Non-deterministic computation and the Jayne Rogers Theorem

Arno Pauly Matthew de Brecht

Clare College, University of Cambridge

National Institute of Information and Communications Technology, Kyoto

Developments in Computational Models 2012, Cambridge

The message

Thinking in terms of computational models can be useful to solve unrelated problems.

The story

A simulation result from the study of Type 2 hyper-computation (namely, singlevalued functions computable by non-deterministic machines with locally compact advice spaces can already be computed by machines making finitely many mindchanges) implies a variant of the Jayne Rogers theorem from descriptive set theory, stating that Δ^0_2-measurable functions between complete metric spaces are piecewise continuous.

Outline

Turing machines running forever

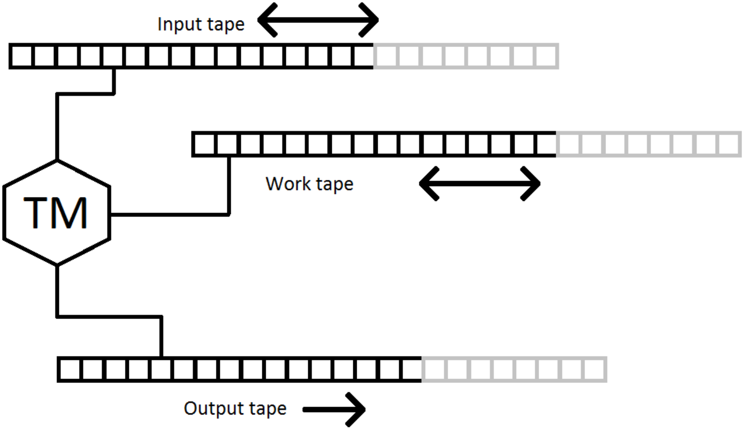

Figure: The core model

Basic concept

A non-deterministic machine…

1. makes a guess,
2. and then either produces a correct output
3. **or** rejects the guess.

Non-deterministic Type 2 machines

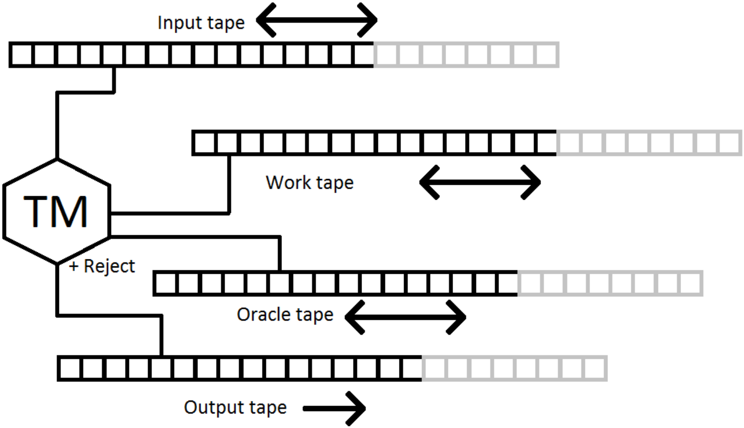

Figure: The core model

Formal definition

$$\forall p \in \mathrm{dom}(f)$$
$$(\exists z \in Z \; . \; M(p,z)\downarrow) \wedge$$
$$(\forall z \in Z \; . \; M(p,z)\downarrow \Rightarrow M(p,z) \in f(p))$$

The rôle of Z

We are promised that our guess comes from some fixed set Z (encoded suitably over infinite sequences). These have a strong impact on computational power:

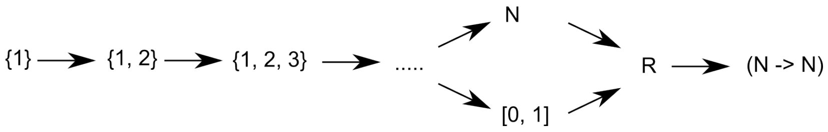

Figure: Computational power for various advice spaces

Examples

Example

Guessing some $z \in \{0,1\}^{\mathbb{N}}$ does not allow to decide equality over $\{0,1\}^{\mathbb{N}}$, but guessing some $z \in \mathbb{N}$ does.

Example

Guessing some $z \in \{0,1\}^{\mathbb{N}}$ does allow to find zeros of monotone functions $f : [0,1] \to [-1,1]$, $f(0) = -1 = -f(1)$; guessing some $z \in \mathbb{N}$ does not.

Complete problems

Definition

For any $\mathbf{X}$, define $C_\mathbf{X} :\subseteq \mathcal{A}(\mathbf{X}) \rightrightarrows \mathbf{X}$ via $x \in C_\mathbf{X}(A)$ iff $x \in A$.

Theorem (Brattka, de Brecht & P.)

f is non-deterministically computable guessing some $z \in \mathbf{Z}$, iff $f \leq_W C_\mathbf{Z}$.

