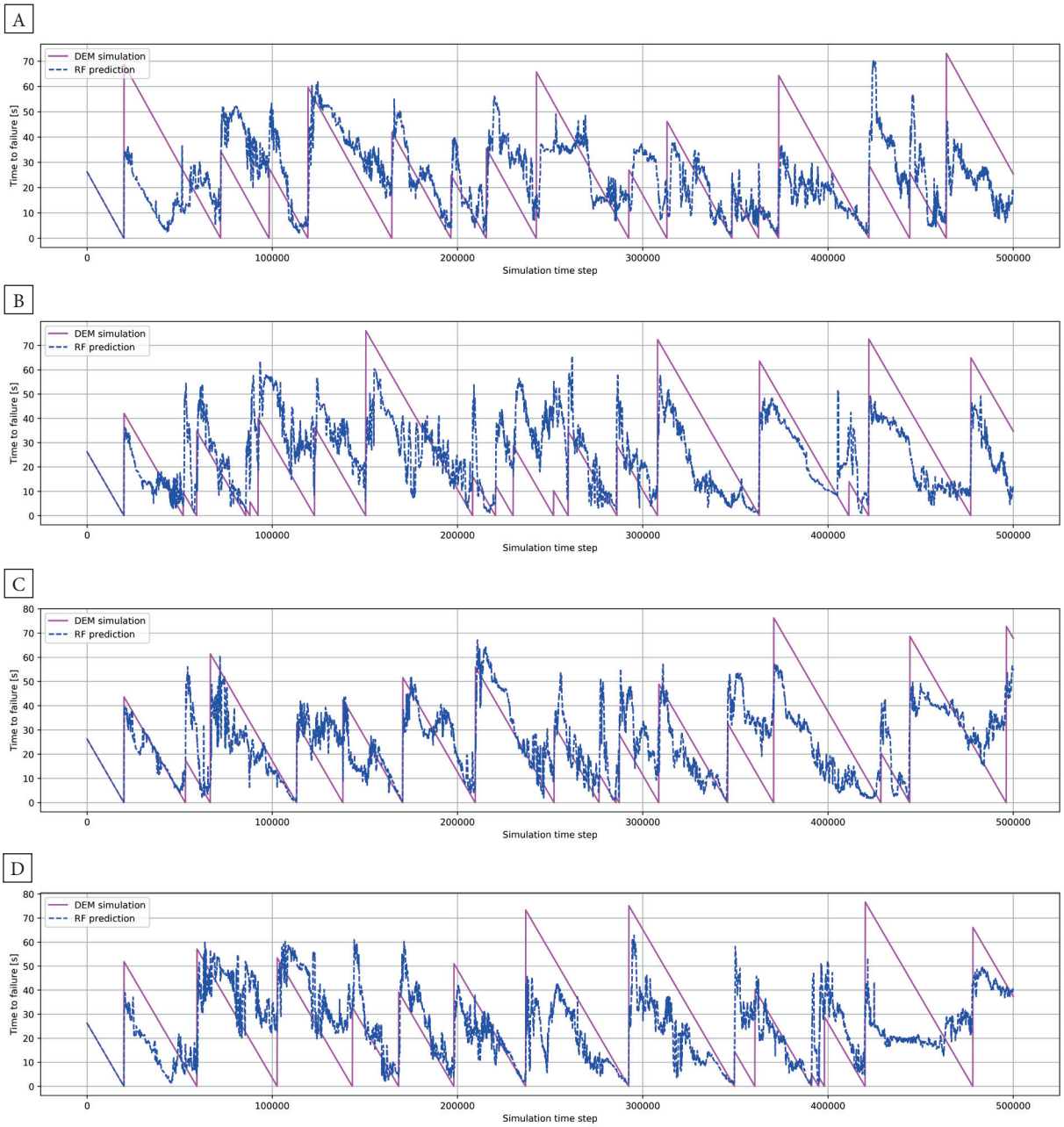


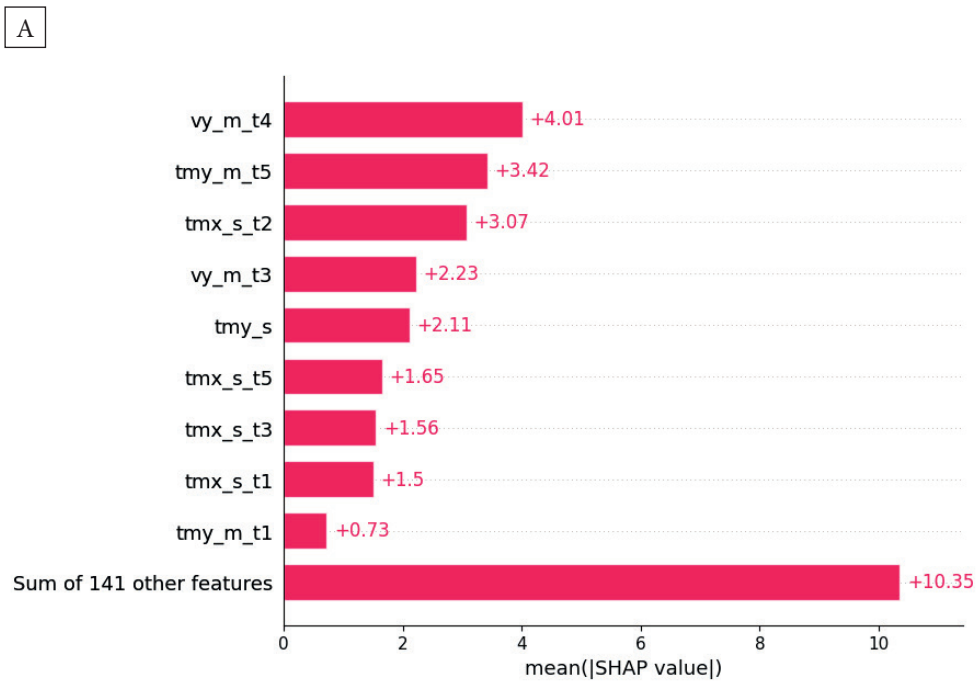

***Fig. 9.*** *Comparison of DEM simulation results and RF algorithm predictions for four experiments: 5 (A), 6 (B), 7 (C), and 8 (D). The algorithm was trained for experiments 1–4, and experiments 5–8 were a set of new test data for it*

