MULTI-STEP PERTURBATION SOLUTION OF NONLINEAR RATIONAL EXPECTATIONS MODELS*

Peter A. Zadrozny**
Bureau of Labor Statistics
2 Massachusetts Ave., NE
Washington, DC 20212
e-mail: zadrozny.peter@bls.gov

Baoline Chen
Bureau Of Economic Analysis
1441 L Street, NW
Washington, DC 20230
e-mail: baoline.chen@bea.gov

February 14, 2006

---

*The paper represents the authors' views and does not represent any official positions of the Bureau of Labor Statistics or the Bureau of Economic Analysis.

**Affiliated as Research Fellow with the Center for Financial Studies (CFS), J.W. Goethe University, Frankfurt, Germany, and with the Center for Economic Studies and Ifo Institute for Economic Research (CESifo), Munich, Germany.

ABSTRACT

Recently, perturbation has received attention as a numerical method for computing an approximate solution of a nonlinear dynamic stochastic model, which we call a nonlinear rational expectations (NLRE) model. To date perturbation methods have been described and applied as single-step perturbation (SSP). If a solution of an NLRE model is a function $\varphi(x)$ of vector x, then, SSP aims to compute a kth-order Taylor approximation of $\varphi(x)$, centered at x_0. In classical SSP, where x_0 is a nonstochastic steady state of the dynamical system, a kth-order approximation is accurate on the order of $||\Delta x||^{k+1}$, where $\Delta x = x - x_0$ and $||\cdot||$ is a vector norm. Thus, for given k and computed x_0, classical SSP is accurate only locally, near x_0. SSP's accuracy can be improved only by increasing k, which beyond small values results in large computing costs, especially for deriving kth-order analytical derivatives of the model's equations. So far, research has not fully solved the problem in SSP of maintaining any desired accuracy while freeing x_0 from the nonstochastic steady state, so that, for given k, SSP can be arbitrarily accurate for any Δx. Multi-step perturbation (MSP) fully solves this problem and, thus, globalizes SSP. In SSP, we approximate $\varphi(x)$ with a single Taylor approximation centered at x_0 and, thus, effectively move from x_0 to x in one step. In MSP, we move in a straight line from x_0 to x in h steps of equal length. At each step, we approximate φ at the x at the end of the step with a Taylor approximation centered at the x at the beginning of the step. After h steps and Taylor approximations, we obtain an approximation of $\varphi(x)$ which is accurate on the order of h^{-k}. Thus, although in MSP we also set x_0 to a nonstochastic steady state, unlike in SSP, we can achieve any desired accuracy for any x_0, x, and k, simply by using sufficiently many steps. Thus, we free the accuracy from dependence on k and $||\Delta x||$ and effectively globalize SSP. Whereas increasing k requires new derivations and programming, increasing h requires only passing more times through an already programmed loop, typically at only moderately more computing time. In the paper, we derive an MSP algorithm in standard linear-algebraic notation, for a 4th-order approximation of a general NLRE model, and illustrate the algorithm and its accuracy by applying it to a stochastic one-sector optimal growth model.

MULTI-STEP PERTURBATION SOLUTION OF

NONLINEAR RATIONAL EXPECTATIONS MODELS

Baoline Chen
Bureau of Economic Analysis
Washington, DC, USA
Email: baoline.chen@bea.gov

and

Peter A. Zadrozny
Bureau of Labor Statistics
Washington, DC, USA
Email: zadrozny.peter@bls.gov*

NBER/NSF/CFS Workshop on Time-Series Analysis
University of Munich, Munich, Germany

September 20-21, 2005.

---

* This paper represents the authors' views and not necessarily any official positions of BLS or BEA.

Introduction

- **This paper is motivated by the desire to accurately compute the solution of nonlinear rational expectations models.**

- **We describe and illustrate the Multiple Step Perturbation (MSP) method for quickly and accurately computing the 4th-order polynomial (Taylor series) approximation of the solution of nonlinear rational expectations models.**

- **Plan of the presentation:**

1. **Nonlinear Rational Expectations Models (NLREM).**
2. **An Optimal Growth Example of NLREM.**
3. **Single-Step vs. Multi-Step Perturbation (SSP vs. MSP).**
4. **2nd-Order MSP Solution Equations.**
5. **MSP solution of the Optimal Growth Model.**
6. **Conclusion.**

1. Nonlinear rational expectations models (NLREM).

- **First statement of NLREM:**

(1) $E_t c(y_{t+1}, y_t, y_{t-1}, \varepsilon_t, \varepsilon_{t+1}) = 0_{n\times 1}$;

y_t = $n\times 1$ vector of endogenous variables;

ε_t = $n\times 1$ vector of exogenous disturbances;

c = nonlinear function which maps R^{5nx1} to R^{nx1} and is k-times differentiable;

E_t = expectation w.r.t. probability distribution of ε_{t+1}, conditional on period t information, specified in terms of ε's moments;

$s_i = E_t(\varepsilon_{t+1} \otimes \ldots \otimes \varepsilon_{t+1})$ = ith moment of ε_{t+1}, with k-1 Kronecker products.

- **Second statement of NLREM:**

  - **Want a solution in the form of a feedback decision rule:**

(2) $\mathbf{y}_t = \phi(\mathbf{x}_t)$,

ϕ = **nonlinear solution function which maps $R^{2n\times1}$ to R^n and is k-times differentiable;**

$\mathbf{x}_t = (\mathbf{y}_{t-1}^T, \varepsilon_t^T)^T$ = **2n×1 state vector;**

  - **Let $\theta\eta_{t+1} = \varepsilon_{t+1}$, where $0 \le \theta \le 1$ scales uncertainty, drop t everywhere, and write $E_t c(\cdot) = 0_{n\times1}$ as**

(3) $Ec(\phi(\phi(\mathbf{x}), \theta\eta), \phi(\mathbf{x}), \mathbf{x}, \theta\eta) = 0_{nx1}$.

- **Third statement of NLREM:**

ϕ depends on uncertainty scale θ, so ϕ is a function of θ and (3) becomes

$$(4) \quad Ec(\phi(\phi(\mathbf{x}, \theta), \theta\eta, \theta), \phi(\mathbf{x}, \theta), \mathbf{x}, \theta\eta) = 0_{nx1}.$$

- **What is a solution?**

A function $\phi(\mathbf{x},\theta)$ that satisfies (4) for $\theta = 1$ at the given value of $\mathbf{x}$.

