

# Materials 35 - ...And still estimating the anchoring function

Laura Gáti

June 30, 2020

## Overview

<b>1</b>	<b>Estimation procedure</b>	<b>2</b>
<b>2</b>	<b>1D function</b>	<b>3</b>
2.1	Simulated data, evaluate loss on a $6^6 = 46656$ grid . . . . .	3
2.2	Simulated data, $\alpha^{true} \in (0, 0.1)$ and convex . . . . .	5
2.3	100000 starting points, top 10 minima . . . . .	7
2.4	Add moments . . . . .	7
2.5	But is it robust? . . . . .	7
2.6	Is the L-shape a residue of too many coefficients? . . . . .	10
<b>3</b>	<b>Redo VFI and PEA using the best approximation of the gain evolution function</b>	<b>11</b>
<b>4</b>	<b>Other analysis</b>	<b>12</b>
<b>A</b>	<b>Model summary</b>	<b>13</b>
<b>B</b>	<b>Target criterion</b>	<b>13</b>

# 1 Estimation procedure

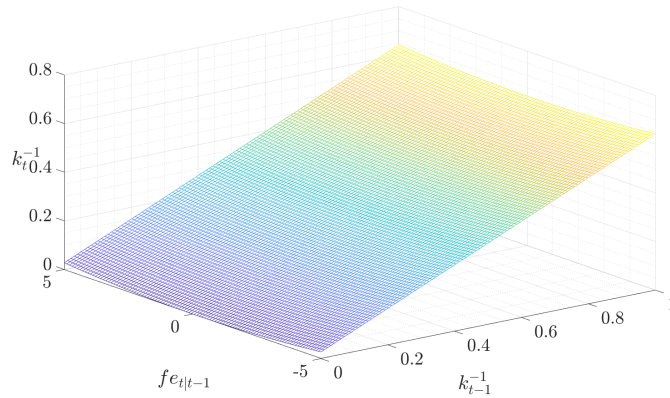
Instead of the AR(1) anchoring function used so far (Equation A.6), I use the following equation

$$k_t^{-1} = \alpha s(X) \quad (1)$$

where  $X = (k_{t-1}^{-1}, fe_{t|t-1})$  and I use piecewise linear interpolation. I initialize  $\alpha_0$  by specifying a grid for  $X$ , passing the grid through Equation (A.6) to generate  $k_t^{-1}$ -values, and approximating by fitting the grid to the  $k_t^{-1}$ -values. See Fig. 1.

Then I estimate  $\alpha$  using GMM, targeting the autocovariance structure of inflation, the output gap and the nominal interest rate (federal funds rate) in the data.

**Figure 1:** Initialization via Equation (A.6) implies this functional relationship

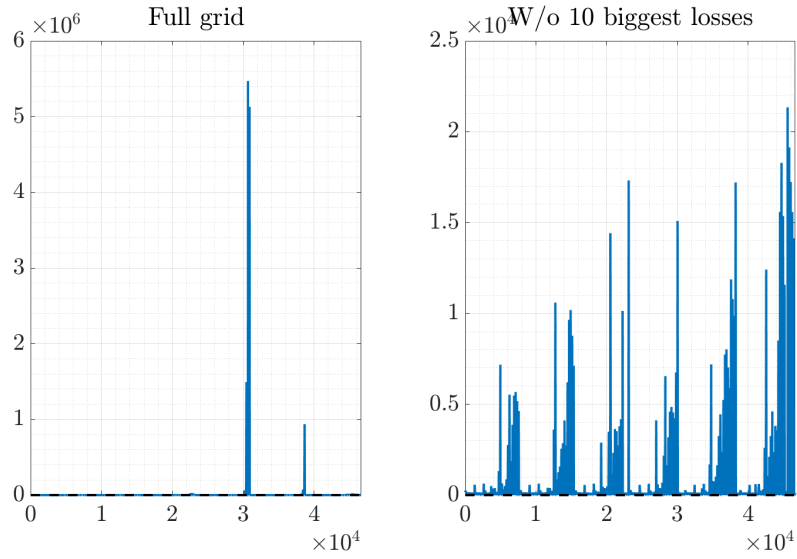


$T = 233$  before BK-filtering,  $T = 209$  after BK-filtering. Using the “constant-only, inflation-only” learning PLM. I drop the  $ndrop = 5$  initial values. I restrict  $\alpha \in (0, 1)$ , the support of  $k^{-1}$  in the grid. I target the lag  $0, \dots, 4$  autocovariance matrices, dropping repeated entries at lag 0, leaving me with 42 moments.

