Chapter 17 Notes
Solving Rational Expectations Models

Kenneth L. Judd
Hoover Institution

November 20, 2002
The Lucas Asset Pricing Model

- **Model:**
  - A single asset paying dividends, $y_{t+1}$:
    \[ y_{t+1} = \rho y_t + \varepsilon_{t+1}, \]  
    \[ \text{(17.1.1)} \]
    where $\varepsilon_t$ is an i.i.d. innovation process.
  - Representative agent consumes only dividends and has utility function $E \left\{ \sum_{t=0}^{\infty} \beta^t u(c_t) \right\}$.
  - $p_t$ price of one unit of asset at time $t$
  - Supply is normalized to be one.
  - Equilibrium price process must satisfy
    \[ u'(y_t)p_t = \beta E \left\{ u'(y_{t+1}) (y_{t+1} + p_{t+1}) \mid y_t \right\}. \]  
    \[ \text{(17.1.2)} \]

- **Recursive solution**
  - Let $p(y)$ be the ex-dividend price of a share when the dividend is $y$
  - Lucas (1978) shows that $p(y)$ solves
    \[ u'(y_t)p(y_t) = \beta E \left\{ u'(y_{t+1}) (y_{t+1} + p(y_{t+1})) \mid y_t \right\}. \]  
    \[ \text{(17.1.3)} \]
  - Rewrite (17.1.3) as
    \[ p(y_t) = \beta E \left\{ \frac{u'(y_{t+1})}{u'(y_t)} y_{t+1} \mid y_t \right\} + \beta E \left\{ \frac{u'(y_{t+1})}{u'(y_t)} p(y_{t+1}) \mid y_t \right\}. \]  
    \[ \text{(17.1.4)} \]
  - (17.1.4) is a linear Fredholm integral equation
    \[ p(y) = g(y) + \beta \int K(y, z) p(z) \, dz, \]  
    \[ \text{(17.1.5)} \]
    where
    \[ g(y) = \beta \int \frac{u'(z)}{u'(y)} z q(z \mid y) \, dz, \]
    \[ K(y, z) = \frac{u'(z)}{u'(y)} q(z \mid y), \]
    and $q(z \mid y)$ is the density of the time $t + 1$ dividend conditional on the time $t$ dividend.
Solution by Simulation

• Present value arguments show that $p(y)$ is

$$p(y_0) = \left(u'(y_0)\right)^{-1}E \left\{ \sum_{t=1}^{\infty} \beta^t u'(y_t) \right| y_0 \right\}. \quad (17.1.6)$$

• Sampling scheme:

  - Draw a sequence of $T$ i.i.d. innovations, $\varepsilon_t$
  - Form dividend sequence $y_t, t = 1, \cdots T$, implied by the process (17.1.1) with $y_0$ given.
  - Compute the sum

$$P(y_0; \varepsilon) = \left(u'(y_0)\right)^{-1} \sum_{t=1}^{T} \beta^t u'(y_t). \quad (17.1.7)$$

  - Repeat this for several draws of $\varepsilon \in R^T$; denote the $j$’th sequence as $\varepsilon^j, j = 1, \ldots, m$.
  - Each $\varepsilon^j$ sequence implies a different dividend sequence and a different value for $P(y_0; \varepsilon^j)$.
  - Estimate of $p(y_0)$, denoted $\hat{p}(y_0)$, is the average of the $P(y_0; \varepsilon)$:

$$\hat{p}(y_0) = \frac{1}{m} \sum_{j=1}^{m} P(y_0; \varepsilon^j). \quad (17.1.8)$$

  - Repeat (17.1.8) for $N$ values of $y_0$, denoted $y_0^i, i = 1, \ldots, N$

• Simulation methods

  - Dominated by other methods if $y$ process has low dimension
  - Are appropriate if desired accuracy is moderate and dividend process is complicated. For example, if

$$y_t = a_0 + a_1 y_{t-1} + a_2 y_{t-2} + \cdots + a_\ell y_{t-\ell} + \varepsilon_t,$$

process is $\ell$-dimensional and would challenge other methods.
Lucas Model: Discrete-State Approximation

- Tauchen (1991) recognized that (17.1.3) is a linear Fredholm integral equation of the second kind, and applied the procedures similar to those described in Section 10.8.

- Use Gauss-Hermite quadrature nodes $y_i$ and weights $\omega_i$.

- Prices $p(y_i)$, satisfy the linear equation

\[
p(y_i) = g(y_i) + \beta \sum_{j=1}^{n} p(y_j) \frac{u'(y_j)}{u'(y_i)} \omega_j \frac{q(y_j \mid y_i)}{w(y_j)}. \tag{17.1.9}
\]

We solve (17.1.9) to compute approximations to $p(y)$ for $y_i \in Y$; call the solutions $\hat{p}_i$. The procedure so far just approximates $p(y)$ on $Y$.

- To approximate $p$ globally
  - Use Nystrom extension, equation (10.8.10), to get

\[
\hat{p}(y) = g(y) + \beta \sum_{j=1}^{n} \hat{p}_j \frac{u'(y_j)}{u'(y)} \omega_j \frac{q(y_j \mid y)}{w(y_j)}, \tag{17.1.10}
\]

which is defined for all $y$.

