*Numerical Methods in Economics*

MIT Press, 1998

# Notes for Chapter 5: Nonlinear Equations

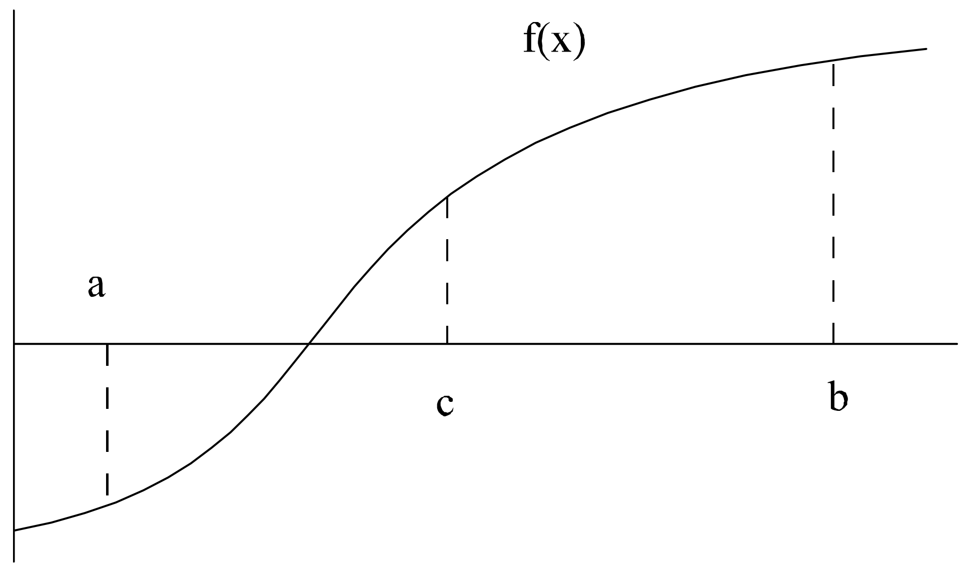
October 7, 2007

# Nonlinear Equations

- Two forms of equations: zeros and fixed points of $f : R^n \to R^n$
  - A *zero of* $f$ is any $x$ such that $f(x) = 0$
  - A *fixed point* of $f$ is any $x$ such that $f(x) = x$.
  - Note: $x$ is a fixed point of $f(x)$ iff it is a zero of $f(x) - x$.
- Existence of solutions is examined in Brouwer's theorem and its extensions.
- Examples
  - Arrow-Debreu general equilibrium: find a price at which excess demand is zero
  - Nash equilibrium of games with continuous strategies
  - Transition paths of deterministic dynamic systems
  - Approximate policy functions in nonlinear dynamic problems

# One-Dimensional Problems: Bisection

- Suppose that $f(a) < 0 < f(b)$
- Step 1: Pick a point $c \in (a, b)$
  - If $f(c) = 0$, stop
  - If $f(c) < 0$, reduce interval to $(c, b)$
  - If $f(c) > 0$, reduce interval to $(a, c)$
- Repeat



# One-Dimensional Problems: Newton's Method

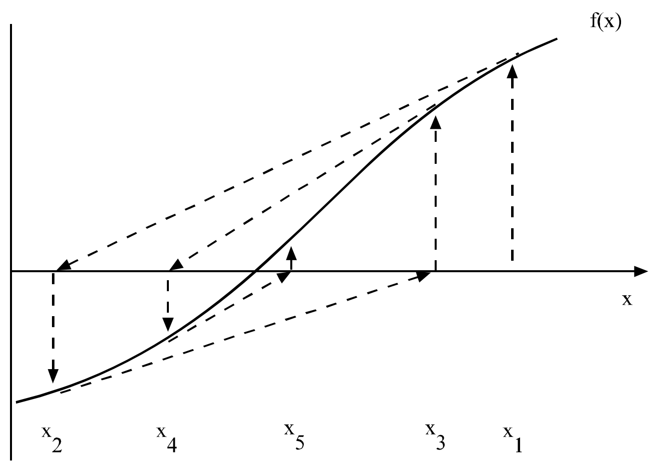
- Given guess $x_k$, compute linear approximation

$$f(x) \doteq f(x_k) + f'(x_k)(x - x_k)$$

and let $x_{k+1}$ be zero of linear approximation:

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} \tag{5.2.1}$$

- Graph of Newton's method:




- Convergence: Suppose $f$ is $C^2$ and $f(x^*) = 0$. If $x_0$ is close to $x^*$, $f'(x^*) \neq 0$, and $|f''(x^*)/f'(x^*))| < \infty$, then (5.2.1) converges to $x^*$ quadratically; that is,

$$\limsup_{k\to\infty} \quad \frac{|x_{k+1} - x^*|}{|x_k - x^*|^2} = \frac{1}{2}\,\frac{|f''(x^*)|}{|f'(x^*)|} < \infty\ . \tag{5.2.2}$$

