Estimation of dynamic models with occasionally binding constraints.

Tom D. Holden[1], School of Economics, University of Surrey

Abstract: We present an algorithm for estimating non-linear dynamic models, including those featuring occasionally binding constraints. The algorithm extends the Cubature Kalman Filter of Arasaratnam and Haykin (2009) with dynamic state space reduction, to give adequate speed in the presence of occasionally binding constraints, and to ensure that it can handle the large state spaces generated by pruned perturbation solutions to medium-scale DSGE models. We further extend the base algorithm to allow for alternative cubature procedures to improve the tracking of non-linearities. The algorithm relies on the solution method for models with occasionally binding constraints of Holden (2016b). We illustrate that the method can solve some of the identification problems that plague linearized DSGE models.

This version: 30 July 2016

The latest version of this paper may be downloaded from:

https://github.com/tholden/dynareOBC/raw/master/EstimationPaper.pdf

---

1. Introduction

Traditionally, DSGE models have been linearized prior to solution and estimation. Assuming that the driving shocks are normally distributed, the resulting solution is a linear-Gaussian state space model that may be estimated without additional approximation error via the standard Kalman filter. Thanks to linearity, and the special properties of the normal distribution, this is remarkably tractable, and has enabled the estimation of rich macroeconomic models, as popularized by Smets and Wouters (2003). However, any small departure from linearity or Gaussianity means that tracking the distribution of the state without approximation error would require keeping track of the full distribution of the state, which is an infinite dimensional object. As a result, any non-linear filter will inevitably introduce additional approximation error.

One important departure from linearity is the presence of occasionally binding constraints. Such constraints are ubiquitous in modern macroeconomic models, appearing, variously, in the zero lower bound in nominal interest rates, in the irreversibility of investment, and in borrowing and collateral constraints. If they are ignored during estimation, the resulting parameters can be severely biased, as lower bounded variables will tend to have higher mean in an accurate approximation than they do under a first order one. Furthermore, when occasionally binding constraints are ignored, there can be failures of identification, as some of the model's parameters may only alter behaviour when at the bound.

Models with occasionally binding constraints present a variety of difficulties. As established by Holden (2016a), they may possess multiple solutions in some states, and no solutions at all in others, even when a terminal condition is specified. For estimation, this means both that the likelihood may be minus infinity for some parameters, and that unless care is taken, the likelihood may be discontinuous in the parameters, invalidating standard inference. Holden (2016b) is the first simulation algorithm for models with occasionally binding constraints that is always guaranteed to select the same solution in the presence of multiplicity, restoring continuity in the parameters, and hence prior to that paper it was impossible to perform conventional statistical inference in models with occasionally binding constraints. Furthermore, Holden (2016b) establishes that computing solutions to models with occasionally binding constraints is a computationally difficult problem, in a sense which that paper makes precise. Indeed, Holden (2016b) is the first algorithm for simulating such models that is guaranteed to complete in finite time. Since almost all non-linear estimation methods are based upon repeated simulation, this in turn implies that estimating models with occasionally binding constraints is likely to be particularly

difficult, and essentially impossible if the Holden (2016b) simulation algorithm is not used.

Some progress has recently been made in estimating non-linear DSGE models using relatively standard particle filters (see e.g. Fernández-Villaverde and Rubio-Ramírez 2007; Fernández-Villaverde, Guerrón-Quintana, and Rubio-Ramírez 2015). The standard particle filter has the advantage that its estimate of the likelihood is unbiased (though noisy), and this is sufficient for correct Bayesian inference at least (Andrieu, Doucet, and Holenstein 2010). However, these methods require too many simulation-steps to be computationally tractable in the presence of occasionally binding constraints, at least on current computer hardware. Furthermore, one of the chief reasons macroeconomists were led towards Bayesian methods was due to the weak identification of many linearized DSGE models. Since using non-linear filters potentially solves the weak identification issue, we would ideally like to be able to return to maximum likelihood estimation, which is less econometrically divisive. However, due to the noise in the standard particle filter's estimate of the likelihood, numerically maximising the likelihood is almost impossible in this case. While techniques such as the smooth particle filter of Pitt (2002) may remedy this, they come at the cost of introducing substantial sampling noise into the final parameter estimates.

An alternative track of the literature has sought to exploit the properties of pruned perturbation solutions (Kim et al. 2008) to enable the use of the standard Kalman filter, despite the non-linearity, albeit with some additional approximation error. One prominent example of this approach is that of Kollmann (2013), who exploits the existence of an augmented state-space in which the pruned perturbation solution is linear, though non-Gaussian. Another is that of Meyer-Gohde (2014), who exploits the ability to get a closed form first order approximation around the mean of an underlying pruned perturbation approximation. However, both of these methods are restricted to models which are everywhere differentiable, ruling out occasionally binding constraints.

In this paper, we take an intermediary route between these two extremes, based on the cubature Kalman filter of Arasaratnam and Haykin (2009). Like the standard particle filter approach, we will rely on approximating the distribution of the state via simulation at a collection of points. Indeed, the approach we take is sometimes referred to as a type of particle filter (see e.g. Adjemian et al. 2016). Like the alternative approach though, we will maintain a Gaussian approximation to the state of the model, and we use an underlying simulation algorithm from Holden (2016b) that exploits the pruned perturbation structure.

Relative to Arasaratnam and Haykin (2009), our approach expands and improves along several lines. Firstly, we give an algorithm for the initialization of the state

distribution. Secondly, we present a technique that ensures the dimensionality of both this initial state, and subsequent states, are as low as possible, without introducing major inaccuracies. Given that pruned perturbation solutions produce very large state spaces, this is essential for medium-scale DSGE applications. Thirdly, we introduce a choice of cubature methods, which will produce better estimates at some additional computational cost. Fourthly, we modify the method to support models in which shocks enter the model non-linearly. Fifthly, we discuss techniques for the numerical maximisation of the likelihood, which needs considerable care given the potential multi-modality. Such difficulties in numerical maximisation were another key reason why the profession shifted towards Bayesian methods, so addressing them is crucial if we are to argue for using maximum likelihood estimates. Finally, we discuss the computation of standard errors, given the non-normality of the actual surprise content of observations.

Our paper is structured as follows. In the following section, we present our estimation algorithm, and discuss assorted extensions and econometric issues. In section 3, we then test the algorithm's performance, and illustrate how non-linear estimation can address the weak identification of linearized DSGE models. Section 4 concludes. All files needed for the replication of this paper's numerical results are included in the "Examples" directory of the author's DynareOBC toolkit.[2]

2.   Our algorithm for estimating non-linear models

In this section, we first describe the core algorithm, leaving unspecified how the Gaussian integrals are evaluated. We then go on to discuss alternative methods of performing the Gaussian integration. The following sub-section discusses the computation of the initial distribution of the state. This is followed by material on the practical numerical maximisation of the likelihood, and the computation of standard errors. The section concludes with a discussion of our implementation of this algorithm in the open source DynareOBC toolkit, which is freely available from http://github.org/tholden/dynareOBC.

