Optimization techniques for tensor approximation:

Newton Trust Region TT$\longrightarrow$ Canonical

Vladimir Kazeev
vladimir.kazeev@gmail.com

Institute of Numerical Mathematics
Russian Academy of Sciences

Workshop "Tensor Methods in Multi-Dimensional Boundary-Value and Spectral Problems"
MPI MiS Leipzig, May 3-5, 2010

Canonical Decomposition

$$\boldsymbol{S}(i) = \sum_{\beta=1}^{R} U_1(i_1, \beta) \dots U_d(i_d, \beta)$$

$$\boldsymbol{S} = \sum_{\beta=1}^{R} \boldsymbol{S}_\beta,$$
$$\boldsymbol{S}_\beta = U_1(:, \beta) \otimes \dots \otimes U_d(:, \beta)$$

dnR parameters
compact representation,
linear complexity in respect to d

Canonical low-rank approximation

$$\begin{aligned} \boldsymbol{S} &\approx (V_1 \dots V_d) = \boldsymbol{T} \\ \|\boldsymbol{T} - \boldsymbol{S}\|^2 &= \sum_{\boldsymbol{i} \in \mathcal{I}}^{r} \left(\boldsymbol{T}(\boldsymbol{i}) - \boldsymbol{S}(\boldsymbol{i})\right)^2 \end{aligned}$$

Optimization problem

$$\begin{aligned} \boldsymbol{T} &= (V_1 \dots V_d) = \underset{\substack{V^{(1)} \in \mathbb{R}^{n \times r} \\ \dots \\ V^{(d)} \in \mathbb{R}^{n \times r}}}{\operatorname{argmin}} \varphi, \\ \varphi &= \frac{1}{2\|\boldsymbol{S}\|^2} \|\boldsymbol{T} - \boldsymbol{S}\|^2 + \frac{\kappa}{2\|\boldsymbol{S}\|^2} \sum_{\beta=1}^{r} \|\boldsymbol{T}_\beta\|^2 \end{aligned}$$

Canonical low-rank approximation

The Frobenius norm minimization problem may not have a solution

Let $x_k, y_k \in \mathbb{R}^n$, $k = 1 \dots 3$, be pairs of linearly independent vectors.
Consider two tensors:

$$\begin{aligned} \boldsymbol{S} &= x_1 \otimes x_2 \otimes y_3 + x_1 \otimes y_2 \otimes x_3 + y_1 \otimes x_2 \otimes y_3, \\ \boldsymbol{T}_\varepsilon &= \frac{1}{\varepsilon}(x_1 + \varepsilon y_1) \otimes (x_2 + \varepsilon y_2) \otimes (x_3 + \varepsilon y_3) - \frac{1}{\varepsilon} x_1 \otimes x_2 \otimes x_3. \end{aligned}$$

Then $\operatorname{rank} \boldsymbol{S} = 3$ and $\operatorname{rank} \boldsymbol{T}_\varepsilon = 2$ for all $\varepsilon > 0$
while $\|\boldsymbol{T}_\varepsilon - \boldsymbol{S}\| = \mathcal{O}(\varepsilon)$.

(*de Silva* and *Lim*, 2008)

Canonical low-rank approximation

Solution of the Frobenius norm minimization problem may be essentially ununique

$$\sin\left(\sum_{k=1}^{d} x_k\right) = \sum_{k=1}^{d} \sin(x_k) \prod_{\substack{\zeta=1 \\ \zeta \neq k}}^{d} \frac{\sin(x_\zeta + \alpha_\zeta - \alpha_k)}{\sin(\alpha_\zeta - \alpha_k)}$$

for all $\{\alpha_k\}_{k=1}^{d}$:

$$\sin(\alpha_k - \alpha_\zeta) \neq 0 \quad \forall k, \zeta = 1 \ldots d : \quad \zeta \neq k$$

(*Mohlenkamp* and *Monzón*, 2005)

Related works

- (2001) *Zhang*, *Golub*. *Rank-one approximation to high order tensors*
- (2005) *Beylkin*, *Mohlenkamp*. *Algorithms for numerical analysis in high dimensions*
- (2006) *Oseledets*, *Savostyanov*. *Minimization methods for approximating tensors and their comparison*
- (2008) *Espig*. *Effiziente Bestapproximation mittels Summen von Elementartensoren in hohen Dimensionen*
- (2009) *Acar*, *Kolda*, *Dunlavy*. *An optimization approach for fitting canonical tensor decompositions*
- (2010) *Kazeev*, *Tyrtyshnikov*. Structure of the Hessian matrix and an economical implementation of Newton method in the problem of canonical tensor approximation

Basic ideas of the method

- Newton minimization of quadratic model of objective function
- global convergence due to Trust Region strategy
- PCG as a solver for Trust Region minimization subproblem
- specific strategy of quitting iterations (*Steihaug*, 1983)

Trust Region approach

Given v_0, $\delta_{\max}$, $\delta_0 \leq \delta_{\max}$, $\epsilon_g > 0$, κ, K, ρ, σ, β:
$0 < \rho \leq \sigma < 1$, $0 \leq \beta < \frac{1}{4}$, $0 < \kappa < 1 < K$.

1. quit iterations if $\|\boldsymbol{g}_k\| \leq \epsilon_g$
2. find (accurate or approximate) solution w_k of a constrained minimization subproblem
3. evaluate

$$\mathrm{kred}_k = \frac{\mathrm{ared}_k}{\mathrm{pred}_k} = \frac{\varphi(v_k) - \varphi(v_k + w_k)}{-\langle \boldsymbol{g}_k, w_k \rangle - \frac{1}{2}\langle w_k, \boldsymbol{H}_k w_k \rangle}.$$

4. let

$$v_{k+1} = \begin{cases} v_k + w_k, & \text{if } \mathrm{kred}_k \geq \beta, \\ v_k & \text{otherwise.} \end{cases}$$

5. let

$$\delta_{k+1} = \begin{cases} \min\{\delta_{\max}, K\|w_k\|\}, & \text{if } \mathrm{kred}_k \geq \sigma, \\ \kappa\|w_k\|, & \text{if } \mathrm{kred}_k \leq \rho, \end{cases}$$

Subproblem solution

Preconditioned CG with Trust Region constraints:

1. if $\langle p_k, \boldsymbol{H} p_k \rangle \leq 0$, let $w_{k+1} = w_k + \tau p_k$, $\tau > 0$ s. t. $\|w_{k+1}\|_{\boldsymbol{A}} = \delta$, and break subiterations,
2. if $\|w_{k+1}\|_{\boldsymbol{A}} > \delta$, let $w_{k+1} = w_k + \tau p_k$, $\tau > 0$ s. t. $\|w_{k+1}\|_{\boldsymbol{A}} = \delta$, and break subiterations,

What do we have to do?