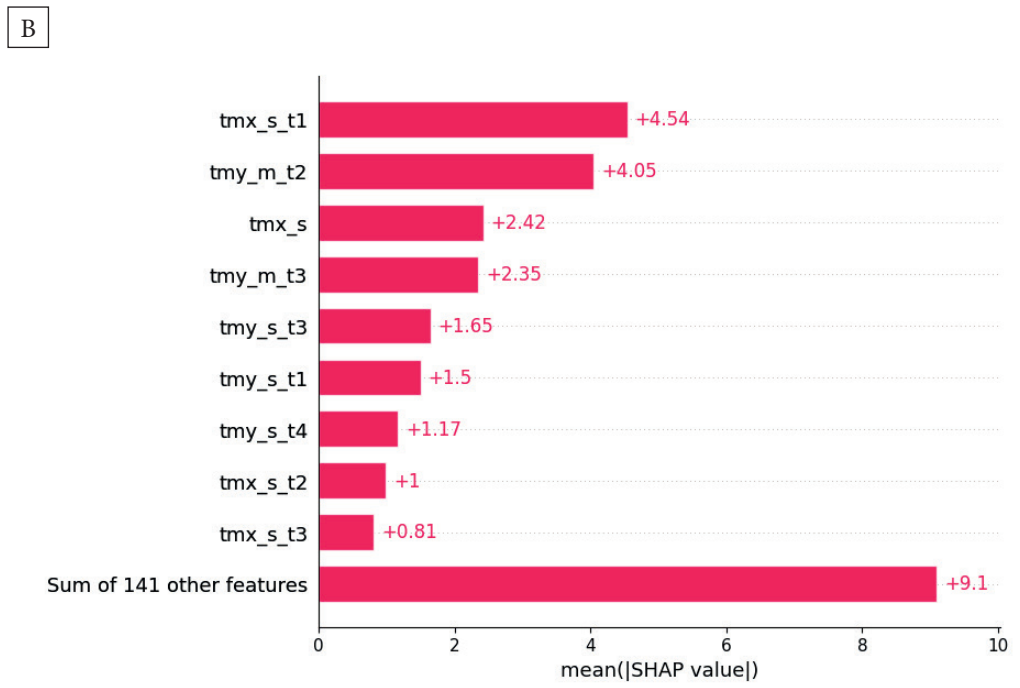
## In-depth analysis of the impact of features on predictions

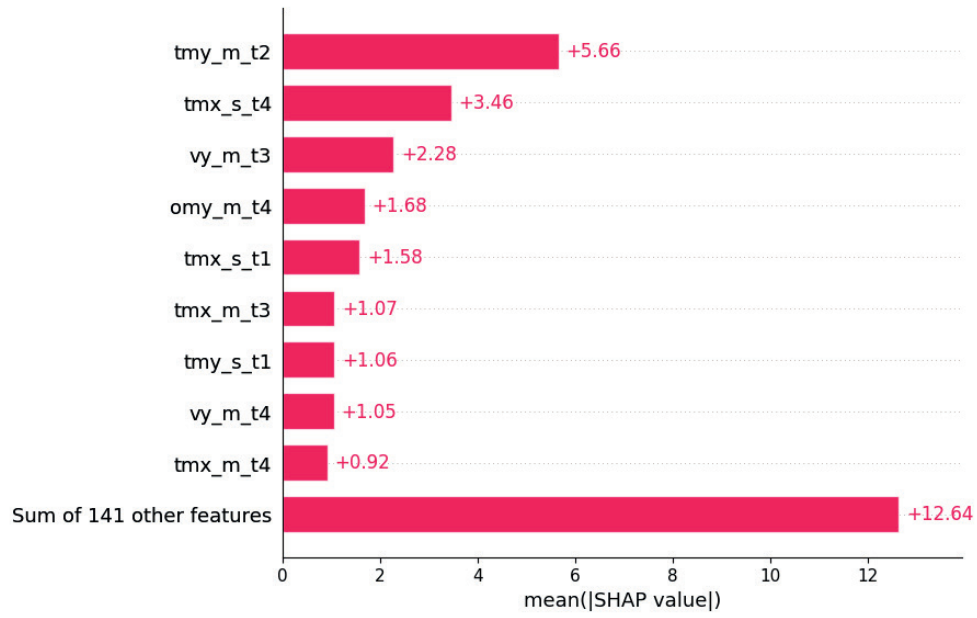
Features from Table 1, i.e., mean and standard deviation of particles' total displacement (tmx_m, tmx_s, tmy_m, tmy_s, tmz_m, tmz_s), mean and standard deviation of particles' one step displacement (omx_m, omx_s, omy_m, omy_s, omz_m, omz_s), mean and standard deviation of the particles' velocity (vx_m, vx_s, vy_m, vy_s, vz_m, vz_s), mean and standard deviation of the net force acting on the particles (fx_m, fx_s, fy_m, fy_s, fz_m, fz_s), and particles' average kinetic energy (ek), were also calculated individually for individual layers (Fig. 2), adding the symbols of individual layers t1, t2, t3, t4, and t5 in the designations. In this way, a dataset of 150 features was obtained. Feature Importance was used and the 20 most important features from the new dataset were presented for each of the eight experiments (Table 2).

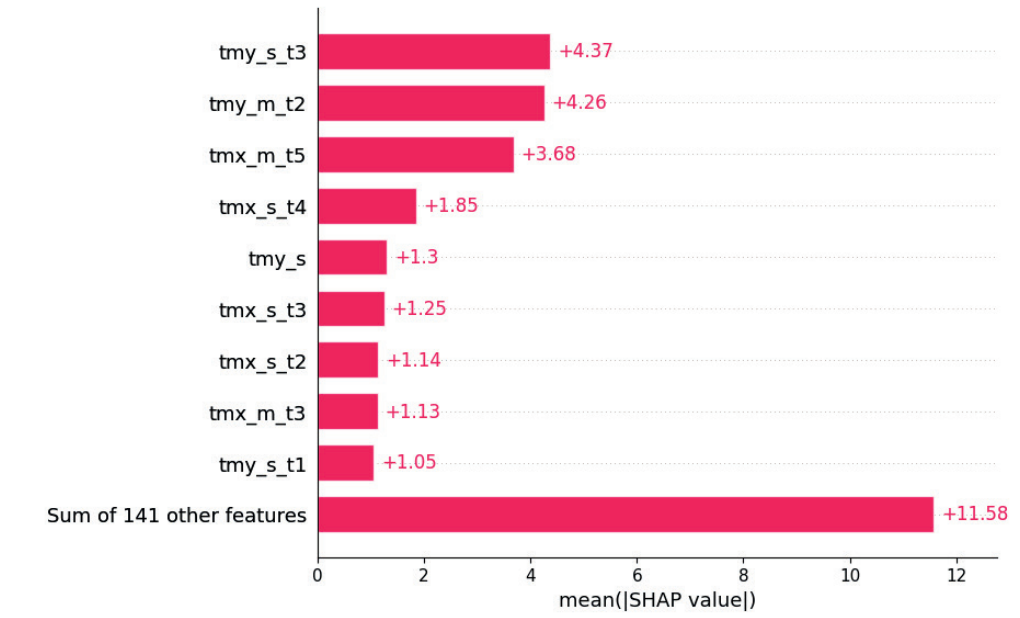
***Table 2***
*Feature Importance for the 20 features arranged in ascending order and calculated for each of the eight experiments 1–8*

