CLOUD BASED LARGE SCALE MULTIDIMENSIONAL CUBIC SPLINE INTERPOLATION FOR WATER QUALITY ESTIMATION

Sorin N. CIOLOFAN[1], Mariana MOCANU[2], Valentin CRISTEA[3]

Reliable and real time estimation of water pollution is critical in water management information systems. Sometimes professional software (such as DHI's MIKE11) is not available online to run simulations for various pollution scenarios. A system able to accurately assess the concentration of the pollutant at any points along a river in respect to a given pollution scenario was implemented. This system reuses historical offline data resulted from previous executions of MIKE11 software. The pollution scenario consists of a set of user specified input parameters (chainage, pollutant concentration, discharged volume, type of pollutant, etc.). Numerical methods are used to address the problem of multivariate interpolation and compute the result. The validation of the system was done using data from a real scenario on Dîmboviţa river, the obtained results being estimated with a mean percentage error less than 1.3%. To efficiently cope with millions of records, the computing intensive application was deployed on Jelastic Cloud in order to take the advantage of on demand elastic RAM and CPU resources.

Keywords: Big Data, computing intensive application, water quality estimation, multivariate interpolation, Cloud

1. Introduction

A critical feature of decision support systems for water quality is the ability to run simulations based on user supplied pollution scenarios and to produce estimations of the magnitude of the pollution phenomena [1,2,3,4]. The execution time of the simulation must allow decision makers to emit warnings and act accordingly before the disaster takes place. During the Cyberwater project a 30-km long section of the Dîmboviţa river situated between Căţelu and Budeşti stations is studied for the assessment of accidental pollution and the development of prototype distributed decision support system [5]. Along this river there are a few economic agents (plants) and also inhabited sites. Often there are situations when various types of pollutants (such as detergents, petroleum, etc) are discharged in the river's water, putting at risk the home consumers that live near

[1] Eng., Dept.of Computer Science, University POLITEHNICA of Bucharest, Romania, e-mail: sorin.ciolofan@cs.pub.ro
[2] Prof., Dept.of Computer Science, University POLITEHNICA of Bucharest, Romania, e-mail: mariana.mocanu@cs.pub.ro
[3] Prof., Dept.of Computer Science, University POLITEHNICA of Bucharest, Romania, e-mail: valentin.cristea@cs.pub.ro

the river and use its water directly for domestic activities. MIKE11 powered by DHI uses a 1-D model of a river, in order to simulate water quality and sediment transport in rivers, flow and water level [6,7,8]. MIKE11 software is usually installed on a dedicated server, this approach having some inherent disadvantages:

- The server where MIKE11 is installed can have downtime periods. For each simulation to be ran on the MIKE11 server a license should be valid and plugged into the server
- There can be connection problems on the network between applications/clients that access the MIKE11 server.

There are critical situations when running of pollution simulations is needed immediately in order to warn the population. If the MIKE11 server is not available when it is needed this fact can have serious negative consequences. From this perspective, it is important to develop an alternative to MIKE11 that can overpass the drawbacks enumerated above. It became clear that in the design of such a black box modeling problem we must consider deploying the application on Cloud facilities because we have to face the following design requirements:

- *Big Data*, since we will have to deal with tens of millions of records, so we will need to have horizontal/vertical scaling on demand. We expect to have big RAM and CPU requirements for our computational intensive application.
- *100% Availability*, due to SLA uptime

The main contribution of this paper is to propose a solution for the problem of black box modeling of MIKE11 based on multivariate interpolation and to implement this solution on a Cloud platform.

2. Paper contents

This article is organized as follows: Section 3 presents a critical State Of the Art in the field of function interpolation and spatial interpolation and a couple of relevant examples from environmental management are given. Section 4 presents in more details the setup of MIKE11 in order to run simulations of water pollution for the segment of Dîmboviţa river situated between Budeşti and Căţelu stations. Then we show two possible alternatives to using MIKE11, and show why they are practically not feasible. In Chapter 5 we discuss the theoretical aspects of the proposed solution based on multivariate spline interpolation while on Section 6 we focus on the implementation details and Cloud deployment. Section 7 is dedicated to experimental validation of the proposed implementation. Chapter 8 presents the conclusions.

