

Slovenská technická univerzita v Bratislave  
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

# **Virtuálny vesmír**

Peter Orosi

# 1 Obsah

2	Základné pojmy a uvedenie do problematiky .....	2
2.1	Čo je počítač a čo je program ? .....	2
2.2	Čo je to vesmír ? .....	4
3	Zostrojenie virtuálneho vesmíru.....	9
3.1	Čo je to algoritmus ?.....	9
3.2	Čo je to priestor, čo je to čas ?.....	11
3.3	Čo je to poznanie a informácia ? .....	14
4	Simulácia virtuálneho vesmíru.....	21
5	Zdroje.....	26

## 2 Základné pojmy a uvedenie do problematiky

V tejto časti sa pokúsím „populárnym“ spôsobom vysvetliť základné myšlienky, ktoré ma viedli k vytvoreniu toho čo nazývam virtuálny vesmír a pokúsím sa zamyslieť či a ako by to prípadne mohlo súvisieť s našou realitou. Kvôli tomu, aby som dal veci do súvisu uvádzam najskôr niektoré všeobecne známe pojmy a pokúsím sa na ne potom pozrieť trochu iným spôsobom. V prvom rade, keďže chcem vytvoriť virtuálny vesmír, ktorý budem simulovať na počítači, musím sa najskôr zamyslieť nad tým čo je počítač a potom čo je to vesmír a ako by to prípadne mohlo spolu súvisieť.

### 2.1 Čo je počítač a čo je program ?

Použijem takúto zjednodušenú definíciu, počítač =

- Procesor
- Operačná pamäť
- Vstupno-výstupné zariadenia

Používateľ napr. napíše do klávesnice nejaké slovo, to sa prevedie na postupnosť jednotiek o núl, tie sa uložia do pamäte a nad nimi sa spustí algoritmus, ktorý tieto informácie transformuje na základe nejakých pravidiel – na inú postupnosť jednotiek a núl. Výstupné zariadenie ako napr. monitor potom dokáže toto písmeno graficky reprezentovať. Čo je algoritmus ? Bežne tak označujeme postupnosť krokov, ktorá prevedie konečne veľký vstup na konečne veľký výstup v konečnom čase (konečne veľkom počte krokov). V počítači je algoritmus tiež uložený ako postupnosť jednotiek a núl, spolu s informáciami nad ktorými sa algoritmus vykonáva, obe jestvujú iba ako postupnosť jednotiek a núl – fyzicky reprezentované stavom tranzistorov operačnej pamäte. A tu je hneď prvý zaujímavý dôsledok – algoritmus je tiež akoby informácia – inak povedané, bez toho, aby jestvovala informácia, by nemohol jestvovať ani algoritmus. A navyše ak by aj jestvoval iba algoritmus (nejakým pre mňa nepredstaviteľným spôsobom bez toho, že by musela jestvovať informácia) – takýto algoritmus by bol bez informácie zbytočný, keďže by sa nemal nad čím vykonávať.

Na druhej strane, ak by neexistoval algoritmus iba informácia, celý počítač by sa redukoval iba na množinu informácii, ktoré by ale boli zbytočné, pretože by boli nemenné a nedali by sa medzi sebou transformovať – inak povedané: existencia informácie bez algoritmov by takisto nemala zmysel. To je napr. dôvod, prečo majú rôzne typy súborov rôzne prípony – prípona akoby definovala, ktoré algoritmy sa majú použiť, bez vhodného algoritmu by pre nás napr. JPEG súbor ostal iba nič nehovoriaca postupnosť bitov, je to práve algoritmus, ktorý rozlišuje medzi nezmyselnou postupnosťou jednotiek a núl a inou postupnosťou jednotiek a núl, ktorá je pre nás zrozumiteľná (vstup do monitora po poskladaní obrázku je vlastne tiež iba postupnosť 1,0). Inými slovami – snažím sa naznačiť, že informácia a algoritmus spolu súvisia a to trochu hlbším spôsobom ako v klasickom ponímaní, kde sme ich zvyknutý oddeľovať (celé to spresním v ďalších častiach dokumentu.)

Teraz na chvíľu trochu odbočím k niečomu, čo sa nazýva Turingov test. Ako je známe, jedná sa o test, pri ktorom spolu konverzujú človek a počítač prostredníctvom klávesnice a obrazovky, pričom existuje tretí pozorovateľ - prípadne celá komisia, ktorá rozhovor sleduje bez toho, aby vedela ktorý účastník je človek a ktorý počítač. Počítač prejde testom vtedy ak sa pozorovateľovi javí na nerozoznanie od skutočného človeka. Predstavme si, že niekedy v budúcnosti by sa podarilo počítaču prejsť Turingovým testom. Skúste si

odpovedať na otázku či si myslíte, že je to možné. Áno ? Nie ? Ja osobne by som si tipol že áno, možno nie dnes, možno nie zajtra, ale z hľadiska rozvoja umelej inteligencie som optimista a verím, že raz na tom budú naše počítače podobne dobre ak nie lepšie ako napr. počítače zo Star-Treku. Ak pripustíme, že je možné, aby teda počítač prešiel Turingovým testom, potom na jednej strane by to znamenalo obrovský úspech umelej inteligencie – to je očividné. Možno menej očividná je opačná strana implikácie – ak by sa toto podarilo, znamenalo by to v podstate to, že ľudské myšlienky a logické operácie nad jednotkami a nulami sú medzi sebou zameniteľné, alebo dokonca – že sú jedno a to isté. Mohli by sme sa na celú konverzáciu pozeráť napr. takto: človek procesom, ktorý nazývame myslením vytvorí odpoveď a napíše ju do klávesnice. Tá sa transformuje na postupnosť 1, 0, počítač ju spracuje, vlastným „myslením“, ktoré je na nerozoznanie od ľudského a výstup (zase postupnosť 1,0) sa objaví na monitore, ktoré človek pochopí (zase pomocou procesu myslenia) a dá naň odpoveď (myslenie) atď. Človek by v podstate tiež iba preberal vstupy (monitor – výstup počítača), transformoval ich (na základe myslenia) a poslal na výstup (písanie do klávesnice – čo je vlastne vstupom počítaču). Otázka teda znie – mohol by byť človek tiež iba nejaký počítač / program ? Toto možno na prvý pohľad vyzerá nezmyselne, avšak skúsme tento pohľad skombinovať s poznatkami súčasnej biológie. Ak si odmyslíme niečo ako duša duchovno a ďalšie metafyzické záležitosti (v týchto otázkach nám bohužiaľ súčasná veda nevie poradiť), potom podľa súčasného poznania je práve DNA (genotyp) niečo ako zdrojový kód nášho programu. DNA riadi syntézu bielkovín a tým determinuje náš fenotyp. Fenotyp (telo) je iba interakcia DNA (programu) s prostredím – DNA bez adekvátneho prostredia by totiž nemala zmysel, ak by sme jej neprivádzali na vstup potrebné chemické zlúčeniny, nikdy by sa z nej nevytvorilo niečo ako ľudské telo.

Ak by jestvovala iba DNA a žiadne prostredie, nikdy by sa z nej nemohol vytvoriť fenotyp. Ak by sme jej zmenili prostredie – napr. privádzali na vstup namiesto potrebných živín dávku gama žiarenia, určite by sme mali očakávať, že tento program bude dávať úplne iné výstupy (pravdepodobne sa naruší jeho štruktúra a nebude dávať žiadne výstupy – podobne ako keď skončí program v počítači). Fenotyp je teda to, čo sa vytvorí z DNA na základe prostredia. Je to naše telo, napr. konkrétny človek, ktorý sa zúčastňuje Turingovho testu. Ten zase preberá vstupy a tie transformuje na výstup. Čiže fenotyp je z tohto pohľadu tiež iba akoby program, ktorý transformuje vstupy na výstupy. Na druhej strane sme ale povedali že fenotyp je determinovaný genotypom (DNA) a teda z „programátorského“ hľadiska nemá zmysel rozlišovať medzi genotypom a fenotypom. Človek ako taký, je z tohto pohľadu v konečnom dôsledku iba interakcia informácie obsiahnutej v DNA s prostredím a na základe rôznych vstupov prostredia potom dáva rôzne výstupy. DNA a jej hlavná informácia ktorú nesie je iba postupnosť jednotiek a núl - štvoricu báz adenozín, cytozín, guanín, tymín by sme vedeli ľahko digitalizovať ak by sme poznali ich presné poradie) – túto postupnosť 1, 0 by sme mohli nazvať „nehmotná podstata dedičnosti“. Ak by sme túto postupnosť mali uloženú v počítači, znamenalo by to, že vlastne máme niečo čo (pravdepodobne) determinuje celý život jedného človeka v závislosti od prostredia kam tento program vložíme. (ak pripustíme, že niečo ako ľudská duša neexistuje, prípadne ak existuje, potom že je tiež nejakým spôsobom zakódovaná v DNA.) Tento vzor uložený v DNA (predpis, postupnosť 1, 0 atď.) sa ale stáva zmysluplnou ak sa realizuje - ak sa vloží do nejakého prostredia, ak sa jej dá nejaká forma – napr. ako stočená štruktúra štyroch základných báz a privedieme k nej vstupy. Samozrejme je tu otázka, či tento algoritmus je deterministický, či je to algoritmus v pravom zmysle slova, či to nie je v protirečení so slobodnou vôľou človeka a pod. Tu si treba uvedomiť, že akýkoľvek nedeterminizmus, ktorý sa nám javí na výstupe môže byť iba výsledkom nedeterminizmu (náhodných) dát na vstupe. Napr. mutácia v DNA je vlastné prepísanie programu teda je to podobné, ako keby algoritmus v dôsledku vstupov zmenil svoju vnútornú štruktúru a na niektoré vstupy tak začal dávať iné výstupy (vieme, že

z informatického hľadiska existujú samomodifikujúce sa programy) a teda je to stále program a jeho začiatkový stav spolu s prostredím (ktoré môže byť aj náhodné) ho plne determinujú. Otázka slobodnej vôle je zložitejšia, ak by sme ale pripustili, že množina vstupov aj výstupov v jednom momente sa nemusia nutne determinovať, potom však vždy môžeme povedať, že vyvinutý fenotyp obsahuje v sebe niečo ako generátor náhodných čísel (ako zdroj nedeterminizmu) pričom na dva rovnaké vstupy by mohol reagovať aj rôznym spôsobom a spôsob akým sa tento generátor vytvorí je zakódované v DNA. V konečnom dôsledku teda stále platí, že aj na niečo tak zložité ako je človek sa môžeme z určitého uhlu pohľadu. pozrieť ako na program, pričom zdrojový kód je postupnosť 1,0 vložená do vhodného prostredia.

Možno samozrejme namietat', že tento pohľad je dosť pritiaľnutého za vlasy – hlavne kvôli metafyzickým otázkam na ktoré nepoznáme zaručenú odpoveď. (či existuje slobodná vôľa, či existuje duša atď.) O čo mi ide je poukázať, že nemusí byť až také nezmyselné skúsiť sa pozrieť na vec aj takýmto spôsobom. Nejde mi o to presvedčať, že to tak musí byť, ako skôr vzbudiť podozrenie typu „čo ak predsa na tom niečo je ?“. Venujem tomu preto takúto pozornosť, aby som ukázal, že ak sa môžeme pozerat' takýmto spôsobom na človeka (najzložitejší objekt vesmíru aký poznáme), potom sa tak vlastne môžeme pozerat' na všetko vo vesmíre - ako na niečo, čo je riadené programom. Netvrďím, že tento pohľad je správny, ale ako som uviedol - nepripadá mi byť zas až na toľko absurdný, aby som sa ním nemohol zaoberať.

