Homotopy

Philipp Müller
Gregor Reich

March 30, 2020

Motivational Example

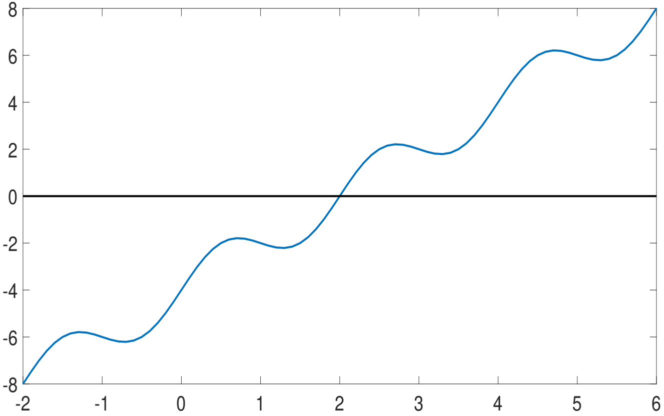
- Consider the problem of finding the root of

$$F : \mathbb{R} \to \mathbb{R}, \qquad F(x) := 2x - 4 + \sin(\pi x) = 0.$$

- Nonlinear equations are
  - **omnipresent** in economics, and
  - can be **hard to solve.**
- They implicitly define, e.g., the equilibria in dynamic models and competive general equilibria.

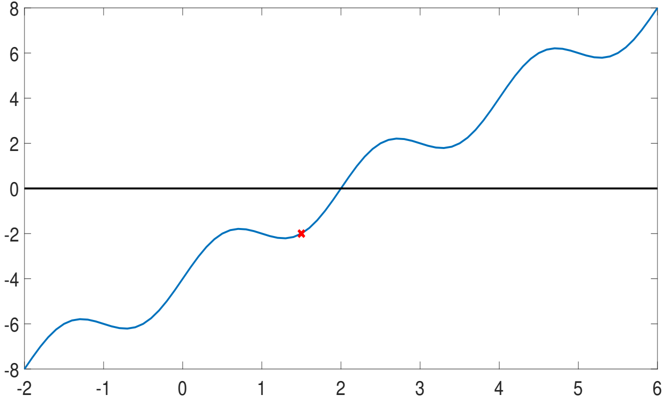
Motivational Example

$$F(x) := 2x - 4 + \sin(\pi x) = 0$$



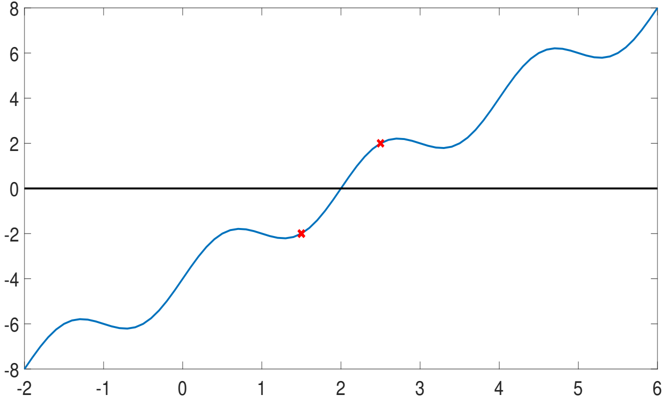
Motivational Example: Apply Matlab's **fsolve**

$$F(x) := 2x - 4 + \sin(\pi x) = 0, \; x_0 = 1.5$$



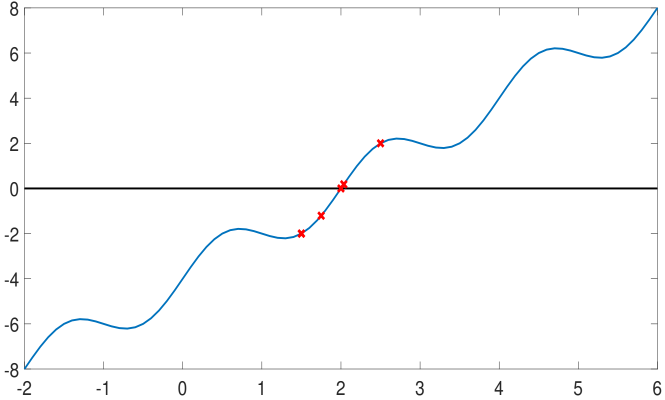
Motivational Example: Apply Matlab's **fsolve**

$$F(x) := 2x - 4 + \sin(\pi x) = 0, \; x_0 = 1.5$$



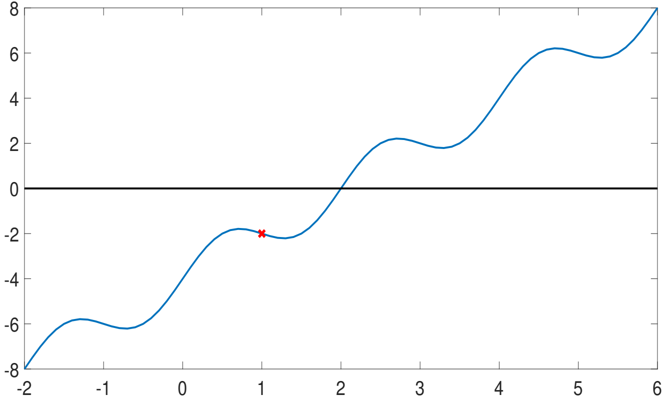
Motivational Example: Apply Matlab's **fsolve**

$$F(x) := 2x - 4 + \sin(\pi x) = 0, \; x_0 = 1.5$$



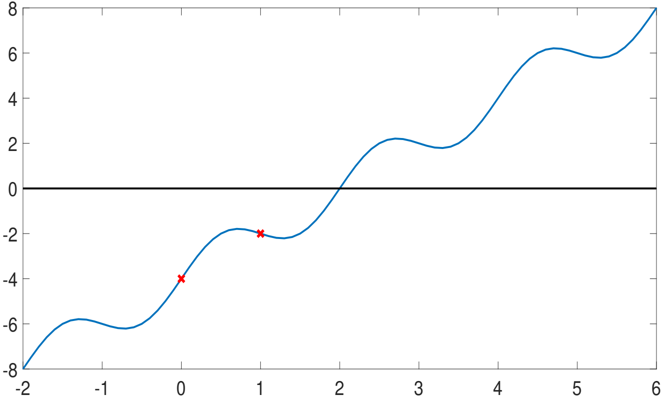
Motivational Example: Apply Matlab's **fsolve**

$$F(x) := 2x - 4 + \sin(\pi x) = 0, \; x_0 = 1.0$$



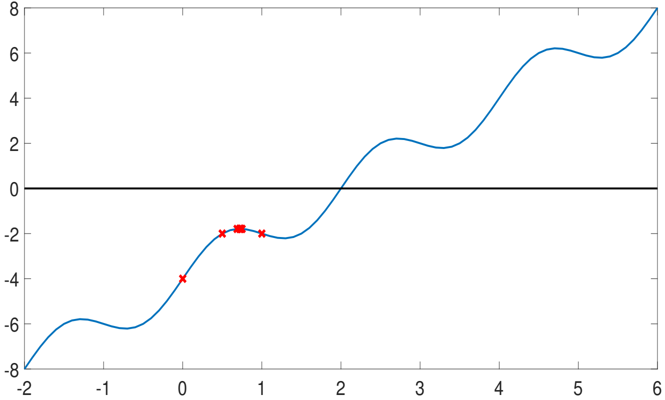
Motivational Example: Apply Matlab's **fsolve**

