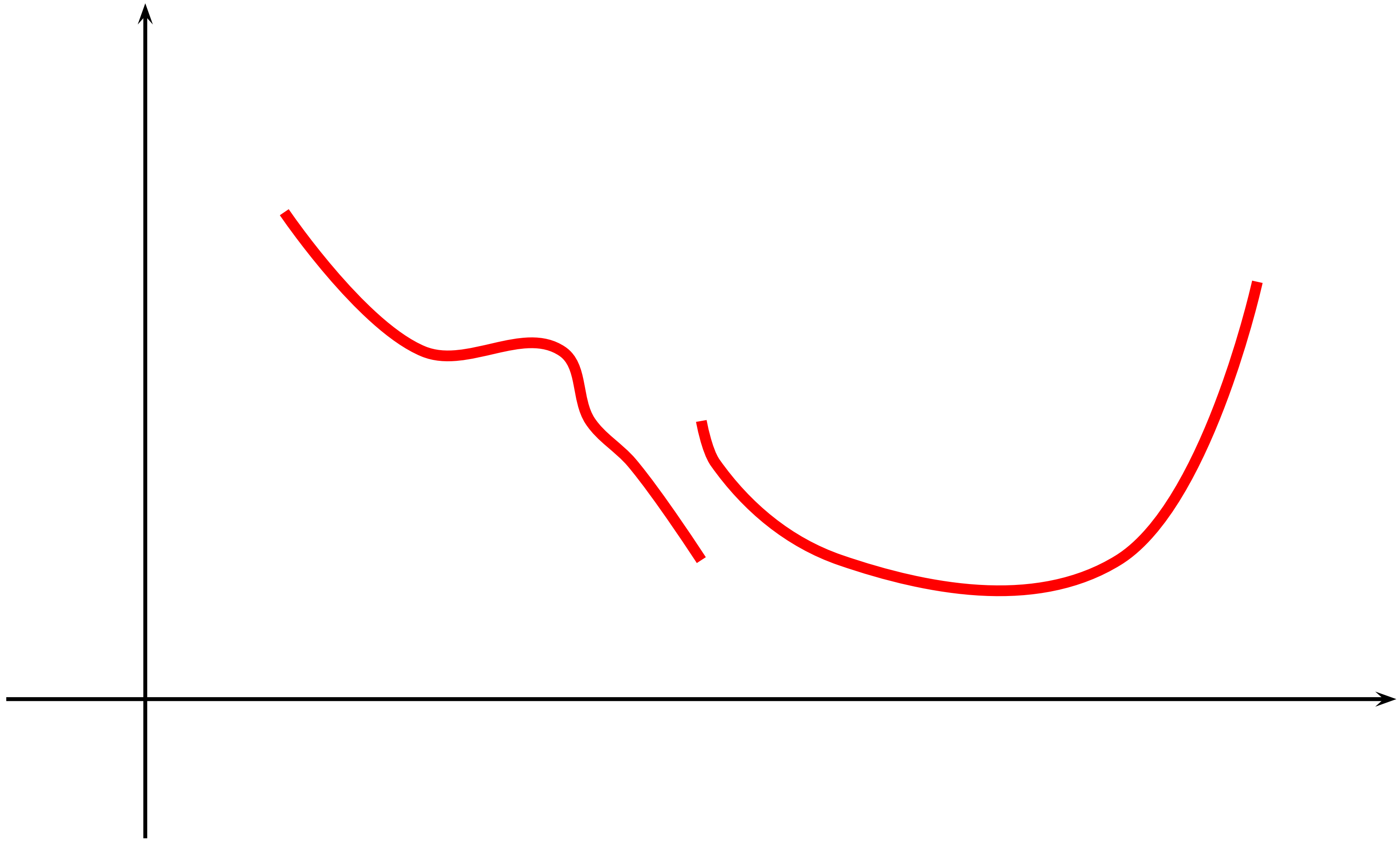
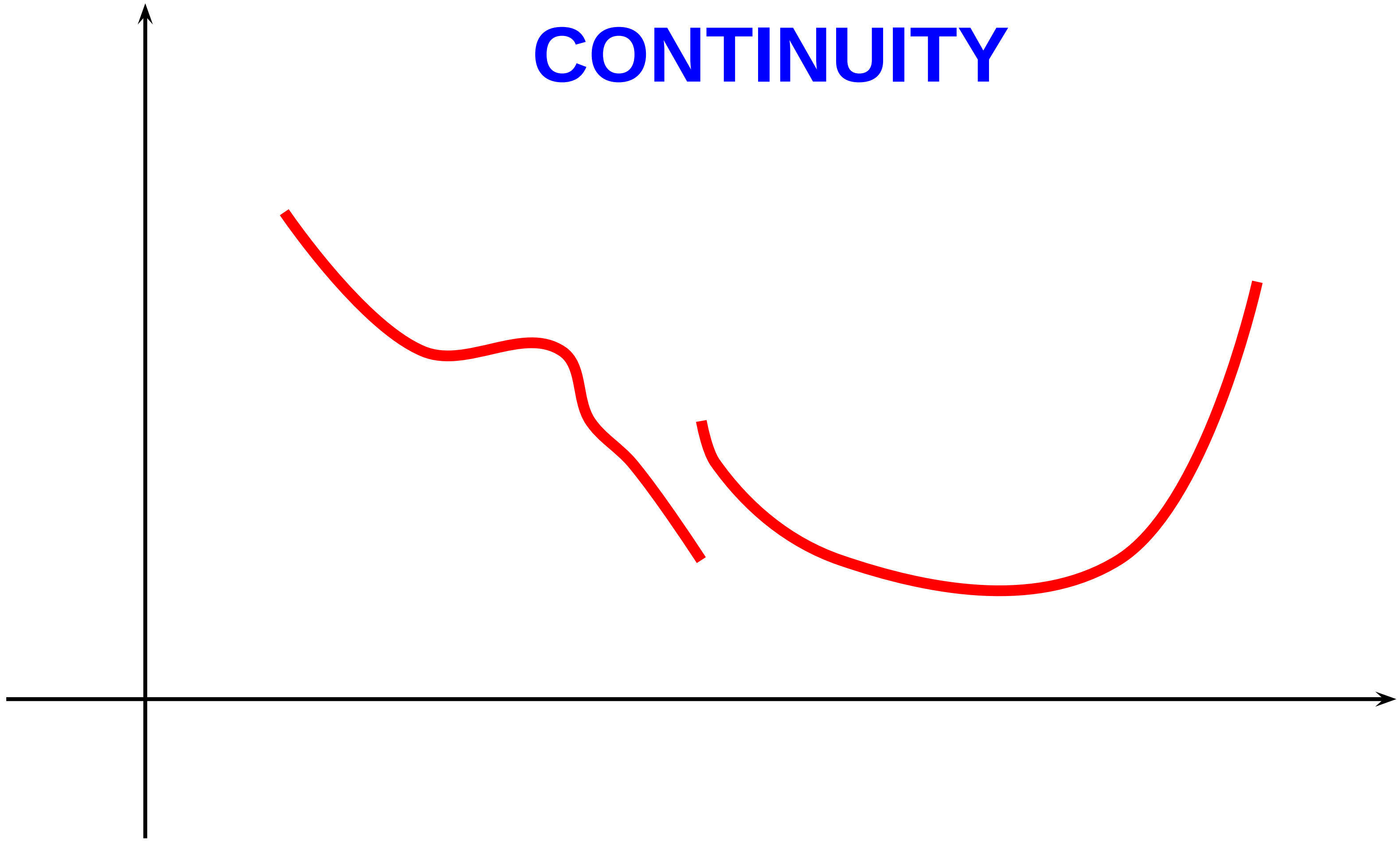
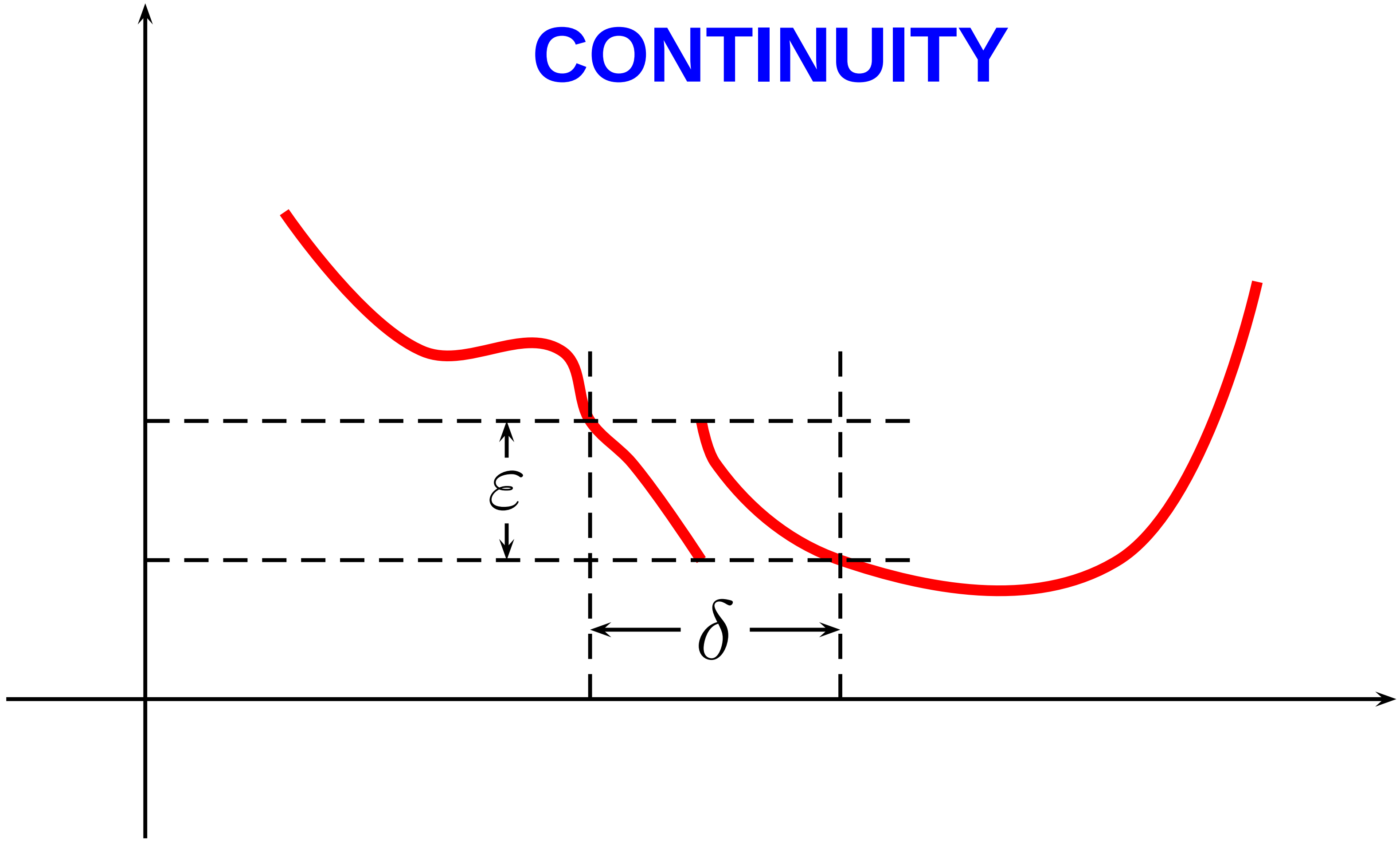
Focus Slide