Mojim cieľom v tejto práci primárne nie je poukázať na to, že by vesmír mal byť nejakým spôsobom program, prípadne počítač skôr mi išlo o zostrojenie virtuálneho vesmíru, ktorý by sa podobal na skutočný. V tomto virtuálnom vesmíre som však objavil určité náznaky, ktoré ma naviedli práve týmto smerom, pokúsiť sa zamyslieť či tieto dva vesmíry (virtuálny a reálny) spolu dáko súvisia, preto sa nevyhýbam ani tejto problematike. V skutočnosti sa nemôžem pozerat' na tento problém iba z hľadiska informatiky, pretože je to problém, v ktorom sa prelína viacero hľadísk ako napr. fyzika, biológia, filozofia. Občas sa preto pokúsim venovať aj takýmto problémom.

## **2.2 Čo je to vesmír ?**

Ako už bolo naznačené, pokúsime sa na vesmír pozrieť najmä očami programátora. Ak chceme o čomkoľvek uvažovať ako o programe určite najskôr musíme vysloviť predpoklady, že toto „niečo“ je konečné a diskkrétne (diskkrétne čiže nespojité. Spojité automaticky implikuje nekonečné) a pokiaľ hovoríme o virtuálnom vesmíre, máme na mysli, že budeme vykonávať jeho simuláciu. A pod simuláciou budeme rozumieť zmenu stavu vesmíru v jednom časovom kvante do nového stavu v nasledujúcom časovom kvante (simulačnom kroku).

Ako by asi mal vesmír v čo najvšeobecnejšom priblížení vyzerat' ? Tak napr. mohla by byť známa hra „Game Of Life“ vesmírom ? Z určitého uhlu pohľadu áno. Jedná sa v podstate o štvorcovú plochu (niečo ako priestor nášho vesmíru) vyplnenú čiernymi a bielymi bodmi (atómami). Zmena v časovom kvante znamená, že farba každého bodu v priestore sa na základe určitých pravidiel buď preklopí na opačnú, alebo ostáva. Takýto vesmír teda má svoj vlastný priestor, svoj vlastný čas, aj pravidlá ktoré ho riadia v každom simulačnom kroku. Napriek tomu vieme, že zložitou sa ani náhodou nepribližuje k skutočnému vesmíru. Ak by sme napr. priestor z hry „Game Of Life“ previedli na veľkosť podobnú rozmerom vesmíru, pravdepodobne by sme tým na zložitosti nezískali nič nové. Stále by sa tam objavovali už známe obrazce, ktoré je možné odsimulovať aj na ploche napr. 32 x 32 štvorcov. Čo teda odlišuje takýto „vesmír“ od toho skutočného ? (okrem toho že tu sa jedná o dvojrozmerný priestor, čo ale nehrá rolu). Napadá ma hneď viacero bodov, voči ktorým možno namietat':

1. Odkiaľ sa zobrali pravidlá, ktoré ho riadia ? Prečo by farba atómu mala záležať iba od všetkých najbližších susedných atómov a nie napr. od všetkých v najbližších s polomerom 5 alebo 10, alebo iba od všetkých nepárnych a prečo by sa mala farba iba preklápať z bielej na čiernu ? Prečo by napr. body nemohli putovať po ploche a pôsobiť na seba aj na diaľku ? (niečo čo by sme v našom vesmíre nazývali silovým pôsobením, napr. gravitáciou)
2. Odkiaľ sa vzal začiatočný stav takéhoto vesmíru ? Obvykle človek vloží „nejaký“ útvar a pozoruje ako sa mení v čase, bez toho, aby bolo nejako logicky zdôvodnené prečo práve takýto útvar by mal byť zaujímavý.
3. Prečo práve určité rozmery vesmíru, napr. 32x32 štvorcov a nie iné ? Keďže takýto vesmír je deterministický a konečný, po určitej dobe sa generované obrazce musia začať opakovať. Ako to súvisí s rozmermi vesmíru ?
4. Prečo by mal vesmír byť práve deterministický ? (prípadne: prečo by mal byť nedeterministický ak by sme ho spravili nedeterministickým)

Ani na jednu z týchto otázok hra „Game Of Life“ nedáva uspokojivú odpoveď (samozrejme Game Of Life je tu použitá iba ako príklad, predpokladám že ambíciou autorov nebolo vytvorenie vesmíru). Vesmír, ktorý je určitým spôsobom „dobrý“ by mal dávať odpoveď na tieto otázky, spôsobom, ktorý by bol čo najmenej spochybniteľný. Na koncept virtuálneho vesmíru ma priviedli tri (štyri, z toho tá posledná je tam z programátorských dôvodov zarátaná viacmenej implicitne) základne myšlienky, ktoré ma napadli v rôznych časoch a potom sa akoby spojili v jednu komplexnú ideu, ktorá bola ich logickým vyústením. Boli to približne podobné úvahy:

1. Pri štúdiu knihy Evolučné algoritmy [1] ma zaujala strana 143, kde sa opisuje virtuálny svet Tierra ( <http://www.his.atr.jp/~ray/tierra/> ). Jedná sa o softvér umožňujúci digitálnu evolúciu jednobunkových organizmov, ktorým bol na začiatku vložený jednoduchý kód (v podobe inštrukcií, ktoré boli simulované na virtuálnom počítači) umožňujúci vlastnú reprodukciu a počas simulácie tohto digitálneho sveta dochádzalo k vzniku úplne nových „zdrojových kódov“ organizmov. Iba pomocou mutácii a náhodných zmien v ich kóde sa vytvorili rôzne formy života, ktoré napr. na sebe parazitujú a pod. Zároveň sa na stránke uvádza, že s vývojom niečoho ako inteligentný život sú spojené problémy. Tento koncept digitálnej evolúcie sa mi vtedy dosť zapáčil a zároveň som sa pokúšal nájsť vysvetlenie, prečo je vývoj inteligentného života v tomto virtuálnom svete spojený s ťažkosťami. Takmer okamžite ma napadlo, že hlavný problém bude pravdepodobne v jednoduchosti prostredia. Napr. ak by náš vesmír celý pozostával iba z plochej dosky a bol by vyplnený potravou (v dostatočnom množstve pre každého), pričom každý tvor v ňom by sa živil iba touto potravou – neexistoval by žiaden dôvod k evolúcii. Byť inteligentným by nepredstavovalo žiadnu evolučnú výhodu, dokonca evolúcia samotná by sa stala zbytočnou, keďže všetko akoby už fungovalo dokonale. Je to práve prostredie, ktoré je katalyzátorom evolúcie napr. stačí ak sa zmení a v tom okamihu prežívajú tí, ktorí sa mu dokážu lepšie prispôbiť. Z tohto pohľadu má napr. človek ako inteligentný tvor oproti ostatným organizmom na planéte obrovskú konkurenčnú výhodu, čím lepšie chápeme zákonitosti vesmíru, tým viac ich dokážeme využívať vo svoj prospech. Ak napr. prostredie poskytuje možnosť vytvorenia strelnej zbrane (čo si napr. vyžaduje znalosti z chémie ohľadne reaktívnosti použitej zmesi na vymrštenie projektilu, mechanické konštrukčné princípy, znalosti o materiáli z ktorého je vyrobená zbraň, aby bola odolná a aby sa pri explózii nerozletela, alebo o spôsobe ako skonštruovať projektil, aby čo najlepšie vyhovoval zákonom aerodynamiky atď.) – tak potom znalosť ako takúto zbraň skonštruovať znamená veľkú konkurenčnú výhodu oproti ostatným tvorom, ktoré takouto schopnosťou

nedisponujú – inak povedané, ak by človek chcel mohol by si brať z prostredia na úkor ostatných a žiaden „neinteligentný“ tvor by sa mu nevedel vzoprieť. Bez toho, aby vesmír v sebe obsahoval „problémy“ na lúštenie, prípadne možnosti ako vyššie poznanie uplatniť vo svoj prospech, bez týchto podmienok by inteligentný život takmer určite nevznikol – inteligencia je z môjho pohľadu akoby pevne prepojená s prostredím. Dokonca aj na vznik neinteligentného života musí vesmír disponovať určitými zákonmi, ktoré ho menia v čase (ak by bol nemenný, neexistovala by ani evolúcia) A tieto zákony predstavujú tiež nejakú informáciu, alebo poznanie. Inak povedané, čím väčšie poznanie a čím zložitejšie zákony v sebe vesmír sústreďuje tým pravdepodobnejší je výskyt evolúcie a tým zložitejšie (inteligentnejšie) organizmy môžeme očakávať. Preto ako jednu z podmienok svojho virtuálneho vesmíru som si stanovil, aby v sebe sústreďoval „nejakým“ spôsobom poznanie (vysvetlím ďalej čo si predstavuje pod pojmom poznanie).

2. Položme si teraz otázku – ako možno predpovedať budúcnosť vesmíru ? Zoberme si jednoduchý príklad s predpoveďou počasia – dá sa to ? V súčasnosti existujú určité výpočtové modely, ktoré dokážu na krátku dobu pomerne úspešne počasie predvídať avšak z dlhodobého hľadiska sa to dobre nedá – platí niečo známe pod pojmom efekt motýľích krídel, t.j. aj mala lokálna zmena môže mať vplyv na globálnu klímu. Teraz si skúsme predstaviť, čo by to vyžadovalo ak by sme chceli predpovedať počasie na nejakú dlhšiu dobu – povedzme aspoň rok dopredu. Stačilo by k tomu, aby sme odsimulovali pohyb všetkých atómov v ovzduší a ich vzájomné interakcie ? Pripusťme, že by sme toto dokázali (aj keď prakticky si sotva možno predstaviť ten potrebný výpočtový výkon a takisto, že by sme presne vedeli odmerať stav všetkých molekúl v jednom momente). Výsledky by boli pravdepodobne presnejšie, avšak vždy sa môže stať niečo s čím sme „nerátali“. Napr. môže vybuchnúť sopka a uvoľnená síra v ovzduší zbúra naše výpočty ako domček z karát. Ešte horšie čo sa môže stať je, že sa Zem zrazí s meteorom a vyvrhnutý prach zatemní oblohu na dlhé obdobie. To v našom modeli na výpočet počasia ťažko zahrnieme, ale bez uváženia aj takýchto „externých“ udalostí nikdy nemôžeme tvrdiť, že máme stopercentne fungujúci stroj na predpoveď počasia. A nemusí pri tom dôjsť ani ku katastrofe, človek ako tvor žijúci na Zemi sám vplýva na počasie. Napr. ak by sme chceli na začiatku 20. storočia predpovedať počasie až po súčasnosť (vyše 100 rokov), museli by sme do nej zhrnúť aj poznanie o vývoji ľudskej civilizácie, pretože rozmach automobilového, leteckého, energetického atď. priemyslu má takisto vplyv na počasie (znečistenie). Avšak na ľudí vplýva strašne veľa vecí, napr. aj svetlo vzdialených hviezd (obzvlášť ak napr. dôjde k výbuchu supernovy, ktorý je viditeľný aj voľným okom). Inými slovami – predpovedať počasie presne a na dlhý čas dopredu je rovnako ťažký problém, ako predpovedanie budúcnosti vesmíru ako celku so všetkými jeho časťami (t.j. napr. aj s ľuďmi, všetkou medzihviezdou hmotou aj hviezdami atď.), čo je poriadne ťažký problém a vyžadovalo by si to prakticky poznanie stavu celého vesmíru v určitom čase a potom prepočítanie všetkých interakcií tých najelementárnejších častíc aké existujú – pravdepodobne každá s každou. Výpočtový výkon, ktorý by sme k tomu potrebovali by bol v merítke našich súčasných výpočtových možnosti nekonečný. Napriek tomu, ak chceme presne predpovedať budúcnosť niet inej možnosti. Fyzikálne zákony pomerne dobre opisujú makrosvet (pohyb planét), mikrosvet (kvantovo mechanický princíp – popis reality na základe pravdepodobností). Žiaden fyzik však v súčasnosti nevie predpovedať ako sa ja ako človek so slobodnou vôľou rozhodnem, hoci som tiež súčasťou vesmíru moje konanie je pre súčasnú fyziku s jej poznaním fyzikálnych zákonov „nevysvetliteľné“. Tu sa dostávame k otázke determinizmu vs. nedeterminizmu vesmíru. Keď chcem vytvoriť svoj vlastný virtuálny vesmír musím sa rozhodnúť medzi tým či bude deterministický alebo nie. Treba si uvedomiť aké možnosti ja ako programátor mám k dispozícii. Dokážem nejakým spôsobom naprogramovať náhodu ?

