

SINTETIČNA METAFIZIKA

Sprejeto

10. 2. 2023

Pregledano

9. 10. 2023

Izdano

22. 12. 2023

ANDRAŽ BOLE

Univerza v Ljubljani, Ljubljana, Slovenija
bolelojs@gmail.com

DOPISNI AVTOR

bolelojs@gmail.com

Ključne besede

komputacija,
metafizika,
sinteza,
algoritem,
odločljivost

Izvleček V prispevku je obravnavana povezava komputacije in metafizike z dveh zornih kotov. V prvem delu so skozi zgodovinski pregled predstavljeni avtorji, ki so naredili ključni teoretični doprinos na poti do modernega razumevanja komputacije. V drugem delu pa je razdelan koncept “sintetične metafizike”, ki napeljuje na to, da je komputacija posplošena oblika vsake metafizike. Omejitve komputacije so hkrati končne omejitve metafizike, zato slednja v svoji najbolj dovršeni obliki ne more biti analitična.

SYNTHETIC METAPHYSICS

ANDRAŽ BOLE

University of Ljubljana, Ljubljana, Slovenia
bolelojs@gmail.comCORRESPONDING AUTHOR
bolelojs@gmail.comAccepted
10. 2. 2023Revised
9. 10. 2023Published
22. 12. 2023

Abstract The following article considers the connection between computation and metaphysics from two points of view. The first embraces a historical overview of authors who made key theoretical contributions to the modern understanding of computation. The second view elaborates on the concept of “synthetic metaphysics” which speculates that computation is a generalised form of every possible metaphysics. Limits of computation are limits of metaphysics so its most developed form cannot be analytical.

Keywords
computation,
metaphysics,
synthesis,
algorithm,
decidability

Matematika je ali preobsežna za človeški um, ali pa je človeški um nekaj drugega kot stroj.

(Goldblatt, 2014)

Takole je Kurta Gödela citiral Robert Goldblatt v svojem priročniku za matematično logiko. Na prvo oko bi pričakovali, da bo misel stekla v obe smeri. Da je matematika preobsežna za človeški um ali pa da je človeški um preobsežen za matematiko. Tako pa pridemo do nekakšne povezave med matematiko, umom in strojem, ki napeljuje na njihovo latentno divergentnost. Preobsežnost matematike je po Gödelovo vpeta v princip spoznavnih kapacitet človeškega uma. Če privzamemo, da je možno spoznavne kapacitete človeškega uma razumeti s principi stroja, kaj naj bi to pomenilo za obsežnost matematike?" Da le stroj, ki v principu izvaja sekvenčno razreševanje odločitvenega problema ("Decision problem", n.d.), lahko popiše ves njen izrazni prostor, ali da je matematika sama le univerzalna oblika vseh partikularnih strojev? Kaj v takšnem primeru sploh storiti z umom? Še večja zagata pa se pojavi, ko vpeljemo v vprašanje novo razmejitev med epistemološko in ontološko perspektivo. Če nekaj lahko razumemo in popišemo na matematičen način, je to daleč od trivialnega zaključka, da tudi samo *je* matematično v nekem Platoničnem smislu.

V središče teh divergentnosti postavljam pojem komputacije. Komputacija danes zaseda prestol v znanstveno-epistemološkem ogrodju bivajočega, v praktičnem in teoretičnem smislu. Njena prediktivna in opisna moč neusmiljeno koraka čez vse ostale metode in zdi se, kot da je kriterij njene učinkovitosti izrinil vse ostale načine opisov razumevanja. Če parafraziramo Quinea (Quine, 1948), je grobi povzetek komputacijske modernosti nekako takšen : "Da nekaj biva, mu mora biti dodeljena spremenljivka, kot rezultat komputacijskega postopka."

V pričujočem prispevku se bomo omenjenih divergentnosti lotili v dveh delih. Začeli bomo s pregledom ključnih avtorjev v zgodovini komputacije in njihovih filozofsko relevantnih teoretičnih izhodišč ter nato na tej osnovi predstavili koncept sintetične metafizike.

Preden začnemo s pregledom ključnih avtorjev, se lotimo za trenutek pojma komputacija in na kakšen način ga bomo uporabljali tekom celotnega prispevka. Oxfordski slovar definira komputacijo kot proces matematičnega računanja ("Computation", n.d.). Wolframova matematična knjižnica definira komputacijo kot proces, ki vodi v jasno določen rezultat z jasno določenimi pravili ("Computation",

2002). Zavaljo temeljitosti pogledimo še na kakšen način lahko slovarsko ogradimo pojem računanja ali kalkuliranja. Oxfordski slovar ponuja opis kot matematični postopek determiniranja števil ("Calculation", n.d.). Vsi trije nivoji opisov imajo nekakšno povezavo z manipuliranjem števil in sledenjem pravilom. Poskusimo jih združiti v bolj elegantno celoto: Komputacija je vsak primer aritmetičnega ali ne aritmetičnega manipuliranja vhodnih parametrov, ki vračajo rezultat po sekvenčno izvedenem zaporedju korakov. To bo naše delovno izhodišče za preostanek sestavka, kot osnova za pojem komputacije.

Sedaj pa nadaljujmo z vprašanjem začetka, ki ima tako kot pri vsakem zgodovinopisnem projektu tudi pri zgodovini komputacije močno konotacijo arbitrarne razmejitve. Zavedam se, da ne bomo uspeli v celoti izdelati pregleda vseh avtorjev, ki so tako ali drugače obravnavali števila in metafiziko. Izbrali pa bomo tiste avtorje, katerih doprinos razumem kot ključen pri razvoju glavne ideje, sintetične metafizike.

