PROJECTION METHODS FOR DYNAMIC MODELS

Kenneth L. Judd

Hoover Institution and NBER

June 28, 2006

Functional Problems

- Many problems involve solving for some unknown function
  - Dynamic programming
  - Consumption and investment policy functions
  - Pricing functions in asset pricing models
  - Strategies in dynamic games
- The projection method is a robust method for solving such problems

An Ordinary Differential Equation Example

- Consider the differential equation

$$y' - y = 0, \quad y(0) = 1, \quad 0 \le x \le 3. \tag{11.1.1}$$

- Define L

$$Ly \equiv y' - y \ . \tag{11.1.2}$$

  - L is an operator mapping functions to functions; domain is C^1 functions and range is C^0.
  - Define $Y = \{y(x) | y \in C^1, \quad y(0) = 1\}$
  - (11.1.1) wants to find a $y \in Y$ such that $Ly = 0$.

- Approximate functions: consider family

$$\hat{y}(x; a) = 1 + \sum_{j=1}^{n} a_j x^j . \tag{11.1.3}$$

  - An affine subset of the vector space of polynomials.
  - Note that $\hat{y}(0; a) = 1$ for any choice of a, so $\hat{y}(0; a) \in Y$ for any a.

- Objective: find a s.t. $\hat{y}(x; a)$ "nearly" solves differential equation (11.1.1).

- Define *residual function*

$$R\,(x;a) \equiv L\hat{y} = -1 + \sum_{j=1}^{n} a_j (j x^{j-1} - x^j) \tag{11.1.4}$$

  - $R\,(x;a)$ is deviation of $L\hat{y}$ from zero, the target value.
  - A projection method adjusts a until it finds a "good" a that makes $R(x;a)$ "nearly" the zero function.
  - Different projection methods use different notions of "good" and "nearly."

- Consider

$$\hat{y}(x;a) = 1 + \sum_{j=1}^{3} a_j x^j$$

- Least Squares:

  - Find a that minimizes the total squared residual

$$\min_a \int_0^3 R(x;a)^2\,dx. \tag{11.1.5}$$

- Method of moments:
  - Idea: If $R(x;a)$ were zero, then $\int_0^3 R(x;a)\, f(x)\, dx = 0$ for all $f(x)$.
  - Use low powers of x to identify a via projection conditions

$$0 = \int_0^3 R(x;a)\, x^j\, dx \ , \quad j = 0, 1, 2. \tag{11.1.9}$$

- Galerkin
  - Idea: use basis elements, x, x^2, and x^3 in projection conditions
  - Form projections of R against the basis elements

$$0 = \int_0^3 R(x;a)\, x^j\, dx \ , \quad j = 1, 2, 3.$$

- Collocation
  - Idea: If $R(x;a) = 0$ then it is zero at all x.
  - Specify a finite set of X and choose a so that $R(x;a)$ is zero $x \in X$. If $X = \{0, 3/2, 3\}$, the uniform grid, this reduces to linear equations

- Chebyshev Collocation
  - Idea: interpolation at Chebyshev points is best
  - List the zeroes of $T_3(x)$ adapted to [0,3]

$$X = \left\{ \frac{3}{2}\,(\cos\frac{\pi}{6} + 1),\, \frac{3}{2},\, \frac{3}{2}\,(\cos\frac{5\pi}{6} + 1) \right\}$$

- Solutions

Table 11.1: Solutions for Coefficients in (11.1.3)

Scheme:	a_1	a_2	a_3
Least Squares	1.290	-.806	.659
Galerkin	2.286	-1.429	.952
Chebyshev Collocation	1.692	-1.231	.821
Uniform Collocation	1.000	-1.000	.667
Optimal L_2	1.754	-.838	.779

Table 11.2: Projection Methods Applied to (11.1.2): L_2 errors of solutions

	Uniform	Chebyshev	Least		
n	Collocation	Collocation	Squares	Galerkin	Best poly.
3	5.3(0)	2.2(0)	3.2(0)	5.3(-1)	1.7(-1)
4	1.3(0)	2.9(-1)	1.5(-1)	3.6(-2)	2.4(-2)
5	1.5(-1)	2.5(-2)	4.9(-3)	4.1(-3)	2.9(-3)
6	2.0(-2)	1.9(-3)	4.2(-4)	4.2(-4)	3.0(-4)
7	2.2(-3)	1.4(-4)	3.8(-5)	3.9(-5)	2.8(-5)
8	2.4(-4)	9.9(-6)	3.2(-6)	3.2(-6)	2.3(-6)
9	2.2(-5)	6.6(-7)	2.3(-7)	2.4(-7)	1.7(-7)
10	2.1(-6)	4.0(-8)	1.6(-8)	1.6(-8)	1.2(-8)

