USING MATLAB FOR THERMAL PROCESSES MODELING AND PREDICTION AT MINING DUMPS

R. Hajovsky, B. Filipova

VSB – Technical University of Ostrava

Faculty of Electrotechnical Engineering and Computer Science

Department of Cybernetics and Biomedical Engineering

17. listopadu 15, Ostrava-Poruba, 708 33

Abstract

The paper deals with the modeling and prediction of thermal processes occurring in the old mining dumps in the Ostrava region. In this region there are many places that arose as a result of long-term mining industry in the past time. One of the most affected mine dumps is Hedvika. Hedvika was founded around 1910 and registered in the end of the 50th years, and since this time mining dump has been burning and some effort has been done in putting the fire out. Nowadays, Hedvika is still very thermally active. In this paper the temperature distribution throughout the mining dump Hedvika area by used the Matlab interpolation functions is described.

1 Introduction

In the Ostrava region there are places that arose as a result of mining industry in the local region. In this article we will focus on mining dumps incurred by the systematic gathering of tailings from the Ostrava mines. We chose mining dump Hedvika, which is located between Ostrava and Petrvald. Hedvika was founded around 1910 and registered in the end of the 50th years, and since this time mining dump has been burning and some effort has been done in putting the fire out. Nowadays, Hedvika is still thermally active. Temperatures inside the mining dump can be generally very high and so regular monitoring of the mining dump is necessary. During the solution of TACR project, the partial goal is to create temperature model distribution and model of its dynamic changes at mentioned dump. For this purpose, system Matlab together with interpolation functions is used.

2 Mining dump Hedvika

Mining dump Hedvika is a slag dump that belongs to former factory Důl Julius Fučík. The dump is mighty, huge and spacious formation of slag landfills. The landfills of slag rock have been carried out since 1903. Transportation of the slag rock to the heap was performed by means of mine pushcarts. Photography of the Hedvika mine dump is illustrated in Figure 1. The Hedvika mining dump is equipped with several tens of measurement probes that provide regular temperature measurements in 2 depth levels (3 and 6 m). The long-term temperature measurement at given depth levels is provided through a special measurement system which is based on use of special measurement probes together with telemetric station. Telemetric station provides primary data logging and subsequently wireless data transfer via GPRS technology to central data storage. In detail, this system in chapter 3 is described.

Figure 1: Hedvika mining dump photo

3 Measurement system

For long-term monitoring of temperatures and gas concentrations there have been designed and successfully implemented complex measurement systems including a large sensoric network containing both temperature and concentration sensors CO and CH_4. The system is centralized measurement system that contains particular sensors of temperature and gas (ones to tens), primary data concentrators and a central telemetric station. This station sends the data by GPRS technology to the central server data storage which is determined for data archiving and online visualization in the forms of trend lines on web pages. It also allows automatic data transfer to private own server via FTP technology. Measurement of temperatures and gas concentrations is realized in tens of probes covering affected area. The most typical spacing of the probes is 25 m.

Telemetric stations consist of universal data logger and GSM/GPRS communication module. Common battery operation is designed for multiple years of operation. Telemetric stations make it possible to set up large monitoring networks independent of external power sources and can be combined not only for temperature monitoring but also for other physical and electrical quantities. Depending on the type of I/O circuit, a single telemetric station can measure multiple input signals (temperature, gas concentrations, pressure etc). Due to these top-class features it possible to cover large areas by autonomous monitoring system, capable of performing measurements and sending data for years without human intervention.

