Toda's theorem - real and complex

Saugata Basu

Purdue University

BIRS, Feb 15, 2010

Outline

Outline

Outline

Outline

Outline

Outline

Some motivation

- The Blum-Shub-Smale model is a natural model to study complexity questions questions of algebraic problems over real as well as complex numbers.
- The role of convexity is mysterious. For instance, semi-definite programming is unlikely to be $\mathbf{NP}_{\mathbb{R}}$-complete but not known to be in $\mathbf{P}_{\mathbb{R}}$ either. (cf. the problem of deciding whether a real quartic polynomial has a zero in $\mathbb{R}^n$ is already $\mathbf{NP}_{\mathbb{R}}$-complete.)
- However, there are various structural complexity results in the B-S-S model that mirrors those in the classical discrete complexity theory.
- In particular, this talk will be on the B-S-S analogue of "counting".

Some motivation

- The Blum-Shub-Smale model is a natural model to study complexity questions questions of algebraic problems over real as well as complex numbers.
- The role of convexity is mysterious. For instance, semi-definite programming is unlikely to be $\mathbf{NP}_{\mathbb{R}}$-complete but not known to be in $\mathbf{P}_{\mathbb{R}}$ either. (cf. the problem of deciding whether a real quartic polynomial has a zero in $\mathbb{R}^n$ is already $\mathbf{NP}_{\mathbb{R}}$-complete.)
- However, there are various structural complexity results in the B-S-S model that mirrors those in the classical discrete complexity theory.
- In particular, this talk will be on the B-S-S analogue of "counting".

Some motivation

- The Blum-Shub-Smale model is a natural model to study complexity questions questions of algebraic problems over real as well as complex numbers.
- The role of convexity is mysterious. For instance, semi-definite programming is unlikely to be $\mathbf{NP}_{\mathbb{R}}$-complete but not known to be in $\mathbf{P}_{\mathbb{R}}$ either. (cf. the problem of deciding whether a real quartic polynomial has a zero in $\mathbb{R}^n$ is already $\mathbf{NP}_{\mathbb{R}}$-complete.)
- However, there are various structural complexity results in the B-S-S model that mirrors those in the classical discrete complexity theory.
- In particular, this talk will be on the B-S-S analogue of "counting".

Some motivation

- The Blum-Shub-Smale model is a natural model to study complexity questions questions of algebraic problems over real as well as complex numbers.
- The role of convexity is mysterious. For instance, semi-definite programming is unlikely to be $\mathbf{NP}_{\mathbb{R}}$-complete but not known to be in $\mathbf{P}_{\mathbb{R}}$ either. (cf. the problem of deciding whether a real quartic polynomial has a zero in $\mathbb{R}^n$ is already $\mathbf{NP}_{\mathbb{R}}$-complete.)
- However, there are various structural complexity results in the B-S-S model that mirrors those in the classical discrete complexity theory.
- In particular, this talk will be on the B-S-S analogue of "counting".

A quick primer of basic definitions and notation

- Initially let $k = \mathbb{Z}/2\mathbb{Z} = \{\bar{0}, \bar{1}\}$.
- A ***language*** L is a set

$$\bigcup_{n>0} L_n, \quad L_n \subset k^n$$

(abusing notation a little we will identify L with the sequence $(L_n)_{n>0}$).

- A language

$$L = (L_n)_{n>0} \in \mathbf{P}$$

if there exists a Turing machine M that given $\mathbf{x} \in k^n$ decides whether $\mathbf{x} \in L_n$ or not in $n^{O(1)}$ time.

A quick primer of basic definitions and notation

- Initially let $k = \mathbb{Z}/2\mathbb{Z} = \{\bar{0}, \bar{1}\}$.
- A ***language*** L is a set

$$\bigcup_{n>0} L_n, \quad L_n \subset k^n$$

(abusing notation a little we will identify L with the sequence $(L_n)_{n>0}$).

- A language

$$L = (L_n)_{n>0} \in \mathbf{P}$$

if there exists a Turing machine M that given $\mathbf{x} \in k^n$ decides whether $\mathbf{x} \in L_n$ or not in $n^{O(1)}$ time.

A quick primer of basic definitions and notation

- Initially let $k = \mathbb{Z}/2\mathbb{Z} = \{\bar{0}, \bar{1}\}$.
- A ***language*** L is a set

$$\bigcup_{n>0} L_n, \quad L_n \subset k^n$$

(abusing notation a little we will identify L with the sequence $(L_n)_{n>0}$).

- A language

$$L = (L_n)_{n>0} \in \mathbf{P}$$

if there exists a Turing machine M that given $\mathbf{x} \in k^n$ decides whether $\mathbf{x} \in L_n$ or not in $n^{O(1)}$ time.

Primer (cont.)

- A language

$$L = (L_n)_{n>0} \in \mathbf{NP}$$

if there exists a polynomial $m(n)$, and a language $L' = (L'_n)_{n>0} \in \mathbf{P}$ such that

$$\mathbf{x} \in L_n \iff (\exists\ \mathbf{y} \in k^{m(n)})\ \ (\mathbf{y}, \mathbf{x}) \in L'_{m+n}.$$

- A language

$$L = (L_n)_{n>0} \in \mathbf{coNP}$$

if there exists a polynomial $m(n)$, and a language $L' = (L'_n)_{n>0} \in \mathbf{P}$ such that

$$\mathbf{x} \in L_n \iff \left(\forall\ \mathbf{y} \in k^{m(n)}\right)\ \ (\mathbf{y}, \mathbf{x}) \in L'_{m+n}.$$

Primer (cont.)

- A language
$$L = (L_n)_{n>0} \in \mathbf{NP}$$
if there exists a polynomial $m(n)$, and a language $L' = (L'_n)_{n>0} \in \mathbf{P}$ such that
$$\mathbf{x} \in L_n \iff (\exists\, \mathbf{y} \in k^{m(n)}) \;\; (\mathbf{y}, \mathbf{x}) \in L'_{m+n}.$$

- A language
$$L = (L_n)_{n>0} \in \mathbf{coNP}$$
if there exists a polynomial $m(n)$, and a language $L' = (L'_n)_{n>0} \in \mathbf{P}$ such that
$$\mathbf{x} \in L_n \iff \left(\forall\, \mathbf{y} \in k^{m(n)}\right) \;\; (\mathbf{y}, \mathbf{x}) \in L'_{m+n}.$$

Discrete Polynomial Time Hierarchy– A Quick Reminder

A language

$$L = (L_n)_{n>0} \in \Sigma_\omega$$

if there exists a language $L' = (L'_n)_{n>0} \in \mathbf{P}$ such that

$$\mathbf{x} \in L_n$$
$$\Updownarrow$$
$$(Q_1 \mathbf{y}^1 \in k^{m_1})(Q_2 \mathbf{y}^2 \in k^{m_2}) \ldots (Q_\omega \mathbf{y}^\omega \in k^{m_\omega})$$
$$(\mathbf{y}^1, \ldots, \mathbf{y}^\omega, \mathbf{x}) \in L'_{m+n}$$

where $m(n) = m_1(n) + \cdots + m_\omega(n) = n^{O(1)}$ and for $1 \leq i \leq \omega$, $Q_i \in \{\exists, \forall\}$, $Q_1 = \exists$.

Discrete Polynomial Time Hierarchy– A Quick Reminder

A language

$$L = (L_n)_{n>0} \in \Sigma_\omega$$

if there exists a language $L' = (L'_n)_{n>0} \in \mathbf{P}$ such that

$$\mathbf{x} \in L_n$$

$$\Updownarrow$$

$$(Q_1\mathbf{y}^1 \in k^{m_1})(Q_2\mathbf{y}^2 \in k^{m_2}) \dots (Q_\omega\mathbf{y}^\omega \in k^{m_\omega})$$
$$(\mathbf{y}^1, \dots, \mathbf{y}^\omega, \mathbf{x}) \in L'_{m+n}$$

where $m(n) = m_1(n) + \cdots + m_\omega(n) = n^{O(1)}$ and for $1 \leq i \leq \omega$, $Q_i \in \{\exists, \forall\}$, $Q_1 = \exists$.