2.1.   Core algorithm

The solution to a general non-linear DSGE model takes the following form for all $t \in \mathbb{N}^+$:

$$\begin{bmatrix} x_t \\ y_t \\ z_t \end{bmatrix} = f(x_{t-1}, \varepsilon_t),$$

where $x_t \in \mathbb{R}^{n_x}$, $f: \mathbb{R}^{n_x} \times \mathbb{R}^{n_\varepsilon} \to \mathbb{R}^{n_x + n_y + n_z}$, and $\varepsilon_t \sim \text{NIID}(0_{n_\varepsilon}, I_{n_\varepsilon \times n_\varepsilon})$. The restriction to normal shocks is without loss of generality, since if $\varepsilon \sim \text{N}(0,1)$, we can generate shocks from a distribution with cumulative distribution function H by evaluating

---

$H^{-1}(\Phi(\varepsilon))$, where Φ is the cumulative distribution function of the standard normal. Given that the shocks may enter non-linearly, such an expression may be incorporated into our f.

We allow f to return more than just the state variables, as often it is impossible (or at least inconvenient) to substitute control variables out of a model's equations. y_t will include all of the controls which have an impact on the measurement equations, and z_t includes any other controls.

We suppose that rather than observing x_t, y_t and/or z_t, for all $t \in \mathbb{N}^+$, we instead observe:

$$m_t = g\left(\begin{bmatrix} x_t \\ y_t \end{bmatrix}\right) + v_t,$$

where $m_t \in \mathbb{R}^{n_m}$ and $v_t \sim \text{NIID}(0_{n_m}, \Lambda)$, where $\Lambda \in \mathbb{R}^{n_m \times n_m}$ is diagonal. Restricting ourselves to additive, uncorrelated, Gaussian measurement error is again without loss of generality, as richer measurement error processes can be directly incorporated into y_t. Not allowing m_t to directly depend on ε_t is also without loss of generality, as components of y_t can pass through ε_t if really required. It is important though that we do always allow for some additional measurement error, since our state space dimension reduction procedure may possibly induce stochastic singularity even in models with as many shocks as observables, if some shocks make a small enough contribution.

Now, suppose that we believe that $x_{t-1}|\mathcal{F}_{t-1} \sim \text{N}(\hat{x}_{t-1|t-1}, S_{t-1|t-1}S'_{t-1|t-1})$, where $\mathcal{F}_{t-1} := \{m_0, \ldots, m_{t-1}\}$ is the period $t-1$ information set, and where $S_{t-1|t-1}$ is not necessarily square. As in the standard Kalman filter, we start with a predict step, i.e. by calculating the distribution of $x_t|\mathcal{F}_{t-1}$. Let $\zeta \in \mathbb{R}^{k_{t-1|t-1}}$ be a draw from $\text{N}(0_{k_{t-1|t-1}}, I_{k_{t-1|t-1} \times k_{t-1|t-1}})$, where $k_{t-1|t-1} := \text{cols}\, S_{t-1|t-1}$, then note that the distribution of $x_{t-1}|\mathcal{F}_{t-1}$ is equal to that of $\hat{x}_{t-1|t-1} + S_{t-1|t-1}\zeta$. Hence:

$$\mathbb{E}\left[\begin{bmatrix} x_t \\ y_t \\ z_t \end{bmatrix}\middle|\mathcal{F}_{t-1}\right] = \int_{\mathbb{R}^{n_\varepsilon}} \int_{\mathbb{R}^{k_{t-1|t-1}}} f(\hat{x}_{t-1|t-1} + S_{t-1|t-1}\zeta, \varepsilon)\phi_{k_{t-1|t-1}+n_\varepsilon}\left(\begin{bmatrix} \zeta \\ \varepsilon \end{bmatrix}\right) d\zeta\, d\varepsilon,$$

$$\text{var}\left[\begin{bmatrix} x_t \\ y_t \\ z_t \end{bmatrix}\middle|\mathcal{F}_{t-1}\right] = \int_{\mathbb{R}^{n_\varepsilon}} \int_{\mathbb{R}^{k_{t-1|t-1}}} a_{t|t-1}(\zeta, \varepsilon)a_{t|t-1}(\zeta, \varepsilon)'\phi_{k_{t-1|t-1}+n_\varepsilon}\left(\begin{bmatrix} \zeta \\ \varepsilon \end{bmatrix}\right) d\zeta\, d\varepsilon,$$

where:

$$a_{t|t-1}(\zeta, \varepsilon) := f(\hat{x}_{t-1|t-1} + S_{t-1|t-1}\zeta, \varepsilon) - \mathbb{E}\left[\begin{bmatrix} x_t \\ y_t \\ z_t \end{bmatrix}\middle|\mathcal{F}_{t-1}\right],$$

and where $\phi_k: \mathbb{R}^k \to \mathbb{R}^+$ is the probability density function of a standard k-dimensional normal variable. Thus, to derive a Gaussian approximation to the distribution of $x_t|\mathcal{F}_{t-1}$, we just need to evaluate a pair of $k_{t-1|t-1} + n_\varepsilon$-dimensional standard Gaussian integrals.

Methods for approximately performing this integration will be discussed in the following section, but we observe here that any integration rule will take the form of a weighted sum over a set of sample points. Hence, the two integrals may be evaluated simultaneously by first evaluating $f(\hat{x}_{t-1|t-1} + S_{t-1|t-1}\xi, \varepsilon)$ at the sample points, then using the integration rule to calculate the approximation to $\mathbb{E}[x_t|\mathcal{F}_{t-1}]$, which then enables us to calculate $a_{t|t-1}(\xi, \varepsilon)$ without further evaluations of f. Using these approximate integrals, we may define:

$$\hat{x}_{t|t-1} :\approx \mathbb{E}[x_t|\mathcal{F}_{t-1}],$$
$$\hat{y}_{t|t-1} :\approx \mathbb{E}[y_t|\mathcal{F}_{t-1}],$$

and

$$P_{t|t-1} :\approx \operatorname{var}\left[\begin{bmatrix} x_t \\ y_t \end{bmatrix} \middle| \mathcal{F}_{t-1}\right],$$

where ":$\approx$" is a shorthand for "is defined to be an approximation to". Finally, we approximate the distribution of $\begin{bmatrix} x_t \\ y_t \end{bmatrix}|\mathcal{F}_{t-1}$ by $\mathrm{N}\left(\begin{bmatrix} \hat{x}_{t|t-1} \\ \hat{y}_{t|t-1} \end{bmatrix}, P_{t|t-1}\right)$. While the normal approximation will not be exact for non-linear models, in practice it will often beat particle approximations unless implausibly large numbers of particles are used.

We then proceed to the update step. Recall that in the standard Kalman filter, the update step is based on the surprise in the observed measurement. Thus, we need to evaluate the distribution of $m_t|\mathcal{F}_{t-1}$. Since m_t is only a function of x_t, y_t and v_t, and all are at least approximately Gaussian conditional on $\mathcal{F}_{t-1}$, this will require a further Gaussian integral. To perform this integration, first let $U_{t|t-1}D_{t|t-1}U'_{t|t-1}$ be the Schur decomposition of $P_{t|t-1}$, where $U_{t|t-1}$ is orthogonal, and $D_{t|t-1}$ is diagonal and weakly positive, since $P_{t|t-1}$ is positive semi-definite (henceforth, p.s.d.). Let $d_{t|t-1} := \operatorname{diag} D_{t|t-1}$. Without loss of generality, suppose that only the $k_{t|t-1}$ first elements of $d_{t|t-1}$ are bigger than some threshold κ, where in the below we set $\kappa := 10^{-6}$. Now let $U_{t|t-1,\cdot 1}$ be the first $k_{t|t-1}$ columns of $U_{t|t-1}$, and $d_{t|t-1,1}$ be the first $k_{t|t-1}$ rows of $d_{t|t-1}$. Then, providing κ is small, $P_{t|t-1} \approx U_{t|t-1,\cdot 1} \operatorname{diag} d_{t|t-1,1} U'_{t|t-1,\cdot 1}$, giving a reduced rank approximation to $P_{t|t-1}$.