Začnimo najprej z omembo nekaj zgodovinskih stebrov. Začetki računanja in matematike kot sistema abstrakcij, vsaj kolikor jih lahko dosledno dokumentiramo, sodijo med 3. in 2. stoletje pred našim štetjem med začetke Mezopotamije. Še posebej Sumerci so imeli za naš premislek skrajno zanimiv odnos do števil, saj so za razliko od večino ostalih kultur skozi zgodovino uporabljali seksagesimalni sistem, temeljujoč na številu 60. Glavna motivacija za takšno, na prvi pogled ne-intuitivno odločitev, se je nahajala v izražanju racionalnih števil. Prav tako se je v 3. stoletju pred našim štetjem v njihovi kulturi prvič pojavil Abakus, instrument za pomoč pri računanju (Ifrah, 2002). Preprosta tablica z vrstami iz žice, ki nosijo kuglice, je omogočala, da so posamezniki lažje in hitreje reprezentirali števila ter z njimi manipulirali. S stališča pričujočega prispevka za takšno dejavnost še ne moremo uporabiti termina komputacija, kot smo ga zastavili na začetku, toda Abakus navsezadnje že predstavlja obliko proto-algoritma. To mislim na način, da se prvič pojavi priložnost, kako dosledno sledenje nekemu pravilu reprezentiranja privede do željenega izračuna ali pa se zadane ob končne izrazne oblike naprave - te so bile, tako kot danes, popolnoma fizikalne narave. Velikost tablice in z njo število vrst iz žice ter število kuglic so bile prvotne oblike tega, kar bi danes imenovali računalniški spomin.

Nadaljujmo s Pitagoro. V zahodni tradiciji filozofiranja Pitagorov vpliv predstavlja temelje spekulativnih izhodišč, ki so utemeljeni na številu, geometriji in njihovih simbolizmov. Kot je zapisal Aristoteles, naj bi Pitagorejci števila in njihovo organizacijo v matematični sistem prvi uporabljali v mistične namene, brez specifičnih nagnjenj k praktičnim aplikacijam, kot so to denimo počeli Sumerci, Babilonci in Egipčani pred njimi (Burkert, 1972). Sam bi nadgradil Aristotelesovo pripombo z mislijo, da je Pitagorejska razširitev števila dejansko ta, da mu dodeli mesto ontološke kategorije. Število ni več "samo" praktično orodje za manipulacijo okolja, kot je po večini to delovalo nekaj tisoč let prej v Rodovitnem Polmesecu, temveč postane ontološka kategorija, natančnejše model bivajočega. Ker bomo tekom sestavka večkrat naleteli na pojem modela, je smiselno opozoriti, da le-ta nosi različne zanimive konotacije, o katerih tu ne bomo razglabljali, vseeno pa pogledjmo na kakšen način bomo pojem uporabljali v našem kontekstu. Model je konsistenten sistem pravil kodiranja, ki se v času ne spreminjajo. Kodiranje pa je transformacija parametrov v števila, ki jih lahko po lastni volji naprej manipuliramo. Če trditvi združimo, je model torej konsistenten sistem pravil transformacije parametrov v števila, ki se v času ne spreminja. Abstrakcijo takšne trditve bi lahko razširili na bolj mehke pojme, kot recimo interpretacija ali pa celo jezik. Vsaka spoznavna spremenljivka zvedljiva na izražanje se vedno vede na način modeliranja, saj transformira ene parametre bivajočega v druge. S tem poudarkom, seveda, da z enimi lahko naprej manipuliramo, z drugimi pa (še) ne. S tega zornega kota lahko opazimo, da je možno kakršno koli obliko izražanja tretirati kot *nekakšno* vrsto komputacije, če se opremo na rabo pojma, kot smo jo opisali na začetku.

Toda pustimo to misel za kasneje in se vrnimo nazaj na Pitagorejce. Njihov ontološki model sveta je mistično oprt na števila. Narava stvari so bila števila. Monada, kot točka, Eno, stabilno božje iz katerega se manifestira mnoštvo. Dijada, je predstavljala dvojnost, razdvojenost, začetek mnoštva. Nekateri raziskovalci, kot denimo Charles Kahn iz Univerze v Pennsylvaniji, pravi, da je Dijado moč razumeti celo kot osnovo za materijo (Kahn, 2001). Trijada pa je predstavljala harmonijo, razsežnost števila 3, ki ima popolno ravnovesje, saj je edino število, katerih šesteti ali zmnoženi elementi vrnejo isti rezultat. Prav tako je edino število, ki samo sebe popolno razkosa na začetek, sredino in konec. Iz teh in še nekaj ostalih aritmetičnih in geometričnih reprezentacij se formira in levi celotno mnoštvo bivajočega. Pitagorejcem je formalno znanje matematike ponujalo trdno orodje za postavitev celotne numerologije, s katero so vzpostavljali kriterije vedenja na družbenem, kulturnem in tudi religioznem nivoju. Če pustimo ob strani kontroverzna stališča okoli izvora in

dokazov Pitagorovega izreka, je njihovo formalno znanje matematike nedvomno seglo zelo daleč. Od astronomije, teorije proporc, frekvenčnih razmerij v glasbi, itd. Ni dvoma, da so veliko kalkulirali oziroma opravljali aritmetične in geometrične operacije, toda komputacije kot sistema še niso razumeli.

Po Pitagorejcih sledi več stoletij razvoja praktičnega in teoretičnega dela matematičnih struktur in števil. Evklid, Arhimed, Diofant, ibn Musa al-Khwarizmi, Descartes, Pascal, Newton itn. Naj omenimo tudi Platona, ampak bolj kot opombo. V filozofiji matematike je zelo vplivno t.i. Platonistično stališče do matematičnih struktur. Čeprav Platon sam na to temo ni imel izrecno razdelanega stališča, se njegovo teorijo Idej nasploh jemlje kot interpretativno ogrodje na naslednji način. Števila obstajajo v ločeni domeni bivajočega na svoj, konkreten ontološki način. Njihova domena je drugačna, a bolj pristna od domene dostopne subjektom.

