AGH DRILLING, OIL, GAS • Vol. 32 • No. 1 • 2015

http://dx.doi.org/10.7494/drill.2015.32.1.141

Jan Barbacki*

APPLICATION FOR QUANTIFYING THE LITHOLOGY OF A ROCK FORMATION USING GEOPHYSICAL WELL LOGS AND CONSTRAINED OPTIMIZATION METHODS

1. INTRODUCTION

Porosity and water saturation are the key reservoir parameters for hydrocarbon and water prospecting. To determine the mentioned parameters the set of equations that combine geological parameters with well logs is constructed [1, 3]. The goal of this paper is to present a comprehensive application for processing and interpreting these well logs.

2. PROBLEM

Porosity and lithology are calculated from the set of equations describing model of a rock formation (1). These equations combine porosity and percentage volume of rock components with corresponding logs through the given geological factors and parameters of formation water. Water saturation is obtained from resistivity log (RT or RX0) [3].

$$P_i \pm \delta_i w_i = \sum_{j=1}^{N} \left(a_{ij} \times V_j \right) \tag{1}$$

where:

- P_i – geophysical measurement,
- δ_i – measurement error,
- w_i – weight of measurement,
- a_{ij} – geological factor,
- V_j – percentage volume of component (mineral, porosity, water saturation),
- N – total number of components.

* AGH University of Science and Technology, Faculty of Drilling, Oil and Gas, Krakow, Poland

Optimal solution to above-mentioned set of equations is obtained by minimizing the error function, defined as a difference between theoretical value and observed value [1, 3]. The objective function is given by formula (2):

$$f\left(\phi, V_j, S_w\right) = \sum_{i=1}^{M} \left(\frac{P_i^{theor} - P_i^{obs}}{P_i^{obs}} \right)^2 \tag{2}$$

where:

- ϕ – porosity,
- V_j – percentage volume of mineral component,
- S_w – water saturation,
- P_i^{theor} – theoretical value,
- P_i^{obs} – observed value,
- M – total number of measurements.

Assumptions:

1) percentage volume of each component is between 0 and 1;
2) total percentage volume of mineral components and porosity is equal 1.

3. SOLUTION

Application for well log processing and interpretation was written in C# programming language, using integrated development environment Microsoft Visual Studio 2008 Professional and ALGLIB – a cross-platform numerical analysis and data processing library. For finding a proper solution to (2), *minbleic* subpackage was used – a part of ALGLIB library which supports boundary and linear equality/inequality constraints [4]. This subpackage allows to use BLEIC (Boundary, Linear Equality – Inequality Constraints) algorithm, based on nonlinear conjugate gradient method, to solve following optimization problems (3):

$$\begin{aligned} &\min f(x) \ \ s.t. \\ &l_i \le x \le u_i, \qquad l, x, u, \in R^N \\ &Ax \circ b, \qquad A \in R^{K \times N}, \text{where } \circ \text{ is any combination of } \le, =, \ge \end{aligned} \tag{3}$$

For every depth *minbleicoptimize* was performed using the optimal solution from previous depth as starting points. For the first depth starting points was obtained by the Monte Carlo method [2]. Gradient was calculated by ALGLIB through numerical differentiation provided by *minbleiccreatef* method.

Optimization methods

Nonlinear conjugate gradient method (NCGM) is generalized form of the conjugate gradient method adapted to nonlinear optimization. In general, the NCGM is an deterministic optimization algorithm that finds the local minimum of a nonlinear function using its gradient. Monte Carlo method (MCM) is used for finding a global minimum of the objective function by random sampling. The MCM is a time-consumed method, therefore number of iteration was limited to 10^5 iterations.

ALGLIB library – example of use

Minimize:

$$f(x, y) = (1 - x)^2 + 100(y - x^2)^2$$

subject to:

