Μη Υπολογισιμότητα

Δημήτρης Φωτάκης

Τμήμα Μηχανικών Πληροφοριακών και Επικοινωνιακών Συστημάτων
Πανεπιστήμιο Αιγαίου

Η Θέση των Church-Turing

- Δεν μπορέσαμε να ενισχύσουμε Μ.Τ. με επιπλέον χαρακτηριστικά (π.χ. πολλαπλές ταινίες, πολλαπλές κεφαλές, μη ντετερμινισμός).
- **Κανένα** γνωστό υπολογιστικό μοντέλο δεν είναι **ισχυρότερο** από Τ.Μ. (π.χ. αναδρομικές συναρτήσεις, Gödel, Herbrand - Gödel, μηχανή με μνήμη τυχαίας προσπέλασης, Church, Markov, Post)
- **Θέση των Church-Turing**: Τ.Μ. που **τερματίζει πάντα** αποτελεί τυπικό ορισμό της διαισθητικής έννοιας του αλγόριθμου.
- Διαισθητική έννοια αλγόριθμου **ταυτίζεται** με Turing-υπολογισιμότητα.
- Θέση Church-Turing **δεν μπορεί** να αποδειχθεί. Είναι θεωρητικά **δυνατό** αλλά **όχι πιθανό** να βρεθεί στο μέλλον πιο ισχυρό μοντέλο υπολογισμού.
- Τυπικές **γλώσσες** κωδικοποιούν **προβλήματα** (απόφασης):

$$A = \{\langle G\rangle \in \{0,1\}^* : G \text{ είναι συνεκτικό γράφημα}\}$$

- Στο εξής εστιάζουμε κυρίως σε **αλγόριθμους** για προβήματα και λιγότερο σε Μ.Τ. για αναγνώριση γλωσσών.

Μη Υπολογισιμότητα

- Υπάρχουν μετρήσιμα άπειρες Μ.Τ. και μη μετρήσιμα άπειρες γλώσσες (τα "προβλήματα" είναι πολύ περισσότερα από τις "λύσεις").
- Για κάποιες γλώσσες **δεν υπάρχουν Μ.Τ.** που τις αποφασίζουν.
- Μπορούμε όμως να βρούμε **κάποια συγκεκριμένη** γλώσσα που δεν αποφασίζεται από Μ.Τ.;
- Θδο δεν μπορεί να αποφασιστεί αν Μ.Τ. M **τερματίζει** με είσοδο w.
- Ορισμός **προβλήματος τερματισμού** και απόδειξη απαιτούν ορισμό **καθολικών** Μ.Τ.

Καθολικές Μηχανές Turing

- Μ.Τ. είναι υπολογιστής **καθορισμένου προγράμματος**.
- **Καθολική** Μ.Τ. U: Μ.Τ. που προσομοιώνει κάθε άλλη Μ.Τ. $U(M;w)$: με είσοδο κωδικοποίηση Μ.Τ. M και εισόδου w, προσομοιώνει $M(w)$. Βλ. **διερμηνευτής** γραμμένος σε κάποια ΓΠ για προγράμματα της ίδιας ΓΠ.
- **Καθολική** Μ.Τ. U είναι **προγραμματιζόμενος** υπολογιστής.
  - Μ.Τ. M είναι το **πρόγραμμα** της U.
  - w είναι η είσοδος του προγράμματος M.
- Μ.Τ. $M = (K, \Sigma, \delta, s, H)$. Ελάχιστα i, j: $2^i \geq |K|$ και $2^j \geq |\Sigma| + 2$.
  - Κατάσταση q_ℓ: qu, u δυαδική κωδικοποίηση ℓ μήκους i $(q000, q001, \ldots)$.
  - Σύμβολο a_ℓ: au, u δυαδική κωδικοποίηση ℓ μήκους j $(\sqcup : a000, \triangleright : a001, \leftarrow : a010, \rightarrow : a011, \alpha : a100, \beta : a101, \ldots)$
- $\langle M\rangle$: κωδικοποίηση **συνάρτησης μετάβασης** δ. $\delta(q,a) = (p,b)$ γράφεται (q,a,p,b) και ταξινομούνται **λεξικογραφικά**. $\langle w\rangle$: κωδικοποίηση **εισόδου** w.

