

## Jump Dominance on the Contaminant Transport Residual Error Estimator\*

J.P. MARTINS<sup>1\*\*</sup>, A. FIRMIANO<sup>2</sup> and E. WENDLAND<sup>1</sup>

Received on November 10, 2013 / Accepted on April 9, 2014

**ABSTRACT.** Considering the fact that the transport of contaminants occur in small advection regime, a residual estimator is used to evaluate the parabolic equation that describes the phenomena of advection-diffusion-reaction in the saturated porous medium. The correspondent numerical solution is obtained by the finite element method using the  $\theta$  A-stable scheme and a Python code. The residual error estimator considers its component parts and enables analysis and comparisons of contributions to residual error. This analysis considers a problem sequence with a different number of elements in computational mesh. As a result of numerical simulations, there is a dominance of the jump residuals compared to other residual estimates and this dominance increases with both, the growth of elements number in the computational mesh and with time. Furthermore, the considered problem requires additional effort for the calculation of contributions associated with the  $L^2$  projection of the contaminant source function on the finite element space.

**Keywords:** residual error estimator, advection-diffusion-reaction equation, small advection regime, finite elements, contaminant transport in porous media.

### 1 INTRODUCTION

Analytical solutions are far from encompassing the variety of phenomena present in contaminant transport and numerical methods are required to obtain an approximate solution of the corresponding mathematical model. Moreover, the reliability of computational methods depends on the discretization technique and the quality of the finite element mesh adopted. Although estimates of *a priori* are available, *a posteriori* estimates are fundamental for practical problems involving finite element method [13]. Once the numerical result is obtained, the *a posteriori* error estimator can be used to provide general or specific information about the quality of the numerical solution [3].

---

\*Paper presented at CMAC-SE 2013.

\*\*Corresponding author: João Paulo Martins dos Santos

<sup>1</sup>Department of Hydraulics and Sanitary Engineering, EESC – USP, Av. Trabalhador Sancarlense, 400, 13566-590 São Carlos, SP, Brazil. E-mails: jp2@usp.br; ew@sc.usp.br

<sup>2</sup>Air Force Academy. E-mail: alessandroafj@afa.aer.mil.br

**2 THE MODEL PROBLEM FOR THE CONTAMINANT TRANSPORT**

The linear second order parabolic equation is used to describe the contaminant transport in a domain  $\Omega \subset \mathbb{R}^2$ , with general unknown concentration solution  $C = C(x, y, t)$ , data functions  $\mathbf{D}, \mathbf{v}, \lambda, f, g, C_0$  and final time  $t_{final}$  is arbitrary, but is kept constant.

The data are real valued functions that may depend on space and time while the initial condition depends only on space [17]. Here, the equation describes the phenomena of advection-dispersion-reaction (ADR model) in a saturated porous media and, according to [1], can be presented by:

$$\begin{aligned}
 \partial_t C - \mathbf{div}(\mathbf{D}\nabla C) + \mathbf{v} \cdot \nabla C + \lambda C &= f && \text{in } \Omega \times (0, t_{final}] \\
 C &= 0 && \text{in } \Gamma_D \times (0, t_{final}] \\
 \mathbf{n} \cdot \mathbf{D}\nabla C &= g && \text{in } \Gamma_N \times (0, t_{final}] \\
 C &= C_0 && \text{in } \Omega \text{ for } t = 0
 \end{aligned} \tag{2.1}$$

where  $\Omega \subset \mathbb{R}^2$  is a polygonal cross-section with a Lipschitz boundary  $\Gamma$  consisting of two disjoint parts  $\Gamma_D$  and  $\Gamma_N$ . The space dependent function  $C = C_0$  in  $\Omega$  for  $t = 0$  is the initial condition while

$$C = C_D \text{ in } \Gamma_D \times (0, t_{final}] \text{ and } \mathbf{n} \cdot \mathbf{D}\nabla C = g \text{ in } \Gamma_N \times (0, t_{final}]$$

are the Dirichlet and Neumann boundary conditions. Assuming that the data satisfy the additional conditions [20]:

