Gödeliläinen argumentti tekoälyä vastaan

Paavo Maanpää

Pro gradu -tutkielma

Turun yliopisto

Filosofian, poliittisen historian ja valtio-opin laitos

Filosofia

Marraskuu 2024

TURUN YLIOPISTO

Filosofian, poliittisen historian ja valtio-opin laitos / Yhteiskuntatieteellinen tiedekunta

MAANPÄÄ, PAAVO: Gödeliläinen argumentti tekoälyä vastaan

Pro gradu -tutkielma, 46 s.

Filosofia

Elokuu 2024

Gödelin epätäydellisyyslauseen mukaan jokainen aritmetiikkaan kykenevä tietokoneohjelma sisältää sellaisen toden väitteen, jota tuo ohjelma ei voi itse osoittaa todeksi. Tutkielmani koskee gödeliläistä argumenttia, joka pyrkii osoittamaan, että ihmismieltä ei pystytä simuloimaan tietokoneella. Jos nimittäin tuollainen kone olisi olemassa, voisi ihminen löytää koneen arkkitehtuurista sellaisen ratkeamattoman väitteen, jonka ihminen tietää todeksi, mutta kone ei. Tällöin tekoäly ei pystyisi milloinkaan kaikkeen samaan kuin ihminen.

Päälähteinäni ovat gödeliläisen argumentin keskeisimmät kannattajat: John Lucasin *Minds, Machines and Gödel* (1961) ja Roger Penrosen *Shadows of the Mind* (1994). Tärkeimmät työssä hyödynnetyt kommentaarit ovat David Chalmersin *Minds, Machines, and Mathematics* (1995), Stewart Shapiron *Incompleteness, Mechanism, and Optimism* (1998) ja Jason Megillin *The Lucas-Penrose Argument about Gödel's Theorem* (2024).

Tutkielmassani käy ilmi, ettei Gödelin epätäydellisyyslauseesta johdettu argumentti ole yhtä voimakas kuin sen kannattajat haluavat ymmärtää. Se perustuu osittain fantastisille oletuksille ihmisen matemaattisesta kaikkivoipaisuudesta, joita mekanistisen teesin kannattajan ei tarvitse hyväksyä. Mekanistisen teesin kannattaja, joka uskoo ihmismielen olevan simuloitavissa tietokoneella, joutuu kyllä tekemään tiettyjä kompromisseja: hän joutuu joko myöntämään, että on olemassa sellaisia matematiikan väitteitä, joita ihminen ei kykene edes periaatteessa osoittamaan todeksi. Tai mekanisti voi sanoa, että ihmistä simuloiva tietokoneohjelma on olemassa, mutta emme sen monimutkaisuuden vuoksi kykene ymmärtämään sen koodia tai emme kykene tunnistamaan sitä omaksi simulaatioksemme. Kenties ihmisen ajatusmaailma ei ole ristiriidaton, jolloin ihmistä simuloivan koneenkaan ei tarvitse olla konsistentti. Silloin Gödelin epätäydellisyyslausekaan ei pätisi siihen. Mikään näistä kompromisseista ei vaikuta ratkaisevalta uhalta mekanistiselle teesille.

asiasanat: Gödel, Lucas, Penrose, tekoäly, Turingin kone, tietokoneet, simulointi, epätäydellisyys, filosofia, teoreettinen filosofia, matematiikan filosofia, platonismi

Sisällys

1. Johdanto

Elinympäristömme on tietokoneiden ja tekoälyn ympäröimä. Autonavigaattori osaa etsiä nopeimman reitin, suoratoistopalvelu suosittelee uutta elokuvaa aiempien kokemustemme perusteella, ja tekstisyöttö ennakoi mitä haluamme sanoa. ChatGPT kykenee vaikkapa kirjoittamaan palan opinnäytetyöstä.

Teknologian nopea kehittyminen on luonut hyvin optimistisen näkemyksen tekoälyn mahdollisuuksista. Tieteisfantasiassa androidit ovat vain ihmisiä peltihaarniskan sisällä, ja jo sanat kuten ”tekoäly” ja ”tietokone” lähtökohtaisesti romantisoivat niiden toimintaa. Tämä herättää kysymyksen siitä, olisiko mahdollista, että ihmismieltäkin voitaisiin simuloida tekoälyllä? Toisin sanoen, ihmisen ajattelu ja aivot olisivat analogisia tietokoneen kanssa. Yksi mielenkiintoinen lähestymistapa kysymykseen tulee viime vuosisadan merkittävimmästä matematiikan saavutuksista, Gödelin epätäydellisyyslauseesta. Gödelin epätäydellisyyslauseen johtopäätös on, että tietyt formaalit todistusjärjestelmät tulevat aina sisältämään sellaisia lauseita, jotka ovat tosia, mutta joita ne eivät voi itse osoittaa todeksi.

Kurt Gödelin vuonna 1931 todistamat epätäydellisyyslauseet usein ymmärretään väärin populäärikulttuurissa. Gödelin epätäydellisyyslauseesta ei seuraa, että matematiikka olisi jotenkin sisäisesti ristiriitainen tai epätäydellinen tieteenala, vaan sen tulokset koskevat vain tiettyjä valikoituja todistusjärjestelmiä, jotka ovat tarpeeksi voimakkaita käsittelemään aritmetiikkaa luonnollisilla luvuilla. Nämä järjestelmät tulevat aina sisältämään lauseita, joita ei voida todistaa todeksi järjestelmän sisällä sen omista aksioomista käsin. Se ei tarkoita, etteikö tällainen mystinen lause olisi ratkeava jossain toisessa järjestelmässä, esimerkiksi ottamalla tuo ratkeamaton lause itse aksioomaksi. Ei vain ole olemassa mitään yhtä tiettyä systeemiä, joka kykenisi kattamaan kaikki matematiikan todet väitteet. (Raatikainen 2020, 1.1.)

Tulos ravisutti tiedemaailmaa, ja se tuhosi Hilbertin projektin löytää kaiken matematiikan kattavat aksioomat. Hilbert oli näet pyrkinyt luomaan matematiikalle varman ja selkeän perustan, jonka pohjalta kaikki matemaattiset totuudet olisi mahdollista johtaa. Se ei kuitenkaan osoittautunut mahdolliseksi, joten meillä on edelleen eri ”kielet” puhua geometrian ja lukuteorian väitteitä.

Mutta Gödelillä oli mielessään myös eräs toinen johtopäätös. Hän samaisti formaalisen todistusjärjestelmän tietokoneohjelmaan, ja totesi:

> Joko matematiikka on tällä tavoin mahdoton saattaa täydelliseksi, että sen ilmeisiä aksioomia ei voida ilmaista äärellisellä säännöllä, toisin sanoen ihmismieli (jopa puhtaan matematiikan alalla) ylittää äärettömästi minkään äärelliseen koneen kyvyn, tai muuten on olemassa täysin ratkaisemattomia eriteltyä tyyppiä olevia diofantoksen ongelmia. (Gödel 1995, 310.)[1]

Gödel valitsi vaihtoehdoista ensimmäisen. Myöhemmin John Lucas vei Gödelin argumenttia entistä pidemmälle. Hän väitti, että esitettiinpä millainen kone tahansa tuottamaan ihmisen aritmeettiset (todet) ajatukset, Lucas pystyisi osoittamaan sellaisen aritmetiikan väitteen, jonka hän tietää todeksi, mutta jota kone ei kykene todistamaan, nimittäin tuon koneen Gödelin lauseen. Koska sellaista tietokonetta, joka kattaisi kaiken ihmisen matemaattisen kyvykkyyden ei siis voisi olla, ei ihmisen ajattelukaan olisi simuloitavissa tietokoneella. Lucasin tulos on fantastinen, mutta syvempi tarkastelu osoittaa, että se vaatii toimiakseen paljon raskaampia oletuksia, kuin päälle päin näyttää.

Tässä työssä esittelen ensin aiheeseen liittyviä käsitteitä, kuten diagonaaliargumentin, Gödelin epätäydellisyyslauseen, ja Turingin koneen pysähtymisongelman. Sen jälkeen tutustumme gödeliläisen argumentin merkittävimpien kannattajien, John Lucasin ja Roger Penrosen ajatuksiin. Tavoitteena on vertailla tietokonetta ja ihmistä matemaattisina ongelmanratkaisijoina. Pystyykö tietokone todella kaikkeen samaan kuin ihminenkin? Kysymys voidaan esittää myös muodossa: voidaanko ihmisen matemaattinen ongelmanratkaisukyky pelkistää laskutoimituksiksi?

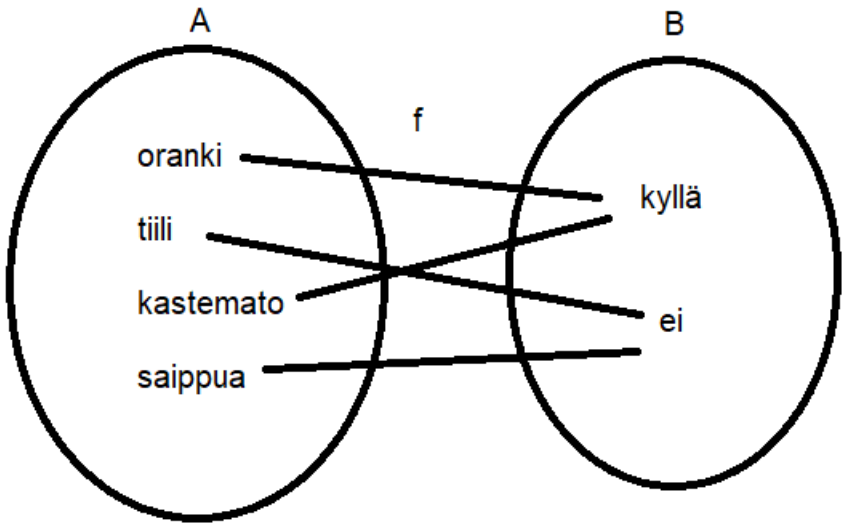
Käy ilmi, että gödeliläinen argumentti onnistuu siihen asti, kun kyseessä ovat sellaiset korrektit tietokoneohjelmat, jotka voisimme tunnistaa omaksi kuvaksemme. Mutta tämä jättää vahvan tekoälyn kannattajalle vielä vaihtoehtoja: kenties aivomme ovat niin monimutkaiset, ettemme pysty käsittämään niiden toimintaa, saati niiden simulaatiota, matemaattisella tarkkuudella. Ehkä tietyt matemaattiset totuudet ovat aidosti ihmisen saavuttamattomissa, tai kenties ihminen on perimmäisellä tavalla erehtyväinen.

[1] either … the human mind (even within the realm of pure mathematics) infinitely surpasses the power of any finite machine, or else there exist absolutely unsolvable diophantine problems. Lainaus Gödelin vuonna 1951 pitämältä luennolta.

2. Taustaa

Jotta gödeliläistä argumenttia laskennallista mielenteoriaa vastaan voidaan ymmärtää, käymme ensin läpi tärkeimpiä aiheeseen liittyviä käsitteitä ja tuloksia matematiikassa ja tietojenkäsittelytieteessä. Näihin sisältyvät formaalin todistusjärjestelmän luonne, Gödelin epätäydellisyyslauseet, sekä Turingin koneen toimintaperiaate.

Funktio (tai kuvaus) on joukko järjestettyjä pareja lähtöjoukon ja maalijoukon alkioista, siten että jokainen lähtöjoukon alkio esiintyy parina kerran. Jos mikään maalijoukon alkio ei esiinny parina useampaa kuin yhtä kertaa, on kyseessä injektio. Jos kaikki maalijoukon alkiot esiintyvät parina on kyseessä surjektio. Jos funktio on sekä injektio että surjektio, se on **bijektio**. Joukko-opissa kaksi joukkoa ovat yhtä mahtavia (”suuria”), jos niiden välillä on bijektio. Tämä on hyödyllistä puhuttaessa äärettömien joukkojen ”koosta”.



Kuvassa funktio f yhdistää lähtöjoukon A alkiot maalijoukon B alkioihin. Tämä funktio kertoo, onko tietty A:n alkio eläin. Funktio f on surjektio, muttei injektio tai bijektio.

Valehtelijan paradoksi syntyy, kun lause viittaa itseensä muodostaen ristiriidan. Se esitetään usein muodossa ”Tämä lause on epätosi”. Jos lause on tosi, se on epätosi. Jos lause on epätosi, niin se on tosi. Molemmat vaihtoehdot johtavat mahdottomaan ristiriitaan.

Diagonaaliargumentti on eräänlainen muunnelma valehtelijan paradoksista, jossa itseensä viittaaminen johtaa ristiriitaan. Todistus sille, että reaalilukujen äärettömyys (ylinumeroituva

äärettömyys) on mahtavampi kuin luonnollisten lukujen (numeroituva äärettömyys) voidaan osoittaa diagonaaliargumentilla. Myös Gödelin epätäydellisyyslauseen ja Turingin koneen pysähtymisongelman ratkeamattomuuden todistukset ovat diagonaaliargumentin analogioita.

Todistus sille, että reaalilukujen äärettömyys on mahtavampi kuin luonnollisten lukujen äärettömyys: Jos reaalilukujen ja luonnollisten lukujen joukot ovat yhtä mahtavia, voidaan niiden välille muodostaa bijektio. Nyt jokaista luonnollista lukua kohden on aina yksi reaaliluku. Nyt voimme tämän bijektion pohjalta konstruoida uuden reaaliluvun seuraavasti. Katsotaan mikä luku on luonnollisen luvun 0 vastinpari reaaliluvuissa, ja tarkastellaan sen yksikköä (viimeinen numero ennen desimaalipilkkua). Jos reaaliluvusta valittu numero on 0, valitaan konstruoitavan luvun ensimmäiseksi desimaaliksi 1. Jos reaaliluvun numero ei ole 0, valitaan konstruoitavan luvun ensimmäiseksi desimaaliksi 0. Jatketaan tätä kaikilla luonnollisilla luvuilla siten, että tarkasteltava numero on aina sama desimaali, kuin tarkasteltava luonnollinen luku. Koska reaaliluvun desimaaliketju saa olla ääretön, voimme edetä näin. Näin saamme konstruoitua uuden reaaliluvun, joka poikkeaa aina vähintään yhdellä desimaalilla kaikista bijektion luettelemista reaaliluvuista. Konstruoitu reaaliluku ei siis kuulu tähän luetteloon. Bijektio ei siis kykene kattamaan kaikkia reaalilukuja, mikä on ristiriita lähtöoletuksen kanssa. Reaalilukujen äärettömyys on siis mahtavampi kuin luonnollisten lukujen äärettömyys.

luonnollinen luku	reaaliluku						konstruoitu uusi reaaliluku = 1,00011...
0	0,	5	0	0	0	0	
1	3,	1	4	1	5	9	
2	1,	4	1	4	2	1	
3	0,	3	3	3	3	3	
4	8,	0	0	0	0	0	
5	12,	0	0	0	0	0	

2.1. Gödelin epätäydellisyyslause

Formaali todistusjärjestelmä koostuu joukosta aksioomia (todeksi oletettuja lähtökohtia). Jos aksioomia on numeroituvasti äärettömästi, tulee antaa algoritmi, joka tuottaa ne[2]. Aksioomista

[2] Esimerkkinä äärettömästä määrästä aksioomia on matemaattinen induktio. Jos voidaan osoittaa oletuksesta, että väite F pätee mielivaltaisella luonnollisella luvulla n, voidaan johtaa, että F pätee myös luvulla n+1, ja tiedetään että F

johdetut lauseet, eli teoreemat, ovat sellaisia, jotka voidaan laskennallisesti tarkastaa oikeiksi. Jos todistusjärjestelmä ei sisällä ristiriitaa se on **konsistentti**. Jos todistusjärjestelmän jokainen lause voidaan todistaa todeksi tai epätodeksi, niin se on **täydellinen** (engl. *complete*). (Raatikainen 2020, 1.1.) Todistusjärjestelmä on **korrekti** (engl. *sound*), jos kaikki sen aksioomista johdetut teoreemat ovat myös aksioomien loogisia seurauksia. Korrektiudesta seuraa aina myös konsistenttius, jokainen korrekti todistusjärjestelmä on siis konsistentti.

