Work after

✓1) I wasn't computing the TC correctly b/c in the last H periods, should feed in $hx^j s_t$ for shock $\{s_t\}_{T-H}^{T}$.

✓2) If I don't impose TC, should impose TR w/ a res, or w/o a res but then have a res in A1 or A2.

3) Should check the fsolve thing.

2) Adding TR as a residual eq doesn't change much at all for $\{\pi, x, i\}_t$! (stopped prematurely.) Making the norm the loss concept however gets me much closer to the Taylor-rule! Adding TR as a res did bring things closer to the TR-outcome for $\{x, i\}_t$ as input. Here the norm doesn't make a big diff. (stopped prematurely). Adding TR for $\{i_t\}$ as only input. (local min but VERY unstable.) For norm unstable too, stopped premat. But it no longer blows up in my face :)

1) Computing the arch π w/ correct $E(\cdot)$ doesn't change things. Does the norm? It finds a local min, but a strange one. (Both for $\{i_t\}$ only.)

For $\{x_t, i_t\}$, norm or no norm makes no diff, but I'm in general much closer to TR than before, if I initialize there. If not, then further.

For $\{\pi_{t-1}, x_{t-1}, i_t\}$ as input, I get the same as before. w/ norm almost the same.
Not if I initialize at TR, I get a strange local min. With norm too.

↳ Either the method is unstable or I'm not quite doing it correctly

→ Tomorrow: do fsolve thing

✓ · Fix H   8 April 2020

✓ · Fix too long figtitles

✓ · Fix fsolve for TR & RE-TC → fixed! (resids are not defined for $t=1$)

- String is f' up w/ the anch-TC.

↳ Or: I don't know if the sol. procedure is unstable, or if I'm doing stg wrong.

Back to VFI: Eric Sims does it by simpl' plug in the constraint to rewrite the problem as a fct of the future state:

We had: $V^{l+1}(k_t) = \max_{c_t} \ln(c_t) + \beta V^l(k_{t+1})$

$$\text{s.t. } k_t^\alpha - c_t \geq k_{t+1}$$

$$\rightarrow c_t = k_t^\alpha - k_{t+1}$$

$$V^{l+1}(k_t) = \max_{k_{t+1}} \ln(k_t^\alpha - k_{t+1}) + \beta V^l(k_{t+1})$$

Rewrite in terms of k, k'

$$V^{l+1}(k) = \max_{k'} \ln(k^\alpha - k') + \beta V^l(k')$$

Ok that's fine but I'm idiot – we need the grid for k_t nonetheless! In the nonstochastic world, we just need no evaluation of expectations and possibly no markov chains.

Btw, Eric Sims calls the choice of k' the policy fct. But I guess that's isomorphic w/ choosing c_t.

I've found Collard's value function iteration notes. (value_function_collard_lectnotes7.pdf)

→ it seems to suggest that you use the grid for k and k' for $i = 1 : n_{grid}$

$k_t = kgrid(i)$

for $j = 1 : n_{grid}$ * there's a complication, but ignore for now.

$k_{t+1} = kgrid(j)$

$c(i,j) = c(k_t, k_{t+1}) \rightarrow U(i,j) = u(c(i,j))$

$n_{grid} \times 1$

end

$j^* = \max[\, U(c(i,j)), 2 \,]$

$k^*_{t+1} = kgrid(\, j^* \,)$

$\underset{1\times 1}{V^{new}(i)} = u(c(i,j^*)) + \beta \cdot \underset{1\times 1}{V(j^*)}$

end

$crit = \max(abs(V^{new} - V))$

Actually Collard does this

$\underset{1\times n_{grid}}{c(i,:)}$ is c when $k_t = kgrid(i)$, and for all k_{t+1}

$\rightarrow util(i,:) = U(c(i,:))$ $1 \times n_{grid}$

$[V^{new}, j^*] = \max[\, u(c(i,:)) + \beta \underset{n_{grid}\times 1}{V(:)} \,]$

* the additional complication is that $c_t, k_{t+1} > 0$

$$c_t > 0 \Rightarrow c_t = k_t^\alpha - k_{t+1} > 0$$

$$k_{t+1} > 0 \Rightarrow \boxed{k_t^\alpha > k_{t+1} > 0}$$

The grid makes sure that $k_{t+1} > 0$, but I need to check that k_{t+1} isn't $> k_t^\alpha(i)$ for each i.

Collard's VFI notes are helpful b/c they clarify what the maximization means and what $V(k_{t+1})$ means. <u>9 April 2020</u>

$V^{old}(k_{t+1})$ just means $V^{old}(:)$ for all values of the kgrid (k_{t+1}).

Similarly, you evaluate $c(k_t(i), :)$ for all values of the kgrid (k_{t+1}).

The maximization then is just to choose the index j^* that max $u(c(k_t(i), :) + \beta V^{old}(:)$, and that for each i. So $k_{t+1}(i) = \text{kgrid}(j)$.

and $V^{new}(i) = u(c(k_t(i), j^*) + \beta V^{old}(j^*)$.

FSOLVE / FMINCON

↳ could try to attack from several, or smart initialization points.

A potential problem for the anchoring TC: overparameterized?

I input T+H = 60 periods, but I can evaluate only T residuals.

(1) Comparing FSOLVE vs. FMINCON

(1.1) Taylor-rule

Fmincon usually gets more solutions, even when fsolve says no sol.

(1.2) RE-TC

Same

(1.3) Anch-TC

Same.

I think this just means that fmincon is a little more easy-going w/ the sol. crit, since fsolve seeks to set $F(x) = 0$, while fmincon just tries to min $F(x)$

FSOLVE never finds a sol (it stalls once: $\{x, i\}_{+}$ in RE-TC, rand init.)
FMINCON always finds at least 1 or 2 local mins.
But they are of 3 groups
- nonsensical: extremely volatile & huge
- really close to TR
  - ↳ for TR & RE-TC → makes sense b/c you're either implementing the TR or the TR is a good implementation of RE-TC
- less volatile for x & i and more for π for anch-TC
  - ↳ anch-TC calls for a less aggressive TR!

I actually think I should prioritize the "approximating the reaction function" approach of Peter over the value function iteration b/c it's more promising in terms of results.

But first I try reoptimizing by inputting the first $\{i_t^*\}$ as an initial guess for $\{i_t\}$.
→ doesn't work either.

command_approx_reaction.m

Problems:

- ✓ g_π is blowing up → a problem for the previous exercise too.
- Besides, in previous ex., was still evaluating the anchTC wrong b/c at each t, need to account for CB's E(·) of future shocks
- loss isn't decreasing in any direction.

✓ forgot $g_{\bar{\pi}}$.

$$\frac{\partial}{\partial \bar{\pi}} \frac{1}{(d\,fe)^2} = \frac{\partial fe}{\partial \bar{\pi}} \left(-2 \frac{1}{(d fe)^2} fe^{-1} \right)$$

$$= 2 \frac{1}{(d fe)^2} fe^{-1} =: g_{\bar{\pi}}^{-1}$$

The problem w/ this is that $fe^{-3} \to \infty$ if fe is small.

I've inverted g_π and $g_{\bar{\pi}}$ b/c in the materials, I've defined $k_t^{-1} = g(\cdot)$, while in the code I have $k_t = g(\cdot)$.

✓Next: need to be smart about anchoring fcst. Correct evaluation of anchTC in the fsolve/fmincon exercise. Regenerate.

$fe = \pi_t - (\bar{\pi}_{t-1} + b s_{t-1})$

$$k_t = g(fe) = \frac{1}{d fe^2}$$

$$k_t^{-1} = g(fe)^{-1} = d fe^2$$

$$\frac{\partial k_t^{-1}}{\partial \pi_t} = 2 d fe =: g_\pi \quad \text{vs.} \quad g_\pi^{-1}$$

$$\frac{\partial k_t^{-1}}{\partial \bar{\pi}_{t-1}} = -2 d fe =: g_{\bar{\pi}} \quad \text{vs.} \quad g_{\bar{\pi}}^{-1}$$

- Still loss isn't decreasing
- Still TC eval is wrong for previous.
  ↳ I think I fixed that now – it takes waay longer to run → 25-30 min?
  → Took 17 minutes! (47 iter)
  Or up to 30!

While it's running, let's work thru Collard's VFI

$$u(c) = \frac{c^{1-\gamma} - 1}{1-\gamma}$$

$$k' = k^{\alpha} - c + (1-\delta)k$$

$$\Rightarrow V(k) = \max_{c} \frac{c^{1-\gamma}-1}{1-\gamma} + \beta V(k'), \text{ or,}$$

plugging in $LOM(k)$ for c,

$$V(k) \max_{k'} \frac{(k^{\alpha} + (1-\delta)k - k')^{1-\gamma} - 1}{1-\gamma} + \beta V(k')$$

Let's make sure that for no grid value of $k(i)$, $k'(j)$ is $c<0$ → for any $k(i)$, $k^{\alpha}(i) + (1-\delta)k(i) - k'(j) \overset{!}{>} 0$

I think I'm still evaluating the TC wrong 10 April 2020
- in all exercises, b/c if you input zero innovations, that's gonna change fa & fb too. NO - it's correct b/c those are fb & fa that the CB *expects* people to have. *Phew!*

Now I have the problem that sim_learnLH.m doesn't work. It doesn't produce IRFs, and the simulation doesn't seem to converge to RE either.

Sorry - it was b/c I turned σ & ρ of shocks to 0. IRFs are working.

And sorry, it is converging, I plotted diff and it does.

I'm cleaning up!

sim_learnLH.m is no longer touched!

sim_learnLH_clean.m = simlearnLH.m and is meant as a basis for subsequent work. (materials24.m compares the two.)

sim_leanLH_clean_smooth.m is meant for the smooth anchoring function only!

Ryan meeting 10 April 2020

→ should get a resid of zero if init at Taylor rule sequence.

→ feed fsolve the T residuals and not the square but the level whereas fminicon will need the square.

Don't use fminicon b/c it takes lots of info and since it uses a scalar

lsqnonlin

↳ similar to fminicon

Fsolve is the way to go when you know a resid of 0 exists. Else lsqnonlin b/c it won't hit 0.

fsolve

1st thing to check: # of exog vars & # equations

→ you can check that unstable in EE or in the last period

A brute force: if $T=200$, $H=100$, then at $t=150$, the $E(\cdot)$ in 2nd line of (B.1) is fixed.

An implementable TC sounds interesting for a discussion. It can also be an option for this optimization. But it's not a concern for the CB unless communicating w/ the public is a concern.

Work after

- So back to fsolve: need to find in the "implement the TR" setup where the mistake lies.
- just some thoughts:

  Ryan seems to think that Peter's approach is kinda "ass-backwards" (he said "circular", I think)

  → why not max the (-loss) directly?

⇒ which makes me wonder if I can solve for the optimal $\{i_t\}$ w/o value fct iteration?

I mean, $\min \sum (\pi_t^2 + \lambda_x x_t^2)$

s.t. model equations?

→ Why bother w/ VFI???

Hmm, hmmm, hmmm...!