- evaluate the objective function

$$\varphi = \frac{1}{2\|\boldsymbol{S}\|^2}\|\boldsymbol{T} - \boldsymbol{S}\|^2 + \frac{\alpha}{2\|\boldsymbol{S}\|^2}\sum_{\beta=1}^{r}\|\boldsymbol{T}_j\|^2$$

- compute its gradient
- compute its Hessian

There is no need to compute the Hessian explicitly, only the matvec operation is required

Convergence theorems

Theorem (Steihaug, 1983)

Let a function f be bounded below and its gradient $g = \nabla f$ be uniformly continuous. If the subproblem of every k-th iteration is solved with relative error ϵ_k such that $\sup_k \epsilon_k < 1$, then

$$\liminf_{k \to \infty} g_k = 0.$$

Further, if the sequence $\{H_k\}_{k=1}^{\infty}$ is bounded, then

$$\lim_{k \to \infty} g_k = 0.$$

Convergence theorems

Theorem (Steihaug, 1983)

Let f be bounded below and twice continuously differentiable in the neighbourhood of a solution x^* and $\nabla^2 f(x^*) > 0$. Then there exist such $\varepsilon > 0$ and $\Delta > 0$ that if $\Delta_0 \leq \Delta$ and $\|x_0 - x^*\| \leq \varepsilon$ then $\{x_k\}_{k=1}^{\infty}$ converges to x^*.

Tensor Train

Oseledets and *Tyrtyshnikov*, 2009:

$$\boldsymbol{S}(\boldsymbol{i}) = \sum_{\boldsymbol{\alpha} \in \mathcal{A}} U_1(i_1, \alpha_1) \cdot \cdot U_2(\alpha_1, i_2, \alpha_2) \ldots U_{d-1}(\alpha_{d-2}, i_{d-1}, \alpha_{d-1}) \cdot \cdot U_d(\alpha_{d-1}, i_d)$$

for $U_k(\alpha_{k-1}, i_k, \alpha_k) \in \mathbb{R}^{R_{k-1} \times n \times R_k}$

\+ robust algorithms for computations and manipulations

TT to Canonical

$$\begin{aligned} \boldsymbol{S} = (U_1 \dots U_d) &\approx (V_1 \dots V_d) = \boldsymbol{T} \\ R_1 \dots R_{d-1} &\quad r \\ \|\boldsymbol{T} - \boldsymbol{S}\|^2 &= \sum_{\boldsymbol{i} \in \mathcal{I}} \left(\boldsymbol{T}(\boldsymbol{i}) - \boldsymbol{S}(\boldsymbol{i})\right)^2 \end{aligned}$$

Optimization problem

$$\begin{aligned} \boldsymbol{T} &= (V_1 \dots V_d) = \underset{\substack{V^{(1)} \in \mathbb{R}^{n \times r} \\ \dots \\ V^{(d)} \in \mathbb{R}^{n \times r}}}{\operatorname{argmin}} \varphi, \\ \varphi &= \frac{1}{2\|\boldsymbol{S}\|^2} \|\boldsymbol{T} - \boldsymbol{S}\|^2 + \frac{\kappa}{2\|\boldsymbol{S}\|^2} \sum_{\beta=1}^{r} \|\boldsymbol{T}_\beta\|^2 \end{aligned}$$

Matrices of quadratic model

Define $r \times r$-matrices, which depend on decomposition of $\boldsymbol{T}$:

$$\begin{aligned}
\Psi_k(\beta, \beta') &= \sum_{i_k=1}^{n_k} V_k(i_k, \beta) V_k(i_k, \beta'), \\
Q(\beta, \beta') &= (1 + \kappa \boldsymbol{\delta}(\beta, \beta')) \prod_{k=1}^{d} \Psi_k(\beta, \beta'), \\
Q_\zeta(\beta, \beta') &= (1 + \kappa \boldsymbol{\delta}(\beta, \beta')) \prod_{\substack{k=1 \\ k \neq \zeta}}^{d} \Psi_k(\beta, \beta'), \\
Q_{\zeta\zeta'}(\beta, \beta') &= (1 + \kappa \boldsymbol{\delta}(\beta, \beta')) \prod_{\substack{k=1 \\ k \neq \zeta, \zeta'}}^{d} \Psi_k(\beta, \beta').
\end{aligned}$$

Matrices of quadratic model

Define the following matrices, which depend on decompositions of both $\boldsymbol{S}$ and $\boldsymbol{T}$

$$\Phi_k\left(\alpha_{k-1}, \beta, \alpha_k\right) = \sum_{i_k=1}^{n_k} U_k\left(\alpha_{k-1}, i_k, \alpha_k\right) V_k\left(i_k, \beta\right)$$

Evaluation of the objective function

In terms of matrices of quadratic model:

$$\varphi = \frac{1}{2}\sum_{\beta=1}^{r}\sum_{\beta'=1}^{r} Q\left(\beta,\beta'\right) - \sum_{\beta=1}^{r}\sum_{\boldsymbol{\alpha}\in\mathcal{A}}\prod_{k=1}^{d}\Phi_k\left(\alpha_{k-1},\beta,\alpha_k\right) + \frac{1}{2}$$

Evaluation of the gradient

In terms of matrices of quadratic model:

$$\begin{aligned} \boldsymbol{g}_\zeta(\xi,\eta) &= \frac{\partial \varphi}{\partial V_\zeta(\xi,\eta)} = \\ &= \sum_{\beta=1}^{r} V_\zeta(\xi,\beta)\, Q_\zeta(\beta,\eta) \\ &- \sum_{\boldsymbol{\alpha}\in\mathcal{A}} U_\zeta(\alpha_{\zeta-1},\xi,\alpha_\zeta) \prod_{\substack{k=1\\k\neq\zeta}}^{d} \Phi_k(\alpha_{k-1},\eta,\alpha_k) \end{aligned}$$

Evaluation of matvec with the Hessian

In terms of matrices of quadratic model:

$$
\begin{aligned}
\boldsymbol{H}_{\zeta\zeta'}\left(\xi,\eta;\xi',\eta'\right) &= \frac{\partial^2\varphi}{\partial V_{\zeta'}\left(\xi',\eta'\right)\partial V_{\zeta}\left(\xi,\eta\right)}, \\
\boldsymbol{H} &= \boldsymbol{A}+\boldsymbol{B}+\boldsymbol{C}-\boldsymbol{D}, \\
\boldsymbol{A}_{\zeta\zeta'}\left(\xi,\eta;\xi',\eta'\right) &= \boldsymbol{\delta}\left(\zeta',\zeta\right)\boldsymbol{\delta}\left(\xi',\xi\right)Q_{\zeta}\left(\eta,\eta'\right), \\
\boldsymbol{B}_{\zeta\zeta'}\left(\xi,\eta;\xi',\eta'\right) &= \overline{\boldsymbol{\delta}}\left(\zeta',\zeta\right)V_{\zeta'}\left(\xi',\eta\right)V_{\zeta}\left(\xi,\eta'\right)Q_{\zeta\zeta'}\left(\eta,\eta'\right).
\end{aligned}
$$

