CAMP: a configurable graphic user interface for three-dimensional flow and transport in variably saturated porous media

M.L. Clemente, E. Leonardi, G. Lecca & C. Gallo
CRS4 - Centre for Research, Development and Advanced Studies in Sardinia
via Nazario Sauro 10, 09123 Cagliari, Italy
E-Mail: {clem,ermy,giuditta,fender}@crs4.it

Abstract

The increasing complexity of the numerical codes describing flow and transport in groundwater makes the availability of a graphic user interface (GUI) very appealing. In this context, CAMP (Configurable Application Manager Program), a GUI written in Tcl/Tk object library language, is presented. CAMP features are described in detail including its portability on different architectures and the ease to adapt the interface to different two- and three-dimensional numerical codes. In particular, the interface integration to CODESA-3D (COupled DEnsity dependent flow and transport in variably SAturated 3-Dimensional porous media) code [1] is considered. CODESA-3D is a finite element code for simulating saltwater intrusion in coastal aquifers and groundwater brine transport. A three-dimensional test case is presented along with the description through the GUI of the mesh generation and refinement processes, the parameter setup, and the graphical post-processing of model outputs. Future developments of CAMP will also be discussed.

1 Introduction

Contamination of groundwater is a major concern in the management of water resources. Any addition of undesirable substances to ground-

water caused by human activities is considered to be contamination. Groundwater often spreads the effects of dumps and spills far beyond the site of the original contamination. Mathematical models can be effectively used to study the migration and the fate of contaminants and their effects on water resources, and to design and analyze remediation strategies. These models are solved numerically by means of complex computer codes which are often very difficult to handle for non-expert users.

In the last decade, an increasing effort has been devoted to providing numerical codes with user-interfaces aimed at simplifying the use the code itself. Generally, a user interface is built *ad hoc* for a specific code and is not reusable for others. In this context, a Configurable Application Manager Program (CAMP) is presented. CAMP is a graphic-user-interface aimed at interfacing subsurface flow and transport finite element codes with some public-domain or commercial visualization tools; "Configurable" because it is possible to adapt CAMP to virtually any numerical code by means of a configuration file. This file is loaded at start-up time, so that CAMP is already customized for the current code.

In this work CAMP is used for running simulations with the CODESA-3D code [1], a numerical simulator for modeling density-dependent flow and mass transport in porous media.

2 Coupled Flow and Transport: the CODESA-3D simulator

The CODESA-3D mathematical model is based on two governing equations: the groundwater and the solute mass conservation. The two equations are coupled through the Darcy velocity (flow $\rightarrow$ transport) and the solute concentration (transport $\rightarrow$ flow). The governing equations for a three-dimensional coupled system of flow of water in a variably saturated porous medium and transport of miscible salt can be written in the form

$$\sigma \frac{\partial \psi}{\partial t} = \nabla \cdot \left[K_s \frac{1 + \epsilon c}{1 + \epsilon' c} k_r \left(\nabla \psi + (1 + \epsilon c) \eta_z \right) \right] - \phi S_w \epsilon \frac{\partial c}{\partial t} + \frac{\rho}{\rho_o} q \qquad (1)$$

$$\mathbf{v} = - K_s \frac{1 + \epsilon c}{1 + \epsilon' c} K_r \left(\nabla \psi + (1 + \epsilon c) \eta_z \right) \qquad (2)$$

$$\phi \frac{\partial S_w c}{\partial t} = \nabla \cdot (D \nabla c) - \nabla \cdot (c \mathbf{v}) + q c^* + f \qquad (3)$$

where $\psi = p/(\rho_o g)$ is the equivalent freshwater pressure head, p is the pressure, ρ_o is the freshwater density, g is the gravitational constant, and z is the vertical coordinate directed upward, $\epsilon = (\rho_s - \rho_o)/\rho_o$ is the density ratio, where ρ_s is the maximum solution density, and $\alpha = (\mu_s - \mu_o)/\mu_o$ is the viscosity ratio, where μ_s and μ_0 are the maximum and freshwater viscosity, respectively; ∇ is the gradient operator, K_s is the saturated hydraulic conductivity tensor at the reference density, $k_r(\psi)$ is the relative conductivity, η_z is a vector equal to zero in its x and y components and 1 in its z component, $\sigma(\psi, c)$ is the general storage term or overall storage coefficient, t is time, ϕ is the porosity, $S_w(\psi)$ is the water saturation, q is the injected (positive)/extracted (negative) volumetric flow rate, $\mathbf{v}$ is the Darcy velocity vector, D is the dispersion tensor, c^* is the normalized concentration of salt in the injected/extracted fluid, and f is the volumetric rate of injected (positive)/extracted (negative) solute that does not affect the velocity field. Initial conditions and Dirichlet, Neumann, or Cauchy boundary conditions are added to complete the mathematical formulation of the flow and transport problem.