$$F(x) := 2x - 4 + \sin(\pi x) = 0, \; x_0 = 1.0$$



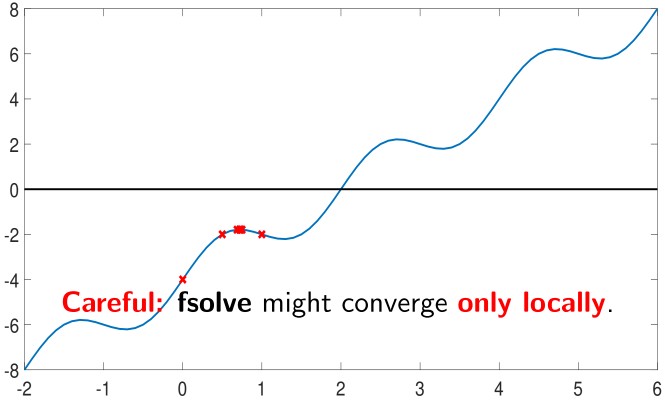
Motivational Example: Apply Matlab's **fsolve**

$$F(x) := 2x - 4 + \sin(\pi x) = 0, \; x_0 = 1.0$$



Motivational Example: Apply Matlab's **fsolve**

$$F(x) := 2x - 4 + \sin(\pi x) = 0, \; x_0 = 1.0$$



Careful: **fsolve** might converge **only locally**.

Simple Continuation Method: Deformation

Deformation Start from a *simple* function g and deform g into our target function F.

Define the homotopy map $H(x, \lambda_i)$ as

$$H : \mathbb{R}^N \times [0,1] \to \mathbb{R}^N, \quad H(x,\lambda) = (1-\lambda)g(x) + \lambda F(x),$$

with the homotopy parameter λ.

Note: It starts at $H(x, \lambda = 0) = g(x)$ and ends at $H(x, \lambda = 1) = F(x)$.

Simple Continuation Method: Algorithm

Objective Find the root of $F(x) = 0$.

Initialize Define the homotopy map $H(x, \lambda) := (1 - \lambda)g(x) + \lambda F(x)$.

Step 1 Start at $\lambda = 0$ and solve $H(x, \lambda = 0) = g(x) = 0$ for an **arbitrary** starting point x_0.

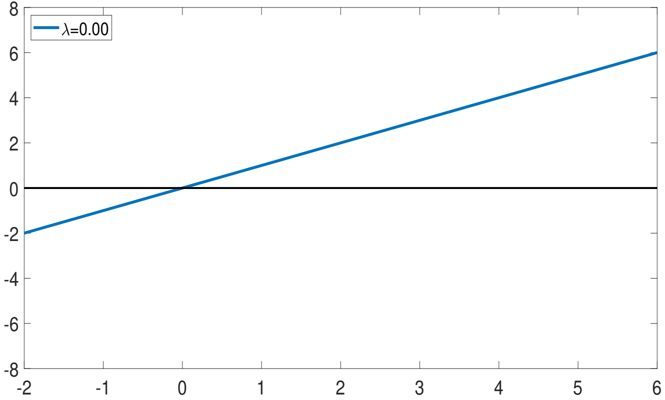
Step 2 Increase λ step-wise and solve $H(x_i, \lambda_i) = 0$ **for each** $\boldsymbol{\lambda_i}$. Use the solution from the **previous step** $i - 1$ as start point.

Result For $\lambda = 1$, we found the solution $\bar{x}$ solving $H(\bar{x}, 1) = 0$ and

$$\mathbf{F(\bar{x}) = 0}$$

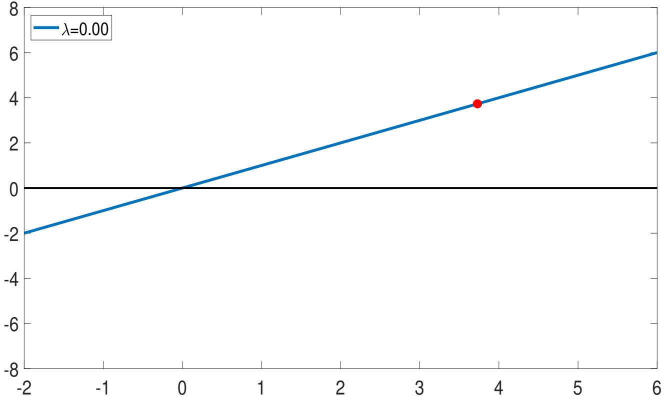
Simple Continuation Method

$$H(x, \lambda) := (1 - \lambda)x + \lambda\left(2x - 4 + \sin(\pi x)\right) = 0$$