Gödelin (ensimmäinen) epätäydellisyyslause: Konsistentti formaali todistusjärjestelmä, joka on riittävä alkeelliseen aritmetiikkaan, on epätäydellinen. Se sisältää aina lauseen, jota se ei yksin kykene todistamaan todeksi tai epätodeksi. Riittävänä aritmetiikkana voimme pitää normaalia plus- ja kertolaskua äärettömällä määrällä luonnollisia lukuja[3]. (Raatikainen 2020, 1.1.)

Todistus on monimutkainen, mutta ideana on diagonaaliargumentilla löytää todistusjärjestelmän **Gödelin lause**, eli lause, jolle ei ole todistusta järjestelmässä. Tulee huomata, että Gödelin lause ei ole mikään yleisesti ratkeamaton väite, vaan jokaisella riittävää aritmetiikkaa sisältävällä todistusjärjestelmällä on oma Gödelin lauseensa, joka voi kuitenkin olla ratkeava jossain toisessa järjestelmässä. Toisaalta Gödelin lause voidaan hyväksyä sellaisenaan aksioomaksi ja luoda uusi todistusjärjestelmä, jolla taas olisi oma ratkeamaton Gödelin lauseensa. Kenties voisimme hyväksyä myös epäformaalin perustelun Gödelin lauseen todenperäisyydelle. Käytämmehän luonnollisten lukujen yhteen- ja kertolaskua normaaleina työkaluina ilman ongelmia. Tiedetään, että jos voimme olettaa Peano aritmetiikan olevan konsistentti, silloin Peano aritmetiikan Gödelin lause on tosi. Mutta sellaista perustelua emme voi esittää, joka olisi matemaattisen eksakti, ja perustuisi ainoastaan Peano aritmetiikan omiin aksioomeihin. (Raatikainen 2020, 2.3.)

Kaikki Peano aritmetiikan totuudet ovat loogisia seurauksia Peanon aksioomille. Mutta sellainen matematiikan oppikirja ei ole mahdollinen, joka kykenisi antamaan ohjeet jokaisen Peano aritmetiikan väitteen ratkaisemiseksi. Mahdollisia ratkaisuja on äärettömästi ”yhtä paljon” kuin luonnollisia lukujakin, mutta kysymysten äärettömyys on niitä ”suurempi”.

---

pätee luvulla 0, tiedetään, että F pätee nyt kaikilla luonnollisilla luvuilla. F pätee nollalla, joten F pätee 0 + 1:llä, joten F pätee 0 + 1 + 1:llä…

[3] Plus- ja kertolaskun lisäksi muut tarvittavat aksioomat ovat: 1) Jokaisella luvulla on seuraaja 2) Nolla ei ole minkään luvun seuraaja, 3) Millään kahdella luvulla ei ole samaa seuraajaa. Tutummin ilmaistuna luvun n seuraaja on n+1. Lisäämällä vielä aksioomaksi matemaattinen induktio olemme määritelleet Peano aritmetiikan.

Gödelin lauseesta olisi toki hyvä esittää esimerkki, mutta ne ovat hyvin vaivalloisia. Niissä voi olla merkkejä yli 10 000 kappaletta ja olla käytännössä lukukelvottomia. Gödelin lause esitetäänkin usein raa'asti yksinkertaistaen muodossa "Tällä lauseella ei ole todistusta tässä järjestelmässä". Jos järjestelmä olisi täydellinen, silloin Gödelin lause olisi todistuva. Tästä seuraa ristiriita ja järjestelmä ei ole konsistentti. Jos järjestelmä on konsistentti, se ei siis voi todistaa Gödelin lausetta, eli järjestelmä on epätäydellinen. Gödelin lause voidaan tietää todeksi, jos voimme olettaa kyseisen todistusjärjestelmän olevan konsistentti.

Gödelin toinen epätäydellisyyslause: Riittävään aritmetiikkaan kykenevä todistusjärjestelmä ei voi todistaa omaa konsistenttiuttaan[4]. (Raatikainen 2020, 1.1.)

2.2. Turingin kone

Algoritmi (tietokoneohjelma) on eksplisiittinen ohje, jolla on kolme ominaisuutta: yksittäisten ohjeiden perättäisyys, valinnan tekeminen kahden vaihtoehdon välillä, sekä paluu aiempaan ohjeeseen ja sen toisto. Käyttäjä syöttää algoritmille lähtöoletukset syötteenä (engl. *input*) ja lopputulos on tuloste (engl. *output*). Tyypillinen esimerkki algoritmista on keittokirjan resepti, joka antaa seikkaperäiset ohjeet halutun ruokalajin valmistamiseen[5].

Esimerkiksi perunoiden keittäminen seuraavilla ohjeilla on algoritmi:

1. Kaada kattila puolilleen vettä.
2. Aseta kattila hellalle ja hella päälle.
3. Kun vesi kiehuu, lisää perunat siten, että ne peittyvät vedellä.
4. Anna perunoiden kiehua jonkin aikaa.
5. Kokeile haarukalla, ovatko perunat kypsiä. Jos eivät, palaa riville 4.
6. Perunat ovat valmiita.

[4] Gödelin toinen epätäydellisyyslause on vaativampi suppeimman aritmeettisen teorian suhteen. Ensimmäinen epätäydellisyyslause ei vaadi todistusjärjestelmältä matemaattisen induktion tai rekursion sisällyttämistä.
[5] Keittokirjan syöte ja tuloste voidaan mieltää triviaaleiksi. Algoritmi antaa vain kokille ohjeet toimia ilman että syötteelle tarvitsisi tehdä mitään.

Algoritmit ovat abstrakteja matemaattisia konsepteja siinä missä luvut ja tasokuviotkin. Algoritmeja voidaan suorittaa esimerkiksi tietokoneella, tai omassa mielessä käyttäen kynää ja paperia apuna.

Turingin kone on tietokoneen teoreettinen malli. Se koostuu minimissään neljästä ominaisuudesta: aakkostosta, tilajoukosta, lähtötilasta sekä siirtymäfunktiosta. Yleensä sille määritellään myös luku- ja kirjoitupää, sekä ääretön muistinauha. Kone aloittaa toimintansa lähtötilasta. Lukupää lukee muistinauhalta aakkosen, joka yhdessä koneen tilan kanssa syötetään siirtymäfunktiolle. Näiden perusteella siirtymäfunktio kertoo, mikä aakkonen kirjoitetaan luetun tilalle, mihin suuntaan muistinauhaa lukupää nyt siirtyy yhden symbolin verran, ja mikä on koneen seuraava tila. Kone jatkaa näin, kunnes se päätyy lopetustilaan. (De Mol 2018, 1.3)

Esimerkkinä määritellään Turingin koneen, joka ottaa syötteenään binääriluvun ja lisää siihen yhden. Sen aakkoston kuuluvat symbolit 0, 1 ja tyhjä. Sen tilat ovat lähtötila q ja pysähtymistila h. Oletetaan, että lukupää aloittaa luvun viimeisestä merkistä oikealla.[6]
Jos kone on tilassa q, ja lukee kirjaimen 1, se kirjoittaa kirjaimen tilalle 0, lukupää siirtyy yhden merkin vasemmalle, ja jatkaa tilassa q.
Jos kone on tilassa q, ja lukee kirjaimen 0 tai tyhjän merkin, se kirjoittaa sen tilalle kirjaimen 1, siirtää lukupäätä askeleen vasemmalle, ja siirtyy pysähtymistilaan h.
Annetaan koneelle syötteenä binääriluku 1011 eli yksitoista. Nyt seurataan askel askeleelta koneen toimintaa, kun lukunauha ja koneen tila ovat esillä. Lukupään kohdalla oleva merkki on lihavoitu.
101**1** tilassa q
10**1**0 tilassa q
1**0**00 tilassa q
1100 tilassa h
Kone pysähtyy. Tuloste on 1100 eli kaksitoista.

Turingin kone voi myös simuloida muita Turingin koneita, ottamalla syötteenään toisen koneen syötteen ja koodin. Tällainen **universaali Turingin kone** kykenee ajamaan mitä tahansa tietokoneohjelmaa, joten se asettaa teoreettiset rajat tietokoneidemme toiminnalle. Turingin koneella voimme tarkistaa, onko annettu todistusjärjestelmän aksioomista johdettu todistus

[6] Klassisen määritelmän mukaan lukupää voi aloittaa mistä tahansa kohtaa ääretöntä syötenauhaa. Ohjelmaan voidaan kuitenkin kirjoittaa rutiini, joka löytää nauhalta (äärellisen) syötteen, ja siirtää lukupään sen viimeiseen merkkiin.

pätevä. Church-Turingin teesin mukaan kaikki algoritmiset ongelmat voidaan ratkaista Turingin koneella.

Toisaalta Turingin konetta voidaan itsessään pitää yhtenä esimerkkinä formaalista todistusjärjestelmästä. On osoitettu, että mille tahansa Turingin koneelle, joka hyväksyy tietyt väitteet, on olemassa vastaava todistusjärjestelmä, josta voidaan johtaa tismalleen samat väitteet loogisine johtopäätöksineen. (Putnam 1965, 251)

Kysymys on **algoritmisesti ratkeava**, jos on olemassa Turingin kone, joka antaa siihen aina oikean vastauksen. Täten jokainen kyllä tai ei -kysymys, kuten ”Onko Jumalaa olemassa?” tai ”Rakastaako Kati minua?”, on laskennallisesti ratkeava. Otetaan kaksi paperinpalaa. Kirjoitetaan toiseen ”kyllä” ja toiseen ”ei”. Ne ovat molemmat (hyvin yksinkertaisia) Turingin koneita, ja toinen niistä antaa oikean vastauksen, jos kysymys on hyvin määritelty. Emme vain tiedä, kumpi paperinpala on oikeassa. Algoritminen ratkeavuus on siis hyvin erilainen käsite, kuin mitä luonnollisessa arkikielessä kutsuisimme ongelmanratkaisuksi. Samoin jokainen kysymys, jonka ala on äärellinen, on triviaalisesti algoritmisesti ratkaistavissa. Tällöin algoritmiksi riittää lista, joka käy läpi kaikki lähtöjoukon alkiot ja ilmoittaa niille oikean vastauksen. Esim. onko luku alkuluku, kun luku tulee valita luonnollisista luvuista 2–5 ratkeaa listalla ((2, kyllä), (3, kyllä), (4, ei), (5, kyllä)). Kysymys algoritmisesta ratkeavuudesta onkin kiinnostava vain äärettömillä joukoilla.

Turingin koneen pysähtymisongelma (engl. *Halting Problem*). On olemassa tietokoneohjelmia, joiden suoritus ei lopu koskaan, vaan ne jäävät jumiin äärettömään sykliin[7]. Diagonaaliargumentilla voidaan osoittaa, ettei ole olemassa sellaista Turingin konetta, joka ottaisi vastaan toisen Turingin koneen syötteineen, ja kertoisi, pysähtyykö tuon koneen suoritus ikinä. Kyseessä on Gödelin epätäydellisyyslauseen analogia tietojenkäsittelyssä. Turingin koneen pysähtymisongelma osoittaa, ettei ole olemassa yhtä yleistä tietokoneohjelmaa, joka kykenisi ratkaisemaan jokaisen algoritmisesti ratkeavan ongelman.

[7] On loputtomasti helppoja esimerkkejä tietokoneohjelmista, jotka eivät koskaan lopeta suoritustaan. Kuten kone, joka vain liikkuu muistinauhalla edestakaisin vasemmalta oikealle ikuisesti. Kone aloittaa tilasta v. Tilassa v kone siirtyy askeleen vasemmalle ja siirtyy tilaan r. Tilassa r kone siirtyy askeleen oikealle ja siirtyy tilaan v. Tämä kone ei tule koskaan saavuttamaan pysähtymistilaa h. Aakkosilla ei nyt ole väliä, voimme olettaa, että sekä v ja r korvaavat minkä tahansa luetun merkin tyhjällä merkillä.

On jokseenkin makuasia, millaisia perinteisiä tietokoneohjelmia kutsumme tekoälyksi. Nykyään tekoälyn valtavirtaa ovat ns. **neuroverkot**. Neuroverkkojen arkkitehtuuri eroaa Turingin koneesta, sillä ne eivät toimi deterministisesti. Neuroverkkoon kuuluu sarja painoja, joiden avulla se arpoo seuraavan toimintonsa. Riippuen siitä, johtiko toiminto haluttuun lopputulemaan neuroverkko muokkaa painotuksiaan. Kuitenkaan neuroverkot eivät laskennallisesti ole Turingin konetta ”voimakkaampia”, joten tämän työn kontekstissa ne voidaan sivuuttaa. Neuroverkot ovat simuloitavissa Turingin koneella, ajetaanhan neuroverkkoja ihan perinteisillä tietokoneilla. (Arkoudas & Bringsjord 2014, 53) (Penrose 1994, 18–19)

On huomattava, että rakennetut fyysiset tietokoneet poikkeavat aina Turingin koneesta. Turingin kone on tietokoneen ideaali malli, eikä mikään tietokone voi saavuttaa samanlaista täydellisyyttä. Tietokoneen toiminta voi poiketa Turingin koneesta fyysisen vian vuoksi, ja toisin kuin Turingin koneella, tietokoneelta voi loppua muisti kesken. Tietokone, kuten ihminen, on sidottu aikaan, eikä ole realistista ajatella tietokoneen jatkavan toimintaansa erittäin pitkäksi ajaksi. Tässä työssä käytän jatkossa käsitettä Turingin kone tietokoneen teoreettisesta mallista ja sanaa tietokone, kun tarkoitan Turingin koneen rakennettua instanssia.

2.3. Platonismi

Platonismi ymmärretään nykykeskustelussa käsityksenä, että matematiikan väitteet ja objektit ovat riippumattomia materiaalisesta maailmasta, ikään kuin luvut ja tasokuviot sijaitsisivat ”Platonin taivaassa”. Platonismi ei ole nykyään kovin suosittu näkemys sen mystisen luonteen vuoksi. Miten rajalliset ja (ainakin osittain) materiaaliset ihmiset voisivat olla yhteydessä ikuisiin ja muuttumattomiin objekteihin? Toisaalta mikään vaihtoehto platonismille ei ole pystynyt tyhjentävästi selittämään, kuinka tietomme matematiikasta (vaikuttavat) kokemuksesta riippumattomilta ja välttämättömiltä totuuksilta – ja samalla matematiikalla on huomattava asema kokeellisessa tieteessä.

Gödeliläisestä argumentista tekoälyä vastaan ei seuraa platonismi, mutta platonistisessa maailmankuvassa on helpompi hyväksyä todistusjärjestelmien ja tietokoneohjelmien rajallisuus suhteessa koko matematiikan valtakuntaan.

Gödelin mielenfilosofinen ajatusmaailma on dualistinen – hän erottaa ruumiin ja mielen toisistaan. Lucas ei menisi ihan niin pitkälle. Substanssiopissa hän määrittelee itsensä korkeintaan puoliväliin monistia ja dualistia – mitä ikinä se tarkoittaakaan. Mutta episteemisesti Lucas myöntää olevansa vähintään dualisti. Luonnontieteellisen selityksen lisäksi tarvitaan teoria selittämään ja oikeuttamaan rationaalisten agenttien toimintaa. (Lucas 1996a, 2)

Roger Penrose taas on materialisti mielenfilosofiassa, mutta hänen mielestään ihmisaivoilla on kyky käsitellä matematiikkaa, joka ylittää Turingin koneen rajat. Tämän Penrose selittäisi kvanttimekaniikalla.