Existuje na to nejaký algoritmus ? Inými slovami, existuje algoritmus, ktorý na dva rovnaké vstupy môže dať dva rôzne výstupy ? O žiadnom takom neviem. Znamenalo by to vlastne, že jedna inštrukcia s rovnakými operandami (pri rovnakom začiatočnom stave spustenia programu) sa môže vyhodnotiť na dva rôzne výsledky. Toto by som určite nenazýval nedeterminizmom, ale skôr chybou procesora – som presvedčený o tom, že počítač, ktorý mám na stole sa takto nespráva, inak by ani nebolo možné aby fungoval. V súčasnosti už síce boli vyvinuté aj procesory, ktoré obsahujú generátor náhodných čísel (na podporu šifrovania), ale v konečnom dôsledku táto náhodnosť sa nezískava žiadnym algoritmom ale z externých zariadení ako napr. senzor teploty procesora a pod. Analogicky ak máme softvérový vytvorený generátor náhodných čísel, ten sa správa úplne deterministicky, jediné čo vyvoláva ilúziu náhody je náhodný vstup, ktorý býva väčšinou získaný opäť z externého zariadenia – napr. systémového časovača (na milisekundy presný čas v ktorom bol program spustený je náhodný parameter – čiže vstup nášho programu). Bez náhodných vstupov neviem zabezpečiť náhodné správanie algoritmu. Na druhej strane vesmír ako taký predstavuje uzavretý systém, nič doňho nevstupuje, nič z neho neodchádza, jestvuje sám o sebe a preto z programátorského hľadiska nemám možnosť ako zabezpečiť náhodné vstupy. Inak povedané, ak má byť môj virtuálny vesmír algoritmizovateľným a zároveň uzavretým systémom (a teda nemôže mať žiadne vstupy ani výstupy) potom musí byť deterministický – ja ako programátor nemám k dispozícii nič čo by sme mohli označiť pojmom náhodný algoritmus. Jedným z dôsledkov takejto uzavretosti je aj to, aby začiatočný stav vesmíru vplynul akosi „sám od seba“ (t.j. ani začiatočný stav by minimálne nemal byť náhodný). Zároveň ak náš virtuálny vesmír je deterministický potom pre každý krok simulácie vieme presne predpovedať jeho nasledujúci stav. Spôsob akým to dosiahneme by na základe toho čo bolo povedané o predpovedaní budúcnosti mal byť zrejmý. Základný princíp je, že všetko so všetkým súvisí a preto vstupom algoritmu by mal byť celý aktuálny stav vesmíru a jeho výstupom celý nasledujúci stav (alebo nejaký iný spôsob pri ktorom máme zaručené, že všetko so všetkým súvisí). To je trochu iný spôsob ako by k tomu pristupovala fyzika. Vo fyzike by sme sa snažili celú sústavu rozložiť na jednotlivé „atómy“ a odsimulovať všetky interakcie ktoré medzi nimi existujú na základe fyzikálnych zákonov. Pri tak veľkom počte atómov aký obsahuje vesmír by to však bolo veľmi (pravdepodobne exponenciálne časovo) náročné, minimálne rovnako komplikované na programovanie a navyše ani nemáme úplnú znalosť fyzikálnych zákonov. Okrem toho by tu vznikla otázka odkiaľ sa vzali fyzikálne zákony a prečo som použil práve takéto zákony a nie iné (ak by som sa ich napr. pokúšal vytvoriť pre svoj virtuálny vesmír „nanovo“, nezávisle od zákonov reálneho vesmíru). Ja ako programátor sa určite nechcem vydať takouto zložitou cestou. Môj prístup je všeobecnejší – nepopisujem čo presne sa má pri zmene stavu odohrať (zatiaľ sa tým nezaobieram), iba že sa to má zmeniť podľa základného princípu všetko so všetkým súvisí. Tým že je zadefinovaný takto všeobecne, výstupom algoritmu môže byť prakticky čokoľvek a teda ľubovoľný zákon fyziky môže byť jeho podmnožinou. T.j. ak si vesmír bude „želať“ ešte vždy sa môže správať podľa zákonov fyziky.

3. Ak je vesmír deterministický a riadi sa určitými pravidlami, potom určite musí platiť, že v každom stave, v ktorom sa vesmír ocitne existuje nejaké pravidlo, ktoré jeho aktuálny stav zmení na nový stav a tiež, že toto pravidlo je zároveň jediné pravidlo, ktoré existuje (ak by existovali dve rôzne pravidlá ako zmeniť aktuálny stav vesmíru do nového, potom by sme už nemohli hovoriť o determinizme). Na druhej strane ak by bol vesmír vytvorený nejakým programátorom a naša realita je výsledkom behu takéhoto programu, potom musíme konštatovať, že autor (stvoriteľ) napísal tento kód naozaj dobre, neporovnateľne lepšie ako čokoľvek čo kedy človek vytvoril – už len z dôvodu, že stále ešte nechápeme pravidlám, ktorými sa vesmír riadi. Ak teda nazveme vesmír programom, potom ako



každý program je zložený z nejakých algoritmov a transformuje informácie (stav vesmíru z jedného na druhý). Ako celok teda sústreďuje v sebe informácie a algoritmy, pričom ku každému stavu vesmíru by malo existovať práve jedno pravidlo, ktoré ho transformuje (ak je deterministický, čo sme vyslovili ako predpoklad). Ak by tento program mal byť v nejakom zmysle „dokonalý“, resp. elegantný, potom by ale logickou požiadavkou malo byť to, aby v sebe neobsahoval nič nadbytočné, nič čo by nenašlo vo vesmíre uplatnenie. Ak by napr. vesmír obsahoval algoritmy na výpočet diferenciálnych rovníc, ale súčasne celý vesmír by sa zredukoval na jeden jediný bod, ktorý by bol konštantne nastavený na jednotku, v takom prípade by kód takéhoto vesmíru nebol elegantný. Bola by tu veľká nadbytočnosť v tom, čo je vesmír v rámci svojho stavu schopný reálne dosiahnuť a zároveň aké algoritmy v ňom jestvujú. Dobrý programátor, ktorý by chcel mať svoje dielo nejakým spôsobom elegantné by takto určite nepostupoval. Okrem toho by to bola zbytočná robota. Preto vyslovím ďalší predpoklad, ktorý som si pri vytváraní virtuálneho vesmíru stanovil. Tým predpokladom je, aby ku každému stavu vesmíru existoval práve jeden algoritmus ako ho zmeniť na nový stav a súčasne ku každému algoritmu musí existovať nejaký stav, na ktorý sa tento algoritmus použije (inak by bol zbytočný). Zároveň však vieme, že vesmír neexistuje iba ako celok – ako jeden objekt so svojim stavom, ale v rámci neho existujú aj ďalšie podobjedy – napr. planéty, hviezdy, atómy atď. a tie sa tiež akoby riadia svojimi zákonmi (napr. pohyb planéty po eliptickej dráhe dokážeme pochopiť ako jednoduchšiu časť vesmíru (bez toho aby sme nutne museli vedieť ako sa správa zvyšok vesmíru) a v jej pohybe ľahko objavíme závislosť (ktorá sa dá vyjadriť algoritmom). To znamená, že aj pre každú podmnožinu stavu vesmíru by mala existovať nejaká podmnožina z množiny algoritmov, ktorá ho riadi.

4. Vyšší programátorský princíp – z programátorského hľadiska sú pekné čísla vždy tie, ktoré sú mocninou dvojky, pretože sa s nimi ľahšie (elegantnejšie) narába. (preto sa budeme sústreďovať najmä na takéto čísla) a zároveň predpokladáme, že všetky informácie aj algoritmy máme zapísané v binárnom kóde.

Takže aby som to zhrnul, štyri základné princípy, ktoré som si stanovil sú tieto:

1. Poznanie – vesmír by mal v sebe sústreďovať informácie
2. Determinizmus a reciprocita – všetko so všetkým súvisí a je algoritimizovateľné
3. Bijektivnosť medzi algoritmi a stavmi vesmíru a zároveň aj medzi každou podmnožinou stavu vesmíru a podmnožinou algoritmu.
4. Binárny princíp – všetky informácie aj algoritmy budem mať zakódované binárne a navyše číslu dva priradím špeciálne postavenie (napr. ak budem potrebovať vyjadriť rozmer vesmíru ako číslo malo by byť nejakou mocninou dvojky a pod.)

Zároveň tieto predpoklady som si stanovil ešte pred tým, než som začal čokoľvek programovať (intúcia mi našepkávala, že by to mohlo k „niečomu“ viesť) – nebolo to teda tak, že by som tieto predpoklady prispôbil, aby sedeli s výsledkami počítačových simulácií podobne som pri svojich predpokladoch nevychádzal ani zo žiadnych poznatkov fyziky či biológie, ktoré by som sa tam nejakým spôsobom snažil nasilu „natlačiť“ – všetko bolo založené iba na mojich logických úvahách. Ak by som v týchto štyroch predpokladoch nevidel určitú možnosť, že môžu k „niečomu“ viesť, tak by som určite ani nezačal nič programovať, alebo sa nad tým hlbšie zamýšľať.

## 3 Zostrojovanie virtuálneho vesmíru

V tejto časti popíšem ako som pri zostrojovaní virtuálneho vesmíru postupoval. Pokračujem spôsobom kladenia otázok.

### 3.1 Čo je to algoritmus ?

Algoritmus už bol čiastočne definovaný v úvode. Avšak potom, čo som napísal v predošlých štyroch základných predpokladoch ohľadne determinizmu a recipacity túto definíciu trochu pozmením. Zoberme si klasický algoritmus zapísaný pomocou strojového kódu – ako postupnosť jednotiek a núl. Na to aby bol tento strojový kód zrozumiteľný a vykonateľný, musí existovať procesor, ktorý mu rozumie, (rozumie všetkým typom inštrukcií). A tu je skutočný problém ak by sme chceli vyjadriť niečo ako univerzálny strojový kód vesmíru. Niečo čo by bolo nespochybniteľné a „jediné správne“. Opäť si pomôžem virtuálnym svetom digitálnej evolúcie Tierra. Autor vytvoril inštrukčnú sadu takým spôsobom, aby každá postupnosť jednotiek a núl bola nejaká vykonateľná inštrukcia. Toto je aj podľa mojho názoru nutná podmienka ak by malo niečo ako univerzálne platný kód vesmíru existovať – zatiaľ je táto myšlienka elegantná. Čo je však už menej elegantné je, že autor Thierry vytvoril celkom komplikovanú množinu inštrukcií, ktoré majú často veľmi komplikované správanie aj komplikované operandy. Inak povedané, zdôvodniť prečo by práve takáto množina inštrukcií mala byť „to najlepšie čo môže byť“ by bolo nemožné. Je to opäť založené na skúsenostiach a poznatkoch autora Thierry a niekto iný (ak by sa na to podujal) by to urobil inak. Sám som sa pokúšal o vytvorenie nejakej elegantnej inštrukčnej sady, ktorá by bola povedzme nejakým spôsobom nespochybniteľná, ale na nič som neprišiel. Je to dosť komplikované z dôvodu, že sa vyžaduje zvoliť si určitú veľkosť operandov pre jednu inštrukciu (aj samotnú veľkosť jednej inštrukcie, otázka znie - prečo práve takú ? je napr. 32 bitov to čo potrebujeme ? Alebo skôr 64 ? Atd. ) Takisto by sme mali vysloviť predpoklady o tom, akú veľkosť pamäte by tieto inštrukcie mali byť schopné adresovať a podobne. Jedným veľkým problémom je podľa môjho názoru, aj existencia inštrukcie cyklu (skoku). Vďaka tomu totiž niektoré algoritmy môžu skončiť v nezmyselnej nekonečnej slučke (inak povedané, vesmír by mohol „zamrznúť“). Myslím, že z matematického hľadiska už bolo dokázané, že neexistuje algoritmus, ktorý by o inom algoritme rozhodol, či skončí v konečnom čase alebo nie (ak obsahuje cyklus). Nasadenie komplikovaných procedurálnych algoritmov na riadenie vesmíru by z tohto pohľadu mohlo byť takpovediac „riskantné“.