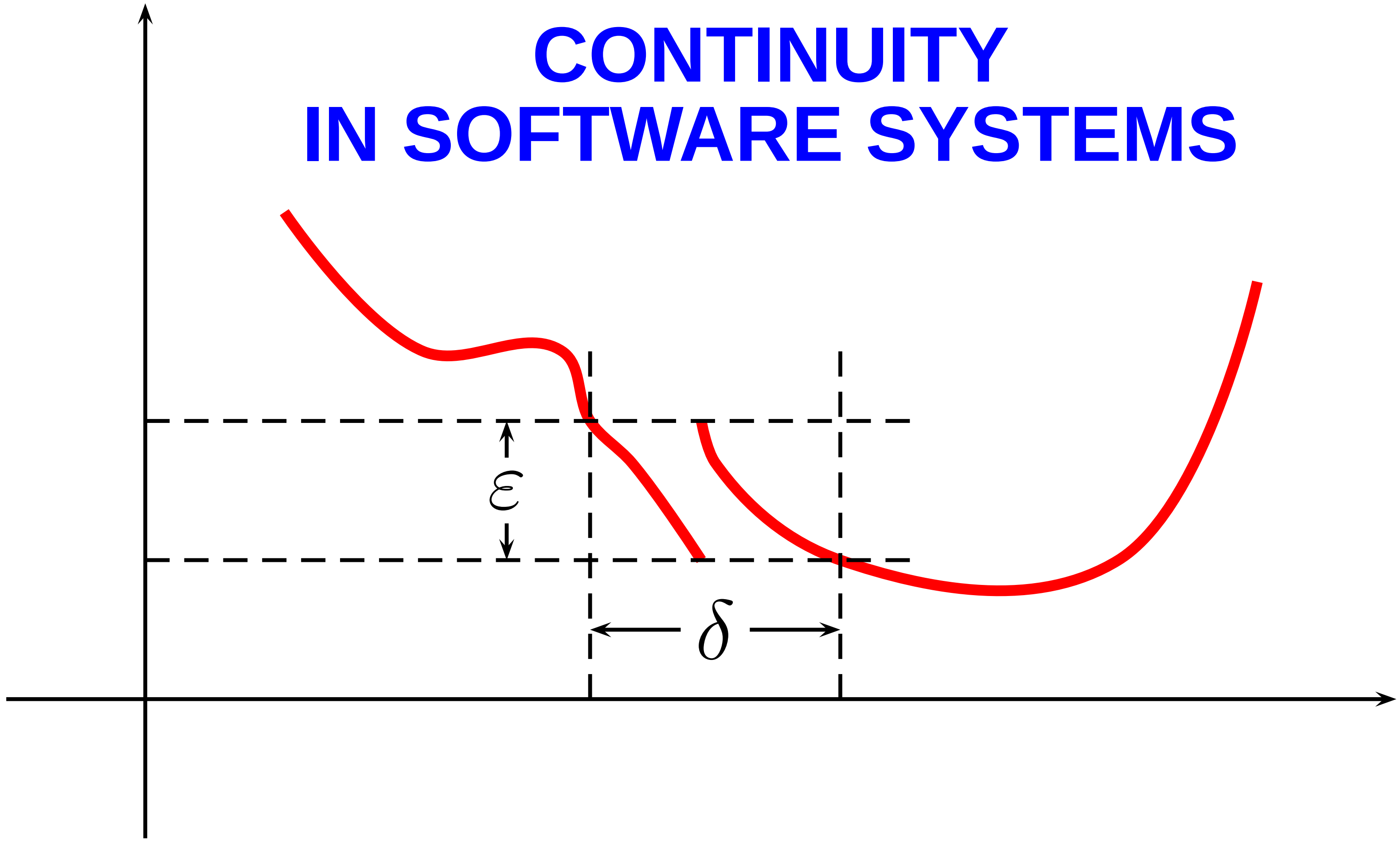
CONTINUITY



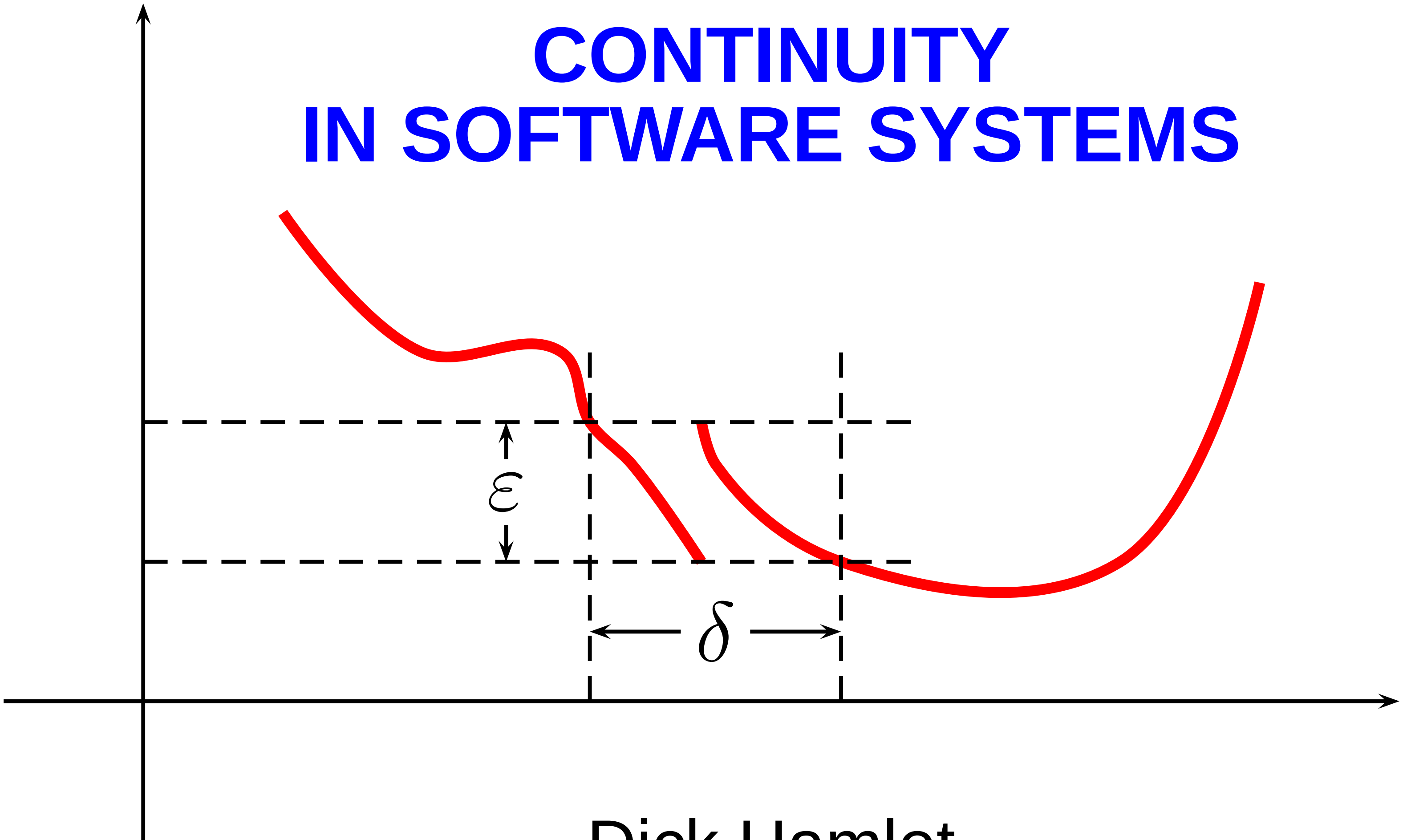
CONTINUITY



CONTINUITY IN SOFTWARE SYSTEMS



CONTINUITY
IN SOFTWARE SYSTEMS



Dick Hamlet
Portland State University
Portland, OR, USA

Outline of the Talk

I. Continuity in the Real World

II. Defining Continuity

III. Testing and Analyzing ‘Continuity’

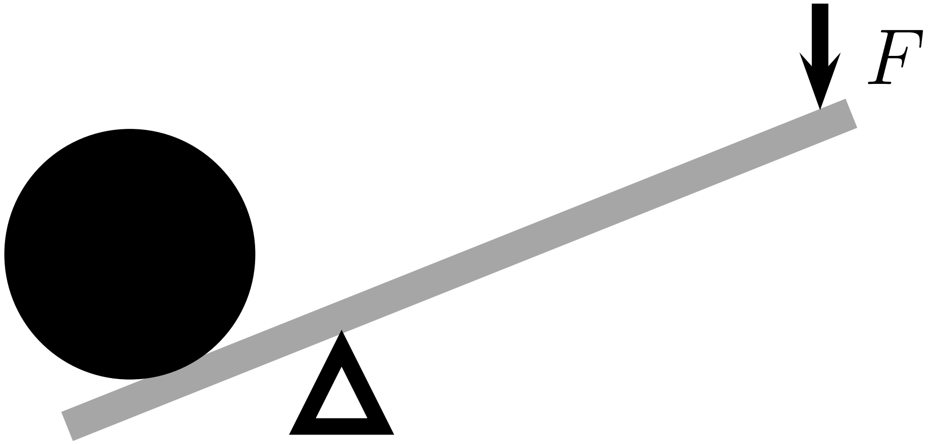
Outline of the Talk

I. Continuity in the Real World

