The Averaged Gradient Technique in Sensitivity Analysis

Jan Chleboun

Department of Mathematics, Faculty of Civil Engineering, Prague

Dolní Maxov, June 1–6, 2008

Outline

Averaged (a.k.a. Recovered) Gradient

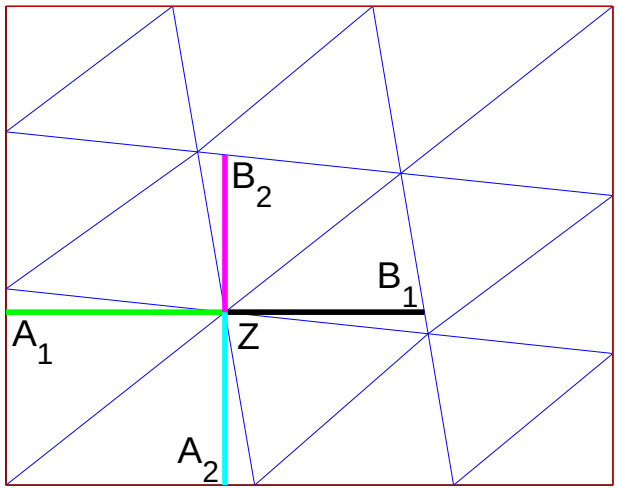
Idea (in 2D): Let (u_x^a, u_y^a) be an approximation of ∇u. By processing (u_x^a, u_y^a), we wish to obtain $(\hat{u}_x^a, \hat{u}_y^a)$, a better (more accurate) approximation of ∇u.

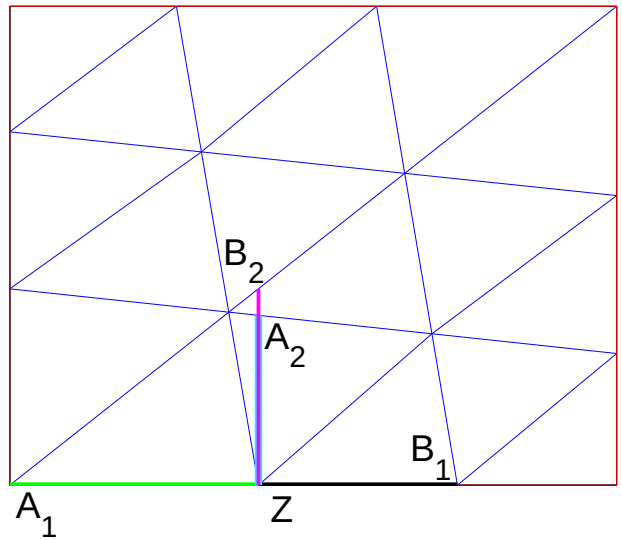
Application: In FEM, $\|\nabla u - (u_{hx}, u_{hy})\|_{L^2} \leq Ch$, where u_h is a FE approximation to a BVP solution u. It can be $\|\nabla u - (\hat{u}_{hx}, \hat{u}_{hy})\|_{L^2} \leq Ch^2$.

Various methods. We will use the method proposed in

I. Hlaváček, M. Křížek, V. Pištora: *How to recover the gradient of linear elements on nonuniform triangulations*, Appl. Math. 41 (1996), 241-267

Recovered Gradient: Technique





$$a_i = (A_i - Z)_i, \quad b_i = (B_i - Z)_i, \quad i = 1, 2(\ldots d)$$

The weighted averaged gradient of function v at node Z

$$(G_h v(Z))_i = \alpha_i v(A_i) - (\alpha_i + \beta_i) v(Z) + \beta_i v(B_i)$$

where

$$\alpha_i = \frac{b_i}{a_i(b_i - a_i)}, \quad \beta_i = \frac{a_i}{b_i(a_i - b_i)}.$$

Properties I

Assumptions:

$\Omega \subset \mathbb{R}^2$ ($\mathbb{R}^3$), a bounded domain with a polygonal Lipschitz boundary.

$\mathcal{F}$, a strongly regular family of triangulations $\{T_h\}_{h \to 0}$, i.e.,

$$\exists \varkappa > 0 \quad \forall T_h \in \mathcal{F} \quad \forall K \in T_h \quad \varkappa h \leq \varrho_K ,$$

where ϱ_K is the radius of a ball $\mathcal{B}_K$ such that $\mathcal{B}_K \subset K$.