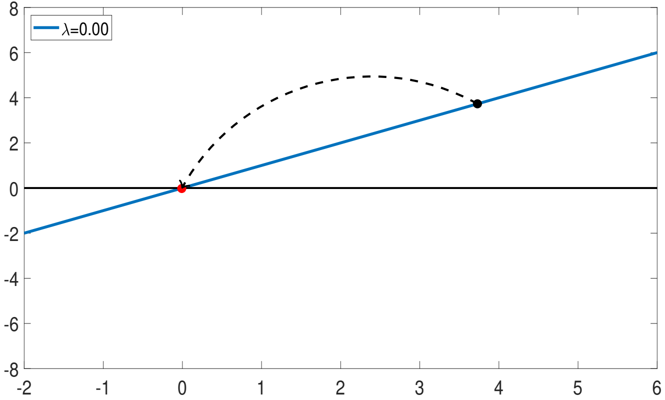
Simple Continuation Method

$$H(x,\lambda) := (1-\lambda)x + \lambda\left(2x - 4 + \sin(\pi x)\right) = 0$$



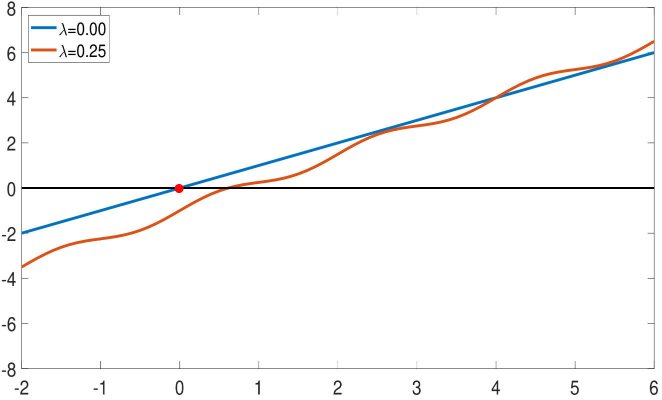
Simple Continuation Method

$$H(x,\lambda) := (1-\lambda)x + \lambda\,(2x - 4 + \sin(\pi x)) = 0$$



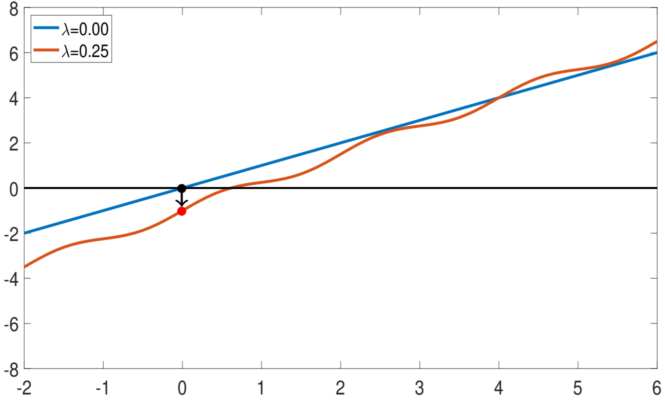
Simple Continuation Method

$$H(x,\lambda) := (1-\lambda)x + \lambda\left(2x - 4 + \sin(\pi x)\right) = 0$$



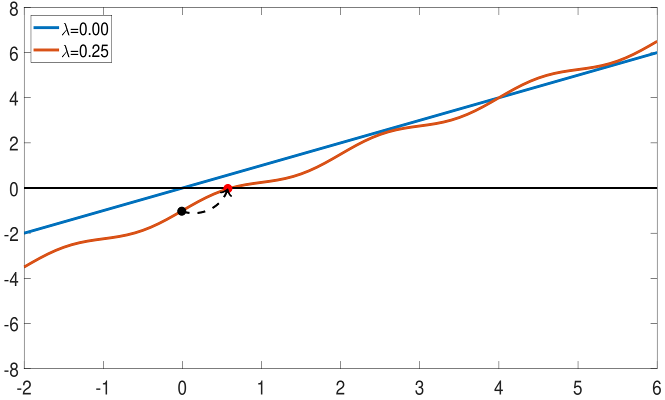
Simple Continuation Method

$$H(x, \lambda) := (1 - \lambda)x + \lambda\,(2x - 4 + \sin(\pi x)) = 0$$



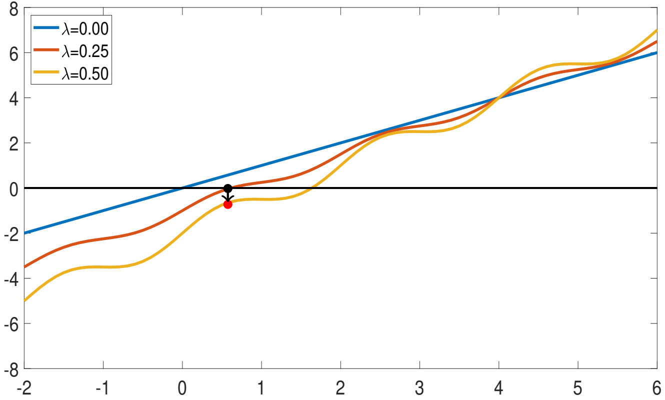
Simple Continuation Method

$$H(x,\lambda) := (1-\lambda)x + \lambda\,(2x - 4 + \sin(\pi x)) = 0$$



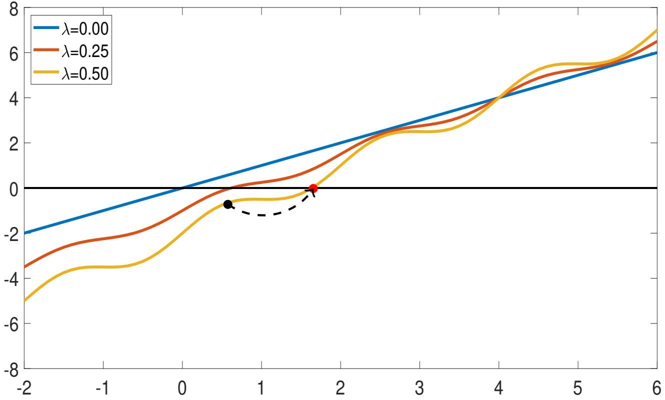
Simple Continuation Method

$$H(x,\lambda) := (1-\lambda)x + \lambda\,(2x - 4 + \sin(\pi x)) = 0$$



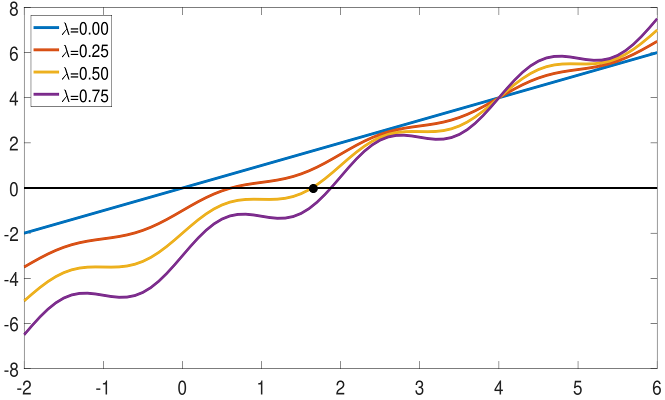