Simple Example: One-Sector Growth

- Consider

$$\max_{c_t} \sum_{t=1}^{\infty} \beta^t u(c_t)$$
$$k_{t+1} = f(k_t) - c_t$$

- Optimality implies that c_t satisfies

$$u'(c_t) = \beta u'(c_{t+1}) f'(k_{t+1})$$

- Problem: The number of unknowns $c_t,\ t = 1, 2, ...$ is infinite.

- **Step 0:** Express solution in terms of an unknown function

$$c_t = C(k_t) : \text{consumption function}$$

  - Consumption function $C(k)$ must satisfy the functional equation:

$$\begin{aligned} 0 &= u'(C(k)) - \beta u'(C(f(k) - C(k))) f'(f(k) - C(k)) \\ &\equiv (\mathcal{N}(C))(k) \end{aligned}$$

  - This defines the operator

$$\mathcal{N} : C^0_+ \rightarrow C^0_+$$

  - Equilibrium solves the operator equation

$$0 = \mathcal{N}(C)$$

- **Step 1:** Create approximation:
  - Find
    $$\widehat{C} \equiv \sum_{i=0}^{n} a_i k^i$$
    which "nearly" solves
    $$\mathcal{N}\ (\widehat{C}) = 0$$
  - Convert an infinite-dimensional problem to a finite-dimensional problem in R^n
    - No discretization of state space
    - A form of discretization, but in spectral domain
- **Step 2:** Compute Euler equation error function:
  $$R\ (k;\ \vec{a}) = u'(\widehat{C}(k)) - \beta u'(\widehat{C}(f(k) - \widehat{C}(k)))f'(f(k) - \widehat{C}(k))$$

- **Step 3**: Choose $\vec{a}$ to make $R(\cdot;\vec{a})$ "small" in some sense**:**
  - Least-Squares: minimize sum of squared Euler equation errors

$$\min_{\vec{a}} \int R(\cdot;\vec{a})^2 dk$$

  - Galerkin: zero out weighted averages of Euler equation errors

$$P_i(\vec{a}) \equiv \int R(k;\vec{a})\psi_i(k)dk = 0,\ \ i = 1, \cdots, n$$

    for n weighting functions $\psi_i(k)$.
  - Collocation: zero out Euler equation errors at $k \in \{k_1, k_2, \cdots, k_n\}$:

$$P_i(\vec{a}) \equiv R(k_i;\vec{a}) = 0\ ,\ \ i = 1, \cdots, n$$

- Details of $\int ...dk$ computation:
  - Exact integration seldom possible in nonlinear problems.
  - Use quadrature formulas – they tell us what are *good* points.
  - Monte Carlo – often mistakenly used for high–dimension integrals
  - Number Theoretic methods – best for large dimension
- Details of solving $\vec{a}$:
  - Jacobian, $\vec{P}_{\vec{a}}(\vec{a})$, should be well-conditioned
  - Newton's method is quadratically convergent since it uses Jacobian
  - Functional iteration and time iteration ignore Jacobian and are linearly convergent.
  - Homotopy methods are almost surely globally convergent
  - Least squares may be ill-conditioned (that is, be flat in some directions).

Bounded Rationality Accuracy Measure

Consider the one-period relative Euler equation error:

$$E(k) = 1 - \frac{(u')^{-1}\left(\beta u'\left(C\left(f(k) - C(k)\right)\right) f'\left(f(k) - C(k)\right)\right)}{C(k)}$$

- Equilibrium requires it to be zero.
- $E(k)$ is measure of optimization error
  - 1 is unacceptably large
  - Values such as .00001 is a limit for people.
  - $E(k)$ is unit-free.
- Define the L^p, $1 \leq p < \infty$, *bounded rationality accuracy* to be

$$\log_{10} \parallel E(k) \parallel_p$$

- The L^∞ error is the maximum value of $E(k)$.

Numerical Results

- Machine: Compaq 386/20 w/ Weitek 1167
- Speed: Deterministic case: < 15 seconds
- Accuracy: Deterministic case: 8th order polynomial agrees with 250,000–point discretization to within 1/100,000.