**P1** – The dispersion  $\mathbf{D} = \mathbf{D}(x, y, t)$  is a continuously differentiable matrix-valued function and symmetric, uniformly positive definite and uniformly isotropic. Formally,

$$\varepsilon = \inf_{0 \leq t \leq t_{final}, (x,y) \in \Omega} \min_{\mathbb{R}^2 - \{0\}} \frac{z^T \mathbf{D}(x, y, t) z}{z^T z} > 0, \tag{2.2}$$

and

$$\kappa = \varepsilon^{-1} \sup_{0 \leq t \leq t_{final}, (x,y) \in \Omega} \max_{z \in \mathbb{R}^2 - \{0\}} \frac{z^T \mathbf{D}(x, y, t) z}{z^T z} \tag{2.3}$$

is moderate size constant.

**P2** – The velocity  $\mathbf{v} = \mathbf{v}(x, y, t) = (v_x(x, y, t), v_y(x, y, t))$  is a continuously differentiable vector-field and scaled such that

$$\sup_{0 \leq t \leq t_{final}, (x,y) \in \Omega} |\mathbf{v}(x, y, t)| \leq 1.$$

**P3** – The reaction  $\lambda = \lambda(x, y, t)$  is a continuous non-negative scalar function.

**P4** – There is a constant  $\beta$  such that  $\lambda - (1/2)\mathbf{div}(\mathbf{v}) \geq \beta$  for almost all  $(x, y) \in \Omega$  and  $0 \leq t \leq t_{final}$ . Moreover there is and a constant  $c_b \geq 0$  of moderate size such that

$$\sup_{0 \leq t \leq t_{final}, (x,y) \in \Omega} |\lambda(x, y, t)| \leq c_b \beta.$$

**P5** – The Dirichlet boundary  $\Gamma_D$  has positive measure  $(d - 1)$ -dimensional and includes the inflow boundary

$$\bigcup_{0 < t \leq t_{final}} \{(x, y) \in \Gamma : \mathbf{v} \cdot \mathbf{n}(x, y) < 0\}.$$

With these additional assumptions, the contaminant transport regime can be classified into:

- dominant dispersion:

$$\sup_{0 \leq t \leq t_{final}, (x,y) \in \Omega} |\mathbf{v}(x, y, t)| \leq c_c \varepsilon \quad \text{and} \quad \beta \leq c' \varepsilon; \tag{2.4}$$

with constants of moderate size;

- regime of dominant reaction:

$$\sup_{0 \leq t \leq t_{final}, (x,y) \in \Omega} |\mathbf{v}(x, y, t)| \leq c_c \varepsilon \quad \text{and} \quad \beta \gg \varepsilon; \tag{2.5}$$

with a constant  $c_c$  of moderate size;

- regime of dominant advection:  $c_v \gg \varepsilon$ .

A detailed discussion of these elements can be found in D. Praetorius [17] which is an extension of the results from Verfürth [20].

To derive the space-time discretization of (2.1), we consider a test function  $w \in H_D^1(\Omega)$  where  $H_D^1(\Omega)$  denotes the subspace of the Sobolev space  $H^1(\Omega) = W^{1,2}(\Omega)$ , with functions that vanish on the Dirichlet boundary  $\Gamma_D$ . Multiply equation (2.1) by a test function and use integration by parts to derive the weak form

$$\int_{\Omega} (\partial_t C w + \nabla C \cdot \mathbf{D} \nabla w + \mathbf{v} \cdot \nabla C w + \lambda C w) d\Omega = \int_{\Omega} f w d\Omega + \int_{\Gamma_N} g w dS. \tag{2.6}$$

Next, consider the partition  $I = \{[t_{n-1}, t_n] : 1 \leq n \leq N_I\}$  of the time interval  $[0, t_{final}]$  such that  $0 = t_0 < t_1 \dots < t_{N_I} = t_{final}$ .