- This is an example of projection method with Gauss-Hermite polynomials and orthogonal collocation.
Example: Stochastic Dynamic General Equilibrium

- Model

\[
\max_{c_t} E \left\{ \sum_{t=1}^{\infty} \beta^t u(c_t) \right\}
\]

\[
k_{t+1} = \theta_t f(k_t) - c_t
\]

\[
\ln \theta_{t+1} = \rho \ln \theta_t + \varepsilon_t
\]

- Euler equation

\[
u'(c_t) = \beta E \{ u'(c_{t+1}) \theta_{t+1} f'(k_{t+1}) | \theta \} \]

Consumption is determined by recursive function

\[
c_t = C(k_t, \theta_t)
\]

- \(C(k, \theta)\) satisfies functional equation

\[
0 = u'(C(k, \theta)) - \beta E \left\{ u' \left( C \left( \theta f(k) - C(k, \theta), \tilde{\theta} \right) \right) \times \tilde{\theta} f'(\theta f(k) - C(k, \theta)) | \theta \right\}
\]

- Transform Euler equation into the more linear form

\[
0 = C(k, \theta) - (u')^{-1} \left( \beta E \left\{ u' \left( C(\theta f(k) - C(k, \theta), \tilde{\theta}) \right) \times \tilde{\theta} f'(\theta f(k) - C(k, \theta)) | \theta \right\} \right) \equiv \mathcal{N}(C)(k, \theta)
\]
• Approximate policy function

$$\tilde{C}(k, \theta; \mathbf{a}) = \sum_{i=1}^{n_k} \sum_{j=1}^{n_\theta} a_{ij} \psi_{ij}(k, \theta)$$

$$\psi_{ij}(k, \theta) \equiv T_{i-1} \left( \frac{2k - k_m}{k_M - k_m} - 1 \right) T_{j-1} \left( \frac{2\theta - \theta_m}{\theta_M - \theta_m} - 1 \right)$$

• Define integrand of expectations

$$I(k, \theta, \mathbf{a}, z) = u' \left( \tilde{C} \left( \theta f(k) - \tilde{C}(k, \theta; \mathbf{a}), e^{\sigma z} \theta^\rho, \mathbf{a} \right) \right) \times e^{\sigma z} \theta^\rho f'(\theta f(k) - \tilde{C}(k, \theta; \mathbf{a})) \pi^{-\frac{1}{2}}$$

• $\mathcal{N} \left( \tilde{C}(\cdot, \cdot; \mathbf{a}) \right) (k, \theta)$ becomes

$$\tilde{C}(k, \theta; \mathbf{a}) - (u')^{-1} \left( \beta \int_{-\infty}^{\infty} I(k, \theta; \mathbf{a}, z) \frac{e^{-z^2/2}}{\sqrt{2\pi}} dz \right)$$

• Use Gauss-Hermite quadrature over $z$:

$$\int_{-\infty}^{\infty} I(k, \theta, \mathbf{a}, z) \frac{e^{-z^2/2}}{\sqrt{2}} dz = \sum_{j=1}^{m_z} I \left( k, \theta, \mathbf{a}, \sqrt{2}z_j \right) \omega_j$$

where $\omega_j, z_j$ are Gauss-Hermite quadrature weights and points.

• The computable residual function is

$$R(k, \theta; \mathbf{a}) = \tilde{C}(k, \theta; \mathbf{a}) - (u')^{-1} \left( \beta \sum_{j=1}^{m_z} I \left( k, \theta, \mathbf{a}, \sqrt{2}z_j \right) w_j \right)$$

$$\equiv \tilde{N} \left( \tilde{C}(\cdot, \cdot; \mathbf{a}) \right) (k, \theta).$$
• Fitting Criteria:
  
  – Collocation:
    
    * Choose \( n_k \) capital stocks, \( \{k_i\}_{i=1}^{n_k} \), and \( n_\theta \) productivity levels, \( \{\theta_i\}_{j=1}^{n_\theta} \).
    * Find \( \mathbf{a} \) such that
      \[
      R(k_i, \theta_j; \mathbf{a}) = 0, \quad i = 1, \ldots, n_k, \quad j = 1, \ldots, n_\theta
      \]
  
  – Galerkin:
    
    * Compute the \( n_k n_\theta \) projections
      \[
      P_{ij}(\mathbf{a}) \equiv \int_{k_m}^{k_M} \int_{\theta_m}^{\theta_M} R(k, \theta; \mathbf{a}) \psi_{ij}(k, \theta) d\theta dk
      \]
    * Approximate projections by Gauss-Chebyshev quadrature
      \[
      \hat{P}_{ij}(\mathbf{a}) \equiv \sum_{\ell_k=1}^{m_k} \sum_{\ell_\theta=1}^{m_\theta} R(k_{\ell_k}, \theta_{\ell_\theta}; \mathbf{a}) \psi_{ij}(k_{\ell_k}, \theta_{\ell_\theta}),
      \]
      where
      \[
      k_{\ell_\theta} = k_m + \frac{1}{2}(k_M - k_m) \left(z_{\ell_k}^{m_k} + 1\right), \quad \ell_k = 1, \ldots, m_k
      \]
      \[
      \theta_{\ell_\theta} = \theta_m + \frac{1}{2}(\theta_M - \theta_m) \left(z_{\ell_\theta}^{m_\theta} + 1\right), \quad \ell_\theta = 1, \ldots, m_\theta
      \]
      \[
      z_{\ell}^n \equiv \cos \left(\frac{(2i - 1)\pi}{2n}\right), \quad \ell = 1, \ldots, n
      \]
    * Coefficients, \( \mathbf{a} \), are fixed by the system
      \[
      \hat{P}_{ij}(\mathbf{a}) = 0, \quad i = 1, \ldots, n_k, \quad j = 1, \ldots, n_\theta
      \]
• Bounded Rationality Accuracy Measure