- **Suppressing θ, we approximate ϕ as a kth-order polynomial:**

$$(5) \quad \hat{\phi}(\mathbf{x}) = \phi_0 + \nabla\phi_0\Delta\mathbf{x} + (1/2)(\Delta\mathbf{x}^T \otimes I_n)\nabla^2\phi_0 + \ldots$$

$$+ (1/k!)(\Delta\mathbf{x}^T \otimes \cdots \otimes \Delta\mathbf{x}^T \otimes I_n)\nabla^k\phi_0\Delta\mathbf{x},$$

where $\Delta\mathbf{x} = \mathbf{x}_h - \mathbf{x}_0$, $\phi_0 = \phi(\mathbf{x}_0)$,

$\nabla\phi_0, \ldots, \nabla^k\phi_0$ = matrices of 1st- to kth-order derivatives of ϕ at $\mathbf{x}_0$,

$\mathbf{x}_0$ = nonstochastic steady state.

3. Optimal growth example of NLREM.

- **Basic equations of the model:**

(6) $u(c_t) = (1-\gamma)^{-1}c_t^{1-\gamma}$ (utility function),

(7) $f(k_{t-1},\tau_t) = \tau_t k_{t-1}^{\alpha}$ (production function),

(8) $\tau_t = \tau_{t-1}^{\rho}\exp(\varepsilon_{\tau,t})$ (technology law of motion),

(9) $k_t = (1-\delta)k_{t-1} + \tau_t k_{t-1}^{\alpha} - c_t + \varepsilon_{k,t}$, (capital law of motion),

$\varepsilon_{t+1} = (\varepsilon_{k,t+1},\ \varepsilon_{\theta,t+1})^T \sim N(0,\Sigma_\varepsilon)$,

$\gamma < 1,\ 0 < \alpha,\ \delta,\ \rho < 1.$

- **Objective: maximize expected present value of utility:**

(10) $\max E_t\sum_{i=0}^{\infty}\beta^i u(c_{t+i})$ w.r.t. $\{c_{t+i}\}_{i=0}^{\infty}$, for $0 < \beta < 1$.

- **Optimal feedback decision rule**:

Eliminate c, so that k is the decision variable in y:

$$(11) \quad \mathbf{y}_t = \phi(\mathbf{x}_t),$$

where $\mathbf{y}_t = (k_t, \tau_t)^T$, $\mathbf{x}_t = (\mathbf{y}_{t-1}^T, \varepsilon_t)^T$, $\varepsilon_t = (\varepsilon_{kt}, \varepsilon_{\tau t})^T$.

- **Model's structural equation:**

$$(12) \quad Ec(\cdot) = E\begin{bmatrix} c_1 \\ \\ c_2 \end{bmatrix} = \begin{bmatrix} \text{Euler equation} \\ \\ \text{Technology l.o.m.} \end{bmatrix} = E\begin{bmatrix} \beta u'_{t+1} f_{k,t+1} - u'_t \\ \\ \tau_t - \tau_{t-1}^{\rho} \exp(\varepsilon_{\tau,t}) \end{bmatrix} = 0_{2\times 1}.$$

3. Single-step vs. multi-step perturbation (SSP vs. MSP).

- **Previous work on SSP in economics:**

  Anderson, Chen & Zadrozny, Collard & Juillard, Judd, Kim & Kim, Schmitt-Grohe & Uribe, Sims, ...

- **Commonality between SSP and MSP:**

  SSP and MSP apply implicit function theorem k times:

  Example: Compute 1st-order solution $\hat{\phi}(\mathbf{x}) = \phi_0 + \nabla\phi_0(\mathbf{x}-\mathbf{x}_0)$ at $\mathbf{x}_0$ for the model $c(\phi(\mathbf{x}),\mathbf{x}) = 0_{n\times1}$:

  - **Solve $c(\mathbf{x}_0,\mathbf{x}_0) = 0_{n\times1}$ for nonstochastic steady state $\mathbf{x}_0$ and set $\phi_0 = \mathbf{x}_0$;**
  - **Differentiate c to obtain n×2n Jacobian matrix of 1st-order partial derivatives of c at $\mathbf{x}_0$: $\nabla c_0 = [\nabla c_{1,0},\nabla c_{2,0}]$;**
  - **Compute $\nabla\phi_0 = -(\nabla c_{1,0})^{-1}\nabla c_{2,0}$.**

- **Differences of error-properties of SSP and MSP:**

a. **SSP has local error properties:**

- **SSP error:** $\varepsilon_{SSP} = \leq \alpha|\Delta \mathbf{x}|^{k+1}$, **where** $\alpha = |\nabla^{k+1}\phi(\xi)|$ **and** ξ **= point in a sphere centered at** $\mathbf{x}_0$ **with radius** $|\Delta \mathbf{x}|$**.**

- **SSP error's order of magnitude:** $O(\varepsilon_{SSP}) = |\Delta \mathbf{x}|^{k+1}$, **for** $\alpha \leq 1$**.**