Simple Continuation Method

$$H(x,\lambda) := (1-\lambda)x + \lambda\,(2x - 4 + \sin(\pi x)) = 0$$



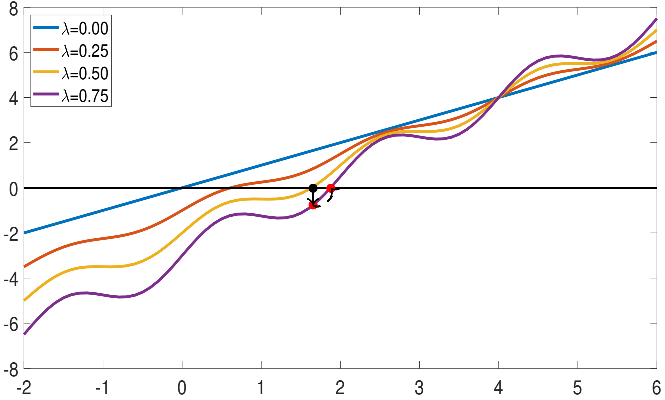
Simple Continuation Method

$$H(x,\lambda) := (1-\lambda)x + \lambda\left(2x - 4 + \sin(\pi x)\right) = 0$$



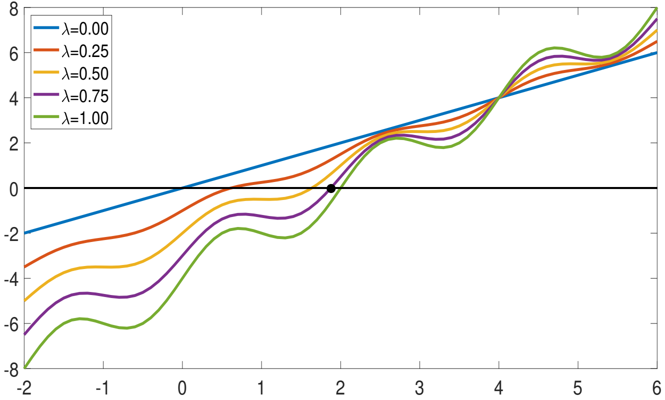
Simple Continuation Method

$$H(x,\lambda) := (1-\lambda)x + \lambda\left(2x - 4 + \sin(\pi x)\right) = 0$$



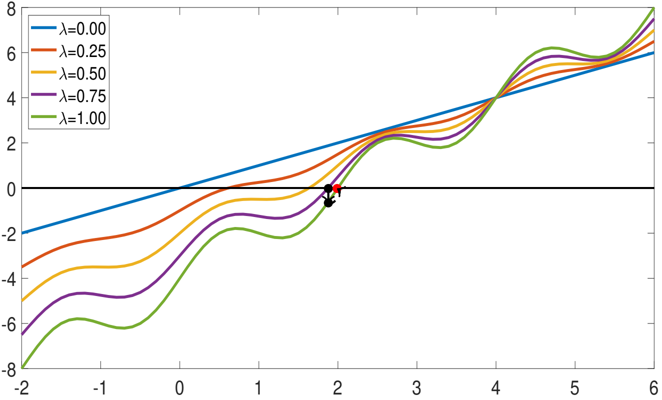
Simple Continuation Method

$$H(x,\lambda) := (1-\lambda)x + \lambda\left(2x - 4 + \sin(\pi x)\right) = 0$$

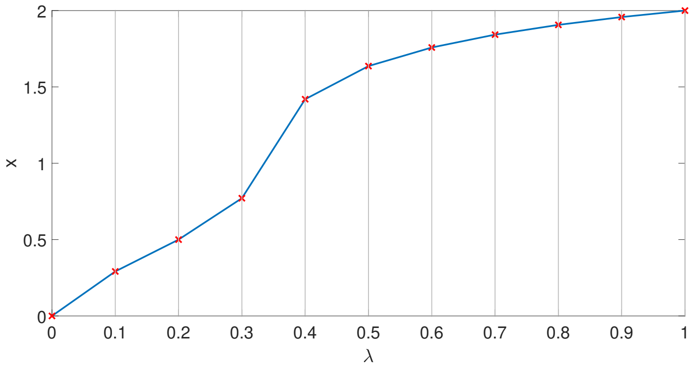


Simple Continuation Method

$$H(x,\lambda) := (1-\lambda)x + \lambda\left(2x - 4 + \sin(\pi x)\right) = 0$$



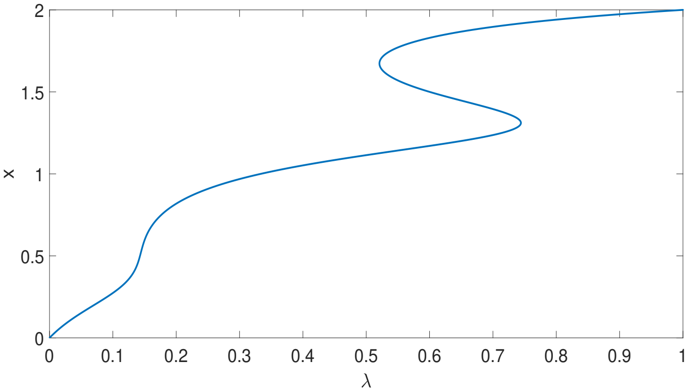
Simple Continuation: Solution Set



Note: The plot shows the curve $c := \{(x, \lambda) : H(x, \lambda) = 0\}$,
i.e. **the solution set**!