There are two benefits to taking reduced rank approximations. Firstly, by reducing the dimensionality of the space over which we have to integrate, it will greatly speed up the computation. Secondly, integration rules are often much better behaved in lower dimensions. This may mean they evaluate less far from the centre, avoiding distortions caused by extreme tail non-linearities, or it may mean that they have more uniform weights, avoiding e.g. failures of positive semi-definiteness caused by a negative weight.

We go on to set:
$$S_{t|t-1} := U_{t|t-1,\cdot 1} \operatorname{diag} \sqrt{d_{t|t-1,1}}$$
(where square-root operates element-wise), giving $P_{t|t-1} \approx S_{t|t-1}S'_{t|t-1}$. Hence, if $\eta \in \mathbb{R}^{k_{t|t-1}}$ is a draw from $\mathrm{N}\big(0_{k_{t|t-1}}, I_{k_{t|t-1} \times k_{t|t-1}}\big)$, then the distribution of $\begin{bmatrix} x_t \\ y_t \end{bmatrix} | \mathcal{F}_{t-1}$ is approximately equal to that of $\begin{bmatrix} \hat{x}_{t|t-1} \\ \hat{y}_{t|t-1} \end{bmatrix} + S_{t|t-1}\eta$.

Consequently:
$$\mathbb{E}[m_t|\mathcal{F}_{t-1}] \approx \int_{\mathbb{R}^{k_{t|t-1}}} g\left(\begin{bmatrix} \hat{x}_{t|t-1} \\ \hat{y}_{t|t-1} \end{bmatrix} + S_{t|t-1}\eta \right) \phi_{k_{t|t-1}}(\eta)\, d\eta,$$
$$\operatorname{var}[m_t|\mathcal{F}_{t-1}] \approx \int_{\mathbb{R}^{k_{t-1|t-1}}} b_{t|t-1}(\eta)b_{t|t-1}(\eta)' \phi_{k_{t|t-1}}(\eta)\, d\eta + \Lambda,$$
$$\operatorname{cov}[x_t, m_t|\mathcal{F}_{t-1}] \approx \int_{\mathbb{R}^{k_{t-1|t-1}}} S_{t|t-1,1}\eta b_{t|t-1}(\eta)' \phi_{k_{t|t-1}}(\eta)\, d\eta,$$
where $S_{t|t-1,1}$ is the top n_x rows of $S_{t|t-1}$ and:
$$b_{t|t-1}(\eta) := g\left(\begin{bmatrix} \hat{x}_{t|t-1} \\ \hat{y}_{t|t-1} \end{bmatrix} + S_{t|t-1}\eta \right) - \mathbb{E}[m_t|\mathcal{F}_{t-1}].$$

As before these Gaussian integrals may be simultaneously approximated by the cubature methods we will discuss in the next section. Using this approximation, we define:
$$\widehat{m}_{t|t-1} :\approx \mathbb{E}[m_t|\mathcal{F}_{t-1}],$$
$$Q_{t|t-1} :\approx \operatorname{var}[m_t|\mathcal{F}_{t-1}],$$
$$R_{t|t-1} :\approx \operatorname{cov}[x_t, m_t|\mathcal{F}_{t-1}].$$
Then $\begin{bmatrix} x_t \\ m_t \end{bmatrix} | \mathcal{F}_{t-1}$ is approximately distributed as:
$$\mathrm{N}\left(\begin{bmatrix} \hat{x}_{t|t-1} \\ \widehat{m}_{t|t-1} \end{bmatrix}, \begin{bmatrix} P_{t|t-1,11} & R_{t|t-1} \\ R'_{t|t-1} & Q_{t|t-1} \end{bmatrix} \right),$$
where $P_{t|t-1,11}$ is the upper left $n_x \times n_x$ block of $P_{t|t-1}$.

We want to know the distribution of $x_t|\mathcal{F}_t$, but by the definition of $\mathcal{F}_t$, this is just equal to the distribution of $x_t|m_t, \mathcal{F}_{t-1}$. Hence, by standard results on conditional normal distributions, $x_t|\mathcal{F}_t$ is approximately distributed as:
$$\mathrm{N}\big(\hat{x}_{t|t-1} + R_{t|t-1}Q_{t|t-1}^{-1}(m_t - \widehat{m}_{t|t-1}), P_{t|t-1,11} - R_{t|t-1}Q_{t|t-1}^{-1}R'_{t|t-1}\big).$$
Hence, we define:
$$\hat{x}_{t|t} := \hat{x}_{t|t-1} + R_{t|t-1}Q_{t|t-1}^{-1}(m_t - \widehat{m}_{t|t-1}),$$
$$P_{t|t} := P_{t|t-1,11} - R_{t|t-1}Q_{t|t-1}^{-1}R'_{t|t-1}.$$

To complete the inductive step, we have to find $S_{t|t}$ such that $S_{t|t}S'_{t|t} \approx P_{t|t}$. We proceed much as before, first constructing $U_{t|t}$ (orthogonal) and $d_{t|t}$ such that $P_{t|t} = U_{t|t} \operatorname{diag} d_{t|t} U'_{t|t}$, and then proceeding to set:
$$S_{t|t} := U_{t|t,\cdot 1} \operatorname{diag} \sqrt{d_{t|t,1}},$$

and where $d_{t|t,1}$ contains the ($k_{t|t}$) elements of $d_{t|t}$ that are greater than κ. After this, we have that the distribution of $x_t|\mathcal{F}_t$ is approximately:
$$N(\hat{x}_{t|t}, S_{t|t}S'_{t|t}),$$
which completes one time-step.

We close this section by noting that we can use these calculations and approximations to obtain the approximate likelihood, in the standard way. In particular:

$$\log p(\mathcal{F}_t) = \log p(\mathcal{F}_{t-1}) + \log p(m_t|\mathcal{F}_{t-1})$$
$$\approx \log p(\mathcal{F}_{t-1}) - \frac{1}{2}\log|Q_{t|t-1}| - \frac{1}{2}(m_t - \widehat{m}_{t|t-1})'Q_{t|t-1}^{-1}(m_t - \widehat{m}_{t|t-1})$$
$$- \frac{n_m}{2}\log 2\pi.$$

This gives an iterative formula for progressively calculating the approximate log-likelihood.

2.2. Cubature methods

Degree 3 monomial rule Arasaratnam and Haykin (2009) suggest approximating the Gaussian integrals via degree three monomial cubature. The degree three rule they advocate is based upon the following approximation:

$$\int_{\mathbb{R}^k} h(z)\phi_k(z)\,dz \approx \frac{1}{2k}\sum_{j=1}^{k}\left[h\left(\sqrt{k}\right) + h\left(-\sqrt{k}\right)\right],$$

which is exact when h is a sum of monomials of at most degree 3. Indeed, this is the degree 3 rule requiring the minimum possible number of function evaluations, making it highly computationally efficient.