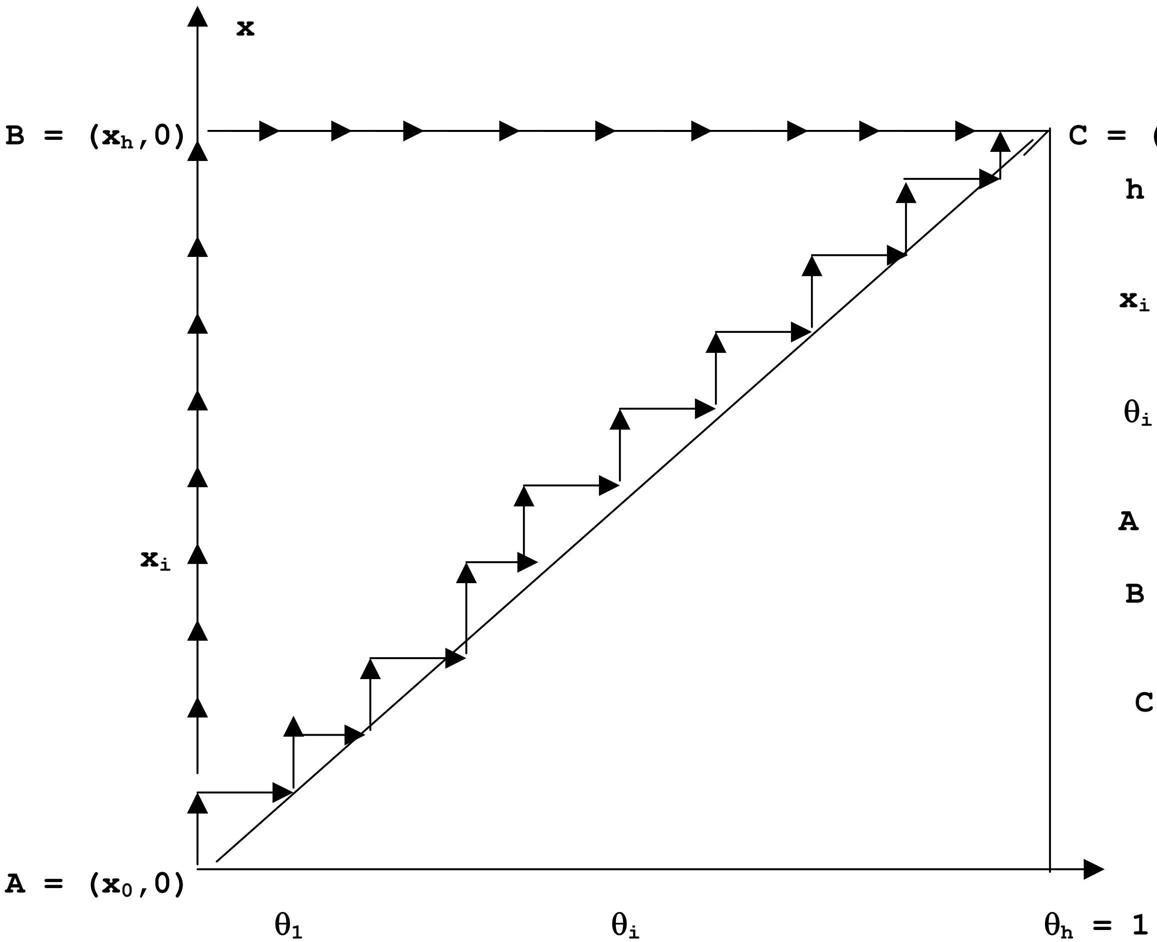
b. **MSP globalizes SSP's local error properties:**

- **SSP "moves in one big step" and is local because its error increases quickly with** $|\Delta \mathbf{x}|$**;**

- **MSP "moves in many small steps" and is global because its error can be limited to any chosen size by making the steps sufficiently small.**

- **<u>Figure 1:</u> <u>SSP and MSP paths in state space</u>.**

SSP goes in 1 step from A to C along 45^0 diagonal and MSP goes in 2h steps from A to B to C:



h = no. of nonstochastic and stochastic steps;

x_i = x at end of nonstochastic step i;

θ_i = θ at end of stochastic step i;

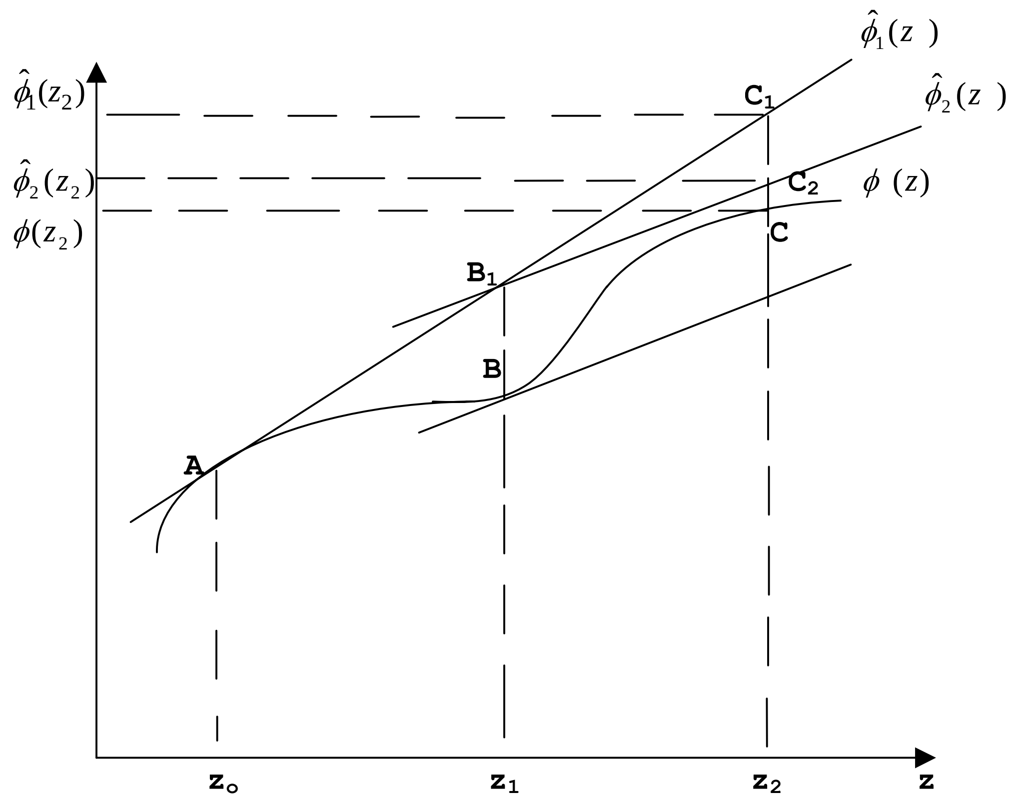
A = $(x_0,0)$ = initial state;

B = $(x_h,0)$ = state after h nonstochastic steps;

C = $(x_h,1)$ = final state after h stochastic steps.

- **Figure 2: SSP vs. MSP accuracy**

SSP goes in one step from z_o to z_2, with error $C_1 - C = |\phi(z_2) - \hat{\phi}_1(z_2)|$, for $z = (x^T, \theta)^T$, and MSP goes in two steps from z_o to z_1 to z_2, with smaller error $C_2 - C = |\phi(z_2) - \hat{\phi}_2(z_2)|$.