For every  $n$  with  $1 \leq n \leq N_I$  denote by  $I_n = [t_{n-1}, t_n]$  the  $n$ -th subinterval and  $\tau_n = t_n - t_{n-1}$  its length.

With every intermediate time  $t_n$ ,  $0 \leq n \leq N_I$  associate an admissible, affine equivalent, shape regular partition  $T_n$  of  $\Omega$  and a corresponding finite element space  $X_n$ . In addition, the partitions  $I$  and  $T_n$  and the spaces  $X_n$  must satisfy the following assumptions:

- $\Omega \cup \Gamma$  is the union of all elements in  $T_n$ ;
- Affine equivalence, Admissibility and Shape-regularity;
- Non-degeneracy, transition condition and degree condition.

With a time discretization parameter  $\theta \in [\frac{1}{2}, 1]$  and the abbreviations  $C^n = C(x, y, t_n)$ ,  $\mathbf{D}^n = \mathbf{D}(x, y, t_n)$ ,  $\mathbf{v}^n = \mathbf{v}(x, y, t_n)$ ,  $\lambda^n = \lambda(x, y, t_n)$ ,  $f^n = f(x, y, t_n)$ ,  $g^n = g(x, y, t_n)$  the finite element approximation with  $\theta$ -A-stable-scheme is obtained by replacing the approximations in the weak form of the parabolic problem and is given by

$$\begin{aligned}
 &\text{Find } C^n \in X_n, 0 \leq n \leq N_I, \text{ such that } C^0 = \pi_0 C_0 \text{ and for } n = 1, 2, \dots, N_I \\
 &\int_{\Omega} \frac{1}{\tau_n} (C^n - C^{n-1}) w_n d\Omega + \int_{\Omega} (\theta \nabla C^n + (1 - \theta) \nabla C^{n-1}) \cdot D^n \nabla w_n d\Omega \\
 &+ \int_{\Omega} \mathbf{v}^n \cdot \nabla (\theta C^n + (1 - \theta) C^{n-1}) w_n d\Omega + \int_{\Omega} \lambda^n (\theta C^n + (1 - \theta) C^{n-1}) w_n d\Omega \tag{2.7} \\
 &= \int_{\Omega} f^n + (1 - \theta) f^{n-1} w_n d\Omega + \int_{\Gamma_N} (\theta g^n + (1 - \theta) g^{n-1}) w_n dS \text{ for all } w_n \in X_n.
 \end{aligned}$$

where  $C^n = C^n_{T_n}$  and  $w_n = w_{T_n}$ .

In particular,  $\theta = 1/2$  gives the *Crank-Nicolson scheme* and  $\theta = 1$  gives *implicit Euler scheme* [20]. Thus, finite element formulation (2.7) can be rewritten as  $a(C^n_n, w_n) = L(w_n)$ . The term  $a(C^n_n, w_n)$  is called the bilinear form and is defined by expression (2.8)

$$\begin{aligned}
 a(C^n, w_n) &= \int_{\Omega} \frac{1}{\tau_n} C^n w_n d\Omega + \int_{\Omega} (\theta \nabla C^n) \cdot D^n \nabla w_n d\Omega \\
 &+ \int_{\Omega} \mathbf{v}^n \cdot \nabla (\theta C^n) w_n d\Omega + \int_{\Omega} \lambda^n (\theta C^n) w_n d\Omega \tag{2.8} \\
 &+ \int_{\Gamma_N} \theta g^n w_n dS,
 \end{aligned}$$

while the term  $L(w_n)$  is called linear form and defined by expression (2.9)

$$\begin{aligned}
 L(w_n) &= \int_{\Omega} \frac{1}{\tau_n} C^{n-1} w_n d\Omega + \int_{\Omega} ((\theta - 1) \nabla C^{n-1}) \cdot D^n \nabla w_n d\Omega \\
 &+ \int_{\Omega} \mathbf{v}^n \cdot \nabla ((\theta - 1) C^{n-1}) w_n d\Omega + \int_{\Omega} \lambda^n ((\theta - 1) C^{n-1}) w_n d\Omega \tag{2.9} \\
 &+ \int_{\Omega} (\theta f^n + (1 - \theta) f^{n-1}) w_n d\Omega + \int_{\Gamma_N} (1 - \theta) g^{n-1} w_n dS
 \end{aligned}$$