$$U(\langle M\rangle\,\langle w\rangle) = \langle M(w)\rangle$$

Παράδειγμα Μηχανής Turing

$$M = (K, \Sigma, \delta, s, \{h\})$$
$$K = \{s, q, h\}$$
$$\Sigma = \{a, \sqcup, \triangleright\}$$

q	σ	$\delta(q,\sigma)$
s	a	$(q, \sqcup)$
s	$\sqcup$	$(h, \sqcup)$
s	$\triangleright$	$(s, \rightarrow)$
q	a	(s, a)
q	$\sqcup$	$(s, \rightarrow)$
q	$\triangleright$	$(q, \rightarrow)$

Αναπαρ.	
s	$q00$
q	$q01$
h	$q11$
$\sqcup$	$a000$
$\triangleright$	$a001$
$\leftarrow$	$a010$
$\rightarrow$	$a011$
a	$a100$

Αναπαράσταση $\langle M \rangle$
$(q00, a100, q01, a000)$
$(q00, a000, q11, a000)$
$(q00, a001, q00, a011)$
$(q01, a100, q00, a100)$
$(q01, a000, q00, a011)$
$(q01, a001, q01, a011)$

- Αναπαράσταση περιεχομένων ταινίας $\triangleright aa\sqcup a$:

$$\langle \triangleright aa\sqcup a \rangle = a001a100a100a000a100$$

Προσομοίωση από Καθολική Μ.Τ.

- Εύκολα, αν καθολική Μ.Τ. U έχει **τρεις** ταινίες:
  - 1η ταινία περιέχει τρέχον περιεχόμενο της **ταινίας** $M(w)$ και κεφαλή σε θέση αντίστοιχη με κεφαλή του $M(w)$.
  - 2η ταινία περιέχει $\langle M \rangle$.
  - 3η ταινία **τρέχουσα κατάσταση** της $M(w)$ (αρχικά s).
  - Εύκολος εντοπισμός και εκτέλεση μετάβασης.
- $U(\langle M \rangle \langle w \rangle)$ **υπολογίζει** ότι ακριβώς υπολογίζει $M(w)$
  - Τερματίζει ανν $M(w)$ τερματίζει.
  - Τερματίζει σε κατάστ. y / n ανν $M(w)$ τερματίζει σε κατάστ. y / n.
  - Τερματίζει σε κατάστ. h με έξοδο u ανν $M(w) = u$.

Το Πρόβλημα του Τερματισμού

- $H = \{\langle M \rangle \langle w \rangle : M$ είναι Μ.Τ. και **τερματίζει** με είσοδο $w\}$
- H είναι **αναδρομικά απαριθμήσιμη** (ημιαποφασίσιμη) από **καθολική** Μ.Τ.
  H αναδρομική ανν Αναδρομικά Απαριθμήσιμες $=$ Αναδρομικές
- Έστω M_L ημιαποφασίζει L και M_H αποφασίζει H.

$$\left.\begin{array}{l} M_H(\langle M_L \rangle \langle w \rangle) = y \Rightarrow w \in L \\ M_H(\langle M_L \rangle \langle w \rangle) = n \Rightarrow w \notin L \end{array}\right\} \Rightarrow L \text{ αναδρομική}$$

- H **δεν** είναι **αναδρομική** (αποφασίσιμη)!
  Αναδρομικές Γλώσσες $\subset$ Αναδρομικά Απαριθμήσιμες Γλώσσες.
- Έστω Μ.Τ. $M_H(\langle M \rangle, \langle w \rangle)$ που **αποφασίζει** αν $M(w)$ **τερματίζει** για κάθε Μ.Τ. M και είσοδο w.
- Θεωρούμε Μ.Τ. $D(\langle M \rangle)$:

  if $M_H(\langle M \rangle, \langle M \rangle) = y$ then **run forever** else **halt**
- $D(\langle M \rangle)$ **τερματίζει** ανν $M(\langle M \rangle)$ **δεν τερματίζει**!
- $D(\langle D \rangle)$ τερματίζει **ανν** $D(\langle D \rangle)$ δεν τερματίζει! **Άτοπο**

