

**Aku Pasanen**

# **Tekoäly Texas Hold'em pokerissa**

Tietotekniikan kandidaatintutkielma

29. huhtikuuta 2022

Jyväskylän yliopisto

Informaatioteknologian tiedekunta

**Tekijä:** Aku Pasanen

**Yhteystiedot:** aku.s.a.pasanen@student.jyu.fi

**Ohjaaja:** Tuomo Rossi

**Työn nimi:** Tekoäly Texas Hold'em pokerissa

**Title in English:** AI in Texas Hold'em poker

**Työ:** Kandidaatintutkielma

**Opintosuunta:** Tietotekniikka

**Sivumäärä:** 27+1

**Tiivistelmä:** Tämä tutkielma kartoittaa millaisia menetelmiä ja tekniikoita soveltaen Texas Hold'em pokeria pelaavia tekoälyohjelmistoja kehitetään. Sekä sen millaisella tasolla kyseiset ohjelmistot suoriutuvat suhteessa parhaimpiin ihmispelaajiin.

**Avainsanat:** Tekoäly, AI, Epätäydellisen informaation pelit, Pokeri, Texas Hold'em, Peliteoria, Kandidaatintutkielmat

**Abstract:** This thesis explores the methods and techniques used to develop artificial intelligence programs for Texas Hold'em poker. As well as the level at which those programs performs relative to the best human players.

**Keywords:** Artificial Intelligence, AI, Imperfect information games, Poker, Texas Hold'em, Game Theory, Bachelor's Theses

## **Kuviot**

Kuvio 1. Abstraktion mallinnus .....	9
Kuvio 2. Julkinen pelipuu .....	16

## Sisällys

1	JOHDANTO .....	1
2	TEKOÄLYTUTKIMUS JA PELIT .....	2
3	TEXAS HOLD'EM.....	5
3.1	Limit .....	5
3.2	No Limit .....	6
3.3	Pelaajamäärä.....	6
3.4	Pelin kulku ja säännöt .....	6
4	TEKOÄLY TEXAS HOLD'EMISSA .....	8
4.1	Peliteoria ja Nashin tasapaino .....	8
4.2	Abstraktointi .....	9
4.3	Counterfactual regret minimization .....	11
5	ANSIOITUNEIMMAT TEXAS HOLD'EM TEKOÄLYOHJELMISTOT .....	13
5.1	Heads Up Limit Texas Hold'em ratkaistu.....	13
5.2	Libratus .....	14
5.3	DeepStack .....	15
5.4	Pluribus .....	17
6	YHTEENVETO.....	21
	LÄHTEET .....	22
	LIITTEET.....	24

# 1 Johdanto

Tämän tutkielman tarkoituksena on selvittää millaisia tekniikoita käyttäen Texas Hold'em pokerin kaltaisia suuria epätäydellisen informaation pelejä pelaavia tekoälyohjelmistoja pystytään kehittämään. Sekä millä tasolla kyseiset tekoälyohjelmistot suoriutuvat tällä hetkellä suhteessa pelin parhaimpiin ihmispelaajiin. Pokeri on pelinä mielenkiintoinen, sillä se kuvastaa hyvin todellisen maailman ongelmia, sen sisältäessä pelaajilta kätkeytyä informaatiota. Tekoälyn menestyksekkäs soveltaminen pokerin kaltaisiin peleihin on askel kohti tekoälyn sovellutuksia tosi elämän ongelmien ratkaisussa. Tutkielma on toteutettu tutustumalla tunnetuimpien tutkimusryhmien julkaisuihin aiheesta. Tutkimuksen alussa käymme läpi lyhyesti tekoälyn historiaa, sekä tekoälytutkimuksen ja pelin välistä suhdetta. Seuraavissa luvuissa paneudumme Texas Hold'em pokeriin ja sen sääntöihin, peliteorian konsepteihin pelien ratkaisun taustalla, ja ratkaisut mahdollistaviin teknologioihin. Viidennessä luvussa käymme läpi alalla tunnetuimpia ja menestyneimpiä tekoälyohjelmistoja eri tutkimusryhmiltä.

## 2 Tekoälytutkimus ja pelit

Yksi maailman tunnetuimmista ja vanhin englanninkielinen tietosanakirja Britannica määrittelee termin tekoäly olevan tietokoneen tai tietokoneen ohjaaman robotin kyky toteuttaa tehtäviä, jotka yleisesti yhdistetään älyllisiin toimijoihin. Termiä käytetään usein järjestelmien kehityksessä, jotka jäljittelevät ihmisille tyypillisiä älyllisiä toimenpiteitä, kuten kykyä järkeillä, ymmärtää tarkoitusta, yleistää ja oppia menneistä kokemukista ("Artificial Intelligence" 2022).

Tekoälyn historia kantautuu aina 1900-luvun alkupuolelle asti, jolloin tieteisfiktiokirjallisuudessa ja -elokuvissa nousi pinnalle ajatus älyllisistä koneista ja roboteista. Ensimmäisiä tämän kaltaisia teoksia on L. Frank Baum'n vuonna 1900 kirjoittama klassikkoteos Ihmemaan Oz (The Wonderful Wizard of Oz), jossa esiintyy sydämetön robotti Tinamies ("The History of Artificial Intelligence" 2017).

1950-luvulla ansioitunut englantilainen matemaatikko Alan Turing julkaisi artikkelinsa "Computing Machinery and Intelligence", jossa tämä kuvasi kuinka luoda älyllisiä koneita, ja etenkin sitä kuinka testata näiden älyllisyyttä. Tätä Turingin testiksi nimitettyä testiä pidetään edelleen tärkeimpänä suorituskykytestinä testatessa tekoälysovellusten älyllisyyttä: Mikäli ihminen on vuorovaikutuksessa toisen ihmisen ja koneen kanssa, eikä ihminen pysty erottamaan konetta toisesta ihmisestä, on kone älyllinen.

Englanninkielinen termi "Artificial Intelligence" eli "tekoäly", esitettiin ensimmäisen kerran Marvin Minskyn ja John McCarthyn järjestämässä, noin kahdeksan viikkoa kestäneessä seminaarissa "Dartmouth Summer Research Project on Artificial Intelligence". Tapahtuman tarkoituksena oli saattaa yhteen eri alojen tutkijoita ja perustaa uusi tutkimusalue jonka määränä on kehittää koneita, jotka pystyvät simuloimaan ihmisten älykkyyttä (Haenlein ja Kaplan 2019).

Erilaisilla peleillä on pitkä historia tekoälytutkimuksen suorituskykytesteinä. Jo ennen tekoälykäsitteen syntyä ensimmäiset tekoälystä kiinnostuneet tutkijat kuten yhdysvaltalainen matemaatikko Claude Shannon, pyrkivät kehittämään shakkia pelaavan tietokoneohjelmiston. Shannonin vuonna 1950 julkaisemassaan artikkelissa "Programming a Computer for Playing

Chess” Shannon kuvailee shakkiohjelmiston olevan ihanteellinen lähtökohta tekoälykehitykselle, sillä shakin kaltaisessa pelissä ongelma on tarkoin rajattu, niin sallittujen siirtojen kuin lopullisen päämäärän suhteen, eikä se ole ongelmana liian helppo ollakseen triviaali, tai liian haastava tyydyttävään ratkaisuun. Shakin taitavan pelaamisen oletetaan yleensä myös vaativan ajattelua, joten sen ratkaiseminen pakottaisi meidät myöntämään mahdollisuuden koneelliselle ajattelulle tai rajaamaan ajattelun määritettä tarkemmin. Shakin rakenne sopii hyvin nykyisten tietokoneiden digitaaliseen luonteeseen (Shannon 1950)