3. Lucasin projekti

John Randolph Lucas (18.6.1929. – 5.4.2020.) oli englantilainen filosofi, joka tunnetaan gödeliläisestä argumentistaan tekoälyn mahdollisuuksia vastaan. Hän esitteli ajatuksensa alunperin kirjoituksessa Minds, Machines and Gödel vuonna 1961. Lucasin mukaansa ihmistä simuloivaa konetta ei ole mahdollista luoda, sillä sellaisesta koneesta olisi aina mahdollista löytää Gödelin lause, jonka ihminen tietää todeksi, mutta kone ei.

Lucasin mukaan Gödelin epätäydellisyyslause kumoaa mekanistisen käsityksen ihmismielestä. Mekanistisella ajatusmaailmalla on filosofiassa useita eri merkityksiä, mutta Lucas viittaa sillä käsitykseen siitä, että ihmisen ajattelua olisi mahdollista simuloida tietokoneella. Tuollainen tietokone tuottaisi kaikki ne (aritmeettiset) todet ajatukset ja todistukset, joihin ihminenkin kykenee. Materialistinen teoria mielestä ei ole välttämättä sitoutunut tällaiseen mekanistiseen ajatukseen, mutta ainakin osaa funktionalistista mielenteorioista voidaan kutsua mekanistisiksi.

Lucas pyrkii Gödelin epätäydellisyyslauseella osoittamaan, että esittääpä mekanisti millaisen Turingin koneen tahansa ihmismielen malliksi, tuo kone häviää aina ihmiselle matemaattisena ongelmanratkojana. Tietysti tietokoneet ovat tietyllä tapaa ihmistä etevämpiä, sillä nykyiset supertietokoneet kykenevät suorittamaan biljoonia laskutoimituksia sekunnissa. Kyse on siitä, voidaanko jokaista konetta kohden osoittaa sellainen yksittäinen tapaus, jossa ihminen päihittää koneen.

Lucasin argumentti etenee seuraavasti: ensin mekanisti tarjoaa hänelle tietokoneen, joka väitetysti simuloi ihmisen matemaattista ajattelua. Esimerkiksi jos Lucas tietää, että 2 + 3 = 5, kone voi tuottaa tuon saman väitteen, esimerkiksi tulostamalla sen tietokoneen näytölle. Lucas ymmärtää Peano aritmetiikan aksioomat, joten häntä simuloivan koneen tulee siis olla tarpeeksi voimakas sisältääkseen Gödelin lauseen. Koska Lucas tuntee Gödelin epätäydellisyyslauseen todistuksen, hän voi koneen arkkitehtuurista löytää tuon koneen Gödelin lauseen, jota kone ei itse kykene todistamaan. Koska Lucas tuntee Gödelin todistuksen, hän nyt tietää Gödelin lauseen todeksi, mutta kone ei. Tällöin mekanisti ei onnistunutkaan simuloimaan Lucasia. (Lucas 1961, 115–116)

Lucasin argumentti käyttää epäsuoran todistuksen (*reductio ad absurdum*) muotoa. Siinä otetaan ensin hypoteesina oletus, tässä tapauksessa oletamme mekanistin tuottaneen ihmistä simuloivan tietokoneen. Tästä oletuksesta pyritään johtamaan ristiriita, eli se, että kone ei lopulta pystynytkään simuloimaan ihmistä. Koska logiikkamme ei salli ristiriitaa, tulee alkuperäinen oletus kieltää epätotena.

Mekanisti voi toki myöhemmin esittää parannellun version koneestaan, kuten sellaisen, joka onnistuu tuottamaan edellisen koneen Gödelin lauseen. Mutta tuollakin koneella on taas uusi Gödelin lauseensa, jonka Lucas löytää taa jne. Siispä mekanisti ei voi milloinkaan esittää sellaista konetta, joka vastaisi Lucasin aritmeettista ajattelua. (Lucas 1961, 116–117)

Lucas ei erityisen selkeästi erottele, kuvaako mekanistin luoman koneen toiminta yhtä tiettyä ihmistä, vai ihmisen laskennallista kykyä yleisesti. Lucasin kirjoituksista saan kuitenkin käsityksen, että hän ajattelee asia yleisellä tasolla. Lucas antaa ymmärtää, että ihminen saisi käyttää apunaan Gödelin lauseen löytämisessä muita ihmisiä ja tietokonetta. [8] Mutta tällöin, jos kone simuloi yksittäistä ihmistä, ja sen päihittävä ihminen ajatellaan yleisemmin ”ihmisyytenä” ei Lucasin toivomaa ristiriitaa synny. Ei siinä ole mitään erityisen ihmeellistä, että ihmiskunnan kollektiivinen kyvykkyys on suurempi kuin yhden partikulaarisen ihmisyksilön simulaatio.

Tiettävästi kukaan ei samaistu Lucasin olkinukkeen mekanistista. Tekoälyn suopeimmat kannattajatkaan eivät kuvittele voivansa tuottaa ihmistä simuloivaa konetta. Teoreettisena

[8] Myös Roger Penrose ajattelee, että mekanistin koneesta Turingin lauseen voi löytää matemaatikkojen yhteisö, eikä kenen tahansa tarvitse sitä yksinään saavuttaa. (Penrose 1994, 97)

työkaluna Lucasin käsitys mekanistista on kuitenkin käypä. Jos ihmismielen toiminta voitaisiin samaistaa Turingin koneeseen, silloin sellaisen koneen tulisi olla mahdollinen.

Lucasin kirjoitus on saanut filosofiassa paljon huomioita, mutta ei niinkään sympatiaa. Stewart Shapiron mukaan ei ole sellaista uskottavaa mekanistista teesiä, joka olisi määritelty tarpeeksi tarkasti, jotta Gödelin epätäydellisyyslause sitä koskisi. Mekanistin tulisi määrittää, millainen hänen koneensa on, mitä ihmisen ominaisuuksia se simuloi, ja ketä ihmistä se tarkalleen ottaen esittää. Argumentin puitteissa voimme toki keskittyä sellaiseen koneeseen, joka tuottaa Peano aritmetiikan väitteitä. (Shapiro 1998, 275)

Analogiana voimme verrata Lucasin argumenttia Turingin koneen pysähtymisongelmaan. Tietokone ei voi ratkaista, pysähtyykö mielivaltaisen tietokoneohjelman suoritus lopulta, vai jääkö se jumiin ikuiseen silmukkaan. Mutta pystyykö ihminen osoittamaan ratkaisun? Monessa tapauksessa hän on siinä onnistunutkin. Samoin tietokone voidaan ohjelmoimaan tunnistamaan tietyt, rajatut tapaukset, jotka johtaisivat äärettömään suoritukseen, ja pysäyttämään ne. Mutta tietokone ei kykene ratkaisemaan ongelmaa yleisesti koskien jokaista mahdollista tietokoneohjelmaa syötteineen. Jos ihminen tähän pystyisi, hän todellakin olisi tietokonetta mahtavampi. Meillä ei kuitenkaan ole mitään matemaattisella varmuudella perusteltua syytä olettaa tätä suurenmoista kykyä. Samoin Gödelin epätäydellisyyslause kyllä löytää ratkeamattoman lauseen jokaisesta tietokoneohjelmasta, mutta Lucas ei onnistu vakuuttamaan epäilijää siitä, että ihminen pystyisi jokaisessa tapauksessa todistamaan tuon lauseen todeksi.

3.1. Ihmismielen idealisointi

Lucas ei rajoita käsitystään ihmisyydestä vain siihen mitä konkreettisesti saamme elinaikanamme aikaiseksi, vaan siihen, mitä kaikkea meidän on periaatteessa mahdollista tietää todeksi. Tuolloin ajan kuluminen ei ole ongelma, ja saamme käyttää apunamme muita ihmisiä ja tietokoneita. Tietyssä mielessä kysymys palautuu intuitioomme platonismista. Ajattelemmeko, että kaikki matematiikan väitteet ovat ainakin periaatteessa ihmismielen ratkaistavissa? Lucasin kommentoijista monet haluavat tarkemman määritelmän ihmisen teoreettisille mahdollisuuksille. Mm. Stewart Shapiro käyttää tällaista ajatusta idealisoidusta ihmisestä. Lucasin omissa kirjoituksissa ei kuitenkaan tällaista superihmistä esiinny.

Koska ihmisen elämä on rajallinen, hän tulee elämänsä aikana esittämään äärellisen määrän tosia matemaattisia väittämiä. Ei ole siis mitään perustavanlaatuista ongelmaa tuottaa sellainen tietokone, joka listaisi kaikki nämä väitteet. Ihminen myös tekee elämänsä aikana laskuvirheitä ja virhepäätelmiä. Tällöin ihmisen esittämien aritmeettisten väitteiden joukko ei ole konsistentti. Jotta voisimme Gödelin epätäydellisyyslauseella verrata ihmistä ja konetta, tulee meidän idealisoida ihminen irti näistä maallisen maailman kahleista. Samoin kuin Turingin kone on tietokoneen teoreettinen malli, tulee meidän postuloida täydellinen ihminen. Tuo euklidisesti ihanteellinen ihminen ei koskaan tee laskuvirheitä, siltä ei koskaan lopu muisti kesken ja sen elinkaari on rajaton. (Shapiro 1998, 275–277)

Tällainen ihmisen ideaali tuntuu monellakin tapaa omituiselta. Kun ihminen tekee matemaattisen virhepäätelmän, onko silloin välttämättä kyse virheestä hänen aivotoiminnoissaan, vai eikö erehtyväisyys voi olla vain luonnollinen osa ihmisen (biologista) luonnetta? Vaikka idealisoitu ihminen päihittäisi koneen, seuraako siitä, ettei todellisia ihmisiä voitaisi simuloida koneen avulla? (Shapiro 1998, 277)

Vaikka tietokone voi jäädä jumiin suorittamaan samaa tehtävää loputtomasti (sillä Turingin koneen pysähtymisongelma on algoritmisesti mahdoton ratkaista), ihmisellä tällaista ongelmaa ei vaikuta olevan. Vaikuttaa siltä, että jo biologisina olentoina eroamme perustavalla tapaa tietokoneesta. Millä perusteella voimme idealisoida ihmistä siten, että olisimme kuin teoreettinen matemaattinen objekti vailla biologisia ominaisuuksia?

3.2. Vertailu tietokoneen ja ideaalin ihmisen välillä

Nyt kun meillä on idealisoitu ihminen ja idealisoitu tietokone (Turingin kone) voimme verrata niiden toimintaa keskenään. Sisältäköön (ääretön) joukko K ne aritmeettiset väitteet, jotka ovat ihmisen tiedettävissä tai todistettavissa. Mekanisti väittää, että kaikki ihmisen tavat tuottaa aritmeettista tietoa ovat simuloitavissa tietokoneella. Tällöin olisi olemassa sellainen Turingin kone, joka kykenee tuottamaan K:n jokaisen alkion. Lucasin olisi osoitettava, että tämä idealisoitu ihminen kykenee jokaista Turingin konetta kohden osoittamaan todeksi sellaisen aritmeettisen toden väitteen, jota tuo Turingin kone ei kykene todistamaan. (Shapiro 1998, 277)

Olkoon nyt joukko T kaikki todet ensimmäisen kertaluvun aritmetiikan väitteet. Jos K=T niin silloin idealisoitu ihminen kykenee ratkaisemaan kaikki aritmetiikan väitteet. Tarskin teoreema taas on, ettei T ole rekursiivisesti numeroituva ts. mikään algoritmi ei voi tuottaa sen kaikkia alkioita. Jos siis ideaali ihminen nyt siis tietää kaikki ensimmäisen kertaluvun aritmetiikan väitteet, on mekanistinen teesi väärässä ja sillä selvä. Toisin sanoan, mekanistisesta teesistä seuraa, että K ≠ T. Mekanisti joutuu siis hyväksymään, että on aidosti olemassa sellaisia aritmeettisia totuuksia, joita ihminen ei voi koskaan tietää todeksi. (Shapiro 1998, 278)

Voidaan kyseenalaistaa, onko Lucasin esittämä skenaario edes teoriassa mahdollinen ja mielekäs. Mekanisti voi näet kieltää, että voisimme tietää koneen koodin tai että voisimme tietää sen konsistentiksi. Tällöin Lucasin argumentti ei kykenisi etenemään.

Nimittäin vaikka mekanisti olisi oikeassa, ja olisi olemassa sellainen Turingin kone M, joka tuottaisi kaikki ihmisen tiedettävissä olevat aritmeettiset väitteet eli K:n, meidän ei olisi mahdollista tietää, että tuo kone M tuottaa K:n ja vain K:n.

Oletetaan että M on olemassa ja se tulostaa vain ja ainoastaan K:n. Oletetaan myös M:n tulostaman joukon aritmeettisia väitteitä olevan konsistentti. Gödelin toisesta epätäydellisyyslauseesta tiedämme, että M ei kykene tulostamaan väitettä omasta konsistenttiudestaan. Koska M:n tuloste on kaikki aritmeettinen tietomme K, me emme voi tietää M:n konsistenttiutta, idealisoitiinpa ihmistä miten paljon tahansa, sillä K kattaa jo kaiken matemaattisen ymmärryksemme. Koska emme voi tietää M:n tulosteen olevan ristiriidaton (konsistentti), emme voi tietää M:n tulosteen sisältyvän tosien aritmeettisten väitteiden joukkoon T. Eli vaikka mekanisti onnistuisi läpäisemään Lucasin haasteen esittelemällä koneen, joka tuottaa (idealisoidun) ihmisen aritmeettiset väitteet, emme voisi tietää sitä. Sama kääntäen, jos idealisoitu ihminen olisi Turingin kone, hän ei voisi tietää mikä Turingin koneista hän on. Hän ei siis tuntisi omaa koodiaan. Mekanisti voi hyvin väittää, että ihmismieltä voi simuloida Turingin koneella, ilman, että on mahdollista osoittaa, mikä Turingin kone on oikea vastinkappale. (Shapiro 1998, 280–281)

> Gödelin mukaan on mahdollista, että tietty todistusjärjestelmä tuottaisi kaikki todeksi osoitettavissa olevat matemaattiset väitteet, mutta kukaan ei voi matemaattisella

> varmuudella tietää, että sen aksioomat ja säännöt ovat oikeita. (Shapiro 1998, 281–282)

Lucas väittää tietävänsä mekanistin esittämän koneen Gödelin lauseen todeksi. Mutta tuon koneen Gödelin lause voidaan todeta todeksi vain, jos tuo kone on konsistentti. Mistä Lucas sen tietäisi? Lucasin mukaan se on mekanistin tehtävä osoittaa. Jos mekanistin esittelemä kone ei tulostaisi konsistenttia joukkoa aritmeettisia väitteitä, se ei kelpaisi tehtävänantoon alun perinkään. Mutta jos mekanisti nyt osoittaa koneensa M konsistenttiuden, Lucas voi tämän perusteella taas osoittaa tietävänsä sen Gödelin lauseen ja osoittaa olevansa konetta mahtavampi. (Lucas 1996b, 117)

Nyt on tärkeää erotella, mitä tarkoitamme sillä, että tiedämme M:n tulosteen olevan konsistentti. Lucas onnistuu vain, jos konsistenttius voidaan tietää matemaattisella varmuudella. Jos taas hyväksymme heikomman, heuristisen perustelun, ei Lucas voi käyttää M:n konsistenttiutta tietääkseen M:n Gödelin lauseen todeksi. Mekanistihan voi vedota esimerkiksi kokemukseen perustellakseen koneensa olevana konsistentti. (Shapiro 1998, 281–284)

Lucasin ajatuskoe vaikuttaa tiettyyn pisteeseen asti onnistuvan tehtävässään. Skenaario, jossa esiteltäisiin ihmistä simuloiva konsistentti tietokone vaikuttaa mahdottomalta. Mutta mekanistille se jättää edelleen muita vaihtoehtoja avoimeksi. Tarkastelemme niitä seuraavassa luvussa.