Theorem: Let $q \in (1, \infty)$ and $v \in W_q^3(\Omega)$. Then exists a constant $C > 0$ such that

$$\|\nabla v - G_h v\|_{0,q} \leq C h^2 |v|_{3,q}$$

for any $T_h \in \mathcal{F}$.

Properties II

An elliptic BVP

$$\begin{aligned} -\operatorname{div}(\lambda \nabla u) &= f \quad \text{in } \Omega, \\ u &= 0 \quad \text{on } \partial\Omega, \end{aligned}$$

where λ is a matrix comprising sufficiently smooth functions ($\lambda \in [W^2_{2+\varepsilon}(\Omega)]^{2\times 2}$).

It is assumed that $u \in W^3_q(\Omega)$, $q > 2$.

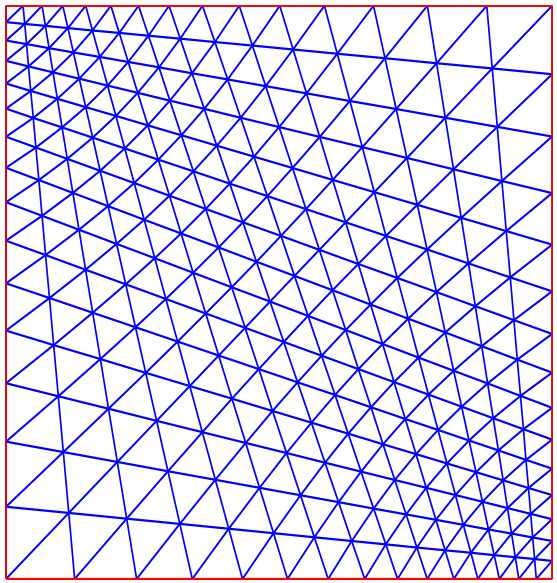
Let us consider u_h, the finite element approximation to u (p.-w. linear basis functions).

Is $G_h u_h$ close to ∇u?

Properties II

Assumptions: The family $\mathcal{F} = \{T_h\}_{h \to 0}$ is generated by smooth distortions of uniform triangulations of square grids.

(Then the linear interpolation of u is sufficiently close to u.)



Ω_0, a domain such that $\overline{\Omega}_0 \subset \Omega$

Theorem: $\|\nabla u - G_h u_h\|_{0,2,\Omega_0} \leq C(u)h^2$.

Sensitivity Analysis: Motivation

The weak formulation of the (simplified) BVP

$$\int_\Omega \lambda(x)\nabla u \cdot \nabla v \, dx = \int_\Omega fv \, dx \quad \forall v \in H_0^1(\Omega),\ u \in H_0^1(\Omega).$$

A unique solution $u(\lambda; \cdot)$.

Criterion-functional: $\Psi(\lambda) = \Phi(u(\lambda))$

Example (Parameter Identification)

A function w is given. The goal is to find λ that minimizes

$$\Psi(\lambda) = \int_\Omega (w(x) - u(\lambda; x))^2 \, dx.$$

Analogous settings appear in optimal design problems, the worst scenario method, or PDE/ODE-constrained optimization.

Sensitivity Analysis

In the search for

$$\arg\min_{\lambda \in \mathcal{U}_{\text{ad}}} \Psi(\lambda),$$

where $\mathcal{U}_{\text{ad}}$ is an admissible set of parameters, the derivative Ψ' of Ψ w.r.t. λ is beneficial.

Remark: In discretized problems (FEM and a discretization of $\mathcal{U}_{\text{ad}}$), $\Psi(\lambda)$ is a function of several variables $\hat{\lambda}_1, \ldots, \hat{\lambda}_n$ and Ψ' becomes $\nabla\Psi$.

Let us focus on a particular approach to the Ψ' calculation: the formula for Ψ' is derived from the "continuous" weak problem, and then approximated by the solutions of discretized problems.

