Aldea Alexia
grupa 144

Examen LFA

O mașină Turing (TM) este un model matematic ce constă dintr-o bandă (tape) de lungime infinită care este împărțită în celule, prin care este dat input-ul. Ea constă într-un head ~~care~~ care citește tape-ul și un registru de stare, care stochează starea mașinii Turing. După citirea unui caracter din input, acesta este înlocuit cu un altul, starea internă se schimbă și se mută de la o celulă la stânga sau la dreapta. Dacă Mașina Turing ajunge în starea finală, string-ul de input este acceptat, iar în caz contrar, acesta este respins.

O mașină Turing poate fi definită formal ca un 7-tuplu $(Q, X, \Sigma, \delta, q_0, B, F)$, unde:

- Q este un set finit de stări
- X este alfabetul de tape-uri
- Σ este alfabetul pentru input
- δ este o funcție de tranziție

  $\delta : Q \times X \rightarrow Q \times X \times \{Left\_shift, Right\_shift\}$
- q_0 este starea inițială
- B este caracterul blank
- F este setul de stări finale

O mașină Turing cu tape infinit dublu este similar cu o mașină Turing normală, dar tape-ul lui este infinit la stânga și la fel și în dreapta. Tape-ul este inițial umplut cu blankspace. Apoi se adaugă inputul și în continuarea acestuia este umplut cu blankspace-uri. Computația este definită, ~~de dinainte~~ ca de obicei, singura diferență fiind că head-ul nu întâlnește niciodată un sfârșit al tape-ului mișcân-du-se spre stânga.

O TM cu bandă infinită dublă poate simula o mașină Turing normală. Ea marchează capătul din stânga al input-ului ca să detecteze și să prevină head-ul din a depăși acel capăt. Pentru ~~a~~ a simula o TM cu tape dublu infinit cu o TM normală, arătăm cum să o simulăm cu o mașină Turing cu două ~~tape~~ tape-uri, care a fost deja demonstrată ca fiind echivalentă ca și putere cu 2 mașini Turing normale. Primul tape al TM cu 2 benzi este scris cu string-ul de input și al doilea tape este blank. Împărțim banda mașinii în 2 părți, în celula de start a string-ului de input. Porțiunea cu string-ul de input și toate spațiile blank din dreapta lui apar pe primul tape al TM cu 2 tape-uri. Porțiunea din stânga string-ului de input apare pe banda a doua, în ordine inversă.

TM infinită în ambele direcții:

- tape-ul infinit al TM infinită în ambele direcții este nelegat în ambele ~~direcții~~ direcții

- TM infinită în ambele direcții poate fi ~~simul~~ simulată de o mașină Turing standard

Putem totuși să observăm alte modele de computație și să vedem cum se pot compara cu alte ~~ma~~ TM. De exemplu, sunt varii metode prin care am putea mări puterea unei TM, cum ar fi o TM cu 2 tape-uri, cu un head de read/write pe fiecare tape. La fiecare pas al computației, o TM cu 2 tape-uri citește simbolul de sub capul de citire/scriere pe ~~ame~~ ambele benzi. Bazat pe aceste simboluri și pe starea sa curentă, poate scrie un nou simbol pe fiecare tape, să mute independent capul de read/write al fiecărui tape, o celulă la stânga sau la dreapta și să schimbe starea.

Ar putea părea că având 2 tape-uri la dispoziție, TM cu 2 tapes ar putea fi capabile să facă unele computații imposibile pt. TM obișnuite. De fapt, ~~tot~~ totuși, nu este cazul. Motivul este simularea: dându-se orice TM cu 2 tapes, este posibil să se construiască o TM cu 1 tape care ~~simula~~ simulează computația pas cu pas a TM-ului cu 2 tapes.

Fie M o TM cu 2 tapes. Pt. a-l simula pe M cu o mașină cu 1 tape, K, trebuie să stocăm conținutul ~~al~~ ambelor tape-uri ale lui M cu un singur tape și trebuie să ținem cont de pozițiile ~~head-ului~~ ambelor head-uri de read/write ale lui M.

Fie @ și $ 2 simboluri care nu fac parte din alfabetul lui M. @ va fi utilizat pentru a marca poziția unui cap de read/write, iar $ va fi utilizat în delimitarea celor 2 părți ce formează cele 2 tapes ale lui M. De exemplu, să presupunem că una dintre benzile lui M conține simbolurile ~~"#ab~~ "abb##cca" cu head-ul pe primul simbol b și că celălalt tape conține "01#111#001" cu head-ul pe ultimul 1. Această configurare va fi reprezentată pe cele 12 tapes astfel: "$a@bb##cca$01#111#00@1$". Pentru a simula un pas al lui M, K trebuie să scaneze întreaga bandă, în căutarea lui @ și să noteze simbolul de la ~~dr~~ dreapta lui @. Pe baza acestei informații K poate să își modifice tape-ul și propria stare pentru a replica noua configurație a lui M după un pas al computerului.

Evident, K va face mai mulți pași decât M și va opera mult mai încet, dar acest argument clarifică faptul că mașinile Turing cu 1 tape pot face orice este făcut de TM cu 2 tapes.