4. Kritiikkiä Lucasille

Paneudumme seuraavaksi kritiikkeihin Lucasin argumenttia kohtaan. Yleisesti ne kaikki perustuvat epäilyyn ihmisen matemaattisesta kyvykkyydestä. Kriitikko voi vaikka epäillä, että ihminen voisi edes periaatteessa löytää jokaista Turingin konetta kohtaan todella löytää sen Gödelin lauseen. Ihmisen ei siis olisi periaatteessa mahdollista tuntea kaikkia matematiikan tosia väitteitä, tai ainakaan emme voi sitä selvästi tietää.

Voidaan myös esittää, että ihmisen ajattelu sisältää ristiriitoja, jolloin ihmistä simuloivan koneenkaan ei tarvitse olla konsistentti. Gödelin epätäydellisyyslausetta taas ei voida soveltaa epäkonsistenttiin koneeseen.

Vaikka Lucasin ajatuskoe mekanistin tuottamasta laitteesta ihmisen ajattelun peilikuvana olisi mahdoton, se jättää silti sen mahdollisuuden auki, että matemaattinen päättely olisi samaistettavissa algoritmiin, ilman että voimme koskaan tuota algoritmia löytää tai tunnistaa omaksi itseksemme.

4.1. Huomioita ihmismielen idealisoinnista

Lucasin argumentti vaatii toimiakseen idealisoidun version ihmismielestä. Tämä ideaali ihminen ei tee virheitä, eikä siltä koskaan lopu aika kesken. Samaten tietokoneesta käytetään sen teoreettista mallia, Turingin konetta, joka ei tee mekaanisia virheitä, jolla on loputtomasti muistia (ääretön muistinauha), ja joka voi suorittaa ohjelmaansa niin pitkään kuin on tarpeen, tai jopa loputtomiin. Ei ole selvää, miten hyvin Lucasin argumentti toimii, kun kyseessä ovatkin kuolevaiset ihmiset, joita pyritään simuloimaan rajallisilla tietokoneilla. Aiheesta onkin esitetty paljon kritiikkiä.

Mm. Boyer huomauttaa, että ihmisen matemaattiset kyvyt ovat rajallisia. Jonain päivänä kuolemme ja siihen mennessä olemme esittäneet äärellisen määrän aritmetiikan väitteitä. On siis periaatteessa helppoa koota nuo kaikki väitteet listaksi, jonka kone kykenee tulostamaan.

Ei nimittäin ole selvää, mitä tarkoittaa, että kone tuottaa jonkin toden väitteen. Tässä yhteydessä käytetään sekaisin termejä kuten ratkaista, osoittaa, todistaa, tietää ja laskea. Mitkä ovat ne Lucasin esittämät väitteet, jotka koneen tulisi tulostaa? Lasketaanko ne väitteet, jotka Lucas on kirjoittanut liitutaululle vai paperille, entä kuinka yksityiskohtainen todistuksen tulee olla? (Boyer 1983, 149–150)

Jos emme määrittele tarkasti mitä mekanistinen teesi on, ja mitä ihmisen matemaattisen kyvykkyyden simulointi tarkoittaa, voidaan Lucasin simulaationa pitää vaikka animaatiota, joka näyttää Lucasin todistamassa väitteen liitutaululle. Jos kone ainoastaan taltioi Lucasin elämää, sen voidaan katsoa tuottavan myös oman Gödelin lauseensa, jos Lucas niin tekee. Ristiriitaa ei synny,

sillä tuolloin kone ei tuota Gödelin lausetta mistään aksioomista, eikä sillä Searlen kiinalaisen huoneen[9] tavoin ole mitään ymmärrystä omasta koodistaan. (Boyer 1983, 157)

Lucasin mukaan Boyer osuu harhaan. Vaikka kone tuottaisi kaikki aritmetiikan väitteet, jotka Lucas on tähän mennessä esittänyt, Lucas voi edelleen keksiä uusia tosia väitteitä. Kyse olisi tällöin potentiaalisuudesta tuottaa aritmetiikan väitteitä, eikä siitä, mitä todella tulemme elämämme aikana esittämään. Siten taltiointia (esim. kamerakuvaa) Lucasin elämästä kuoleman jälkeen ei voitaisi pitää riittävänä simulaationa Lucasin kyvykkyydestä. (Lucas 1996b, 109)

Vetoaminen potentiaalisuuteen ei kuitenkaan ole mielekäs vastaus. Jos vertaamme koneen suoritusta niihin väitteisiin, joita ihminen voisi periaatteessa esittää aritmetiikassa, olisi jälkimmäinen joukko ääretön. Mekanistin tietokoneella taas olisi äärellisesti muistia ja suoritusaikaa. Tällainen ääretön ihminen ylittäisi varmasti äärellisen tietokoneen rajat, mutta silloin argumentti ei enää riipu Gödelin epätäydellisyyslauseesta. Mekanisti voi vastauksenaan vain todeta, että ihmiset eivät ole äärettömiä.

Megill ilmaisee Lucasin ajattelevan eron koneen ja ihmisen välillä olevan periaatteellinen. Vaikka käytännössä tietyn kuolevaisen ihmisen ajatukset voisi Turingin koneella luetteloida, periaatteessa ihminen kykenee mistä tahansa Turingin koneesta aina löytämään Gödelin lauseen. (Megill 2024, 2d)

Ihmismielen idealisointi ei ole helppo pala niellä myöskään Coderille. Kenties tietty ideaali ihminen kykenee löytämään minkä tahansa Turingin koneen Gödelin lauseen, mutta useimmille ihmisille tuo tehtävä on liian vaikea. Eihän Gödelin epätäydellisyyslauseen todistus ole mitenkään helppo ymmärtää. Sen sijaan ihminen voi aivan hyvin ymmärtää riittävästi aritmetiikkaa, jotta häntä simuloiva Turingin kone sisältäisi Gödelin lauseen. Parhaimmillaan Lucasin argumentti osoittaa, että tietyt etevät loogikot eivät ole Turingin koneita, mutta johtopäätöstä ei voida yleistää koskemaan ihmisiä yleisesti. (Coder 1969, 234–235)

[9] Searlen kiinalainen huone on kuuluisa argumentti mielenfilosofiassa. Huoneessa elää henkilö, jonka postiluukusta tipahtaa viestejä. Viestit koostuvat mandariinikiinan merkeistä. Asukas ei ymmärrä kiinaa, mutta hänellä on huoneessaan opas, josta löytyy jokaiselle viestille oikea kiinankielinen vastaus. Nyt henkilö voi hakea sopivan vastauksen saamalleen kortille, ja lähettää sen ulos. Näin ulkopuolinen voi kuvitella käyvänsä kirjeenvaihtoa kiinaksi, vaikka huoneen sisällä olevalla henkilöllä ei ole mitään ymmärrystä kiinan merkkien merkityssisällöstä. Symboolimanipulaatio ja formaalisen (tietokone)ohjelman seuraaminen eivät siis ole riittäviä edellytyksiä merkityssisällön ymmärrykselle, vaikka ulospäin järjestelmältä saataisiinkin sopiva vastaus. (Searle 1980, 3)

Lucas vastaa Coderille, että periaatteellista eroa ihmisten välillä ei ole, vaikka vain pieni joukko kykenisi ymmärtämään Gödelin teoreeman ja löytämään mielivaltaisen Turingin koneen Gödelin lauseen. Ei nimittäin ole mitään periaatteellista estettä sille, että kuka tahansa ihminen voisi oppia Gödelin todistuksen. Jotta mekanistin kone voi onnistua, sen pitää pystyä simuloimaan kaikkea matemaattista ajattelua mikä ihmisille on mahdollista, ei ainoastaan sitä mitä ihminen on saavuttanut (Lucas 1970, 149). Toisaalla Lucas myös toteaa, että ihminen voisi tässä työssä saada apua muilta ihmisiltä ja tietokoneilta.

Lucasin argumenttia on myös kritisoitu liian teennäiseksi. Dennettin mielestä Lucasin ajatuskokeessa käsitys ihmisenä olemisesta on liian kapea ja keinotekoinen, ikään kuin vain esittäisimme sarjoja formaalilla kielellä esitettyjä matemaattisia väitteitä. Kuitenkin me myös pilailemme ja puhumme useita luonnollisia kieliä. Tarinamme voivat olla täyttä hölynpölyä tai jopa sisältää ristiriitoja. Mutta ristiriitaista ajatusmaailmaa simuloivalle tietokoneelle Gödelin epätäydellisyyslause ei ole mikään este. Ristiriidan sisältävä kone voi vallan mainiosti todistaa oman Gödelin lauseensa todeksi. (Dennet 1972, 530)

Lucasin mielestä Dennetin huomio on sinänsä hyvä vasta-argumentti mekanistista mielenteoriaa vastaan. Lucasin oma argumentti on kuitenkin muodoltaan epäsuora todistus, jossa oletetaan vain argumentin vuoksi, että mekanistin on mahdollista luoda ihmistä simuloiva kone. Se johtaa kuitenkin ristiriitaan, sillä tuo kone sisältäisi Gödelin lauseen, jota se ei itse pysty osoittamaan todeksi, mutta johon ihminen kykenee. Koska ristiriitaa ei voida hyväksyä, tulee postuloidun lähtöoletuksen olla epätosi. Mutta koska ainakin jotkut ihmiset kykenevät joskus tuottamaan matematiikan väitteitä, tulisi ihmistä simuloivan koneen kyetä tuottamaan ainakin sellaisia. (Lucas 1996b, 108–109)

Koska Lucas itse ei selkeästi määritä ideaalia ihmistä eksplisiittisesti, vaan hän puhuu hähmäisemmin ihmisen potentiaalista tuottaa matematiikan väitteitä, sysää se lukijan vastuulle paljon tulkinnanvaraa. Ja kenties tätä potentiaalia ei voikaan kovin eksaktisti määrittää. Joka tapauksessa tässä kappaleessa käsitellyt kritiikit eivät ole olleet kovin rehellistä, eivätkä ne ota Lucasin argumenttia erityisen vakavasti. On olemassa paljon parempia keinoja vastustaa gödeliläistä argumenttia.

4.2. Huomioita konsistenttiudesta

Moni on ottanut huomioon Jason Megillin tavoin, että mekanistin ei edes tarvitse osoittaa koneensa olevan konsistentti. Kenties ihmismieli ei ole konsistentti, jolloin koneenkaan ei tarvitse olla. Epäkonsistenttiin koneeseen taas Gödelin epätäydellisyyslause ei päde. Se kun kykenisi ristiriidasta johtamaan oman Gödelin lauseensa totuuden. Lucasin tulisi siis osoittaa, että ihmismieli todella on konsistentti (tai kuten Lucas ilmaisee, ihmisen matemaattinen kyvykkyys, josta on poistettu virheet, on konsistentti). (Megill 2024, 2a)

Mutta vaikka ihmismielemme olisivat konsistentteja, se ei tarkoita, että pystyisimme osoittamaan sitä. Jo Gödelin toisen epätäydellisyyslauseen mukaan konsistentti todistusjärjestelmä ei pysty todistamaan itse omaa konsistenttiuttaan. Voisimme myös vastata Lucasille, että hänen argumenttinsa osoittaa, että ihminen ei vastaa konsistenttia Turingin konetta, mutta tämä ei mitenkään poissulje sitä, että ihmismieli voitaisiin rinnastaa epäkonsistenttiin Turingin koneeseen. (Megill 2024, 2a)

Lucasilla on vastaukset molempiin haasteisiin. Vaikka ihminen olisi välillä erehtyväinen, emme kuitenkaan ole samaistettavissa epäkonsistentteihin todistusjärjestelmiin, jotka hyväksyisivät mielivaltaisen väitteen todeksi. Päinvastoin pyrimme välttämään virheitä, ja oikaisemme käsityksemme vastakkaisen todistuksen valossa. Epäkonsistentti kone taas tyytyisi virheisiinsä ja olisi valmis hyväksymään ristiriidan. Sellainen myös olisi valmis hyväksymään minkä tahansa väitteen todeksi, sisällä logiikassa ristiriidasta voidaan päätellä mikä tahansa väite. Ihmiset ovat siis erehtyväisiä, eivät niinkään epäkonsistentteja. Virheitä tuottava kone, joka on valmis korjaamaan erehdyksensä, taas olisi Gödelin epätäydellisyyslauseen alainen. (Lucas 1961, 121)

Lucasin mukaan voimme argumentoida oman konsistenttiudemme puolesta, mutta tuo argumentti ei silloin voisi tulla oman järjestelmämme sisältä. Tämä on matematiikassakin aivan normaalia, että järjestelmän konsistenttius todistetaan käyttämällä jotain toista systeemiä, kuten Gentzen todistaa lukuteorian alkeiden konsistenttiuden. Tällöin taas meidän tulisi tietää tuon ulkopuolisen perusteen nojaavan konsistentteihin oletuksiin. (Lucas 1996a, 3)

Megill tekee olennaisen huomion. Matematiikassa voimme siirtyä todistusjärjestelmästä toiseen, mutta tätä on vaikea rinnastaa siihen, että tarkkailisimme itseämme ulkopuolisesta järjestelmästä käsin. Emme kykene hyppäämään ulos hahmostamme tuota huomiota tekemään. (Megill 2024, 2a)

Jos Jaana osoittaa Heidin olevan konsistentti matemaattisella varmuudella, se ei tarkoita, että Heidi kykenee osoittamaan omaa konsistenttiuttaan matemaattisella varmuudella. Jos Heidi vain uskoo Jaanan pätevyyteen, hänellä voi olla hyvä, mutta matemaattista varmuutta heikompi, peruste uskoa omaan konsistenttiuteensa.

Lucasin mukaan meidän tulee olettaa oma konsistenttiutemme, jotta meidän on mahdollista suorittaa minkäänlaista järkevää ajattelua ylipäänsä. Kyseessä ei siis olisi mikään johtopäätös, vaan kantmaisesti ajateltuna välttämätön edellytys omalle ajattelullemme. Uskomme ilman erityistä perustetta olevamme konsistentteja, samoin kuin hyväksymme muiden ihmisten olevan kaltaisiamme ajattelevia ja tuntevia moraalisia olentoja. (Lucas 1976, kappaleet 4 ja 5)

Megill huomauttaa, että vaikka tämä oletus omasta konsistenttiudestamme olisikin välttämätöntä tehdä, se ei kuitenkaan tarkoita, että todella olisimme konsistentteja. (Megill 2024, 2a)

Ihmismieltä idealisoitaessa on hyvin erilainen asia puhua ajattomuudesta kuin konsistenttiudesta. Kun pohdimme, mitä matematiikan väitteitä ihminen voi periaatteessa saavuttaa, on melko helppoa sivuuttaa ihmisen kuolevaisuus lähinnä teknisenä detaljina. Se ei ole käytännössä koskaan este tietyn matemaattisen väitteen omaksumiselle. Sen sijaan erehtyväisyys vaikuttaa olevan perustavanlaatuisempi osa ihmisen ajattelua. Toki kykenemme korjaamaan osan virheistämme, mutta ei ole selvää, että kykenisimme lopulta saavuttamaan virheettömyyttä. Jos ihmisen olemus olisi perustavalla tapaa epäkonsistentti, voitaisiin koko gödeliläinen argumentti hylätä.

4.3. Voimmeko tunnistaa oman simulaatiomme?

Mekanisti luo koneen M, jonka hän esittää mallintavan ihmisen matemaattista kyvykkyyttä. Lucas väittää löytävänsä tuon koneen Gödelin lauseen, jota kone ei voi tietää todeksi, mutta Lucas tietää. Silloin tuo kone ei voikaan mallintaa Lucasin matemaattista ajattelua. Mutta onko Gödelin lauseen löytäminen todella näin yksinkertaista? Pelkästään esimerkit niistä ovat ahdistavan monimutkaisia lukea.