Tekoälykehityksen yksi suurimmista haasteista seuraavien vuosikymmenien aikana olikin päihittää shakin hallitseva maailmanmestari normaalipituudessa otelussa. Vasta vuonna 1977 tietokoneohjelmisto voitti ensimmäisen kerran shakin suurmestarin arvonimen ansainneen pelaajan shakin blitz-pelimuodossa, jossa kullakin pelaajalla on käytössään vain 5 minuuttia aikaa päätöstensä pohtimiseen koko pelin aikana. Tietokoneiden etu ihmisiä vastaan tuli-kin esiin juuri pikashakissa, sillä jopa maailman parhaat ihmispelaajat ajautuvat tekemään virheitä nopeissa otteluissa. Tästä voitosta kesti vielä 11 vuotta, ennen kuin tekoäly pystyi päihittämään suurmestarin pelin normaalipituudessa otelussa (Hsu 1999). IBM:n kehittämä shakkitietokone Deep Blue onnistui ensimmäisenä maailmassa shakkitietokoneiden suurimmassa haasteessa, voittaessaan shakin hallitsevan maailmanmestarin, venäläisen suurmestarin Garri Kasparovin, vuonna 1997 käydyssä historiallisessa uusintaottelussa, sen hävittyä kaksikon ensimmäisen ottelun vain vuotta aikaisemmin (“IBM” 2022).

Luontaisena jatkumona tekoälytutkimus siirsi katseensa suurempiin haasteisiin pelien saralla, shakkia huomattavasti monimutkaisempaan, yli 3000 vuotta sitten Kiinassa kehittyneeseen lautapeliin nimeltä Go. Britannialaisen tekoälytutkimukseen erikoistuneen yrityksen, DeepMind Technologiesin artikkelin mukaan Go on tunnettu haastavimpana klassisena pelinä tekoälytutkimuksen parissa sen monimutkaisuuden vuoksi. Peli sisältää  $10^{170}$  erilaista pelitilannetta (pelinappuloiden sijainti laudalla), joka on enemmän kuin atomien määrä koko universumissa. DeepMindin kehittämä Go-tekoälyohjelmisto AlphaGo onnistui ensimmäisenä maailmassa päihittämään ammattilaispelaajan, kolminkertaisen euroopanmestarin, Fan Huin tuloksella 5-0. Vain vuotta myöhemmin vuonna 2016, AlpaGo haastoi maailman parhaana ja pelin legendana pidetyn ihmispelaajan, 18 kertaisen Go:n maailmanmestarin Lee Sedolin, ja päihitti tämän tuloksella 4-1. (“DeepMind” 2022)

Edellämainituissa peleissä shakissa ja Go:ssa on yhteistä se, että kullakin pelaajalla on käytössään kaikki tieto pelin sen hetkisestä tilasta, eli pelinappuloiden sijainnista pelilaudalla. Tämän kaltaisia pelejä nimitetään täydellisen informaation peleiksi. Modernin peliteorian kehittäjä ja tietojenkäsittelyn edelläkävijä, John Von Neumann visioi päättelystä epätäydellisen informaation peleissä seuraavasti: ”Real life is not like that. Real life consists of bluffing, of little tactics of deception, of asking yourself what is the other man going to think I mean to do. And that is what games are about in my theory”. Yksi epätäydellisen informaation peleistä, joka erityisesti kiehtoi Von Neumannia oli pokeri. (Moravčík ym. 2017). Epätäydellisen informaation peleissä jokin osa pelin informaatiosta on piilossa pelaajilta, esimerkiksi vastustajalla voi olla hallussaan salaisia pelikortteja, kuten pokerissa pelaajille jaettavat yksityiset taskukortit. Epätäydellinen informaatio on läsnä kaikissa todellisen maailman strategisissa vuorovaikutuksissa, kuten liiketoimintastrategioissa, neuvotteluissa, hinnoittelussa, taloudessa, kyberturvallisuudessa ja sotilaallisissa toimissa. Tämän takia epätäydellisen informaation omaavien pelien tutkiminen tekoälytutkimuksessa on erityisen tärkeää. Kätkeyty informaatio tekee peleistä todella paljon monimutkaisempia lukuisista eri syistä johtuen. Sen sijaan että etsisimme optimaalisia järjestyksiä toteuttaa pelitoimintoja, epätäydellisen informaation pelien tekoälyohjelmistojen tulee pystyä tasapainottamaan toimintonsa asianmukaisesti, siten ettei vastustaja saa selville liikaa tekoälyn hallitsemasta yksityisestä informaatiosta. Esimerkiksi bluffaaminen on oleellinen osa jokaista kilpailullista pokeristrategiaa, mutta bluffaaminen jokaisessa tilanteessa olisi huono strategia. (Brown ja Sandholm 2018) Seuraavassa kappaleessa käsittelemme lisää tutkimuksen keskeistä aihetta, pokeria ja sen suosituinta pelimuotoa Texas Hold’emia, joka on muodostunut shakin ja Go:n tapaan tekoälykehityksen keskeiseksi haasteeksi pelin ominaisuuksien vuoksi.



### 3 Texas Hold'em

Pokerilla on pitkä historia tieteellisen tutkimuksen aiheena, sitä ovat tutkineet niin matemaatikot ja taloustieteilijät kehittäessään peliteorian konsepteja, ja tekoälytutkijat kehittäessään ohjelmistoja jotka pyrkivät ratkaisemaan haastavia ongelmia epätäydellisen informaation aiheuttaessa epävarmuutta päätöksenteossa. Kaikki pokerin pelimuodot ovat niin sanottuja nol-lasummapelejä, eli yhden pelaajan voitot ovat suoraan pois toisilta pelaajilta. Pelin tavoite onkin siis päihittää toiset pelaajat ja kerätä heidän rahat itselleen. Moni saattaa virheellisesti ajatella pokerin olevan vain uhkapeli sen sisältäessä satunnaisuutta jaettujen korttien muodossa, mutta koska peliä pelataan toisia pelaajia vastaan ja se sisältää strategisia päätöksiä, johtaa se tilanteeseen jossa parempi pelaaja jää pitkällä aikavälillä aina voitolle huonompia pelaajia vastaan pelatessaan.

Texas Hold'em on maailman suosituin pokerin pelimuoto, niin netissä, kuin kasinoilla ja pelisaleissa ympäri maailmaa. Vuosittaisissa Yhdysvaltojen Las Vegasissa järjestettävissä pokerin maailmanmestaruuskilpailuissa, World Series Of Pokerissa (lyhenne WSOP), maailmanmestaruus ratkaistaan juurikin 10 000 dollarin sisäänoston No Limit Texas Hold'em pääturnauksessa, jonka voittaja haalii itselleen miljoonien dollareiden lisäksi arvostetun kultaisen WSOP-rannekkeen. WSOP:n pääturnaus vetää puoleensa vuosittain tuhansia osanottajia; ennen koronaviruspandemian alkua vuonna 2019 pääturnauksessa oli huimat 8569 osallistujaa, ja turnauksen voittajalle kertyi palkinnoksi 10 miljoonaa dollaria ("WSOP" 2019). Epätäydellisen informaation lisäksi Texas Hold'emista haastavan pelin niin ihmisille kuin tekoälyllekin tekee pelin suuri koko. Pelin "No Limit" variantissa on yli  $10^{160}$  eri päätöskohtaa (Johanson 2013), eli sen voidaan nähdä olevan lähes yhtä monimutkainen ja laajahaarainen peli, kuin aiemmassa luvussa käsitelty klassinen peli Go.

#### 3.1 Limit

Termi "Limit", tarkoittaa, että pelissä käytetyille panostuksille ja korotuksille on ennalta määrätty koko. Panostuksien rajaamisesta johtuen Texas Hold'em:n Limit pelimuoto on kooltaan huomattavasti sen No Limit pelimuotoa pienempi, sisältäen vain noin  $10^{14}$  päätöskohtaa.