3. Related Work

Various methods for interpolation of a function data as well as for spatial data were presented previously in the literature [9]. For *spatial data*, interpolation methods estimate values of some properties at sites that were not sampled

considering the proximity of this sites where observations are known. In this category fits methods such as IDW (Inverse distance weighted), NNIDW (Natural neighbor inverse distance weighted), TREND, kriging, etc. In IDW method it is considered that closer points have a higher impact than the far away ones so they receive higher weights (does not provide good performance for peaks or mountain areas). NNIDW is a variation of the IDW which works well with clustered points. TREND uses a global polynomial interpolation to fit a given surface while Kriging adopts a statistical approach where a Gaussian process is used to model the interpolated values. In [10] the authors performed IDW and kriging methods to estimate Dissolved Oxygen, salinity and water temperature based on data acquired in 9 years of observations in the waters of Chesapeake Bay and concluded that kriging methods outperforms IDW for all parameters. To estimate the risk of groundwater nitrate pollution on Granada aquifer in S-E of Spain Chica-Olmo, Mario, et al. successfully used Indicator Kriging method to produce probability and categorical charts [11]. Since our model in MIKE11 is a 1-D model our data does not provide geographically spatial features, so the above-mentioned methods are not best suitable (these methods are applicable in environmental applications with points in bidimensional or tridimensional space). For *function data*, polynomial interpolation has the disadvantage of oscillation. Hermite polynomials also present big oscillation for large number of nodes [12]. Trigonometric interpolation, in particular Fast Fourier Transform (FFT), can be applied for functions that exhibit a periodic behavior. In our case we are dealing with large number of nodes (millions order of magnitude) and with accidental pollution so the previous methods are not suitable. The most used form is piecewise-polynomial interpolation, especially cubic spline interpolation in cases where only the function values are known and a high degree of accuracy is desired [13]. Because we are working with a function of more than one variable we will focus on multivariate spline interpolation rather on univariate interpolation (see Section 5).

4. River modeling and pollution estimation using MIKE11

MIKE 11 software computes the concentration of a pollutant in the water of a river based on the following differential equation for 1D Advection-Dispersion [14]:

$$\frac{\partial AC}{\partial t} + \frac{\partial QC}{\partial x} - \frac{\partial}{\partial x}\left(AD\frac{\partial C}{\partial x}\right) = -AKC + C_2 q \qquad (1)$$

where $C(x,t)$ is the solution and represents the concentration depending on time t and position along river, x. D is the dispersion coefficient, $A(x)$ is the cross-sectional area, K the linear decay coefficient, C_2 source/sink concentration, q the lateral inflow.

MIKE 11 solves the above equation and outputs a discrete format (tabulated values) for C(x,t), considering the following input information is given (Fig. 1).

- River profile (the topology of the river on the segment of interest)
- Concentration of the discharged pollutant (mg/l)
- Type of pollutant
- Chainage (distance measured from the reference point where the pollutant was dispensed)
- Start/Stop time of simulation
- Initial conditions as an array of elements Q[t], where for a given timestamp t, Q[t] represents the quantity [l] of pollutant that is accidentally released in the river's water.

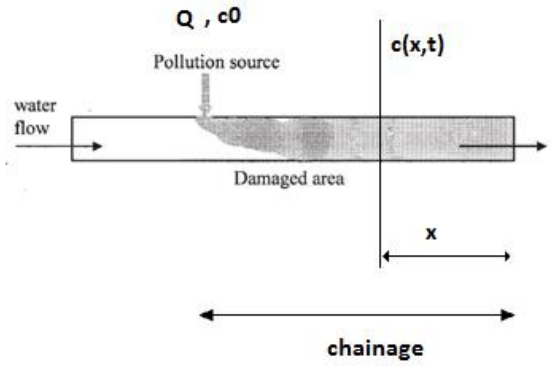
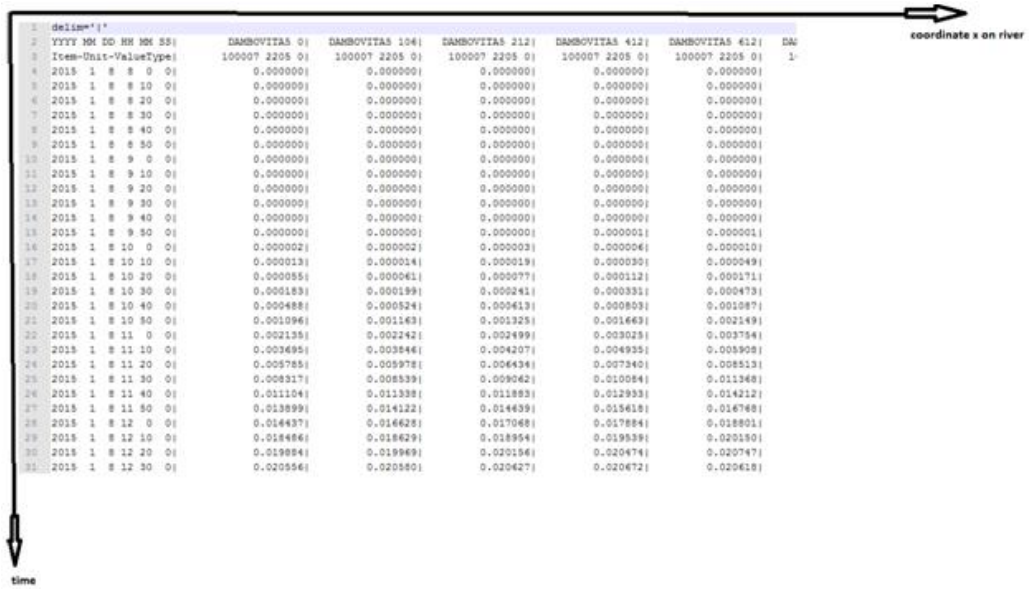