Back to Fsolve

It's the NKIS that isn't satisfied if you input $\{\pi_t, x_t, i_t\}$ or $\{x_t, i_t\}$.

If you input only $\{i_t\}$, it's the Taylor rule resids that explode at the end, just as $\{x_t, \pi_t\}$ explode.

↳ I mean, in this case A9 & A10 have to hold, b/c they're used to generate $\{x_t, \pi_t\}$. And they do. But they lead to $\{x_t, \pi_t\}$ exploding, which is why the Taylor-rule doesn't hold.

$$\underbrace{\{i_t^{seq}\}}_{\text{given } \neq \infty} = \psi_\pi \underbrace{\pi_t}_{\to\infty} + \psi_x \underbrace{x_t}_{\to\infty}$$

⇒ I'm returning to thinking that it's the expectations...

↳ that's why it's NKIS that doesn't hold! B/c it's a function of f_b, which is the expectation that explodes a lot more!

But now I'm in the world where $b = gx \cdot hx = RE$, and $a = \begin{bmatrix}\bar{\pi} \\ 0 \\ 0\end{bmatrix}$ so $f_a(2) = f_a(3) = f_\beta(2) = f_\beta(3) = 0$

Therefore the $-\sigma\beta f_b(3)$ shouldn't cause headaches. But if that's so, why are $E(\cdot)$ exploding when we input the $\{i_t^{TR}\}$?

So I'm really back to the idea that the model somehow doesn't think that the TR is in place when it's not actively used, so it's not E-stable.

→ But this is kinda at odds w/ f_a, f_b behaving nicely the moment you input x_t too.

Is it? Maybe not.

Input only i_t:

$$\text{expl} \quad x_t = -\sigma i_t + E(\cdot) \quad \text{expl.}$$

$$\text{expl.} \quad \pi_t = \kappa x_t + E(\cdot)$$

Input x_t, i_t:

$$\text{res} \quad x_t = -\sigma i_t + E(\cdot)$$

$$\pi_t = \kappa x_t + E(\cdot)$$

$$i_t = \psi_\pi \pi_t + \psi_x x_t$$

here expectations seem disciplined somehow. (they converge slowly but yeah)

Indeed, as soon as you add x_t, $\bar{\pi}$ doesn't explode anymore. What happens to k_t?

$\rightarrow k_t^{-1} \rightarrow 0.14475$ (not zero) when input $\{i_t\}$.

Hmm. res(NKIS) seems to mirror $\bar{\pi}$ in shape

Yes. Quite closely, except res(NKIS) $\simeq 100 \cdot \bar{\pi}$

Erm $\rightarrow$ $f_b(1) \simeq 100 \cdot \bar{\pi} = \frac{1}{1-\beta}\bar{\pi}$ (!)

Of course! I've set $hx = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ ($\rho_i = 0 \quad i = r^n, \bar{i}, u$)

So $fa_t = \frac{1}{1-\alpha\beta} a_{t-1} = \frac{1}{1-\alpha\beta}\begin{bmatrix} \bar{\pi}_{t-1} \\ 0 \\ 0 \end{bmatrix} = 9.1743 \begin{bmatrix} \bar{\pi}_{t-1} \\ 0 \\ 0 \end{bmatrix}$

$fb = \frac{1}{1-\beta} a_{t-1}$

$\Rightarrow$ So $x_t = -\sigma i_t + \sigma f_b(1) + 0$ (again b/c $hx = 0$)

and since $\sigma = 1$,

$$\boxed{x_t = -i_t + 100 \cdot \bar{\pi}_{t-1}}$$

$\Rightarrow$ which is why $\underline{\text{res(NKIS)} = x_t + i_t = 100\bar{\pi}_{t-1}}$

What about π_t?

$$\pi_t = \kappa x_t + \overbrace{(1-\alpha)\beta}^{0.099} fa(1) = \kappa x_t + 0.9083\bar{\pi}_{t-1}$$

Supp we input $\{x_t, i_t\}$:

$$i_t = \{i_t^{TR}\} = \psi_\pi \pi_t + \psi_x x_t \to res = 0$$

$$x_t = -i_t + 100 \cdot \bar{\pi}_{t-1} \to res_{A9} = 100 \bar{\pi}_{t-1}$$

$$\pi_t = \kappa x_t + 0.9083 \bar{\pi}_{t-1} \to res_{A10} = 0.$$

Supp we input $\{\pi_t, x_t, i_t\}$

$$i_t = \{i_t^{TR}\} = \psi_\pi \pi_t + \psi_x x_t \to res = 0$$

$$x_t = -i_t + 100 \cdot \bar{\pi}_{t-1} \to res_{A9} = 100 \bar{\pi}_{t-1}$$

$$\pi_t = \kappa x_t + 0.9083 \bar{\pi}_{t-1} \to res_{A10} \text{ should be}$$

$$0.9083 \bar{\pi}_{t-1}$$

But it's not.

Moreover, $res_{A9} \neq 100 \bar{\pi}_{t-1}$ exactly.

Inputting $\{x_t, i_t\}$ vs. $\{\pi_t, x_t, i_t\}$ doesn't seem to make a diff? Can it be b/c $\psi_x = 0$? No. Setting $\psi_x = 0.5$ doesn't change things a lot.

$\to$ Why isn't there a diff btwn inputting $\{x_t, i_t\}$ and $\{\pi_t, x_t, i_t\}$?

Maybe the reason inputting $\{\pi_t, x_t, i_t\}$ isn't diff from $\{x_t, i_t\}$, b/c we input the same $\{x_t, i_t\}$ which in the case where π_t is endog determined pins down the TR π_t, which coincides w/ the one I'm endog inputting. So this should break if I initialize at random sequences?

→ Yes it does!

1) TR doesn't hold anymore.
2) NKPC holds if you input $\{x_t, i_t\}$, but no longer does when you input $\{\pi_t, x_t, i_t\}$. Which I think is exactly as it should be.

Now I'm back to inputting the Taylor-rule sequences. All equations are fulfilled, except the TR. If I input $\{\pi_t, x_t, i_t\}$ or $\{x_t, i_t\}$, then the TR is fulfilled (by design), NKPC is either fulfilled by design or due to x_t being given, so it's always the eq. That's not

fulfilled where $E(\cdot)$ show up, or find an outlet.
The $E(\cdot)$ are clearly not the same as in the TR-simulation.

$\Rightarrow$ So somehow the same sequence isn't generating the same expectations.

In particular, k is basically constant in the exog sequence, while in the Taylor-rule sequence it's basically decreasing.

So why is this happening?

$$\bar{\pi}_t = \bar{\pi}_{t-1} + k_t^{-1} \underbrace{\left(\pi_t - \bar{\pi}_{t-1} - b_1 s_{t-1}\right)}_{fe^{eve}_{t|t-1}}$$

$$k_t^{-1} = k_{t-1}^{-1} + g\left(fe^{eve}_{t|t-1}\right)$$

when $t = 1$, $k_t = k_t^{TR} = \bar{g}^{-1}$

$t = 2$, $k_t = k_t^{TR}$

$t = 3$, $k_t > k_t^{TR}$ $\Rightarrow$ it must be that fe are different! I am suspecting that even a small difference in fe will be amplified by $g(\cdot)$.

Strange – setting a dgain doesn't change anything, except things look quantitatively diff, but qualitatively, they're the same: they blow up.

So for dgain, if I input $\{x_t, i_t\}$, $\bar{\pi} = \bar{\pi}^{TR}$
$k^{-1} = k^{-1\,TR}$, and $fe^{eve}_{t+1|t-1} = fe^{eve\,TR}_{t+1|t-1}$.

and $y = y^{TR}$!

But NKIS still isn't fulfilled!

Same for cgain!

If I only input $\{i_t\}$, then $\begin{pmatrix} \bar{\pi} \\ k^{-1} \\ fe \end{pmatrix} \neq \begin{pmatrix} \bar{\pi} \\ k^{-1} \\ fe \end{pmatrix}^{TR} \rightarrow y \neq y^{TR}$!

and TR is the one that isn't fulfilled.

You guys... this is true for the smooth endog gain too!

But if E are the same, and y are the same, then why are NKIS resids $\neq 0$?

$\rightarrow$ Now I'm thinking again that it might be the A-matrices. Let me check if the ones from Mathematica (materials13_true_baseline.nb) equal the ones in materials2 (and 3 & 4):

$$A_a = \begin{bmatrix} \frac{(1-\alpha)\beta(1+b\Psi_x)}{w} & \frac{(1+b\Psi_x)\kappa\alpha\beta}{w} & 0 \\ -\frac{(1-\alpha)\beta b\Psi_\pi}{w} & -\frac{b\Psi_\pi\kappa\alpha\beta}{w} & 0 \\ \frac{(1-\alpha)\beta\Psi_\pi}{w} & \frac{\Psi_\pi\kappa\alpha\beta}{w} & 0 \end{bmatrix}$$

should be

$$\frac{\Psi_\pi(1-\alpha)\beta(1+b\Psi_x) - \Psi_x(1-\alpha)\beta b\Psi_\pi}{w}$$

$$\text{num} = (1-\alpha)\beta\Psi_\pi\left[1+\cancel{b\Psi_x}-\cancel{\Psi_x b}\right]$$ ✓

should be

$$\frac{\Psi_\pi(1+b\Psi_x)\kappa\alpha\beta - \Psi_x b\Psi_\pi\kappa\alpha\beta}{w} = \frac{\Psi_\pi\kappa\alpha\beta\left(1+\cancel{b\Psi_x}-\cancel{b\Psi_x}\right)}{w}$$ ✓

Aa is fine! What about Ab?

$$A_0 = \begin{bmatrix} \frac{\kappa\sigma(1-\beta\psi_\pi)}{w} & \frac{\kappa(1-\beta-\beta\sigma\psi_x)}{w} & 0 \\ \frac{\sigma-\beta\sigma\psi_\pi}{w} & \frac{1-\beta-\beta\sigma\psi_x}{w} & 0 \\ \frac{\sigma(1-\beta\psi_\pi)(\psi_x+\kappa\psi_\pi)}{w} & \frac{(1-\beta-\beta\sigma\psi_x)(\psi_x+\kappa\psi_\pi)}{w} & 0 \end{bmatrix}$$

num =

$\psi_\pi \kappa\sigma(1-\beta\psi_\pi) + \psi_x\sigma(1-\beta\psi_\pi)$

$= \sigma(1-\beta\psi_\pi)(\psi_\pi\kappa + \psi_x)$ ✓

num =

$\psi_\pi\kappa(1-\beta-\beta\sigma\psi_x) + \psi_x(1-\beta-\beta\sigma\psi_x)$

$= (\psi_\pi\kappa + \psi_x)(1-\beta-\beta\sigma\psi_x)$ ✓

Ok and at this point I guess I'll just trust that A_S is correct too.

