Solving the Stochastic Growth Model with a Finite Element Method

Ellen R. McGrattan

Federal Reserve Bank of Minneapolis

First Draft: January, 1993

Current Draft: July, 1993

ABSTRACT

Since it is the dominant paradigm of the business cycle and growth literatures, the stochastic growth model has been used to test the performance of alternative numerical methods. This paper applies the finite element method to this example. I show that the method is easy to apply and, for examples such as the stochastic growth method, gives accurate solutions within a second or two on a desktop computer. I also show how inequality constraints can be handled by redefining the optimization problem with penalty functions.

Correspondence to: Research Department, Federal Reserve Bank of Minneapolis, 250 Marquette Avenue, Minneapolis, MN 55401-0291.

This article is based in part on conversations with Jeff Eischen at North Carolina State University. I would like to thank Tom Sargent and Martie Starr for helpful comments and the National Science Foundation for their support under grant #SES-9108758.

1. Introduction

Most applied economists agree that for practical problems they must go beyond analytically tractable models to confront the theory with the data. For this reason, there has been a surge in the development of numerical methods designed to compute equilibria in problems that do not allow for analytical solutions. To date, the big trade-off has been computing time versus accuracy of the solution. For example, some numerical methods rely on approximating nonlinear decision functions by linear functions.[1] Calculating linear decision functions requires little computer time. In some cases, however, the true decision functions are highly nonlinear, and linear approximations are therefore inaccurate.[2]

Since the dominant paradigm of the business cycle and growth literatures has been the stochastic growth model, most practitioners have focused on it in their attempts to narrow the gap between computational time and the accuracy of the solution. Taylor and Uhlig (1990), for example, compare a variety of different algorithms for computing equilibria in the stochastic growth model.[3] They find that none of these performed well in all respects and illustrate, through a battery of tests, the need for improved, less computer-intensive methods.

In this paper, I apply a method which is widely used in engineering applications such as structural analysis and aerodynamic design to this problem. The method, called the finite element method, is an algorithm for solving functional equations and its advantages are that, for certain problems, it is both fast and accurate. Using the Taylor and Uhlig (1990) application, I demonstrate that the finite element method works extremely well when applied to a case with an analytical solution.[4] For cases that do not have an analytical solution, I show that the method yields similar decision functions to discretized dynamic programming in a fraction of the computing time. Finally, I show that the method can also be applied to problems with inequality constraints. In the growth example, inequality constraints arise from the assumption that investment is positive for all realizations of the capital stock and the shock to technology. Thus, the finite element method does a good

[1] See Kydland and Prescott (1982) for an example.

[2] Linear approximations are also inaccurate in models with large shocks. Braun and McGrattan (1993), who consider very large fiscal shocks, use a finite element approximation such as that discussed here.

[3] Computational methods were the subject of a conference held at the Federal Reserve Bank of Minneapolis in 1988. The result of the meeting was a collection of papers which includes Taylor and Uhlig (1990) in volume 8 of the *Journal of Business and Economic Statistics.*

[4] My purpose here is to describe the method by way of examples. For a general treatment of the finite element method, see Hughes (1987) or Reddy (1992).

job of narrowing the computational time and accuracy trade-off.

The model described in Taylor and Uhlig (1990) has two state variables and, therefore, requires computation of approximate functions that are two-dimensional. Because the one-dimensional approximations are easier to visualize, I start with a simple one-dimensional textbook example. This is done in section 2. In section 3, I apply the finite element method to the stochastic growth model. Concluding remarks are given in section 4.

2. A simple example

Let me start with a simple example. Consider the following differential equation:

$$\frac{df(x)}{dx} + f(x) = 0, \quad f(0) = 1. \tag{1}$$

I am interested in finding the function $f(x)$ defined on the interval $[0,1]$ which solves the differential equation in (1). It turns out that the exact solution to the problem is the exponential function, $f(x) = \exp(-x)$. Suppose, however, that I do not know the exact solution and want some approximate solution that is "close" to the exact solution. Let $f^h(x)$ be one approximate solution and assume that $f^h(0) = 1$. Because f^h is approximate, $\frac{df^h(x)}{dx} + f^h(x)$ is not necessarily equal to 0 for all x in $[0,1]$. Thus, I need some concept of "close." I also need to choose the space of functions in which the approximations, f^h, lie.

In the case of finite element methods, "close" means that the approximate solution satisfies the *weak formulation* of the problem in (1):

$$\int_0^1 w(x)\left(\frac{df^h(x)}{dx} + f^h(x)\right)dx = 0, \quad f^h(0) = 1, \tag{2}$$

where $w(x)$ is a weighting function that satisfies (a) $w(0) = 0$ and (b) $\int_0^1 [w'(x)]^2 dx < \infty$.[5] In the version of the finite element method that I use here, the functions $f^h(x)$ and $w(x)$ are assumed to be piecewise linear, e.g.

$$f^h(x) = \sum_{a=1}^{n} f_a N_a(x), \quad w(x) = \sum_{a=1}^{n} w_a N_a(x), \tag{3}$$

where

$$N_a(x) = \begin{cases} \frac{x - x_{a-1}}{x_a - x_{a-1}}, & x_{a-1} \le x \le x_a \\ \frac{x_{a+1} - x}{x_{a+1} - x_a}, & x_a \le x \le x_{a+1} \\ 0, & \text{elsewhere} \end{cases}, \tag{4}$$

[5] The restrictions on w and f^h ensure integrability of the weak formulation and satisfaction of the boundary constraints. See Reddy (1992) for a discussion of boundary constraints.

and f_1, ..., f_n and w_1, ... w_n are constants. The functions $N_a(x)$, $a = 1, \ldots n$, are called shape functions or basis functions. If the exact solution to the problem is nonlinear, it is necessary to use a sufficient number of grid points (or nodes) to resolve the nonlinearities. Alternatively, higher order polynomials can be used in place of the linear basis functions, $N_a(x)$, and require fewer grid points for resolving nonlinearities.

A typical shape function, $N_a(x)$, is displayed in figure 1. Notice that $N_a(x)$ is equal to zero for all x except those that lie in the intervals (or 'elements' in the language of the finite element method): $[x_{a-1}, x_a]$ and $[x_a, x_{a+1}]$. In figure 2, all of the shape functions on [0,1] are displayed. Notice that they overlap each other. The piecewise linear approximation $f^h(x)$ in figure 3 can be constructed with the shape functions. For the interval $[x_a, x_{a+1}]$, $f^h(x)$ is given by a weighted sum of the functions $N_a(x)$ and $N_{a+1}(x)$, i.e., $f_a N_a(x) + f_{a+1} N_{a+1}(x)$. Note that in the first interval, the boundary condition, $f^h(0) = 1$, and the restriction on the weighting function, $w(0) = 0$, must be enforced. Since $N_1(0) = 1$ and $N_2(0) = 0$, the approximate function at $x = 0$ is equal to f_1, and the weighting matrix at $x = 0$ is equal to w_1. Therefore, to impose the boundary condition, I set $f_1 = 1$ and $w_1 = 0$.