Preto som sa skúsil pozrieť na algoritmus trochu iným spôsobom. Môj pohľad je neprocedurálny, pozerám sa na algoritmus skôr ako na funkciu (konkrétne boolovskú funkciu). Algoritmus vezme vstupnú postupnosť bitov a k nej vráti výstupnú postupnosť bitov. Takáto definícia algoritmu má rovnakú „silu“ ako keď sa naň pozeráme z hľadiska klasickej definície. V klasickej definícii tiež máme konečnú postupnosť bitov na vstupe a konečnú postupnosť bitov na výstupe. Rozdiel je v tom, že kým pri klasickej algoritme chápeme jeho spustenie ako určitú postupnosť operácií, ktoré majú určitú časovú zložitosť, v mojom ponímaní vieme výsledok algoritmu okamžite, cenou za túto rýchlosť je, že musíme mať všetky kombinácie vstupov a výstupov uložené v pamäti (v podstate akoby predpočítané). Napríklad ak by sme chceli definovať algoritmus, ktorý vezme dve jednobitové čísla a vráti výsledné dvojbítové číslo, musíme to zadefinovať tabuľkou, napr. takto:

operand A	operand B	SUM (A + B)
0	0	00
0	1	01
1	0	01
1	1	10

**Tab. 1**

Ak by sme chceli sčítavať dve 32 bitové čísla a zároveň chceme, aby náš algoritmus bol úplný – zadaný pre všetky kombinácie na vstupe, ľahko si overíme koľko riadkov by musela mať naša tabuľka. Presne  $2^{64}$  riadkov, čo je vskutku ohromné číslo a určite by sa mohli vyskytnúť námietky, či takéto definovanie „algoritmu“ je to čo hľadáme. Ako som vysvetlil vyššie – ak by sme sa snažili zachytiť algoritmus klasicky pomocou inštrukcií, máme veľké problémy v ich zdôvodnení odkiaľ sa vzali, s tým ako funguje inštrukcia skoku a prečo práve takéto inštrukcie atď. Naproti tomu definícia algoritmu pomocou tabuľky ak si odmyslíme pamäťové nároky je veľmi jednoduchá a elegantná. Je to to najjednoduchšie a najelegantnejšie čo si dokážem predstaviť. Minimálne ja sám som na nič lepšie zatiaľ neprišiel - ak by mal existovať nejaký univerzálny strojový kód vesmíru neviem si momentálne predstaviť ako inak by mohol vyzeráť ako práve pomocou tabuľky a neprocedurálne. A čo je dobré si uvedomiť – ak tabuľka obsahuje všetky možné kombinácie vstupných bitov, potom vlastne nemusíme tieto vstupné kombinácie uchovávať. Stačí ak si do tabuľky uložíme iba výstupné bity, t.j. napr. predošlú tabuľku môžeme prepísať aj takto:

SUM (A + B)
00010110

**Tab. 2**

Stačí iba ak vieme, ako namapovať vstupnú postupnosť bitov na výstupnú a tým máme algoritmus „hotový“. Keď toto aplikujem na problém nášho virtuálneho vesmíru - algoritmus, ktorý pretransformuje stav nášho vesmíru z aktuálneho na nový môžeme definovať ako množinu:

$$ALG = \{ nInputBits, nOutputBits, Seq \}$$

kde:

- *nInputBits* predstavuje počet bitov na vstupe
- *nOutputBits* predstavuje počet bitov na výstupe
- *Seq* predstavuje sekvenciu (postupnosť) bitov (podobne ako v tabuľke 2). Zároveň ak vesmír počas svojej existencie prejde *nStates* stavmi a na reprezentáciu každého stavu potrebujeme *nStateSize* bitov, potom na zachytenie transformácie stavu vesmíru pre každé časové kvantum by nám malo stačiť (toto uvádzam ako príklad, v tomto momente som ešte stále nepovedal nič o tom ako vlastne algoritmus transformácie na nový stav získame):

$$sizeof(Seq) = (nStates) * (nStateSize) [bits]$$

Pre algoritmus ako bol zadaný vyššie a pre všeobecný prípad (keď všetky vstupy algoritmu sú zadané a teda nepotrebujeme ich uchovávať v tabuľke a stačia nám iba

výstupne postupnosti) ľahko určíme počet potrebných bitov, ktoré potrebujeme na uloženie algoritmu ako:

$$\text{sizeof(Seq)} = n \lg \text{Bits} = 2^{n\text{InputBits}} * n\text{OutputBits}$$

Ak tieto dve rovnice dáme do rovnosti dostaneme:

$$n\text{States} * n\text{StateSize} = 2^{n\text{InputBits}} * n\text{OutputBits}$$

Neskôr sa k tejto jednoduchšej rovnici vrátim. Pred tým sa ale pokúsim vyjasniť akým spôsobom budem vo svojom virtuálnom vesmíre reprezentovať priestor a čas.

### **3.2 Čo je to priestor, čo je to čas ?**

Predstavme si že vesmír je konečný a v každom čase existuje niečo ako stav vesmíru. Z dôvodu konečnosti vesmíru (konečnosti informácie ktorú v sebe vesmír nesie) by malo vyplývať, že jeho stav je digitalizovateľný a teda, že každý stav možno reprezentovať ako konečnú postupnosť bitov. Minimálne pre náš virtuálny vesmír toto určite bude platiť. Jeden bit nazveme atómom vesmíru, nič jednoduchšie z hľadiska jeho binárnej reprezentácie totiž nejstuje. A teraz urobme malú úvahu ohľadne počtu rozmerov a času. Čo bude v našom vesmíre čas ? A ako chápeme priestor a čas v našom reálnom vesmíre ? Vie to niekto exaktne definovať ? Existoval čas a priestor aj pred veľkým treskom ? A čo rozpínanie vesmíru ? Znamená to, že „veľkosť priestoru“ sa zväčšuje ? Zväčšuje v zmysle ak by vesmír bol konečný, znamená jeho rozpínanie, že sa zväčšuje aj množstvo bitov, ktoré potrebujeme na jeho reprezentáciu v nasledujúcom stave ? Podľa vysvetlení, ktoré podáva moderná fyzika sa vesmírny priestor (priestoročas) skutočne rozťahuje. Z tohto mi ale stále nie je príliš jasné či sa mení iba mierka, alebo reálne stúpa počet bitov v jednom stave. a zároveň uvádza, že objekty ktoré na seba pôsobia napr. gravitáciou sa toto rozťahovanie priestoru nevzťahuje, pretože zákony fyziky sú nezávislé od zvoleného metrického systému priestoročasu (tak nejak som to aspoň pochopil). Pre mňa ako „nefyzika“ je podobné zdôvodnenie síce dosť ťažko pochopiteľné (prečo sa napr. potom galaxie od seba vzdľahujú ak na seba tiež pôsobia gravitáciou ? Dobrovoľne sa priznávam, že v tejto veci vôbec nemám jasno a asi je to spôsobené mojimi nedostatočnými poznatkami z oblasti fyziky). Podľa súčasných poznatkov fyziky existuje substancia nazývaná „temná hmota“, ktorú sme zatiaľ schopní evidovať iba na základe jej gravitačných účinkov a podľa odhadov tvorí okolo 25 % hmoty vo vesmíre (klasická viditeľná hmota iba okolo 4%). Ďalších 75 % tvorí tzv. „temná energia“, ktorú takisto nemôžeme vidieť žiadnymi meraniami v oblasti elektromagnetického spektra. Ak teda existuje takáto forma energie, ktorú nevidíme iba vieme že existuje v dôsledku gravitačných účinkov, pokojne si viem predstaviť energiu prípadne hmotu ktorú nevidíme ani o jej existencii zatiaľ nevieme (napr. ak by existovalo niečo ako záporná gravitácia, ktorou by sa jej gravitačné účinky rušili – toto je samozrejme veľmi špekulatívne) Celý priestor by napr. mohol byť vyplnený „nejakou“ hmotou a čas by mohol jestvovať večne, ale v dôsledku determinizmu a jeho konečnosti by po konečnom počte krokov musel dôjsť do niektorého stavu dvakrát (ideálne začiatočného bodu), čím by sa vlastne stal periodickým. Okolité priestor by teda mohol jestvovať „večne“ a veľký tresk by bol iba vyvrhnutie nami viditeľnej hmoty do existujúceho priestoru. Napr. by takáto hmota vznikala interakciou s okolitými inak neviditeľnými časticami . Z toho dôvodu by sa galaxie od seba vzdľahovali v priestore, ktorý nie je prázdny, ale je v každom svojom bode vyplnený nejakou časticou. Je mi jasné že takýto

prístup k veľkému tresku a štruktúre priestoru prípadne času je už naozaj krajne špekulatívny a vzhľadom na moje veľmi obmedzené znalosti fyziky bude asi lepšia ak tu prestanem. (na druhej strane fyzika myslím doteraz žiadnym spôsobom nevysvetlila, prečo veľký tresk vznikol, iba popisuje jednotlivé javy ktoré v ňom prebehli v pokiaľ možno čo najkratšom časovom intervale – tesne po veľkom tresku).

Vo svojom virtuálnom vesmíre som nezahrnul žiadne predpoklady o tom, že by sa priestor mal zväčšovať. Stav vesmíru v každom čase reprezentujem konštantne veľkou postupnosťou bitov. Ak by sa vesmír v čase skutočne zväčšoval možno by bolo dobré aby som túto svoju reprezentáciu pozmenil. Na druhej strane ak je počet bitov pre jeden stav zvolený tak, že aj okamih v ktorom mal vesmír najväčšie rozmery sa doň „zmestí“, potom ľubovoľný „menší“ priestor s menším počtom atómov sa doňho tiež zmestí, tým pádom by záviselo od toho ako si daný stav interpretujem. Jednou zo zaujímavých vecí pri vytváraní virtuálneho vesmíru je, že nič také ako priestor nepotrebujem vytvárať. Z môjho programátorského hľadiska sa priestor javí iba ako ilúzia spôsobená konkrétnou interpretáciou lineárneho vektora bitov. Vzniknutý priestor môže byť vlastne ľubovoľných rozmerov (vysvetlím ďalej), teraz sa pokúsim dať do súvisu priestor s časom v mojom virtuálnom vesmíre. Povedzme, že chceme vesmír s dvojrozmerným priestorom, ktorý sa bude v jednotlivých simulačných krokoch meniť a túto zmenu nazveme časom. Ako niečo takéto reprezentovať v počítači? Z programátorského hľadiska je to podobné ako keby sme si pustili na počítači video, to je dvojrozmerné a mení sa v čase a to ako sa mení máme uložené v pamäti – najjednoduchší spôsob je pomocou trojrozmerného poľa napr.:

```
Bit bitArray[ VIDEO_SIZE_X][VIDEO_SIZE_Y][N_FRAMES];
```

Za predpokladu, že video je čiernobiele (na každý pixel videa nám stačí jeden bit) spolu potrebujeme

```
nBits = (VIDEO_SIZE_X * VIDEO_SIZE_Y * N_FRAMES);
```

Ako je vidieť, jedná sa o trojrozmerné pole, ak by napr. bola šírka videa zhodná s výškou a navyše by aj počet frameov (obrázkov videa) bol rovnaký (pre jednoduchosť predpokladajme že sú), potom by bol potrebný počet bitov vlastne treťou mocninou:

$nBits = SIZE^3$ . Takéto trojrozmerné pole je analógiou trojrozmernému priestoru, pre každú súradnicu v trojrozmernom priestore (x, y, z) máme priradenú hodnotu atómu buď jednotku alebo nulu (true - false, biela - čierna). Ak si predstavíme, že treťou súradnicou je čas, teda hodnotu pixla pre naše video v danom čase dostaneme jednoducho ako funkciu (x, y, t), ktorá vráti hodnotu atómu na tejto pozícii. V bežných programovacích jazykoch ku prvku takéhoto poľa pristupujeme pomocou trojice indexov napr.:

```
atomAtXYZ = bitArray[x][y][z];
```

Fyzicky v počítači je takéto pole uchovávané ako lineárna postupnosť bitov - t.j. ako jednorozmerné pole, ktorého jednotlivé bity nasledujú za sebou a pole ako celok má veľkosť  $SIZE^3$  bitov, pričom prekladač pre nás vytvorí ilúziu, že narábame s trojrozmerným poľom. Trojica indexov x, y, z sa musí preložiť na nejaký offset vzhľadom k začiatku tohto jednorozmerného poľa obyčajne napr. takto:

```
atomAtXYZ = bitArray [x][y][z] = bitArray [ SIZE2 * x +SIZE * y + z ]
```

Toto však nie je jediný možný spôsob prekladu, existuje ich oveľa viac jedinou podmienkou je, aby medzi trojicou (x, y, z) a offsetom existovala bijekcia a po zachovaní tejto podmienky máme zachovanú ilúziu dokonalej trojrozmernosti jednorozmerného poľa. Toto vyzerá veľmi jednoducho a zrozumiteľne, teraz skúsme dosadiť za SIZE konkrétne číslo – napr.

$$SIZE = 16 = 2^4 \Rightarrow nBits = (2^4)^3 = 2^{12} = 4096 [bits]$$

Koľko rozmerne pole sme dostali ? Nezabúdajme, že počet rozmerov je iba ilúzia, ktorú pre nás vytvoril prekladač, fyzicky vždy pracujem s jednorozmerným poľom, preto si položme otázku či okrem trojrozmernej a jednorozmernej interpretácie existuje aj nejaký iný spôsob ako sa na takéto pole pozerať. Po kratšej úvahe zbadáme že existuje a je ich hneď viacero napr.:

```
Bit bitArray[4][1024];  
Bit bitArray[8][2][4][64];  
Bit bitArray[8][8][8][8];  
Bit bitArray[64][64];
```

Je ich samozrejme oveľa viac, najmä posledné dve sú však zaujímavé – ukazujú nám totiž, že jeden priestor typu  $SIZE^3$  môžeme zameniť za priestor rozmerov  $SIZE^2$  alebo aj  $SIZE^4$ . Inak povedané, naše video by sme si mohli predstaviť nielen uložené v trojrozmernom priestore, ale napr. aj v dvojrozmernom štvorci dostatočne veľkých rozmerov, alebo v štvorrozmernej kocke atď. Je naozaj ťažké rozhodnúť koľko rozmerne pole o 4096 bitoch objektívne je. Priestor je iba ilúzia a ak mám nejaký jednorozmerný reťazec 4096 bitov, potom si túto postupnosť môžem predstaviť v trojrozmernej kocke ako nemennú štruktúru bitov (pre trojrozmernú kocku nič také ako čas nejestvuje, je to iba ďalšia súradnica priestoru, kocka je statická a vyplnená bielymi alebo čiernymi malými kockami podľa toho, či sa na danej trojici súradníc nachádza jednotka alebo nula.) alebo aj ako rozpochybované video v dvojrozmernom priestore, prípadne jednorozmerný priestor, v ktorom sa zvyšné dve súradnice zamenia za čas.

Skúsme teraz vyjadriť koľko atómov sa nachádza v našom vesmíre, presnejšie povedané koľko bitov potrebujeme na uloženie stavu vesmíru (ak predpokladáme, že čas je iba jednou zložkou priestoru). Z dôvodu, že uchováваме hodnoty bitov pre všetky možné trojice súradníc, nie je potrebné zahrnúť ich do dátovej časti – opäť stačí ak si predstavíme existenciu nejakého mapovania, ak vieme pre každý bit ku ktorej trojici súradníc patrí.

```
nDataBits = nSpaceSizenDimensions * nAtomBits
```

Keďže každému bodu v priestoročase priradíme práve jeden atóm a ten má hodnotu 1 / 0, môžeme za veľkosť atómu (počet bitov atómu) dosadiť jednotku. Skúsme si to porovnať s rovnicou pre veľkosť algoritmu, ktorá bola uvedená vyššie:

$$nA \lg \text{Bits} = 2^{n\text{InputBits}} * n\text{OutputBits}$$

$$n\text{DataBits} = n\text{SpaceSize}^{n\text{Dimensions}} * n\text{AtomBits}$$

Ak predpokladáme, že Veľkosť priestoru je nejakou mocninou dvojky a dáme tieto dve rovnice do rovnosti, potom dostaneme:

$$2^{n\text{InputBits}} * n\text{OutputBits} = 2^{(n\text{SpaceSize Bits} * n\text{Dimension s})} * n\text{AtomBits}$$

Tu sa zjavne črtá analógia, akoby algoritmus aj dáta v ňom boli vlastne jedno a to isté. Túto analógiu som aj využil vo svojom programe. Algoritmus, ktorý by vesmír riadil v podstate neexistuje. Na všetko sa pozerám iba z hľadiska trojrozmernej kocky, ktorá je nemenná. Z tohto pohľadu je mojím algoritmom trojica vstupov (x, y, t), ktorá pre ľubovoľnú trojicu vráti jednobitovú hodnotu 1 / 0, čiže atómu na daných súradniciach. Jeden stav vesmíru (premietnutý do dvojrozmernej plochy) je potom množina takých trojíc { (x1, y1, t1), (x2, y2, t2), ... (xn, yn, tn) } ktoré majú rovnakú časovú súradnicu (t1 = t2 = ... tn) a spolu zaplnia celú dvojrozmernú plochu. Existuje aj viacero iných možností ako sa pozrieť na to čo bude pre mňa algoritmom, tie ostatné som však zatiaľ neodsimuloval. Okrem toho je dobré si všimnúť, že pri takomto spôsobe chápania algoritmu máme zabezpečenú podmienku bijekcie medzi algoritmom a stavom vesmíru a zároveň reciprocitu. Pri podmienke reciprocitu to však nie je až také isté, jeden stav vesmíru vyžaduje síce ako vstup všetky kombinácie x, y, ale možno by sa to dalo urobiť aj inak (tomuto problému sa ale viac nebudem venovať). Toto je opäť veľmi jednoduché, už stačí iba „maličkosť“ – ako vyplniť jednotlivé body priestoru tými „správnymi“ hodnotami a nájdenie toho správneho mapovania (x, y, t) na ten správny bod kocky (tým, že kocka v sebe obsahuje všetky kombinácie (x, y, t) a my si ako súčasť stavu vesmíru neuchováваме informáciu, ktorý vektor (x, y, t) sa viaže na ktorý bod kocky, je potrebné nájsť nejaké implicitné mapovanie, ktoré by bolo určitým spôsobom „logickejšie“, než ostatné. Pred tým sa však pokúsme zamyslieť ako vyplniť jednotlivé body kocky informáciou.

### 3.3 Čo je to poznanie a informácia ?

Stačí postupnosť jednotiek a núl, aby sme to mohli nazvať informáciou ? Pravdepodobne áno. Ako som však už v úvode naznačil – informácia bez algoritmu nemá zmysel. Napr. Shakespeare sám by nerozumel vlastným dielam, ak by mu ich niekto dal prečítať v slovenskom preklade. To čo informáciu robí pre človeka užitočnou nie je konkrétna postupnosť jednotiek a núl (jazyk atď), ale spôsob akým ich interpretuje. Informáciu si nedokážem predstaviť oddelene bez algoritmu a a ich spoločné koexistovanie by som nazval myslením – toto je veľmi voľná definícia, ale nič lepšie ma nenapadá. Napr. Shakespeare do svojich diel vložil určité myšlienky, ktoré zhodou okolností vyjadril v jazyku a v písanej forme. Na začiatku to bola myšlienka, ktorá sa pretransformovala na informáciu. Text vieme ľahko digitalizovať, takže v podstate Shakespeare vytvoril postupnosť jednotiek a núl. Z takéhoto pohľadu by sme mohli povedať, že vlastne Shakespeare napísal nejakú informáciu. Avšak tvrdiť, že Shakespeare napísal len informáciu a nič viac je trochu nespravodlivé. Ak čítame text a rozumieme jeho jazyku, sme schopní pochopiť aj myšlienkam, ktoré do svojich diel Shakespeare vložil (dokonca aj to ako by jedno jeho dielo chápali dvaja rôzni ľudia bude pravdepodobne v čomsi iné). Skúsme si predstaviť desiatky rôznych prekladov Shakespeareových diel do rôznych jazykov, pričom všetky by niesli zhruba rovnakú myšlienku

(ak rozumieme tomu konkrétnemu jazyku) ale boli by všetky reprezentované úplne odlišnými postupnosťami bitov. Inak povedané - jedná a tá istá myšlienka môže byť vyjadrená úplne rozličným spôsobom a záleží na čitateľovi ako si túto informáciu interpretuje. A interpretácia závisí od toho, aký algoritmus použijem na dekodovanie informácie (ak sa pozerám na vesmír ako na deterministický a riadený algoritmom). Z hľadiska nášho vesmíru, ak v ňom chceme sústrediť niečo ako poznanie preto nie je dôležité aby sme v ňom sústredovali niektoré konkrétne bitové reťazce, ale skôr „možnosť vzniku myšlienok“. Dalo by sa to odôvodniť asi takto: čím viac „myšlienok“ v sebe vesmír obsahuje, tým viac obsahuje zmysluplných informácií aj algoritmov (keďže myšlienky informácie aj algoritmy považujem za tesne späté) Skúsme sa pozrieť na takýto binárny reťazec:

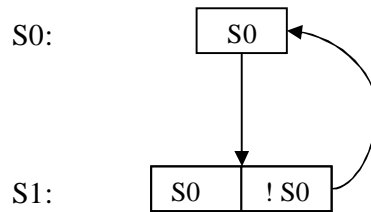
1111111111

Koľko informácie v sebe nesie ? Podľa bežných vzorcov na výpočet entropie je pravdepodobnosť výskytu jednotky 100% a preto entropia bude rovná nule. A čo ak by sme si zobrali postupnosť celej stovky samých jednotiek ? Zmení sa tým niečo ? Ak by sme brali do úvahy entropiu tak oba reťazce nesú rovnakú (nulovú) informáciu, inak povedané sú dokonale usporiadané, bez najmenej známky chaosu. V skutočnosti ak napíšem 100 jednotiek za sebou a iba desať jednotiek niečo sa predsa len zmení – je to práve ich počet. Aj ten vyjadruje informáciu. Ak vopred neviem akú veľkosť reťazec bude mať, tak na zachytenie informácie o reťazci dĺžky  $N$  jednotiek potrebujem  $\log_2 N$  bitov. Jedná sa v podstate o akúsi jednotkovú číselnú sústavu. Pre potreby virtuálneho vesmíru budem preto merať množstvo informácií, ktoré určitý reťazec obsahuje v počte možných zobrazení jeho podpostupností na množinu prirodzených čísel. Napr. ak napíšem postupnosť desať jednotiek za sebou, potom dokážem objaviť práve desať podpostupností, ktoré sú navzájom od seba rôzne a možno ich mapovať na nejaké číslo z množiny prirodzených čísel.