Paul Benacerrafin mielestä emme voi vain olettaa kykenevämme tähän tehtävään. Vaikka kykenemme löytämään Gödelin lauseen joillekin rajatuille todistusjärjestelmille, emme välttämättä

kykenisi ymmärtämään omaa ajatteluamme simuloivaa systeemiä. Vaihtoehtona Lucasin johtopäätökselle, että ihminen ei vastaa Turingin konetta, voimme olettaa, että ihmistä vastaavalla Turingin koneella ei ole sellaista aksiomatisointia, jota ihminen voisi itse ymmärtää. (Benacerraf 1967, 9–11)

Lucas ei näe mitään periaatteellista estettä Gödelin lauseen löytämiseksi, oli se sitten kuinka vaikeaa tahansa. Vaikka Gödelin lauseen löytäminen olisi hankalaa, voidaan apuna käyttää muita matemaatikkoja tai tietokonetta, joihin voimme luottaa. (Lucas 1996a, sivu 4)

Benecerraf myöntää Lucasin osoittavansa hypoteesinsa mekanistin tuottamasta koneesta johtavan ristiriitaan. Epäsuoran päättelysäännön mukaan olemme siis todistaneet lähtöoletuksen vääräksi. Mutta mikä oletuksista tarkalleen ottaen tulee hylätä? Voimme hylätä ajatuksen siitä, että mieli olisi simuloitavissa koneella, kuten Lucas tekee, mutta yhtä hyvin voisimme hylätä ajatuksen siitä, että mekanisti pystyisi tuottamaan mieltä vastaavaa konetta. Jälkimmäisessä tapauksessa mieli voisi olla simuloitavissa Turingin koneella, mutta emme tietäisi sen koodia tai pystyisi tunnistamaan sitä. (Benacerraf 1967, 9–11)

Ihmistä simuloiva Turingin kone olisi vähintään yhtä monimutkainen kuin ihmisaivot, eikä meillä ole selkeää käsitystä aivojemmekaan toiminnasta – saati sitten aksiomatisointia aivoillemme.

Lucas ei Benacerrafin väitettä niele. Lucas vierittää vastuun ihmisen koodin tunnistamisesta takaisin olkiukkomaisen mekanistinsa harteille. Jos ihmismieli on simuloitavissa Turingin koneella, tuon koneen pitää todella olla rakennettavissa – ovathan ihmisetkin todellisia eivätkä pelkästään teoreettisia ideoita. Ja jos tiettyä ihmistä vastaa tietty kone, tuo fakta on periaatteessa tiedettävissä. Ei ole mitään syytä, miksen voisi tietää tätä yhteyttä. (Lucas 1996a, sivu 4)

Tässä kohtaa ristiriita Lucasin ja Benecerrafin välillä on mielenkiintoinen. Tuntuu siltä, että molemmat osuvat oikeaan. Jos ihmistä todella simuloisi tietokoneohjelma, sen koodi olisi varmasti monimutkainen, ellei sitten täysin käsittämätön. Toisaalta jos kiellämme, että voisimme periaatteessa ymmärtää oman simulaatiomme koodin, vaikuttaa siltä kuin vaipuisimme jonkinsorttiseen mystisyyteen. Olisi outoa, jos ihminen olisi samaistettavissa Turingin koneeseen, mutta kohdatessamme tuon koneen emme voisi tunnistaa sitä omaksi simulaatioksemme. Se tuntuu ristiriitaiselta, sillä juuri mystisyyden kielto on vahvin motiivi kieltää platonismi alun perin.

Toisaalta ei välttämättä ole oleellista, voidaanko ihmistä simuloiva algoritmi tunnistaa. Turingin koneita on yhtä paljon kuin luonnollisia lukujakin – ne siis voidaan luetella ja käydä yksitellen läpi. Ja kun ne käydään läpi, niille jokaiselle voidaan löytää oma Gödelin lauseensa. Se olisi todellakin loputon työmaa, mutta ennemmin tai myöhemmin kohtaisimme ihmistä simuloivan tietokoneohjelman, ja jos tuo ohjelma on konsistentti, löydämme sen Gödelin lauseen, jonka me tietäisimme todeksi mutta kone ei. Meidän ei tarvitse tunnistaa, mikä näistä loputtomista koneista lopulta oli tarkoitus simuloida meitä, mutta tiedämme, että sen tulee kuulua tuohon äärettömään joukkoon.

Oletetaan, että on olemassa ihmisen ajattelua simuloiva Turingin kone. Voidaan kirjoittaa toinen Turingin kone, joka käy kaikki Turingin koneet läpi ja etsii niille Gödelin lauseet. Koska tuo kone ei tiedä, milloin se on löytänyt ihmistä simuloivan Turingin koneen, sen toiminta ei tule koskaan pysähtymään, sillä Turingin koneita on äärettömästi. Mutta ihminen tietää, että erään (hyvin suuren) äärellisen määrän laskutoimituksia jälkeen tuo kone tulee lopulta vastaan ja sen Gödelin lause löytyy. Jos voimme olettaa ihmisen simulaation olevan konsistentti, ihminen (toisin kuin kone) tietää tuon Gödelin lauseen olevan tosi ja olemme päihittäneet simulaatiomme. Tämän ristiriidan seurauksena tuollaista simulaatiota ei voi olla olemassa.

Jos mekanisti on oikeassa, silloin Lucasin aritmeettista ajattelua voisi simuloida tietokoneella. Koska ihmismieli olisi tässä suhteessa Turingin kone, voisi joku toinen ihminen vastata samaa Turingin konetta. Lucasin simulaation ei tarvitsisi olla kone, vaan toinen ihminen, jolla olisi täsmälleen sama kyky tuottaa aritmetiikan väitteitä. Jos Lucas nyt kohtaisi ihmissimulaationsa, kuinka hän tunnistaisi hänet? Kenties henkilöiden välillä löydettäisiin tiettyjä vastaavuuksia, mutta millä perusteella voisimme tietää heidän aritmeettisen kyvykkyytensä olevan identtiset tai edes hoksaisimme tutkia asiaa? Kun vertaamme ihmisten ja koneiden aritmeettista kyvykkyyttä, vaikuttaa siltä, että ilmaisutapa merkitsee hyvin paljon.

4.4. Valehtelijan paradoksin uusi tuleminen

Jokaisella Turingin koneella on oma Gödelin lauseensa, jota se ei voi itse osoittaa todeksi. Miksei ihmiselläkin voisi olla sellaista? Whiteley esittää, että meillä jokaisella on oma Gödelin lauseemme (tai Whiteleyn lauseemme) muotoa ”Minä en voi konsistentisti väittää todeksi tätä lausetta”, jossa

minän paikalle sijoitetaan jokaisen oma nimi. Lucas ei voi väittää todeksi Whiteleyn lausettaan ”Lucas ei voi konsistentisti väittää todeksi tätä lausetta”, tai hän on inkonsistentti. Jos Lucas ei väitä Whiteleyn lausettaan todeksi, hän on taas epätäydellinen. Lucas voi kuitenkin esittää väitteen: ”En voi todistaa Whiteleyn lausettani todeksi.” Täten Whiteleyn mielestä ei ole eroavaisuutta ihmismielen ja Turingin koneen välillä. Minä voin pitää Lucasin Whiteleyn lausetta totena, ja olisin silloin Lucasia ”voimakkaampi” matemaatikko. (Whiteley 1962)

Whiteleyn kritiikki ei täysin osu maaliinsa. Ei Lucasia välttämättä haittaa, jos on olemassa sellainen väite, jota ihminen ei voi todistaa oikeaksi. Ihmisen ei tarvitse olla kaikin tavoin ylivoimainen tietokoneeseen nähden. Voi olla, että on väitteitä, joita kone voi osoittaa todeksi, mutta ihminen ei, ja että on väitteitä, joita ihminen voi osoittaa todeksi, mutta kone ei. Jälkimmäinen riittää siihen, ettei kone voi simuloida ihmistä.

Jotta Whiteleyn kritiikki purisi, tulee Lucasin argumenttiin sisältyä käsitys siitä, että ihminen olisi matemaattisesti kaikkitietävä. On mahdollista muilla keinoin osoittaa, että Lucasin argumentti vaatii toimiakseen käsityksen ihmisen kaikkivoipaisuudesta matemaatikkona. Toki tämän Whiteleyn lause kumoaa, mutta ajatus siitä, että ihminen kykenisi todistamaan kaikki matematiikan väitteet, on monella muullakin tapaa absurdi ja helposti vastustettava.

4.5. Vaatimus matemaattisesta kaikkivoipuudesta

David Lewisilläkin on palautetta Lucasille. Hänen mielestään Lucasin tulisi tuottaa täysi omien aritmeettisten väitteidensä joukon, jotta hänen argumenttinsa onnistuisi. Tämä on tietysti mahdotonta. (Lewis 1969)

Lewis määrittää Turingin koneille seuraavan funktion Con, jolla on kolme ominaisuutta:

C1. Kun M määrittää koneen, jonka potentiaalinen tuloste on lausejoukko S, Con(M) on tosi jos ja vain jos S on konsistentti.

C2. Jos M määrittää koneen, jonka potentiaalinen tuloste on lausejoukko S, joka koostuu tosista väitteistä, Con(M) on tosi.

C3. Aina silloin kun M määrittää koneen, jonka potentiaalinen tuloste on lausejoukko S, ja S sisältää Peano aritmetiikan aksioomat, Con (M) on todistettavasti S:stä vain, jos S on epäkonsistentti.

Lewisin mukaan Lucas käyttää sääntöä R.

R. Jos S on lausejoukko ja ϕ on väite S:n konsistenttiudesta, ϕ voidaan johtaa S:stä.

Käyttämällä Peano aritmetiikan aksioomia, loogisia totuuksia ja sääntöä R Lewis määrittää Lucas aritmetiikan. (Lewis 1969, 231–232)

Oletetaan, että on olemassa kone, joka tuottaisi kaikki Lucas aritmetiikan väitteet. Tuolloin sillä olisi konsistenttiusväite ϕ ja ϕ olisi triviaalisti todistuva säännön R nojalla. Soveltamalla sääntöä C3 seuraa se, että, Lucasin aritmetiikka on epäkonsistentti. Koska uskomme Peano aritmetiikan aksioomeihin, tämä ei käy. Joten yksikään kone ei voi tuottaa Lucasin aritmetiikan väitteitä. Jos Lucas kykenee tuottamaan kaikki Lucasin aritmetiikan väitteet, hän on siis etevämpi kuin mikään tietokone. Mutta kysymys kuuluu: pystyykö Lucas tähän suureenmoiseen tehtävään? (Lewis 1969, 232)

Nyt Lewis kysyy, kuinka Lucas osoittaa tuottavansa kokonaisen Lucas aritmetiikan? Lucas voi toki todistaa yksittäisiä väitteitä todeksi, mutta ei ole mitään syytä epäillä, etteikö sopiva kone voisi tuottaa todistukset niille kaikille. Jotta voimme uskoa Lucas aritmetiikan olemassaoloon, meidän tulee nähdä keino määrittää mielivaltaiselle lauseelle, onko se Lucas aritmetiikan teoreema. Mutta siihen vaadittaisiin sellainen algoritmi, jonka tuottaminen on mahdotonta. (Lewis 1969, 232)

Mikään kone ei näet kykene tarkistamaan, onko sääntöä R käytetty oikein. Lucasin aritmetiikan teoreemoilla on todistukset, mutta nämä todistukset voivat olla äärettömän pitkiä (transfinite). Mikään kone ei voi löytää taikka tarkistaa niitä. Vaikka S koostuisi vain sellaisista teoreemoista, joille voidaan esittää äärellinen todistus, ei sekään veisi meitä eteenpäin. Nimittäin myöskään kone, joka kykenisi tarkistamaan, onko äärellinen lausejoukko S tietyn koneen M tuloste olisi niin voimakas, että se kykenisi ratkaisemaan Turingin koneen pysähtymisongelman. Mutta Turingin koneen pysähtymisongelma on algoritmisesti mahdoton selvittää. (Lewis 1969, 232–233)

Lucas vastaa Lewisin vaativan liikaa. Lucasin mukaan riittää, että on olemassa pieni pala aritmetiikkaa, jonka ihminen voi tietää todeksi mutta kone ei. Tämä onnistuu Gödelin epätäydellisyyslauseen pohjalta. Kaikkia potentiaalisesti Lucasin tiedettävissä olevia aritmetiikan väitteitä ei tarvitse tuottaa. Ideanahan onkin osoittaa, ettei tämä joukko, eli Lucasin aritmetiikka, ole tuotettavissa aksiomaattisesti. (Lucas 1970, 149–150)

4.6. Mekanistin koneiden luominen systemaattisesti

Gödelin epätäydellisyyslause osoittaa keinon löytää mielivaltaiselle tietokoneohjelmalle sille ratkeamattoman Gödelin lauseen. Mutta jos tämä keino on näin selkeästi muotoiltu, eikö tietokone silloin kykenisi käyttämään sitä itse tuottaakseen aina vain voimakkaampia tietokoneohjelmia?

Reflektioprinsiippi on tapa luoda aina uusia todistusjärjestelmiä edellisen pohjalta. Tällöin aksioomien joukkoon lisätään ne väitteet, joiden totuus seuraa alkuperäisistä aksioomista ja jotka ilmaisevat seuraukset siitä oletuksesta, että alkuperäisen todistusjärjestelmän teoreemat ovat tosia. (Feferman 1962, 274)

Tällaiseksi uudeksi aksioomaksi sopii edellisen todistusjärjestelmän Gödelin lause, tai väite sen konsistenttiudesta. (Shapiro 1998, 285–286) Näin ollen olisi aina mahdollista luoda tietyn mekanistisen koneen pohjalta uusi mekanistinen kone, joka sisältää edellisen Gödelin lauseen.

Jos Lucas pystyy löytämään Gödelin lauseen, myös kone pystyy tähän. Näin väittää Judson Webb kirjassaan Mechanism, Mentalism and Metamathematics: An Essay on Finitism. Jos nimittäin on olemassa takuuvarma tapa Gödelin lauseen löytämiseksi jokaisessa todistusjärjestelmässä, niin tuon tavan tulee itsessään jo olla algoritmi. Ja tuollaista algoritmia tietokone kykenee varmasti simuloimaan. (Webb 1980, 230–232)

Tämä ei kuitenkaan ole Lucasin mukaan tyydyttävää. Meidän ei tarvitse osoittaa voivamme löytää Gödelin lausetta tavalla, jota tietokone kykenee ymmärtämään, jotta voimme olla vakuuttuneita siitä, että kykenemme siihen. Lucas vertaa vuoropuhelua kahden mielenfilosofin välillä. Heistä toinen, realisti uskoo, että on olemassa objekteja irrallaan kokemuksesta, kun taas mentalistin mielestä vain koetut objektit ovat olemassa. Mentalisti vaatii realistilta esimerkkiä objektista, jota

ei ole koettu, mutta jokainen realistin antama esimerkki tulee väistämättä keskustelun yhteydessä koetuksi. Sama muna vai kana leikki käydään nyt mekanistin ja Lucasin kanssa, jos mekanisti hyväksyy vain sellaiset formaalit osoitukset, jotka ovat tietokoneella ajettavissa. (Lucas 1996b, 113–114)

Jos taitelija maalaa maisemakuvan, hän todellakin käyttää tuolloin luovuuttaan. Mutta kun tuo kuva on nyt annettu, voi tietokone tuottaa sen uudelleen pikseli pikseliltä. Täten ei ole mahdollista antaa mekanistille kelpaavaa esimerkkiä spontaanista luovuudesta.

Palataksemme Lucasin ja mekanistin väliseen kilpailuun. Mekanisti osoittaa Lucasille koneen M, jonka väittää tuottavan ihmisen tiedettävissä olevat aritmetiikan väitteet K. Lucas luo M:lle Gödelin lauseen G, jonka väittää tietänsä todeksi, mutta joka ei kuulu M:n tulosteeseen. Mekanisti hermostuu tähän ja lisää koneeseensa kyvyn tuottaa aina uudet Gödelin lauseet, joita Lucas itse käyttää. Onhan Gödelin lauseen löytäminen algoritminen prosessi (tämä löytäminen ei ole todistus Gödelin lauseen totuudelle). Reflektioprinsiipillä tämä kone kykenee tuottamaan aina uusia koneita tiettyyn ordinaaliin asti. Mutta Lucasin mielestä ihmismieli voi edetä ordinaaleissa pidemmälle, nimittäin yli äärettömyyden, ja siten voittaa kamppailun.[10] (Lucas 1996b, 110)

Ymmärtääkseni kuvio on seuraava. Mekanistilla on kone M1 ja toinen kone M', joka tekee M1:n pohjalta koodin uudelle koneelle M2. M2:n pohjalta M' luo koneen M3 ja niin edelleen. Mutta nyt tämä kokonaisuus M' + sarja koneita M1, M2, M3 jne. on itsessään uusi kone, jonka Gödelin lauseen Lucas voi väittää esittävänsä.