$$0 \le x \le 3$$
$$0 \le y \le 3$$
$$x + y = 4$$

Code:

```
class Program
{
    static void Main(string[] args)
    {
      gradient();
    }
    static void gradient()
    {
      double[] x = new double[] {0, 0}; //initial values x = x[0], y = x[1]
      double[,] c = {
      { 1, 0, 0}, //x[0] >= 0
      { 1, 0, 3}, //x[0] <= 3

      { 0, 1, 0}, //x[1] >= 0
      { 0, 1, 3}, //x[1] <= 3

      { 1, 1, 4} //x[0] + x[1] = 4
      };

      int[] ct = new int[] { 1, –1, 1, –1, 0 }; // >= <= >= <= =

      alglib.minbleicstate state;
      alglib.minbleicreport rep;
```

```
        //stopping conditions for the optimizer
        double epsg = 0;
        double epsf = 0;
        double epsx = 0;
        int maxits = 0;

        //differentiation step
        double diffstep = 1.0e–6;

        alglib.minbleiccreatef(x, diffstep, out state); //creating optimizer
        alglib.minbleicsetlc(state, c, ct); //adding linear constraints
        alglib.minbleicsetcond(state, epsg, epsf, epsx, maxits); //tuning stopping condi-
        tions
        alglib.minbleicoptimize(state, rosenbrock, null, null); //optimization
        alglib.minbleicresults(state, out x, out rep); //obtaining results

        //printing values of x[0] and x[1]
        Console.WriteLine("x = " + Math.Round(x[0], 3).ToString());
        Console.WriteLine("y = " + Math.Round(x[1], 3).ToString());

        //printing the value of f(x[0], x[1])
        double rval = 0;
        object o = new object();
        rosenbrock(x, ref rval, o);
        Console.WriteLine("f(x, y) = " + Math.Round(rval, 3).ToString());

        Console.ReadKey(); }

    static void rosenbrock(double[] x, ref double func, object obj)
    {
        func = (1 – x[0]) * (1 – x[0]) + 100 * (x[1] – x[0] * x[0]) * (x[1] – x[0] * x[0]);
    }
}
```

Results:
x = 1,561
y = 2,439
f(x, y) = 0,315

Program description

For calculations the following set of logs was given:

- interval transit time DT [us/m],
- gamma radiation GR [API],
- neutron porosity NPHI [% in limestone scale],
- bulk density RHOB [g/cm^3],
- resistivity of the uninvaded formation RT [Ω·m],
- resistivity of the invaded zone RX0 [Ω·m].

The expected input was – beside parameters defined by user – the file with logs encrypted in LAS format (Fig. 1).

```
~Version Information Block
VERS.                    1.20:   CWLS log ASCII Standard -WERSJA 1.20
WRAP.                      NO:   Dane z tej samej glebokosci w jednej linii
~Well Information Block
STRT.M                2190.000:
STOP.M                2365.000:
STEP.M                   0.250:
NULL.                 -999.250:   Wartosc nieokreslona
~Curve Information Block
#MNEM.UNIT      API CODE      Curve Description
#---------    -------------   ------------------------------
DEPT.M                                   : Depth
CALI.MM                                  : Prof.srednicy otworu w mm
DT  .US/M                                : Prof.czasu interwalowego w us/m
GR  .API                                 : Prof.gamma naturalne w API
GRS .API                                 : Prof.gamma bez uranu w API
LLD .OHMM                                : Prof.opornosci laterologiem dl.w omm
LLS .OHMM                                : Prof.opornosci laterologiem sr.w omm
NPHI.PERC                                : Prof.porowatosci neutronowej w %
RHOB.G/CM3                               : Prof.gestosci w g/cm3
~A DEPT        CALI        DT        GR        GRS        LLD        LLS        NPHI        RHOB
2190.000     240.168    169.674     8.730     4.336    460.977     46.466      0.305      2.975
2190.250     236.726    169.656     8.224     4.501    376.792     43.944      0.332      2.981
2190.500     236.139    169.893     7.700     4.925    251.683     40.131      0.438      3.001
2190.750     233.887    169.214     8.963     5.544    168.487     38.387      0.885      2.989
2191.000     232.103    169.062    11.711     6.319    124.447    750.117      2.377      2.903
2191.250     230.671    171.987    15.715     8.185     84.151    741.617      6.871      2.777
```

Fig. 1. Example of input file in LAS format showed in external text editor

LAS format description

LAS (*Log ASCII Standard*) is a standard file-format to store well log information [5]. Each LAS file contains numerous sections and each section begins with a tilde (~) mark. The sections that make up an LAS file are as follows:

~V – version and wrap mode information [mandatory],
~W – well identification [mandatory],
~C – curve information [mandatory],
~P – parameters or constants [optional],
~O – other information such as comments [optional],
~A – ASCII log data [mandatory].