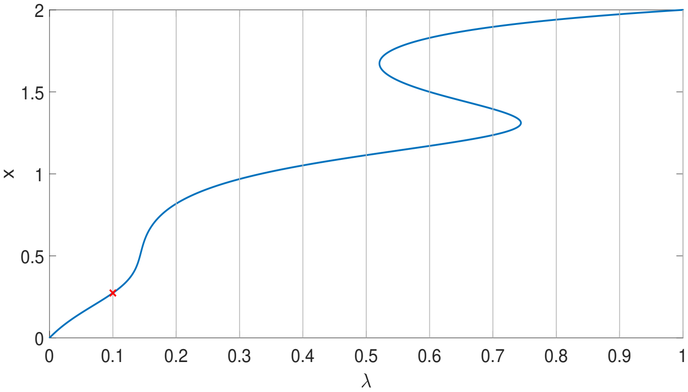
Where Simple Continuation Fails

$$H(x,\lambda) := (1-\lambda)x + \lambda\,(2x - 4 + \sin(2\pi x)) = 0$$



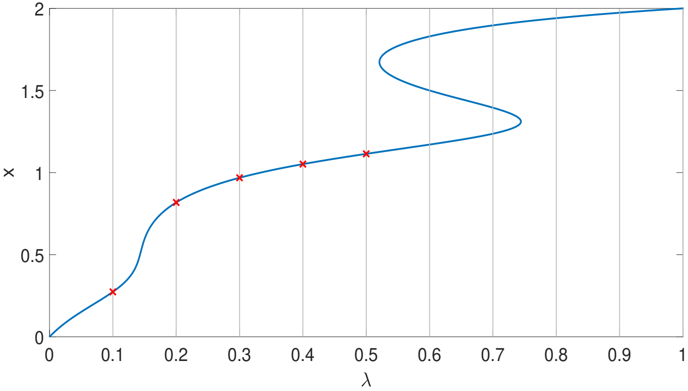
Where Simple Continuation Fails

$$H(x, \lambda) := (1 - \lambda)x + \lambda\,(2x - 4 + \sin(2\pi x)) = 0$$



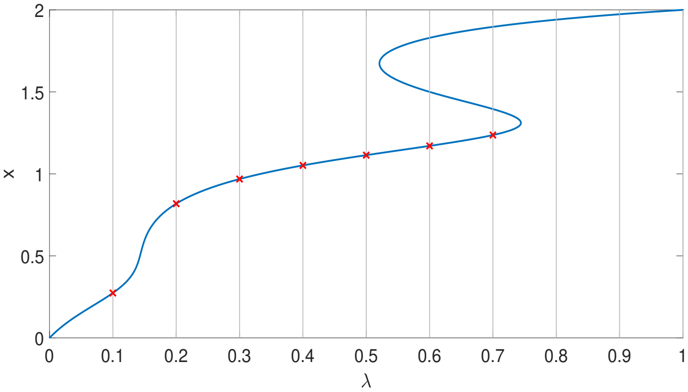
Where Simple Continuation Fails

$$H(x, \lambda) := (1 - \lambda)x + \lambda\,(2x - 4 + \sin(2\pi x)) = 0$$



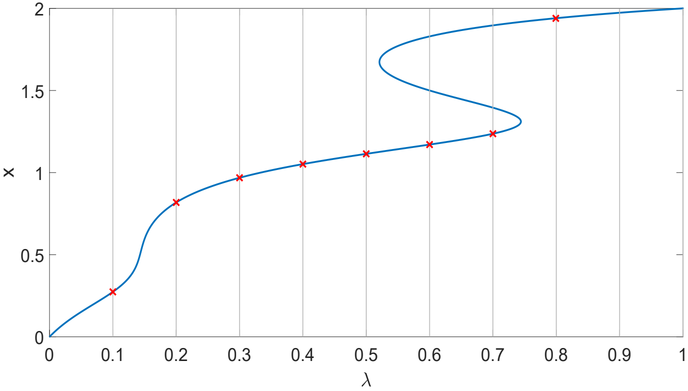
Where Simple Continuation Fails

$$H(x,\lambda) := (1-\lambda)x + \lambda\,(2x - 4 + \sin(2\pi x)) = 0$$



Where Simple Continuation Fails

$$H(x, \lambda) := (1 - \lambda)x + \lambda\,(2x - 4 + \sin(2\pi x)) = 0$$



Where Simple Continuation Fails

$$H(x, \lambda) := (1-\lambda)x + \lambda\left(2x - 4 + \sin(2\pi x)\right) = 0$$

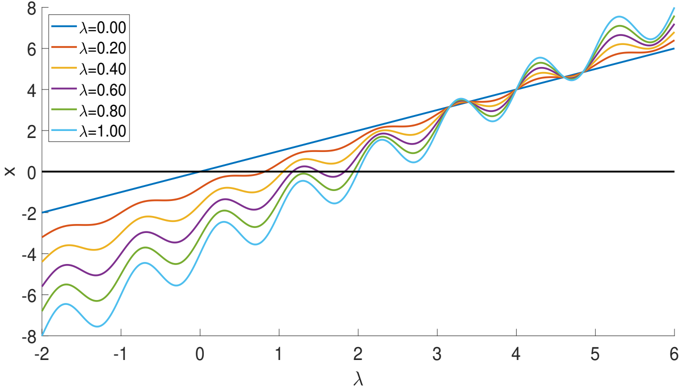
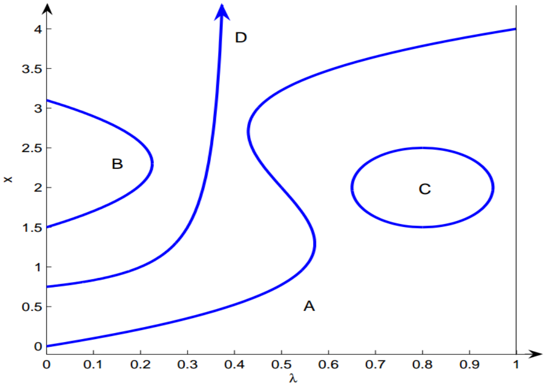
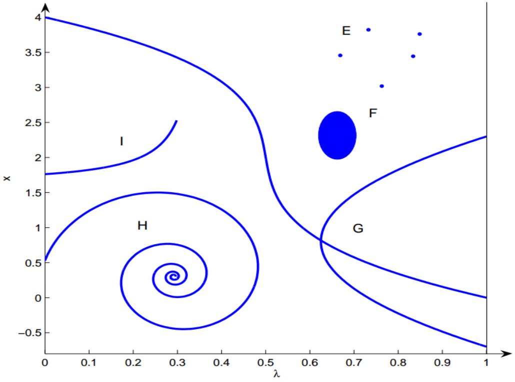


Illustration of Possible Regular Solution Sets



Source: Borkovsky et al. [2010].

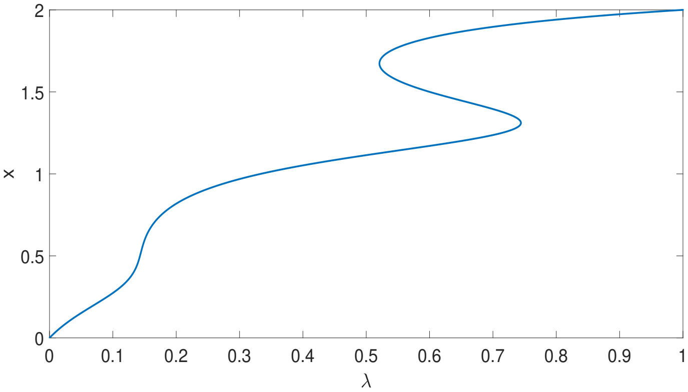
Illustration of Possible Non Regular Solution Sets



Source: Borkovsky et al. [2010].

Towards Predictor Corrector Methods

$$c := \{(x, \lambda) : H(x, \lambda) = 0\}$$



Towards Predictor Corrector Methods

Objective Find the root of $F(x) = 0$ by tracing the curve $c := \{(x, \lambda) : H(x, \lambda) = 0\}$.

Approach Use the **arclength** s as parameterisation for the curve c.

$\Rightarrow$ The homotopy map changes to $H(x(s), \lambda(s)) = 0$!

Towards Predictor Corrector Methods: ODE-Theory

Objective Find the root of $F(x) = 0$ by tracing the curve $c := \{(x, \lambda) : H(x(s), \lambda(s)) = 0\}$.

- Differentiating $H(x(s), \lambda(s))$ w.r.t. s, yields the initial and boundary value problem (IBVP)

$$x(0) = x_0, \quad \lambda(0) = 0, \quad ||(x'(s), \lambda'(s))||_2^2 = 1, \qquad (1)$$

$$\frac{\partial H(x(s), \lambda(s))}{\partial x} x'(s) + \frac{\partial H(x(s), \lambda(s))}{\partial \lambda} \lambda'(s) = 0. \qquad (2)$$

- ODE-theory algorithms can solve the IBVP (1) - (2) to follow the curve c closely.

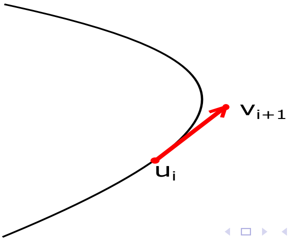
Predictor Corrector Methods: Algorithm

Approach Trace c by *alternating* **prediction** and **correction** steps.

Predictor Use e.g., Euler's explicit step to predict

$$v_{i+1} = u_i + h \cdot H'(x(s_i), \lambda(s_i)).$$

Corrector Use the predicted point v_{i+1} and improve prediction by e.g., Newton-type methods.



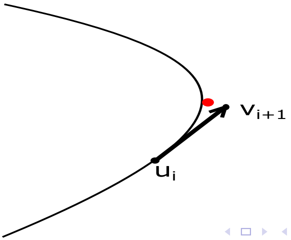
Predictor Corrector Methods: Algorithm

Approach Trace c by *alternating* **prediction** and **correction** steps.