Discrete Polynomial Time Hierarchy– A Quick Reminder

A language

$$L = (L_n)_{n>0} \in \Sigma_\omega$$

if there exists a language $L' = (L'_n)_{n>0} \in \mathbf{P}$ such that

$$\mathbf{x} \in L_n$$

$$\Updownarrow$$

$$(Q_1 \mathbf{y}^1 \in k^{m_1})(Q_2 \mathbf{y}^2 \in k^{m_2}) \ldots (Q_\omega \mathbf{y}^\omega \in k^{m_\omega})$$

$$(\mathbf{y}^1, \ldots, \mathbf{y}^\omega, \mathbf{x}) \in L'_{m+n}$$

where $m(n) = m_1(n) + \cdots + m_\omega(n) = n^{O(1)}$ and for $1 \leq i \leq \omega$, $Q_i \in \{\exists, \forall\}$, $Q_1 = \exists$.

Reminder (cont.)

Similarly a language

$$L = (L_n)_{n>0} \in \Pi_\omega$$

if there exists a language $L' = (L'_n)_{n>0} \in \mathbf{P}$ such that

$$\mathbf{x} \in L_n$$

$$\Updownarrow$$

$$(Q_1\mathbf{y}^1 \in k^{m_1})(Q_2\mathbf{y}^2 \in k^{m_2}) \cdots (Q_\omega\mathbf{y}^\omega \in k^{m_\omega})$$
$$(\mathbf{y}^1, \ldots, \mathbf{y}^\omega, \mathbf{x}) \in L'_{m+n}$$

where $m(n) = m_1(n) + \cdots + m_\omega(n) = n^{O(1)}$ and for $1 \le i \le \omega$, $Q_i \in \{\exists, \forall\}$, $Q_1 = \forall$. Notice that

$$\mathbf{P} = \Sigma_0 = \Pi_0,$$

$$\mathbf{NP} = \Sigma_1, \quad \mathbf{coNP} = \Pi_1.$$

Reminder (cont.)

Similarly a language

$$L = (L_n)_{n>0} \in \Pi_\omega$$

if there exists a language $L' = (L'_n)_{n>0} \in \mathbf{P}$ such that

$$\mathbf{x} \in L_n$$

$$\Updownarrow$$

$$(Q_1 \mathbf{y}^1 \in k^{m_1})(Q_2 \mathbf{y}^2 \in k^{m_2}) \cdots (Q_\omega \mathbf{y}^\omega \in k^{m_\omega})$$
$$(\mathbf{y}^1, \ldots, \mathbf{y}^\omega, \mathbf{x}) \in L'_{m+n}$$

where $m(n) = m_1(n) + \cdots + m_\omega(n) = n^{O(1)}$ and for $1 \leq i \leq \omega$, $Q_i \in \{\exists, \forall\}$, $Q_1 = \forall$. Notice that

$$\mathbf{P} = \Sigma_0 = \Pi_0,$$

$$\mathbf{NP} = \Sigma_1, \quad \mathbf{coNP} = \Pi_1.$$

Reminder (cont.)

Similarly a language

$$L = (L_n)_{n>0} \in \Pi_\omega$$

if there exists a language $L' = (L'_n)_{n>0} \in \mathbf{P}$ such that

$$\mathbf{x} \in L_n$$
$$\Updownarrow$$
$$(Q_1 \mathbf{y}^1 \in k^{m_1})(Q_2 \mathbf{y}^2 \in k^{m_2}) \cdots (Q_\omega \mathbf{y}^\omega \in k^{m_\omega})$$
$$(\mathbf{y}^1, \ldots, \mathbf{y}^\omega, \mathbf{x}) \in L'_{m+n}$$

where $m(n) = m_1(n) + \cdots + m_\omega(n) = n^{O(1)}$ and for $1 \leq i \leq \omega$, $Q_i \in \{\exists, \forall\}$, $Q_1 = \forall$. Notice that

$$\mathbf{P} = \Sigma_0 = \Pi_0,$$

$$\mathbf{NP} = \Sigma_1, \quad \mathbf{coNP} = \Pi_1.$$

Reminder (cont.)

Similarly a language

$$L = (L_n)_{n>0} \in \Pi_\omega$$

if there exists a language $L' = (L'_n)_{n>0} \in \mathbf{P}$ such that

$$\mathbf{x} \in L_n$$
$$\Updownarrow$$
$$(Q_1\mathbf{y}^1 \in k^{m_1})(Q_2\mathbf{y}^2 \in k^{m_2})\cdots(Q_\omega\mathbf{y}^\omega \in k^{m_\omega})$$
$$(\mathbf{y}^1,\ldots,\mathbf{y}^\omega,\mathbf{x}) \in L'_{m+n}$$

where $m(n) = m_1(n) + \cdots + m_\omega(n) = n^{O(1)}$ and for $1 \le i \le \omega$, $Q_i \in \{\exists,\forall\}$, $Q_1 = \forall$. Notice that

$$\mathbf{P} = \Sigma_0 = \Pi_0,$$

$$\mathbf{NP} = \Sigma_1, \quad \mathbf{coNP} = \Pi_1.$$

The polynomial time hierarchy

- Also, notice the inclusions

$$\Sigma_i \subset \Pi_{i+1}, \Sigma_i \subset \Sigma_{i+1}$$
$$\Pi_i \subset \Sigma_{i+1}, \Pi_i \subset \Pi_{i+1}$$

- The *polynomial time hierarchy* is defined to be

$$\mathbf{PH} \stackrel{\text{def}}{=} \bigcup_{\omega \geq 0} (\Sigma_\omega \cup \Pi_\omega) = \bigcup_{\omega \geq 0} \Sigma_\omega = \bigcup_{\omega \geq 0} \Pi_\omega.$$

- *Central problem of CS is to prove that* **PH** *is a proper hierarchy (as is widely believed), and in particular to prove* $\mathbf{P} \neq \mathbf{NP}$.

The polynomial time hierarchy

- Also, notice the inclusions

$$\Sigma_i \subset \Pi_{i+1}, \Sigma_i \subset \Sigma_{i+1}$$
$$\Pi_i \subset \Sigma_{i+1}, \Pi_i \subset \Pi_{i+1}$$

- The *polynomial time hierarchy* is defined to be

$$\mathbf{PH} \stackrel{\text{def}}{=} \bigcup_{\omega \geq 0} (\Sigma_\omega \cup \Pi_\omega) = \bigcup_{\omega \geq 0} \Sigma_\omega = \bigcup_{\omega \geq 0} \Pi_\omega.$$

- *Central problem of CS is to prove that* **PH** *is a proper hierarchy (as is widely believed), and in particular to prove* **P** $\neq$ **NP**.

The polynomial time hierarchy

- Also, notice the inclusions

$$\Sigma_i \subset \Pi_{i+1}, \Sigma_i \subset \Sigma_{i+1}$$
$$\Pi_i \subset \Sigma_{i+1}, \Pi_i \subset \Pi_{i+1}$$

- The *polynomial time hierarchy* is defined to be

$$\mathbf{PH} \stackrel{\text{def}}{=} \bigcup_{\omega \geq 0} (\Sigma_\omega \cup \Pi_\omega) = \bigcup_{\omega \geq 0} \Sigma_\omega = \bigcup_{\omega \geq 0} \Pi_\omega.$$

- *Central problem of CS is to prove that* **PH** *is a proper hierarchy (as is widely believed), and in particular to prove* $\mathbf{P} \neq \mathbf{NP}$.

