A lowest order mixed finite element code for Forchheimer flow

C. Kim & J.E. Pasciak

August 15, 1997

Abstract

This code is based on lowest order Raviart-Thomas elements on a topologically regular triangulation contained in R^2. It uses a nonlinear Uzawa type iteration as a solver and can simulate Forchheimer wells as well as various combinations of Dirichlet and Neumann boundary conditions.

1 Introduction

This report provides a description of numerical algorithms employed in the code to implement the Forchheimer well model discussed in [4]. Also included are the results of some numerical simulations.

First, the Forchheimer flow equations and the variables therein are reviewed.

Then, the numerical algorithms are introduced. For discretization of the Forchheimer flow equations, we use mixed finite element method with the lowest order Raviart-Thomas elements on triangles. And to solve the nonlinear system thus formulated, we apply the inexact Uzawa algorithm.

The mixed method gives a better approximation on the velocity of the flow than the usual finite element method. This is an advantage especially in multi-phase flows with transport, which is a phenomenon goverened by the velocity (see e.g. [3]).

Finally, some results of convergence behavior and test runs performed on a single well model are presented. The tests are done for two cases: (1) where the bottom hole pressure is kept constant and the flow rate is to be determined and (2) where the flow rate is prescribed and the bottom hole pressure is to be calculated. In both cases, the effect of the Forchheimer tensor β are observed and compared with Darcy flow, where β is not present.

2 Problem formulation

The code implements the Forchheimer nonlinearity along with a Forchheimer well model. The Forchheimer flow equations in x-y geometry in non-dimensional form are given by

$$\begin{aligned} \frac{\partial(\phi\tilde{\rho})}{\partial t} + \nabla\cdot(u\tilde{\rho}) &= q, \\ -(\nabla p + \rho g \nabla Z) &= (\mu K^{-1} + \rho\beta|u|)u. \end{aligned} \tag{1}$$

Here is a brief description of the variables:

ϕ:	porosity of the rock
$\tilde{\rho}$:	molar density of the gas, ρ/MW, where MW is the molecular weight of the gas
ρ:	mass density of the gas
q:	source/sink term for gas
p:	gas pressure
g:	gravitational acceleration scalar
Z:	z-direction vector, $(0,0,z)$
μ:	fluid viscosity
K:	permeability tensor
β:	Forchheimer tensor
u:	gas velocity $=(u_x,u_y,u_z)$
$\|u\|$:	gas speed $=\sqrt{u_x^2+u_y^2+u_z^2}$

It must be noted that some of the above variables depend on the pressure p, namely $\rho=\rho(p)$, $\tilde{\rho}=\tilde{\rho}(p)$, $\phi=\phi(p)$, $\mu=\mu(p)$, and $u=u(p)$.

3 Code summary

The Raviart-Thomas spaces are defined on deformations of a typical regular triangulation as depicted in Figure 1. The velocity space V_h consists of those functions which are piecewise linear and have constant continuous normal components on the edges of the triangles. The pressure space Π_h consists of functions which are piecewise constant on the triangles (see, e.g. [2]).

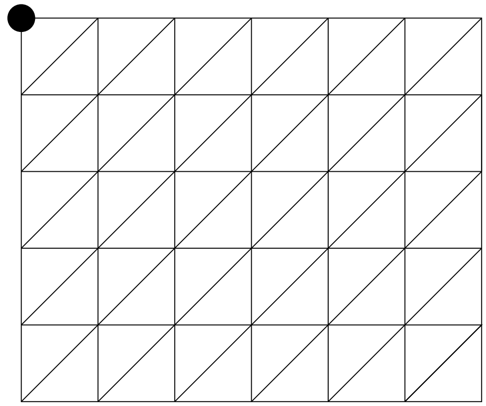

Figure 1: A 5×6 grid with a well at the upper left corner

At each time step, the mixed approximation is defined to be the pair (U^{n+1}, P^{n+1}) satisfying the semi-discrete weak form,

$$< \rho^{-1}(\mu K^{-1} + \beta|U^{n+1}|)U^{n+1}, \theta > - < P^{n+1}, \nabla \cdot \theta >= - < \rho g \nabla Z, \theta >$$

$$- < \nabla \cdot U^{n+1}, \varphi > - (\Delta t)^{-1} < \phi^{n+1}\rho^{n+1}, \varphi >$$
$$= -MW \ < q, \varphi > -(\Delta t)^{-1} < \phi^n \rho^n, \varphi > .$$