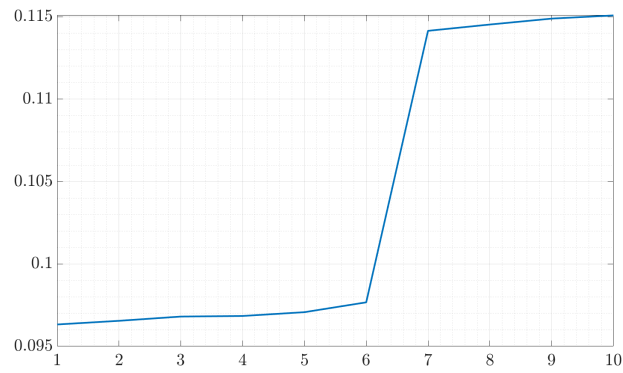
## 2 1D function

### 2.1 Simulated data, evaluate loss on a $6^6 = 46656$ grid

**Figure 2:** Objective function values on a grid with values  $(0; 0.2; 0.4; 0.6; 0.8; 1)$

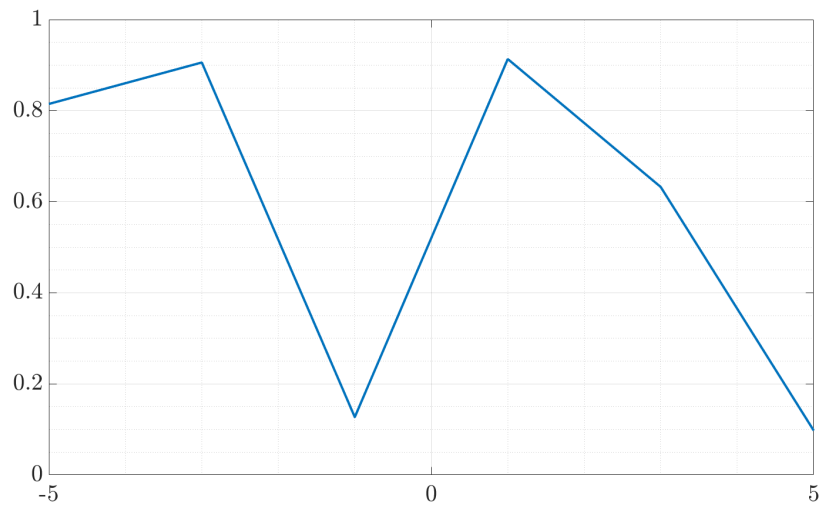


(a) Full grid and excluding the highest 10

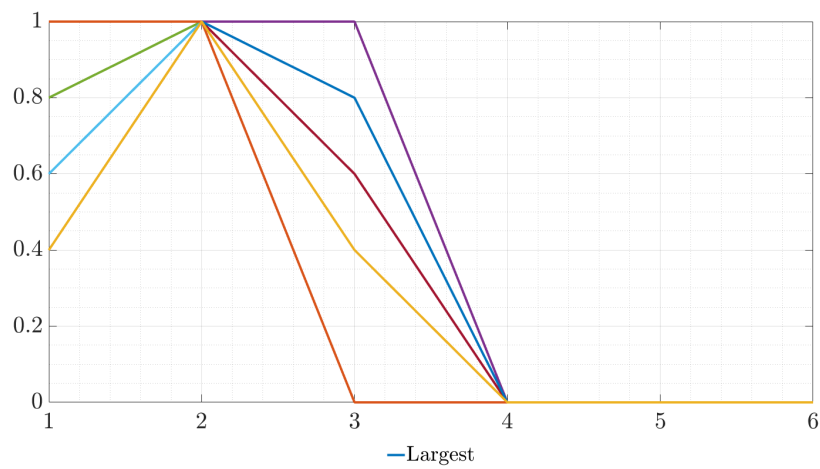


(b) The lowest 10 losses

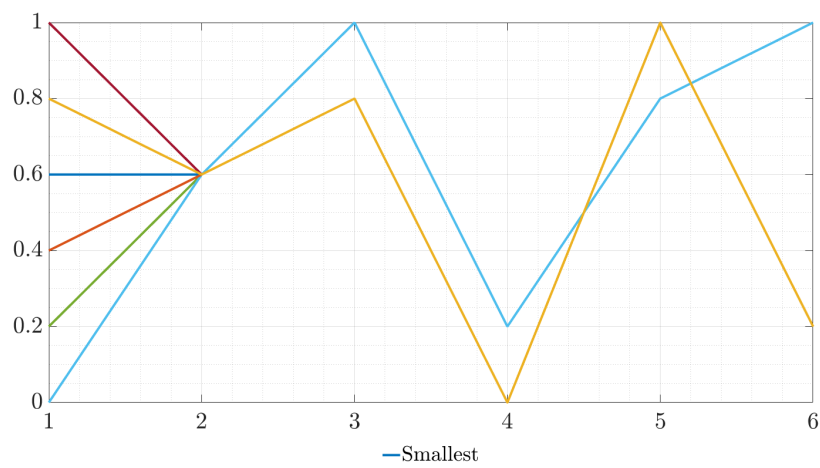
**Figure 3:** The  $\alpha$ s with highest and lowest objective function values