As for temperature sensors, the system uses temperature sensors PT 100 or PT 1000 and gas sensors Figaro Engineering Inc. of series TGS 2442, TGS 2611, TGS 3870 etc. One of the main disadvantages of this system is mainly necessity of cablings connecting sensors with primary data concentrator and from concentrators to telemetric station. Block scheme of the complete measurement system is shown in Figure 2. [4]

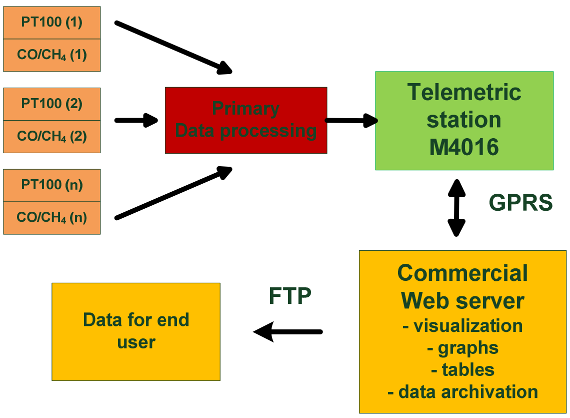


Figure 2. Block scheme of measurement system based on telemetric station

4 Interpolation in Matlab

Data interpolation is an application area based on underlying geometric algorithms. Data may be uniform, that is, sampling occurs over uniform intervals. Data may also be scattered, that is, sampling occurs over irregular intervals.

For uniform data Matlab applies functions interp1, interp2, interp3 in [5]. These functions use polynomial techniques, fitting the supplied data with polynomial functions between data points and evaluating the appropriate function at the desired interpolation points. When the sample data is scattered, the interpolation techniques use a triangulation-based approach as a basis for computing interpolated values. Matlab function griddata in [2] use the Delaunay triangulation for interpolation.

Interpolation is a method for estimating the value at a query location that lies within the domain of a set of sample data points. A sample data set defined by locations X and corresponding values V can be interpolated to produce a function of the form V = F(X). This function can then be used to evaluate a query point Xq, to give Vq = F(Xq). This is a single-valued function; for any query Xq within the domain of X it will produce a unique value Vq. The sample data is assumed to respect this property in order to produce a satisfactory interpolation. One other interesting characteristic is that the interpolating function passes through the data points. This is an important distinction between interpolation and curve/surface fitting. In fitting, the function does not necessarily pass through the sample data points.

The computation of the value Vq is generally based on the data points in the neighborhood of the query point Xq. There are numerous approaches to performing interpolation. In MATLAB interpolation is classified into two categories depending on the structure of the sample data. The sample data may be ordered in a axis-aligned grid or they may be scattered. In the case of a gridded distribution of sample points, you can leverage the organized structure of the data to efficiently find the sample points in the neighborhood of the query. Interpolation of scattered data on the other hand requires a triangulation of the data points, and this introduces an additional level of computation.

4.1 One – dimensional data

The function interp1 performs one-dimensional interpolation. The basic syntax is: yi = interp1(x,Y,xi), interpolates to find yi, the values of the underlying function Y at the points in the vector or array xi. x must be a vector. Y can be a scalar, a vector, or an array of any dimension, subject to the following conditions:

- If Y is a vector, it must have the same length as x. A scalar value for Y is expanded to have the same length as x. xi can be a scalar, a vector, or a multidimensional array, and yi has the same size as xi.
- If Y is an array that is not a vector, the size of Y must have the form [n,d1,d2,...,dk], where n is the length of x. The interpolation is performed for each d1-by-d2-by-...-dk value in Y. The sizes of xi and yi are related as follows:
    - If xi is a scalar or vector, size(yi) equals [length(xi), d1, d2, ..., dk].
    - If xi is an array of size [m1,m2,...,mj], yi has size [m1,m2,...,mj,d1,d2,...,dk].

yi = interp1(Y,xi) assumes that x = 1:N, where N is the length of Y for vector Y, or size(Y,1) for matrix Y.

yi = interp1(x,Y,xi,method) interpolates using alternative methods:

'nearest': Nearest neighbor interpolation

'linear': Linear interpolation (default)

'spline': Cubic spline interpolation

'pchip': Piecewise cubic Hermite interpolation

'cubic': (Same as 'pchip')

4.2 Two – dimensional data

The function interp2 performs two-dimensional interpolation. The basic syntax is: ZI = interp2(X,Y,Z,XI,YI) returns matrix ZI containing elements corresponding to the elements of XI and YI and determined by interpolation within the two-dimensional function specified by matrices X, Y, and Z. X and Y must be monotonic, and have the same format ("plaid") as if they were produced by meshgrid. Matrices X and Y specify the points at which the data Z is given. Out of range values are returned as NaNs. XI and YI can be matrices, in which case interp2 returns the values of Z corresponding to the points (XI(i,j),YI(i,j)). Alternatively, you can pass in the row and column vectors xi and yi, respectively. In this case, interp2 interprets these vectors as if you issued the command meshgrid(xi,yi).