If the approximations in (3) are substituted into the weak form of the problem given by (2), the result is an equation that depends on the constants of the approximate solution, $\vec{f} = [1,\ f_2,\ f_3,\ \ldots,\ f_n]'$ and the coefficients of the weighting function $\vec{w} = [0,\ w_2,\ w_3,\ \ldots,\ w_n]'$, namely

$$\vec{w}' \int_0^1 (\vec{N} d\vec{N}'/dx + \vec{N}\vec{N}')\vec{f} dx = 0, \tag{5}$$

where $\vec{N} = [N_1(x), N_2(x), \ldots, N_n(x)]'$ and $d\vec{N}/dx$ is a vector of first derivatives of the basis functions. Since $\vec{w}$ and $\vec{f}$ are vectors of constants, they can be brought outside of the integral. Assuming that (5) holds for any arbitrary weights, $\vec{w}$, it must be the case that

$$\int_0^1 N_a(x)(\frac{d\vec{N}'}{dx} + \vec{N}')\vec{f} dx = 0, \quad a = 2, \ldots n. \tag{6}$$

In (6), there are $n-1$ equations and $n-1$ unknowns ($\vec{f}$). I can solve these equations to determine the finite element approximation to $f(x)$.

Consider the following simple examples. First, assume that there is one element and nodes at 0 and 1. In this case, $N_1(x) = 1 - x$ and $N_2(x) = x$ on [0,1]. Therefore, the integral in (5) is

$$[0 \quad w_2] \int_0^1 \begin{bmatrix} x - 1 + (1-x)^2 & 1 - x + (1-x)x \\ -x + x(1-x) & x + x^2 \end{bmatrix} \begin{bmatrix} 1 \\ f_2 \end{bmatrix} dx = 0 \tag{7}$$

or, after integration,

$$[0 \quad w_2]\begin{bmatrix} -1/6 & 2/3 \\ -1/3 & 5/6 \end{bmatrix}\begin{bmatrix} 1 \\ f_2 \end{bmatrix} = w_2(-\frac{1}{3} + \frac{5}{6}f_2) = 0.$$

The equation in (7) must be satisfied for any arbitrary but nonzero value for w_2. Thus, the approximate solution for the one element case has $f^h(1) = f_2 = 0.4$. The exact solution at $x = 1$ is $f(1) = 0.37$. For the second example, assume that there are two elements with nodes at 0, 0.5 and 1. By the definition of the basis functions in (4), I have $N_1(x) = 1 - 2x$, $N_2(x) = 2x$, $N_3(x) = 0$ on the interval [0,0.5] and $N_1(x) = 0$, $N_2(x) = 2 - 2x$, $N_3(x) = 2x - 1$ on the interval [0.5,1]. After carrying out the integration in (5) for this case, I have

$$[0 \quad w_2 \quad w_3]\begin{bmatrix} -1/3 & 7/12 & 0 \\ -5/12 & 1/3 & 7/12 \\ 0 & -5/12 & 2/3 \end{bmatrix}\begin{bmatrix} 1 \\ f_2 \\ f_3 \end{bmatrix} = 0.$$

If this equation is to be satisfied for arbitrary values of w_2 and w_3, then f_2 and f_3 must satisfy

$$\begin{bmatrix} 1/3 & 7/12 \\ -5/12 & 2/3 \end{bmatrix}\begin{bmatrix} f_2 \\ f_3 \end{bmatrix} = \begin{bmatrix} 5/12 \\ 0 \end{bmatrix}.$$

The constants that satisfy these two equations are $f_2 = 0.6$ and $f_3 = 0.37$. The exact solution at $x = 0.5$ is $f(0.5) = 0.61$ and the exact solution at $x = 1$ is $f(1) = 0.37$.

Note that the integral in (2) can also be calculated numerically. For example, if Gaussian quadrature is used, then the integral of a function $g(x)$ over the interval $[a, b]$ is given by

$$\int_a^b g(x)dx \approx \sum_{i=1}^{m} \gamma_i g(x_i)$$

for m-point quadrature, where γ_i is the weight given to the function at the point x_i.[6]

In the next section, I apply the finite element method to the stochastic growth model. The main difference between the growth example and the example of equation (1) is the dimension of the functions being computed. For the example of equation (1), the function is one-dimensional. For the growth example, I compute functions that are two-dimensional. Since the one-dimensional functions are easier to visualize, I started with

[6] See Press, et. al. (1986) for simple algorithms that calculate the weights, γ_i, and abscissas, x_i, $i = 1, \ldots, m$ with inputs a, b, and m.

the example of equation (1). However, extending the algorithm to higher dimensions is relatively straightforward.

3. The stochastic growth model

In this section, I apply the finite element method to a simple but widely used model in economics: the stochastic growth model. I consider three specific examples. For the first example, I choose a parameterization of the model that allows for an analytical solution. This example serves as a test case for evaluating the accuracy of the algorithm. The second example is a case considered by Taylor and Uhlig (1990). Since no analytical solution exists for the parameterizations that they consider, I compare the results for the second example with the results reported in Taylor and Uhlig (1990). The third example is one studied by Christiano and Fisher (1993) who impose nonnegativity constraints on investment. For this example, I show how inequality constraints can be handled without changing the implementation of the finite element method.

3.1. Statement of the problem

Let c_t be consumption and k_t be the capital stock at date t. Agents are assumed to maximize expected utility:

$$E\left[\sum_{t=0}^{\infty} \beta^t \frac{c_t^{1-\tau}}{1-\tau} \,\middle|\, k_{-1}\right], \qquad 0 < \beta < 1,\ \tau > 0,\ k_{-1} \text{ given} \tag{8}$$

subject to the following resource constraint

$$c_t + k_t - (1-\delta)k_{t-1} = \theta_t k_{t-1}^{\alpha}, \qquad 0 < \alpha < 1,\ 0 \leq \delta \leq 1,$$

and nonnegativity constraints $c_t \geq 0$, $k_t \geq 0$ at all dates t. The term $k_t - (1-\delta)k_{t-1}$ is investment at date t. Note that δ of the stock depreciates between dates $t-1$ and t. The term $\theta_t k_{t-1}^{\alpha}$ is the output produced with k_{t-1} units of the capital stock if the level of technology is given by θ_t. For this example, I assume that the level of technology varies over time and the process for θ_t is given by

$$\ln \theta_t = \rho \ln \theta_{t-1} + \varepsilon_t,$$

where ε_t is a serially uncorrelated, normally distributed random variable with mean zero and variance σ_ε^2.