## 3.2 No Limit

Termi "No Limit" tarkoittaa sitä, että pelaaja voi vuorollaan panostaa haluamansa määrän rahaa tai pelimerkkejä, eli tämä voi halutessaan korottaa vaikka kaikki pelimerkkinsä, tässä tapauksessa pelaajan sanotaan menevän "All in". No Limit on Texas Hold'em:n suosituin pelimuoto.

## 3.3 Pelaajamäärä

Texas Hold'emia pelataan niin käteispeleinä kuin turnauksina. Netissä pelattavista käteispeleistä suurin osa pelataan kuusikätsenä eli kuuden pelaajan kesken, mutta myös kasinoilla suosittuja "täysiä" 9 pelaajan pelipöytiä on tarjolla. Turnauksissa voi olla jopa satoja tai tuhansia osallistujia. Turnausten alussa pelaajat jaetaan yleensä 6-9 pelaajan pöytiin ja pelaajien pudotessa pois turnauksesta hävittyään kaikki pelimerkkinsä, pelipöytiä yhdistetään niin että turnauksen säännöissä määrätty pöytäkohtainen pelaajamäärä säilyy ennallaan. Turnausten viimeistä jäljellä olevaa pöytää kutsutaan finaali-pöydäksi. Puhuttaessa pokerin pelaajamäärästä, puhutaan siis pelin "kätisyydestä"; kuuden pelaajan peli on kuusikätsistä ja niin edelleen, poikkeuksena kaksin pelattava pelimuoto, jota pokeritermein kutsutaan "heads up":iksi.

## 3.4 Pelin kulku ja säännöt

Pelikierroksen alussa kaksi pelaajaa kuvainnollisen jakajannapin vasemmalla puolella myötapäivään kiertäessä, asettavat pakolliset panokset, eli niin sanotut sokkohanokset. Sokkohanoksia on kahden kokoisia, iso ja pieni. Pieni on kooltaan puolet isosta sokkohanoksesta, esimerkiksi 1€/2€. No Limit Texas Hold'em käteispelissä pieni sokkohanos on 1€ ja iso sokkohanos 2€. Sokkohanosten asettamisen jälkeen kullekin pelaajalle jaetaan kaksi kuvapuoli alaspäin olevaa korttia, jotka vain he itse saavat katsoa. Näitä kortteja kutsutaan pokeritermillä "taskukortit" (hole cards). Ison sokkohanoksen vasemmalla puolella oleva pelaaja aloittaa ensimmäisenä panostuskierroksen. Hän voi vuorollaan joko luovuttaa kätensä, eli foldata, maksaa ison sokkohanoksen tai korottaa. Kaikilla pelaajilla on vuorollaan käytössä samat vaihtoehdot: foldata, maksaa aiemmin tehty korotus tai korottaa aiemman korotuksen

päälle. Usein taidokkaat ihmispelaajat välttävät ison sokkopanoksen maksamista ja päätyvät korottamaan avatakseen pelin, sillä näin he voivat voittaa käden suoraan, mikäli muut pelaajat luovuttavat kätensä. Mikäli kaksi tai useampi pelaaja on päätenyt osallistumaan pelikierrokselle ja ovat maksaneet tarvittavat panostukset, jakaja jakaa pöydälle kuvapuoli ylöspäin kolme korttia, joita kutsutaan flopiksi, flopin kortit ovat kaikille pelaajille yhteisiä. Flopilla tapahtuu seuraava panostuskierrös, jolloin pelaajat vuorollaan voivat joko sököttää eli jättää panostamatta, panostaa tai korottaa aiemmin tehdyn panostuksen päälle. Mikäli kaksi tai useampi pelaaja päättää jatkaa peliä ja maksaa tarvittavat panostukset, jakaja jakaa pöydälle yhden yhteisen kortin, jota kutsutaan turniksi. Turnin panostuskierrös noudattaa samaa kaavaa kuin flopilla. Pelin jatkuessa turnin jälkeen jakaja jakaa pöydälle viidennen ja viimeisen yhteisen kortin eli riverin, jonka jälkeen seuraa vielä viimeinen panostuskierrös samoilla säännöillä. Mikäli kaksi tai useampi pelaaja maksaa tarvittavat panostukset, näyttävät pelaajat taskukorttinsa. Käden voittaa pelaaja, joka on onnistunut muodostamaan parhaan viiden kortin pokerikäden käyttäen kahta yksityistä taskukorttiaan ja viittä pöydällä olevista yhteisistä korteista. Texas Hold'emissa käden voi siis voittaa näyttämällä parhaan käden riverillä, tai panostamalla vastustajansa luovuttamaan kätensä millä tahansa panostuskierröksellä.

## 4 Tekoäly Texas Hold'emissa

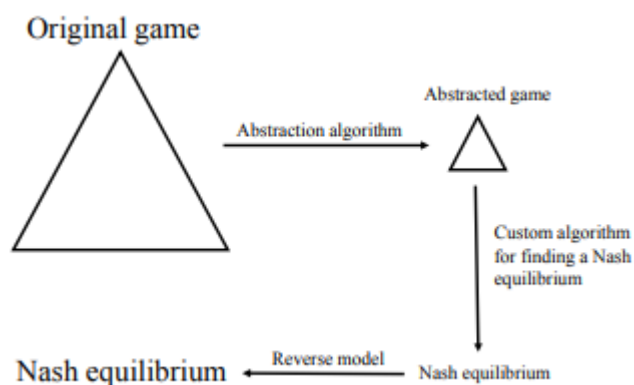
Tutustuttaemme Texas Hold'emiin edellisessä luvussa, voimme nyt siirtyä käsittelemään tutkimuksen pääaihetta, tekoälyä Texas Hold'emissa. Tässä kappaleessa käsittelem tärkeimpiä menetelmiä ja konsepteja Texas Hold'emia pelaavien tekoälyohjelmistojen kehityksessä.

### 4.1 Peliteoria ja Nashin tasapaino

Kaikissa aiemmin käsitellyissä esimerkeissä ja mainittujen tekoälyohjelmistojen saavutuksissa eri pelien parissa, yhteistä on niiden strateginen menettely. Lähes jokainen peleihin erikoistunut tekoälyohjelmisto pyrkii ratkaisemaan pelin ja rakentamaan strategiansa peliteorian menetelmiä apunaan käyttäen. Sen sijaan että etsisimme heikkouksia vastustajamme strategiasta ja käyttäisimme näitä strategisia heikkouksia hyväksemme, tekoälyohjelmistot pyrkivät arvioimaan ja ratkaisemaan pelissä vallitsevan Nashin tasapainoa mukailevan strategian. Kehittäjänsä John Nashin mukaan nimetyllä Nashin tasapainolla tarkoitetaan strategisen päättelyn tilaa, jossa kummankaan pelaajan ei ole mahdollista muuttaa strategiaansa saavuttaakseen itselleen parempaa lopputulosta. Nashin tasapainotilan on todistettu olevan olemassa kaikissa rajallisissa peleissä, joskin tasapainotilan löytäminen voi osoittautua haastavaksi. Heads up Texas Hold'em in kaltaisessa, kahden pelaajan välisessä nollasummapelissä, pelaaja joka käyttää Nashin tasapainon mukaista strategiaa, ei tule häviämään odostuarvossa, riippumatta siitä mitä strategiaa vastustaja käyttää. Toisinsanoen Nashin tasapainon mukaisista strategiaa on mahdoton päihittää; molempien pelaajien pelatessa Nashin tasapainon mukaista strategiaa, päättyy peli pitkällä aikavälillä tasatulokseen. Kuitenkin Texas Hold'em in kaltaisissa monimutkaisissa nollasummapeleissä tasapaino-strategian toteuttaminen on erityisen hankalaa. Vastustajan tehdessä virheitä, eli poiketessaan tasapainotilasta, Nashin tasapainotilan mukaista strategiaa pelaava pelaaja voittaa odotusarvossa, Nashin tasapainon määritelmän mukaisesti. (Brown ja Sandholm 2019)

