Occasionally Binding Constraints using Perturbation Techniques: Exogenous & Endogenous Regime Switching

Wouter J. Den Haan

Regular linear methods

- Advantages and disadvantages of linearized methods
  - Advantages: Fast methods that can deal with a (very) large state space. Models can be estimated.
  - Disadvantages: Uses local approximations, so accuracy only guaranteed around steady state. Certainty equivalence, so no precautionary savings. Regular perturbation cannot deal with inequality constraints unless they *always* bind.

Constraints considered here

- **Regime switching**, that is, exogenous *switching* between regime when constraint binds and regime when it does not bind. Quite a few models, including ZLB models, fall in this category. This part of the slides is based on work by Pontus Rendahl.
- **OccBin**: The approach of Guerrieri and Iacoviello, which allows for some endogeneity.

Overview of Pontus' approach

The underlying idea is simple enough.

- Consider a two state version.
- You will have one linear system in each state. Pontus uses a clever way to obtain these. See Pontus (2017).
- But with some probability you will, in the next period, jump to the other state (and vice versa).
- The fact that you may jump **will influence** what the linear system looks like in each state **and will capture nonlinear aspect of the model.**

But let's start with some basic tips and tricks about linear(ized) systems.

Overview of Pontus' approach

Before dealing with the occasionally binding constraints, we first

- Describe a simple method to find linear approximation around the steady state (without using Dynare). This is "DIY linearization".
- Extend this procedure to find linear approximations around different points.

With these tools in place, we will be ready to deal with occasionally binding constraints.

DIY Linearization with Time Iteration

- Without occasionally binding constraints, most models can be written in the following way,

$$\mathbb{E}_t[F(x_t, x_{t+1}, x_{t+2})] = 0$$

- Where x is a vector of endogenous and exogenous (possibly stochastic) variables
- The non-stochastic steady state, x^*, satisfies

$$F(x^*, x^*, x^*) = 0$$

- In a standard neoclassical growth model this amounts to a steady state capital stock, k^*, such that

$$1 = \beta(1 + f'(k^*) - \delta)$$

DIY Linearization with Time Iteration

- Linearisation techniques are very simple.
- Take a first order Taylor expansion of

$$\mathbb{E}_t[F(x_t, x_{t+1}, x_{t+2})] = 0$$

  around $x_t = x_{t+1} = x_{t+2} = x^*$
- and we get

$$F(x^*, x^*, x^*) + J_{x_t}(x_t - x^*) + J_{x_{t+1}}(\mathbb{E}_t[x_{t+1}] - x^*) + J_{x_{t+2}}(\mathbb{E}_t[x_{t+2}] - x^*) = 0$$

DIY Linearization with Time Iteration

- Or simply

  $$J_{x_t}(x_t - x^*) + J_{x_{t+1}}(\mathbb{E}_t[x_{t+1}] - x^*) + J_{x_{t+2}}(\mathbb{E}_t[x_{t+2}] - x^*) = 0$$

  where J_{x_t} is the Jacobian of $F(x_t, x_{t+1}, x_{t+2})$ with respect to x_t evaluated at $x_t = x_{t+1} = x_{t+2} = x^*$.
- The convenient part of this is that uncertainty vanishes, and we can focus on expected variables instead (certainty equivalence).

DIY Linearization with Time Iteration

This is written as

$$Au_{t-1} + Bu_t + Cu_{t+1} = 0$$

with $x_t - x_t^* = u_{t-1}$

- Where u_{t-1} is a vector of predetermined variables, u_t is a vector of choice variables, and u_{t+1} a vector of forward looking variables.
- Note that we have switched to "Dynare" notation

DIY Linearization with Time Iteration

Why has the expectations operator, $\mathbb{E}_t$ disappeared?

- Consider the standard RBC model with stochastic productivity, z_t, which follows an AR(1)
- The system of equations contains z_t and z_{t+1}.
- If we use the AR(1) assumption, then the system of equations contains z_{t-1}, z_t, and the innovation ε_{t+1}. The latter "disappears" because of the linearization.