Well Information section defines the following mnemonics:

STRT.M – the first depth in the file;
STOP.M – the last depth in the file;
STEP.M – the depth increment used;
NULL. – the null value.

The hash sign (#) is used to indicate that the line is a comment line.

Application in use

The main window (Fig. 2) allows to:

- read from LAS file;
- define depth interval used in interpretation;
- activate *Parameters* window.

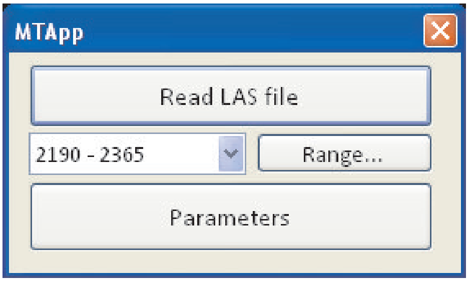

Fig. 2. Main window

The *Parameters* window (Fig. 3–6) allows to:

– mask logs i.e. to choose logs participating in interpretation;
– select resistivity log for optimization (RT/RX0);
– define: general parameters, mud parameters, mud filtrate parameters, formation water parameters, hydrocarbons parameters, loam parameters;
– select the model parameters;
– mask minerals i.e. to choose minerals participating in interpretation;
– define mineral properties (Fig. 7);
– set output parameters;
– start calculations.