(a) True coefficients



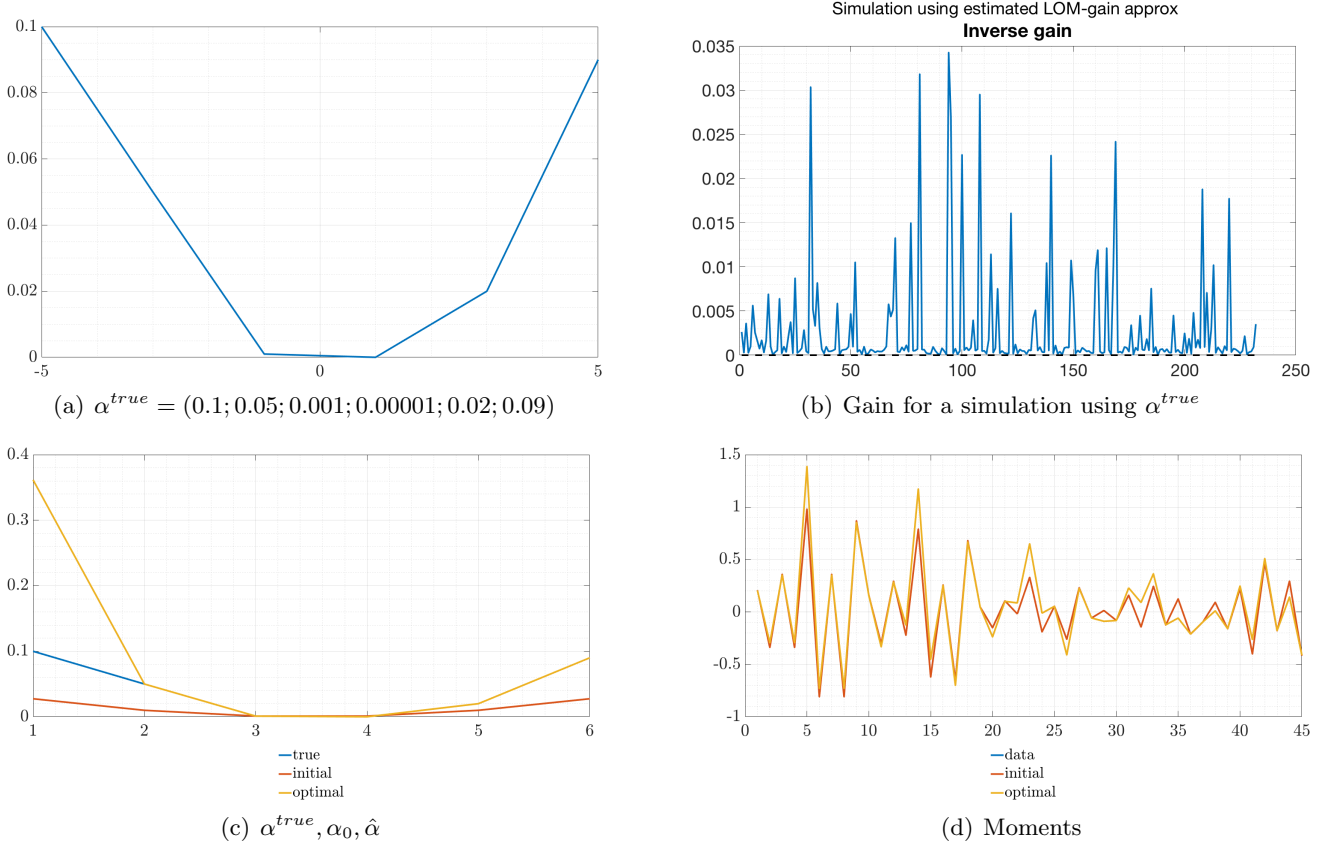
(b) Highest loss



(c) Lowest loss

## 2.2 Simulated data, $\alpha^{true} \in (0, 0.1)$ and convex

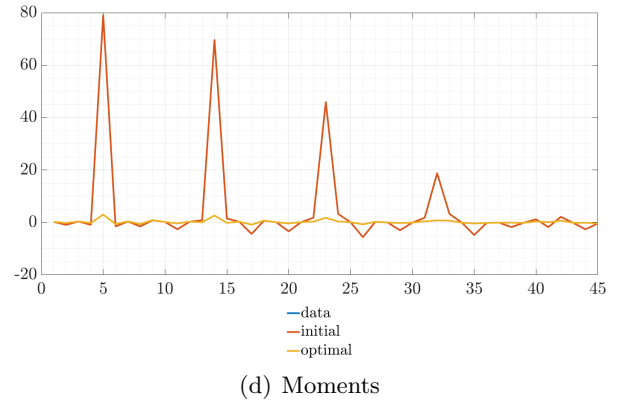
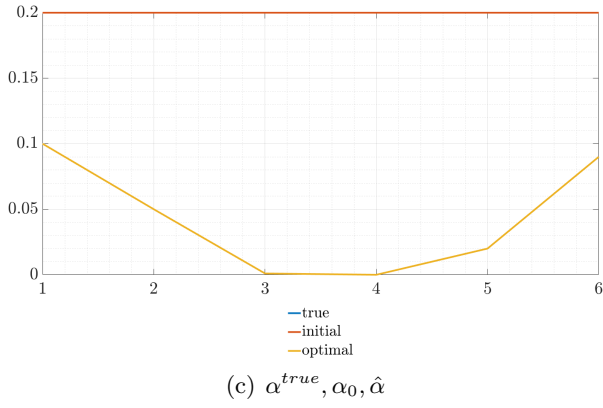
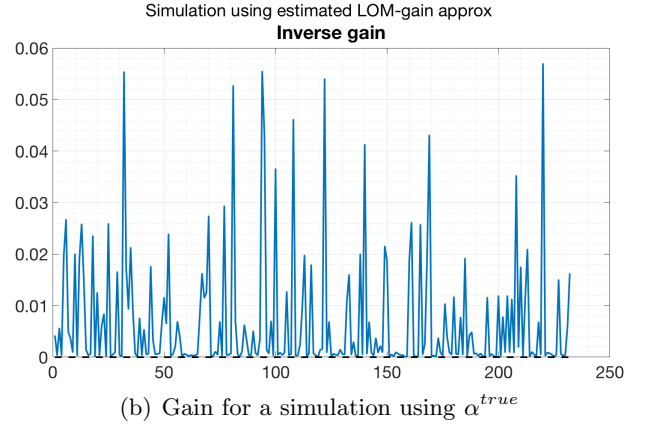
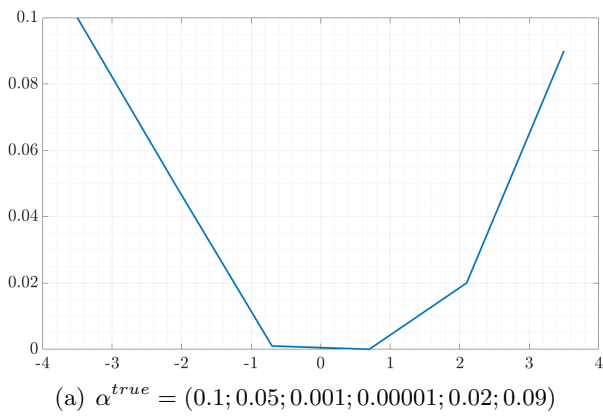
**Figure 4:** Effects of making the truth i) convex ii) between 0 and 0.1



Actually, this improves the solver's ability to get close (up to  $\alpha(fe = -5)$ ) dramatically! But this indicates the finite-element issue: there might not be any data in the -5 range for the forecast error. If that's so, then redoing this spiel with the true data involving a smaller forecast error support should do the trick.

BAM! That did it! PTO.

