AgFoodTrade

**New Issues in Agricultural,
Food & Bioenergy Trade**

Comparing numerical methods for solving the competitive storage model

Christophe Gouel (*INRA-AgroParisTech*)

Working Paper 2010-03

Comparing numerical methods for solving the competitive storage model[*]

Christophe Gouel[†]

May 3, 2010

Abstract

This paper compares numerical methods for solving the competitive storage model. Since storage implies an inequality constraint, the solution methods must be considered carefully. The model is solved using value function iteration, and several projection approaches, including parameterised expectations and decision rules approximation. Using a penalty function approach to smooth the inequality constraint, perturbation methods are also applied. Parameterised expectations proves the most accurate method, while perturbation techniques are shown inadequate for solving this highly nonlinear model. The endogenous grid method allows rapid solution if supply is assumed to be inelastic.

Keywords: binding constraint, nonlinear rational expectations models, numerical methods.

JEL classification: C63, D84, E37.

[†]INRA, CEPII and Ecole Polytechnique (`christophe.gouel@inra.grignon.fr`)

1 Introduction

The competitive storage model is the workhorse of neoclassical studies on the price behaviour of storable commodities (Williams and Wright, 1991, Wright, 2001). Although its empirical properties continue to be debated (Cafiero and Wright, 2006), this model provides justification for the main properties of commodity prices: positive serial correlation, skewness, successions of long periods of doldrums and short periods of high prices. It also serves as a normative benchmark for analysing public intervention in commodity markets (Miranda and Helmberger, 1988). Like most dynamic stochastic problems, it cannot be solved analytically. For this reason, understanding the properties of available numerical solutions is important for securing precise econometric estimates or reliable policy conclusions. One example of the importance of good numerical solutions is provided by Cafiero et al. (2010), who show that Deaton and Laroque's (1995, 1996) econometric estimates are not reliable due to the imprecise methods used to approximate the model. Another aspect that requires good knowledge of numerical solutions is its extension to higher dimensional problems. Many important questions on commodity prices (e.g., the comovement of commodity prices, the relationship between trade and prices, the consequences of price stabilisation programmes, and the effect of monetary policies on prices) imply problems with several state and decision variables that become very challenging to solve numerically.

This paper compares different approaches to solving the competitive storage model. It implements three methods not tested previously on this model: perturbation, endogenous grid methods and storage rule approximation. As perturbation methods only work with smooth problems, two alternative models are considered: one with a non-negativity constraint on storage and the other with a precautionary motive for storage that rules out zero stock situations and smoothes the model. Comparing solution methods for this smooth model is the object of Miranda's (1997) study. Our approach differs from Miranda's in considering the model with a non-negativity constraint—the more traditional representation of storage problems—and in its different focus. Miranda (1997) studies mainly the effect of varying approximation methods for a given projection algorithm which parameterises current price. The present paper tests various algorithms, but limits variations between approaches by holding the set of tools constant: functions are approximated by cubic splines; convergence to the rational expectations equilibrium is achieved by successive approximations; value function iteration and projection problems are resolved by collocation; and the same degree of approximation is used for each method. The precision of the solutions is assessed using a measure of the Euler equation error that is derived to account for the switching-regime behaviour.

There is a large body of work on the comparison of numerical methods in economics. Most concerns the stochastic growth model (see Aruoba et al. (2006) and Heer and Maußner (2008) for two recent studies). This literature covers only some of the methods used for the storage model. The specificities of the storage model come from the non-negativity constraint on stocks. For these kinds of constraints, Christiano and Fisher (2000) study numerical methods in a stochastic growth model with irreversible investment. This paper uses some of the same methods as Christiano and Fisher, but proposes also additional ones.

The study of numerical methods for the storage model presents interests beyond the study of commodities behaviour. The storage model, in effect, is formally very close to the consumption/saving problem under income uncertainty (Deaton, 1991, Carroll, 2001). Their problematics are identical: how much an agent should consume and save today when future resources are volatile. In the optimal consumption problem, a consumer's borrowing capacity is limited or he applies a limit as a precaution. At each period, the consumer chooses how much to save to protect his future consumption against adverse income shocks. In the storage model, there are three agents: a consumer, a storer acting competitively, and a producer using a stochastic

production function. Without distortion, this problem can be stated as a planner problem. The planner must choose how much grain to carry over to the next period. There is one main difference between the two models: storage is costly and grain may get spoilt during storage, while in a consumption/saving problem savings are remunerated. But the impatience of consumers makes saving costly, which renders the two model almost identical. This similarity was first noted by Deaton, who worked on both fields (Deaton, 1991, Deaton and Laroque, 1992). The longstanding ignorance about this resemblance had some consequences: a numerical method for the storage model was proposed by Gustafson (1958), while the consumption/saving problem under income uncertainty was solved using the misleading certainty equivalence or log-linearisation until the work by Zeldes (Barsky et al., 1986, Zeldes, 1989).

This paper provides several notable results. First, the parameterised expectations approach stands out as the most precise method. It has well known good properties for solving models with occasionally binding constraints (Christiano and Fisher, 2000), but in our case its good performances extend also to the smooth version of the model. Indeed, unlike other methods that approximate highly nonlinear functions, this method approximates a function that is almost linear. Second, perturbation methods are shown to be inadequate for this model. Given the nonlinearity of storage behaviour, low order perturbations are imprecise and lead to negative storage, whereas high order perturbations present diverging behaviour. Third, when supply is assumed to be inelastic, as is the case in most econometric studies, the endogenous grid method is by far the fastest, which makes it especially valuable for computationally intensive work.

2 The competitive storage model

2.1 Model equations

The storage model analysed in Wright and Williams (1982) is used throughout the paper. It features a market for a storable commodity with a competitive storer, a producer whose output is subject to multiplicative shocks and a final demand.

The activity of the competitive risk-neutral storer is to transfer the commodity from one period to the next. Storing the quantity S_t from period t to period $t+1$ entails a physical cost $\Phi(S_t)$, a purchase cost $P_t S_t$, with P_t the market price, and an opportunity cost. A share δ of the commodity deteriorates during storage. The benefits valued in period t are $P_{t+1} S_t \cdot (1-\delta)/(1+r)$, with r the interest rate. The profit expected by the storer is

$$\mathrm{E}_t\left(\Pi_{t+1}^S\right) = \left[\frac{1-\delta}{1+r}\,\mathrm{E}_t\left(P_{t+1}\right) - P_t\right] S_t - \Phi(S_t), \tag{1}$$

where E_t is the expectation operator conditional on period t information. Taking into account the possibility of a corner solution (i.e., the non-negativity constraint of storage), expected profit maximisation yields the following complementary condition[1]

$$S_t \geq 0 \quad \perp \quad \frac{1-\delta}{1+r}\,\mathrm{E}_t\left(P_{t+1}\right) - P_t - \Phi'(S_t) \leq 0, \tag{2}$$

which means that inventories are null when the marginal cost of storage, including purchase cost, is not covered by the expected marginal benefits; for positive inventories, the arbitrage equation holds with equality.

---

[1]Complementarity conditions are written in what follows using the "perp" notation ($\perp$). It means that the expressions on each side of the sign are orthogonal. One holds with strict inequality when the other holds with strict equality.

A representative producer makes his productive choice one period before bringing the output to market. He plans in period t a production level H_t for period $t+1$, but a disturbance affects final production (e.g., weather disturbances). The expected profit can be written as

$$\mathrm{E}_t\left(\Pi_{t+1}^H\right) = \frac{\mathrm{E}_t\left(P_{t+1}H_t\epsilon_{t+1}\right)}{1+r} - \Psi\left(H_t\right), \tag{3}$$

with $\Psi\left(H_t\right)$ the cost of planning the production H_t, and $H_t\epsilon_{t+1}$ the realised production. ϵ_{t+1} is the realisation of a stochastic process, supposed i.i.d. and distributed following a normal centred on 1 with standard error σ. The planned production derived from expected profit maximisation satisfies

$$\mathrm{E}_t\left(P_{t+1}\epsilon_{t+1}\right) = \left(1+r\right)\Psi'\left(H_t\right). \tag{4}$$

Final demand is given by the inverse demand function $P\left(D\right)$, where D is the quantity consumed. The shocks being i.i.d., the state of the model is defined by total availability $A_t = \left(1-\delta\right)S_{t-1} + H_{t-1}\epsilon_t$. Accounting for carry over storage, market equilibrium can be written as

$$A_t = D_t + S_t. \tag{5}$$

Using Equation (5), the current price can be eliminated through the storage arbitrage condition (2):

$$S_t \geq 0 \quad \perp \quad \frac{1-\delta}{1+r}\mathrm{E}_t\left(P_{t+1}\right) - P\left(A_t - S_t\right) - \Phi'\left(S_t\right) \leq 0. \tag{6}$$

The storage model is defined by its first-order conditions (4) and (6). A restriction of this model to inelastic supply is the standard tool for econometric works. It reduces to Equation (6), with H_t taken as constant. Although this simplified model is not the object of this work, we will discuss some of its numerical specificities.

2.2 Model parameterisation

Physical storage costs are usually assumed to be proportional to the amount stored (Wright and Williams, 1982). In our first parameterisation, we take $\Phi'\left(S\right) = 0.01$ (i.e., 1% of the deterministic steady state price assumed to be 1). One alternative is to include in the storage cost a convenience yield, which would tend to take high negative values for low stocks, preventing the occurrence of stockouts. Miranda (1997) uses a logarithmic function to achieve this. We follow his approach by taking $\Phi'\left(S\right) = 0.3 + 0.1\log\left(S\right)$, which conveniently removes the complementarity condition in Equation (6). As the logarithm tends to minus infinity for low stocks, stocks always remain positive. The two models show similar, but distinctive, behaviours (see Figure 1), with the model with convenience yield being smoother than the model with the non-negativity constraint. Negative marginal storage costs can be justified on the grounds that they account for both physical storage and convenience yield. To reconcile the fact that low stocks exist for apparently negative returns to storage, some authors, such as Kaldor (1939) and Working (1948), conclude that storers can expect a convenience yield from holding stock despite an apparent negative yield. This convenience yield comes from the possibility of being able to use of stock at any moment. Considering storage costs that include a convenience yield is equivalent to storage responding to different motives: a speculative motive that allows stockouts and a precautionary motive that precludes them (Carter and Revoredo Giha, 2007).