## 4.2 Abstraktointi

Abstraktoinnilla tarkoitetaan menetelmää, jossa iso ongelma, esimerkiksi peli, pilkotaan pienempiin paremmin hahmotettaviin kokonaisuuksiin tai osiin, pyrkien kuitenkin säilyttämään mahdollisimman paljon alkuperäisen pelin strategisista ominaisuuksista. Abstraktointi on muodostunut tärkeäksi menetelmäksi suurten epätäydellisen informaation pelien, kuten Texas Hold'em pokerin, ratkaisuun pyrkivien tekoälyohjelmistojen luomisessa useiden vuosien aikana. Syitä pelin abstraktoimiselle voi olla; Se, että alkuperäinen abstraktoimaton peli on liian suuri ratkaistavaksi Nashin tasapainoon. Pelin ratkaisemiseen käytettävä algoritmi voi vaatia pelin yksinkertaistamista ratkaisua varten, esimerkiksi peleille usein pitää pystyä määrittelemään tarkka tila ja mahdolliset toiminnot. Alkuperäinen peli on liian monimutkainen tai suuri ylöskirjattavaksi yksityiskohtaisesti, joten tarvitsemme abstraktiota mallintaaksemme peliä. (Sandholm 2015)



Kuvio 1. Abstraktion mallinnus (Sandholm 2015)

Pelien abstraktoimisen yleisin muoto on informaation abstraktio (information abstraction), jossa joitain pelin informaatiotiloja niputetaan yhteen (Sandholm 2015). Informaatio abstraktiosta yksinkertaisena esimerkkinä voi olla Texas Hold'emissa erilaisten käsikategorioiden käsitteleminen samana; ennen floppia taskukortit pataässä ja patakuningas (AsKs, s = "spades") ovat samanarvoisia, ja tästä syystä strategisesti identtisiä herttaässä ja herttakuninkaan (AhKh, h = "hearts") kanssa. Tämän kaltaisia aloituskäsiä (taskukortteja) pokerissa usein kutsutaankin vain niiden ominaisuuksien mukaan; "Ace-King suited" eli Ässä-Kuningas sa-

maa maata (AKs, s = "suited"). Myös flopin jälkeen muodostuneita vahvempia pokerikäsiä voidaan kategorisoida ja käsitellä samanarvoisina tilanteesta riippuen.

Toinen yleinen pelien abstraktoimisen muoto on toimintojen abstraktio (action abstraction), jossa jotkin alkuperäisessä pelissä mahdollisista toiminnoista ajatellaan mahdottomiksi, ja ne poistetaan pelaajien käytöstä (Sandholm 2015). Kuten aiemmassa kappaleessa käsitelimme, No Limit pokerissa pelaaja voi vuorollaan panostaa haluamansa määrän pelimerkkejä, joten yleinen esimerkkitapaus toimintojen abstraktiosta No Limit Texas Hold'emissa on käytettävien panosten koon rajaaminen. Panostuksen 100 \$ ja 101 \$ välillä voidaan katsoa olevan niin vähän eroa, että kummankin vaihtoehdon sisällyttäminen abstraktoituun peliin on kannattamatonta. Sen sijaan voisimme sallia pelaajien panostaa 100 \$ -lisäyksin 100 \$ ja 20000 \$ välillä. (Brown ja Sandholm 2018). Tämän kaltainen abstraktio vähentää mahdollisia päätösvaihtoehtoja runsaasti ja pienentää ratkaistavan pelin kokoa merkittävästi.

Abstraktion luominen on pitkään ollut manuaalinen prosessi, mutta viime vuosina siihen on kehitetty lukuisia tehokkaita algoritmeja, jotka pyrkivät optimoimaan abstraktion tason kussakin pelin vaiheessa (Sandholm 2015).

Abstraktioon perustuvat ratkaisut ja strategiat olettavat että myöskään vastustaja ei voi toimia abstraktion ulkopuolella olevien toimintojen mukaisesti, mutta todellisuudessa asia ei ole niin. Kohdatessaan abstraktion ulkopuolisen toiminnon, tulee tekoälyohjelmiston pystyä kartoittamaan tämä toiminto johonkin abstraktoidun pelin toimintoon. Tähän käyttötarkoitukseen kehitettyjä algoritmeja nimitetään toimintojen kartoittamis -algoritmeiksi ("action mapping", "reverse mapping", "action translation"). Ne pyrkivät löytämään abstraktoidusta pelistä parhaimman mahdollisen vastineen oikeassa pelissä kohdatusta toiminnosta. Toimintojen kartoittaminen on yksi isoimmista heikkouksista abstraktioon perustuvissa No Limit Texas Hold'em -tekoälyohjelmistoissa, sillä parhaimmat ihmispelaajat pystyvät paikantamaan heikkouksia tekoälyn strategiasta käyttämällä nokkelia ja vaihtelevia panoskokoja abstraktoidun pelin ulkopuolelta. Abstraktioon perustuva ratkaisu on siis sitä heikompi, mitä enemmän abstraktoitu peli eroaa alkuperäisestä pelistä. (Sandholm 2015)

### 4.3 Counterfactual regret minimization

Abstraktoinnin lisäksi tarvitsemme keinon ratkaista abstraktoidun pelin Nashin tasapainoon. Suurten epätäydellisen informaation pelien ratkaisussa menestyneimpiä algoritmeja ovat olleet niin sanotut Counterfactual regret minimization -sukuiset algoritmit ja niiden eri muunnelmat. Nimensä mukaisesti counterfactual regret minimization -algoritmit pyrkivät minimoimaan ”katumuksen” pelin jokaisen päätöksen kohdalla. (Zinkevich ym. 2007). CFR on iteratiivinen ”self-play” algoritmi, joka konvergoi kohti Nashin tasapainoa kaikissa kahden pelaajan nollasummapeleissä, kuten heads-up pokerissa, pelaamalla itseään vastaan lukuisia pelikierroksia. Sen eri ilmentymät ovat olleet käytössä kaikissa viime vuosien menestyneimmissä Texas Hold'em -tekoälyohjelmistoissa, sekä jokaisessa kilpailukykyisessä ohjelmistossa vuosittaisissa tietokonepokerin kilpailussa (Annual Computer Poker Competition) viimeisen kuuden vuoden aikana. (Brown ym. 2019)

Brown ja Sandholm 2018 kuvaavat käyttämänsä counterfactual regret minimization -algoritmin muunnelman, Monte Carlo CFR:n (”MCCFR”) toimintaa, artikkelissaan ”Superhuman AI for heads-up no-limit poker: Libratus beats top professionals” seuraavasti; Jokaisella simuloitulla pelikierroksella algoritmi valitsee yhden pelaajan, joka tutkii kaikki mahdolliset päätösvaihtoehdot ja päivittää päätösکوhtaiset katumus-arvonsa ja strategiansa. Tätä pelaajaa kutsutaan ”traverseriksi”, eli läpikulkijaksi. Toinen pelaajista pelaa aiemmin määritettyjen katumus-arvojen mukaisella strategialla. Roolit vaihtuvat pelikierrosten välissä, sallien siis kummankin pelaajista päivittää strategiaansa vuorollaan läpikulkijana. Prosessin alussa tekoälyohjelmisto ei ole vielä oppinut mitään, joten se pelaa täysin satunnaisella strategialla. Pelikierroksen päätyttyä ja lopputuloksen selvittyä, läpikulkija laskee jokaiselle päätökselleen katumus-arvon, ja päivittää strategiaansa siten, että se suosii jatkossa suuremmalla todennäköisyydellä päätösvaihtoehtoja, joiden käyttämättä jättäminen aiheutti eniten katumusta edellisellä pelikierroksella. Pelikierros pokerin tapauksessa tarkoittaa yhtä pokerikättä. Kun enemmän ja enemmän pelikierroksia simuloidaan, takaa MCCFR sen että suurella todennäköisyydellä pelaajan keskimääräinen päätösکوhtainen katumus lähestyy nollaa ja Nashin tasapainoa.