Fig. 1. Graphical representation of parameters involved in estimation of river's pollution

MIKE11 runs a simulation for each input pair and produces an output file (text file, Fig.2), in form of a matrix *M(m,n)* with *m*n* values, where *M(i,j)* represents concentration of pollutant at time t_i on the node x_j. The number of rows, *n*, is given by the timesteps for which the simulation is done and the number of columns *m* is given by the nodes on the river (location *x* along the river). In our specific case study we run simulations over a period of 9 hours with time steps of 10 minutes (*n*=55) and have a number *m*=155 nodes placed at a distance of about 200m one from the other. One MIKE simulation output consists of a 55x155 matrix with 8525 float values representing concentration values. It worth to be noted that the convention is to consider the reference point x=0, the downstream of the river.



	delim='|'										
2	YYYY MM DD HH MM SS|					DAMBOVITA5 0|	DAMBOVITA5 104|	DAMBOVITA5 212|	DAMBOVITA5 412|	DAMBOVITA5 612|	DA
3	Item-Unit-ValueType|					100007 2205 0|	100007 2205 0|	100007 2205 0|	100007 2205 0|	100007 2205 0|	1
4	2015	1	8	8	0 0|	0.000000|	0.000000|	0.000000|	0.000000|	0.000000|	
5	2015	1	8	8	10 0|	0.000000|	0.000000|	0.000000|	0.000000|	0.000000|	
6	2015	1	8	8	20 0|	0.000000|	0.000000|	0.000000|	0.000000|	0.000000|	
7	2015	1	8	8	30 0|	0.000000|	0.000000|	0.000000|	0.000000|	0.000000|	
8	2015	1	8	8	40 0|	0.000000|	0.000000|	0.000000|	0.000000|	0.000000|	
9	2015	1	8	8	50 0|	0.000000|	0.000000|	0.000000|	0.000000|	0.000000|	
10	2015	1	8	9	0 0|	0.000000|	0.000000|	0.000000|	0.000000|	0.000000|	
11	2015	1	8	9	10 0|	0.000000|	0.000000|	0.000000|	0.000000|	0.000000|	
12	2015	1	8	9	20 0|	0.000000|	0.000000|	0.000000|	0.000000|	0.000000|	
13	2015	1	8	9	30 0|	0.000000|	0.000000|	0.000000|	0.000000|	0.000000|	
14	2015	1	8	9	40 0|	0.000000|	0.000000|	0.000000|	0.000000|	0.000000|	
15	2015	1	8	9	50 0|	0.000000|	0.000000|	0.000000|	0.000001|	0.000001|	
16	2015	1	8	10	0 0|	0.000002|	0.000002|	0.000003|	0.000006|	0.000010|	
17	2015	1	8	10	10 0|	0.000013|	0.000014|	0.000019|	0.000030|	0.000049|	
18	2015	1	8	10	20 0|	0.000055|	0.000061|	0.000077|	0.000112|	0.000171|	
19	2015	1	8	10	30 0|	0.000183|	0.000199|	0.000241|	0.000331|	0.000473|	
20	2015	1	8	10	40 0|	0.000488|	0.000524|	0.000613|	0.000803|	0.001087|	
21	2015	1	8	10	50 0|	0.001096|	0.001163|	0.001325|	0.001663|	0.002149|	
22	2015	1	8	11	0 0|	0.002135|	0.002242|	0.002499|	0.003025|	0.003754|	
23	2015	1	8	11	10 0|	0.003695|	0.003846|	0.004207|	0.004935|	0.005908|	
24	2015	1	8	11	20 0|	0.005785|	0.005978|	0.006434|	0.007340|	0.008513|	
25	2015	1	8	11	30 0|	0.008317|	0.008539|	0.009062|	0.010084|	0.011348|	
26	2015	1	8	11	40 0|	0.011104|	0.011338|	0.011883|	0.012933|	0.014212|	
27	2015	1	8	11	50 0|	0.013899|	0.014122|	0.014639|	0.015618|	0.016748|	
28	2015	1	8	12	0 0|	0.016437|	0.016628|	0.017068|	0.017884|	0.018801|	
29	2015	1	8	12	10 0|	0.018486|	0.018629|	0.018954|	0.019539|	0.020150|	
30	2015	1	8	12	20 0|	0.019884|	0.019969|	0.020156|	0.020474|	0.020747|	
31	2015	1	8	12	30 0|	0.020556|	0.020580|	0.020627|	0.020672|	0.020618|	