Although variational formulation (2.7) is the key in finite element method, forms (2.8) and (2.9) are essential for implementation using the FEniCS Project methodology [4]. These forms generate the linear system to be solved in each step of the simulation process. The solution in  $(n - 1)$ -th time step and the solution on the  $n$ -th time step are used to provide an error measure.

A detailed description of the residual estimator and assumptions presented here are in references [20, 19, 16, 17]. For finite elements method, a detailed discussion can be found in reference [2].

In the residual method, an element residuals  $R_K$  is defined by:

$$R_K = f_I - \frac{1}{\tau_n} (C^n - C^{n-1}) + \text{div}(\mathbf{D}^n \nabla(\theta C^n + (1 - \theta)C^{n-1})) - \mathbf{v}^n \cdot \nabla(\theta C^n + (1 - \theta)C^{n-1}) - \lambda^n(\theta C^n + (1 - \theta)C^{n-1}) \tag{2.10}$$

while an edge or face residual  $R_E$  is defined by:

$$R_E = \begin{cases} -\mathbf{J}_E(\mathbf{n}_E \cdot \mathbf{D}^n \nabla(\theta C^n + (1 - \theta)C^{n-1})) & \text{if } E \not\subseteq \Gamma \\ g_I - \mathbf{n}_E \cdot \mathbf{D}^n \nabla(\theta C^n + (1 - \theta)C^{n-1}) & \text{if } E \subset \Gamma_N \\ 0 & \text{if } E \subset \Gamma_D \end{cases} \tag{2.11}$$

where  $\mathbf{J}$  is the jump operator,  $f_I(x, y, t) = \pi_n(\theta f(x, y, t_n) + (1 - \theta)f(x, y, t_{n-1}))$  and  $g_I(x, y, t) = \pi_n(\theta g(x, y, t_n) + (1 - \theta)g(x, y, t_{n-1}))$  the projection functions on the finite element space  $X_n$  [20]. According to Verfürth [20], if the transport regime is of small advection then the residual estimator is given by equation (2.12)

$$\hat{\eta}_I = \left\{ \|C_0 - \pi_0 C_0\|_{L^2(\Omega)}^2 + \sum_{n=1}^{N_I} \tau_n \left[ (\eta^n)^2 + \|C^n - C^{n-1}\|^2 \right] \right\}^{\frac{1}{2}} \tag{2.12}$$

where

$$(\eta_{T_n}^n)^2 = \sum_K \alpha_K^2 \|R_K\|_{L^2(K)}^2 + \sum_E \varepsilon^{-\frac{1}{2}} \alpha_E \|R_E\|_{L^2(E)}^2, \tag{2.13}$$

and

$$\|C^n - C^{n-1}\|^2 = \varepsilon \|\nabla(C^n - C^{n-1})\|_{L^2(\Omega)}^2 + \beta \|C^n - C^{n-1}\|_{L^2(\Omega)}^2, \tag{2.14}$$

with weighting factors  $\alpha_S = \min\{h_S \varepsilon^{-\frac{1}{2}}, -\beta^{\frac{1}{2}}\}$ , where  $S = \{K, E\}$  is an element or an edge/face and  $\beta^{-\frac{1}{2}} = \infty$  if  $\beta = 0$ .