Suurten epätäydellisten informaation pelien osalta, tuotetut ratkaisut ovat arvioita Nashin tasapainosta, eli ne poikkeavat täydellisestä ratkaisusta ja todellisesta Nashin tasapainosta.

Tuotetun ratkaisun laatua voidaan arvioida sen ”exploitabilityn” mukaan, eli siis sillä kuinka paljon arvioitu strategia häviää pahimman tapauksen vastustajalleen odotusarvoisesti.



## 5 Ansioituneimmat Texas Hold'em tekoälyohjelmistot

Tässä kappaleessa esittelen alalla tunnetuimpia ja menestyneimpiä tekoälyohjelmistoja eri tutkimusryhmiltä, jotka ovat onnistuneet päihittämään jopa maailman parhaimpiin lukeutuvia ihmispelaajia heidän leipäpelissään.

### 5.1 Heads Up Limit Texas Hold'em ratkaistu

Bowling ym. 2015 kirjoittavat artikkelissaan ”Heads-up limit hold'em poker is solved” kehittämästään Heads up Limit Texas Hold'em -tekoälyohjelmistosta. Tutkimusryhmän mukaan he ovat pystyneet ”heikosti” ratkaisemaan yli  $10^{14}$  päätöskohtaa sisältävän pokerin variantin. Heidän määritelmänsä mukaan peli voidaan julistaa ratkaistuksi, mikäli yli 95% varmuudella tuotettua ratkaisua ei voida erottaa täydellisestä ratkaisusta koko elämän kestävästä pelaamisesta huolimatta. Tämä käytännössä tarkoittaa siis sitä, että pelatessaan pokeria 200 pelikierrosta tunnissa, 12 tuntia päivässä, 70 vuoden ajan; heidän kehittämänsä tekoälyohjelmisto ei tule suurella varmuudella ja riittävällä marginaalilla häviämään pahimman tapauksen vastustajalleen, joka pystyy käyttämään hyväksi (”exploit”) tekoälyohjelmiston strategian heikkouksia, noin 1 milli-ison- sokkopianoksen verran jokaista pelikierrosta kohden (mbb/g, milli-big-blind/game). Tämä johtuu pokerin satunnaisuudesta ja sen tuottamasta keskihajonnasta, joka Limit Hold'em:n tapauksessa on arvioitu olevan noin 5 milli-isoa-sokkopianosta pelikierrosta kohden. Keskihajonnasta johtuen artikkelissa käsitelty tekoälyohjelmisto pystyy jopa voittamaan pahimman tapauksen vastustajansa kerran kahdestakymmenestä simuloidusta 70 vuoden ajanjaksosta.

Ratkaisussaan he hyödynsivät CFR+ nimistä counterfactual regret minimization -algoritmiä ja tehokasta tiedonpakkausta. Kirjoittajien mukaan CFR+ algoritmi vaatii huomattavasti vähemmän laskentatehoa ja -aikaa kuin aiemmat CFR:n variantit, mikä mahdollistaa yhä suurempien epätäydellisten informaation pelien ratkaisun. Ilman tiedonpakkaamista Limit Hold'em:n kokoisen pelin ratkaisussa tarvittavien CFR:n tuottamien katumusarvojen ja strategian tallentaminen edellyttäisi 168 teratavuun verran tallennustilaa. Tehokkaan, jopa 28:1, tiedonpakkauksen ansiosta ratkaisu pystyttiin pakkaamaan yhteensä alle 28 teratavuun tallennustilaa.

## 5.2 Libratus

Yhdysvalloissa sijaitsevan Carnegie Mellonin huippuyliopiston tietojenkäsittelytieteen laitoksen tutkijoiden Noah Brownin ja Tuomas Sandholmin kehittämä tekoälyohjelmisto ”Libratus”, pystyi ensimmäisten joukossa maailmassa päihittämään ryhmän maailman parhaimpia ihmispelaajia heidän leipäpelissään, Heads up No Limit Texas Hold’emissa (Brown ja Sandholm 2018). Libratuksen ja neljän ihmispelaajan; Dong Kimin, Jason Lesin, Jimmy Choun ja Daniel McAulayn välinen kolme viikkoa ja 120 000 pokerikättä kestänyt ottelu miteltiin Rivers Casinolla Pittsburghissa vuonna 2017 järjestetyssä ”Brains vs Artificial Intelligence: Upping the Ante” tapahtumassa (”Carnegie Mellon University” 2017). Tapahtuman nimi on viittaus kaksi vuotta aikaisempaan tapahtumaan, jossa samaisen tutkimusryhmän kehittämä tekoälyohjelmisto ”Claudico” hävisi 80 000 kättä kestäneen haasteottelunsa ihmispelaajia vastaan samaisella Riversin Casinolla vuonna 2015. Claudico hävisi ihmispelaajille 91 mbb/g marginaalilla, joka tapahtumasta uutisoineen pokeriuutissivuston Pokerfusen mukaan on merkittävä tulos (”Pokerfuse” 2015). Libratuksen ja ihmispelaajien ottelu päättyi Libratuksen voittoon 147 mbb/g marginaalilla, jonka tilastollinen merkittävyys on huikat 99,98%.

Libratuksen takana oleva arkkitehtuuri sisältää kolme erillistä moduulia, joilla jokaisella on olennainen rooli sen menestyksessä. Ensimmäistä moduulia kirjoittajat kutsuvat ”blueprint” strategiaksi, eli pohjapiirros-strategiaksi. Libratuksen pohjapiirros-strategia on abstraktioon perustuva, ja ennen pelaamista ratkaistu käyttäen luvussa 4.3 kuvailtua monte carlo counterfactual regret minimization -algoritmiä. Suurin osa abstraktiossa käytetyistä panosko’oista (”bet sizes”) on valittu analysoimalla menestyneimpien tekoälyohjelmistojen käyttämiä panoskokoja menneiden vuosien tietokonepokerin kilpailuista. Jotkin käytetyt panoskoot pelin aikaisilla panostuskierroksilla määriteltiin riippumattoman parametrien optimointialgoritmin avulla, joka konvergoi paikallisesti optimaaliseen panoskokojoukkoon. Ensimmäisellä kahdella panostuskierroksella, ennen floppia ja flopilla, Libratuksen abstraktiossa ei käytetä korttien abstraktiota. Kolmannella panostuskierroksella (turnilla) 55 miljoonaa mahdollista pokerikättä niputetaan 2,5 miljoonaksi abstraktoiduksi käsikategoriaksi (”buckets”). Neljännellä panostuskierroksella (riverillä) 2,4 miljardia mahdollista pokerikättä ryhmiteltiin vastaavasti 1,25 miljoonaan käsikategoriaan. Huomattavaa kuitenkin on, että Libratus

ei hyödynnä pohjapiirros-strategiaa kahdella viimeisellä panostuskierroksella, vaan sen sijaan käyttää toisen moduulin ”nested subgame solving” menetelmää, eli sisäkkäistä alipelien ratkaisua, joka ei sisällä korttien abstraktiota. Libratuksen toinen moduuli rakentaa pelin kahdelle viimeiselle panostuskierrokselle yksityiskohtaisemman abstraktion ja ratkaisee alipelin reaaliajassa pelikierroksen aikana. Alipelien ratkaisussa käytettävää abstraktiota voidaan myös vaihtaa reaaliajassa vastaamaan esimerkiksi vastustajan käyttämiä abstraktion ulkopuolisia panoskokoja. Toisin kuin täydellisen informaation peleissä, epätäydellisen informaation peleissä alipelin ratkaisua ei voi tehdä eristyksessä muusta pelistä, sillä optimaalinen strategia voi olla riippuvainen joistain pelin saavuttamattomista alipeleistä. Libratuksen rakentaakin alipelin ratkaisunsa siten että se sopii sen laajempaan pohjapiirros-strategiaan. Kehittyneen reaaliaikaisen alipelien ratkaisun voidaankin nähdä olevan kriittinen osa Libratuksen yli-inhimilliselle tasolle yltävää suorituskykyä. Libratuksen kolmas moduuli on niin sanottu ”self-improver” eli itsensä parantaja, joka mahdollistaa Libratuksen oppimisen aiemmista pelikokemuksista. Se täydentää Libratuksen pohjapiirros-strategiaa puuttuvien alipelien osalta, käyttäen vastustajan pelissä käyttämiä toimintoja ja panoskokoja. (Brown ja Sandholm 2018)