In our problem, let
$\lambda(x_1, x_2) = \lambda_1 + \lambda_2 x_1 + \lambda_3 x_2 + \lambda_4 x_1^2 + \lambda_5 x_1 x_2 + + \lambda_6 x_2^2$.
Then, for instance,

$$\frac{\partial \Psi}{\partial \lambda_4}(\lambda) = -\int_\Omega x_1^2 \nabla u(x) \cdot \nabla z(x)\, dx,$$

where $z \in H_0^1(\Omega)$ solves the adjoint equation

$$\int_\Omega \lambda(x) \nabla z \cdot \nabla v\, dx = \int_\Omega 2(u - w)\, dx \quad \forall v \in H_0^1(\Omega)$$

(the right-hand side equals the derivative of $\Phi(u)$ w.r.t. u).

Generally (not only in our example!), Ψ' is evaluated through the integration of an expression containing ∇u and ∇z. Since the accuracy of Ψ' depends on the accuracy of the approximations of ∇u and ∇z, the idea of recovering ∇u_h and ∇z_h, the FEM-computed gradients, emerged.

Generally (not only in our example!), Ψ' is evaluated through the integration of an expression containing ∇u and ∇z. Since the accuracy of Ψ' depends on the accuracy of the approximations of ∇u and ∇z, the idea of recovering ∇u_h and ∇z_h, the FEM-computed gradients, emerged.

The idea failed.

The gradient of Ψ computed by means of the recovered gradient technique is not better than the gradient computed directly.

Why?

Reasons:

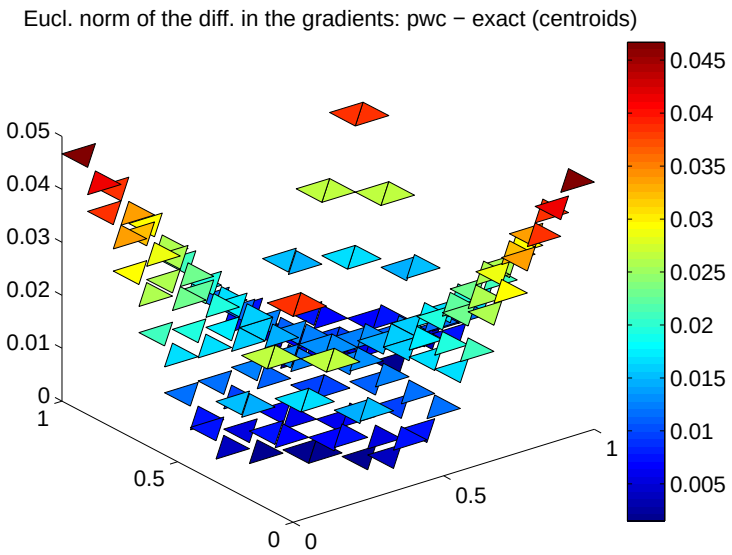
1. A bug in the code?

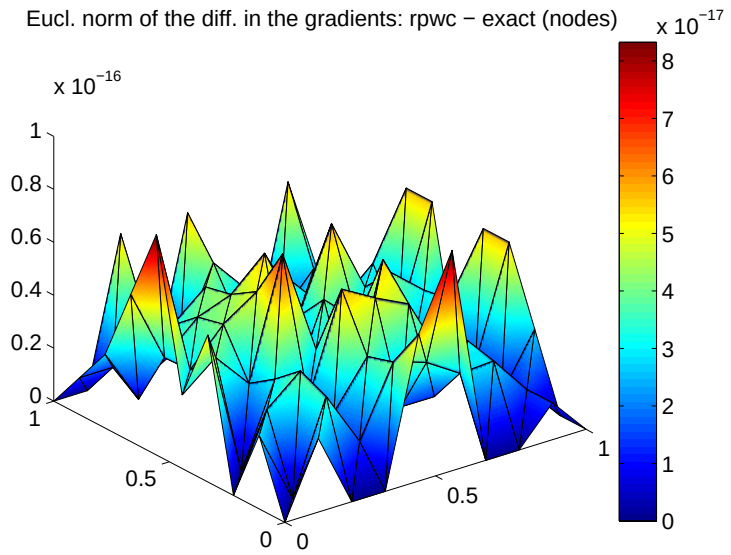
Why?