### 3 IMPLEMENTATION

The Python numerical code considers the available methodology for the FEniCS Project. A complete description of this project can be found in references [8, 9, 10, 11, 12, 14] or at <http://fenicsproject.org> [4]. For graphical display of numerical solutions, Matplotlib/Scitools was used [15, 18]. Using Python language, the contaminant transport equation is implemented using the available tools from [4]. These tools provide conditions for setting the transport of contaminants directly through the use of bilinear and linear forms. Furthermore, mesh, initial conditions, dispersions, velocity field, projections and boundary conditions are defined using the available classes/tools described in documentation [4]. For example, `mesh = UnitSquareMesh( $n_x, n_y, 'crossed'$ )` gives a mesh with  $n_x, n_y$  triangular elements in each coordinate direction and *crossed* orientation. Reference [4] is a complete description of the available classes and tools.

After the solution in the  $n$ -th step is obtained, the residual error estimates (2.13) are implemented through the summation of the quantities defined in (2.10) and (2.11). Formally,

$$(\eta^n_{T_n})^2 = \underbrace{\sum_K \alpha_K^2 \|R_K\|_{L^2(K)}^2}_{EC^n} + \underbrace{\sum_{E \notin \Gamma} \varepsilon^{-\frac{1}{2}} \alpha_E \|R_E\|_{L^2(E)}^2}_{JC^n} + \underbrace{\sum_{E \subseteq \Gamma_N} \varepsilon^{-\frac{1}{2}} \alpha_E \|R_E\|_{L^2(E)}^2}_{BC^n}. \quad (3.1)$$

According to the designations adopted by Verfürth [20] and D. Praetorius [17], components  $EC^n$ ,  $JC^n$ ,  $BC^n$ ,  $\| \|C^n - C^{n-1}\| \|^2$  will be called, respectively: element, jump, boundary and time contributions for the  $n$ -th time step. These quantities allow us to rewrite equation (2.12) as a sum of spatial and temporal contributions, obtained in each step, weighted by time step. Formally,

$$\hat{\eta}_I = \left\{ \|C_0 - \pi_0 C_0\|_{L(\Omega)}^2 + \sum_{n=1}^{N_I} \tau_n (ES^n)^2 \right\}^{\frac{1}{2}}. \quad (3.2)$$

From equation (3.2), the amount  $(ES^n)^2 = (\eta^n)^2 + \| \|C^n - C^{n-1}\| \|^2$  may be regarded as the residual error at each time step.

The code that implements the element, jump, boundary and time contributions follows an example available in [6]. This allows the calculation of  $\eta_I$  and enables comparisons of components or individual evaluation of contributions. This paper considers:

- Jump contribution versus element contribution  $R^n_{Jump/Element}$ , defined by

$$R^n_{Jump/Element} = \frac{JC^n}{EC^n} = \frac{\sum_{E \notin \Gamma} \varepsilon^{-\frac{1}{2}} \alpha_E \|R_E\|_{L^2(E)}^2}{\sum_K \alpha_K^2 \|R_K\|_{L^2(K)}^2} \quad (3.3)$$

which provides information on the magnitude of the jumps on the elements;

- Time contributions weighted by time time step, defined by  $\tau_n \| \|C^n - C^{n-1}\| \|^2$  which provides information on time residual error.

### 4 RESULTS AND DISCUSSION

With an adaptation of a problem described in [7] and [17], this paper implements the contaminant transport equation in a two-dimensional rectangular domain  $\Omega$  defined by points  $\mathbf{x} = (x, y) \in \mathbb{R}^2$ . A spatial adaptation of that problem consider  $\Omega$  defined by points  $(x_0, y_0) = (0, 0)$ ,  $(x_1, y_0) = (80, 0)$ ,  $(x_1, y_1) = (80, 40)$  and  $(x_0, y_1) = (0, 40)$ , initial condition  $C_0 = C(x, y, 0) = 0$  and contaminant source  $f$  defined by

$$f(x, y, t) = \begin{cases} 1 & \text{if } 9.375 \leq x \leq 1.625, \quad 19.375 \leq y \leq 20.625 \\ 0 & \text{otherwise} \end{cases}. \quad (4.1)$$