The Class $\#\mathbf{P}$

- In order to develop an "algebraic" version of complexity theory Valiant introduced certain complexity classes of *functions*;
- A sequence of functions

$$(f_n : k^n \to \mathbb{N})_{n>0}$$

is said to be in the class $\#\mathbf{P}$ if there exists $L = (L_n)_{n>0} \in \mathbf{P}$ such that for $\mathbf{x} \in k^n$

$$f_n(\mathbf{x}) = \mathrm{card}(L_{m+n,\mathbf{x}}), \quad m = n^{O(1)},$$

where $L_{m+n,\mathbf{x}}$ is the fibre $\pi^{-1}(\mathbf{x}) \cap L_{m+n}$, and $\pi : k^{m+n} \to k^n$ the projection map on the last n co-ordinates.

The Class $\#\mathbf{P}$

- In order to develop an "algebraic" version of complexity theory Valiant introduced certain complexity classes of *functions*;
- A sequence of functions

$$(f_n : k^n \to \mathbb{N})_{n>0}$$

is said to be in the class $\#\mathbf{P}$ if there exists $L = (L_n)_{n>0} \in \mathbf{P}$ such that for $\mathbf{x} \in k^n$

$$f_n(\mathbf{x}) = \mathrm{card}(L_{m+n,\mathbf{x}}), \quad m = n^{O(1)},$$

where $L_{m+n,\mathbf{x}}$ is the fibre $\pi^{-1}(\mathbf{x}) \cap L_{m+n}$, and $\pi : k^{m+n} \to k^n$ the projection map on the last n co-ordinates.

The Class $\#\mathbf{P}$

- In order to develop an "algebraic" version of complexity theory Valiant introduced certain complexity classes of *functions*;
- A sequence of functions

$$(f_n : k^n \to \mathbb{N})_{n>0}$$

is said to be in the class $\#\mathbf{P}$ if there exists $L = (L_n)_{n>0} \in \mathbf{P}$ such that for $\mathbf{x} \in k^n$

$$f_n(\mathbf{x}) = \mathrm{card}(L_{m+n,\mathbf{x}}), \quad m = n^{O(1)},$$

where $L_{m+n,\mathbf{x}}$ is the fibre $\pi^{-1}(\mathbf{x}) \cap L_{m+n}$, and $\pi : k^{m+n} \to k^n$ the projection map on the last n co-ordinates.

The Class $\#\mathbf{P}$

- In order to develop an "algebraic" version of complexity theory Valiant introduced certain complexity classes of *functions*;
- A sequence of functions

$$(f_n : k^n \to \mathbb{N})_{n>0}$$

is said to be in the class $\#\mathbf{P}$ if there exists $L = (L_n)_{n>0} \in \mathbf{P}$ such that for $\mathbf{x} \in k^n$

$$f_n(\mathbf{x}) = \mathrm{card}(L_{m+n,\mathbf{x}}), \quad m = n^{O(1)},$$

where $L_{m+n,\mathbf{x}}$ is the fibre $\pi^{-1}(\mathbf{x}) \cap L_{m+n}$, and $\pi : k^{m+n} \to k^n$ the projection map on the last n co-ordinates.

The Class $\#\mathbf{P}$

- In order to develop an "algebraic" version of complexity theory Valiant introduced certain complexity classes of *functions*;
- A sequence of functions

$$(f_n : k^n \to \mathbb{N})_{n>0}$$

is said to be in the class $\#\mathbf{P}$ if there exists $L = (L_n)_{n>0} \in \mathbf{P}$ such that for $\mathbf{x} \in k^n$

$$f_n(\mathbf{x}) = \mathrm{card}(L_{m+n,\mathbf{x}}), \quad m = n^{O(1)},$$

where $L_{m+n,\mathbf{x}}$ is the fibre $\pi^{-1}(\mathbf{x}) \cap L_{m+n}$, and $\pi : k^{m+n} \to k^n$ the projection map on the last n co-ordinates.

Toda's Theorem

Toda's theorem is a seminal result in discrete complexity theory and gives the following inclusion.

Theorem (Toda (1989))

$$\mathbf{PH} \subset \mathbf{P}^{\#\mathbf{P}}$$

"illustrates the power of counting"

Toda's Theorem

Toda's theorem is a seminal result in discrete complexity theory and gives the following inclusion.

Theorem (Toda (1989))

$$\mathbf{PH} \subset \mathbf{P}^{\#\mathbf{P}}$$

"illustrates the power of counting"

Toda's Theorem

Toda's theorem is a seminal result in discrete complexity theory and gives the following inclusion.

Theorem (Toda (1989))

$$\mathbf{PH} \subset \mathbf{P}^{\#\mathbf{P}}$$

"illustrates the power of counting"

Toda's Theorem

Toda's theorem is a seminal result in discrete complexity theory and gives the following inclusion.

Theorem (Toda (1989))

$$\mathbf{PH} \subset \mathbf{P}^{\#\mathbf{P}}$$

"illustrates the power of counting"

Blum-Shub-Smale model

- Generalized TM where k is allowed to be any ring (we restrict ourselves to the cases $k = \mathbb{C}$ or $\mathbb{R}$).
- Setting $k = \mathbb{Z}/2\mathbb{Z}$ (or any finite field) recovers the classical complexity classes.
- Informally, such a TM should be thought of as a program that accepts as input $\mathbf{x} \in k^n$, and at each step
  1. either makes a ring computation $z_i \leftarrow z_j * z_\ell$;
  2. or branches according to a test $z_j\{=, \neq\}0$ in case $k = \mathbb{C}$, or the test $z_j\{>, <, =\}0$ in case $k = \mathbb{R}$;
  3. or accepts/rejects.
- A B-S-S machine accepts for every n a subset $S_n \subset k^n$.

Blum-Shub-Smale model

- Generalized TM where k is allowed to be any ring (we restrict ourselves to the cases $k = \mathbb{C}$ or $\mathbb{R}$).
- Setting $k = \mathbb{Z}/2\mathbb{Z}$ (or any finite field) recovers the classical complexity classes.
- Informally, such a TM should be thought of as a program that accepts as input $\mathbf{x} \in k^n$, and at each step
  1. either makes a ring computation $z_i \leftarrow z_j * z_\ell$;
  2. or branches according to a test $z_j\{=,\neq\}0$ in case $k = \mathbb{C}$, or the test $z_j\{>,<,=\}0$ in case $k = \mathbb{R}$;
  3. or accepts/rejects.
- A B-S-S machine accepts for every n a subset $S_n \subset k^n$.

Blum-Shub-Smale model

- Generalized TM where k is allowed to be any ring (we restrict ourselves to the cases $k = \mathbb{C}$ or $\mathbb{R}$).
- Setting $k = \mathbb{Z}/2\mathbb{Z}$ (or any finite field) recovers the classical complexity classes.
- Informally, such a TM should be thought of as a program that accepts as input $\mathbf{x} \in k^n$, and at each step
  1. either makes a ring computation $z_i \leftarrow z_j * z_\ell$;
  2. or branches according to a test $z_j\{=, \neq\}0$ in case $k = \mathbb{C}$, or the test $z_j\{>, <, =\}0$ in case $k = \mathbb{R}$;
  3. or accepts/rejects.
- A B-S-S machine accepts for every n a subset $S_n \subset k^n$.
  1. In case $k = \mathbb{C}$, each S_n is a constructible set.
  2. In case $k = \mathbb{R}$, each S_n is a semi-algebraic set.

Blum-Shub-Smale model

- Generalized TM where k is allowed to be any ring (we restrict ourselves to the cases $k = \mathbb{C}$ or $\mathbb{R}$).
- Setting $k = \mathbb{Z}/2\mathbb{Z}$ (or any finite field) recovers the classical complexity classes.
- Informally, such a TM should be thought of as a program that accepts as input $\mathbf{x} \in k^n$, and at each step
  1. either makes a ring computation $z_i \leftarrow z_j * z_\ell$;
  2. or branches according to a test $z_j\{=,\neq\}0$ in case $k = \mathbb{C}$, or the test $z_j\{>,<,=\}0$ in case $k = \mathbb{R}$;
  3. or accepts/rejects.
- A B-S-S machine accepts for every n a subset $S_n \subset k^n$.