Διαγωνιοποίηση

- Συστηματική απαρίθμηση όλων των Μ.Τ.: $M_1, M_2, M_3, M_4, \ldots, D, \ldots$
- Αν H **αναδρομική**, D **τερματίζει** και έχουμε **άτοπο**.

  if $M_H(\langle M \rangle, \langle M \rangle) = y$ then **run forever** else **halt**

	$\langle M_1 \rangle$	$\langle M_2 \rangle$	$\langle M_3 \rangle$	$\langle M_4 \rangle$	$\cdots$	$\langle D \rangle$	$\cdots$
M_1	**T**	**Δ**	**Δ**	**T**		**T**	
M_2	**T**	**T**	**Δ**	**T**	$\cdots$	**Δ**	
M_3	**Δ**	**T**	**Δ**	**Δ**		**T**	
M_4	**T**	**Δ**	**T**	**T**		**T**	
$\vdots$			$\vdots$		$\ddots$		
D	**Δ**	**Δ**	**T**	**Δ**		**???**	
$\vdots$			$\vdots$				$\ddots$

Συνέπειες

- $H = \{\langle M\rangle\langle w\rangle : M$ είναι Μ.Τ. και **τερματίζει** με είσοδο $w\}$
  $H_1 = \{\langle M\rangle : M$ είναι Μ.Τ. και **τερματίζει** με είσοδο $\langle M\rangle\}$
- H, H_1 δεν είναι **αναδρομικές**, δηλ. δεν είναι **υπολογίσιμες**!
- **Μη επιλύσιμα** προβλήματα: αντιστοιχούν σε **μη αναδρομικές** γλώσσες.
  Π.χ. δεν υπάρχει αλγόριθμος που για κάθε M και w αποφασίζει αν $M(w)$ τερματίζει.
- Πρόβλημα τερματισμού μπορεί να λύνεται (εύκολα) σε ειδικές περιπτώσεις.
  Δεν υπάρχει **γενική μέθοδος** που απαντάει σωστά σε όλες τις περιπτώσεις.
- Γλώσσα L **αναδρομική** **ανν** L και $\overline{L}$ **αναδρομικά απαριθμήσιμες**.
- $\overline{H}, \overline{H_1}$ δεν είναι **αναδρομικά απαριθμήσιμες**!
- Αναδρομικά απαριθμήσιμες γλώσσες όχι κλειστές ως προς **συμπλήρωμα**.
  Άλλες κλειστότητες (τομή, ένωση, παράθεση, *) για αναδρομ. απαριθμ.;
  Κλειστότητες (συμπλήρωμα, τομή, ένωση, παράθεση, *) για αναδρομικές;

Αναγωγή

- Πρόβλημα A **ανάγεται στο** B ($A \le_R B$) αν από
  τη λύση του B προκύπτει ("εύκολα") η λύση του A.
- Αν A μη επιλύσιμο, τότε και B **μη επιλύσιμο**.
- **Αναγωγή** μη επιλύσιμου προβλήματος σε πρόβλημα Π
  για νδο Π **μη επιλύσιμο** (χωρίς εφαρμογή διαγωνιοποίησης).
- **Αναγωγή** L_1 **στην** L_2 ($L_1 \le_R L_2$) είναι μια **αναδρομική** συνάρτηση
  $f : \Sigma^* \mapsto \Sigma^*$ τέτοια ώστε $w \in L_1$ **ανν** $f(w) \in L_2$.
- Αν L_2 είναι αναδρομική και $L_1 \le_R L_2$, L_1 είναι **αναδρομική**.
- Αν L_1 δεν είναι αναδρομική και $L_1 \le_R L_2$, L_2 **δεν** είναι **αναδρομική**.
- Αν L_2 αποφασίζεται από Τ.Μ. M_2 και αναγωγή L_1 στην L_2
  υπολογίζεται από Τ.Μ. M_f, τότε $M_f M_2$ **αποφασίζει** L_1!