Here $< \cdot, \cdot >$ denotes the inner product in L^2. Note that the velocity variable U^{n+1} is an approximation to ρu. Also note that the no flow boundary condition rids of the boundary term in the first equation. More explicitly, this can be written

$$a(U^{n+1}, \theta) + b(\theta, P^{n+1}) = - \int_\Omega \rho g \nabla Z \cdot \theta dx,$$
$$b(U^{n+1}, \varphi) - c(\phi^{n+1}\rho^{n+1}, \varphi) = -MW \int_\Omega q\varphi dx - (\Delta t^{-1}) \int_\Omega \phi^n \rho^n \varphi dx,$$

where

$$a(u, \theta) = \int_\Omega \rho^{-1}(\mu K^{-1} + \beta\,|u|)u \cdot \theta dx,$$
$$b(u, p) = - \int_\Omega p \nabla \cdot u dx,$$
$$c(p, \varphi) = (\Delta t)^{-1} \int_\Omega p\varphi dx.$$

As can be easily seen from the weak formulation, explicit Euler scheme is used for discretization in time.

The above equations are reduced to matrix equations by introducing finite element bases for the spaces V_h and Π_h. This leads to a saddle-point problem of the form

$$\begin{pmatrix} A & B \\ B^t & -C \end{pmatrix} \begin{pmatrix} X \\ Y \end{pmatrix} = \begin{pmatrix} F \\ G \end{pmatrix}.$$

Here A has the Forchheimer nonlinearity while C is a nonlinear matrix from the time term.

The code utilizes a nonlinear solver based on the inexact Uzawa algorithm (for details, see [1]). This is a two stage iteration which is defined as follows:

Inexact Uzawa Algorithm

1. Assume that starting approximations (X_0, Y_0) are given.
2. For $i = 1, 2, \dots$ do
   (a) Set $X_{i,0} = X_{i-1}$.
   (b) For $j = 1, \dots, k$ do
   $$X_{i,j} = X_{i,j-1} + \tau_1 Q_A (F - BY_{i-1} - AX_{i,j-1}).$$
   (c) Set $X_i = X_{i,k}$.
   (d) Define Y_i by
   $$Y_i = Y_{i-1} + \tau_2 Q_C (-CY_{i-1} + B^t X_i - G).$$

The operators Q_A and Q_C are preconditioners. Currently, Q_A is a diagonal operator with weights depending on the most recent iterate $X_{i,j-1}$. The preconditioner Q_C is somewhat more critical to future code efficiency and is under development. Currently Q_C is the identity (no preconditioning). The positive constants τ_1 and τ_2 are iteration parameters which are determined by trial and error. Generally, one uses $k = 1$ or $k = 2$ and τ_1 fixed. Then one does some preliminary runs to determine an appropriate value of τ_2.

When q in (1) is a point source/sink term, e.g. $q = Q\delta$, the well model involves two effective radii and has the form

$$P_0 - P_w = \frac{Q\mu}{2\pi K} log(r_1/r_w) + \frac{\rho\beta Q|Q|}{4\pi^2}(r_w^{-1} - r_2^{-1}).$$

Here r_1 and r_2 are effective radii. These can be computed experimentally (numerically) or analytically based on some simplifying assumptions near the well. The analytical calculation (for lowest order Raviart-Thomas mixed finite elements where the well is placed at the corner opposite the hypothenuse in an isoceles right triangle) suggests

$$r_1 = \frac{2\sqrt{2}}{3} e^{-\pi/6},$$
$$r_2 = \frac{12\sqrt{2}}{18+\pi^2}.$$

The above is, of course, only valid for two dimensional calculations under simplifying assumptions (e.g. constant coefficients).

The code allows wells of either bottom hole type (fixed bottom hole pressure) or prescribed flow rate. In the case of a prescribed flow rate, one is allowed to specify a minimal well bore pressure. The computation shuts down if the bottom hole pressure goes below the minimum well bore pressure.

4 Test

4.1 Convergence of the numerical method

Here, we verify that the method employed in the code gives a result which converges to a known analytical solution.

4.1.1 Equation

The code was tested on the parabolic initial value problem

$$\frac{\partial p}{\partial t} - \Delta p = q\delta, \tag{2}$$

where δ is the Kronecker delta function at the well.