The physical parameters are the velocity field  $\mathbf{v} = (v_x, v_y) = (0.864, 0)\frac{m}{d}$ , the dispersion  $D = \alpha_D v_x I_{2 \times 2}$  with  $I_{2 \times 2}$  the identity matrix of order two and  $\alpha_D = 0.05m$  the diffusivity. Neumann boundary condition is defined, using bilinear and linear forms, by  $g = \mathbf{n} \cdot \mathbf{D}\nabla C$  on  $\Gamma_N = \{x_1\} \times (y_0, y_1)$  while Dirichlet boundaries are defined, using the available class, by  $C = 0$  on  $\Gamma_D = \Gamma \setminus \Gamma_N$ .

For all  $t_n, 1 \leq n \leq N_I$ , the numerical result of contaminant transport provides the distribution function  $C^n \in \mathbb{R}^2$ , which describes the contaminant concentration at each point of  $\Omega$ . Figure (1) presents the numerical solution for  $t_{70} = 70days, \tau_n = 1.0day$ , lagrangian functions of order two,  $\theta = 1.0$  and finite element mesh with  $n_x = 200 = 2n_y$  triangular elements in each direction and *left/rigth* orientation.

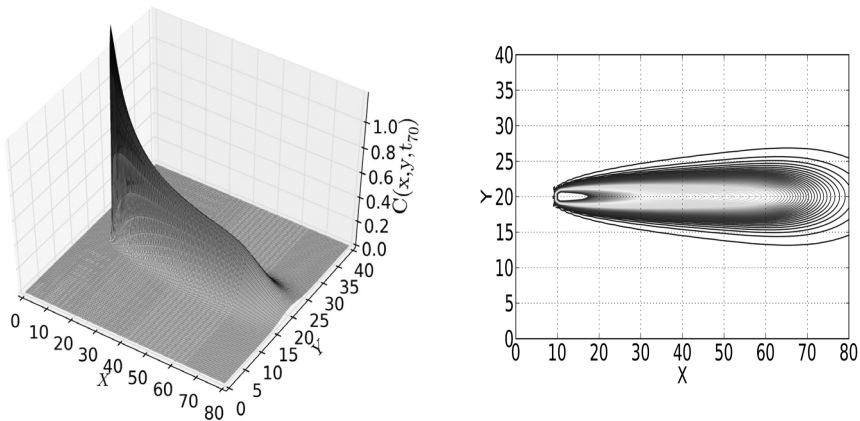


Figure 1: Numerical solution and level curves for  $t_{70} = 70days$  with  $\tau_n = 1.0day$ , lagrangian functions of order two,  $\theta = 1.0$  and finite element mesh with  $n_x = 200 = 2n_y$  triangular elements in each direction and *left/rigth* orientation.

For residual estimates, the conditions **P1-P5** were verified and contaminant transport was classified in advection dominated [17]. However, the regime is the small advection since  $C_c = v/\epsilon$  is a constant of moderate size [5].

The numerical simulations consider finite element meshes with  $n_x = 2n_y$  triangular elements in each coordinate direction with *left/right* orientation,  $\tau_n = 0.50, t_{final} = 100.0days, \theta = 1.0$  and lagrangian functions of order two. Figures (2)-a and (2)-b presents the results for temporal estimates and for ratio  $R_{Jump/Element}^n$  for a sequence of problems with  $n_x = [100, 200, 300, 400]$ .

Numerical results provide that temporal residual decreases with time and as the number of elements in mesh increases. On the other hand,  $R_{Jump/Element}^n$  increases with time and as the number of elements in mesh increases. The increasing in  $R_{Jump/Element}^n$  with time is due to the contaminant front advances, which, in turn, reflects in the dominance of jump contributions under element contributions already in the coarse mesh. This dominance characterizes the jump contributions as the most relevant part of  $\eta_I$  because  $\tau_n |||C^n - C^{n-1}|||^2 \approx 0$  and  $g = \mathbf{n} \cdot \mathbf{D}\nabla C$  is, by definition, such that  $BC^n = 0$ .

