Subdivision Approximation for Curves and Integral Analysis

Chee Yap
Courant Institute of Mathematical Sciences
New York University

Joint work with
M.Burr, S.Choi, B.Galehouse, F.Krahmer, V.Sharma

ABSTRACT

Geometric operations on curves and surfaces can be based on algebraic techniques (e.g., cylindrical algebraic decomposition, resultants) or on numerical/geometric techniques (e.g., subdivision methods, marching cubes). The latter techniques have adaptive complexity but are usually incomplete. To achieve completeness, hybrid techniques which combine numerical with algebraic techniques are usually used. In this talk, we focus on purely numerical techniques.

Vegter-Plantinga gave the first numerical subdivision algorithm that is guaranteed to compute isotopic approximations for implicit curves and surfaces that are non-singular. The computational model is non-algebraic, using only evaluation functions and the interval evaluation of functions and their derivatives. We show how to achieve isotopic approximation of implicit curves with isolated singularities within the Vegter-Plantinga model.

The complexity analysis of adaptive algorithms is an major challenge. We shall consider the 1-D version of their algorithm: this amounts to real root isolation. We introduce general framework and a novel integral formula for the complexity of EVAL, a version of their root isolation algorithm. We also show that for the benchmark problem where the input polynomial f is a integer polynomial of degree d with L-bit coefficient size, EVAL has $O(d^2L)$ complexity.

Our analysis technique might be called "continuous amortization argument", and exploits an evaluation-form of the Mahler-Davenport type bounds.

PART I.
Introduction

Computational Curves and Surfaces

- TWO APPROACHES
  - ∗ Algebraic Approach
  - ∗ Numerical/Geometric Approach

- PROS and CONS
  - ∗ ALGEBRAIC: robust and complete BUT inefficent, hard-to-implement, non-local
  - ∗ GEOMETRIC: fast, simple-to-implement, local BUT incomplete and non-robust
  - ∗ HYBRID METHOD: e.g., subdivision WITH algebraic primitives

- SUBDIVISION METHODs in Geometric Approaches
  - ∗ STRONG FORM: Snyder, Mourrain
  - ∗ WEAK FORM: Vegter-Plantinga (see below)

Computational Curves and Surfaces

- TWO APPROACHES
  - ∗ Algebraic Approach
  - ∗ Numerical/Geometric Approach

- PROS and CONS
  - ∗ ALGEBRAIC: robust and complete BUT inefficent, hard-to-implement, non-local
  - ∗ GEOMETRIC: fast, simple-to-implement, local BUT incomplete and non-robust
  - ∗ HYBRID METHOD: e.g., subdivision WITH algebraic primitives

- SUBDIVISION METHODs in Geometric Approaches
  - ∗ STRONG FORM: Snyder, Mourrain
  - ∗ WEAK FORM: Vegter-Plantinga (see below)

Computational Curves and Surfaces

- TWO APPROACHES
  - ∗ Algebraic Approach
  - ∗ Numerical/Geometric Approach

- PROS and CONS
  - ∗ ALGEBRAIC: robust and complete BUT inefficent, hard-to-implement, non-local
  - ∗ GEOMETRIC: fast, simple-to-implement, local BUT incomplete and non-robust
  - ∗ HYBRID METHOD: e.g., subdivision WITH algebraic primitives

- SUBDIVISION METHODs in Geometric Approaches
  - ∗ STRONG FORM: Snyder, Mourrain
  - ∗ WEAK FORM: Vegter-Plantinga (see below)

Computational Curves and Surfaces

- TWO APPROACHES
  - $*$ Algebraic Approach
  - $*$ Numerical/Geometric Approach

- PROS and CONS
  - $*$ ALGEBRAIC: robust and complete BUT inefficent, hard-to-implement, non-local
  - $*$ GEOMETRIC: fast, simple-to-implement, local BUT incomplete and non-robust
  - $*$ HYBRID METHOD: e.g., subdivision WITH algebraic primitives

- SUBDIVISION METHODs in Geometric Approaches
  - $*$ STRONG FORM: Snyder, Mourrain
  - $*$ WEAK FORM: Vegter-Plantinga (see below)

General Objectives (CHALLENGES)

- OF COURSE...
  - ∗ Correct, Robust, Efficient

- (A) ADAPTIVE METHODS
  - ∗ Fast on typical or non-degenerate inputs
  - ∗ THIS TALK: subdivision method

- (B) PURELY NUMERICAL METHODS
  - ∗ No manipulation of algebraic numbers, resultants, root isolation
  - ∗ ADVANTAGES: simpler, applies to non-algebraic geometry
  - ∗ THIS TALK: purely numerical subdivision

- (C) COMPLETE METHODS
  - ∗ AVOID Assumptions on Inputs
  - ∗ E.g., non-singularity, Morseness, general position
  - ∗ THIS TALK: isolated singularities

- (D) COMPLEXITY ANALYSIS
  - ∗ Need intrinsic measures: E.g., condition numbers, precision-sensitivity
  - ∗ THIS TALK: integral analysis

General Objectives (CHALLENGES)

- OF COURSE...
  - ∗ Correct, Robust, Efficient
- (A) ADAPTIVE METHODS
  - ∗ Fast on typical or non-degenerate inputs
  - ∗ THIS TALK: subdivision method
- (B) PURELY NUMERICAL METHODS
  - ∗ No manipulation of algebraic numbers, resultants, root isolation
  - ∗ ADVANTAGES: simpler, applies to non-algebraic geometry
  - ∗ THIS TALK: purely numerical subdivision
- (C) COMPLETE METHODS
  - ∗ AVOID Assumptions on Inputs
  - ∗ E.g., non-singularity, Morseness, general position
  - ∗ THIS TALK: isolated singularities
- (D) COMPLEXITY ANALYSIS
  - ∗ Need intrinsic measures: E.g., condition numbers, precision-sensitivity
  - ∗ THIS TALK: integral analysis

General Objectives (CHALLENGES)

- OF COURSE...
  - ∗ Correct, Robust, Efficient
- (A) ADAPTIVE METHODS
  - ∗ Fast on typical or non-degenerate inputs
  - ∗ THIS TALK: subdivision method
- (B) PURELY NUMERICAL METHODS
  - ∗ No manipulation of algebraic numbers, resultants, root isolation
  - ∗ ADVANTAGES: simpler, applies to non-algebraic geometry
  - ∗ THIS TALK: purely numerical subdivision
- (C) COMPLETE METHODS
  - ∗ AVOID Assumptions on Inputs
  - ∗ E.g., non-singularity, Morseness, general position
  - ∗ THIS TALK: isolated singularities
- (D) COMPLEXITY ANALYSIS
  - ∗ Need intrinsic measures: E.g., condition numbers, precision-sensitivity
  - ∗ THIS TALK: integral analysis

General Objectives (CHALLENGES)

- OF COURSE...
  - ∗ Correct, Robust, Efficient
- (A) ADAPTIVE METHODS
  - ∗ Fast on typical or non-degenerate inputs
  - ∗ THIS TALK: subdivision method
- (B) PURELY NUMERICAL METHODS
  - ∗ No manipulation of algebraic numbers, resultants, root isolation
  - ∗ ADVANTAGES: simpler, applies to non-algebraic geometry
  - ∗ THIS TALK: purely numerical subdivision
- (C) COMPLETE METHODS
  - ∗ AVOID Assumptions on Inputs
  - ∗ E.g., non-singularity, Morseness, general position
  - ∗ THIS TALK: isolated singularities
- (D) COMPLEXITY ANALYSIS
  - ∗ Need intrinsic measures: E.g., condition numbers, precision-sensitivity
  - ∗ THIS TALK: integral analysis

General Objectives (CHALLENGES)

- OF COURSE...
  - ∗ Correct, Robust, Efficient
- (A) ADAPTIVE METHODS
  - ∗ Fast on typical or non-degenerate inputs
  - ∗ THIS TALK: subdivision method
- (B) PURELY NUMERICAL METHODS
  - ∗ No manipulation of algebraic numbers, resultants, root isolation
  - ∗ ADVANTAGES: simpler, applies to non-algebraic geometry
  - ∗ THIS TALK: purely numerical subdivision
- (C) COMPLETE METHODS
  - ∗ AVOID Assumptions on Inputs
  - ∗ E.g., non-singularity, Morseness, general position
  - ∗ THIS TALK: isolated singularities
- (D) COMPLEXITY ANALYSIS
  - ∗ Need intrinsic measures: E.g., condition numbers, precision-sensitivity
  - ∗ THIS TALK: integral analysis

General Objectives (CHALLENGES)

- OF COURSE...
  - ∗ Correct, Robust, Efficient
- (A) ADAPTIVE METHODS
  - ∗ Fast on typical or non-degenerate inputs
  - ∗ THIS TALK: subdivision method
- (B) PURELY NUMERICAL METHODS
  - ∗ No manipulation of algebraic numbers, resultants, root isolation
  - ∗ ADVANTAGES: simpler, applies to non-algebraic geometry
  - ∗ THIS TALK: purely numerical subdivision
- (C) COMPLETE METHODS
  - ∗ AVOID Assumptions on Inputs
  - ∗ E.g., non-singularity, Morseness, general position
  - ∗ THIS TALK: isolated singularities
- (D) COMPLEXITY ANALYSIS
  - ∗ Need intrinsic measures: E.g., condition numbers, precision-sensitivity
  - ∗ THIS TALK: integral analysis

Direct Precursors of this Work

- Complete Bezier Curve Intersection [SoCG'06]
  - How can we detect tangential intersection?
  - Adaptive Application of Geometric Separation Bounds
- Near-Optimal Analysis of Descartes' Method [ISSAC'06]
  - Amortized Analysis – what does it mean?
  - Use of Mahler-Davenport Bound
- Isotopic Approximation of Curves and Surfaces [Vegter-Plantinga'04-07]
  - Introduces a computational model

Direct Precursors of this Work

- Complete Bezier Curve Intersection [SoCG'06]
    - ∗ How can we detect tangential intersection?
    - ∗ Adaptive Application of Geometric Separation Bounds
- Near-Optimal Analysis of Descartes' Method [ISSAC'06]
    - ∗ Amortized Analysis – what does it mean?
    - ∗ Use of Mahler-Davenport Bound
- Isotopic Approximation of Curves and Surfaces [Vegter-Plantinga'04-07]
    - ∗ Introduces a computational model

Direct Precursors of this Work

- Complete Bezier Curve Intersection [SoCG'06]
  - ∗ How can we detect tangential intersection?
  - ∗ Adaptive Application of Geometric Separation Bounds

- Near-Optimal Analysis of Descartes' Method [ISSAC'06]
  - ∗ Amortized Analysis – what does it mean?
  - ∗ Use of Mahler-Davenport Bound

- Isotopic Approximation of Curves and Surfaces [Vegter-Plantinga'04-07]
  - ∗ Introduces a computational model

Direct Precursors of this Work

- Complete Bezier Curve Intersection [SoCG'06]
  - ∗ How can we detect tangential intersection?
  - ∗ Adaptive Application of Geometric Separation Bounds
- Near-Optimal Analysis of Descartes' Method [ISSAC'06]
  - ∗ Amortized Analysis – what does it mean?
  - ∗ Use of Mahler-Davenport Bound
- Isotopic Approximation of Curves and Surfaces [Vegter-Plantinga'04-07]
  - ∗ Introduces a computational model

OVERVIEW OF TALK

- PART II: Problem and Review
  - ∗ The Vegter-Plantinga computational model
- PART III: One-Dimensional Case
  - ∗ I.e., root isolation in presence of multiple roots
- PART IV: Integral Complexity Analysis
  - ∗ Continuous amortization arguments
  - ∗ Integral Analysis for non-singular 1-D
- PART V: Complete subdivision algorithm
  - ∗ Extension of Vegter-Plantinga to singular curves
- CONCLUSIONS

OVERVIEW OF TALK

- PART II: Problem and Review
  - ∗ The Vegter-Plantinga computational model
- PART III: One-Dimensional Case
  - ∗ I.e., root isolation in presence of multiple roots
- PART IV: Integral Complexity Analysis
  - ∗ Continuous amortization arguments
  - ∗ Integral Analysis for non-singular 1-D
- PART V: Complete subdivision algorithm
  - ∗ Extension of Vegter-Plantinga to singular curves
- CONCLUSIONS

OVERVIEW OF TALK

- PART II: Problem and Review
  - ∗ The Vegter-Plantinga computational model
- PART III: One-Dimensional Case
  - ∗ I.e., root isolation in presence of multiple roots
- PART IV: Integral Complexity Analysis
  - ∗ Continuous amortization arguments
  - ∗ Integral Analysis for non-singular 1-D
- PART V: Complete subdivision algorithm
  - ∗ Extension of Vegter-Plantinga to singular curves
- CONCLUSIONS

OVERVIEW OF TALK

- PART II: Problem and Review
  - ∗ The Vegter-Plantinga computational model
- PART III: One-Dimensional Case
  - ∗ I.e., root isolation in presence of multiple roots
- PART IV: Integral Complexity Analysis
  - ∗ Continuous amortization arguments
  - ∗ Integral Analysis for non-singular 1-D
- PART V: Complete subdivision algorithm
  - ∗ Extension of Vegter-Plantinga to singular curves
- CONCLUSIONS

OVERVIEW OF TALK

- PART II: Problem and Review
  - ∗ The Vegter-Plantinga computational model
- PART III: One-Dimensional Case
  - ∗ I.e., root isolation in presence of multiple roots
- PART IV: Integral Complexity Analysis
  - ∗ Continuous amortization arguments
  - ∗ Integral Analysis for non-singular 1-D
- PART V: Complete subdivision algorithm
  - ∗ Extension of Vegter-Plantinga to singular curves
- CONCLUSIONS

OVERVIEW OF TALK

- PART II: Problem and Review
  - ∗ The Vegter-Plantinga computational model
- PART III: One-Dimensional Case
  - ∗ I.e., root isolation in presence of multiple roots
- PART IV: Integral Complexity Analysis
  - ∗ Continuous amortization arguments
  - ∗ Integral Analysis for non-singular 1-D
- PART V: Complete subdivision algorithm
  - ∗ Extension of Vegter-Plantinga to singular curves
- CONCLUSIONS

PART II.
PROBLEM AND REVIEW

Problem Statement, Computational Model

- Meshing of Implicit Curve
  - ∗ GIVEN: $f : \mathbb{R}^n \longrightarrow \mathbb{R}$ and $\varepsilon > 0$
  - ∗ COMPUTE: a polygonal approximation $\widetilde{S}$ for the surface $S : f = 0$.
  - ∗ WHERE: $\widetilde{S}$ is isotopic to S, and $d(\widetilde{S}, S) \leq \varepsilon$.

- The Vegter-Plantinga Computational Model
  - ∗ (1) [Like Marching Cube] Evaluate sign of $f(x)$ for $x \in \mathbb{F}^n \subseteq \mathbb{R}^n$
  - ∗ (2) [Interval Arithmetic] Evaluate interval versions of f and its derivatives

- NOTE: Model is applicable to non-algebraic f
  - ∗ No root isolation, polynomial manipulation, or resultants

Problem Statement, Computational Model

- Meshing of Implicit Curve
  - ∗ GIVEN: $f : \mathbb{R}^n \longrightarrow \mathbb{R}$ and $\varepsilon > 0$
  - ∗ COMPUTE: a polygonal approximation $\widetilde{S}$ for the surface $S : f = 0$.
  - ∗ WHERE: $\widetilde{S}$ is isotopic to S, and $d(\widetilde{S}, S) \leq \varepsilon$.

- The Vegter-Plantinga Computational Model
  - ∗ (1) [Like Marching Cube] Evaluate sign of $f(x)$ for $x \in \mathbb{F}^n \subseteq \mathbb{R}^n$
  - ∗ (2) [Interval Arithmetic] Evaluate interval versions of f and its derivatives

- NOTE: Model is applicable to non-algebraic f
  - ∗ No root isolation, polynomial manipulation, or resultants

Problem Statement, Computational Model

- Meshing of Implicit Curve
    - ∗ GIVEN: $f : \mathbb{R}^n \longrightarrow \mathbb{R}$ and $\varepsilon > 0$
    - ∗ COMPUTE: a polygonal approximation $\widetilde{S}$ for the surface $S : f = 0$.
    - ∗ WHERE: $\widetilde{S}$ is isotopic to S, and $d(\widetilde{S}, S) \leq \varepsilon$.

- The Vegter-Plantinga Computational Model
    - ∗ (1) [Like Marching Cube] Evaluate sign of $f(x)$ for $x \in \mathbb{F}^n \subseteq \mathbb{R}^n$
    - ∗ (2) [Interval Arithmetic] Evaluate interval versions of f and its derivatives

- NOTE: Model is applicable to non-algebraic f
    - ∗ No root isolation, polynomial manipulation, or resultants

Problem Statement, Computational Model

- Meshing of Implicit Curve
  - ∗ GIVEN: $f : \mathbb{R}^n \longrightarrow \mathbb{R}$ and $\varepsilon > 0$
  - ∗ COMPUTE: a polygonal approximation $\widetilde{S}$ for the surface $S : f = 0$.
  - ∗ WHERE: $\widetilde{S}$ is isotopic to S, and $d(\widetilde{S}, S) \leq \varepsilon$.

- The Vegter-Plantinga Computational Model
  - ∗ (1) [Like Marching Cube] Evaluate sign of $f(x)$ for $x \in \mathbb{F}^n \subseteq \mathbb{R}^n$
  - ∗ (2) [Interval Arithmetic] Evaluate interval versions of f and its derivatives

- NOTE: Model is applicable to non-algebraic f
  - ∗ No root isolation, polynomial manipulation, or resultants

Interval Arithmetic

- Let $[a, b] = I \subseteq \mathbb{R}$ be a (closed) interval
  - $*$ Width: $w(I) = b - a$.
  - $*$ Midpoint: $m(I) = (a + b)/2$.

- For $S \subseteq \mathbb{R}$, let $\square S$ denote the set of all intervals contained in S
  - $*$ n-Boxes: $B \in \square\mathbb{R}^n$
  - $*$ For $f : \mathbb{R}^n \longrightarrow \mathbb{R}$, its interval version is denoted

$\square f : \square\mathbb{R}^n \longrightarrow \square\mathbb{R}$

Interval Arithmetic

- Let $[a, b] = I \subseteq \mathbb{R}$ be a (closed) interval
  - $*$ Width: $w(I) = b - a$.
  - $*$ Midpoint: $m(I) = (a + b)/2$.

- For $S \subseteq \mathbb{R}$, let $\square S$ denote the set of all intervals contained in S
  - $*$ n-Boxes: $B \in \square \mathbb{R}^n$
  - $*$ For $f : \mathbb{R}^n \longrightarrow \mathbb{R}$, its interval version is denoted

$\square f : \square \mathbb{R}^n \longrightarrow \square \mathbb{R}$

Interval Arithmetic

- Let $[a, b] = I \subseteq \mathbb{R}$ be a (closed) interval
  - $*$ Width: $w(I) = b - a$.
  - $*$ Midpoint: $m(I) = (a + b)/2$.

