J Grid Computing (2021) 19: 26
https://doi.org/10.1007/s10723-021-09566-y

A Pilot Applied Physics Grid Computing Infrastructure for Developing Applications Predicting the Qualities of Industrial Coatings

Vasiliy Chitanov · Michael Krieger · Paul Heinzlreiter · Lilyana Kolaklieva · Roumen Kakanakov

Received: 19 April 2020 / Accepted: 8 June 2021 / Published online: 1 July 2021

Abstract A pilot grid computing test infrastructure has been created between the Central Laboratory of Applied Physics (CLAP), part of Bulgarian Academy of Sciences, and RISC Software GmbH, a subsidiary research company of Johannes Kepler University in Austria. The infrastructure is used by specialists in applied physics for training in the application of grid technologies. A specific application has been developed for predicting the qualities of industrial coatings researched at CLAP. "SKLEARN "Python library, including six predictive models, is used by the application. The nanohardness of different coatings is predicted and compared with data from actual measurements for the validation of the modeling results. The results show that of all methods, Gaussian Process Regression (GPR) gives the closest predictions for most of the research coatings.

Keywords Grid computing · Predictive models · Nanohardness · Coatings

V. Chitanov (✉) · L. Kolaklieva · R. Kakanakov
Central Laboratory of Applied Physics, Bulgarian Academy of Sciences, 69 Bld. Sankt Petersburg, 4000 Plovdiv, Bulgaria
e-mail: vchitanov@gmail.com

L. Kolaklieva
e-mail: ohmic@mbox.digsys.bg

R. Kakanakov
e-mail: ipfban@mbox.digsys.bg

M. Krieger
Dynatrace Austria GmbH, Freistädter Straße 313, 4040 Linz, Austria
e-mail: mk@druide.at

M. Krieger · P. Heinzlreiter
RISC Software GmbH, Johannes Kepler University, Softwarepark 35 (IT-Center) A-4232, Hagenberg, Austria

P. Heinzlreiter
e-mail: Paul.Heinzlreiter@risc-software.at

1 Introduction

Scientific research is increasingly applying computing resources, hence the higher requirements for infrastructure facilities. Sciences such as physics, biology and chemistry use complex software applications executed on High Performance Computing (HPC) infrastructures as well as clouds and distributed computing [1]. In this context, grid computing [2] is applied by researchers working with distributed data to access computer resources outside the control of their organizations [3]. With the improvement and enlargement of computing infrastructures, research in grid computing ranges from sharing resources, standardization of data access and data storage [4], to the development of grid applications for secure access, reliability, big data analysis and incorporating artificial intelligence [5]. While the basic technical issues in grid systems have been addressed, there is still the need for improving the usability of grid systems as well as the training of specialists in accessing grid infrastructure. The main challenges in the development of grids are therefore the training of administrators and users, standardization of the interfaces and the creation of user-friendly applications for specific research tasks [6]. Nowadays, cloud computing is gaining priority in this technology, where resources are provisioned on demand from a cloud service provider and once the assigned task is completed the resources are terminated. In grid computing, when the user logs into a network, any unused resources are assigned to the execution of a particular job, speeding up task execution and eliminating failures if any burn-out occurs, distributing the risk between the different nodes.

The setup of an independent grid test center opens wide possibilities for the education of new specialists in administration and organization of grid data and structures. Such organization opens up perspectives for different education plans and experiments with different software jobs. The organization of the data could follow the specific needs and expectations of the client.

This paper presents a new grid application in the area of applied physics for predicting the quality of superhard coatings as well as improvements in grid usability.

1.1 Grid Applications for Applied Physics

In physical sciences, grid computing is applied to tasks implementing complex theories. The pioneer project requiring a large amount of computing resources and thus advancing grid technologies was given by the CERN infrastructure for High Energy Physics experiments [7]. The project concentrates expensive equipment, used by different laboratories across the world. The application of grid technologies supports hardware sharing, virtualization and execution of tasks by different organizations. The worldwide Large Hadron Collider Computing Grid (WLCG) project organizes more than 170 computing centers in 42 countries. In 2018, the mission of the project was to store, distribute and analyze about 50 Petabytes of data (http://wlcg.web.cern.ch/). The worldwide LHC computing grid is partnered with EGI (European Grid Infrastructure) [8], OSG (Open Science Grid) [9], and NeIC (Nordic e-Infrastructure Collaboration). Other applications in physics using grid calculations are, for example, plasma and fusion research, gravitational waves, and modeling black holes with one of the most complex equations in physics [10] In all these applications, the data flow is increasing rapidly and additional data analysis algorithms have to be implemented.

1.2 Grid Computing Principles

There are different definitions for grid computing [11]. The goal of grids is to guarantee secure remote access to decentralized systems and to support the acquisition of new research knowledge. Grid computing is characterized by the utilization of computing resources across organizational boundaries. These resources could be distributed computers, big calculation infrastructures, middleware and software applications. The main features of grid computing are resource management for the distribution of available resources between the processes and users, security which has to guarantee secure and authorized use of resources, and data management – the movement and processing of data and job management – for initiation and execution of grid jobs. The organization of grid architectures with all these functions requires development of middleware protocols, interfaces and standards. There are different middleware implementations among which Globus toolkit is a widely used open platform [12]. Globus middleware is characterized by parallel and partial data and file transfer and enhanced Grid Security Infrastructure (GSI). The main advantage of Globus toolkit is that it enables metacomputing calculations. The standardized resource management, communications, security and remote data access make it applicable for desktop supercomputing, smart instruments data analysis, collaborative and distributive calculations. There are examples such as Distributed Infrastructure with Remote Agent Control[1] (DIRAC) [13, 14] projects developed by CERN for coordination of resources for large physics simulations. The authors argue that this system is less complex and more scalable, but DIRAC implements its own data management system and is particularly applicable to large systems. Other systems like NorduGrid[2] [15, 16] step on GLOBUS and employ user grid interface to submit job requests and obtain system information. There are different middlewares, such as EMI, gLite, Gridway, OpenLava among others, developed for the specific requirements of a project. Still, Globus toolkit is the most generalized and accepted by the community and it was implemented in the development of the educational grid center. The security part of Globus middleware includes authentication of

[1] http://diracgrid.org
[2] http://www.nordugrid.org/arc

users, authorization of access, and secure transfer of data between grid nodes. The resource management part is responsible for the remote job scheduling and execution. The data management procedures provide secure data sharing and transfer. Therefore, the toolkit allows the building of distributed applications.