Prvi naslednji avtor, ki nas je pripeljal tik pred vrata komputacije in je hkrati tudi slutil na njeno intrinzično povezavo z metafiziko, pa je bil Leibniz. Bil je polimat, ki se je ukvarjal s celo vrsto panog, od filozofije, matematike, fizike, psihologije, družboslovja, celo pravom. Kar bomo obravnavali je ozek izrez iz njegove kreativne kariere, a po mojem mnenju najbolj pomemben, saj se dotika vseh vej njegovega angažiranja. Ko pomislimo na Leibnizov doprinos h matematiki in formalnim sistemom, se verjetno najprej spomnimo na njegov "spor" z Newtonom glede izvora matematične analize, ki sta jo oba s svojega konca razvila posebej. Toda ključna inovacija v razmišljanju v navezavi na kalkuliranje in komputacijo je ležala drugje. In sicer v njegovih dveh teoretičnih konceptih : *Machina ratiocinatrix* ter *Characteristica universalis*. Koncepta delujeta z roko v roki, saj prvi opisuje teoretični univerzalni računski aparat (kot nekakšen napreden abakus, ki avtomatsko premika kuglice), drugi pa opisuje univerzalni izrazni jezik tega računskega aparata. Okoli Machine ratiocinatrix obstaja več interpretativnih okvirov (Fearnley-Sander, 1982), v katere se ne bomo spuščali, niti za naš premislek niso pomembni. Kar je pomembno je le to, da dosledno izvaja vse dovoljene stavke poljubnega jezika, ki ga definira Characteristica universalis. Slednja ima globoko vizionarsko konotacijo univerzalnega programskega jezika, katerega je v celoti konsistentno izrazil šele Alan Turing 200 let za tem. O njem bomo povedali več malo kasneje. Ta koncept univerzalnega izražanja je imel centralno mesto v Leibnizovem mega projektu enciklopedije vsega vedenja, abecede človeške misli. Sistem simbolov, s katerim bi lahko brez dvoumnosti izrazil kakršne koli trditve o naravoslovju, matematiki, filozofiji, humanistiki itd. Kar je pomembno poudariti je to, da ni šlo "samo" za

formalni sistem sklepanja ali pa avtomatizacijo logike, ampak za ideografski sistem znakov, ki bi operiral na nivoju stvari (kot npr. kitajske pismenke) in ne izključno na nivoju znaka. Izredno zanimiva izvirnost takšnega projekta torej ni nekakšna predhodnica Fregejeve *Begriffsschrift*, temveč poskus avtomatizacije izražanja za epistemološko sintezo empiričnih znanosti, matematike, družboslovja, humanistike in metafizike s piktografijo. Še več, medtem ko je ugotavljal na kakšen način bi taka sinteza lahko dobila obliko, je s koncepti lastnega časa prišel do teoretične osnove moderne programske opreme. Binarni zapis, s katerimi lahko izrazimo katero koli število, bi bil reprezentiran s kuglicami - podobno kot pri Abakusu, celotna stanja pa bi "shranili" s sistemom luknjanih kartic. Vsaka lokacija je preluknjana ali pa ne, s čimer povzame prisotnost ali ne-prisotnost kuglice in s tem binarni zapis. Ta korak reprezentacije avtomata s specifično konfiguracijo razmerja kuglic je ključen pri premiku od surove kalkulacije h komputaciji kot (algoritično povezanemu) vhodno-izhodnem sistemu. Zakaj točno bomo videli kasneje pri Turingu, ampak na tej točki omenimo le to, da se s tem korakom odpre možnost manipulacije v času, saj se z vzpostavitvijo reprezentacije stanja omogoči tudi njegovo hranjenje - spomin.

Le špekuliramo lahko, ali je most do metafizike, katerega bomo omenili, nastal kot posledica ali kot vzrok njegovega enciklopedičnega projekta, ki bi temeljil na *Machine Ratiocinatrix* in *Characteristici universalis*. Toda dejstvo ostaja, da je v 5. in 6. delu *Diskurza o Metafiziki* nakazal na centralno stično točko med komputacijo in metafiziko. Trdi, da Bog hkrati maksimizira raznolikost in bogatost sveta ter minimizira konceptualno kompleksnost idej, s katerimi svet opisuje. Ideja se opira na primer, da za vsako končno množico točk vedno obstaja zakon, ki določa njihove pozicije. Toda, če so točke izbrane naključno, ti zakoni postajajo nemogoče kompleksni. Zakonitosti/formule/teorije morajo tako imeti končno kompleksnost, saj je v nasprotnem primeru sam koncept zakona popolnoma nesmiseln. Ta spekulativni uvid nosi resnično profetične konotacije, saj nehote odpre Pandorino skrinjico s vprašanjem: "Kakšna je dejanska razlika med kompleksnostjo in preprostostjo?" Na kakšen način lahko to razliko kvantificiramo, brez katere sploh ne moremo konkretno zamejiti ločnice med učinkovito ali neučinkovito teorijo/sistemom?

Trdim, da je epistemološka posledica *Machine Ratiocinatrix* in *Characteristici Universalis* ta, da so sistemi in teorije nekakšne vrste programi, ki izvajajo/komputirajo dejstva, razmerja in posledice s ključno dodatno distinkcijo:

manj kot imajo odvečnih elementov, bolj so natančni. Vsebujejo proces optimizacije, ki konvergira k stanju maksimalnega izločanja. Torej, da poskuša najti strukturo, od katere ne more izločiti ničesar več, brez da strukture ne spremeni v nekaj drugega. Povedano z drugimi besedami, če ne moremo opisati fenomena s krajšim opisom, kot je sam, ne moremo govoriti, da imamo opravka s spoznanjem. In, dalje, da vsaka sistemizacija poskuša opisati čim več s čim manj. Rečeno na tak način lahko opredelimo stališče tudi kot premik Occamove britve kot načela filozofiranja v načelo modeliranja kot spoznavni sistem. Če parafraziramo, je tako Leibnizova posredna posledica, da je nehoti spoznavni proces opisal kot vrsto kompresije. Osnovna izhodišča Leibnizove ideje so 250 let kasneje razvili Kolmogorov, Chaitin in Solomonoff v algoritmčno teorijo informacij (Chaitin, 1977), služila pa je tudi kot muza Zusejevi in Wheelerjevi digitalni fiziki ter Wolframovi, Chaitinovi in Fredkinovi digitalni filozofiji.