Blum-Shub-Smale model

- Generalized TM where k is allowed to be any ring (we restrict ourselves to the cases $k = \mathbb{C}$ or $\mathbb{R}$).
- Setting $k = \mathbb{Z}/2\mathbb{Z}$ (or any finite field) recovers the classical complexity classes.
- Informally, such a TM should be thought of as a program that accepts as input $\mathbf{x} \in k^n$, and at each step
  1. either makes a ring computation $z_i \leftarrow z_j * z_\ell$;
  2. or branches according to a test $z_j\{=, \neq\}0$ in case $k = \mathbb{C}$, or the test $z_j\{>, <, =\}0$ in case $k = \mathbb{R}$;
  3. or accepts/rejects.
- A B-S-S machine accepts for every n a subset $S_n \subset k^n$.
  1. In case $k = \mathbb{C}$, each S_n is a constructible subset of $\mathbb{C}^n$.
  2. In case $k = \mathbb{R}$, each S_n is a semi-algebraic subset of $\mathbb{R}^n$.

Blum-Shub-Smale model

- Generalized TM where k is allowed to be any ring (we restrict ourselves to the cases $k = \mathbb{C}$ or $\mathbb{R}$).
- Setting $k = \mathbb{Z}/2\mathbb{Z}$ (or any finite field) recovers the classical complexity classes.
- Informally, such a TM should be thought of as a program that accepts as input $\mathbf{x} \in k^n$, and at each step
  1. either makes a ring computation $z_i \leftarrow z_j * z_\ell$;
  2. or branches according to a test $z_j\{=, \neq\}0$ in case $k = \mathbb{C}$, or the test $z_j\{>, <, =\}0$ in case $k = \mathbb{R}$;
  3. or accepts/rejects.
- A B-S-S machine accepts for every n a subset $S_n \subset k^n$.
  1. In case $k = \mathbb{C}$, each S_n is a constructible subset of $\mathbb{C}^n$.
  2. In case $k = \mathbb{R}$, each S_n is a semi-algebraic subset of $\mathbb{R}^n$.

Blum-Shub-Smale model

- Generalized TM where k is allowed to be any ring (we restrict ourselves to the cases $k = \mathbb{C}$ or $\mathbb{R}$).
- Setting $k = \mathbb{Z}/2\mathbb{Z}$ (or any finite field) recovers the classical complexity classes.
- Informally, such a TM should be thought of as a program that accepts as input $\mathbf{x} \in k^n$, and at each step
  1. either makes a ring computation $z_i \leftarrow z_j * z_\ell$;
  2. or branches according to a test $z_j\{=, \neq\}0$ in case $k = \mathbb{C}$, or the test $z_j\{>, <, =\}0$ in case $k = \mathbb{R}$;
  3. or accepts/rejects.
- A B-S-S machine accepts for every n a subset $S_n \subset k^n$.
  1. In case $k = \mathbb{C}$, each S_n is a *constructible* subset of $\mathbb{C}^n$,
  2. in case $k = \mathbb{R}$, each S_n is a *semi-algebraic* subset of $\mathbb{R}^n$.

Blum-Shub-Smale model

- Generalized TM where k is allowed to be any ring (we restrict ourselves to the cases $k = \mathbb{C}$ or $\mathbb{R}$).
- Setting $k = \mathbb{Z}/2\mathbb{Z}$ (or any finite field) recovers the classical complexity classes.
- Informally, such a TM should be thought of as a program that accepts as input $\mathbf{x} \in k^n$, and at each step
  1. either makes a ring computation $z_i \leftarrow z_j * z_\ell$;
  2. or branches according to a test $z_j\{=,\neq\}0$ in case $k = \mathbb{C}$, or the test $z_j\{>,<,=\}0$ in case $k = \mathbb{R}$;
  3. or accepts/rejects.
- A B-S-S machine accepts for every n a subset $S_n \subset k^n$.
  1. In case $k = \mathbb{C}$, each S_n is a *constructible* subset of $\mathbb{C}^n$,
  2. in case $k = \mathbb{R}$, each S_n is a *semi-algebraic* subset of $\mathbb{R}^n$.

Blum-Shub-Smale model

- Generalized TM where k is allowed to be any ring (we restrict ourselves to the cases $k = \mathbb{C}$ or $\mathbb{R}$).
- Setting $k = \mathbb{Z}/2\mathbb{Z}$ (or any finite field) recovers the classical complexity classes.
- Informally, such a TM should be thought of as a program that accepts as input $\mathbf{x} \in k^n$, and at each step
  1. either makes a ring computation $z_i \leftarrow z_j * z_\ell$;
  2. or branches according to a test $z_j\{=, \neq\}0$ in case $k = \mathbb{C}$, or the test $z_j\{>, <, =\}0$ in case $k = \mathbb{R}$;
  3. or accepts/rejects.
- A B-S-S machine accepts for every n a subset $S_n \subset k^n$.
  1. In case $k = \mathbb{C}$, each S_n is a *constructible* subset of $\mathbb{C}^n$,
  2. in case $k = \mathbb{R}$, each S_n is a *semi-algebraic* subset of $\mathbb{R}^n$.

Blum-Shub-Smale model

- Generalized TM where k is allowed to be any ring (we restrict ourselves to the cases $k = \mathbb{C}$ or $\mathbb{R}$).
- Setting $k = \mathbb{Z}/2\mathbb{Z}$ (or any finite field) recovers the classical complexity classes.
- Informally, such a TM should be thought of as a program that accepts as input $\mathbf{x} \in k^n$, and at each step
  1. either makes a ring computation $z_i \leftarrow z_j * z_\ell$;
  2. or branches according to a test $z_j\{=, \neq\}0$ in case $k = \mathbb{C}$, or the test $z_j\{>, <, =\}0$ in case $k = \mathbb{R}$;
  3. or accepts/rejects.
- A B-S-S machine accepts for every n a subset $S_n \subset k^n$.
  1. In case $k = \mathbb{C}$, each S_n is a *constructible* subset of $\mathbb{C}^n$,
  2. in case $k = \mathbb{R}$, each S_n is a *semi-algebraic* subset of $\mathbb{R}^n$.

Complexity Classes

- Complexity classes $\mathbf{P}_k, \mathbf{NP}_k, \mathbf{coNP}_k$ and more generally $\mathbf{PH}_k$ are defined as before (for $k = \mathbb{C}, \mathbb{R}$).
- B-S-S developed a theory of **NP**-completeness.
- In case, $k = \mathbb{C}$ the problem of determining if a system of $n+1$ polynomial equations in n variables has a common zero in $\mathbb{C}^n$ is $\mathbf{NP}_{\mathbb{C}}$-complete.
- In case, $k = \mathbb{R}$ the problem of determining if a quartic polynomial in n variables has a common zero in $\mathbb{R}^n$ is $\mathbf{NP}_{\mathbb{R}}$-complete.
- It is unknown if $\mathbf{P}_{\mathbb{C}} = \mathbf{NP}_{\mathbb{C}}$ (respectively, $\mathbf{P}_{\mathbb{R}} = \mathbf{NP}_{\mathbb{R}}$) just as in the discrete case.

Complexity Classes

- Complexity classes $\mathbf{P}_k, \mathbf{NP}_k, \mathbf{coNP}_k$ and more generally $\mathbf{PH}_k$ are defined as before (for $k = \mathbb{C}, \mathbb{R}$).
- B-S-S developed a theory of **NP**-completeness.
- In case, $k = \mathbb{C}$ the problem of determining if a system of $n + 1$ polynomial equations in n variables has a common zero in $\mathbb{C}^n$ is $\mathbf{NP}_{\mathbb{C}}$-complete.
- In case, $k = \mathbb{R}$ the problem of determining if a quartic polynomial in n variables has a common zero in $\mathbb{R}^n$ is $\mathbf{NP}_{\mathbb{R}}$-complete.
- It is unknown if $\mathbf{P}_{\mathbb{C}} = \mathbf{NP}_{\mathbb{C}}$ (respectively, $\mathbf{P}_{\mathbb{R}} = \mathbf{NP}_{\mathbb{R}}$) just as in the discrete case.