  – Consider the computable Euler equation error

\[
E(k, \theta) = \frac{\hat{N}(\hat{C}(\cdot, \cdot; \mathbf{a}))(k, \theta)}{\hat{C}(k, \theta; \mathbf{a})}
\]

where \( \hat{N} \) uses some integration formula for \( E\{\cdot\} \); need not be the same as used in computing \( R(k, \theta; \mathbf{a}) \). In fact, should use better one.

  – Define the \( L^p, 1 \leq p < \infty \), bounded rationality accuracy to be

\[
\log_{10} \| E(k) \|_p
\]

• Verify solution: Accept solution to projection equations, \( \mathbf{a} \), only if it passes tests

  – Check stability

  * For example, there should be positive savings at low \( k \), high \( \theta \)
  * Could simulate capital stock process implied by \( \hat{C}(k, \theta; \mathbf{a}) \) to see if it has a stationary distribution

  – Check Euler equation errors

  * \( E(k, \theta) \) should be moderate for most \( (k, \theta) \) points in \([k_m, k_M] \times [\theta_m, \theta_M]\)
  * \( E(k, \theta) \) should be small for most \( (k, \theta) \) points frequently visited

  – If \( \hat{C}(k, \theta; \mathbf{a}) \) does not pass these tests, go back and use higher values for \( n_k \) and \( n_\theta \), and increase \( m_k \), and \( m_\theta \)
• Numerical Results

- Basis: Chebyshev polynomials
- Initial guess: Linear rule through deterministic steady state and zero.
- \( k \in [.333, 2.000] \)
- Method: Collocation and Galerkin.
- Newton’s method solved projection equations, \( P_i(a) = 0 \), for \( a \).
- Machine: Compaq 386/20 (old, but relative speeds are still valid)
- Speed: Stochastic case: under two minutes for a 60 parameter fit.
- Errors: 2% for 6 parameter fit, .1% for 60 parameter fit – about a penny loss per $10,000 dollar expenditure
- Orth. poly. + orthog. collocation + Gaussian quad. + Newton outperforms naive methods by factor of 10 or greater; exceeded Monte Carlo methods by factor of 100+.
- \( \hat{C}(k, \theta; a) \) is an \( \varepsilon \)-equilibrium with small \( \varepsilon \) – a bounded rationality interpretation.
Table 17.1: $\log_{10}$ Euler Equation Errors

<table>
<thead>
<tr>
<th>$\gamma$</th>
<th>$\rho$</th>
<th>$\sigma$</th>
<th>$|E|_\infty$</th>
<th>$|E|_1$</th>
<th>$|E|_\infty$</th>
<th>$|E|_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>-15.00</td>
<td>0.80</td>
<td>0.01</td>
<td>-2.13</td>
<td>-2.80</td>
<td>-3.00</td>
<td>-3.83</td>
</tr>
<tr>
<td>-15.00</td>
<td>0.80</td>
<td>0.04</td>
<td>-1.89</td>
<td>-2.54</td>
<td>-2.44</td>
<td>-2.87</td>
</tr>
<tr>
<td>-15.00</td>
<td>0.30</td>
<td>0.04</td>
<td>-2.13</td>
<td>-2.80</td>
<td>-2.97</td>
<td>-3.83</td>
</tr>
<tr>
<td>-0.10</td>
<td>0.80</td>
<td>0.01</td>
<td>-0.01</td>
<td>-1.22</td>
<td>-1.68</td>
<td>-2.65</td>
</tr>
<tr>
<td>-0.10</td>
<td>0.80</td>
<td>0.04</td>
<td>0.01</td>
<td>-1.19</td>
<td>-1.48</td>
<td>-2.22</td>
</tr>
<tr>
<td>-0.10</td>
<td>0.30</td>
<td>0.04</td>
<td>0.18</td>
<td>-1.22</td>
<td>-1.63</td>
<td>-2.65</td>
</tr>
<tr>
<td>-0.10</td>
<td>0.80</td>
<td>0.01</td>
<td>-4.28</td>
<td>-5.19</td>
<td>-4.43</td>
<td>-5.18</td>
</tr>
<tr>
<td>-0.10</td>
<td>0.80</td>
<td>0.04</td>
<td>-3.36</td>
<td>-4.00</td>
<td>-3.30</td>
<td>-3.95</td>
</tr>
<tr>
<td>-0.10</td>
<td>0.30</td>
<td>0.04</td>
<td>-4.24</td>
<td>-5.19</td>
<td>-4.38</td>
<td>-5.18</td>
</tr>
<tr>
<td>-0.10</td>
<td>0.80</td>
<td>0.01</td>
<td>-3.40</td>
<td>-4.37</td>
<td>-3.47</td>
<td>-4.39</td>
</tr>
<tr>
<td>-0.10</td>
<td>0.80</td>
<td>0.04</td>
<td>-2.50</td>
<td>-3.22</td>
<td>-2.60</td>
<td>-3.17</td>
</tr>
<tr>
<td>-0.10</td>
<td>0.30</td>
<td>0.04</td>
<td>-3.43</td>
<td>-4.37</td>
<td>-3.49</td>
<td>-4.39</td>
</tr>
<tr>
<td>-0.10</td>
<td>0.80</td>
<td>0.01</td>
<td>-5.48</td>
<td>-6.43</td>
<td>-5.61</td>
<td>-6.42</td>
</tr>
<tr>
<td>-0.10</td>
<td>0.80</td>
<td>0.04</td>
<td>-3.81</td>
<td>-4.38</td>
<td>-3.88</td>
<td>-4.37</td>
</tr>
<tr>
<td>-0.10</td>
<td>0.30</td>
<td>0.04</td>
<td>-5.45</td>
<td>-6.43</td>
<td>-5.57</td>
<td>-6.42</td>
</tr>
<tr>
<td>-0.10</td>
<td>0.80</td>
<td>0.01</td>
<td>-5.09</td>
<td>-6.12</td>
<td>-5.17</td>
<td>-6.15</td>
</tr>
<tr>
<td>-0.10</td>
<td>0.80</td>
<td>0.04</td>
<td>-2.99</td>
<td>-3.68</td>
<td>-3.09</td>
<td>-3.64</td>
</tr>
<tr>
<td>-0.10</td>
<td>0.30</td>
<td>0.04</td>
<td>-5.17</td>
<td>-6.12</td>
<td>-5.23</td>
<td>-6.14</td>
</tr>
</tbody>
</table>