**Figure 5:** Effects of making the truth i) convex ii) between 0 and 0.1 iii) shrinking the true forecast error support to  $(-3.5, 3.5)$



## 2.3 100000 starting points, top 10 minima

Didn't improve on 1000, besides the nonconvex truth was too much of a challenge, so not pursuing this.

## 2.4 Add moments

### 2.4.1 Strict priors: anchoring function should be convex

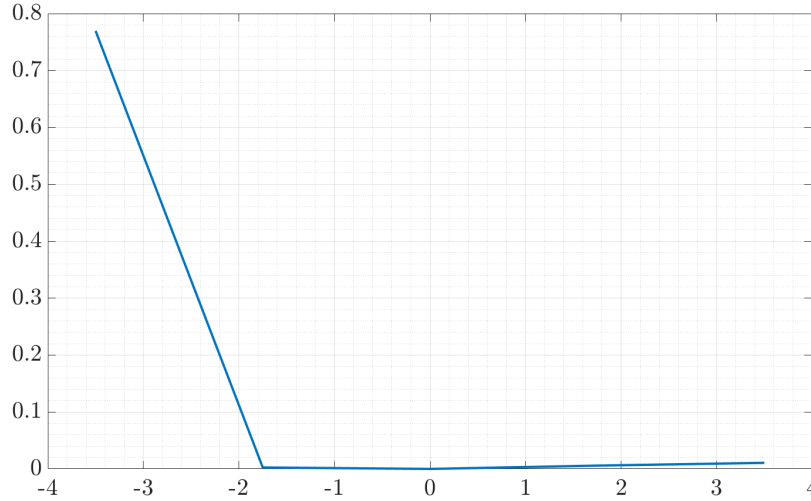
### 2.4.2 Calibrated moments: e.g. average gain in simulation should be 0.05

Starting with simulated data, and then turning to the real data, I've found that

- without (and also with!) additional moments, data is able identify 5 parameters;
- the convexity moment forces the solution to be convex (otherwise it is often not convex);
- the mean moment helps pinning down the solution when the convexity moment is in place.