- For $S \subseteq \mathbb{R}$, let $\square S$ denote the set of all intervals contained in S
  - $*$ n-Boxes: $B \in \square\mathbb{R}^n$
  - $*$ For $f : \mathbb{R}^n \longrightarrow \mathbb{R}$, its interval version is denoted

$$\square f : \square\mathbb{R}^n \longrightarrow \square\mathbb{R}$$

Review of Vegter-Plantinga Algorithm

- Assume $S : f = 0$ is a non-singular curve
  - ∗ Begin with any square box B_0
  - ∗ We want to approximate $S \cap B_0$

- TWO TESTS
  - ∗ $C_0(B)$: $\qquad 0 \notin \square f(B)$
  - ∗ $C_1(B)$: $\qquad 0 \notin (\square f_x(B))^2 + (\square f_y(B)^2)$

- STEP 1: MAIN LOOP

```
        Initialize $Q$ with $\{B_0\}$
        while $Q$ is non-empty
              Remove $B$ from $Q$
1.            if $C_0(B)$, discard $B$
2.            elif $C_1(B)$, put $B$ into $Q^*$
3.            else split $B$ into 4 squares and place in $Q$
```

Review of Vegter-Plantinga Algorithm

- Assume $S : f = 0$ is a non-singular curve
  - ∗ Begin with any square box B_0
  - ∗ We want to approximate $S \cap B_0$

- TWO TESTS
  - ∗ $C_0(B) : \quad 0 \notin \square f(B)$
  - ∗ $C_1(B) : \quad 0 \notin (\square f_x(B))^2 + (\square f_y(B)^2)$

- STEP 1: MAIN LOOP

```
        Initialize $Q$ with $\{B_0\}$
        while $Q$ is non-empty
              Remove $B$ from $Q$
1.            if $C_0(B)$, discard $B$
2.            elif $C_1(B)$, put $B$ into $Q^*$
3.            else split $B$ into 4 squares and place in $Q$
```

Review of Vegter-Plantinga Algorithm

- Assume $S : f = 0$ is a non-singular curve
  - ∗ Begin with any square box B_0
  - ∗ We want to approximate $S \cap B_0$

- TWO TESTS
  - ∗ $C_0(B)$: $\quad 0 \notin \Box f(B)$
  - ∗ $C_1(B)$: $\quad 0 \notin (\Box f_x(B))^2 + (\Box f_y(B)^2)$

- STEP 1: MAIN LOOP

```
        Initialize $Q$ with $\{B_0\}$
        while $Q$ is non-empty
            Remove $B$ from $Q$
1.          if $C_0(B)$, discard $B$
2.          elif $C_1(B)$, put $B$ into $Q^*$
3.          else split $B$ into 4 squares and place in $Q$
```

Review of Vegter-Plantinga Algorithm

- Assume $S : f = 0$ is a non-singular curve
  - $*$ Begin with any square box B_0
  - $*$ We want to approximate $S \cap B_0$

- TWO TESTS
  - $*$ $C_0(B)$: $\quad 0 \notin \square f(B)$
  - $*$ $C_1(B)$: $\quad 0 \notin (\square f_x(B))^2 + (\square f_y(B)^2)$

- STEP 1: MAIN LOOP

```
      Initialize $Q$ with $\{B_0\}$
      while $Q$ is non-empty
          Remove $B$ from $Q$
1.        if $C_0(B)$, discard $B$
2.        elif $C_1(B)$, put $B$ into $Q^*$
3.        else split $B$ into 4 squares and place in $Q$
```

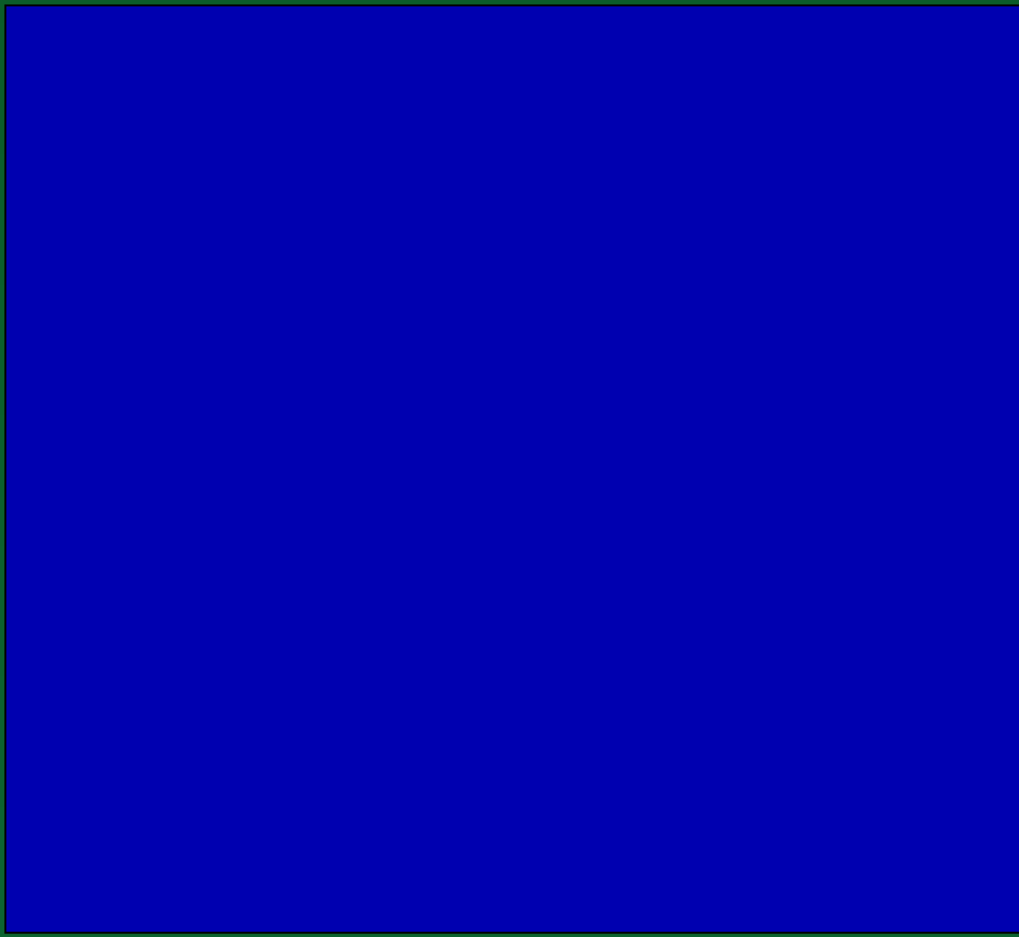
- STEP 2: Make Q^* balanced
  - I.e., adjacent boxes in Q^* differ by at most one level

- STEP 3: Form graph G from Q^*, and output G

  - Insert Vertices and Connect to get G

- STEP 2: Make Q^* balanced
  - $*$ I.e., adjacent boxes in Q^* differ by at most one level

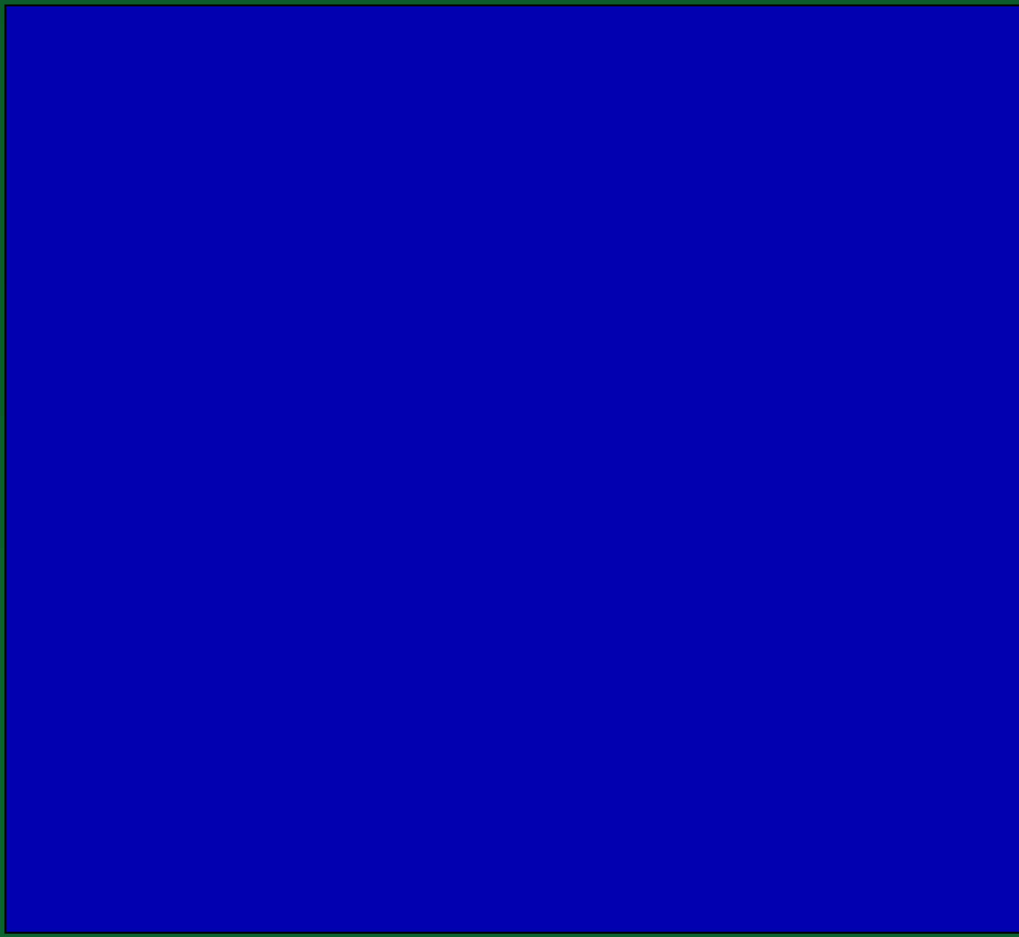
- STEP 3: Form graph G from Q^*, and output G



  - $*$ Insert Vertices and Connect to get G

- STEP 2: Make Q^* balanced
  - $*$ I.e., adjacent boxes in Q^* differ by at most one level

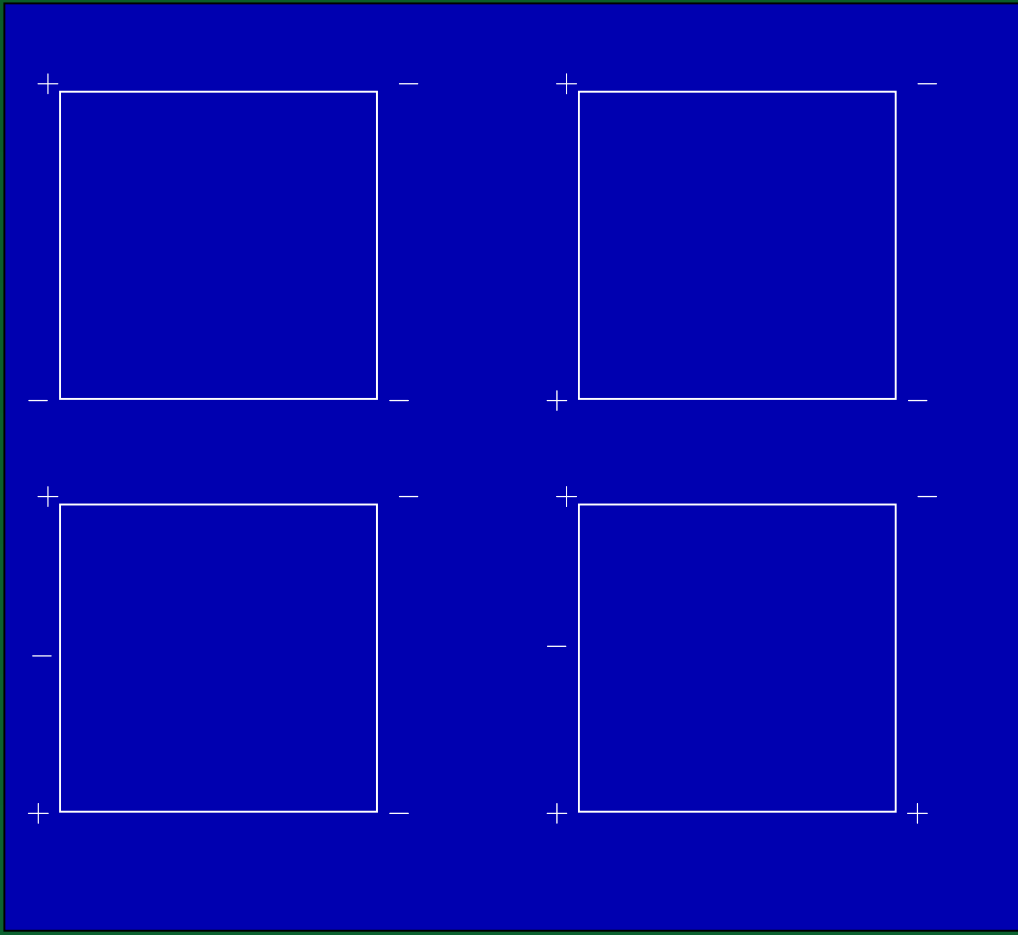
- STEP 3: Form graph G from Q^*, and output G



  - $*$ Insert Vertices and Connect to get G

- STEP 2: Make Q^* balanced
  - $*$ I.e., adjacent boxes in Q^* differ by at most one level

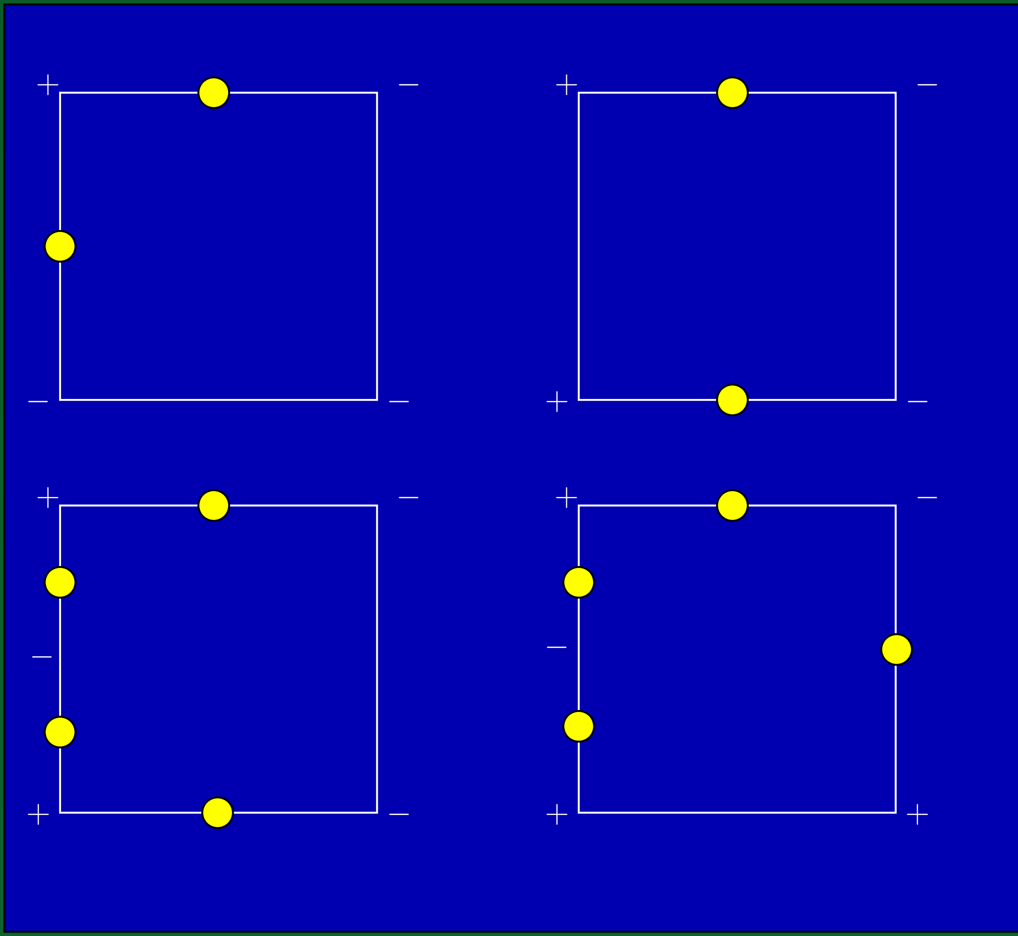
- STEP 3: Form graph G from Q^*, and output G



  - $*$ Insert Vertices and Connect to get G

- STEP 2: Make Q^* balanced
  - $*$ I.e., adjacent boxes in Q^* differ by at most one level

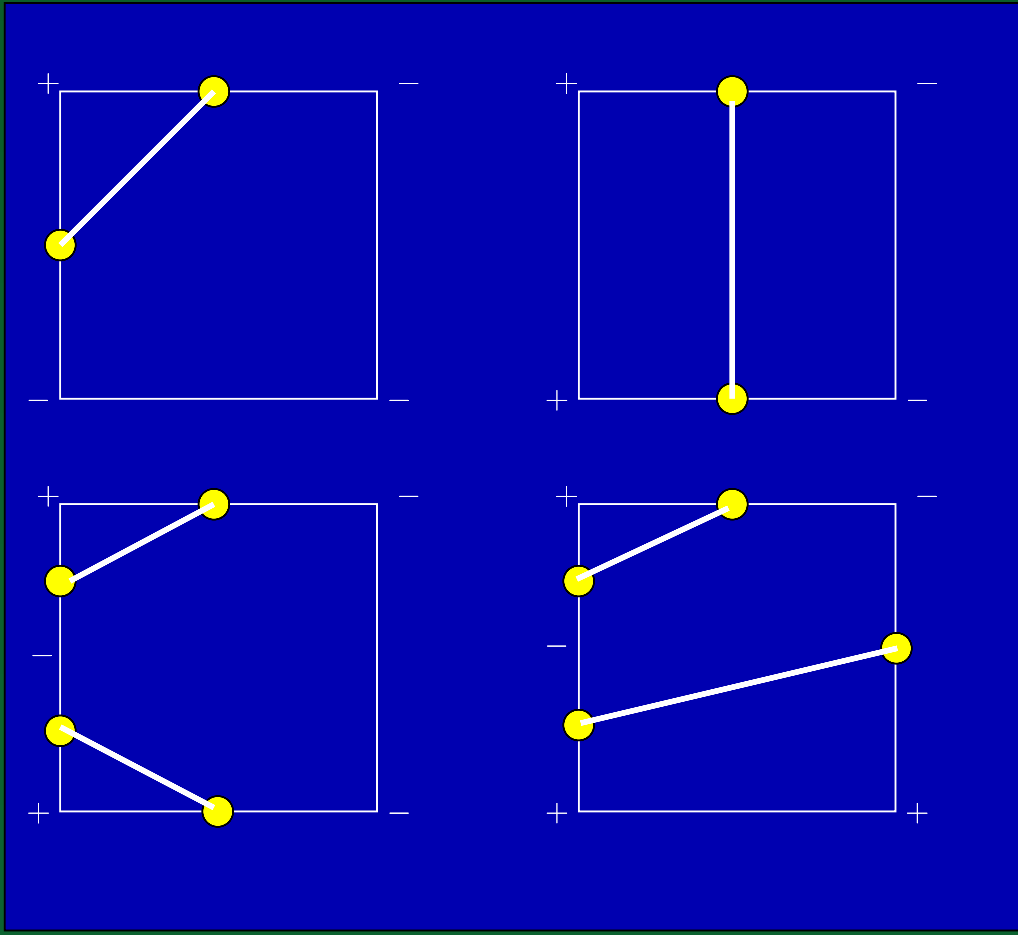
- STEP 3: Form graph G from Q^*, and output G