Evaluation of matvec with the Hessian

In terms of matrices of quadratic model:

$$
\begin{aligned}
\boldsymbol{C}_{\zeta\zeta'}\left(\xi,\eta;\xi',\eta'\right) &= \overline{\boldsymbol{\delta}}\left(\zeta',\zeta\right)\boldsymbol{\delta}\left(\eta',\eta\right) \\
&\cdot \sum_{\beta=1}^{r} V_{\zeta}\left(\xi,\beta\right) V_{\zeta'}\left(\xi',\beta\right) Q_{\zeta}\left(\beta,\eta\right), \\
\boldsymbol{D}_{\zeta\zeta'}\left(\xi,\eta;\xi',\eta'\right) &= \overline{\boldsymbol{\delta}}\left(\zeta',\zeta\right)\boldsymbol{\delta}\left(\eta',\eta\right) \\
&\cdot \sum_{\boldsymbol{\alpha}\in\mathcal{A}} U_{\zeta}\left(\alpha_{\zeta-1},\xi,\alpha_{\zeta}\right) U_{\zeta'}\left(\alpha_{\zeta'-1},\xi',\alpha_{\zeta'}\right) \\
&\cdot \prod_{\substack{k=1 \\ k\neq\zeta,\zeta'}}^{d} \Phi_{k}\left(\alpha_{k-1},\eta,\alpha_{k}\right).
\end{aligned}
$$

Computation of the matrices of quadratic model

Computational costs are

$$\begin{aligned} \mathbf{Work} &= \mathcal{O}\left((d+n)dr^2 + d^2nrR(R+n)\right), \\ \mathbf{Mem} &= \mathcal{O}\left(d^2r^2 + d(dn+R)rR\right) \end{aligned}$$

In case $n = 2$

$$\begin{aligned} \mathbf{Work} &= \mathcal{O}\left(d^2r^2 + d^2rR^2\right), \\ \mathbf{Mem} &= \mathcal{O}\left(d^2r^2 + d(d+R)rR\right) \end{aligned}$$

Computation costs

Objective function φ

$$\begin{aligned} \textbf{Work} &= \mathcal{O}\left(r^2 + rR)\right), \\ \textbf{Mem} &= \mathcal{O}\left(1\right). \end{aligned}$$

Its gradient $\boldsymbol{g}$

$$\begin{aligned} \textbf{Work} &= \mathcal{O}\left(dnr^2 + dnrR)\right), \\ \textbf{Mem} &= \mathcal{O}\left(1\right). \end{aligned}$$

Computation costs

$\boldsymbol{A} > 0$

- preconditioner for the Hessian:

$$\textbf{Work} = \mathcal{O}\left(dr^3\right), \quad \textbf{Mem} = \mathcal{O}\left(dr^2\right)$$

- matvec:

$$\textbf{Work} = \mathcal{O}\left(dnr^2\right), \quad \textbf{Mem} = \mathcal{O}\left(1\right)$$

Computation costs

B, C or $B + C$

- matvec:

$$\textbf{Work} = \mathcal{O}\left((d+n)dr^2\right), \quad \textbf{Mem} = \mathcal{O}\left(dr^2\right)$$

D

- matvec:

$$\textbf{Work} = \mathcal{O}\left(d^2n^2r\right), \quad \textbf{Mem} = \mathcal{O}\left(1\right)$$

The tensor

$$\begin{pmatrix} c_1 & c_3 \\ c_2 & c_4 \end{pmatrix} = \begin{pmatrix} a_1 & a_3 \\ a_2 & a_4 \end{pmatrix} \begin{pmatrix} b_1 & b_3 \\ b_2 & b_4 \end{pmatrix},$$

$$c_k = \sum_{i=1}^{4} \sum_{j=1}^{4} s_{ijk} a_i b_j,$$

$S = \{s_{ijk}\}$ is a $4 \times 4 \times 4$-tensor: $d = 3$, $n = 4$, 8 ones and 56 zeros. Once we have a decomposition $S = (U, V, W)$ of rank 7, we may compute

$$c_k = \sum_{s=1}^{7} w_{ks} \sum_{i=1}^{4} u_{is} a_i \sum_{j=1}^{4} v_{js} b_j.$$

Results

2×2: $d = 3$, $n = 4$, $R = 8$, $r = 7$, $drn = 84$

	$\boldsymbol{e}_0$	IT	τ, s.	$\|\boldsymbol{g}\|_2$	$\boldsymbol{e}$
TT to Can, MATLAB	$1.0 \cdot 10^0$	23	2.4	$2.3 \cdot 10^{-8}$	10^{-8}
Can to Can, C	$1.0 \cdot 10^0$	21	0	$7.5 \cdot 10^{-12}$	$2.6 \cdot 10^{-8}$

