Bounded Arithmetic and Randomized Computation

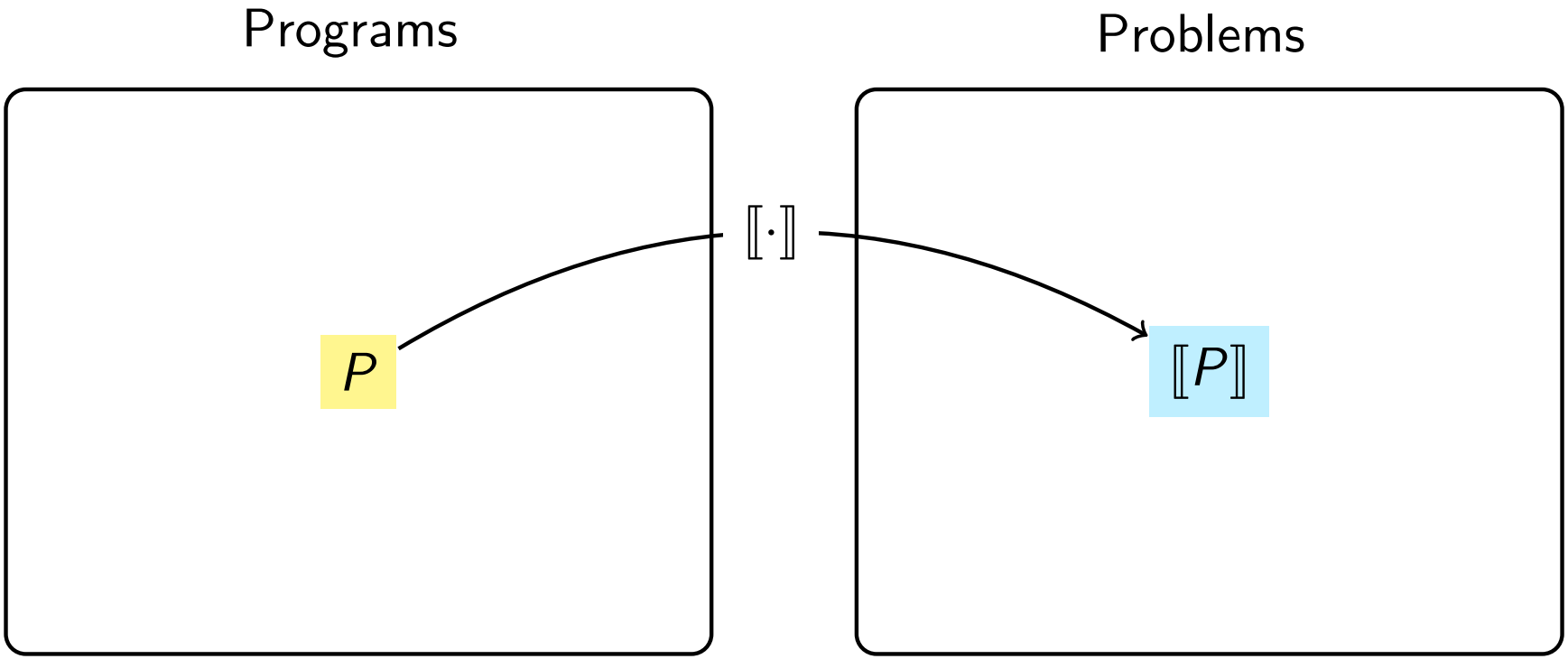
Davide Davoli

Alma Mater Studiorum - Università di Bologna

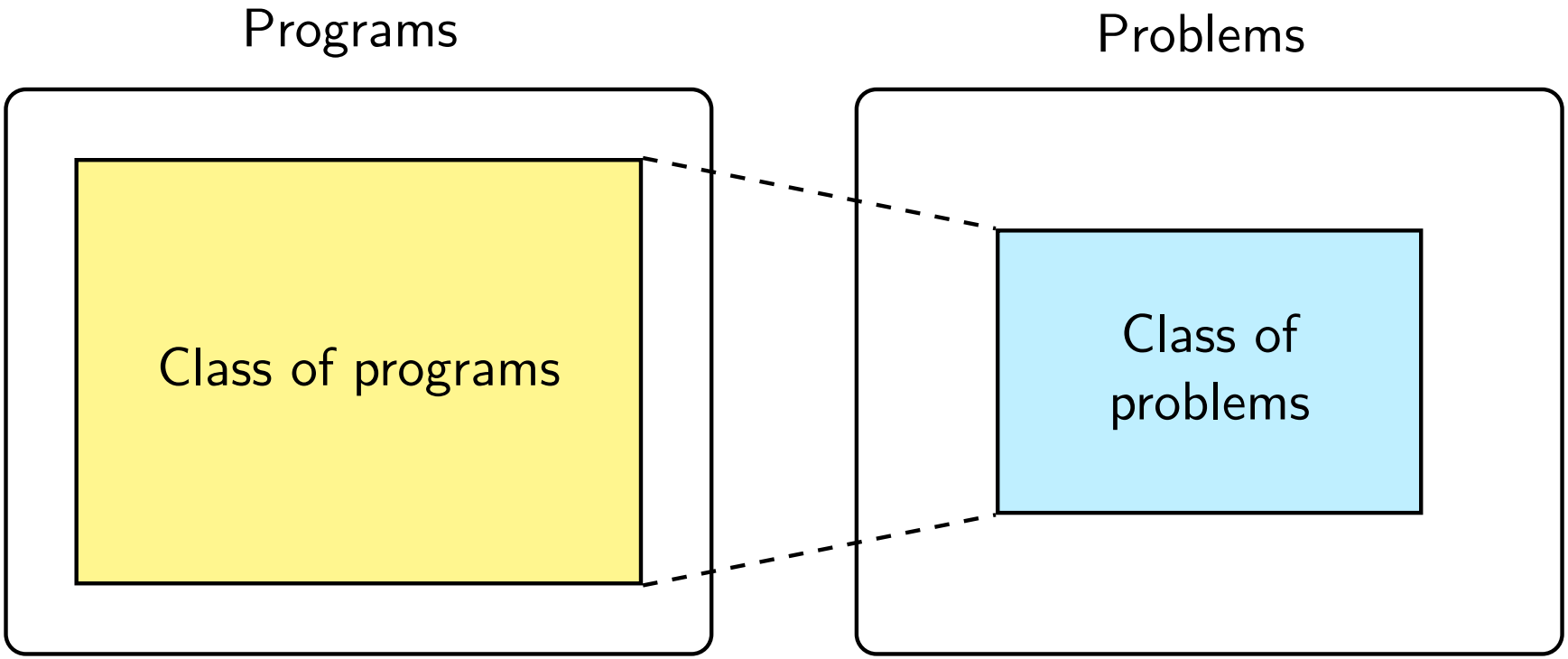
13th July 2022 - Alma Mater Studiorum - Università di Bologna