Fig. 2. Output of MIKE11 simulation

In Fig.3-a, the concentration c(x) is depicted where x is the location along the river using steps of 1km. With different colors is represented c(x) for various values of time (t), choosing time step of 10 min. The initial conditions for these simulations are chosen according to a real pollution scenario where a tank transporting detergents with a capacity of 5 tons falls down at the middle of the studied segment of the river (chainage=15000m, c_0=1000000 and Q[0]=5000l).
It can be observed that the shape of the curve is about a Gauss bell with amplitude decreasing as the time passes. In Fig.3-b is represented c(t) where t is the time of the simulation (at 30 min steps). With different colors is represented c(t) for various nodes on the river (from km0 to km15, with a step of 5km). The initial conditions are the same as for Fig.3-a). The shapes, again, resemble a Gaussian bell with the amplitude decreasing as we go further from the place where the pollutant was discharged. One possible alternative to using MIKE11 could be to implement a numerical solving of equation (1), which is not feasible in practice because of difficulties it rises (topological modeling of the riverbed, computational costs of numerical solving of Equation (1)). Another alternative we explored was the Machine Learning API (respectively Google Prediction API [15]). A Google Prediction API model was trained. This model is defined based on values of previous runs of MIKE11. Once the model learns from previous values it can be used further to predict what will be the value for pollution given a user supplied input set of parameters. The model was trained with data from N=1210 simulations of Mike, respectively 10315250 records and the training process, which needs to be done only once, took about 29 minutes. The model was far away from the desired accuracy (the maximum relative percent error for

the predicted values was 90%), so we have focused our attention to another approach, respectively multidimensional spline interpolation which will be described in more details in the next chapter.

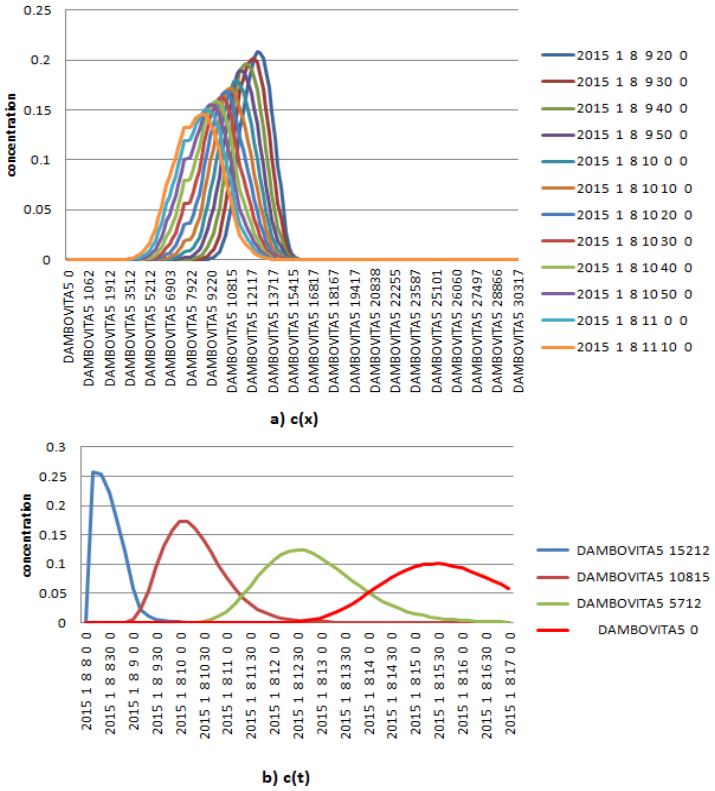


Fig. 3. Concentration of pollutant as a function of distance plotted at various moments in time (a) and Concentration of pollutant as function of time plotted for various nodes on the river (b)