Convergence

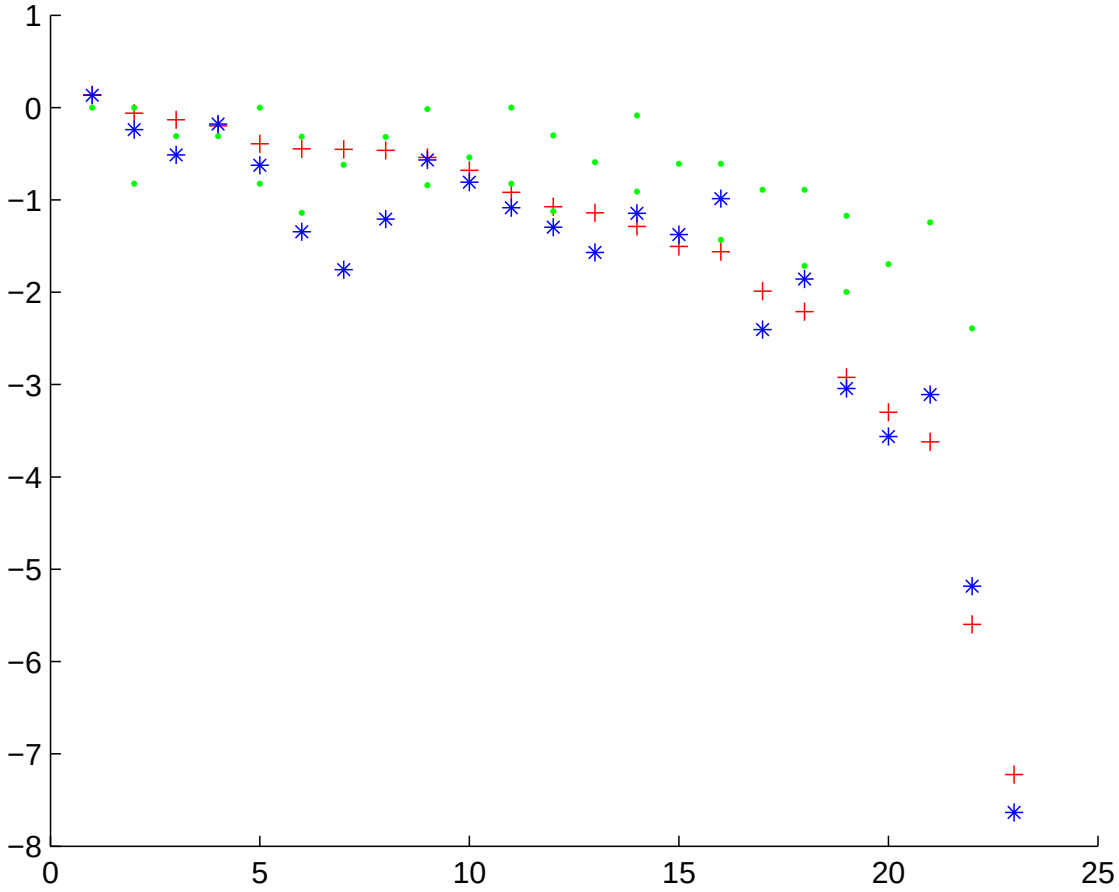


Рис.: e, $\|g\|_2$ and δ vs. iteration no.

Basic idea

- given a function $f : [0, 1] \to \mathbb{R}$, which is a sum of exponentials, and a mesh $\{x_i\}_{i=1}^{N} \subset [0, 1]$, $N = 2^d$
- consider a Q-mesh and a Q-vector: $\boldsymbol{S}\left(i_1 \dots i_d\right) = f\left(\sum_{k=1}^{d} 2^{k-1} i_k\right)$
- convert to TT: $\boldsymbol{S} = (U_1 \dots U_d)$
- approximate in Canonical with rank r: $\boldsymbol{T} = (V_1 \dots V_d) \approx (U_1 \dots U_d)$
- hope that $\mathrm{vec}\left(\boldsymbol{T}_\beta\right)$ is an exponential

Two exponentials

$f(x) = 0.1e^x - 0.003e^{5x}$, $d = 12$

$\boldsymbol{e}_0$	IT	τ, s.	$\|\boldsymbol{g}\|_2$	$\boldsymbol{e}$
$1.0 \cdot 10^0$	35	9.7	$1.6 \cdot 10^{-7}$	$1.5 \cdot 10^{-8}$

the same f with multiplicative 10% noise, $d = 12$

$\boldsymbol{e}_0$	IT	τ, s.	$\|\boldsymbol{g}\|_2$	$\boldsymbol{e}$
$1.0 \cdot 10^0$	42	10.8	$9.7 \cdot 10^{-12}$	$5.8 \cdot 10^{-2}$

Convergence (no noise)