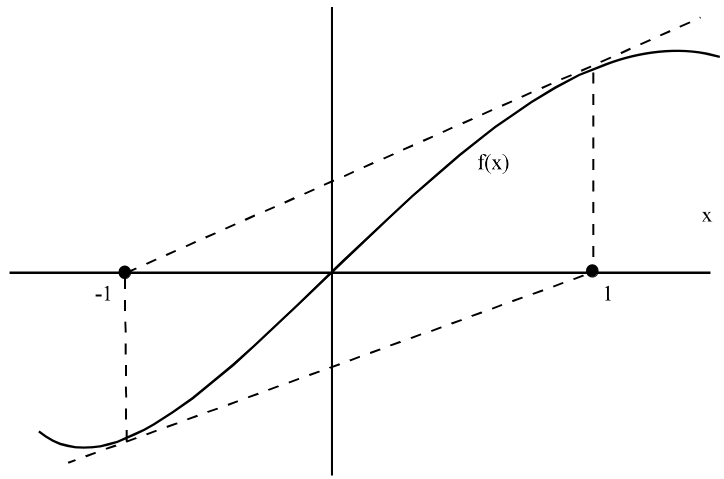
# Pathological Examples

- Newton's method works well when it works, but it can fail.
- Example: $f(x) = x^{1/3}e^{-x^2}$.
  - Unique zero of $f$ is at $x = 0$.
  - Newton's method is

$$x_{n+1} = x_n \left(1 - \frac{3}{1 - 6x_n^2}\right) \tag{5.2.4}$$

    which has two pathologies.
    * For $x_n$ small, (5.2.4) reduces to $x_{n+1} = -2x_n$; hence, (5.2.4) converges to 0 only if $x_0 = 0$ is the initial guess.
    * For $x_n$ large, (5.2.4) becomes $x_{n+1} = x_n(1 + \frac{2}{x_n^2})$, which diverges, but will eventually satisfy stopping rule at some large $x_n$.
  - Divergence due to $f''(0)/f'(0) = \infty$
  - "Convergence" arises because $e^{-x^2}$ factor squashes $f$ at large $x$; in some sense, since $f(\pm\infty) = 0$.

- Example: convergence to a cycle:



# A General Equilibrium Example

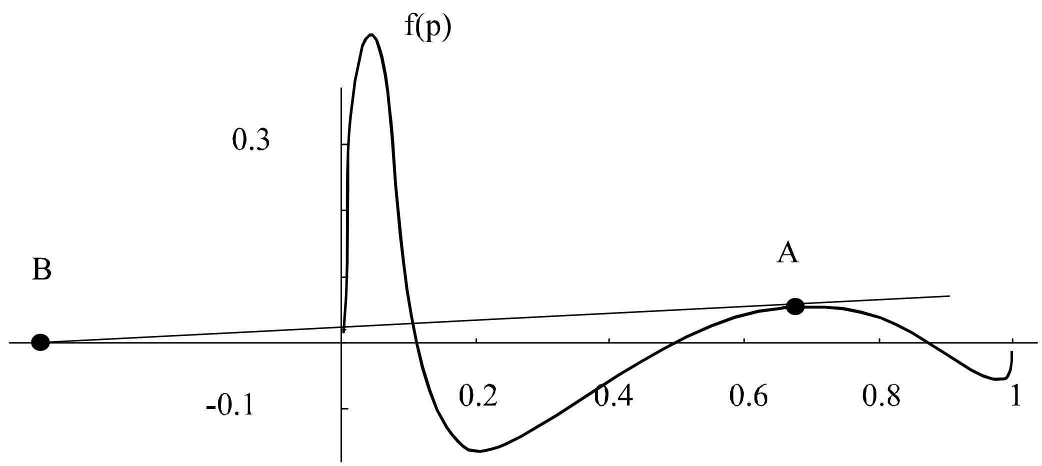
- Demand function is

$$d^i_j(p) = \theta^i_j I^i p_j^{-\eta_i}$$

$$\theta^i_j \equiv (a^i_j)^{\eta_i} \Big/ \sum_{\ell=1}^{2} (a^i_\ell)^{\eta_i} p_\ell^{(1-\eta_i)}$$

- Three equilibria:. $(0.5,\ 0.5)$, $(0.1129,\ 0.8871)$, $(0.8871,\ 0.1129)$.
- Reduce to a one-variable problem by $p_2 = 1 - p_1$, producing

$$f(p_1) \equiv \sum_{i=1}^{2} d^i_1(p_1, 1 - p_1) - \sum_{i=1}^{2} e^i_1 = 0 \tag{5.2.6}$$




- Notice: Newton's method may send $p$ negative.