The test was run on the domain $\Omega = (0,10) \times (0,10)$ with time interval $[0,2]$. A well with $q = 20.0$ was placed on the upper left corner of the domain as shown in Figure 1. The initial pressure was set $p_i = 250.0$ and the following boundary conditions were imposed.

$$\frac{\partial u}{\partial n} = 0 \quad \text{on } \{0\} \times [0,10] \text{ and } [0,10] \times \{10\},$$
$$p = p_i \quad \text{on } \{10\} \times [0,10] \text{ and } [0,10] \times \{0\}.$$

In other words, we have no flow boundary condition on the upper and left boundaries and Dirichlet boundary condition on the bottom and right boundaries.

The analytic solution for the infinite domain problem, i.e. problem (2) on $(0,\infty) \times (0,\infty)$ with $p = p_i$ at infinity and initial pressure p_i, is given by

$$p(r,t) = p_i - \frac{q}{\pi} E_1\left(\frac{r^2}{4t}\right), \tag{3}$$

where $E_1(x)$ is the exponential integral whose series expansion is

$$E_1(x) = 0.57721 + \ln x + \sum_{n=1}^{\infty} \frac{x^n}{n \cdot n!}$$

4.1.2 Error

First, discrete l_2-norm error along the diagonal was calculated, once on the whole diagonal (i.e. $0 < r < 10\sqrt{2}$), denoted e_0, and then away from the singularity (i.e. $1 < r < 10\sqrt{2}$), denoted e_1. Here,

$$r = \left\{x^2 + (z-10)^2\right\}^{1/2}$$

and the discrete l_2-norm is defined

$$\|p - p_h\| = \left(\frac{1}{N}\sum_{i=1}^{N}(p(x_i) - p_h(x_i))^2\right)^{1/2}.$$

Triangles on diagonal are numbered from 1 to N and x_i is the centroid of the i-th triangle. p_h is the calculated, i.e. numerical, solution and p is as in (3).

The results are given in Tables 1, 2, and 3.

Table 1: Discrete l_2-norm error on the diagonal for fixed $\Delta t = 0.02$

h	$e_1 = l_2$ error from 1 to $10\sqrt{2}$	ratio	$e_0 = l_2$ error from 0 to $10\sqrt{2}$	ratio
.500	3.237e-02		3.280e-01	
.250	1.490e-02	2.172	2.348e-01	1.397
.125	8.662e-03	1.720	1.670e-01	1.406

Then, an approximate L^2-norm error on the domain was calculated. Also because of singularity, the calculations were performed on the whole domain (i.e. for $r > 0$) and away from the singularity (i.e. for $r \geq 1$). The definition of the norm is as follows.

$$\|p - p_h\| = \left(\sum_{\tau_i} |(p - p_h)(x_i)|^2 |\tau_i|\right)^{1/2},$$

Table 2: Discrete l_2-norm error on the diagonal for fixed $\Delta t = 0.01$

h	$e_1 = l_2$ error from 1 to $10\sqrt{2}$	ratio	$e_0 = l_2$ error from 0 to $10\sqrt{2}$	ratio
.500	3.001e-02		3.252e-01	
.250	1.242e-02	2.416	2.324e-01	1.399
.125	5.782e-03	2.148	1.650e-01	1.408

Table 3: Discrete l_2-norm error on the diagonal for fixed $h = .125$

Δt	l_2 error from 1 to $10\sqrt{2}$	ratio	l_2 error from 0 to $10\sqrt{2}$	ratio
.08	2.456e-02		1.804e-01	
.04	1.458e-02	1.684	1.714e-01	1.053
.02	8.662e-03	1.683	1.670e-01	1.026
.01	5.782e-03	1.498	1.650e-01	1.012

where i runs through all triangles and x_i is the centroid of the triangle τ_i. The results are given in Tables 4, 5, and 6.

Table 4: L^2 error for fixed $\Delta t = 0.02$

h	L_2 error (for $r \geq 1$)	ratio	L_2 error (for $r > 0$)	ratio
.500	1.429e-01		7.448e-01	
.250	5.775e-02	2.474	3.793e-01	1.964
.125	3.567e-02	1.619	1.945e-01	1.950

4.2 Dependence on β

Reported in this section are the results of computations for various Forchheimer tensors β. In the following, β will denote $7.6e7\beta I$ and calculations for $\beta = 0, 1$, and 10 will be performed.