Well, what I can do is if $h_x = 0$, then $i_a = I_3 = i_b$ then, according to materials2, A_S becomes

$$A_S = \begin{bmatrix} \frac{1+\sigma\psi_x}{w} & \frac{\kappa\sigma}{w} & -\frac{\kappa\sigma}{w} \\ \frac{-\sigma\psi_\pi}{w} & \frac{\sigma}{w} & -\frac{\sigma}{w} \\ \cdot & \cdot & \cdot \end{bmatrix}$$

are these switched?

$$1 - \frac{\kappa b \psi_\pi}{w} = \frac{w - \kappa b \psi_\pi}{w} = \frac{1 + b\psi_x + \cancel{\kappa b \psi_\pi} - \cancel{\kappa b \psi_\pi}}{w}$$

1st el. of last row:

$$\psi_\pi \frac{1 + b\psi_x}{w} + \psi_x \left(-\frac{b\psi_\pi}{w} \right) = \frac{\psi_\pi + \cancel{b\psi_\pi \psi_x} - \cancel{b\psi_\pi \psi_x}}{w}$$

$$= \frac{\psi_\pi}{w}$$

2nd el. of last row

$$\psi_\pi \frac{\kappa b}{w} + \psi_x \left(\frac{b}{w} \right) + 1 = \frac{\kappa b \psi_\pi + b\psi_x + w}{w} \quad ?$$

$\leftarrow = 0$ if there's no mon. pol. shock $\rightarrow \frac{\kappa b \psi_\pi + b\psi_x}{w}$

3rd el. of last row

$$\psi_\pi \left(-\frac{\kappa b}{w} \right) + \psi_x \left(-\frac{b}{w} \right) = -\frac{(\kappa b \psi_\pi + b\psi_x)}{w}$$

↳ then

$$A_s(3,:) = \left[\frac{\psi_\pi}{w}, \quad \frac{\kappa b \psi_\pi + b\psi_x}{w}, \quad -\frac{(\kappa b \psi_\pi + b\psi_x)}{w} \right]$$

As isn't quite right.

→ would this be the reason?

All right - let's assume iid shocks ($h_x = 0$) and pencil & paper solve for A_a, A_b, A_s.

(A.9) & (A.10) becomes

$$x_t = -\tilde{b}\, i_t + [b \;\; 1-\beta \;\; -b\beta] f_b + \tilde{b}\, r_t^n$$

depth $$\pi_t = \kappa x_t + [(1-\alpha)\beta, \kappa\alpha\beta, 0] f_a + u_t$$

$=: \beta$ $$i_t = \psi_\pi \pi_t + \psi_x x_t$$

$$\begin{bmatrix} 0 & 1 & \tilde{b} \\ 1 & -\kappa & 0 \\ -\psi_\pi & -\psi_x & 1 \end{bmatrix} \begin{bmatrix} \pi_t \\ x_t \\ i_t \end{bmatrix} = \begin{bmatrix} \text{stuff}_1 f_b + \tilde{b} r_t^n \\ \text{stuff}_3 f_a + u_t \\ 0 \end{bmatrix}$$

maybe I should try replacing A w/ this in matlab!

$$\begin{bmatrix} \pi_t \\ x_t \\ i_t \end{bmatrix} = \begin{bmatrix} \frac{\kappa}{w} & \frac{1+\tilde{b}\psi_x}{w} & \kappa\tilde{b} \\ \frac{1}{w} & \frac{-\tilde{b}\psi_\pi}{w} & \frac{\tilde{b}}{w} \\ \frac{\psi_x+\kappa\psi_\pi}{w} & \frac{\psi_\pi}{w} & \frac{1}{w} \end{bmatrix} \begin{bmatrix} \text{stuff}_1 f_b + \tilde{b} r_t^n \\ \text{stuff}_3 f_a + u_t \\ 0 \end{bmatrix}$$

$$\pi_t = \frac{\kappa}{w} \text{stuff}_1 f_b + \left(\frac{1+\tilde{b}\psi_x}{w}\right) \text{stuff}_3 f_a + \frac{\kappa}{w} b r_t^n + \left(\frac{1+\tilde{b}\psi_x}{w}\right) u_t$$

$$x_t = \frac{1}{w} \text{stuff}_1 f_b + \frac{-\tilde{b}\psi_\pi}{w} \text{stuff}_3 f_a + \frac{1}{w} \tilde{b} r_t^n + \frac{-\tilde{b}\psi_\pi}{w} u_t$$

$$i_t = \frac{\psi_x+\kappa\psi_\pi}{w} \text{stuff}_1 f_b + \frac{\psi_\pi}{w} \text{stuff}_3 f_a + \frac{\psi_x+\kappa\psi_\pi}{w} b r_t^n + \frac{\psi_\pi}{w} u_t$$

Tomorrow: try to implement the TR-sequence using
← that system in Matlab instead of the A-matrices!

Also check the inverse

$$\begin{bmatrix} 0 & 1 & b \\ 1 & -k & 0 \\ -\Psi_\pi & -\Psi_x & 1 \end{bmatrix}^{-1} \overset{?}{=} \begin{bmatrix} \frac{k}{w} & \frac{1+b\Psi_x}{w} & -\frac{kb}{w} \\ \frac{1}{w} & \frac{-b\Psi_\pi}{w} & -\frac{b}{w} \\ \frac{\Psi_x+k\Psi_\pi}{w} & \frac{\Psi_\pi}{w} & \frac{1}{w} \end{bmatrix}$$

Now it's good. ✓

No! ⇗ those two els are diff. 11 April 2020

$$\begin{bmatrix} \pi_t \\ x_t \\ i_t \end{bmatrix} = \begin{bmatrix} \frac{k}{w} & \frac{1+b\Psi_x}{w} & -\frac{kb}{w} \\ \frac{1}{w} & \frac{-b\Psi_\pi}{w} & -\frac{b}{w} \\ \frac{\Psi_x+k\Psi_\pi}{w} & \frac{\Psi_\pi}{w} & \frac{1}{w} \end{bmatrix} \begin{bmatrix} shuff_1 f_b + shuff_2 \cdot s_t \\ shuff_3 f_x + shuff_4 \cdot s_t \\ 0 \quad shuff_5 \cdot s_t \end{bmatrix}$$

$$A_a = \begin{bmatrix} \frac{1+b\Psi_x}{w} \\ \frac{-b\Psi_\pi}{w} \\ \frac{\Psi_\pi}{w} \end{bmatrix} [shuff_3] \qquad A_b = \begin{bmatrix} k/w \\ 1/w \\ \frac{\Psi_x+k\Psi_\pi}{w} \end{bmatrix} [shuff\,1]$$

almost equal to old Matlab stuff.

Not equal.

minus

$$A_s = \begin{bmatrix} \frac{k}{w} shuff_2 + \frac{1+b\Psi_x}{w} shuff_4 - \frac{kb}{w} shuff_5 \\ 1/w\, shuff_2 + \frac{-b\Psi_\pi}{w} shuff_4 - \frac{b}{w} shuff_5 \\ \frac{\Psi_x+k\Psi_\pi}{w} shuff_2 + \frac{\Psi_\pi}{w} shuff_4 + \frac{1}{w} shuff_5 \end{bmatrix}$$

← Equal except dependence on i_t

Good news and a bad:

Good: w/ these "april"-A-matrices, the TR sim & the exog. sim coincide. The bad news is when the TR sim diverges → of course! B/C w/o expectations internalizing a Taylor-principle, the model is not E-stable!

⟹ But overall this means that I need a single, unified way to derive the A-matrices, and I think my best bet is to do as few steps analytically as possible.

For both implementations, (A.9.) & (A.10) describes the DGP for x_t, i_t

$$x_t = -\sigma i_t + \text{stuff}_1 f_b + \text{stuff}_2 \cdot s_t$$

$$\pi_t = \kappa x_t + \text{stuff}_3 f_a + \text{stuff}_4 \cdot s_t$$

and for the TR implementation, i_t is given by a TR

$$i_t = \psi_\pi \pi_t + \psi_x x_t \qquad \text{(w/o a shock)}$$

This is already what A3A10 uses to generate
x_t, π_t given inputs.
→ it's clearly saying that w/ this configuration of the model, even a Taylor-rule system isn't E-stable.

$$x_t = -\sigma i_t + \text{stuff}_1 f_b + \text{stuff}_2 \cdot s_t$$

$$\pi_t = \kappa x_t + \text{stuff}_3 f_a + \text{stuff}_4 \cdot s_t$$

$$i_t = \psi_\pi \pi_t + \psi_x x_t$$

$$\underbrace{\begin{bmatrix} 0 & 1 & \sigma \\ 1 & -\kappa & 0 \\ -\psi_\pi & -\psi_x & 1 \end{bmatrix}}_{=:\aleph^s} \begin{bmatrix} \pi_t \\ x_t \\ i_t \end{bmatrix} = \underbrace{\begin{bmatrix} \text{stuff}_1 f_b + \text{stuff}_2 \cdot s_t \\ \text{stuff}_3 f_a + \text{stuff}_4 \cdot s_t \\ 0 \ \left(\text{or } \text{stuff}_5 \cdot s_t = [0\ 1\ 0]\, s_t \right) \end{bmatrix}}_{=:\gimel \text{ ("gimel")}}$$

if you wanna include a mon-pol shock

$$z_t = {\aleph^s}^{-1} \cdot \gimel$$

Matlab: $\Longrightarrow$ z_t = aleph_gimel.m

Ok, btw now I've corrected ${\aleph^s}^{-1}$ on materials25.nb and now it equals ${\aleph^s}^{-1}$ from evaluating the inverse numerically. (see red corrections)

Simulation using aleph_gmod.m yields EXACTLY the same thing when TR-sim or exog-sim. But, both are not E-stable.
In fact, w/ the correction, so does simulation using matrices_A_25_true_baseline.m.

↳ Ok, so this suggests that it's not that the earlier A-matrices are wrong per se, it's just that they embody a particular info-ass.
I'm looking at materials13_true_baseline.nb
Now, w/ corrections, compare (A_a, A_b, A_s) w/ $(A_a, A_b, A_s)^{april}$

$A_a = A_a^{april}$ ✓

$A_b - A_b^{april} = \begin{bmatrix} * & 0 & * \\ * & 0 & * \\ * & 0 & * \end{bmatrix}$ → loadings of z_t on $f_b(1)$ and $f_b(3)$ are different

$A_s - A_s^{april} = \begin{bmatrix} 0 & * & 0 \\ 0 & * & 0 \\ 0 & * & 0 \end{bmatrix}$ → loadings of z_t on $\hat{i}_t$ are diff.

Let's deal w/ A_s first, as I think this just comes from whether I include a mon. pol shock:

$A_s - A_s^{april} = \begin{bmatrix} 0 & * & 0 \\ 0 & * & 0 \\ 0 & * & 0 \end{bmatrix}$ → loadings of z_t on i_t are diff.

→ yes, w/ the blue corrections

$A_s = A_s^{april}$

Ok so now to Ab.

$A_b - A_b^{april} = \begin{bmatrix} * & 0 & * \\ * & 0 & * \\ * & 0 & * \end{bmatrix}$ → loadings of z_t on $f_b(1)$ and $f_b(3)$ are different

↳ this should change if $\psi_x \neq 0$. → Exactly! It does!