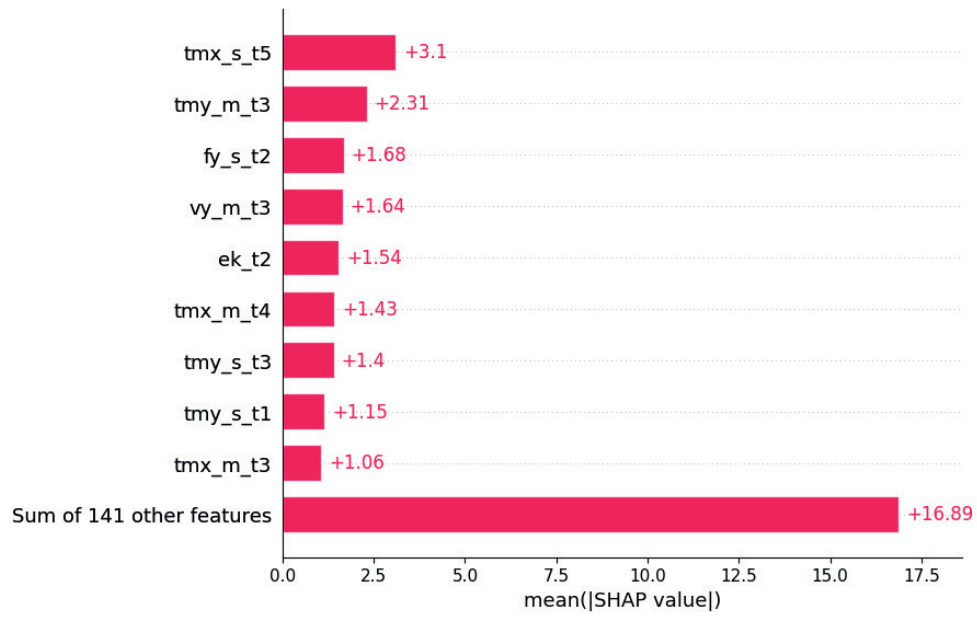
| **Experiment 1** | | **Experiment 2** | | **Experiment 3** | | **Experiment 4** | |
|---|---|---|---|---|---|---|---|
| tmy_s_t3 | 0.007 | tmx_m_t3 | 0.006 | tmy_m_t3 | 0.011 | tmx_s_t5 | 0.010 |
| tmy_m_t4 | 0.008 | tmx_m_t1 | 0.007 | tmy_s_t1 | 0.012 | vy_m | 0.010 |
| ek_t3 | 0.009 | vy_m_t5 | 0.007 | tmx_m_t5 | 0.013 | vy_m_t4 | 0.011 |
| tmx_s_t4 | 0.009 | vy_m_t3 | 0.008 | tmx_s | 0.013 | tmx_s_t1 | 0.011 |
| tmx_m_t1 | 0.010 | tmx_m_t4 | 0.010 | tmx_m_t2 | 0.013 | tmy_m | 0.012 |
| tmy_s_t4 | 0.010 | tmx_s_t5 | 0.010 | tmy_s | 0.014 | tmy_s_t2 | 0.012 |
| tmy_m_t1 | 0.012 | tmx_m | 0.013 | tmx_s_t5 | 0.015 | tmx_m | 0.014 |
| tmy_m_t2 | 0.013 | tmy_s_t2 | 0.017 | vy_m_t4 | 0.015 | tmy_s_t4 | 0.016 |
| tmy_m_t3 | 0.014 | vy_m_t4 | 0.026 | tmx_s_t3 | 0.018 | tmy_m_t1 | 0.017 |
| tmy_s_t5 | 0.015 | tmx_s_t3 | 0.030 | tmx_m_t1 | 0.020 | tmy_s_t5 | 0.018 |
| tmx_m_t4 | 0.015 | tmx_s_t2 | 0.032 | vx_s | 0.020 | tmx_s_t2 | 0.022 |
| tmy_s | 0.028 | omy_m | 0.034 | tmy_s_t3 | 0.023 | tmx_s | 0.023 |
| fy_s_t4 | 0.035 | vx_m_t3 | 0.038 | tmx_m_t3 | 0.042 | omy_m_t3 | 0.029 |
| vy_m_t3 | 0.043 | tmy_s_t3 | 0.040 | tmx_m_t4 | 0.043 | tmx_m_t3 | 0.035 |
| tmx_s_t1 | 0.048 | tmy_s_t4 | 0.041 | tmx_s_t2 | 0.044 | tmy_s | 0.044 |
| tmx_s_t3 | 0.052 | tmx_s_t4 | 0.043 | tmx_s_t1 | 0.045 | tmx_s_t3 | 0.067 |
| tmx_s_t5 | 0.063 | tmy_s_t1 | 0.056 | vy_m_t3 | 0.048 | tmx_s_t4 | 0.086 |
| tmx_s_t2 | 0.093 | tmx_s | 0.119 | tmx_s_t4 | 0.133 | tmy_m_t2 | 0.111 |
| tmy_m_t5 | 0.110 | tmx_s_t1 | 0.150 | omy_m_t4 | 0.144 | tmx_m_t5 | 0.117 |
| vy_m_t4 | 0.275 | tmy_m_t2 | 0.234 | tmy_m_t2 | 0.182 | tmy_s_t3 | 0.208 |
| **Experiment 5** | | **Experiment 6** | | **Experiment 7** | | **Experiment 8** | |
| tmx_s | 0.013 | tmx_s_t4 | 0.010 | vy_m_t3 | 0.005 | vy_m_t4 | 0.014 |
| tmy_s_t1 | 0.015 | tmx_m_t1 | 0.011 | tmx_m_t3 | 0.006 | vy_m | 0.015 |
| vy_m_t3 | 0.017 | omy_m_t4 | 0.011 | tmx_m_t2 | 0.006 | tmx_s | 0.016 |
| omy_m | 0.018 | tmy_s_t1 | 0.012 | tmx_s_t1 | 0.010 | tmx_s_t2 | 0.017 |
| tmy_s | 0.019 | tmx_m_t4 | 0.013 | tmy_s | 0.012 | tmy_m_t1 | 0.017 |
| tmx_s_t1 | 0.019 | vx_s_t2 | 0.015 | tmy_m_t2 | 0.015 | tmy_s_t4 | 0.018 |
| tmx_m | 0.019 | vy_m_t3 | 0.016 | tmx_s | 0.015 | vx_s | 0.018 |
| omy_m_t4 | 0.021 | tmy_s | 0.017 | tmy_s_t4 | 0.017 | omy_m_t3 | 0.019 |
| tmx_m_t3 | 0.021 | tmy_m_t3 | 0.021 | tmx_m_t5 | 0.026 | vy_m_t3 | 0.020 |
| tmy_m_t2 | 0.027 | tmx_m_t5 | 0.026 | tmx_s_t4 | 0.036 | tmx_s_t5 | 0.021 |
| tmx_s_t2 | 0.032 | vx_s | 0.028 | tmx_s_t3 | 0.037 | tmx_m_t4 | 0.022 |
| vy_m_t4 | 0.035 | omy_m_t1 | 0.031 | tmy_s_t2 | 0.041 | tmy_s_t5 | 0.028 |
| tmx_m_t4 | 0.041 | tmx_s_t3 | 0.034 | tmy_s_t1 | 0.046 | tmx_m_t2 | 0.032 |
| omy_m_t1 | 0.042 | tmx_s | 0.035 | tmx_m_t1 | 0.053 | tmx_s_t3 | 0.036 |
| tmx_s_t3 | 0.044 | vx_s_t1 | 0.040 | tmx_s_t2 | 0.054 | tmy_s_t3 | 0.040 |
| ek_t2 | 0.051 | tmx_s_t5 | 0.048 | tmx_m_t4 | 0.056 | tmy_s_t2 | 0.041 |
| tmy_m_t3 | 0.058 | tmx_s_t2 | 0.055 | tmx_s_t5 | 0.070 | tmy_m_t5 | 0.057 |
| fy_s_t2 | 0.066 | tmx_s_t1 | 0.092 | vx_m_t3 | 0.090 | omy_m | 0.058 |
| tmy_s_t3 | 0.095 | tmy_m_t2 | 0.130 | tmy_s_t3 | 0.107 | tmx_s_t4 | 0.123 |
| tmx_s_t5 | 0.128 | tmy_s_t2 | 0.208 | tmy_m_t3 | 0.220 | tmy_m_t2 | 0.215 |