The inverse demand function is taken to be isoelastic:

$$P = D^{1/e}. \tag{7}$$

4

The production cost function also follows an isoelastic form

$$\Psi\left(H\right) = \frac{H^{\alpha+1}}{\left(1+r\right)\left(\alpha+1\right)}. \tag{8}$$

The production cost is normalised by $1+r$ to ensure that the deterministic steady state production is at 1. Table 1 presents the parameters used to calibrate the model. They are assumed such that the deterministic steady state of the model with non-negativity constraint is 1 for availability, price and production. Convenience yield implies that there is always some stock, even when price is constant and there is no speculative opportunity, so the deterministic steady state differs slightly with stock, availability, price and production respectively at 0.0338, 1.0336, 0.9998 and 1.0001. Supply elasticity is 0.2 and demand elasticity -0.3.

Table 1. Calibrated parameters

Parameter	δ	r	e	α	σ
Value	0.01	0.03	-0.3	5	0.15

For this calibration, the storage model reproduces some stylised facts about agricultural commodities markets (Table 2). Prices are positively serially correlated. Their distribution is asymmetric with a higher frequency of high than low prices, because storage alleviates low prices by stockpiling, but cannot alleviate all episodes of high prices as stocks cannot be negative. The occurrence of zero stock periods should be small to preserve the serial correlation (it is storage that creates autocorrelation in prices) and to limit episodes of high prices. Because market availability is smoothed by the storage and supply reaction, final demand oscillates very little (6% and 9% of coefficient of variation). In opposition to prices, final consumption is positively skewed, because high consumption is prevented by storage, and periods of low consumption occur because of stockouts.

Table 2. Storage model behaviour

	With non-negativity constraint	With convenience yield
Occurrence of stockouts (%)	9	0
Price		
Autocorrelation	0.26	0.17
Coefficient of variation	0.32	0.45
Skewness	7.37	7.50
Consumption demand		
Coefficient of variation	0.06	0.09
Skewness	-2.33	-0.90

Figure 1 illustrates the behaviour of the storage model. It shows the numerical difficulties we encounter when deciding about the different approximation schemes. The difficulties arise from the non-negativity constraint of storage. Below a threshold availability, the expected profit from stockpiling grains is negative, as the expected price does not cover storage costs and the high current price. For a negative expected profit, there is no stock carried over to the next period. So the storage rule encompasses two regimes: a no-stock regime below the threshold, and a stock increasing with availability above the threshold. This behaviour affects all the variables in the model. Below the threshold, the market price is set by consumption

demand, but above it, the demand for storage adds to consumption demand. So, the market demand function presents a kink at the availability threshold. The behaviour of producers depends on the expected price in the next period. This expected price is constant for low availability because there is no stock to connect successive periods. Therefore, planned production is constant until the threshold availability is reached and then decreases with the increase in stocks. For a more complete description of the properties of the storage model see Wright (2001).

The model parameterised with convenience yield has no threshold and is smooth. However, its behaviour is similar to the model with inequality constraint: it displays highly nonlinear curves that feature a transition between a regime with stocks driven not by speculation but by convenience yield to a speculative regime. Planned production or expected price curves are strongly nonlinear when represented with respect to availability, but much more linear when stock is used in x-axis (Figure 1c and 1d).

3   Solution methods

The seminal paper by Gustafson (1958) proposes several solution methods: iterating the value function, iterating the marginal value function, and approximating the storage rule by a piecewise linear function, but he carries his work with value function iteration. Gardner (1979) extends Gustafson's value function iteration method to a case with elastic supply (for an early review of the dynamic programming approach to the storage problem, see Plato and Gordon, 1983). Value function iteration is known to be a reliable method. It has convergence properties based on dynamic programming theory, but it is slow and, since it solves a central planner problem, it does not enable finding a suboptimal competitive equilibrium.

Two more recent methods are generally used to solve the storage model. Wright and Williams (1982) were the first to propose a solution based on Euler equations. They use low order polynomials to fit expected price to current stock. They believe that low order polynomials provide sufficient precision because the expected price function is smooth and not too nonlinear (the small plot in Figure 1d). They apply their method to various settings, such as the storage-trade model, a model featuring the interaction of public buffer-stock and private stock, and a storage model with news arrival (Williams and Wright, 1991). This numerical strategy is applied in several works on the storage model, such as Miranda and Helmberger (1988), Lence and Hayes (2002), and Park (2006).

Another method, which has two alternative forms, was proposed by Deaton and Laroque (1992) (a method similar to that applied to optimal consumption problems under liquidity constraints by Deaton, 1991) and Miranda and Glauber (1995). Both forms approximate the current price function. The first solves the system using a fixed-point algorithm; the second applies a collocation method. Miranda (1997) compares various approximation schemes for this latter method.

The methods described above do not exhaust the possibilities, however. For a model with convenience yield, Judd (1998) suggests direct approximation of the storage rule, but does not apply it. Approximating decision rules is a usual method for solving the stochastic growth model (Aruoba et al., 2006) and other dynamic models. The method is often used because it allows direct simulations once decision rules are identified. It is generally not used for the storage model because the storage rule is kinked, which makes it difficult to approximate. However, its limited accuracy could be compensated by higher speed, therefore, in the present paper we implement decision rules approximation.

Newbery and Stiglitz (1982) propose to expand the storage rule as a Taylor series at the kink point. We adopt a similar method in this paper. The model is approximated at low orders around its steady state. The
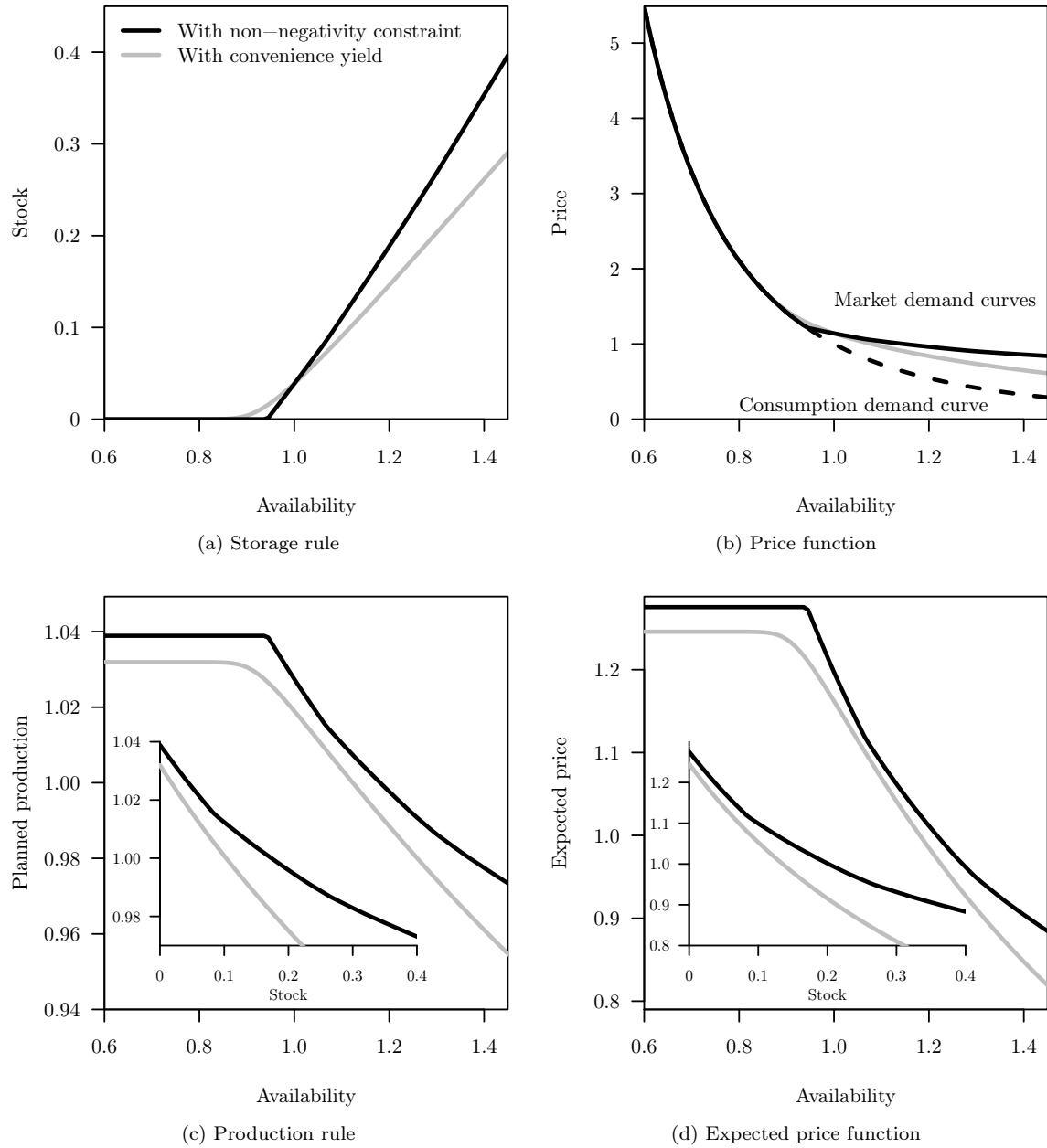
Figure 1. Characterisation of the storage model behaviour. Black curves correspond to the model with non-negativity constraint, grey curves to the smoother model with convenience yield. For the price curves (b), the dashed line corresponds to demand for final consumption; total market demand (solid lines) includes demand for storage. For the production rule (c) and the expected price function (d), the plot includes another version of the same functions, but drawn against the decision variable, stock, rather than the state variable, availability. These additional curves show the important differences in nonlinearity depending on the explanatory variable.

use of a Taylor expansion is restricted to smooth functions, so we apply it only to the case with convenience yield. This method, the perturbation method, is commonly used with the stochastic growth model.