*($n_k$, $n_\theta$, $m_k$, $m_\theta$)
Table 17.2: Alternative Implementations

\[ n_k = 7, n_\theta = 5, m_k = 7, m_\theta = 5 \]

<table>
<thead>
<tr>
<th>( \gamma )</th>
<th>( \beta )</th>
<th>( \sigma )</th>
<th>( G^a )</th>
<th>( P^b )</th>
<th>( U^c )</th>
<th>( UP^d )</th>
</tr>
</thead>
<tbody>
<tr>
<td>error</td>
<td>time</td>
<td>error</td>
<td>time</td>
<td>error</td>
<td>time</td>
<td>error</td>
</tr>
<tr>
<td>.3 .01</td>
<td>-4.35</td>
<td>:11</td>
<td>-4.35</td>
<td>:52</td>
<td>-4.07</td>
<td>:08</td>
</tr>
<tr>
<td>.3 .01</td>
<td>-4.03</td>
<td>:07</td>
<td>-4.03</td>
<td>:30</td>
<td>-3.76</td>
<td>:07</td>
</tr>
<tr>
<td>-.1 .8 .04</td>
<td>-2.50</td>
<td>:07</td>
<td>-2.50</td>
<td>:41</td>
<td>-2.52</td>
<td>:06</td>
</tr>
<tr>
<td>.3 .01</td>
<td>-3.42</td>
<td>:08</td>
<td>-3.42</td>
<td>1:30</td>
<td>-3.18</td>
<td>:07</td>
</tr>
</tbody>
</table>

\[ n_k = 10, n_\theta = 6, m_k = 25, m_\theta = 15 \]

<table>
<thead>
<tr>
<th>( \gamma )</th>
<th>( \beta )</th>
<th>( \sigma )</th>
<th>( G^a )</th>
<th>( P^b )</th>
<th>( U^c )</th>
<th>( UP^d )</th>
</tr>
</thead>
<tbody>
<tr>
<td>error</td>
<td>time</td>
<td>error</td>
<td>time</td>
<td>error</td>
<td>time</td>
<td>error</td>
</tr>
<tr>
<td>.3 .01</td>
<td>-5.68</td>
<td>2:19</td>
<td>-5.14</td>
<td>11:31</td>
<td>-5.49</td>
<td>2:14</td>
</tr>
<tr>
<td>-.9 .8 .04</td>
<td>-4.00</td>
<td>1:31</td>
<td>-4.00</td>
<td>5:17</td>
<td>-4.01</td>
<td>1:31</td>
</tr>
<tr>
<td>.3 .01</td>
<td>-5.40</td>
<td>1:23</td>
<td>-4.63</td>
<td>7:13</td>
<td>-5.25</td>
<td>1:20</td>
</tr>
<tr>
<td>.3 .01</td>
<td>-5.27</td>
<td>1:32</td>
<td>-4.02</td>
<td>7:25</td>
<td>-5.09</td>
<td>1:27</td>
</tr>
</tbody>
</table>

\[ ^a \text{Chebyshev polynomial basis, Chebyshev zeroes used in evaluating fit} \]
\[ ^b \text{Ordinary polynomial basis, Chebyshev zeroes used in evaluating fit} \]
\[ ^c \text{Chebyshev polynomial basis, uniform grid points} \]
\[ ^d \text{Ordinary polynomial basis, uniform grid points} \]
\[ ^e \text{error measure is } \| E(k) \|_\infty \]
Table 17.3: Tensor Product vs. Complete Polynomials\(^a\)