The computation was performed in a square of side length 5000(ft) with the upper left corner at the origin, where the well is placed. We

Table 5: L^2 error for fixed $\Delta t = 0.01$

h	L_2 error (for $r \geq 1$)	ratio	L_2 error (for $r > 0$)	ratio
.500	1.401e-01		7.382e-01	
.250	4.886e-02	2.867	3.741e-01	1.973
.125	2.067e-02	2.364	1.889e-01	1.980

Table 6: L^2 error for fixed $h = .125$

Δt	L_2 error (for $r \geq 1$)	ratio	L_2 error (for $r > 0$)	ratio
.08	1.258e-01		2.574e-01	
.04	6.819e-02	1.845	2.120e-01	1.214
.02	3.567e-02	1.912	1.945e-01	1.090
.01	2.067e-02	1.726	1.889e-01	1.030

assume no flow boundary conditions on all boundaries. Then, by symmetry, this will depict the lower right quarter of the whole domain. Other assumptions are as follows:

$$r_w = 0.35(\text{ft}) \text{ (well radius)}$$
$$p_i = 5000(\text{psi}) \text{ (initial pressure)}$$

The grid size was kept at $\Delta x = \Delta z = 5000/N$ with $N = 30$. But, the number of timesteps N_t was varied and will be stated for each specific case.

4.2.1 Wells of prescribed flow rate

Here, the flow rate is prescribed to be

$$q = 80 \text{ (source/sink term)}.$$

Bottom whole pressures for $\beta = 0$, and 1 were calculated for $t = 400$ (days) and $N_t = 40$. The result is shown in Figure 2.

4.2.2 Wells of bottom hole type

In this case, the bottom hole pressure is fixed at

$$p_w = 1000(\text{psi})$$

and calculations are done for $\beta = 0, 1$, and 10. Plots of q, $\int q dt$, and average pressure for 2,000 days are presented in Figures 3, 4, and 5. In these cases, N_t was 20 and 32 for $t = 0$ to $t = 400$ and $t = 400$ to $t = 2000$(days), respectively.

Finally, pressure along diagonal after 100 days is sketched in Figure 6. N_t was 10.

References

[1] J.H. Bramble, J.E. Pasciak, and A.T. Vassilev, *Analysis of the Inexact Uzawa Algorithm for Saddle Point Problems*, SIAM J. of Numerical Analysis (June 1997), 1072-1092.

[2] F. Brezzi, and M. Fortin, *Mixed and Hybrid Finite Element Methods*, Springer-Verlag, New York, 1991.

[3] G. Chavent and J. Jaffré, *Mathematical Models and Finite Elements for Reservoir Simulation*, North-Holland, Amsterdam, 1986.

[4] R.E. Ewing, R.D. Lazarov, and J.E. Pasciak, *Numerical Well Model for Non-Darcy Flow*, Technical Report ISC-97-03-MATH, Institute for Scientific Computation, Texas A&M University, 1997.