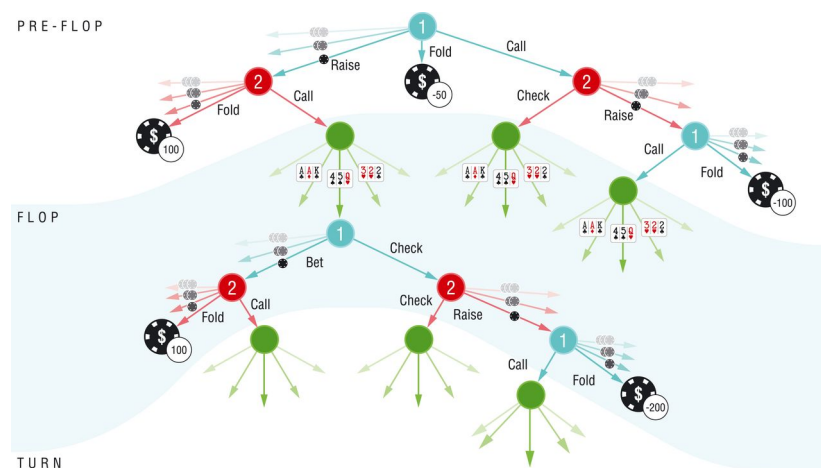
### **5.3 DeepStack**

Artikkelissaan ”DeepStack: Expert-level artificial intelligence in heads-up no-limit poker” Moravčík ym. 2017 esittelevät ”DeepStack”:iksi nimeämänsä epätäydellisen informaation pelien ratkaisuun kehitetyn yleiskäyttöisen algoritmin. DeepStack onnistui Libratuksen tapaan ensimmäisten joukossa päihittämään joukon pokeriammattilaisia Heads up No Limit Texas Hold’emissa. DeepStack kilpaili kokonaisuudessaan 33 eri ihmispelaajaa vastaan, pelaten 44 852 pokerikättä. DeepStackin ja sen haastajien välisiä otteluita lähetettiin suorana Twitch.tv suoratoistopalvelussa, ja ne ovat suurelta osin katseltavissa tallenteina YouTube:ssa. DeepStackin voiton marginaaliksi tutkijat ilmoittavat uskomattomat 492 mbb/g, mutta arvioivat tekoälyn olleen paikoitellen onnekas.

Kirjoittajien mukaan DeepStackin kehityksessä on käytetty aiemmista ratkaisuista eroavaa lähetymistapaa; DeepStack käyttää strategiansa päättelyyn muiden tekoälyohjelmistojen tapaan CFR-algoritmiä, mutta sen sijaan että tekoäly ratkaisisi pelin ennen pelaamista ja tal-

lentaisi tuotetun strategian levyille, DeepStack käsittelee pelitilanteita reaaliajassa niiden ilmaantuessa, mutta ei kuitenkaan eristyksissä muusta pelistä. Se välttää päättelyn koko pelin loppuosasta korvaamalla tietyn syvyyden ylittävän laskennan nopealla likimääräisellä arviolalla. Tätä ominaisuutta kehittäjät kuvailevat DeepStackin intuitioksi, aavistukseksi siitä kuinka arvokasta on omata tietyt yksityiset taskukortit kussakin pelitilanteessa. Samoin kuin ihmistenkin, tekoälyn intuitiota tulee kehittää, tähän tarkoitukseen DeepStackin arkkitehtuurissa hyödynnetään syväoppimis-menetelmää. (Moravčík ym. 2017)

Kirjoittajat hahmottavat kuinka pokeripelin julkista tilaa voidaan mallintaa puurakenteen avulla. Pokeripelin julkinen tila koostuu julkisista yhteisistä korteista, sekä pelaajien tekemistä toiminnoista, kuten panostuksista. DeepStackin algoritmi rakentuu kolmesta elementistä; paikallisesta strategialaskennasta sen hetkiselle julkiselle tilalle, syvyydeltä rajoitetusta ennakoinnista ("look-ahead"), jossa käytetään opittua arvofunktiota, sekä rajoitetusta määrästä sallittuja ennakoitintoimintoja. Kirjoittajien mukaan nämä kolme ainesosaa käsitteellisellä tasolla kuvaavat heuristista hakua ("heuristic search"), joka on ollut menetelmänä käytössä täydellisen informaation pelien tekoälyohjelmistoissa. Heuristisen haun menestyksellään käytöstä epätäydellisen informaation pelien parissa ei ole kuitenkaan ennen DeepStackia raportoitu. (Moravčík ym. 2017)



Kuvio 2. Julkinen pelipuu (Moravčík ym. 2017)

DeepStack ratkaisee optimaalisen strategian kussakin pelitilanteessa toiminto kerrallaan "continual re-solving"- eli jatkuvaa uudelleenratkaisu -menetelmää käyttäen, vasta silloin kun on

sen vuoro toimia pelissä. Algoritmi ei tallenna kokonaista strategiaa missään ratkaisun vaiheessa, vaan itseasiassa se pyrkii muistamaan kussakin tilanteessa vain oleellisen osan aiemmista pelitapahtumista; Ratkaistakseen kunkin pelivaiheen, DeepStack käyttää agentin omaavia taskukortteja ja niiden muodostamaa ”rangea”, eli skaalaa kaikista mahdollisista taskukorteista jotka pelaajalla voi olla hallussaan pelin kussakin vaiheessa, sekä vastustajan CFR:n tuottamia kontrafaktuaalisia arvoja (”counterfactual values”) kuvaavaa vektoria. Toiminnon tehdessään DeepStack päivittää oman rangensa tuotetun strategian mukaisesti, sekä vastustajan arvovektorin kuvaamaan pelin uutta julkista tilaa. Sen ei tarvitse missään kohtaa tiedostaa vastustajan rangea tai sen tekemiä toimintoja, mikä on tärkeä ominaisuus algoritmin tehokkuuden kannalta. Kuten aiemmin on mainittu, epätäydellisen informaation pelien tapauksessa kunkin pelitilanteen optimaalinen strategia on sidoksissa laajemman pelin strategiaan. Kiertääkseen tästä muodostuneen ongelman yksittäistä pelitilannetta ratkaistessaan DeepStack tekee hienostuneita oletuksia muun pelin tilasta opitun kontrafaktuaalisen arvofunktion avulla. Kontrafaktuaalisena arvofunktiona DeepStackin arkkitehtuurissa käytetään syvää neuroverkkoa, jota koulutettiin yli 10 miljoonalla satunnaisella pokeritilanteella. DeepStackin haku jatkuvan uudelleenratkaisun aikana on rajattu syvyydeltään neljään. Tämä abstraktion kaltainen menettely pienentää uudelleenratkaistavan pelin kokoa alkuperäisestä  $10^{160}$ :sta päätöskohdasta noin  $10^{17}$  päätös kohtaan. Syvyyden lisäksi ratkaisun aikaista hakupuuta harvennetaan antamalla DeepStackin valita toiminnoikseen vain luovutuksen, maksamisen, kaksi tai kolme eri panostustoimintoa, sekä ”all-in”-panostuksen. Syvyysrajatun ja harvennetun uudelleen ratkaistavan puun kooksi jää vain  $10^7$  päätös kohtaa ja sen ratkaisuun kuluu aikaa alle 5 sekuntia, mahdollistaen DeepStackin pelaamisen ihmisille tyypillisellä nopeudella. (Moravčík ym. 2017)