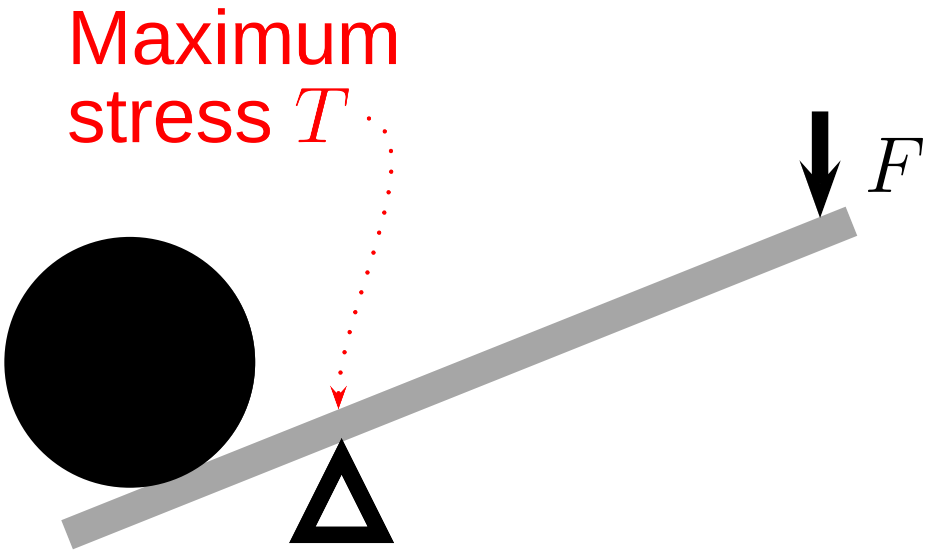
II. Defining Continuity

III. Testing and Analyzing ‘Continuity’

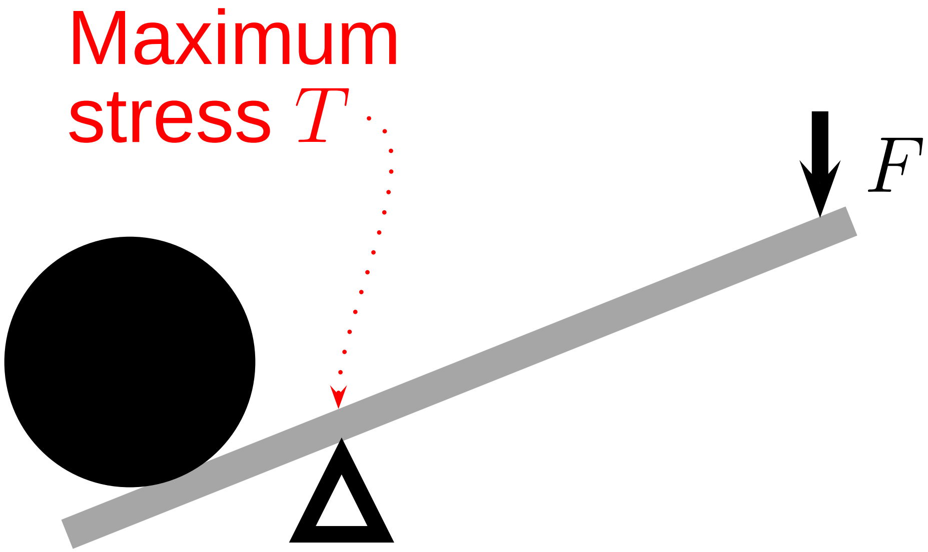
The Trustworthy Lever

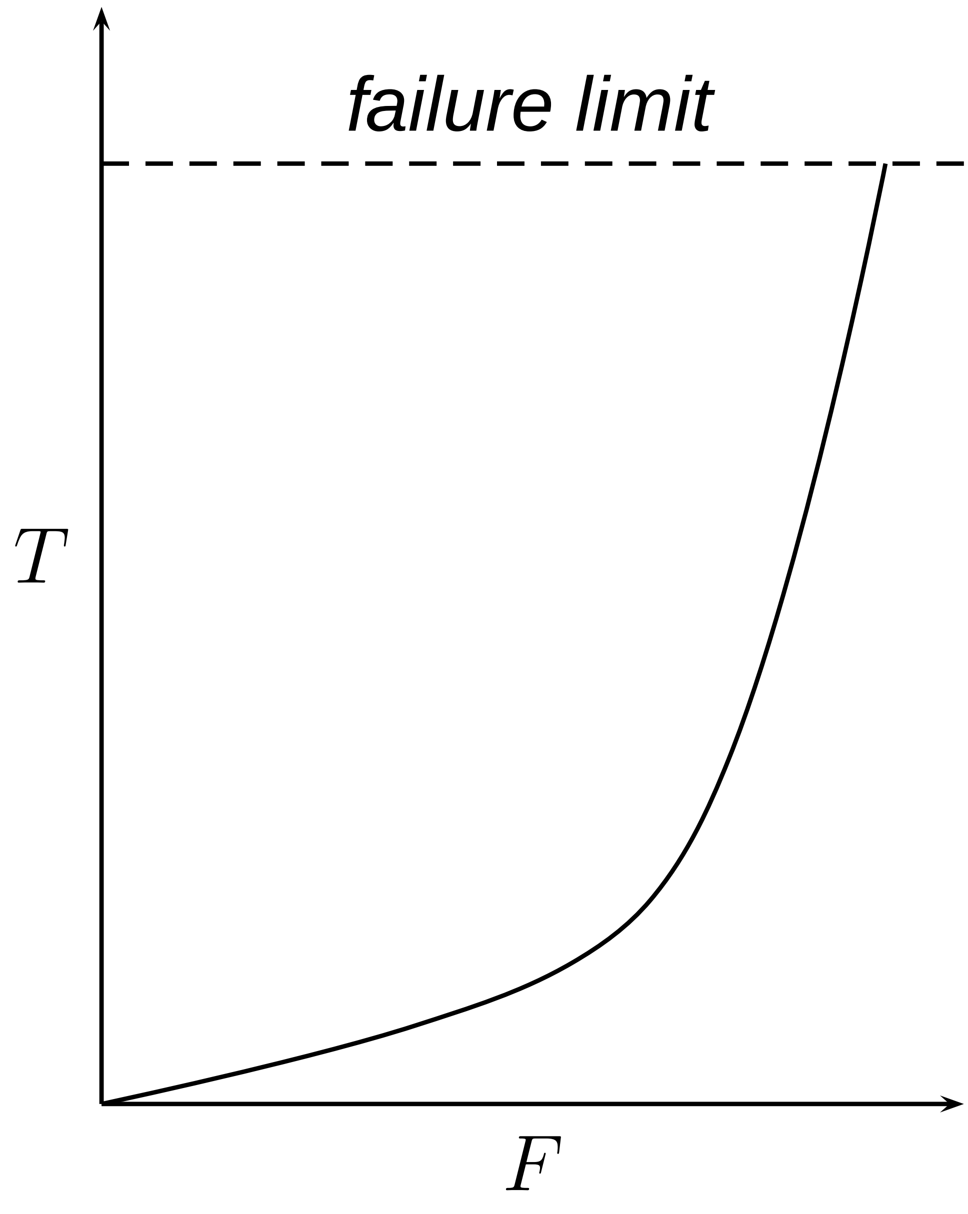


The Trustworthy Lever

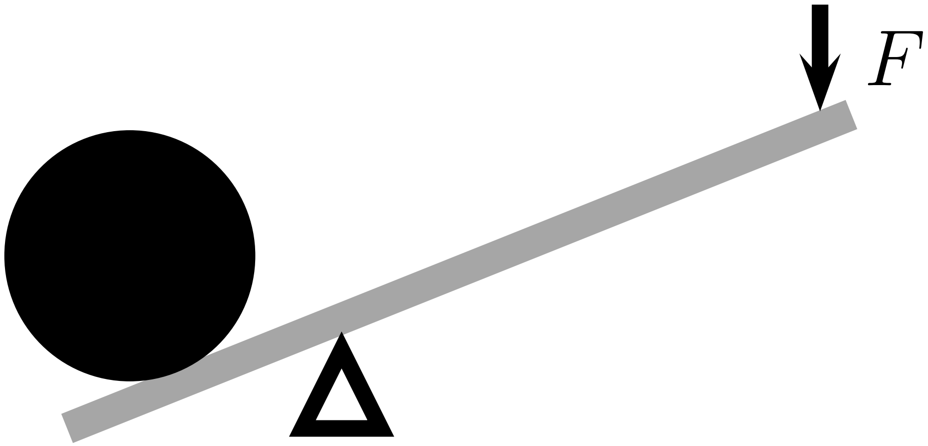


The Trustworthy Lever

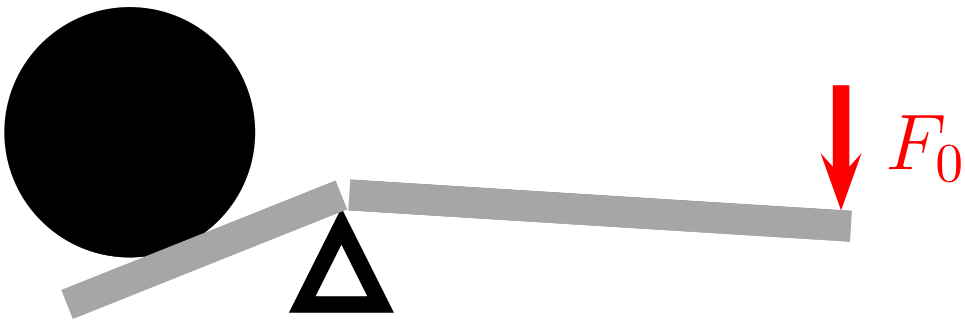




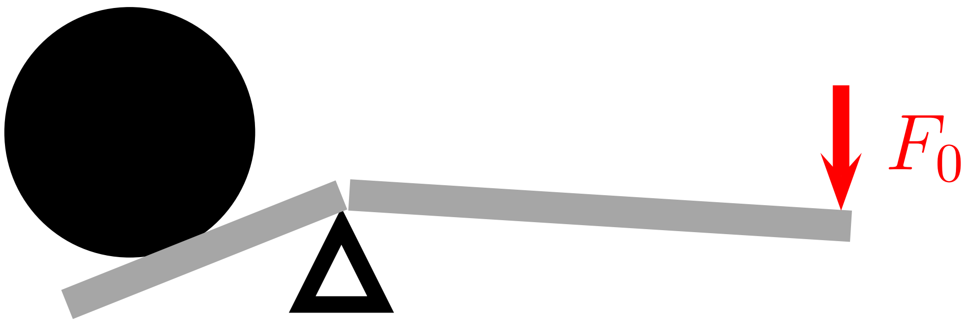
Untrustworthy Behavior

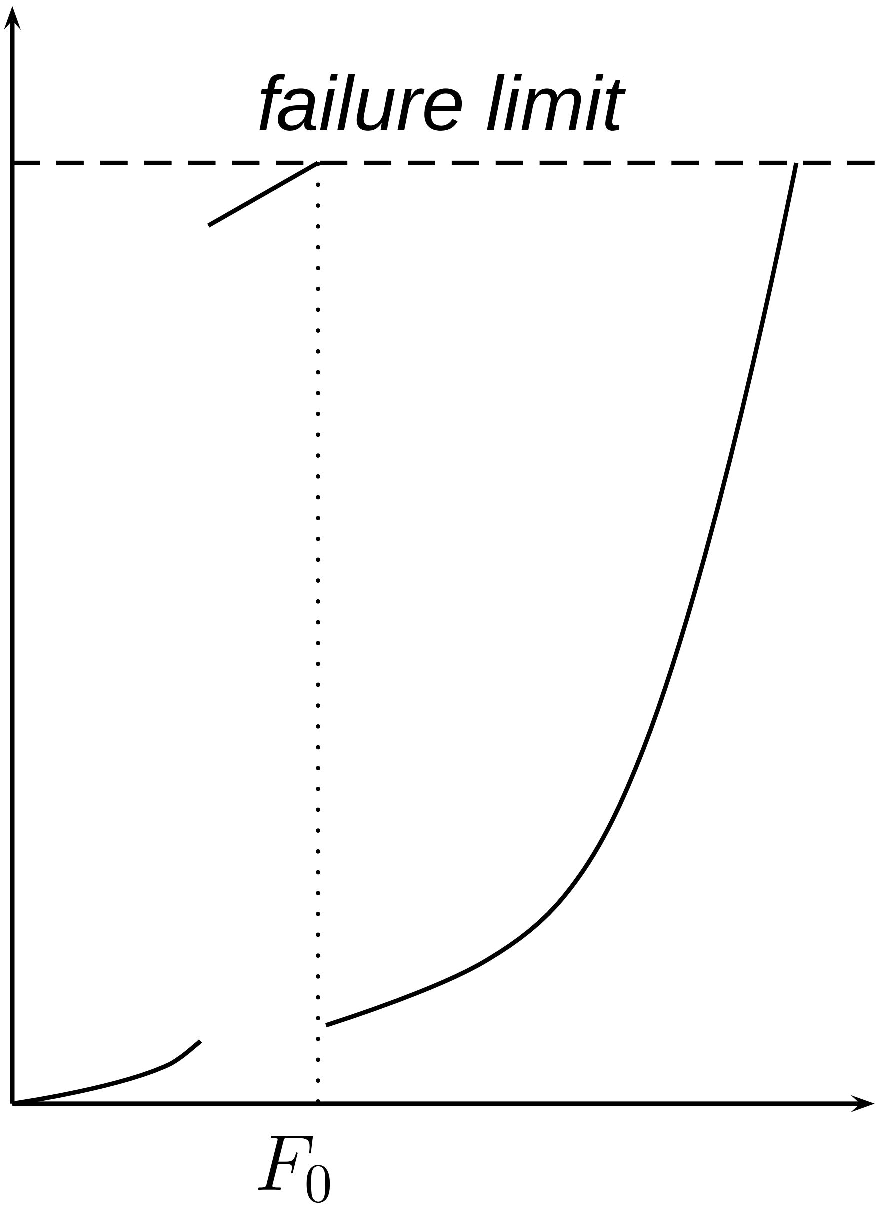