Equations (1-3) are approximated in space using the Galerkin finite-element method with linear basis function, and in time with a weighted finite difference scheme. These discretizations yield a nonlinear system which is solved by applying the Newton-Raphson linearization iterative scheme (for solving the nonlinear unsaturated flow equation) and the Picard (or fixed-point) iterative scheme (for the global coupling between the flow and the transport equation). The linear systems obtained after the linearization are solved using an iterative linear solver for nonsymmetric sparse systems [2].

Four main parts can be recognized in the code: input (mesh informations, soil and solute properties, initial and boundary conditions, simulation parameters), build-up of topological data structures (nodes reordering, element areas neighbors edges, etc.), flow equation solution (which yields the pressure and velocity fields), and transport equation solution (which yields the concentration field). CODESA-3D can handle complex features including: unstructured meshes, heterogeneous soils, pumping wells, localized and distributed immissions, temporally and spatially variable boundary conditions, etc.

3 Tcl/Tk programming library for CAMP

CAMP is written using the public-domain programming library **Tcl/Tk** [3, 4]. **Tcl** is a string-oriented language which provides high-level generic programming facilities for handling variables, procedures, conditional and repetitive loops, etc. Programming in **Tcl** requires no variable or subroutine declarations. **Tk** is a toolkit for the X Window System which contains some preselected objects for handling windows, like: buttons linked to commands, entries, listboxes, scrollbars, text widgets, canvases, etc. [3, 4]. **Tcl/Tk** libraries can be easily interfaced with other libraries and programming languages (such as C) and allow an easy management of both graphical and system functions.

4 How CAMP works

In this section CAMP capabilities are illustrated by means of a rather simple test case: the fate of a plume of dense contaminant in a saturated aquifer. Thus, it has been considered a three-dimensional domain 100 m long, 100 m wide and 30 m high, discretized into 758 triangular elements in the x-y plane by means of an unstructured mesh. The 2D grid has been replicated along the z direction to obtain the 2940-nodes/13644-tetrahedral element 3D-mesh. An impermeable block of clay was assumed to be present in the aquifer. This block was considered to span over all the depth of the aquifer and it was removed from the global mesh on each layer: no-flow boundary conditions were assumed in the corresponding internal boundary. The domain was assumed to be isotropic and heterogeneous: two horizontal zones of different hydraulic conductivity were identified, the first one, from 30 to 15 m, with a value of 0.1 m/day, and the second one, from 15 m to the domain bottom, with a value of 1.0 m/day. An initial condition of uniform potential head distribution was assumed over the whole domain. A Darcy flux on the x-z plane was obtained by imposing a prescribed potential head of $h = 50$ m (Dirichlet flow boundary condition) on the left side of domain; moreover, a constant infiltration rate of 4.32×10^{-5} m/day was imposed on the top surface. The domain was assumed to be initially free of contaminant, i.e., $c(x, t = 0) = 0$, and then the contaminant entered the aquifer from a salt deposit located in a central zone at the top of the

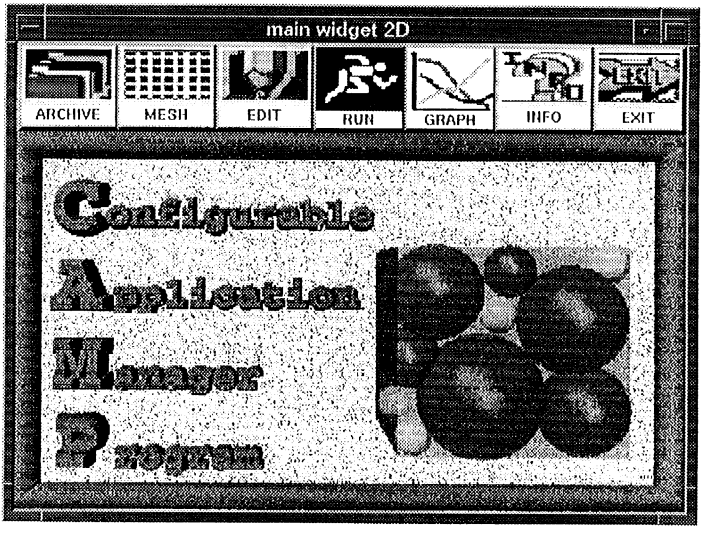


Figure 1: Main window of CAMP program.

domain (Dirichlet transport boundary condition, $c(x^*, t) = 1$, with x^* belonging to the selected zone). The density ratio was $\epsilon = 0.07$, the viscosity ratio and the diffusion coefficient were set to zero, and the dispersivity coefficients were imposed $\alpha_L = \alpha_T = 10$ m.