- **Table 1: ε_{MSP} = MSP order of magnitude of accuracy:**
  - **$|\Delta x|$ = 1 and kth-order approximation => $O(\varepsilon_{SSP}) = 1$.**
  - **$|\Delta x|$ = 1, kth-order approximation, h steps, h^{-1} stepsize => $O(\varepsilon_{MSP}) = h^{-k}$.**
  - **Table is based on: for given ε_{MSP} and h, $O(\varepsilon_{MSP}) = h^{-k} \leq \varepsilon_{MSP}$ requires h = smallest integer $\geq \varepsilon^{-1/k}$.**

ε	k	h^{-1}	h
Semi-Single Precision: $\varepsilon = O(10^{-4})$	1	1.00×10^{-4}	10^{4}
	2	1.00×10^{-2}	10^{2}
	3	4.55×10^{-2}	22
	4	1.00×10^{-1}	10
	5	1.43×10^{-1}	7
	6	2.00×10^{-1}	5
Single Precision: $\varepsilon = O(10^{-8})$	1	1.00×10^{-8}	10^{8}
	2	1.00×10^{-4}	10^{4}
	3	2.15×10^{-3}	465
	4	1.00×10^{-2}	100
	5	2.50×10^{-2}	40
	6	4.55×10^{-2}	22
Double Precision: $\varepsilon = O(10^{-16})$	1	1.00×10^{-16}	10^{16}
	2	1.00×10^{-8}	10^{8}
	3	4.64×10^{-6}	215,444
	4	1.00×10^{-4}	10^{4}
	5	6.31×10^{-4}	1585
	6	2.15×10^{-3}	465

- **<u>MSP advantages over SSP</u>:**

➢ **<u>In SSP</u>:**

1) $O(\varepsilon_{SSP}) = |\Delta x|^k$ increases quickly with $|\Delta x|$;

2) $|\Delta x| < 1 \Rightarrow O(\varepsilon_{SSP})$ can be reduced to any size by increasing k, which costs much more derivation and programming time;

3) $|\Delta x| > 1 \Rightarrow O(\varepsilon_{SSP})$ cannot be reduced below 1.

➢ **<u>In MSP</u>:**

1) Given $|\Delta x| \Rightarrow O(\varepsilon_{MSP}) = h^{-k}$ can be reduced to any size by increasing h or k, preferably by increasing h.

2) Nonstochastic and stochastic h can be different: may need stochastic h > nonstochastic h, to account for enough disturbance moments, hence, enough uncertainty.

4. MSP solution equations.

- **Definitions and rules of matrix differentiation:**

  - For **x** = m×1 and y = n×1, differentiate y = f(**x**), to obtain n×1 1st-order differential vector of f(**x**),

    (13) $$dy = \nabla f(\mathbf{x})\cdot d\mathbf{x},$$

    where $\nabla f(\mathbf{x})$ = n×m matrix of 1st-partial derivatives of f(**x**).

  - Differentiate $dy = \nabla f(\mathbf{x})\cdot d\mathbf{x}$, to obtain n×1 2nd-order differential vector of f(**x**),

    (14) $$d^2y = d\nabla f(\mathbf{x})\cdot d\mathbf{x} = (d\mathbf{x}^T \otimes I_n)\cdot\nabla^2 f(\mathbf{x})\cdot d\mathbf{x},$$

    where $\nabla^2 f(\mathbf{x})$ = nmxm matrix of 2nd-order derivatives of f(**x**).

  - Continue and obtain the kth-order $nm^{k-1}\times 1$ differential vector of f(**x**),

    (15) $$d(\mathrm{vec}(\nabla^{k-1} f(\mathbf{x}))) = (d\mathbf{x}^T \otimes \cdot\cdot\cdot \otimes d\mathbf{x}^T \otimes I_n)\nabla^k f(\mathbf{x})d\mathbf{x},$$

    where $\nabla^k f(\mathbf{x})$ = $nm^{k-1}\times m$ matrix of kth-order partial derivatives of f(**x**).

- **Implications**:

  - $\nabla^k f(x)$ is the Jacobian of the column vector of $\nabla^{k-1} f(x)$:

  - (16) $d^{j+k}y = (dx^T \otimes \cdot\cdot\cdot \otimes dx^T \otimes I_n) d^j \nabla^k f(x) \cdot dx$,

    so we can use partial derivatives in a mixed differential-gradient form.

- **Use 3 rules to derive MSP solution equations**:

  - **Vectorization rule**:

(17) $vec(ABC) = (C^T \otimes A) \cdot vec(B)$.

  - **Product rule**:

(18) $d[f(x)g(x)] = df(x) \cdot g(x) + f(x) \cdot dg(x)$.

  - **Chain rule**:

(19) $\nabla g[f(x)] = \nabla g[f(x)] \cdot \nabla f(x)$.

- **Overall steps of MSP computational algorithm.**

> **4 on-line steps**:

1) **Start: Compute steady state $\mathbf{x}_0$, s.t. $C(\mathbf{x}_o, \mathbf{x}_o, \mathbf{x}_o, \varepsilon_o, \varepsilon_o) = 0_{nx1}$**

2) **Apply h nonstochastic steps to update $\hat{\phi}(\mathbf{x}_i)$ and $\hat{\gamma}(\phi(\mathbf{x}_i))$;**

3) **Apply h stochastic steps to update $\hat{\phi}(\mathbf{x}_h, \theta_j\eta)$ and $\hat{\gamma}(\phi(\mathbf{x}_h), \theta_j\eta)$;**

4) **Finish: combine updated $\hat{\phi}(\mathbf{x}_h, \theta_h\eta)$ coefficients.**

> **1 off-line step**:

1) **Check solution accuracy at end of stochastic steps.**

- **2nd-order nonstochastic equations**:

*(20) $\nabla c_1 \nabla \phi_1^2 + \nabla c_2 \nabla \phi_1 + \nabla c_3 = 0_{n\times n}$,

*(21) $\nabla \phi_2 = -(\nabla c_1 \nabla \phi_1 + \nabla c_2)^{-1} \nabla c_4$,

(22) $\mathrm{vec}(d\nabla c_i) = [(\nabla^2 c_{i,1} \nabla \phi_1 + \nabla^2 c_{i,2})\nabla \phi_{1+2} + \nabla^2 c_{i,3+4}]\mathbf{dx}$,

*(23) $\mathrm{vec}(d\nabla \phi_1) = -[\nabla \phi_1^T \otimes \nabla c_1 + I_n \otimes (\nabla c_1 \nabla \phi_1 + \nabla c_2)]^{-1}$

$$\times\ \mathrm{vec}(d\nabla c_1 \nabla \phi_1^2 + d\nabla c_2 \nabla \phi_1 + d\nabla c_3),$$

*(24) $d\nabla \phi_2 = -(\nabla c_1 \nabla \phi_1 + \nabla c_2)^{-1} \times [(d\nabla c_1 \nabla \phi_1 + \nabla c_1 d\nabla \phi_1) + d\nabla c_2)\nabla \phi_2 + d\nabla c_4]$,

(25) $\Delta \phi_{1+2} = [\nabla \phi_{1+2} + (1/2) d\nabla \phi_{1+2}]\mathbf{dx}$,

(26) $\Delta \gamma_{1+2} = [\nabla \phi_1 \nabla \phi_{1+2} + (1/2)(d\nabla \phi_1 \nabla \phi_{1+2} + \nabla \phi_1 d\nabla \phi_{1+2})]\mathbf{dx}$,

where $\nabla^2 c_{i,3+4} = [\nabla^2 c_{i,3},\ \nabla^2 c_{i,4}]$, $\nabla \phi_{1+2} = [\nabla \phi_1,\ \nabla \phi_2]$, $d\nabla \phi_{1+2} = [d\nabla \phi_1,\ d\nabla \phi_2]$.