When $\psi_x = 0.5$, $A_b - A_b^{april} = \begin{bmatrix} * & * & * \\ * & * & * \\ * & * & * \end{bmatrix}$

Ok, so now I know that the reason there's a diff between $A_b - A_b^{april}$ is b/c of how $E(\cdot)$ are treated.

$stuff_1 = [1 \quad 0.01 \quad -0.90]$ (w/ current params, $\psi_x = 0.5$)

$$A_b^{\text{april}} = \begin{bmatrix} k/w \\ 1/w \\ \frac{\psi_x + k\psi_\pi}{w} \end{bmatrix} [\text{stuff}_1]$$

Ok. My conjecture is that for the old A_b, stuff_1^{old} is diff:

$$A_b^{old} = \begin{bmatrix} k/w \\ 1/w \\ \frac{\psi_x + k\psi_\pi}{w} \end{bmatrix} \text{stuff}_1^{old}$$

$\text{stuff}_1 = [b \; 1-\beta, -b\beta]$. If we impose (*) from my info-assumptions work in January, i.e. we impose $f_\beta(3) = \psi_\pi f_\beta(1) + \psi_x f_\beta(2)$, stuff_1^{old} becomes $[b - b\beta\psi_\pi, \; 1-\beta - b\beta\psi_x, \; 0] = \text{stuff}_1^{old}$.

↳ and it's correct!

⇒ So now:

matrices_A_25_true_baseline.m gives you the same as aleph_gimel.m. Whether they give you the same A-matrices as

matrices_A_13_true_baseline.m

depends on whether you impose that $stuff_1 = stuff_1^{old}$.

W/ $stuff_1^{old}$, you impose condition (*), i.e. that agents know the Taylor-rule. In that case, the model is E-stable, and all equations are fulfilled. If you impose the new $stuff_1$, the model is NOT E-stable, but all equations are fulfilled.

Now the challenge is:

① it doesn't make sense to simulate a target TR simulation in which agents don't internalize the TR → need to impose (*) for TR.

② but if for the exog-seq, I do not impose (*), the equations won't be fulfilled.

③ But it doesn't really make sense to impose (*) when the CB isn't using a TR → that amounts to assuming that agents believe the CB to follow a TR, while it's really just choosing the i-sequences.

⇒ Yet I feel ③ is my only option b/c in the target TR-sim, agents should know the TR and the model needs to be E-stable, yet for the exog-seq thing to work, the eqs. <u>need</u> to match. So I can say: suppose the CB thinks agents believe in a TR-type regime, then what sequences to choose?

(3) works when initialized at TR, and when initialized at random seq., <u>except</u> if you input π_t as well. (then "stopped prematurely")

Let's just try diff. seeds for rng(0) to confirm that this is independent of initialization. → Yup!

⇒ Ok, so I know the fsolve machinery works now. But several challenges arise. Supp. I want to implement the RE-TC. Now the TR is <u>NOT</u> among the model equations. So I need to simulate the model w/o recourse to a Taylor-rule. But this is not expectationally stable unless I impose condition (*) (stuff_1^{old}). Right now I feel it's the only choice I have. But it seems like this problem has an economic interpretation which is interesting. It highlights the <u>communication role</u> of the Taylor rule (or potentially of any int rate rule);

not only does i_t itself stabilize, but the majority of its effectiveness comes from its effect on LR expectations, so that they become/remain anchored, and the model thus becomes E-stable.
→ ideally, my choice of optimal reaction function would need to reflect this as well.

I just implemented the numerical Ramsey. It seems like it finds sols (although if $\lambda_x = 0$, it doesn't seem to.)
It finds min at $\lambda_x = 0.01$, but it seems to take longer the smaller λ_x. It also seems to find the same seq every time, but incurs higher loss when λ_x is higher.
Let's initialize somewhere else. Even if we initialize at TR, we get the same thing.
And if we input $\{x_1, i_1\}$? But that's not conceptually correct!
No you get sthg diff.

Honestly, I don't know why bother w/ implementing the TC numerically when you can simply solve the Ramsey problem numerically.
If this and the "implement-anchTC" exercise are both correct, and my understanding is also correct, then I should obtain the same sequences w/ Ramsey & the "implement-anchTC" approach.

What I don't like: π & x don't move, even if i changes $\rightarrow$ suggests to me that A9A10.m still isn't quite right. (Even if you input $i = \text{ones}(\cdot)$.
But there is nothing in A9A10.m that "could" be wrong - I mean, there's barely anything in it!
$\Rightarrow$ things do move! Inputting ones$(\cdot)$ vs. some sequence and computing $y^{ones} - y^{seq} \neq 0$.
But the point is that π doesn't move a lot b/c

$$\pi_t = \underset{\uparrow_{\text{small}}}{K} x_t = i_t + E(\cdot) \longrightarrow \text{whether you set}$$

higher or smaller i_t is only a function of how much there's a concern for i-stabilization (λ_i), since it doesn't move π_t by a lot, and moving x_t is the same as moving π_t, just by more.

→ but it seems then that the TR is optimal, or its optimality seems to stem from stabilizing $E(\cdot)$.

↳ I wonder if this obtains in vector learning.

I also worry about the optimal loss being huge.

Ok: tomorrow: try to implement RE-TC w/ the assumption of "TR-Expectations".

And think thru this: why do $E(\cdot)$ matter in a scalar learning environment? Is it that I change (how agents react to) π-$E(\cdot)$ that makes the diff?

→ I think so!

A9 & A10: 12 April 2020

$$\text{A9}: \quad x_t = -\sigma i_t + \overbrace{[\sigma, 1-\beta, -\sigma\beta]}^{\text{stuff}_1} f_b + \text{shocks}$$

$$+ \underbrace{[\sigma - \sigma\beta\psi_\pi, 1-\beta-\sigma\beta\psi_x, 0]}_{\text{stuff}_1^{\text{old}}} f_b$$

$$\text{A10}: \quad \pi_t = \kappa x_t + [(1-\alpha)\beta, \kappa\alpha\beta, 0] f_a + \text{shocks}$$

$\uparrow_{E(i)}$ don't play a role here, so i's effect on $E(\cdot)$ goes all thru the NKIS.

Suppose $\psi_x < 0$. Then the difference between stuff_1 & $\text{stuff}_1^{\text{old}}$ is that <u>x_t depends on $E(\pi)$ w/ σ instead of w/ $\sigma - \sigma\beta\psi_\pi$</u>, and on $E(i)$ w/ $-\sigma\beta$ instead of 0.

↳ I think this is the part that matters in scalar learning, b/c $E(i) = b \cdot s_t$ anyway. (so "fixed")

So $x_t = \sigma f_\beta(1)$ vs. $x_t = \sigma(1-\beta\psi_\pi) f_\beta(1)$

The creepy thing is, this is the same thing that causes the oscillations b/c $|\sigma| > |\sigma(1-\beta\psi_\pi)| > 0 > \sigma(1-\beta\psi_\pi)$

In vector learning, this wouldn't make a diff b/c $E(i)$ would converge to $\psi_\pi E(\pi) + \psi_x E(x)$.

Since π_t is just a scaled-down version of x_t, the CB can only affect π_t by affecting x_t $\frac{1}{\kappa}$ more. And if x_t barely moves b/c it loads on $E(\pi)$ by a small (negative) number, $\sigma(1-\beta\psi_\pi)$, if $E(\cdot)$ anchored, so gain is small, oscillations become very small, $fe(\pi)$ is small, $E(\cdot)$ anchored, and moving i a lot doesn't have a huge effect, can't get $E(\cdot)$ unanchored b/c π is moving so little that fe is too small. (→ need a better g-function).

↳ The point is that Taylor-rule beliefs (together w/ this form of the anchoring function) stabilize so much that the chosen i-sequence doesn't stir much water. → That's why vector learning is more interesting b/c there a given i-sequence would allow (or not) Taylor-rule beliefs to emerge.

→ check work given vector learning.

The problem is, I haven't implemented g for vector learning. Also, the entire anchoring TC is derived for scalar learning; it doesn't even make sense for vector learning. But I could get an optimal t-sequence for vector learning if I generalize g.

Does it even make sense to continue w/ implementing the scalar anch TC? → I don't think so b/c in that scalar case, if beliefs are not tied to the i somehow, things explode. But in a sense, this also has implications for the analytical TC. It should really just be seen as a characterization of what the sol. would look like in such a simple setting that otherwise doesn't make sense.

Ramsey i-sequence for vector learning and general g

$\Rightarrow$ sim_learnLH_clean_g.m (from sim_learnLH_clean.m)

this is as a first step just a g-function where I wrote w/ k^{-1} to start w/. Let me refer to it in the code as k1. (scalar)

then implement vector

fk_smooth.m

this stuff still blows up in my face. It's anonymous fct. not the right time to switch to k^{-1}!

$$k1 = k1t\_1 + g(\cdot)$$

$$k_t^{-1} = k_{t-1}^{-1} + g(\cdot)$$

where $g(\cdot)$ should be sthg like $d \cdot fe^2 + c$

$$\underset{ny \times 1}{fe}\ \underset{1 \times ny}{fe'} \rightarrow ny \times ny \quad \text{vs.} \quad \underset{1 \times ny}{fe'}\ \underset{ny \times 1}{fe} \Rightarrow 1 \times 1$$

$$\begin{bmatrix} fe_\pi \\ fe_x \\ fe_i \end{bmatrix} \begin{bmatrix} fe_\pi & fe_x & fe_i \end{bmatrix} = \begin{matrix} fe_\pi^2 & \underline{fe_\pi fe_x} & \underline{\underline{fe_\pi fe_i}} \\ \underline{fe_\pi fe_x} & fe_x^2 & \underline{\underline{\underline{fe_x fe_i}}} \\ \underline{\underline{fe_\pi fe_i}} & \underline{\underline{\underline{fe_x fe_i}}} & fe_i^2 \end{matrix}$$

$$\begin{bmatrix} fe_\pi & fe_x & fe_i \end{bmatrix} \begin{bmatrix} fe_\pi \\ fe_x \\ fe_i \end{bmatrix} = \begin{bmatrix} fe_\pi^2 + fe_x^2 + fe_i^2 \end{bmatrix}$$

that's fine for now.

In fact, even the vector version isn't cool. Not the right time

→ But the good news is: for vector learning, we don't necessarily need a "vector-criterion" for b. it's fine if agents only evaluate $f_e(\pi)$, not $f_e(z)$.

Ok, right now sim_learnLH_clean_g.m
= sim_learnLH_clean.m
and I've set it (_clean_g) to use $f_e(\pi)$ only to set k_t, but it's doing vector learning. Now try to input a sequence of i in there.

DEF

Vector vs scalar learning:

$$e_l = \begin{bmatrix}1\\1\\1\end{bmatrix} \rightarrow e_l \cdot \phi$$

Vector vs. scalar anchoring:

$$\underset{n_y \times 1}{f_e = f_e(z_t)} \quad \text{vs} \quad \underset{1 \times 1}{f_e = f_e(\pi_t)}$$

Now I need sim_learnLH_clean_g_given_seq.