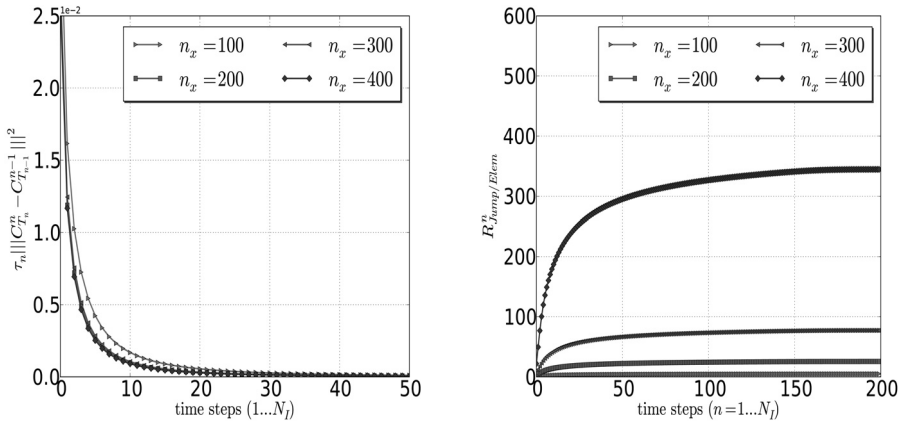


Figure 2: Temporal contributions  $\tau_n |||C^n - C^{n-1}|||^2$  and the ratio  $R_{Jump/Element}^n = JC^n/EC^n$  for  $\tau_n = 0.50$ ,  $t_{final} = 100days$ ,  $\theta = 1.0$ , lagrangian functions of order two,  $n_x = 2n_y = [100, 200, 300, 400]$ .

With the results,  $\eta_I$  can be approximated by

$$\hat{\eta}_I = \left\{ |||C_0 - \pi_0 C_0|||_{L_\Omega}^2 + \sum_{n=1}^{N_I} \tau_n (ES^n)^2 \right\}^{\frac{1}{2}} = \left\{ \sum_{n=1}^{N_I} \tau_n (JC^n)^2 \right\}^{\frac{1}{2}} \tag{4.2}$$

where  $|||C^0 - \pi_0 C_0|||_{L_\Omega}^2 = 0$  and  $(ES^n)^2 = (\eta^n)^2 + |||C^n - C^{n-1}|||^2 \approx EC^n + JC^n \approx JC^n$ .

This results and the approximation for  $\eta_I$  reveals that jump error is not a negligible quantity and complements the results obtained by Firmiano [5]. In fact the jump residual can be the most important part of the residual estimates.

### 5 CONCLUSION

The presence of residual estimates enables an analysis of the numerical solution, which provides information about the quality of the numerical results. Error estimator partition allows us to analyze and compare the component parts of residual error. This provides a better understanding of residual behavior for changing number of elements in mesh or time step. As a result of simulations, jump dominance manifests for all adopted meshes, which is due to the advances of contaminant front in the computational domain. As the number of finite elements increases, the magnitude of dominance becomes more significant, making elements and temporal contributions negligible when compared to jump residual. This dominance shows that jump residual is a important quantity in residual error estimator and can not be neglected.

**RESUMO.** Considerando o fato de que o transporte de contaminantes ocorre em regime de pequena advecção, um estimador residual é usado para analisar a equação parabólica



que descreve o fenômeno de advecção-difusão-reação em meio poroso saturado. A solução numérica correspondente é obtida pelo método de elementos finitos usando um esquema  $\theta$  A-estável em código Python. O estimador de erro residual avalia separadamente as partes componentes do erro e permite a análise e comparação das contribuições para o erro residual. Essa análise considera uma sequência de problemas com diferentes números de elementos na malha computacional. As simulações numéricas indicam que o residual do salto é dominante em comparação com outros resíduos estimados e a dominância cresce com o número de elementos na malha e com o tempo. Adicionalmente, o problema considerado requer esforço adicional para o cálculo das contribuições associadas com a projeção  $L^2$  da função fonte de contaminante no espaço de elementos finitos.