Περισσότερα Μη Επιλύσιμα Προβλήματα

- **Μη επιλύσιμο** πρόβλημα Π_1:
  "Δεδομένης Μ.Τ. M, τερματίζει η M αρχίζοντας από κενή ταινία;"
- **Αναγωγή** του προβλήματος του τερματισμού στο Π_1.
- Έστω Μ.Τ. M και είσοδος w. Θέλουμε να μάθουμε αν $M(w)$ τερματίζει.
- Κατασκευάζουμε Μ.Τ. M_w που όταν ξεκινάει με κενή ταινία, $M_w(\varepsilon)$, τότε :
  – **γράφει** w στην ταινία της, και
  – **προσομοιώνει** $M(w)$.
- Αναγωγή είναι αναδρομική (υπολογίσιμη).
- $M_w(\varepsilon)$ τερματίζει ανν $M(w)$ τερματίζει.
- Αν υπήρχε αλγόριθμος που αποφασίζει αν $M_w(\varepsilon)$ τερματίζει,
  θα μπορούσαμε να αποφασίσουμε αν $M(w)$ τερματίζει.

Περισσότερα Μη Επιλύσιμα Προβλήματα

- **Μη επιλύσιμα** προβλήματα
  Π_2: "Δεδομένης Μ.Τ. M, $\exists w \in \Sigma^*$ ώστε $M(w)$ τερματίζει;"
  Π_3: "Δεδομένης Μ.Τ. M, $\forall w \in \Sigma^*$, $M(w)$ τερματίζει;"
- **Αναγωγή** του Π_1 στα Π_2 και Π_3.
- Έστω Μ.Τ. M. Θέλουμε να μάθουμε αν $M(\varepsilon)$ τερματίζει.
- Κατασκευάζουμε Μ.Τ. M' που:
  – **αγνοεί** την είσοδο της (π.χ. διαγραφή), και
  – **προσομοιώνει** $M(\varepsilon)$.
- $\forall w \in \Sigma^*$, $M'(w)$ τερματίζει ανν $M(\varepsilon)$ τερματίζει.
- M' **τερματίζει**:
  – είτε για **όλες** τις εισόδους w (ανν $M(\varepsilon)$ τερματίζει)
  – είτε για **καμία** είσοδο w (ανν $M(\varepsilon)$ δεν τερματίζει).
- Αλγόριθμος για Π_2 ή Π_3, αποφασίζει αν $M(\varepsilon)$ τερματίζει.

Περισσότερα Μη Επιλύσιμα Προβλήματα

- **Μη επιλύσιμο** πρόβλημα Π_4:
  "Δεδομένων Μ.Τ. M_1 και M_2, τερματίζουν για ίδιο σύνολο εισόδων;"
- **Αναγωγή** του Π_3 στο Π_4.
- Έστω Μ.Τ. M. Ρωτάμε αν M τερματίζει **για κάθε είσοδο**.
- Έστω y Μ.Τ. που τερματίζει αποδεχόμενη κάθε είσοδο.
- Μ.Τ. M και y τερματίζουν για ίδιο σύνολο εισόδων ανν
  M τερματίζει για κάθε είσοδο.