Complexity Classes

- Complexity classes $\mathbf{P}_k$, $\mathbf{NP}_k$, $\mathbf{coNP}_k$ and more generally $\mathbf{PH}_k$ are defined as before (for $k = \mathbb{C}, \mathbb{R}$).
- B-S-S developed a theory of **NP**-completeness.
- In case, $k = \mathbb{C}$ the problem of determining if a system of $n+1$ polynomial equations in n variables has a common zero in $\mathbb{C}^n$ is $\mathbf{NP}_{\mathbb{C}}$-complete.
- In case, $k = \mathbb{R}$ the problem of determining if a quartic polynomial in n variables has a common zero in $\mathbb{R}^n$ is $\mathbf{NP}_{\mathbb{R}}$-complete.
- It is unknown if $\mathbf{P}_{\mathbb{C}} = \mathbf{NP}_{\mathbb{C}}$ (respectively, $\mathbf{P}_{\mathbb{R}} = \mathbf{NP}_{\mathbb{R}}$) just as in the discrete case.

Complexity Classes

- Complexity classes $\mathbf{P}_k, \mathbf{NP}_k, \mathbf{coNP}_k$ and more generally $\mathbf{PH}_k$ are defined as before (for $k = \mathbb{C}, \mathbb{R}$).
- B-S-S developed a theory of **NP**-completeness.
- In case, $k = \mathbb{C}$ the problem of determining if a system of $n+1$ polynomial equations in n variables has a common zero in $\mathbb{C}^n$ is $\mathbf{NP}_{\mathbb{C}}$-complete.
- In case, $k = \mathbb{R}$ the problem of determining if a quartic polynomial in n variables has a common zero in $\mathbb{R}^n$ is $\mathbf{NP}_{\mathbb{R}}$-complete.
- It is unknown if $\mathbf{P}_{\mathbb{C}} = \mathbf{NP}_{\mathbb{C}}$ (respectively, $\mathbf{P}_{\mathbb{R}} = \mathbf{NP}_{\mathbb{R}}$) just as in the discrete case.

Complexity Classes

- Complexity classes $\mathbf{P}_k, \mathbf{NP}_k, \mathbf{coNP}_k$ and more generally $\mathbf{PH}_k$ are defined as before (for $k = \mathbb{C}, \mathbb{R}$).
- B-S-S developed a theory of **NP**-completeness.
- In case, $k = \mathbb{C}$ the problem of determining if a system of $n+1$ polynomial equations in n variables has a common zero in $\mathbb{C}^n$ is $\mathbf{NP}_{\mathbb{C}}$-complete.
- In case, $k = \mathbb{R}$ the problem of determining if a quartic polynomial in n variables has a common zero in $\mathbb{R}^n$ is $\mathbf{NP}_{\mathbb{R}}$-complete.
- It is unknown if $\mathbf{P}_{\mathbb{C}} = \mathbf{NP}_{\mathbb{C}}$ (respectively, $\mathbf{P}_{\mathbb{R}} = \mathbf{NP}_{\mathbb{R}}$) just as in the discrete case.

Two classes of problems

The most important algorithmic problems studied in this area fall into two broad sub-classes:

1. the problem of quantifier elimination, and its special cases such as *deciding* a sentence in the first order theory of reals/complex numbers, or deciding emptiness of semi-algebraic/constructible sets.
2. the problem of *computing* topological invariants of semi-algebraic/constructible sets, such as the number of connected components, Euler-Poincaré characteristic, and more generally all the Betti numbers of semi-algebraic/constructible sets.

Two classes of problems

The most important algorithmic problems studied in this area fall into two broad sub-classes:

1. the problem of quantifier elimination, and its special cases such as *deciding* a sentence in the first order theory of reals/complex numbers, or deciding emptiness of semi-algebraic/constructible sets.
2. the problem of *computing* topological invariants of semi-algebraic/constructible sets, such as the number of connected components, Euler-Poincaré characteristic, and more generally all the Betti numbers of semi-algebraic/constructible sets.

Two classes of problems

The most important algorithmic problems studied in this area fall into two broad sub-classes:

1. the problem of quantifier elimination, and its special cases such as *deciding* a sentence in the first order theory of reals/complex numbers, or deciding emptiness of semi-algebraic/constructible sets.
2. the problem of *computing* topological invariants of semi-algebraic/constructible sets, such as the number of connected components, Euler-Poincaré characteristic, and more generally all the Betti numbers of semi-algebraic/constructible sets.

Analogy with Toda's Theorem

- The classes **PH** and $\#$**P** appearing in the two sides of the inclusion in Toda's Theorem can be identified with the two broad classes of problems in algorithmic algebraic/semi-algebraic geometry;
- the class **PH** with the problem of deciding sentences with a fixed number of quantifier alternations;
- the class $\#$**P** with the problem of computing topological invariants of semi-algebraic/constructible sets, namely their Betti numbers, which generalizes the notion of cardinality for finite sets;
- it is thus quite natural to seek a real as well as complex analogue of Toda's theorem.

Analogy with Toda's Theorem

- The classes **PH** and $\#$**P** appearing in the two sides of the inclusion in Toda's Theorem can be identified with the two broad classes of problems in algorithmic algebraic/semi-algebraic geometry;
- the class **PH** with the problem of deciding sentences with a fixed number of quantifier alternations;
- the class $\#$**P** with the problem of computing topological invariants of semi-algebraic/constructible sets, namely their Betti numbers, which generalizes the notion of cardinality for finite sets;
- it is thus quite natural to seek a real as well as complex analogue of Toda's theorem.

Analogy with Toda's Theorem

- The classes **PH** and $\#$**P** appearing in the two sides of the inclusion in Toda's Theorem can be identified with the two broad classes of problems in algorithmic algebraic/semi-algebraic geometry;
- the class **PH** with the problem of deciding sentences with a fixed number of quantifier alternations;
- the class $\#$**P** with the problem of computing topological invariants of semi-algebraic/constructible sets, namely their Betti numbers, which generalizes the notion of cardinality for finite sets;
- it is thus quite natural to seek a real as well as complex analogue of Toda's theorem.

Analogy with Toda's Theorem

- The classes **PH** and $\#$**P** appearing in the two sides of the inclusion in Toda's Theorem can be identified with the two broad classes of problems in algorithmic algebraic/semi-algebraic geometry;
- the class **PH** with the problem of deciding sentences with a fixed number of quantifier alternations;
- the class $\#$**P** with the problem of computing topological invariants of semi-algebraic/constructible sets, namely their Betti numbers, which generalizes the notion of cardinality for finite sets;
- it is thus quite natural to seek a real as well as complex analogue of Toda's theorem.

Real/complex analogue of $\#\mathbf{P}$

- In order to define real analogues of counting complexity classes of discrete complexity theory, it is necessary to identify the proper notion of "counting" in the context of algebraic/semi-algebraic geometry.
- Counting complexity classes over the reals/complex numbers have been defined previously by Meer (2000) and studied extensively by other authors Burgisser, Cucker et al (2006). These authors used a straightforward generalization to semi-algebraic/constructible sets of counting in the case of finite sets; namely

$$\begin{aligned} f(S) &= \mathrm{card}(S), \text{ if } \mathrm{card}(S) < \infty; \\ &= \infty \text{ otherwise.} \end{aligned}$$

Real/complex analogue of $\#\mathbf{P}$

- In order to define real analogues of counting complexity classes of discrete complexity theory, it is necessary to identify the proper notion of "counting" in the context of algebraic/semi-algebraic geometry.
- Counting complexity classes over the reals/complex numbers have been defined previously by Meer (2000) and studied extensively by other authors Burgisser, Cucker et al (2006). These authors used a straightforward generalization to semi-algebraic/constructible sets of counting in the case of finite sets; namely

$$\begin{aligned} f(S) &= \mathrm{card}(S), \text{ if } \mathrm{card}(S) < \infty; \\ &= \infty \text{ otherwise.} \end{aligned}$$

An alternative definition

- In our view this is not fully satisfactory, since the count gives no information when the set is infinite, and *most interesting semi-algebraic/constructible sets are infinite*.
- If one thinks of "counting" a semi-algebraic/constructible set $S \subset \mathbb{R}^k$ or $\mathbb{C}^k$ as computing certain discrete invariants, then a natural mathematical candidate is its sequence of Betti numbers, $b_0(S), \ldots, b_{k-1}(S)$, or more succinctly
- the ***Poincaré polynomial*** of S, namely

$$P_S(T) \stackrel{\text{def}}{=} \sum_{i \geq 0} b_i(S)\, T^i.$$

- In case $\mathrm{card}(S) < \infty$, we have that $b_0(S) = P_S(0) = \mathrm{card}(S)$.