**Palavras-chave:** estimador de erro residual, equação de advecção-difusão-reação, regime de pequena advecção, elementos finitos, transporte de contaminantes em meios porosos.

## REFERENCES

- [1] J. Bear. *Hydraulics of Groudwater*, Dover Publications Inc., New York, (1979).
- [2] S.C. Brenner. *The Mathematical Theory of Finite Element Methods*, em “Texts in Applied Mathematics, v.15” (J.E. Marsden, L. Sirovich and S.S. Antman, eds.), Springer-Verlag, New York, (1994).
- [3] R.E. Ewing. *A posteriori* error estimation. *Computer Methods in Applied Mechanics and Engineering*, **82**(1-3): 59–72, September (1990).
- [4] Fenics project documentation. <http://fenicsproject.org/>, 2012. Accessed in 05/2012.
- [5] A. Firmiano. “Um estimador de erro *a posteriori* para a equação do transporte de contaminantes em regime de pequena advecção”. Tese de Doutorado, SHS/EESC/USP, (2010).
- [6] <https://answers.launchpad.net/dolphin/+question/177108>.
- [7] G. Hofinger & F. Judex. Pollution in groundwater flow: definition of ARGESIM comparison C19. *Simulation News Europe*, 44/45: 51–52, (2005).
- [8] R.C. Kirby. Algorithm 839: Fiat, a new paradigm for computing finite element basis functions. *ACM Transactions on Mathematical Software*, **30**(4) (2004), 502–516.
- [9] R.C. Kirby. Optimizing the evaluation of finite element matrices. *SIAM Journal on Scientific Computing*, **27**(3) (2005), 741–758.
- [10] R.C. Kirby. A compiler for variational forms. *ACM Transactions on Mathematical Software*, **32**(3), (2006).
- [11] R.C. Kirby. Efficient compilation of a class of variational forms. *ACM Transactions on Mathematical Software*, **33**(3), (2007).
- [12] R.C. Kirby. Geometric optimization of the evaluation of finite element matrices. *SIAM Journal on Scientific Computing*, **29**(2) (2007), 827–841.
- [13] D. Kuzmin. Goal-oriented *a posteriori* error estimates for transport problems. *Mathematics and Computers in Simulation*, **80**(8): 1674–1683, April (2010).

- [14] A. Logg. Automating the finite element method. *Archives of Computational Methods in Engineering*, **14**(2) (2007), 93–138.
- [15] Matplotlib. <http://matplotlib.sourceforge.net/index.html>, 2012. Accessed in 06/2012.
- [16] A. Papastavrou. *A posteriori* error estimators for stationary convection-diffusion problems: a computational comparison. *Computer Methods in Applied Mechanics and Engineering*, **189**(2): 449–462, September (2000).
- [17] D. Praetorius. A space-time adaptive algorithm for linear parabolic problems, Asc report 07/2008, Institute for Analysis and Scientific Computing Vienna University of Technology-TU Wien, 2008, disponível em [www.asc.tuwien.ac.at](http://www.asc.tuwien.ac.at) ISBN 978-3-902627-00-1.
- [18] Scitools: Python library for scientific computing. <http://code.google.com/p/scitools/>, 2012. Accessed in 06/2012.
- [19] R. Verfürth. A review of *a posteriori* error estimation techniques for elasticity problems. *Computer Methods in Applied Mechanics and Engineering*, **176**(1-4): 419–440, July (1999).
- [20] R. Verfürth. Adaptive finite element methods lecture notes winter term 2007/08, 2007/2008, disponível em <http://citeseerx.ist.psu.edu/viewdoc/summary-doi=10.1.1.155.4064>.