<table>
<thead>
<tr>
<th>(\gamma)</th>
<th>(\rho)</th>
<th>(\sigma)</th>
<th>Tensor Product (n = 3)</th>
<th>Tensor Product (n = 6)</th>
<th>Tensor Product (n = 10)</th>
<th>Complete Polynomials (n = 3)</th>
<th>Complete Polynomials (n = 6)</th>
<th>Complete Polynomials (n = 10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>-15.0</td>
<td>.8</td>
<td>.04</td>
<td>-2.34(^b)</td>
<td>-3.26</td>
<td>-3.48</td>
<td>-1.89</td>
<td>-3.10</td>
<td>-4.06</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>:01(^c)</td>
<td>:13</td>
<td>14:21</td>
<td>:03</td>
<td>:07</td>
<td>1:09</td>
</tr>
<tr>
<td>-0.9</td>
<td>.3</td>
<td>.10</td>
<td>-2.19</td>
<td>-3.60</td>
<td>-5.27</td>
<td>-2.14</td>
<td>-3.55</td>
<td>-5.22</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>:01</td>
<td>:08</td>
<td>1:21</td>
<td>:01</td>
<td>:05</td>
<td>:32</td>
</tr>
<tr>
<td>-0.1</td>
<td>.3</td>
<td>.01</td>
<td>-1.00</td>
<td>-2.84</td>
<td>-5.21</td>
<td>-0.99</td>
<td>-2.83</td>
<td>-5.17</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>:01</td>
<td>:08</td>
<td>1:24</td>
<td>:01</td>
<td>:05</td>
<td>:35</td>
</tr>
</tbody>
</table>

\(^a\) Tensor product cases used orthogonal collocation with \(n_k = n_\theta = m_k = m_\theta = n\) to identify the \(n^2\) free parameters. Complete polynomial cases used Galerkin projections to identify the \(1 + n + n(n + 1)/2\) free parameters.

\(^b\) \(\log_{10} \parallel E \parallel_\infty\)

\(^c\) Computation time expressed in minutes :seconds.

- General Observations:
  - Tensor product of degree \(n\) takes more time, but achieves higher accuracy
  - For a specific level of accuracy, complete polynomial method is faster
Examples: Multiagent Dynamic General Equilibrium

• Model:
  - $n$ types of agents, utility functions, $u_i(c), i = 1, 2, \ldots, n$,
  - Common discount factor $\beta$.
  - Equity is the only asset
  - $c_i = C_i(k)$, wealth distribution is $k = (k_1, k_2, \ldots, k_n)$

• Approximate $c_i = \hat{C}_i(k, \theta; a)$.

• Euler equation for type $i = 1, 2, \ldots, n$

\[
R_i(k, \theta, C) = u_i'(C_i(k, \theta)) - \beta E \{u'(C_i(Y_i(k, \theta) - C(k, \theta), \tilde{\theta})) \\
\times F_k(Y_i(k, \theta) - C(k, \theta), \tilde{\theta}) \mid \theta\}
\]

where

\[
Y_i(k, \theta) = k_i F_1(k, \theta) + w(k, \theta), \; i = 1, \ldots, n
\]

\[
w(k, \theta) = F(k, \theta) - k F_1(k, \theta)
\]

\[
k = \sum_i k_i.
\]
• Approximate residual function for agents of type $i = 1, 2, ..., n$
\[
\hat{R}_i(k, \theta, \hat{C}(\cdot; a)) = \hat{C}_i(k, \theta; a) - (u_i')^{-1} \left( \beta \hat{E} \left\{ u_i' (c^+) F_k \left( k^+, \hat{\theta} \right) | \theta \right\} \right) \\
\]
\[
c_i^+ \equiv \hat{C}_i(y^+, \hat{\theta}; a) \\
k^+ \equiv Y(k, \theta; a) - \hat{C}(k, \theta; a)
\]

where $\hat{E}$ is a numerical approximation of the integral. Use product Gaussian quadrature

• Identifying projections are
\[
P_{ij}(a) \equiv \int_{\theta_m}^{\theta_M} \int_{k_m}^{k_M} \cdots \int_{k_m}^{k_M} \hat{R}_i(k, \theta, \hat{C}(\cdot; a)) \psi_j(k, \theta) w(k, \theta) dk_1 \cdots dk_n d\theta
\]
where $i = 1, ..., n$, and $j = 1, ..., m$.

• Let $\hat{P}(a)$ denote a numerical integration approximation of $P(a)$; we will use product Gaussian quadrature

• Solution chooses $a$ so that $\hat{P}(a) = 0$. 
Representation: Tensor vs. Complete Polynomials

- Tensor method:
  \[
  \hat{C}_i(k, \theta; a) = \sum_{j_1=0}^{n_k} \cdots \sum_{j_n=0}^{n_k} \sum_{\ell=0}^{n_\theta} a_{j_1 \ldots j_n \ell}^i \varphi_i(k_1) \cdots \varphi_n(k_n) \psi_\ell(\theta), \quad i = 1, \ldots, n
  \]
  where \( \varphi_i(k_j) \) (\( \psi_\ell(\theta) \)) is a degree \( i - 1 \) (\( \ell - 1 \)) polynomial in \( k_j(\theta) \) from some orthogonal family.