## Secant Method

- Problem: $f'(x)$ may be costly.
- Solution: *secant method* approximates $f'(x_k)$ with secant of $f$ between $x_k$ and $x_{k-1}$:

$$x_{k+1} = x_k - \frac{f(x_k)\,(x_k - x_{k-1})}{f(x_k) - f(x_{k-1})} \tag{5.3.1}$$

- Convergence: If $f(x^*) = 0$, $f'(x^*) \neq 0$, and $f''(x)$ is continuous near $x^*$, then (5.3.1) converges at rate $(1+\sqrt{5})/2$, that is

$$\limsup_{k\to\infty} \frac{|x_{k+1} - x^*|}{|x_k - x^*|^{(1+\sqrt{5})/2}} < \infty \tag{5.3.3}$$

# Multivariate Equations: Gauss-Jacobi Algorithm

- Suppose $f : R^n \to R^n$, and we want to solve $f(x) = 0$:

$$\begin{aligned} f^1(x_1, x_2, \cdots, x_n) &= 0, \\ &\vdots \\ f^n(x_1, x_2, \cdots, x_n) &= 0. \end{aligned} \tag{5.4.1}$$

- Gauss-Jacobi method.
  - Given $k$th iterate, $x^k$, use equation $i$ to compute $x_i^{k+1}$:

$$\begin{aligned} f^1(x_1^{k+1}, x_2^k, x_3^k, \cdots, x_n^k) &= 0, \\ f^2(x_1^k, x_2^{k+1}, x_3^k, \cdots, x_n^k) &= 0, \\ &\vdots \\ f^n(x_1^k, x_2^k, \cdots, x_{n-1}^k, x_n^{k+1}) &= 0. \end{aligned} \tag{5.4.2}$$

  - Gauss-Jacobi repeatedly solves $n$ equations in one unknown.
  - Gauss-Jacobi is affected by the indexing scheme.
    * Otherwise, there are $n(n-1)/2$ different Gauss-Jacobi schemes.
    * Sometimes there is a natural scheme implying diagonal dominance (or, gross substitutes)
    * Strategy: choose indexing which makes Jacobian nearly diagonal

# Multivariate Equations: Gauss-Seidel Algorithm

- Gauss-Jacobi: use new guess of $x_i$, $x_i^{k+1}$, only after we have computed the entire vector of new values, $x^{k+1}$.
- Gauss-Seidel: use new guess, $x_i^{k+1}$,as soon as it is available.
- Formal definition: construct $x^{k+1}$ componentwise by solving

$$\begin{aligned} f^1\left(x_1^{k+1}, x_2^k, x_3^k, \cdots, x_n^k\right) &= 0, \\ f^2\left(x_1^{k+1}, x_2^{k+1}, x_3^k, \cdots, x_n^k\right) &= 0, \\ &\vdots \\ f^{n-1}\left(x_1^{k+1}, \cdots, x_{n-2}^{k+1}, x_{n-1}^{k+1}, x_n^k\right) &= 0, \\ f^n\left(x_1^{k+1}, \cdots, x_{n-2}^{k+1}, x_{n-1}^{k+1}, x_n^{k+1}\right) &= 0. \end{aligned} \tag{5.4.4}$$

- Both indexing and ordering matter in GS.
  - Back-substitution on triangular system is GS
  - Strategy: choose indexing and ordering which makes Jacobian nearly triangular

- Features of Gaussian methods:
  - Each step in GJ or GS is a nonlinear equation
    * Usually solved by some iterative method.
    * Economize on effort at each iteration with loose stopping rule.
  - Can apply extrapolation and acceleration methods
  - Can apply ideas at block level - "block GJ, block GS"
    * Find groups of variables and orderings such that Jacobian is nearly block diagonal or block triangular.
    * Example: in {apples, oranges, cheddar cheese, swiss cheese} problem, put cheeses in one block, fruit in the other, and use Newton to solve blocks
  - Convergence is at best linear
    * Discussion of convergence in chapter 3 applies here.
    * Key fact: for any $x^{k+1} = G(x^k)$ the spectral radius of $G_x(x^*)$ is asymptotic linear rate of convergence.

## Fixed-Point Iteration

- The simplest iterative method for solving $x = f(x)$ is

$$x^{k+1} = f(x^k) \tag{5.4.8}$$

  called fixed-point iteration; also known as *successive approximation*, *successive substitution*, or *function iteration*.