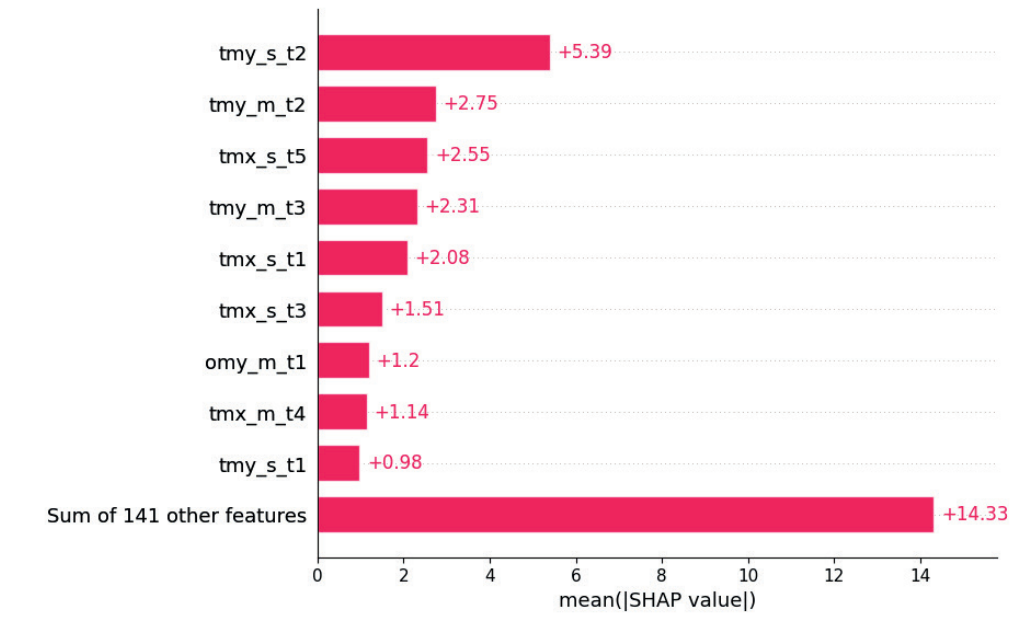


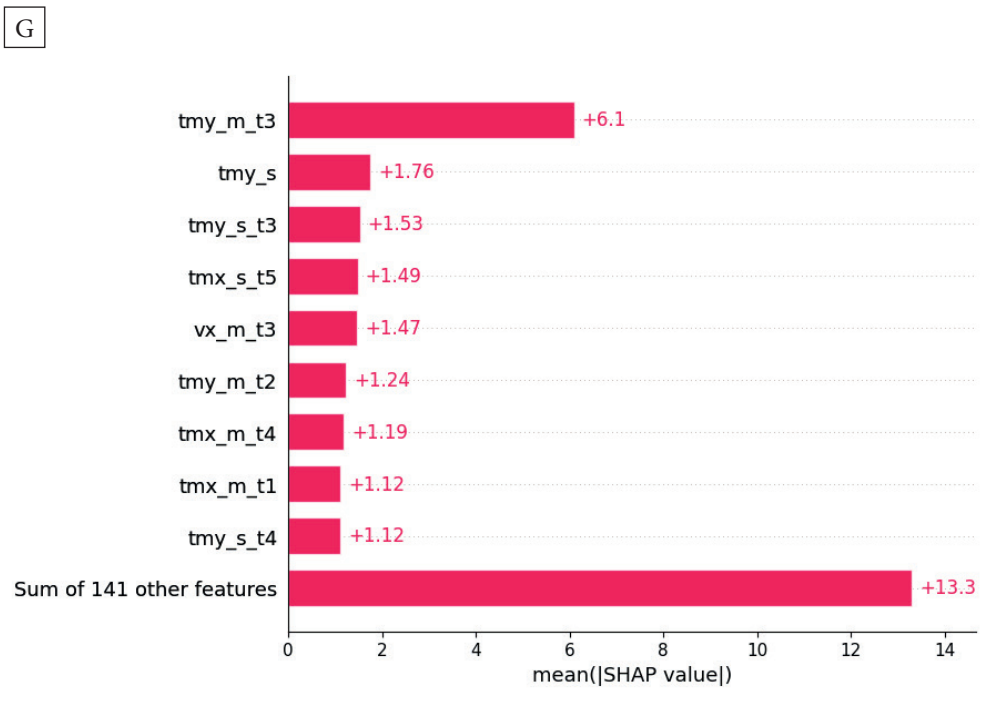



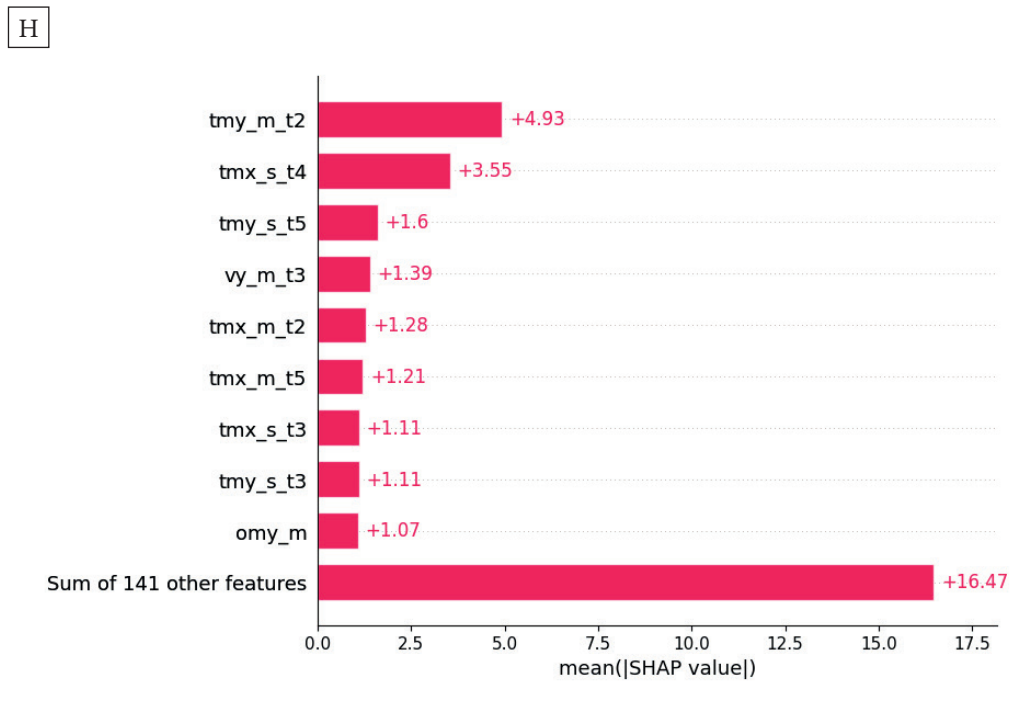