With the exception of the perturbation methods, all the methods applied in this paper share the same set of numerical issues: how to pass from an integral over ϵ to a finite dimensional problem, i.e., how to discretise the shocks; how to solve the first order conditions (4) and (6), which present specific problems caused by the complementarity condition; and which method to use to approximate the unknown functions. Following the treatment of these common issues, we present the different methods.

3.1 Shocks discretisation

Productivity shocks ϵ are discretised and the integrals over the shocks are calculated using a 9-node Gauss-Hermite quadrature, allowing the exact integration of degree 19 polynomials weighted by a normal distribution. The Gaussian formula transforms an expectation term into a sum. For example, next-period expected price expressed as a function of next-period availability is approximated as

$$\mathrm{E}\left[P\left(A\right)\right] = \mathrm{E}\left[P\left(\left(1-\delta\right)S + H\epsilon\right)\right] \approx \sum_{l=1}^{L} w_l P\left(\left(1-\delta\right)S + H\epsilon_l\right),$$

with ϵ_l and w_l the nodes and weights of the quadrature.

3.2 Solving the equations

The competitive storage model reduces to two equations: the storage arbitrage condition (6) and the producer incentive equation (4). The arbitrage condition takes a complementarity form when marginal storage costs are constant. This is not a traditional nonlinear equation that can be solved with a Newton solver. We solve this complementarity problem by using the `ncpsolve` solver (Miranda and Fackler, 2002). When the inequality constraint is smoothed by the convenience yield hypothesis, the problem can be solved using any of the traditional nonlinear solvers.[2] In our case, we use also the `ncpsolve` solver, which can also solve simple nonlinear problems. Equations are solved with a precision of 10^{-9}.

Given the simplicity of the problem, we could avoid having to use a complementarity solver. The two equations can be solved by any nonlinear solver and, once a solution is found, the positivity of storage can be checked. For negative storage, we would force it at zero and solve for the planned production. This is the approach taken by Williams and Wright. However, this is not possible for a problem with more state variables, such as a multi-commodity model or a storage-trade model. A complementarity solver introduces only limited overheads compared to a traditional nonlinear solver, so we prefer this more direct solution.

With inelastic supply, it is necessary to solve only the storage arbitrage condition. The problem can be written as a fixed-point problem and solved through simple operations without the need for solvers (Deaton and Laroque, 1992, Judd, 1998). Deaton and Laroque (1992) propose this solution and it has been applied in most econometric work on the storage model because of its speed (Chambers and Bailey, 1996, Michaelides and Ng, 2000, Osborne, 2004).

3.3 Approximation methods

The first numerical methods for the storage model involve approximating the value function by space discretisation (Gustafson, 1958, Gardner, 1979). Later, Wright and Williams (1984) use low-order polynomials

---

[2]Even if smoother this problem is not simple to solve. Indeed, the logarithm does not tolerate negative storage level and the solver must be prevented from exploring this region.

to approximate conditional expectations. The polynomials are fitted to grid values by ordinary least-squares. Miranda and Glauber (1995) solve the storage model using Chebychev collocation methods.

The virtues of the various approximation schemes used to study dynamic economic problems have been discussed by various authors (see e.g., Judd, 1992, 1998, Christiano and Fisher, 2000). The choice simply reduces to splines versus Chebychev polynomials. In this context, the storage model presents a specific difficulty. Because of the non-negativity constraint on stocks, most functions of the model present a kink. They are continuous, but not derivable (see Figure 1), which creates problems for both interpolation schemes and requires a high number of free parameters to achieve a satisfactory level of precision. The difficulty is greater with polynomials, since the discontinuity affects the entire interpolation interval (Miranda and Fackler, 2002), which means that, in this case, splines should behave better. For the storage model with convenience yield, which has no kink, Miranda (1997) compares various approximation schemes. He finds that approximation by cubic splines is preferable to the other approaches. We follow his result and use only cubic splines interpolation.

The interpolation is done with the Matlab *compecon* toolbox, developed to accompany Miranda and Fackler (2002). Functions defined over storage are approximated over the interval $[0, 1.11]$ ($[2.2 \times 10^{-16}, 1.11]$ with convenience yield). Functions defined over availability are approximated over the interval $[\min(\epsilon_l), 2.4]$. These intervals were chosen to include the values we are most likely to find during a simulation. They are determined by tâtonnement until the smallest intervals including most points inside the empirical distribution were found.

Values outside intervals may be encountered during the resolution of the functional equations problem, and, in these cases, there is no extrapolation: values are taken as approximations of the last points of the interval. For all methods except perturbation where we rely on a separate software, convergence to the true approximating function is assured by a successive approximation algorithm.

Algorithm performance and convergence depend greatly on the initial guess about the approximated function. Our initial guess is based on the corresponding functions at steady state values. When price or expected price are approximated, we correct the first guess for the expected effects of storage. Storage will occur in periods of low prices and, thus, will prevent situations of extremely low prices. We, therefore, restrict the first guess to values above 70% of the steady state.

Approximation methods make it possible to transform an infinite dimension problem (finding a function satisfying some conditions over a continuum) into a finite dimension problem (finding the values satisfying the conditions at some nodes and approximate between the nodes). The function $f(\cdot)$—to be defined later for each method—is approximated by $f_n(\cdot)$: $f(x) \approx f_n(x) = \sum_{j=1}^{J} \theta_j^n B_j(x)$. θ_j^n are basis coefficients to be determined and $B_j(\cdot)$ are the basis functions of a cubic spline (Judd, 1998). At each iteration of n, the basis coefficients are updated in a fitting step. Except for endogenous grid method where the coefficients are updated through a least-squares regression, the fitting step consists of solving the linear system of J equations in the unknown θ_j^n:

$$\sum_{j=1}^{J} \theta_j^n B_j(x_i) = f(x_i) \quad \text{for } i = 1, \ldots, J, \tag{9}$$

with $\{x_1, x_2, \ldots, x_J\}$, the collocation nodes.

3.4 Value function iteration

The model can be re-cast as a planner problem (Scheinkman and Schechtman, 1983). Its recursive dynamic formulation is

$$V\left(A_{t}\right) = \max_{S_{t} \geq 0, H_{t}} \int_{\bar{D}}^{A_{t}-S_{t}} P\left(D\right) \mathrm{d}D - \Phi\left(S_{t}\right) - \Psi\left(H_{t}\right) + \frac{1}{1+r} \mathrm{E}_{t} V\left(\left(1-\delta\right) S_{t} + H_{t}\epsilon_{t+1}\right), \qquad (10)$$

where $V\left(\cdot\right)$ is the value function of the problem. When $\bar{D}$ is a positive number, $\int_{\bar{D}}^{A_{t}-S_{t}} P\left(D\right) \mathrm{d}D$ is gross consumer benefit. Without any loss of generality, we make $\bar{D}$ sufficiently high such that $\int_{\bar{D}}^{A_{t}-S_{t}} P\left(D\right) \mathrm{d}D = \left(A_{t} - S_{t}\right)^{1+1/e} / \left(1 + 1/e\right)$.

To solve by value function iteration (VFI), we define a grid on availability, $\{A_1, A_2, \ldots, A_J\}$. For a given value function, it is necessary to determine optimal storage and planned production by solving the two arbitrage conditions. The value function approximation can be updated using these optimal decisions. We stop the iterations when changes in the value function decrease below a threshold. The algorithm runs as follows:

1. Initialise by taking the first guess: $V_0\left(A\right) = A^{\left(1+1/e\right)} / \left(1 + 1/e\right) - \Psi\left(H^{SS}\right)$,[3] with $V_0\left(\cdot\right)$ a J-breakpoint spline.

2. For each A_i in $\{A_1, A_2, \ldots, A_J\}$, find S_i and H_i that solve the first order conditions

$$S_i \geq 0 \quad \perp \quad -P\left(A_i - S_i\right) - \Phi'\left(S_i\right) + \frac{1-\delta}{1+r} \sum_{l=1}^{L} w_l V_n'\left(\left(1-\delta\right) S_i + H_i\epsilon_l\right) \leq 0, \qquad (11)$$

$$-\left(1+r\right) \Psi'\left(H_i\right) + \sum_{l=1}^{L} w_l \epsilon_l V_n'\left(\left(1-\delta\right) S_i + H_i\epsilon_l\right) = 0. \qquad (12)$$

3. Update the J-breakpoint spline $V_{n+1}\left(\cdot\right)$ using the system of J equations:

$$V_{n+1}\left(A_i\right) = \frac{\left(A_i - S_i\right)^{1+1/e}}{1 + 1/e} - \Phi\left(S_i\right) - \Psi\left(H_i\right) + \frac{1}{1+r} \sum_{l=1}^{L} w_l V_n\left(\left(1-\delta\right) S_i + H_i\epsilon_l\right) \text{ for } i = 1, \ldots, J. \quad (13)$$

   $V_n\left(\left(1-\delta\right) S_i + H_i\epsilon_l\right)$ and its derivate are computed by interpolation.

4. If $\|V_{n+1}\left(A\right) - V_n\left(A\right)\|_2 \geq 10^{-7}$ then increment n to $n+1$ and go to step 2.

Comparing this crude dynamic programming algorithm with the projection methods that follow is unfair, since there are several procedures that could accelerate a value function iteration algorithm (Rust, 1996). But, given VFI is limited to models amenable to a planner problem, a deeper analysis of this algorithm does not seem relevant.