(1), (11), (111) ... (111111111)

Ak teda existuje práve 10 podpostupností, môžem ich namapovať napr. na prvých 10 prirodzených čísel. Čím viac takýchto mapovaní dokážem urobiť, tým viac informácie reťazec nesie. Ak by sme si zobrali napr. CD – ľubovoľná informácia na ňom môže byť vyjadrená jedným veľkým prirodzeným číslom, samotné CD je schopné niesť približne  $2^{8 \cdot 700 \cdot 1000000}$  takýchto mapovaní na prirodzené čísla, preto množstvo informácií, ktoré je schopné v sebe zahrnúť je naozaj obrovské, ale pokiaľ by sme zostrojili reťazec pozostávajúci zo samých jednotiek a bol by dĺžky  $2^{8 \cdot 700 \cdot 1000000}$  vyjadrili sme úplne rovnaké množstvo informácie (dokonca možno ešte väčšie) na aké nám stačí  $8 \cdot 700 \cdot 100000$  bitov (ľubovoľná postupnosť bitov akú možno uložiť na CD z hľadiska mapovania na množinu prirodzených čísel určite nebude väčšie číslo ako práve najväčšie možné, ktoré je pomocou takéhoto počtu bitov vyjadriť). Skúsme sa zamyslieť čo sa zmení ak ku jednotke pridáme aj nejakú podľa možnosti čo najjednoduchšiu operáciu. Najjednoduchšie čo ma napadá je negácia. Operácie, ktoré budeme vykonávať môžem zachytiť pomocou nasledujúceho diagramu (výkričníkom označíme negáciu):





**Schéma 1**

Jedná sa o rekurzívnu procedúru, kedy výstup v kroku  $i$ , je vstupom v kroku  $i + 1$  (akási spätná väzba). Ak v prvom kroku inicializujeme  $S_0$  na nulu dostávame po štvrtej iterácii takúto postupnosť:

$S = 01101001\ 10010110$

Je táto postupnosť niečím zaujímavá? Pri pozornejšom skúmaní môžeme objaviť, že takýmto jednoduchým pravidlom sme dosiahli, že výstupná postupnosť na seba dokáže namapovať všetky možné štvorbitové postupnosti. Zoberme si napríklad postupnosť  $T = 0100$  a pokúsme sa ju namapovať. Budeme postupovať binárnym delením sekvencie  $S$ . Prvé číslo z postupnosti  $T$  je nula a teda pôjdeme do ľavej polovice postupnosti  $S$ . Druhé číslo je jednotka, takže vo zvyšnej polovici pôjdeme doprava atď. Situáciu zachytáva nasledujúca tabuľka:

N	N-tý bit z T	Binárny podreťazec získaný rozpolcovaním S
1	0	01101001
2	1	1001
3	0	10
4	0	1

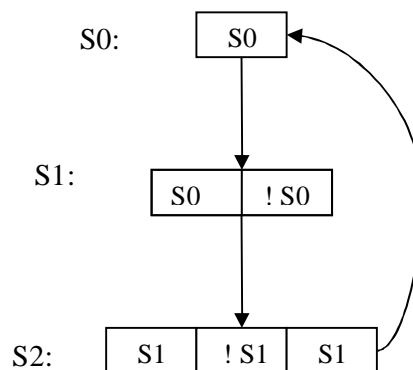
**Tab. 3**

Ak by sme poukladali čísla z tretieho stĺpca za sebou dostaneme: 01101001 1001 10 1. Takto získaná postupnosť bitov je pre každú štvorbitovú číselnú postupnosť jedinečná. To ale vlastne znamená, že pomocou tohto jednoduchého pravidla sme namapovali všetkých 16 štvorbitových postupností do jedného 16 bitového reťazca. Ak by sme také niečo chceli urobiť „klasicky“, potom by sme potrebovali  $16 * 4 = 64$  bitov. Tento rozdiel sa stáva ešte viac očividným ak ho prevedieme na mierku CD. Nech CD má  $K$  bitov ( $K \approx 700 * 8 * 1000000$ ). Potom na zachytenie všetkých možných stavov CD by nám pomocou tohto jednoduchého algoritmu stačilo  $2^K$  bitov (ak by sme chceli uložiť „klasicky“ všetky binárne reťazce aké sa na CD môžu vyskytnúť tak by sme potrebovali  $K * 2^K$  bitov, čo je podstatne viac, hoci počet mapovaní na množinu prirodzených čísel by bol zhodný s tým úspornejším zápisom. Zároveň je zaujímavé, že takto vygenerovaný reťazec 16tich (prípadne  $K$  bitov) bitov v sebe obsahuje nielen všetky bijekcie na všetky 4 bitové čísla, ale aj 3, 2, 1 aj „nula“ bitové. Tým sme oproti zápisu 16tich jednotiek získali vyššiu „hustotu“ informácií. Pomocou 16tich jednotiek je možné zachytiť 16 rôznych stavov (16

mapovaní na množinu prirodzených čísel), kým pomocou takéhoto zápisu dokážeme zachytiť až  $2^4 + 2^3 + 2^2 + 2^1 + 2^0 = 31$  rôznych stavov (mapovaní). Pravdepodobne by bolo možné dokázať, že žiadna 16bitová postupnosť neobsahuje v sebe viac zobrazení a preto podľa toho ako som zdefinoval množstvo informácie, pre naše potreby sa hodí najlepšie.

A teraz sa zamyslime – ak existuje nejaká konečná množina ľudského poznania, ktorú počas svojej existencie ľudstvo nadobudlo a táto množina by bola vyjadrená v bitoch, potom existuje vyššie zmieňovaný doslova triviálny algoritmus, ktorý by každú podpostupnosť z tejto globálnej masy informácií dokázal na seba namapovať. Nie je to fascinujúce ? Ak by napríklad vesmír takto naozaj fungoval, potom prakticky môžeme povedať, že to kam sa až môžeme počas trvania evolúcie dopracovať je limitované poznaním, ktoré bolo vložené do vesmíru pri jeho vzniku – žiadna väčšia informácia ako vesmír sám v našom vnímaní reality neexistuje. Inak povedané Shakespeare nevymyslel Hamleta, on ho iba objavil, podobne ako sochár nevyrába materiál z ktorého sochu vyrobí ale skôr iba objaví tvar ukrytý v mramorovom kvádri.

Neskôr som začal experimentovať aj so zložitejšími generátormi z čoho ma zaujal najmä nasledujúci:



**Schéma 2**

Naprogramovať niečo takéto je extrémne jednoduché, na ilustráciu sem teraz dávam výstup z takéhoto generátora po troch iteráciách:

011 001 100 110 100 110 011 001
011 001 100 110 100 110 011 001
011 001 100 110 100 110 011 001
100 110 011 001 011 001 100 110
100 110 011 001 011 001 100 110
100 110 011 001 011 001 100 110
011 001 100 110 100 110 011 001
011 001 100 110 100 110 011 001
011 001 100 110 100 110 011 001

**Tab. 4**

Pridaním ďalšieho bloku sme získali zvýšenie zložitosti výstupnej postupnosti bitov. Ako ale túto postupnosť interpretovať? Kým v minulom prípade sa jednalo o delenie jednorozmerného priestoru na polovicu a nachádzanie zobrazení na množinu prirodzených čísel (existovali iba dva základné atómy 0, 1) teraz po pozornejšom skúmaní môžeme objaviť štyri základné stavebné prvky:

011, 001, 100, 110

Je to akoby delenie štvorcovej plochy na menšie štvorce, mohli by sme ich namapovať napr. spôsobom ako je naznačené v nasledujúcej tabuľke, pričom každým rozdelením väčšieho štvorca získame štvorec s polovičnou hranou (a teda so štvrtinovým počtom atómov)

011	001
100	110

**Tab.5**

Ak sa však pozornejšie zahľadíme určite objavíme jednu zaujímavú vlastnosť, atómy (011, 001) a (100, 110) sa vyskytujú vždy vo dvojici a sú navzájom komplementárne (ak by sme ich preklopili okolo osi tak jednotky a nuly do seba presne zapadnú). Toto sa náramne podobá na bázy DNA, adenozín, cytozín, tymín a guanín. Je to iba náhoda? Potom ako som si to dal dokopy s podmienkou na bijektivnosť informácií a algoritmov ma napadlo, že DNA sama o sebe predstavuje skvelý nástroj na zabezpečenie bijekcie z dôvodu, že ku každej báze existuje jej práve jeden komplementárny element (zaujímavé, že mi to pred tým uniklo). Na základe môjho pohľadu by som potom dokonca mohol tvrdiť, že DNA je niečo ako algoritmus aj informácia súčasne a zároveň a môžeme si ju predstaviť ako strom deliaci v každom svojom kroku plochu na štyri časti. (toto je z mojej strany iba čistá špekulácia). Mimochodom podmienku na bijektivnosť som formuloval dávno pred tým než som si takto náhodou uvedomil, že DNA je vlastne veľmi vhodným prostriedkom na jej dosiahnutie (pri tvorbe virtuálneho vesmíru ma vôbec nenapadlo rozmýšľať nad niečím ako je biológia). A zároveň samotná DNA sa javí ako niečo dosť podobné, k čomu som dospel pri pokuse nájsť „čosi“ ako univerzálny strojový kód vesmíru (ako som písal, že by mal byť neprocedurálny, spájať v sebe informácie aj algoritmy a vytvárať medzi nimi bijekciu a byť definovaný tabuľkou) – pritom som k tomu dospel úplne odlišnými postupmi ako kritériami biológie – možno je táto podobnosť iba náhoda a možno nie, kto vie?

Spôsob akým som vytvoril svoj virtuálny vesmír však nevychádza z bijektivnosti medzi štyrmi základnými atómami (podmienku bijektivnosti som zabezpečil iným spôsobom), táto myšlienka by však mala byť tiež určite vyskúšaná – považujem ju za celkom perspektívnu a logickú. Mimochodom - aké množstvo informácie uchováva takto vygenerovaný reťazec z tabuľky 4? Toto som bližšie neskúmal, spoľahol som sa iba na to, že

aj voľným okom je viditeľné, že obsahuje jednoducho „veľa“ mapovaní do množiny prirodzených čísel a takisto na to, že je tam určitá analógia s DNA. Postupnosť takto vygenerovaných bitov som preto použil ako zdrojový kód svojho vesmíru - počet iterácii som zvýšil na 9. Zaplnil som ním jednotlivé atómy, potom už stačilo iba namapovať jednotlivé bity jednorozmerného a nemenného vesmíru na súradnice (x,y,t), čím ho bolo možné rozpochybovať. Môj vesmír je teda lineárny (jednorozmerný) vektor takto vygenerovaných postupností bitov. Zároveň som takto jednoducho a elegantne splnil všetky štyri základné podmienky, ktoré som si stanovil (sústredenie poznania, reciprocita, determinizmus a algoritmizovateľnosť, bijekcia medzi algoritmami a informáciami, binárnosť).