Predictor Use e.g., Euler's explicit step to predict

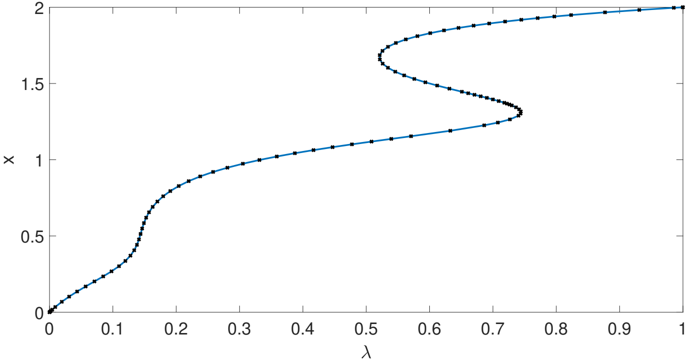
$$v_{i+1} = u_i + h \cdot H'(x(s_i), \lambda(s_i)).$$

Corrector Use the predicted point v_{i+1} and improve prediction by e.g., Newton-type methods.



Predictor Corrector Methods: ODE-based Algorithm

$$H(x, \lambda) = (1 - \lambda)x + \lambda(2x - 4 + \sin(2\pi x))$$



Source: M-Hompack

Probability One Globally Convergent Homotopy Methods

Given: $H(x, \lambda, a) = (1-\lambda)(x-a) + \lambda f(x)$

Theorem

Let $f : \mathrm{R}^p \to \mathrm{R}^p$ be a C^2 map, $H : R^p \times [0,1) \times \mathrm{R}^p \to R^p$ a C^2 map.
Suppose that
1) H is transversal to zero. (=full rank on $H^{-1}(0)$)
Suppose also that for each fixed $a \in \mathrm{R}^p$,
2) $H(0, x) = 0$ has a unique nonsingular solution x_0,
3) $H(1, x) = f(x)$ for $\forall x \in \mathrm{R}^p$.
4) $H_a^{-1}(0)$ is bounded,
then H reaches a point $(1; x)$ such that $f(x) = 0$. Furthermore, if $Df(\bar{x})$ is invertible, then $H^{-1}(0)$ has finite arc length.

Competitive General Equilibrium

Goods $j = 1, \ldots, D$ (subscripts)

Prices $(p_1, \ldots, p_D)$

Agents $i = 1, \ldots, I$ (superscripts)

Endowment $(w_1^i, \ldots w_D^i)$

Utility u^i

Agent i solves her utility maximization problem

$$\max_{x^i} u(x^i) \tag{3}$$

$$\text{s.t. } px^i = pw^i \tag{4}$$

Competitive General Equilibrium

For each of the I agents, we derive D FOCs

$$\frac{\partial u^i(x^i)}{\partial x^i_j} - \lambda^i p_j = 0 \qquad i = 1, \ldots, I, j = 1, \ldots, D \tag{5}$$

The derivatives w.r.t. the lagrangian multipliers yield the I budget constraints from above

Market clearing must hold

$$\sum_{i=1}^{I} x^i_j - w^i_j = 0 \qquad j = 1, \ldots, D \tag{6}$$

Simplex normalization $\sum_{j=1}^{D} p_j = 1$.

Competitive General Equilibrium

System of nonlinear equations with $ID + I + D$ equations and $ID + I + D$ unknowns

Unknowns are

- consumption allocations x_j^i
- Lagrange multipliers λ^i
- prices p_j

Parameter Continuation

Objective Solve parameterised non-linear equations of type $F(x, \alpha) = 0$ for $\alpha \in [a, b]$.
α could be e.g. the *discount factor* in your model.

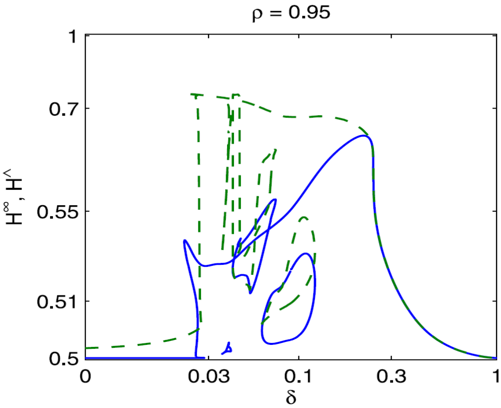
Approach Apply homotopy methods by defining the homotopy map as

$$H(x, \lambda) := F(x, \lambda) = 0,$$

from $F(x, a)$ to $F(x, b)$.

Benefit: Predictor corrector methods find *"all"* solutions for $\lambda \in [a, b]$.

Parameter Continuation



Parameter continuation in Doraszelski et al. [2010].

Recap: Homotopy Algorithms in Economics

- We can use homotopy algorithms to **solve nonlinear systems of equations**

  by deforming a *globally solvable* system of equations $g(x)$ into our target system of equations $F(x)$
- Other applications include e.g.,
  - **solving parameterized families of nonlinear equations** depending on a single parameter, and
  - finding all solutions to polynomial equations by Judd et al. [2012],
  - in engineering: solving nonlinear programming problems, where optimizer fail [WATSON, 1999].

HOMPACK90

- A collection of predictor corrector methods implemented in Fortran 90 by Watson et al. [1997]
- First application in economics by Schmedders [1998] for solving for equilibria in GE models with incomplete asset markets
- Well-established algorithms which take care of e.g.,
  1. adaptive step sizes, and
  2. an efficient implementation of the corrector step.
- Required user input: homotopy map H and its Jacobian JH as **Fortran subroutines**

M-Hompack

- We have implemented and provide M-Hompack, an interface between Matlab and HOMPACK90
- We chose to provide a Matlab interface for the beginning as it is widely used as modeling language
- Researchers do not need to implement **any** Fortran code, but can implement the homotopy map and its Jacobian within Matlab

⇒ The application of homotopy methods becomes more feasible and even straight-forward

Employed Computational Features

- Automatic differentiation
  - AD algorithms transform source codes of functions to their derivative

  ⇒ **Homotopy methods rely on Jacobians; AD eliminates the need for any analytic differentiation**
- Sparsity
  - (If applicable) support of existing sparsity patterns in the Jacobian

Counterfactual Analysis in Dynamic Models

Objective Evaluate the effect of a parameter change *(= policy change)* on the equilibrium.

- In models with **multiple equilibria**, the "correct" equilibrium is not well-defined without further information
  - In **factual scenarios** observe data select the equilibrium by maximum likelihood estimation.
  - For **counterfactual scenario**, we cannot observe these data.
- There exists **no** well-established algorithm for counterfactual analysis.

Toy Model Aguirregabiria [2012], Modified.