Turing machines changing their minds

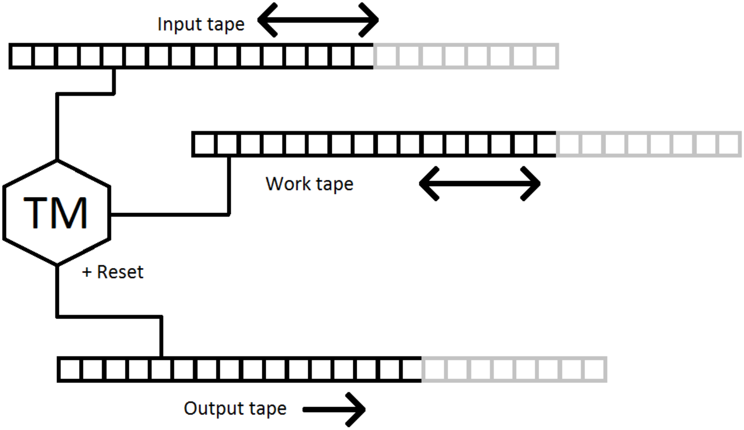

Figure: Computation with mindchanges

Equivalent models

Theorem

The following computational models are equivalent:

1. *Non-deterministic Type 2 machines guessing from $\mathbb{N}$.*
2. *Type 2 machines making finitely many mindchanges.*
3. *Type 2 machines using an oracle for equality testing finitely many times.*

The base case

Theorem (Brattka & Gherardi)

If $f : \mathbf{X} \to \mathbf{Y}$ is singlevalued, $\mathbf{Y}$ admissible and f non-deterministically computable with advice space $\{0,1\}^{\mathbb{N}}$, then f is already computable.

More generally

Theorem (Brattka, de Brecht & P.)

If $f : \mathbf{X} \to \mathbf{Y}$ is singlevalued, $\mathbf{Y}$ admissible and f non-deterministically computable with advice space $\{0,1\}^{\mathbb{N}} \times Z$, then f is already non-deterministically computable with advice space Z.

Corollary

For singlevalued functions into admissible spaces, non-deterministic computation with advice space $\mathbb{N} \times \{0, 1\}^{\mathbb{N}}$ (or equivalently $\mathbb{R}$) is no stronger than computation with finitely many mindchanges.

Definitions I

- Let $\mathbf{\Sigma}^0_1(\mathbf{X}) = \mathcal{O}(\mathbf{X})$.
- Let $\mathbf{\Pi}^0_\alpha(\mathbf{X}) = \{X \setminus U \mid U \in \mathbf{\Sigma}^0_\alpha(\mathbf{X})\}$.
- Let $\mathbf{\Sigma}^0_{n+1}(\mathbf{X}) = \{\left(\bigcup_{n\in\mathbb{N}} U_n\right) \mid U_n \in \mathbf{\Pi}^0_\alpha(\mathbf{X})\}$.
- Let $\mathbf{\Delta}^0_\alpha(\mathbf{X}) = \mathbf{\Sigma}^0_\alpha(\mathbf{X}) \cap \mathbf{\Pi}^0_\alpha(\mathbf{X})$

Definitions II

Definition

$f : \mathbf{X} \to \mathbf{Y}$ is Δ^0_2-measurable, iff $f^{-1}(U) \in \Delta^0_2(\mathbf{X})$ for each $U \in \mathcal{O}(\mathbf{Y})$.

Definitions III

Definition

$f : \mathbf{X} \to \mathbf{Y}$ is piecewise continuous, if there is a closed cover $(A_n)_{n\in\mathbb{N}}$ of $\mathbf{X}$ such that any $f_{|A_n}$ is continuous.

The Jayne Rogers Theorem

Theorem (Jayne & Rogers)

Let $\mathbf{X}$, $\mathbf{Y}$ *be complete metric spaces. Then* $f : \mathbf{X}$ *is* Δ^0_2*-measurable iff it is piecewise continuous.*

Computable topology

Idea: A set being open means that membership is provable, closed means membership is refutable.

Piecewise continuity and non-deterministic computation

Theorem

Evaluation of piecewise continuous functions is non-deterministically computable with advice space $\mathbb{N}$, and complete.

Proof.

Given $x \in \mathbf{X}$, guess $n \in \mathbb{N}$ such that $x \in A_n$. Then use that evaluation for continuous functions is computable. For the other direction, let A_n be the set of inputs where n is a correct guess. □

Evaluation of Δ^0_2-measurable functions

Theorem

Evaluation of Δ^0_2-measurable functions is non-deterministically computable with advice space $\mathbb{N} \times \{0,1\}^{\mathbb{N}}$.

The proof idea

1. Guess $n \in \mathbb{N}$ and $p \in \{0,1\}^{\mathbb{N}}$ encoding some $y \in \mathbf{Y}$.
2. Compute $\mathbf{Y} \setminus \{y\} \in \mathcal{O}(\mathbf{Y})$.
3. Compute $f^{-1}(\mathbf{Y} \setminus \{y\}) = \bigcap_{i \in \mathbb{N}} O_i$.
4. Test $x \in O_i$ for all $i \leq n$, and reject if all answers are positive.
5. Output y.

The computable Jayne Rogers Theorem

Theorem

From a name of a Δ^0_2-measurable function we can compute a name for the same function as a piecewise continuous function.

Caveat: Do all Δ^0_2-measurable functions have corresponding names?

Bibliography

M. Ziegler.
Revising type-2 computation and degrees of discontinuity.
Electronic Notes in Theoretical Computer Science, 167:255–274, 2007.

M. Ziegler.
Real hypercomputation and continuity.
Theory of Computing Systems, 41:177–206, 2007.

Bibliography II

Vasco Brattka, Matthew de Brecht and Arno Pauly.
Closed Choice and a Uniform Low Basis Theorem.
Annals of Pure and Applied Logic, 8:968–1008, 2012.

J. E. Jayne and C. A. Rogers.
First level Borel functions and isomorphisms.
Journal de Mathématiques Pures et Appliquées, 61:177–205, 1982.