DIY Linearization with Time Iteration

$$Au_{t-1} + Bu_t + Cu_{t+1} = 0$$

- The great thing about this is that systems like these are
  1. Arbitrarily general (can be of very high dimensions)
  2. Dead-easy to solve
  3. Blazing fast
  4. Uniqueness/stability and so on can be checked by the Blanchard and Kahn's (1980) conditions.
- It is always smart to solve models using linearisation techniques first to check that you get something sensible.

DIY Linearization with Time Iteration

$$Au_{t-1} + Bu_t + Cu_{t+1} = 0$$

- So how do we solve them?
- There are many ways, but Pontus' first insight is that a very easy way is *Time Iteration*. Although, this will require calculating an inverse, this matrix inversion is less problematic than the one of regular perturbation, so you do not have to worry about things like Schur decompositions.
- We are looking for a linear solution $u_t = Fu_{t-1}$
  1. Here u_{t-1} is the state, and u_t the "choice variable".
  2. F is a matrix of the same dimensionality as the Jacobians above.

DIY Linearization with Time Iteration

$$Au_{t-1} + Bu_t + Cu_{t+1} = 0$$

- The principle of time iteration: Take as given how you act tomorrow, then solve for the optimal choice today.
- If the initial guess for F is called F_0, then using this for tomorrow's behavior implies

  $$Au_{t-1} + Bu_t + CF_0 u_t = 0.$$

  from this, we get an update for F, that is an updated relationship between u_t and u_{t-1}.

DIY Linearization with Time Iteration

- More generally, for some $n \geq 0$ we find u_t as

$$Au_{t-1} + Bu_t + CF_n u_t = 0$$

and update F_n to F_{n+1} until convergence.

DIY Linearization with Time Iteration

- More generally, for some $n \geq 0$ we find u_t as

$$Au_{t-1} + Bu_t + CF_n u_t = 0$$

  and update F_n to F_{n+1} until convergence.
- Solving for u_t

$$u_t = \underbrace{(B + CF_n)^{-1}(-A)}_{F_{n+1}} u_{t-1}$$

DIY Linearization with Time Iteration

- Thus iterate on

$$F_{n+1} = (B + CF_n)^{-1}(-A),$$

until

$$||A + BF_{n+1} + CF_{n+1}^2|| \approx 0$$

- Since this goes fast, you can/should use a tight convergence criterion, like 1e(-12).

DIY Linearization with Time Iteration

- Is the solution stable?
- If the eigenvalues of F are less than one in absolute value it is.
- Are there other stable solutions too?
- Try

$$u_{t-1} = Su_t$$

and iterate on

$$S_{n+1} = (B + AS_n)^{-1}(-C),$$

- And if the eigenvalues of S are less than one in absolute value, then. there are no other stable solutions.

Linearization around an arbitrary point

- Before introducing occasionally binding constraint, we generalize the procedure to allow expansion around an arbitrary point.
- The model is again

$$\mathbb{E}_t[F(x_t, x_{t+1}, x_{t+2})] = 0$$

- Now suppose we take a first-order Taylor expansion around $\bar{x} \neq x^*$, and that

$$F(\bar{x}, \bar{x}, \bar{x}) = D$$

Linearization around an arbitrary point

- We then get

$$D + J_{x_t}(x_t - \bar{x}) + J_{x_{t+1}}(\mathbb{E}_t x_{t+1} - \bar{x}) \\ + J_{x_{t+2}}(\mathbb{E}_t x_{t+2} - \bar{x}) = 0$$

- where J_{x_t} is the Jacobian of $F(x_t, x_{t+1}, x_{t+2})$ with respect to x_t evaluated at $x_t = x_{t+1} = x_{t+2} = \bar{x}$.