Figure 1 shows CAMP main window with some of the included functionalities already present: ARCHIVE, MESH, EDIT, RUN, GRAPH, INFO, and EXIT. Each of these labels activates, when clicked, a menu in which a list of functions are made available. Thus, for example, ARCHIVE activates the window for handling the archive of a previous set of simulations and for storing new test cases I/O compressed data; MESH activates functionalities related to mesh generation, visualization, and data formatting for the numerical code. EDIT enables the editing of input files (such as initial and boundary condition, soil characteristics, and other simulation parameters; RUN is devoted to run, stop, and change the priority status of the running code; and GRAPH postprocess and visualize model results in a two and three dimensional fashion.

The MESH window contains all the functions for creating the computational mesh. The boundary of the computational domain can be drawn clicking the points (points can be added, cancelled, moved, etc.) with the mouse, as shown in Figure 2; then the file *[filename].poly* is created and represents the starting step for the tri-

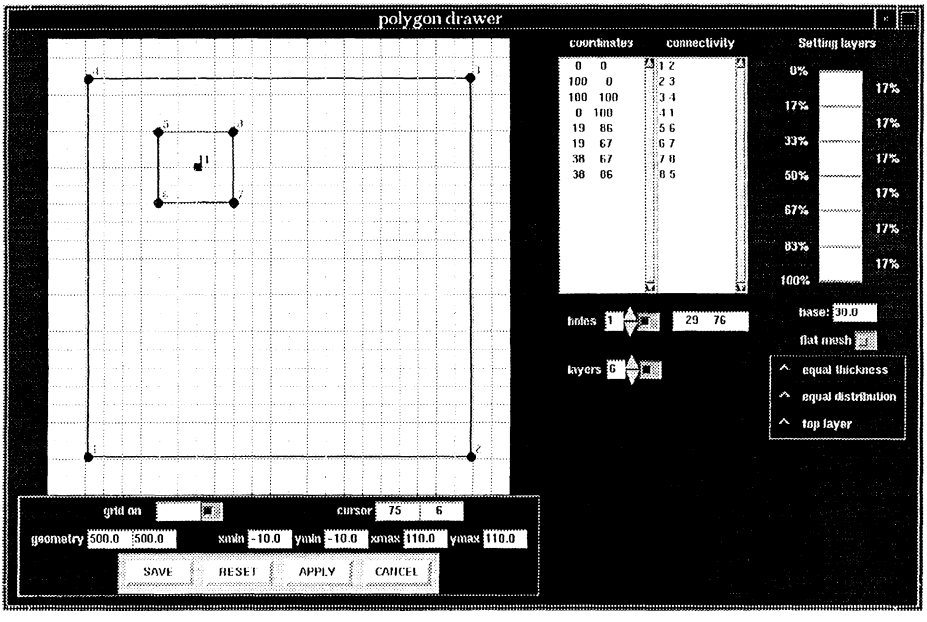


Figure 2: Polygon drawer window.

angulation. The number of layers and relative thicknesses for the generation of the 3D computational domain can be easily imposed.

The triangulation is obtained using the Triangle command panel, and is visualized by clicking the *Show_mesh* button , as illustrated in Figure 3. By clicking the *Save_Mesh* button, the 2D grid is replicated the desired number of times and saved in file *[filename].grid*; now it is also possible to see the new mesh, as shown in Figure 4.

After generating the grid file, it is possible to run the CODESA-3D code by clicking the *RUN* button on the main window (see Figure 1). The window GRAPH enables the user to select some graphical tools (such as PLOTMTV, XMGR) for visualizing results generated by the simulation which has been run. Concentration, potential head, pressure, and velocity profiles can easily be shown at any desired time step and on each layer of the computational domain, using the frame selector panel (see Figure 5).

For example, Figure 6 represents the concentration profiles at time $T = 1000$ days on the first, and third layer, while in Figure 7 the corresponding slices are plotted at time $T = 4000$ days, when the PLOTMTV visualizer is selected.