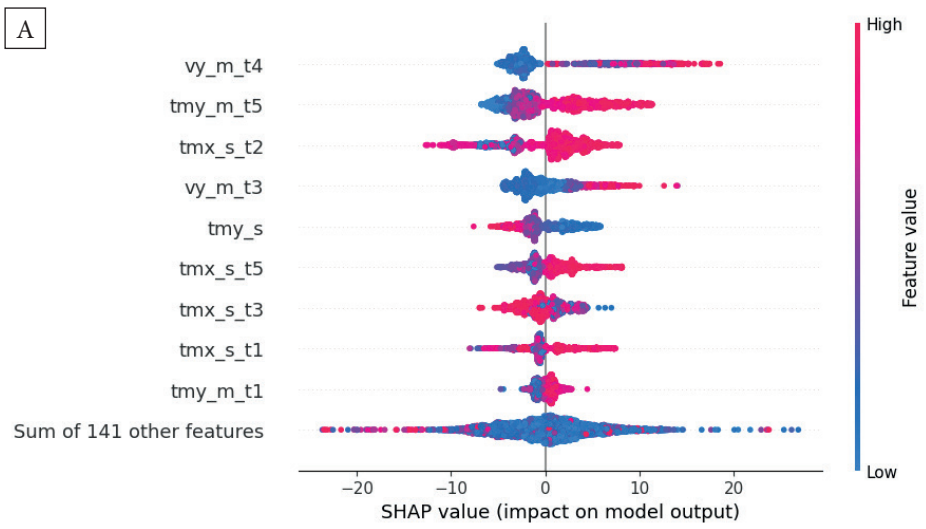
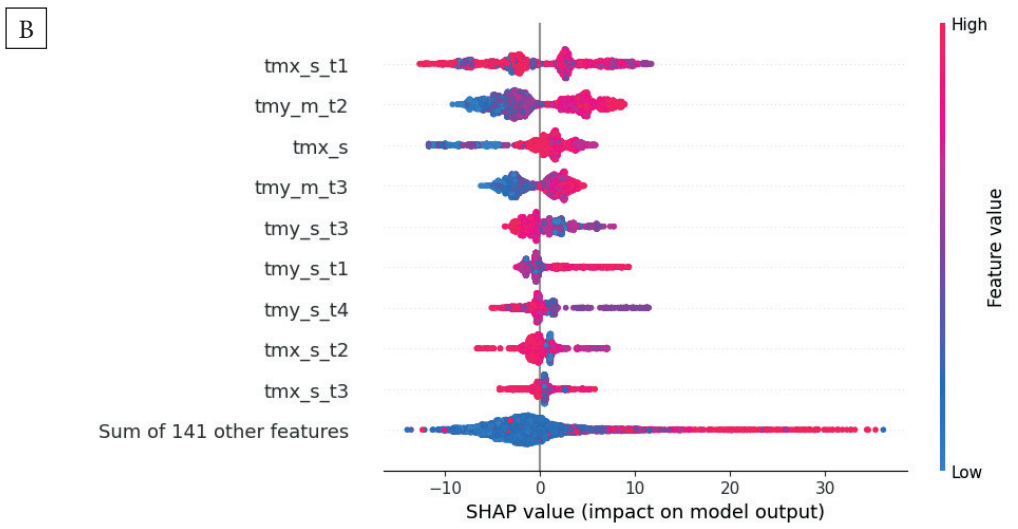
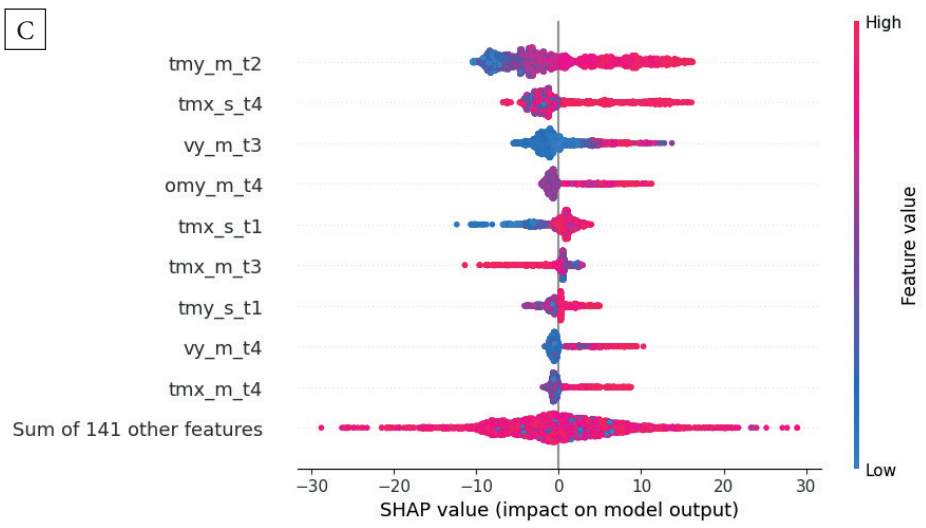
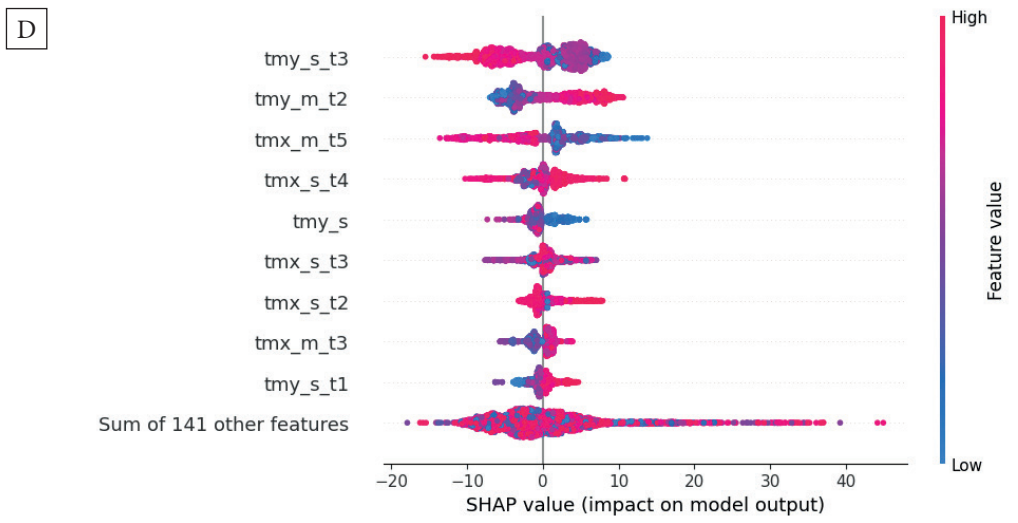