3.5 Projection methods

Projection methods are the most standard methods applied to the storage model (Wright and Williams, 1982, Miranda and Helmberger, 1988). Its use was formalised in economics by Judd (1992) and consists

---

[3]Variables with the superscript SS are the deterministic steady state values.

of defining an approximation scheme and using it to minimise a residual function that defines the rational expectations equilibrium. Storage model is a perfect test-case for projections methods, since they can be applied in various ways depending on the functions being approximated. The present paper demonstrate that choosing the smoothest function for approximation can result in significant precision gains.

Projection methods can differ in terms of which functions they approximate, but the residual is always defined by the two Euler equations (4) and (6). As it is not possible to make the residual function zero for all points, several possibilities exist for defining objectives. They differ in their definition of the norm over the residual. The Galerkin method forces the inner product of the residual with the basis functions to be zero. The collocation method requires the interpolant to make the residual function equal to zero for all nodes. Collocation only obliges that the model be solved for the same number of points as the number of free parameters in the interpolant, and has proved slightly less precise than Galerkin for the resolution of significantly fewer points (Judd, 1992); thus in this paper we use collocation for all projection methods, except endogenous grid.

We follow the usual practice in the literature on storage model in implementing the simplest method to converge to the true rational expectations equilibrium, i.e., time iteration. At each iteration step, the new approximation is defined by the application of the current approximation to the next-period problem. The algorithm stops when the function used for two successive periods are almost identical. The collocation conditions could also be approached as a nonlinear problem and solved using a Newton solver.

We present three versions of the projection method which differ only in the functions they approximate: current price approximation; parameterisation of the expected price; storage rule approximation.

3.5.1   Parameterised current price algorithms

The literature proposes two approaches to solving the storage model by parameterising the price function. Deaton and Laroque (1992) propose a fixed-point algorithm, which is very fast because it does not require any rootfinding step, but is applicable only to cases without supply reaction. So it cannot be used here. The other approach is suggested by Miranda and Glauber (1995), who use a projection algorithm with Chebychev collocation.

In this paper, we implement two algorithms to solve the model with a parameterised price function. First, Miranda and Glauber's approach, which we call time iteration approach. Second, a fixed-point algorithm, based on the endogenous grid method (EGM) proposed by Carroll (2006), which is faster than Deaton and Laroque's algorithm and is applicable to all the situations of interest in this paper.

The time iteration approach   Miranda and Glauber's algorithm and the value function iteration are almost identical. The former iterates the marginal of the value function—i.e., the price function—rather than the value function. More precise results can be expected because VFI requires a good approximation of both the value function and its derivative.

1. Initialise $g_0(A) = \max\left[P(A), 0.7P^{SS}\right]$ with $g_0(\cdot)$ a J-breakpoint spline.

2. For each A_i in $\{A_1, A_2, \ldots, A_J\}$, find S_i and H_i that solve the first order conditions

$$S_i \geq 0 \quad \perp \quad -P(A_i - S_i) - \Phi'(S_i) + \frac{1-\delta}{1+r} \sum_{l=1}^{L} w_l g_n((1-\delta)S_i + H_i\epsilon_l) \leq 0, \tag{14}$$

$$-(1+r)\Psi'(H_i) + \sum_{l=1}^{L} w_l \epsilon_l g_n((1-\delta)S_i + H_i\epsilon_l) = 0. \tag{15}$$

3. Determine the J-breakpoint spline $g_{n+1}(\cdot)$ that solves the system:

$$g_{n+1}(A_i) = P(A_i - S_i) \quad \text{for } i = 1, \ldots, J. \tag{16}$$

4. If $\|g_{n+1}(A) - g_n(A)\|_2 \geq 10^{-7}$ then increment n to $n+1$ and go to step 2.

The endogenous grid method Carroll (2006) proposes a very efficient method for solving dynamic stochastic optimisation problems. The idea is to solve the Euler equation by iterations of the end-of-period decision variable rather than the state variable. This suppresses the need to solve a non-linear equation and replaces the rootfinding step by arithmetic operations. Applied to the model with convenience yield and inelastic supply, it would simply consist of iterating the function $k_n(A)$ on a grid of storage points, $\{S_1, S_2, \ldots, S_M\}$. $k_n(A)$ approximates the current price function, with

$$A_i = S_i + P^{-1}\left(\frac{1-\delta}{1+r} \mathrm{E}\left[k_n((1-\delta)S_i + H\epsilon_l)\right] - \Phi'(S_i)\right), \tag{17}$$

and $k_{n+1}(A_i) = P(A_i - S_i)$.

Carroll (2006) explains also how the method can be applied to problems with inequality constraints. One needs only to change the iteration step in k_n and make sure that the grid on S includes zero:

$$k_{n+1}(A) \text{ approximates } \begin{cases} P(A_i - S_i) & \text{for all grid points,} \\ P(A) & \text{for } A \leq \min(A_i). \end{cases}$$

As soon as there is more than one decision variable, the methods becomes trickier. Barillas and Fernández-Villaverde (2007) extend the method to a case with two decision variables. As these decision variables must be mutually consistent, it is not possible to define a grid on both variables. The idea is to define the grid on one decision variable and to find the other by solving the corresponding first order equation. In our case, we define a grid on storage, solve Equation (4) for planned production, and compute availability and price with (17). We lose one of the interests of EGM, i.e., avoiding rootfinding step, but in this case rootfinding is quite simple and we can accelerate the algorithm by applying it every two or three iterations. The algorithm runs as follows:

1. Initialise $k_0(A) = \max\left[P(A), 0.7P^{SS}\right]$ with $k_0(\cdot)$ a J-breakpoint spline.

2. For each storage level S_i in $\{S_1, S_2, \ldots, S_M\}$ find H_i that solve the Euler equation for production

$$-(1+r)\Psi'(H_i) + \sum_{l=1}^{L} w_l \epsilon_l k_n((1-\delta)S_i + H_i\epsilon_l) = 0. \tag{18}$$

To accelerate the algorithm, this step can be applied every two or three iterations.

3. For each combination $\{S_i, H_i\}$ calculate A_i with

$$A_i = S_i + P^{-1}\left(\frac{1-\delta}{1+r}\sum_{l=1}^{L} w_l k_n\left((1-\delta)S_i + H_i\epsilon_l\right) - \Phi'\left(S_i\right)\right).$$

(19)

4. Determine the J-breakpoint spline $k_{n+1}\left(\cdot\right)$ that best fits (in a least-squares meaning) the points $\{A_i, P\left(A_i - S_i\right)\}$ and the function $P\left(A\right)$ for $A < \min\left(A_i\right)$.

5. If $\|k_{n+1}\left(A\right) - k_n\left(A\right)\|_2 \geq 10^{-7}$ then increment n to $n+1$ and go to step 2.

Unlike the other methods, we take more grid points for the EGM than the number of breakpoints in the approximating function. Consequently, the approximation is updated by a least-squares step. This is required by the fact that the definition of the spline requires at least one observation between two breakpoints, which cannot be guaranteed if we take as many observations as breakpoints because breakpoints and grid points are chosen independently.

3.5.2 Parameterised expectations algorithm

The parameterised expectations algorithm (PEA) approximates the expectation terms that appear in the Euler equations. Popularised in the macroeconomics literature by den Haan and Marcet (1990), this algorithm was originally developed by Wright and Williams (1982) for the storage model.[4] Christiano and Fisher (2000) note that the two methods are slightly different. Den Haan and Marcet approximate conditional expectations with respect to the state variables when Wright and Williams use conditional expectations with respect to the current-period decision.[5]

In the case of the storage model, the den Haan and Marcet PEA would approximate $\mathrm{E}\left(P_{t+1}|A_t\right)$, whereas Williams and Wright parameterise $\mathrm{E}\left(P_{t+1}|S_t\right)$. The latter function should be smoother than the former, because expected price with respect to availability shows two regimes with a kink, and is quite nonlinear. For low availabilities, the expected price is constant. It is the stockout case. After a threshold, expected price decreases with current availability. The relation between expected price and storage is smoother and more linear, as it does not include the first stockout regime (compare the two plots in Figure 1d). This difference was already appreciated in the case of the stochastic growth model with irreversible investment in Christiano and Fisher (2000). In what follows, we implement only Wright and Williams's approach.

The algorithm below is adapted from Wright and Williams (1984). This method requires approximation of two functions: expected price with respect to storage, $f_n^S\left(S\right) \approx \mathrm{E}\left(P\,|\,S\right)$, and producer incentive price with respect to storage, $f_n^H\left(S\right) \approx \mathrm{E}\left(P\epsilon\,|\,S\right)$.[6]

1. Initialise $f_0^H\left(S\right) = \mathrm{E}\left\{\max\left[P\left((1-\delta)S + H^{SS}\epsilon - S^{SS}\right), 0.7P^{SS}\right]\epsilon\right\}$ and
   $f_0^S\left(S\right) = \mathrm{E}\left\{\max\left[P\left((1-\delta)S + H^{SS}\epsilon - S^{SS}\right), 0.7P^{SS}\right]\right\}$, both are J-breakpoint splines.

---

[4]They describe the algorithm in Wright and Williams (1984).

[5]The two versions differ also in the method used to compute the approximation. Wright and Williams use a projection method, and den Haan and Marcet implement a stochastic simulation approach, dubbed by Judd et al. (2009) as simulation-based PEA. Stochastic PEA may be more intuitive than its projection counterpart, but it is less accurate and its convergence is not warranted, so it is not implemented here.

[6]Wright and Williams (1984) approximate directly the value of planned production instead of the producer incentive price. The two schemes are equivalent, but to follow the logic of PEA we parameterise the expectation term.