- Method is sensitive to transformations: Consider

$$x^3 - x - 1 = 0 \tag{5.3.3}$$

  – Rewrite as $x = (x+1)^{1/3}$; then the iteration

$$x_{k+1} = (x_k + 1)^{1/3}. \tag{5.3.4}$$

  converges to a solution of (5.3.3) if $x_0 = 1$.

  – Rewrite (5.3.3) as $x = x^3 - 1$; then the iteration

$$x_{k+1} = x_k^3 - 1 \tag{5.3.5}$$

  diverges to $-\infty$ if $x_0 = 1$.

- Naive implementations of the fixed-point iteration approach often fail.
- However, most algorithms have the form $x_{k+1} = f(x_k)$.
- Aim: construct fixed-point iteration which works.

## Contraction Mapping Case of Function Iteration

- For a special class of functions, fixed-point iteration will work well.
- A *differentiable contraction map on* $D$ is any $C^1$ $f : D \rightarrow R^n$ defined on a closed, bounded, convex set $D \subset R^n$ such that
  - $f(D) \subset D$, and
  - $\max_{x \in D} \| J(x) \|_\infty < 1$, $J(x)$ is Jacobian of $f$.
- *(Contraction mapping theorem)* If $f$ is a differentiable contraction map on $D$, then
  - $x = f(x)$ has a unique solution, $x^* \in D$;
  - $x^{k+1} = f(x^k)$ converges to $x^*$; and
  - there is a sequence $\epsilon_k \rightarrow 0$ such that

$$\| x^* - x^{k+1} \|_\infty \leq (\| J(x^*) \|_\infty + \epsilon_k) \| x^* - x^k \|_\infty$$

- If $f(x^*) = x^*$, $f$ is Lipschitz at $x^*$, and $\rho(J(x^*)) < 1$, then for $x^0$ close to $x^*$, $x^{k+1} = f(x^k)$ is convergent.

# Stopping Rule Problems for Multivariate Systems

- Use ideas from chapter 1
- First, use a rule for stopping.
  - If we want $\| x^k - x^* \| < \epsilon$, we continue until $\| x^{k+1} - x^k \| \leq (1-\beta)\epsilon$ where $\beta = \rho(G_x(x^*))$.
  - Sometimes we know $\beta$, as with some contraction mappings
  - Otherwise, estimate $\beta$ with

$$\hat{\beta} = \left( \frac{\| x^k - x^{k+1} \|}{\| x^{k-L} - x^{k+1} \|} \right)^{1/L}$$

    for some $L$.
- Second, check that $f(x^k)$ is close to zero.
  - Require that $\| f(x^k) \| \leq \delta$ for some small $\delta$.
  - You should have each component of $f$ small
  - Be careful about units; check should be unit-free
- $\delta$ and $\epsilon$ should not be less than square root of error in computing $f$.

# Newton's Method for Multivariate Equations

- Sequential linear approximations:
  - Replace $f$ with a linear approximation at $x^k$
  - Solve linear approximation for $x^{k+1}$
- Formally:
  - Newton approx around $x^k$ is $f(x) \doteq f(x^k) + J(x^k)\,(x - x^k)$.
  - Zero of approx is

$$x^{k+1} = x^k - J(x^k)^{-1}\, f(x^k) \tag{5.5.1}$$

- Convergence: If $f(x^*) = 0$, $\det(J(x^*)) \neq 0$ and $J(x)$ is Lipschitz near $x^*$, then for $x^0$ near $x^*$, the sequence defined by (5.5.1) satisfies

$$\lim_{k\to\infty} \frac{\| x^{k+1} - x^* \|}{\| x^k - x^* \|^2} < \infty \tag{5.5.2}$$

- Problems with Newton method
  - Jacobian, $J(x)$, may be expensive to compute (but not if you use automatic differentiation)
  - May not converge
  - Should really be called the Newton-Raphson-Fourier-Simpson method
- Solutions
  - Broyden approximates $J(x)$
  - Powell hybrid improves likelihood of convergence.
  - Homotopy methods will converge

# Secant Method (Broyden)

- Jacobian, $J(x)$, is costly to compute
  - Analytic expressions are difficult to compute
  - Finite-difference approximations require $n^2$ evaluations of $f$.
- In $R$, we used the secant; can we do this for $R^n$?
- Broyden method
  - Start with initial Jacobian guess, $A_0$
  - Use $A_k$ to compute the Newton step, $s^k$: $A_k s^k = -f(x^k)$
  - Set $x^{k+1} = x^k + s^k$.
  - Choose $A_{k+1}$ to be
    * close to $A_k$
    * consistent with secant equation $f(x^{k+1}) - f(x^k) = A_{k+1} s^k$
    * for any direction $q$ orthogonal to $s^k$, want $A_{k+1} q = A_k q$, i.e., no change in directions orthogonal to Newton step

– Broyden update is

$$A_{k+1} = A_k + \frac{(y_k - A_k s^k)(s^k)^\top}{(s^k)^\top s^k}$$
$$y_k \equiv f(x^{k+1}) - f(x^k)$$

– Stop iteration when $f(x^k)$ is close to zero, or when $s^k$ is small.