Linearization around an arbitrary point

- Or simply

$$Au_{t-1} + Bu_t + Cu_{t+1} + D = 0$$

with $x_t - \bar{x} = u_{t-1}$

Linearization around an arbitrary point

- Now, our solution is not of the type

$$u_t = F u_{t-1}$$

but instead

$$u_t = E + F u_{t-1}$$

Linearization around an arbitrary point

- With time iteration we are searching for a u_t such that

$$Au_{t-1} + Bu_t + C(E_n + F_n u_t) + D = 0$$

- Thus,

$$u_t = \underbrace{(B + CF_n)^{-1}(-(D + CE_n))}_{E_{n+1}} + \underbrace{(B + CF_n)^{-1}(-A)}_{F_{n+1}} u_{t-1}$$

- Notice that F_n can be updated without information of E_n or E_{n+1}.

Linearization around an arbitrary point

- Therefore we iterate as usual

$$F_{n+1} = (B + CF_n)^{-1}(-A)$$

- Until

$$\|A + BF_{n+1} + CF_{n+1}^2\| \approx 0$$

- And once F_n has converged, we find E as the solution to

$$E = (B + CF)^{-1}(-(D + CE))$$

or simply

$$E = (B + C + CF)^{-1}(-D)$$

Regime switching systems

- Previously we looked at models that could be written in the following way,

$$\mathbb{E}_t[F(x_t, x_{t+1}, x_{t+2})] = 0$$

- Where x was a vector of endogenous and exogenous (possibly stochastic) variables
- Now we are going to look at models that are given by

$$\mathbb{E}_t[F(x_t, z_t; x_{t+1}, z_{t+1}; x_{t+2})] = 0$$

- Where z_t is a discrete stochastic variable with some transition matrix T.
- (The vector x_t can still contain other stochastic variables if you'd like, but wlog, it is assumed here that it doesn't't)

Regime switching systems

- Suppose z_t can take on values in $Z = \{z^1, z^2, \ldots, z^I\}$.
- We will not linearize with respect to z but only with respect to x.
- That is, *for each* $z^i \in Z$ we will linearize the system around $\bar{x}$, such that

$$\mathbb{E}_j F(\bar{x}, z^i; \bar{x}, z^j; \bar{x}) = D^i$$

- In fact, we could linearize around a different $\bar{x}$ for each z^i if we would like to, but let's keep things simple.

Regime switching systems

- We indicate this period's state with superscript i and next-period's state with superscript j.
- The optimal choice of x_{t+1} will depend on z^i. Thus x_{t+2} will in turn depend on z^j (the exogenous state "tomorrow").
- Next period's state is not known, but it has a discrete distribution. So think of $\mathbb{E}$ as a sum and note that we have one realization of x_{t+2} for each j.

Regime switching systems

- Linearization of the system of equations gives

$$D^i + J^i_{x_t}(x_t - \bar{x}) + J^i_{x_{t+1}}(x_{t+1} - \bar{x}) \\ + \mathbb{E}_j[J^j_{x_{t+2}}(x_{t+2}(j) - \bar{x})|i] = 0,$$

- where $J^i_{x_t}$ is the Jacobian of $\mathbb{E}_j[F(\bar{x}, z^i; \bar{x}, z^j; \bar{x})]$ with respect to the first argument, $J^i_{x_{t+1}}$ is the Jacobian with respect to the third argument, and $J^j_{x_{t+2}}$ is the Jacobian with respect to $x_{t+2}(j)$.

Regime switching systems

- We can again write this as

$$A^i u_{t-1}(i) + B^i u_t(i) + \sum_{j=1}^{I} C^j u_{t+1}(j) + D^i = 0, \quad \text{for } i = 1, \ldots, I$$

Regime switching systems

- We can again write this as

$$A^i u_{t-1}(i) + B^i u_t(i) + \sum_{j=1}^{I} C^j u_{t+1}(j) + D^i = 0, \quad \text{for } i = 1, \ldots, I$$

- Looks complicated? Let's make it more concrete.