1.3 Main Contribution of the Study

The development of computational physics in parallel to experimental work for small but complex pilot installations and infrastructures has increased the demand for specialists trained in accessing and using HPC infrastructure. The quantity of sensors and characterization data is also constantly growing and new data science specialists are therefore being required. The development of educational infrastructures for application of grid technologies and data analysis methods will be indispensable for specialists in different professional fields. In this study, a test grid infrastructure for specialists in applied physics is presented. A grid application, including models predicting the quality of hard and super-hard coatings, was developed and adapted in accordance with the test infrastructure and the requirements of the coating researchers.

2 Related Work – What we Rely on

Scientific infrastructures working with big experimental data rely on three basic components: HPC and cloud systems, grid computing and middleware, and data analysis.

2.1 HPC – Cloud and Grid Infrastructures

Within research, HPC is applied to computing research problems requiring modeling and data analysis tasks in parallel. From a historical perspective, in the 1990s cluster computing was commonly applied in HPC calculations, especially in less complex but still time-consuming calculations due to the advancements in hardware calculation speed, network connections and Linux opportunities [1]. The parallel execution of different jobs on distributed heterogeneous clusters motivated the use of grid technologies. The administration of user access to distributed infrastructures in combination with reduced costs supported the development of grid technologies. With the intensification of research requiring data access and implementation of virtualization, cloud computing technologies appeared as a new option for remote big data calculations. The main advantage of clouds is their flexibility in the use of computing resources.

2.2 Grid Middleware

The Globus toolkit provides user authentication and authorization based on the Grid Security Infrastructure (GSI) [17]. The GSI includes GSI-SSH commands and procedures for authentication with a proxy certificate. The certificate is created and used when the grid system is accessed. Certificates are issued for users and computing nodes. A Certificate Authority (CA) can be created for issuing certificates, if no certificates from official CAs are available. Another function of the Globus toolkit is to identify computing resources and their status in the grid, which is made available to a resource broker through the Grid Information Service [18]. Using the Grid Information Service, a job scheduler is being used to determine an available resource. Globus itself does not provide such schedulers, but job managers exist which can interact with specially developed job schedulers like PBS, when jobs need to be distributed across computing clusters. Secure data management in Globus is handled by the Grid Access to Secondary Storage (GASS). The most commonly used element is probably GridFTP, which was developed based on the standard FTP protocol but includes additional authentication and authorization procedures. Once a grid certificate is obtained, users can distribute data without the use of login commands. The Grid Resource Allocation Manager (GRAM) is applied to initiate job execution on a particular resource. It can check the status of the job and be used to retrieve results.

2.3 Data Analysis

Data analysis (DA) is applied to data from business and scientific research to perform generalization and verification of data and predictive models, and to support specialists in their work. The Python module "SKLEARN "is appropriately applicable because of the incorporated machine learning algorithms for solving supervised and unsupervised problems. It is preferred by non-specialists in machine learning because of the fast language prototyping and easy use of library documentation [19]. It is distributed under the BSD procedures and is applicable to free OS, which are mainly used in grid computing. With “SKLEARN “it is possible to develop machine learning algorithms for classification, regression, clustering and model validation tasks.

3 Description of the Test Infrastructure

The grid infrastructure presented in Fig. 1 is used to connect two research organizations located in Bulgaria and Austria. One part of the infrastructure is located at the Central Laboratory of Applied Physics, Bulgarian Academy of Sciences (CLAP-BAS), the second part at RISC Software GmbH (RISC), a subsidiary of Johannes Kepler University Linz. At CLAP-BAS, a physical machine with Intel Core i7–3770 3.4GHz and 8GB RAM is used for hosting a local Grid Certificate Authority as well as a grid endpoint in multiple virtual machines. At RISC, a virtual machine of the organization’s internal cloud infrastructure is used as grid endpoint, providing access to compute resources at RISC as well as a GPGPU (NVIDIA Quadro K2000). The host systems at RISC used for the presented research have 2 × 8 Core Intel Xeon CPUs and 128 GB of RAM and run Ubuntu 16.04 LTS as operating systems. I/O is limited by the underlying file system, which is built on top of a distributed Ceph file system spanning multiple machines.

As grid middleware, a standard installation of Globus toolkit 6[3] [20] is used, which serves three different purposes: to organize the job execution, to provide secure access to the infrastructure with a simple interface for access and management of the different nodes, and to enable fast data transfers between all resources participating in this distributed grid infrastructure. GRAM [21] is used for remote job submission and execution. For secure access, each organization relies on a certificate authority. At CLAP-BAS, a simple certificate authority (CA) [22] provides host and user certificates for authentication, authorization, and delegation of processes, whereas RISC uses official Austrian grid certificates for hosts and users. With these certificates users are able to use GSI-SSH for connecting to available grid resources. For secure and efficient file transfer, all endpoints allow the use of GridFTP [23].

During installation of such test grid infrastructure, one of the bottlenecks is the network settings. The different nodes in the network could be accessed through their host name and IP address. A second machine could be added to the physical machine, on which user grid applications and jobs to be executed on the grid test infrastructure.