Reasons:

1. A bug in the code?
2. A "boundary layer", where the averaging is inaccurate.

Interpolation features of G_h, uniform mesh

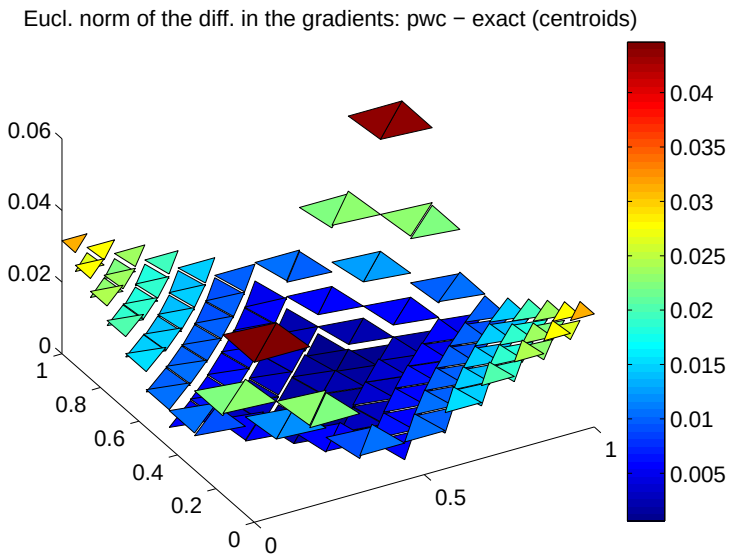


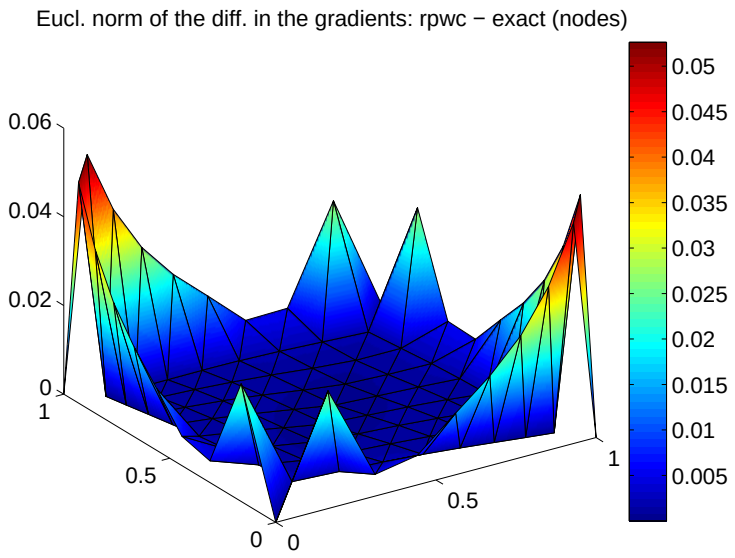


Interpolated function: $xy(1-x)(1-y)$

Euclid. norm of the difference between the calculated and the exact gradient: p.-w. constant gradient (left), recovered gradient (right).