Introduction

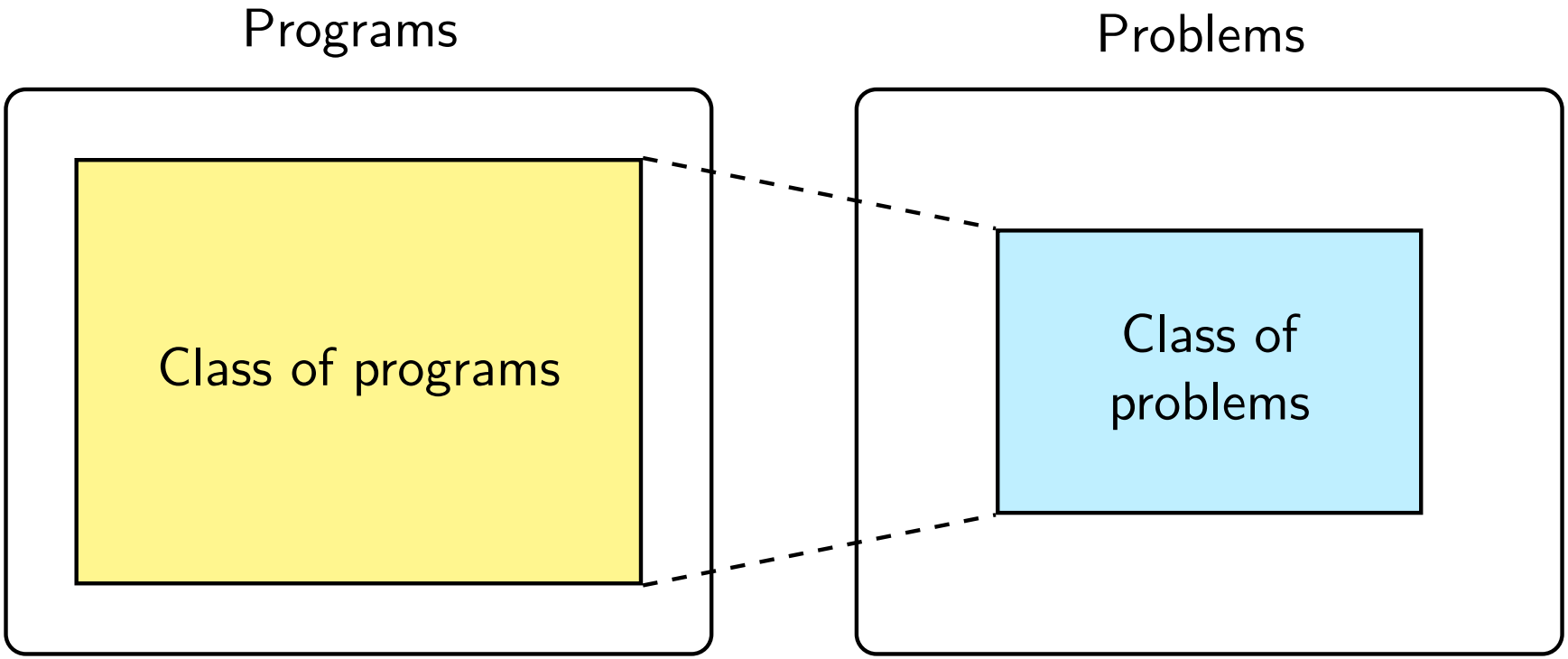
Implicit Computational Complexity



Implicit Computational Complexity



Implicit Computational Complexity



Idea: Study a complexity class describing the programs belonging to its preimage with respect to $[\![\cdot]\!]$.

Purpose of a Randomized Bounded Arithmetic

Bounded arithmetics are logical theories which allow us to:

- Represent programs by means of proofs.
- Capture complexity classes by means of classes of forumlæ.

Purpose of a Randomized Bounded Arithmetic

Bounded arithmetics are logical theories which allow us to:

- Represent programs by means of proofs.
- Capture complexity classes by means of classes of forumlæ.

A bounded arithmetic with randomness would enable us to:

- Capture the **PPT** functions.
- Characterize some probabilistic complexity classes, e.g. **BPP**.
- Find a *recursively enumerable* non-trivial subset of **BPP**.

S. Buss' Bounded Arithmetic

Buss 1986 introduces a Bounded Arithmetic for the class **FP**.

S. Buss' Bounded Arithmetic

Buss 1986 introduces a Bounded Arithmetic for the class **FP**.

Theorem (Buss' Theorem Buss 1986)

$f \in \mathbf{FP}$ if and only if there is a Σ_1^b formula G_f such that:

$$S_2^1 \vdash \forall x.\exists! y.G_f(x,y)$$

S. Buss' Bounded Arithmetic

Buss 1986 introduces a Bounded Arithmetic for the class **FP**.

Theorem (Buss' Theorem Buss 1986)

*$f \in$ **FP** if and only if there is a Σ_1^b formula G_f such that:*

$$S_2^1 \vdash \forall x.\exists! y.G_f(x,y)$$

- The proof system S_2^1 captures the complexity of f.