The Globus toolkit was used for the application described in this paper mainly because of technology transfer reasons. Besides enabling access to a larger set of resources for the application, one of the main goals of the collaborative project between the institutions involved was to strengthen the technological grid knowledge of the partners at CLAP. To achieve this goal, Globus toolkit was picked as grid technology for the project as opposed to a readily available European infrastructure. Consequently, the application partners at CLAP were forced on a steeper technological learning curve, while on the other hand increasing their technological preparedness for the use of grid infrastructures.

The extensions of the software in the direction of grid-awareness as well as the training of CLAP staff in grid technologies can provide a good basis for moving the application to European grid research infrastructures if the need for more computational resources should arises.

The focus of the collaborative project on grid know-how transfer also did rule out the use of cloud infrastructures as an alternative source of computational resources. In addition, the use of commercial public clouds was not considered for financial reasons.

4 Workflow

The process data are organized in data text .csv files. The files are converted to data frame objects applied with the Python libraries in the developed application. An application for predicting the mechanical properties of hard and super-hard coatings was developed under Python, including the “SKLEARN “library and six statistical methods. The execution of the application code could be done in two ways using the web interface through Jupyter (http://jupyter.org/) or by the grid infrastructure as Python script. The Jupyter application was installed on the grid virtual server machine and on the RISC machine. With Jupyter, the Python code can be developed faster and the results can be visualized very easily. The application is split into three executing code parts. The workflow of the code execution is presented in Fig. 2.

[3] http://toolkit.globus.org/toolkit

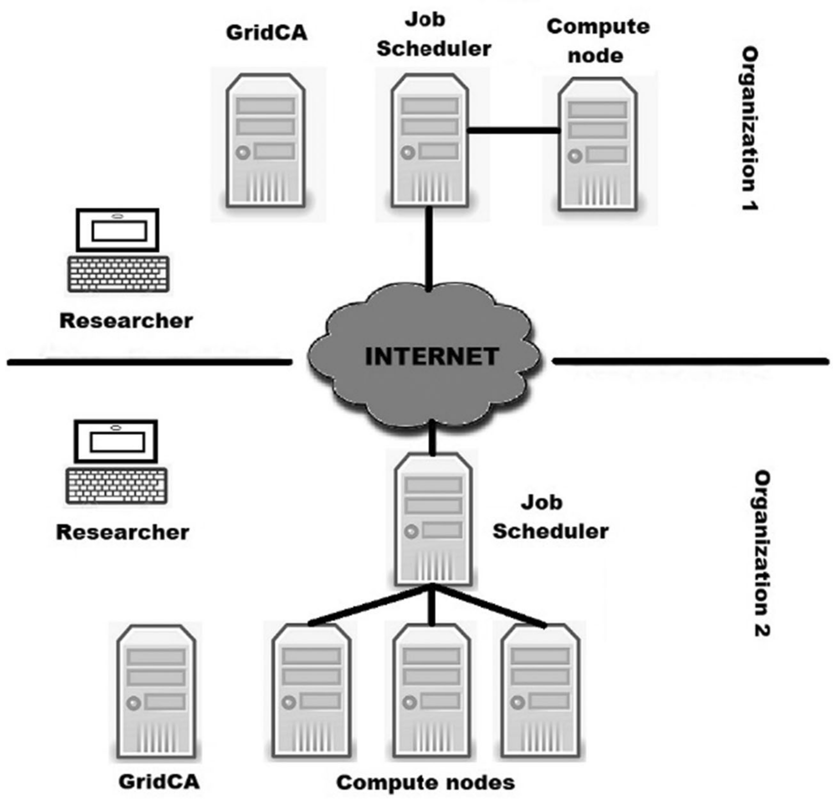

Fig. 1 Architecture of test infrastructure

The first part of the code performs visualization of the process data. The graphics output of the software shows the dependency of the predicted parameter on the different inputs. In this way, the user of the application can obtain a preliminary impression of the dependencies of the actual input parameters of the process and their distribution in the modeling space. The next Fig. 3 shows the graphical output of the Jupyter code. The second code part is organized as Python script and is executed under the grid infrastructure with the Globus grid commands. The script code includes the building of the predictive models using the defined regression methods developed based on the Python "SKLEARN "library (http://scikit-learn.org). The first method applied is "Linear Regression" (LR). This method fits a linear model minimizing the sum of squares between the observed research datasets and the predictions from the linear approximation. The second method is "KNeighbours Regressor" (KNR). It uses a Neighbors-based classification, which is a type of instance-based learning. It does not attempt to construct a general internal model, but simply stores instances of the training data. The classification is made from chosen nearest neighbors of each point. A checking point is assigned to the data set, which has the most representatives within the nearest neighbors of the calculating point. The third method, "GaussianProcessRegressor" (GPR), is based on Gaussian Processes (GP) applied for regression purposes.

GP is a generic supervised learning method. It performs interpolation of the observations and the predictions are probabilistic and also adaptive fitting is possible. Different kernels can be applied. The next method, "SGDRegressor" (SGDR), uses a linear model. SGD means Stochastic Gradient Descent. The gradient of loss is estimated for each sample data set and the model is updated with a decreasing strength schedule. The next library, "SVM.SVR", applies support vector machines (SVM), a set of supervised learning methods. The method uses a subset of training points called support vectors in the decision function. The method is memory efficient and can also use different kernel functions. The last method applied is "GradientBoostingRegressor" (GBR). This method supports different loss functions for regression, which can be specified in the method. The default regression function used is the least squares.