ZI = interp2(Z,XI,YI) assumes that X = 1:n and Y = 1:m, where [m,n] = size(Z).

ZI = interp2(Z,ntimes) expands Z by interleaving interpolates between every element, working recursively for ntimes. interp2(Z) is the same as interp2(Z,1).

ZI = interp2(X,Y,Z,XI,YI,method) specifies an alternative interpolation method:

'nearest': Nearest neighbor interpolation 'linear': Linear interpolation (default)

'spline': Cubic spline interpolation

'cubic': Cubic interpolation, as long as data is uniformly-spaced. Otherwise, this method is the same as 'spline'.

The following picture illustrates code in Matlab, which calculates and shows temperature distribution at dump Hedvika.

```
%% Cubic interpolation
maticeZ_3=griddata(x,y,z_3,vektorX,vektorY,'cubic');% cubic interpolatio
figure(21)
mesh(vektorX,vektorY,maticeZ_3); %surf
xlabel('x'),ylabel('y'),zlabel('Teplota - 3m ')
title('Cubic interpolation - 3m')
colorbar
caxis([mmin_3 mmax_3]); %configure the same range of colors
hold on
plot3(x,y,z_3,'ro','MarkerFaceColor','r'),
%% The temperature distribution - 3m
figure(222)
pcolor(vektorX,vektorY,maticeZ_3)%The temperature distribution
colorbar
caxis([mmin_3 mmax_3]); %configure the same range of colors
hold on
plot(x,y,'ro','MarkerFaceColor','r'),
xlabel('x'),ylabel('y')
shading interp %no grid
```

The following pictures illustrate temperature distribution at mining dump Hedvika for the period between 08/2011 and 10/2011. The data for those graphs are acquired by means of large autonomous measurement system containing 24 temperature sensors in 12 probes. The other pictures show graphs of temperature trend over given time period and it demonstrates temperature increase in some dump´s places. Those results correspond with results of temperature distribution calculated by means of interpolation function (color changes).