Consumption/Savings problem with unemployment

Euler equations for employed and unemployed agent are

$$\begin{aligned}
0 = -u'(a_t(1+r) + w - a_{t+1}) \\
+\beta(1+r)[T_{e,e}u'(a_{t+1}(1+r) + w - a_{t+2}(e)) \\
+ T_{e,u}u'(a_{t+1}(1+r) - a_{t+2}(u))]
\end{aligned}$$

$$\begin{aligned}
0 = -u'(a_t(1+r) - a_{t+1}) \\
+\beta(1+r)[T_{u,e}u'(a_{t+1}(1+r) + w - a_{t+2}(e)) \\
+ T_{u,u}u'(a_{t+1}(1+r) - a_{t+2}(u))]
\end{aligned}$$

- Can take Jacobian w.r.t a_t, a_{t+1} and $a_{t+2}(i)$, $i = e, u$, and evaluate around $\bar{a}$

Consumption/Savings problem with unemployment

The linearized regime switching system is given by

$$A^e u_{t-1}(e) + B^e u_t(e) + C^{e,e} u_{t+1}(e) + C^{e,u} u_{t+1}(u) + D^e = 0$$
$$A^u u_{t-1}(u) + B^u u_t(u) + C^{u,e} u_{t+1}(e) + C^{u,u} u_{t+1}(u) + D^u = 0$$

- We would look for solutions $u_t = E^i + F^i u_{t-1}$, $i = e, u$.

Regime switching systems

- Let's go back to the general formulation:

$$A^i u_{t-1}(i) + B^i u_t(i) + \sum_{j=1}^{I} C^j u_{t+1}(j) + D^i = 0, \quad \text{for } i = 1, \dots, I$$

- We are looking for I policy functions of the type

$$u_t(i) = E^i + F^i u_{t-1}(i), \quad i = 1, 2, \dots, I$$

Regime switching systems

- Time iteration means to find u_t as the solution to

$$A^i u_{t-1}(i) + B^i u_t(i) + \sum_{j=1}^{I} C^j (E_n^j + F_n^j u_t(i)) + D^i = 0,$$

$$\text{for } i = 1, \ldots, I$$

and update the coefficients E_{n+1}^i and F_{n+1}^i accordingly.

Regime switching systems

- Therefore we iterate on the equations

$$E^i_{n+1} = (B^i + \sum_{j=1}^{l} C^j F^j_n)^{-1}(-(D + \sum_{j=1}^{l} C^j E^j_n))$$

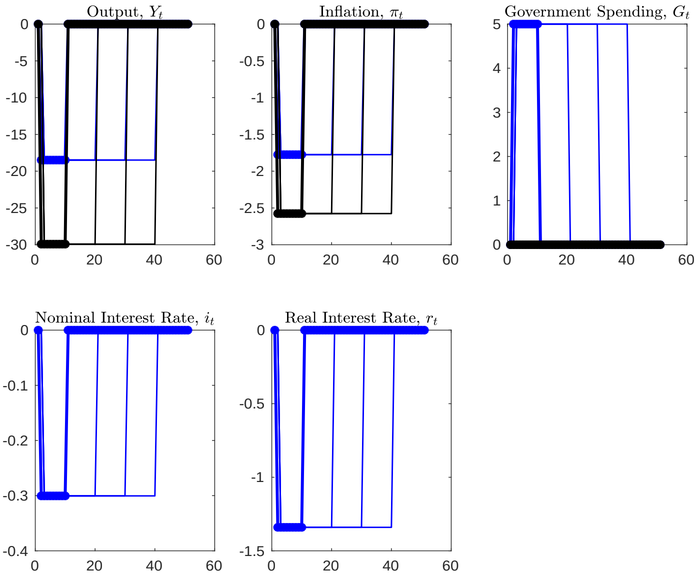
$$F^i_{n+1} = (B^i + \sum_{j=1}^{l} C^j F^j_n)^{-1}(-A)$$

for $i = 1, 2, \ldots, l$

- Until

$$\|(A + BF^i_{n+1} + \sum_{j=1}^{l} C^j F^j_{n+1} F^i_{n+1}) + \mathbb{1}[BE^i_{n+1} + \sum_{j=1}^{l} C^j (E^j_{n+1} + F^j_{n+1} E^i_{n+1})]\| \approx 0$$