- **2nd-order stochastic equations**:

$$*(27)\quad d\nabla\phi_3 = -[\nabla c_1(\nabla\phi_1 + I_n) + \nabla c_2]^{-1}[(s_2^T/h) \otimes I_n]$$

$$\times [(\nabla\phi_2^T \otimes \nabla\phi_2^T \otimes I_n)\mathrm{vec}(\nabla^2 c_{1,1}) + (I_n \otimes \nabla\phi_2^T \otimes I_n)\mathrm{vec}(\nabla^2 c_{1,5})$$

$$+ (\nabla\phi_2^T \otimes I_n)\mathrm{vec}(\nabla^2 c_{5,1}) + \mathrm{vec}(\nabla^2 c_{5,5})],$$

$$(28)\quad \Delta\phi_3 = (1/2)d\nabla\phi_3,$$

$$(29)\quad \Delta\gamma_3 = (1/2)(I_n + \nabla\phi_1)d\nabla\phi_3,$$

where $s_2 = E(\varepsilon_{t+1} \otimes \varepsilon_{t+1})$.

- **Final updates**:

$$(30)\quad \Delta\phi = \Delta\phi_{1+2} + \Delta\phi_3,$$

$$(31)\quad \Delta\gamma = \Delta\gamma_{1+2} + \Delta\gamma_3.$$

- **Check solution accuracy**:

  Have final approximate solution $\hat{\phi}(\mathbf{x}) = \phi_0 + \nabla\phi_0\Delta\mathbf{x} + (1/2)(\Delta\mathbf{x}^T \otimes I_n)\nabla^2\phi_0$, for $\theta = 1$. Use Gauss-Hermite quadrature to evaluate the absolute $n\times1$ error vector

  $$(32)\quad e = |Ec(\hat{\phi}(\hat{\phi}(\mathbf{x}), \eta), \hat{\phi}(\mathbf{x}), \mathbf{x}, \eta)|,$$

  where $|\cdot|$ = vector of absolute values.

- **Computational complexity**:

➢ **Nonstochastic equations**:

1) Solve 1 n-dimensional quadratic equation, (20);
2) Solve 1 n^2-dimensional linear equation, (23);
3) Solve 2 n-dimensional linear equations, (21) and (24).

➢ **Stochastic equations**:

1) Solve 1 n-dimensional linear equation, (27).

5. SSP applied to an optimal growth model.

Table 2: Accuracy test of 2nd-order Taylor approximation: nonstochastic case.

	Euler equation: $C_1 = \beta u'_{t+1} f_{k,t+1} - u'_t$				Technology l.o.m.: $C_2 = \tau_t - \tau_{t-1}^{\rho} e^{\varepsilon\theta,t}$			
h	max	min	mean	stdv.	max	min	mean	stdv.
1	5.09E-06	1.06E-08	1.17E-06	1.01E-06	1.01E-06	2.18E-09	2.26E-07	2.23E-07
10	3.07E-07	3.57E-10	7.74E-08	6.43E-08	1.24E-08	2.67E-11	7.17E-09	6.08E-09
100	2.94E-08	5.00E-11	7.17E-9	6.08E-09	3.51E-10	7.58E-13	7.86E-11	7.74E-11
1000	2.93E-09	5.66E-12	7.11E-10	6.04E-10	2.62E-11	5.77E-14	5.88E-12	5.78E-12
10,000	2.86E-10	5.54E-13	6.94E-11	5.89E-11	2.45E-12	0.00E+00	5.53E-13	5.44E-13

$E_t \sum_{i=0}^{\infty} \beta^i u(c_{t+i})$, $u(c_t) = (1-\gamma)^{-1} c_t^{1-\gamma}$, $f(k_{t-1}, \tau_t) = \tau_t k_{t-1}^{\alpha}$,

$\tau_t = \tau_{t-1}^{\rho} \exp(\varepsilon_{\tau,t})$, $k_t = (1-\delta) k_{t-1} + \tau_t k_{t-1}^{\alpha} - c_t + \varepsilon_{k,t}$,

$(\beta, \gamma, \alpha, \rho, \delta) = (.95, .5, .33, .95, .1)$,

$(k^*, \tau^*) = (.1771, 1.)$, $k_{t-1}/k^* \in \{.9, 1.1\}$, $\tau_{t-1}/\tau^* \in \{.9, 1.1\}$.

Table 3: Accuracy test of 2nd-order Taylor approximation: stochastic case.

	$EC_1 = \beta u'_{t+1} f_{k,t+1} - u'_t$, $\Sigma = \begin{bmatrix} .0001 & .00005 \\ .00005 & .0001 \end{bmatrix}$, $\lvert\eta_{t+1}\rvert \in (.0001,\ .015)$				$C_1 = \beta u'_{t+1} f_{k,t+1} - u'_t$, $\Sigma = \begin{bmatrix} .001 & .0005 \\ .0005 & .001 \end{bmatrix}$, $\lvert\eta_{t+1}\rvert \in (.0005,\ .054)$			
h	**Max**	**Min**	**Mean**	**Std.**	**Max**	**Min**	**Mean**	**Std.**
1	1.33E-02	1.00E-03	6.12E-03	3.60E-03	1.20E-02	9.37E-04	5.66E-03	3.34E-03
10	1.33E-03	4.31E-05	5.93E-04	3.66E-04	1.25E-03	5.69E-05	5.55E-04	3.38E-04
100	1.35E-04	5.66E-06	5.99E-05	3.66E-05	1.43E-04	2.34E-05	7.43E-05	3.37E-05
1000	1.55E-05	2.51E-06	7.94E-06	3.66E-06	3.25E-05	2.06E-05	2.56E-05	3.38E-06
10,000	3.76E-06	2.13E-06	2.79E-06	4.60E-07	2.17E-05	2.02E-05	2.08E-05	4.24E-07