  - $*$ Insert Vertices and Connect to get G

- STEP 2: Make Q^* balanced
  - $*$ I.e., adjacent boxes in Q^* differ by at most one level

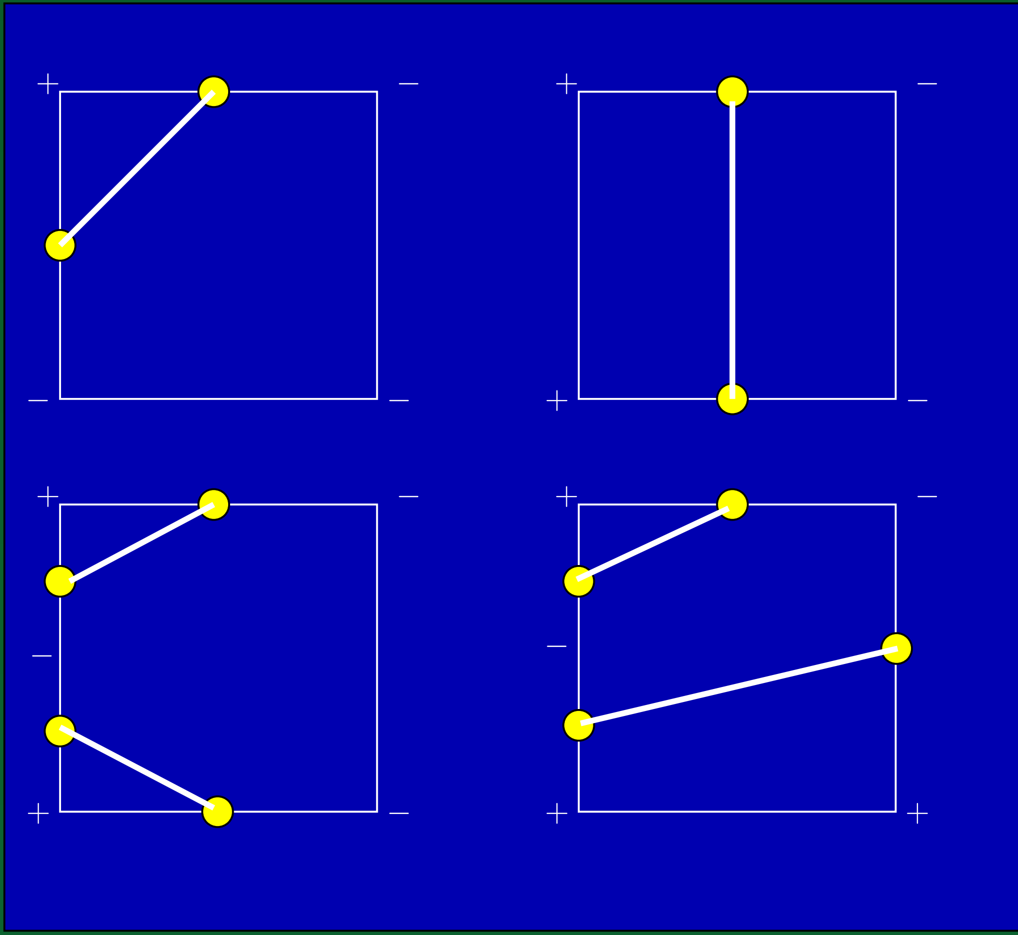
- STEP 3: Form graph G from Q^*, and output G



$*$ Insert Vertices and Connect to get G

- STEP 2: Make Q^* balanced
  - $*$ I.e., adjacent boxes in Q^* differ by at most one level

- STEP 3: Form graph G from Q^*, and output G



$*$ Insert Vertices and Connect to get G

- THEOREM (Vegter-Plantinga)
  - Assume $S \subseteq B_0$
  - Then the output graph G is isotopic to $B_0 \cap S$

- "Weak Subdivision": the topology on the boundary of boxes are not guaranteed!

- THEOREM (Vegter-Plantinga)
  - Assume $S \subseteq B_0$
  - Then the output graph G is isotopic to $B_0 \cap S$

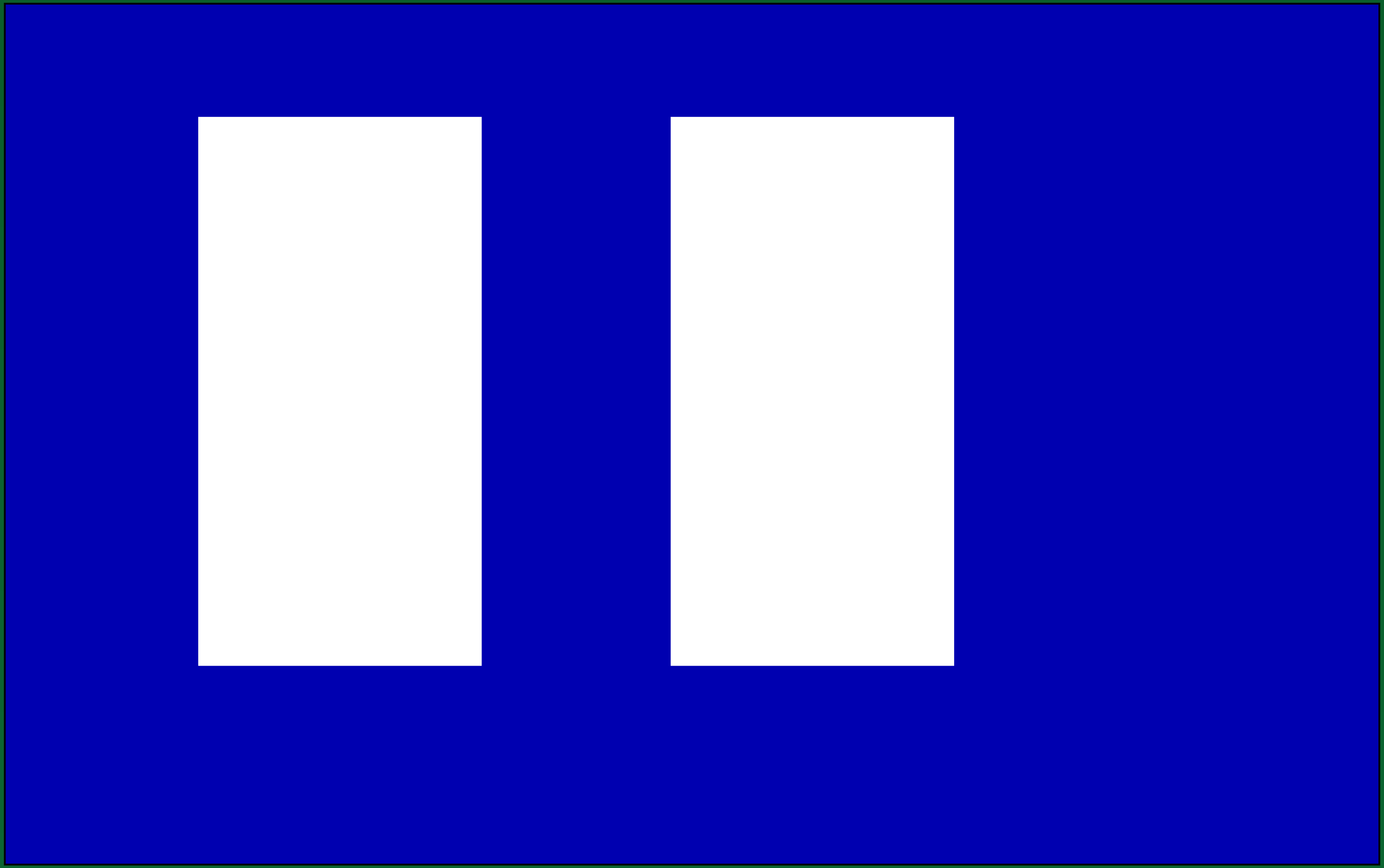
- “Weak Subdivision”: the topology on the boundary of boxes are not guaranteed!

- THEOREM (Vegter-Plantinga)
  - ∗ Assume $S \subseteq B_0$
  - ∗ Then the output graph G is isotopic to $B_0 \cap S$

- "Weak Subdivision": the topology on the boundary of boxes are not guaranteed!

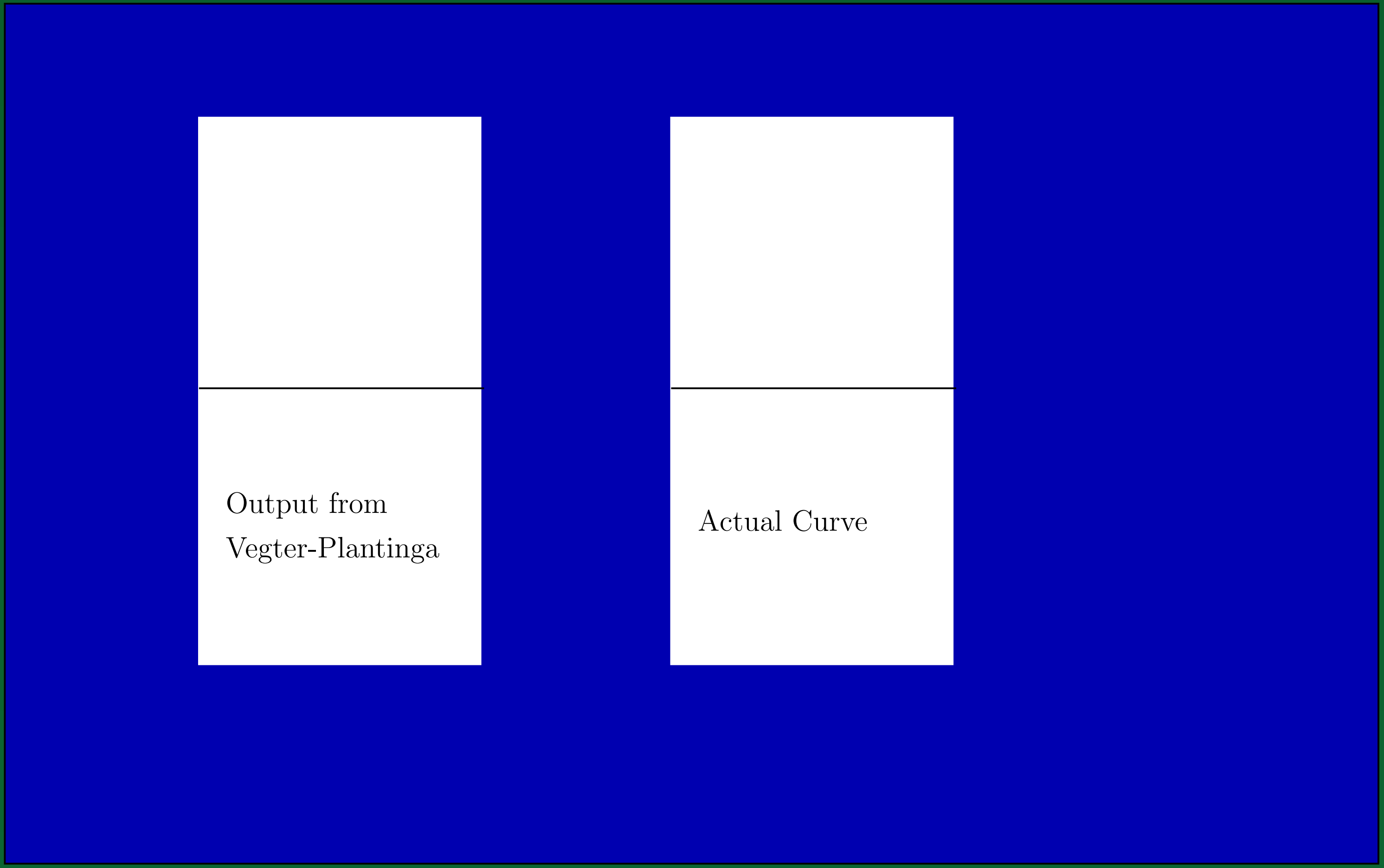
- THEOREM (Vegter-Plantinga)
  - $*$ Assume $S \subseteq B_0$
  - $*$ Then the output graph G is isotopic to $B_0 \cap S$

- "Weak Subdivision": the topology on the boundary of boxes are not guaranteed!



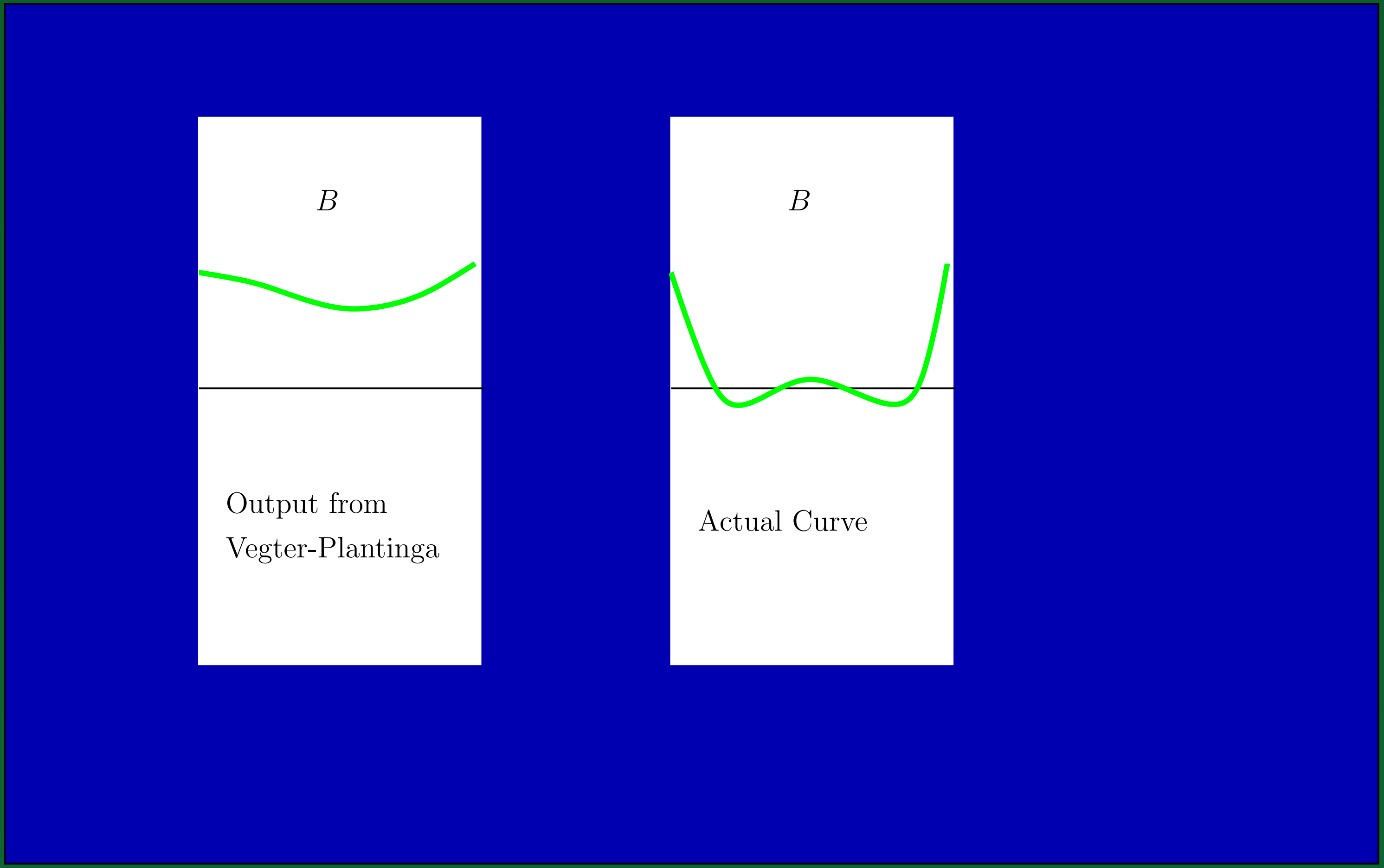
- THEOREM (Vegter-Plantinga)
  - $*$ Assume $S \subseteq B_0$
  - $*$ Then the output graph G is isotopic to $B_0 \cap S$

- "Weak Subdivision": the topology on the boundary of boxes are not guaranteed!



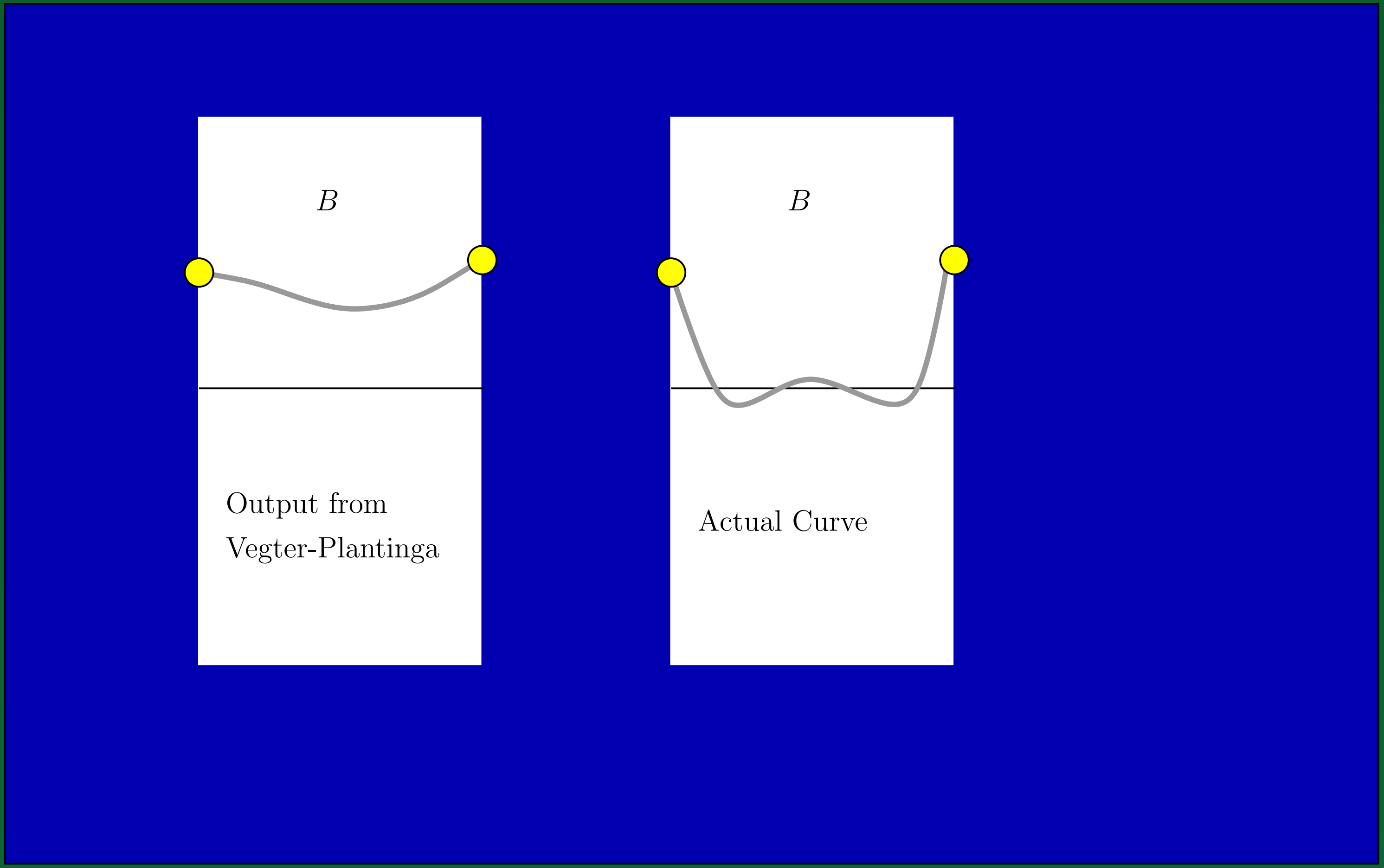
- THEOREM (Vegter-Plantinga)
  - $*$ Assume $S \subseteq B_0$
  - $*$ Then the output graph G is isotopic to $B_0 \cap S$

- "Weak Subdivision": the topology on the boundary of boxes are not guaranteed!