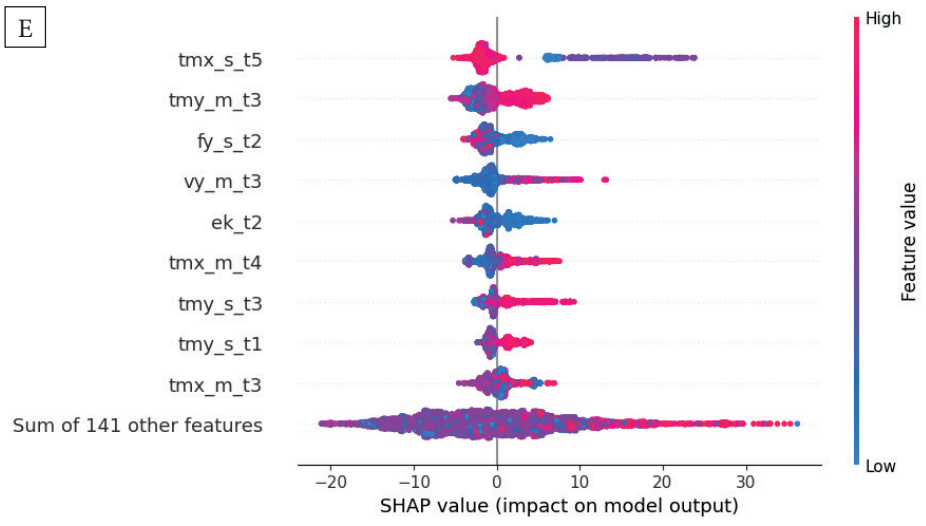

***Fig. 10.*** *The average value of the SHAP function for the features that most influence the prediction quality of the machine learning algorithm for all eight experiments: 1 (A), 2 (B), 3 (C), 4 (D), 5 (E), 6 (F), 7 (G), and 8 (H)*

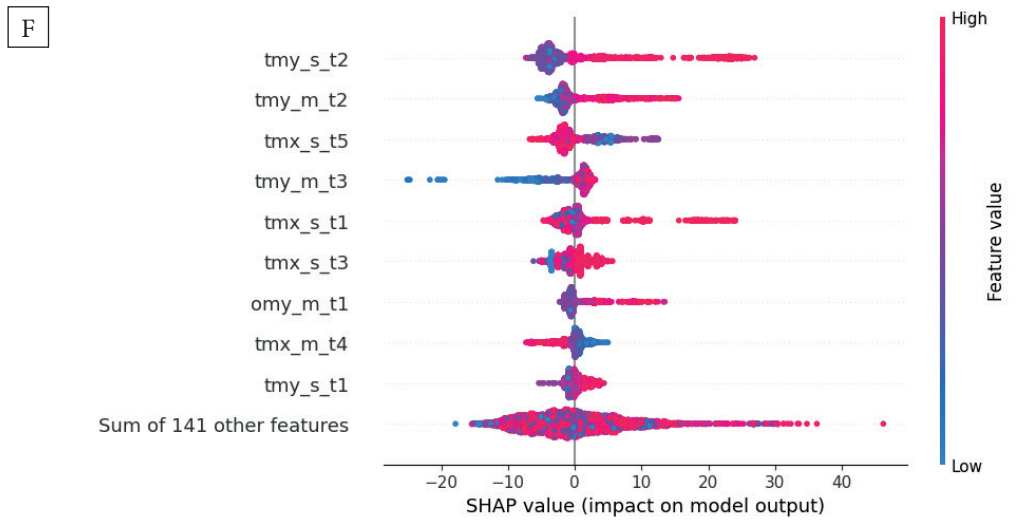


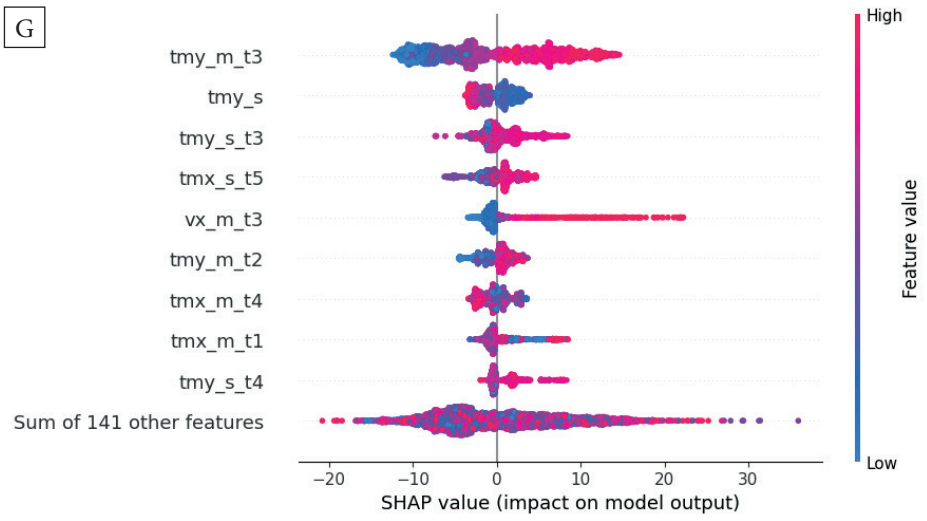



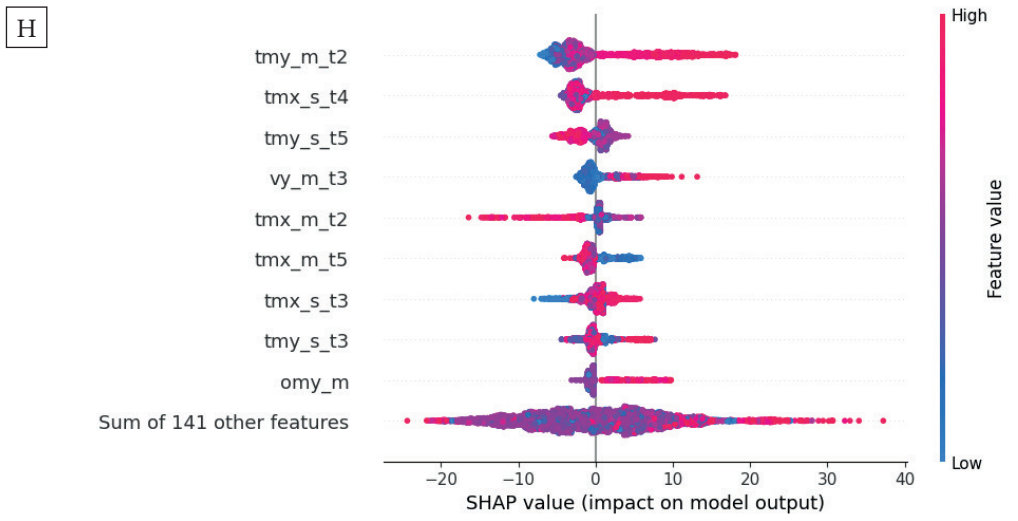




***Fig. 11.*** *The use of the SHAP function to analyze the impact of individual features on the final predictions of the machine learning algorithm, taking into account the impact of individual layers in the model for all eight experiments: 1 (A), 2 (B), 3 (C), 4 (D), 5 (E), 6 (F), 7 (G), and 8 (H)*