Sedaj pa smo prišli do zadnjega posameznika v prvem delu prispevka, ki je Leibnizovo delo realiziral v konkretnih okvirih in postavil ontološko osnovo komputaciji. To je Alan Turing. Preden ga vzamemo pod drobnogled, naredimo najprej kratek ovinek v kontekst problematike časa, znotraj katerega je Turing deloval. V 20ih letih 20. stoletja sta matematika Hilbert in Ackermann v popolni gesti kristalizacije duha časa zastavila izhodiščno tezo, da mora obstajati učinkovit algoritem ali metoda, s katerim lahko na univerzalen način dobimo binarni odgovor "da" in "ne" na katero koli matematično vprašanje, ter tudi ali je takšen odgovor možno izpeljati iz aksiomov s pomočjo logičnih zakonitosti. Poljubna problematika je v takšnem primeru s formaliziranim postopkom tako rekoč diskretizirana na potrditev ali ovrženje. Ta teza je bila del širšega meta-matematičnega projekta Hilberta, kjer je hotel pokazati, da je matematika konsistentna, kompletna in odločljiva. *Entscheidungsproblem* oziroma "problem odločitve" je predstavljal kulminacijo Hilbertovega ambicioznega programa, s katerim naj bi utemeljil matematiko kot čisto znanost in celotno polje matematičnega izpeljal iz končnega nabora dokazljivo konsistentnih aksiomov. Turingovo delo se je dotikalo v bistvu le ene, a ključne besedne zveze Hilbertove in Ackermannove teze. In sicer, kako izgleda učinkovit algoritem, ki na poljubno matematično trditev odgovori z Da ali Ne.

V svojem legendarnem prispevku '*On computable numbers, with an application to the Entscheidungsproblem*' je Turing zastavil teoretični koncept t.i. Turingovega stroja, TS - Turing ga v članku sicer imenuje računski stroj (Turing, 1937). Njegova ideja je bila poskus razstavljanja koncepta metode na jasne elementarne korake, kjer lahko

vsakega od njih preverimo za učinkovito razrešitev odločitvenega problema. Čeprav je Turingov stroj po simboliki avtomatizacije stroj, je Turingovo izhodišče mehanizacija najbolj splošnega postopka, ki ga lahko navsezadnje izvaja človek s papirjem in pisalom. Pričel je celo z otroško slikanico, ki je služila kot osnova za idealizacijo postopka, ki opisuje, kaj lahko doseže človeški um, če dosledno sledi izvajanju pravil. Čeprav je bil teoretični koncept Turingovega stroja namenjen abstrakciji, je navsezadnje še vedno predstavljal zasnovo za potencialno praktično realizacijo. Osnova za okvir prispodobne je bil t.i. teleprinter, naprava ki so jo v Turingovem času uporabljali za pošiljanje in sprejemanje sporočil na daljavo. Ključne lastnosti Turingovega stroja so: deluje na neskončno dolgem traku, ki je razdeljen na celice z binarno zapisano vsebino. Nad trakom deluje t.i. bralna glava, ki se premika preko celic na traku. Na njih izvaja osnovne aritmetične operacije tako, da v skladu z zatečenim internim stanjem stroja in vsebino prebrane celice spreminja vrednost v celici in interno stanje stroja. Vedno ima dostop do stanja stroja (izračunanega na prejšnjem koraku) in vrednostjo v prebrani celici. Ključna poanta je tu ta, da se ne sme/more ustaviti. Za nevtrarno delovanje mora po domače rečeno neprestano "delati nič".

Takoj lahko opazimo dva zanimiva uvida Turingovega razmisleka o pojmovanju učinkovite formalizacije metode. Prvi je ta, da je trak oziroma spomin stroja neskončen in da je čas neomejen. Takšna poteza je bila narejena z mislijo, da se potencialne definicije komputacije ne bi zožale zaradi enostavnih praktičnih omejitev naših sposobnosti repliciranja Turingovih strojev v realnosti. Omogoči to, da ne obstaja nobena izračunljiva funkcija, ki je Turingov stroj ne bi mogel izračunati izključno in samo zaradi tega, ker je omejen z resursi. Rečeno drugače, vsakršen problem, ki ga lahko zapišemo na končnem traku poljubne velikosti, je izračunljiv. Kar pa seveda ne pomeni nujno, da funkcij, ki so izračunljive, ampak ne s Turingovim strojem, ni. Drug zanimiv uvid na tej točki opisa pa je dejstvo, da se Turingov stroj nahaja v diskretiziranem prostoru in času. Prostor premikanja - trak, je razdeljen na celice, med katerimi se ne nahaja ničesar, prav tako pa je elementarna časovna enota premikanja stroja definirana s premikom bralne glave med celicami.

S takšnim konceptom stroja se je Turing lotil velikanskega izziva, ki ga je narekoval *Entscheidungsproblem*. Lotil se je horizonta definicije izračunljivega, oziroma kakšna števila so v principu izračunljiva, da bi lahko dalje osnoval ali ovrigel splošnost učinkovite metode logičnega sklepanja. Izračunljiva števila, kot jih je definirali Turing, so tista realna števila, ki so izračunljiva s končnimi sredstvi: "*Izračunljiva števila lahko*

na kratko opišemo kot tista realna števila, katerih izrazi $\approx$ decimalnim mestom so izračunljivi s končnimi sredstvi? (Turing, 1937, str.1). Oziroma natančneje, števila katerih števila lahko izpiše stroj. Omenimo pomembno nianso, da izpis s končnimi sredstvi ne pomeni, da takšen stroj ni sposoben izpisa realnega števila z neskončno decimalnimi mesti. To pa zato, ker lahko zapis takšnega števila naredimo v obliki funkcije, ki zakodira njegovo obliko tako, da je oblika sama zapisana s končnim številom konfiguracij, rezultat izvajanja pa teče v nedogled (na primer, število π). Tu leži ključna poanta algoritma kot metode iskanja. Ko nekaj algoritmiziramo, od Turinga dalje, v osnovi postavljamo razvejevalne kriterije metode iskanja, ki po neki poljubni iskalni domeni izloča elemente. Tisto kar ostane je rezultat. In teh kriterijev je vedno končno mnogo, medtem ko je izločenih elementov lahko neskončno mnogo - metoda lahko deluje v nedogled, dokler seveda ne zadane v fizikalne omejitve delovanja. Celo idejo Turingovega stroja lahko dejansko z moderno govorico razumemo kot nekakšen računalnik, ki je narejen zato, da izvaja le točno določeno nalogo. Koncept računalnika kot ga poznamo danes je bližje ideji t.i. univerzalnega Turingovega stroja - UTS, ki je posplošena ideja navadnega Turingovega stroja. Ima to razliko, da je programabilen oziroma da lahko za vnos na traku (poleg podatkov) vzame poljubno obliko Turingovega stroja. Rečeno drugače, lahko simulira katerikoli možno obliko navadnega Turingovega stroja. Njegovo komputacijsko funkcionalnost lahko prosto prilagajamo z različno formalno semantiko (zapisom na traku) - v današnjem smislu z različnimi programskimi jeziki - in tako rešujemo poljubne komputacijske probleme.