The first order conditions for the optimization problem in (8) imply that the following equations must be satisfied by the optimal decision functions for all values of k and θ:

$$R(k,\theta;c) = c(k,\theta)^{-\tau} - \beta \int_{-\infty}^{\infty} c(\tilde{k},\tilde{\theta})^{-\tau}(\alpha\tilde{\theta}\tilde{k}^{\alpha-1} + 1 - \delta)f(\epsilon)d\epsilon = 0, \qquad c(0,\theta) = 0, \quad (9)$$

where

$$\tilde{k} = \theta k^{\alpha} + (1-\delta)k - c(k,\theta),$$
$$\tilde{\theta} = \theta^{\rho}\exp(\epsilon)$$

and $f(\epsilon)$ is the normal density function. The objective is to find the consumption function c such that residual $R(k,\theta;c)$ in equation (9) is approximately equal to zero for all values of k and θ. This is done in the next section.

3.2. Solution method

In this section, I describe how to apply a finite element method to the problem of section 3.1. As I noted in section 2, two choices must be made. I must decide on the space of functions from which I choose an approximation to the consumption function (c^h). I must also decide on a concept of "close" when seeking functions that approximately satisfy the first order conditions in (9).

For the functions, I choose from the space of piecewise bilinear functions. If the grid over the capital stock and the technology shock is rectangular, then each element can be defined as a rectangular interval over k and θ. Consumption on element e is defined to be

$$c^e(k,\theta) = \sum_{j=1}^{4} c^e_{(j)} N^e_{(j)}(k,\theta), \qquad (10)$$

where $N^e_{(j)}$, $j = 1, \ldots 4$ are functions to be defined below and $c^e_{(j)}$, $j = 1, \ldots 4$ are constants. The consumption function, unlike the function of section 2, is defined over two dimensions.[7] In section 2, the functions on any element e are defined as a weighted sum of two linear functions. Here, the function on element e is a weighted sum of four functions.

In figure 4, I display a typical rectangular grid. The grid is divided into smaller rectangles called elements. The elements are marked by encircled numbers. Each point on the grid, or node, is numbered. When referring to the nodes of a particular element, it is convenient to refer to the local nodes, which are marked 1 through 4 and appear

[7] From the two-dimensional case it is easy to see how higher dimensions can be handled.

in parentheses in figure 4. To calculate consumption for element 3 of the example grid in figure 4, I need to know the values of the constants $c^3_{(j)}$ and the functions $N^3_{(j)}(k,\theta)$, $j = 1,2,3,4$, defined on the rectangle with (global) nodes 3, 4, 8, and 7. Assume that the rectangle for element e is $[\underline{k}^e, \overline{k}^e] \times [\underline{\theta}^e, \overline{\theta}^e]$. The functions $N^e_{(j)}(k,\theta)$, $j = 1,2,3,4$ are assumed to be bilinear functions and are defined as follows

$$
\begin{aligned}
N^e_{(1)}(k,\theta) &= \frac{\overline{k}^e - k}{\overline{k}^e - \underline{k}^e} \cdot \frac{\overline{\theta}^e - \theta}{\overline{\theta}^e - \underline{\theta}^e} \\
N^e_{(2)}(k,\theta) &= \frac{k - \underline{k}^e}{\overline{k}^e - \underline{k}^e} \cdot \frac{\overline{\theta}^e - \theta}{\overline{\theta}^e - \underline{\theta}^e} \\
N^e_{(3)}(k,\theta) &= \frac{k - \underline{k}^e}{\overline{k}^e - \underline{k}^e} \cdot \frac{\theta - \underline{\theta}^e}{\overline{\theta}^e - \underline{\theta}^e} \\
N^e_{(4)}(k,\theta) &= \frac{\overline{k}^e - k}{\overline{k}^e - \underline{k}^e} \cdot \frac{\theta - \underline{\theta}^e}{\overline{\theta}^e - \underline{\theta}^e}.
\end{aligned}
$$

The subscript on N denotes the local node and the superscript denotes the element. Note that the bilinear function $N^e_{(1)}(k,\theta)$ is equal to 1 at $(\underline{k}^e, \underline{\theta}^e)$ and 0 at all other local nodes in element e. In general, the function $N^e_{(j)}$ is equal to 1 at local node j and zero at all other local nodes in element e. With $N^e_{(j)}$ and the constants $c^e_{(j)}$, $j = 1,2,3,4$ that are to be computed, I can calculate consumption for any point in element e.

I have defined the basis or shape functions for a typical element. I can also define them over the entire domain. Suppose that node a is at the intersection of four elements marked e_1, e_2, e_3, and e_4. Since a is at the intersection of the four elements, it is a local node for each. Suppose that global node a is also local node (j) for element e_j, $j = 1,2,3,4$. Define $N_a(k,\theta)$ as follows: on element e_j, $N_a(k,\theta)$ is equal to $N^{e_j}_{(j)}(k,\theta)$, $j = 1,2,3,4$ and everywhere else $N_a(k,\theta)$ is equal to 0. Suppose, for example, that $a = 6$ in figure 4. Node 6 is local node (1) for element 5, local node (2) for element 4, local node (3) for element 1, and local node (4) for element 2. Thus, I set $e_1 = 5$, $e_2 = 4$, $e_3 = 1$, and $e_4 = 2$. In this case, $N_6(k,\theta)$ is equal to $N^5_{(1)}$ on element 5, $N^4_{(2)}$ on element 4, $N^1_{(3)}$ on element 1, $N^2_{(4)}$ on element 2, and 0 everywhere else. In figure 5, I display $N_a(k,\theta)$ for a typical node a. Its shape is like that of a pyramid. The function is positive for values of (k,θ) in the four elements surrounding node a and zero elsewhere. At a, $N_a(k,\theta)$ is equal to 1.

The approximate consumption function can also be defined over the entire domain, e.g.,

$$c^h(k,\theta) = \sum_{a=1}^{n} c_a N_a(k,\theta), \tag{11}$$

where n is the total number of (global) nodes and c_a is the value of consumption at node a. Note that the constants c_a, $a = 1, \ldots n$, in equation (11) are related to the constants $c^e_{(j)}$, $j = 1, 2, 3, 4$ in equation (10). Consider the mesh of figure 4 as an example. The constant $c^3_{(2)}$ for the local node number 2 in element 3 is equal to the constant c_4 which is the value of consumption at the global node number 4.