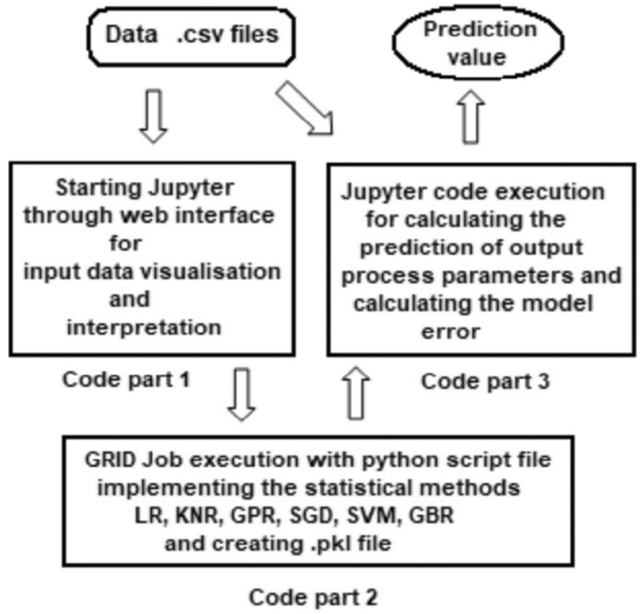

Fig. 2 Workflow of the application

The input .csv data files with the process data are divided into two types: for modeling and for validation. For each modeling step, the testing inputs and outputs are defined. Then, a fitting and training process with the modeling data files is conducted for each implemented statistical method. Finally, the created and educated predictive models are saved as Python object .pkl files.

The second code, involving the predictive model creation, is schedule and job executed on the grid server as Python script. The user first initiates a proxy

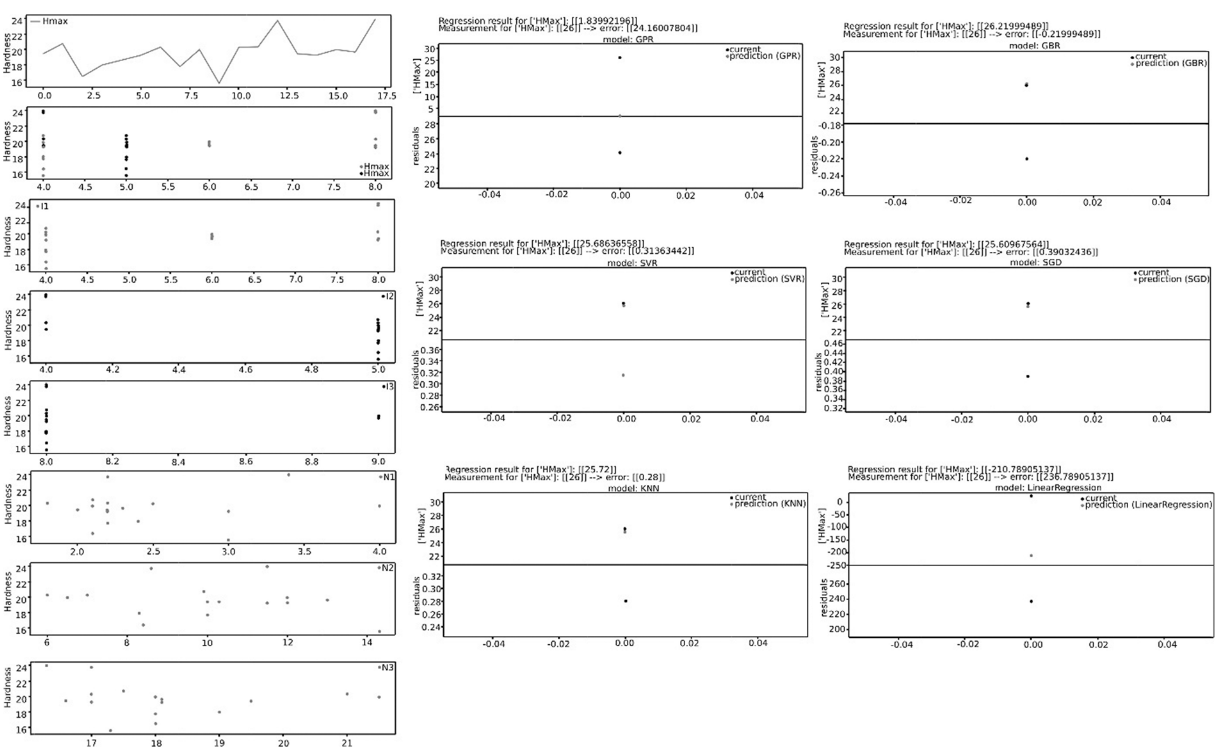

Fig. 3 Jupyter graphical output

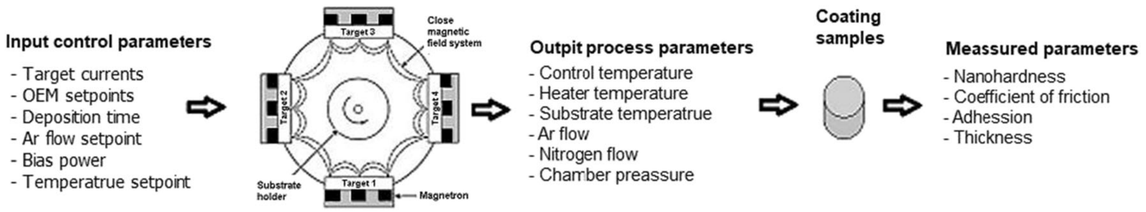

Fig. 4 Input and output process parameters

certificate with the command grid-proxy-init. After that, with the globus-job-submit and globus-job-status commands, the Python script file is started and the job execution status is checked. After the job is completed, the standard output could be retrieved with the command globus-job-get-output. The created files could be retrieved over GridFTP with the command globus-url-copy. The usage of the grid infrastructure allows starting the process of creating a new model independently of the execution of the other two code modules.