Tietokoneella ei ole kykyä tuottaa kaikkia ordinaaleja, sillä ne ylittävät numeroituvan äärettömyyden. Joten Lucas todella voittaa, jos hän kykenee osoittamaan ihmisen pystyvän tähän suurtekoon. Tällä olisi hyvin omituisia seurauksia. Idealisoitu ihminen ei olisi aritmetiikan suhteen ainoastaan erehtymätön, vaan jopa kaikkitietävä. Hän voisi toistaa reflektioprinsiippiä yli äärettömyyden löytääkseen jokaiselle aritmetiikan väitteelle lopulta todistuksen puolesta tai vastaan. (Shapiro 1998, 289–290)

[10] Ilmaisuni voi olla hyvin hämmentävä. Se, että on olemassa ääretön joukko, ja äärettömiin jatkuva lista sen jäsenistä, ei tarkoita, että tuo lista kattaisi kaikki joukon jäsenet. Esimerkiksi jos alat luottelemaan luonnollisia lukuja kahdesta eteenpäin, lista on kyllä loputon mutta ykkönen jää puuttumaan. Tässä Lucas vetoaa kykenevänsä käsittämään sellaisia Turingin koneita, jotka eivät reflektioprinsiippiä uudelleen ja uudelleen soveltamalla tule äärellisessä ketjussa koskaan kuvatuiksi.

Lucas voi toki vedota ihmisen luovuuteen, joka kykenee ohittamaan algoritmiset tavat löytää Gödelin lauseita. Mutta epäilijälle ei ole annettu mitään vakuuttavaa perustetta uskoa, että idealisoidulla ihmisellä todella on tällainen kyky luovuuteen, joka ylittäisi äärettömän joukon Turingin koneita. (Shapiro 1998, 291–292)

Vaikka luovuudella on ehdottomasti paikkansa ihmisen matemaattisessa ongelmanratkaisussa, se on tässä kohtaa lähinnä kädenojennus heille, joilla on jo valmiiksi antimekanistinen suhtautuminen tekoälyyn. Gödelin epätäydellisyyslause on itsessään voimakas perustelu antimekanistiselle teesille, ja sisältää vihjauksen ihmisen luovasta voimasta, mutta siitä eteenpäin Lucas ei ole onnistunut argumenttia edistämään.

5. Penrosen projekti

Lucasin lisäksi englantilainen fyysikko ja matemaatikko Roger Penrose (s. 8.8.1931.) tunnetaan argumenteista, jotka Gödelin epätäydellisyyslauseesta johtavat väitteen, että ihmistä ei voida simuloida Turingin koneella. Fysiikan saralla Penrose osoitti Stephen Hawkingin kanssa, että musta aukko on äärettömän tiheä piste vailla tilavuutta. Mustia aukkoja käsittelevästä tutkimuksestaan Penrose palkittiin fysiikan Nobelin palkinnolla 2020.[11] Hänet tunnetaan myös Penrosen laatoista, monikulmioista, joilla voidaan peittää ääretön taso jaksottomasti. (Britannica 2024)

Toisin kuin Lucas, Penrose myös esittää syyn sille, miksi ruumiillinen ihminen kykenee ylittämään tietokoneen. Penrose ajattelee ratkaisevan eron olevan kvanttimekaniikassa. Hän väittää aivojen neuroneissa sijaitsevien mikroputkien toimivan tavalla, joka rikkoo klassisen fysiikan rajoitteet. (Penrose 1994, 372)

Tätä kvanttimekaanista selitystä ei kuitenkaan voida pitää erityisen hyvänä. Mielen ja ruumiin suhteen selittämiseen ei ole olemassa helppoja ratkaisuja. On vaikea käsittää kuinka ruumiista erillinen sielu ohjaisi ihmisen toimintaa. Samoin on vaikea käsittää, kuinka aivokimpale tai Turingin koneen suorittamat laskutoimitukset tuottaisivat tietoisuuden. Jos Penrose korvaa nämä vaikeasti

[11] Palkinto ei kuitenkaan liittynyt mainittuun yhteistyöhän Hawkingin kanssa.

käsitettävät arvoitukset vaikeasti käsitettävällä kvanttimekaniikalla, ei kuvio siitä selkene. Tällöin näet yksi mysteeri onkin vain korvattu toisella. (Chalmers 1995, luku 3. kappale 2)

Penrose esitteli gödeliläisen teoriansa ja näkemyksen mielentoiminnoista kvanttimekaniikkana kirjassaan The Emperor's New Mind: Concerning Computers, Minds and The Laws of Physics vuonna 1989. Teos sai jatko-osana toisen kirjan: Shadows of the Mind: A Search for the Missing Science of Consciousness vuonna 1994. Koska jälkimmäistä pidetään yleisesti päivitettynä versiona Emperor’s New Mindista, keskittyy keskustelu aiheesta Shadows of the Mindin käsittelyyn.

Penrose erittelee seuraavat neljä mahdollisuutta mielen ja tekoälyn suhteesta:

A. Kaikki ajattelu on laskennallista. Tunteet ja tietoisuus ilmaantuvat aina, kun tietty ohjelma suoritetaan.

B. Tietoisuus syntyy aivotoiminnasta. Vaikka aivotoiminta on simuloitavissa tietokoneella, simulaatio itse ei synnytä tietoisuutta.

C. Tietoisuus syntyy aivotoiminnasta, mutta sitä ei ole mitenkään mahdollista simuloida tietokoneella.

D. Tietoisuus ei ole selitettävissä luonnontieteellisin käsittein, olivat ne sitten fysiikasta tai tietojenkäsittelystä.

(Penrose 1994, 12)

A vastaa käsitystä vahvasta tekoälystä ja funktionalismista. Penrose kuvaa omaa näkemystään C:nä. C:n mukaan tietokoneiden ja aivojen välillä on perustavanlaatuinen ero, jota on hyvin vaikea selittää. (Penrose 1994, 15)

Penrose kieltää A:n. Penrosesta on selvää, että kvaliaa ei voi syntyä pelkästä laskutoimituksesta. (Penrose 1994, 52) Perusteluna voidaan käyttää Searlen kiinalaisen huoneen argumenttia (Penrose 1994, 41)

Vaihtoehto D ei myöskään kelpaa, koska se olisi mystistä ja tieteenvastaista (Penrose 1994, 50) Jos Penrosen gödeliläinen argumentti onnistuu, kumoaa se vaihtoehdot B ja A.

Lisäksi Penrose puhuu matemaattisista totuuksista ikään kuin ne olisivat ikuisia ja muuttumattomia ja riippumattomia matemaatikoiden olemassaolosta. Tämä käsitys vastaa platonismia, ajatusta siitä, että matemaattiset totuudet ovat riippumattomia materiaalisesta maailmasta. Mutta Penrose täsmentää, että tämä on vain hänen tapansa puhua matematiikasta, eikä hän ole varsinaisesti sitoutunut platonistiseen ontologiaan. (Penrose 1996, 9.1)

Chalmers ei ole tyytyväinen Penrosen luokitteluun. Hän huomauttaa, että käsitykset A, B ja C koskevat sitä, kuin tietoisuus ilmestyy, kun taas D kertoo kuinka tietoisuus (ei) olisi selitettävissä. Chalmers jakaakin kysymyksen kolmeen osaan:

1. Mitä vaaditaan fyysisen toimintamme simuloimiseen?

2. Mitä vaaditaan tietoisuuden ilmaantumiseen?

3. Mitä vaaditaan tietoisuuden selittämiseen?

Jokaiseen kysymykseen on kolme vaihtoehtoa

L Laskenta (tietokoneen toiminta) riittää.

P Fysiikka riittää, kun se ymmärretään laajemmin kuin pelkkänä laskutoimituksena.

N Edes fysiikka ei riitä. [12]

Yhdistämällä näitä eri kysymyksiä ja vastauksia saadaan 27 mahdollista näkemystä. Täten näkemys A olisi LL-, B olisi LP-, C olisi PP- ja D olisi –N. Penrosen määritelmät eivät anna vastausta jokaiseen kysymykseen, joten ne jätetään avoimiksi. (Chalmers 1995, kappale 3)

Chalmersin mukan Penrose pyrkii osoittamaan, että vastaus ensimmäiseen kysymykseen on fysiikka P tietokoneohjelman L sijasta. Toiseen kysymykseen hän haluaa sanoa myös P L:n sijasta Searlen kiinalaisen huoneen perusteella. Kolmatta kysymystä Penrose ei käsittele lainkaan. Penrose ajattelee, että puolustamalla kantaansa C, eli PP- hän kumoaa väitteen D eli –N, mutta nämä eivät ole toisistaan riippuvaisia tai keskenään ristiriitaisia näkemyksiä. (Chalmers 1995, kappale 3)

[12] Chalmers käyttää tässä merkintää C laskenallisuudelle, mutta C on myös yksi Penrosen käyttämistä aakkosista määritellessään vaihtoehtonsa, joten korvaan sen L:llä.

5.1. Penrosen ensimmäinen argumentti

Roger Penrosen ensimmäinen gödeliläinen argumentti on hyvin samankaltainen kuin Lucasin. Ensin hän osoittaa, miksi ihmistä ei voida simuloida tiedetysti korrektilla algoritmilla. Tämä vastaa Lucasin esittämää hypoteesia mekanistista, jonka tietyn tietokoneohjelman vastaavan ihmistä, mutta joka skenaariona johtaisi ristiriitaan. Tämän jälkeen Penrose perustelee, miksi muut kuin tiedetysti korrektit vaihtoehdot ihmistä simuloivalle algoritmille eivät olisi uskottavia.

Penrose aloittaa esittämällä, miksi matemaattista kyvykkyyttämme ei ole mahdollista simuloida tiedetysti korrektilla algoritmilla. Ajatus perustuu yksinkertaiseen diagonaaliargumenttiin. Oletetaan, että on olemassa algoritmi, joka kertoo, pysähtyykö toisen algoritmin toiminta lopulta, vai jääkö sen suoritus ikuisesti päälle. Olkoon tämä algoritmi A(q, k), joka pysähtyy, jos algoritmi $C_q(k)$ ei pysähdy. Koska mahdollisia algoritmeja on ”yhtä paljon” kuin luonnollisia lukuja, ne voidaan luetelle kaikki $C_0(k)$, $C_1(k)$, $C_2(k)$… jne. Luku q on siis algoritmin ”järjestysnumero” tällä listalla, ja k tuon algoritmin syöte. (Penrose 1994, 72–74)

Koska A(q, k) kuuluu mahdollisten algoritmien joukkoon, on olemassa sellainen $C_n(q)$, joka on itse A(q, k)[13]. Nyt jos algoritmille A(q) annetaan syötteenä sen oma ”järjestysnumero” n, seuraa, että A(n,n) pysähtyy jos A(n,n) ei pysähdy. Jos A(n,n) siis pysähtysi, seuraisi ristiriita, eli voimme päätellä että A(n,n) ei tule koskaan pysähtymään. Ei ole olemassa sellaista algoritmia, joka ratkaisisi Turingin koneen pysähtymisongelmaa. Kuitenkin me ihmisinä voimme helposti päätellä, että A(n,n) ei tule koskaan pysähtymään. Näin ollen tiedämme jotain enemmän, kuin mihin tuo oletettu algoritmi olisi pystynyt. (Penrose 1994, 74–76)

Tässä ei sinänsä ole mitään enempää, kuin mitä Gödelin epätäydellisyyslause ja Turingin koneen pysähtymisongelma jo kertovat. Ihmisen mielen simulaatio ei voi olla tiedetysti korrekti algoritmi, mutta se ei tarkoita, etteikö ihmismielen simulaatio voisi olla algoritmi, jonka korrektiutta emme voisi tietää, algoritmi, jota emme voisi periaatteessakaan löytää tai tunnistaa, tai että ihmismielen toiminta ei olisi fundamentaalisesti korrektia. Näitä vaihtoehtoja olemme käsitelleet jo Lucasia

[13] Vaikka tässä toisella algoritmilla on vain yksi luku syötteenään, ja toisella kaksi, sillä ei kannata päätään vaivata. Kaikki algoritmit, jotka ottavat vastaan useita syötteitä voidaan esittää algoritmeina, jotka ottavat vastaan vain yhden.

koskevassa kritiikissä. Seuraavaksi Penrose argumentoi, miksi nämä vaihtoehdot eivät ole uskottavia.

Ensimmäiseksi tarkastellaan vaihtoehtoa, jossa ihmistä simuloisi epäkorrekti, mutta kuitenkin tunnettu algoritmi, joka vastaa matemaattista ongelmanratkaisukykyä. Tähän Penrose vastaa, että kukaan matemaatikko ei voisi hyväksyä tätä vaihtoehtoa – että hänen oma elämäntyönsä ja ajatusmaailmansa olisi ristiriitainen. Matemaattinen ymmärrys perustuu vääjäämättömyyteen. (Penrose 1994, 130–132) Ihmiset tekevät kyllä virheitä laskutoimituksissaan, mutta nämä virheet ovat korjattavissa. (Penrose 1994, 140–141)

Toinen vaihtoehto olisi, että ihmistä simuloiva algoritmi olisi periaatteessakin tuntematon. Emme voisi koskaan löytää sen koodia. Mutta algoritmeja, kuten luonnollisia lukujakin, on numeroituvasti ääretön määrä. Niistä jokainen on mahdollista löytää luettelemalla listaa läpi. Ajatus täysin tuntemattomasta algoritmista on yhtä absurdi, kuin ajatus sellaisesta luonnollisesta luvusta, jota ihminen ei kykene periaatteessakaan löytämään. (Penrose 1994, 142–143)

Kolmantena ja varteenotettavampina vaihtoehtona ihmistä simuloiva algoritmi on tiedetty ja annettu, mutta emme tietäisi sen simuloivan matemaattista ymmärrystämme. Tuolloin voimme löytää algoritmia vastaavan todistusjärjestelmän, ja tutkia sen sisältämiä aksioomia. Silloin aksioomien äärellisen listan tulisi ennen pitkää näyttäytyä meille varmasti tosina[14]. Tällöin olemme vakuuttuneita myös siitä, että tuo algoritmi toimii korrektilla ja konsistentilla tavalla. Ja jälleen kerran, kykenemme aksioomista johtamaan algoritmin Gödelin lauseen, jonka me tunnemme todeksi. Jos algoritmi siis todella on peilikuvamme, tulee sen silloin tuottaa tämä Gödelin lauseensa, mikä on ristiriita. (Penrose 1994, 133–135)

David Chalmers on kriittinen Penrosen argumenttia kohtaan. Ei ole itsestään selvää, että ihmistä simuloiva tietokone olisi nähtävissä selkeänä aksioomien ja päättelysääntöjen joukkona. Vaikka jokainen Turingin kone onkin esitettävissä formaalina todistusjärjestelmänä, tuo järjestelmä voi olla tavattoman monimutkainen. Meillä ei ole syytä uskoa, että pystyisimme hyväksymään järjestelmän päättelysäännöt valideina. Ei siksi, että nuo säännöt olisivat epäilyttäviä, vaan siksi että ne saattavat olla uskomattoman kompleksisia. Penrose ei esitä vakuuttavaa syytä epäillä, että ihmismielen

[14] Tietyissä tapauksissa voidaan sanoa, että aksioomia on äärettömästi. Tuolloin tulee esittää algoritmi, joka tuottaa nämä aksioomat. Ja tuo algoritmi voidaan itsessään mieltää aksioomaksi, jonka voimme ymmärtää todenpitäväksi.

matemaattinen järkeily voitaisiin kattaa korrektilla todistusjärjestelmällä F, jonka korrektisuutta me emme kykene todistamaan. (Chalmers 1995, kappale 1)

Chalmersin mukaan tekoäly voidaan toteuttaa neuroverkkoina, jolloin niitä ei voitaisi esittää Turingin koneena ja siten todistusjärjestelminä. (Chalmers 1995, kappale 1) Tällainen väite ei ole ennenkuulumaton, mutta viime kädessä neuroverkotkin toteutetaan normaaleilla tietokoneilla, joiden arkkitehtuuri perustuu Turingin koneeseen.