The constants, c_a, $a = 1, \ldots n$, of the approximate solution in equation (11) are chosen to satisfy the weak form of the problem,

$$\int_0^{\overline{k}} \int_{\underline{\theta}}^{\overline{\theta}} w(k, \theta) R(k, \theta; c^h) dk \, d\theta = 0,$$

where $w(k, \theta)$ is a piecewise bilinear weighting function that satisfies $w(0, \theta) = 0$ for all $\theta > 0$. If the weak form of the problem is to be satisfied for an arbitrary weighting function, then the problem is to find constants c_a at all nodes a for which the capital stock is nonzero. That is, the main computational task is to find the unknown constants c_a such that

$$\int_0^{\overline{k}} \int_{\underline{\theta}}^{\overline{\theta}} N_a(k, \theta) R\big(k, \theta; c^h\big) dk \, d\theta = 0, \qquad \text{for all } a \in \mathcal{A}. \tag{12}$$

The set $\mathcal{A}$ includes all nodes except those at the $k = 0$ boundary. Let $\vec{c}$ be a vector with elements c_a, $a \in \mathcal{A}$ and denote the system of equations in (12) by $H(\vec{c}) = 0$. If the partition on $[0,\bar{k}]$ has n_k points and the partition on $[\underline{\theta}, \overline{\theta}]$ has n_θ points (i.e., $n = n_k \times n_\theta$), then the set $\mathcal{A}$ has $(n_k - 1)n_\theta$ elements. Thus, the set of equations given by $H(\vec{c}) = 0$ is a system of $(n_k - 1)n_\theta$ equations with $(n_k - 1)n_\theta$ unknowns.

In the example of section 2, a linear system of equations is solved to compute the approximate solution. In the growth example, I have a system of nonlinear equations to solve, i.e., $H(\vec{c}) = 0$. If I use a Newton-Raphson algorithm to find the vector $\vec{c}$ which satisfies the nonlinear system of equations, then I choose some initial guess, say $\vec{c}_0$, and iterate as follows:

$$\vec{c}_{\ell+1} = \vec{c}_\ell - J(\vec{c}_\ell)^{-1} H(\vec{c}_\ell), \tag{13}$$

where $\vec{c}_\ell$ is the guess of $\vec{c}$ at iteration ℓ and J is the Jacobian matrix of H. The (i, j) element of J is the derivative of the ith equation in H with respect to the jth element of $\vec{c}$. The iterations in (13) require algorithms to compute x in $Ax = b$ for $A = J(\vec{c})$, $b = H(\vec{c})$. In most cases, A is sparse. Thus, if the application has many state variables, the matrix A can be stored in compressed form for more efficient storage and direct or iterative methods can be used to solve the large linear system.

Integration is required to compute both the residual, R, in (9) and the system of equations, H, of the weak formulation of the problem. To calculate the integral in R, I first substitute the expressions for next period's capital and technology $(\tilde{k}, \tilde{\theta})$ into the integrand. The integrand is, thus, a function of current capital, k, current technology, θ, and the innovation in technology, ϵ. Therefore, for each value of (k, θ), I can compute the integral in (9) by Gaussian quadrature. In this case, I use Gauss-Legendre quadrature which is described in detail in Press, et. al. (1986). To do this, I need to choose a finite interval for ϵ and the number of points at which I evaluate the integrand. Given $R(k, \theta; c^h)$, I can compute the elements of $H(\vec{c})$. Notice, however, that for a typical element of $H(\vec{c})$, the integrand is nonzero for only a small part of the domain. Take, for example, the element of H associated with node a. In this case, I only have to integrate over the 4 elements that surround node a because $N_a(k, \theta)$ is zero for all other values of capital and the technology shock.

In the appendix, I describe the main steps of the algorithm in more detail.[8] In specifying the algorithm for the growth example, I assume that certain parameters are given. First, I need a parameterization of the utility and production functions, i.e. choices of β, δ, τ, ρ, α, and σ_ϵ. I also need to specify the grid over the capital stock and the technology shock, i.e., choices of $\overline{k}$, $\underline{\theta}$, and $\overline{\theta}$ and partitions over $[0, \overline{k}]$ and $[\underline{\theta}, \overline{\theta}]$. Let $\vec{k}$ $= [0, k_2, k_3, \ldots, k_{n_k}]$ be a partition for the capital stock and let $\vec{\theta} = [\theta_1, \theta_2, \ldots, \theta_{n_\theta}]$ be a partition for the technology shock.[9] For each (k_i, θ_j) pair, $i = 2, \ldots, n_k$, $j = 1, \ldots, n_\theta$, I need an initial guess for consumption, $c(k_i, \theta_j)$. For integration, I need to choose an interval for the innovations in technology, ϵ, and the number of points to be used for Gaussian quadrature when computing integrals in (9) and (12). For computing the integral in (9), assume that $[\underline{\epsilon}, \overline{\epsilon}]$ is the interval and that m_ϵ points are used in quadrature. For computing the integral in (12), I assume that a different number of quadrature points can be used for each interval in the partitions of k and θ. Let $m_{k,i}$ denote the number of quadrature points used for integration with respect to the capital stock on interval $[k_i, k_{i+1}]$. Similarly, let $m_{\theta,i}$ denote the number of quadrature points for integration with respect to the technology shock on element $[\theta_i, \theta_{i+1}]$.

In the next two sections, I illustrate the performance of the finite element method

---

[8] The appendix and codes written in Fortran 77 are available from the author upon request to erm@ellen.mpls.frb.fed.us.

[9] The subscripts on k and θ index nodes in the partition of the grid. Time subscripts are not needed here because the solutions are time-invariant.

with two specific parameterizations. The first is a test case, for which we have a known solution. The second is an example studied by Taylor and Uhlig (1990).[10]

3.3. *A test case*

If I assume that the capital stock fully depreciates each period (i.e., $\delta = 1$) and that the utility function is logarithmic (i.e., $\tau = 1$), then I can obtain an analytical solution to the problem stated in section 3.1. The decision function for consumption when $\delta = 1$ and $\tau = 1$ is given by

$$c(k_{t-1}, \theta_t) = (1 - \beta\alpha)\theta_t k_{t-1}^{\alpha}.$$

Investment, in this case, is $\beta\alpha\theta_t k_{t-1}^{\alpha}$. Therefore, in this economy, the level of the capital stock tends to the value of $(\beta\alpha)^{1/(1-\alpha)}$ for small values of σ_ϵ.

To obtain the finite element approximation of the consumption function, I need to specify the model and algorithmic parameters. Let $\beta = 0.95$, $\alpha = 0.33$, $\rho = 0.95$, and $\sigma_\epsilon = 0.1$. This choice implies that ϵ is in the interval $[-0.288, 0.288]$ approximately 99.6 percent of the time and that θ is in the interval $[0.744, 1.345]$ approximately 99.6 percent of the time. I use these estimates to set $[\underline{\epsilon}, \overline{\epsilon}]$ and $[\underline{\theta}, \overline{\theta}]$. For the upper bound on capital, I set $\overline{k} = \overline{\theta}^{\frac{1}{1-\alpha}}$ or 1.56 which is the maximum sustainable capital stock for $\theta = \overline{\theta}$.