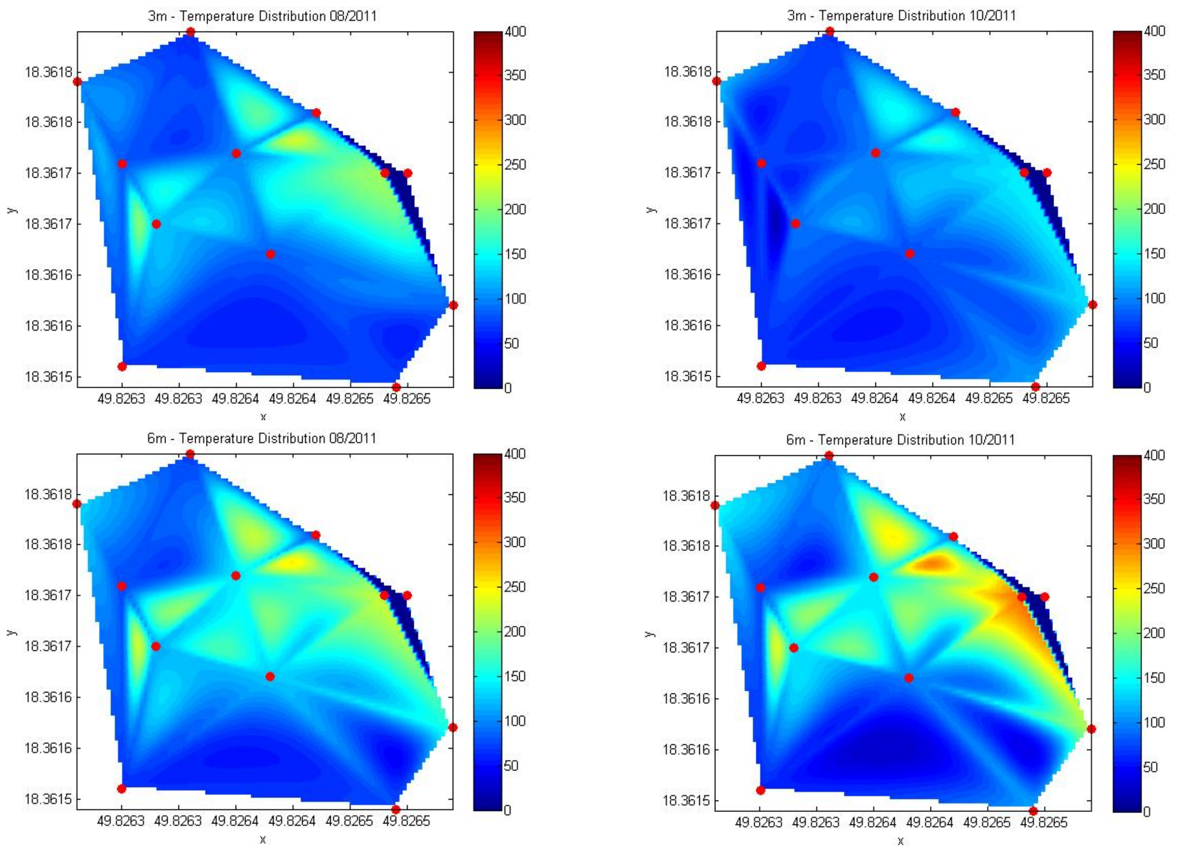
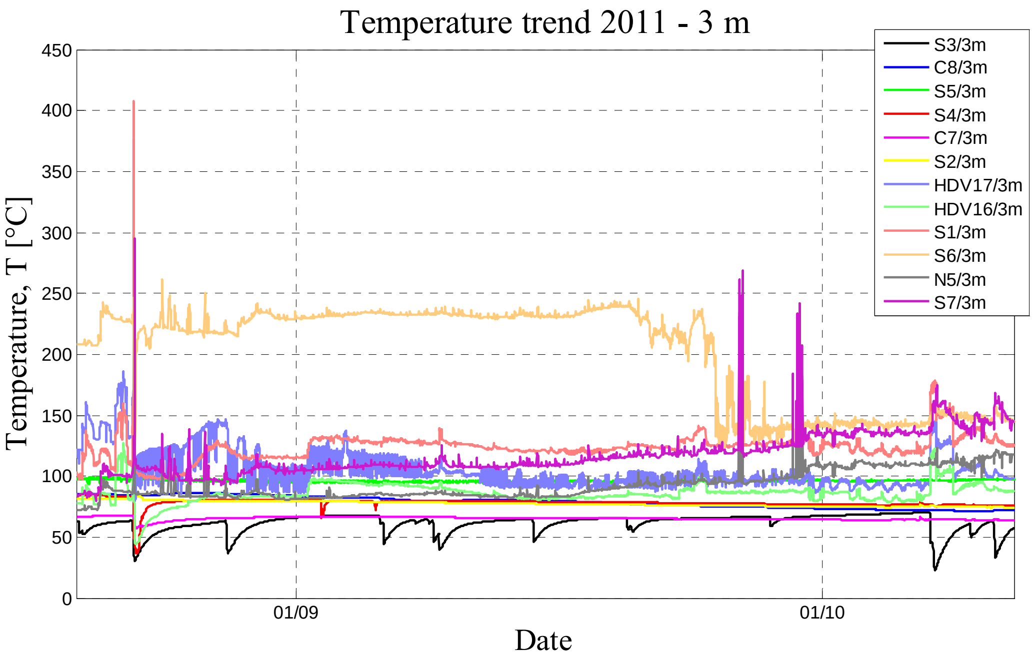