Convergence Properties of Galerkin Methods

- Zeidler (1989): If the nonlinear operator $\mathcal{N}$ is monotone, coercive, and satisfies a growth condition then Galerkin method proves existence and works numerically.

- Krasnosel'skii and Zabreiko (1984): If $\mathcal{N}$ satisfies certain degree conditions, then a large set of projection methods (e.g., Galerkin methods with numerical quadrature) converge.

- Convergence is neither sufficient nor necessary
  - Usually only locally valid
  - Convergence theorems don't tell you when to stop.
  - Non-convergent methods are no worse if they satisfy stopping rules

Coefficients of Solution

- Theoretical predictions
  - Approximation theory says that the Chebyshev coefficients should fall rapidly if $C(k)$ is smooth.
  - Orthogonal basis should imply that coefficients do not change as we increase n.
- Table 16.1 verifies these predictions.

Table 16.1: Chebyshev Coefficients for Consumption Function

k	$n=2$	$n=5$	$n=9$	$n=15$
1	0.0589755899	0.0600095844	0.0600137797	0.0600137922
2	0.0281934398	0.0284278730	0.0284329464	0.0284329804
3		−0.0114191783	−0.0113529374	−0.0113529464
4		0.0007725731	0.0006990930	0.0006988353
5		−0.0001616767	−0.0001633928	−0.0001634209
6			0.0000427201	0.0000430853
7			−0.0000123570	−0.0000122160
8			0.0000042498	0.0000036367
9			−0.0000011464	−0.0000011212
10				0.0000003557
11				−0.0000001147
12				0.0000000370

Each entry is the coefficient of the k'th Chebyshev polynomial (over the interval [.333, 1.667]) in the n-term approximation of the consumption policy function in (4.3) for the case discussed in Section 4.2.

Errors in Consumption Policy Function

- "Truth" computed by a 1,000,000 state discrete approximation
- "True solution" also has some error because of discretization
- Table 16.2 displays difference between approximations and "truth"

Table 16.2: Policy Function Errors

k	y	c	$n = 20$	$n = 10$	$n = 7$	$n = 4$	$n = 2$
0.5	0.1253211	0.1010611	1(-7)	5(-7)	5(-7)	2(-7)	5(-5)
0.6	0.1331736	0.1132936	2(-6)	1(-7)	1(-7)	2(-6)	8(-5)
0.7	0.1401954	0.1250054	2(-6)	3(-7)	3(-7)	1(-6)	2(-4)
0.8	0.1465765	0.1362965	1(-6)	4(-7)	4(-7)	4(-6)	2(-4)
0.9	0.1524457	0.1472357	1(-6)	3(-7)	3(-7)	5(-6)	2(-4)
1.0	0.1578947	0.1578947	4(-6)	0(-7)	1(-7)	2(-6)	1(-4)
1.1	0.1629916	0.1683016	4(-6)	2(-7)	2(-7)	1(-6)	9(-5)
1.2	0.1677882	0.1784982	3(-6)	2(-7)	2(-7)	4(-6)	7(-6)
1.3	0.1723252	0.1884952	7(-7)	4(-7)	4(-7)	3(-6)	9(-5)

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 17.4: Projection Method Menu

Approximation	Integration	Projections	Equation Solver
Piecewise Linear	Newton-Cotes	Galerkin	Newton
Polynomials	Gaussian Rules	Collocation	Powell
Splines	Monte Carlo	M. of Moments	Fixed-pt. iteration
Neural Networks	Quasi-M.C.	Subdomain	Time iteration
Rational Functions	Monomial Rules		Homotopy
Problem Specific	Asymptotics		

- Unifies literature: Previous work can be classified and compared

Choices			
Authors	**Approximation**	**Integration**	**Sol'n Method**
Gustafson(1959)	piecewise linear	Newt.-Cotes	S.A.-time it.
Wright-W.(1982,4)	poly. (of cond. exp.)	Newt.-Cotes	S.A.-time it.
Miranda-H.(1986)	polynomials	Newt.-Cotes	S.A.-learning
Coleman(1990)	finite element	Gaussian	S.A.-time it.
den Haan-M.(1990)	poly. (of cond. exp.)	Sim. M.C.	S.A.-learning
Judd(1992)	orthogonal poly.	Gaussian	Newton