The third code part of the application is visualization of results. It is also executed under Jupyter. It uses the .pkl file models created by the second code part for predicting the process results or for validation of their accuracy. The predicted results can be plotted and compared to actual process outputs. The developed application code, executed under Jupyter, is started through web-interface and the visualization of the results is carried out under the control of the user. The “MATPLOTLIB “library with the “PYPLOT “function is used to plot the modeling results. Finally, the calculation of the prediction and the results for the model error are printed.

The execution of the code 1 and 3 parts under Jupiter allows faster changes in the code structure and variables, and immediate visualization of the graphical results for validating the code changes.

5 Process Data Description

The process data used were obtained from equipment situated at the Central Laboratory of Applied Physics, Bulgarian Academy of Sciences. Hi-tech equipment for unbalanced magnetron sputtering (UMS) [24] UDP 850–4, Teer Coatings Ltd., England, was used for deposition of different hard coatings. The UDP850/4 chamber diameter was 700 mm, the chamber height was 1000 mm, and the target size was 725 × 175 mm (http://www.teercoatings.co.uk). Reactive N_2 gas was controlled by an optical emission monitoring (OEM) system. A configuration of Ti, Cr or Al targets was used to obtain coatings with different composition.

The coatings were deposited on HSS substrates. Several main technological parameters determine the coating structure and composition during the deposition process. Once

Table 1 Data frame object for a TiCrN coating

	SAMPLE	DATA	I1	I2	I3	N1	N2	N3	Tsub	Tset	T1
0	#194–016	1	5	5	7	0.5	6.5	8.7	153	200	189
1	#199–016	1	5	5	7	0.5	10.5	12.3	201	200	272
2	#203–016	1	5	5	7	0.8	9.4	12.3	200	165	274
3	#01–017	1	5	5	7	2.5	9.3	13.6	198	165	284
4	#03–017	1	5	5	7	0.8	13.0	13.2	198	165	284
5	#11–017	1	5	5	7	2.9	11.0	18.2	133	0	278
Th	time	D	H10	Hmax	H20	E10	Emax	E20	Mu	Ad1	Ad2
320	90	0.00	0	21	20.0	0	303	248	0.377	8.50	3
312	75	0.00	0	19	20.0	0	312	314	0.457	2.30	2
315	60	0.00	0	20	18.5	0	275	265	0.282	11.00	4
315	90	1.40	0	28	23.0	0	381	313	0.350	4.15	3
318	90	1.10	0	19	19.0	0	261	261	0.180	11.40	3
95	90	1.14	0	21	0.0	0	321	0	0.144	25.40	5

Table 2 Description of data frame parameters

Column name	Column description
Sample	Sample number
data	Option for eliminating concrete samples from the model calculations
I1, I2, I3	Currents applied on the targets, [A]
N1, N2, N3	Nitrogen flow during the gradient increase and main coating deposition step, [sccm]
Tsub	Temperature on the substrate, [°C]
Tset	Temperature setpoint, [°C]
T1	Control temperature, [°C]
Th	Temperature of the heater, [°C]
time	Coating deposition time of the main coating step, [min]
D	Measured thickness of the coating, [μm]
H10	Nanohardness following 10% rule, [GPa]
Hmax	Maximum measured nanohardness, [GPa]
H20	Nanohardness measured at 20 [mN] load, [GPa]
E10, Emax, E20	Modulus of elasticity correlated with measured hardness, [GPa]
Mu	Coefficient of friction, nondimensional parameter between 0 and 1
Ad	Critical load at which the coating failed during the scratch test, [N]
Ad2	Coating adhesion, 1- very bad, 2 - bad, 3 - good, 4 - very good, 5 - excellent.

loaded in the chamber, the samples were first heated and cleaned in Ar plasma at a bias voltage. The cleaning time, bias voltage, and Ar flow have a critical role for the cleaned surface quality and consequently for the adhesion of the coating to the substrate.

Coating structuring starts with an interface metal layer. In the UMS process, power is applied to the magnetron [25]. The magnetron power and bias voltages are basic technological parameters in PVD processes that determine the target material deposition and coating composition.

The interface layer is followed by a gradient layer. It is deposited at an increasing flow of reactive N_2 gas. In reactive sputtering, N_2 reacts with the target surface as well as the sputter deposited metal atoms, thus forming a compound thin film on the substrate as well as on the chamber walls [26]. The reactive process is controlled by the N_2 flow, which is a parameter used for producing a coating with predefined stoichiometry. The deposition time of the different layers determines their thickness. The proper set of process parameters, such as cathode and bias power, Ar and nitrogen flows, working temperatures, process time etc. determine the composition and morphology of the coatings, consequently, their mechanical properties [27]. The research input and output process parameters and coating mechanical properties are shown in Fig. 4.

The mechanical parameters of the coatings were measured with Compact platform CPX (MHT/NHT) CSM Instruments, Anton Paar (Austria) situated at CLAP (www.anton-paar.com). The equipment software automatically calculates the nanohardness and Elastic Module depending on the displacement depth. The scratch test evaluates the coefficient of friction and the critical loads at which the coating fails. The thickness of the coatings was measured by the calotesting technique.

Six different types of hard coatings were investigated: TiN, CrN, TiCrN, CrTiN, CrTiAlN and TiCrAlN. For each coating a table with the researched technological samples is provided. For each sample, the technological parameters applied during its deposition are determined: nitrogen flow, current, temperature and the relevant mechanical parameters like nanohardness and adhesion. Different number of samples are included in the tables. The smallest number of samples is 4, the maximum number is 36. Technological experiments are too expensive and it is difficult to collect a large amount of data. The data are organized as .txt files and subsequently converted to .csv files. The Python software transform them and stores the data as data frame objects. Seven files were used in the project. The smallest size is 170 bytes and the biggest is 3.2 kb.