Figure 3. 3m and 6m Temperature distribution on the mining dump Hedvika



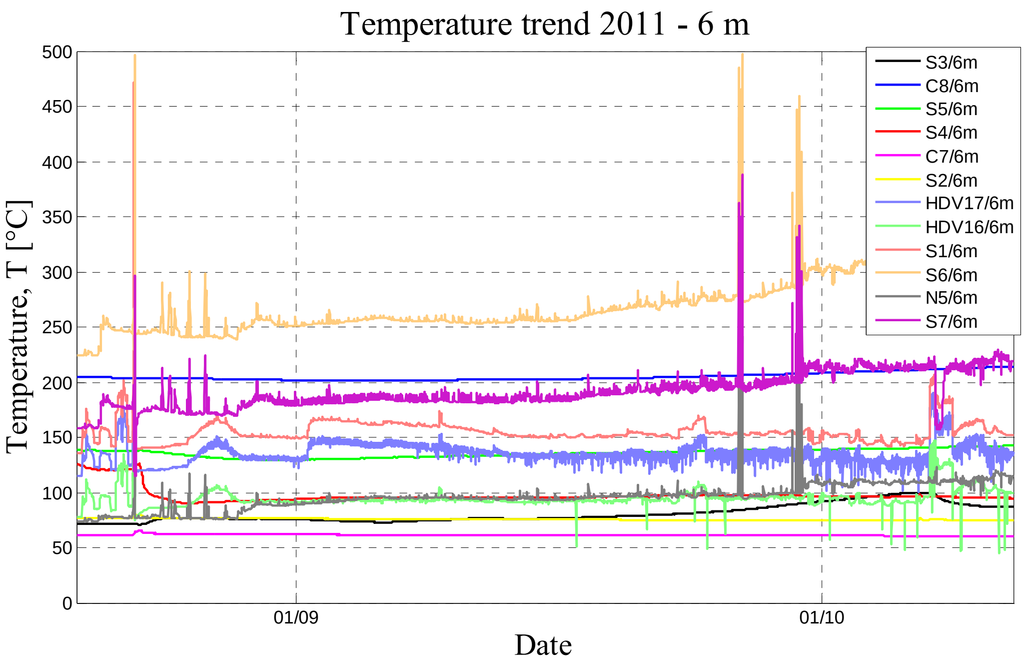


Figure 4. Temperature trends at mining dump Hedvika

5 Matlab and MySQL connection

The Matlab's Database Toolbox can be favourably used for import and processing of the data stored in MySQL database. Figure. 5 illustrates an example of connection between Matlab and MySQL. This examples launches data connect to the database and consequently selects all the temperatures from Hedvika table that follows FROM-TO condition for the date. Finally it plots the graphs with temperature distribution in 1m depth.

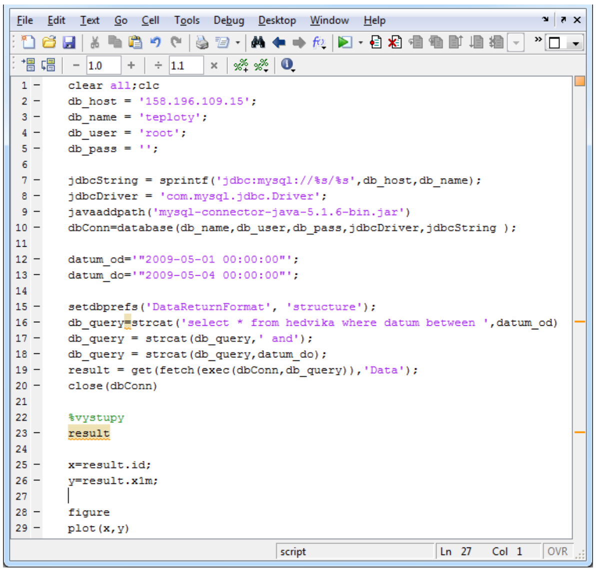
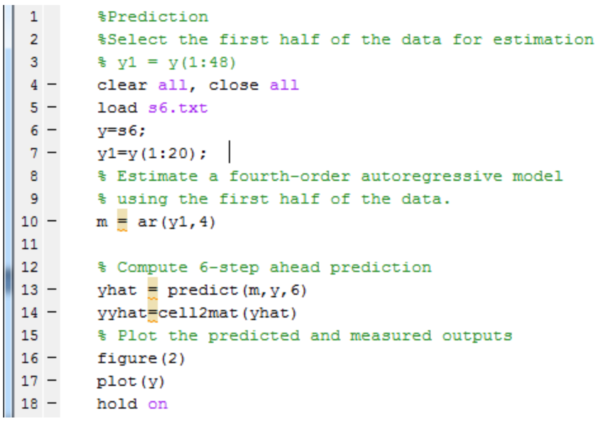