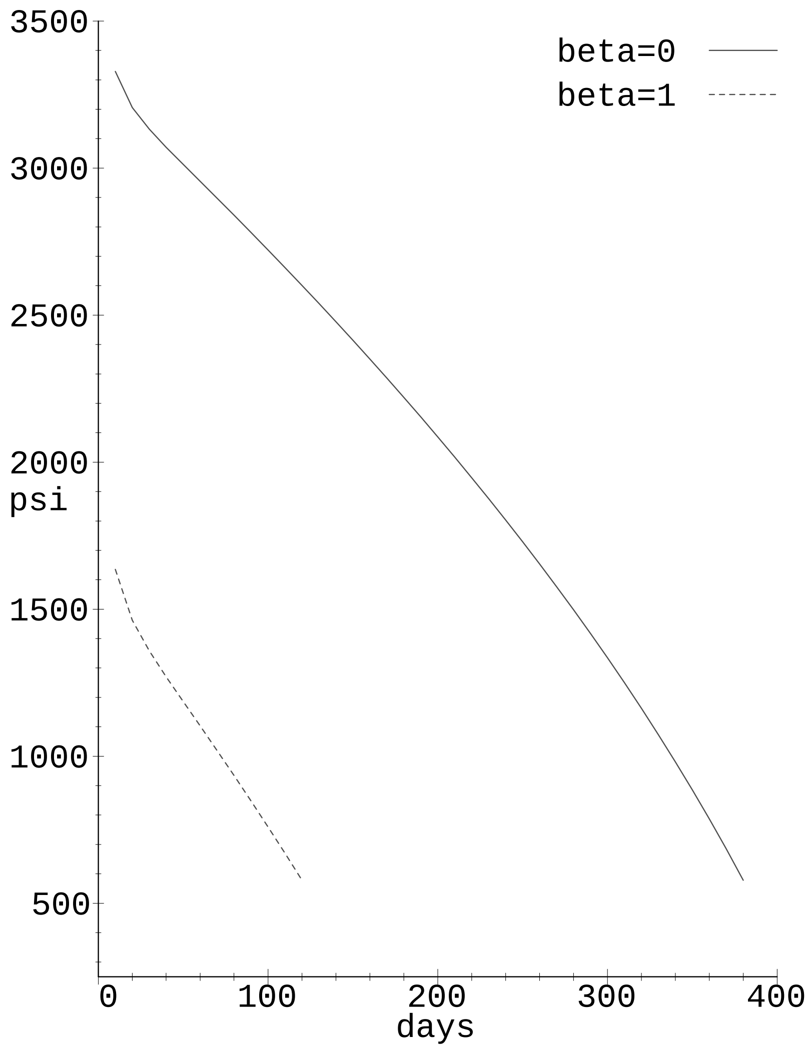

Figure 2: Bottom hole pressure for $q = 80$

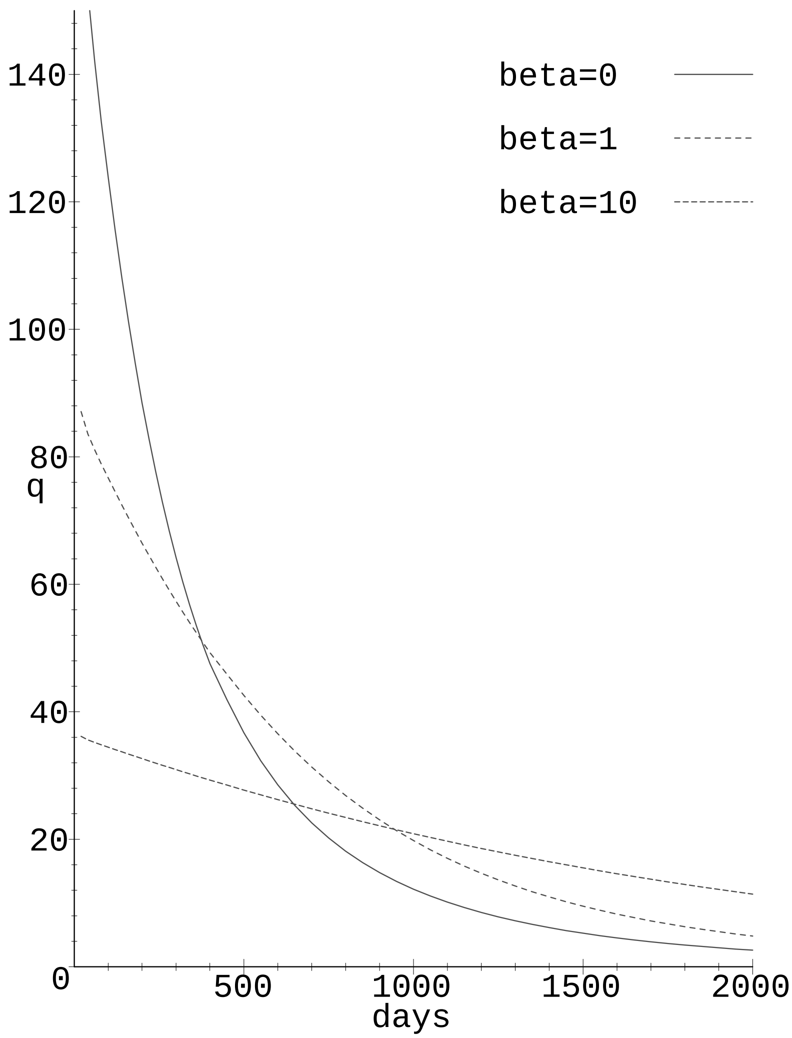