To illustrate how the approximation changes as I change the grid, I report results for two partitions of the grid over the capital stocks and technology shocks. I first assume that $\vec{k}$ = [0, 0.01, 0.1, 0.5, 1.0, 1.56] and $\vec{\theta}$ = [0.744, 1, 1.345], which implies that there are 10 elements. The second partition is given by $\vec{k}$ = [0, 0.01, 0.05, 0.1, 0.25, 0.5, 0.75, 1.0, 1.25, 1.56] and $\vec{\theta}$ = [0.744, 0.9, 1, 1.15, 1.345]. In the second case, there are 36 elements. For both partitions, I set the number of quadrature points for integration with respect to the capital stock and the technology shock equal to 3 in each element, i.e., $m_{\theta,e} = 3$, $m_{k,e} = 3$ for all e. For integration over ϵ, I set the number of quadrature points, m_ϵ, equal to 10. In both cases, the initial guess for the consumption function is the linear-quadratic

---

[10] Judd (1992) applies spectral methods to the examples in Taylor and Uhlig (1990). Like the finite element method, spectral methods are used to solve functional equations such as equation (9). The main difference is the function space from which the approximations are chosen. Spectral methods use shape functions which are almost everywhere nonzero. Typically these functions are polynomials. The finite element method uses piecewise shape functions. Because it does not require that the true function be smooth and have small high-order derivatives, the finite element method is more general than the spectral methods. Furthermore, because the shape functions are zero on a large portion of the domain, the Jacobian matrix in (13) is very sparse. Thus, methods for solving sparse linear systems can be used. This results in a dramatic reduction in computational time, especially when there are a large number of state variables.

approximation, i.e.,

$$c_a = \theta_a k_a^\alpha - (0.119 + 0.33 k_a + 0.177 \ln(\theta_a)),$$

for all a such that $k_a > 0$, where k_a is the level of capital stock at node a and θ_a is the technology shock at node a. I set c_a equal to 0 at all nodes a on the $k = 0$ boundary. I assume that the iterations in (13) are converged when $||\vec{c}_{\ell+1} - \vec{c}_\ell|| < 10^{-5}$ where $||\vec{x}||$ is the vector norm equal to $(\sum_{i=1}^n x_i^2)^{\frac{1}{2}}/n$.

Starting from the linear-quadratic solution, the computation of the 10-element approximation takes approximately 0.075 seconds and convergence is achieved in 4 steps of the Newton-Raphson algorithm.[11] The per-iteration computation time (excluding input-output overhead) is approximately 0.016 seconds. In the 36-element case, convergence is achieved in 4 iterations of (13) with 0.32 seconds for the entire computation. The per-iteration computation time is approximately 0.08 seconds. In figure 6, I plot the exact solution, the finite element approximation with 10 elements, and the finite approximation with 36 elements for $\theta = 1.15$. The exact solution is given by the solid line, the 36-element approximation is given by the dashed line and the 10-element approximation is given by the scatter plot with '+'. Notice that in both cases, it is difficult to distinguish the approximation from the exact solution.

3.4. An example from Taylor and Uhlig (1990)

In the examples of Taylor and Uhlig (1990), the rate of depreciation is equal to zero (i.e., $\delta = 0$) and no analytical solutions exist. In this section, I consider their 'case 2' which has $\delta = 0$, $\tau = 1.5$, $\beta = 0.95$, $\alpha = 0.33$, $\rho = 0.95$, and $\sigma_\epsilon = 0.1$. I assume that the upper bound for the capital stock is $\overline{k} = 25$. In the interval [0,5], the partition is given by [0, .01, .05, .13, .29, .51, 1.15, 2.43, 5], which puts a cluster of points where the gradient is large. Nodes between 5 and 25 are equally spaced with partition length equal to 2.5. Therefore, the number of points for the grid on k is 17. The partition for the technology shock is given by $\vec{\theta} = [.4, .7, 1, 1.3, 1.6]$. I set the number of quadrature points for integration with respect to the capital stock and the technology shock equal to 3 in each element, i.e., $m_{\theta,e} = 3$, $m_{k,e} = 3$ for all e. For integration over ϵ, I set the number of quadrature points, m_ϵ, equal to 10. For the initial consumption function, I use $.14(\theta k^\alpha + k - \delta k)$. This initial

[11] All computation times reported in this paper are based on runs of Fortran code on a Silicon Graphics Indigo R-4000.

guess assumes that a constant fraction of output is used for consumption and a constant fraction is used for the purchase of new capital. The fraction 0.14 is chosen because it implies that the correct marginal propensity to consume when σ_ϵ is small. I assume that the iterations in (13) are converged when $||\vec{c}_{\ell+1} - \vec{c}_\ell|| < 10^{-5}$ where $||\vec{x}||$ is the vector norm equal to $(\sum_{i=1}^n x_i^2)^{\frac{1}{2}}/n$.

In figure 7, I plot both the finite element approximation and the approximation from Coleman's (1990) method which is reported in Taylor and Uhlig (1990). Coleman's approximation is given by the points marked '+'. Each curve is a finite-element approximation for consumption as a function of capital and some fixed level of the technology shock. Values of the technology shock are displayed next to the curves. Notice that each of these curves coincides with the solution of Coleman. Taylor and Uhlig (1990) report a computational time of 110 seconds for Coleman who ran a Fortran code on an Amdahl 5890-300. For the finite element approximation, the computational time is 1.32 seconds on the SGI R-4000 with 7 iterations of (13). The per-iteration computation time is approximately 0.18 seconds.

3.5. An example with inequality constraints

If there are inequality constraints that bind for certain values of the capital stock and the technology shock, the algorithm as described in section 3.3 will not enforce the constraints. Suppose for example, that investment cannot fall below zero. Then, the solution to the problem of section 3.1 must satisfy

$$c(k_{t-1}, \theta_t) \leq \theta_t k_{t-1}^\alpha. \tag{14}$$

Can we modify the problem or the algorithm so that the solution satisfies (14)?

The approach that I take here is to modify the problem. In particular, I replace the objective function of (8) by

$$E\Big[\sum_{t=0}^{\infty} \beta^t \{\frac{c_t^{1-\tau}}{1-\tau} - \gamma[(c_t - \theta_t k_{t-1}^\alpha)^3 + |(c_t - \theta_t k_{t-1}^\alpha)^3|]\}\big|k_{-1}\Big], \qquad 0 < \beta < 1, \ \tau > 0. \tag{8$'$}$$

Notice that I have included a penalty function of the form $x^3 + |x^3|$ which is equal to 0 for values of x less than 0 and $2x^3$ for nonnegative values of x.[12] If the constraint is violated

[12] See Fletcher (1987) for more details on penalty functions.

and investment is negative, then there is a loss in utility. The larger is consumption relative to income, the larger is the penalty.