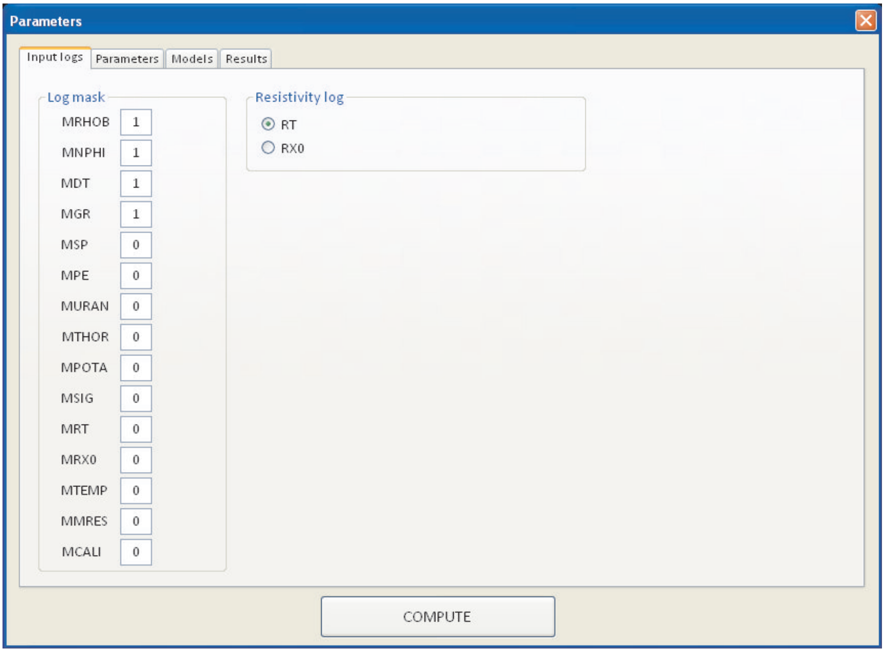

Fig. 3. *Input logs* tab in the *Parameters* window

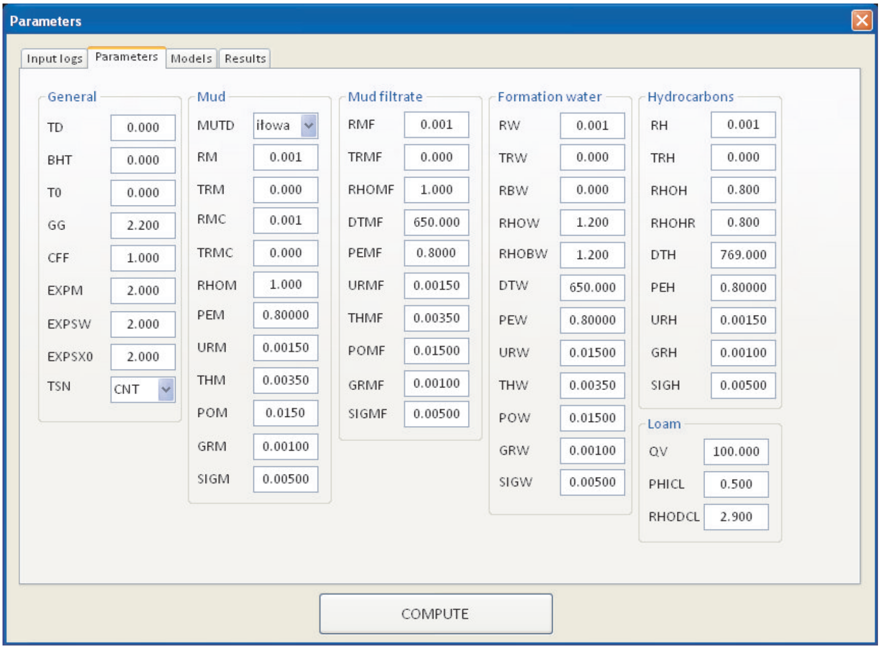

Fig. 4. *Parameters* tab in the *Parameters* window

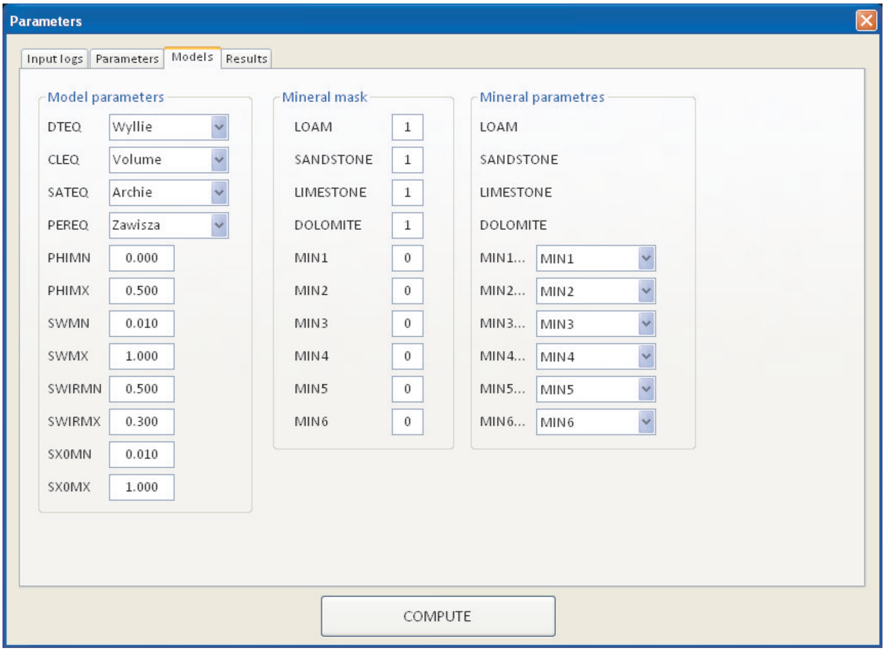

Fig. 5. *Models* tab in the *Parameters* window

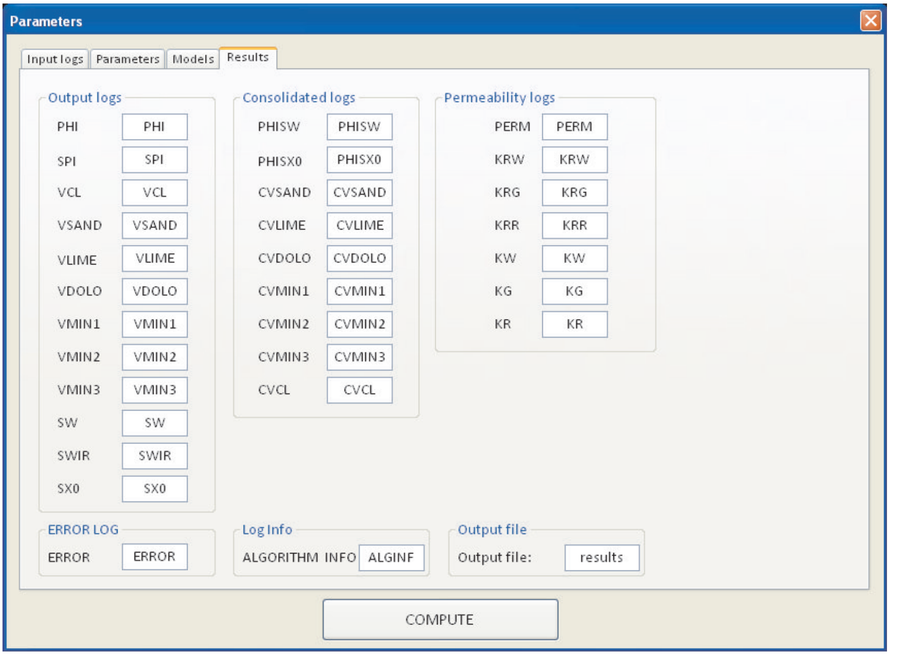

Fig. 6. *Results* tab in the *Parameters* window

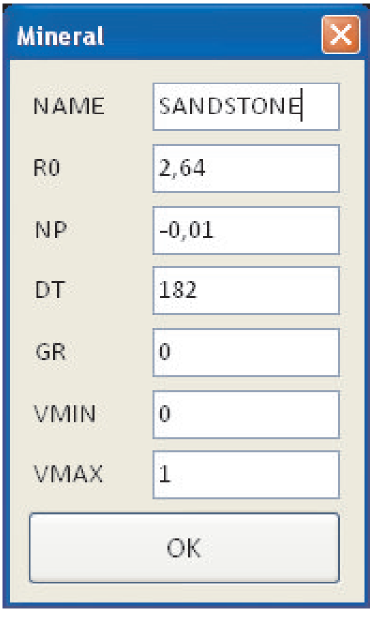

Fig. 7. *Mineral* window

Every component contained in *Parameters* window can provide short information (hint) when the user places the pointer on its name. This mechanism is known as *tooltip* (Fig. 8).