2. For each storage level S_i in $\{S_1, S_2, \ldots, S_J\}$ and shocks ϵ_l define the availability

$$A_{i,l} = (1 - \delta) S_i + (\Psi')^{-1} \left(f_n^H (S_i) / (1 + r) \right) \epsilon_l. \tag{20}$$

3. For each element $A_{i,l}$ solve the storage Euler equation to find $S_{i,l}$

$$S_{i,l} \geq 0 \quad \perp \quad \frac{1 - \delta}{1 + r} f_n^S (S_{i,l}) - P (A_{i,l} - S_{i,l}) - \Phi' (S_{i,l}) \leq 0. \tag{21}$$

4. Determine the J-breakpoint splines $f_{n+1}^S (\cdot)$ and $f_{n+1}^H (\cdot)$ that solve the systems

$$f_{n+1}^S (S_i) = \sum_{l=1}^{L} w_l P (A_{i,l} - S_{i,l}) \qquad \text{for } i = 1, \ldots, J, \tag{22}$$

$$f_{n+1}^H (S_i) = \sum_{l=1}^{L} w_l \epsilon_l P (A_{i,l} - S_{i,l}) \qquad \text{for } i = 1, \ldots, J. \tag{23}$$

5. If $\|f_{n+1}^S (S) - f_n^S (S)\|_2 \geq 10^{-7}$ or $\|f_{n+1}^H (S) - f_n^H (S)\|_2 \geq 10^{-7}$ then increment n to $n + 1$ and go to step 2.

We expect this method to be more precise than the others, since the conditional expectations are smoother than most of the other functions to be approximated in the model. However, this method requires two conditional expectations to be approximated, each corresponding to a different decision variable. The parameterised current price approach or VFI, on the other hand, are more parsimonious, since they require only one function approximation.

3.5.3 Storage rule approximation

Although widely used with other dynamic models, decision rules approximation has not been applied to solving the storage model. The method was suggested in Judd (1998) for the case with convenience yield, but here we apply it to both cases: convenience yield and non-negativity constraint. In a time iteration approach, the approximated storage rule is used inside the expectation terms to define next-period storage. Storage undertaken at the current period, found by solving the first order conditions over a grid of availability points, allows the approximation to be updated. The production rule is not necessary to define the rational expectations equilibrium.

1. Initialise $s_0 (A) = 0$ with $s_0 (\cdot)$ a J-breakpoint spline.

2. For each A_i in $\{A_1, A_2, \ldots, A_J\}$, find S_i and H_i that solve the first order conditions

$$S_i \geq 0 \quad \perp \quad -P (A_i - S_i) - \Phi' (S_i)$$
$$+ \frac{1 - \delta}{1 + r} \sum_{l=1}^{L} w_l P \big((1 - \delta) S_i + H_i \epsilon_l - s_n ((1 - \delta) S_i + H_i \epsilon_l) \big) \leq 0, \tag{24}$$

$$- (1 + r) \Psi' (H_i) + \sum_{l=1}^{L} w_l \epsilon_l P \big((1 - \delta) S_i + H_i \epsilon_l - s_n ((1 - \delta) S_i + H_i \epsilon_l) \big) = 0. \tag{25}$$

14

3. Update the approximation by finding the J-breakpoint spline $s_{n+1}(\cdot)$ that solves the system

$$s_{n+1}(A_i) = S_i \quad \text{for } i = 1, \ldots, J. \tag{26}$$

4. If $\|s_{n+1}(A) - s_n(A)\|_2 \geq 10^{-7}$ then increment n to $n+1$ and go to step 2.

The storage rule approximation method can also be solved by fixed-point iteration, as suggested in Judd (1998) or by EGM.

3.6 Perturbation methods

For the storage model, perturbation methods have been barely considered. Miranda (1997) tries linearisation of the decision rules, but not higher order approximations. In other settings, particularly dynamic stochastic general equilibrium modelling, perturbation methods are widely used because they allow one to work with a large number of state variables and are easy to implement through user-friendly packages such as *Dynare*. The method consists of approximating the dynamic problem around the steady state by its Taylor development, which requires a smooth problem.

The storage problem is normally not amenable to such methods because of the non-negativity constraint. The introduction of convenience yield smooths the model and allows the use of perturbation. This transformation to a smooth problem from a problem with inequality constraint is not uncommon in numerical methods, where it is called the penalty function or barrier method (Judd, 1998, p. 123–125). It is used in the macroeconomic context by Preston and Roca (2007) and Kim et al. (2010) to smooth the borrowing constraint in a consumption/savings decision.

The storage model with convenience yield is approximated around its steady state from the first to the fifth order by using the software *Dynare++* v. 4.1.0.

4 Numerical results

A benchmark solution is used to generate the true distribution of the state variable. The benchmark solution is the PEA with 5,000 nodes. The benchmark simulation is defined over 10,000 periods. The same shocks are used for both versions of the model.

4.1 Euler equation error

Since the storage model has no analytical solution, we cannot compare the approximate solutions obtained from the various methods with an exact benchmark. For this case, Judd (1992) designed a bounded rationality measure, the Euler equation error, to measure by how much solutions violate the optimising conditions. This measure defines the resources wasted by using approximated decision rules instead of true ones. In what follows, we derive the definitions of the two measures of the Euler equation error for the storage model. This differs from the literature on numerical methods in that the measure has to account for the existence of two regimes in the Euler equation.

Expressed in terms of the policy functions, $S(A)$ and $H(A)$, the storage Euler equation is

$$S(A_t) \geq 0 \quad \perp \quad \frac{1-\delta}{1+r} \mathrm{E}_t \left[P\big(((1-\delta)S(A_t) + H(A_t)\epsilon_{t+1} - S((1-\delta)S(A_t) + H(A_t)\epsilon_{t+1}) \big) \right]$$
$$- P\big(A_t - S(A_t)\big) - \Phi'\big(S(A_t)\big) \leq 0. \quad (27)$$

To simplify the notations in the following equations, we use $A_{t+1} = (1-\delta)S(A_t) + H(A_t)\epsilon_{t+1}$. We need to distinguish two situations. First, when storage costs include a convenience yield, the Euler equation holds exactly without a complementarity condition. This leads to the following Euler equation

$$P\big(A_t - S(A_t)\big) = \frac{1-\delta}{1+r} \mathrm{E}_t \left[P\big(A_{t+1} - S(A_{t+1})\big) \right] - \Phi'\big(S(A_t)\big). \quad (28)$$

As the decision rules are approximations, this equation will not hold exactly for all availabilities. We can define the Euler equation error function with convenience yield as

$$EE_S(A_t) = 1 - \frac{P^{-1}\left(\frac{1-\delta}{1+r} \mathrm{E}_t \left[P\big(A_{t+1} - S(A_{t+1})\big) \right] - \Phi'\big(S(A_t)\big) \right)}{A_t - S(A_t)}. \quad (29)$$

Second, for the case without convenience yield, we can rewrite the Euler equation as

$$P\big(A_t - S(A_t)\big) = \max\left\{ P(A_t), \frac{1-\delta}{1+r} \mathrm{E}_t \left[P\big(A_{t+1} - S(A_{t+1})\big) \right] - \Phi'\big(S(A_t)\big) \right\}. \quad (30)$$

This formulation displays the two regimes. For low next-period expected price, there is no storage, and the current price is given simply by the inverse demand function applied to current availabilities. For positive storage, current price is linked to next-period price. As the decision rules are approximations, this equation does not hold exactly for all availabilities. We can define the Euler equation error as

$$EE_S(A_t) = 1 - \frac{P^{-1}\left(\max\left\{ P(A_t), \frac{1-\delta}{1+r} \mathrm{E}_t \left[P\big(A_{t+1} - S(A_{t+1})\big) \right] - \Phi'\big(S(A_t)\big) \right\} \right)}{A_t - S(A_t)}. \quad (31)$$

As $D_t = A_t - S(A_t)$, this error can be interpreted as a unit-free representation of the amount of commodity consumption that should adjust to make the Euler equation hold exactly.

The error for the production Euler equation is derived in the same way:

$$EE_H(A_t) = 1 - \frac{(\Psi')^{-1}\left(\frac{1}{1+r} \mathrm{E}_t \left[P\big(A_{t+1} - S(A_{t+1})\big)\epsilon_{t+1} \right] \right)}{H(A_t)}. \quad (32)$$

It is equivalent to the level of production that should be adjusted to make the Euler equation hold.

We consider two measures of the Euler equation error, its maximum and its average. Its maximum is taken over an interval of availability defined by the first and the last percentiles of the benchmark simulation. The maximum error is useful to identify the algorithm limits. Average error is the error arising from the use of decision rules for a sequence of decisions. We calculate it by averaging absolute Euler equation errors over the benchmark simulation. To facilitate their reading, the errors are presented in a base-10 logarithm. A result of -3 for $\log_{10}|EE_S|$ should be read as a \$1 error in every \$1,000 of commodity consumption; -6, \$1 error for \$1,000,000 of consumption.