It turns out that in most cases the tmy and tmx functions had some of the greatest influence on the predictions, i.e., the total displacement from the beginning of the simulation to a given moment in the direction of the *x*- and *y*-axes. This rule is repeated for each experiment, but the details are different for each experiment. Respectively, the three highest Feature Importance coefficients for each experiment were as follows:

- Experiment 1: tmx_s_t2 (0.093), tmy_m_t5 (0.110), vy_m_t4 (0.275);
- Experiment 2: tmx_s (0.119), tmx_s_t1 (0.150), tmy_m_t2 (0.234);
- Experiment 3: tmx_s_t4 (0.133), omy_m_t4 (0.144), tmy_m_t2 (0.182);
- Experiment 4: tmy_m_t2 (0.111), tmx_m_t5 (0.117), tmy_s_t3 (0.208);
- Experiment 5: fy_s_t2 (0.066), tmy_s_t3 (0.095), tmx_s_t5 (0.128);
- Experiment 6: tmx_s_t1 (0.092), tmy_m_t2 (0.130), tmy_s_t2 (0.208);
- Experiment 7: vx_m_t3 (0.090), tmy_s_t3 (0.107), tmy_m_t3 (0.220);
- Experiment 8: omy_m (0.058), tmx_s_t4 (0.123), tmy_m_t2 (0.215).

However, there is no clear influence of individual layers on the algorithm's predictions.

The results after applying the average SHAP function for all eight experiments are similarly distributed (Fig. 10 on the interleaf), confirming the leading role of the total displacement in influencing the final predictions. SHAP values, not averaged, but individual for individual features, are presented in Figure 11 (on the interleaf).

It is discernible that for the vx speed, lower feature values result in a lower SHAP value, i.e., a smaller impact on the model result. However, in the case of total displacements, tmx or tmy, no general dependencies are visible.

## DISCUSSION AND CONCLUSIONS

Short-term deterministic earthquake prediction is impossible according to the current state of knowledge. No general rules have been discovered that can predict the time and magnitude of the next event with a high degree of probability (Senatorski 2023). Machine learning, which is great at finding hidden and non-obvious relationships between data, offers new opportunities. For example, supervised machine learning methods help to create new generations of earthquake catalogs (Beroza et al. 2021, Kubo et al. 2024). It represents progress in probabilistic earthquake prediction (Beroza et al. 2021). However, there is still a lack of complete information about earthquake physics. For practical reasons, direct access to faults for measurements is limited. Laboratory and numerical experiments help to understand such phenomena. Datasets collected in repeatable laboratory conditions and during numerical simulations are a natural environment for the application of machine learning. As shown in previous works (Rouet-Leduc et al. 2017, Ren et al. 2019, Huang et al. 2024), machine learning can be very effective in such predictions. However, real faults are extremely complex structures that form interacting systems (Senatorski 2023), an example of which can be triggered earthquakes (Ferdowsi 2024). Therefore, simplified laboratory and numerical setups are not able to recreate all the nuances present in reality.

Machine learning helps by automatically searching for patterns and relationships from huge datasets, especially non-linear relationships and this work is a continuation of scientific research in this area. Its new contribution consists of proposing a numerical model in which the external forces acting on a numerical fault change in a random and irregular manner. Based on eight numerical experiments, it was possible to use the Random Forest algorithm for predictions both on individual experiments and on their entirety. Very good $R_2$ statistics were achieved on the training set, while the results on the test set are promising, but require further research. Feature Importance and the SHAP value were used to determine the contribution of individual features to the final predictions and it was found that the highest contribution comes from total shifts from the *x*- and *y*-axes directions.

All readily available parameters in the DEM model describing the state of particles (25 features) were used to obtain the largest possible dataset for the machine learning algorithm and to improve the quality of prediction.

Some of these parameters are available only in simulation and are difficult or impossible to measure in real measurements, which should lead to a reduction in the number of parameters used in future simulations, e.g. to those related to velocity and acceleration (because they are related to acoustic emission). It is a well-known fact that machine learning algorithms often perform well within the dataset on which they were trained and tested, but fail on completely new data. The random nature of the changes in the force acting on the model made these cycles challenging for the algorithm, but it can be seen that the shape of the curves in Figure 7 is roughly reproduced. The $R_2$ coefficients were equal to: Experiment 5 – 0.006, Experiment 6 – 0.204, Experiment 7 – 0.363, Experiment 8 – 0.154, respectively. This indicates that the algorithm needs more data to better learn the relationships in the dataset. Reducing the resolution of the time window and extending the simulations themselves would provide more data and therefore should improve the quality of the predictions.

*The author would like to acknowledge the support of the NCN MINIATURA 6 grant (National Science Centre, Poland) No. DEC-2022/06/X/ST10/01581. The author would also like to thank the statutory funds of the Institute of Geophysics, Polish Academy of Sciences for their support. Prof. Guilhem Mollon is thanked for supplying the opportunity of a scientific stay in Lyon in June 2023. The work of the authors of ESyS-Particle Tutorial (Abe et al. 2014) is appreciated and the author would also like to thank the anonymous Reviewers of the article for their contributions. The author would like to thank the Scientific Information and Publishers Department of IG PAS for the language correction.*

*Special thanks are due to my Mother and Uncle from Gdynia for their support.*

*Code and results are available in the Data Portal of the Institute of Geophysics Polish Academy of Sciences (IG PAS):*

- *licence: Creative Commons Attribution 4.0 International (CC BY 4.0);*
- *DOI: https://doi.org/10.25171/InstGeoph_PAS_IG-Data_Predictions_of_Numerical_Earthquakes;*
- *link: https://dataportal.igf.edu.pl/dataset/predictions_of_numerical_earthquakes.*

## REFERENCES

Abe S., Boros V., Hancock W. & Weatherley D., 2014. *ESyS-Particle Tutorial and User's Guide. Version 2.3.1*. https://launchpad.net/esys-particle [access: 30.06.2025].

Beroza G.C., Segou M. & Mousavi S.M., 2021. Machine learning and earthquake forecasting – next steps. *Nature Communications*, 12, 4761. https://doi.org/10.1038/s41467-021-24952-6.

Bolton D.C., Shreedharan S., Rivière J. & Marone C., 2020. Acoustic energy release during the laboratory seismic cycle: Insights on laboratory earthquake precursors and prediction. *Journal of Geophysical Research: Solid Earth*, 125, e2019JB018975. https://doi.org/10.1029/2019JB018975.