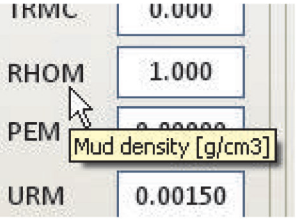

Fig. 8. Tooltip example

After clicking the *COMPUTE* button a progress bar (Fig. 9) which visualize how many of depths was already computed became visible. After completion of calculations the *Results* window is shown.

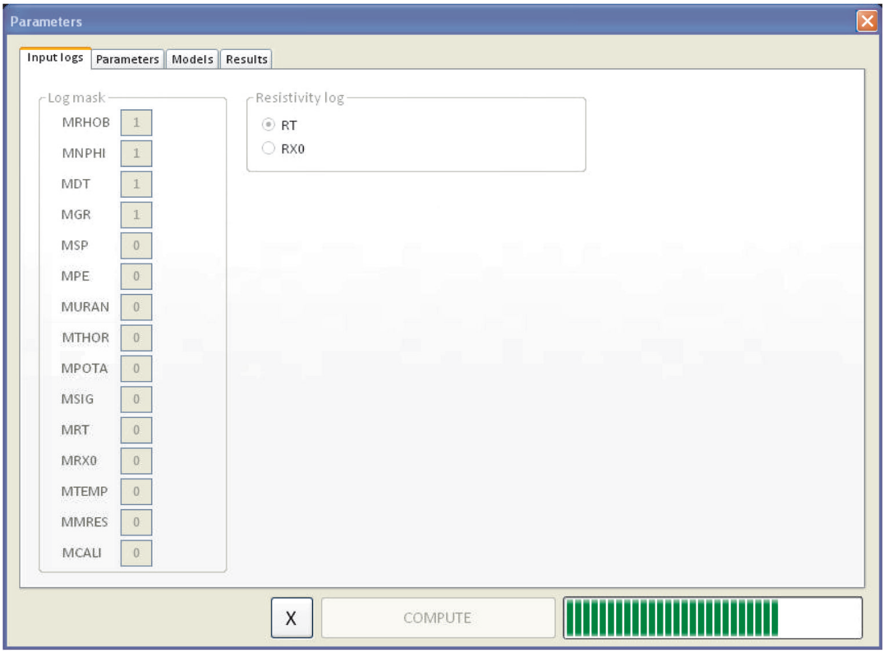

Fig. 9. *Parameters* window during computation

The *Results* window (Fig. 10) consists of:

– graphs of logs participating in interpretation (both experimental and theoretical – Fig. 12),
– RT/RX0 graph,

– lithology graph,
– water saturation graph,
– total error graph.