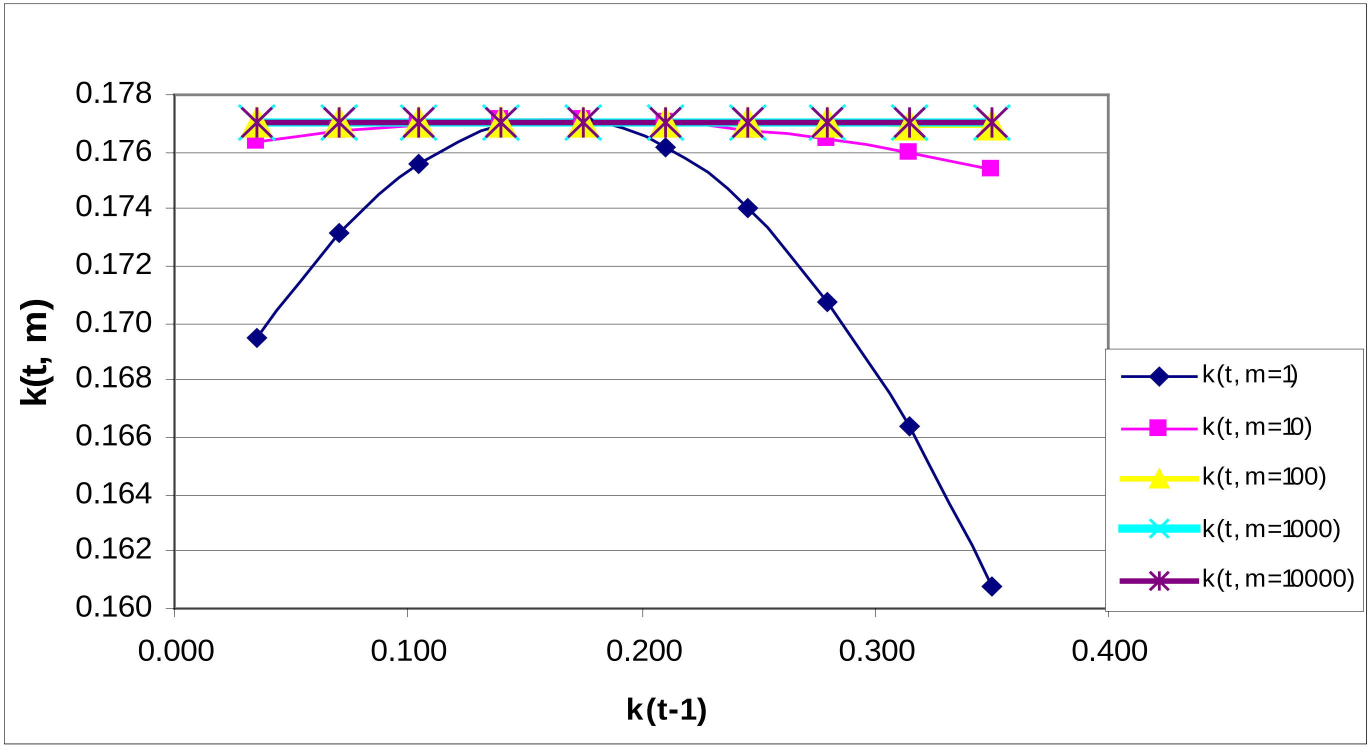
$E_t \sum_{i=0}^{\infty} \beta^i u(c_{t+i})$, $u(c_t) = (1-\gamma)^{-1} c_t^{1-\gamma}$, $f(k_{t-1}, \tau_t) = \tau_t k_{t-1}^{\alpha}$, $\Sigma = E\eta\eta^T$,

$\tau_t = \tau_{t-1}^{\rho} \exp(\varepsilon_{\tau,t})$, $k_t = (1-\delta) k_{t-1} + \tau_t k_{t-1}^{\alpha} - c_t + \varepsilon_{k,t}$, $\varepsilon_{t+1} = \varepsilon_t + \theta\eta_{t+1}$,

$(\beta,\ \gamma,\ \alpha,\ \rho,\ \delta) = (.95,\ .5,\ .33,\ .95,\ .1)$,

$(k^*,\ \tau^*) = (.1771,\ 1.0)$, $k_{t-1}/k^* \in \{.9,\ 1.1\}$, $\tau_{t-1}/\tau^* \in \{.9,\ 1.1\}$.

Figure 3.a: 2nd-order Taylor approximation of k_t: nonstochastic case.