- Complete polynomial method
  \[
  C_i(k, \theta; a) = \sum_{0 \leq j_1 + \cdots + j_n + \ell \leq d} a_{j_1 \ldots j_n \ell}^i \varphi_{j_1}(k_1) \cdots \varphi_{j_n}(k_n) \psi_\ell(\theta)
  \]

- Number of unknown coefficients are far smaller in complete poly case, but not as flexible.
Solution Methods

• Successive Approximation: at grid of \((k, \theta)\) points (e.g., Chebyshev zeroes) and given iteration \(j\) for \(a\) (denoted \(a^j\)), \(\hat{C}_i(k, \theta; a^j)\), generate data

\[
\hat{C}_i(k, \theta; a^{j+1}) = (u')^{-1} \left( \beta \hat{E} \left\{ u' \left( \hat{C}_i \left( Y(k, \theta) - \hat{C}_i(k, \theta; a^j), \tilde{\theta}; a^j \right) \right) \right. \right.
\times F_k \left. \left( Y(k, \theta) - \hat{C}_i(k, \theta; a^j), \tilde{\theta} \right) \bigg| \theta \right\} \]  

and set coefficients \(a^{j+1}\) through interpolation or regression

• Time Iteration: same procedure except not generate data for \(\hat{C}_i(k, \theta; a^{j+1})\) by solving

\[
\hat{C}_i(k, \theta; a^{j+1}) = (u')^{-1} \left( \beta \hat{E} \left\{ u' \left( \hat{C}_i \left( Y(k, \theta) - \hat{C}_i(k, \theta; a^{j+1}), \tilde{\theta}; a^{j+1} \right) \right) \right. \right.
\times F_k \left. \left( Y(k, \theta) - \hat{C}_i(k, \theta; a^{j+1}), \tilde{\theta} \right) \bigg| \theta \right\} \]  

• Newton’s Method: just solve nonlinear equations \(\hat{P}(a) = 0\)
Table 5: Time and Accuracy Comparisons

<table>
<thead>
<tr>
<th>agents</th>
<th>$\gamma$</th>
<th>deg</th>
<th>basis</th>
<th>coef's</th>
<th>time</th>
<th>acc'cy</th>
<th>time</th>
<th>accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-2</td>
<td>t</td>
<td>4</td>
<td>:0.05</td>
<td>-2.7</td>
<td>:0.2</td>
<td>-2.7</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>c</td>
<td>3</td>
<td>:0.06</td>
<td>-2.6</td>
<td>:0.4</td>
<td>-2.6</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>t</td>
<td>9</td>
<td>:0.22</td>
<td>-3.4</td>
<td>:01</td>
<td>-3.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>c</td>
<td>6</td>
<td>:0.17</td>
<td>-3.3</td>
<td>:01</td>
<td>-3.3</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>t</td>
<td>16</td>
<td>:0.71</td>
<td>-4.1</td>
<td>:01</td>
<td>-4.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>c</td>
<td>10</td>
<td>:0.49</td>
<td>-4.0</td>
<td>:02</td>
<td>-4.0</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>t</td>
<td>25</td>
<td>:02</td>
<td>-4.8</td>
<td>:02</td>
<td>-4.9</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>c</td>
<td>30</td>
<td>:0.99</td>
<td>-4.7</td>
<td>:03</td>
<td>-4.6</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>-1</td>
<td>t</td>
<td>16</td>
<td>:0.66</td>
<td>-3.1</td>
<td>:01</td>
<td>-3.1</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>c</td>
<td>6</td>
<td>:0.38</td>
<td>-2.7</td>
<td>:01</td>
<td>-2.7</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>t</td>
<td>54</td>
<td>:07</td>
<td>-4.1</td>
<td>:08</td>
<td>-4.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>c</td>
<td>20</td>
<td>:02</td>
<td>-3.4</td>
<td>:06</td>
<td>-3.4</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>t</td>
<td>128</td>
<td>1:22</td>
<td>-5.0</td>
<td>:33</td>
<td>-4.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>c</td>
<td>40</td>
<td>:11</td>
<td>-4.1</td>
<td>:21</td>
<td>-4.1</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>t</td>
<td>250</td>
<td>12:34</td>
<td>-5.9</td>
<td>1:48</td>
<td>-4.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>c</td>
<td>70</td>
<td>:45</td>
<td>-4.8</td>
<td>:56</td>
<td>-4.7</td>
<td></td>
</tr>
</tbody>
</table>

Note: “inf” means infeasible. “h hrs n : m.l” means “h hours n minutes, m.l seconds”.

Table 5: Time and Accuracy Comparisons (Continued)