– Convergence: There exists $\epsilon > 0$ such that if $\| x^0 - x^* \| < \epsilon$ and $\| A_0 - J(x^*) \| < \epsilon$, then the Broyden method converges superlinearly.

– Key properties of Broyden versus Newton

  * Convergence asserted only $x^k$, not $A_k$
  * Need good initial guess for $A_0$
  * Each iteration of the Broyden method is cheap to compute
  * Broyden method will need more iterations than Newton's method.
  * For large systems, Broyden dominates

# Use Least Squares To Improve Chances of Convergence

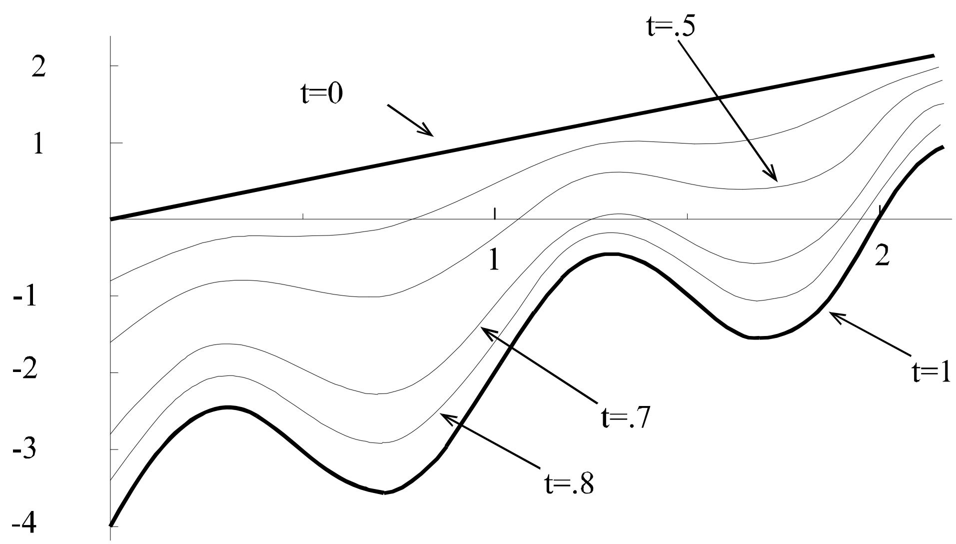
- Nonlinear Equations as an optimization problem
  - Any solution to $f(x) = 0$ is a *global* solution of

$$0 = \min_x \sum_{i=1}^{n} f^i(x)^2 \equiv SSR(x) \tag{5.6.1}$$

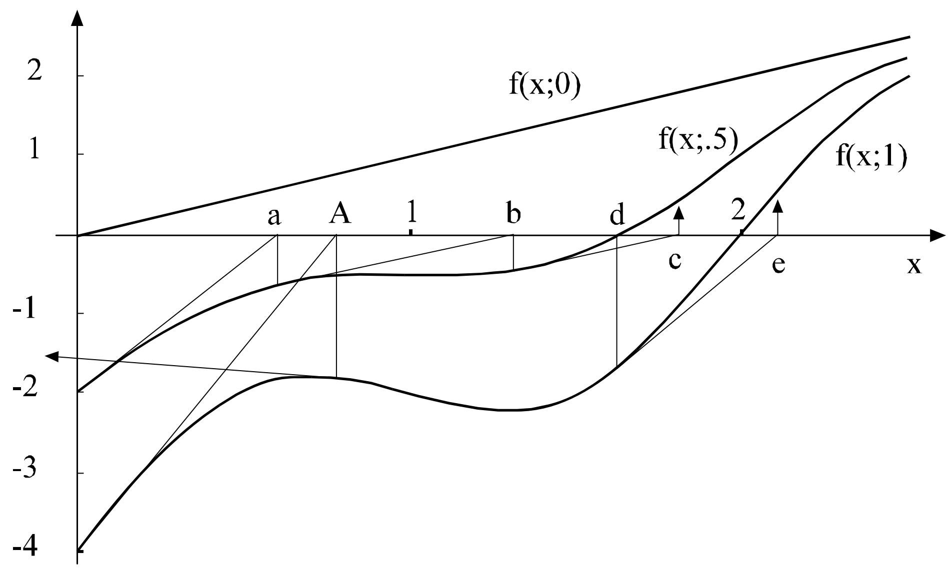
  - Benefits of (5.6.1)
    * Can use optimization procedures
    * Will always converge to something
    * May give a good initial guess for any solver
  - Problems with (5.6.1):
    * Hessian is generally ill-conditioned; roughly equals the square of the condition number of $J(x)$
    * (5.6.1) may have many local minima
- Powell's Hybrid Method
  - Do Newton, except check if Newton step reduces the value of $SSR(x)$
  - If not, then switch to least squares
  - Powell (1970) implemented procedure which avoids some conditioning problems of naive scheme.