Untrustworthy Behavior



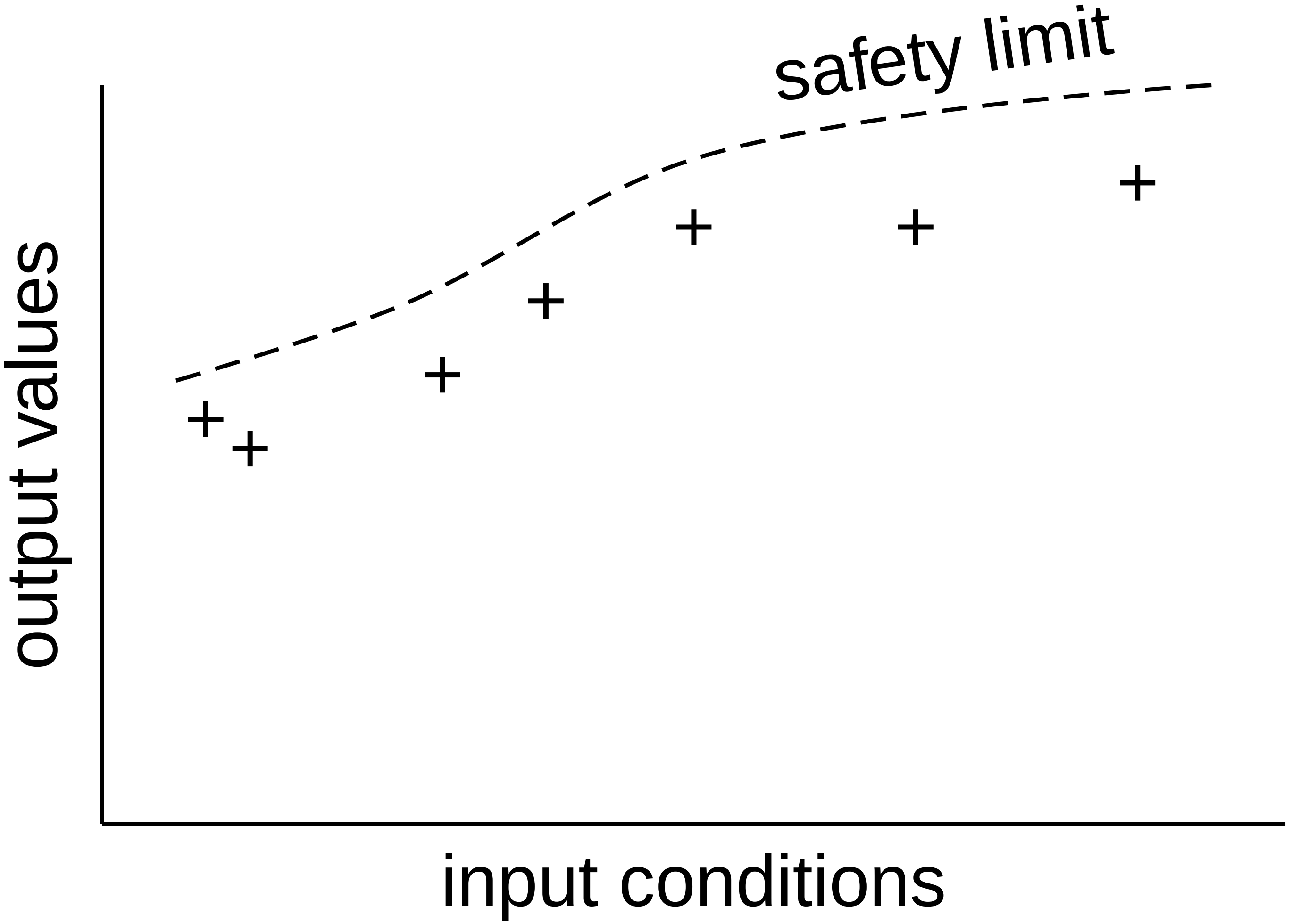
Untrustworthy Behavior





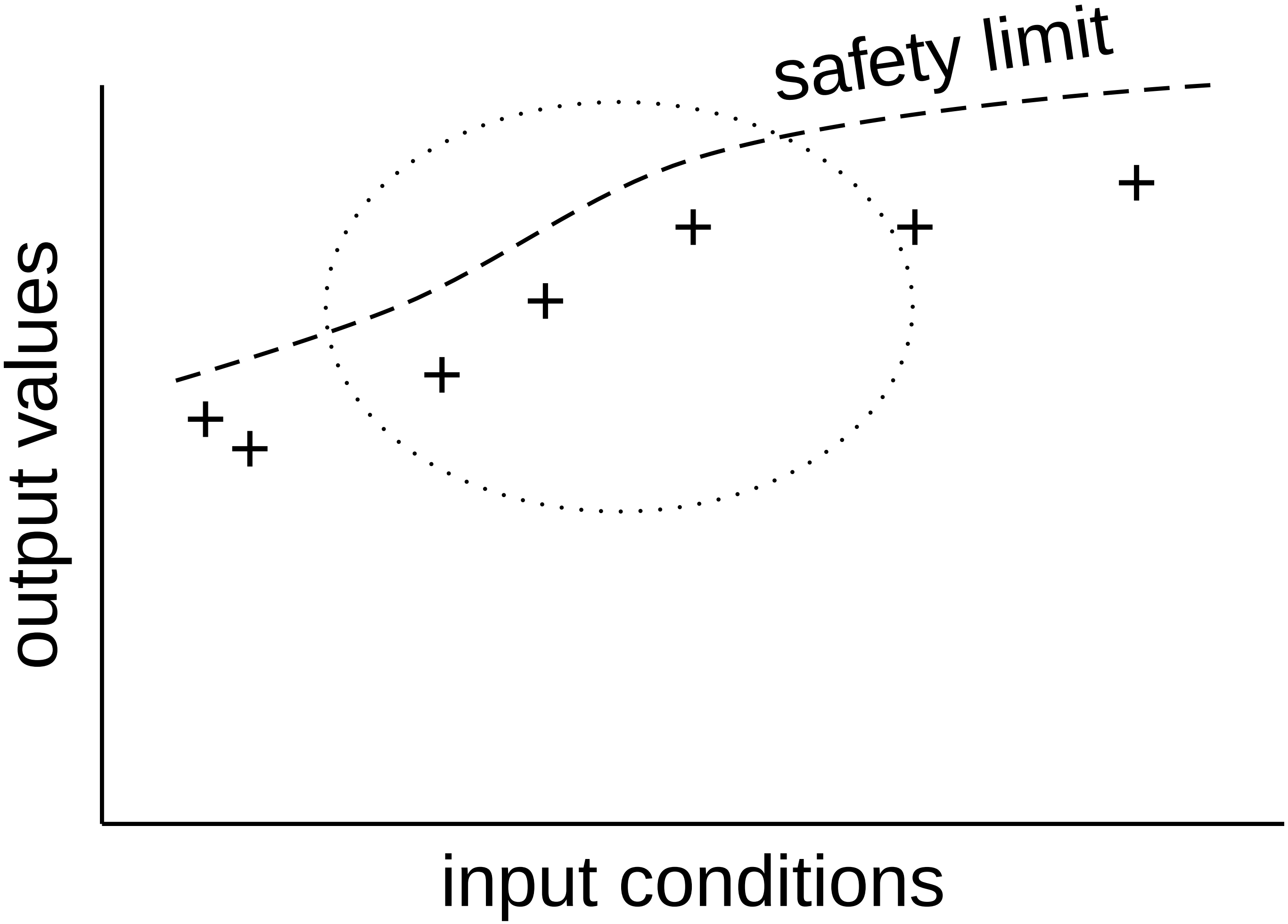
Testing a System for Trustworthiness

Sample the behavior often enough that continuity covers the space between samples



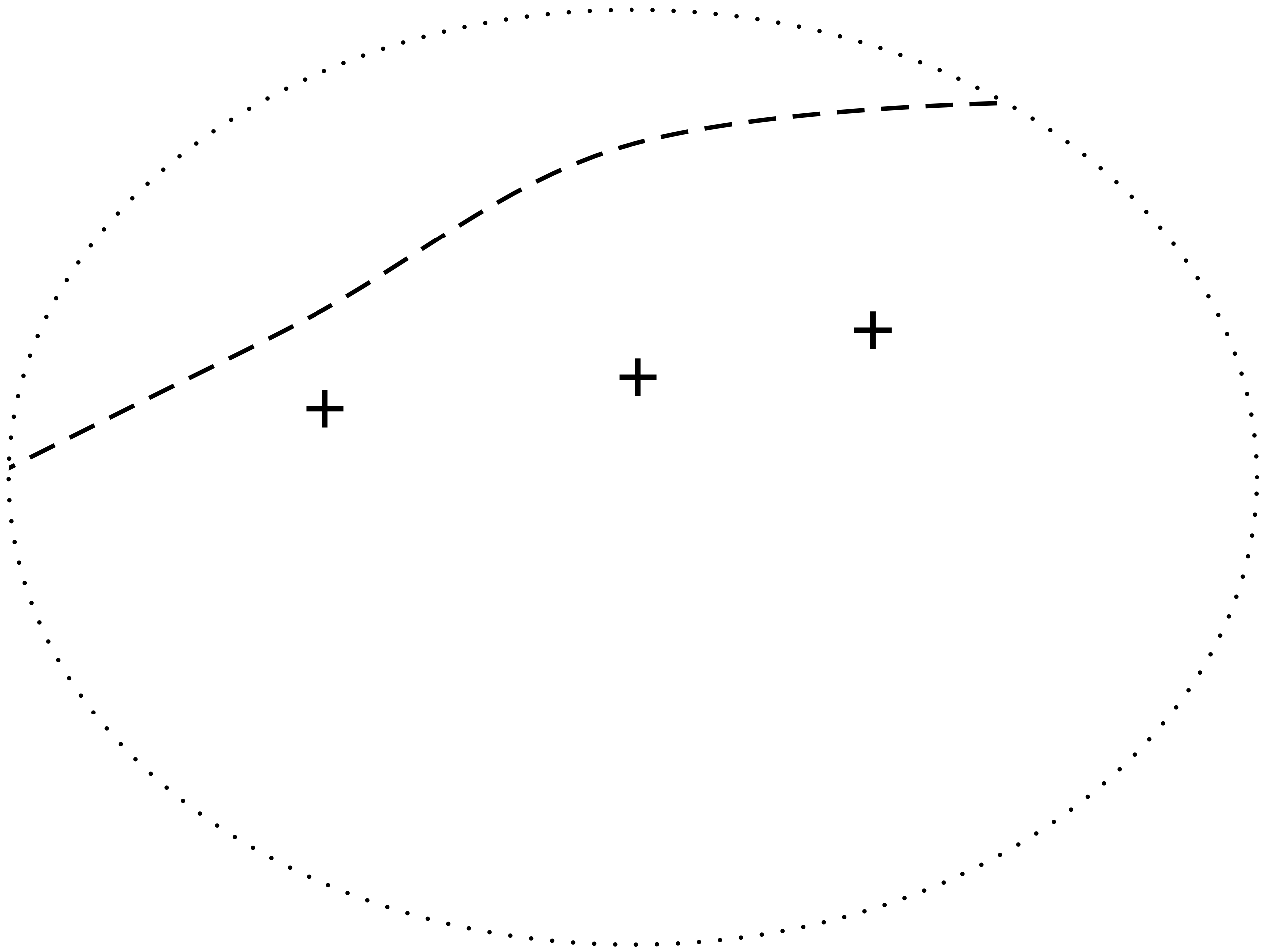
Testing a System for Trustworthiness

Sample the behavior often enough that continuity covers the space between samples