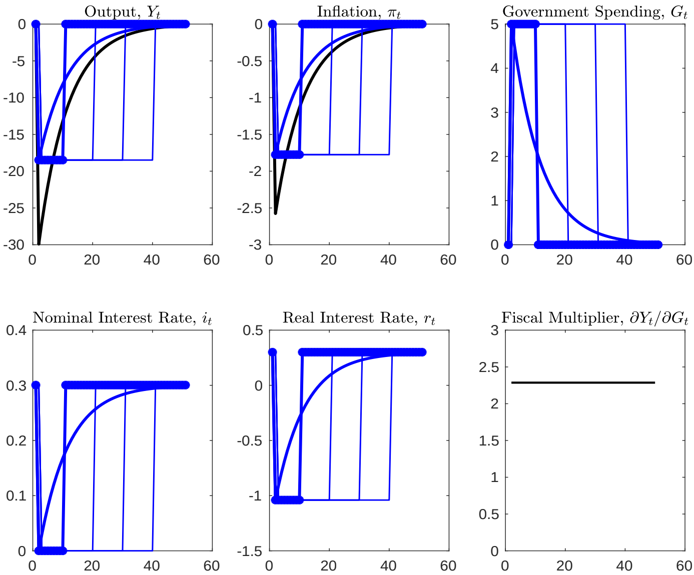
Time path conditional on state



Regime switching systems

- Looks ok, but it's not pretty.
- Plot all possible sample paths? That would be 50^2. Or more generally if T is the length of the impulse response and N is the number of elements in Z, then there are T^N possible paths.
- Popular alternative: Plot the expected paths.
  - Quite good because this is what econometricians would pick up if they had access to the data generated by the model.

Impulse Responses (blue with G increase)



Regime switching systems

- Better!
- How is this done?
- One possibility: calculate all T^N paths, weigh them by their respective probability of occurring, and sum.
- But with these sort of linear policy functions we can be smarter than that.

Regime switching systems

- Denote the expected value of u_{t+s} conditional on information available at time t, *and conditional on being in state* $z_{t+s} = z_j$ as $\mathbb{E}_t[u_{t+s}|z_{t+s} = z_j]$.
- Because of the linearities of the policy functions, this can be written as

$$\begin{aligned}\mathbb{E}_t[u_{t+s}|z_{t+s} = z_j] = \sum_{i=1}^{I} & Pr(z_{t+s-1} = z_i|z_{t+s} = z_j) \\ & \times (E^j + F^j \mathbb{E}_t[u_{t+s-1}|z_{t+s-1} = z_i])\end{aligned}$$

Regime switching systems

$$\mathbb{E}_t[u_{t+s}|z_{t+s} = z_j] = \sum_{i=1}^{I} Pr(z_{t+s-1} = z_i|z_{t+s} = z_j)$$
$$\times (E^j + F^j \mathbb{E}_t[u_{t+s-1}|z_{t+s-1} = z_i])$$

- Bayes' rule states that

$$P(A|B) = P(B|A)\frac{P(A)}{P(B)}$$

Regime switching systems

$$\mathbb{E}_t[u_{t+s}|z_{t+s} = z_j] = \sum_{i=1}^{I} Pr(z_{t+s-1} = z_i|z_{t+s} = z_j)$$
$$\times (E^j + F^j \mathbb{E}_t[u_{t+s-1}|z_{t+s-1} = z_i])$$

- Thus

$$Pr(z_{t+s-1} = z_i|z_{t+s} = z_j) = Pr(z_{t+s} = z_j|z_{t+s-1} = z_i)$$
$$\times \frac{Pr(z_{t+s-1} = z_i)}{Pr(z_{t+s} = z_j)}$$

Regime switching systems

- If z follows transition matrix T, this can be written as

$$
\begin{aligned}
Pr(z_{t+s-1} = z_i | z_{t+s} = z_j) &= Pr(z_{t+s} = z_j | z_{t+s-1} = z_i) \\
&\quad \times \frac{Pr(z_{t+s-1} = z_i)}{Pr(z_{t+s} = z_j)} \\
&= T_{ij} \frac{v_{t+s-1,i}}{v_{t+s,j}}
\end{aligned}
$$

- Where T_{ij} is the (i,j)th element of transition matrix T, and $v_{t+s,j}$ is the jth element of the vector

$$v_{t+s} = v_{t+s-1} \times T$$

for some initial v_t.