Figure 5. Demonstration of connection between MySQL and Matlab

The introduced example introduces the possibility of connection between Matlab and MySQL and data import from the database. In this case MySQL is used as central data storage which serves as the source for needed measured data for consequent calculation of temperature distribution at the dump and for prediction of their spread. For these purposes Matlab provides specialized functions, mainly for temperature distribution modeling (interp).

6 Thermal processes prediction

For first time prediction testing in Matlab, we used *Predict* function.



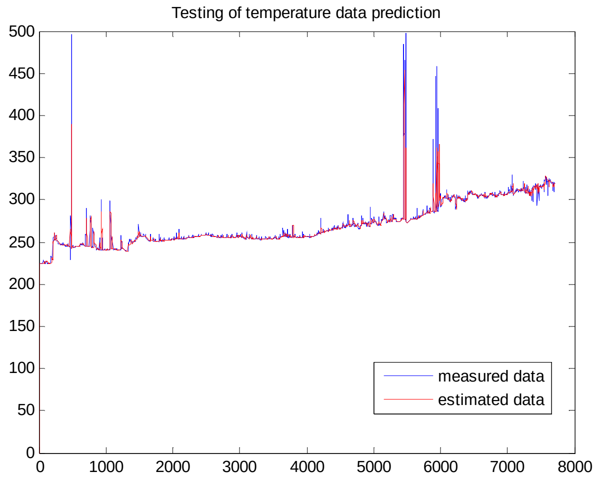


Figure 6. Demonstration of prediction in Matlab

7 Results

The paper dealt with a description of system Matlab for thermal processes modeling and prediction at old mining dumps in Ostrava region. By the end of 2011 it is planned to run the web pages with on-line visualization of measured data. They will contain both current values of the temperature in given probes and concentrations of CO gas. Visualization will also contain transparent historical trends over previous time periods.

References

[1] Stoer, J., Bulirsch, R.: *Introduction to Numerical Analysis.* Springer Science + Business Media, New York, USA, 2002. ISBN 978-0-387-95452-3.

[2] Wikipedia, URL:http://en.wikipedia.org/wiki/ Delaunay_triangulation.

[3] B. Filipova, R. Hajovsky, Using MATLAB for Modeling of Thermal Processes in a Mining Dump, *9th IASME /WSEAS Int. Conf. on HEAT TRANSFER, THERMAL ENGINEERING and ENVIRONMENT (HTE '11),* Florence, Italy 2011.

[4] R. Hajovsky, S. Ozana, Long Term Temperature Monitoring and Thermal Processes Prediction Within Mining Dumps, *The 6th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications,*Prague, Czech republic 2011

[5] Matlab help

Acknowledgment

The work and the contribution were supported by the project Grant Agency of Czech Republic – TAČR TA01020282 “Enhancement of quality of environment with respect to occurrence of endogenous fires in mine dumps and industrial waste dumps, including its modeling and spread prediction”.

Ing. Blanka Filipová, Ph.D.
VŠB-TU Ostrava, FEI, Katedra kybernetiky a biomedicínského inženýrství, 17. listopadu 15, Ostrava-Poruba, 708 33, tel: +420597323229, e-mail: blanka.filipova@vsb.cz

Ing. Radovan Hájovský, Ph.D.
VŠB-TU Ostrava, FEI, Katedra kybernetiky a biomedicínského inženýrství, 17. listopadu 15, Ostrava-Poruba, 708 33, tel: +420597324221, e-mail: radovan.hajovsky@vsb.cz