Περισσότερα Μη Επιλύσιμα Προβλήματα

- Για Μ.Τ. M, έστω $L^{(s)}(M)$ η γλώσσα που ημιαποφασίζεται από M:
  $$L^{(s)}(M) = \{w \in \Sigma^* : M(w) \text{ τερματίζει}\}$$
- **Μη επιλύσιμα** προβλήματα
  Π_{reg}: "Δεδομένης Μ.Τ. M, είναι $L^{(s)}(M)$ κανονική;"
  Π_{cf}: "Δεδομένης Μ.Τ. M, είναι $L^{(s)}(M)$ χωρίς συμφραζόμενα;"
  Π_{rec}: "Δεδομένης Μ.Τ. M, είναι $L^{(s)}(M)$ αναδρομική;"
- **Αναγωγή** του Π_2 στα Π_{reg}, Π_{cf}, και Π_{rec}.
  Έστω Μ.Τ. M. Θέλουμε να μάθουμε αν $M(\varepsilon)$ τερματίζει.
- M': $L^{(s)}(M') = \begin{cases} \emptyset & \textbf{κανονική} & \text{αν } M(\varepsilon) \text{ δεν τερματίζει} \\ H & \textbf{μη αναδρομική} & \text{αν } M(\varepsilon) \text{ τερματίζει} \end{cases}$
- Αλγόριθμος για "$L^{(s)}(M')$ κανονική;", αποφασίζει αν $M(\varepsilon)$ τερματίζει.
- $M'(w)$: Προσομοιώνει $M(\varepsilon)$. Στη συνέχεια, προσομοιώνει $U(w)$.

Γενίκευση: Θεώρημα του Rice

- $\mathcal{C}$ **μη-κενό γνήσιο** υποσύνολο όλων των αναδρομ. απαριθμ. γλωσσών.
  Μη επιλύσιμο πρόβλημα Π_{R}: "Δεδομένης Μ.Τ. M, είναι $L^{(s)}(M) \in \mathcal{C}$;"
- **Αναγωγή** του "$M(\varepsilon)$ τερματίζει;" στο Π_{R}.
- Έστω $\emptyset \notin \mathcal{C}$ (αλλιώς αποφασίζουμε ισοδύναμα αν $L^{(s)}(M) \notin \mathcal{C}$).
  Κάποια $L \in \mathcal{C}$ και Μ.Τ. M_L που ημιαποφασίζει L (δηλ. $L^{(s)}(M_L) \in \mathcal{C}$).
- Έστω Μ.Τ. M. Θέλουμε να μάθουμε αν $M(\varepsilon)$ τερματίζει.
- Ορίζουμε M': $L^{(s)}(M') = \begin{cases} \emptyset & \notin \mathcal{C} & \text{αν } M(\varepsilon) \text{ δεν τερματίζει} \\ L & \in \mathcal{C} & \text{αν } M(\varepsilon) \text{ τερματίζει} \end{cases}$
- Αλγόριθμος για "$L^{(s)}(M') \in \mathcal{C}$;", αποφασίζει αν $M(\varepsilon)$ τερματίζει.
- $M'(w)$: Προσομοιώνει $M(\varepsilon)$. Στη συνέχεια, προσομοιώνει $M_L(w)$.

Μηχανές Turing ως Απαριθμητές

- Μ.Τ. E **απαριθμεί** γλώσσα L ανν για 'ειδική" κατάσταση q της M:
  $$L = \{w : (s, \triangleright\sqcup) \vdash_E^* (q, \triangleright\sqcup w)\}$$
- Γλώσσα L (Turing-)**απαριθμήσιμη** ανν $\exists$ Μ.Τ. E που απαριθμεί L.
- E ξεκινά από αρχική κατάσταση s με κενή ταινία.
- E εισέρχεται περιοδικά σε "ειδική" **μη-τερματική** κατάσταση q.
- E σε κατάσταση q δηλώνει ότι **συμβ/ρά** στην ταινία **ανήκει στην** L.
  E αφήνει q και επανέρχεται με άλλο μέλος της L.
- **Όλες** οι συμβ/ρές της L (και μόνο αυτές) **παράγονται** με αυτό τον τρόπο.
  Μέλη L μπορεί να παρατεθούν με οποιαδήποτε σειρά και με επαναλήψεις.