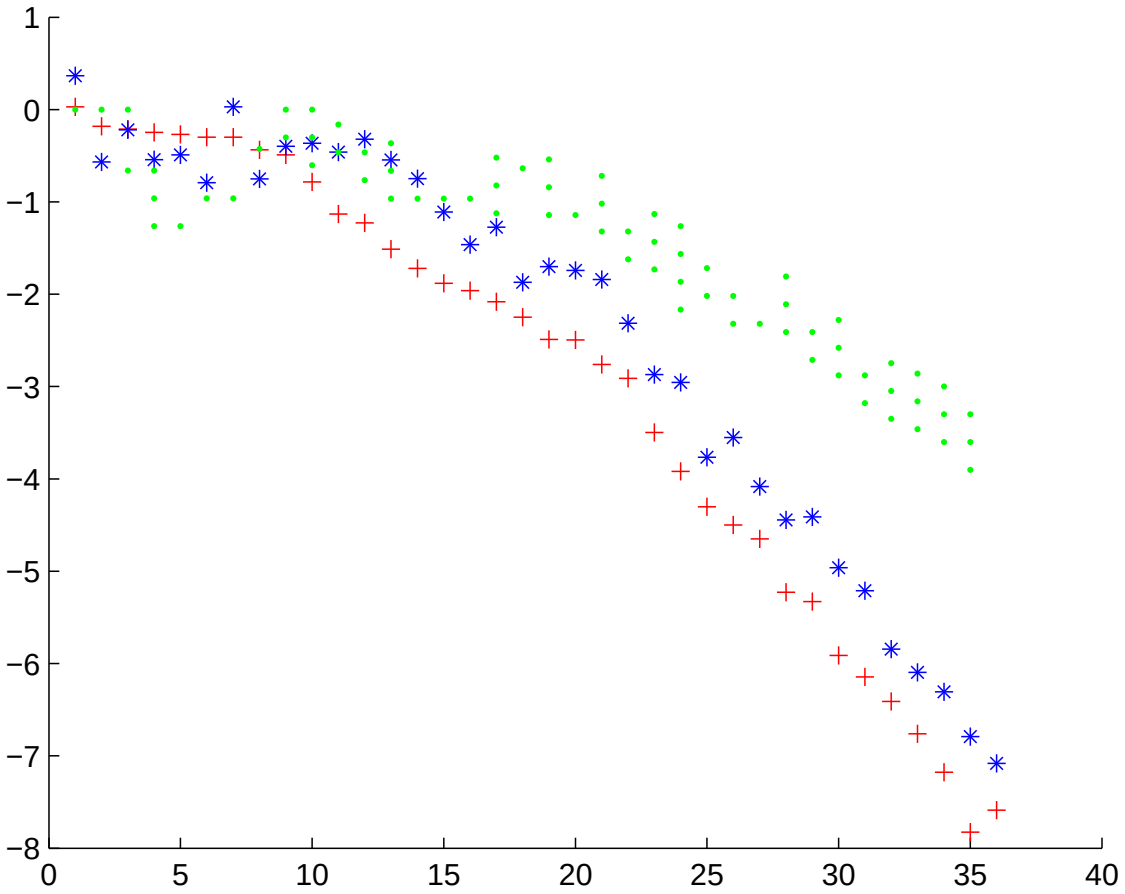
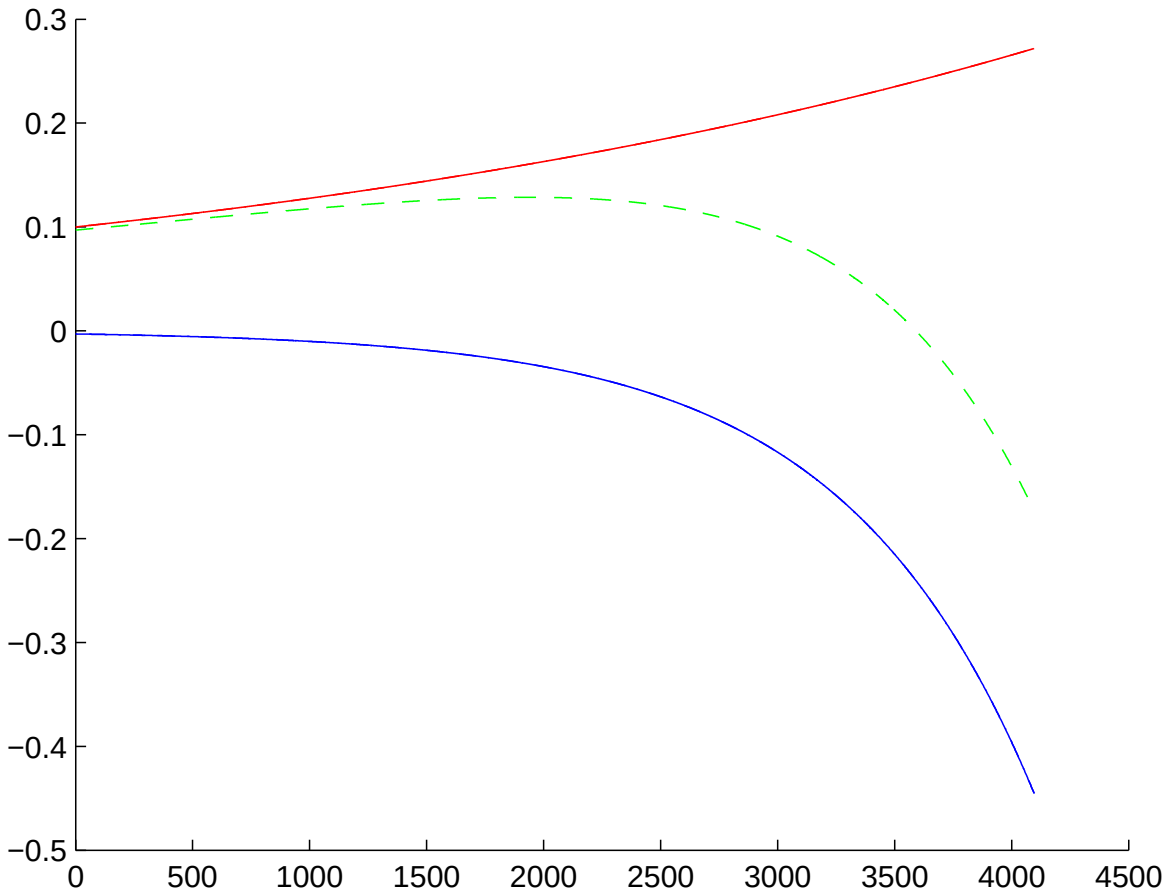
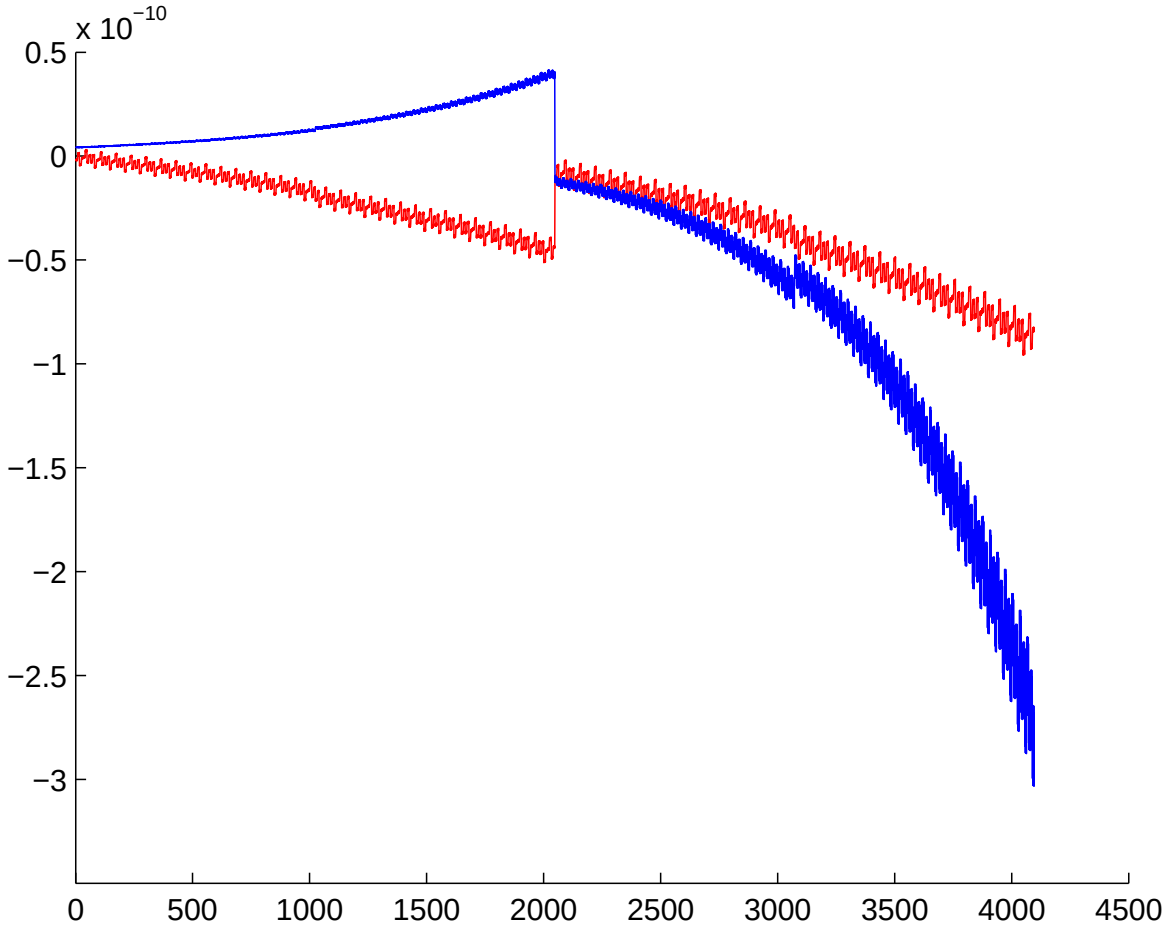


Рис.: e, $\|g\|_2$ and δ vs. iteration no.