The size of the penalty is determined by the value of the parameter γ. To compute the optimal decision function, I solve a sequence of optimization problems, each indexed by a different penalty parameter. In other words, I first choose a sequence $\gamma^{(j)}$, such as $\{1, 10, 10^2, 10^3, \ldots\}$, which has $\gamma^{(j)} \to \infty$. For each parameter in the sequence, I calculate the finite element approximation. I stop when the constraint in (14) is approximately satisfied for all values of capital and technology, e.g., $c(k,\theta) - \theta k^{\alpha} < \xi$, for all k and θ, where ξ is a tolerance parameter.

There are two main advantages of the penalty functions over the method that is applied by Christiano and Fisher (1993). First, with the approach used here, I do not compute the points at which the constraint binds. Christiano and Fisher (1993) assume that at some level of the capital stock, $\tilde{k}$, the constraint is binding and it is binding at all levels of the stock above $\tilde{k}$. In computing the consumption function, they use the fact that some $\tilde{k}$ exists and impose the constraint in (14) at all levels of the capital stock above $\tilde{k}$. Thus, they have to find the value of $\tilde{k}$ where the constraint binds. However, in problems with more than one continuous state variable, it is difficult to keep track of the regions where the constraints bind. The second advantage of the approach used here is that I do not have to calculate the Lagrange multipliers associated with the constraints in (14).

For an example, consider the parameterization of the model used by Christiano and Fisher (1993). In their case, they assume that $\beta = 1.03^{-.25}$, $\delta = 0.02$, $\alpha = 0.3$, $\tau = 1$, and that θ_t is independently and identically distributed. The technology shock at t, θ_t, takes on the value $\exp(.22)$ with probability 1/2 and $\exp(-.22)$ with probability 1/2. In figure 8, I plot the finite element approximations for the investment function (i.e., $i(k,\theta) = \theta k^{\alpha} - c(k,\theta)$) for the unconstrained and constrained problems. Notice that the constrained and unconstrained solutions are approximately the same at levels of capital stocks where the constraint doesn't bind. For the constrained problem, equation (14) is satisfied to within 5×10^{-5}. For both the constrained and unconstrained problems, the interval for the capital stock is taken to be [15,51] with subintervals of length 1. Figure 8 should be compared to figure 8 of Christiano and Fisher. Both figures show the constraint binding around $k = 34$. Both have the constrained investment equation for $\theta = \exp(.22)$ lying below the unconstrained equation. If plotted together, it would be difficult to distinguish the solution reported here from that reported in Christiano and Fisher.

4. Conclusion

This paper describes the finite element method by way of several examples. I show that the method is easy to apply and, for examples such as the stochastic growth method, gives accurate solutions within a second or two on a desktop computer. I also show how inequality constraints can be handled by redefining the optimization problem with penalty functions.

References

Braun, R. Anton and Ellen R. McGrattan. 1993. "The Macroeconomics of War and Peace." forthcoming in *NBER Macroeconomics Annual.*

Christiano, Lawrence J. and Jonas D. Fisher. 1993. "Prototype algorithms for solving the one sector stochastic growth model when gross investment is constrained to be non-negative," mimeo, Northwestern University.

Coleman, Wilber J., III. 1990. "Solving the stochastic growth model by policy-function iteration," *Journal of Business and Economic Statistics*, 8:27-30.

Fletcher, Roger. 1987. *Practical Methods of Optimization.* New York: Wiley and Sons.

Hughes, Thomas J.R. 1987. *The Finite Element Method.* New Jersey: Prentice Hall.

Judd, Kenneth L. 1992. "Projection methods for solving aggregate growth models." *Journal of Economic Theory*, 58:410-452.

Kydland, Finn and Edward C. Prescott. 1982. Time to build and aggregate fluctuations, Econometrica 50, 1345-1370.

Press, W.H., B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling. 1986. *Numerical Recipes: The art of scientific computing.*

Reddy, J.N. 1993. *An Introduction to the Finite Element Method.* New York: McGraw-Hill.

Taylor, John B. and Harold Uhlig. 1990. "Solving nonlinear stochastic growth models: a comparison of alternative solution methods." *Journal of Business and Economic Statistics*, 8:1-18.

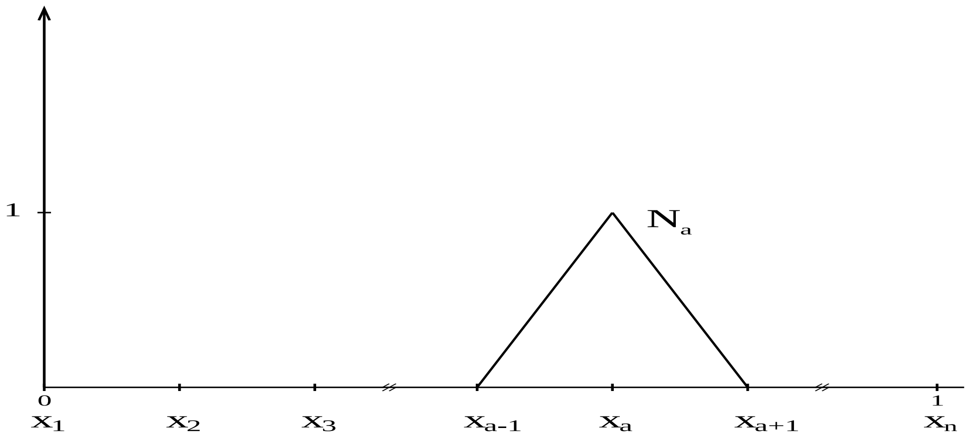

Figure 1. Shape function for node a, $N_a(x)$.