In a model without occasionally binding constraints, under a first order approximation the rule will give the exact mean and variance, and under a second or third order approximation, it will give the exact mean, but only an approximate variance. In the presence of occasionally binding constraints, it will only give approximate means and variances, whatever the order of approximation.

However, in practice this rule tends to perform remarkably well. As discussed in Holden (2016b), degree 3 monomial cubature rules are particularly robust since they have positive, equal weights. All known higher degree integration rules that do not use more than polynomial in k nodes also feature negative weights on at least some nodes (Cools 2003), which means that their result is not guaranteed to lie within the convex hull of the source evaluations, and it means that the approximated covariance matrix need not be p.s.d..

A downside of the rule is that when k is large, the rule evaluates a long distance from the mean. When the true integrand is actually a sum of monomials of at most degree 3, this obviously does not matter, but in reality h often has substantially more curvature. Indeed, in the presence of occasionally binding constraints, h may feature

extreme behaviour in the tails. Hence, when h is evaluated far into the tails, we are likely to obtain biased estimates of the integral. Our dimension reduction algorithm obviously helps with this, but still in large models this may be problematic.

Genz and Keister (1996) rules By way of motivation, note that with the standard cubature Kalman filter, if we obtained a non-p.s.d. covariance matrix at one step, it would produce a catastrophic failure of the likelihood evaluation. However, given our dimension reduction algorithm, in our context this is not the case. To see this, suppose that $P_{t|t-1}$ is not p.s.d.. It will nonetheless be symmetric though, as it is being approximated by a weighted sum of symmetric matrices. Hence, by the spectral theorem for real symmetric matrices, the Schur decomposition will still give us an orthogonal $U_{t|t-1}$ and a diagonal $D_{t|t-1}$ such that $U_{t|t-1}D_{t|t-1}U'_{t|t-1} = P_{t|t-1}$. Thus, we can still select the values on the diagonal of $D_{t|t-1}$ that are greater than κ to give a p.s.d. approximation to $P_{t|t-1}$, as before. This will be a reasonable approximation to the true covariance of interest provided (plausibly enough) that the reason we ended up with a non-p.s.d. value for $P_{t|t-1}$ was that the true covariance has some very small eigenvalues.

In light of this discussion, it is natural for us to reappraise the use of higher degree cubature rules in our setting. Holden (2016b) found that the rules of Genz and Keister (1996) performed very well in a different context, with not excessively high computational cost, so these rules seem a natural thing to try here too. These rules rules allow one to choose the maximum degree of monomial that should be integrated exactly, up to a maximum order of 51. The number of points used is $O(k^d)$, where $2d + 1$ is the degree of monomial that is integrated exactly. When $d > 0$ and $k > 1$, the rule features negative weights on at least one node, but this enables it to ensure that the maximum over the absolute vectors of integration points is independent of k. This contrasts with the aforementioned rule in which the higher is k, the further into the tails of the distribution one has to evaluate the integrand, which may lead to poor performance as discussed previously. Furthermore, by using a higher degree rule, we can generally obtain a better approximation to the integrand, leading to improved estimates. The Genz and Keister (1996) rules are likely to be particularly useful for performing the integrals in the update step, as these integrals do not require performing costly simulation steps at each node, so using many nodes is quite tractable.

2.3. Selecting the initial distribution of the state

In order to evaluate the likelihood, we need to be able to calculate the unconditional distribution of m_1, which in turn requires the unconditional distribution of x_0. Now, if we have a model without occasionally binding constraints, then it is possible to get the mean and covariance of x_0 without simulation if we are using a pruned solution. Even

if we are not using a pruned solution, we can at least derive reasonable approximations to these quantities without resorting to simulation. However, in the presence of occasionally binding constraints, or with more general non-linearities, we will not be able to calculate a reasonable approximation to the stationary distribution of x_t without resorting to simulation.

One approach then would be to simulate a long run from the model, discard a burn-in period, and then take the mean and covariance of the remainder. This has two drawbacks. Firstly, due to the high degree of auto-correlation in many DSGE models, removing all sampling variation from the estimate would require a prohibitively long simulation run. Even if the same random seed was used for each run, so the objective was still continuous in the parameters even in finite samples, the result would just be that the sampling variation was transmitted to the final parameter estimates. This is essentially the same problem as is encountered by the particle filter in maximum likelihood contexts.

Secondly, it is not clear that the stationary distribution of the model is actually what is needed here. In general, thanks to the approximations intrinsic in any variant of the cubature Kalman filter, including ours, the value to which $\hat{x}_{t|t}$ and $P_{t|t}$ would converge given an infinite string of completely uninformative observations will not agree with the stationary distribution of the model. If it is agreed that parameter estimates should not change when the data set is augmented by a run of initial missing observations, then rather than trying to evaluate the stationary distribution of x_t, we should be trying to evaluate the limit of $\hat{x}_{t|t}$ and $P_{t|t}$ when no information arrives.

This is the approach we take here. In particular, we imagine that we were tasked with running the cubature Kalman filter on an infinite run of missing observations. Since this works in "pseudo-time", to distinguish "pseudo-times" from real times, we place ˜ over all "pseudo-times" in the following. We start by initializing $\hat{x}_{\tilde{0}|\tilde{0}}$ and $P_{\tilde{0}|\tilde{0}}$ with some easily computable approximation, such as that derived from the pruned perturbation approximation to the model, omitting occasionally binding constraints. We then run the cubature Kalman filter forward, skipping the update step, meaning that $\hat{x}_{\tilde{t}|\tilde{t}} = \hat{x}_{\tilde{t}|\widetilde{t-1}}$ and $P_{\tilde{t}|\tilde{t}} = P_{\tilde{t}|\widetilde{t-1},11}$ for all $\tilde{t} \in \mathbb{N}^+$. We continue until the change in $\hat{x}_{\tilde{t}|\tilde{t}}$ and $P_{\tilde{t}|\tilde{t}}$ is sufficiently small (e.g. on the order of 10^{-8}). For some models, this procedure may not converge exactly, in which case rather than making a full step from $\hat{x}_{\widetilde{t-1}|\widetilde{t-1}}$ to $\hat{x}_{\tilde{t}|\tilde{t}}$ and $P_{\widetilde{t-1}|\widetilde{t-1}}$ to $P_{\tilde{t}|\tilde{t}}$, we instead make a partial step to a weighted average of the old and new points. DynareOBC contains code for dynamically adjusting the weight which works well in practice, ensuring convergence. When $\hat{x}_{\tilde{t}|\tilde{t}}$ and $P_{\tilde{t}|\tilde{t}}$ have converged, we set $\hat{x}_{0|0}$ and $P_{0|0}$ to the found limiting values.

2.4.   Maximising the likelihood and computing standard errors

Traditionally, DSGE models have chiefly been estimated by Bayesian methods. This has apparently been for two reasons. Firstly, many parameters in linearized DSGE models are either unidentified or just weakly identified. By placing a prior over the parameter space, although the likelihood may be flat in places, the posterior density will not be, ensuring that any numerical maximisation algorithm will return the same maximum a posterior estimate. Nonetheless, the prior does not solve the underlying non-identification. Instead, when the likelihood is flat, then the prior becomes highly "informative". Of course, if the prior reflects true external information (e.g. from panel micro-data), then it is completely appropriate to incorporate this information into the final estimate. However, most of the time the prior used in macroeconomic modelling is instead a product (albeit indirect) of the same data on which the model is now being estimated. It is preferable then to attempt to fix the underlying weak identification problem, which is what non-linear estimation potentially permits. This is thanks to the fact that two parameters may have identical effects on dynamics very close to the steady-state, but quite different effects further away.

