GEOFLOW 1.0: Motor of calculation to model the problem of flow confined in grounds with finite differences in irregular meshes.

Pallares M. Myriam Rocío[†] , Rodríguez C. Wilson[*]

[†] Grupo de Investigación en Simulación y Control Numérico (SICON)
Universidad Santo Tomás
Colombia
e-mail: myriampallares@correo.usta.edu.co

[*] Centro de Investigación en Modelación Numérica y Desarrollo de Software CAE (CIMON)
Universidad de La Salle
Colombia
e-mail: wrodríguez@unisalle.edu.co

Key words: *over-relaxation, finite differences*

Abstract: *The present work consists of the development and implementation of the finite differences method for over-relaxation adapted to irregular meshes to model the two dimensional porous media problems. These are the flow under concrete dams and the flow in the domain of a sheet-pile wall. GID like a pre/post-process platform and FORTRAN77 like a programming language of the motor of differences GEOFLOW 1.0, are used. The domain is constituted by rectangles divided in non uniform rectangular meshes. Additionally, a comparison with obtained results is made in the MATLAB PDETOOL.*

1 INTRODUCTION

The finite differences method is a simple numerical technique that we can use to solve partial differential equations. A solution with finite differences to Laplace’s equation or Poisson’s equation can be obtained in two steps. First, approaching to differential equation and the boundary conditions by means of a group linear algebraic equations called equations of differences, into the points of a grid located inside of the solution region, and Second, solving this group of algebraic equations.

In this work we apply the finite differences method to determine the flow under concrete dams and sheet-pile walls. In this article, we present a basic description of the electric analogy with the problem of the porous media flow, the implementation with GiD and the use of the differences program. For the size restriction of this document two models are presented only, however, Geoflow 1.0 was checked in several problems and it was compared with the Matlab PDETool solutions, satisfactorily.

2 ELECTRIC ANALOGY

The figure 1 shows the relationship between the variables of the two problems. This analogy is employed to develop “Geoflow 1.0” and the Matlab models.

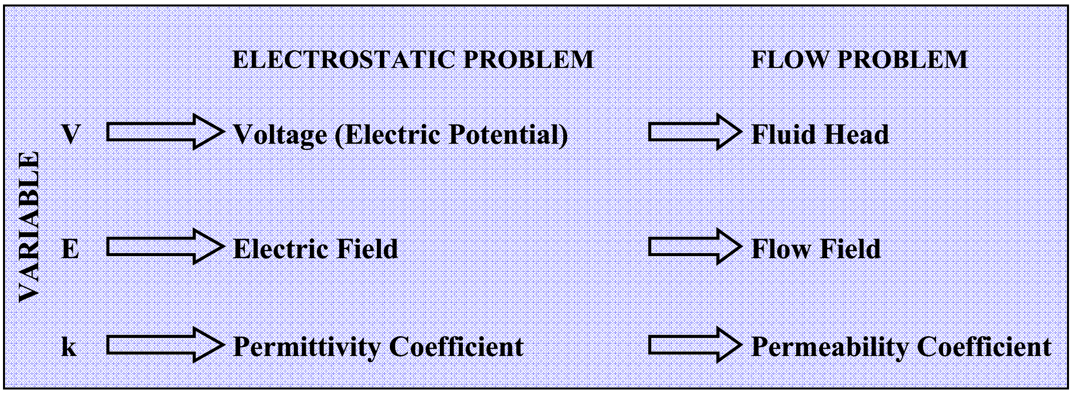

Figure 1 : Electric analogy of seepage problem

3 IMPLEMENTATION

3.1 GiD interface and Geoflow implementation

The GiD program makes the pre and post-process activities, however, when a calculation motor is developed, is necessary to generate an interface to keep in mind the models particularities. This process is made through six files explained next, shortly:

- ✓ **Geoflow.prb**: it provides the window to load the information of the problem general parameters.
- ✓ **Geoflow.cnd**: it informs GiD about the conditions imposed to the model (on lines and points): Voltage and Flow Density.
- ✓ **Geoflow.mat**: it provides information about the materials physical properties (permittivity).
- ✓ **Geoflow.bas**: it provides the format of file data which will be exchanged between GID and the code
- ✓ **Geoflow.bat**: it throws the calculation module. The GiD “Calculate” option executes this file.
- ✓ **Geoflow.exe**: it is the calculation motor developed in FORTRAN (it solves the problem for approach in finite differences)