S. Buss' Bounded Arithmetic

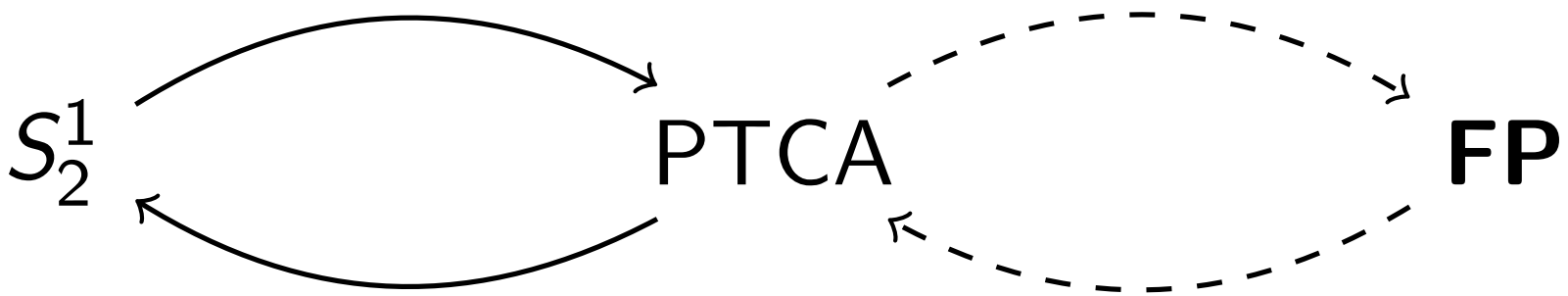
Buss 1986 introduces a Bounded Arithmetic for the class **FP**.

Theorem (Buss' Theorem Buss 1986)

*$f \in$ **FP** if and only if there is a Σ_1^b formula G_f such that:*

$$S_2^1 \vdash \forall x.\exists! y.G_f(x,y)$$

- The proof system S_2^1 captures the complexity of f.



Schema of the proof given by Ferreira 1988.

Main Theorem

$\mathcal{L}$ is a first-order word language including the unary predicate `Flip` and associated with a *quantitative* semantic inspired by Antonelli et al. 2021.

Main Theorem

$\mathcal{L}$ is a first-order word language including the unary predicate `Flip` and associated with a *quantitative* semantic inspired by Antonelli et al. 2021.

Theorem

The **PPT** *functions are exactly the functions f for which there exists a Σ_1^b formula of $\mathcal{L}$, G_f, such that:*

- $RS_2^1 \vdash \forall x \exists! y . G_f(x, y)$.
- *The interpretation of $G_f(x, y)$ has measure equal to $Pr[f(x) = y]$.*

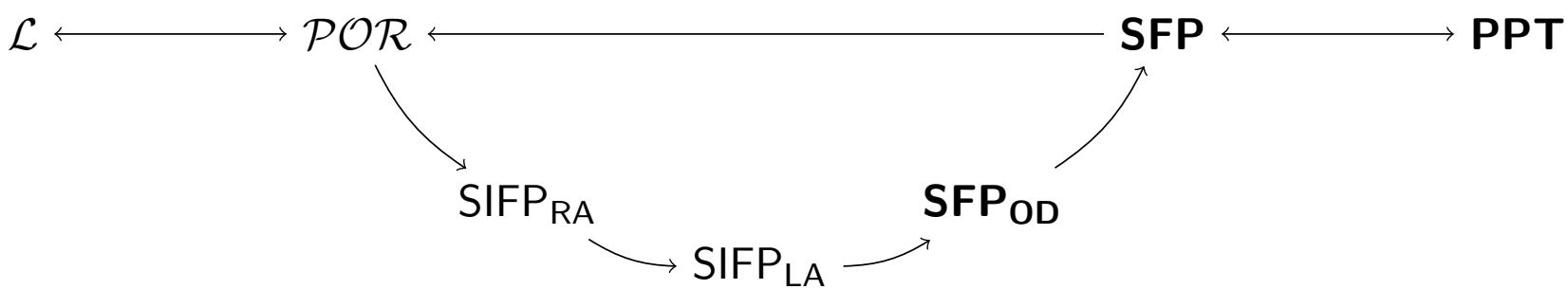
Main Theorem

$\mathcal{L}$ is a first-order word language including the unary predicate `Flip` and associated with a *quantitative* semantic inspired by Antonelli et al. 2021.