## 5.4 Pluribus

Aiemmissä aliluvuissa kuvatut tekoälyohjelmistot pystyivät yli-inhimillisiin suorituksiin kahden pelaajan Heads up No Limit Texas Hold’emissa. Pokeria kuitenkin yleensä pelataan isommalla pelaajamäärällä, mikä on tekoälyn kehityksen kannalta paljon monimutkaisempi ja haastavampi pelimuoto. Libratuksen kehittäjät, Carnegie Mellonin yliopiston tekoälytutkijat, Noah Brown ja Tuomas Sandholm kehittivät ”Pluribus” nimisen tekoälyohjelmiston,

joka opetteli pelaamaan kuusikäistä No Limit Texas Hold'emia pelaamalla viittä itsensä kopiota vastaan. Kokeellisessa arvioinnissaan Pluribus pelasi kokonaisuudessaan 20 000 pokerikättä joukkoa huippuluokan ihmispelaajia vastaan, joista jokainen on voittanut vähintään miljoona dollaria ammattilaisurallaan. Tutkimus oli jaettu kahteen osaan, joista ensimmäisessä yksi kopio Pluribuksesta pelasi samassa pelipöydässä viiden ihmispelaajan kanssa, ja toisessa vaiheessa yksi ihmispelaaja pelasi viittä Pluribusta vastaan. Tutkimuksen ensimmäiseen osaan osallistui 13 ihmispelaajaa, joihin kuului mm. Linus "LLinusLLove" Loeliger, jota pidetään yhtenä parhaimmista No Limit Texas Hold'em pelaajista maailmassa. Ensimmäinen vaihe kesti kokonaisuudessaan 12 päivää ja 10 000 pokerikättä. Päivittäin pelaajaryhmästä valittiin viisi vapaaehtoista ihmispelaajaa saatavuuden mukaan. Tutkimuksen toisessa vaiheessa kaksi ihmispelaajaa, Chris "Jesus" Ferguson ja Darren Elias, pelasivat kumpikin 5000 pokerikättä Pluribuksen kopioita vastaan. Kussakin tutkimuksen vaiheessa pelaajilla oli selkeät rahalliset motivaattorit pelata parhaan kykynsä mukaan, sekä pelaajat käyttivät peitenimiä, jotta toiset pelaajat eivät voisi tietää ketä vastaan pelaavat. Kokeen tuloksiin käytettiin "AIVAT" varianssinvähennys-tekniikkaa, jotta pokerin tuuri-elementti saataisiin minimoitua. Ensimmäisessä vaiheessa Pluribus onnistui päihittämään viisi ihmispelaajaa 48 mbb/g marginaalilla, joka on tutkijoiden mukaan todella korkea voittosuhde kuusikäisessä pokerissa. Myös toisen vaiheen kokeessa Pluribus onnistui päihittämään ihmispelaajat 32 mbb/g marginaalilla. (Brown ja Sandholm 2019)

Vaikka Nashin tasapainon on pystytty todistamaan olevan olemassa jokaisessa rajallisessa pelissä, sen löytäminen voi olla erityisen hankalaa. Tehokkaita algoritmeja Nashin tasapainon löytämiseen on löydetty vain tietyn tyyppiselle peleille, joista Heads up pokerin tyyppiset kahden pelaajan väliset nollasummapelit ovat erityisen edustettuina. Yli kahden pelaajan nollasummapeleissä Nashin tasapainon löytäminen on vaikeaa ja sen arvioiminen on teoriasakin liki mahdotonta, sillä jopa tehokkaimmat algoritmit ovat kykenettömiä käsittelemään yli kourallista strategioita pelaajaa kohden. Mikäli Nashin tasapaino pystyttäisiin tehokkaasti laskemaan yli kahden pelaajan pelille, on myöskin epäselvää olisiko tämän strategian noudattaminen kannattavaa; jos jokainen pelin pelaajista itsenäisesti laskee ja pelaa Nashin tasapainon mukaista strategiaa, lista pelattavista strategioista ei välttämättä ole Nashin tasapainossa. Tämän syynä on se, että monen pelaajan pelissä Nashin tasapainoon voidaan päätyä usealla eri tavalla, josta seuraa se, että Nashin tasapainojakin on useita. Sen sijaan että pyrki-

sivät ratkaisemaan monikäätisen pokerin, tutkijoiden tavoitteena olikin kehittää tekoälyohjel-  
misto, joka pystyisi empiirisesti ja johdonmukaisesti päihittämään parhaimpia ihmispelaajia.  
(Brown ja Sandholm 2019)

Pluribuksen pohjana toimii Libratuksen tapaan ennen pelaamista ratkaistu ja abstraktioon  
perustuva pohjapiirros-strategia. Pluribuksen pohjapiirros-strategian toimintojen abstraktio  
sisältää panoskokojen rajaamisen 1 ja 14 eri panoskoon väliltä, pelitilanteesta riippuen. Myös  
informaation abstraktiota hyödynnetään niputtamalla yhteen samankaltaisia pokerikäsiä, jois-  
ta kirjoittajat antavat esimerkiksi 10-korkean ja 9-korkean suoran, jotka ovat eri vahvuisia  
pokerikäsiä, mutta strategisesti hyvin samankaltaisia. Tuottaakseen tarkempia strategioita,  
Pluribus hyödyntää informaation abstraktiota vain tulevien panostuskierrosten osalta, eikä  
ikänä sen hetkiselällä panostuskierroksella. Pohjapiirros-strategia on ratkaistu käyttäen paran-  
neltua versiota Monte Carlo CFR -algoritmistä, joka mm. välttää selkeästi heikkojen pää-  
tösvaihtoehtojen tutkimista suurella osalla iteraatioista tehostaakseen algoritmin toimintaa.  
Luvussa 4.3 mainittu CFR:n takaus konvergoinnista Nashin tasapainoon on voimassa vain  
kahden pelaajan nollasummapeleissä. Se kuitenkin takaa sen, että kaikissa rajatuissa peleis-  
sä kontrafaktuaaliset katumusarvot kasvavat sublineaarisesti suhteessa iteraatioiden määrän  
kanssa. Tämä ominaisuus puolestaan takaa sen, että CFR:n jokaisen pelatun iteraation kes-  
kimääräinen suorituskyky vastaa parhaan mahdollisen yksittäisen kiinteän strategian keski-  
määräistä suorituskykyä. Pluribuksen pohjapiirros-strategian laskemiseen kului aikaa 8 päi-  
vää, käyttäen 64 ytimisellä palvelimella 12 400 CPU-ydintuntia ja se vaati käyttöönsä alle  
512 gigatavua muistia. (Brown ja Sandholm 2019)