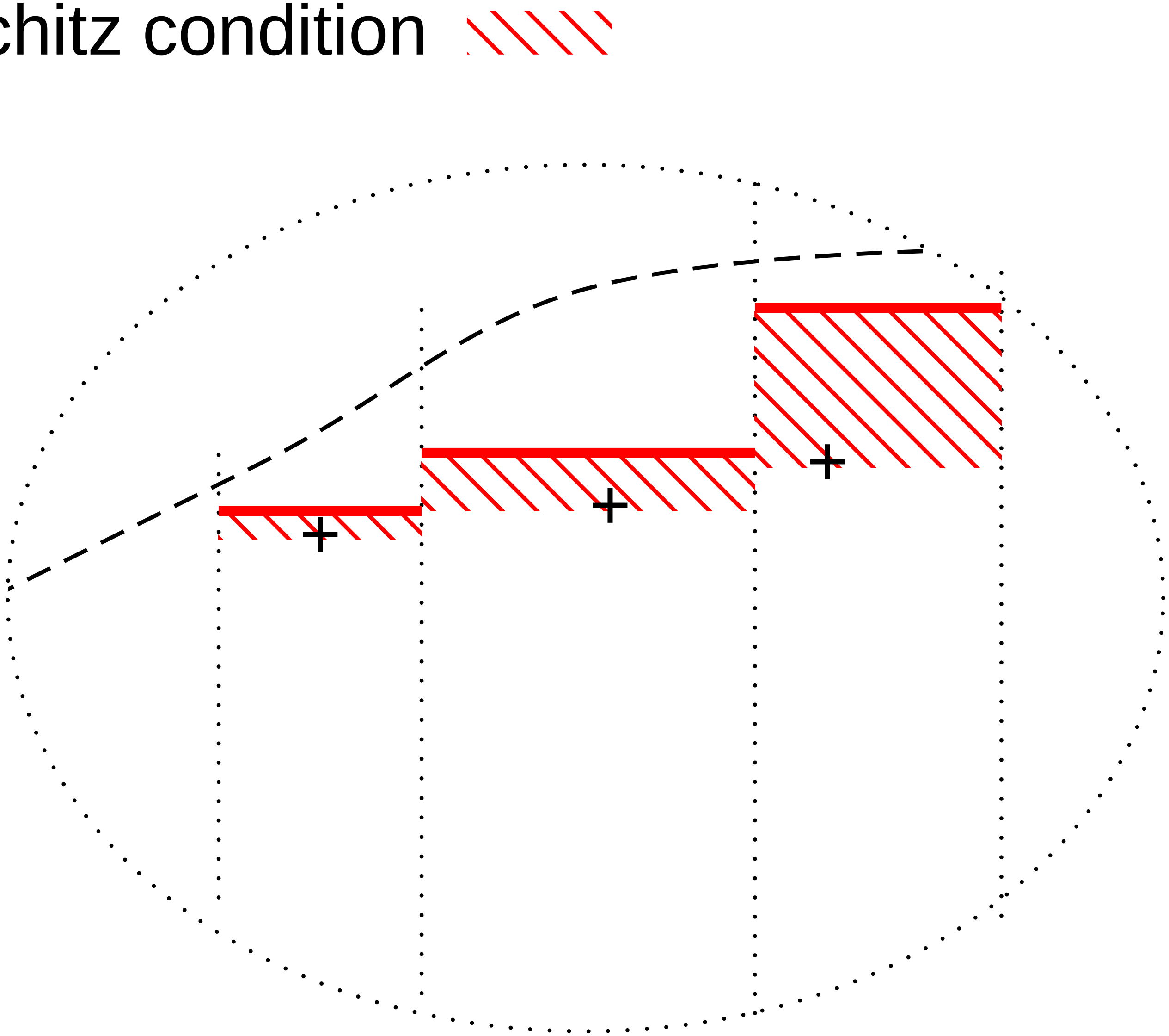
Safety Factors

Continuity isn't enough – something needed like a Lipschitz condition



Safety Factors

Continuity isn't enough – something needed like a Lipschitz condition



Outline of the Talk

I. Continuity in the Real World

II. Defining Continuity

III. Testing and Analyzing ‘Continuity’

The Real-analysis Definition

The famous '$\varepsilon - \delta$' version:

DEFINITION: A real function f is *continuous* at x_0 iff: Given any $\epsilon > 0,\ \exists\delta > 0$ such that

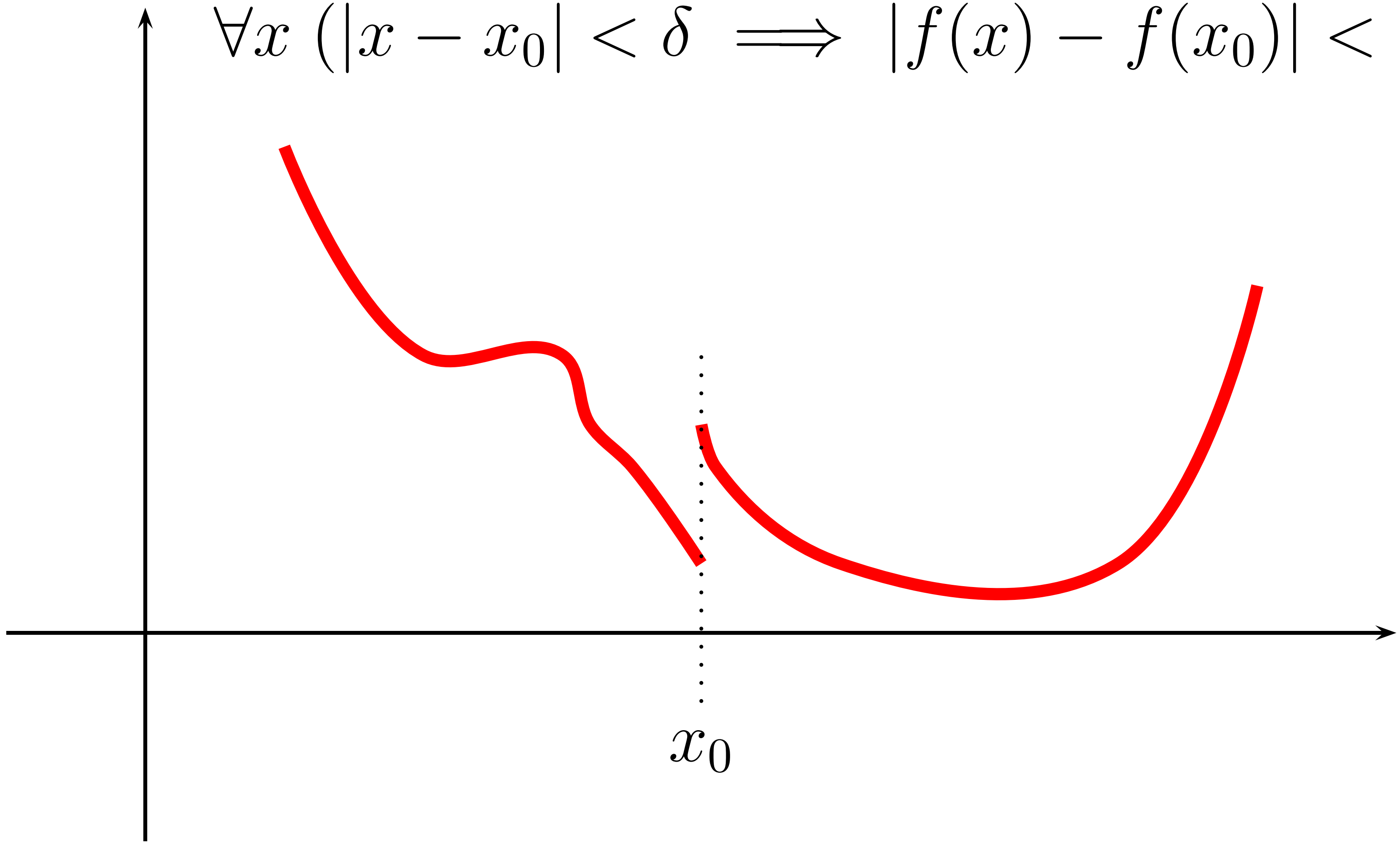
$$\forall x\ (|x - x_0| < \delta \implies |f(x) - f(x_0)| < \epsilon)$$

The Real-analysis Definition

The famous '$\varepsilon - \delta$' version:

DEFINITION: A real function f is *continuous* at x_0 iff: Given any $\epsilon > 0,\ \exists\delta > 0$ such that

$$\forall x \ (|x - x_0| < \delta \implies |f(x) - f(x_0)| < \epsilon)$$

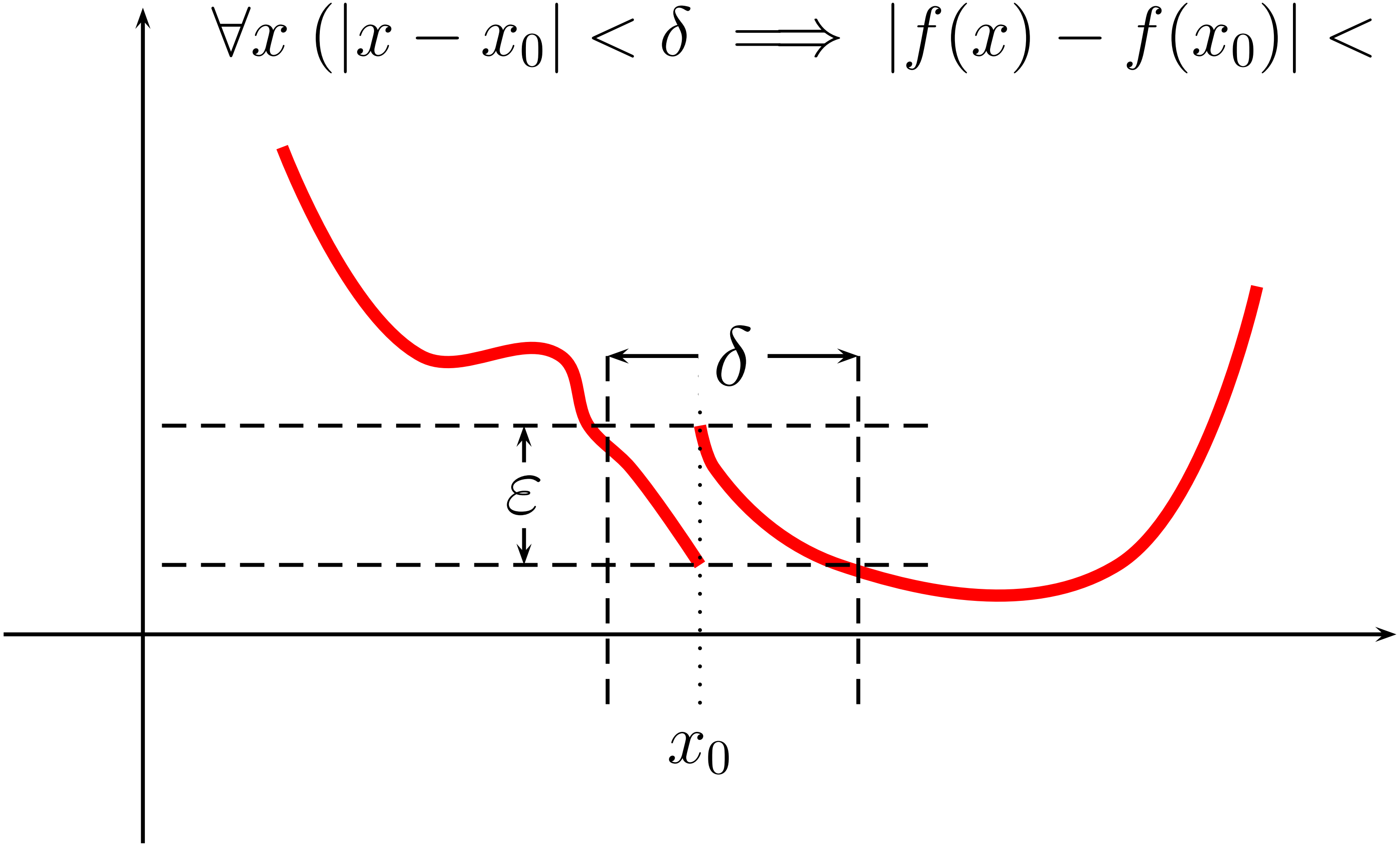


The Real-analysis Definition

The famous '$\varepsilon - \delta$' version:

DEFINITION: A real function f is *continuous* at x_0 iff: Given any $\epsilon > 0,\ \exists\delta > 0$ such that

$$\forall x\ (|x - x_0| < \delta \implies |f(x) - f(x_0)| < \epsilon)$$

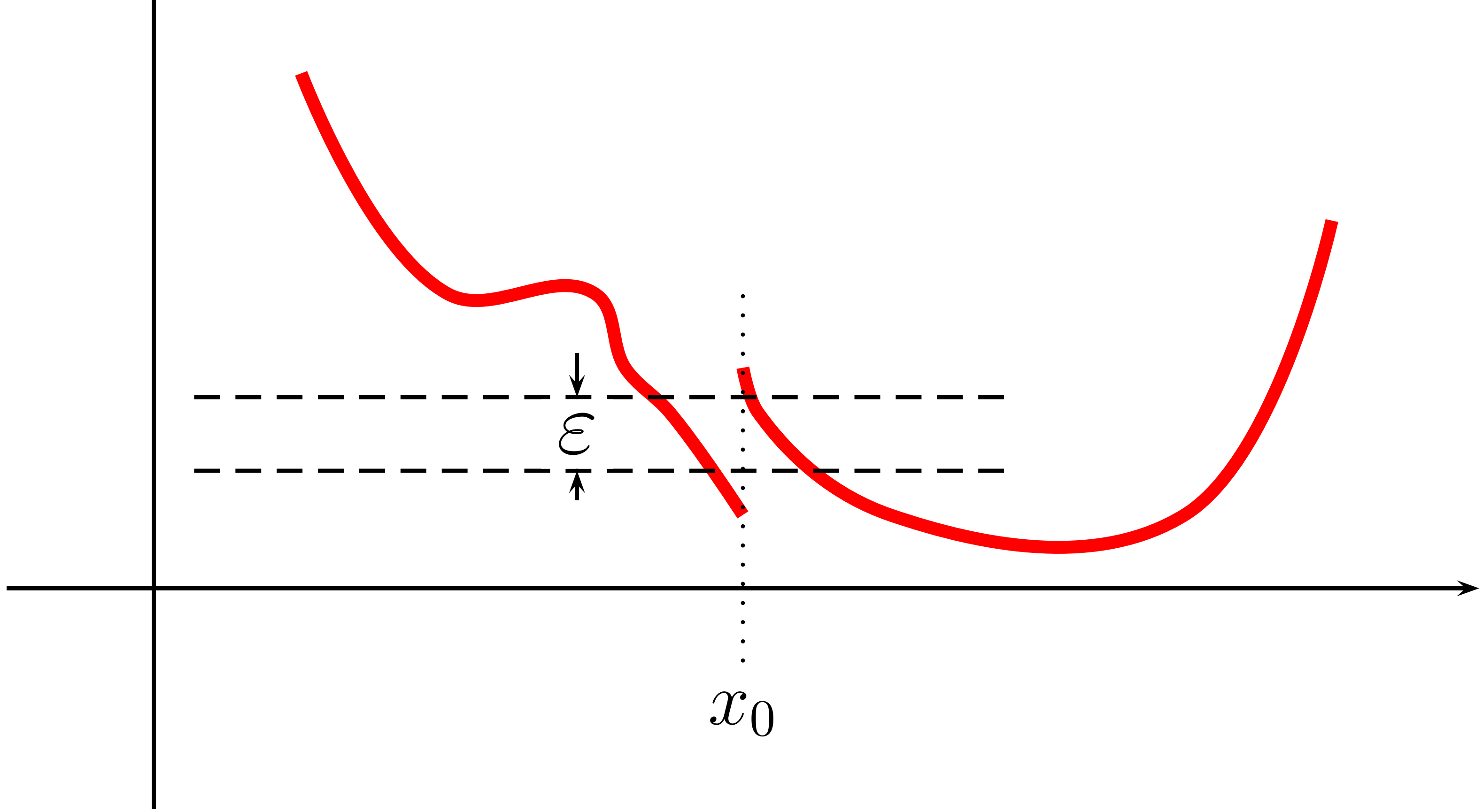


The Real-analysis Definition

The famous '$\varepsilon - \delta$' version:

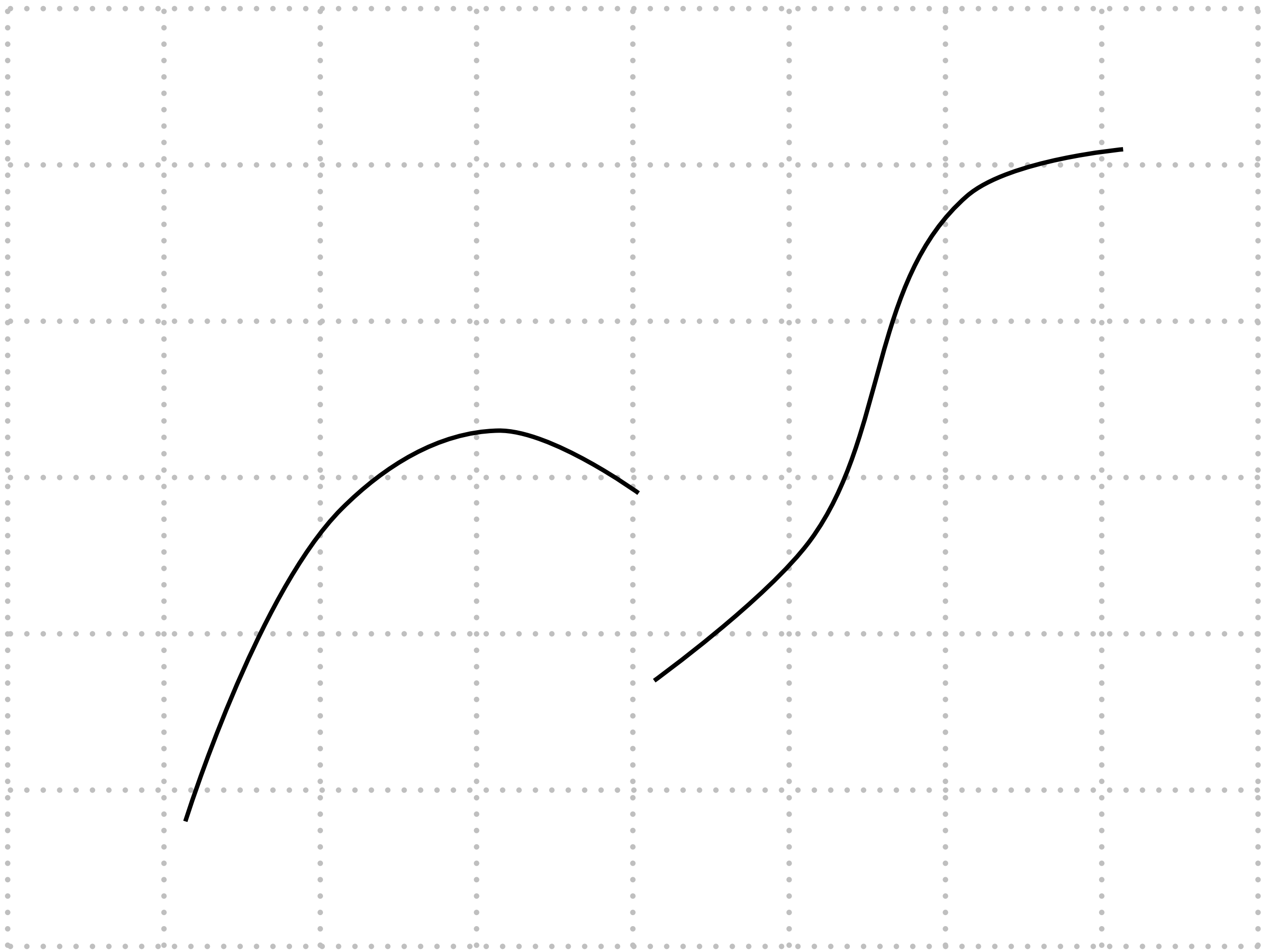
DEFINITION: A real function f is *continuous* at x_0 iff: Given any $\epsilon > 0, \ \exists\delta > 0$ such that

$$\forall x \ (|x - x_0| < \delta \implies |f(x) - f(x_0)| < \epsilon)$$



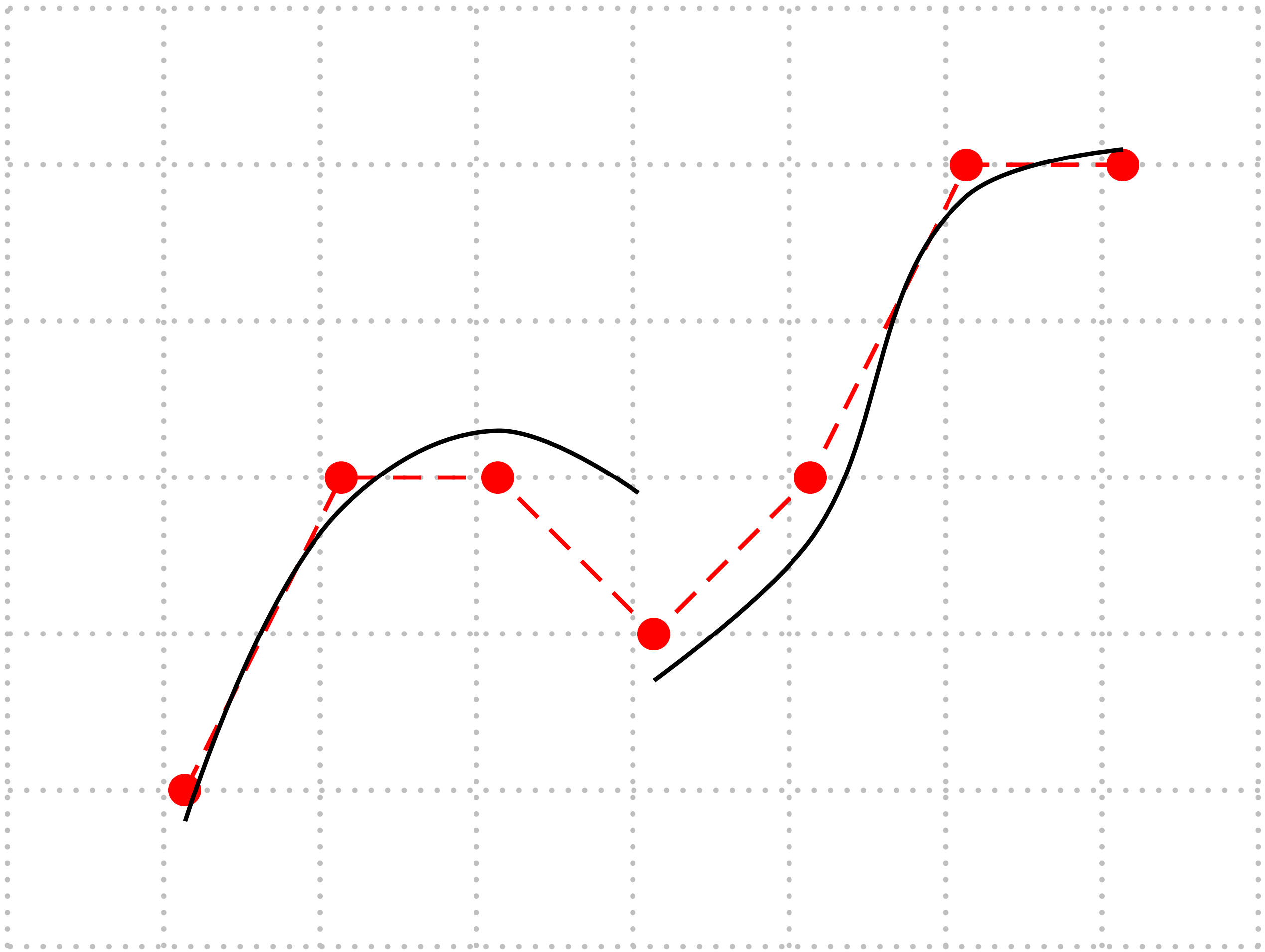
Discrete Functions

Approximating a function $f(\;)$



Discrete Functions

Approximating a function f ( )with a discrete approximation f_d ( ), $f_d(x) = \mathrm{rnd}(f(x))$, integer x