Interpolation features of G_h, distorted mesh

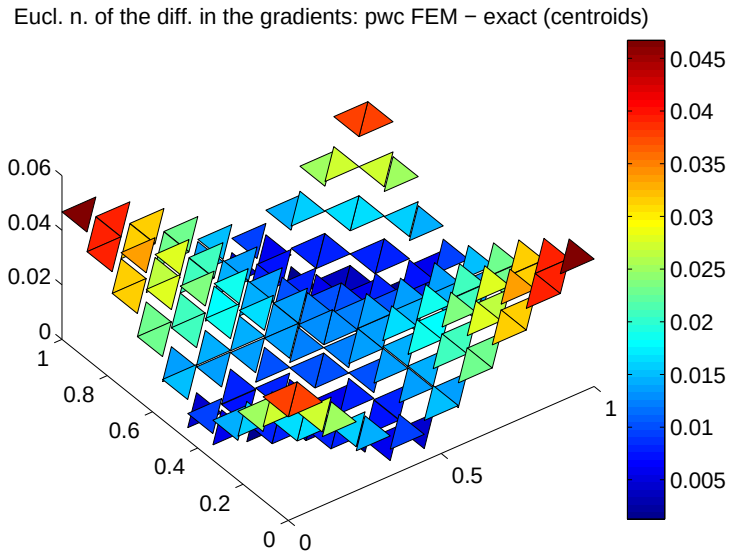


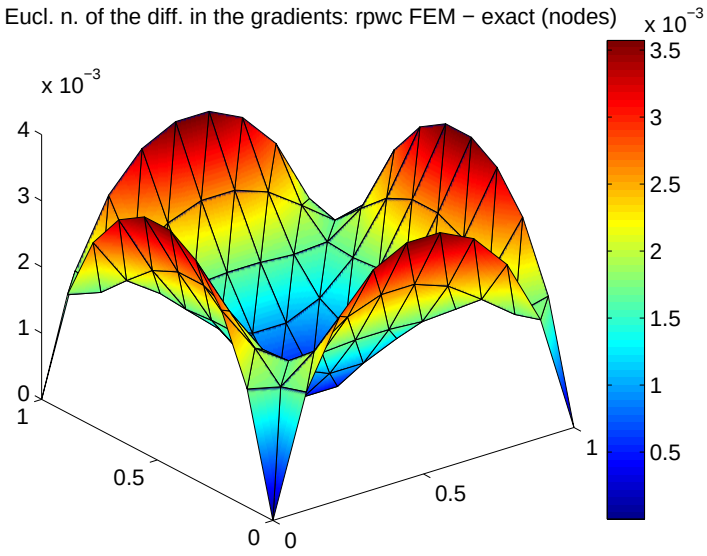


Interpolated function: $xy(1-x)(1-y)$

Euclid. norm of the difference between the calculated and the exact gradient: p.-w. constant gradient (left), recovered gradient (right).

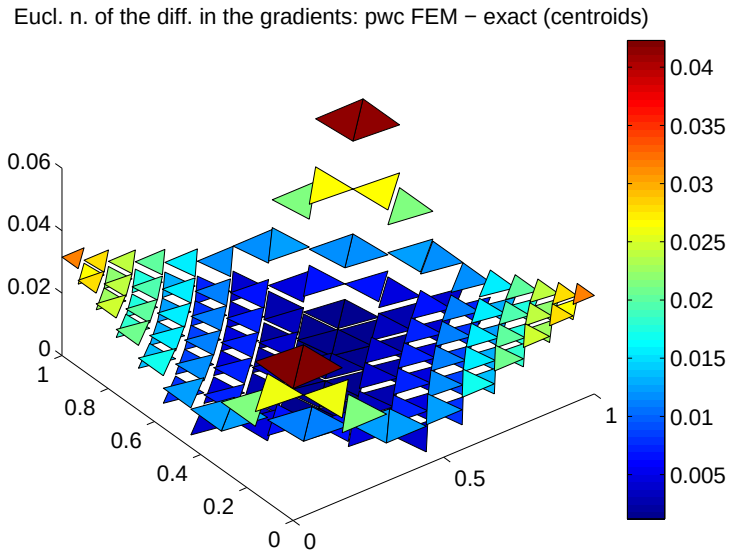
FEM features of G_h, uniform mesh

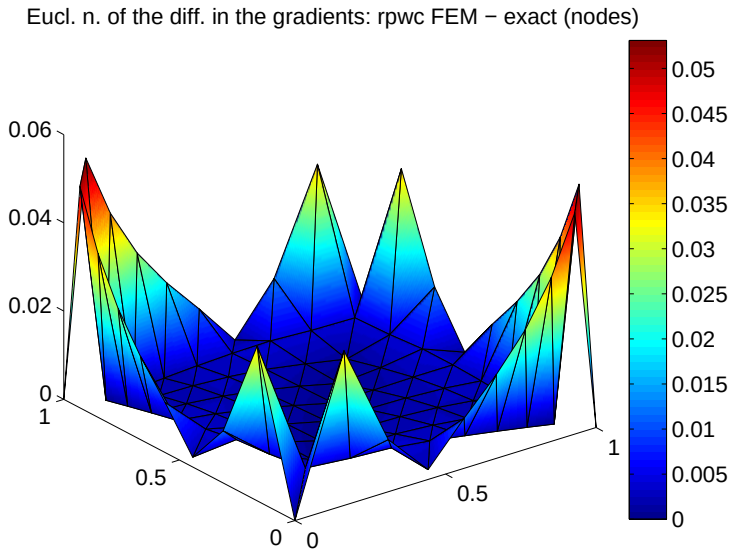




FEM solution of the Poisson equation with the exact solution $xy(1-x)(1-y)$.
Euclid. norm of the difference between the calculated and the exact gradient: p.-w. constant gradient (left), recovered gradient (right).