<table>
<thead>
<tr>
<th>agents</th>
<th>γ</th>
<th>deg</th>
<th>basis</th>
<th>num coef’s</th>
<th>time</th>
<th>accuracy</th>
<th>time</th>
<th>accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>-1</td>
<td>1</td>
<td>t</td>
<td>48</td>
<td>:07</td>
<td>-3.4</td>
<td>:07</td>
<td>-3.4</td>
</tr>
<tr>
<td>-2</td>
<td>c</td>
<td>15</td>
<td>1.48</td>
<td>:05</td>
<td>-2.8</td>
<td>2:11</td>
<td>-2.8</td>
<td></td>
</tr>
<tr>
<td>-3</td>
<td>2</td>
<td>t</td>
<td>243</td>
<td>7:07</td>
<td>-4.6</td>
<td>2:11</td>
<td>-4.5</td>
<td></td>
</tr>
<tr>
<td>c</td>
<td>63</td>
<td>:21</td>
<td>-3.6</td>
<td>:36</td>
<td>-3.6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>t</td>
<td>768</td>
<td>inf</td>
<td>inf</td>
<td>19:57</td>
<td>-4.6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c</td>
<td>105</td>
<td>4:05</td>
<td>-4.3</td>
<td>3:09</td>
<td>-4.3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>t</td>
<td>1875</td>
<td>inf</td>
<td>inf</td>
<td>1 hr 56</td>
<td>-4.6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c</td>
<td>210</td>
<td>46:58</td>
<td>-4.9</td>
<td>12:45</td>
<td>-4.8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>-.5</td>
<td>1</td>
<td>t</td>
<td>128</td>
<td>1:09</td>
<td>-3.5</td>
<td>:33</td>
<td>-3.5</td>
</tr>
<tr>
<td>-1</td>
<td>c</td>
<td>24</td>
<td>:5.10</td>
<td>-2.9</td>
<td>:13</td>
<td>-2.9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-2</td>
<td>2</td>
<td>t</td>
<td>972</td>
<td>inf</td>
<td>inf</td>
<td>24:57</td>
<td>-4.6</td>
<td></td>
</tr>
<tr>
<td>c</td>
<td>84</td>
<td>2:47</td>
<td>-3.7</td>
<td>3:04</td>
<td>-3.7</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>t</td>
<td>4096</td>
<td>inf</td>
<td>inf</td>
<td>7 hr 13</td>
<td>-4.6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c</td>
<td>224</td>
<td>52:11</td>
<td>-4.4</td>
<td>26:01</td>
<td>-4.4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>-.5</td>
<td>1</td>
<td>t</td>
<td>320</td>
<td>8:52</td>
<td>-3.6</td>
<td>2:48</td>
<td>-3.6</td>
</tr>
<tr>
<td>-1</td>
<td>c</td>
<td>35</td>
<td>:17.90</td>
<td>-3.0</td>
<td>:38</td>
<td>-3.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-2</td>
<td>2</td>
<td>t</td>
<td>3645</td>
<td>inf</td>
<td>inf</td>
<td>5 hr 16</td>
<td>-4.6</td>
<td></td>
</tr>
<tr>
<td>-3</td>
<td>c</td>
<td>140</td>
<td>12:18</td>
<td>-3.8</td>
<td>10:18</td>
<td>-3.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-4</td>
<td>3</td>
<td>t</td>
<td>20,480</td>
<td>inf</td>
<td>inf</td>
<td>inf</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c</td>
<td>420</td>
<td>13 hr</td>
<td>-4.5</td>
<td>3 hr 27</td>
<td>-4.5</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note: “inf” means infeasible. “h hrs n : m.l” means “h hours n minutes, m.l seconds”.
<table>
<thead>
<tr>
<th>Method:</th>
<th>Basis:</th>
<th>Solution Method:</th>
<th>Advantages:</th>
<th>Disadvantages:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Taylor</td>
<td>Complete</td>
<td>Eigenvalues, Fast linear eq’ns</td>
<td>Local validity</td>
<td></td>
</tr>
<tr>
<td>Series</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Projection methods</td>
<td>Tensor or complete</td>
<td>Newton Quadratic conv.</td>
<td>Infeasible for large problems</td>
<td></td>
</tr>
<tr>
<td>Tensor or Successive complete approx.</td>
<td>Easy possible Iterations nonconv.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Solving Asymmetric Information Asset Models

• General Problem
  – Different agents have different information
  – Question: how much information is revealed by information?

• Grossman (1976)
  – Find all information revealed by trading
  – Finds no incentive to acquire information
  – Assumed special functional forms
  – Assumed and limited type of assets

• More generally
  – Equilibrium is fully revealing if prices are continuous and states finite
  – Equilibrium is often not fully revealing
  – Need more general models which are tractable
- **A Gamma-Gaussian Model**

  - Investors have \( W \) to invest in two assets
  - Safe asset \( R \)
  - Risky asset \( - \log \tilde{Z} \) is \( N(m, w) \).
  - \( m \sim N(\mu_m, \sigma^2_m) \) and \( w \sim \Gamma(\alpha, \beta) \) (If the variance of \( w \) is zero, we have Grossman’s model.)
  - Type \( i \) informed traders know \( y_i \sim N(m, w) \) plus Gaussian noise, \( i = 1, 2, 3 \).
  - \( \omega_i \) is number of shares held by type \( i = 1, 2, 3 \)
  - First-order conditions: Agent \( i, i = 1, 2, 3 \), knows \( p \) and \( y_i \). His FOC for \( \omega^i(p, y_i) \) is
    \[
    0 = E_{y, Z} \left\{ u'_i(C_i(y, Z))(Z - p(y) R) \mid p, y_i \right\} \tag{17.3.2}
    \]
    \[
    C_i(y, Z) \equiv (W - \omega_i(p(y), y)p(y)) R + \omega_i(p(y), y_i)Z
    \]

- **Equilibrium:**

  - \( \omega^i(p, y_i) \) satisfying FOC
  - Market-clearing: \( p(y) \) satisfying
    \[
    1 = \sum_{i=1,2,3} \omega^i(p(y), y_i)) \]
    for all states \( y \).
Numerical implementation of the conditional expectation:

• Definition of conditional expectation:

\[ Z(X) = \mathbb{E}\{Y \mid X\} \]

if and only if for all continuous functions \( \phi \)

\[
0 = \mathbb{E}\{(Z(X) - Y)\phi(X)\} \\
= \int (Z(X(\omega)) - Y(\omega))\phi(X(\omega))d\omega
\]

• The definition replaces the conditional expectation with an infinite number of unconditional expectation conditions.