4.2 Limitations of perturbation methods

Because perturbation methods perform significantly worse than other methods, we give them special treatment in order to understand why they are of limited interest in this case whereas it is the method of choice in several other settings.[7] Figure 2 presents the storage rules obtained by perturbation methods, at various orders, and compares them with the true one, obtained using the benchmark method. A figure for the production rule would be similar, so is not displayed here.
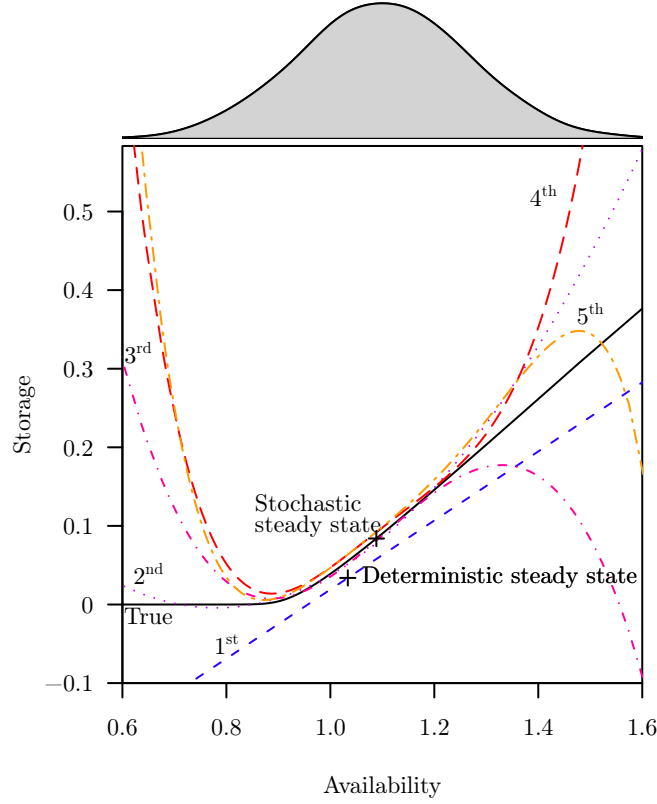
Figure 2. Storage rules obtained with perturbation methods up to the 5th order. Density of availability from the benchmark model above the plot. The "True" storage rule is generated by the benchmark method. Deterministic steady state is the state reached in the absence of shocks and ignoring future shocks. Stochastic steady state is the fixed point of the stochastic problem in the absence of shocks, i.e., the state reached in the absence of shocks but accounting for the likelihood of future shocks.

Given the nonlinearity of decision rules, a first order perturbation could not be expected to yield an appropriate approximation. For low availability, the first order storage rule proposes negative storage and presents a marginal propensity to store current availability much lower than the true rule. These are the classical properties of the certainty equivalent solution to the consumption/saving model under uncertainty (Zeldes, 1989). The certainty equivalent means that the variance of future shocks does not matter. The first order storage rule passes by the deterministic steady state, which is not part of the true storage rule,

---

[7]They are also treated apart because the storage rules found by perturbation are negative for some availability. The Euler equation error, used to assess the precision of the other methods, is not defined for negative storage, because of the logarithm in $\Phi'(\cdot)$.

because, under certainty, there is no speculative motive for storing, only a convenience yield. Beyond first order perturbation, the approximated decision rules follow the true one quite closely for availability between 0.95 and 1.28, which corresponds to 70% of the distribution. Below or beyond this, approximations are very poor. Odd order approximations present negative storage for high availability. And, despite the fact that Blanchard-Kahn conditions are satisfied, all solutions are unstable. They are locally stable around the steady state, but lose stability farther away from it. The best precision is achieved with second order approximation. This degradation of precision with perturbation order was identified by Feigenbaum (2005) for the consumption problem and by Aruoba et al. (2006) for one parameterisation of the stochastic growth model.

There are two problems related to perturbation methods applied to the competitive storage model. Even smoothed by a penalty function, the storage model remains highly non-linear, which calls for high order perturbation. As perturbation methods are in essence local methods, they should behave correctly around the steady state, which is what we observe. But the typical shocks in agricultural production drive the behaviour in regions distant from the steady state. This work confirms some recent evidence that higher-order perturbation may not be well-behaved. Feigenbaum (2005) and den Haan and de Wind (2009) note that a drawback of perturbation methods is the lack of control over the radius of convergence. With projection methods, however, the space of convergence is defined arbitrarily. In this case, we see that the radius of convergence covers only a small part of the availability distribution. The use of a logarithm as the barrier function may have increased the lack of convergence of the Taylor development, because its radius of convergence is limited. But den Haan and de Wind (2009) show that this problem arises also with a well behaved barrier, such as an exponential.

4.3 Comparison of methods precision

Before comparing methods, it is useful to assess the relation between storage and production Euler equation errors, and the precision achievable from the most precise method. Table 3 presents detailed results for the PEA—proved below to be the most accurate algorithm—when the number of breakpoints in the spline approximation varies.

The production Euler equation error is systematically lower than the storage Euler equation error. However, they are close and vary in the same way. Augmenting the number of nodes improves the precision slowly for the model with non-negativity constraint. The general shape is well approximated at a low order (for information, the papers of Williams and Wright that have derived most of theoretical properties of the storage model used an order-5 polynomial). With convenience yield, the expected price approximation improves faster with the number of breakpoints. Based on these first results, we decide to restrict presentation of the subsequent results to the storage Euler equation error and to a 20-breakpoint approximation.

Table 4 presents the precision achieved by the various methods. The PEA is consistently more precise than other methods. Value function iteration, current price approximation and storage rule approximation all show similar precision, although the endogenous grid method is rather less precise.

Figure 3 presents the evolution of the storage Euler equation error with availability. The curves are hump-shaped, because the precision is very high at breakpoints (the principle of the collocation method is to find a solution where the function satisfies all constraints at the breakpoints) and decreases between them. The lowest precision appears around the kink of no storage.

Figure 4 compares methods at various orders of approximation. The logarithm of the Euler equation error decreases linearly with the logarithm of the approximation order. The mean slope is -2, which means a

Table 3. The effect of approximation order on precision ($\log_{10}|EE|$) for the parameterised expectations algorithm

| Order	Storage Euler equation		Production Euler equation	
	Max error	Average error	Max error	Average error
With non-negativity constraint				
5	−2.20	−2.67	−2.45	−2.93
10	−2.69	−3.47	−2.93	−3.74
20	−2.92	−3.95	−3.15	−4.21
50	−3.22	−4.53	−3.45	−4.78
100	−3.56	−5.16	−3.79	−5.41
200	−4.13	−5.70	−4.36	−5.95
1000	−4.92	−7.23	−5.15	−7.49
With convenience yield				
5	−2.41	−2.83	−2.78	−3.10
10	−3.68	−4.31	−4.13	−4.68
20	−4.66	−5.18	−5.08	−5.48
50	−6.42	−7.15	−6.63	−7.42
100	−7.83	−8.48	−8.04	−8.74
200	−9.08	−9.73	−9.29	−9.99
1000	−9.93	−11.43	−11.00	−11.69

Table 4. Storage Euler equation error ($\log_{10}|EE_S|$) for various methods with a 20-breakpoint spline approximation

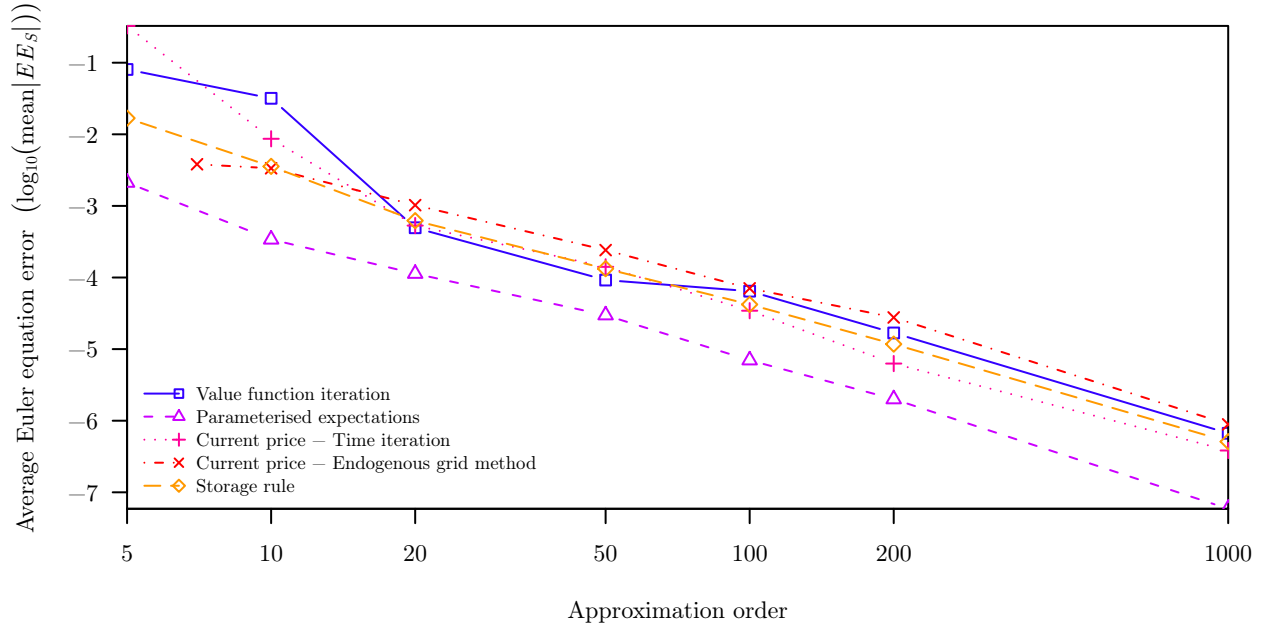| Method	With non-negativity constraint		With convenience yield	
	Max error	Average error	Max error	Average error
Value function iteration	−2.57	−3.31	−2.94	−3.24
Parameterised expectations	−2.92	−3.95	−4.66	−5.18
Current price approximation				
Time iteration	−2.69	−3.27	−3.30	−3.61
Endogenous grid method	−2.34	−2.99	−2.82	−3.33
Storage rule	−2.43	−3.21	−3.49	−3.86

10-fold increase in the number of breakpoints leads to an error decrease of 2. If at 20 breakpoints, the average error is $1 in every $1,000; at 200 breakpoints it will be $1 in every $100,000. PEA is consistently the most precise method. At very low orders, the current price approximation solved by time iteration has a very poor precision of $1 error for every $10 decisions. The exponential form of the price function (Figure 1b) makes it difficult to approximate with only 5 breakpoints. At the breakpoints, approximation coincides with the price function, but between them it oscillates widely.