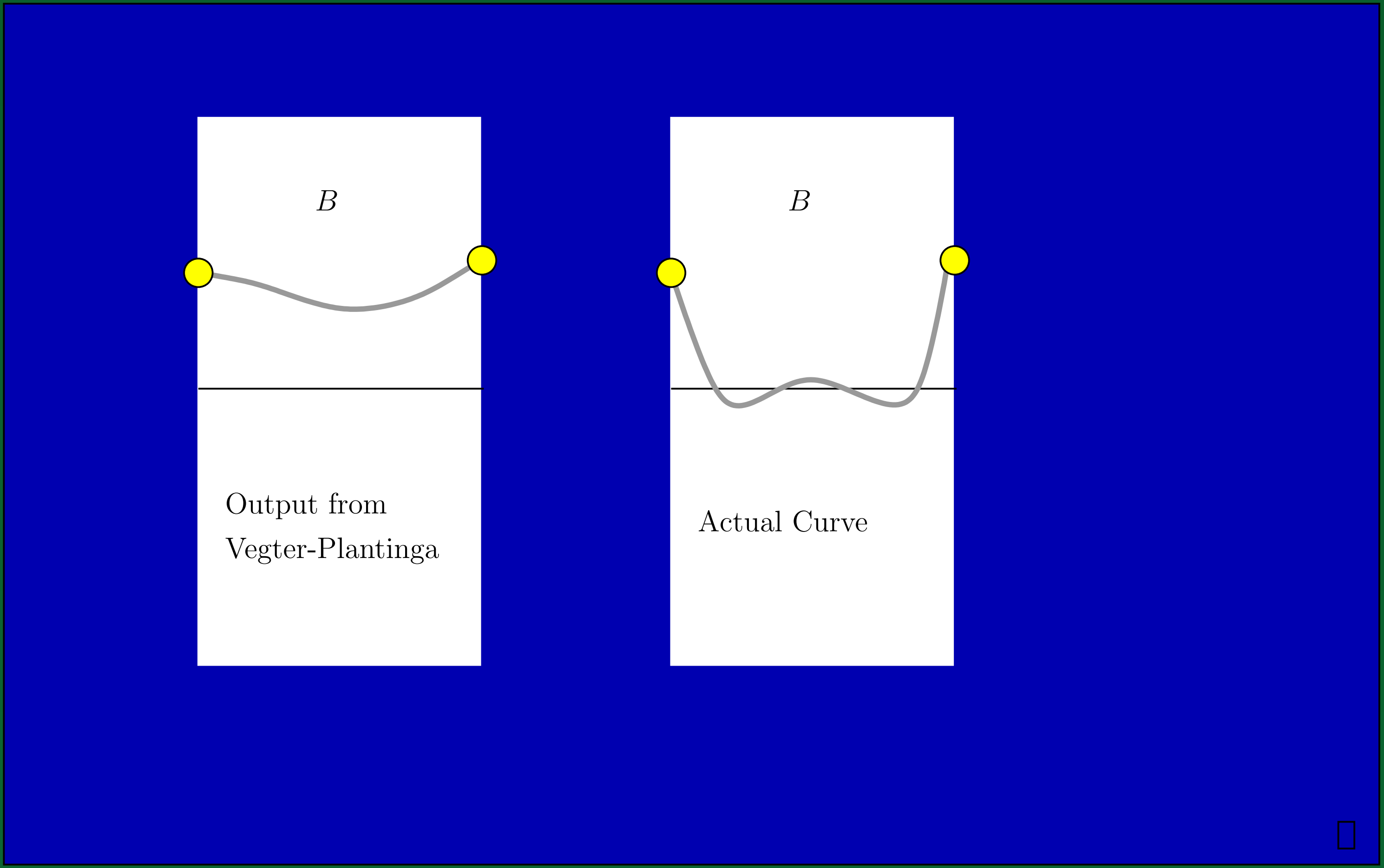
- THEOREM (Vegter-Plantinga)
  - $*$ Assume $S \subseteq B_0$
  - $*$ Then the output graph G is isotopic to $B_0 \cap S$

- "Weak Subdivision": the topology on the boundary of boxes are not guaranteed!



- THEOREM (Vegter-Plantinga)
  - $*$ Assume $S \subseteq B_0$
  - $*$ Then the output graph G is isotopic to $B_0 \cap S$

- "Weak Subdivision": the topology on the boundary of boxes are not guaranteed!



- We will need EXTENSION 1: If S is not contained in B_0
  - ∗ Can modify so that G is ε-isotopic to $B_0^+ \cap S$

- We will need EXTENSION 2: If S may be singular outside B_0
  - ∗ Can modify so that G is ε-isotopic to $B_0^+ \cap S$
  - ∗ FURTHERMORE B_0 need not be simply connected
  - ∗ E.g., B is an annular region

- We will need EXTENSION 1: If S is not contained in B_0
  - ∗ Can modify so that G is ε-isotopic to $B_0^+ \cap S$

- We will need EXTENSION 2: If S may be singular outside B_0
  - ∗ Can modify so that G is ε-isotopic to $B_0^+ \cap S$
  - ∗ FURTHERMORE B_0 need not be simply connected
  - ∗ E.g., B is an annular region

- We will need EXTENSION 1: If S is not contained in B_0
  - $*$ Can modify so that G is ε-isotopic to $B_0^+ \cap S$

- We will need EXTENSION 2: If S may be singular outside B_0
  - $*$ Can modify so that G is ε-isotopic to $B_0^+ \cap S$
  - $*$ FURTHERMORE B_0 need not be simply connected
  - $*$ E.g., B is an annular region

The Main Challenge: Singularities

- Completeness Issue:
  - ∗ How to confirm singularities in the in the Vegter-Plantinga Model?
- Meshing curves is a 2-D problem
  - ∗ The 1-D analogue is Root Isolation!
- STRATEGY: First understand the 1-D case

The Main Challenge: Singularities

- Completeness Issue:
  - ∗ How to confirm singularities in the in the Vegter-Plantinga Model?

- Meshing curves is a 2-D problem
  - ∗ The 1-D analogue is Root Isolation!

- STRATEGY: First understand the 1-D case

The Main Challenge: Singularities

- Completeness Issue:
  - ∗ How to confirm singularities in the in the Vegter-Plantinga Model?

- Meshing curves is a 2-D problem
  - ∗ The 1-D analogue is Root Isolation!

- STRATEGY: First understand the 1-D case

The Main Challenge: Singularities

- Completeness Issue:
  - ∗ How to confirm singularities in the in the Vegter-Plantinga Model?
- Meshing curves is a 2-D problem
  - ∗ The 1-D analogue is Root Isolation!
- STRATEGY: First understand the 1-D case

PART III.
THE SINGULAR CASE IN 1-D

The Algorithm EVAL

- EVAL: an adaptation of Vegter-Plantinga to 1-D
  - ∗ STEP 1: MAIN LOOP

```
        Initialize $Q$ with $\{B_0\}$
        while $Q$ is non-empty
              Remove $B$ from $Q$
1.            if $C_0(B)$, discard $B$
2.            elif $C_1(B)$, put $B$ [if necessary] into $Q^*$
3.            else split $B$ into 4 [2 not 4] squares and place in $Q$
```

- STEP 2: Process Q^*
  - ∗ NOTE: $C_1(B)$ says "$0 \notin \square f'(B)$"
  - ∗ Output $B = [a, b]$ iff $f(a)f(b) \leq 0$.

- Bolzano Theorem: if $f(a)f(b) < 0$, then $f(c) = 0$ for some $a < c < b$
  - ∗ Other Evaluation Methods: based on Newton operator

The Algorithm EVAL

- EVAL: an adaptation of Vegter-Plantinga to 1-D
  - $*$ STEP 1: MAIN LOOP

```
       Initialize $Q$ with $\{B_0\}$
       while $Q$ is non-empty
            Remove $B$ from $Q$
1.          if $C_0(B)$, discard $B$
2.          elif $C_1(B)$, put $B$ [if necessary] into $Q^*$
3.          else split $B$ into 4 [2 not 4] squares and place in $Q$
```

- STEP 2: Process Q^*
  - $*$ NOTE: $C_1(B)$ says "$0 \notin \square f'(B)$"
  - $*$ Output $B = [a, b]$ iff $f(a)f(b) \leq 0$.

- Bolzano Theorem: if $f(a)f(b) < 0$, then $f(c) = 0$ for some $a < c < b$
  - $*$ Other Evaluation Methods: based on Newton operator

The Algorithm EVAL

- EVAL: an adaptation of Vegter-Plantinga to 1-D
  - STEP 1: MAIN LOOP

```
      Initialize $Q$ with $\{B_0\}$
      while $Q$ is non-empty
            Remove $B$ from $Q$
1.          if $C_0(B)$, discard $B$
2.          elif $C_1(B)$, put $B$ [if necessary] into $Q^*$
3.          else split $B$ into 4 [2 not 4] squares and place in $Q$
```

- STEP 2: Process Q^*
  - NOTE: $C_1(B)$ says "$0 \notin \square f'(B)$"
  - Output $B = [a, b]$ iff $f(a)f(b) \leq 0$.

- Bolzano Theorem: if $f(a)f(b) < 0$, then $f(c) = 0$ for some $a < c < b$
  - Other Evaluation Methods: based on Newton operator

The Algorithm EVAL

- EVAL: an adaptation of Vegter-Plantinga to 1-D
  - ∗ STEP 1: MAIN LOOP

```
        Initialize $Q$ with $\{B_0\}$
        while $Q$ is non-empty
            Remove $B$ from $Q$
1.          if $C_0(B)$, discard $B$
2.          elif $C_1(B)$, put $B$ [if necessary] into $Q^*$
3.          else split $B$ into 4 [2 not 4] squares and place in $Q$
```

- STEP 2: Process Q^*
  - ∗ NOTE: $C_1(B)$ says "$0 \notin \square f'(B)$"
  - ∗ Output $B = [a, b]$ iff $f(a)f(b) \leq 0$.

- Bolzano Theorem: if $f(a)f(b) < 0$, then $f(c) = 0$ for some $a < c < b$
  - ∗ Other Evaluation Methods: based on Newton operator

The Algorithm EVAL

- EVAL: an adaptation of Vegter-Plantinga to 1-D
  - ∗ STEP 1: MAIN LOOP

```
   Initialize $Q$ with $\{B_0\}$
   while $Q$ is non-empty
       Remove $B$ from $Q$
1.     if $C_0(B)$, discard $B$
2.     elif $C_1(B)$, put $B$ [if necessary] into $Q^*$
3.     else split $B$ into 4 [2 not 4] squares and place in $Q$
```

- STEP 2: Process Q^*
  - ∗ NOTE: $C_1(B)$ says "$0 \notin \square f'(B)$"
  - ∗ Output $B = [a, b]$ iff $f(a)f(b) \leq 0$.

- Bolzano Theorem: if $f(a)f(b) < 0$, then $f(c) = 0$ for some $a < c < b$
  - ∗ Other Evaluation Methods: based on Newton operator

How to do the Singular Case

- We need various bounds:
  - K_I: Lipschitz constant for f over I
  - $\Delta(f,g)$: Separation bound for zeros of f and g
  - $\mathrm{EV}(f,g)$: Evaluation Bound, $= \min\{|f(x)| : g(x)=0, f(x)\neq 0\}$

- ALSO: let $\Delta(f) \equiv \Delta(f,f')$, $\mathrm{EV}(f) \equiv \mathrm{EV}(f,f')$

- NonAdaptic MultipleRoot Isolation
  - Subdivide as long as interval is larger than

$$BOUND := \min\left\{\Delta(f),\ \Delta(f'),\ \Delta(f,f'),\ \frac{\mathrm{EV}(f)}{3K},\ \frac{\mathrm{EV}(f',f)}{2K'}\right\}$$

How to do the Singular Case

- We need various bounds:
  - K_I: Lipschitz constant for f over I
  - $\Delta(f, g)$: Separation bound for zeros of f and g
  - $\mathrm{EV}(f, g)$: Evaluation Bound, $= \min\{|f(x)| : g(x) = 0, f(x) \neq 0\}$
- ALSO: let $\Delta(f) \equiv \Delta(f, f')$, $\mathrm{EV}(f) \equiv \mathrm{EV}(f, f')$
- NonAdaptic MultipleRoot Isolation
  - Subdivide as long as interval is larger than

$$BOUND := \min\left\{\Delta(f),\ \Delta(f'),\ \Delta(f, f'),\ \frac{\mathrm{EV}(f)}{3K},\ \frac{\mathrm{EV}(f', f)}{2K'}\right\}$$

How to do the Singular Case

- We need various bounds:
  - K_I: Lipschitz constant for f over I
  - $\Delta(f, g)$: Separation bound for zeros of f and g
  - $\mathrm{EV}(f, g)$: Evaluation Bound, $= \min\{|f(x)| : g(x) = 0, f(x) \neq 0\}$

- ALSO: let $\Delta(f) \equiv \Delta(f, f')$, $\mathrm{EV}(f) \equiv \mathrm{EV}(f, f')$

- NonAdaptic MultipleRoot Isolation
  - Subdivide as long as interval is larger than

$$BOUND := \min \left\{ \Delta(f),\ \Delta(f'),\ \Delta(f, f'),\ \frac{\mathrm{EV}(f)}{3K},\ \frac{\mathrm{EV}(f', f)}{2K'} \right\}$$

How to do the Singular Case

- We need various bounds:
  - ∗ K_I: Lipschitz constant for f over I
  - ∗ $\Delta(f,g)$: Separation bound for zeros of f and g
  - ∗ $\mathrm{EV}(f,g)$: Evaluation Bound, $= \min\{|f(x)| : g(x)=0, f(x)\neq 0\}$

- ALSO: let $\Delta(f) \equiv \Delta(f,f')$, $\mathrm{EV}(f) \equiv \mathrm{EV}(f,f')$

- NonAdaptic MultipleRoot Isolation
  - ∗ Subdivide as long as interval is larger than

$$BOUND := \min\left\{\Delta(f),\ \Delta(f'),\ \Delta(f,f'),\ \frac{\mathrm{EV}(f)}{3K},\ \frac{\mathrm{EV}(f',f)}{2K'}\right\}$$

```
ROOTISOL Input: $F : \mathbb{R} \longrightarrow \mathbb{R}$, and interval $I_0$
Output: A list $F$ of isolating intervals
      Initialize $Q$ to $I_0$.
      while $Q \neq \emptyset$
            $I = [a, b] \leftarrow Q.remove()$.
            if $|I| > BOUND$
                  if the midpoint of $I$, $m = (a + b)$, is a root of $F$
                        if midpoint of $I$ is also a root of $F'$, put $[m, m]$ into $L$ (singular).
                        else put $[m, m]$ into $L$ (nonsingular)
                  Split $I$ in two equal halves, and put them in $Q$.
            else
                  if $(F'(a)F'(b) \leq 0)$
1.                      if $|f(a)| \leq \mathrm{EV}(F, F')/3$, put $I$ into $L$ (singular).
2.                      else Discard $I$.
                  else % Thus, $F'(a)F'(b) > 0$
                        if $(f(a)f(b) \leq 0)$
3.                            if $(|F'(a)| \leq \mathrm{EV}(F', F)/3)$, put $I$ into $L$ (singular)
4.                            else put $I$ into $L$ (nonsingular).
5.                      else Discard $I$.
```

- Adaptive version
  - ∗ Too long to fit the slide — see the end

- Adaptive version
    - ∗ Too long to fit the slide — see the end

Generalizing the 1-D Solution?

- What we learned from the 1-D case:
  - ∗ Need Separation Bounds: $\Delta(f), \Delta(f'), \Delta(f, f')$
  - ∗ Need Evaluation Bounds: $\mathrm{EV}(f)$, $\mathrm{EV}(f', f)$
  - ∗ Need Lipschitz Constants: $\mathrm{EV}(f)$, $\mathrm{EV}(f', f)$
  - ∗ KEY POINT: these bounds ONLY serves as stopping criteria, preserving adaptivity

- BUT direct generalization of these ideas don't work

Generalizing the 1-D Solution?

- What we learned from the 1-D case:
  - ∗ Need Separation Bounds: $\Delta(f), \Delta(f'), \Delta(f, f')$
  - ∗ Need Evaluation Bounds: $\mathrm{EV}(f)$, $\mathrm{EV}(f', f)$
  - ∗ Need Lipschitz Constants: $\mathrm{EV}(f)$, $\mathrm{EV}(f', f)$
  - ∗ KEY POINT: these bounds ONLY serves as stopping criteria, preserving adaptivity

- BUT direct generalization of these ideas don't work

Generalizing the 1-D Solution?

- What we learned from the 1-D case:
  - ∗ Need Separation Bounds: $\Delta(f), \Delta(f'), \Delta(f, f')$
  - ∗ Need Evaluation Bounds: $\mathrm{EV}(f)$, $\mathrm{EV}(f', f)$
  - ∗ Need Lipschitz Constants: $\mathrm{EV}(f)$, $\mathrm{EV}(f', f)$
  - ∗ KEY POINT: these bounds ONLY serves as stopping criteria, preserving adaptivity
- BUT direct generalization of these ideas don't work

PART IV.
COMPLEXITY ANALYSIS OF EVAL

Perspective on EVAL

- 3 Subdivision Methods for Isolating real zeros

$$STURM > DESCARTES > BOLZANO$$

- PRIMITIVES (decreasing complexity):
  - ∗ Sturm Query: Exact number of roots in I
  - ∗ Descartes/Bernstein Query: Rule of Sign
  - ∗ Bolzano Queries: $C_0(I)$ and $C_1(I)$

- WHAT IS THE SIZE OF THE SEARCH TREE?
  - ∗ Sturm Tree: $O(dL)$ [Davenport'86]
  - ∗ Descartes Tree: $O(dL)$ [Eigenwillig-Sharma-Y'06]
  - ∗ Bolzano Tree: $O(dL)$ for optimal □ f [Burr-Sharma-Y'07]

- Empirically, Descartes is faster than Sturm
  - ∗ Attributable to the cheaper primitive!
  - ∗ Can the same hold true for Bolzano?