The second reason people have not traditionally pursued maximum likelihood estimation of DSGE models is because the likelihoods tend to be highly multi-modal. The hope is that with a strong enough prior, the posterior density might be unimodal, even though the likelihood is not. This hope is somewhat naïve though, since at least asymptotically the likelihood will asymptotically dominate the prior along any dimensions in which there is identification. Evidence for the practical relevance of this is provided by Herbst and Schorfheide (2014) who show that a range of popular DSGE models actually possess multimodal posteriors, though this was previously missed due to the difficulties of integrating over high dimensional spaces.

Consequently, the only way that a Bayesian approach could have a computational advantage over a classical approach would be if somehow it was at least easier to integrate over high dimensional spaces than it was to maximise in them. But this cannot be true. If one has an algorithm that can sample from a distribution in a high dimensional space, then by starting standard local maximisation procedures from these draws, one will eventually find the global maximum. Furthermore, since local maximisation requires far fewer evaluations than (say) MCMC would to explore a mode, this will be faster. Of course one can never guarantee that a maximisation procedure has found the true global maximum, but neither can one guarantee that an integration procedure has explored every mode. The fact that most MCMC implementations start by using conventional methods to search for a global mode provides further evidence that integration must be harder than maximisation.

In short then, the standard arguments for a Bayesian approach do not seem relevant when the model is estimated non-linearly. To make a maximum likelihood approach practical though, we must provide a global search algorithm with decent performance. By default, DynareOBC uses a customised version of the CMA-ES algorithm of Hansen et al. (1995; 2006). While first order (and first generation) evolutionary algorithms evolve a population of parameter vectors by combining parameters from multiple "parents" and adding independent noise to each component, the second order (and second generation) CMA-ES algorithm draws noise from a covariance matrix which mirrors the shape of the objective. This covariance matrix is dynamically updated over time in an entirely parameter-free way, and the result is an algorithm which is almost competitive with local algorithms on unimodal objectives,[3] but which also possesses good global search properties. The modified version in DynareOBC is designed to exploit parallel computing environments, further speeding up the search.

Once we have found the location of the maximum of the likelihood function, we then need to compute standard errors. Now, recall that the likelihood is coming from normal approximations to the distributions of $m_t | \mathcal{F}_{t-1}$ for $t \in \{1, \dots, T\}$. In non-linear models, these distributions may be quite a way from normality, hence, what we have here is in fact a quasi-maximum likelihood estimate (or at least, an approximate quasi-maximum likelihood one, given integral approximation error). Quasi-maximum likelihood estimates enjoy the same consistency and asymptotic normality properties of standard maximum likelihood (White 1982), but they require standard errors to be computed using a sandwich formula, as detailed in e.g. Canova (2007).

2.5.   Further details on the DynareOBC toolkit

Code implementing the estimation algorithm discussed here is contained in the author's "DynareOBC" toolkit which is a suite of MATLAB files designed to augment the abilities of Dynare (Adjemian et al. 2011). The toolkit may be freely downloaded from http://github.org/tholden/dynareOBC, and this site also contains complete documentation for its assorted options.[4] To use it for simulation, one merely has to include a "max", "min" or "abs" in the MOD file describing the DSGE model to be simulated, and then to invoke DynareOBC with the MATLAB command "dynareOBC ModFileName.MOD". Using it for estimation is almost as easy, and the examples in the "Examples\BoundedProductivityEstimation" sub-folder should make it clear to the user how to proceed.

While base Dynare now supports using the cubature Kalman filter for estimating second order approximations to models (Adjemian et al. 2016), it does not implement either the state initialization, or the state space reduction technique developed here; it

---

[3] CMA-ES requires around ten times more function evaluations than local methods on quadratic objectives.

[4] A PDF of the toolkit's documentation is available from: https://github.com/tholden/dynareOBC/raw/master/ReadMe.pdf.

does not support third order approximations; it does not calculate quasi-Maximum likelihood standard errors; and it cannot handle occasionally binding constraints. In all these regards then, the DynareOBC estimation procedure is an improvement on that already contained in base Dynare.

3. A test of the performance of our algorithm

In this section, we give some indications of the accuracy of our approach, by applying the implementation of it in the DynareOBC toolkit to two simple non-linear models, one of which contains occasionally binding constraints. We restrict ourselves to models for which the results of Holden (2016a) imply there is a unique solution in all states of the world, to avoid additional uncertainty coming from equilibrium selection, though the algorithm of Holden (2016b) does give a natural procedure for doing this. Likewise, we restrict ourselves to models for which an exact solution is available, to enable us to assess the impact of approximation error on the final parameter estimates. We start by describing the two models, and their identification properties, then we go on to present the estimation results.

3.1. A simple model without occasionally binding constraints

Suppose the representative household in an economy chooses consumption C_t and zero net supply bond holdings B_t to maximise:

$$\mathbb{E}_t \sum_{k=0}^{\infty} \beta^k \frac{C_{t+k}^{1-\gamma} - 1}{1 - \gamma},$$

subject to the restriction that:

$$A_t + R_{t-1}B_{t-1} = C_t + B_t$$

for all $t \in \mathbb{Z}$, where A_t's evolution is given by: $\log A_t = \log A_{t-1} + g_t$, where

$$g_t = (1 - \rho)\bar{g} + \rho g_{t-1} + \sigma \varepsilon_t$$

and $\varepsilon_t \sim \mathrm{N}(0,1)$. Market clearing implies $A_t = C_t$ and $B_t = 0$ for all $t \in \mathbb{Z}$, implying that the first order condition may be written as:

$$1 = \beta R_t \mathbb{E}_t \exp(-\gamma g_{t+1}),$$

and from this, it is easy to see that in the exact solution:

$$R_t = \left[\beta \exp\left(\frac{\gamma^2 \sigma^2}{2} - \gamma \mu_t \right) \right]^{-1},$$

where $\mu_t = (1 - \rho)\bar{g} + \rho g_t$.

To make our estimation task more challenging, we suppose that the econometrician only observes $\log R_t$, not g_t. The model has five parameters, but with this limited information set, only three of them can be identified. To see this, first note that since γ just scales $-\gamma g_{t+1}$, and $\log \beta$ just shifts $\log \beta - \gamma g_{t+1}$, the parameter vector $(\beta, \gamma, \bar{g}, \rho, \sigma)$ must be observationally equivalent to the parameter vector $(1, 1, \gamma\bar{g} - \log \beta, \rho, \gamma\sigma)$. In light of this, when we estimate the model, we fix β and γ at their true

values. These estimates will be based on an 1000 period artificial data-set constructed from the exact solution using the following parameters: $\beta := 0.99, \gamma := 5, \bar{g} := 0.005, \rho := 0.95$ and $\sigma := 0.007$ (after discarding 100 periods of burn-in).