3.2 Geoflow development

The motor is made of seven subroutines. The first one reads the data of the “*.dat” file. The second one organizes the nodes according to coordinates. This way, a reticular configuration is obtained. The third subroutine stores the coordinates matrix, voltages matrix and cases[1] matrix. Then, the Dirichlet conditions are applied (this step is previous to the use of the subroutine over-relaxation). In the fourth subroutine the over-relaxation method is implemented. This subroutine uses a cases selector, classified according to the quantity of

[1] There are four cases that define the form of calculation of each node of the finite differences mesh.

neighboring nodes and type of assigned contour conditions. When Neumann conditions are identified, fictitious nodes are used. In order to control the iterations, the approximate absolute error is calculated and it is compared with the tolerance specified by the user. Once satisfied the tolerance, the cycle of voltage calculation concludes. The fifth subroutine calculates the electric field using the voltage values. The sixth subroutine generates the "*.flavia.res" file. The voltage and field values that GiD uses to execute the post-process phase are stored in this file.

```
        DECLARATION OF VARIABLES
(1)     DATA READING FROM *.DAT FILE (THIS FILE IS GENERATED FOR GiD)
(2)     NODES CLASSIFICATION ACCORDING TO COORDINATES
(3)     NODES MATRIX, COORDINATES MATRIX, VOLTAGES MATRIX AND CASES MATRIX
        ASSEMBLING (THE CASES MATRIX IDENTIFY THE CALCULATION FORM OF EACH
        NODE)
(4)     OVER-RELAXATION METHOD ADAPTED TO IRREGULAR STEP MESHES
(5)     ELECTRIC FIELD CALCULATION
(6)     RESULTS WRITTING
        END
```

Table 1 : Geoflow chart

4 RESULTS, POST-PROCCES AND MATLAB PDETOOL COMPARISON

The figures 2, 3, 4, 5, 6, 7, 8 and 9 show the GID and MATLAB solution (potential contours and flow vectors) of two flow problems (a concrete dam and a sheet-pile wall).

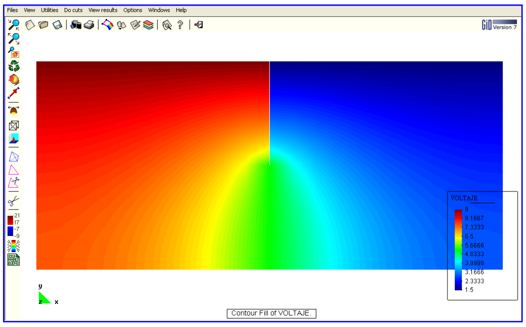

Figure 2 : GiD potential iso-contours (Sheet-pile Model)

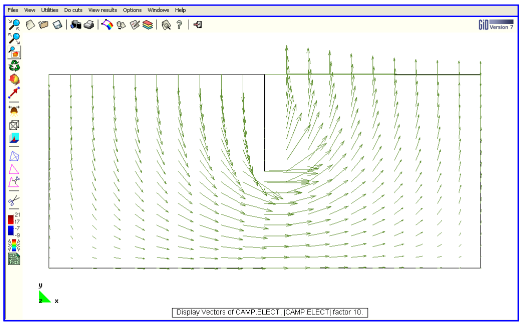

Figure 3 : GiD flow vectors (Sheet-pile Model)

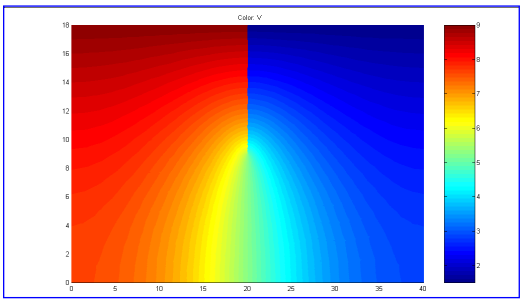

Figure 4 : Matlab potential iso-contours (Sheet-pile Model)