Regime switching systems

- Thus our nasty equation

$$\mathbb{E}_t[u_{t+s}|z_{t+s} = z_j] = \sum_{i=1}^{I} Pr(z_{t+s-1} = z_i|z_{t+s} = z_j) \times (E^j + F^j \mathbb{E}_t[u_{t+s-1}|z_{t+s-1} = z_i])$$

turns into something more pleasant

$$\mathbb{E}_t[u_{t+s}|z_{t+s} = z_j] = \sum_{i=1}^{I} T_{ij} \frac{v_{t+s-1,i}}{v_{t+s,j}} \times (E^j + F^j \mathbb{E}_t[u_{t+s-1}|z_{t+s-1} = z_i])$$

- And

$$\mathbb{E}_t[u_{t+s}] = \sum_{j=1}^{I} v_{t+s,j} \mathbb{E}_t[u_{t+s}|z_{t+s} = z_j]$$

Regime switching systems

- To implement this procedure, we still need to answer the following:
- What is the initial condition, u_{t-1}?
- What is v_t?

Regime switching systems

What is u_{t-1}?

- This is somewhat arbitrary, but a good start is to assume that the economy is at it's long run expected value in period t; u_{ss}.
- Given a long-run distribution v, this is given by

$$u_{ss} = \sum_{j=1}^{I} u_{j,ss} v_j$$

- Where $u_{j,ss}$ solves

$$u_{j,ss} = \sum_{i=1}^{I} T_{ij} \frac{v_i}{v_j} \times (E^j + F^j u_{i,ss}), \quad j = 1, \ldots, I$$

- We can either iterate to find $u_{j,ss}$, or to set it up as a linear system of equations.

Regime switching systems

- The nice thing about this starting value is that the expected value

$$\mathbb{E}_t[u_{t+s}] = \sum_{j=1}^{I} v_{t+s,j}\mathbb{E}_t[u_{t+s}|z_{t+s} = z_j]$$

will converge to u_{ss} as s goes to infinity.

- That is

$$\lim_{s\to\infty} \mathbb{E}_t[u_{t+s}] = u_{ss}$$

Regime switching systems

What is v_t?

- This is entirely up to you, and forms the basis of your impulse response.
- Setting $v_t = [0, 0, 1, 0, 0, \ldots]$ means that you know with certainty that you are in state 3 in period t.

Occbin - Guerrieri & Iocoviello

- In some models, there is not an exogenous variable that determines the relevant regime.
- Then we would need to determine endogenously in which regime we are.
- Occbin makes some assumptions that seem strong (but are shown not too affect accuracy too much in at least some applications). See Guerrieri and Iacoviello (2015).

Equations of the 2 regimes

- Reference regime M1 when constraint is slack: Linearized system can be expressed as

$$C_t \mathbb{E}_t[u_{t+1}] + Bu_t + Au_{t-1} + H\varepsilon_t = 0 \tag{1}$$

- Reference regime M2 when constraint binds: Linearized system can be expressed as

$$C^* \mathbb{E}_t[u_{t+1}] + B^* u_t + A^* u_{t-1} + D^* + H^* \varepsilon_t = 0 \tag{2}$$

- The constant D reflects that $u = 0$ is not a steady state of this equation.

Assumptions

- BK conditions hold in regime M1
- If $\varepsilon_t = 0$ for all future t, then the system would return to M1 within a finite number of periods

Algorithm overview: M1

- Regime M1: the solution is equal to the regular first-order perturbation solution. That is, the solution does not take into account the possibility of hitting the constraint in the future.
- Linearized system:

$$u_t = Fu_{t-1} + G\varepsilon_t$$

- You only have to check whether the constraint is indeed slack

Algorithm overview in regime M2

- Regime M2 in period t given u_{t-1}, ε_t:
  - Guess the value of T such that for $\tau \geq T$, we are in regime M1 in perpetuity
    - Here we assume that the system will be in M2 until then, but the algorithm allows for some additional switching between M1 & M2 until then.
  - Verify whether this is correct.
  - Update if necessary.