3.2. A simple model with occasionally binding constraints

To produce a model with occasionally binding constraints, we modify the previous model, changing the law of motion for g_t. In particular, we suppose that for all $t \in \mathbb{Z}$:

$$g_t = \max\{0, (1-\rho)\bar{g} + \rho g_{t-1} + \sigma\varepsilon_t\}$$

where $\varepsilon_t \sim N(0,1)$. This specification may be thought of as capturing the fact that technologies cannot be un-invented. The first order condition of this model is as before, but now, in the exact solution, slightly more onerous calculation gives us that:

$$R_t = \left[\beta \left[\left(1 - \Phi\left(\frac{\mu_t}{\sigma}\right) \right) + \left(1 - \Phi\left(\frac{\gamma\sigma^2 - \mu_t}{\sigma}\right) \right) \exp\left(\frac{\gamma^2\sigma^2}{2} - \gamma\mu_t\right) \right] \right]^{-1},$$

where μ_t is as before.

This model also contains five parameters, but thanks to the additional non-linearity, four of them can be identified despite only $\log R_t$ being observed. While γ still just scales the $-\gamma g_{t+1}$ term in the first order condition, varying $\log\beta$ is no longer equivalent to shifting $\bar{g}$, due to the zero lower bound on productivity. Thus, when we estimate this model, we only need to fix γ at its true value. For comparison though, we also perform runs with β fixed too. These estimates will use an artificial data set constructed from the exact solution exactly as before, with identical parameters.

3.3. Estimation results

Results from estimating both models are contained in Table 1 below. We also include the results of estimating the unbounded model on data from the bounded model, to illustrate the biases that can occur when bounds are ignored. Furthermore, to illustrate the potential costs of different types of approximation error, we include both results where the simulations in the filter predict step are performed using the exact solution, and results where they are performed using the approximate solution algorithm from Holden (2016b), either without cubature or with degree 3 monomial cubature in the internal simulation algorithm (referred to as "fast cubature" in the below). If cubature is not incorporated into the inner simulation algorithm, then expectations are not fully rational, as agents are continually surprised by the presence of the bound. Using cubature in the inner solution algorithm fixes this, producing more accurate simulations, and, hopefully, better estimates.

When using this approximate solution algorithm without cubature, we try both with an order one approximation to the underlying model, and with an order three one. The latter illustrates our algorithm's dimension reduction method, since it turns out that in these models, order two and order three solutions agree. Finally, when we use the exact solution algorithm, we try both with degree 3 monomial cubature for the

update step integrals, and with the degree 51 Genz and Keister (1996) rules, which are essentially exact. All other integrals are performed with the degree 3 monomial rule.

Bound in model	Bound in d.g.p.	Simulation approximation	Update int. degree	$\log p(\mathcal{F}_t)$ true params.	$\log p(\mathcal{F}_t)$ estimate dparams.	β	$\bar{g}$	ρ	σ	$\sqrt{\Lambda}$
No	No	Order 1	3	1991.00	2005.96	*Fixed*	2.72E-03	9.24E-01	7.04E-03	0.00E+00
							(NaN)	*(8.09E-03)*	*(2.95E-05)*	*(5.53E-11)*
No	No	Order 3	3	1991.02	2005.96	*Fixed*	2.84E-03	9.24E-01	7.04E-03	0.00E+00
							(NaN)	*(1.07E-02)*	*(5.26E-05)*	*(1.13E-10)*
No	No	Exact	3	1991.02	2005.96	*Fixed*	2.84E-03	9.24E-01	7.04E-03	0.00E+00
							(1.57E-01)	*(2.51E+01)*	*(4.14E-02)*	*(1.76E-07)*
No	No	Exact	51	1991.02	2005.96	*Fixed*	2.84E-03	9.24E-01	7.04E-03	0.00E+00
							(2.60E-05)	*(4.89E-03)*	*(4.46E-05)*	*(1.35E-10)*
No	Yes	Order 1	3	2116.99	2193.82	*Fixed*	1.51E-02	8.98E-01	6.00E-03	0.00E+00
							(NaN)	*(1.27E-02)*	*(9.19E-05)*	*(3.22E-10)*
No	Yes	Order 3	3	2116.92	2193.82	*Fixed*	1.52E-02	8.98E-01	6.00E-03	0.00E+00
							(NaN)	*(2.31E-02)*	*(1.11E-04)*	*(5.06E-10)*
No	Yes	Exact	3	2116.92	2193.82	*Fixed*	1.52E-02	8.98E-01	6.00E-03	0.00E+00
							(1.21E-04)	*(2.57E-03)*	*(5.09E-05)*	*(1.92E-10)*
No	Yes	Exact	51	2116.92	2193.82	*Fixed*	1.52E-02	8.98E-01	6.00E-03	0.00E+00
							(6.71E-05)	*(1.70E-02)*	*(2.85E-06)*	*(9.73E-11)*
Yes	Yes	Order 1 No cubature	3	2181.32	2239.61	*Fixed*	1.02E-02	9.21E-01	6.25E-03	2.12E-11
							(NaN)	*(7.45E-02)*	*(9.36E-05)*	*(1.68E-16)*
Yes	Yes	Order 3 No cubature	3	2187.98	2242.91	*Fixed*	1.06E-02	9.20E-01	6.31E-03	4.25E-11
							(NaN)	*(5.27E-03)*	*(6.43E-05)*	*(4.53E-15)*
Yes	Yes	Order 1 Fast cubature	3	2232.11	2258.29	*Fixed*	3.13E-03	9.44E-01	7.02E-03	5.23E-09
							(NaN)	*(7.47E-03)*	*(1.65E-12)*	*(1.05E-10)*
Yes	Yes	Exact	3	2233.50	2254.06	*Fixed*	2.89E-03	9.46E-01	6.59E-03	3.80E-05
							(2.78E-05)	*(5.51E-03)*	*(6.23E-05)*	*(2.38E-07)*
Yes	Yes	Exact	51	2236.48	2255.92	*Fixed*	1.36E-03	9.49E-01	6.69E-03	3.73E-05
							(3.79E-05)	*(2.39E-02)*	*(1.60E-04)*	*(1.34E-07)*
Yes	Yes	Order 1 No cubature	3	2181.32	2253.06	9.75E-01	1.00E-06	9.44E-01	6.37E-03	1.40E-09
						(NaN)	*(NaN)*	*(2.55E-02)*	*(4.97E-04)*	*(6.27E-10)*
Yes	Yes	Order 3 No cubature	3	2187.98	2253.07	9.75E-01	1.00E-06	9.44E-01	6.37E-03	0.00E+00
						(NaN)	*(NaN)*	*(4.02E-03)*	*(5.57E-05)*	*(1.11E-10)*
Yes	Yes	Order 1 Fast cubature	3	2232.11	2262.63	9.93E-01	6.53E-03	9.33E-01	6.91E-03	1.38E-11
						(NaN)	*(NaN)*	*(8.46E-03)*	*(5.58E-05)*	*(1.58E-11)*
Yes	Yes	Exact	3	2233.50	2257.95	9.96E-01	9.75E-03	9.23E-01	6.85E-03	2.11E-11
						(1.94E-03)	*(1.37E-04)*	*(3.92E-03)*	*(6.77E-05)*	*(3.94E-16)*
Yes	Yes	Exact	51	2236.48	2259.46	9.96E-01	9.58E-03	9.24E-01	6.95E-03	0.00E+00
						(2.50E-08)	*(2.10E-10)*	*(2.03E-08)*	*(1.52E-10)*	*(1.56E-14)*