Perspective on EVAL

- 3 Subdivision Methods for Isolating real zeros

$$STURM > DESCARTES > BOLZANO$$

- PRIMITIVES (decreasing complexity):
  - ∗ Sturm Query: Exact number of roots in I
  - ∗ Descartes/Bernstein Query: Rule of Sign
  - ∗ Bolzano Queries: $C_0(I)$ and $C_1(I)$
- WHAT IS THE SIZE OF THE SEARCH TREE?
  - ∗ Sturm Tree: $O(dL)$ [Davenport'86]
  - ∗ Descartes Tree: $O(dL)$ [Eigenwillig-Sharma-Y'06]
  - ∗ Bolzano Tree: $O(dL)$ for optimal □ f [Burr-Sharma-Y'07]
- Empirically, Descartes is faster than Sturm
  - ∗ Attributable to the cheaper primitive!
  - ∗ Can the same hold true for Bolzano?

Perspective on EVAL

- 3 Subdivision Methods for Isolating real zeros

$$STURM > DESCARTES > BOLZANO$$

- PRIMITIVES (decreasing complexity):
  - ∗ Sturm Query: Exact number of roots in I
  - ∗ Descartes/Bernstein Query: Rule of Sign
  - ∗ Bolzano Queries: $C_0(I)$ and $C_1(I)$

- WHAT IS THE SIZE OF THE SEARCH TREE?
  - ∗ Sturm Tree: $O(dL)$ [Davenport'86]
  - ∗ Descartes Tree: $O(dL)$ [Eigenwillig-Sharma-Y'06]
  - ∗ Bolzano Tree: $O(dL)$ for optimal $\square f$ [Burr-Sharma-Y'07]

- Empirically, Descartes is faster than Sturm
  - ∗ Attributable to the cheaper primitive!
  - ∗ Can the same hold true for Bolzano?

Perspective on EVAL

- 3 Subdivision Methods for Isolating real zeros

$$STURM > DESCARTES > BOLZANO$$

- PRIMITIVES (decreasing complexity):
  - ∗ Sturm Query: Exact number of roots in I
  - ∗ Descartes/Bernstein Query: Rule of Sign
  - ∗ Bolzano Queries: $C_0(I)$ and $C_1(I)$
- WHAT IS THE SIZE OF THE SEARCH TREE?
  - ∗ Sturm Tree: $O(dL)$ [Davenport'86]
  - ∗ Descartes Tree: $O(dL)$ [Eigenwillig-Sharma-Y'06]
  - ∗ Bolzano Tree: $O(dL)$ for optimal □ f [Burr-Sharma-Y'07]
- Empirically, Descartes is faster than Sturm
  - ∗ Attributable to the cheaper primitive!
  - ∗ Can the same hold true for Bolzano?

Perspective on EVAL

- 3 Subdivision Methods for Isolating real zeros

$$STURM > DESCARTES > BOLZANO$$

- PRIMITIVES (decreasing complexity):
  - ∗ Sturm Query: Exact number of roots in I
  - ∗ Descartes/Bernstein Query: Rule of Sign
  - ∗ Bolzano Queries: $C_0(I)$ and $C_1(I)$

- WHAT IS THE SIZE OF THE SEARCH TREE?
  - ∗ Sturm Tree: $O(dL)$ [Davenport'86]
  - ∗ Descartes Tree: $O(dL)$ [Eigenwillig-Sharma-Y'06]
  - ∗ Bolzano Tree: $O(dL)$ for optimal $\Box f$ [Burr-Sharma-Y'07]

- Empirically, Descartes is faster than Sturm
  - ∗ Attributable to the cheaper primitive!
  - ∗ Can the same hold true for Bolzano?

Lipschitz Constants and Box Functions

- Analysis of EVAL for non-singular function $f : \mathbb{R} \to \mathbb{R}$

- Lipschitz Constant of f for interval X:
  - $K_X = K_X(f) := \max_{a \in X} \sum_{i=1}^{n} \frac{|f^{(i)}(a)|}{i!} w(X)^{i-1}$.
  - ALSO: let K'_X for $K_X(f')$

- Centered Form box function [Ratschek-Rokne]
  - $\square f(X) := \sum_{i=0}^{n} \frac{|f^{(i)}(m(X))|}{i!} \left(\frac{w(X)}{2}[-1,1] \right)^i$.

- PROPERTY: Let $I \subseteq X$
  - Linear Convergence: $w(\square f(I)) \le K_X \cdot w(I)$

Lipschitz Constants and Box Functions

- Analysis of EVAL for non-singular function $f : \mathbb{R} \to \mathbb{R}$

- Lipschitz Constant of f for interval X:
  - $K_X = K_X(f) := \max_{a \in X} \sum_{i=1}^{n} \frac{|f^{(i)}(a)|}{i!} w(X)^{i-1}$.
  - ALSO: let K'_X for $K_X(f')$

- Centered Form box function [Ratschek-Rokne]
  - $\square f(X) := \sum_{i=0}^{n} \frac{|f^{(i)}(m(X))|}{i!} \left(\frac{w(X)}{2}[-1, 1] \right)^{i}$.

- PROPERTY: Let $I \subseteq X$
  - Linear Convergence: $w(\square f(I)) \leq K_X \cdot w(I)$

Lipschitz Constants and Box Functions

- Analysis of EVAL for non-singular function $f : \mathbb{R} \to \mathbb{R}$

- Lipschitz Constant of f for interval X:
    - $K_X = K_X(f) := \max_{a \in X} \sum_{i=1}^{n} \frac{|f^{(i)}(a)|}{i!} w(X)^{i-1}$.
    - ALSO: let K'_X for $K_X(f')$

- Centered Form box function [Ratschek-Rokne]
    - $\Box f(X) := \sum_{i=0}^{n} \frac{|f^{(i)}(m(X))|}{i!} \left(\frac{w(X)}{2}[-1, 1] \right)^{i}$.

- PROPERTY: Let $I \subseteq X$
    - Linear Convergence: $w(\Box f(I)) \leq K_X \cdot w(I)$

Lipschitz Constants and Box Functions

- Analysis of EVAL for non-singular function $f : \mathbb{R} \to \mathbb{R}$

- Lipschitz Constant of f for interval X:
  - $K_X = K_X(f) := \max_{a \in X} \sum_{i=1}^{n} \frac{|f^{(i)}(a)|}{i!} w(X)^{i-1}$.
  - ALSO: let K'_X for $K_X(f')$

- Centered Form box function [Ratschek-Rokne]
  - $\square f(X) := \sum_{i=0}^{n} \frac{|f^{(i)}(m(X))|}{i!} \left(\frac{w(X)}{2} [-1, 1] \right)^{i}$.

- PROPERTY: Let $I \subseteq X$
  - Linear Convergence: $w(\square f(I)) \leq K_X \cdot w(I)$

Lipschitz Constants and Box Functions

- Analysis of EVAL for non-singular function $f : \mathbb{R} \rightarrow \mathbb{R}$

- Lipschitz Constant of f for interval X:
  - $K_X = K_X(f) := \max_{a \in X} \sum_{i=1}^{n} \frac{|f^{(i)}(a)|}{i!} w(X)^{i-1}$.
  - ALSO: let K'_X for $K_X(f')$

- Centered Form box function [Ratschek-Rokne]
  - $\square f(X) := \sum_{i=0}^{n} \frac{|f^{(i)}(m(X))|}{i!} \left(\frac{w(X)}{2}[-1,1] \right)^{i}$.

- PROPERTY: Let $I \subseteq X$
  - Linear Convergence: $w(\square f(I)) \leq K_X \cdot w(I)$

General Framework of Stopping Functions

- BASIC LEMMA
  - ∗ Let $a \in J$ and $0 \in \Box f(J)$
  - ∗ Then $w(J) \geq |f(a)|/K_J$

- Let $g : \mathbb{R} \longrightarrow \mathbb{R}$ be continuous.
  - ∗ An interval J is large (relative to g) if for all $a \in J$, $w(J) \geq g(a)$.
  - ∗ Call J big (relative to g) if for all $a \in J$, $w(J) \geq \frac{1}{2}g(a)$.
  - ∗ Call J terminal if either $C_0(J)$ or $C_1(J)$.
  - ∗ Call g a stopping function if an interval J that is not large must be terminal.

- A subdivision P of I is a partition
  - ∗ obtained from $P = \{I\}$ by performing repeated bisections of intervals $J \in P$
  - ∗ P is a big subdivision if each $J \in P$ is big
  - ∗ Let $\#(P)$ be the size of P.

- LEMMA
  - ∗ Let P be a big subdivision of I relative to g. Then $\#(P) \leq 2 \int_I \frac{da}{g(a)}$.

General Framework of Stopping Functions

- BASIC LEMMA
  - ∗ Let $a \in J$ and $0 \in \square f(J)$
  - ∗ Then $w(J) \geq |f(a)|/K_J$

- Let $g : \mathbb{R} \rightarrow \mathbb{R}$ be continuous.
  - ∗ An interval J is large (relative to g) if for all $a \in J$, $w(J) \geq g(a)$.
  - ∗ Call J big (relative to g) if for all $a \in J$, $w(J) \geq \frac{1}{2}g(a)$.
  - ∗ Call J terminal if either $C_0(J)$ or $C_1(J)$.
  - ∗ Call g a stopping function if an interval J that is not large must be terminal.

- A subdivision P of I is a partition
  - ∗ obtained from $P = \{I\}$ by performing repeated bisections of intervals $J \in P$
  - ∗ P is a big subdivision if each $J \in P$ is big
  - ∗ Let $\#(P)$ be the size of P.

- LEMMA
  - ∗ Let P be a big subdivision of I relative to g. Then $\#(P) \leq 2\int_I \frac{da}{g(a)}$.

General Framework of Stopping Functions

- BASIC LEMMA
  - Let $a \in J$ and $0 \in \square f(J)$
  - Then $w(J) \geq |f(a)|/K_J$

- Let $g : \mathbb{R} \longrightarrow \mathbb{R}$ be continuous.
  - An interval J is large (relative to g) if for all $a \in J$, $w(J) \geq g(a)$.
  - Call J big (relative to g) if for all $a \in J$, $w(J) \geq \frac{1}{2}g(a)$.
  - Call J terminal if either $C_0(J)$ or $C_1(J)$.
  - Call g a stopping function if an interval J that is not large must be terminal.

- A subdivision P of I is a partition
  - obtained from $P = \{I\}$ by performing repeated bisections of intervals $J \in P$
  - P is a big subdivision if each $J \in P$ is big
  - Let $\#(P)$ be the size of P.

- LEMMA
  - Let P be a big subdivision of I relative to g. Then $\#(P) \leq 2\int_I \frac{da}{g(a)}$.

General Framework of Stopping Functions

- BASIC LEMMA
  - $*$ Let $a \in J$ and $0 \in \square f(J)$
  - $*$ Then $w(J) \geq |f(a)|/K_J$

- Let $g : \mathbb{R} \longrightarrow \mathbb{R}$ be continuous.
  - $*$ An interval J is large (relative to g) if for all $a \in J$, $w(J) \geq g(a)$.
  - $*$ Call J big (relative to g) if for all $a \in J$, $w(J) \geq \frac{1}{2}g(a)$.
  - $*$ Call J terminal if either $C_0(J)$ or $C_1(J)$.
  - $*$ Call g a stopping function if an interval J that is not large must be terminal.

- A subdivision P of I is a partition
  - $*$ obtained from $P = \{I\}$ by performing repeated bisections of intervals $J \in P$
  - $*$ P is a big subdivision if each $J \in P$ is big
  - $*$ Let $\#(P)$ be the size of P.

- LEMMA
  - $*$ Let P be a big subdivision of I relative to g. Then $\#(P) \leq 2\int_I \frac{da}{g(a)}$.

General Framework of Stopping Functions

- BASIC LEMMA
  - ∗ Let $a \in J$ and $0 \in \square f(J)$
  - ∗ Then $w(J) \geq |f(a)|/K_J$

- Let $g : \mathbb{R} \longrightarrow \mathbb{R}$ be continuous.
  - ∗ An interval J is large (relative to g) if for all $a \in J$, $w(J) \geq g(a)$.
  - ∗ Call J big (relative to g) if for all $a \in J$, $w(J) \geq \frac{1}{2}g(a)$.
  - ∗ Call J terminal if either $C_0(J)$ or $C_1(J)$.
  - ∗ Call g a stopping function if an interval J that is not large must be terminal.

- A subdivision P of I is a partition
  - ∗ obtained from $P = \{I\}$ by performing repeated bisections of intervals $J \in P$
  - ∗ P is a big subdivision if each $J \in P$ is big
  - ∗ Let $\#(P)$ be the size of P.

- LEMMA
  - ∗ Let P be a big subdivision of I relative to g. Then $\#(P) \leq 2\int_I \frac{da}{g(a)}$.

- COROLLARY
  - ∗ The Subdivision P of I at the end of STEP 1 of EVAL has size $\max\{1, \int_I \frac{2da}{g(a)}\}$.

- COROLLARY
  - ∗ The Subdivision P of I at the end of STEP 1 of EVAL has size $\max\{1, \int_I \frac{2da}{g(a)}\}$.

Examples of Stopping Functions

- Global Lipschitz Constants
  - $*$ LEMMA: The function $g_0(a) = \max\{\frac{|f(a)|}{K_I}, \frac{|f'(a)|}{K'_I}\}$ is a stopping function

- Local Lipschitz Constants, K_a for any $a \in I$.
  - $*$ LEMMA: The function $g_1(a) = \max\{\frac{|f(a)|}{K_a}, \frac{|f'(a)|}{K'_a}\}$ is a stopping function

- "Ideal stopping function", $g_*(a) = \max\{\left|\frac{|f(a)|}{f'(a)}\right|, \left|\frac{|f'(a)|}{f''(a)}\right|\}$.

Examples of Stopping Functions

- Global Lipschitz Constants
  - LEMMA: The function $g_0(a) = \max\{\frac{|f(a)|}{K_I}, \frac{|f'(a)|}{K'_I}\}$ is a stopping function
- Local Lipschitz Constants, K_a for any $a \in I$.
  - LEMMA: The function $g_1(a) = \max\{\frac{|f(a)|}{K_a}, \frac{|f'(a)|}{K'_a}\}$ is a stopping function
- "Ideal stopping function", $g_*(a) = \max\{\left|\frac{|f(a)|}{f'(a)}\right|, \left|\frac{|f'(a)|}{f''(a)}\right|\}$.

Examples of Stopping Functions

- Global Lipschitz Constants
  - $*$ LEMMA: The function $g_0(a) = \max\{\frac{|f(a)|}{K_I}, \frac{|f'(a)|}{K'_I}\}$ is a stopping function

- Local Lipschitz Constants, K_a for any $a \in I$.
  - $*$ LEMMA: The function $g_1(a) = \max\{\frac{|f(a)|}{K_a}, \frac{|f'(a)|}{K'_a}\}$ is a stopping function

- "Ideal stopping function", $g_*(a) = \max\{\left|\frac{|f(a)|}{f'(a)}\right|, \left|\frac{|f'(a)|}{f''(a)}\right|\}$.

Examples of Stopping Functions

- Global Lipschitz Constants
  - $*$ LEMMA: The function $g_0(a) = \max\{\frac{|f(a)|}{K_I}, \frac{|f'(a)|}{K'_I}\}$ is a stopping function

- Local Lipschitz Constants, K_a for any $a \in I$.
  - $*$ LEMMA: The function $g_1(a) = \max\{\frac{|f(a)|}{K_a}, \frac{|f'(a)|}{K'_a}\}$ is a stopping function

- "Ideal stopping function", $g_*(a) = \max\{\left|\frac{|f(a)|}{f'(a)}\right|, \left|\frac{|f'(a)|}{f''(a)}\right|\}$.

Bounding the Ideal Stopping Function

- BENCHMARK PROBLEM: let $f(x)$ an integer polynomial of degree d and the logarithmic height is $L = \lg \|f\|$
  - $*$ What is the complexity of isolating all the roots of f?
  - $*$ We know that $O(dL)$ is optimal (assuming $L \geq \log d$)

- Consider ideal case: $R := \int_a^b \frac{da}{g_*(a)} := \int_a^b \min\{\left|\frac{|f'(a)|}{f(a)}\right|, \left|\frac{|f''(a)|}{f'(a)}\right|\}da.$
  - $*$ THEOREM: $R = O(dL)$
  - $*$ SKETCH: R can be broken up into regions $D \subseteq [a, b]$ where f'/f, $-f'/f$, f''/f', $-f''/f'$ dominate. Consider f'/f: $\int_D f'/f = \log \prod_i f(b_i)/f(a_i)$

- Amortization Arguments for Evaluation: let $f, h \in \mathbb{Z}[x]$ have degrees m, n (resp.)
  - $*$ Let $\texttt{lead}(h) = b$ and $\beta_1, \ldots, \beta_k$ be complex zeros of h.
  - $*$ Then $\prod_i |f(\beta_i)| \leq (m+1)\|f\|^k (M(h)/b)^m$
  - $*$ If h, f are relatively prime then $\prod_i |f(\beta_i)| \geq (m+1)\|f\|^{k-n}(M(h)/b)^{-m}$

Bounding the Ideal Stopping Function

- BENCHMARK PROBLEM: let $f(x)$ an integer polynomial of degree d and the logarithmic height is $L = \lg \|f\|$
  - ∗ What is the complexity of isolating all the roots of f?
  - ∗ We know that $O(dL)$ is optimal (assuming $L \geq \log d$)

- Consider ideal case: $R := \int_a^b \frac{da}{g_*(a)} := \int_a^b \min\{\left|\frac{|f'(a)|}{f(a)}\right|, \left|\frac{|f''(a)|}{f'(a)}\right|\}da.$
  - ∗ THEOREM: $R = O(dL)$
  - ∗ SKETCH: R can be broken up into regions $D \subseteq [a,b]$ where f'/f, $-f'/f$, f''/f', $-f''/f'$ dominate. Consider f'/f: $\int_D f'/f = \log \prod_i f(b_i)/f(a_i)$

- Amortization Arguments for Evaluation: let $f, h \in \mathbb{Z}[x]$ have degrees m, n (resp.)
  - ∗ Let $\texttt{lead}(h) = b$ and $\beta_1, \ldots, \beta_k$ be complex zeros of h.
  - ∗ Then $\prod_i |f(\beta_i)| \leq (m+1)\|f\|^k (M(h)/b)^m$
  - ∗ If h, f are relatively prime then $\prod_i |f(\beta_i)| \geq (m+1)\|f\|^{k-n}(M(h)/b)^{-m}$

Bounding the Ideal Stopping Function

- BENCHMARK PROBLEM: let $f(x)$ an integer polynomial of degree d and the logarithmic height is $L = \lg \|f\|$
    - ∗ What is the complexity of isolating all the roots of f?
    - ∗ We know that $O(dL)$ is optimal (assuming $L \geq \log d$)

- Consider ideal case: $R := \int_a^b \frac{da}{g_*(a)} := \int_a^b \min\{\left|\frac{|f'(a)|}{f(a)}\right|, \left|\frac{|f''(a)|}{f'(a)}\right|\} da.$
    - ∗ THEOREM: $R = O(dL)$
    - ∗ SKETCH: R can be broken up into regions $D \subseteq [a, b]$ where f'/f, $-f'/f$, f''/f', $-f''/f'$ dominate. Consider f'/f: $\int_D f'/f = \log \prod_i f(b_i)/f(a_i)$

- Amortization Arguments for Evaluation: let $f, h \in \mathbb{Z}[x]$ have degrees m, n (resp.)
    - ∗ Let $\texttt{lead}(h) = b$ and $\beta_1, \ldots, \beta_k$ be complex zeros of h.
    - ∗ Then $\prod_i |f(\beta_i)| \leq (m + 1)\|f\|^k (M(h)/b)^m$
    - ∗ If h, f are relatively prime then $\prod_i |f(\beta_i)| \geq (m + 1)\|f\|^{k-n}(M(h)/b)^{-m}$

Bounding the Ideal Stopping Function

- BENCHMARK PROBLEM: let $f(x)$ an integer polynomial of degree d and the logarithmic height is $L = \lg \|f\|$
  - $*$ What is the complexity of isolating all the roots of f?
  - $*$ We know that $O(dL)$ is optimal (assuming $L \geq \log d$)

- Consider ideal case: $R := \int_a^b \frac{da}{g_*(a)} := \int_a^b \min\{\left|\frac{|f'(a)|}{f(a)}\right|, \left|\frac{|f''(a)|}{f'(a)}\right|\} da.$
  - $*$ THEOREM: $R = O(dL)$
  - $*$ SKETCH: R can be broken up into regions $D \subseteq [a,b]$ where f'/f, $-f'/f$, f''/f', $-f''/f'$ dominate. Consider f'/f: $\int_D f'/f = \log \prod_i f(b_i)/f(a_i)$

- Amortization Arguments for Evaluation: let $f, h \in \mathbb{Z}[x]$ have degrees m, n (resp.)
  - $*$ Let $\texttt{lead}(h) = b$ and $\beta_1, \ldots, \beta_k$ be complex zeros of h.
  - $*$ Then $\prod_i |f(\beta_i)| \leq (m+1)\|f\|^k (M(h)/b)^m$
  - $*$ If h, f are relatively prime then $\prod_i |f(\beta_i)| \geq (m+1)\|f\|^{k-n}(M(h)/b)^{-m}$

A Priori Complexity Bound

- MAIN THEOREM: EVAL makes $O(d^2 L)$ subdivisions
  - Current proof assumes f', f'' relatively prime

- Ideas: need “gamma function” $\gamma(x) = \gamma_f(x) = \max_{i \geq 2} \left(\frac{|f^{(i)}(x)|}{i!|f'(x)|} \right)^{1/(i-1)}$.
  - Roughly, $1/\gamma(x)$ is the radius of Newton convergence

- Let $G(a) := \min \left\{ \frac{1}{2\gamma(a)}, \frac{|f(a)|}{2d|f'(a)|} \right\}$.
  - Note: Let $G'(a)$ be similarly defined, but for f' instead of f.