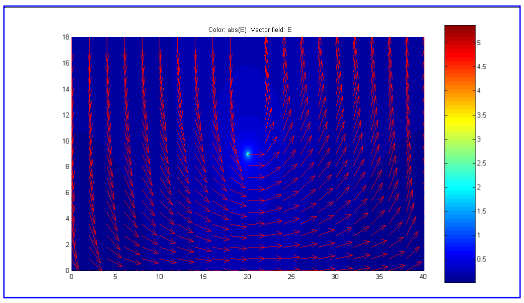

Figure 5 : Matlab flow vectors (Sheet-pile Model)

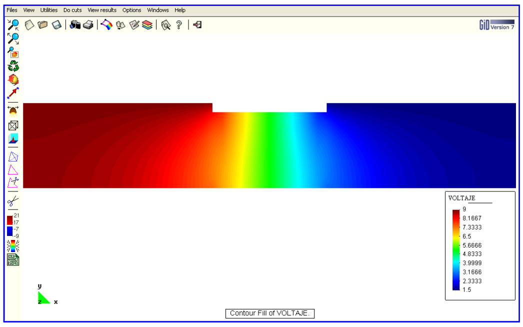

Figure 6 : GiD potential iso-contours (Concrete Dam)

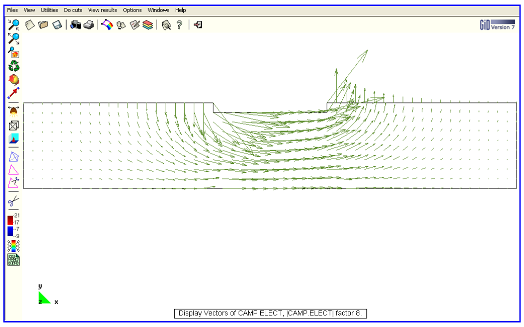

Figure 7 : GiD flow vectors (Concrete Dam)

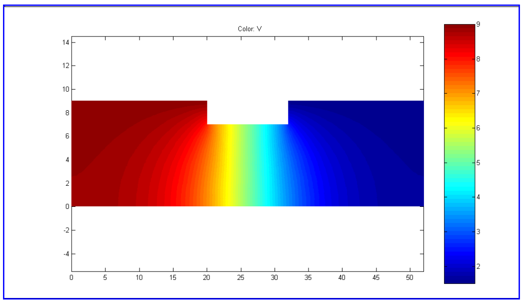

Figure 8 : Matlab potential iso-contours (Concrete Dam)

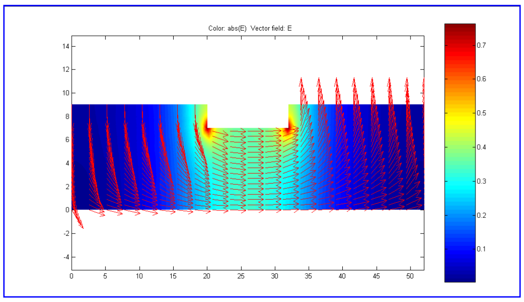

Figure 9 : Matlab flow vectors (Concrete Dam)

5 CONCLUSIONS

- The finite differences are advantageous, since it is a method of quick solution, the matrix obtained it is full and follow the model of grid. This way, a domain with a good number of nodes doesn't imply bigger storage for the matrix voltage.
- The over-relaxation factor accelerates the convergence, satisfactorily. Particularly, 1.60 is the best value that reduces the iterations number.
- The adaptation of the finite differences method to not uniform grids implemented in this application, it allows refining the mesh in interesting areas.
- The numerical methods are a great tool to the modeling engineering problems, the same thing that the computational tools developed for it, such it is the case of GiD, a pre and post-process platform excellent to model this problem type.

REFERENCES

[i] Geral, Weley, Análisis Numérico con Aplicaciones, Sexta Edición, 595-695
[ii] Mathews Jhon H., D. Fink Kurtis, Métodos Numéricos con Matlab, 557-582
[iii] Mora Javier, Oñate Eugenio, Canet Juan Miquel, Electromagnetismo, CIMNE 1999
[iv] Ribó R., GID User Manual, CIMNE, 1999
[v] Swanson Analysis Systems, Inc., “Ansys User Manual”, revision 5.0, Houston
[vi] Zárate F., Oñate E., CALTEP: Programa para el Cálculo Transitorio de la Ecuación de Poisson, CIMNE 1993
[vii] Zienkiewicz, O. C. y Taylor, R. L., “El Método de los Elementos Finitos”, Vol. 1, edit. Mc. Graw – Hill, 4ª edición, Barcelona, 1994.