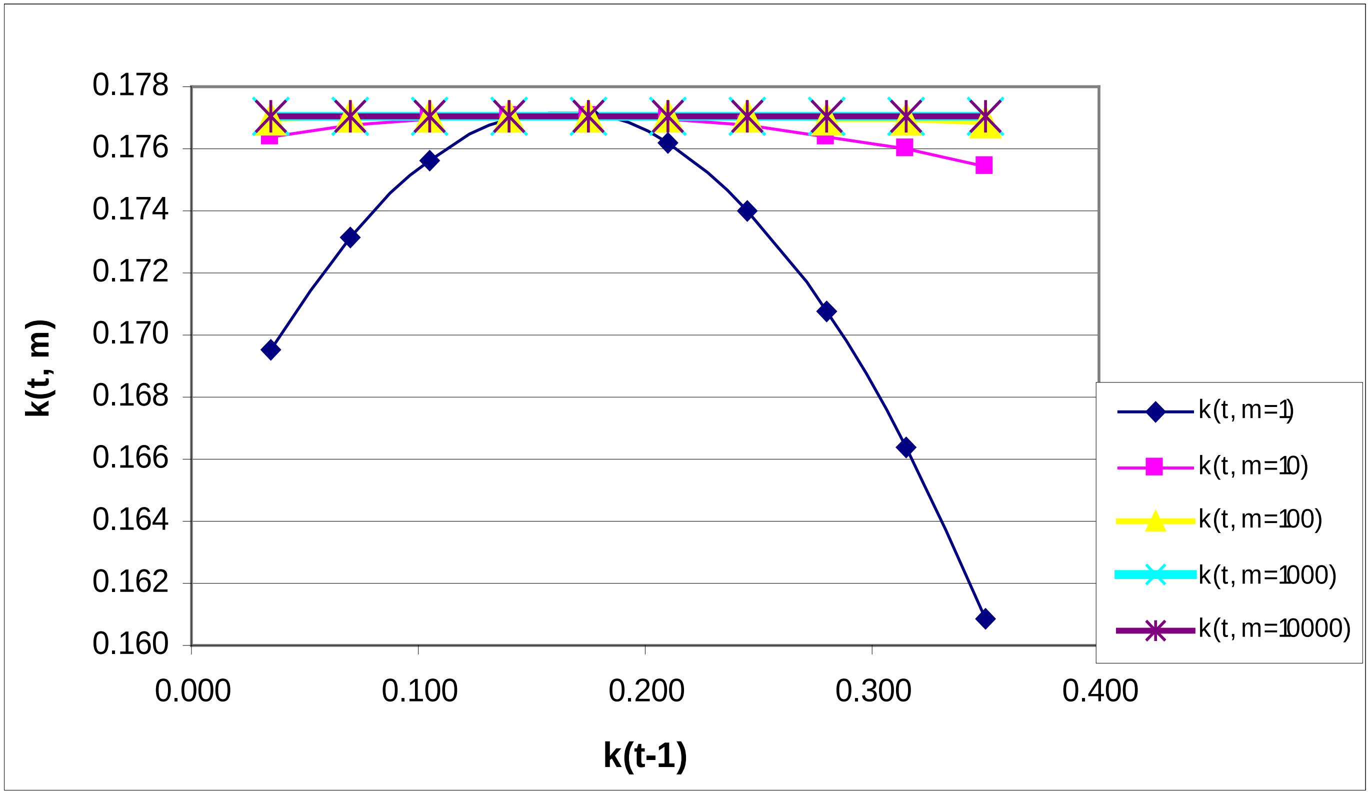


$\mathbf{k_{t-1}} \in \{.035, .07, .105, .14, .21, .245, .28, .315, .35\}$, $\tau_{t-1} = .95$,

$\mathbf{k_{t-1}/k^*} \in \{.2, .4, .6, .8, 1., 1.2, 1.4, 1.6\}$, $(k^*, \tau^*) = (.1770581, 1.)$,

$\hat{C}_1 \in \{3.94\text{x}10^{-3}, 1.29\text{x}10^{-9}\}$, $\hat{C}_2 \in \{5.94\text{x}10^{-5}, 1.46\text{x}10^{-10}\}$.

Figure 3.b: 2nd-order Taylor approximation of k_t: stochastic case.

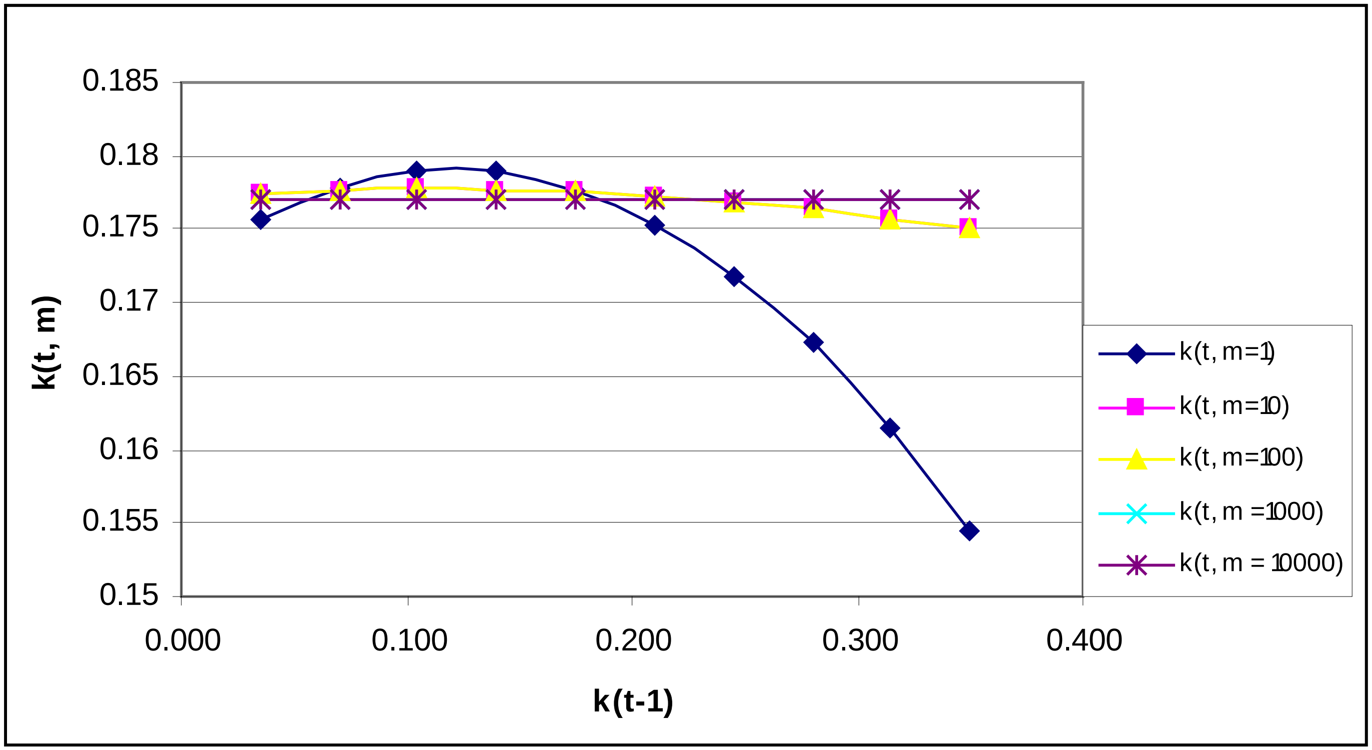


$\mathbf{k_{t-1} \in \{.035,\ .07,\ .105,\ .14,\ .21,\ .245,\ .28,\ .315,\ .35\},\quad \tau_{t-1} = .95,}$

$\mathbf{k_{t-1}/k^* \in \{.2,\ .4,\ .6,\ .8,\ 1.,\ 1.2,\ 1.4,\ 1.6\},\quad (k^*,\tau^*) = (.1770581,\ 1.),}$

$\mathbf{E\hat{C}_1 \in \{2.72x10^{-2},\ 4.81x10^{-6}\},\quad |\eta_{t+1}| \in \{.001,\ .054\}.}$

Figure 4.a: 2nd-order Taylor approximation of k_t: stochastic case.