Μηχανές Turing ως Απαριθμητές

- Γλώσσα L **αναδρομικά απαριθμήσιμη** ανν **Turing-απαριθμήσιμη** .
- ($\Rightarrow$) Έστω Μ.Τ. M που ημιαποφασίζει $L \in \Sigma^*$.
  Κατασκευάζουμε Μ.Τ. E **που απαριθμεί την** L:
  - Έστω $s_1, s_2, \ldots$ μια απαρίθμηση όλων των συμβ/ρών του Σ^*.
  - Για κάθε $i = 1, 2, 3, \ldots$:
    - E προσομοιώνει i **βήματα** M **για** i **πρώτες** συμβ/ρές $s_1, \ldots, s_i$.
    - Όταν $M(s_j)$ τερματίζει, E σε **"ειδική"** κατάστ. q με s_j στην ταινία.
- Κάθε w που $M(w)$ τερματίζει (άρα κάθε $w \in L$) απαριθμείται από E.
- Ο ορισμός της E σαν $M(s_1)\, M(s_2)\, M(s_3)\ \ldots$ δεν είναι σωστός!
  Αν $M(s_j)$ δεν τερματίζει, $s_{j+1}, s_{j+2}, \ldots$ δεν απαριθμούνται.

Μηχανές Turing ως Απαριθμητές

- Γλώσσα L **αναδρομικά απαριθμήσιμη** ανν **Turing-απαριθμήσιμη** .
- ($\Leftarrow$) Έστω Μ.Τ. E που απαριθμεί $L \in \Sigma^*$.
  Κατασκευάζουμε Μ.Τ. M **που ημιαποφασίζει την** L:
  - $M(w)$ **προσομοιώνει** E.
  - Όταν E απαριθμεί συμβ/ρά u, M συγκρίνει w με u.
    - Αν $w = u$, $M(w)$ **τερματίζει** (θα συμβεί ανν $w \in L$).
    - Αν $w \neq u$, $M(w)$ **συνεχίζει** προσομοίωση E.

Μηχανές Turing ως Λεξικογραφικοί Απαριθμητές

- Έστω Μ.Τ. E που απαριθμεί γλώσσα L.
- E **απαριθμεί λεξικογραφικά** L αν για 'ειδική" κατάσταση q της E:
  - όποτε $(q, \triangleright\sqcup w) \vdash_M^+ (q, \triangleright\sqcup w')$,
  - τότε w' βρίσκεται **λεξικογραφικά μετά το** w.
- Γλώσσα L (Turing-)**λεξικογραφικά απαριθμήσιμη** ανν
  υπάρχει Μ.Τ. E που απαριθμεί λεξικογραφικά την L.

Μηχανές Turing ως Λεξικογραφικοί Απαριθμητές

- Γλώσσα L **αναδρομική** ανν **Turing-λεξικογραφικά απαριθμήσιμη** .
- ($\Rightarrow$) Έστω Μ.Τ. M που αποφασίζει $L \in \Sigma^*$.
  Κατασκευάζουμε Μ.Τ. E **που απαριθμεί λεξικογραφικά την** L:
  - Έστω $s_1, s_2, \ldots$ λεξικογραφική απαρίθμηση συμβ/ρών Σ^*.
  - Για κάθε $i = 1, 2, 3, \ldots$, E **προσομοιώνει** $M(s_i)$.
    - Αν $M(s_i)$ αποδέχεται, E **απαριθμεί** s_i .
    - Αν $M(s_i)$ απορρίπτει, E συνεχίζει με $M(s_{i+1})$.
- ($\Leftarrow$) Έστω Μ.Τ. E που απαριθμεί λεξικογραφικά $L \in \Sigma^*$.
  Κατασκευάζουμε Μ.Τ. M **που αποφασίζει την** L:
  - $M(w)$ **προσομοιώνει** E .
  - Όταν E απαριθμεί συμβ/ρά u, M συγκρίνει w με u.
    - Αν $w = u$, M **αποδέχεται** w (θα συμβεί ανν $w \in L$).
    - Αν $w <_{\text{lex}} u$, $M(w)$ συνεχίζει προσομοίωση E.
    - Αν $w >_{\text{lex}} u$, M **απορρίπτει** w .