→ ha! Listen:

- sim_learnLH_clean_g_given_seq.m;
- set up w/ vector learning
- initialized at TR seq
- NOT setting $shifts_1 = shifts_1^{old}$

... *drumroll* ....

... doesn't explode!!!

If you initialize it elsewhere, it explodes, c.p. What is puzzling though is that if you input anything else than the TR-i, it explodes. Even if you input $i_t = 1.6 \cdot \pi_t^{TR}$. Oh I think I see why. B/c also that is an exog i-sequence that doesn't correspond to what π_t turns out to be in the simulation. But this means that inputting an exog. sequence is never gonna work unless it coincides exactly w/ a TR that's used to generate that very sequence. The reason is that if you do the following (which I called "thingy" in command_numerical_ramsey.m"):

- generate y^{TR} using the TR
- input in sim_learnLH_clean_g_given_seq.m (w/ vector learning, scalar anchoring) anything else than $\psi_\pi \pi_t^{TR}$ ($\psi_x = 0$)

→ then in the model, agents won't learn the TR b/c in their world, other $\pi_t \neq \pi_t^{TR}$ will be realized, so the relationship $\psi_\pi \pi_t$ won't hold!

Ok so what I've learnt in all this is: inputting exog i-sequence doesn't make sense if agents aren't learning the Taylor-rule, AND, if they are learning the TR, they won't learn the rule you're following if it's not conditioned on the π_t they're observing.

→ the point is → what makes sense to do is to find the optimal reaction-function for i, not an exog. sequence.

Time-out.

1) Why not simply min CB-loss over exog $\{i_t\}$?
→ b/c exog-$\{i_t\}$ will always explode, even w/ vector learning.

2) So why not simply min CB-loss over $\psi_\pi, \psi_x, \psi_{\pi^2}, \psi_{x^2}$?
→ I think I've already done that (at least over ψ_π) and it answers the wrong question: it answers: "given adhering to a TR-rule (or a higher-order thing), what are optimal params?"

3.) The right question is: "what's the optimal reaction fct?"
→ optimal means that it implements the target criterion. But: the TC is only valid for my scalar-learning and highly simplified anchoring fct setting. That's no good. So what I can do is a mix of 2) & 3)

i) min CB loss over ψ_π, ψ_x in a vector learning world (the latter to avoid the "they don't know the TR, so things explode or they do know the TR and so nothing matters" issue)

(grid_search.m did this step in a constant-only, vector learning world with the CUSUM-cont.)

compute residual

ii) min residual over $\psi_\pi, \psi_x, \psi_{\pi^2}, \psi_{x^2}, \psi_{\pi x}$

keep going until residual is small enough.

Do this for other input candidates

$$\pi_{t-1}, \quad \hat{E}_t(\pi_{t+1})$$

For simplicity, set $\psi_x = 0$.

Some confusion is caused by the recursive nature of this process. Why can't I start w/ the most generous specification? B/c it's "∞"!

What can I do?

- I can evaluate the archTC for the TR to see how optimal the TR is
- I can min overall loss wrt. a very general TR and see how that does for the archTC.

↳ I wonder if the CBloss = res(archTC).

I think for today I'll prepare the presentation slides, maybe cont w/ Matlab in the evening.

I'm looking to see if one or two <u>13 April 2020</u>
iterations on analytical VFI can't help me guess what form the optimal *policy function* for i should be.

$$\left(-\kappa\delta i_t + \Omega_4 \bar{\pi}_{t-1} + \Omega_5 s_t\right)^2$$

$$= (\kappa\delta)^2 i_t^2 + \Omega_4^2 \bar{\pi}_{t-1}^2 + \Omega_5^2 s_t^2$$

$$-\kappa\delta\Omega_4 i_t \bar{\pi}_{t-1} - \kappa\delta\Omega_5 i_t s_t + \Omega_4\Omega_5 \bar{\pi}_{t-1} s_t$$

→ Eqs (32) - (34) in materials25 describe the VFI problem.

Suppose $s_t = \begin{bmatrix} r_t^n \\ u_t \end{bmatrix} \sim iid$ so $E_t(s_t) = 0$.

Guess: $V^0 = 0$.

Step 0.

$$V^1 = \max_{i_t} -\left\{ \Omega_6 i_t^2 + \Omega_7 \bar{\pi}_{t-1}^2 + \Omega_8 i_t \bar{\pi}_{t-1} + \Omega_9 i_t s_t + \Omega_{10} \bar{\pi}_{t-1} s_t + \Omega_{11} s_t \right\}$$

s.t.

$$\bar{\pi}_t = \bar{\pi}_{t-1} + k_t^{-1}\left(-\kappa\sigma i_t + \Omega_{12}\bar{\pi}_{t-1} + \Omega_5 s_t - b_1 s_{t-1}\right) \quad (33)$$

$$k_t^{-1} = k_{t-1}^{-1} + g(i_t, \bar{\pi}_{t-1}, s_t, s_{t-1}) \quad (34)$$

FOC i: $-2\Omega_6 i_t + \Omega_8 \bar{\pi}_{t-1} + \Omega_9 s_t = 0$

$$i_t = \frac{\Omega_8}{2\Omega_6}\bar{\pi}_{t-1} + \frac{\Omega_9}{2\Omega_6} s_t \quad =: P^1$$

(policy fct)

I'm not sure if the constraints should be involved here. I think maybe only in the next step to sub out $t+1$ states. Plug policy in V^1:

$$V^1 = -\left\{ \Omega_6 \left(\frac{\Omega_8}{2\Omega_6}\bar{\pi}_{t-1} + \frac{\Omega_9}{2\Omega_6} s_t \right)^2 + \Omega_7 \bar{\pi}_{t-1} + \Omega_8 i_t \bar{\pi}_{t-1} + \Omega_9 i_t s_t + \Omega_{10} \bar{\pi}_{t-1} s_t \right\}$$

$$
\begin{aligned}
V^1_{(x_t)} = -\Big\{ & \Omega_6 \left(\frac{\Omega_8}{2\Omega_6} \bar{\pi}_{t-1} + \frac{\Omega_9}{2\Omega_6} s_t \right)^2 + \Omega_7 \bar{\pi}_{t-1} + \Omega_8 i_t \bar{\pi}_{t-1} + \Omega_9 i_t s_t \\
& + \Omega_{10} \bar{\pi}_{t-1} s_t \Big\}
\end{aligned}
$$

$$
\begin{aligned}
= -\Big\{ & \Omega_6 \left(\frac{\Omega_8}{2\Omega_6} \right)^2 \bar{\pi}_{t-1}^2 + \Omega_6 \left(\frac{\Omega_9}{2\Omega_6} \right)^2 s_t^2 + \Omega_6 \frac{\Omega_8 \Omega_9}{(2\Omega_6)^2} \bar{\pi}_{t-1} s_t \\
& + \Omega_7 \bar{\pi}_{t-1} + \Omega_8 i_t \bar{\pi}_{t-1} + \Omega_9 i_t s_t + \Omega_{10} \bar{\pi}_{t-1} s_t \Big\}
\end{aligned}
$$

$$
\begin{aligned}
= -\Big\{ & \frac{\Omega_8^2}{4\Omega_6} \bar{\pi}_{t-1}^2 + \frac{\Omega_9^2}{4\Omega_6} s_t^2 + \Omega_7 \bar{\pi}_{t-1} + \Omega_8 i_t \bar{\pi}_{t-1} + \Omega_9 i_t s_t \\
& + \left(\frac{\Omega_8 \Omega_9}{4\Omega_6} + \Omega_{10} \right) \bar{\pi}_{t-1} s_t \Big\}
\end{aligned}
$$

Forgot to sub in policy for i_t here

$$
\begin{aligned}
= -\Big\{ & \frac{\Omega_8^2}{4\Omega_6} \bar{\pi}_{t-1}^2 + \frac{\Omega_9^2}{4\Omega_6} s_t^2 + \Omega_7 \bar{\pi}_{t-1} \\
& + \Omega_8 \bar{\pi}_{t-1} \left(\frac{\Omega_8}{2\Omega_6} \bar{\pi}_{t-1} + \frac{\Omega_9}{2\Omega_6} s_t \right) \\
& + \Omega_9 s_t \left(\frac{\Omega_8}{2\Omega_6} \bar{\pi}_{t-1} + \frac{\Omega_9}{2\Omega_6} s_t \right) \\
& + \left(\frac{\Omega_8 \Omega_9}{4\Omega_6} + \Omega_{10} \right) \bar{\pi}_{t-1} s_t \Big\}
\end{aligned}
$$

$$V^1(x_t) = -\Big\{ \frac{\Omega_8^2}{4\Omega_6}\bar{\pi}_{t-1}^2 + \frac{\Omega_9^2}{4\Omega_6}s_t^2 + \Omega_7\bar{\pi}_{t-1}$$

$$+\Omega_8\bar{\pi}_{t-1}\left(\frac{\Omega_8}{2\Omega_6}\bar{\pi}_{t-1} + \frac{\Omega_9}{2\Omega_6}s_t\right)$$

$$+\Omega_9 s_t\left(\frac{\Omega_8}{2\Omega_6}\bar{\pi}_{t-1} + \frac{\Omega_9}{2\Omega_6}s_t\right)$$

$$+\left(\frac{\Omega_8\Omega_9}{4\Omega_6} + \Omega_{10}\right)\bar{\pi}_{t-1}s_t\Big\}$$

$$= -\Big\{ \frac{\Omega_8^2}{4\Omega_6}\bar{\pi}_{t-1}^2 + \frac{\Omega_9^2}{4\Omega_6}s_t^2 + \Omega_7\bar{\pi}_{t-1}$$

$$+\left(\frac{\Omega_8^2}{2\Omega_6}\bar{\pi}_{t-1}^2 + \Omega_8\frac{\Omega_9}{2\Omega_6}\bar{\pi}_{t-1}s_t\right.$$

$$+\left(\frac{\Omega_8}{2\Omega_6}\Omega_9\bar{\pi}_{t-1}s_t + \frac{\Omega_9^2}{2\Omega_6}s_t^2\right)$$

$$+\left(\frac{\Omega_8\Omega_9}{4\Omega_6} + \Omega_{10}\right)\bar{\pi}_{t-1}s_t\Big\}$$

$$V^1(x_t) = -\left\{\frac{3\Omega_8^2}{4\Omega_6}\bar{\pi}_{t-1}^2 + \frac{3\Omega_9^2}{4\Omega_6}s_t^2 + \Omega_7\bar{\pi}_{t-1} + \left(\frac{5\Omega_8\Omega_9}{4\Omega_6} + \Omega_{10}\right)\bar{\pi}_{t-1}s_t\right\}$$

Here the iid ass comes in handy b/c $E[s_{t+1}] = 0$ $E[s_{t+1}^2] = \Sigma$

$$E_t V^1(x_{t+1}) = -\left\{\frac{3\Omega_8^2}{4\Omega_6}\bar{\pi}_t^2 + \frac{3\Omega_9^2}{4\Omega_6}\Sigma + \Omega_7\bar{\pi}_t\right\}$$