Figure 3: Source/sink term q

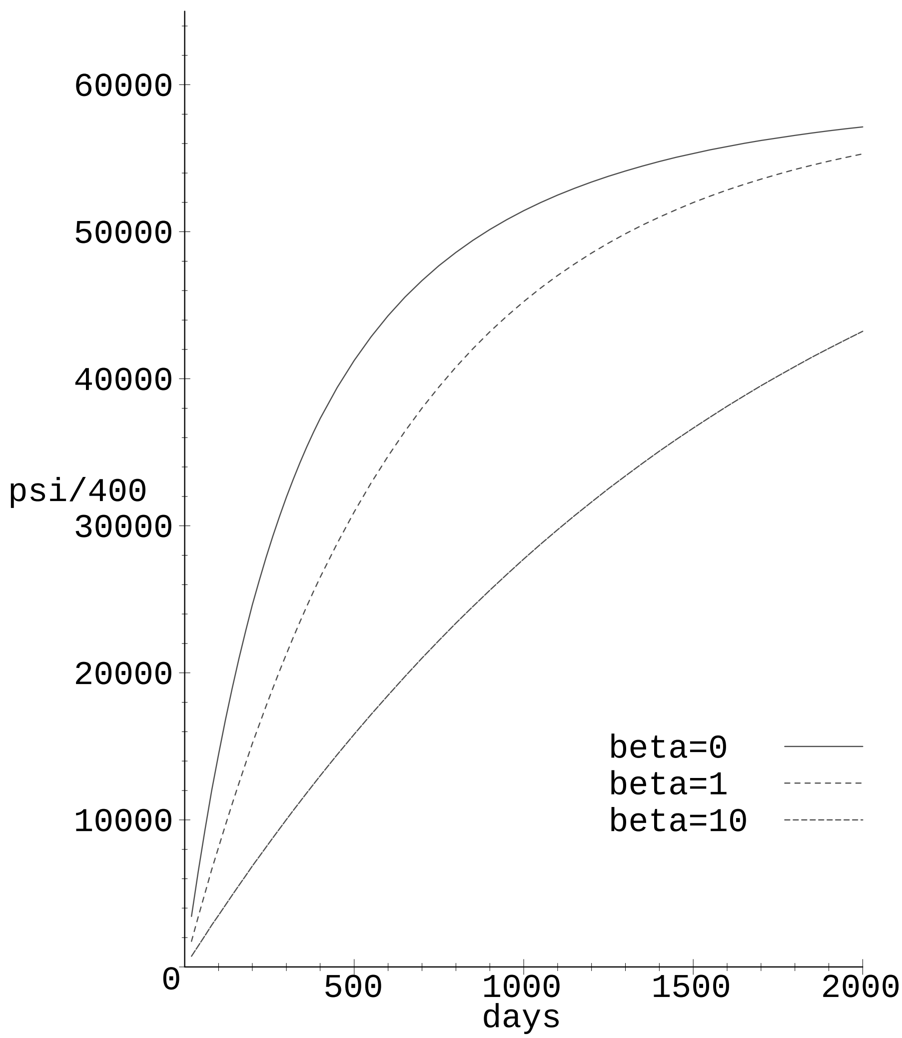

Figure 4: $\int q dt$

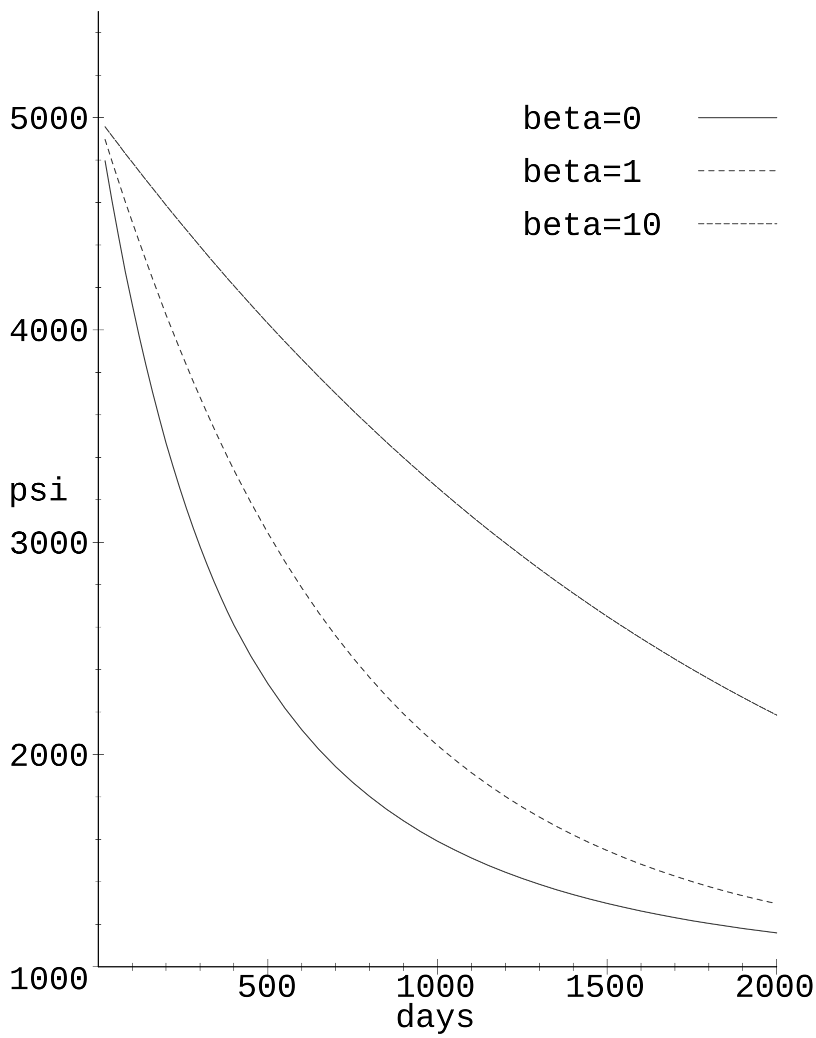