Table 1: Estimates (Standard errors in brackets. *Standard errors in italics may be unreliable due to reported numerical difficulties inverting the Hessian. NaN stands for "not a number" and is indicative of near zero Eigenvalues in both the Hessian and the Fisher information matrix, resulting in 0/0 expressions.*)

All numerical maximisation was performed with the CMA-ES algorithm, with the results polished by MATLAB's "fmincon". We start CMA-ES from the true parameters, with the measurement error standard deviation, $\sqrt{\Lambda}$, set to 0.0001, but thanks to the initial broad search undertaken by the CMA-ES algorithm, it soon moves away from this point, so identical results would be derived with a different initial

point. We constraint β and ρ to lie in $[10^{-6}, 1 - 10^{-6}]$, $\bar{g}$ to lie in $[10^{-6}, \infty)$, and σ and $\sqrt{\Lambda}$ to lie in $[0, \infty)$. Estimation times range from a few minutes for the models without bounds, to around four hours for the runs with "fast cubature" in the internal simulation algorithm, running on either a 12 or 20 core machine. We stress though that thanks to our dimension reduction techniques, and the fact that simulation speed is almost independent of model size, running times should be of a similar order of magnitude even with medium-scale models.

Turning to the results, we first note that the estimates of $\bar{g}$ are all quite poor. For example, in the model without occasionally binding constraints, $\bar{g}$ is roughly half its true value, even with the exact simulation algorithm, which turns this into a standard linear filtering problem. However, in all cases this is just driven by sampling error. Given the high persistence in g_t, $\bar{g}$ cannot be precisely estimated. To provide further intuition, note that in the model without occasionally binding constraints:

$$\mathbb{E} \log R_t = \gamma \bar{g} - \frac{\gamma^2 \sigma^2}{2} - \log \beta.$$

Hence, conditional on the other parameters, we can estimate $\bar{g}$ by evaluating:

$$\frac{1}{\gamma} \left[\frac{1}{T} \sum_{t=1}^{T} \log R_t + \frac{\gamma^2 \sigma^2}{2} + \log \beta \right],$$

which, on our generated data set without occasionally binding constraints, gives a value of 2.58×10^{-3}, broadly in line (though somewhat worse) than our estimates.

Next, note that without occasionally binding constraints, three of our methods give identical results. This is due to the fact that the third order approximation is exact in this case, and to the fact that the exact measurement equation is linear. The standard errors do not perfectly agree here though, essentially due to a blow up in numerical errors coming from the poor-conditioning of the hessian, which itself comes from the weak identification of $\bar{g}$.

We now examine the estimates generated by running the model without occasionally binding constraints on the data generated with an occasionally binding constraint. Here, we see a big upward bias in $\bar{g}$, and a big downwards bias in ρ and σ. The former is due to the fact that the mean of a zero lower bounded process is higher than the mean of the process without bound. The latter is due to the fact that hitting the bound prevents g_t from getting any lower, both compressing its range, and reducing its mean half-life, as returning from e.g. $g_t = -0.01$ takes longer than returning from $g_t = 0$. This illustrates the severe biases that may accompany an attempt to estimate a model without occasionally binding constraints when there clearly are such constraints in the data generating process.

The estimates using a model with occasionally binding constraints unsurprisingly fare much better. With "fast cubature", we get results very close to those using the exact solution, suggesting that even in models for which an exact solution is not

available, we are likely to get good results using "fast cubature". The estimates of ρ seem particularly impressive, which is most likely explained by the additional information coming from the zero lower bound, as ρ will now have a non-linear effect.

When we attempt to estimate β as well, understandably, the estimates of other parameters suffer somewhat. For example, β is biased downwards, and $\bar{g}$ is driven to its lower bound of 10^{-6} when we do not incorporate cubature in the simulation algorithm. To understand this, observe that in the absence of cubature, our simulation algorithm essentially ignores the bound when computing expectations. In particular, if d is a dummy which equals 0 under a first order approximation, and 1 under a third order approximation, then our solution's approximation in the absence of cubature, which we shall denote $\tilde{R}_t^{(d)}$ solves:

$$1 = \beta \tilde{R}_t^{(d)} \exp\left(d\frac{\gamma^2 \sigma^2}{2} - \gamma \mu_t \right).$$

Hence:

$$\log \tilde{R}_t^{(d)} = \gamma \mu_t - d\frac{\gamma^2 \sigma^2}{2} - \log \beta$$
$$= \gamma\big((1-\rho)\bar{g} + \rho g_t\big) - d\frac{\gamma^2 \sigma^2}{2} - \log \beta$$
$$\geq \gamma(1-\rho)\bar{g} - d\frac{\gamma^2 \sigma^2}{2} - \log \beta,$$

as $g_t \geq 0$. Thus, using this approximation, a reasonable estimate of $\log \beta$ conditional on the other parameters would be:

$$\gamma(1-\rho)\bar{g} - d\frac{\gamma^2 \sigma^2}{2} - \min_{t=1,\ldots,T} \log \tilde{R}_t.$$

Note though, that this is likely to be substantially biased down, since our approximation to $\mathbb{E}_t \exp(-\gamma g_{t+1})$ is biased upwards, as it ignores the impact of the bound. Now suppose we are trying to simultaneously estimate $\bar{g}$ as well. Given our approximation, the natural estimate of $\bar{g}$ is the solution for $\bar{g}$ to:

$$\gamma\big((1-\rho)\bar{g} + \rho\mathbb{E}g_t\big) = \frac{1}{T}\sum_{t=1}^{T} \log \tilde{R}_t^{(d)} + d\frac{\gamma^2 \sigma^2}{2} + \log \beta,$$

which, using our estimate of $\log \beta$ is the solution for $\bar{g}$ to:

$$\gamma\rho\mathbb{E}g_t = \frac{1}{T}\sum_{t=1}^{T} \log \tilde{R}_t^{(d)} - \min_{t=1,\ldots,T} \log \tilde{R}_t. \tag{1}$$

Since $\mathbb{E}g_t$ depends positively on $\bar{g}$, and our estimate of $\log \beta$ is biased down, so too will be our estimate of $\bar{g}$. This in turn causes further downwards bias in the estimate of $\log \beta$, which further pushes down $\bar{g}$, and so on.