Such a solution seems quite robust to starting points (there are some starting points that lead the solver not to converge). The solution looks like this:

**Figure 6:** The candidate solution

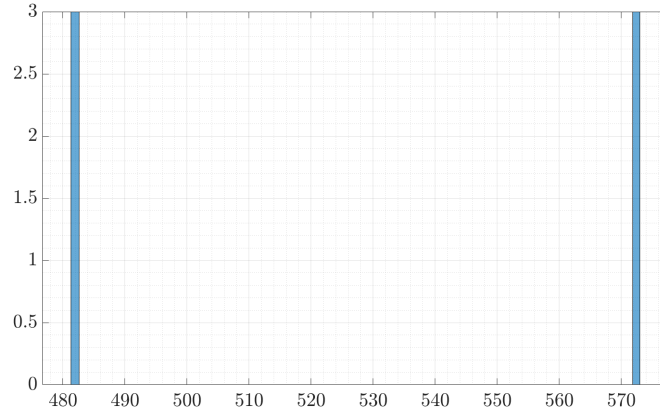


$n_\alpha = 5, fe \in (-3.5, 3.5), \alpha \in (0, 1)$ , convexity and mean moment imposed, starting point is given by the AR(1) gain function on this grid.  $\hat{\alpha} = (0.7696; 0.0026; 0; 0.0058; 0.0107)$

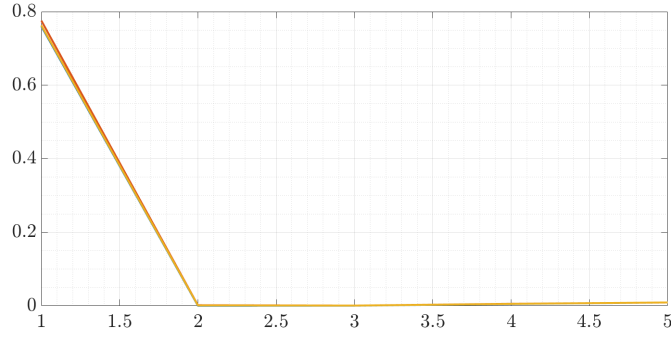
## 2.5 But is it robust?

Initialization for robustness check: random straight lines between (0,1). I've also done completely random sets of points and I get similar, but less clean results because initializing at non-convex points impairs the solver's behavior (gets stuck at more local minima).

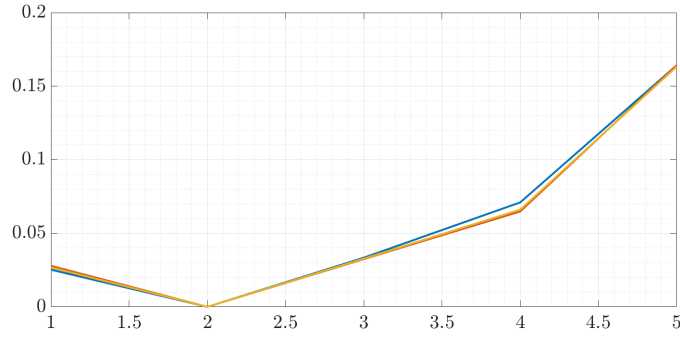
**Figure 7:** Converged solutions for 20 runs (6 converged)



(a) Residuals of converged solutions



(b)  $\hat{\alpha}^{Point1}$



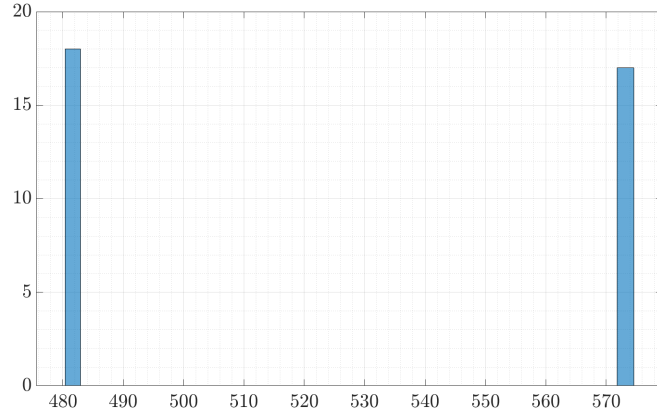
(c)  $\hat{\alpha}^{Point2}$

$n_\alpha = 5, fe \in (-3.5, 3.5), \alpha \in (0, 1)$ , convexity and mean moment imposed, from different straight lines between (0,1) as initial points.