Na tej osnovi pa stoji Turingov zaključek, s katerim je odgovoril na *Entscheidungsproblem* z odločnim "Ne". Ne obstaja, niti ne more obstajati v principu, znotraj aksiomov matematike kot jo poznamo, nobena učinkovita metoda, s katero lahko poljubno formalno trditev zvedemo na binarno drevo odločanja. V jedru tega odgovora se nahaja t.i. "Halting problem". Gre za spekulacijo ali lahko v splošnem najdemo takšno konfiguracijo Turingovega stroja, ki bo odločila za katerikoli drugi računski stroj in vnosni niz ali se kadarkoli ustavi s sekvenčnim računskim procesiranjem ali ne. Turing je dokaz izpeljal z razširitvijo Cantorjevega diagonalizacijskega argumenta na izračunljiva realna števila in pokazal, da takšen hipotetični računski stroj ne more izračunati niti lastnega opisa, kaj šele iskanega diagonalnega števila po Cantorjevem načelu. (Chaitin, 2007, str.81-84)

Problem kontinuuma realnih števil, na katerega je opozoril Cantor, napeljuje na količino raznolikosti v neštevniosti, zaradi katere je nemogoče vzpostaviti splošen kriterij razvrstitve - korespondenco z naravnimi števili. Kontinuum je torej tako velik, da s sekvenčnim postopkom nikoli ne moremo priti do vseh elementov, tudi v neskončnem času, čeprav jasno vidimo, da morajo obstajati. Kar pomeni tudi, da je večina števil kontinuuma izven domene iskanja Turingovih strojev in tako izven domene vsakega "učinkovitega algoritma". Kot smo že omenili, je Turingovo izhodišče, da je število izračunljivo, če obstaja način, da ga najdemo s končnimi sredstvi. Turingovi stroji pa so konceptualni okvir vsakega algoritma, ki je splošna implementacija "končnega sredstva" za poljuben primer iskanja. Problem kontinuuma nam že tako odpre težko spoznavno izhodišče, da objekt izgubi ontološki primat, saj je "le" posledica konfiguracije opisa v končnem številu korakov. Rečeno drugače, objekt je karkoli je možno opisati končnim številom konfiguracij. Ontološka perspektiva se sintetizira s spoznavno, saj objekt biva kot posledica omejenosti informacije. "Halting problem" pa razkriva še bolj radikalno stališče, da tudi znotraj števil, za katere imamo algoritem in jih lahko v principu preštejemo dosledno natančno, obstajajo elementi, ki se nam bodo izmaknili. Objekt kot tak, kakorkoli bivajoč, nikoli ne more biti kompleten. Oziroma, je presekan. Kompleten je lahko le naivno, kot posledica pragmatičnega kompromisa.

Globinska posledica tega dela je, da logika prvega reda v splošnem ni odločljiva. Oziroma tudi če bi bila, za njo ne bi mogla obstajati učinkovita sekvenčna metoda s katero bi lahko odločljivost analitično našli/utemeljili. Pokaže, da obstajajo matematične trditve, ki so neizračunljive. Ostajajo neodločene. Veliko vlogo pri tem uvidu, ki je zdesetkal Hilbertov program, je imel tudi Kurt Godel s svojim delom na completeness in consistency formalnih sistemov, toda njegovo mesto bomo obravnavali kdaj drugič. Že omenjeni Gregory Chaitin pa je par desetletij kasneje pokazal, da dejansko neodločljive trditve niso izjeme v matematičnem vesolju temveč, da je velika večina matematičnih trditev nezvedljiva s sekvenčnim postopkom komputacije ("Chaitin's constant", n.d.).

V svojem doktorskem raziskovanju se je Turing bolj posvetil teoretični spekulaciji neizračunljivega in implikacij, ki jih je prinesel "Halting problem". Prišlo je do nekakšnega premika v razmisleku o človeškem umu od sledenja pravilom k fascinaciji s stanji uma, ko ne sledi pravilom. Jedrno točko teh premislekov je po mojem mnenju najbolje povzel s konceptom Oraklja, t.i. "Oracle", za katerega je Turing določil prav neposredno čarobne sposobnosti. Koncept klasičnih Turingovih

strojev je razširil z novo mogočo instrukcijo, ki se preprosto imenuje "kliči orakelj", s katerim omogoča stroju "posvetovanje" z orakljem. Ta opravlja funkcijo analiziranja ("računanja") neizračunljivega in določene korake "preskoči" na poti do zaključnega koraka izračuna. Do rezultata pride na nejasen način. Njihova vloga je v tem, da so postavili ogrodje premisleku idejnega prostora procesiranja stanj, katerih se ne da izvajati izključno z mehanskim postopkom kalkuliranja, t.i. hiperkomputaciji. To lahko ponazorim z vzporednimi ravninami elementov, ki imajo celoštevilčne koordinate. Z UTS pridemo od poljubnega elementa v katerikoli od teh ravnin do poljubnega elementa v isti ravnini, ne pa od poljubnega elementa ene ravnine do kateregakoli elementa v drugi ravnini, čeprav imata oba celoštevilčne koordinate. Orakelj omogoča preskoke med temi ravninami. Kot je povedal sam : *"Ne bomo se spuščali dlje v naravo tega Oraklja, razen v dejstvo, da ne more biti stroj"* (Turing, 1939, p.173). Neizračunljivo torej ni po definiciji nedostopno, gre le za to, da je nedostopno postopku komputacije, ki je končno omejeno s principom Univerzalnega Turingovega Stroja. Orakelj odpira vprašanje relativne in ne absolutne komputacije, ki bi jo na drugačen, bolj pogovorni način lahko imenovali tudi intuicija ali neke vrste spontani uvid. Implikacija takšnega premisleka za funkcionalno spoznavno sposobnost je v tem, da se zdi, kot da so ljudem dostopni deli formalnih sistemov, ki jih stroji nikoli ne bodo mogli doseči - vsaj dokler privzemamo trenutne sisteme formalne logike in bodo stroji operirali na ontologiji Univerzalnih Turingovih Strojov.