Teraz sa ešte skúsme zamyslieť prečo by rekúzia (rekurzívne definovanie poznania) mala byť to „pravé“ čo hľadáme. Na jednej strane som čiastočne naznačil (toto by ešte malo byť podrobne preskúmané), že práve použité generátory postupností dávajú v určitom zmysle najviac mapovaní do množiny prirodzených čísel. Skúsme sa na to pozrieť aj z druhej strany – je rekúzia to, čo bežne vidíme v prírode? Rekúziu si pokojne môžeme zameniť aj slovom sebedobnosť. Ako vieme v prírode existuje veľa príkladov sebedobnosti, typickým príkladom sú napr. rastliny a stromy (kmeň stromu sa rozvetvuje a to sa zase podobá na konár ktorý sa rozvetvuje atď.) Ale to nie je všetko. Okrem toho sú sebedobné napr. aj atómy, ktoré majú svoje jadrá, okolo ktorých obiehajú elektróny, atómy tvoria zložitejšie štruktúry ako napr. živé bunky. Živé bunky opäť majú svoje jadrá a tvoria zložitejšie organizmy. Ďalej existujú planéty ktoré krúžia okolo Slnka, a Slnko ktoré obieha okolo stredu galaxie a samotné galaxie sa tiež pohybujú. Ak sa na to pozrieme opäť z hľadiska, programátorského a teda, že vesmír je riadený algoritmom, potom aj algoritmy sú si sebedobné, nielen štruktúra objektov ktoré riadia. Napr. pohyb elektrónu okolo jadra atómu je podobný pohybu planéty okolo Slnka v tom, že elektrón neskolabuje do jadra, ani nevyletí mimo jeho dosahu (podobne ako planéta sa drží v dosahu Slnka), na druhej strane ale jeho pohyb zďaleka nepripomína usporiadaný pohyb planéty, ale skôr sa riadi kvantovomechanickým princípom neurčitosti (t.j. všetko čo je sebedobné je zároveň aj určitým spôsobom odlišné, akoby sa napr. jednalo o inú hĺbku rekúzie, ktorá vždy spôsobí vykonanie nejakého iného variantu). Je zaujímavé, že ak chceme dosiahnuť bijekciu medzi algoritmami a informáciami a zároveň sebedobnosť, rekúzia je asi jediným spôsobom ako to docieľiť. Rekúzia naozaj môže fungovať aj na algoritmus aj na štruktúru objektov súčasne. Existujú napr. paradigmy programovacích jazykov (jeden z mojich obľúbených predmetov bolo aj Funkcionálne a logické programovanie, jazyky Prolog, Lisp atď.), ktoré odmietajú cyklus a namiesto toho používajú rekúziu. Vyjadrovacia sila týchto jazykov je pritom rovnaká ako v jazykoch ktoré cyklus používajú – avšak existujú určité praktické dôvody, prečo sa na väčšinu problémov používajú práve procedurálne jazyky a nie jazyky bez podpory cyklu (oni ho síce umožňujú tiež používať, ale potom to už nie je „čisté“ funkcionálne programovanie). Za stanovenia mojich východiskových štyroch podmienok som však neobjavil žiadnu inú možnosť ako by vesmírny kód (v prípade, že je to naozaj „len“ nejaký druh veľmi sofistikovaného programu) mohol fungovať. Riešenie na ktoré som prišiel sa mi zdá byť vcelku logicky odôvodnené, vysvetlil som, prečo by vesmír mal byť deterministický, aké zákony by ho mali riadiť, prečo by mal v sebe sústreďovať poznanie, ako ho tam vložiť, aký by mal byť začiatkový stav vesmíru - stačí jediný bit 0 alebo 1 atď. V skutočnosti je však 1 alebo 0 iba pomôcka pre našu predstavivosť, správne je definovať jednotku ako nejaké slovo (postupnosť) na symbolickej úrovni označme ju S0 a nula bude reprezentovaná ako uplatnenie jedinej a zároveň najjednoduchšej operácie, ktorú sme nad týmto symbolom zadefinovali – čiže negácie:

„1“ = (S0)

„0“ = (! S0)

A postupnosti budú vlastne zložené z týchto postupnosti. Každá postupnosť je pritom sebedobná sama na seba a teda ak vytvoríme ďalšie postupnosti zložené z týchto počiatočných, v určitom okamihu dôjde k dokonalej sebedobnosti medzi začiatkovou sekvenciou a tou, ktorá z nej bola vytvorená – čiže niečo ako samoimplikácia existencie vesmíru. Stručne povedané – tým, že existuje „niečo“, môže existovať aj takáto rekurzívna postupnosť. A z faktu, že ak takáto postupnosť naozaj existuje spätne vyplýva, že existencia (existencia niečoho – napr. vesmíru) môže existovať (to je veľmi stručne povedané). Podľa takéhoto prístupu by sme objektívnou realitou nazvali práve jednorozmernú postupnosť jednotiek a núl, ktorá je nemenná a našou realitou potom spôsob interpretácie takejto postupnosti. Ako bolo povedané nikde v tejto postupnosti bitov nemáme explicitne zapísané, ktorým spôsobom ju interpretovať. A záleží teda na nás ako ju vnímame. Je pravdepodobné, že ak by sme chceli vytvoriť trojrozmerný vesmír existuje nejaké podobné pravidlo, ktoré vygeneruje osem základných atómov, čím by vlastne jeden atóm vyjadroval delenie priestoru na osem menších kociek (podobne ako v prípade štvorca to bolo delenie na štyri menšie plochy). Stav vesmíru v čase si potom môžeme predstaviť ako rozvetvený strom (napr. binárny, kvartálny, oktálny a pod. – závisí od počtu dimenzií), ktorý zo svojich vstupov presne vymedzí vetvu ako sa v priestore zo stredu nemennej objektívne jestvujúcej kocky uberať až kým dôjde ku konkrétnemu atómu, ktorý bude mať hodnotu 0 alebo 1 (táto hodnota sa mu priradí pomocou jednoduchého algoritmu „generátora poznania“ ako bolo uvedené vyššie).

Takúto stromovitou štruktúrou som aj ja uplatnil na vizualizáciu trojrozmerného priestoročasu do dvojrozmernej plochy. Podobná stromovitá štruktúra sa zdá byť z programátorského hľadiska elegantnejšia, ako keby sme napr. zobrali iba postupne všetky bity postupnosti sekvenčne za sebou. Určitý princíp ktorý som zatiaľ nezdôraznil ale považujem ho za veľmi dôležitý je symetria. Ak má byť vesmír určitým spôsobom dokonalý, potom by mal byť symetrický, napr. vyzerá byť logickejšie, aby vesmír svoj aktuálny stav definoval zo svojho stredu (napr. náš reálny štvorrozmerný priestor je podľa tohto prístupu ak sa zbavíme ilúzie času iba štvorrozmerná kocka vyplnená nulami a jednotkami a vieme si predstaviť kde leží jej stred) na všetky strany pomocou rozvetveného stromu, ako keby bol zadefinovaný niekde z rohu kocky ako postupnosť bitov idúcich za sebou (v praxi som aj overil, že tento druhý prístup naozaj nevedie k niečomu zvlášť zmysluplnému). Tu sa žiada zdôrazniť, že pri takto zadefinovanom vesmíre je hlavným problémom pamäť potrebná na rekuziu. Čím väčší vesmír (väčšie rozmery) zvolím, tým viac poznania v sebe bude obsahovať a tým viac pamäte potrebujem mať k dispozícii. Na jednej strane je to dobré, pretože čím väčší vesmír, tým komplikovanejšie štruktúry môžem očakávať, že v ňom spontánne vyemergujú. Na druhej strane ale nároky na pamäť rastú exponenciálne (súčasne s nimi ale exponenciálne rastú aj rozmery vesmíru) a preto som zatiaľ nebol schopný odsimulovať dvojrozmerný vesmír, ktorý by bol väčší ako výrazne väčší ako 32x32 štvorcov (takáto rekuzia zároveň spôsobuje, že všetko so všetkým súvisí).

Myslím si, že s týmto problémom je možné sa vysporiadať a v budúcnosti by sa mohlo podariť spustiť aj simuláciu o väčšom počte atómov bez takej vysokej pamäťovej náročnosti (neuchovávali by sme si celý trojrozmerný vesmír v pamäti, ale vždy by sme si nejakým spôsobom generovali iba tú časť, ktorú potrebujeme.) Na druhej strane som však presvedčený, že ak ma niečo ako život, prípadne inteligentný život vzniknúť a to navyše spontánne – potom popísaný postup je to najlepšie čo v súčasnosti poznám na to ako to dosiahnuť. Evolúcia v takomto ponímaní nie je náhodná (v dôsledku veľkosti vesmíru a množstva informácií, ktoré obsahuje sa tak síce môže javiť), ale je akoby dopredu predurčená (kým však nespustíme simuláciu a neprejdeme ju krok za krokom nevieme ako sa bude presne vyvíjať). Je takýto pohľad na evolúciu nelogický? V klasickom ponímaní za všetko môže náhoda.

V skutočnosti však veda nedokáže presne vysvetliť ako život iba za pomoci náhody mohol vzniknúť. Je to síce teoreticky možné, ale nevidím dôvod, prečo by to malo byť logickejšie ako evolúcia riadená poznaním (hlavne ak sa ukáže, že vloženie poznania do prostredia je v podstate veľmi jednoduchá záležitosť rekurzcie. A keď si predstavíme koľko molekúl obsahuje len obyčajný pohár vody (pričom ich stav je nejaká informácia) a ak si predstavíme aký obrovský je vesmír a aké množstvo informácii v sebe nesie – nepripadá mi byť takéto chápanie evolúcie v podstate vôbec odporujúce logike. Celá planéta Zem so všetkými informáciami ktoré v sebe obsahuje je iba kvapkou v informačnom mori vesmíru. A preto evolúcia sa síce môže javiť ako náhodná v skutočnosti sa však môže jednať iba o niečo ako deterministický chaos. A deterministický chaos v merkách niečoho tak obrovského ako vesmír sa pre nás môže javiť ako dokonalý nedeterminizmus – čiže náhoda. Mimochodom je známy Einsteinov výrok „Boh predsa nehrá kocky“. A z pohľadu takto postavenej teórie - kvantová mechanika môže naozaj dobre popisovať realitu na základe štatistiky a súčasne samotný vesmír je dokonale deterministický. Iba z dôvodu, že všetko so všetkým súvisí a z dôvodu, že vesmír obsahuje tak veľa informácií na úrovni mikrosvetu ho nevieme popísať lepšie ako princípom neurčitosti.

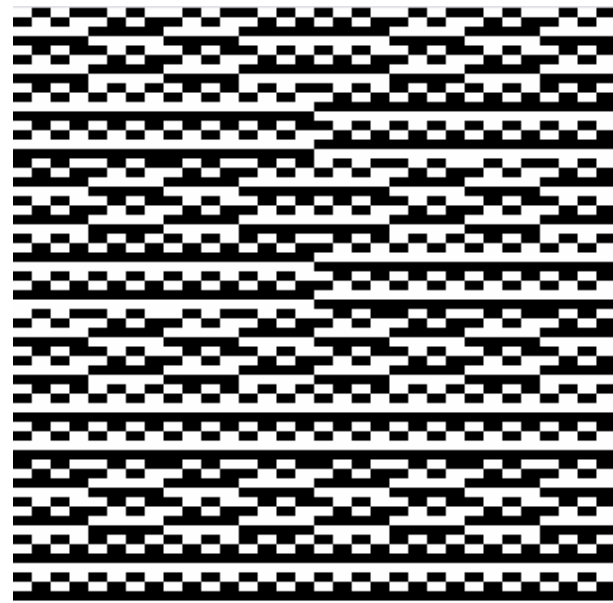
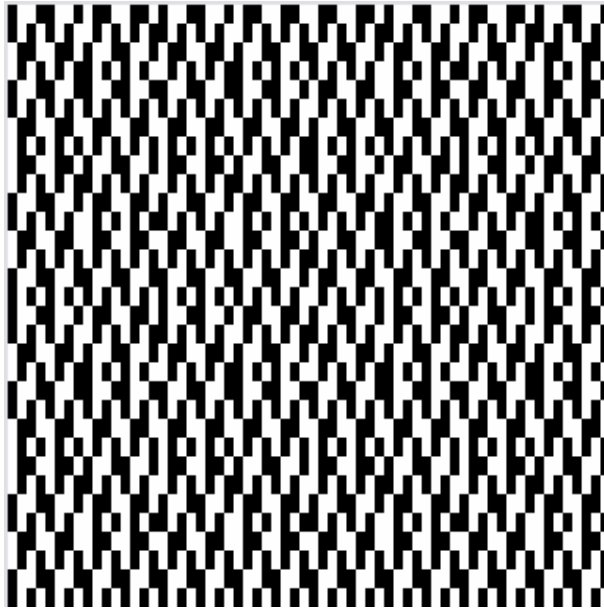
Ešte ostáva zamyslieť sa nad otázkou slobodnej vôle a tým, či je všetko vo vesmíre naozaj predurčené. V našom virtuálnom vesmíre určite áno a ak by sme sa pozreli na skutočný vesmír a fungoval by podľa podobných princípov, potom jediný spôsob akoby mohol poskytovať dokonalú slobodnú vôľu je, že by musel byť nekonečný. V tom prípade jeho nekonečnosť by priamo implikovala nekonečné množstvo informácií a teda aj nekonečné množstvo pravidiel, ktoré ho riadia a preto by sa pre nás javil ako dokonale nedeterministický (aj keď svojmu stvoriteľovi – ak existuje, by sa stále mohol javiť ako deterministický). Ak by bol vesmír konečný, potom ešte vždy nemusí nutne dochádzať k paradoxu slobodnej vôle, pretože ak aby sme chceli odsimulovať jeho stav v nejakom čase, pravdepodobne by sme k tomu potrebovali viac prostriedkov (pamäť, čas), ako je vesmír samotný z dôvodu svojej konečnosti schopný poskytnúť.