5. Water quality assessment based on multivariate spline interpolation

The pollution of water on the river can be seen as a function of more variables, in our case it depends at least on the following: time, place on the river, concentration of pollutant discharged, place where the pollutant was discharged accidentally, quantity of the pollutant, type of pollutant (detergents, petroleum, etc).

$$c = c(x, t, c_0, x_0, q, \dots)$$

Since we can have access to past simulation data (concentration values for various values of the other parameters) the next step is to interpolate these into the N-dimensional space (in our case, 6-D space).

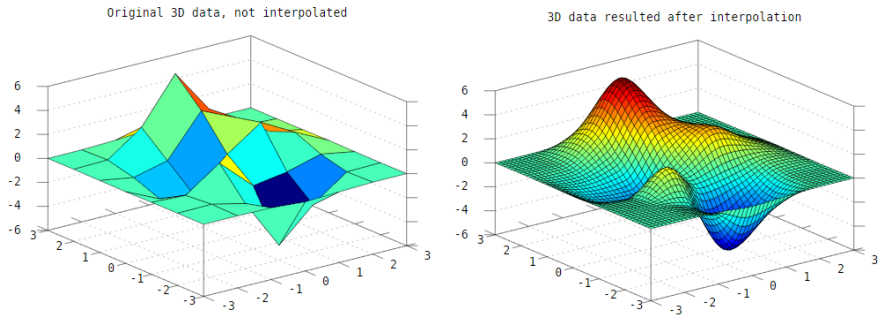


Fig. 4. Interpolation in 3D space

Fig. 4 shows interpolation in 3D space; for n>3 the visualization of n-dimensional objects and n-dimensional interpolation is less intuitive since humans are familiar to 3D space. Because a pollution event is determined in our case by three variables (c_0, x_0, q), to keep track of such accidental events we propose to define "pollution scenarios" (abbreviated PS). Each PS is defined by 3 parameters: c_0, x_0, q. One PS has one corresponding MIKE11 simulation (which produces one output file, ResAD.txt, which contains the matrix of concentrations presented in Fig.3). The next step is to do multivariate interpolation based on the set of simulation output files. This can be achieved using recursivity, from multivariate interpolation back to univariate interpolation. To store the historical data a *hyper-rectangular array* is used (an array with *n* dimensions; to access a specific element from this array an n-indexing scheme is necessary, such as $v[i_1][i_2][i_3]\ldots[i_n]$, where i_k are integer indexes). For n=2, the function $z=f(x,y)$ can be interpolated at a given dataset of points ($z_i = f(x_i, y_i)$) and a value computed for an arbitrary given point $(\tilde{x}, \tilde{y})$ using univariate interpolation; first considering the function in one variable y, $f(x_i, y)$ and giving values for x_i. For each value of x_i we interpolate the function in one variable and obtain a formula for $f(x_i, y)$ where the only variable is y. Let P_1(y) be that formula. Then we evaluate f($x_i, \tilde{y}$)= $P_1(\tilde{y})$. Second step is to consider the function $f(x,\tilde{y})$ where x is the only variable and to fit this using univariate interpolation. Once the formula is determined, let it be P_2(x), we can then obtain the desired value, by making $x=\tilde{x}$

and the result, $\boldsymbol{P_2(\tilde{x})}$ will be the approximation of $f(\tilde{x}, \tilde{y})$. Any method of interpolation can be used in the method presented above (linear, polynomial, etc), but for the accuracy of the results we have used cubic splines. For the n=3 problem we reduce the degree of the problem to n=2 using univariate interpolation, and from n=2 we can reduce to n=1 as described below. Thus, using mathematical induction, any problem of interpolation of order n can be reduced to n-1 then from n-1 to n-2, and so forth to n=1 (univariate interpolation).
Formally we can use the following notations [16]

$$\boldsymbol{p^{(n)} = \left(x^{(1)}, x^{(2)}, \dots, x^{(n)}\right)}$$

be a generic point in the Euclidean n-space $\boldsymbol{R_n}$ and a given dataset represented as a hyper-rectangular array in *n* dimensions

$$\{(\boldsymbol{x_{i1}^{(1)}, x_{i2}^{(2)}, \dots x_{in}^{(n)}})\}$$

with the corresponding function values

$$\{f((\boldsymbol{x_{i1}^{(1)}, x_{i2}^{(2)}, \dots x_{in}^{(n)}}))\} \; ; \; 0 \leq \boldsymbol{i_j} \leq \boldsymbol{N_j} \text{ and } 0 \leq j \leq n$$