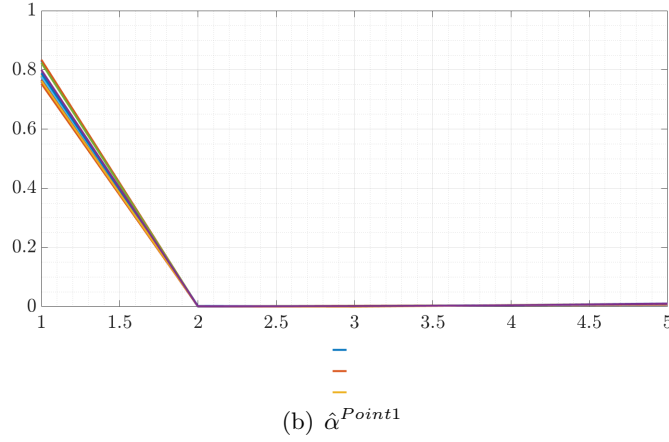
- Mean coefficients in Point 1 = (0.7663; 0.0011; 0.0007; 0.0054; 0.0092)
- Mean coefficients in Point 2 = (0.0267; 0; 0.0329; 0.0673; 0.1639)



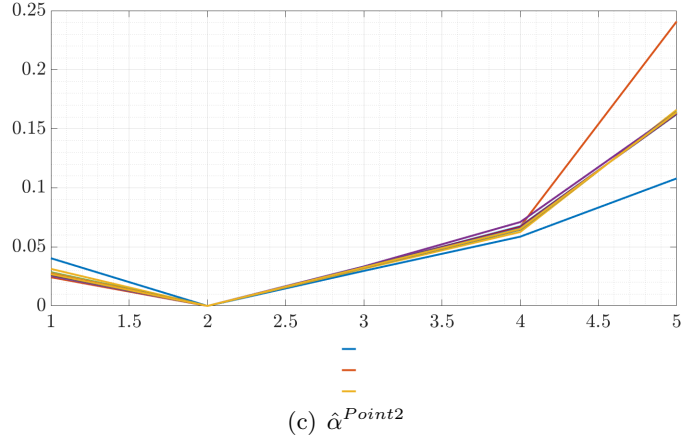
**Figure 8:** Converged solutions for 100 runs (35 converged)



(a) Residuals of converged solutions



(b)  $\hat{\alpha}^{Point1}$



(c)  $\hat{\alpha}^{Point2}$

$n_\alpha = 5, fe \in (-3.5, 3.5), \alpha \in (0, 1)$ , convexity and mean moment imposed, from different straight lines between (0,1) as initial points.

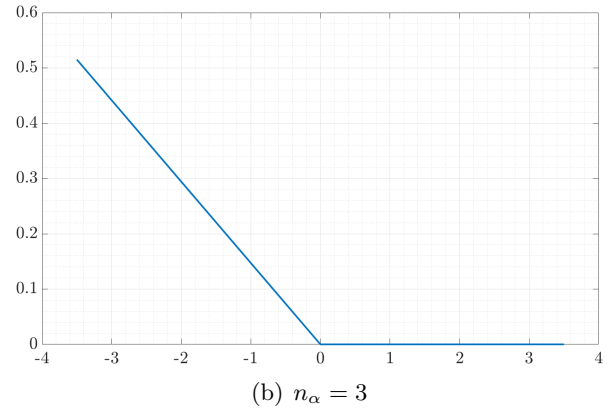
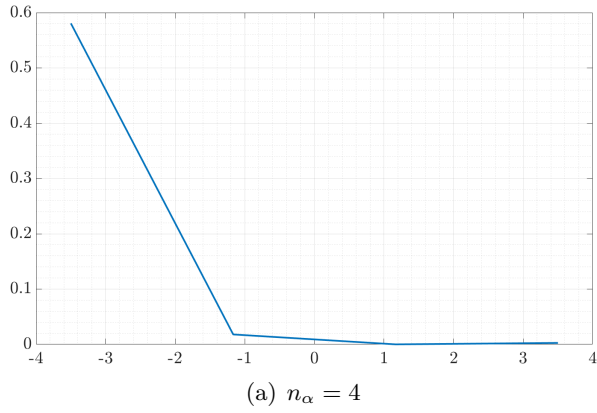
- Mean coefficients in Point 1 = (0.7816; 0.0014; 0.0012; 0.0054; 0.0091)
- Mean coefficients in Point 2 = (0.0279; 0; 0.0324; 0.0654; 0.1652)

- The very best solution:  $(0.8307; 0.0004; 0.0028; 0.0051; 0.0064)$ , residual = 480.3449.