4.4   Simulating the model

We know address an issue often neglected in computational economics, which is how the model should be simulated. Since dynamic models are often solved by methods requiring an approximation of the decision
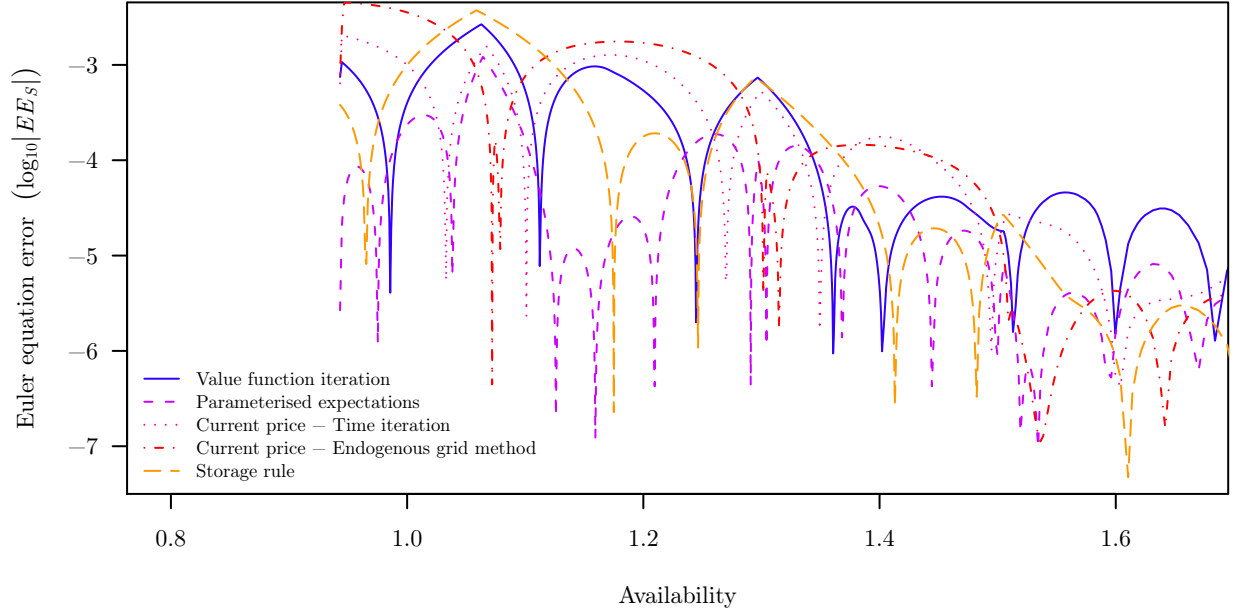
Figure 3. Storage Euler equation errors for the model with non-negativity constraint. Results obtained using a 20-breakpoint spline approximation. The x-axis represents availability from the 1st to the 99th percentile. Below the kink point of the storage rule the error is null, which means that approximation errors have no economic implication because storage does not occur.

Figure 4. Average storage Euler equation error for the model with non-negativity constraint, for all methods, at different approximation orders. No solution was found for the EGM with 5 breakpoints, thus, the EGM curve starts at 7 breakpoints.

rules (e.g., in the case of perturbation and projection methods in Aruoba et al., 2006), simulating a model appears intuitively as a recursive application of the approximated decision rules. This intuition is not always correct as demonstrate Kim et al. (2008). They show that the recursive application of a quadratic approximation creates higher order terms that do not increase accuracy, but can lead to explosive time paths. They propose to prune out the extraneous high-order terms to prevent explosiveness. In the storage model, the problem arises from the fact that the most precise method, PEA, does not produce approximated decision rules.

Wright and Williams (1984), who first proposed the PEA, use the same method for solving and simulating the model. They approximate the conditional expectations terms with respect to storage using an algorithm similar to the one presented in Section 3.5.2. For the simulation, they plug these approximations into the first-order conditions (4) and (6), which they solve for storage and planned production. So each simulation requires the resolution of a nonlinear, complementarity problem.

There may be more convenient solutions. Finding the rational expectations equilibrium and simulating the model are different things; the most appropriate method for finding the equilibrium might not be the most appropriate for simulating the model. All the methods described above for finding the rational expectations equilibrium can be used to simulate the model, but all imply solving a nonlinear complementarity problem. Another approach would be to build approximations of the decision rules, $S(A)$ and $H(A)$, and to use them to simulate the model. This approach is at least 15 times faster than simulation by solving a system of equations.

It is this solution that is most commonly adopted for dynamic models. Some clarification is, however, needed to achieve optimal use of decision rules approximation in simulations. First, since decision rules are less smooth than other of the model's functions (e.g., the price function or conditional expectations), they need to be approximated using a higher number of free parameters. Second, simulating the model by solving the first-order conditions provides greater precision than using the approximated policy functions directly. For example, if the model is solved by approximating the storage rule, it is possible to use this approximation to simulate the model without more equations solving, although this is not advisable. A decision rule approximated by a 20-breakpoint spline generates maximum and average storage Euler equation errors of -1.79 and -2.69 (see Table 5 below), a level of precision significantly lower than what is achieved when using this approximation to solve the first-order conditions (see Table 4 for comparison). This imprecise storage rule allows simulation of the model with reasonable precision when used inside the expectations terms, as in equations (24)–(25). So, when the rational expectations equilibrium is found with the desired precision, the model should be simulated by solving the equations on a grid to define the two approximated decision rules, $S(A)$ and $H(A)$, which should be defined with a higher number of free parameters.

The above is presented more clearly in Table 5, which shows the Euler equation errors achieved by approximating the benchmark decision rules with splines of various orders. For the same order of approximation, the precision is inferior to what is achieved by explicitly solving the first-order conditions as shown in tables 3 and 4. This applies especially to the maximum error, which decreases very slowly with the number of breakpoints because of the difficulty involved in approximating the kinks.

In the econometric work on the storage model, simulations are usually based on approximated decision rules (e.g., Deaton and Laroque, 1995, 1996, Michaelides and Ng, 2000, Osborne, 2004).[8] At small orders, this method lacks precision, and great attention is needed to the number of breakpoints used for spline

---

[8]In fact, these studies use a current price approximation in a model without supply reaction. Without supply reaction, the current price function is sufficient to simulate the model.

Table 5. The effect of approximation order on precision ($\log_{10}|EE|$) for a spline approximation of the true decision rules, for the model with non-negativity constraint.

Order	Storage Euler equation		Production Euler equation	
	Max error	Average error	Max error	Average error
10	−1.49	−2.15	−2.78	−3.05
20	−1.79	−2.69	−2.89	−3.33
50	−2.19	−3.51	−3.53	−4.10
100	−2.49	−4.20	−3.77	−4.76
200	−2.89	−4.86	−4.03	−5.39
1000	−4.10	−6.56	−4.88	−6.87

Note: The decision rules from the benchmark solution are approximated at various orders by shape-preserving splines (function `pchip` in Matlab).

approximation. Deaton and Laroque (1995, 1996) use a 20-breakpoint cubic spline. This approximation is criticised by Cafiero et al. (2010), who show that by using more breakpoints they obtain a better estimation, restoring the empirical relevance of the storage model criticised by previous studies. Indeed, as Table 5 demonstrates, for high orders of approximation, the precision achieved is at an acceptable level.

4.5 Implementation and computing time

This subsection focuses on the time needed to solve the model. As all the existing econometric work uses the model without supply reaction, we include a comparison of computing time over this more restricted model. Comparison of its Euler equation errors would be less interesting since they are very close to those for the model with supply reaction.

The results reported are for a 2.6 GHz Intel PC running Windows Vista and Matlab R2010a. We would expect a faster solve for a model with inelastic supply, because the model is simpler. This is not the case for all methods (see Table 6). While the model has one less equation, the function being approximated is farther away from the first guess and so requires more iterations. The kink point of the storage rule appears at a lower availability level when supply elasticity decreases. For a low availability level, it is better to increase future supply by increasing production rather than by increasing stock, because building stocks are expensive. And the lower the kink, the higher will be the number of iterations needed. This higher number of iterations is not compensated for by the fact that iterations are simpler except for VFI and EGM. In the case of EGM, solving the model without supply reaction requires only simple calculations and there is no nonlinear problem to solve. It is very fast. It is the case where this method shows all its interest.

The value function iteration, as expected, is the most time-consuming method. It converges very slowly (several hundred iterations) to rational expectations. With the exception of EGM in the case without elastic supply, current price approximation with time iteration is the fastest method. The most precise method, parameterised expectations algorithm, is the slowest for a given approximation order (20 breakpoints). However, this is not to imply that it is slower than other methods for a given precision. Actually, for a specified precision PEA is the fastest method.

In terms of speed, EGM outperforms other methods by a factor of at least 3 in the case of inelastic supply. The speed of this method makes it a particularly suitable alternative to Deaton and Laroque's fixed point algorithm, for econometric work. When extended to a situation with elastic supply, EGM loses its interest.

Table 6. Time (seconds) to solve the rational expectations equilibrium with a 20-breakpoint spline approximation

Method	With non-negativity constraint		With convenience yield
	Inelastic supply[a]	Elastic supply	
Value function iteration	3.47	3.51	3.80
Parameterised expectations	0.93	0.57	0.47
Current price approximation			
Time iteration	0.56	0.48	0.20
Endogenous grid method	0.16	0.57	0.32
Storage rule	0.68	0.59	0.25

[a] For inelastic supply, Equation (4) is excluded of the model and planned production is fixed at 1, its deterministic steady state.

It requires a nonlinear solve and so loses its high speed, for a precision comparable to other methods. EGM is only interesting in models with one control variable.

The model with convenience yield is smooth. Its functions are simpler to approximate than those of the model with non-negativity constraint. Hence, it takes less time to solve it; convergence is achieved with fewer iterations.