• Numerically: We accept \( \hat{Z}(X) \) as an approximation to \( Z(X) \) if

\[
0 = \mathbb{E}\{(\hat{Z}(X) - Y)\phi_i(X)\} \\
= \int (\hat{Z}(X(\omega)) - Y(\omega))\phi_i(X(\omega))d\omega, \; i = 1, \ldots, n
\]

for a finite number of \( \phi_i(\cdot) \) functions.
Numerical Approach

- Parameterize unknown functions
  - $H_i(\cdot)$ denotes the degree $i$ Hermite polynomial
  - price function:
    $$ p(y_1, y_2, y_3) = \sum_{0 \leq j+k+l \leq N_p \atop 0 \leq j,k,\ell \leq N_p} a_{j,k,\ell} H_j(y_1) H_k(y_2) H_\ell(y_3) $$
  - stock demand for a type $i$ investor:
    $$ \omega_i(p, y_i) = \sum_{0 \leq j+k \leq N_\theta} b_{i,j}^j H_j(p) H_k(y_i), \ i = 1, 2, 3 $$

- Goal: determine the $a_{j,k,\ell}$ and $b_{i,j}^j$ coefficients for various $N_\omega$ and $N_p$.

- The first-order-condition for a type $i$ investor
  - Theoretical
    $$ E_{y,Z} \left\{ U'(\tilde{c}_i)(\tilde{Z} - p R) \mid y_i, p \right\} = 0, \ i = 1, 2, 3. $$
  - Numerical approximation
    $$ E_{y,Z} \left\{ U'(\tilde{c}_i)(\tilde{Z} - p(y) R) H_j(p(y)) H_k(y_i) \right\} = 0, \ j,k \geq 0, j+k \leq N_\theta. \tag{3} $$

  * The $(i,j)$ condition says that the (excess return) $\times U'(c_i)$ is uncorrelated with $H_j(p(y)) H_k(y_i)$.
  * Eq’ns in (3) are integrals over $y_1, y_2, y_3, z, m, w$ - six dimensions
• Market clearing

  – Theoretical equilibrium condition: for all states $y$

    $$1 = \sum_{i=1,2,3} \omega_i (p(y), y_i)$$

  – Numerical approximations are

    $$E_y \left\{ \left( \sum_{i=1}^{3} \omega_i (p(y), y_i) - 1 \right) H_j(y_1) H_k(y_2) H_l(y_3) \right\} = 0, \quad (4)$$

    $$j, k, l \geq 0, j + k + l \leq N_p$$

• Both are approximations

  – First-order conditions are satisfied only on average, not in any particular state.

  – Market-clearing does not hold in each state

  – Hope: magnitudes of errors are small and solution is close to true equilibrium.
• Numerical Results
   – Four- and Five-digit accuracy on models where we know results
   – Euler equation errors of 1$ per thousand for all of our models
   – Computation time less than 15 minutes on current machines

• Discretization Comments
   – Cannot discretize state space: generic full revelation
   – Must discretize in spectral domain

• Extensions
   – Endogenous information acquisition
   – Other assets - options
   – Two-period model with first-period volume information used in second period
Summary of Projection Method

- Can be used for problems with unknown functions
- Uses approximation ideas
- Utilizes standard optimization and nonlinear equation solving software
- Can exploit a priori information about problem
- Flexible: users choose from a variety of approximation, integration, and nonlinear equation-solving methods

<table>
<thead>
<tr>
<th>Approximation</th>
<th>Integration</th>
<th>Projections</th>
<th>Equation Solver</th>
</tr>
</thead>
<tbody>
<tr>
<td>Piecewise Linear</td>
<td>Newton-Cotes</td>
<td>Galerkin</td>
<td>Newton</td>
</tr>
<tr>
<td>Polynomials</td>
<td>Gaussian Rules</td>
<td>Collocation</td>
<td>Powell</td>
</tr>
<tr>
<td>Splines</td>
<td>Monte Carlo</td>
<td>M. of Moments</td>
<td>Fixed-pt. iteration</td>
</tr>
<tr>
<td>Neural Networks</td>
<td>Quasi-M.C.</td>
<td>Subdomain</td>
<td>Time iteration</td>
</tr>
<tr>
<td>Rational Functions</td>
<td>Monomial Rules</td>
<td></td>
<td>Homotopy</td>
</tr>
<tr>
<td>Problem Specific</td>
<td>Asymptotics</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
- Unifies literature: Previous work can be classified and compared

<table>
<thead>
<tr>
<th>Authors</th>
<th>Approximation</th>
<th>Integration</th>
<th>Sol’n Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gustafson(1959)</td>
<td>piecewise linear</td>
<td>Newt.-Cotes</td>
<td>S.A.-time it.</td>
</tr>
<tr>
<td>Coleman(1990)</td>
<td>finite element</td>
<td>Gaussian</td>
<td>S.A.-time it.</td>
</tr>
<tr>
<td>Judd(1992)</td>
<td>orthogonal poly.</td>
<td>Gaussian</td>
<td>Newton</td>
</tr>
</tbody>
</table>