## 2.6 Is the L-shape a residue of too many coefficients?

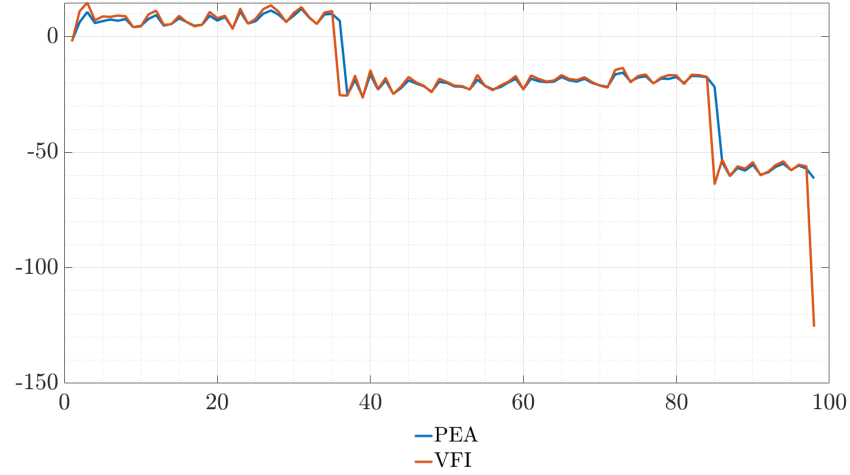
No:

**Figure 9:** The same configurations as above, except less parameters

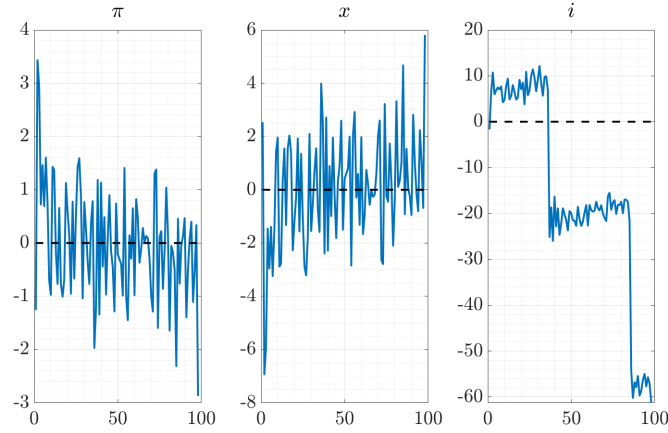


### 3 Redo VFI and PEA using the best approximation of the gain evolution function

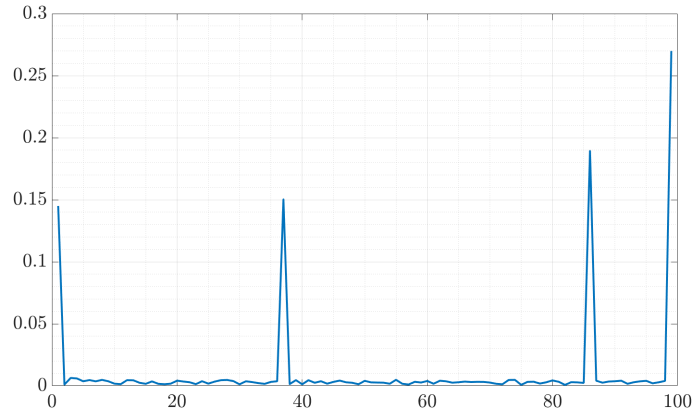
**Figure 10:** Optimal interest rate sequence conditional on a history of shocks



(a) Optimal policy: PEA vs VFI



(b) Observables under optimal policy conditional on a history of shocks

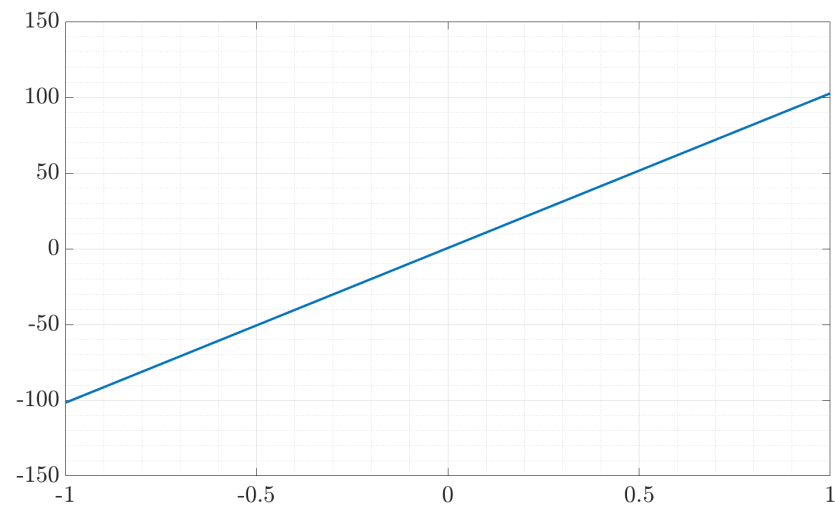


(c) Gain under optimal policy conditional on a history of shocks

---

## 4 Other analysis

**Figure 11:** Comparative statics of the policy function:  $\partial i / \partial \bar{\pi}$  if all other states are kept at their mean