Cundall P.A. & Strack O.D.L., 1979. A discrete numerical model for granular assemblies. *Géotechnique*, 29(1), 47–65. https://doi.org/10.1680/geot.1979.29.1.47.

Dahmen K., Ben-Zion Y. & Uhl J.A., 2011. Simple analytic theory for the statistics of avalanches in sheared granular materials. *Nature Physics*, 7, 554–557. https://doi.org/10.1038/nphys1957.

Dorostkar O., 2018. *Stick-slip dynamics in dry and fluid saturated granular fault gouge investigated by numerical simulations*. ETH Zürich [PhD thesis]. https://doi.org/10.3929/ethz-b-000283977.

Ferdowsi B., 2014. *Discrete element modeling of triggered slip in faults with granular gouge: Application to dynamic earthquake triggering*. ETH Zürich [PhD thesis]. https://doi.org/10.3929/ethz-a-010232908.

Géron A., 2019. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. 2nd ed. O'Reilly Media, Inc, Sebastopol, CA.

Huang W., Gao K. & Feng Y., 2024. Predicting stick-slips in sheared granular fault using machine learning, optimized dense fault dynamics data. *Journal of Marine Science and Engineering*, 12(2), 246. https://doi.org/10.3390/jmse12020246.

Kanamori H. & Anderson D.L., 1975. Theoretical basis of some empirical relations in seismology. *Bulletin of the Seismological Society of America*, 65(5), 1073–1095.

Kubo H., Naoi M. & Kano M., 2024. Recent advances in earthquake seismology using machine learning. *Earth, Planets and Space*, 76(1), 36. https://doi.org/10.1186/s40623-024-01982-0.

Li C.-Q. & Zhou X.-P., 2021. Laboratory earthquake prediction of granite. *Tribology International*, 160, 107003. https://doi.org/10.1016/j.triboint.2021.107003.

Li C.-Q. & Zhou X.-P., 2022. Visualization of stick-slip shear failure process of granite by 3D reconstruction technique and DEM. *Tribology International*, 176, 107923. https://doi.org/10.1016/j.triboint.2022.107923.

Liu Y.H., 2020. *Python Machine Learning by Example: Build Intelligent Systems Using Python, TensorFlow 2, PyTorch, and Scikit-Learn*. 3rd ed. Packt Publishing, Birmingham.

Ma G., Mei J., Gao K., Zhao J., Zhou W. & Wang D., 2022. Machine learning bridges microslips and slip avalanches of sheared granular gouges. *Earth and Planetary Science Letters*, 579, 117366. https://doi.org/10.1016/j.epsl.2022.117366.

Mollon G., Aubry J. & Schubnel A., 2023. Laboratory earthquakes simulations – typical events, fault damage, and gouge production. *Journal of Geophysical Research: Solid Earth*, 128, e2022JB025429. https://doi.org/10.1029/2022JB025429.

Muto J., Nakatani T., Nishikawa O. & Nagahama H., 2015. Fractal particle size distribution of pulverized fault rocks as a function of distance from the fault core. *Geophysical Research Letters*, 42(10), 3811–3819. https://doi.org/10.1002/2015GL064026.

O'Sullivan C., 2011. *Particulate Discrete Element Modelling: A Geomechanics Perspective*. CRC Press, London.

O'Sullivan C. & Bray J.D., 2004. Selecting a suitable time step for discrete element simulations that use the central difference time integration scheme. *Engineering Computations*, 21(2–4), 278–303. https://doi.org/10.1108/02644400410519794.

Prieto G.A., Shearer P.M., Vernon F.L. & Kilb D., 2004. Earthquake source scaling and self-similarity estimation from stacking P and S spectra. *Journal of Geophysical Research: Solid Earth*, 109(B8), B08310. https://doi.org/10.1029/2004JB003084.

Raschka S. & Mirjalili V., 2019. *Python Machine Learning: Machine Learning and Deep Learning with Python, Scikit-Learn, and TensorFlow 2*. 3rd ed. Packt Publishing, Birmingham.

Ren C.X., Dorostkar O., Rouet-Leduc B., Hulbert C., Strebel D., Guyer R.A., Johnson P.A. & Carmeliet J., 2019. Machine learning reveals the state of intermittent frictional dynamics in a sheared granular fault. *Geophysical Research Letters*, 46(13), 7395–7403. https://doi.org/10.1029/2019GL082706.

Rivière J., Lv Z., Johnson P.A. & Marone C., 2018. Evolution of b-value during the seismic cycle: Insights from laboratory experiments on simulated faults. *Earth and Planetary Science Letters*, 482, 407–413. https://doi.org/10.1016/j.epsl.2017.11.036.

Rouet-Leduc B., Hulbert C., Lubbers N., Barros K., Humphreys C.J. & Johnson P.A., 2017. Machine learning predicts laboratory earthquakes. *Geophysical Research Letters*, 44(18), 9276–9282. https://doi.org/10.1002/2017GL074677.

Senatorski P., 2023. Sejsmiczność, trzęsienia ziemi i lingwistyka [Seismicity, earthquakes and linguistics]. *Przegląd Geofizyczny*, 68(3–4), 185–197. https://doi.org/10.32045/PG-2023-042.

Štrumbelj E. & Kononenko I., 2014. Explaining prediction models and individual predictions with feature contributions. *Knowledge and Information Systems*, 41(3), 647–665. https://doi.org/10.1007/s10115-013-0679-x.

Wang Y.C., 2009. A new algorithm to model the dynamics of 3-D bonded rigid bodies with rotations. *Acta Geotechnica*, 4(2), 117–127. https://doi.org/10.1007/s11440-008-0072-1.

Wang Y., Abe S., Latham S. & Mora P., 2006. Implementation of particle-scale rotation in the 3-D lattice solid model. *Pure and Applied Geophysics*, 163(9), 1769–1785. https://doi.org/10.1007/s00024-006-0096-0.

Weatherley D.K., Boros V.E., Hancock W.R. & Abe S., 2010. Scaling benchmark of ESyS-Particle for elastic wave propagation simulations. [in:] *Proceedings 2010 Sixth IEEE International Conference on E-Science: EScience 2010: 7–10 December 2010, Brisbane, Queensland, Australia*, IEEE, Piscataway, 277–283. https://doi.org/10.1109/eScience.2010.40.

Zhou X., Ma W., Yang L., Bi J. & Cheng H., 2018. Experimental study of stick-slip failure processes and effect of physical properties on stick-slip behavior. *Journal of Geophysical Research: Solid Earth*, 123(1), 653–673. https://doi.org/10.1002/2017JB014515.

Zhou X., He Y. & Shou Y., 2021. Experimental investigation of the effects of loading rate, contact roughness, and normal stress on the stick-slip behavior of faults. *Tectonophysics*, 816, 229027. https://doi.org/10.1016/j.tecto.2021.229027.

Zhou X.-P., Li C.-Q. & Tu D., 2023. Evolution of the full-interface shear stress of a fault during frictional instability. *Tribology International*, 182, 108350. https://doi.org/10.1016/j.triboint.2023.108350.