An alternative definition

- In our view this is not fully satisfactory, since the count gives no information when the set is infinite, and *most interesting semi-algebraic/constructible sets are infinite*.
- If one thinks of "counting" a semi-algebraic/constructible set $S \subset \mathbb{R}^k$ or $\mathbb{C}^k$ as computing certain discrete invariants, then a natural mathematical candidate is its sequence of Betti numbers, $b_0(S), \ldots, b_{k-1}(S)$, or more succinctly
- the ***Poincaré polynomial*** of S, namely

$$P_S(T) \stackrel{\text{def}}{=} \sum_{i \geq 0} b_i(S)\, T^i.$$

- In case $\mathrm{card}(S) < \infty$, we have that $b_0(S) = P_S(0) = \mathrm{card}(S)$.

An alternative definition

- In our view this is not fully satisfactory, since the count gives no information when the set is infinite, and *most interesting semi-algebraic/constructible sets are infinite*.
- If one thinks of "counting" a semi-algebraic/constructible set $S \subset \mathbb{R}^k$ or $\mathbb{C}^k$ as computing certain discrete invariants, then a natural mathematical candidate is its sequence of Betti numbers, $b_0(S), \ldots, b_{k-1}(S)$, or more succinctly
- the ***Poincaré polynomial*** of S, namely

$$P_S(T) \stackrel{\text{def}}{=} \sum_{i \geq 0} b_i(S)\, T^i.$$

- In case $\mathrm{card}(S) < \infty$, we have that $b_0(S) = P_S(0) = \mathrm{card}(S)$.

An alternative definition

- In our view this is not fully satisfactory, since the count gives no information when the set is infinite, and *most interesting semi-algebraic/constructible sets are infinite*.
- If one thinks of "counting" a semi-algebraic/constructible set $S \subset \mathbb{R}^k$ or $\mathbb{C}^k$ as computing certain discrete invariants, then a natural mathematical candidate is its sequence of Betti numbers, $b_0(S), \ldots, b_{k-1}(S)$, or more succinctly
- the ***Poincaré polynomial*** of S, namely

$$P_S(T) \stackrel{def}{=} \sum_{i \geq 0} b_i(S)\, T^i.$$

- In case $\mathrm{card}(S) < \infty$, we have that $b_0(S) = P_S(0) = \mathrm{card}(S)$.

Definition of $\#\mathbf{P}^{\dagger}_{\mathbb{R}}$

We call a sequence of functions

$$(f_n : \mathbb{R}^n \to \mathbb{Z}[T])_{n>0}$$

to be in class $\#\mathbf{P}^{\dagger}_{\mathbb{R}}$ if there exists $(S_n \subset \mathbb{R}^n)_{n>0} \in \mathbf{P}_{\mathbb{R}}$ such that for $\mathbf{x} \in \mathbb{R}^n$

$$f_n(\mathbf{x}) = P_{S_{m+n,\mathbf{x}}}, \quad m = n^{O(1)},$$

where $S_{m+n,\mathbf{x}} = S_{m+n} \cap \pi^{-1}(\mathbf{x})$ and $\pi : \mathbb{R}^{m+n} \to \mathbb{R}^n$ is the projection on the last n coordinates.
Similar definition over $\mathbb{C}$ as well.

Definition of $\#\mathbf{P}^{\dagger}_{\mathbb{R}}$

We call a sequence of functions

$$(f_n : \mathbb{R}^n \to \mathbb{Z}[T])_{n>0}$$

to be in class $\#\mathbf{P}^{\dagger}_{\mathbb{R}}$ if there exists $(S_n \subset \mathbb{R}^n)_{n>0} \in \mathbf{P}_{\mathbb{R}}$ such that for $\mathbf{x} \in \mathbb{R}^n$

$$f_n(\mathbf{x}) = P_{S_{m+n,\mathbf{x}}}, \quad m = n^{O(1)},$$

where $S_{m+n,\mathbf{x}} = S_{m+n} \cap \pi^{-1}(\mathbf{x})$ and $\pi : \mathbb{R}^{m+n} \to \mathbb{R}^n$ is the projection on the last n coordinates.

Similar definition over $\mathbb{C}$ as well.

Counting and Betti numbers

- The connection between counting points of varieties and their Betti numbers is more direct over fields of positive characteristic via the zeta function.
- The zeta function of a variety defined over $\mathbb{F}_p$ is the exponential generating function of the sequence whose n-th term is the number of points in the variety over $\mathbb{F}_{p^n}$.
- The zeta function depends on the Betti numbers of the variety with respect to a certain (ℓ-adic) co-homology theory.
- Thus, the problems of "counting" varieties and computing their Betti numbers, are connected at a deeper level, and thus our definition of $\#\mathbf{P}^{\dagger}_{\mathbb{R}}$ is not entirely ad hoc.

Counting and Betti numbers

- The connection between counting points of varieties and their Betti numbers is more direct over fields of positive characteristic via the zeta function.
- The zeta function of a variety defined over $\mathbb{F}_p$ is the exponential generating function of the sequence whose n-th term is the number of points in the variety over $\mathbb{F}_{p^n}$.
- The zeta function depends on the Betti numbers of the variety with respect to a certain (ℓ-adic) co-homology theory.
- Thus, the problems of "counting" varieties and computing their Betti numbers, are connected at a deeper level, and thus our definition of $\#\mathbf{P}^{\dagger}_{\mathbb{R}}$ is not entirely ad hoc.

Counting and Betti numbers

- The connection between counting points of varieties and their Betti numbers is more direct over fields of positive characteristic via the zeta function.
- The zeta function of a variety defined over $\mathbb{F}_p$ is the exponential generating function of the sequence whose n-th term is the number of points in the variety over $\mathbb{F}_{p^n}$.
- The zeta function depends on the Betti numbers of the variety with respect to a certain (ℓ-adic) co-homology theory.
- Thus, the problems of "counting" varieties and computing their Betti numbers, are connected at a deeper level, and thus our definition of $\#\mathbf{P}_{\mathbb{R}}^{\dagger}$ is not entirely ad hoc.

Counting and Betti numbers

- The connection between counting points of varieties and their Betti numbers is more direct over fields of positive characteristic via the zeta function.
- The zeta function of a variety defined over $\mathbb{F}_p$ is the exponential generating function of the sequence whose n-th term is the number of points in the variety over $\mathbb{F}_{p^n}$.
- The zeta function depends on the Betti numbers of the variety with respect to a certain (ℓ-adic) co-homology theory.
- Thus, the problems of "counting" varieties and computing their Betti numbers, are connected at a deeper level, and thus our definition of $\#\mathbf{P}_{\mathbb{R}}^{\dagger}$ is not entirely ad hoc.

Real/Complex analogue of Toda's theorem

It is now natural to formulate the following conjectures.

Conjecture

$$\mathbf{PH}_{\mathbb{R}} \subset \mathbf{P}^{\#\mathbf{P}^{\dagger}_{\mathbb{R}}}$$

Conjecture

$$\mathbf{PH}_{\mathbb{C}} \subset \mathbf{P}^{\#\mathbf{P}^{\dagger}_{\mathbb{C}}}$$

For technical reasons we are unable to prove this without a further compactness hypothesis on the left hand-side.