Splitting (no noise)



Splitting error (no noise)



Convergence (with noise)

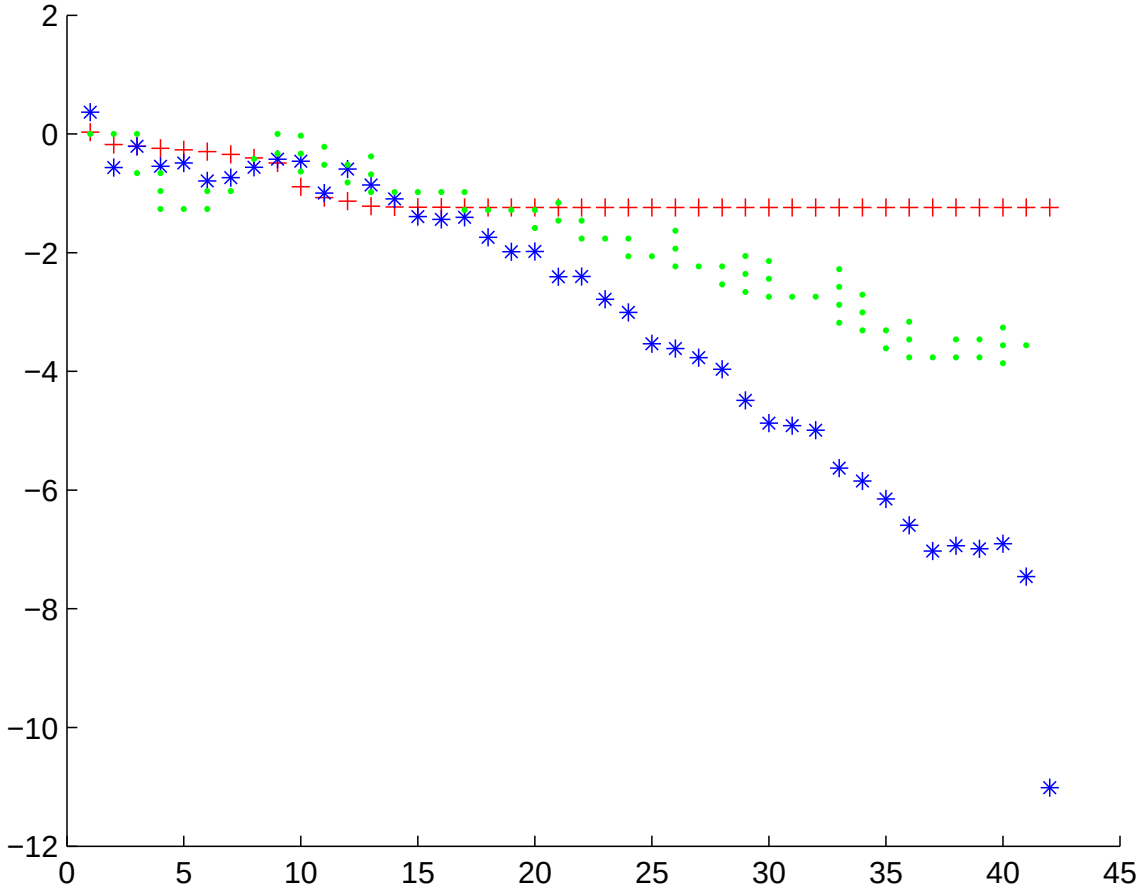
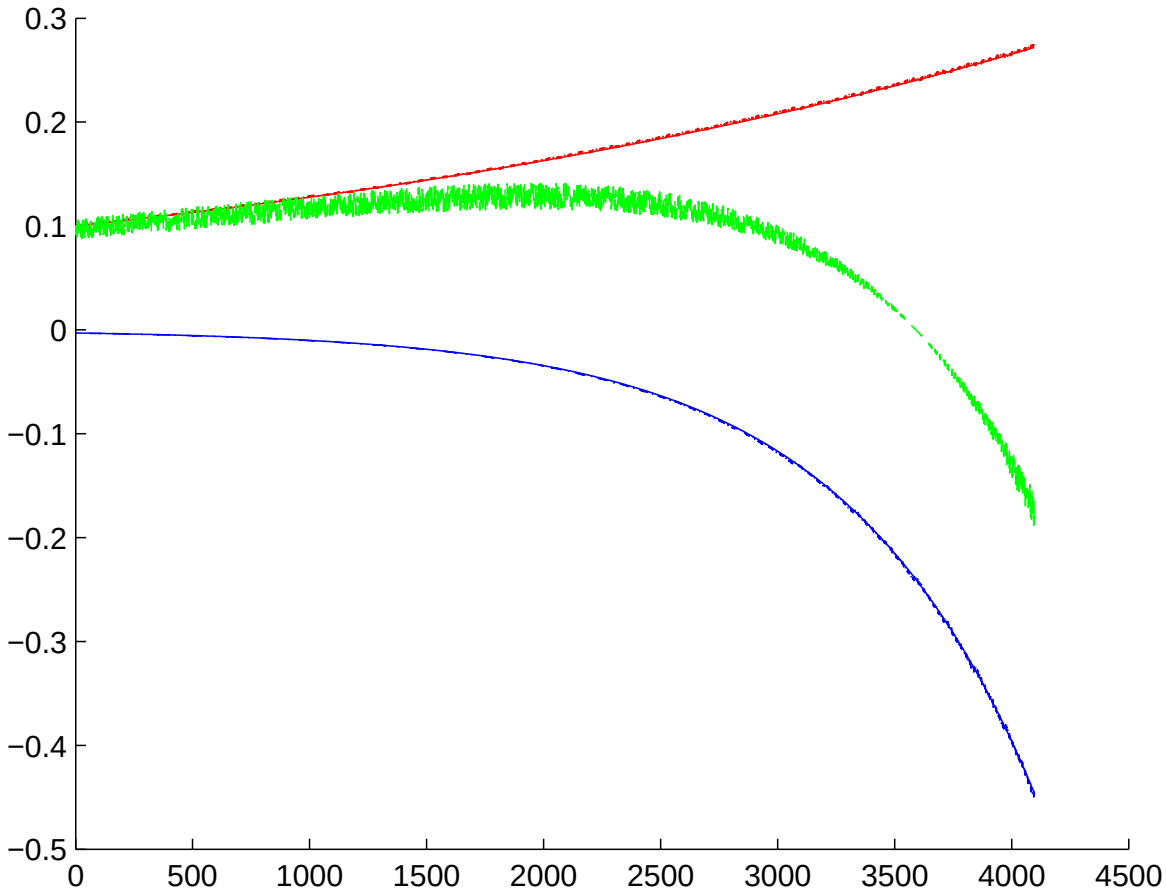
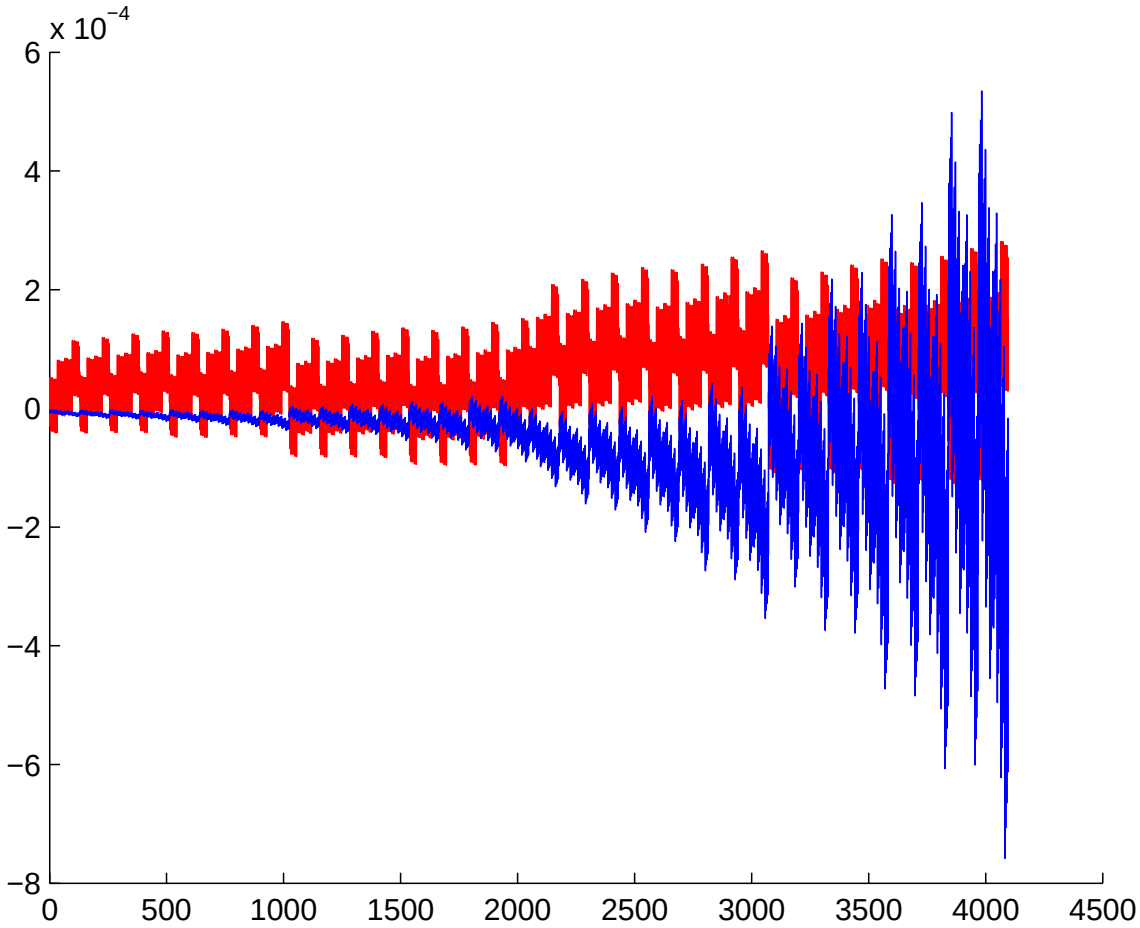