## A Model summary

$$x_t = -\sigma i_t + \hat{\mathbb{E}}_t \sum_{T=t}^{\infty} \beta^{T-t} ((1-\beta)x_{T+1} - \sigma(\beta i_{T+1} - \pi_{T+1}) + \sigma r_T^n) \quad (\text{A.1})$$

$$\pi_t = \kappa x_t + \hat{\mathbb{E}}_t \sum_{T=t}^{\infty} (\alpha\beta)^{T-t} (\kappa\alpha\beta x_{T+1} + (1-\alpha)\beta\pi_{T+1} + u_T) \quad (\text{A.2})$$

$$i_t = \psi_\pi \pi_t + \psi_x x_t + \bar{i}_t \quad (\text{if imposed}) \quad (\text{A.3})$$

$$\text{PLM:} \quad \hat{\mathbb{E}}_t z_{t+h} = a_{t-1} + b h_x^{h-1} s_t \quad \forall h \geq 1 \quad b = g_x h_x \quad (\text{A.4})$$

$$\text{Updating:} \quad a_t = a_{t-1} + k_t^{-1} (z_t - (a_{t-1} + b s_{t-1})) \quad (\text{A.5})$$

$$\text{Anchoring function:} \quad k_t^{-1} = \rho_k k_{t-1}^{-1} + \gamma_k f e_{t-1}^2 \quad (\text{A.6})$$

$$\text{Forecast error:} \quad f e_{t-1} = z_t - (a_{t-1} + b s_{t-1}) \quad (\text{A.7})$$

$$\text{LH expectations:} \quad f_a(t) = \frac{1}{1-\alpha\beta} a_{t-1} + b(\mathbb{I}_{nx} - \alpha\beta h)^{-1} s_t \quad f_b(t) = \frac{1}{1-\beta} a_{t-1} + b(\mathbb{I}_{nx} - \beta h)^{-1} s_t \quad (\text{A.8})$$

This notation captures vector learning ( $z$  learned) for intercept only. For scalar learning,  $a_t = (\bar{\pi}_t \ 0 \ 0)'$  and  $b_1$  designates the first row of  $b$ . The observables  $(\pi, x)$  are determined as:

$$x_t = -\sigma i_t + \begin{bmatrix} \sigma & 1-\beta & -\sigma\beta \end{bmatrix} f_b + \sigma \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} (\mathbb{I}_{nx} - \beta h_x)^{-1} s_t \quad (\text{A.9})$$

$$\pi_t = \kappa x_t + \begin{bmatrix} (1-\alpha)\beta & \kappa\alpha\beta & 0 \end{bmatrix} f_a + \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} (\mathbb{I}_{nx} - \alpha\beta h_x)^{-1} s_t \quad (\text{A.10})$$

## B Target criterion

The target criterion in the simplified model (scalar learning of inflation intercept only,  $k_t^{-1} = \mathbf{g}(f e_{t-1})$ ):

$$\pi_t = -\frac{\lambda_x}{\kappa} \left\{ x_t - \frac{(1-\alpha)\beta}{1-\alpha\beta} \left( k_t^{-1} + ((\pi_t - \bar{\pi}_{t-1} - b_1 s_{t-1})) \mathbf{g}_\pi(t) \right) \right. \\ \left. \left( \mathbb{E}_t \sum_{i=1}^{\infty} x_{t+i} \prod_{j=0}^{i-1} (1 - k_{t+1+j}^{-1} - (\pi_{t+1+j} - \bar{\pi}_{t+j} - b_1 s_{t+j}) \mathbf{g}_{\bar{\pi}}(t+j)) \right) \right\} \quad (\text{B.1})$$

where I'm using the notation that  $\prod_{j=0}^0 \equiv 1$ . For interpretation purposes, let me rewrite this as follows:

$$\pi_t = -\frac{\lambda_x}{\kappa} x_t + \frac{\lambda_x}{\kappa} \frac{(1-\alpha)\beta}{1-\alpha\beta} \left( k_t^{-1} + f e_{t|t-1}^{eve} \mathbf{g}_\pi(t) \right) \mathbb{E}_t \sum_{i=1}^{\infty} x_{t+i} \\ - \frac{\lambda_x}{\kappa} \frac{(1-\alpha)\beta}{1-\alpha\beta} \left( k_t^{-1} + f e_{t|t-1}^{eve} \mathbf{g}_\pi(t) \right) \left( \mathbb{E}_t \sum_{i=1}^{\infty} x_{t+i} \prod_{j=0}^{i-1} (k_{t+1+j}^{-1} + f e_{t+1+j|t+j}^{eve} \mathbf{g}_{\bar{\pi}}(t+j)) \right) \quad (\text{B.2})$$

Interpretation: **tradeoffs from discretion in RE** + **effect of current level and change of the gain on future tradeoffs** + **effect of future expected levels and changes of the gain on future tradeoffs**