When user clicked on the lithology graph the *Legend* window appears (Fig. 11).

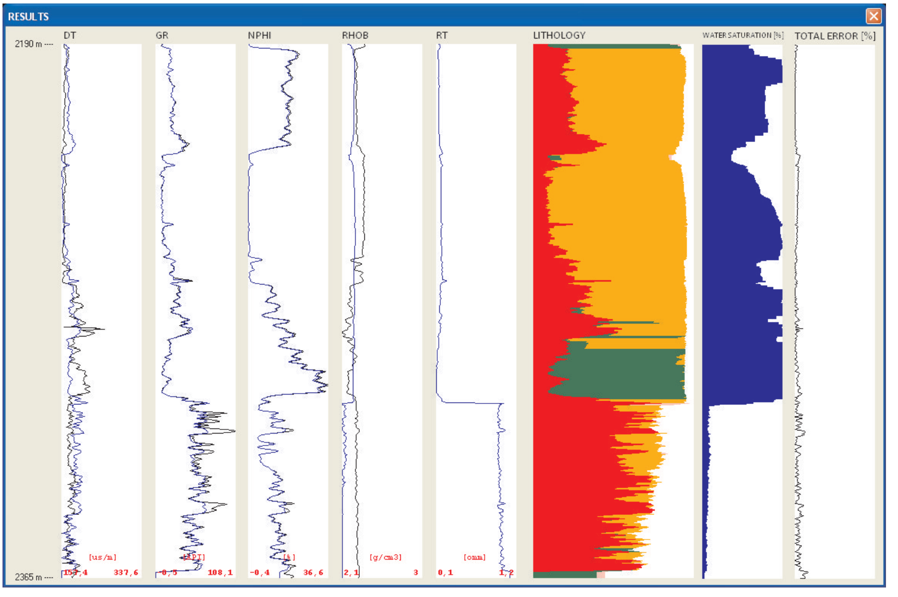

Fig. 10. *Results* window – from the left: DT, GR, NPHI, RHOB, RT, LITHOLOGY, WATER SATURATION [%], TOTAL ERROR [%]

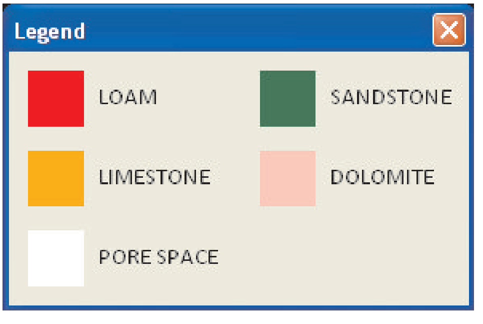

Fig. 11. *Legend* window

Every component contained in the *Results* window can provide basic information such as log name and the range of taken values via *tooltip.* Precise values of percentage volume of selected minerals, pore space and water saturation are available in automatically generated LAS file, placed in folder containing input logs, under the name given by user (Fig. 13).