Theorem

The **PPT** *functions are exactly the functions f for which there exists a Σ_1^b formula of $\mathcal{L}$, G_f, such that:*

- $RS_2^1 \vdash \forall x \exists! y . G_f(x, y)$.
- *The interpretation of $G_f(x, y)$ has measure equal to $Pr[f(x) = y]$.*



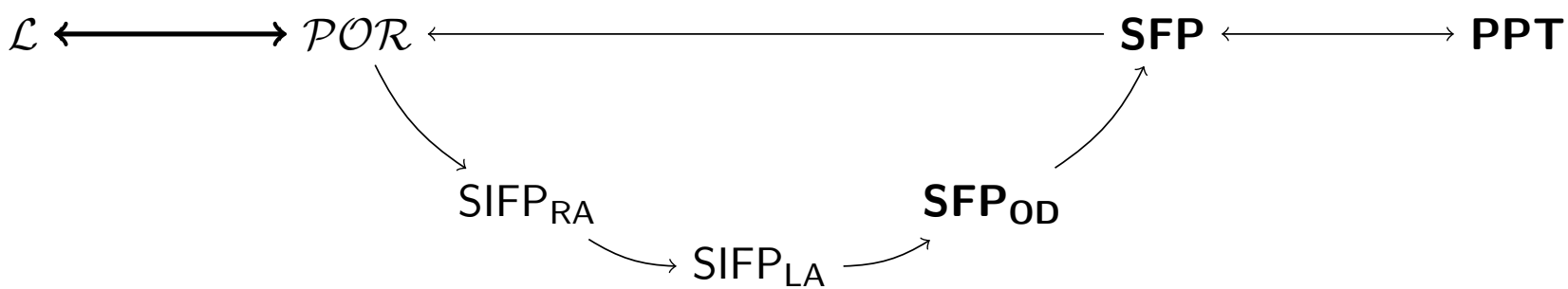
Main Theorem

$\mathcal{L}$ is a first-order word language including the unary predicate `Flip` and associated with a *quantitative* semantic inspired by Antonelli et al. 2021.

Theorem

The **PPT** *functions are exactly the functions f for which there exists a Σ_1^b formula of $\mathcal{L}$, G_f, such that:*

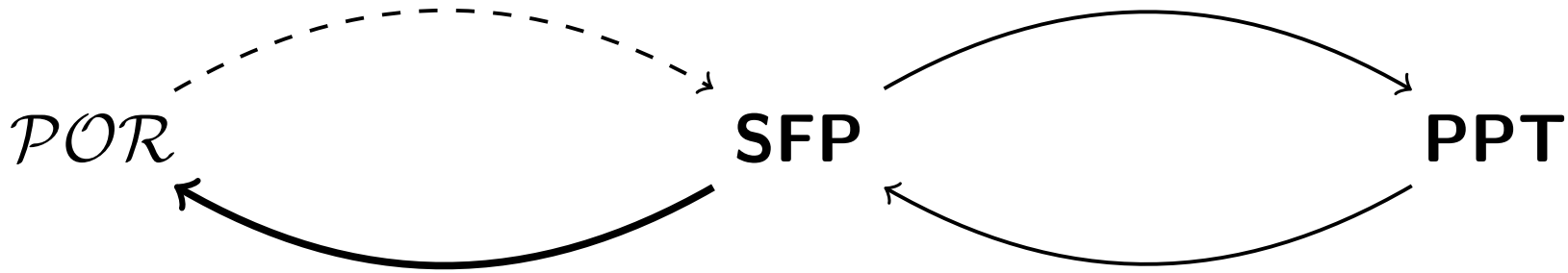
- $RS_2^1 \vdash \forall x \exists! y . G_f(x, y)$.
- *The interpretation of $G_f(x, y)$ has measure equal to $Pr[f(x) = y]$.*



Proved following Cook and Urquhart 1993 and Ferreira 1988.

Some reductions

Expressivity of $\mathcal{POR}$, Part I



The Stream Machines' source of randomness are tapes: functions $\eta \in \{0,1\}^{\mathbb{N}}$, while $\mathcal{POR}$ functions use functions $\omega \in \{0,1\}^{\mathbb{S}}$.