# Simple Continuation Method

- Continuation idea
  - Suppose that we
    - want to solve $f(x; t^*) = 0$
    - know that $x = x^0$ solves $f(x; t^0) = 0$
  - If $t^*$ is near $t^0$, use
    - $x^0$ as initial guess in solving $f(x; t^*) = 0$.
    - Jacobian of $f\left(x; t^0\right)$ as initial guess for Jacobian of $f\left(x; t^0\right)$ - a hot start
  - If $t^*$ is not close to $t^0$, construct sequence of problems $f(x; t^i) = 0$, $t^0 \approx t^1 \approx \cdots \approx t^n \approx t^*$.



- Continuation method often converges in problems where standard methods fail, but continuation may fail.




- Continuation method is an approach to mass production of solutions
  - Solve $f(x;t^0)=0$ - a fixed cost
  - Solve $t$ sequence - a small (hopefully) marginal cost

# Homotopy Methods - Almost Sure Convergence

- Construct homotopy functions, $H(x,t)$, $H : R^{n+1} \rightarrow R^n$, $H \in C^0(R^{n+1})$, that continuously deforms $g$ into $f$:

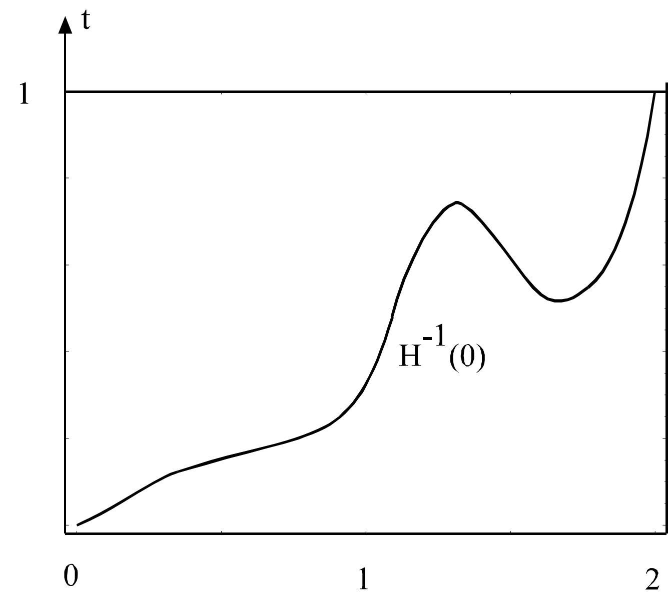
$$H(x,0) = g(x), \quad H(x,1) = f(x) \tag{5.9.1}$$

  - $H(x,0)$ should be a simple function with a unique, obvious zero
  - $H(x,1) = f(x)$
  - *Newton homotopy*: $H(x,t) = f(x) - (1-t)f(x^0)$ for some $x^0$.
  - *Fixed-point homotopy*: $H(x,t) = (1-t)(x - x^0) + tf(x)$ for some $x^0$.
  - *Linear Homotopy*: $H(x,t) = tf(x) + (1-t)g(x)$