We can write:

$$f(\boldsymbol{p^{(n)}}) \approx \boldsymbol{\prod_{j=1}^{n} I\left(x^{(j)}\right) f = P_{N1} P_{N2} \dots P_{Nn}\left(\tilde{x}^{(1)}, \tilde{x}^{(2)}, \dots, \tilde{x}^{(n)}\right)}$$

where $\boldsymbol{I\left(x^{(j)}\right) f}$, denotes the operator that represents univariate interpolation of function *f* with respect to a single variable $\boldsymbol{x^{(j)}}$.

6. Implementation of the water pollution application

The workflow of the application is represented in Fig. 5 bellow. The application has two main phases; first phase is to generate a large number of MIKE executions that will form a historical offline database that serves later, in the second phase, for the multivariate interpolation. A Java application is responsible for the first phase and generates a number of pollution scenarios to be executed on the server where MIKE11 runs (UTCB server). Two MIKE 11 input files are modified, updating the following values: start/end time of simulation, the chainage, concentration of pollutant, the mass of pollutant and time when it was discharged in the river's water. These files are then uploaded via FTP (1) to the MIKE11 server. In the second phase, MIKE 11 executions are launched. The output files of MIKE11, which consists of values of concentration at different locations on the river, are downloaded back (2) to development machine and

stored locally for later post-processing. The post-processing consists in extracting and loading the interest data into a larger CSV file (3) which is packaged into the J2EE application archive (*.war file). The J2EE application is responsible for interpolating the data from the CSV file and is deployed on Jelastic Cloud [17] to benefit by automated on-demand vertically and horizontally scaling. For performance reasons, because of the very large number of nodes to perform interpolation, we implemented segmentation and caching mechanism. One Spline Interpolation Java Object (SIO) is created only on a subset of the pollution parameters and used only when the end-user requests to evaluate the function in a point that corresponds to that range of parameters. The object is kept in memory and used later if other requests for the same range are issued. At the beginning, the user supplies a pollution scenario (PS) giving some pollution parameters as described above and then request a value for the concentration of the pollutant.

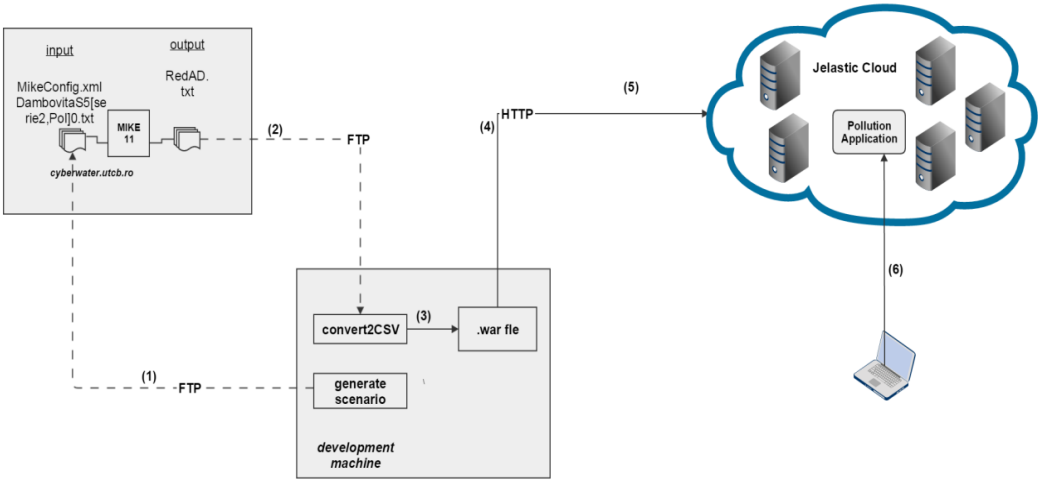


Fig. 5. Workflow of development and deployment for the multivariate interpolation application

The application checks in the memory if such a SIO already exists and if yes, it uses further that SIO to estimate the value of pollutant concentration. If not, it decides which of the CSV files are needed to create a new SIO, creates the object, call the method to estimate the value and place it in memory for further utilization. Using this approach, instead of creating a huge Java object which interpolates over 10^7 points in 6-D space, we create several smaller objects that interpolates over smaller subsets of points (e.g. 10^5) and use these objects when needed.