Verification procedure in regime M2

- Certainty equivalence is imposed, which means that behavior in period t does not depend on variance of shocks.
- Thus, to solve for behavior in period t we set

$$\varepsilon_{t+j} = 0 \text{ for } j \geq 1$$

- for $t + T$, we are in M1. Thus

$$u_T = Fu_{T-1} \tag{3}$$

- If $\varepsilon_t = 0$ for all future t, then the system would return to M1 within a finite number of periods

Verification procedure in regime M2

- Combining equation (3) with the M2 system of equation (2) gives

$$C^* F u_{T-1} + B^* u_{T-1} + A^* u_{T-2} + D^* = 0$$

- From this we can solve for the policy rule of X_{T-1} given X_{T-2}, just the way Pontus solved for linear policy rules using time iteration. That is,

$$u_{T-1} = F_{T-1} u_{T-1} + E_{T-1}$$

- $F_{T-1} = -(C^* F + B^*)^{-1} A^*$
- $E_{T-1} = -(C^* F + B^*)^{-1} D^*$

Verification procedure in regime M2

- Combining this with the M2 set of equations 2 gives

$$C^* F_{T-1} u_{T-2} + B^* u_{T-2} + A^* u_{T-3} + D* = 0$$

- From this we can solve for the policy rule of u_{T-2} given u_{T-3}. That is,

$$u_{T-2} = F_{T-2} u_{T-3} + E_{T-2}$$

- $F_{T-2} = -(C^* F_{T-1} + B^*)^{-1} A^*$
- $E_{T-2} = -(C^* F_{T-1} + B^*)^{-1} D^*$
- Note that E and F are time-varying coefficients

Verification procedure in regime M2

- Continue until you get to t
- With these *time-dependent* policy rules, you can generate a time path for X_{t+j} for $j \geq 1$
- Now check whether the guess for T is indeed correct

Comparison of the two methods

Pontus' regime switching model derives a linear approximation that is consistent with the underlying model, however, switching between the different regimes must be driven fully by an exogenous random variable

Comparison of the two methods

OccBin allows for endogenous switching. However, the derived policy rules are not 100% consistent with the underlying model:

- The policy rules for the regime when the constraint is not binding are *not* affected by the possibility that the constraint identical to the model in which the constraint is never binding, but in the true model they are.
- The policy rules for the regime when the constraint is binding are based on the policy rules for the unconstrained regime, which we know are not quite correct.

Comparison of the two methods

OccBin allows for endogenous switching. However, the derived policy rules are not 100% consistent with the underlying model:

- So being at the constraint with OccBin is like an “MIT” shock, that is, you get there completely unexpectedly and then do not expect to be in that position ever again.

This may or may not be important quantitatively. In quite a few examples it seems fine.

What else could you do?

1. Occasionally binding constraints are typically *not* a complication for projection methods. When policy functions have kinks, then linear splines are likely to be a better choice (although one could use polynomials when the constraint is not binding; when the constraint is binding then the policy rule follows directly from the constraint)
2. Holden (2016) develops a more general procedure that improves on Occbin by having some guaranteed convergence properties (if a solution exist) and some extensions for higher-order perturbation.

References

- Holden, T.D., 2016, Computation of solutions to Dynamic Models with Occastionally Binding Constraints, available at http://www.tholden.org/papers/
- Guerrieri, L. and M. Iacoviello, 2015, Occbin: A Toolkit for Solving Dynamic MOdels with Occasionally Binding Constriants Easily, *Journal of Monetary Economics* 70, 22-38.
- Rendahl, P., 2017, Linear Time Iteration, available at https://sites.google.com/site/pontusrendahl/Research