Рис.: e, $\|g\|_2$ and δ vs. iteration no.

Splitting (with noise)



Splitting error (with noise)



Three exponentials

$f(x) = 0.015e^{3x} - 0.003e^{5x} - 0.0001e^{10x}$, $d = 12$

$\boldsymbol{e}_0$	IT	τ, s.	$\|\boldsymbol{g}\|_2$	$\boldsymbol{e}$
$1.0 \cdot 10^0$	350	115	$3.0 \cdot 10^{-7}$	$6.2 \cdot 10^{-7}$

Convergence (no noise)

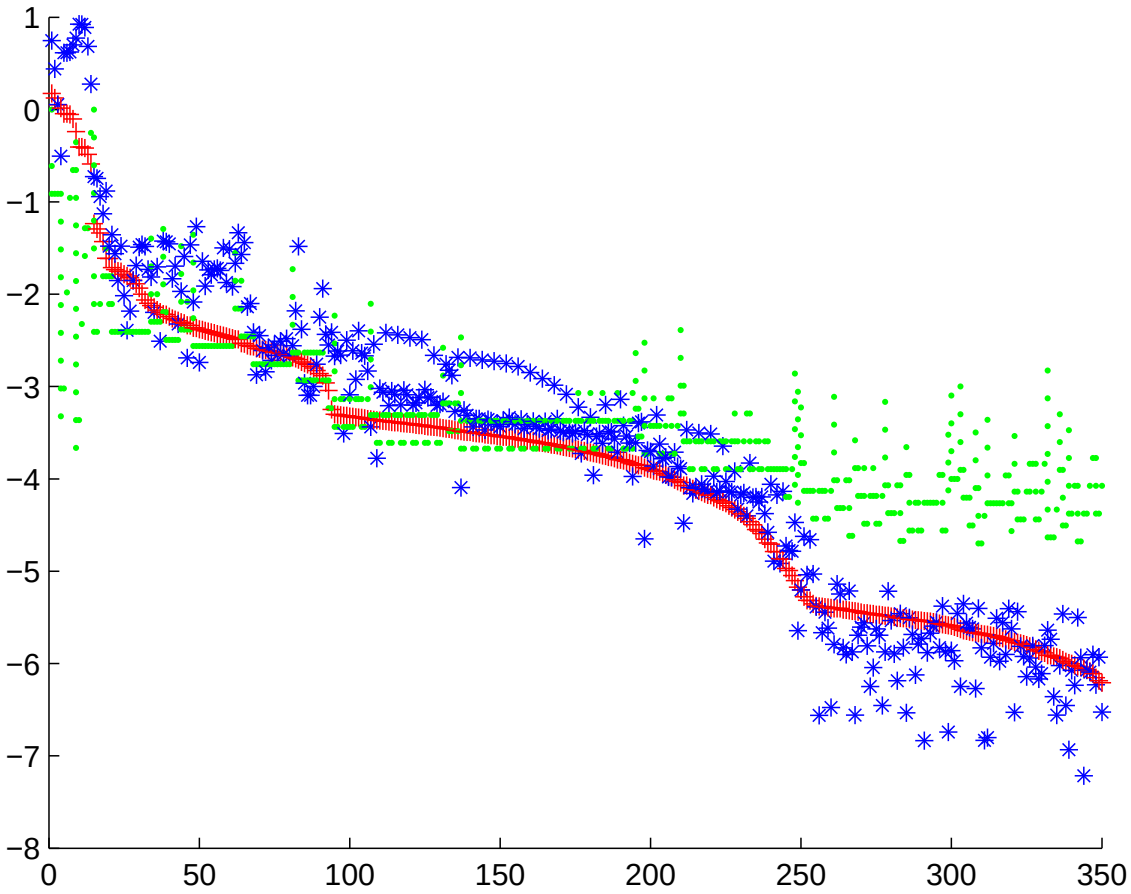
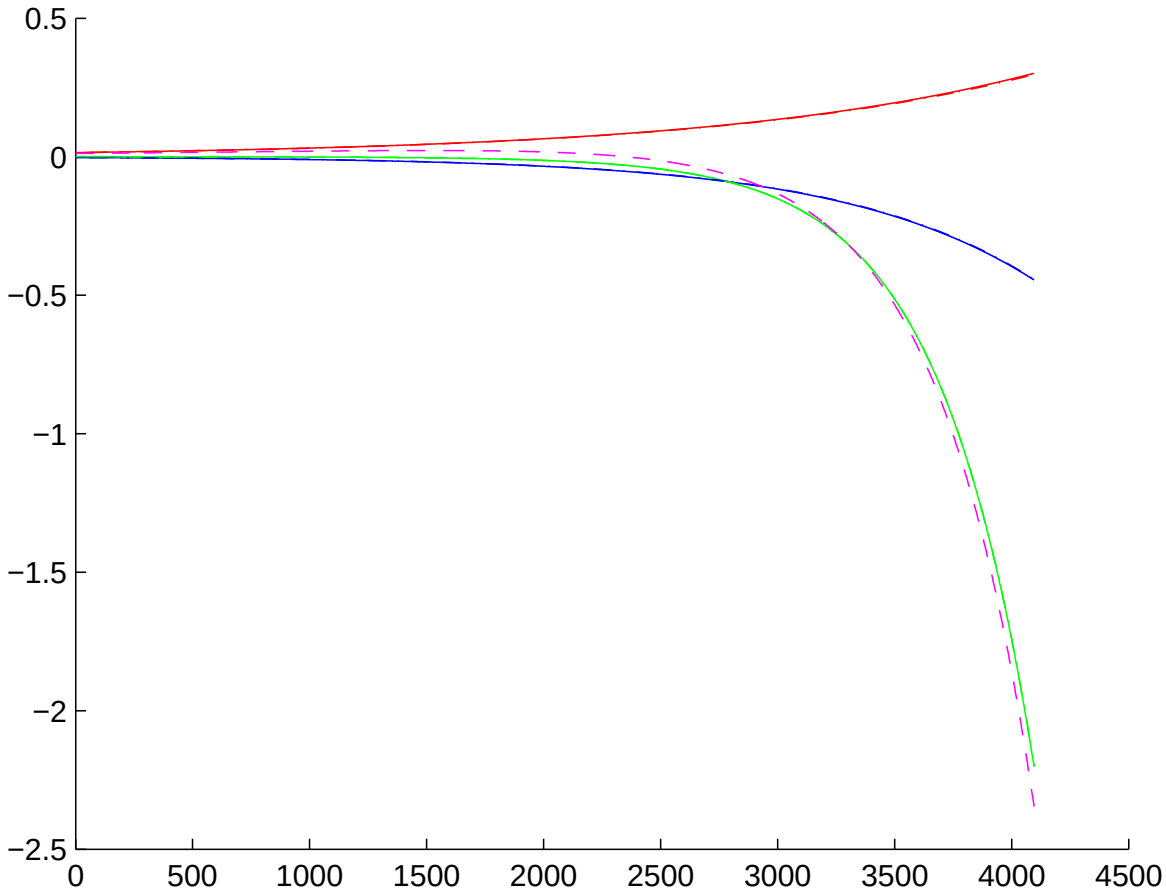
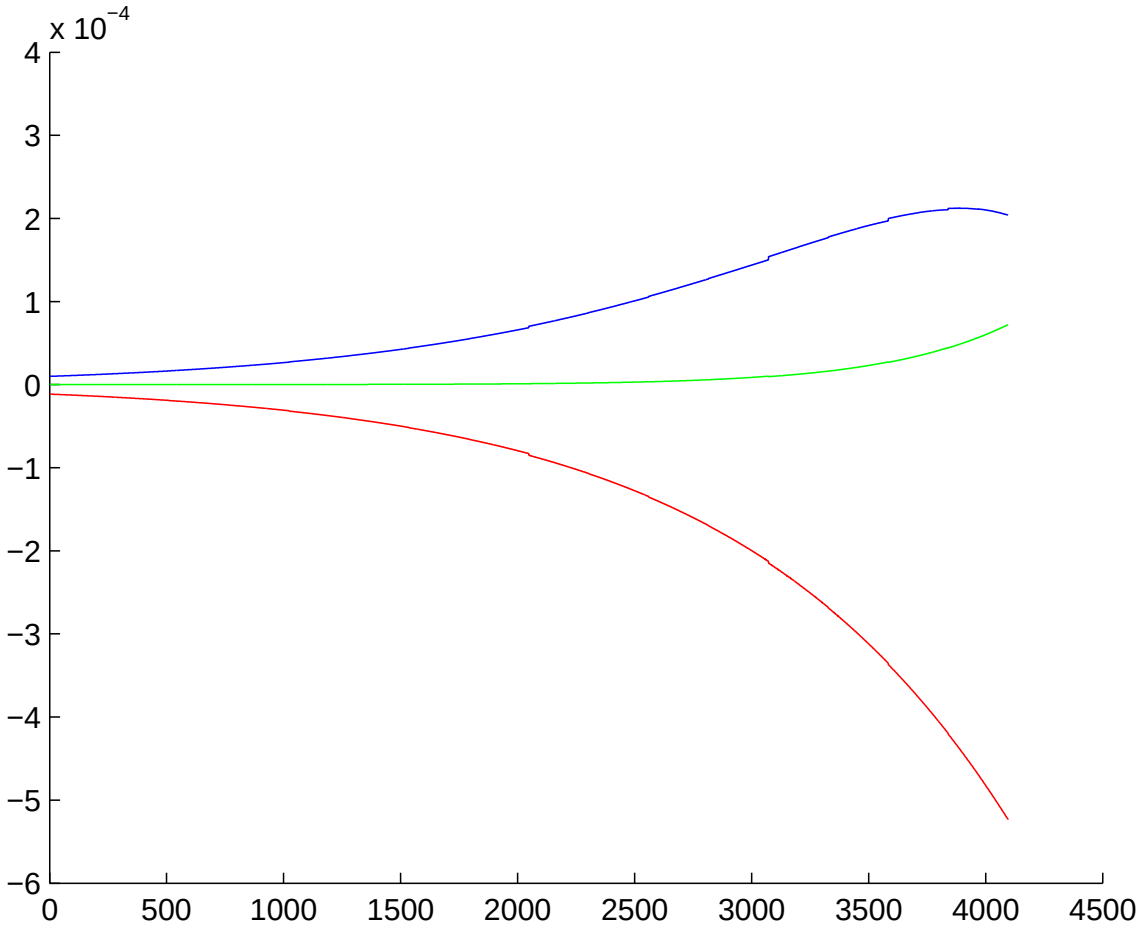


Рис.: e, $\|g\|_2$ and δ vs. iteration no.

Splitting (no noise)



Splitting error (no noise)



Thank you for your attention!