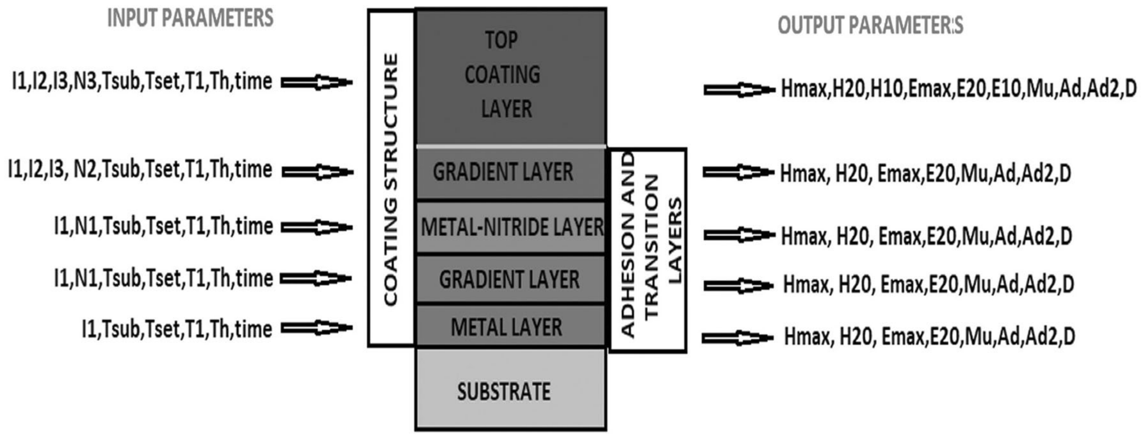

Fig. 5 Process inputs and outputs according to deposited coating structure

The subset of data is accessed in the software using the commands accepted in Python for data frame objects. The data frame objects are necessary for the calculation of the predictive model results. In Table 1, the data frame object for TiCrN coating is presented.

The data frame contains information about the number of input data, sample number, and an option for excluding a sample according to the available measurement data. Values for eleven process parameters and for ten output parameters qualifying the coating properties are included. In this way, each specialist can use different combinations of input and output parameters depending on the specific application developed. The description of the parameters used is given in Table 2.

In Fig. 5, the relations between the process inputs and outputs for each layer of the deposited coating structure are shown. The values of the output parameters are also influenced by the mechanical parameters of the substrate [28]. In this research, the influence of substrate is excluded because all the samples are deposited on the same type of HSS substrate. For the evaluation of the applied predictive models, the following input parameters were used: magnetron currents I1, I2, I3, nitrogen flows N2, N3, substrate temperature Tsub, and deposition time. The modeled and predicted parameter is the nanohardness Hmax as one of the most important coating mechanical parameters [29]. The model input and outputs and the corresponding modeling output results are summarized in Fig. 6.

6 Results

Results for six different types of coatings are presented. For each coating group, a different number of samples is used for education of the predictive models. The first coating type, TiN, is divided in TiN with 36 samples and TiN2 with 4 samples because of the different target configurations applied during the deposition process. The other standard coating researched with 6 samples is CrN. The nanohardness of two triple coatings TiCrN and CrTiN with 6 and 14 samples and two four-layer coatings TiCrAlN and CrTiAlN with 6 and 18 samples, respectively, are also modeled. Figure 7 and Table 3 present the results for the predicted nanohardness of the

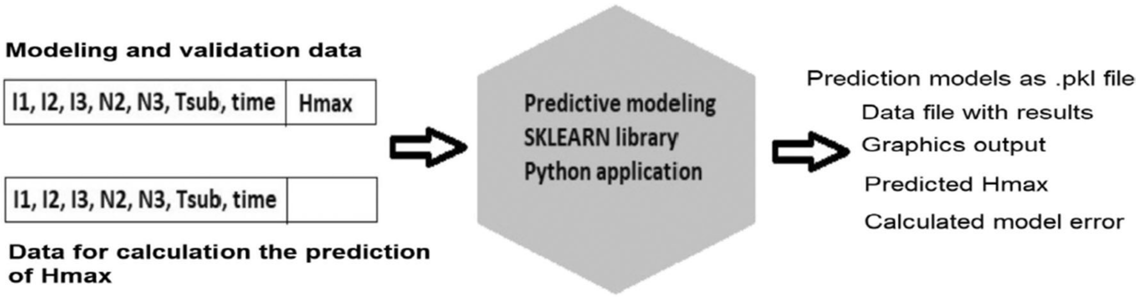

Fig. 6 Coating modeling inputs and output

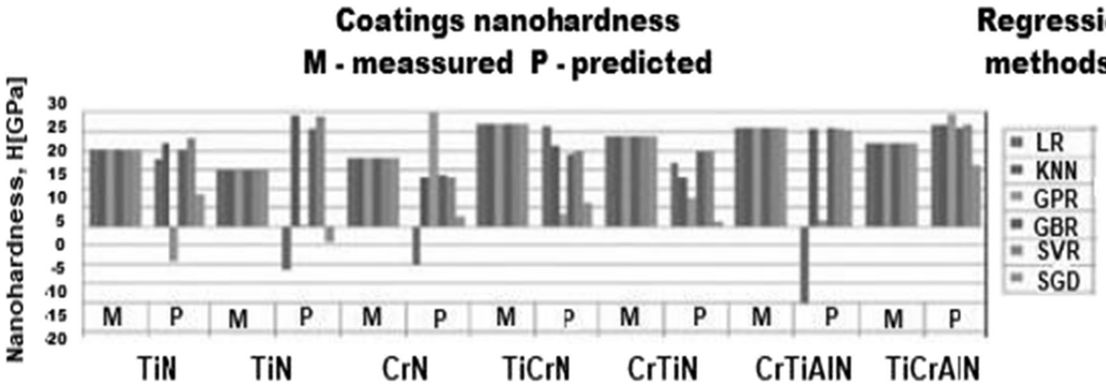

Fig. 7 Results for predicting the nanohardness of different hard coatings

regression methods used and the corresponding actually measured coating value.