Toda ta sklep je morda preveč dvoumen, saj ni jasno kaj zares storiti s principom intuicije. Le-ta res lahko funkcionira kot sintetični generator *novuma*, ki vzpostavi mostove med pred tem nepovezanimi entitetami (Chaitin, 2007, str.320). Toda takšna aktivnost je lahko tudi posledica nezavednih procesov, za katere lahko čisto plavzibilno trdimo, spet v spekulativnem duhu, da funkcionirajo kot sekvenčni proces, ki enostavno vrača rezultat preko spoznavne tančice. Vrne rezultat, ki se nam zdi kot rezulat intuicije - nejasen, vendar le zato, ker ne vidimo postopka s katerim je bil generiran. Seveda pa lahko spekuliramo tudi v drugo smer, da nam nedostopni deli našega spoznavnega instrumenta delujejo na način Turingovega oraklja in so pol-mehanska-pol-nekaj-drugega prikazen, ki procesiranje signalov ne izvaja dosledno na način sekvenčno izvedljivih korakov.

Koncept Turingovega stroja (posledično tudi UTS, saj lahko simulira vsak TS) nudi torej ontološko osnovo vsakršni komputaciji. Komputacija je vsak postopek izraza števila, ki ga izvaja UTS. Ta princip je Turing s svojim doktorskim mentorjem

Alansom Churchom formaliziral v t.i. Church-Turingovi tezi : funkcija nad naravnimi števili je učinkovito izračunljiva izključno in samo, če je izračunljiva za UTS ("Church-Turing thesis", n.d.). Zanimiva plat komputacije pa je ta, da za njo obstaja več možnih ontoloških osnov. Poleg TS, poznamo še Churchov Lambda račun, Godelove rekurzivne funkcije, celične avtomate... in, kar je morda še bolj zanimivo, vse ontologije komputacije so med seboj ekvivalentne! Izrazijo in najdejo lahko popolnoma isto množico izračunljivih trditev (števil), razlikujejo se med seboj le v tem, da so v določenih nalogah počasnejše/hitrejše od ostalih.

Sedaj smo obdelali ključne točke za nadaljni razmislek in s tem odprli pot za drugi del, kjer bomo predstavili koncept "sintetične metafizike". Najprej nekaj opomb na pojem sintetičnosti, da razčistimo morebitne predsodke. S sintetičnim vsekakor ne mislim pomena, ki se navezuje na tehnično raven proizvedeno s pomočjo postopka kemične sinteze. Niti ne mislim na sintetično v splošnem načinu filozofiranja kot predikat, ki je mediiran z neko konfiguracijo izkustva bivajočega. S sintetičnim pojmem interaktiranje med koncepti, ki ima fundamentalno naravo združevanja. Kot cela vrsta avtorjev skozi zgodovino filozofije, se bom tudi jaz oprl na Predsokratike, da postavim temelje za sintezo svojega izhodišča. Toda na malo drugačen način. Temelji sintetične metafizike se opirajo na komputacijo. Natančneje, sintetična metafizika se ponaša kot metafizika posplošena s komputacijskimi principi. In trdim, da je ključno ontologijo komputacije moč razbrati že pri Predsokratikih. Toda izhodišča ne bomo utemeljili na enem samem Predsokratiku, ampak na branju treh Predsokratikov skupaj, kot tri različne plati istega lika. Karikiramo lahko, da je osnovni akt sintetične metafizike že osnovan na združevalnem branju konceptov Heraklita, Parmenida in Demokrita. Naj še poudarim, zavedam se, da za vsakega od njih obstaja cela vrsta različnih interpretacij. Prav tako se zavedam, da pri takšnem branju ostanejo elementi posameznih del, ki ne spadajo v nov-predlagan pogled na proces sintetiziranja metafizike. Ampak vztrajam, da so jedrne ideje vseh treh ohranjene dovolj pristno, ne glede na to kakšno interpretacijo bi vzeli in da elementi njihovih opusov, ki bodo izvzeti, ne bi ničesar drastično spremenili.

Če se osredotočimo na Turingove stroje so v osnovi vedno prisotne tri značilnosti : a) statična oblika ogrodka zapisa ukaza, b) neprestani proces premikanja oziroma zamenjave stanj ter c) jezik, ki zavzema binarno obliko zapisa. Prva, permanentna statična oblika ogrodka zapisa je določljivost algoritma oziroma matematičnega ukaza. Ta funkcijska domena pripada Parmenidu. Poljuben algoritmčni ali funkcijski

zapis, ki narekuje način iskanja, je znotraj lastne dimenzije produciranja, več in nespremenljiv. Kot primer, denimo algoritmičen zapis s katerim izpišemo vsa naravna števila. Je najbolj strnjena oblika kompleksnosti, ki se hkrati nikoli ne spremeni, čeprav nikoli ne sprodi enakega rezultata. Ima končno število znakov, ki bo vračalo neskončno število znakov (izključujoč fizikalne omejitve seveda). Je zaključeno, Eno in ne-deljeno. Opisuje kompleksnost, katerih večine elementov sama sploh ne vsebuje, čeprav vsakega od njih popisuje. Tudi zablodna pot smrtnikov iz Parmenidove pesmi, ki maskira z imeni in je nepristna prikazen Resničnega se lepo ujema s tem ogrožjem TS. Maskirani elementi "nepristnega" so preprosto rezultati iskanja, ki jih vrne "večni" algoritmični zapis. Pot zablode, ki trdi, da stvari minevajo, se iz nič manifestirajo v bit in nato nazaj ven iz nje, je izomorfna dejstvu, da so rezultati iskanja razvrščeni. Kar pomeni, da imajo mesto pred in za poljubno izbrano lokacijo na traku. Ne prisostvujejo v Bivajočem, dokler jih metoda iskanja ne izpiše in morebiti kasneje izbriše in integrira v novo stanje, toda skozi metodo so vedno že nekako latentno prisotni. Prav tako se Parmenidov odnos do ne-bivajočega, kot izločenega iz množice spoznavnega ujema z imperativom, da ukaz ne more biti prazen. Fundamentalna narava ukaza je v tem, da mora vsebovati najmanj lastno načelo izvedbe, v obliki "delaj-nič".