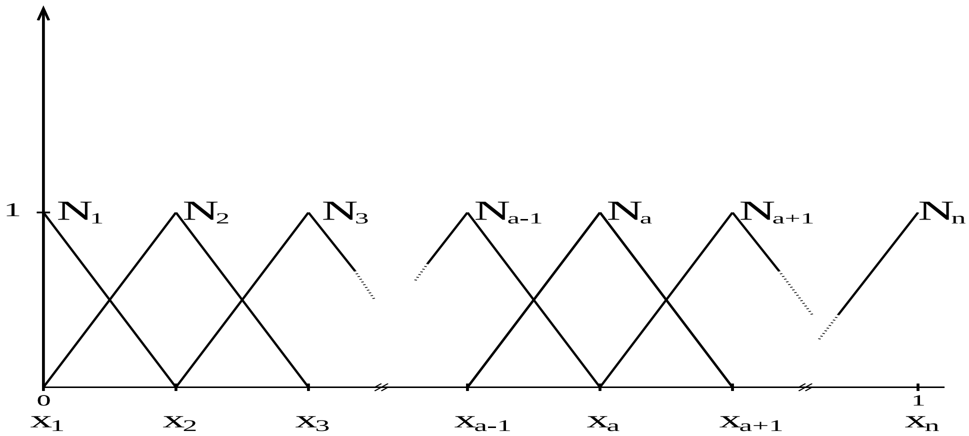

Figure 2. All of the shape functions, $N_a(x)$, $a = 1, \ldots n$.

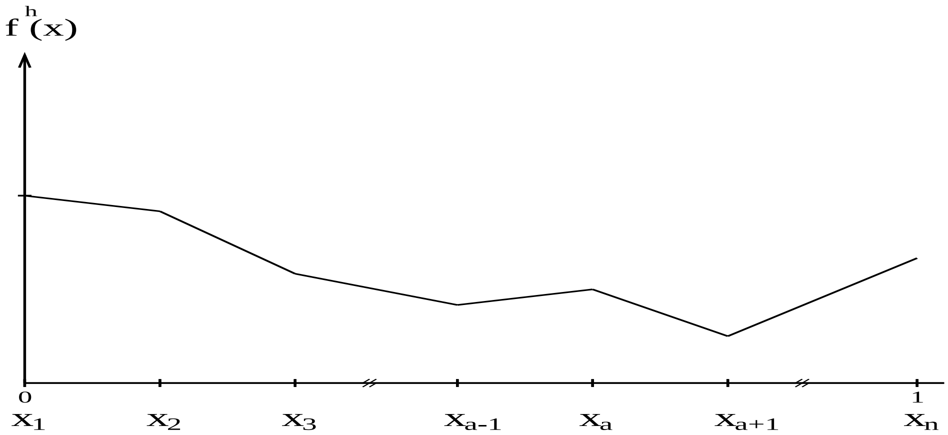

Figure 3. A piecewise linear function, $f^h(x)$.

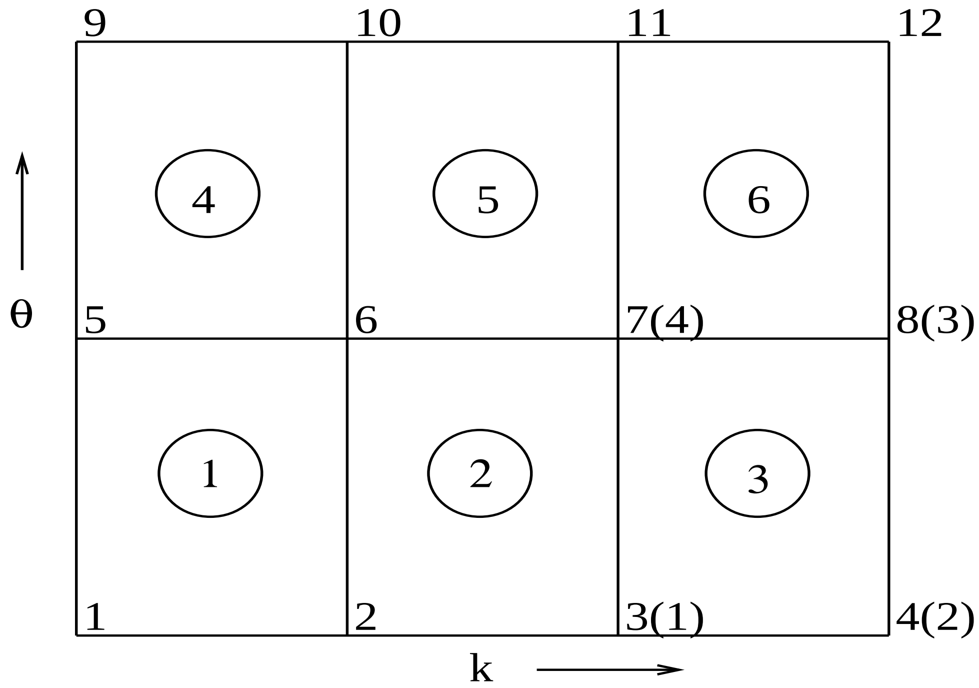

Figure 4. Mesh of rectangular elements (with element numbers circled and local node numbers for element 3 in parentheses).

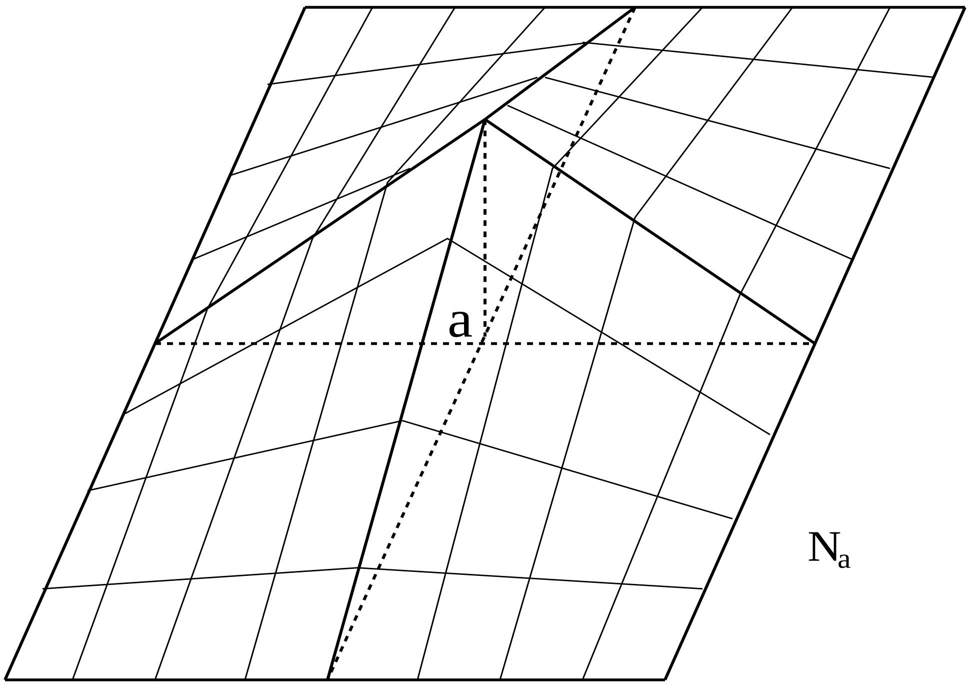

Figure 5. Shape function for node a, $N_a(k, \theta)$.