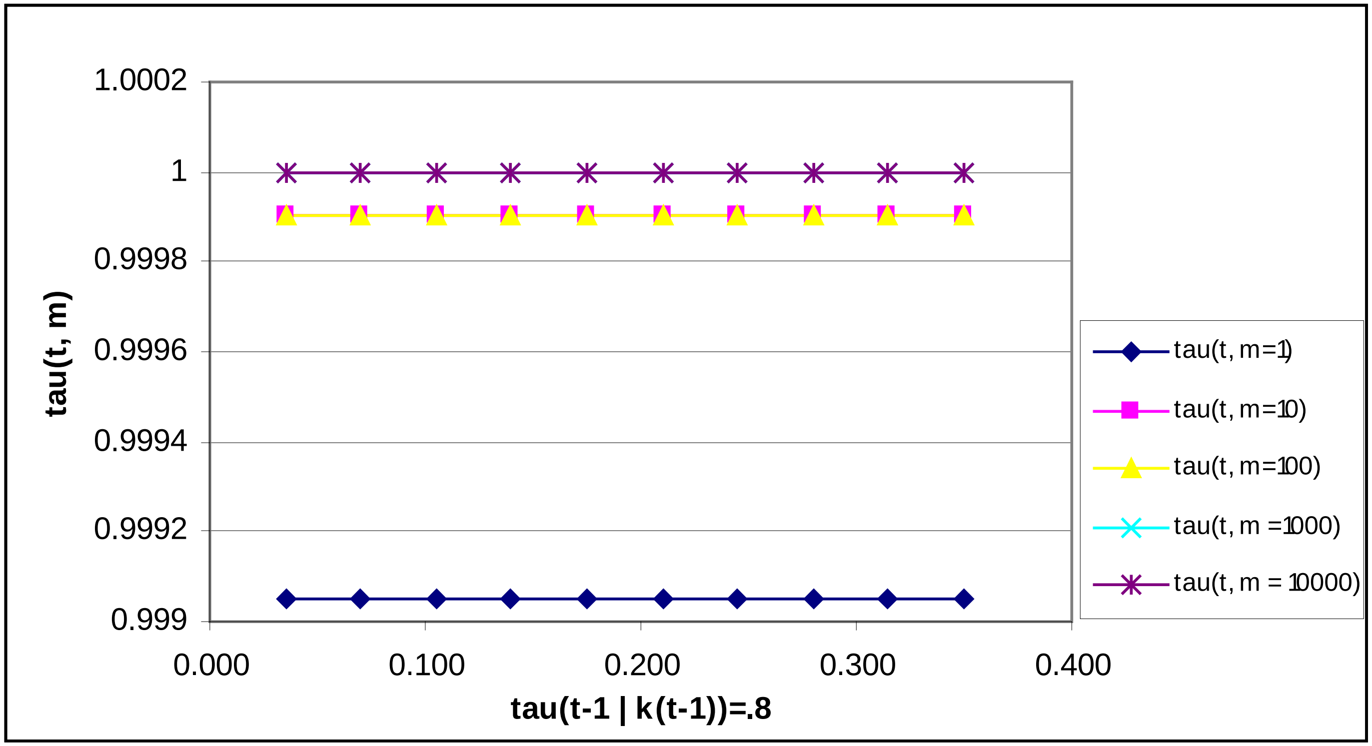


$k_{t-1} \in \{.035, .07, .105, .14, .21, .245, .28, .315, .35\}$, $\tau_{t-1} = .8$,

$k_{t-1}/k^* \in \{.2, .4, .6, .8, 1., 1.2, 1.4, 1.6\}$, $(k^*, \tau^*) = (.1770581, 1.)$,

$E\hat{C}_1 \in \{3.3\text{x}10^{-2}, 3.4\text{x}10^{-6}\}$, $|\eta_{t+1}| \in \{.001, .054\}$.

Figure 4.b: 2nd-order Taylor approximation of τ_t: stochastic case.



$k_{t-1} \in \{.035, .07, .105, .14, .21, .245, .28, .315, .35\}$, $\tau_{t-1} = .8$,

$k_{t-1}/k^* \in \{.2, .4, .6, .8, 1., 1.2, 1.4, 1.6\}$, $(k^*, \tau^*) = (.1770581, 1.)$,

$E\hat{C}_1 \in \{3.3x10^{-2}, 3.4x10^{-6}\}$, $|\eta_{t+1}| \in \{.001, .054\}$.

6. Conclusions.

- **We derived MSP equations for computing 4th-order approximate solutions of NLRE models and illustrated 2nd-order MSP solutions equations and an optimal growth model.**

- **For sufficient approximation order k, SSP provides good local accuracy, but increasing k adds costly derivation and programming time. MSP solutions are accurate on the order of h^{-k} for h number of steps. Increasing h requires only repeating preprogrammed loops of mostly linear operations more times. MSP cheaply extends SSP to be accurate over a much larger region of the state space and, thus, effectively globalizes it.**

- **MSP is easy to program in a matrix oriented programming language because all solution equations are expressed in standard linear-algebraic operations of vectors and matrices. The linear-algebraic form also facilitates analytical understanding of the solution equations.**

- **In addition to rational expectations models, MSP could be applied to various dynamic economic, financial, and statistical models. For example, it has been shown that MSP can be used to compute price indexes and productivity indexes with high accuracy in models that are based on explicit forms of functions to be maximized (Chen & Zadrozny, 2004).**

Dynamic programming

- **Bellman equation**:

(1) $v(x) = \max\ [u(\phi(x),x) + \beta Ev(f(x,\phi(x),\varepsilon))]$ w.r.t ϕ,

x = state vector of predetermined observed and unobserved variables; $\phi(x)$ = decision function to be computed; $f(x,\phi(x),\varepsilon)$ = state transition function; and ε = unobserved disturbance.

- **First-order necessary conditions:**

➢ Differentiate [·] in (1) w.r.t. ϕ:

(2) $u'(\phi(x),x) + \beta Ev'(f(x,\phi(x),\varepsilon))f_2(x,\phi(x),\varepsilon) = 0.$

➢ Differentiate (1) w.r.t. x:

(3) $v'(x) = u_2(x) + \beta Ev'(f(x,\phi(x),\varepsilon))f_1(x,\phi(x),\varepsilon).$

- **Solution objectives**:

➢ In general: Given $u(\cdot)$, $f(\cdot)$, β, compute ϕ and v' which solve (2) & (3).

➢ In MSP: compute polynomial approxs. of ϕ and v' which solve (2) & (3) at x.

➢ Comparison with NLREM: $v'(f(\phi(\cdot)))$ in DP corresponds to $\phi(\phi(\cdot))$ in NLREM.

- **Possible financial time-series application:**

  - Choose $u(\cdot)$ to represent a filtering criterion.

  - Choose a model, $f(\cdot)$, and estimate its parameters.

  - Solve (2) and (3) for filter $\phi(\cdot)$, which estimates stochastic-volatility disturbances.