The results reveal that the higher number of samples used in the creation of the predictive model leads to more accurate prediction results. The best results are for the TiN and CrTiAlN coatings, presented with 36 and 18 samples. From the regression methods, the GBR gave the best prediction among the different coating groups. The SVR method gives slightly better prediction then GBR for the CrTiN coating, but the error for GBR is still acceptable. Only for the TiCrN coating, GBR shows a bigger modelling error and the LR calculates the smallest error. For this coating, however the samples are only six and a general conclusion is difficult to be made. In Table 4, the calculated model errors for all predictions are presented. The smallest error of 0.17 is for the TiN coating predicted with GBR. Of the best predictions for all coatings, TiN2 has the maximum value of 10.86 but for this prediction only data from four samples were used. It has to be mentioned that in the actual measurements of the characterization equipment, differences between 1 and 5 GPa nanohardness are normal for one measured sample because of the specifics of the applied method. In most of the results, the prediction errors are under 5 GPa. The correctness of the model depends on the actual measurement results used. For this research up to 3 measurements per indentation load were made to test and validate the chosen statistical methods. The results show that of all research methods, GBR could be used for predicting the nanohardness of hard and superhard coatings.

7 Conclusion

A pilot test grid infrastructure was built by Bulgarian and Austrian research teams. The Globus toolkit was

Table 3 Predicted values of nanohardness

Regr. Meth.	NANOHARDNESS, H [GPa]													
	M	P	M	P	M	P	M	P	M	P	M	P	M	P
LR	20.4	17.8	15	−11.55	18.1	−10.06	27	26.44	24	16.68	26	−210.6	22	26.64
KNN	20.4	21.9	15	29.44	18.1	12.86	27	21.4	24	12.86	26	25.72	22	26.44
GPR	20.4	−9.3	15	0.44	18.1	34.13	27	3.37	24	7.6	26	1.84	22	29.64
GBR	20.4	20.2	15	25.86	18.1	13.57	27	19	24	19.77	26	26.22	22	25.96
SVR	20.4	23.3	15	28.96	18.1	13.06	27	20.09	24	19.88	26	25.69	22	26.83
SGD	20.4	8.4	15	−3.93	18.1	2.52	27	6.27	24	1.51	26	25.34	22	16.28
Coat. type	TiN		TiN		CrN		TiCrN		CrTiN		CrTiAlN		TiCrAlN	
Samples number	36		4		6		6		14		18		6	

Table 4 Model error values

Coating	TiN	TiN2	CrN	TiCrN	CrTiN	CrTiAlN	TiCrAlN
LR	2.55	26.55	28.16	0.56	7.32	236.79	−7.64
KNN	−1.48	−14.44	5.24	5.6	11.14	0.28	−4.84
GPR	20.4	14.6	−16.03	26.99	24	24.16	−7.64
GBR	0.17	−10.86	4.53	7.99	4.22	−0.22	−3.96
SVR	−2.90	−13.96	5.04	6.91	4.12	0.32	−4.85
SGD	11.95	18.93	15.58	20.73	22.49	0.66	5.72

used for organizing the process of starting and executing grid jobs. A Python application for predicting the nanohardness of real industrial coatings was developed. Six statistical methods were applied to model the mechanical properties of seven groups of industrial coatings. More measurement data are necessary for obtaining an accurate statistical model. The GBR method showed the lowest model error. The test grid center will be developed with the implementation of other statistical algorithms and new approaches like deep learning for extracting new knowledge. As regards the global development of cloud and new grid worldwide infrastructures, the inclusion of Large national and European grid initiatives like EGI will be considered.

Acknowledgments This Work Has Been Supported by the Bilateral Program for Scientific and Technological Cooperation between the Bulgarian National Science Fund and Austrian Agency for International Cooperation in Education and Research under Project ДНТС/Austria 01/5 of 25.06.2013 "Developing pilot grid infrastructure supporting research in the area of applied physics"

Data Availability All data generated or analysed during this study are included in this published article.