Pluribus noudattaa pelatessaan pohjapiirros-strategiaa vain ensimmäisellä panostuskierrok-  
sella, jolloin mahdollisten päätöskohtien määrä on suhteellisen pieni, mikä sallii abstraktion  
olevan myöhempiä panostuskierroksia yksityiskohtaisempi. Myöhemmille panostuskierrok-  
sille Pluribus pyrkii määrittelemään tarkemman ja paremman strategian reaaliaikaisen haun  
avulla. Kirjoittajat kuvaavat DeepStackin käyttämää neuroverkkoon perustuvaa arvofunk-  
tiota laskennallisesti kalliiksi menetelmäksi järkeillä ratkaistavan pelitilanteen ulkopuolisen  
pelin osalta. Sekä Libratuksen käyttämä reaaliaikainen ratkaisu, joka yltää pelipuun loppuun  
asti, on laskennallisesti mahdotonta toteuttaa pelaajamäärän kasvaessa. Sen sijaan Pluribus  
käyttää muunneltua menetelmää, jossa yhden kiinteän strategian sijaan se olettaa, että vas-

tustaja tai vastustajat voivat käyttää pelissä neljää vaihtoehtoista strategiaa. Strategiat joita vastustajan oletetaan käyttävän ovat Pluribuksen pohjapiirros-strategia tai jokin sen muunnelmista; yksi strategioista suosii luovuttamista (fold), toinen maksamista (call) ja kolmas korottamista (raise). Tämän kaltainen menettely auttaa Pluribusta tuottamaan tasapainoisia strategioita reaaliaikaisen ratkaisun aikana. (Brown ja Sandholm 2019)



## 6 Yhteenveto

Tekoälykehitys on tullut pitkälle niin täydellisen informaation, kuin epätäydellin informaation pelien ratkaisussa. Täydellistä ratkaisua No Limit Texas Hold'emille ei ole sen suuresta koosta johtuen pystytty toteuttamaan, eikä luultavasti vielä pitkään aikaan tulla toteuttamaan. Parhaimpia ihmispelaajia voittavia ohjelmistoja on kuitenkin pystytty kehittämään. Pokerin sisältäessä runsaasti satunnaisuutta, on kuitenkin pohdittava sitä, kuinka merkittäviä nämä tulokset ovat, vaikka erinäisiä varianssinvähennys-menetelmiä onkin pyritty käyttämään tulosten analysoimisessa. Kohtalaisen kokeneena pokerinpelaajana osaan sanoa sen, että selvittääkseen kunkin pelaajan todellisen suorituskyvyn tai voittosuhteen pelaajien välillä, tulee pelaajien pelata satojatuhansia pokerikäsiä. Tämän huomioon ottaen heads-upissa Libratuksen voitto tarkkaan valituista neljästä huippuammattilaisesta 120 000 pokerikäden otannalla on mielestäni selkeästi merkittävin saavutus, vaikka DeepStackin saavuttama voittosuhte onkin paljon suurempi pienemmällä otannalla. Listaa DeepStackin kohtaamista pokeriammattilaisista ei tutkimuksessa julkaistu, joten jää epäselväksi olivatko kaikki kokeeseen osallistujista pokerimaailman huipulta vai alemmilta panostasoilta. Myöskin kunkin DeepStackin haastajista pelasi verrattaen pienen määrän pokerikäsiä tekoälyä vastaan; vain 11 pelaajista pelasi pyydetyt 3000 kättä, joten todellisten heikkouksien havainnointi tekoälyn strategiasta on välttämättäkin jäänyt vähäiseksi. Pluribuksen sovellutukset monikäyttöiseen pokeriin ovat myös iso askel eteenpäin todellisen maailman ongelmien ratkaisussa, sillä todellisen maailman ongelmat sisältävät harvoin vain kaksi vastakkaista toimijaa. Sen yli-inhimillinen suorituskyky näyttää sen, että kahden pelaajan välisiin nollasummapeleihin sovellettavat konseptit ovat käytännöllisiä myös monen pelaajan pelien ratkaisussa, vaikkakin ilman selkeää teoreettista taustaa.

## Lähteet

“Artificial Intelligence”. 2022. <https://www.britannica.com/technology/artificial-intelligence>.

Bowling, Michael, Neil Burch, Michael Johanson ja Oskari Tammelin. 2015. “Heads-up limit hold#x2019;em poker is solved”. *Science* 347 (6218): 145–149. <https://doi.org/10.1126/science.1259433>. eprint: <https://www.science.org/doi/pdf/10.1126/science.1259433>. <https://www.science.org/doi/abs/10.1126/science.1259433>.

Brown, Noam, Adam Lerer, Sam Gross ja Tuomas Sandholm. 2019. “Deep Counterfactual Regret Minimization”. Teoksessa *Proceedings of the 36th International Conference on Machine Learning*, toimittanut Kamalika Chaudhuri ja Ruslan Salakhutdinov, 97:793–802. Proceedings of Machine Learning Research. PMLR, syyskuu. <https://proceedings.mlr.press/v97/brown19b.html>.

Brown, Noam, ja Tuomas Sandholm. 2018. “Superhuman AI for heads-up no-limit poker: Libratus beats top professionals”. *Science* 359 (6374): 418–424. <https://doi.org/10.1126/science.aao1733>. eprint: <https://www.science.org/doi/pdf/10.1126/science.aao1733>. <https://www.science.org/doi/abs/10.1126/science.aao1733>.

———. 2019. “Superhuman AI for multiplayer poker”. *Science* 365 (6456): 885–890. <https://doi.org/10.1126/science.aay2400>. eprint: <https://www.science.org/doi/pdf/10.1126/science.aay2400>. <https://www.science.org/doi/abs/10.1126/science.aay2400>.

“Carnegie Mellon University”. 2017. <https://www.cmu.edu/news/stories/archives/2017/january/AI-beats-poker-pros.html>.

“DeepMind”. 2022. <https://deepmind.com/research/case-studies/alphago-the-story-so-far>.

Haenlein, Michael, ja Andreas Kaplan. 2019. “A Brief History of Artificial Intelligence: On the Past, Present, and Future of Artificial Intelligence”. *California Management Review* 61 (4): 5–14. <https://doi.org/10.1177/0008125619864925>. eprint: <https://doi.org/10.1177/0008125619864925>. <https://doi.org/10.1177/0008125619864925>.

Hsu, Feng-Hsiung. 1999. "IBM's Deep Blue Chess grandmaster chips". *IEEE Micro* 19 (2): 70–81. <https://doi.org/10.1109/40.755469>.

"IBM". 2022. <https://www.ibm.com/ibm/history/ibm100/us/en/icons/deepblue/>.

Johanson, Michael. 2013. "Measuring the size of large no-limit poker games". *arXiv preprint arXiv:1302.7008*.

Moravčík, Matej, Martin Schmid, Neil Burch, Viliam Lisý, Dustin Morrill, Nolan Bard, Trevor Davis, Kevin Waugh, Michael Johanson ja Michael Bowling. 2017. "DeepStack: Expert-level artificial intelligence in heads-up no-limit poker". *Science* 356 (6337): 508–513. <https://doi.org/10.1126/science.aam6960>. eprint: <https://www.science.org/doi/pdf/10.1126/science.aam6960>. <https://www.science.org/doi/abs/10.1126/science.aam6960>.

"Pokerfuse". 2015. <https://pokerfuse.com/news/media-and-software/26854-doug-polk-and-team-beat-claudico-win-100000-microsoft/>.

Sandholm, Tuomas. 2015. "Abstraction for Solving Large Incomplete-Information Games", <https://www.aaai.org/ocs/index.php/AAAI/AAAI15/paper/view/10039/9692>.

Shannon, Claude. 1950. "Programming a Computer for Playing Chess". *Philosophical Magazine, Ser 7* (41): 314.

"The History of Artificial Intelligence". 2017. <https://sitn.hms.harvard.edu/flash/2017/history-artificial-intelligence>.

"WSOP". 2019. <https://www.wsop.com/news/2019/Jul/11156/HOSSEIN-ENSAN-IS-THE-2019-WSOP-MAIN-EVENT-CHAMPION.html>.

Zinkevich, Martin, Michael Johanson, Michael Bowling ja Carmelo Piccione. 2007. "Regret minimization in games with incomplete information". *Advances in neural information processing systems* 20.

## **Liitteet**