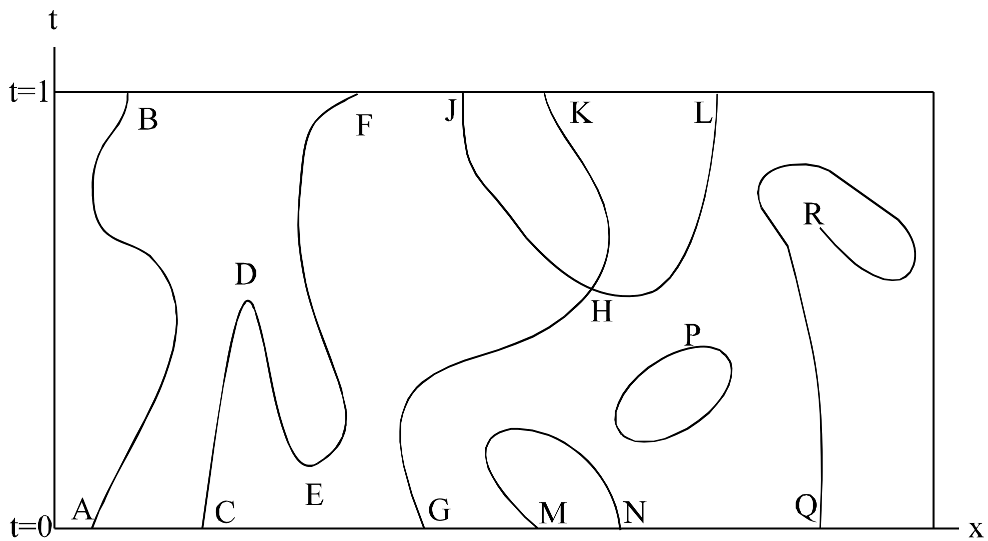
- Examine the set

$$H^{-1}(0) = \{(x,t) \mid H(x,t) = 0\}$$

  - Idea: trace out a path in $H^{-1}(0)$ connecting zeros of $H(x,0)$ to zeros of $f(x) = H(x,1)$.
  - Continuation will fail in the sense that there is no nearby solution at some values of $t$, but homotopy aims at tracing out the path even as it turns down.



- $H^{-1}(0)$ may not be simple; may have
  - turning points
  - branch points
  - extraneous components



- Parametric approach
  - $(x(s),\ t(s))$ traces path in $(x, t)$ as function of parameter $s$.
  - Implicit differentiation of $H(x(s), t(s)) = 0$ w.r.t. $s$ implies

$$\sum_{i=1}^{n} H_{x_i}(x(s), t(s))x_i'(s) + H_t(x(s), t(s))t'(s) = 0 \tag{5.9.3}$$

  - Define $y(s) = (x(s),\ t(s))$; $y$ obeys

$$\frac{dy_i}{ds} = (-1)^i \det\left(\frac{\partial H}{\partial y}\,(y)_{-i}\right)\ ,\quad i = 1, \cdots, n+1 \tag{5.9.4}$$

  - Garcia and Zangwill (1981) call (5.9.4) the *basic differential equation.*
  - (5.9.4) is defined only if $H_y = (H_x, H_t)$ has full rank; i.e., $H$ is *regular*

- Good homotopy choices: some basic theorems

  - Suppose that $f \in C^2$, $D$ compact with nonempty interior. Define

$$H(x,t) = (1-t)(x-x^0) + tf(x).$$

    If $H$ is regular, and $H(x,t) \neq 0$ for all $0 \leq t < 1$ and $x \in \partial D$, then $f$ has a zero at the end of a path joining $t=0$ and $t=1$ in $H^{-1}(0)$.

  - If $B^n \equiv \{x \in R^n \mid |x| < 1\}$ and $f : \overline{B^n} \longrightarrow \overline{B^n}$ is $C^1$, then

$$\begin{aligned} &H : B^n \times (0,1) \times B^n \longrightarrow R^n, \\ &H(a,t,x) = (1-t)(x-a) + t(x - f(x)), \end{aligned}$$

    is regular and for almost all $a \in B^n$, $H^{-1}(0)$ is a smooth curve joining $(0,a)$ to a fixed point of $f$ at $t=1$.

  - There is an open and dense subset of $C^2$ functions, $\mathcal{F}$, such that for $f \in \mathcal{F}$, $H^{-1}(0)$ is a smooth curve of $H(x,t) = tf(x) - x$.

- Simplicial methods (a.k.a. piecewise linear homotopy) are also useful.
- Practical adaptations
  - $H^{-1}(0)$ could have infinite length and oscillate as $t \to 1$; not a major concern in practice
  - Could try more natural homotopies
    - Homotopy parameter could be taste parameters
    - Homotopy parameter could be an endowment parameter
    - Economically intuitive homotopies may lack theory but be better in practice
  - One often should switch to a Newton-style method after partially traversing $H^{-1}(0)$; homotopy methods are only linearly convergent but should get close enough for Newton to work.