- LEMMA: If there exists $b \in J$ such that $w(J) < G(b)$ then $C_0(J)$ holds
  - COR: $g(a) := \max\{G(a), G'(a)\}$ is a stopping function.

- Conceptual algorithm
  - STAGE 1: keep splitting intervals J in a subdivision of I until one of:
  - (a) $\#(J \cap \textsc{Zero}(f f')) > 1$
  - (b) $\#(J \cap \textsc{Zero}(f f')) = 1$ and $w(J) \geq \min\{B(\alpha), \rho(\alpha)/4d(d-1)\}$
  - STAGE 2: for each J, split if $w(J) \geq G(a)$ for all $a \in J$

A Priori Complexity Bound

- MAIN THEOREM: EVAL makes $O(d^2 L)$ subdivisions
  - $*$ Current proof assumes f', f'' relatively prime

- Ideas: need "gamma function" $\gamma(x) = \gamma_f(x) = \max_{i \geq 2} \left(\frac{|f^{(i)}(x)|}{i!|f'(x)|} \right)^{1/(i-1)}$.
  - $*$ Roughly, $1/\gamma(x)$ is the radius of Newton convergence

- Let $G(a) := \min \left\{ \frac{1}{2\gamma(a)}, \frac{|f(a)|}{2d|f'(a)|} \right\}$.
  - $*$ Note: Let $G'(a)$ be similarly defined, but for f' instead of f.

- LEMMA: If there exists $b \in J$ such that $w(J) < G(b)$ then $C_0(J)$ holds
  - $*$ COR: $g(a) := \max\{G(a), G'(a)\}$ is a stopping function.

- Conceptual algorithm
  - $*$ STAGE 1: keep splitting intervals J in a subdivision of I until one of:
  - $*$ (a) $\#(J \cap \textsc{Zero}(ff')) > 1$
  - $*$ (b) $\#(J \cap \textsc{Zero}(ff')) = 1$ and $w(J) \geq \min\{B(\alpha), \rho(\alpha)/4d(d-1)\}$
  - $*$ STAGE 2: for each J, split if $w(J) \geq G(a)$ for all $a \in J$

A Priori Complexity Bound

- MAIN THEOREM: EVAL makes $O(d^2 L)$ subdivisions
  - $*$ Current proof assumes f', f'' relatively prime
- Ideas: need "gamma function" $\gamma(x) = \gamma_f(x) = \max_{i \geq 2} \left(\frac{|f^{(i)}(x)|}{i!|f'(x)|} \right)^{1/(i-1)}$.
  - $*$ Roughly, $1/\gamma(x)$ is the radius of Newton convergence
- Let $G(a) := \min \left\{ \frac{1}{2\gamma(a)}, \frac{|f(a)|}{2d|f'(a)|} \right\}$.
  - $*$ Note: Let $G'(a)$ be similarly defined, but for f' instead of f.
- LEMMA: If there exists $b \in J$ such that $w(J) < G(b)$ then $C_0(J)$ holds
  - $*$ COR: $g(a) := \max\{G(a), G'(a)\}$ is a stopping function.
- Conceptual algorithm
  - $*$ STAGE 1: keep splitting intervals J in a subdivision of I until one of:
  - $*$ (a) $\#(J \cap \text{Zero}(ff')) > 1$
  - $*$ (b) $\#(J \cap \text{Zero}(ff')) = 1$ and $w(J) \geq \min\{B(\alpha), \rho(\alpha)/4d(d-1)\}$
  - $*$ STAGE 2: for each J, split if $w(J) \geq G(a)$ for all $a \in J$

A Priori Complexity Bound

- MAIN THEOREM: EVAL makes $O(d^2L)$ subdivisions
  - $*$ Current proof assumes f', f'' relatively prime

- Ideas: need "gamma function" $\gamma(x) = \gamma_f(x) = \max_{i\geq 2}\left(\frac{|f^{(i)}(x)|}{i!|f'(x)|}\right)^{1/(i-1)}$.
  - $*$ Roughly, $1/\gamma(x)$ is the radius of Newton convergence

- Let $G(a) := \min\left\{\frac{1}{2\gamma(a)}, \frac{|f(a)|}{2d|f'(a)|}\right\}$.
  - $*$ Note: Let $G'(a)$ be similarly defined, but for f' instead of f.

- LEMMA: If there exists $b \in J$ such that $w(J) < G(b)$ then $C_0(J)$ holds
  - $*$ COR: $g(a) := \max\{G(a), G'(a)\}$ is a stopping function.

- Conceptual algorithm
  - $*$ STAGE 1: keep splitting intervals J in a subdivision of I until one of:
  - $*$ (a) $\#(J \cap \text{ZERO}(ff')) > 1$
  - $*$ (b) $\#(J \cap \text{ZERO}(ff')) = 1$ and $w(J) \geq \min\{B(\alpha), \rho(\alpha)/4d(d-1)\}$
  - $*$ STAGE 2: for each J, split if $w(J) \geq G(a)$ for all $a \in J$

A Priori Complexity Bound

- MAIN THEOREM: EVAL makes $O(d^2L)$ subdivisions
  - Current proof assumes f', f'' relatively prime

- Ideas: need "gamma function" $\gamma(x) = \gamma_f(x) = \max_{i\geq 2}\left(\frac{|f^{(i)}(x)|}{i!|f'(x)|}\right)^{1/(i-1)}$.
  - Roughly, $1/\gamma(x)$ is the radius of Newton convergence

- Let $G(a) := \min\left\{\frac{1}{2\gamma(a)}, \frac{|f(a)|}{2d|f'(a)|}\right\}$.
  - Note: Let $G'(a)$ be similarly defined, but for f' instead of f.

- LEMMA: If there exists $b \in J$ such that $w(J) < G(b)$ then $C_0(J)$ holds
  - COR: $g(a) := \max\{G(a), G'(a)\}$ is a stopping function.

- Conceptual algorithm
  - STAGE 1: keep splitting intervals J in a subdivision of I until one of:
  - (a) $\#(J \cap \textsc{Zero}(ff')) > 1$
  - (b) $\#(J \cap \textsc{Zero}(ff')) = 1$ and $w(J) \geq \min\{B(\alpha), \rho(\alpha)/4d(d-1)\}$
  - STAGE 2: for each J, split if $w(J) \geq G(a)$ for all $a \in J$

A Priori Complexity Bound

- MAIN THEOREM: EVAL makes $O(d^2L)$ subdivisions
  - $*$ Current proof assumes f', f'' relatively prime
- Ideas: need "gamma function" $\gamma(x) = \gamma_f(x) = \max_{i\geq 2}\left(\frac{|f^{(i)}(x)|}{i!|f'(x)|}\right)^{1/(i-1)}$.
  - $*$ Roughly, $1/\gamma(x)$ is the radius of Newton convergence
- Let $G(a) := \min\left\{\frac{1}{2\gamma(a)}, \frac{|f(a)|}{2d|f'(a)|}\right\}$.
  - $*$ Note: Let $G'(a)$ be similarly defined, but for f' instead of f.
- LEMMA: If there exists $b \in J$ such that $w(J) < G(b)$ then $C_0(J)$ holds
  - $*$ COR: $g(a) := \max\{G(a), G'(a)\}$ is a stopping function.
- Conceptual algorithm
  - $*$ STAGE 1: keep splitting intervals J in a subdivision of I until one of:
  - $*$ (a) $\#(J \cap \textsc{Zero}(ff')) > 1$
  - $*$ (b) $\#(J \cap \textsc{Zero}(ff')) = 1$ and $w(J) \geq \min\{B(\alpha), \rho(\alpha)/4d(d-1)\}$
  - $*$ STAGE 2: for each J, split if $w(J) \geq G(a)$ for all $a \in J$

- SKETCH
  - Complexity of STAGE 1 is $O(dL)$.
  - Complexity of STAGE 2 can be bounded by integrals similar to the ideal case.

- SKETCH
  - $*$ Complexity of STAGE 1 is $O(dL)$.
  - $*$ Complexity of STAGE 2 can be bounded by integrals similar to the ideal case.

PART V.
Isotopic Approximation of Singular Curves

Ensuring Isolated Singularities

- Let $f \in \mathbb{Z}[X_1, \ldots, X_n]$.
  - ∗ If $n = 1$, then f is square-free implies $Zero(f)$ has no singularites
  - ∗ WANTED: generalization for $n > 1$

- THEOREM: Let $f_1, \ldots, f_m \in \mathbb{Z}[X_1, \ldots, X_n]$ be non-constant.
  - ∗ IF $GCD(f_1, \ldots, f_m) = 1$
  - ∗ THEN $Zero(f_1, \ldots, f_m) \subseteq \mathbb{C}^n$ has dimension at most $n - 2$.
  - ∗ FURTHER, the dimension $n - 2$ can be achieved

- COROLLARY: Let $f \in \mathbb{Z}[X_1, \ldots, X_n]$ be squarefree.
  - ∗ The singularities of the surface $f = 0$ has dimension at most $n - 2$.

- COROLLARY: Let $f \in \mathbb{Z}[X, Y]$ be squarefree.
  - ∗ The curve $f = 0$ has only isolated singularities

Ensuring Isolated Singularities

- Let $f \in \mathbb{Z}[X_1, \ldots, X_n]$.
  - ∗ If $n = 1$, then f is square-free implies $Zero(f)$ has no singularites
  - ∗ WANTED: generalization for $n > 1$

- THEOREM: Let $f_1, \ldots, f_m \in \mathbb{Z}[X_1, \ldots, X_n]$ be non-constant.
  - ∗ IF $GCD(f_1, \ldots, f_m) = 1$
  - ∗ THEN $Zero(f_1, \ldots, f_m) \subseteq \mathbb{C}^n$ has dimension at most $n - 2$.
  - ∗ FURTHER, the dimension $n - 2$ can be achieved

- COROLLARY: Let $f \in \mathbb{Z}[X_1, \ldots, X_n]$ be squarefree.
  - ∗ The singularities of the surface $f = 0$ has dimension at most $n - 2$.

- COROLLARY: Let $f \in \mathbb{Z}[X, Y]$ be squarefree.
  - ∗ The curve $f = 0$ has only isolated singularities

Ensuring Isolated Singularities

- Let $f \in \mathbb{Z}[X_1, \ldots, X_n]$.
  - $*$ If $n = 1$, then f is square-free implies $Zero(f)$ has no singularites
  - $*$ WANTED: generalization for $n > 1$

- THEOREM: Let $f_1, \ldots, f_m \in \mathbb{Z}[X_1, \ldots, X_n]$ be non-constant.
  - $*$ IF $GCD(f_1, \ldots, f_m) = 1$
  - $*$ THEN $Zero(f_1, \ldots, f_m) \subseteq \mathbb{C}^n$ has dimension at most $n - 2$.
  - $*$ FURTHER, the dimension $n - 2$ can be achieved

- COROLLARY: Let $f \in \mathbb{Z}[X_1, \ldots, X_n]$ be squarefree.
  - $*$ The singularities of the surface $f = 0$ has dimension at most $n - 2$.

- COROLLARY: Let $f \in \mathbb{Z}[X, Y]$ be squarefree.
  - $*$ The curve $f = 0$ has only isolated singularities

Ensuring Isolated Singularities

- Let $f \in \mathbb{Z}[X_1, \ldots, X_n]$.
  - $*$ If $n = 1$, then f is square-free implies $Zero(f)$ has no singularites
  - $*$ WANTED: generalization for $n > 1$

- THEOREM: Let $f_1, \ldots, f_m \in \mathbb{Z}[X_1, \ldots, X_n]$ be non-constant.
  - $*$ IF $GCD(f_1, \ldots, f_m) = 1$
  - $*$ THEN $Zero(f_1, \ldots, f_m) \subseteq \mathbb{C}^n$ has dimension at most $n - 2$.
  - $*$ FURTHER, the dimension $n - 2$ can be achieved

- COROLLARY: Let $f \in \mathbb{Z}[X_1, \ldots, X_n]$ be squarefree.
  - $*$ The singularities of the surface $f = 0$ has dimension at most $n - 2$.

- COROLLARY: Let $f \in \mathbb{Z}[X, Y]$ be squarefree.
  - $*$ The curve $f = 0$ has only isolated singularities

Ensuring Isolated Singularities

- Let $f \in \mathbb{Z}[X_1, \ldots, X_n]$.
  - $*$ If $n = 1$, then f is square-free implies $Zero(f)$ has no singularites
  - $*$ WANTED: generalization for $n > 1$

- THEOREM: Let $f_1, \ldots, f_m \in \mathbb{Z}[X_1, \ldots, X_n]$ be non-constant.
  - $*$ IF $GCD(f_1, \ldots, f_m) = 1$
  - $*$ THEN $Zero(f_1, \ldots, f_m) \subseteq \mathbb{C}^n$ has dimension at most $n - 2$.
  - $*$ FURTHER, the dimension $n - 2$ can be achieved

- COROLLARY: Let $f \in \mathbb{Z}[X_1, \ldots, X_n]$ be squarefree.
  - $*$ The singularities of the surface $f = 0$ has dimension at most $n - 2$.

- COROLLARY: Let $f \in \mathbb{Z}[X, Y]$ be squarefree.
  - $*$ The curve $f = 0$ has only isolated singularities

How to Locate Isolated Singularities WITHOUT ALGEBRA

- PREVIOUS METHODS:
  - ∗ Requires resultants (E.g., [Wolpert-Seidel]
  - ∗ Requires root isolation on box boundary (E.g., using topological degree, [Mourrain]

- STEP 1: CONSTRUCT $F = f^2 + f_x^2 + f_y^2$
  - ∗ So: $F(p) = 0$ iff p is a singular point of $f = 0$

- STEP 2: Perturb F
  - ∗ Let $\varepsilon_1 = \frac{1}{2}\mathrm{EV}(F)$
  - ∗ The Function $F - \varepsilon_1$ is non-singular
  - ∗ The curve of $\{F = \varepsilon_1\}$ can be solved by Vegter-Plantinga!

- STEP 3: Consider singular regions of $\{F < \varepsilon_1\}$
  - ∗ Bounded singular regions contains a unique singular point of $f = 0$
  - ∗ Unbounded singular regions may contain singular points
  - ∗ BUT: can choose a tighter $\varepsilon_2 > 0$ to ensure this cannot happen

How to Locate Isolated Singularities WITHOUT ALGEBRA

- PREVIOUS METHODS:
  - ∗ Requires resultants (E.g., [Wolpert-Seidel]
  - ∗ Requires root isolation on box boundary (E.g., using topological degree, [Mourrain]

- STEP 1: CONSTRUCT $F = f^2 + f_x^2 + f_y^2$
  - ∗ So: $F(p) = 0$ iff p is a singular point of $f = 0$

- STEP 2: Perturb F
  - ∗ Let $\varepsilon_1 = \frac{1}{2}\mathrm{EV}(F)$
  - ∗ The Function $F - \varepsilon_1$ is non-singular
  - ∗ The curve of $\{F = \varepsilon_1\}$ can be solved by Vegter-Plantinga!

- STEP 3: Consider singular regions of $\{F < \varepsilon_1\}$
  - ∗ Bounded singular regions contains a unique singular point of $f = 0$
  - ∗ Unbounded singular regions may contain singular points
  - ∗ BUT: can choose a tighter $\varepsilon_2 > 0$ to ensure this cannot happen

How to Locate Isolated Singularities WITHOUT ALGEBRA

- PREVIOUS METHODS:
  - ∗ Requires resultants (E.g., [Wolpert-Seidel]
  - ∗ Requires root isolation on box boundary (E.g., using topological degree, [Mourrain]

- STEP 1: CONSTRUCT $F = f^2 + f_x^2 + f_y^2$
  - ∗ So: $F(p) = 0$ iff p is a singular point of $f = 0$

- STEP 2: Perturb F
  - ∗ Let $\varepsilon_1 = \frac{1}{2}\mathrm{EV}(F)$
  - ∗ The Function $F - \varepsilon_1$ is non-singular
  - ∗ The curve of $\{F = \varepsilon_1\}$ can be solved by Vegter-Plantinga!

- STEP 3: Consider singular regions of $\{F < \varepsilon_1\}$
  - ∗ Bounded singular regions contains a unique singular point of $f = 0$
  - ∗ Unbounded singular regions may contain singular points
  - ∗ BUT: can choose a tighter $\varepsilon_2 > 0$ to ensure this cannot happen

How to Locate Isolated Singularities WITHOUT ALGEBRA

- PREVIOUS METHODS:
  - ∗ Requires resultants (E.g., [Wolpert-Seidel]
  - ∗ Requires root isolation on box boundary (E.g., using topological degree, [Mourrain]

- STEP 1: CONSTRUCT $F = f^2 + f_x^2 + f_y^2$
  - ∗ So: $F(p) = 0$ iff p is a singular point of $f = 0$

- STEP 2: Perturb F
  - ∗ Let $\varepsilon_1 = \frac{1}{2}\mathrm{EV}(F)$
  - ∗ The Function $F - \varepsilon_1$ is non-singular
  - ∗ The curve of $\{F = \varepsilon_1\}$ can be solved by Vegter-Plantinga!