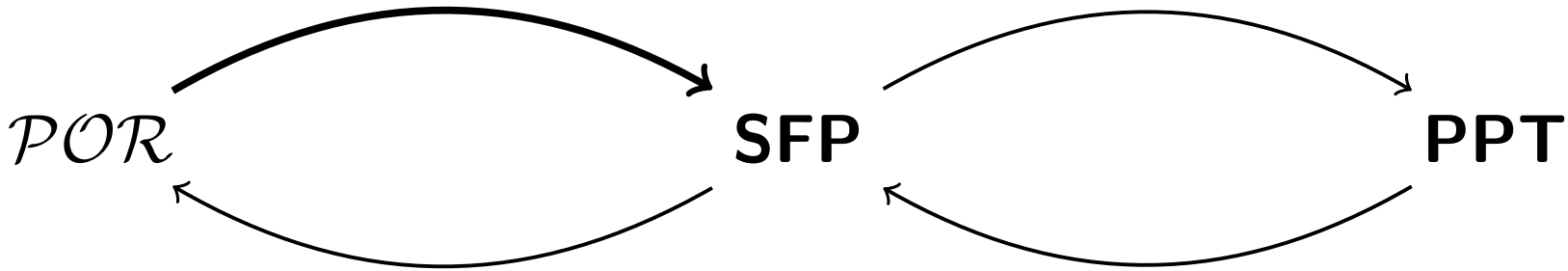
The two classes correspond, but *modulo measure*:

Lemma

*For every $M \in$ **SFP**, there is a f_M in $\mathcal{POR}$ such that:*

$$\mu(\{\eta \in \{0,1\}^{\mathbb{N}} \mid M(x,\eta) = y\}) = \mu(\{\omega \in \{0,1\}^{\mathbb{S}} \mid f_M(x,\omega) = y\})$$

Expressivity of $\mathcal{POR}$, Part II



- The *poly-time* complexity of a function $M \in$ **SFP** bounds the amount of the random bits of η influencing the output.
- A $\mathcal{POR}$ function $f(x, \omega) : \mathbb{S} \times \{0,1\}^{\mathbb{S}} \longrightarrow \mathbb{S}$ can query up to $2^{p(|x|)}$ different values to ω.

Fixed x and ω, each $f \in \mathcal{POR}$ queries at most a polynomial number of values to ω.

Idea: reduce $\mathcal{POR}$ to an imperative formalism, and then to **SFP**.

Expressivity of $\mathcal{POR}$, Part III

$$\mathcal{POR} \longrightarrow \mathsf{SIFP}_{\mathsf{RA}} \dashrightarrow \mathsf{SIFP}_{\mathsf{LA}} \dashrightarrow \mathbf{SFP}_{\mathbf{OD}} \dashrightarrow \mathbf{SFP}$$

Goal: Showing that for each $f \in \mathcal{POR}$ there is a *poly-time* imperative program P_f which can query ω and such that $[\![P_f]\!] = f$.

Expressivity of $\mathcal{POR}$, part IV

$$\mathcal{POR} \longrightarrow \mathsf{SIFP}_{\mathsf{RA}} \longrightarrow \mathsf{SIFP}_{\mathsf{LA}} \dashrightarrow \mathbf{SFP}_{\mathbf{OD}} \dashrightarrow \mathbf{SFP}$$

Goal: Showing that for each *poly-time* $P \in \mathsf{SIFP}_{\mathsf{RA}}$ there is a *poly-time* imperative program $Q \in \mathsf{SIFP}_{\mathsf{LA}}$ such that:

$$\mu(\{\omega \in \{0,1\}^{\mathbb{S}} \mid [\![P]\!](x,\omega) = y\}) = \mu(\{\eta \in \{0,1\}^{\mathbb{N}} \mid [\![Q]\!](x,\eta) = y\})$$

and Q reads random bits sequentially from η.