Rosenfeld's Definition

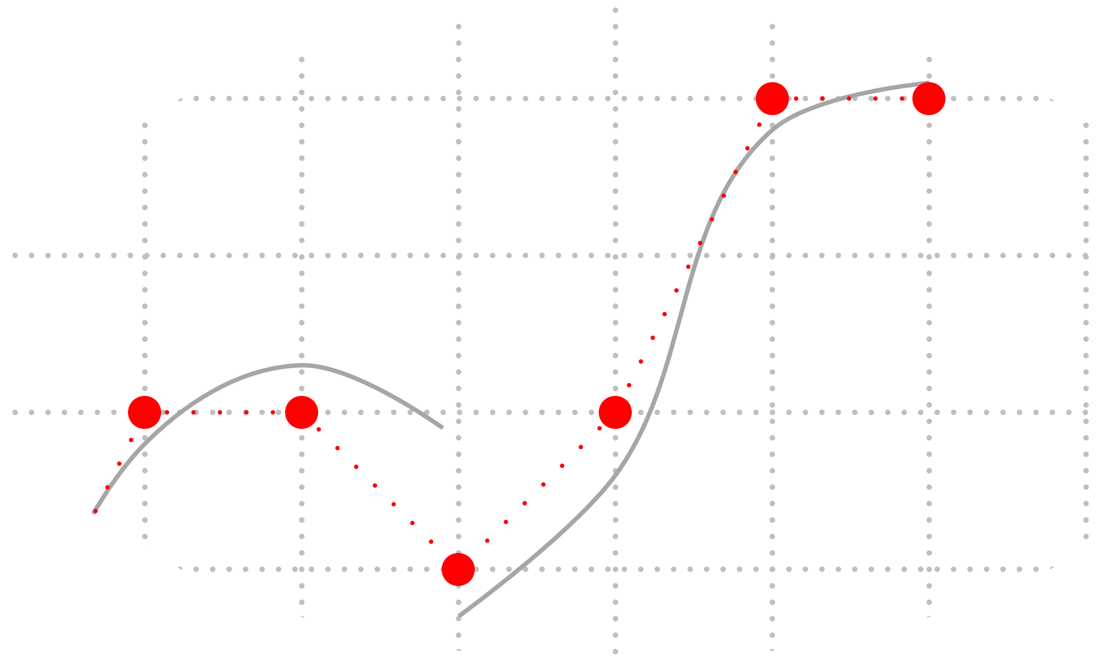
DEFINITION: An integer function f defined on a finite interval of the integers is *discretely continuous* iff:
Given any $\epsilon \geq 1,\ \exists \delta \geq 1$ such that

$$\forall x\ (|x - x_0| \leq \delta \implies |f(x) - f(x_0)| \leq \epsilon)$$

Rosenfeld's Definition

DEFINITION: An integer function f defined on a finite interval of the integers is *discretely continuous* iff:
Given any $\epsilon \geq 1, \ \exists \delta \geq 1$ such that

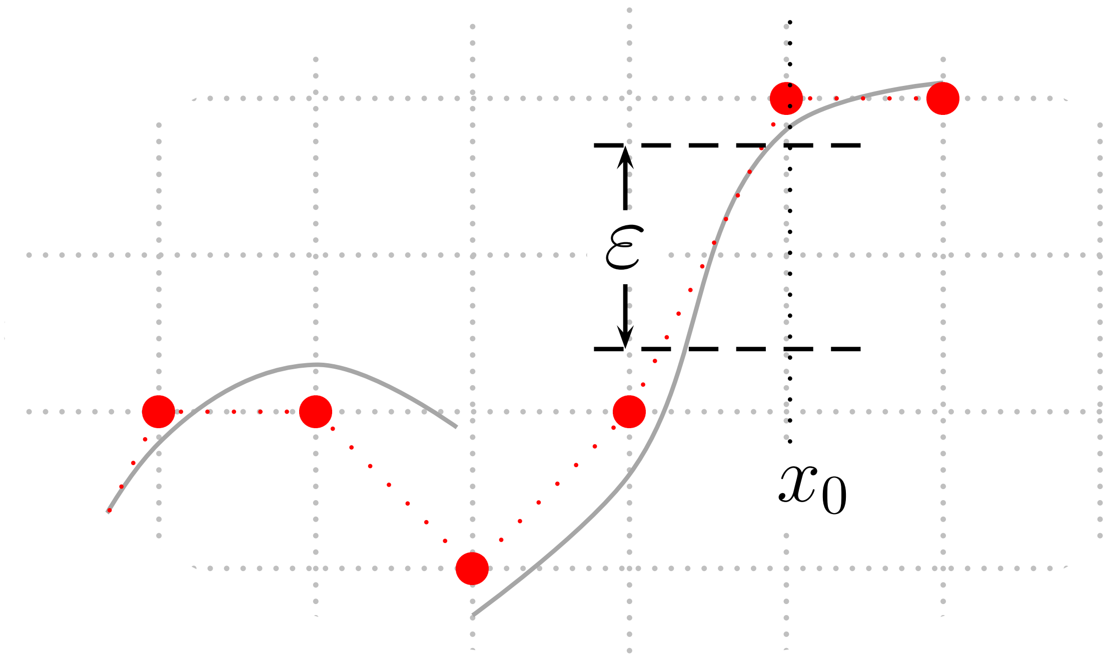
$$\forall x \ (|x - x_0| \leq \delta \implies |f(x) - f(x_0)| \leq \epsilon)$$



Rosenfeld's Definition

DEFINITION: An integer function f defined on a finite interval of the integers is *discretely continuous* iff:
Given any $\epsilon \geq 1, \; \exists \delta \geq 1$ such that

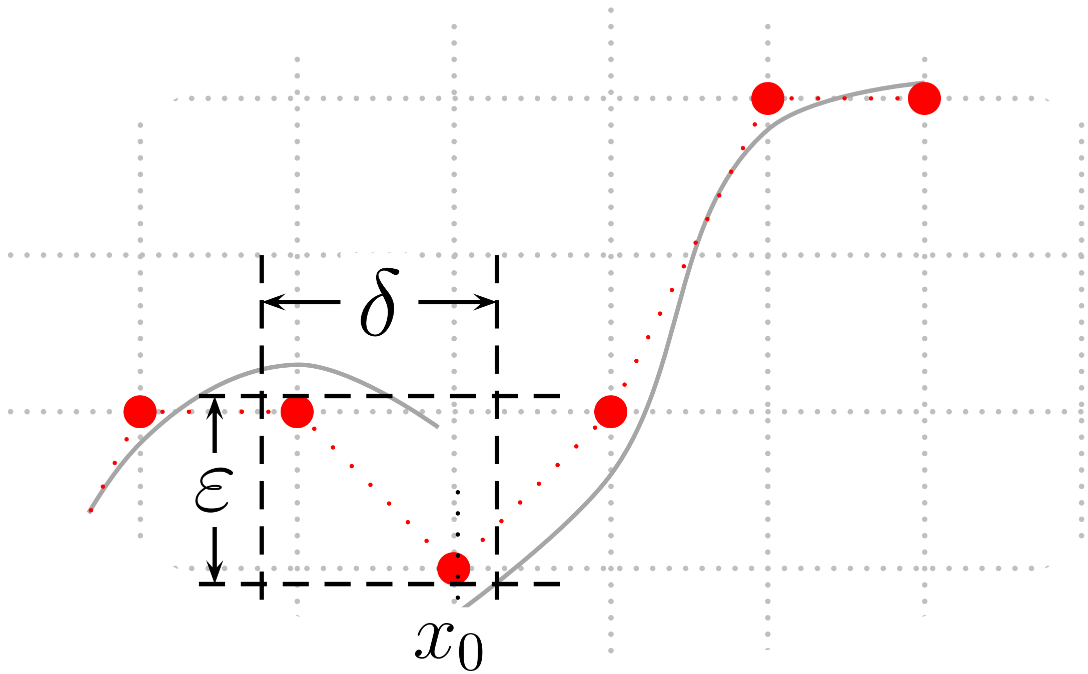
$$\forall x \; (|x - x_0| \leq \delta \implies |f(x) - f(x_0)| \leq \epsilon)$$



Rosenfeld's Definition

DEFINITION: An integer function f defined on a finite interval of the integers is *discretely continuous* iff:
Given any $\epsilon \geq 1, \ \exists \delta \geq 1$ such that

$$\forall x \ (|x - x_0| \leq \delta \implies |f(x) - f(x_0)| \leq \epsilon)$$



Surprises?

The discretely continuous functions:

- have the intermediate value property:
  if $f(x) < m < f(y), \exists z$ such that $f(z) = m$

Surprises?

The discretely continuous functions:

- have the intermediate value property:
  if $f(x) < m < f(y), \exists z$ such that $f(z) = m$
- are closed under composition

Surprises?

The discretely continuous functions:

- have the intermediate value property: if $f(x) < m < f(y), \exists z$ such that $f(z) = m$
- are closed under composition
- are *not closed* under arithmetic operations

Surprises?

The discretely continuous functions:

- have the intermediate value property:
  if $f(x) < m < f(y), \exists z$ such that $f(z) = m$
- are closed under composition
- are *not closed* under arithmetic operations
  - Let $f(x) = x$, for which f_d is discretely continuous everywhere. But $f_d + f_d$ is nowhere discretely continuous.

Floating-point Continuity

A program "computes f to within 1%":

- For all real x, program inputs will approximate x with error at most δ_x, and for all input values t such that $|x - t| < \delta_x$ the program output v_t at t will satisfy $|(f(x) - v_t)/f(x)| < .01$

DEFINITION: The function F computed by a program is floating-point continuous iff it approximates a continuous function to the accuracy of the program's specification.

- Floating-point continuity: almost discrete continuity 'scaled' by floating-point granularity

Failure Continuity

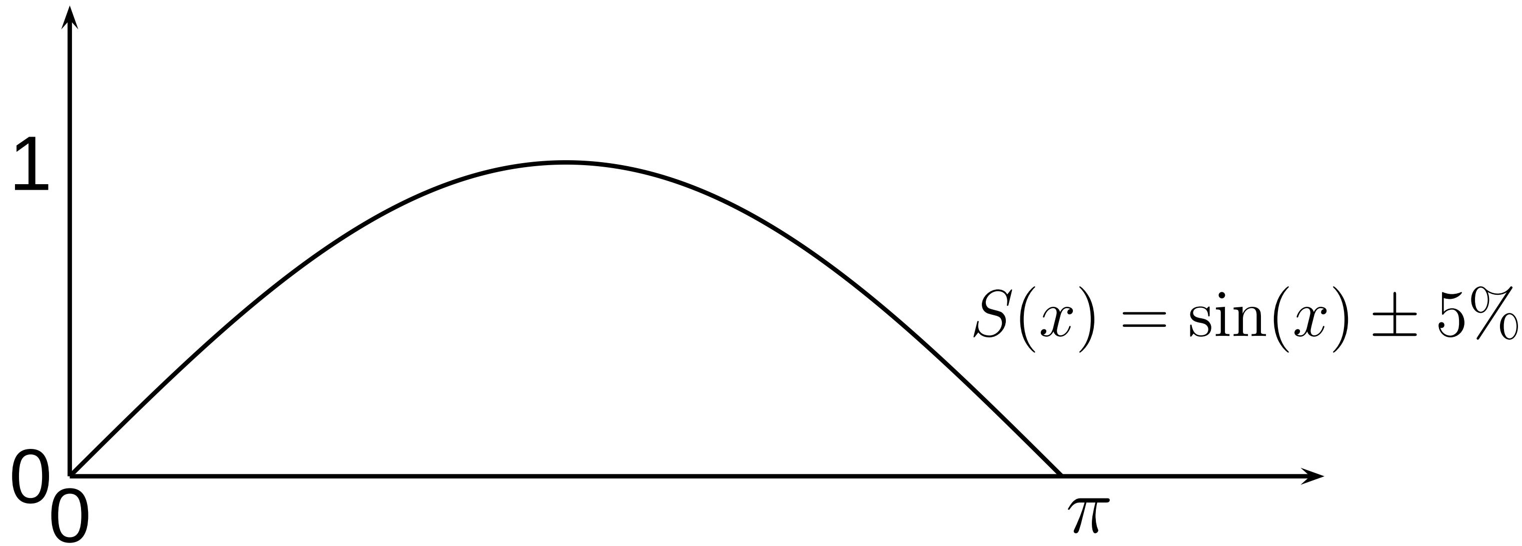
DEFINITION: Program P has specification S. P is *failure continuous* at x_0 iff $\exists b > 0$ such that:

$$P(x_0) \neq S(x_0) \implies \forall t, |x_0 - t| < b \; (P(t) \neq S(t))$$

Failure Continuity

DEFINITION: Program P has specification S. P is *failure continuous* at x_0 iff $\exists b > 0$ such that:

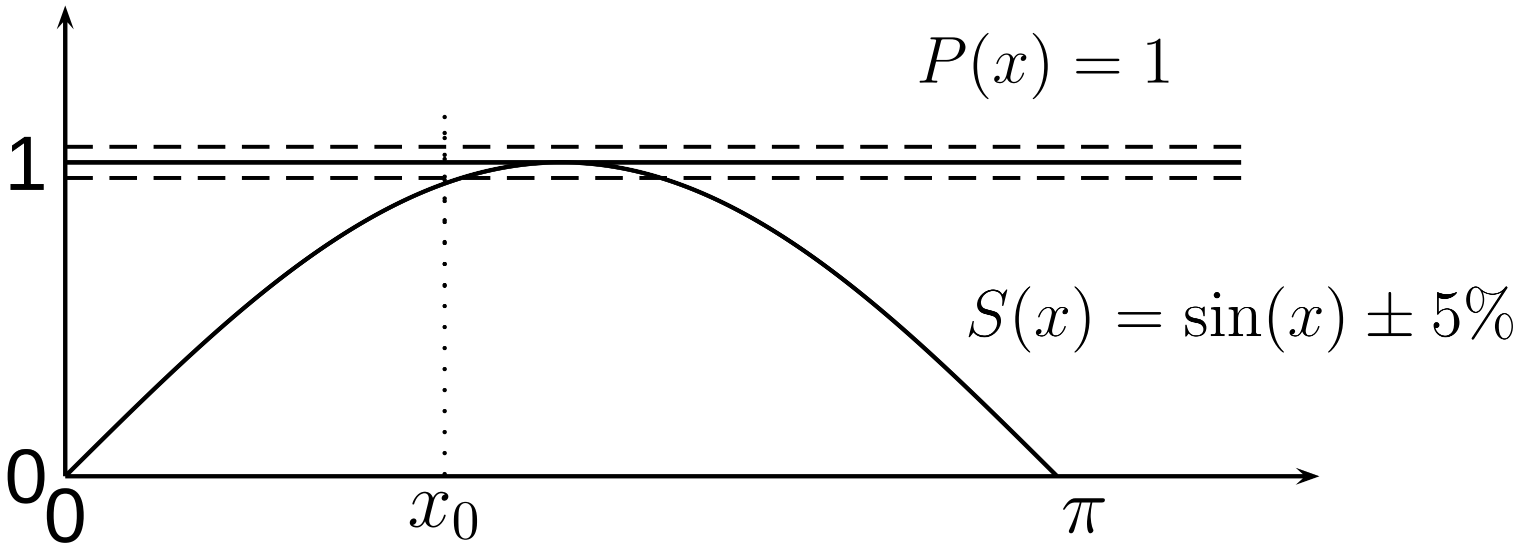
$$P(x_0) \neq S(x_0) \implies \forall t, |x_0 - t| < b \; (P(t) \neq S(t))$$



Failure Continuity

DEFINITION: Program P has specification S. P is *failure continuous* at x_0 iff $\exists b > 0$ such that:

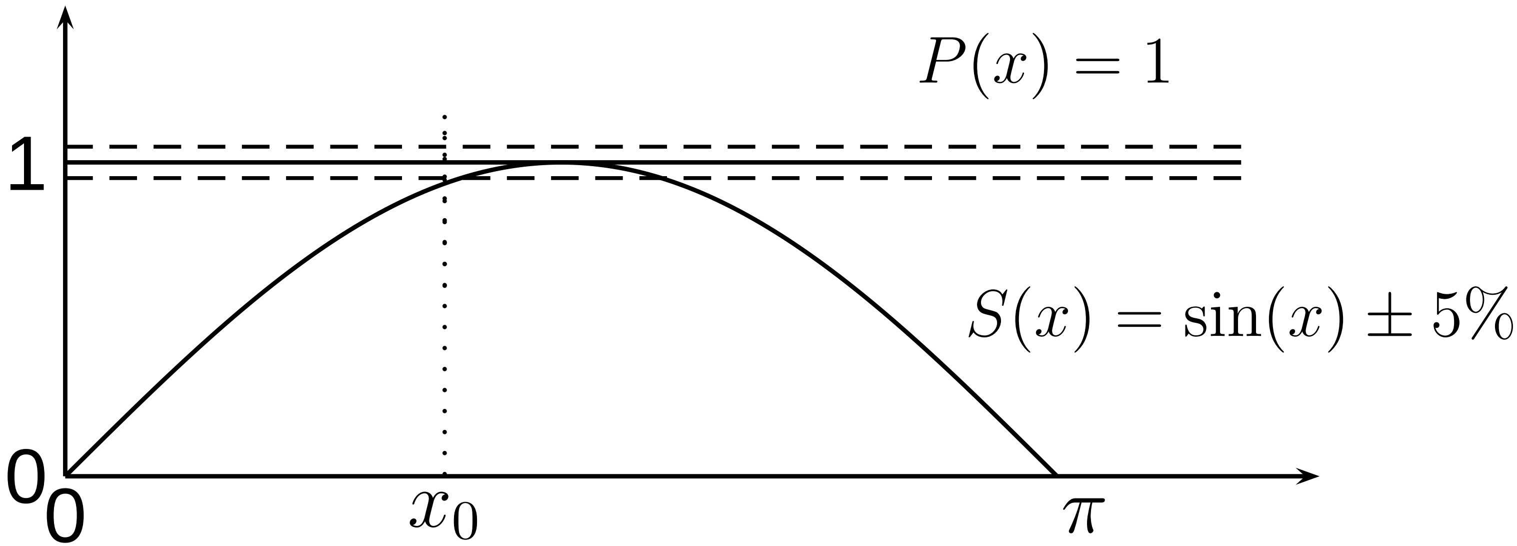
$$P(x_0) \neq S(x_0) \implies \forall t, |x_0 - t| < b \; (P(t) \neq S(t))$$



Failure Continuity

DEFINITION: Program P has specification S. P is *failure continuous* at x_0 iff $\exists b > 0$ such that:

$$P(x_0) \neq S(x_0) \implies \forall t, |x_0 - t| < b \; (P(t) \neq S(t))$$



- Failure continuity is what Howden's 'reliable' subdomains have

Program Analysis with Reals Justified

- Program variables are not the real variables we pretend they are

CONJECTURE: If a program computes by symbolic execution a continuous real-valued function, then: (1) The program is discretely continuous over a suitable interval, and (2) There is a specification accuracy for which the program is floating-point continuous.

Proof? Choose the interval or the required accuracy to be as poor as necessary.

Program Analysis with Reals Justified

- Program variables are not the real variables we pretend they are

CONJECTURE: If a program computes by symbolic execution a continuous real-valued function, then: (1) The program is discretely continuous over a suitable interval, and (2) There is a specification accuracy for which the program is floating-point continuous.

Proof? Choose the interval or the required accuracy to be as poor as necessary.

- The converse is false

Outline of the Talk

I. Continuity in the Real World

II. Defining Continuity

III. Testing and Analyzing ‘Continuity’

Testing a Program for Continuity

- Imperative conditional statements are the source of discontinuity
- On each path subdomain, programs compute a real-variable polynomial
  - Potential discontinuities can occur only on path-subdomain boundaries
  - Testing for continuity across a boundary requires no oracle

Testing a Program for Continuity

- Imperative conditional statements are the source of discontinuity
- On each path subdomain, programs compute a real-variable polynomial
  - Potential discontinuities can occur only on path-subdomain boundaries
  - Testing for continuity across a boundary requires no oracle
- Functional languages might be better – program continuities are closed under composition

Ideas to Explore in Continuity Analysis

Suppose a program for a continuous specification *is* continuous.

What new kinds of analysis are possible?

- With Lipschitz conditions, good behavior on test points spaced at some Δ guarantees correctness

Ideas to Explore in Continuity Analysis

Suppose a program for a continuous specification *is* continuous.

What new kinds of analysis are possible?

- With Lipschitz conditions, good behavior on test points spaced at some Δ guarantees correctness
- "Random structural testing" is a name for using a uniform profile on each Lipschitz neighborhood – it may not be intractable in the ultrareliable region

Ideas to Explore in Continuity Analysis

Suppose a program for a continuous specification *is* continuous.

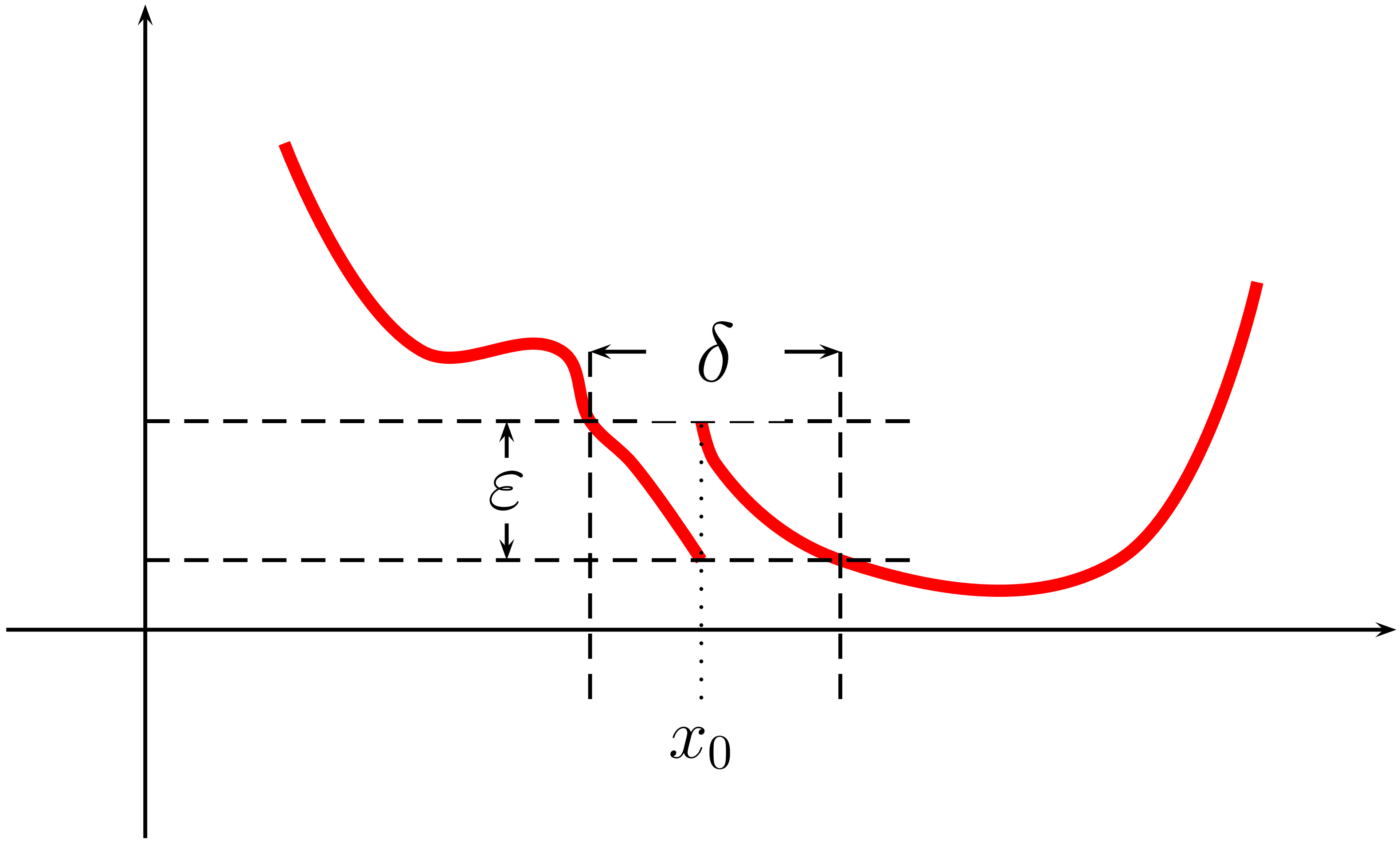
What new kinds of analysis are possible?

- With Lipschitz conditions, good behavior on test points spaced at some Δ guarantees correctness
- “Random structural testing” is a name for using a uniform profile on each Lipschitz neighborhood – it may not be intractable in the ultrareliable region
- Exploit continuity in the self-testing/correcting methods of Blum and Ammann

Inherent Discontinuity

- Continuous specifications are important
  - Flight- and process-control software
  - Simulations of natural systems
  - Regulatory-agency problems with software replacing hardware
- But software's forté is *discontinuous* specifications that no other technology can handle
  - Chess-playing robots
  - Compilers and other character-based processors

QUESTIONS?



QUESTIONS?