The fixed-point equation

$$\psi(P,\beta) = h(\beta)G(P) \tag{7}$$

defines the equilibrium P implicitly. G denotes the Cauchy cumulative density function (CDF)

$$G(P) = \frac{1}{\pi}\text{atan}\left(\frac{P-0.5}{0.15}\right) + 0.5, \tag{8}$$

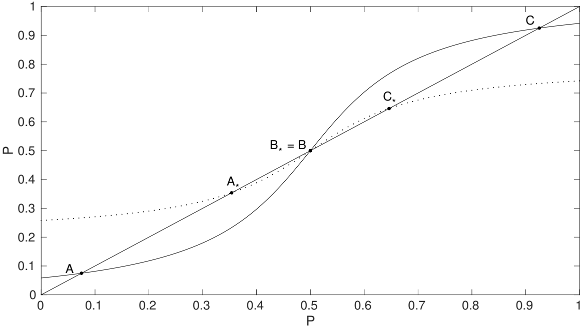
and

$$h(\beta) = \frac{1}{3\beta^5+3}. \tag{9}$$

$\hat{\beta}_0 := 0.85$ denotes the value of β in the factual scenario, and $\beta_* := 1.15$ in the **counterfactual** scenario

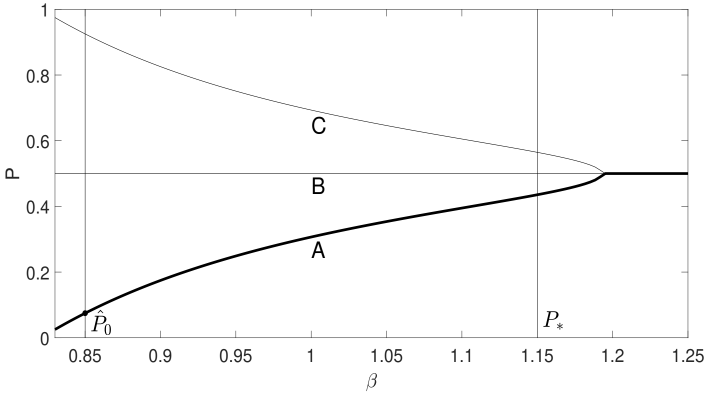
Fixed-Point Plot

$$\psi(P,\beta) = h(\beta)G(P)$$



$\beta = \hat{\beta}_0$ (solid line) and $\beta = \beta_*$ (dotted line) with the equilibria $\{A, B, C\}$

Equilibria Paths: As Function of β



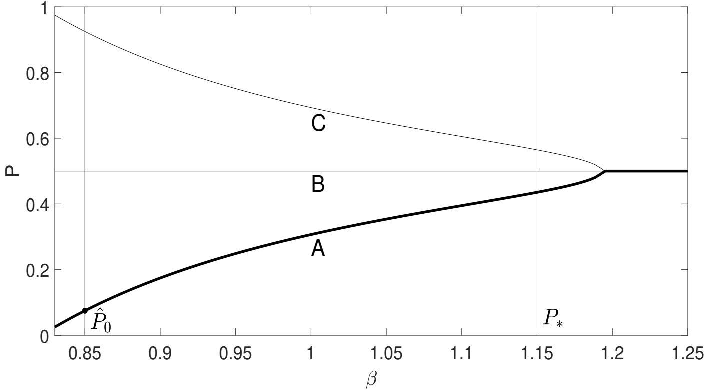
Counterfactual Analysis following Aguirregabiria [2012]

Assume Invariance of equilibrium type (IET) by Aguirregabiria [2012]

Definition (IET): The factual equilibrium $\hat{P}_0$ and the counterfactual equilibrium P_* are connected by a continuous path.

Idea The fixed-point equation for $\hat{P}_0$ and for P_* are connected by a continuous path. **Trace the path** by varying the **parameter of interest** β.

Equilibria Paths: As a Function of β



Note: We can trace these paths with our homotopy methods!

The Homotopy Algorithm

Objective Find the counterfactual equlibrium *"of same type"* for your chosen counterfactual parameter β_*.

Step 1 Build the homotopy map

$$H(P,\lambda) := \psi(P,(1-\lambda)\beta_0 + \lambda\beta_*) - P \tag{10}$$

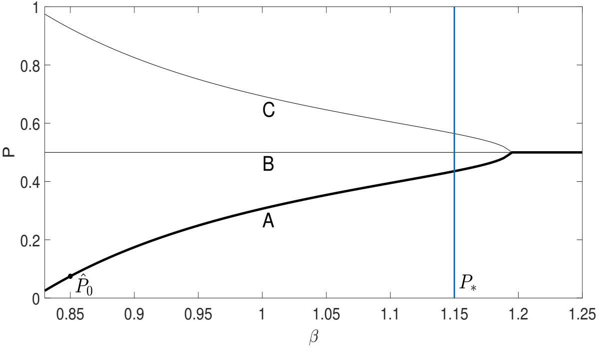
from the factual scenario $\hat{P}_0$ to the counterfactual scenario P_*.

Step 2 Follow the curve c by employing the ODE predictor corrector method.

Step 3 For $\lambda = 1$ and $H(\bar{P},1) = 0$: $\bar{P}$ equals our counterfactual equilibrium P_*.

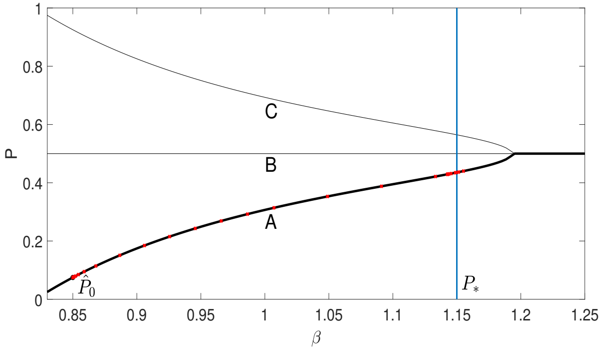
Result M-Hompack

$$\hat{\beta}_0 = 0.85 \text{ and } \beta_* = 1.15$$



Result M-Hompack

$\hat{\beta}_0 = 0.85$ and $\beta_* = 1.15$



Note: M-Hompack solves for the "correct" counterfactual equilibrium, and even yields intermediate equilibria as byproduct!

Aguirregabiria [2012]'s Algorithm

Simple Homotopy Algorithm

Objective Find the counterfactual equlibrium *"of same type"* for your chosen counterfactual parameter β_*.

Step 1 Derive the **first-order Taylor expansion** F_T around the factual equilibrium $F(\beta_0) := \hat{P}_0$ as

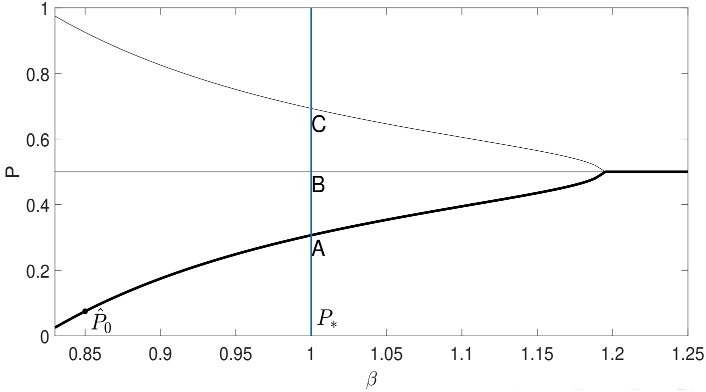
$$F_T(\beta) = F(\hat{\beta}_0) + \frac{\partial F(\hat{\beta}_0)}{\partial \beta}(\beta - \hat{\beta}_0)$$

Step 2 Approximate the counterfactual equilibrium P_* by $\tilde{P}_* = F_T(\beta_*)$

Step 3 Use $\tilde{P}_*$ as starting point for the equilibrium mapping $\psi(P, \beta)$ and iterate up to convergence.