Expressivity of $\mathcal{POR}$, part V

$$\mathcal{POR} \longrightarrow \mathsf{SIFP}_{\mathsf{RA}} \longrightarrow \mathsf{SIFP}_{\mathsf{LA}} \longrightarrow \mathbf{SFP}_{\mathbf{OD}} \longrightarrow \mathbf{SFP}$$

Goal: Showing that for each *poly-time* $P \in \mathsf{SIFP}_{\mathsf{LA}}$ there is a *poly-time* STM $N \in \mathbf{SFP}$ such that:

$$\mu(\{\eta \in \{0,1\}^{\mathbb{S}} | [\![P]\!](x,\eta) = y\}) = \mu(\{\eta \in \{0,1\}^{\mathbb{N}} | [\![N]\!](x,\eta) = y\})$$

Theorem

$\mathcal{POR} \simeq \mathbf{PPT}$.

Characterizing **BPP**

A semantic Characterization of **BPP**

We are investigating what happens if we extend $\mathcal{L}$ with a measure quantifier $\mathbf{C}^{n/m}(F)$, defined by the following semantics:

$$[\![\mathbf{C}^{n/m}(F)]\!]_\xi := \begin{cases} \{0,1\}^{\mathbb{S}} & \text{if } \mu([\![F]\!]_\xi) \geq n/m \\ \emptyset & \text{otherwise} \end{cases}$$

A semantic Characterization of **BPP**

We are investigating what happens if we extend $\mathcal{L}$ with a measure quantifier $\mathbf{C}^{n/m}(F)$, defined by the following semantics:

$$[\![\mathbf{C}^{n/m}(F)]\!]_\xi := \begin{cases} \{0,1\}^{\mathbb{S}} & \text{if } \mu([\![F]\!]_\xi) \geq n/m \\ \emptyset & \text{otherwise} \end{cases}$$

Corollary

BPP *is exactly the set of languages L with characteristic function* f_L *for which there is a* Σ_1^b *formula* G_L *such that:*

$$RS_2^1 \vdash \forall x \exists! y . G_L(x,y)$$

$$\forall \sigma \in \mathbb{S}. \models \mathbf{C}^{2/3}(G_L(\sigma, f_L(\sigma)))$$

Further work

Towards a syntactic characterization of **BPP**

It is well known that **BPP** $\subseteq \Sigma_2^p$, so according to Buss 1986, for each $L \in$ **BPP** there exists a Σ_3^b formula H_L which decides L. Putting together this result with the semantic characterization of **BPP**, it must hold that:

$$\forall x, y.\mathbf{C}^{2/3}(G_L(x,y) \leftrightarrow H_L(x,y))$$

Since all the terms in $\mathcal{L}$ are bounded, $\mathbf{C}^{n/m}$ can be reduced to a counting existential, and then to a bonded existential.

$$\forall x, y.\exists^{\geq k(x)} t_f \preceq t(x).(G_L(x,y) \leftrightarrow H_L(x,y))[\mathtt{Flip}(t) \leftarrow \pi_t(t_f) = 1]$$

Conjecture

There is a non-trivial subset of **BPP**, **PBPP** such that:

$$\forall L \in PBPP.\exists F_L.PA \vdash \forall x.\forall y.F_L(x,y)$$

Contributions

- $\mathcal{POR} \simeq \mathbf{PPT}$.
- As a corollary: Cobham-style Algebras $= \mathbf{FP}$.
- Characterizations of **BPP**, **ZPP**, and other probabailistic complexity classes.

Thanks for your attention

References

- Antonelli, M., U. Dal Lago, and P. Pistone (2021). "On Measure Quantifiers in First-Order Arithmetic". In: *Connecting with Computabiliy*. Ed. by L. De Mol et al. Springer, pp. 12–24.
- Buss, S.R. (1986). *Bounded Arithmetic*. Bibliopolis.
- Cook, Stephen and Alasdair Urquhart (1993). "Functional Interpretations of Feasibly Constructive Arithmetic". In: *Annals of Pure and Applied Logic* 63.2, pp. 103–200. DOI: 10.1016/0168-0072(93)90044-e.
- Ferreira, F. (Dec. 1988). "Polynomial Time Computable Arithmetic and Conservative Extesions". Ph.D. Dissertation.