S to točko pa omenimo drugo dinamiko TS in obenem drugega omenjenega Predsokratika, Heraklita. Kot rečeno v zadnji trditvi o Parmenidu, stanje ničelne spremembe se mora konstantno izvajati na-novo, tako kot vsaka druga sprememba. Vsako interno stanje TS se definira kot kriterij spremembe do vseh ostalih spremenjenih stanj in kjer je tudi mirovanje, le še ena od oblik spreminjanja. TS se nikoli ne ustavi (seveda, ponovno, razen v primeru fizikalnih omejitev), tudi ko čaka, si mora predajati ukaz, za izvajanje prazne zanke.

Prekrivanje s Parmenidom ima še eno dimenzijo, in sicer Heraklitov pogled na unifikacije nasprotij. Bivajoče hkrati je in ni. Heraklitov poskus, da nas s takšnimi trditvami poskuša zdramiti iz dogmatičnega spanca je zelo uspešen. Lastnost razmerja med algoritmičnim zapisom in rezultatom, ki ga vrača ima natanko takšno razmerje. V kontekstu TS, kako misliti obstoj števila, ki ga ni neposredno v kratkem algoritmičnem zapisu, a ga ta še ni vrnil, čeprav ga enkrat bo, saj se nahaja v njegovi iskalni domeni? Zdi se kot da gre za podobno opombo od Romana Ingardna o obstoju glasbenega dela - ali le-to obstaja že v notni zapis, šele, ko se izvaja, ko je poslušan...? Tretji Heraklitov element, ki je izomorfen z drugo dinamiko TS pa je koncept Kozmosa kot kriterija organiziranosti in urejenosti Bivajočega. Metodčnost

in sekvenčna organizacija je ključni element urejenosti delovanja vsakega TS. Prehodi med stanji in stanja imajo jasno sosledico in morajo biti logično-reverzibilna, brez preskokov.

Zadnja dimenzija TS, jezik v obliki binarnega zapisa, pa je seveda ločljivost Demokrita in njegovih naturalističnih principov atomov ter praznine. Aspekt dvojega, ki se združuje v bolj in bolj elaborirane elemente, ki se dalje formirajo na svoje načine je centralna poanta Demokritovega in Levkipovega atomizma. Stoji v ostrem nasprotju z Zenonovo in Parmenidovo linijo razmišljanja, ki trdi, da praznina ne more imeti mesta v bivajočem oziroma, če ga ima, temelji na paradoksu. Praznina ne more biti, saj če je, ni praznina. Toda praznina, kot jo od Demokrita jemljemo tu, je morda semantično nekaj drugega, kot to pojmuje Parmenid. Slednji jo pojmuje v absolutnem smislu, medtem ko se Demokrit do praznine vede bolj na način negativa stvari (Diels & Kranz, 1951, 67A6). Negativ stvari je tisto, kar stvari omogoča biti to, kar je. Praznina je oblika razsežnosti, skozi katero se atomi lahko manifestirajo. Praznina in elementi se neprestano formirajo v mnogote oblike kompleksnosti, ki pa s časom razpadejo nazaj na enake elementarne binarne pare. Princip binarnega zapisa na traku TS ima preprosto posledico, in sicer, da je iz dveh znakov možno generirati katerokoli število in s tem poljubno bogat jezik. Za takšno konstrukcijo potrebujemo veljavno abecedo in nabor pravil, ki se izvajajo na njej. TS lahko obravnavajo poljubno število abeced s poljubnim naborem pravil (ki so v končnem ponovno omejene s fizikalnimi kriteriji). Moderni računalnik, ki je mnogo bližje UTS, lahko implementira celo vrsto različnih jezikov in abeced z nabori pravil, ki so vsi t.i. Turing Complete (lahko izrazijo vse, kar zmore izraziti UTS). Operirajo na različnih nivojih abstrakcije, eni bližje, eni dlje neposrednemu ukazu stroja, kot bi to v teoriji izvajali pri TS, ki ni poznal še nobenih implementabilnih principov programske opreme ali operacijskega sistema. Toda ideja je v tem, da nivoji abstrakcije ponovno nosijo določen element pristnosti, ki se zmanjšuje z oddaljenostjo od neposrednih ukazov stroju - kar je zelo priljubljen princip skozi celotno zgodovino filozofije, dlje kot je nekaj od substance, bolj kot je sestavljeno, manj je pristno. Rečeno drugače, sestavljeni elementi (generirani jeziki z abecedami) so prikazni, pristno bivajoči elementi so atomi in praznina (binarna znaka zapisa). Sam bi rekel, da se pristnost nahaja bolj v perspektivi optimizacije ciljne funkcije. Po Demokritu, Parmenidu, Heraklitu ter še marsikom drugim, bi ohranjanje pristnosti pomenilo premikanje toka na tranzistorjih. Na tak - pristen način bi seveda prav tako lahko spisali to besedilo, samo bi potrebovali verjetno nekaj let, namesto da bi

uporabili urejevalnik besedilana nekem višjem nivoju abstrakcije in se nahajali v dimenziji prikazni.

Na koncu verige se vsi jeziki s svojimi abecedami transformirajo v binarni zapis, s katerim se izvaja sekvenčna komputacija. V tem primeru lahko celo dobesedno naredimo paralelo in rečemo, da se dva znaka najprej formulirata v večje in večje kompleksnosti jezikov, ki zmorejo tudi abstrakcije izražanja preko sebe - izvajata združene koncepte, katerih ponovno ni v regularni abecedi, a jih z njo lahko v celoti sestavimo. Na koncu pa se vedno zreducirajo nazaj na binarni znakovni zapis (ko število izračunamo), tako kot se vse kompleksnosti, po Demokritovo in Levkipovo vedno sčasoma vrnejo nazaj na atome in prazino.