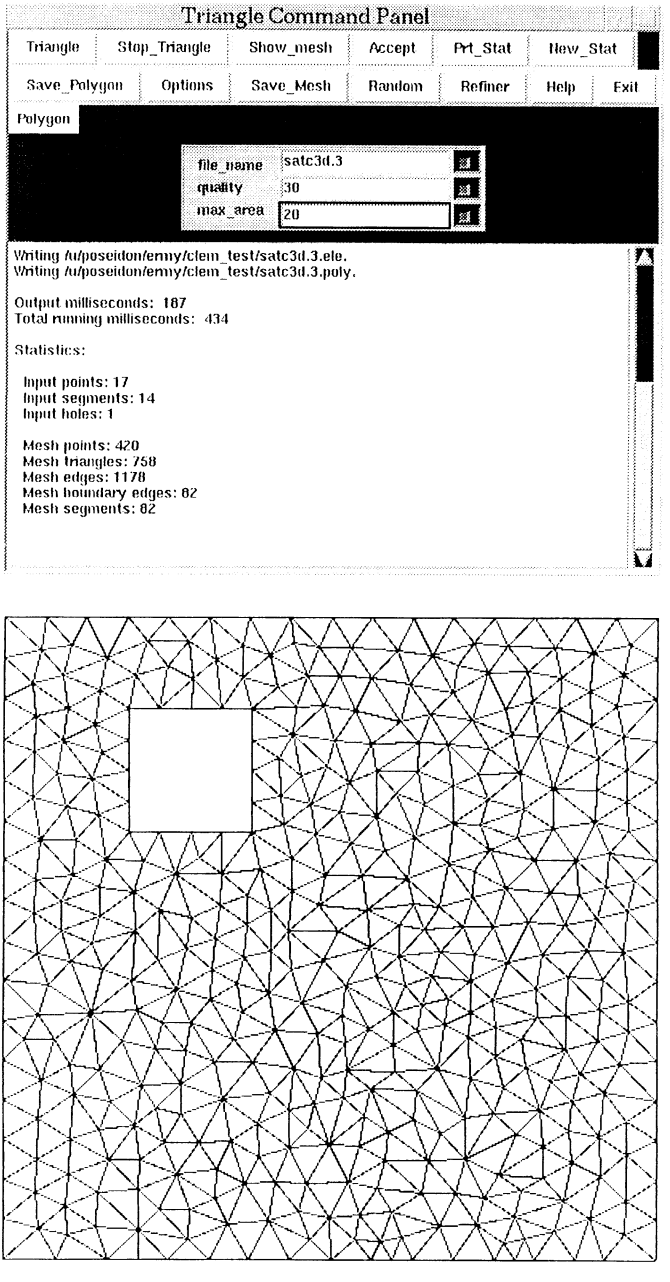


Figure 3: Triangle command window (above), unstructured 2D mesh (below).

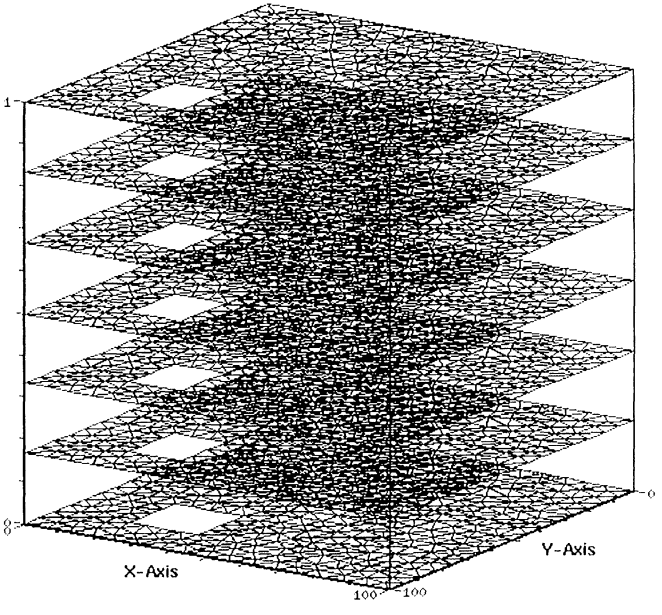


Figure 4: 3D mesh.

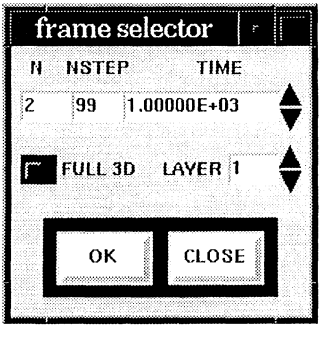


Figure 5: Frame selector.