- STEP 3: Consider singular regions of $\{F < \varepsilon_1\}$
  - ∗ Bounded singular regions contains a unique singular point of $f = 0$
  - ∗ Unbounded singular regions may contain singular points
  - ∗ BUT: can choose a tighter $\varepsilon_2 > 0$ to ensure this cannot happen

How to Locate Isolated Singularities WITHOUT ALGEBRA

- PREVIOUS METHODS:
  - ∗ Requires resultants (E.g., [Wolpert-Seidel]
  - ∗ Requires root isolation on box boundary (E.g., using topological degree, [Mourrain]

- STEP 1: CONSTRUCT $F = f^2 + f_x^2 + f_y^2$
  - ∗ So: $F(p) = 0$ iff p is a singular point of $f = 0$

- STEP 2: Perturb F
  - ∗ Let $\varepsilon_1 = \frac{1}{2}\mathrm{EV}(F)$
  - ∗ The Function $F - \varepsilon_1$ is non-singular
  - ∗ The curve of $\{F = \varepsilon_1\}$ can be solved by Vegter-Plantinga!

- STEP 3: Consider singular regions of $\{F < \varepsilon_1\}$
  - ∗ Bounded singular regions contains a unique singular point of $f = 0$
  - ∗ Unbounded singular regions may contain singular points
  - ∗ BUT: can choose a tighter $\varepsilon_2 > 0$ to ensure this cannot happen

How to Determine the Singularity Degree

- GOAL: Determine the Degree of an isolated singularity
  - i.e., its vertex degree when the curve S is viewed as a graph

- Let δ be smaller than:
  - (1) The separation bound between singular points [SoCG'06]
  - (2) $\|p-q\|$ where p is singular point, $q \in S$ and $\nabla f(q) \| (p-q)$

- PROPERTY: if p is singular point
  - If $q \in S$ and $\|p-q\| < \delta$, then one of the paths from q will converge monontonically to p

How to Determine the Singularity Degree

- GOAL: Determine the Degree of an isolated singularity
  - ∗ i.e., its vertex degree when the curve S is viewed as a graph

- Let δ be smaller than:
  - ∗ (1) The separation bound between singular points [SoCG'06]
  - ∗ (2) $\|p - q\|$ where p is singular point, $q \in S$ and $\nabla f(q) \| (p - q)$

- PROPERTY: if p is singular point

  ∗ If $q \in S$ and $\|p - q\| < \delta$, then one of the paths from q will converge monontonically to p

How to Determine the Singularity Degree

- GOAL: Determine the Degree of an isolated singularity
  - i.e., its vertex degree when the curve S is viewed as a graph

- Let δ be smaller than:
  - (1) The separation bound between singular points [SoCG'06]
  - (2) $\|p - q\|$ where p is singular point, $q \in S$ and $\nabla f(q) \| (p - q)$

- PROPERTY: if p is singular point
  - If $q \in S$ and $\|p - q\| < \delta$, then one of the paths from q will converge monontonically to p

How to Determine the Singularity Degree

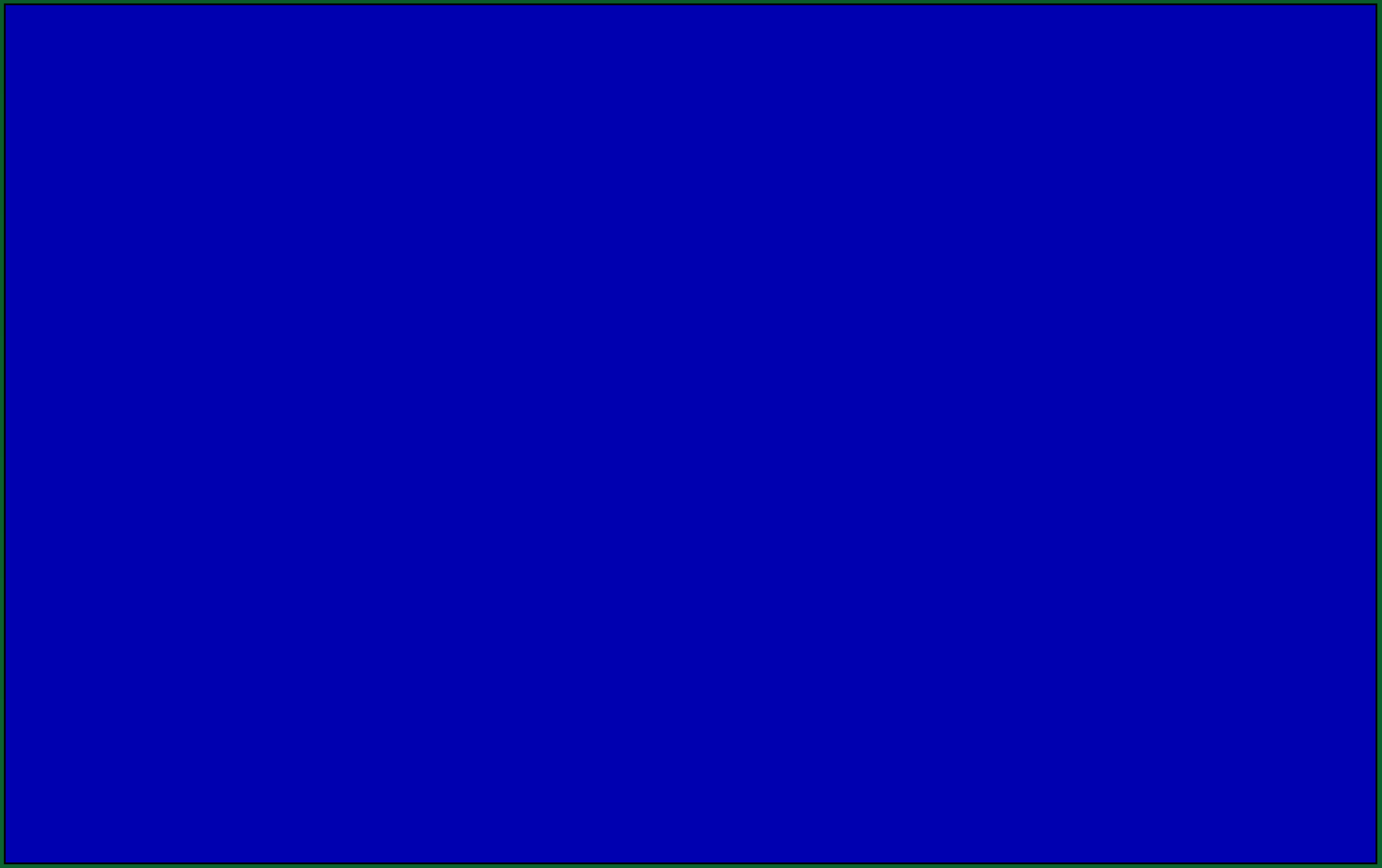
- GOAL: Determine the Degree of an isolated singularity
  - ∗ i.e., its vertex degree when the curve S is viewed as a graph

- Let δ be smaller than:
  - ∗ (1) The separation bound between singular points [SoCG'06]
  - ∗ (2) $\|p - q\|$ where p is singular point, $q \in S$ and $\nabla f(q) \| (p - q)$

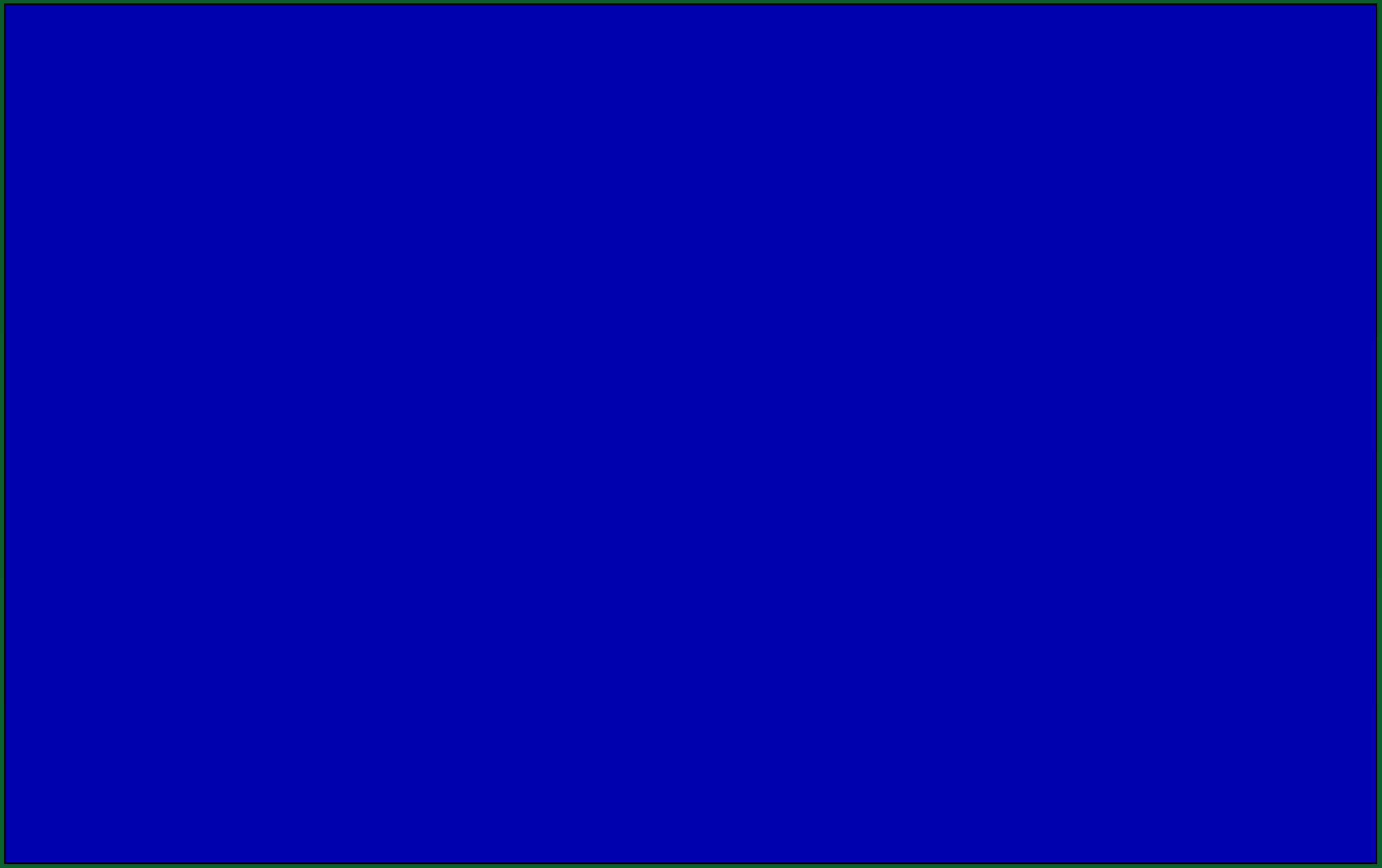
- PROPERTY: if p is singular point

  ∗ If $q \in S$ and $\|p - q\| < \delta$, then one of the paths from q will converge monontonically to p

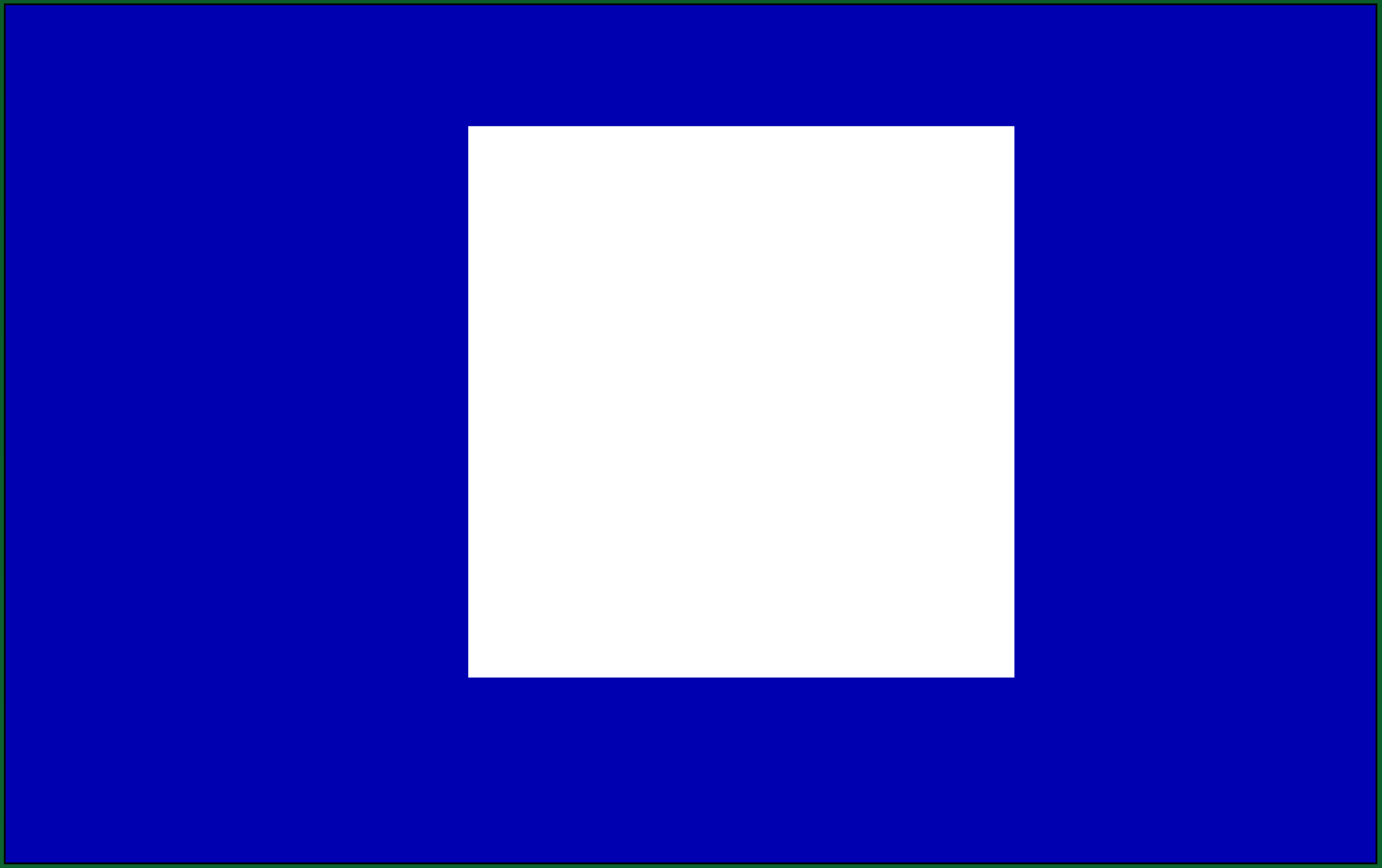
- Let R be an Annular region (B, B')
  - $*$ $w(B) < \delta$ and B' contains a singular point p



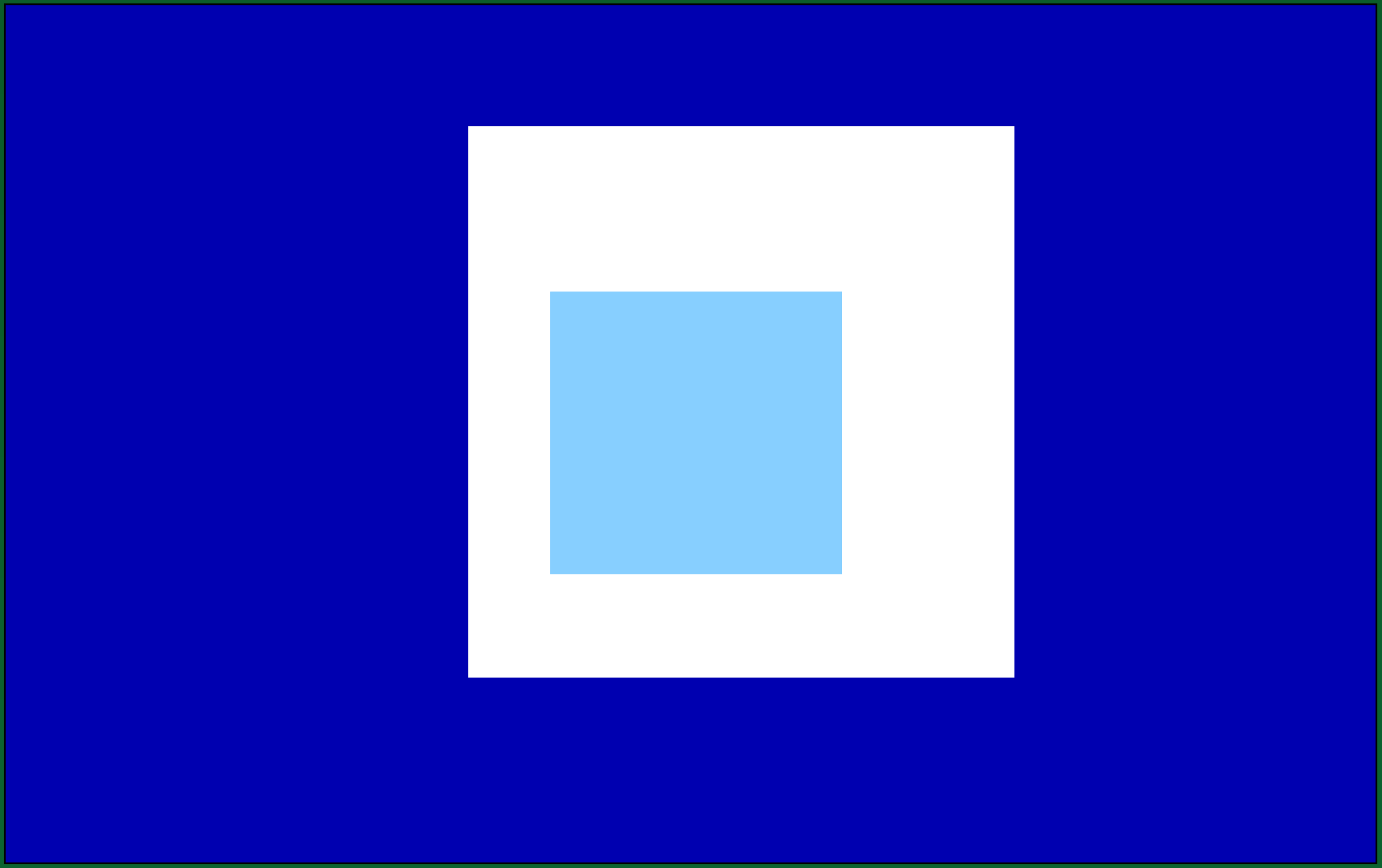
- Let R be an Annular region (B, B')
  - $*$ $w(B) < \delta$ and B' contains a singular point p