Takole orisano omrežje Parmenida, Heraklita in Demokrita nam torej služi kot okvir za idejo, da je komputacija tisti lik, katerega različne plati so v splošnem zavzemali vsak od njih. Je primer najkrajšega opisa sintetičnega združevanja in predstavljam ga kot izhodišče trditve, da je komputacija posplošena metafizika s katero so se - nevedoč, ukvarjali že na samem začetku zgodovine filozofije. Z besedno zvezo posplošena metafizika ciljam na to, da je možno s principi komputacije opisati vsakršno metafiziko, vključujoč samo sebe. Seveda je tudi takšna trditev metafizična trditev. Leibniz, Turing, Godel, Kolmogorov, Chaitin...itd. so sicer operirali med števili in matematičnimi strukturami, toda takšno delovanje je v principu prav tako metafizično. Zato pojma komputacije in metafizike ne zoperstavljam, ampak združujem v posplošitvi. Trditve o eni ne gre zanemarjati v drugi in obratno. Vsako vzorčenje Bivajočega, od prebliska naključnega vtisa, misli, do modelov in sistemov so tako oblike algoritmov, metode iskanja na neki domeni, ki v presečišču z domenskimi elementi proizvedejo subjekt, objekt in njuno razmerje. Sintetična metafizika išče najkrajši univerzalen sistem opisov, ki obsega vse možne partikularne opise bivajočega - ki so sami za sebe, ponovno spet algoritmi. Sintetična metafizika poskuša simulirati vse ostale algoritme - metafizike, tako kot UTS lahko simulira kateri-koli TS. Tiste o Resničnem, o Neresničnem, o Dobrem, o Svetem, o Nedoločenem,...ne kot teorija vsega, ampak kot plastična metoda iskanja, ki neprestano sprejema nove vektorje pogledov in jih poskuša zaobjeti v opisu. Vključno z vsemi prejšnjimi verzijami sebe. Zdi se, da nam ne preostane drugega, kot da se v permanentnem procesu poskušamo približevati razmejivni med kompleksnostjo in preprostostjo, o kateri je pisal Leibniz. Univerzalne metode za njeno iskanje ni, vsaj ne na sekvenčno metodičen način. Lahko rekurzivno bogatimo sisteme opisov s čim večjim številom različnih variacij ter jih poskušamo združiti v

najkrajši možni algoritem. Če pa razlike v velikosti ni, ne moremo govoriti o spoznanju, saj smo prišli do nečesa, kar je zaobjeto že v obstoječem algoritmu in je le njegovo prebesedenje.

Lahko bi rekli, da je celotna zgodovina poskus iskanja algoritmov v nekakšno obliko univerzalnega algoritma, ekvivalent UTS, ki bo zmožen simuliranja vseh možnih metafizičnih sistemov. Uperjen bo v Absolut, toda še vedno bo imel pred seboj nerazrešljivo zagato. Če bo do tja hotel priti na sekvenčni - algoritemski način analitične metodike, bo po Turingovem, Godelovem in Chaitinovem uvidu, ugotovil, da ne samo, da ne more biti Absolut, temveč tudi, da je Absolut zanj nedoločljiv. Za preseganje nedoločljivosti sistema Absoluta pa bo moral seči po kompromisu - metafizičnemu oraklu, ki ne bo ustrezal sekvenčnemu načinu analitične metodike.

Literatura

- Chaitin, G. J. (1977). Algorithmic Information Theory. *IBM Journal of Research and Development*, 21(4), 350–359.
<https://doi.org/10.1147/rd.214.0350>
- Professor Emeritus of Classics Walter Burkert, Burkert, W., & Harvard University Press. (1972). *Lore and science in ancient pythagoreanism* (str. 467 - 468). Harvard University Press.
- Calculation. (n.d.). *Oxford Learner's Dictionaries* | Find definitions, translations, and grammar explanations at Oxford Learner's Dictionaries.
<https://www.oxfordlearnersdictionaries.com/definition/english/calculation>
- Chaitin, G. J. (2007). *Thinking about Godel and Turing: Essays on complexity*, 1970-2007L. World Scientific.
- Chaitin's Constant (n.d.). Wolfram MathWorld: The Web's Most Extensive Mathematics Resource.
<https://mathworld.wolfram.com/ChaitinsConstant.html>
- Church-Turing Thesis (n.d.). Wolfram MathWorld: The Web's Most Extensive Mathematics Resource.
<https://mathworld.wolfram.com/Church-TuringThesis.html>
- Computation. (n.d.). *Oxford Learner's Dictionaries* | Find definitions, translations, and grammar explanations at Oxford Learner's Dictionaries.
<https://www.oxfordlearnersdictionaries.com/definition/english/computation>
- Computation. (2002). Wolfram MathWorld: The Web's Most Extensive Mathematics Resource.
<https://mathworld.wolfram.com/Computation.html>
- Decision Problem (n.d.). Wolfram MathWorld: The Web's Most Extensive Mathematics Resource
<https://mathworld.wolfram.com/DecisionProblem.html>
- Diels, H., & Kranz, W. (1951). *Die Fragmente Der Vorsokratiker: Griechisch und Deutsch*.
- Fearnley-Sander, D. (1982). Hermann Grassmann and the prehistory of universal algebra. *The American Mathematical Monthly*, 89(3), 161 - 166.
<https://doi.org/10.2307/2320198>
- Goldblatt, R. (2014). *Topoi: The categorical analysis of logic*. Elsevier.
- Ifrah, G. (2002). *The universal history of computing: From the abacus to the quantum computer* (str. 11). Wiley.
- Kahn, C. H. (2001). *Pythagoras and the pythagoreans: A brief history*. Hackett Publishing Company.
- Turing, A. M. (1937). On computable numbers, with an application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society*, s2-42(1), 230-265.
<https://doi.org/10.1112/plms/s2-42.1.230>

Turing, A. M. (1939). Systems of logic based on Ordinals. *Proceedings of the London Mathematical Society*, s2-45(1), 161-228.

<https://doi.org/10.1112/plms/s2-45.1.161>

Quine, W. V. (1948). On what there is. *Review of metaphysics*, 2(5), 21-38.