Real/Complex analogue of Toda's theorem

It is now natural to formulate the following conjectures.

Conjecture

$$\mathbf{PH}_{\mathbb{R}} \subset \mathbf{P}^{\#\mathbf{P}^{\dagger}_{\mathbb{R}}}$$

Conjecture

$$\mathbf{PH}_{\mathbb{C}} \subset \mathbf{P}^{\#\mathbf{P}^{\dagger}_{\mathbb{C}}}$$

For technical reasons we are unable to prove this without a further compactness hypothesis on the left hand-side.

Real/Complex analogue of Toda's theorem

It is now natural to formulate the following conjectures.

Conjecture

$$\mathbf{PH}_{\mathbb{R}} \subset \mathbf{P}^{\#\mathbf{P}^{\dagger}_{\mathbb{R}}}$$

Conjecture

$$\mathbf{PH}_{\mathbb{C}} \subset \mathbf{P}^{\#\mathbf{P}^{\dagger}_{\mathbb{C}}}$$

For technical reasons we are unable to prove this without a further compactness hypothesis on the left hand-side.

Real/Complex analogue of Toda's theorem

It is now natural to formulate the following conjectures.

Conjecture

$$\mathbf{PH}_{\mathbb{R}} \subset \mathbf{P}^{\#\mathbf{P}^{\dagger}_{\mathbb{R}}}$$

Conjecture

$$\mathbf{PH}_{\mathbb{C}} \subset \mathbf{P}^{\#\mathbf{P}^{\dagger}_{\mathbb{C}}}$$

For technical reasons we are unable to prove this without a further compactness hypothesis on the left hand-side.

The compact fragment of real polynomial hierarchy

We say that a sequence of semi-algebraic sets

$$(S_n \subset \mathbf{S}^n)_{n>0} \in \Sigma^c_{\mathbb{R},\omega}$$

if there exists another sequence $(S'_n)_{n>0} \in \mathbf{P}_{\mathbb{R}}$ such that each S'_n is compact and

$$x \in S_n$$

if and only if

$$(Q_1 y^1 \in \mathbf{S}^{m_1})(Q_2 y^2 \in \mathbf{S}^{m_2}) \ldots (Q_\omega y^\omega \in \mathbf{S}^{m_\omega})$$
$$(y^1, \ldots, y^\omega, x) \in S'_{m+n}$$

where $m(n) = m_1(n) + \cdots + m_\omega(n) = n^{O(1)}$ and for $1 \leq i \leq \omega$, $Q_i \in \{\exists, \forall\}$, and $Q_j \neq Q_{j+1}, 1 \leq j < \omega$, $Q_1 = \exists$. The compact class $\Pi^c_{\mathbb{R},\omega}$ is defined analogously.

The compact fragment of real polynomial hierarchy

We say that a sequence of semi-algebraic sets

$$(S_n \subset \mathbf{S}^n)_{n>0} \in \Sigma^c_{\mathbb{R},\omega}$$

if there exists another sequence $(S'_n)_{n>0} \in \mathbf{P}_{\mathbb{R}}$ such that each S'_n is compact and

$$x \in S_n$$

$$\text{if and only if}$$

$$(Q_1 y^1 \in \mathbf{S}^{m_1})(Q_2 y^2 \in \mathbf{S}^{m_2}) \ldots (Q_\omega y^\omega \in \mathbf{S}^{m_\omega})$$
$$(y^1, \ldots, y^\omega, x) \in S'_{m+n}$$

where $m(n) = m_1(n) + \cdots + m_\omega(n) = n^{O(1)}$ and for $1 \leq i \leq \omega$, $Q_i \in \{\exists, \forall\}$, and $Q_j \neq Q_{j+1}, 1 \leq j < \omega$, $Q_1 = \exists$. The compact class $\Pi^c_{\mathbb{R},\omega}$ is defined analogously.

The compact fragment of real polynomial hierarchy

We say that a sequence of semi-algebraic sets

$$(S_n \subset \mathbf{S}^n)_{n>0} \in \Sigma^c_{\mathbb{R},\omega}$$

if there exists another sequence $(S'_n)_{n>0} \in \mathbf{P}_{\mathbb{R}}$ such that each S'_n is compact and

$$x \in S_n$$

$$\text{if and only if}$$

$$(Q_1 y^1 \in \mathbf{S}^{m_1})(Q_2 y^2 \in \mathbf{S}^{m_2}) \ldots (Q_\omega y^\omega \in \mathbf{S}^{m_\omega})$$
$$(y^1, \ldots, y^\omega, x) \in S'_{m+n}$$

where $m(n) = m_1(n) + \cdots + m_\omega(n) = n^{O(1)}$ and for $1 \leq i \leq \omega$, $Q_i \in \{\exists, \forall\}$, and $Q_j \neq Q_{j+1}, 1 \leq j < \omega$, $Q_1 = \exists$. The compact class $\Pi^c_{\mathbb{R},\omega}$ is defined analogously.

The compact fragment of real polynomial hierarchy

We say that a sequence of semi-algebraic sets

$$(S_n \subset \mathbf{S}^n)_{n>0} \in \Sigma^c_{\mathbb{R},\omega}$$

if there exists another sequence $(S'_n)_{n>0} \in \mathbf{P}_{\mathbb{R}}$ such that each S'_n is compact and

$$x \in S_n$$

$$\text{if and only if}$$

$$(Q_1 y^1 \in \mathbf{S}^{m_1})(Q_2 y^2 \in \mathbf{S}^{m_2}) \dots (Q_\omega y^\omega \in \mathbf{S}^{m_\omega})$$

$$(y^1, \dots, y^\omega, x) \in S'_{m+n}$$

where $m(n) = m_1(n) + \cdots + m_\omega(n) = n^{O(1)}$ and for $1 \leq i \leq \omega$, $Q_i \in \{\exists, \forall\}$, and $Q_j \neq Q_{j+1}, 1 \leq j < \omega$, $Q_1 = \exists$. The compact class $\Pi^c_{\mathbb{R},\omega}$ is defined analogously.

The compact real polynomial hierarchy (cont.)

We define

$$\mathbf{PH}^c_{\mathbb{R}} \stackrel{\text{def}}{=} \bigcup_{\omega \geq 0} (\Sigma^c_{\mathbb{R},\omega} \cup \Pi^c_{\mathbb{R},\omega}) = \bigcup_{\omega \geq 0} \Sigma^c_{\mathbb{R},\omega} = \bigcup_{\omega \geq 0} {}^c_{\mathbb{R},\omega}.$$

Notice that the semi-algebraic sets belonging to any language in $\mathbf{PH}^c_{\mathbb{R}}$ are all semi-algebraic compact (in fact closed semi-algebraic subsets of spheres). Also, notice the inclusion

$$\mathbf{PH}^c_{\mathbb{R}} \subset \mathbf{PH}_{\mathbb{R}}.$$

The compact real polynomial hierarchy (cont.)