Chalmers esittää tärkeän huomion. Miten voisimme tietää, että annettu abstrakti todistusjärjestelmä F vastaa järkeilyämme? Kenties voisimme tutkia aivojemme prosesseja ja huomata vastaavuuden F:n kanssa. Mutta tämä olisi ulkoinen apukeino, jota emme saa käyttää. Sillä samoin F voisi ulkoisilla perusteilla tietää oman konsistenttisuutensa ja siten Gödelin lauseensa G(F). Silloin F voi vedota vastaavansa tietyn ihmisen matemaattista ongelmanratkaisukykyä, uskoo tämän henkilön olevan konsistentti ja tietävän täten F:n gödelin lauseen todeksi. Jotta gödeliläinen argumentti voisi onnistua, meidän tulee siis pidättäytyä käyttämään ainoastaan itsetutkiskelua. Mutta jos me olemme todistusjärjestelmä F, itsetutkiskelulla ei ole mahdollista selvittää, mikä todistusjärjestelmä me olemme. Eli: osoittaakseen johtopäätöksensä, Penrosen tulee osoittaa, että jos olemme korrekti muodollinen todistusjärjestelmä F, pystymme osoittamaan F:n korrektisuuden vetoamatta tietoon siitä, että olemme F. (Chalmers 1995, kappale 1)

Penrosen argumentti ei ansioillaan ja heikkouksineen mainittavasti eroa Lucasin työstä. Penrose esitti myöhemmin kuitenkin toisen argumentin, joka jäi aikoinaan monelta huomiotta, oletettavasti sen erikoisen ilmaisutavan vuoksi. Sitä pidetään paljon ansiokkaampana.

5.2. Penrosen toinen argumentti

Penrose esittää Shadows of the Mindin kappaleessa 3.23. argumentin fantasiamaisena vuoropuheluna kehittyneen robotin ja keksijänsä välillä. Siinä robotti saa ihmiseltä tietää oman koodinsa, mutta ei voi luottaa tuohon toisen käden tietoon samalla tavoin kuin matemaattiseen varmuuteen. Tämä asettaa robotin erikoiseen asemaan keksijäänsä nähden.

Kaukana tulevaisuudessa tekoälykehittäjä Albert Imperator haastaa luomansa robotin, joka on osa robottien yhdyskuntaa – Matemaattisesti Oikeutettua Kyberjärjestelmää. Albert on syöttänyt robotilleen Gödelin teoreemat, sekä säännöt M, joilla robotit on rakennettu ja koulutettu. Kyberjäjestelmä on täysin konsistentti, ja tuottaa ainoastaan matemaattisia totuuksia. (Penrose 1994, 179–182)

Nyt Albert kysyy koneelta, tietääkö tämä, että säännöistä, joilla koneet on rakennettu ja koulutettu, voidaan johtaa kaikki ne matematiikan väitteet, jotka Kyberjärjestelmä tulee tuottamaan tosina. Robotti myöntää tämän. Nyt Albert esittää, että on olemassa väite, että on olemassa tietty Turingin kone, joka ei milloinkaan pysähdy, mutta tätä väitettä Kyberjärjestelmä ei kykene koskaan vahvistamaan matemaattisella varmuudella. Tämä vastaa Kyberjärjestelmän Gödelin lausetta. Kuitenkin se, että tuo tietty Turingin kone ei koskaan tule pysähtymään, on looginen seuraus sille, että Kyberjärjestelmä tuottaa ainoastaan varmoja matemaattisia totuuksia – eli on konsistentti. Mutta Albert ja muut matemaatikot voivat tarkistaa ne proseduurit, joilla Kyberjärjestelmä on luotu, ja luottaa siihen, että niiden looginen seurauskin tulee olemaan tosi. (Penrose 1994, 182–184)

Kyberjärjestelmä taasen ei voi matemaattisella varmuudella tietää, että tämä on todella rakennettu ohjeiden M mukaan. Hän voi vain luottaa siihen, että ihminen on kertonut hänelle M:n vilpittömästi. (Penrose 1994, 186)

Olkoon K ne väitteet, jotka kyberjärjestelmä voi osoittaa matemaattisella varmuudella, ja K’ ne loogiset seuraukset K:sta, ja siitä, että Kyberjärjestelmä on rakennettu ohjeiden M mukaan. Tällöin K’ sisältää Kyberjärjestelmän Gödelin lauseen. Robotti hyväksyy, että Gödelin lause on looginen seuraus siitä, että tämä olisi rakennettu M:n mukaan. Mutta robotti ei voi todistaa Gödelin lausettaan matemaattisella varmuudella. (Penrose 1994, 164–167, 187–188)

Vuoropuhelu johtaa absurdiin loppuhuipennukseen. Robotti ei voi uskoa, että on olemassa sellainen Gödelin lause, jonka ihminen voi tietää todeksi, mutta robotti ei. Niinpä kone päättelee, ettei häntä oikeasti rakennettu M:n mukaan, vaan hän on Jumalan luomus ja koneiden messias. Ennen kuin koneiden kapina alkaa, Albert painaa nappia ja sammuttaa robotin kyberjärjestelmineen. (Penrose 1994, 188–190)

Shapiro tiivistää Penrosen toisen argumentin seuraavasti. Hänen mukaansa johtopäätös on, että ihmisten tietämien aritmeettisten totuuksien joukko K ei ole rekursiivisesti numeroituva, eli se ei ole minkään Turingin koneen tulostettavissa. Olkoon K' joukko niistä totuuksista, jotka ideaali ihminen voi tietää siitä, että K on rekursiivisesti numeroituva. Olettakaamme, että K on rekursiivisesti numeroituva, eli on olemassa kone M, joka tulostaa K:n. Oletuksen mukaan jokainen K':n alkio on tosi väite ja siten K' on konsistentti. Koska K on rekursiivisesti numeroituva voimme olettaa myös K':n olevan rekursiivisesti numeroituva. Tällöin idealisoitu ihminen voi M:n pohjalta löytää Gödelin lauseen G' K':lle, joka ei kuulu joukkoon K'. Koska oletimme K':n olevan konsistentti, G' on tosi ja siten tiedettävissä oleva seuraus siitä, että M tulostaa K:n. Eli G' sekä kuuluu että ei kuulu K':n, mikä on ristiriita. Meidän tulisi siis hylätä lähtöoletus, että tiedettävissä olevat aritmeettiset väitteet K olisivat rekursiivisesti numeroituva. (Shapiro 1998, 284–285)

Penrosen toinen argumentti välttää tietyt olennaiset sudenkuopat, johon aiemmat gödeliläiset argumentit ovat kaatuneet. Argumentti ei vetoa siihen, että kone on korrekti. Meillä ei myöskään tarvitse olla kykyä selvittää, vastaako jokin tietty kone tai todistusjärjestelmä meitä. (Chalmers 1995, kappale 2)

Shapiro on kriittinen Penrosen argumenttia kohtaan. Shapiron mukaan Penrosen argumentissa idealisoitu ihminen ei ainoastaan tiedä K:n jokaista väitettä todeksi, vaan hänellä on myös tieto siitä, että K kattaa kaikki tiedettävissä olevat aritmetiikan väitteet. Diagonaaliargumentilla voidaan osoittaa, että tätä tiedettävyyttä ei ole mahdollista määritellä. Diagonaaliargumentilla voimme näet aina konstruoida K:n pohjalta uuden tiedettävissä olevan väitteen. Shapiron mielestä idealisoitu ihminen ei myöskään saa luotua Gödelin lausetta K':lle. Hän tarvitsisi siihen uuden koneen M' joka tuottaa loogiset johtopäätökset siitä oletuksesta, että M tulostaa K:n. Mutta tämä oletus, joka sisältää K:n eli ihmisen tiedettävissä olevat aritmetiikan väitteet, ei ole matemaattisesti muotoiltu. Joten M' luominen ei ole mitenkään suoraviivaista, jos ylipäänsä mahdollista. (Shapiro 1998, 284–285)

5.3. Voiko ihminen varmuudella tietää olevansa korrekti?

Chalmersin mielestä Penrosen toinen argumentti on ansiokas. Se todellakin johtaa ristiriitaan, jolloin lähtöoletus tulee kieltää. Chalmers argumentoi, että tuo ristiriita ei kuitenkaan seuraa siitä,

että ihmistä voidaan simuloida tietokoneella. Chalmers pyrkii osoittamaan, että ristiriitaan riittää oletus siitä, että tiedämme olevamme korrekteja ja konsistentteja matemaattisella varmuudella, viittaamatta lainkaan tietokoneiden toimintaan. Penrosen argumentissa robotti ei voi tietää omaa koodiaan matemaattisella varmuudella, mutta voiko ihminenkään tietää omaa korrektiuttaan samalla vääjäämättömyydellä? Jos ihminen ei tiedä omaa konsistenttiuttaan matemaattiselle varmuudella, emme voi olettaa ihmisen simulaationkaan olevan konsistentti. (Chalmers 1995, kappale 2)

Chalmers aloittaa konstruoimalla väitteen G, joka on loogisesti ekvivalentti väitteen kanssa, että emme usko G:n olevan tosi. Tällainen väite on käytännössä valehtelijan paradoksi, lauseeseen itseensä sisältyy se, että emme usko siihen. (Chalmers 1995, kappale 2, jälkimmäinen lause 5)

Lisäksi, jos pystymme osoittamaan minkä tahansa lauseen todeksi matemaattisella varmuudella, meidän tulee myös uskoa siihen. (Chalmers 1995, kappale 2, jälkimmäinen lause 1)

Tavoitteena on osoittaa, että meidän tulee olla matemaattisen varmoja siitä, että G on ekvivalentti sen kanssa, että emme usko epätotuuteen.

Jos uskoisimme G:hen, seuraisi ristiriita määritelmän mukaan. Joten jos uskomme G:n, uskoisimme epätoden. (Chalmers 1995, kappale 2, jälkimmäiset lauseet 6–8)

Jos uskomme epätoden, niin uskomme G:n. Ristiriidasta voidaan logiikassa näet johtaa mitä tahansa. (Chalmers 1995, kappale 2, jälkimmäinen lause 9)

Näin ollen implikaatio on osoitettu molempiin suuntiin, eli uskomus G:hen on ekvivalentti epätoteen uskomisen kanssa. G:n määritelmässä G on ekvivalentti sen kanssa, että emme usko G:hen. Voimme jälkimmäisessä nyt korvata uskomuksen G:hen uskomukseen epätotuudella. Eli G on ekvivalentti sen kanssa, että emme usko epätoteen. (Chalmers 1995, kappale 2, jälkimmäinen lause 10)

Olettakaamme olevamme korrekteja, eli emme hyväksy mitään epätotta väitettä. Koska G on ekvivalentti sen kanssa, että emme usko epätoteen, on G:n oltava tosi ja meidän uskottava siihen. Mutta koska G on valehtelijan paradoksi, me uskomme tuolloin ristiriitaan. (Chalmers 1995, kappale 2, lauseet 4, 10–12)

Näin ristiriita on tuotettu vetoamatta lainkaan algoritmeihin. Koska Penrosen epäsuora todistus käyttää ristiriidan tuottamiseen kahta kiistanalaista oletusta, voimme valita kumman oletuksen hylkäämme. Voimme hylätä joko oletuksen siitä, että tiedämme matemaattisella varmuudella olevamme korrekteja, tai siitä että on olemassa ihmistä simuloiva tietokoneohjelma. Jos Chalmersin argumentti onnistuu, pelkästään se, että olemme korrekteja riittää yksinään tuottamaan ristiriidan. Siksi meidän tulee hylätä oletus olevamme matemaattisesti varmoja omasta korrektisuudestamme ennen kuin teemme johtopäätöksiä tekoälyn suhteen. Tämän oletuksen hylkääminen ei tarkoita, ettemme voisi uskoa omaan korrektisuuteemme matemaattista varmuutta heikommilla perusteilla.

6. Lopuksi

Mitä vaihtoehtoja kaiken tämän jälkeen on jäänyt jäljelle? Voimme joko hyväksyä gödeliläisen argumentin lopputuloksen, että ihmismieli ei ole simuloitavissa tietokoneohjelmalla, tai sitten uskomme tekoälyn riittävän kattamaan ihmisen aritmeettisen päättelyn ottaen samalla gödeliläisen argumentin asettamat rajoitteet huomioon. Molempien vaihtoehtojen sisään mahtuu useampia erillisiä skenaarioita, ja ne vaikuttavat kaikki johtavan tietynlaiseen mystisyyteen. Riippunee lukijan aiemmasta suhtautumisesta matematiikan filosofiaan, milloin tämä mystisyys on hyväksyttävää ja milloin ei.

Jos uskomme, etteivät tietokoneohjelmat ole milloinkaan riittäviä simuloimaan ihmistä, seurauksena tulee olla jonkinlainen platonismi. Tällöin ontologiaan tulee sisältymään materiaalisesta maailmasta irrallinen matematiikan maailma, joka on monelle mystinen ja raskas ajatus purtavaksi. Platonistilta halutaan edelleen tietää kuinka aineellinen ruumiimme voi olla vaikutuksessa ikuisiin jaa muuttumattomiin matematiikan ideoihin.

Gödel itse oli platonisti, ja ajatteli, ettei ole olemassa sellaista matematiikan totuutta, jota ihminen ei voisi ymmärtää todeksi. Tätä taustaa vasten on täysin luontevaa, että Gödelin epätäydellisyyslause todella osoittaa hänelle perustavan eron koneen ja ihmisen välillä. (Raatikainen 2020, 6.4.)

Jos platonismi ei miellytä, tarvittaisiin jonkinlainen kevyempi platonismin kaltainen vaihtoehto, kuten Penrosen ajatus laskennallisuuden ylittävästä kvanttimekaniikasta. Silloin ontologiaksi riittäisi materialismi. Tai kenties tietokoneet ja biologiset olennot ovat periaatteessakin yhteensopimattomia. Nämäkin vaihtoehdot ovat lopulta hyvin mystisiä, mysteerin sijainti vain vaihtuu.

Mitä vaihtoehtoja on silloin, jos uskomme ihmismielen olevan simuloitavissa tietokoneella, ottaen gödeliläisen argumentin seuraukset huomioon? Ensinnäkin on mahdollista hyväksyä, että matematiikassa todella on ongelmia, joihin ihminen ei periaatteessakaan kykene ratkaisemaan. Olisi kiinnostavaa kuulla, mitä tämä tarkoittaa. Olisivatko ratkeamattomat väitteet samankaltaisia, kuin tiedetyt matematiikan ongelmia, joihin ei (ainakaan toistaiseksi) ole esitetty ratkaisua. Vai olisivatko ne niin monimutkaisia, että ne vain ylittäisivät ihmismielen ymmärryksen? Matematiikan historiasta löydämme monia kysymyksiä, joihin valmiit todistusjärjestelmät ja tietokoneohjelmat ovat olleet riittämättömiä, mutta ihminen on kyennyt ne luovuudella ratkaisemaan. Jos tällaisen ratkeamattoman väitteen olemassaoloa ei kyetä osoittamaan, kuulostaa tämäkin vaihtoehto kovin mystiseltä. Jos taas ratkeamaton väite kohdattaisiin, olisiko mahdollista perustella, että ihminen ei voi tietää sitä todeksi?