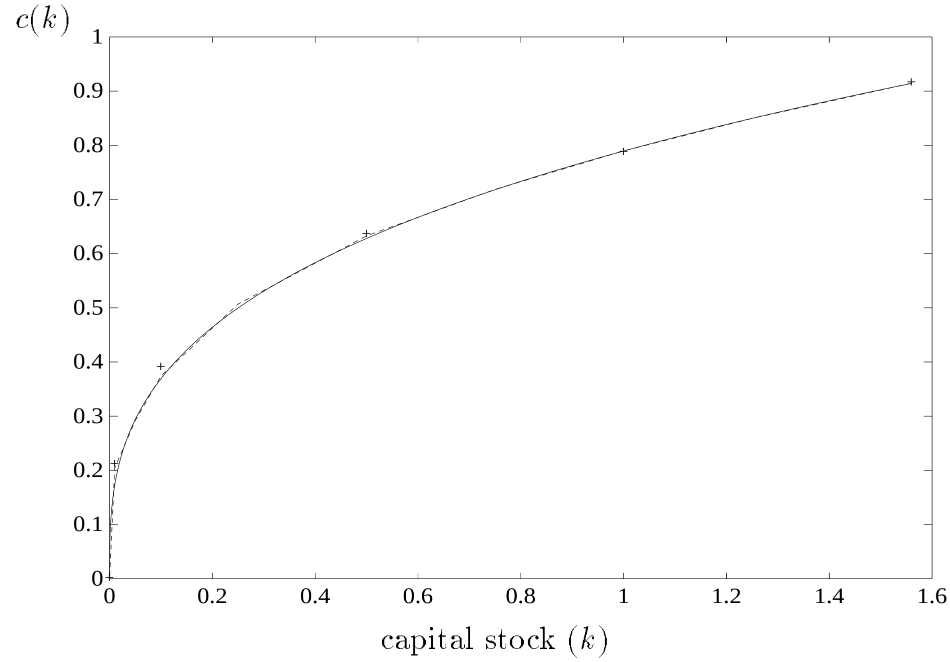

Figure 6. Decision functions for test case. (The solid line is the exact solution, the dashed line is the 36-element approximation, and the points marked '+' are the 10-element approximation.)

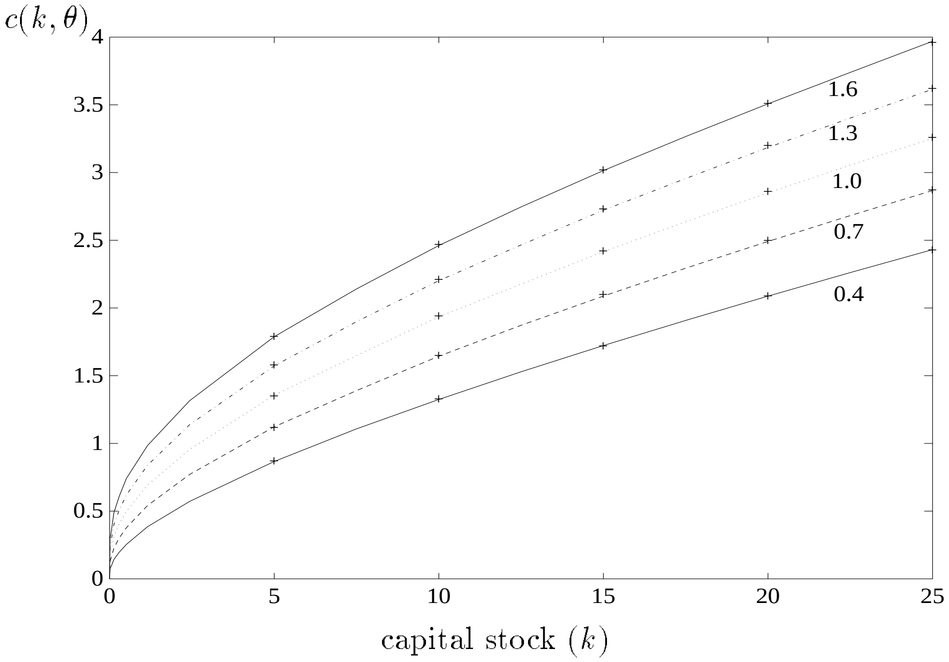

Figure 7. Consumption functions for the stochastic growth model. (Each line corresponds to a different value of θ. Points marked by '+' are Coleman's (1990) solution.)

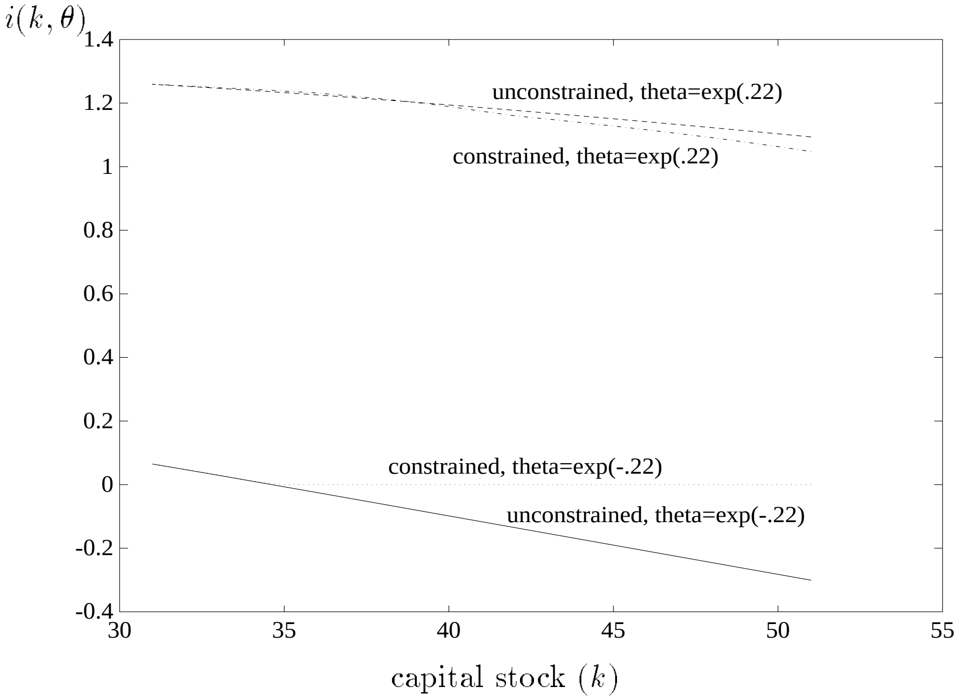

Figure 8. Investment functions for constrained and unconstrained problems.