Now need to use constraints to sub out $\bar{\pi}_t$

Step 1.

$$V^2(x_t) = \max_{i_t} -\Big\{ \Omega_6 i_t^2 + \Omega_7 \bar{\pi}_{t-1}^2 + \Omega_8 i_t \bar{\pi}_{t-1} + \Omega_9 i_t s_t$$

$$+ \Omega_{10} \bar{\pi}_{t-1} s_t + \Omega_{11} s_t + \beta E_t V^1(x_{t+1}) \Big\}$$

s.t. constraints

which is

$$V^2(x_t) = \max_{i_t} -\Big\{ \Omega_6 i_t^2 + \Omega_7 \bar{\pi}_{t-1}^2 + \Omega_8 i_t \bar{\pi}_{t-1} + \Omega_9 i_t s_t$$

$$+ \Omega_{10} \bar{\pi}_{t-1} s_t + \Omega_{11} s_t$$

$$+ \beta \Big[-\Big\{ \frac{3\Omega_8^2}{4\Omega_6} \bar{\pi}_t^2 + \frac{3\Omega_9^2}{4\Omega_6} \Sigma + \Omega_7 \bar{\pi}_t \Big\} \Big] \Big\}$$

s.t. $\bar{\pi}_t = \bar{\pi}_{t-1} + k_t^{-1} (-\kappa\sigma i_t + \Omega_{12} \bar{\pi}_{t-1} + \Omega_5 s_t - b_1 s_{t-1})$

$k_t^{-1} = k_{t-1}^{-1} + g(i_t, \bar{\pi}_{t-1}, s_t, s_{t-1})$

$$V^2(x_t) = \max_{i_t} -\Big\{ \Omega_6 i_t^2 + \Omega_7 \bar{\pi}_{t-1}^2 + \Omega_8 i_t \bar{\pi}_{t-1} + \Omega_9 i_t s_t + \Omega_{10} \bar{\pi}_{t-1} s_t + \Omega_{11} s_t$$

$$- \beta \Big[\frac{3\Omega_8^2}{4\Omega_6} \Big(\bar{\pi}_{t-1} + \big(k_{t-1}^{-1} + g(\cdot)\big)\big(-\kappa\sigma i_t + \Omega_{12} \bar{\pi}_{t-1} + \Omega_5 s_t - b_1 s_{t-1}\big)\Big)^2$$

$$+ \frac{3\Omega_9^2}{4\Omega_6} \Sigma + \Omega_7 \Big(\bar{\pi}_{t-1} + \big(k_{t-1}^{-1} + g(\cdot)\big)\big(-\kappa\sigma i_t + \Omega_{12} \bar{\pi}_{t-1} + \Omega_5 s_t - b_1 s_{t-1}\big)\Big) \Big] \Big\}$$

$$V^2(x_t) = \max_{i_t} -\Big\{ \Omega_6 i_t^2 + \Omega_7 \bar{\pi}_{t-1}^2 + \Omega_8 i_t \bar{\pi}_{t-1} + \Omega_9 i_t s_t + \Omega_{10} \bar{\pi}_{t-1} s_t + \Omega_{11} s_t$$

$$- \beta \Big[\frac{3\Omega_8^2}{4\Omega_6} \Big(\bar{\pi}_{t-1} + \big(k_{t-1}^{-1} + g(\cdot)\big)\big(-k\delta i_t + \Omega_{12} \bar{\pi}_{t-1} + \Omega_5 s_t - b_1 s_{t-1}\big)\Big)^2$$

$$+ \frac{3\Omega_9^2}{4\Omega_6} \Sigma + \Omega_7 \Big(\bar{\pi}_{t-1} + \big(k_{t-1}^{-1} + g(\cdot)\big)\big(-k\delta i_t + \Omega_{12} \bar{\pi}_{t-1} + \Omega_5 s_t - b_1 s_{t-1}\big)\Big)\Big]\Big\}$$

FOC
i

$$-2\Omega_6 i_t + \Omega_8 \bar{\pi}_{t-1} + \Omega_9 s_t$$

$$- \beta \frac{3\Omega_8^2}{4\Omega_6} 2 \Big(\bar{\pi}_{t-1} + \big(k_{t-1}^{-1} + g(\cdot)\big)\big(-k\delta i_t + \Omega_{12} \bar{\pi}_{t-1} + \Omega_5 s_t - b_1 s_{t-1}\big)\Big)$$

$$\cdot \big(-k\delta k_{t-1}^{-1} - k\delta g(\cdot) - k\delta i_t \cdot g_i \big)$$

$$- \beta \Omega_7 \big(-k\delta k_{t-1}^{-1} - k\delta g(\cdot) - k\delta i_t \cdot g_i \big) = 0$$

$$\Leftrightarrow -2\Omega_6 i_t + \Omega_8 \bar{\pi}_{t-1} + \Omega_9 s_t$$

$$-2\beta \frac{3\Omega_8^2}{4\Omega_6} \big(k_{t-1}^{-1} + g(\cdot)\big)(-k\delta i_t)\big(-k\delta k_{t-1}^{-1} - k\delta g(\cdot) - k\delta i_t \cdot g_i \big)$$

$$- 2\beta \frac{3\Omega_8^2}{4\Omega_6} \Big(\bar{\pi}_{t-1} + \big(k_{t-1}^{-1} + g(\cdot)\big)\big(\Omega_{12} \bar{\pi}_{t-1} + \Omega_5 s_t - b_1 s_{t-1}\big)\Big)$$

$$\big(-k\delta k_{t-1}^{-1} - k\delta g(\cdot) - k\delta i_t \cdot g_i \big)$$

$$+ \beta \Omega_7 k\delta g_i i_t - \beta \Omega_7 \big(-k\delta k_{t-1}^{-1} - k\delta g(\cdot) \big) = 0$$

$$-2\Omega_6 i_t + \Omega_8 \bar{\pi}_{t-1} + \Omega_9 s_t$$

$$-2\beta \frac{3\Omega_8^2}{4\Omega_6}\left(k_{t-1}^{-1} + g(\cdot)\right)\left(-\kappa\delta\, i_t\right)\left(-\kappa\delta k_{t-1}^{-1} - \kappa\delta g(\cdot) - \kappa\delta i_t \cdot g_i\right)$$

$$-2\beta \frac{3\Omega_8^2}{4\Omega_6}\left(\bar{\pi}_{t-1} + \left(k_{t-1}^{-1} + g(\cdot)\right)\left(\Omega_{12}\bar{\pi}_{t-1} + \Omega_5 s_t - b_1 s_{t-1}\right)\right)$$

$$\left(-\kappa\delta k_{t-1}^{-1} - \kappa\delta g(\cdot) - \kappa\delta i_t \cdot g_i\right)$$

$$+\beta\Omega_7 \kappa\delta g_i\, i_t - \beta\Omega_7\left(-\kappa\delta k_{t-1}^{-1} - \kappa\delta g(\cdot)\right) = 0$$

$$\Leftrightarrow \underline{-2\Omega_6 i_t} + \Omega_8 \bar{\pi}_{t-1} + \Omega_9 s_t$$

$$-2\beta \frac{3\Omega_8^2}{4\Omega_6}\left(k_{t-1}^{-1} + g(\cdot)\right)(\kappa\delta)^2 g_i\, i_t^2$$

$$\underline{-2\beta \frac{3\Omega_8^2}{4\Omega_6}\left(k_{t-1}^{-1} + g(\cdot)\right)\left(-\kappa\delta k_{t-1}^{-1} - \kappa\delta g(\cdot)\right)\left(-\kappa\delta\, i_t\right)}$$

$$\underline{-2\beta \frac{3\Omega_8^2}{4\Omega_6}\left(\bar{\pi}_{t-1} + \left(k_{t-1}^{-1} + g(\cdot)\right)\left(\Omega_{12}\bar{\pi}_{t-1} + \Omega_5 s_t - b_1 s_{t-1}\right)\right)\left(-\kappa\delta g_i\right) i_t}$$

$$-2\beta \frac{3\Omega_8^2}{4\Omega_6}\left(\bar{\pi}_{t-1} + \left(k_{t-1}^{-1} + g(\cdot)\right)\left(\Omega_{12}\bar{\pi}_{t-1} + \Omega_5 s_t - b_1 s_{t-1}\right)\right)\left(-\kappa\delta k_{t-1}^{-1} - \kappa\delta g(\cdot)\right)$$

$$\underline{+\beta\Omega_7 \kappa\delta g_i\, i_t} - \beta\Omega_7\left(-\kappa\delta k_{t-1}^{-1} - \kappa\delta g(\cdot)\right) = 0$$

$$\Rightarrow 2\beta \frac{3\Omega_8^2}{4\Omega_6}\left(k_{t-1}^{-1} + g(\cdot)\right)(\kappa\delta)^2 g_i\, i_t^2$$

$$+\Big[2\Omega_6 - \beta\Omega_7\kappa\delta g_i + 2\beta \frac{3\Omega_8^2}{4\Omega_6}\left(k_{t-1}^{-1} + g(\cdot)\right)^2(\kappa\delta)^2$$

$$-2\beta \frac{3\Omega_8^2}{4\Omega_6}\left(\bar{\pi}_{t-1} + \left(k_{t-1}^{-1} + g(\cdot)\right)\left(\Omega_{12}\bar{\pi}_{t-1} + \Omega_5 s_t - b_1 s_{t-1}\right)\right)\left(\kappa\delta g_i\right)\Big] i_t$$

$$+\Big[-\Omega_8 \bar{\pi}_t + \Omega_9 s_t + \beta\Omega_7\left(-\kappa\delta k_{t-1}^{-1} - \kappa\delta g(\cdot)\right)$$

$$+2\beta \frac{3\Omega_8^2}{4\Omega_6}\left(\bar{\pi}_{t-1} + \left(k_{t-1}^{-1} + g(\cdot)\right)\left(\Omega_{12}\bar{\pi}_{t-1} + \Omega_5 s_t - b_1 s_{t-1}\right)\right)\left(-\kappa\delta k_{t-1}^{-1} - \kappa\delta g(\cdot)\right)\Big]$$

$= 0$ The policy function becomes a quadratic.

Before we go on the PC, let's interpret this policy:

$$2\beta\frac{3\Omega_8^2}{4\Omega_6}\left(k_{t-1}^{-1}+g(\cdot)\right)(k\delta)^2 g_i\, i_t^2$$
$$+\Big[2\Omega_6-\beta\Omega_7 k\delta g_i+2\beta\frac{3\Omega_8^2}{4\Omega_6}\left(k_{t-1}^{-1}+g(\cdot)\right)^2(k\delta)^2$$
$$-2\beta\frac{3\Omega_8^2}{4\Omega_6}\left(\bar\pi_{t-1}+(k_{t-1}^{-1}+g(\cdot))(\Omega_{12}\bar\pi_{t-1}+\Omega_5 s_t-b_1 s_{t-1})\right)(k\delta g_i)\Big]\, i_t$$
$$+\Big[-\Omega_8\bar\pi_t+\Omega_9 s_t+\beta\Omega_7\left(-k\delta k_{t-1}^{-1}-k\delta g(\cdot)\right)$$
$$+2\beta\frac{3\Omega_8^2}{4\Omega_6}\left(\bar\pi_{t-1}+(k_{t-1}^{-1}+g(\cdot))(\Omega_{12}\bar\pi_{t-1}+\Omega_5 s_t-b_1 s_{t-1})\right)\left(-k\delta k_{t-1}^{-1}-k\delta g(\cdot)\right)\Big]$$
$$=0$$

$$x_{1,2}=\frac{-b\pm\sqrt{b^2-4ac}}{2a}$$ nightmare.