Figure 5: Average pressure

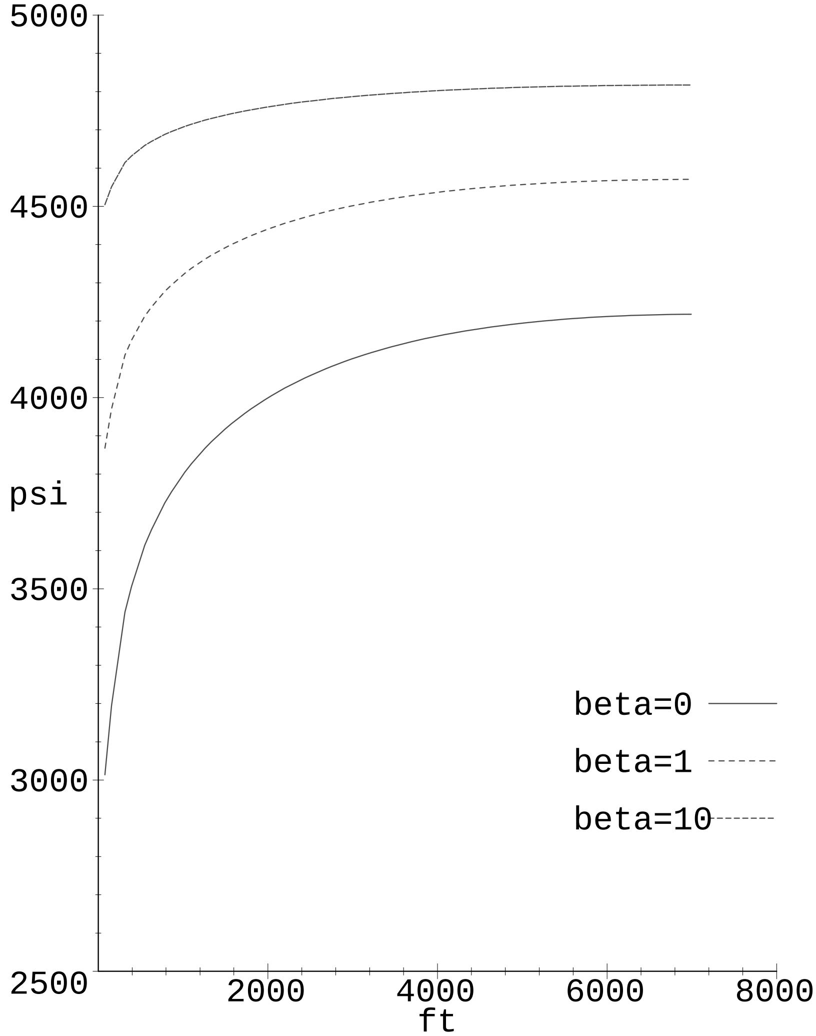

Figure 6: Pressure along the diagonal after $t = 100$ days ($p_w = 1000$)