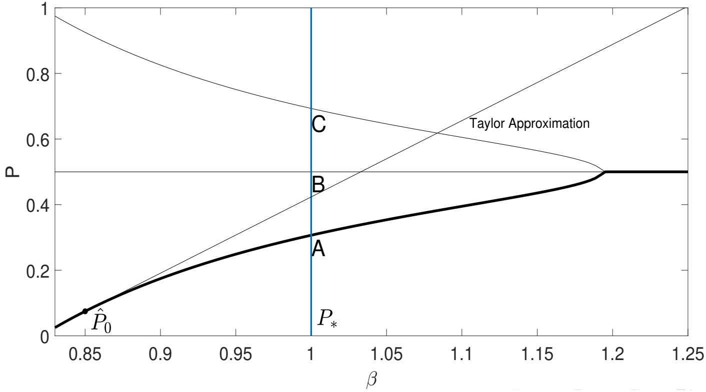
Simple Homotopy Algorithm Aguirregabiria [2012]

$$\hat{\beta}_0 = 0.85 \text{ and } \beta_* = 1.0$$



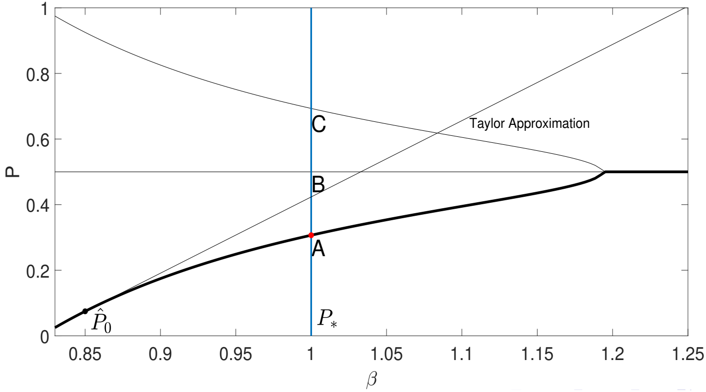
Simple Homotopy Algorithm Aguirregabiria [2012]

$$\hat{\beta}_0 = 0.85 \text{ and } \beta_* = 1.0$$



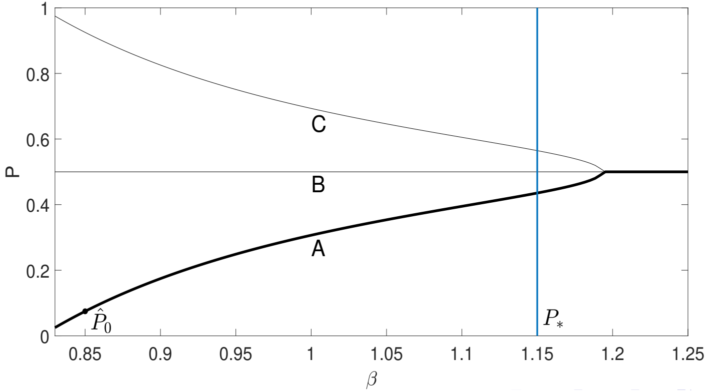
Simple Homotopy Algorithm Aguirregabiria [2012]

$$\hat{\beta}_0 = 0.85 \text{ and } \beta_* = 1.0$$



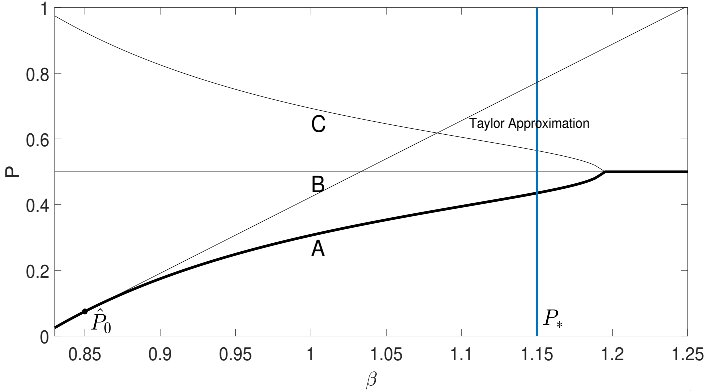
Simple Homotopy Algorithm Aguirregabiria [2012]

$$\hat{\beta}_0 = 0.85 \text{ and } \beta_* = 1.15$$



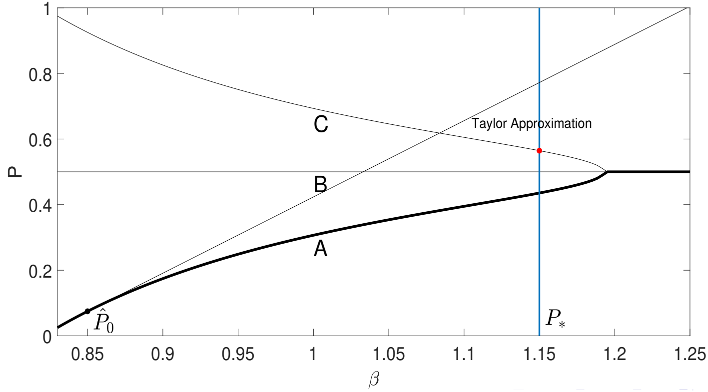
Simple Homotopy Algorithm Aguirregabiria [2012]

$$\hat{\beta}_0 = 0.85 \text{ and } \beta_* = 1.15$$



Simple Homotopy Algorithm Aguirregabiria [2012]

$$\hat{\beta}_0 = 0.85 \text{ and } \beta_* = 1.15$$



Live-Demonstration

- Matlab Live-Demo

References I

V. Aguirregabiria. A method for implementing counterfactual experiments in models with multiple equilibria. *Economics Letters*, 114(2):190–194, 2012.

R. Borkovsky, U. Doraszelski, and Y. Kryukov. A User's Guide to Solving Dynamic Stochastic Games Using the Homotopy Method. *Operations Research*, 58(4):1116 – 1132, 2010.

U. Doraszelski, D. Besanko, and Y. Kryukov. LEARNING-BY-DOING, ORGANIZATIONAL FORGETTING, AND INDUSTRY DYNAMICS. *Econometrica*, 2010.

K. L. Judd, P. Renner, and K. Schmedders. Finding all pure-strategy equilibria in games with continuous strategies. *Quantitative Economics*, 3(2):289–331, 2012.

References II

K. Schmedders. Computing equilibria in the general equilibrium modelwith incomplete asset markets. *Journal of Economic Dynamics and Control*, 22(22):1375–1401, 1998.

L. T. WATSON. Theory of globally convergent probability-one homotopies for nonlinear problems. *SIAM*, 58, 1999.

L. T. Watson, M. Sosonkina, A. P. Morgan, and H. F. Walker. Algorithm 777: HOMPACK90: A Suite of Fortran 90 Codes for Globally Convergent Homotopy Algorithms. *ACM Transactions on Mathematical Software*, 23(4):514–549, 1997.