But i_t is a function of

$$i_t=\rho\left(k_{t-1}^{-1},\, g(\cdot),\, g_i,\, \bar\pi_{t-1},\, s_t,\, s_{t-1}\right)$$

Which makes me think that the optimal reaction fct should be something like this:

$$i_t=r\left(\bar\pi_t,\, k_{t-1}^{-1},\, \bar\pi_{t-1},\, s_t,\, s_{t-1}\right) \qquad \boxed{r(\cdot)}$$

b/c of FE.

interesting that both show up

↳ So one thing one could do next to value fct iter is to consider a rule of the form

$$i_t = \psi_\pi \pi_t + \psi_k k_{t-1}^{-1} + \psi_{\bar{\pi}} \bar{\pi}_{t-1} + \psi_s s_t + \psi_{sl} s_{t-1} \quad (r[\cdot])$$

and maximize (− CB loss) over $(\psi_\pi, \psi_k, \psi_{\bar{\pi}}, \psi_s, \psi_{sl})$ as well as higher orders or interactions.

The previous FOC (policy fct) suggests the following interactions: $k_{t-1}^{-1}\pi_t$, $k_{t-1}^{-1}\bar{\pi}_{t-1}$, $k_{t-1}^{-1} s_t$, $k_{t-1}^{-1} s_{t-1}$

$\pi_t \bar{\pi}_{t-1}$, $\pi_t s_t$, $\pi_t s_{t-1}$.

Ok, but let's turn to Collard's VFI notes to see how I should discretize i) the iid shocks ii) the partly nonlinear LOMs :S

For the iid shocks, let's follow Peter in ass. that they take on only 'high' & 'low' values & are iid.

Then $s_t = \begin{bmatrix} r^n_{high}, u_{high} \\ r^n_{high} \; u_{low} \\ r^n_{low} \; u_{high} \\ r^n_{low} \; u_{low} \end{bmatrix}$ Supp for simplicity these values are $\{-1, 1\}$.

Then the state-space of the exog states is

$$\left\{ \begin{matrix} [1 & 1] \\ [1 & -1] \\ [-1 & 1] \\ [-1 & -1] \end{matrix} \right.$$

$\rightarrow$ sgrid = $\begin{bmatrix} 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \end{bmatrix}$

$\rightarrow$ and a state i is sgrid(:,i) .

Let's also ass. that each of these states is equally likely, so Prob(sgrid(:,i)) = $\frac{1}{4}$. I'm not sure but I think that $E(s_t | \mathcal{I}_{t-1}) = 0$ still.

Now to the endog. states.

Collard's summary of Tauchen & Hussey seems to imply the following logic. For an AR(1) of the form

$$s_{t+1} = \rho s_t + (1-\rho)\bar{s} + \varepsilon_{t+1} \qquad \varepsilon_{t+1} \sim N(0, \sigma^2)$$

<u>Step 1</u>. The likelihood is given by the Normal pdf/cdf (of s)

Step 2. Tauchen & Hussey replace the integral in the Normal pdf/cdf by the quadrature

$$\frac{1}{\sqrt{\pi}} \sum_{j=1}^{n} \underbrace{w_j}_{\text{weights}} \Phi(\underbrace{z_j, z_i}_{\text{quadrature nodes}}, \bar{x})$$

where $\Phi(s_{t+1}, s_t, \bar{s}) = \exp\left\{-\frac{1}{2}\left[\left(\frac{s_{t+1} - \rho s_t - (1-\rho)\bar{s}}{\sigma}\right)^2 - \left(\frac{s_{t+1} - \bar{s}}{\sigma}\right)^2\right]\right\}$

so that $w_j \Phi$ is interpreted as an estimate of the transition prob $\text{Prob}(s_{t+1} = s_j \mid s_t = s_i)$.

I think you get the weights using Gauss-Hermite quadrature → and there's a function GHquadrature.m in Ryan PSG code.

↳ he seems to be doing policy fct iter w/ fsolve ... hmm ...

There's also tauchenhussey.m (Martin Floden) in "International" which does all of it for you.

But my concern is that I can't simply use AR(1)-discretization b/c even the LOM for k_t^{-1} isn't a standard AR(1). If $g(\cdot)$ were the simplest possible form, which is $g(x) = d \cdot x^2 - c$

then $k_t^{-1} = k_{t-1}^{-1} + d\left(-\kappa\sigma i_t + \Omega_{12}\bar{\pi}_{t-1} + \Omega_5 s_t - b_1 s_{t-1}\right)^2 - c$.

→ this may not even be Normal anymore.

As for $\bar{\pi}_t$, even if I don't plug k_t^{-1},

$$\bar{\pi}_t = \bar{\pi}_{t-1} + k_t^{-1}\Omega_{12}\bar{\pi}_{t-1} + k_t^{-1}(-\kappa\sigma i_t + \Omega_5 s_t - b_1 s_{t-1})$$

$$\bar{\pi}_t = \underset{\uparrow \text{ time-varying AR-coeff}}{(1 + k_t^{-1}\Omega_{12})}\bar{\pi}_{t-1} + \underbrace{k_t^{-1}(-\kappa\sigma i_t + \Omega_5 s_t - b_1 s_{t-1})}_{\text{nonlinear MA terms}}$$

→ may also not be Normal, and I haven't even plugged k_{t-1}^{-1}!

<u>Judd 12.5 (doc 113) deals w/ discretization.</u>

But I want to leave that aside for now and see if a rule of the form of $r(\cdot)$ can do better than a TR.

Solving the model for a rule of the form: let's shut these off

$$i_t = \psi_\pi \pi_t + \psi_k k_{t-1}^{-1} + \psi_{\bar{\pi}} \bar{\pi}_{t-1} + \cancel{\psi_s s_t + \psi_{sl} s_{t-1}}$$

First of all, I need to i) either impose vector learning but then change the rule ii) have scalar learning but rewrite the model → rewrite stuff$_1$.

I opt for i) b/c it's more flexible.

So let the rule look like this

$$i_t = \psi_\pi \pi_t + \psi_x x_t + \psi_k k_{t-1}^{-1} + \underbrace{\psi_{\bar{\pi}} \bar{\pi}_{t-1}}_{a(1)} + \underbrace{\psi_{\bar{x}} \bar{x}_{t-1}}_{a(2)} \qquad [r(1)]$$

Still use constant-only learning. $(b = gx \cdot hx)$.

(Can shut of $\psi_x = \psi_{\bar{x}} = 0$, and it might be sufficient actually b/c $\bar{\pi}_{t-1}$ might be a sufficient stat.)

→ Model

$$\begin{bmatrix} 0 & 1 & \sigma \\ 1 & -\kappa & 0 \\ -\psi_\pi & -\psi_x & 1 \end{bmatrix} \begin{bmatrix} \pi_t \\ x_t \\ i_t \end{bmatrix} = \begin{bmatrix} stuff_1 f_b + stuff_2 s_t \\ stuff_3 f_a + stuff_4 s_t \\ \psi_k k_{t-1}^{-1} + \psi_{\bar{\pi}} \bar{\pi}_{t-1} + \psi_{\bar{x}} \bar{x}_{t-1} \end{bmatrix}$$

only new thing

↳ $\boxed{\text{aleph\_gird2.m}}$

It seems to me that in materials25.m, where I'm doing this $(\psi_\pi, \dots \psi_{xbar}) = \text{argmin}$ (Bloss thing), what I'm finding is that adding additional ψ_s does not improve on the TR. On the contrary, the new ψ_s seem to be substitutes to ψ_π in a sense that the model just doesn't want high negative feedback, so it distributes across available ψ_s, but ψ_π can bear all the burden alone.

<u>Need to get smooth anchoring fit to work</u> <u>14 April 2020</u>

- Slide on multipliers, no commitment, App.
- Put TR on slide w/ SR costs vs long-run benefits.
- Need to explain SR costs vs long-run benefits better
- Need to think of episodes the model can explain

Peter meeting 19 April 2020

- Macro lunch
  - Need for a feedback rule: RE too?
  - Implement TC = min CB loss
  - Covid-19?

→ RE: Taylor-principle needs to hold

↳ indeterminate w/ opt mon. pol when

TR is fct of exog shocks at t & t-1.

→ One trick: Ramsey describes mon pol on the eqb

but off the eqb path I assume a policy

↳ leaning hard on RE to say that agents can distinguish between your (CB's) behavior on & off the eqb path.

→ check these references to see if they have some trick

- TR preferred is an interesting result
→ if it's robust, make that another section of the paper.

The same kind of bounded rational E(·) that gives rise to anchored E also makes it hard for the CB to articulate a policy. Even tho in theory it might be optimal to adjust to $\bar{\pi}$ & k, if you in the TR include those in the TR, actually those coeffs are close to zero.

↳ Approximate the decision rules that depend on the nonlinear states as a finite dimensional object.
e.g. guess that Value fct is a linear object
→ iterate until close (but it won't equal) 0.
Fixed point which you won't.
Tschebyschev - polynomials are the way to approx

nonlinear value fits, says Ken Judd.

Work after

· Prezi mit Kritika + M

X Bring back old figure

Need to cut by 10 min.

Slide 2. 2010 – 2020 instead of 2019

✓ → Puzzling interest rate setting 2019.
Fed behavior

X "Taylor rule is not followed" vs.
am Ende "Taylor rule ist doch besser".

Slide 3 ✓ endog gain "fairly unusual" kann man nicht einfach dahinstellen.

Slide 4 ✓ nicht betonen dass "special case" – sollte nicht wie ein failure drüberkommen

Slide 7 buttons lenken ab.
mindestens unterschieben.
X "Appendix: cons, price level"
Doppelpunkte anstelle von "–"

Slide 10 ✓ misspecification is not irrational

Slide 13 ✓ Cut!

Slide 19 ✓ 2 layers of novel intertemporal tradeoffs
✓ – emphasize the E(·) of agents the CB affects
✓ – Take out ⊕

– get in touch w/ Mduwir.

Slide 21 ✓ "already true for exog gain" so why endog?

✓ Weswegen ist es gut den endog gain hier explizit zu sehen? Weil wir deswegen das Fed besser erklären können.

✓ "2nd intertemp tradeoff" banalizes it.

Slide 23. X Betonen was ist SR
Vielleicht L^{CB} schreiben?