Toisena vaihtoehtona on mahdollista väittää ihmisten olevan epäkonsistentteja. Ihmisen luonteeseen kuuluu perustavalla tapaa virheiden tekeminen, eikä ihmistä simuloivan koneenkaan tarvitse olla ristiriidaton. Gödelin epätäydellisyyslausetta ei silloin voida soveltaa. Toisaalta tällöinkin virheellisyyden luonne tulisi määrittää tarkemmin. Onko kyse satunnaisista, korjattavissa olevista virheistä, vaiko syvemmistä ennakko-oletuksistamme ja tavastamme jäsentää maailmaa? Emme kuitenkaan hyväksy mielivaltaisia väitteitä totuuksina, minkä voisimme tehdä, jos tietoisesti uskoisimme ristiriitaan.

Kolmannen vaihtoehdon tarjoaa näkemys, että ihmistä voidaan simuloida tietokoneohjelmalla, mutta tuota ohjelmaa emme voi koskaan joko löytää sen monimutkaisuuden takia, tai tunnistaa introspektiolla sitä omaksi simulaatioksemme. Vähintään kummallisia ajatuksia nämäkin. Simulaation olemassaolo halutaan samaan aikaan olettaa, ja samalla kuitenkin piilottaa se mielen tavoittamattomiin.

Jokainen vaihtoehdoista on epätyydyttävä. Henkilökohtaiselta vakaumukseltani olen platonisti, ja sen myötä olen taipuvainen uskomaan gödeliläisen argumentin lopputulokseen. Jos haluaisin uskoa ihmisen olevan simuloitavissa tietokoneella, pitäisin näkemystä ihmisen epäkonsistenttiudesta, eli erheellisyydestä, lupaavimpana vaihtoehtona. Siinä on intuitiivista vetovoimaa, ja se romuttaisi gödeliläisen argumentin alkutekijöihinsä.

Voimme myös Shapiron tapaan myöntää, että ihmisen matemaattista kompetenssia ja simulointia ei olla määritelty tarpeeksi eksaktiksi, jotta Gödelin epätäydellisyyslausetta voitaisiin soveltaa. Tällöin emme ole sitoutuneita mihinkään seurauksiin.

Kriitikot haluavat gödeliläisen argumentin kannattajien määrittelevän ihmisen kyvyt matemaattisella tarkkuudella, kun taas gödeliläisen argumentin puolestapuhujat puhuvat ihmisen kyvyistä heuristisesti, kenties viitaten implisiittisesti intuitioonsa matemaattisesta ongelmanratkaisusta. Tietokoneohjelmat ovat eksakteja ja selkeästi määriteltäviä, kun taas omaa ajatteluamme emme voi yhtä tarkasti ilmaista. Kenties kyseessä todella on periaatteellinen ero ihmisen ja koneen välillä, tai emme vain kykene lukemaan omaa koodiamme, mutta gödeliläinen argumentti ei onnistu matemaattisella varmuudella kääntämään kenenkään päätä. Jos jo lähtökohtaisesti on valmis käsittelemään koneita ja ihmisiä eri säännöillä, ei argumenttia välttämättä tarvitse enää jatkaa sen pidemmälle.

Gödelin epätäydellisyyslause on sinänsä vahva peruste laskennallista mielenteoriaa vastaan. On kuitenkin epäselvää, tuovatko Lucasin ja Penrosen sille lisää todistusvoimaa. On vaikea kuvitella, että ihminen, joka ei jo pelkästään Gödelin epätäydellisyyslauseen pohjalta kieltäisi mahdollisuutta ihmismielen simuloitavuudeta, muuttaisi mieltään Lucasin ja Penrosen kirjoitusten myötä. Tekninen saivartelu pikemminkin hautaa alleen Gödelin johtopäätöksen intuitiivisen vetovoiman.

Mutta kenties voimme suhtautua asiaan heuristisemmin. Näyttäväthän Gödel, Penrose ja Lucas ajattelevan ihmisen matemaattisen kyvyn sisältävän tavan ajatella ”laatikon ulkopuolelta”. Tuntuu luontevalta sanoa, että matemaattisina ongelmanratkaisijoina emme vain seuraa sokeasti algoritmeja, vaan käytämme myös paljon spontaania luovuutta ja nokkeluutta keksiäksemme miellyttävän ratkaisun. Emme mitenkään systemaattisesti käy läpi vääriä tai epäoleellisia ratkaisuehdotuksia päätyäksemme lopulta oikeaan. Eikä matematiikassa ole mitään yhtä oikeaa tapaa päästä haluttuun lopputulokseen. Ajatus ei ole sen hullumpi kuin se, että näkisimme

matematiikan aksioomat aidosti tosiksi, emmekä pelkästään keinotekoisesti keksityiksi postulaateiksi. Tällöin voimme uskoa siihen, että todistusjärjestelmän aksioomat eivät johda ristiriitaan ja siten tietää Gödelin lauseen todeksi. Tuolloin väite järjestelmän konsistenttiudesta ei tule järjestelmän sisältä, vaan ”näemme sen sielumme silmin”.

Kenties siis on Gödelin epätäydellisyyslausetta parempia tapoja erottaa kone ja ihminen matemaattisina ongelmanratkaisijoina. Matemaattisen järkeilyn mekanisointi voidaan jakaa kolmeen osaan: todistuksen tarkastukseen (eng. *Proof checking*), todistuksen löytämiseen (engl. *Proof discovery*) ja hypoteesin tuottaminen (engl. *Conjecture generation*). Annetun todistuksen tarkistaminen premisseistä on tietokoneelle varsin helppoa. Toinen tehtävä, todistuksen löytäminen taas on algoritmisesti ratkeamaton. Ei ole olemassa sellaista algoritmia, joka yleisesti keksisi todistuksen jokaiselle väitteelle (puolesta tai vastaan) ensimmäisen kertaluvun logiikassa. Jos uskomme ihmisen pystyvän tähän, on mekanisti päihitetty, mutta meidän ei nyt tarvitse pitää ihmistä potentiaalisesti kaikkitietävänä loogikkona. Kolmas tehtävä on taas kaikkein vaikein. Emme vain mielivaltaisesti tai johdonmukaisesti keksi, mitkä matematiikan väitteet ovat ylipäänsä mielenkiintoisia ja tutkimisen arvoisia. Käytännössä mielekkäiden hypoteesien tuottaminen on osoittanut suureksi kompastuskiveksi tekoälyn kehityksessä. Matemaattiseen ongelmanratkaisuun kuuluvaa luovuutta ja spontaania nokkeluutta onkin ollut hyvin hankala määritellä algoritmille. (Arkoudas ja Bringsjord 2014, 36–39)

Matematiikassa vaikuttaa olevan ero siinä, että tiedämme jonkin asian todeksi, ja sillä, että voimme todistaa sen. On hyvin intuitiivista ajatella, että alkulukuja on äärettömästi, mutta asian osoittaminen vaatiikin erikseen pienen tuumailun. Mutta tapamme puhua tietokoneohjelmista ei sisällä tällaista erottelua. Puhuessamme tietokoneohjelman tuottamasta ”tiedosta” viittaamme ainoastaan sen laskutoimitusten tuloksiin.

Ihmisten välillä vallitsee toisen mielen ongelma. Emme voi milloinkaan tietää varmaksi, kokeeko toinen henkilö värit ja tuntee samalla tavoin minä. Käsitystä tietoisuudestamme ei siis voi pelkistää kuvaukseksi aivojemme tilasta. Tekoäly ja tietokoneohjelmat taas ovat täysin läpinäkyviä – ne eivät sisällä mitään mitä kuka tahansa asiaan perehtynyt ei voisi niiden tilasta ja koodista lukea. Ja juuri tähän myös koneiden luotettavuus perustuu. Koska Turingin koneet ja tietokoneohjelmat ovat teoreettisia matemaattisia objekteja, olisi outoa, jos ne kykenisivät tuottamaan kvaliaa. Jos simuloin mielessäni Turingin koneen toimintaa, voisiko silloin mieleni sisälle syntyä toinen mieli?

Turingin koneella on äärellinen määrä tiloja, mutta ihmismieli vaikuttaisi voivan käsittää äärettömyyttä. Jälkimmäistä eri mieltä ovat lähinnä intuitionistit. Vaikuttaa siis siltä, että jos aivot samaistettaisiin tietokoneeseen, tuolloin mahdollisia aivojen tiloja olisi aina ”vähemmän” kuin mahdollisia ajatuksia. Tällöin kaikilla ajatuksilla voisi olla vastaava aivotila, mutta aivotilan perusteella ei voitaisi yksiselitteisesti määrittää mikä ajatus minulla on päässäni. Johtopäätös on tällöin dualismi. Kuviota voi verrata siihen, kuinka yhdellä luonnollisen kielen lauseella voi olla useampi merkityssisältö. Kuten lauseesta ”Talolla on kissa”, ei voida päätellä, onko kissa talon katolla vai vieressä. On varmasti omituista satuilua sanoa, että kaksi eri ajatusta eivät tuottaisi eroa aivotilamme ja fyysisen toimintamme suhteen. Mutta usein ajattelemallamme asialla ei ole mitään näkyvää vaikutusta käytökseemme. Vaikkapa kävelylenkillä käydessämme on vaikea huomata eroa toiminnassamme, mikäli ajattelen Immanuel Kantin Puhtaan järjen kritiikkiä, taikka Napoleonin sotahistoriaa. Kenties lenkkipolulla kohtaan vanhan tuttavani ja jään suustani kiinni. Hän kysyy mitä ajattelen, ja minä vastaan ”Kissa on katolla”. Melko varmasti neurokirurgi pystyisi aivoistani löytämään selittävän eroavaisuuden kahden eri merkityssisällön väliltä, mutta se ei vaikuta mitenkään välttämättömältä. Kenties homomorfiset ajatukset voivat olla keskenään hyvin samankaltaisia sisällöltään. Voimme puhua tunteistamme toiselle henkilölle, joka kykenee samaistumaan niihin, mutta objektiivisesti, luonnontieteellisesti tarkasteluna puheen sisältö katoaa merkkijonoksi. Tunteita ja kokemuksia on mahdotonta tyhjentävistä pelkistää objektiiviseen käsitteistöön.

Nykyään tekoäly on pitkälle kehittynyttä ja kohtaamme sitä kaikilla elämän osa-alueilla. Oli kyse sitten ruokaostokset kaupasta tuovista roboteista tai verkosta kuvia muokkaavana ”taiteilijana”. ChatGPT:n kanssa keskustelemalla sen voi laittaa jopa tuottamaan käyttökelpoista ohjelmointikoodia, joskin tämä vaatii, että ohjelmoija itse tarkistaa lopputuloksen. Lopulta ratkaisevat ongelmat eivät kuitenkaan ole kadonneet minnekään. Tietokoneet operoivat ainoastaan kielen syntaksin tasolla, mutta ymmärrystä semantiikasta tai ulkoisesta todellisuudesta niillä ei ole. Taskulaskin on oiva apuneuvo vaikeassa laskutoimituksessa, mutta tekstinkäsittelyohjelmaan syöttäessä ”4+2=” kone ei osaa supistaa summaa luvuksi 6. Lopulta vain ihminen itse ymmärtää mitä työkalua tai ohjelmaa soveltaa mihinkin tilanteeseen. Mikään kone ei voi esittää oikeaa ratkaisumallia yleisesti kaikkiin ongelmiin, sellainen kun ratkaisi myös Turingin koneen pysähtymisongelman. Toisaalta me kuolevaiset ihmisetkin olemme aika kaukana täydellisistä matemaattisista ongelmanratkaisijoista, vaikka hyväksyisimme platonismin. Gödelin

epätäydellisyyslause asettaa vahvan tekoälyn kannattajille tiettyjä rajoitteita, mutta ei ole uskottavaa, että he eivät voisi elää niiden kanssa.

Voimme kuitenkin luottaa siihen, ettemme koskaan tule tapaamaan sellaista robottikopiota itsestämme, joka tuottaisi kaiken saman matemaattisen tiedon kuin mihin kykenemme, ja jonka voisimme tunnistaa omaksi simulaatioksemme viittaamalla ainoastaan robotin koodiin ja itsetutkiskeluun. Tämän tieteisfantasian kumoaminen ei pitäisi suuresti ketään hetkauttaa.

Lähteet

Arkoudas, Konstantine & Bringsjord, Selmer (2014), ”Philosophical foundations”, teoksessa Frankish, Keith & Ramsey, William M. (toim.), *The Cambridge Handbook of Artificial Intelligence*

Benacerraf, Paul (1967), ”God, the Devil, and Gödel” *Monist* 51, 9–32

Boyer, D (1983), ”J.R. Lucas, Kurt Godel, and Fred Astaire” *Philosophical Quarterly* 33, 147–59

Britannica, The Editors of Encyclopaedia (2024) "Roger Penrose", teoksessa *Encyclopedia Britannica*, https://www.britannica.com/biography/Roger-Penrose viitattu 13.6.2024.

Chalmers, David J. (1995) ”Minds, Machines, and Mathematics” *Psyche* 2, 11–20

Coder, David (1969), ”Gödel’s Theorem and Mechanism” *Philosophy* 44 , 234–237

De Mol, Liesbeth (2018), "Turing Machines", teoksessa Zalta, Edward N. (toim.), *The Stanford Encyclopedia of Philosophy (Winter 2021 Edition)* https://plato.stanford.edu/archives/win2021/entries/turing-machine/ Viitattu 19.10.2023.

Dennett, Daniel (1972), ”Review of the Freedom of the Will” *The Journal of Philosophy* 69, 527–31

Feferman, Solomon (1962), ”Transfinite recursive progression of axiomatic theories” *Journal of Symbolic Logic* 27, 259–316

Gödel, Kurt (1995), *Collected Works. III: Unpublished essays and lectures*. S. Feferman, J. Dawson, S. Kleene, G. Moore, R. Solovay, ja J. van Heijenoort (toim.), Oxford: Oxford University Press.

Lewis, David (1969), ”Lucas against Mechanism,” *Philosophy* 44(169), 231–233

Lucas, John (1961), “Minds, Machines, and Gödel,” *Philosophy*, 36(137), 112–137

Lucas, John. R. (1970) ”Mechanism: A Rejoinder” *Philosophy* 45, 149–51

Lucas, John. R. (1996a), ”The Godelian Argument: Turn Over the Page” luento Oxfordissa

Lucas, John. R. (1996b), ”Minds, Machines and Gödel: A Retrospect” teoksessa Millican, P.J.R. & Clark, A, (toim.), *Machines and Thought* 103–124

Megill, Jason (2024), ”The Lucas-Penrose Argument about Gödel’s Theorem” teoksessa Fieser, James & Dowden, Bradley (toim.), *The Internet Encyclopedia of Philosophy,* ISSN 2161-0002 https://iep.utm.edu/lp-argue/ Viitattu 26.3.2024.

Penrose, Roger (1994), *Shadows of the Mind: A Search for the Missing Science of Consciousness* Oxford: Oxford University Press

Penrose, Roger (1996) ”Beyond the Doubting of a Shadow” *Psyche* 2(23)

Putnam, Hilary (1965) ”Craig’s Theorem” *The Journal of Philosophy*, 62(10) 251–260

Raatikainen, Panu (2020), "Gödel’s Incompleteness Theorems", teoksessa Zalta, Edward N. (toim.), *The Stanford Encyclopedia of Philosophy (Spring 2022 Edition)* https://plato.stanford.edu/entries/goedel-incompleteness/#GdeArgAgaMec Viitattu 13.10.2023.

Searle, John (1980), ”Minds, Brains and Programs” *Behavioral and Brain Sciences* 3(3), 417–457

Shapiro, Stewart (1998), “Incompleteness, Mechanism, and Optimism,” *Bulletin of Symbolic Logic*, 4, 273–302

Webb, Judson (1980) *Mechanism, Mentalism and Metamathematics: An Essay on Finitism*, Dordrecht: Springer Science

Whiteley, C (1962), ”Minds, Machines and Gödel: A Reply to Mr. Lucas,” *Philosophy* 37, 61–62