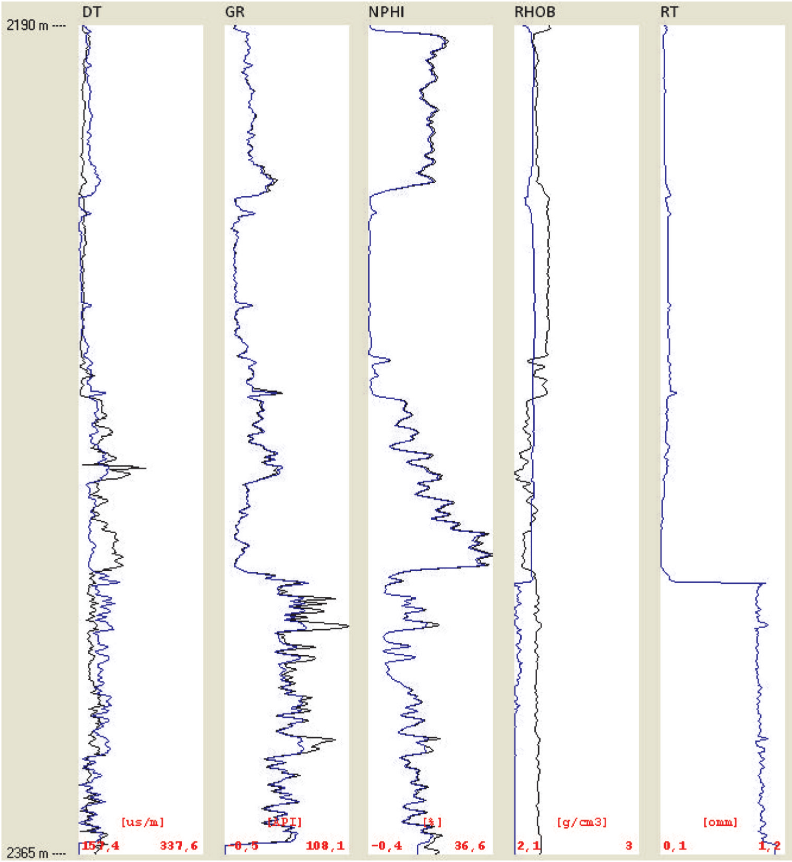

Fig. 12. Experimental logs (black curves) with theoretical logs (blue curves)

```
~Version Information Block
VERS.                        1.20:   CWLS log ASCII Standard -WERSJA 1.20
WRAP.                          NO:   Dane z tej samej glebokosci w jednej linii
~Well Information Block
STRT.M                   2190.000:
STOP.M                   2365.000:
STEP.M                      0.250:
NULL.                    -999.250:   Wartosc nieokreslona
~Curve Information Block
DEPT.M              : Depth
PORO.PERC           : Pore space [%]
LOAM.PERC           : Loam [%]
SAND.PERC           : Sandstone [%]
LIME.PERC           : Limestone [%]
DOLO.PERC           : Dolomite [%]
WATE.PERC           : Water saturation [%]
~A DEPT     PORO        LOAM        SAND        LIME        DOLO        WATE
2190,00     18,971      9,457       68,866      0,000       2,706       26,808
2190,25     18,405      9,341       72,253      0,000       0,000       26,463
2190,50     17,245      8,639       41,556      32,560      0,000       27,717
2190,75     15,960      10,330      29,324      44,386      0,000       29,523
2191,00     12,654      13,226      0,000       74,121      0,000       36,248
2191,25     9,606       17,801      0,000       72,592      0,000       48,477
2191,50     7,516       21,559      0,000       70,926      0,000       63,991
2191,75     7,733       22,756      0,000       69,511      0,000       64,336
2192,00     7,920       21,186      0,707       70,187      0,000       64,396
2192,25     8,002       20,455      1,015       70,527      0,000       64,524
```

Fig. 13. Results of computation in LAS format showed in external text editor

4. SUMMARY

The purpose of this paper was to present a comprehensive application for determining the lithology, porosity and water saturation from well logs. Presented application makes it possible to compare experimental logs with computed logs as well as to visualize a total error, what significantly simplify verification of results correctness, allowing to enhance models that was used in interpretation. Moreover, output data encoded in LAS format allow to perform further processing of obtained results in another programs.

REFERENCES

[1] Jarzyna J. (ed.), Bała M., Cichy A., Gądek W., Gąsior I., Karczewski J., Marzencki K., Stadtmüller M., Twaróg W., Zorski T.: *Przetwarzanie i interpretacja profilowań geofizyki wiertniczej – system GeoWin*. ARBOR, Kraków, 2002.

[2] Jarzyna J. (ed.), Bała M., Cichy A., Gądek W., Karczewski J., Marzencki K., Stadtmüller M., Twaróg W., Zorski T.: *Przetwarzanie i interpretacja profilowań geofizyki wiertniczej – system GeoWin. Cz. II: Nowe aplikacje i uzupełnienia*. ARBOR, Kraków, 2007.

[3] Jarzyna J., Cichy A., Gądek W., Twaróg W.: *InterLog – kompleksowa interpretacja profilowań geofizyki wiertniczej w systemie GeoWin.* Szymbark, wrzesień 2002.

[4] Bochkanov S.: *Bound and linear equality/inequality constrained optimization.* http://www.alglib.net/optimization/boundandlinearlyconstrained.php [24.03.2015].

[5] Canadian Well Logging Society: *LAS 1.2 – A FLOPPY DISK STANDARD FOR LOG DATA*., 1990.

[6] http://www.cwls.org/wp-content/uploads/2014/09/LAS12_Standards.txt [24.03.2015].