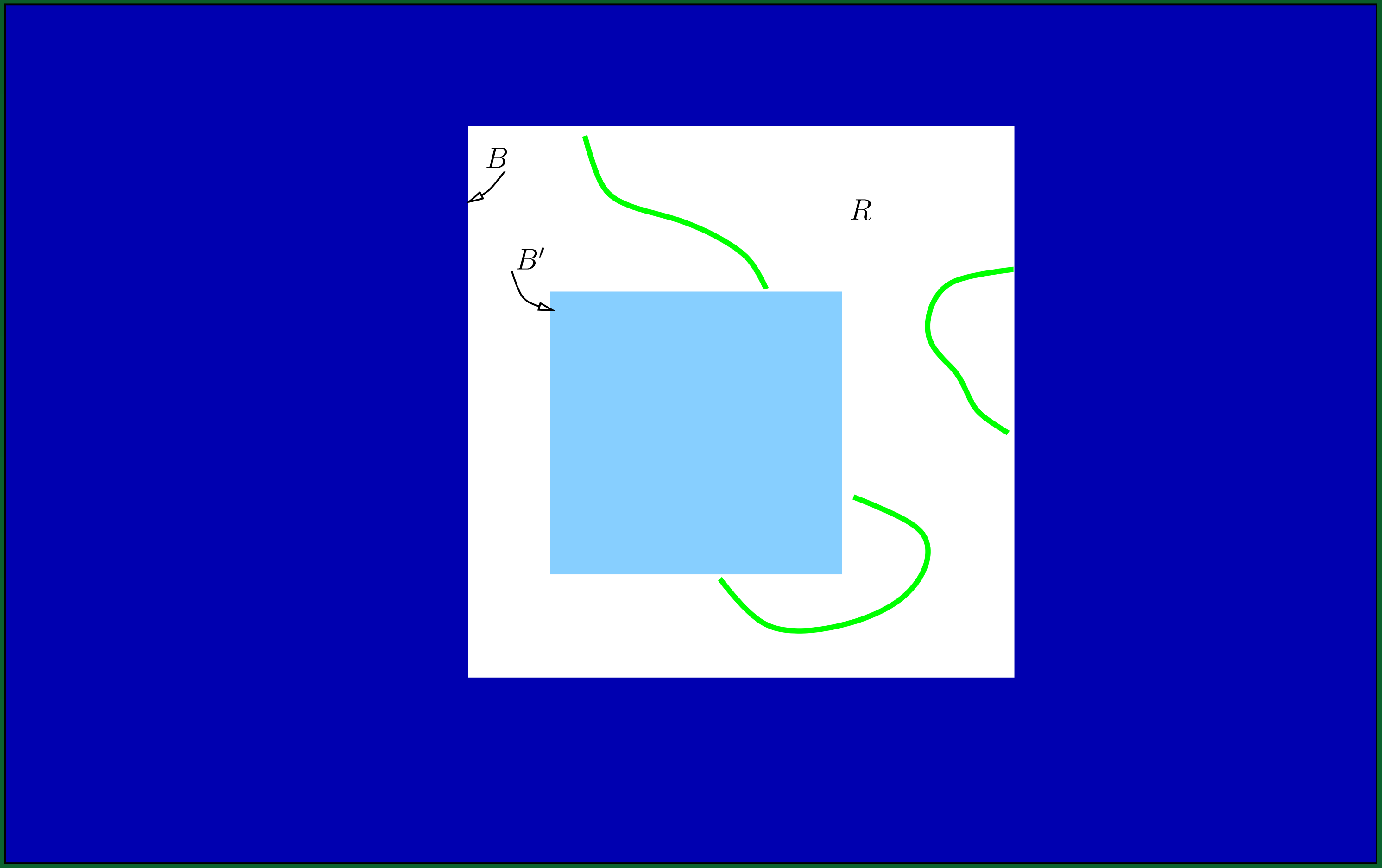
- Let R be an Annular region (B, B')
  - $*$ $w(B) < \delta$ and B' contains a singular point p



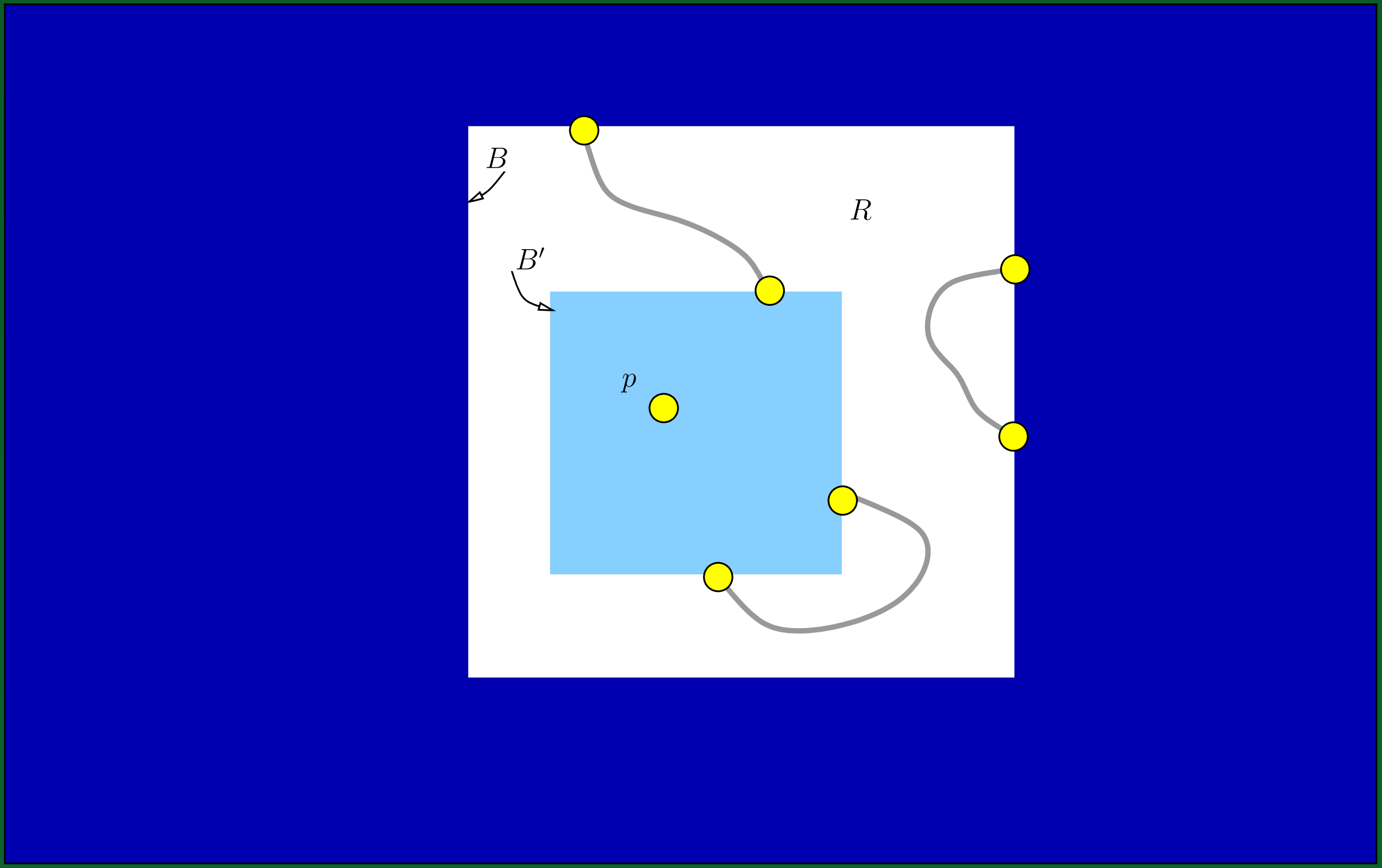
- Let R be an Annular region (B, B')
  - $*$ $w(B) < \delta$ and B' contains a singular point p



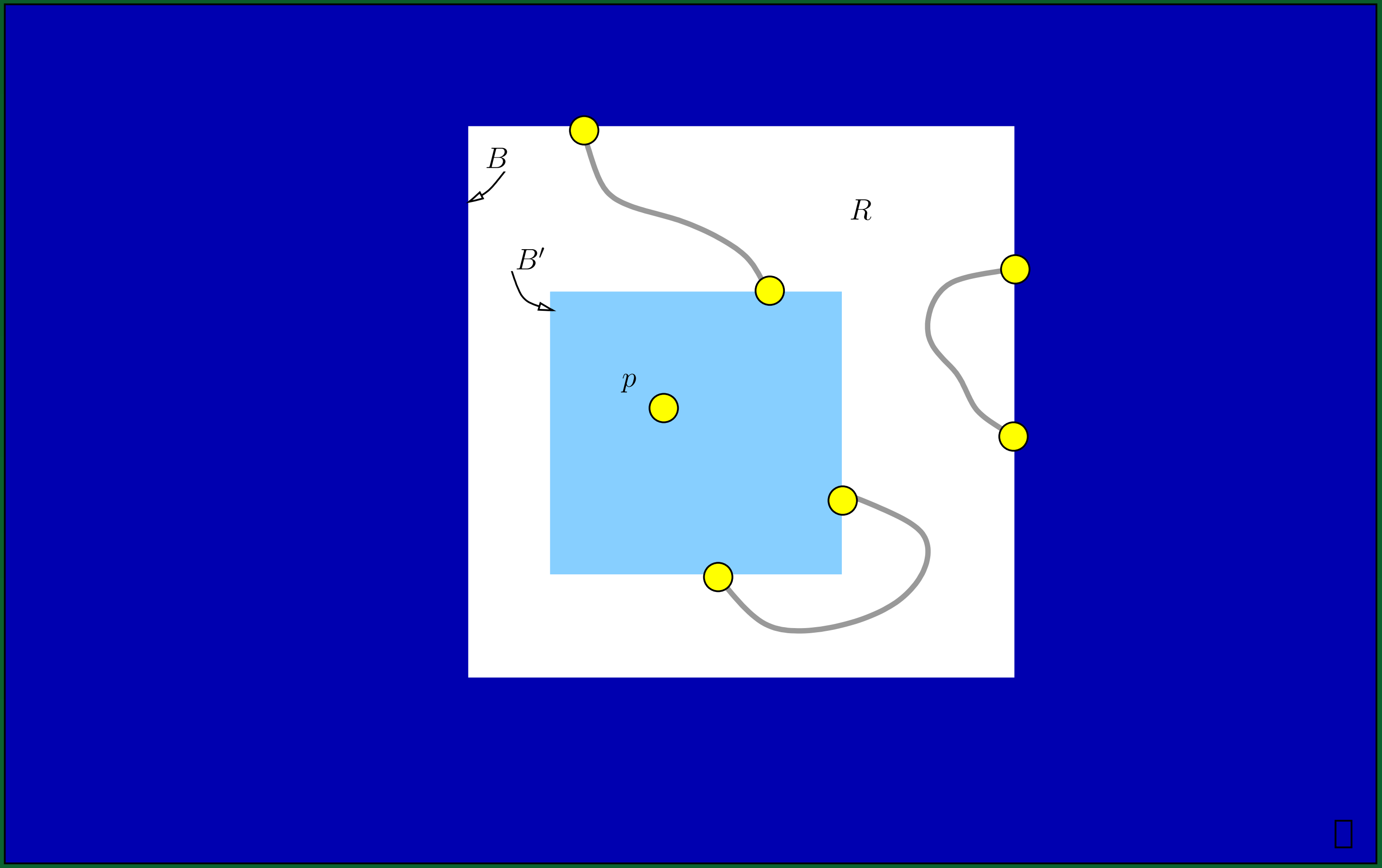
- Let R be an Annular region (B, B')
    - $*$ $w(B) < \delta$ and B' contains a singular point p



- Let R be an Annular region (B, B')
    - $*$ $w(B) < \delta$ and B' contains a singular point p

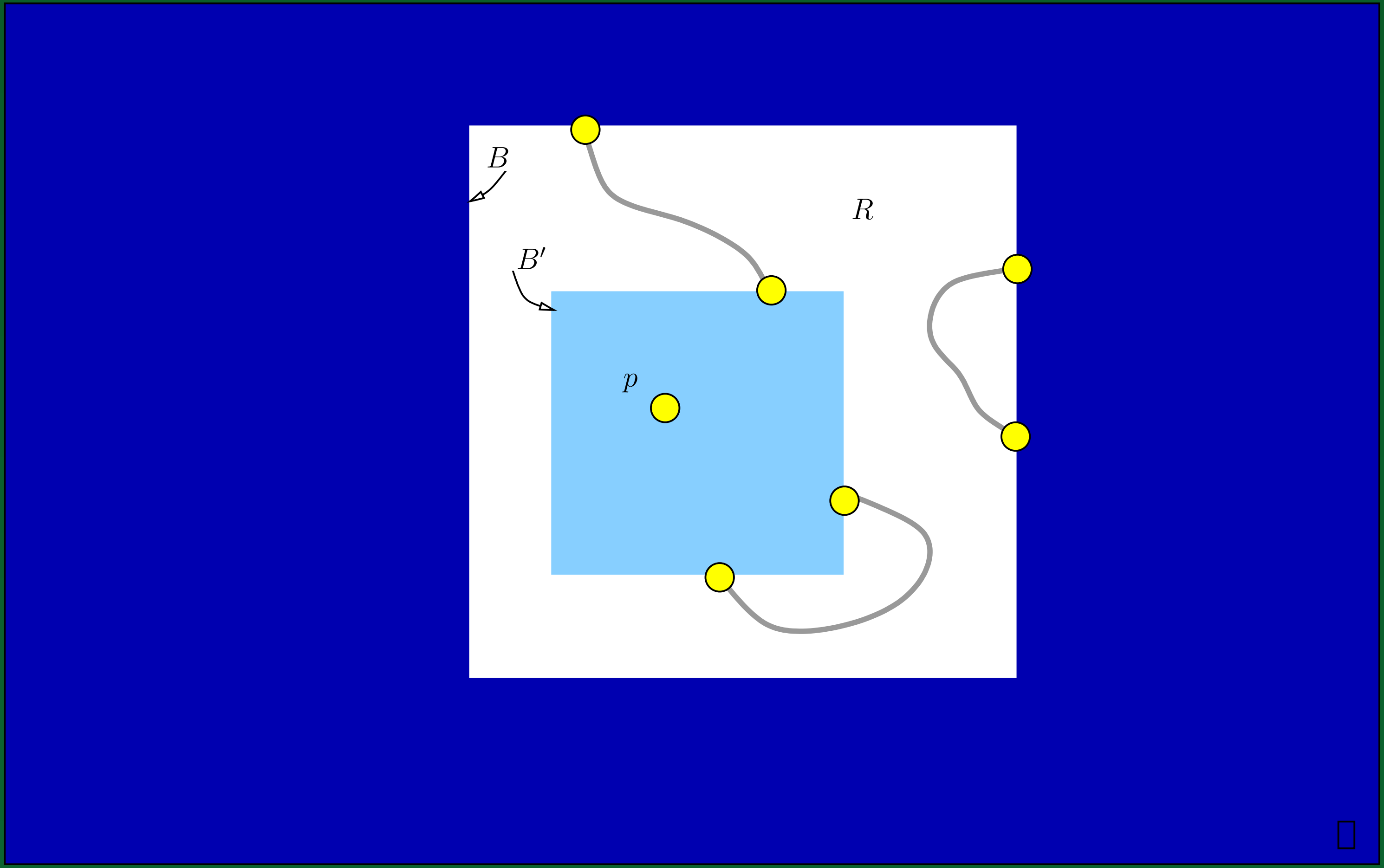


- Let R be an Annular region (B, B')
  - $*$ $w(B) < \delta$ and B' contains a singular point p



- THEOREM:
  - $*$ Degree of p is the number of paths of $R \cap S$ from ∂B to $\partial B'$

- Let R be an Annular region (B, B')
  - $*$ $w(B) < \delta$ and B' contains a singular point p



- THEOREM:
  - $*$ Degree of p is the number of paths of $R \cap S$ from ∂B to $\partial B'$

CONCLUSION

- Why Adaptive Algorithms?
  - ∗ Fast on typical inputs
  - ∗ Simpler algorithms
  - ∗ Exploits local geometry
  - ∗ Generality – not restricted to algebraic case
  - ∗ Algorithm uses, but is decoupled from, root bounds

- Main Challenges of Adaptive Algorithms
  - ∗ (1) Achieve completeness via purely numerical means
  - ∗ (2) Complexity Analysis

- This talk
  - ∗ (1) First complete numerical solution for curve approximation
  - ∗ (2) Integral analysis of EVAL
  - ∗ (3) EVAL has complexity $O(d^2L)$ for benchmark problem

- OPEN PROBLEMS
  - ∗ (1) Extend to 3-D

CONCLUSION

- Why Adaptive Algorithms?
    - ∗ Fast on typical inputs
    - ∗ Simpler algorithms
    - ∗ Exploits local geometry
    - ∗ Generality – not restricted to algebraic case
    - ∗ Algorithm uses, but is decoupled from, root bounds
- Main Challenges of Adaptive Algorithms
    - ∗ (1) Achieve completeness via purely numerical means
    - ∗ (2) Complexity Analysis
- This talk
    - ∗ (1) First complete numerical solution for curve approximation
    - ∗ (2) Integral analysis of EVAL
    - ∗ (3) EVAL has complexity $O(d^2L)$ for benchmark problem
- OPEN PROBLEMS
    - ∗ (1) Extend to 3-D

CONCLUSION

- Why Adaptive Algorithms?
  - ∗ Fast on typical inputs
  - ∗ Simpler algorithms
  - ∗ Exploits local geometry
  - ∗ Generality – not restricted to algebraic case
  - ∗ Algorithm uses, but is decoupled from, root bounds
- Main Challenges of Adaptive Algorithms
  - ∗ (1) Achieve completeness via purely numerical means
  - ∗ (2) Complexity Analysis
- This talk
  - ∗ (1) First complete numerical solution for curve approximation
  - ∗ (2) Integral analysis of EVAL
  - ∗ (3) EVAL has complexity $O(d^2L)$ for benchmark problem
- OPEN PROBLEMS
  - ∗ (1) Extend to 3-D

CONCLUSION

- Why Adaptive Algorithms?
  - ∗ Fast on typical inputs
  - ∗ Simpler algorithms
  - ∗ Exploits local geometry
  - ∗ Generality – not restricted to algebraic case
  - ∗ Algorithm uses, but is decoupled from, root bounds

- Main Challenges of Adaptive Algorithms
  - ∗ (1) Achieve completeness via purely numerical means
  - ∗ (2) Complexity Analysis

- This talk
  - ∗ (1) First complete numerical solution for curve approximation
  - ∗ (2) Integral analysis of EVAL
  - ∗ (3) EVAL has complexity $O(d^2L)$ for benchmark problem

- OPEN PROBLEMS
  - ∗ (1) Extend to 3-D

- $\ast$ (2) Adaptive Complexity Analysis for Singular Case
- $\ast$ (3) Does EVAL have complexity $O(dL)$?

- $*$ (2) Adaptive Complexity Analysis for Singular Case
- $*$ (3) Does EVAL have complexity $O(dL)$?

END of TALK

Thanks for Listening!

FURTHER INFORMATION

- "Complete Subdivision Algorithms I: Intersection of Bezier Curves"
  - ∗ SoCG'06, C.Yap
- "Complete Subdivision Algorithms II: Isotopic Meshing of Singular Algebraic Curves"
  - ∗ M.Burr, S.W.Choi, B.Galehouse and C.Yap
- "Evaluation-based Root Isolation"
  - ∗ M.Burr, V.Sharma, C.Yap
- "Integral Analysis of Evaluation-based Root Isolation"
  - ∗ M.Burr, F.Krahmer, C.Yap

Extra Slides