7. Experimental Validation

To assess the accuracy of the proposed solution we used the interpolation to estimate the concentration given a specific value for concentration, chainage

and quantity of pollutant for the whole set of time and distance nodes. Because there is no actual data available for pollution on the specific river stretch studied to assess the accuracy we compare our estimation with MIKE11 estimations. Due to advanced numerical methods used internally by MIKE11 (6-point Abbott scheme for finite differences) and to high precision floating point numbers used in computations, the error of MIKE11 is negligible (very close to 0). The obtained values from our solution (a matrix of 55x155 values) we compared with the matrix of values resulted from the MIKE11 running for the same parameters. Then we computed the relative percent error:

$$\%\text{Err} = \frac{|c_{estimated} - c_{MIKE}|}{c_{MIKE}} x100\%, \ (c_{MIKE} \neq 0)$$

The relative percent error values present a mean of 1.27 and a standard deviation of 4.31.

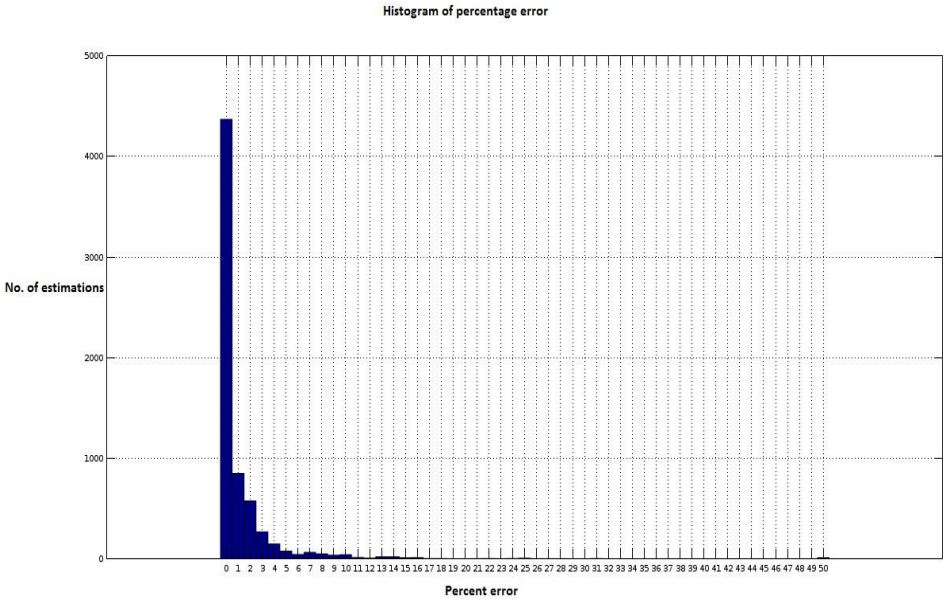


Fig. 6. Histogram representing percent error for multivariate interpolated data

The histogram presented in the Fig.6 above shows that 92% of the estimated data presents a percentage error less than 5%, and 81% of the estimated data has a percentage error less than 2%. One query takes on average 500 ms. The dynamic web application realized in J2EE was deployed in Jelastic Cloud that provides high-level on demand horizontal/vertical scaling. It also offers high availability and load balancing. The basic resource unit in Jelastic is a *cloudlet* that represents 128Mb RAM and 400MHz CPU. The vertical scaling for 1 node in the environment created for test can use, if needed, up to 128 cloudlets (16Gb RAM and 51.2 GHz). Each node has installed Tomcat 7.0.62 and Java 7. The horizontal scaling is achieved by adding more nodes of the same type as described

above. In Fig.7 CPU (GHz), RAM(Gb), Network(Mb), Disk(Gb) and IOPS of the deployed application on the Jelastic cloud are represented. The RAM and CPU are the main challenges of this computing intensive Big Data processing application. The RAM graphic presents two main parts. "Stage I" shows the constant increasing of RAM up to 4 GB and corresponds to a request to create a new SIO and place it in the main memory. In "Stage II" the memory consumption decreases gradually and stabilizes to the 2GB value because the SIO object is already found and can be used to interpolate and provide the result. The CPU consumption shows a peak at 6 GHz and corresponds to "Stage I", when the interpolation object is created.