The time needed to achieve the rational expectations equilibrium is only a small part of the time required for the whole model. It takes much longer to program it. With regard to this criterion, the convenience yield hypothesis is less convenient. It is faster and more precise than the regime-switching model, but the logarithm creates numerical difficulties. It is interesting in that it does not allow negative storage by decreasing to minus infinity when storage tends to zero, but it is necessary to use a nonlinear solver which precludes exploration of negative values and handles the interactions of very low and very high values, which tend to create scaling problems. EGM is very simple to implement in the model with inelastic supply, because it boils down to iterations with arithmetic operations. With elastic supply, it requires a nonlinear solve, so it is comparable to other methods. The various projection methods present the same difficulty of implementation, PEA being perhaps less intuitive because of its two nested loops, the first on storage and the second on availability.

5 Conclusion

This paper compared several solution methods for the competitive storage model. Three methods are popular in the related literature: value function iteration, parameterised expectations algorithm, and current price approximation. We introduced three other methods, borrowed from the literature on numerical analysis: perturbation, endogenous grid, and decision rules approximation. Perturbation methods proved to be no use in this context, because the storage model, smoothed by the convenience yield hypothesis, is too nonlinear, and the disturbances drive approximation out of its radius of convergence. The endogenous grid method is a good alternative to Deaton and Laroque's (1992) algorithm for a fast solve of the model for estimation purpose. Its good performance, however, is limited to the case without supply reaction. One method stands out: the parameterised expectations algorithm, which is very precise, even at small approximation orders, because it approximates smooth and close to linear functions. Although, in this case, it is sufficient to use low order approximants to find the rational expectations equilibrium, the simulation of the model with decision

rules requires the use of more free parameters in the approximation to secure a good precision. These results could be generalised to the consumption/saving problem under uncertainty, where the equations are close to the storage model.

Because of modern computational power, any method, except perturbation, would provide a satisfying solution to the competitive storage model, and in reasonable time. Numerical challenges arise when the model is extended to include several state variables, such as in storage-trade interaction or the multiple commodities model. In these settings, the advantages of the parameterised expectations algorithm over other methods may be even greater, because it is more precise, even for a small number of nodes. Indeed the "curse of dimensionality", i.e., the exponential rise in computing time when problem dimensions increase, is all the more binding such that a solution method requires a larger number of nodes to be precise. Other methods require 50-node splines to reach the same precision as PEA with 10 nodes. In a naive extension to two dimensions, for the same precision, PEA would require 100 nodes, and other methods 2,500 nodes.

References

Aruoba, S. B., Fernández-Villaverde, J., and Rubio-Ramírez, J. F. (2006). Comparing solution methods for dynamic equilibrium economies. *Journal of Economic Dynamics and Control*, 30(12), 2477–2508.

Barillas, F. and Fernández-Villaverde, J. (2007). A generalization of the endogenous grid method. *Journal of Economic Dynamics and Control*, 31(8), 2698–2712.

Barsky, R. B., Mankiw, N. G., and Zeldes, S. P. (1986). Ricardian consumers with Keynesian propensities. *The American Economic Review*, 76(4), 676–691.

Cafiero, C., Bobenrieth, E. S. A., Bobenrieth, J. R. A., and Wright, B. D. (2010). The empirical relevance of the competitive storage model. *Journal of Econometrics*, In press.

Cafiero, C. and Wright, B. D. (2006). Is the storage model a 'closed' empirical issue? The empirical ability of the storage model to explain price dynamics. In A. Sarris and D. Hallam (Eds.), *Agricultural Commodity Markets and Trade. New Approaches to Analyzing Market Structure and Instability* chapter 4, (pp. 89–114). Northampton: FAO/Edward Elgar.

Carroll, C. D. (2001). A theory of the consumption function, with and without liquidity constraints. *The Journal of Economic Perspectives*, 15(3), 23–45.

Carroll, C. D. (2006). The method of endogenous gridpoints for solving dynamic stochastic optimization problems. *Economics Letters*, 91(3), 312–320.

Carter, C. A. and Revoredo Giha, C. L. (2007). The working curve and commodity storage under backwardation. *American Journal of Agricultural Economics*, 89(4), 864–872.

Chambers, M. J. and Bailey, R. E. (1996). A theory of commodity price fluctuations. *The Journal of Political Economy*, 104(5), 924–957.

Christiano, L. J. and Fisher, J. D. M. (2000). Algorithms for solving dynamic models with occasionally binding constraints. *Journal of Economic Dynamics and Control*, 24(8), 1179–1232.

Deaton, A. (1991). Saving and liquidity constraints. *Econometrica*, 59(5), 1221–1248.

Deaton, A. and Laroque, G. (1992). On the behaviour of commodity prices. *Review of Economic Studies*, 59(1), 1–23.

Deaton, A. and Laroque, G. (1995). Estimating a nonlinear rational expectations commodity price model with unobservable state variables. *Journal of Applied Econometrics*, 10, S9–S40.

Deaton, A. and Laroque, G. (1996). Competitive storage and commodity price dynamics. *The Journal of Political Economy*, 104(5), 896–923.

den Haan, W. J. and de Wind, J. (2009). *How well-behaved are higher-order perturbation solutions?* DNB Working Paper 240, De Nederlandsche Bank.

den Haan, W. J. and Marcet, A. (1990). Solving the stochastic growth model by parameterizing expectations. *Journal of Business & Economic Statistics*, 8(1), 31–34.

Feigenbaum, J. (2005). Second-, third-, and higher-order consumption functions: a precautionary tale. *Journal of Economic Dynamics and Control*, 29(8), 1385–1425.

Gardner, B. L. (1979). *Optimal Stockpiling of Grain*. Lexington, Mass.: Lexington Books.

Gustafson, R. (1958). *Carryover Levels for Grains: A Method for Determining Amounts that are Optimal Under Specified Conditions*. Technical Bulletin 1178, US Dept. of Agriculture.

Heer, B. and Maußner, A. (2008). Computation of business cycle models: A comparison of numerical methods. *Macroeconomic Dynamics*, 12(05), 641–663.

Judd, K. L. (1992). Projection methods for solving aggregate growth models. *Journal of Economic Theory*, 58(2), 410–452.

Judd, K. L. (1998). *Numerical Methods in Economics*. Cambridge: MIT Press.

Judd, K. L., Maliar, L., and Maliar, S. (2009). *Numerically Stable Stochastic Simulation Approaches for Solving Dynamic Economic Models*. Working Paper 15296, National Bureau of Economic Research.

Kaldor, N. (1939). Speculation and economic stability. *The Review of Economic Studies*, 7(1), 1–27.

Kim, J., Kim, S. H., Schaumburg, E., and Sims, C. A. (2008). Calculating and using second-order accurate solutions of discrete time dynamic equilibrium models. *Journal of Economic Dynamics and Control*, 32(11), 3397–3414.

Kim, S. H., Kollmann, R., and Kim, J. (2010). Solving the incomplete market model with aggregate uncertainty using a perturbation method. *Journal of Economic Dynamics and Control*, 34(1), 50–58.

Lence, S. H. and Hayes, D. J. (2002). U.S. farm policy and the volatility of commodity prices and farm revenues. *American Journal of Agricultural Economics*, 84(2), 335–351.

Michaelides, A. and Ng, S. (2000). Estimating the rational expectations model of speculative storage: A Monte Carlo comparison of three simulation estimators. *Journal of Econometrics*, 96(2), 231–266.

Miranda, M. J. (1997). Numerical strategies for solving the nonlinear rational expectations commodity market model. *Computational Economics*, 11(1–2), 71–87.

Miranda, M. J. and Fackler, P. (2002). *Applied Computational Economics and Finance*. Cambridge: MIT Press.

Miranda, M. J. and Glauber, J. W. (1995). Solving stochastic models of competitive storage and trade by Chebychev collocation methods. *Agricultural and Resource Economics Review*, 24(1), 70–77.

Miranda, M. J. and Helmberger, P. (1988). The effects of commodity price stabilization programs. *The American Economic Review*, 78(1), 46–58.

Newbery, D. M. G. and Stiglitz, J. E. (1982). Optimal commodity stock-piling rules. *Oxford Economic Papers*, 34(3), 403–427.

Osborne, T. (2004). Market news in commodity price theory: Application to the Ethiopian grain market. *The Review of Economic Studies*, 71(1), 133–164.

Park, A. (2006). Risk and household grain management in developing countries. *Economic Journal*, 116(514),

1088–1115.

Plato, G. and Gordon, D. (1983). Dynamic programming and the economics of optimal grain storage. *Journal of Agricultural Economics Research*, 35(1), 10–22.

Preston, B. and Roca, M. (2007). *Incomplete Markets, Heterogeneity and Macroeconomic Dynamics*. Working Paper 13260, National Bureau of Economic Research.

Rust, J. (1996). Numerical dynamic programming in economics. In H. M. Amman, D. A. Kendrick, and J. Rust (Eds.), *Handbook of Computational Economics*, volume 1 chapter 14, (pp. 619–729). Amsterdam: Elsevier.

Scheinkman, J. A. and Schechtman, J. (1983). A simple competitive model with production and storage. *The Review of Economic Studies*, 50(3), 427–441.

Williams, J. C. and Wright, B. D. (1991). *Storage and Commodity Markets*. New York: Cambridge University Press.

Working, H. (1948). Theory of the inverse carrying charge in futures markets. *Journal of Farm Economics*, 30(1), 1–28.

Wright, B. D. (2001). Storage and price stabilization. In B. L. Gardner and G. C. Rausser (Eds.), *Marketing, Distribution and Consumers*, volume 1, part 2 of *Handbook of Agricultural Economics* chapter 14, (pp. 817–861). Amsterdam: Elsevier.

Wright, B. D. and Williams, J. C. (1982). The economic role of commodity storage. *The Economic Journal*, 92(367), 596–614.

Wright, B. D. and Williams, J. C. (1984). The welfare effects of the introduction of storage. *The Quarterly Journal of Economics*, 99(1), 169–192.

Zeldes, S. P. (1989). Optimal consumption with stochastic income: Deviations from certainty equivalence. *The Quarterly Journal of Economics*, 104(2), 275–298.