References

1. Netto, M.A., Calheiros, R. N., Rodriques, E. R., Cunha, R. F., Buyya, R., HPC cloud for scientific and business applications: taxonomy, vision and research challenges. ACM Comput. Surv., **51**(1), A. 8., (2018), https://dl.acm.org/citation.cfm?doid=3177787.3150224
2. Foster, I., Kesselman, C.: The Grid 2. Morgan Kaufmann Publishers, Blueprint for a new Computing Infrastructure (2003) https://dl.acm.org/citation.cfm?id=996313
3. Foster, I., Kesselman, C., Tuecke, S.: The anatomy of the grid: enabling scalable virtual organizations. International Journal of High Performance Computing Applications. **15**(3), 200–222 (2001). https://doi.org/10.1177/109434200101500302
4. Qureshi, M., Dehnavi, M., Min-Allah, N., Qureshi, M., Hussain, H., Rentifis, I., Tziritas, N., Loukopoulos, T., Khan, S., Xu, C., Zomaya, Y.: Survey on grid resource allocation mechanisms. J Grid Computing. **12**, 399–441 (2014)
5. Vaniachine, A., Golubkov, D., Karpenko, D.: Reliability engineering analysis of ATLAS data reprocessing campaigns, 20th international conference on computing in high energy and Nuclear Physics (CHEP2013). J. Phys. Conf. Ser. **513**, 032101 (2014). https://doi.org/10.1088/17426596/513/3/032101
6. Berman, F., Fox, G., Hey, T.: Grid computing: making the global infrastructure a reality. John Wiley & Sons, Ltd (2003) https://onlinelibrary.wiley.com/doi/book/10.1002/0470867167
7. Stockinger, H., Grid computing in physics and life sciences, Proceedings of the 18th International Conference on Computer Information and Systems Science and Engineering (CISE'06), (2006) https://www.researchgate.net/publication/249824415_Grid_Computing_in_Physics_and_Life_Sciences
8. David, M., Borges, G., Gomes, J., Pina, J., Plasencia, I., Castillo, E., Diaz, I., Fernandez, C., Freire, E., Simon, A., Koumantaros, K., Dreschner, M., Ferrari, T., Solagna, P.: Validation of grid middleware for the European grid infrastructure. Journal of Grid Computing. **12**, 543–558 (2014)
9. Altunay, M., Avery, P., Blackburn, K., Bockelman, B., Ernst, M., Fraser, D., Quick, R., Gardner, R., Goasquen, S., Levshina, T., Livny, M., McGee, J., Olson, D., Pordes, R., Potekhin, M., Rana, A., Roy, A., Sehgal, C., Sfiliqoi, I., Wuerthwein, F.: The open science grid executive board, a science driven production cyberinfrastructure – the Open Science grid. Journal of Grid Computing. **9**, 201–218 (2011)
10. McCurdy, B., Ceperly, D., Koonin, S., Ostriker, J., Reed, D., Wilkins, N. J., Woodward, P. Computation as a tool for discovery in physics, a report to the national science foundation by the steering Committee on Computational Physics, Arlington,Virginia(2012) https://www.nsf.gov/pubs/2002/nsf02176/nsf02176_1.pdf
11. Kovalenko, V.N., Koryagin, D.A.: The grid: analysis of basic principles and ways of application. Program. Comput. Softw. **35**(1), 1834 (2009). https://doi.org/10.1134/S0361768809010046
12. Jacob, B., Ferreira, L., Bieberstein, N., Gilzean, C., Girard, J.-Y., Strachowski, R., Yu, S.: Enabling application of grid computing with Globus. IBM Redbooks. (2003)
13. Tsaregorodtsev, A., Garonne, V., Stokes-Rees, I.: DIRAC: a scalable lightweight architecture for high throughput

computing. Fifth IEEE/ACM International Workshop on Grid Computing. (2004). https://doi.org/10.1109/GRID.2004.22
14. Diaz, R., Ramo, A., Aqüero, A., Fifield, T., Sevior, M.: Belle-DIRAC setup for using Amazon elastic compute cloud. Journal of Grid Computing. **9**, 65–79 (2011)
15. Ellert, M.: At al, advanced resource connector middleware for lightweight computational grids. Futur. Gener. Comput. Syst. **23**, 219–240 (2007). https://doi.org/10.1016/j.future.2006.05.008
16. Krašovec, B., Filipčič, A.: Enhancing the grid with cloud computing. Journal of Grid Computing. **17**, 119–135 (2019)
17. Jacob, B.: Grid computing what are the key components? Contents, High-level overview of grid computing. Taking advantage of grid computing for application enablement. Introductorylecture (2003) http://www.lsi.usp.br/~grid/artigos/ibm/groverview.pdf
18. Czajkowski, K., Fitzgerald, S., Foster, I., Kesselman, C., Grid information services for distributed resource sharing, Proc. of the 10th IEEE International Symposium on High-Performance Distributed Computing (2001) https://ieeexplore.ieee.org/document/945188/
19. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grise, L.O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, É.: SciKit-learn: Machine learning in Python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)
20. Foster, I.: Globus toolkit version 4: software for service-oriented systems, international conference on network and parallel computing. Springer LNCS. **3779**, 2–13 (2006)
21. Czajkowski K., Foster I., Karonis N., Kesselman C., Martin S., Smith W., Tuecke S., A resource management architecture for metacomputing systems, Proc. IPPS/SPDP '98, Workshop on Job Scheduling Strategies for Parallel Processing, pp. 62–82 (1998)
22. Foster I., Kesselman C., Tsudik G., Tuecke S., A security architecture for computational grids, Proc. of the 5th ACM Conference on Computer and Communications Security, pp. 83–92 (1998)
23. Allcock, W., Bresnahan, J., Kettimuthu, R., Link, M., Dumitrescu, C., Raicu, I., Foster, I.: The Globus striped GridFTP framework and server. Proc. of Supercomputing. **2005**, (2005)
24. D. G. Teer, UK patent No. 2 258 343, USA patent No. 5 554 519, European patent No. 0 521 045
25. Rohde, S.: Unbalanced magnetron sputtering. Physics of thin films. **18**, 235–288 (1994)
26. Liljeholm, L. Reactive Sputter deposition of functional thin films, Ph.D. thesis, Faculty of science and technology, Uppsala University, (2012)
27. Mohammadtaheri, M., Yang, Q., Li, Y., Corona-Gomez, The effect of deposition parameters on the structure and mechanical properties of chromium oxide coatings deposited by reactive magnetron sputtering, Coatings, **8** (3), (2018) https://doi.org/10.3390/coatings8030111
28. Ding, X., Zeng, X., Liu, Y., Wei, J., Holiday, P.: Influence of substrate hardness on the properties of PVD hard coatings. Synthesis and Reactivity in Inorganic, Metal-Organic, and Nano-Metal Chemistry. **38**(2), 156–161 (2008). https://doi.org/10.1080/15533170801926028
29. Chen, S., Liu, L., Wang, T.: Investigation of the mechanical properties of thin films by nanoindentation, considering the effects of thickness and different coating–substrate combinations. Surf. Coat. Technol. **191**, 25–32 (2005)