Fig. 7. Jelastic Resource monitoring panel for water pollution application

8. Conclusions

This paper proposes an estimation method to evaluate the concentration of pollutant in river's water, using historical data and multivariate interpolation. The historical data is obtained from past executions of MIKE11 for a real case model on a 30-km long segment on Dîmboviţa river. Because we are doing interpolation for a function of 5 variables and work with millions of points in the 6-D space we have built a *computational intensive Big Data application* that was deployed on Jelastic Cloud. A segmentation method for splitting the hyper-rectangular array into more ranges was proposed in order to optimize the RAM and CPU consumption and avoid any un-necessary computations. Comparing our results from interpolation with the results that would have been obtained with MIKE11 we showed that our results have a mean percent error of 1.25%.

Acknowledgements

This research is part of the CyberWater project supported by the UEFISCDI PN II, PCCA 1, nr.47/2012. We would like also to address many thanks to our project partners from Technical University of Civil Engineering Bucharest, Faculty of Hydrotechnics, for their support related to the MIKE11 infrastructure.

R E F E R E N C E S

[1]. *Giupponi, C.; Sgobbi, A.*, "Decision Support Systems for Water Resources Management in Developing Countries: Learning from Experiences in Africa" Water 2013, *5*, 798-818.

[2] *Zhang, Kejiang, et al*., "Application of decision support systems in water management", Environmental Reviews 22.3 (2013): 189-205.

[3] *Willuweit, Lars, and John J. O'Sullivan*. "A decision support tool for sustainable planning of urban water systems: Presenting the Dynamic Urban Water Simulation Model", Water research 47.20 (2013): 7206-7220.

[4] *Sharma, Asheesh, Madhuri Naidu, and Aabha Sargaonkar*. "Development of computer automated decision support system for surface water quality assessment", Computers & Geosciences 51 (2013): 129-134.

[5]. *Ciolofan, Sorin N., Mariana Mocanu, and Anca Daniela Ionita*. "Cyberinfrastructure architecture to support decision taking in natural resources management." 2013- 19th International Conference on Control Systems and Computer Science.

[6] *Xu, Chang., et al*. "Application of MIKE11 Model in Water Quality Prediction of Hunhe River Basin." *Water Resources and Power* 6 (2013)

[7] *Liang, J., et al*. "MIKE 11 model-based water quality model as a tool for the evaluation of water quality management plans." Journal of Water Supply: Research and Technology- Aqua 64.6 (2015): 708-718.

[8] *Kanda, Edwin K., Job R. Kosgei, and Emmanuel C. Kipkorir*. "Simulation of organic carbon loading using MIKE 11 model: a case of River Nzoia, Kenya."Water Practice and Technology 10.2 (2015): 298-304.

[9] *Li, Jin, and Andrew D. Heap*. "Spatial interpolation methods applied in the environmental sciences: A review." Environmental Modelling & Software 53 (2014): 173-189.

[10] *Murphy, Rebecca R., Frank C. Curriero, and William P. Ball*. "Comparison of spatial interpolation methods for water quality evaluation in the Chesapeake Bay." Journal of Environmental Engineering 136.2 (2009): 160-171.

[11] *Chica-Olmo, Mario, et al.* "Categorical Indicator Kriging for assessing the risk of groundwater nitrate pollution: the case of Vega de Granada aquifer (SE Spain)." Science of the Total Environment 470 (2014): 229-239.

[12] , *R.L. Burden, J.D. Faires, and M. Annette*, "Numerical Analysis. 10-th edition" Cengage Learning (2016).

[13] *Hamming, Richard*. Numerical methods for scientists and engineers. Courier Corporation, 2012.

[14] DHI, MIKE11 Reference Manual, last time (July 2016) available at http://euroaquae.tu-cottbus.de/hydroweb/Platform/Notes/Mike11_Reference.pdf .

[15] Google Prediction API, last time (July 2016) available at https://cloud.google.com/prediction/

[16] *McKinney, Earl H.* "Generalized recursive multivariate interpolation."Mathematics of Computation 26.119 (1972): 723-735.

[17] Jelastic Cloud Documentation, last time (July 2016) available at https://docs.jelastic.com/