To see the magnitude of these expected biases, note that our estimation algorithm approximates the stationary distribution of g_t by a Gaussian, say $N(\psi, \omega^2)$, hence, if $g \sim N(\psi, \omega^2)$ and $\varepsilon \sim N(0,1)$:

$$(1-\rho)\bar{g} + \rho g + \sigma\varepsilon \sim N(\tilde{\psi}, \tilde{\omega}^2),$$

where $\tilde{\psi} := (1 - \rho)\bar{g} + \rho\psi$ and $\tilde{\omega} := \sqrt{\rho^2\omega^2 + \sigma^2}$, so:

$$\psi = \mathbb{E}[\max\{0, (1 - \rho)\bar{g} + \rho g + \sigma\varepsilon\}] = \tilde{\psi}\Phi\left(\frac{\tilde{\psi}}{\tilde{\omega}}\right) + \tilde{\omega}\phi\left(\frac{\tilde{\psi}}{\tilde{\omega}}\right),$$

and:

$$\omega^2 = \mathbb{E}[(\max\{0, (1 - \rho)\bar{g} + \rho g + \sigma\varepsilon\} - \psi)^2]$$
$$= \tilde{\psi}^2\Phi\left(\frac{\tilde{\psi}}{\tilde{\omega}}\right)\left[1 - \Phi\left(\frac{\tilde{\psi}}{\tilde{\omega}}\right)\right] + \tilde{\omega}^2\left[\Phi\left(\frac{\tilde{\psi}}{\tilde{\omega}}\right) - \phi\left(\frac{\tilde{\psi}}{\tilde{\omega}}\right)^2\right] + \tilde{\psi}\tilde{\omega}\phi\left(\frac{\tilde{\psi}}{\tilde{\omega}}\right)\left[1 - 2\Phi\left(\frac{\tilde{\psi}}{\tilde{\omega}}\right)\right],$$

which implicitly define ψ and ω in terms of $\bar{g}$. Substituting $\psi \approx \mathbb{E}g_t$ into equation (1), and solving the resulting non-linear equations gives an estimate of $\bar{g}$ on our artificial data of -0.0080.[5] Since we constrain $\bar{g}$ to be greater than 10^{-6} when we estimate, this means that we should be unsurprised that we hit this bound. Given this, the implied estimate for β is:

$$\exp\left[10^{-6}\gamma(1 - \rho) - d\frac{\gamma^2\sigma^2}{2} - \min_{t=1,\dots,T}\log\tilde{R}_t\right],$$

which, on our data, equals 0.9759 under a first order approximation, and 0.9753 under a third order approximation. These estimates are closely in line with the estimates produced by our procedure, fully explaining our results.

Luckily, when we include cubature in the underlying simulation algorithm, ensuring that agent's expectations are truly rational, all of these biases disappear. Indeed, our estimates with "fast cubature" including the bound are the most accurate estimates of both β and $\bar{g}$ from any of our estimation runs. Given that "fast cubature" is computationally tractable even on large models, this suggests that we should be able to use our estimation procedure to get reliable estimates even in the medium scale DSGE models used by policy makers.

4.   Conclusion

This paper has presented an efficient algorithm for estimating non-linear models, including those with occasionally binding constraints. Thanks to the algorithm's dimension reduction techniques, the algorithm keeps the costs of forming predictive distributions manageable, allowing it to be used even on models for which simulation is expensive, such as those with occasionally binding constraints. We went on to show that identification is easier in non-linear models, and that estimating ignoring occasionally binding constraints may introduce substantial biases. The latter is particularly relevant for the zero lower bound on nominal interest rates, given the amount of time many economies have now spent at the bound. Macro-econometricians thus have no choice but to include the occasionally binding constraint in their model when they estimate medium-scale DSGE models, if they want to use up

---

[5] Replication code for this is contained in the "NaturalEstimate.m" file within the "Examples\BoundedProductivityEstimation" folder of DynareOBC.

to date data. Luckily, the algorithms presented in this paper will readily scale up to handle such models.

Code implementing all of the algorithms discussed here is contained in the author's "DynareOBC" toolkit which augments the abilities of Dynare (Adjemian et al. 2011) with the ability to solve and estimate models with occasionally binding constraints.

5.   References

Adjemian, Stéphane, Houtan Bastani, Fréderic Karamé, Michel Juillard, Junior Maih, Ferhat Mihoubi, George Perendia, Johannes Pfeifer, Marco Ratto, and Sébastien Villemot. 2011. *Dynare: Reference Manual Version 4*. CEPREMAP.

Adjemian, Stéphane, Michel Juillard, Fréderic Karamé, and Ferhat Mihoubi. 2016. *New Methods for Estimating Nonlinear Dynamic General Equilibrium Models with Particle Filters: A New Software Tool to Be Distributed via Dynare*. MACFINROBODS.

Andrieu, Christophe, Arnaud Doucet, and Roman Holenstein. 2010. "Particle Markov Chain Monte Carlo Methods." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 72 (3): 269–342.

Arasaratnam, Ienkaran, and Simon Haykin. 2009. "Cubature Kalman Filters." *Automatic Control, IEEE Transactions on* 54 (6): 1254–1269.

Canova, F. 2007. *Methods for Applied Macroeconomic Research*. Princeton University Press.

Cools, Ronald. 2003. "An Encyclopaedia of Cubature Formulas." *Oberwolfach Special Issue* 19 (3) (June): 445–453.

Fair, Ray C, and John B Taylor. 1983. "Solution and Maximum Likelihood Estimation of Dynamic Nonlinear Rational Expectations Models." *Econometrica* 51 (4) (July): 1169–85.

Fernández-Villaverde, Jesús, Pablo Guerrón-Quintana, and Juan F. Rubio-Ramírez. 2015. "Estimating Dynamic Equilibrium Models with Stochastic Volatility." *Journal of Econometrics* 185 (1): 216 – 229.

Fernández-Villaverde, Jesús, and Juan F. Rubio-Ramírez. 2007. "Estimating Macroeconomic Models: A Likelihood Approach." *The Review of Economic Studies* 74 (4): 1059–1087.

Genz, Alan, and Bradley D. Keister. 1996. "Fully Symmetric Interpolatory Rules for Multiple Integrals over Infinite Regions with Gaussian Weight." *Journal of Computational and Applied Mathematics* 71 (2): 299–309.

Hansen, Nikolaus. 2006. "The CMA Evolution Strategy: A Comparing Review." In *Towards a New Evolutionary Computation*, 75–102. Springer.

Hansen, Nikolaus, Andreas Ostermeier, and Andreas Gawelczyk. 1995. "On the Adaptation of Arbitrary Normal Mutation Distributions in Evolution Strategies: The Generating Set Adaptation." In *ICGA*, 57–64. Citeseer.

Herbst, Edward, and Frank Schorfheide. 2014. "Sequential Monte Carlo Sampling For Dsge Models." *Journal of Applied Econometrics* 29 (7) (November): 1073–1098.

Holden, Tom. 2016a. "Existence and Uniqueness of Solutions to Dynamic Models with Occasionally Binding Constraints." Unpublished.

———. 2016b. "Computation of Solutions to Dynamic Models with Occasionally Binding Constraints." Unpublished.

Kim, Sunghyun Henry, Jinill Kim, Ernst Schaumburg, and Christopher A. Sims. 2008. "Calculating and Using Second Order Accurate Solutions of Discrete Time Dynamic Equilibrium Models." *Journal of Economic Dynamics and Control* 32 (11): 3397–3414.

Kollmann, Robert. 2013. *Tractable Latent State Filtering for Non-Linear DSGE Models Using a Second-Order Approximation*. Working Papers ECARES. ULB – Universite Libre de Bruxelles.

Meyer-Gohde, Alexander. 2014. *Risky Linear Approximations*. SFB 649 Discussion Papers. Sonderforschungsbereich 649, Humboldt University, Berlin, Germany.

Pitt, Michael K. 2002. *Smooth Particle Filters for Likelihood Evaluation and Maximisation*. The Warwick Economics Research Paper Series (TWERPS). University of Warwick, Department of Economics.

Smets, Frank, and Rafael Wouters. 2003. "An Estimated Dynamic Stochastic General Equilibrium Model of the Euro Area." *Journal of the European Economic Association* 1 (5): 1123–1175.

White, Halbert. 1982. "Maximum Likelihood Estimation of Misspecified Models." *Econometrica* 50: 1–25.