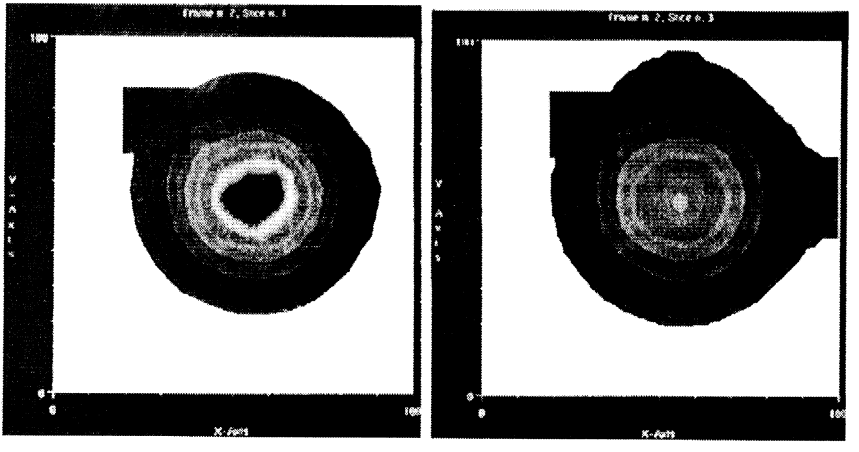


Figure 6: Concentration profiles on the 1^{st} layer (left) and 3^{rd} layer (right) at T=1000 days.

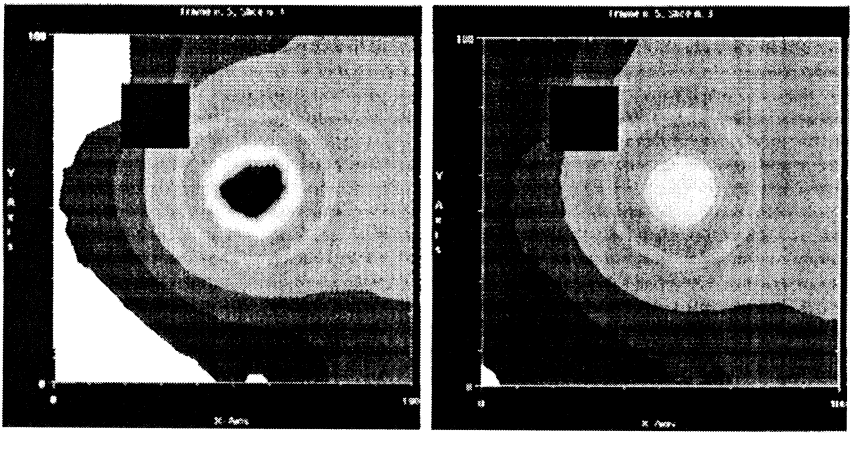


Figure 7: Concentration profiles on the 1^{st} layer (left) and 3^{rd} layer (right) at T=4000 days.

5 Final remarks

The CAMP user interface has been presented. CAMP capabilities have been described and the software package has been applied to the density-dependent CODESA-3D finite element simulator. Configurability is one of the most important features of CAMP and its application to a wide variety of numerical codes is relatively straightforward. CAMP customization is obtained by means of a configuration file in which paths to executables, graphic tools, archive directories, and working directories are to be specified.

Currently CAMP can handle files deriving from previous simulations or from scratch. Future developments will include the possibility to import/export I/O data from and to GIS systems.

Acknowledgments

This work has been financially supported by the Sardinia Regional Authorities and by the European AVICENNE Research Programme (contract AVI-CT93-2-073).

References

[1] G. Gambolati, M. Putti, and C. Paniconi. Three-dimensional model of coupled density-dependent flow and miscible salt transport in groundwater. In J. Bear, A. H.-D. Cheng, I. Herrera, S. Sorek, and D. Ouazar, editors, *Seawater Intrusion in Coastal Aquifers: Concepts, Methods, and Practices*. Kluwer Academic, Dordrecht, Holland, 1998 (to appear).

[2] H. A. Van der Vorst. *Parallel iterative solution methods for linear systems arising from discretized PDE's.* Proc. of the AGARD-VKI Lecture Series on Parallel Computing in Computational Fluid Dynamics, 1995.

[3] J. K. Ousterhout. *Tcl and the Tk Toolkit.* Addison-Welsey professional computing series, 1994.

[4] B. B. Welch. *Practical Programming in Tcl and Tk.* Prentice Hall PTR, 1995.