We define

$$\mathbf{PH}^{c}_{\mathbb{R}} \stackrel{\text{def}}{=} \bigcup_{\omega \geq 0} (\Sigma^{c}_{\mathbb{R},\omega} \cup \Pi^{c}_{\mathbb{R},\omega}) = \bigcup_{\omega \geq 0} \Sigma^{c}_{\mathbb{R},\omega} = \bigcup_{\omega \geq 0} {}^{c}_{\mathbb{R},\omega}.$$

Notice that the semi-algebraic sets belonging to any language in $\mathbf{PH}^{c}_{\mathbb{R}}$ are all semi-algebraic compact (in fact closed semi-algebraic subsets of spheres). Also, notice the inclusion

$$\mathbf{PH}^{c}_{\mathbb{R}} \subset \mathbf{PH}_{\mathbb{R}}.$$

Main theorem

Theorem (B-Zell,2008)

$$\mathbf{PH}_{\mathbb{R}}^{c} \subset \mathbf{P}_{\mathbb{R}}^{\#\mathbf{P}_{\mathbb{R}}^{\dagger}}.$$

Theorem (B.,2009)

$$\mathbf{PH}_{\mathbb{C}}^{c} \subset \mathbf{P}_{\mathbb{C}}^{\#\mathbf{P}_{\mathbb{C}}^{\dagger}}.$$

Main theorem

Theorem (B-Zell,2008)

$$\mathbf{PH}_{\mathbb{R}}^{c} \subset \mathbf{P}_{\mathbb{R}}^{\#\mathbf{P}_{\mathbb{R}}^{\dagger}}.$$

Theorem (B.,2009)

$$\mathbf{PH}_{\mathbb{C}}^{c} \subset \mathbf{P}_{\mathbb{C}}^{\#\mathbf{P}_{\mathbb{C}}^{\dagger}}.$$

Outline

Summary of the Main Idea

- Our main tool is a topological construction which given a semi-algebraic set $S \subset \mathbb{R}^{m+n}$, $p \geq 0$, and $\pi_{\mathbf{Y}} : \mathbb{R}^{m+n} \to \mathbb{R}^n$ denoting the projection along (say) the $\mathbf{Y}$-co-ordinates, constructs *efficiently* a semi-algebraic set, $D^p_{\mathbf{Y}}(S)$, such that
$$b_i(\pi_{\mathbf{Y}}(S)) = b_i(D^p_{\mathbf{Y}}(S)), 0 \leq i < p.$$
- Notice that even if there exists an efficient (i.e. polynomial time) algorithm for checking membership in S, the same need not be true for the image $\pi_{\mathbf{Y}}(S)$.
- A second topological ingredient is *Alexander-Lefschetz duality* which relates the Betti numbers of a compact subset K of the sphere $\mathbf{S}^n$ with those of $\mathbf{S}^n - K$.

Summary of the Main Idea

- Our main tool is a topological construction which given a semi-algebraic set $S \subset \mathbb{R}^{m+n}$, $p \geq 0$, and $\pi_{\mathbf{Y}} : \mathbb{R}^{m+n} \to \mathbb{R}^n$ denoting the projection along (say) the $\mathbf{Y}$-co-ordinates, constructs *efficiently* a semi-algebraic set, $D^p_{\mathbf{Y}}(S)$, such that

$$b_i(\pi_{\mathbf{Y}}(S)) = b_i(D^p_{\mathbf{Y}}(S)), 0 \leq i < p.$$

- Notice that even if there exists an efficient (i.e. polynomial time) algorithm for checking membership in S, the same need not be true for the image $\pi_{\mathbf{Y}}(S)$.
- A second topological ingredient is *Alexander-Lefschetz duality* which relates the Betti numbers of a compact subset K of the sphere $\mathbf{S}^n$ with those of $\mathbf{S}^n - K$.

Summary of the Main Idea

- Our main tool is a topological construction which given a semi-algebraic set $S \subset \mathbb{R}^{m+n}$, $p \geq 0$, and $\pi_{\mathbf{Y}} : \mathbb{R}^{m+n} \to \mathbb{R}^n$ denoting the projection along (say) the $\mathbf{Y}$-co-ordinates, constructs *efficiently* a semi-algebraic set, $D^p_{\mathbf{Y}}(S)$, such that
$$b_i(\pi_{\mathbf{Y}}(S)) = b_i(D^p_{\mathbf{Y}}(S)), 0 \leq i < p.$$
- Notice that even if there exists an efficient (i.e. polynomial time) algorithm for checking membership in S, the same need not be true for the image $\pi_{\mathbf{Y}}(S)$.
- A second topological ingredient is *Alexander-Lefschetz duality* which relates the Betti numbers of a compact subset K of the sphere $\mathbf{S}^n$ with those of $\mathbf{S}^n - K$.

Outline

Complex join fibered over a map

Let $A \subset \mathbb{P}_{\mathbb{C}}^{k} \times \mathbb{P}_{\mathbb{C}}^{\ell}$ be a constructible set defined by a first-order multi-homogeneous formula,

$$\phi(X_0, \ldots, X_k; Y_0, \ldots, Y_\ell)$$

and let $\pi_{\mathbf{Y}} : \mathbb{P}_{\mathbb{C}}^{k} \times \mathbb{P}_{\mathbb{C}}^{\ell} \to \mathbb{P}_{\mathbb{C}}^{k}$ be the projection along the $\mathbf{Y}$-co-ordinates.

Complex join fibered over a map (cont.)

For $p > 0$, the p-fold complex join of A fibered over the map $\pi_{\mathbf{Y}}$, $J^p_{\mathbb{C},\mathbf{Y}}(A) \subset \mathbb{P}^k_{\mathbb{C}} \times \mathbb{P}^{(\ell+1)(p+1)-1}_{\mathbb{C}}$, is defined by the formula

$$\begin{aligned} J^p_{\mathbb{C},\mathbf{Y}}(\phi)(X_0,\ldots,X_k;Y^0_0,\ldots,Y^0_\ell,\ldots,Y^p_0,\ldots,Y^p_\ell) \\ \stackrel{\text{def}}{=} \bigwedge_{i=0}^{p} \phi(X_0,\ldots,X_k;Y^i_0,\ldots,Y^i_\ell). \end{aligned}$$

Main topological theorem

Theorem

Assume that A is closed. Then, for every $p \geq 0$, we have that

$$P_{\pi_{\mathbf{Y}}(A)} = (1 - T^2) P_{J^p_{\mathbb{C},\mathbf{Y}}(A)} \mod T^p.$$

The pseudo-Poincaré polynomial

We denote for any constructible $S \subset \mathbb{P}^n_{\mathbb{C}}$,

$$Q_S(T) \stackrel{\text{def}}{=} \sum_{j \geq 0} (b_{2j}(S) - b_{2j-1}(S)) T^j.$$

In other words:

$$Q_S = P_S^{\text{even}} - T\, P_S^{\text{odd}}.$$

Alexander-Lefschtez duality

Let $A \subset \mathbb{P}^n_{\mathbb{C}}$ be any constructible subset. Then,

$$Q_A(T) = -\mathrm{Rec}_n(Q_{\mathbb{P}^n_{\mathbb{C}} \setminus A}) + \sum_{i=0}^{n} T^i,$$

where for any polynomial $P(T)$,

$$\mathrm{Rec}_n(P) := T^n P(1/T).$$

Future work and open problems

- Remove compactness hypothesis.
- Obtain the classical Toda's theorem via algebro-geometric means.
- Develop a "Valiant type" theory over $\mathbb{R}$ and $\mathbb{C}$ or even more general structures. The "counting functions" considered should not be polynomials (such as the determinant, permanent etc.) as is done over finite fields, but rather *constructible functions*. We have a formulation of a $\mathbf{VP}_k^{\dagger} \neq \mathbf{VNP}_k^{\dagger}$ problem for $k = \mathbb{R}$ or $\mathbb{C}$.

Future work and open problems

- Remove compactness hypothesis.
- Obtain the classical Toda's theorem via algebro-geometric means.
- Develop a "Valiant type" theory over $\mathbb{R}$ and $\mathbb{C}$ or even more general structures. The "counting functions" considered should not be polynomials (such as the determinant, permanent etc.) as is done over finite fields, but rather *constructible functions*. We have a formulation of a $\mathbf{VP}_k^\dagger \neq \mathbf{VNP}_k^\dagger$ problem for $k = \mathbb{R}$ or $\mathbb{C}$.

Future work and open problems

- Remove compactness hypothesis.
- Obtain the classical Toda's theorem via algebro-geometric means.
- Develop a "Valiant type" theory over $\mathbb{R}$ and $\mathbb{C}$ or even more general structures. The "counting functions" considered should not be polynomials (such as the determinant, permanent etc.) as is done over finite fields, but rather *constructible functions*. We have a formulation of a $\mathbf{VP}_k^{\dagger} \neq \mathbf{VNP}_k^{\dagger}$ problem for $k = \mathbb{R}$ or $\mathbb{C}$.