## CGE Problems

- Assume
  - $m$ goods
  - $n$ agents
    - utility of type $i$ is $u^i(x)$, $x \in R^m$.
    - endowment of good $j$ for type $i$ is $e^i_j$.
- Strategy of economic theory:
  - First, construct demand function for each agent given prices $p \in R^m$

$$d^i(p) = \begin{array}{l} \arg\max_x u^i(x) \\ \text{s.t. } \; p \cdot (x - e^i) = 0, \end{array} \tag{5.10.1}$$

  - Construct excess demand function: $E(p) = \sum_i^n (d^i(p) - e^i)$.
  - Equilibrium is $p \in R^m$ such that $E(p) = 0$.
  - By degree zero homogeneity of $d^i(p)$, if $E(p) = 0$ then $E(\lambda p) = 0$ for any $0 \neq \lambda \in R$; find equilibrium on unit simplex.
  - To prove existence, construct $g(p)$ on unit simplex

$$S^{m-1} = \left\{ p \,\middle|\, \sum_{j=1}^{m} p_j = 1 \right\}$$

    such that $E(p^*) \leq 0$ at any fixed point $p^*$ of $g(p)$.

- Computation: Fixed-Point Iteration Method
  - Define
$$g_j(p) = \frac{p_j + \max\left[0, E_j\left(p\right)\right]}{1 + \sum_{j=1}^{m} \max\left[0, E_j\left(p\right)\right]} \tag{5.10.2}$$
  - Since $g : S^{m-1} \rightarrow S^{m-1}$ is continuous, it has a fixed point.
  - Could compute iteration $p^{k+1} = g(p^k)$.
  - Easy if we have closed-form individual demand functions
  - No need to compute Jacobians.
  - Iterates stay in unit simplex.
  - No assurance of convergence.

- Computation: $E(p) = 0$ as a Zero Problem
  - Send $E(p) = 0$ to a nonlinear equation solver.
  - Must consider degree zero homogeneity of $E$. Hence we solve is

$$\begin{aligned} E_1(p) &= 0, \\ &\vdots \\ E_{m-1}(p) &= 0, \\ \textstyle\sum_{i=1}^{m} p_i &= 1. \end{aligned}$$

  - Easy if we have closed-form individual demand functions
  - Without closed-form expressions for $d^i(p)$, we compute $d^i(p)$ numerically.
    * Easy problem: concave objective with linear constraint
    * However, numerical error in $d^i(p)$ computation implies
      · Errors affect numerical approximation of Jacobians
      · Errors in $d^i(p)$ means that convergence criterion for $E(p) = 0$ must be relatively loose

- Computation: First-Order Conditions and Market Balance
  - Create large system with individuals' first-order conditions and market equilibrium. In $m$-good, $n$-agent model
    * First-order condition

$$u_j^i(x^i) = p_j \lambda^i, \quad i = 1, \cdots, n, \quad j = 1, \cdots, m \tag{5.10.3}$$

    * Budget constraint for each agent

$$p \cdot (x^i - e^i) = 0 \quad, \ i = 1, \cdots, n \tag{5.10.4}$$

    * Market balance for goods 1 through $m-1$,

$$\sum_{i=1}^{n} (x_j^i - e_j^i) = 0, \quad j = 1, \cdots, m-1 \tag{5.10.5}$$

    * Simplex condition for prices

$$\sum_j p_j = 1 \tag{5.10.6}$$

    * Unknowns are $p, x^i$, and $\lambda^i, i = 1, \cdots, n$.
  - System is large, but has a sparse Jacobian

- Computation: Negishi Method
  - Key observations
    - Any competitive equilibrium maximizes social welfare

$$\begin{array}{ll} \max_{x^1,x^2,\ldots} & \sum_{i=1}^n \lambda^i u^i(x^i), \\ \text{s.t.} & \sum_{i=1}^n (e^i - x^i) = 0. \end{array} \tag{5.10.7}$$

      for some weights, $\lambda^i > 0$, $i = 1, \ldots, n$.
    - Prices are proportional to marginal utilities since $p_j/p_1 = u^i_j/u^i_1$
    - Negishi approach to computing general equilibrium: Look for $\lambda$ s.t. solution to (5.10.7) is equilibrium $x^i$

– Algorithm:

  * For $\lambda$ vector, compute $X(\lambda) \in R^{m\times n}$ that solves (5.10.7).
  * Compute prices in unit simplex implied by $X(\lambda)$

$$p_j = \frac{u^1_{x^1_j}(X^1(\lambda))}{\sum_{\ell=1}^m u^1_{x^1_\ell}(X^1(\lambda))} \equiv P_j(\lambda)$$

  * Compute, for each $i$, excess wealth $W_i(\lambda) \equiv P(\lambda) \cdot (e^i - X^i(\lambda))$.
  * $P(\lambda)$ are equilibrium prices iff $W_i(\lambda) = 0$
  * Negishi approach solves the system

$$W_i(\lambda) = 0,\ i = 1, \ldots, n \tag{5.10.8}$$

    for $\lambda$, and then computes $P(\lambda)$ to get equilibrium prices.

– Negishi approach is very good if there are $n < m$

– Representative agent model is an example.