Slide 24. ✓ "TR disciplines expectations:"
equation

Slide 25 Cut Taylor-rule preferred
✓ "Next steps: form of reaction function
Puzzling: prelim results prefer simple TR."
Rephrase "TR better approx" and put to a different slide.

Slide 26. "Optimal policy is conditioned on /
✓ conditional on"
"Explains departures..."

Marcel:

Struktur: • Modell geht extrem lang
• Ramsey ist irgendwann fertig, &
Implications ✓ theoretical ← Übergang ist dann flüssig.
Practical 10min zu lang.
• Der Schluss leidet am meisten:
How to implement
Kapitel angeben → TOC / Solving the Ramsey problem"

✓ • Bälle grösser beim Enumeration

X • Grössere margins

✓ • Buttons

✓ • Bindestriche

Notes for me

TC (commitment RF) $\Rightarrow \pi_t = -\frac{\lambda_x}{k} x_t + \frac{\lambda_t}{k} x_{t-1}$

- get in touch w/ Mdnair

- slide 18: exog gain? } the TC doesn't say
- SR vs LR tradeoffs } these LR-SR gains
- examples of other deviating from TR episodes

✓ Not today slides.
✓ PS doesn't know of slide 9. → doesn't know they're identical
✓ Slide 10, space.
✓ Svensson target criterion (2003)
✓ Switch corollary w/ No commitment slide 15.
- Implementation: feedback rule.

Notes prezi

- Ryan: what if in RLS replace s_t for s_{t-1}?
- Vito: normative implications of fwd guidance
  outcome-based vs. instrument-based?

Ryan meeting

Biggest challenge: RLS & prominent question
→ Lucas critique
→ fair critique of E(·) formation

3 response points:
1. Are these expectations good positive theory?
→ Yes, ∃ many papers in the lit incl. CEMP that show that this learning fit beliefs citing some facts that they match
e.g. inertia
2. Good for normative analysis?
RE isn't good positive, so an alternative is

needed.

This may not be the best tool, but it's what economists do! So it may be as good as any we have in terms of being robust to policy.

It may be better than models that assume over- or underreaction directly

b/c o- or -u- reaction depends on policy and so regressions can capture those

3. Again: g isn't a result of an opti-problem but it does give agents some ability to have E-formation respond to changes in environment.

- "I tried putting s_t there and it didn't matter"
- Ramsey problem min L^{CB}

s.t. model equations
and expectation formation

- Hope to get from me
  give a description of the signs of TC
  → do I respond more or less to a cost-push
⇒ !

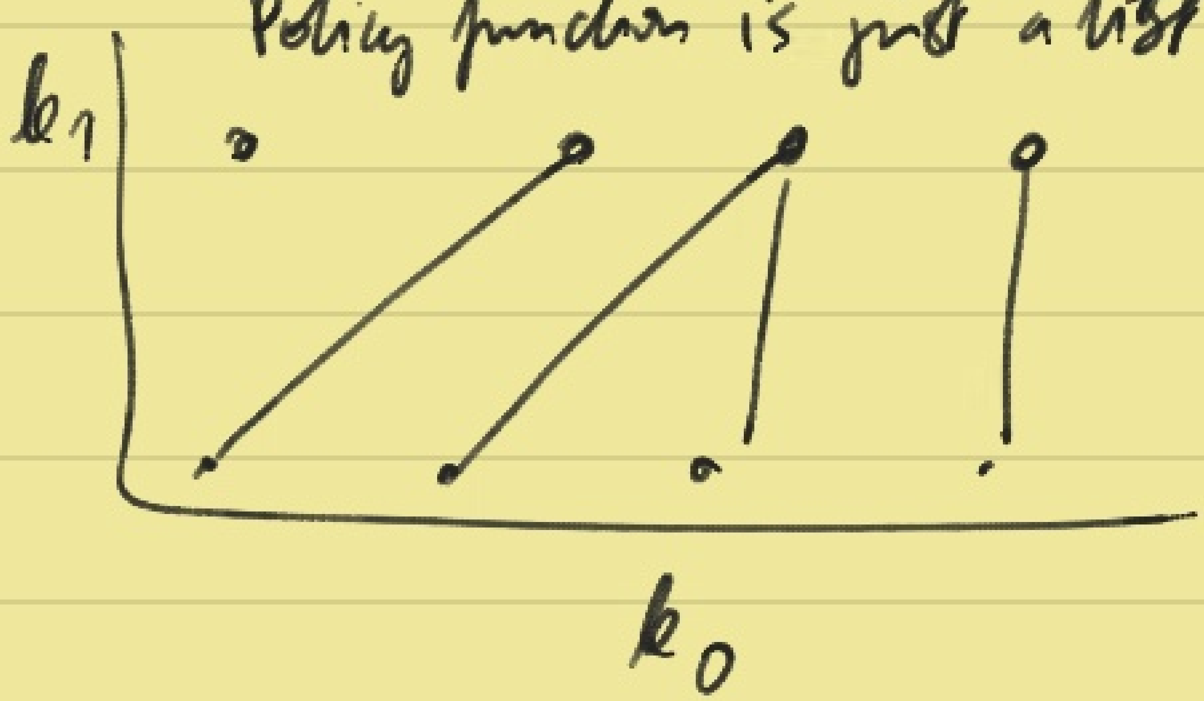
- The safe way is to say not that opti policy is time-inconsistent
  ↳ check time-inconsistent def

talk only about that!
What Ryan wants to see: 48 hours before.
objective function (history of x, i, π; exog states)
↳ complete history of beliefs that go w/ it and residuals to IS, PC & TR.
A starting point from a seq. generated by the Taylor-rule.
Eval obj fct once to show that it's working

· Good bottom line that opt policy would deviate from the TR in which direction
→ in the dir suggested by the TC!

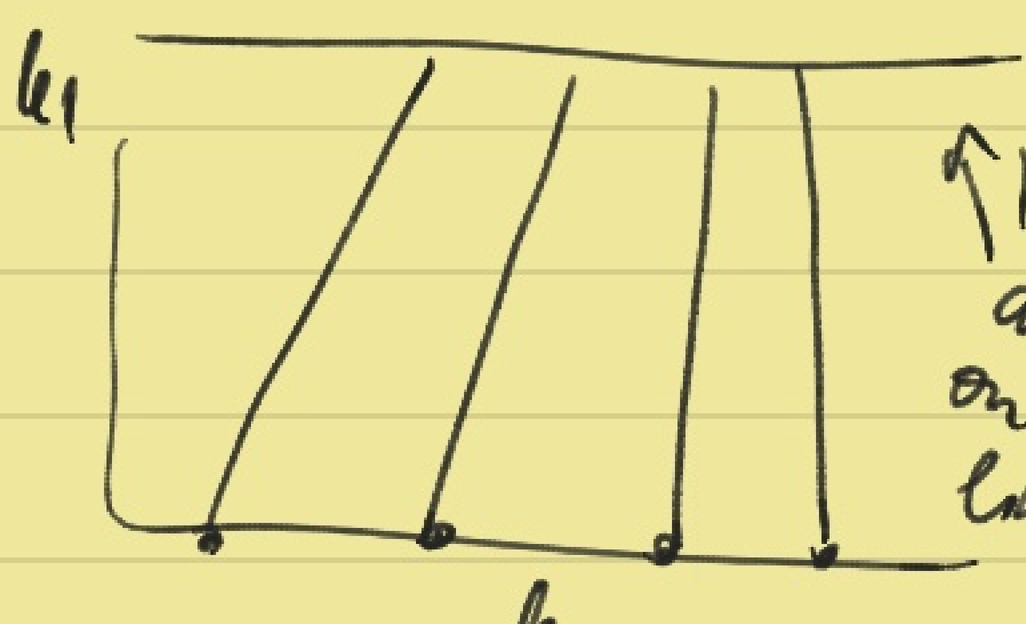
Might learn sthg by putting in a too aggressive TR and how it violates the TC.

Policy function is just a list:



Discretization perspective

vs

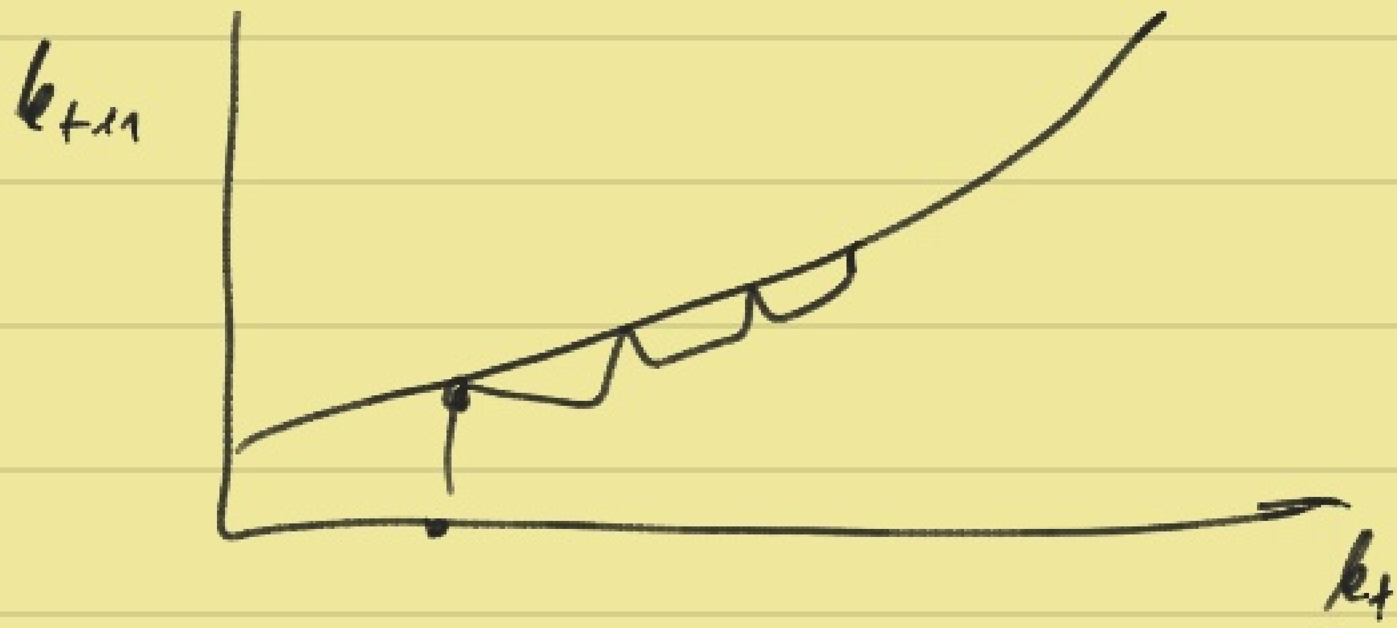


↑ I can go anywhere on the Real line

Integration perspective

⇒ policy function then is a smooth linearly fn

⇒ policy fct:



→ interpolation allows me to have a smooth k_t, k_{t+1}

→ discretization is easier, interp is more efficient

Ryan's approach:

- discretize exog states
- interpolate endog states