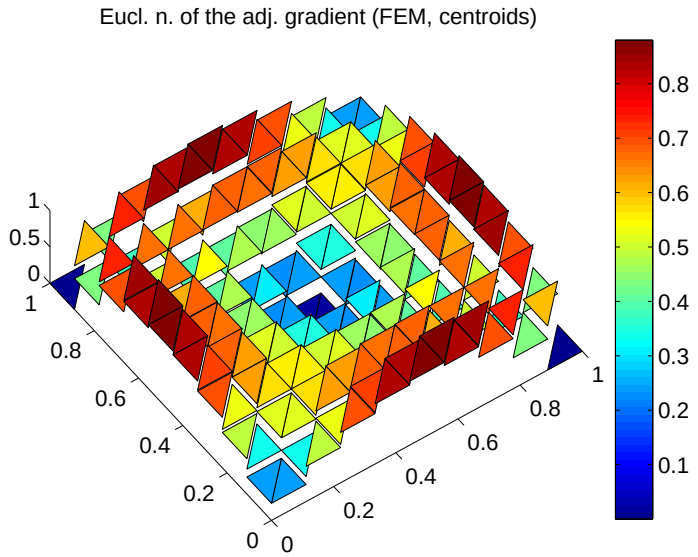
FEM features of G_h, distorted mesh

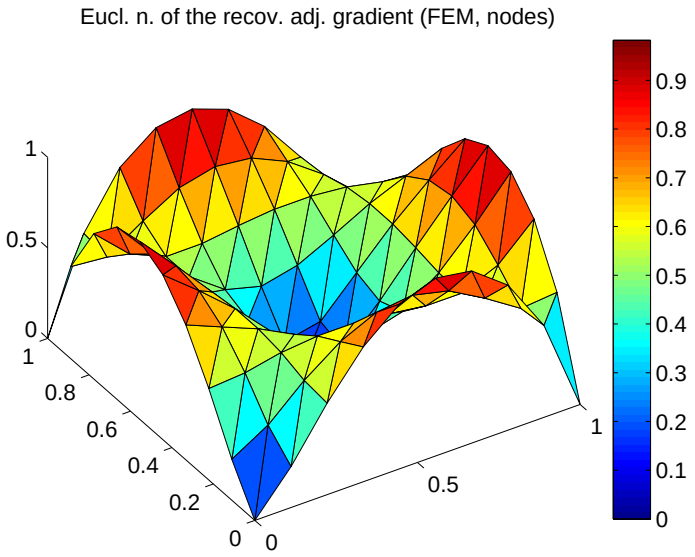




FEM solution of the Poisson equation with the exact solution $xy(1-x)(1-y)$.
Euclid. norm of the difference between the calculated and the exact gradient: p.-w. constant gradient (left), recovered gradient (right).

Gradient of the adjoint state, uniform mesh





The Euclid. norm of the p.-w. constant (left) and recovered gradient of the adjoint solution.

Observations: A boundary layer exists where (a) the recovered gradient of ∇u_h (and ∇z_h) is inaccurate, and (b) the amount of sensitivity information stored in ∇z_h is substantial.

Conclusion: The performance of $\nabla u_h \cdot \nabla z_h$ is poor even for recovered gradients.

Work in progress

- Exact integration in the FEM code.
- Focus not on $\nabla u_h \cdot \nabla z_h$ in the sensitivity formula, but on special criterion-functionals as

$$\Psi(\lambda) = \int_{\Omega_0} (w_{,x_1} - u_{,x_1})^2 + (w_{,x_2} - u_{,x_2})^2 \, dx,$$

where u solves the Poisson equation, w is a given function, and $\overline{\Omega}_0 \subset \Omega$.

Matlab and PDE Toolbox

Two errors discovered in R2007b and R2008a (R2007a too?, not in R2006a):

- The format of the boundary condition matrix "b" as used in the past and as described in the PDE Toolbox help system or handbook causes an error message and the breakdown of the code.
  Will be fixed in R2008b. DIY fixing is easy.
- The PDE Toolbox GUI (`pdetool`) does not solve BVPs with a Dirichlet homogeneous boundary condition. It delivers a solution that is nonzero along the boundary.
  Will be fixed?

Acknowledgement: I thank dr. Tomáš Vejchodský for his routine for numerical integration.

Thank you for your attention