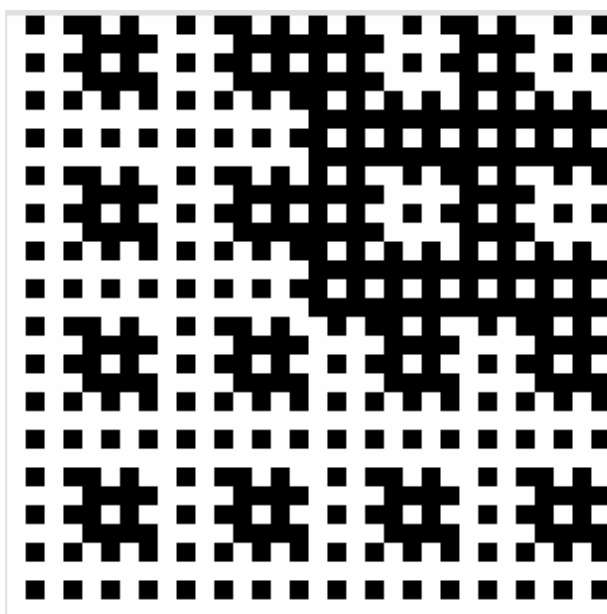
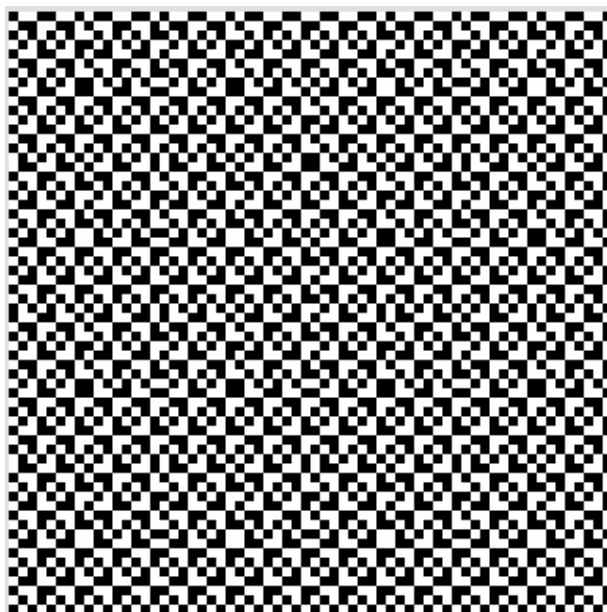
Určite by ešte ostávalo zopár vecí, ktoré by bolo treba napísať k samotnému programu to ako bol vytvorený – nazval by som ich implementačné detaily, bohužiaľ to už časovo nestíham.

Takisto sa stihnem iba veľmi stručne vyjadriť k výsledkom simulácie

## 4 Simulácia virtuálneho vesmíru

Vzhľadom na to, že som pri vytváraní vesmíru doňho nekladal žiadne explicitné poznanie fyzikálnych zákonov, ale nechal som vesmír v podstate „nech sa riadi sám“ vyzerajú získané simulačné výsledky zaujímavo. Pravidelne sa v ňom opakujú obrazce, ktoré majú jasne rozoznateľnú štruktúru. Takmer do oka je bijúca analógia s reálnym svetom o jeho delení na mikrosvet (neusporiadanosť – opis pomocou kvantovej mechaniky) , „medzi“ svet (stredná usporiadanosť – ak sem zahrnieme napr. úroveň na ktorej vníma realitu človek, potom fyzika pre túto oblasť v podstate nemá teóriu ktorá by ho vysvetľovala (napr. konkrétne konanie ľudí a pod.) ) a napokon makrosvet, ktorý je najlepšie organizovaný (opísaný je všeobecnou teóriou relativity – je to úroveň v ktorej dominuje gravitácia). Získané obrazce možno vidieť na nasledujúcich stranách.





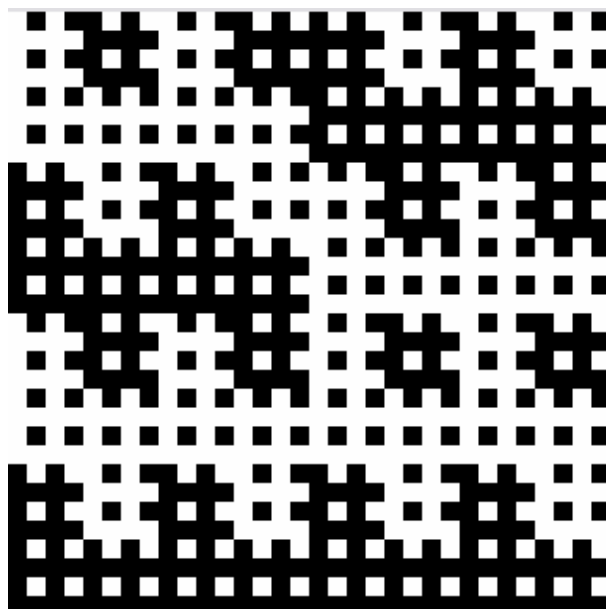
Najmä tento posledný obrazec je vcelku zaujímavý. Pri troche fantázie by som si pokojne vedel predstaviť, že sa jedná napr. o stred galaxie okolo ktorej sa vytvorili hviezdy alebo niečo podobné. Najneusporiadanejší obrazec (mikrosvet) by mohol zase pozostávať z jadier atómov (zložených z malých štvorcov). A na úrovni „medzi“ sveta máme zase niečo čo pripomína dve druhy rovnakej energie niečo ako kladné a záporne pole. Pri ešte inom pohľade by to ale mohli byť aj úseky kódu, ktoré do seba komplementárne zapadajú. V skutočnosti neviem ako mám správne to čo vidím interpretovať. Jedno je však isté, určite sa tu objavila nejakým spôsobom zložitosť, vznik organizovaných aj menej organizovaných štruktúr a to úplne spontánne, bez toho, aby som ja ako programátor musel niečo zložito programovať. Zároveň ak si uvedomíme ako je tento vesmír malý v porovnaní so skutočným (je to iba vesmír rozmeru 32x32 štvorcov) bolo by zaujímavé sledovať ako sa bude vyvíjať napr. pri veľkosti o rozmeroch 1024x1024 pixelov (na to treba najskôr vyriešiť problém



exponenciálnej pamäťovej zložitosti). Vtedy by už v sebe sústreďoval možno dostatočné množstvo poznatkov, aby sa spontánne mohlo vytvoriť niečo ako fyzikálne zákony (pravdepodobne). V prípade, že by sa problémy s pamäťou vyriešili, na základe podobného princípu nevidím problém v zostrojení aj trojrozmerného vesmíru (problém by mohol akurát byť ako vizualizovať získané dáta a ďalej ako vôbec spracovať také množstvo dát). Ešte jedna vec ma pri sledovaní simulácie zaujala, súvisí to s tým vyššie ukázaným obrázkom makrosveta (o ktorom som hovoril, že napr. pripomína niečo ako stred galaxie a pod.) Prečo stred tej galaxie nie je v strede nášho zobrazovaného okna ? Jedným z vysvetlení pravdepodobne je, že som nepoužil presne taký rozmer vesmíru, ktorý by bol tzv. „peknu“

mocninou dvojky  $\left( (2^2)^2 \right)^2$  a teda porušila sa tým „dokonalá“ symetria. Na druhej strane,

keďže nikde nie je presne stanovené ako súradnice x, y namapovať do plochy pokojne by som celý obrázok mohol napr. posunúť doľava alebo doprava a tým by sa vlastne akoby pixle celého vesmíru posunuli o určitý počet jednotiek modulo veľkosť hrany vesmíru. Inak povedané – to čo sa ocitne na jednom kraji vesmíru a prekročí ho sa znovu objaví na jeho opačnom konci. Pre subjektívneho pozorovateľa umiestneného niekde vo vesmíre by neexistovalo nič také ako stred vesmíru. V každom bode, v ktorom by sa nachádzal by bol obklopený celým vesmírom a ten by bol zároveň do seba uzavrený, čiže podobne ako dvojrozmerná bytosť žijúca na povrchu gule. Na povrchu gule takisto nevieme nájsť bod, ktorý by sme mohli označiť ako stred povrchu gule, alebo okraj povrchu gule - namiesto toho si každý pozorovateľ sám vytvára svoju subjektívnu realitu. A takýto pozorovateľ by ani v najmenšom netušil, že sa napr. práve nachádza na okraji vesmíru. V skutočnosti nič také ako okraj virtuálneho vesmíru neexistuje. Zákony vesmíru by pravdepodobne mali fungovať ako už bolo povedané na spôsob modulo aritmetiky. A pokiaľ si uvedomíme, že čas je vlastne tiež iba priestor, potom by aj priestor aj čas mohli byť dokonale relatívne a závislé od pozorovateľa – inak povedané neexistuje nič ako absolutný čas alebo absolutný priestor. V skutku – to čo objektívne jestvuje je iba jednorozmerná postupnosť bitov sústreďujúca v sebe poznanie a všetko ostatné čo sa z nej dá vyvodíť závisí na interpretácii. Na to aby sme to ale naozaj potvrdili by sme si asi museli predstaviť niektorý zo stavov vesmíru na povrchu gule, napr. skúsme to pre tento stav.



Ak by sme si ho dokázali predstaviť na povrchu gule potom by sme mohli objaviť určité nadväznosti, ktoré v ňom na jednotlivých hranách medzi sebou existujú. Ja konkrétne napr. vidím určité nadväznosti, aj keď ako celok si to neviem dosť dobre na povrchu gule predstaviť. Je toho ešte viac čo som chcel napísať ohľadom fyzikálnych zákonov reálneho vesmíru a o tom, či ich napr. nemožno vysvetliť pomocou virtuálneho vesmíru, alebo či by prípadne virtuálny vesmír mohol byť aspoň s nimi konzistentný – zhrniem to stručne v bodoch:

- Je reálny čas diskretný, alebo spojitý ?
- Je stav vesmíru digitalizovateľný ? Ak by napr. bol vesmír konečný, existuje nejaký dôvod myslieť si, že by nemohol byť digitalizovateľný ?
- Je reálny vesmír konečný, alebo nie ? Ak by bol nekonečný o čo nereálnejšia sa javí existencia nekonečného binárneho reťazca ako existencia nekonečného vesmíru ?
- Odkiaľ sa vzali štyri základné interakcie vesmíru ? (silná jadrová, elektromagnetická, slabá jadrová a gravitácia) nesúvisí to náhodou s počtom rozmerov vesmíru ? V mojej simulácii som mal tri rozmery a spontánne akoby vznikli tri interakcie (otázka je samozrejme, či takáto interpretácia toho čo vzniklo je správna) – každá akoby pôsobila na inú úroveň fungovania sveta (úroveň mikro, „medzi“ a makrosveta)
- Dajú sa na základe podobných úvah zdôvodniť hodnoty niektorých fyzikálnych konštánt ? (vypočítať). Súčasná fyzika ich berie za niečo ako axiómu, ktorú namerala z fyzikálnych meraní a ich hodnoty nedokáže zdôvodniť
- Zameniteľnosť hmoty za čas vo virtuálnom priestore - tretiu súradnicu v mojom dvojrozmernom vesmíre si môžem predstaviť nielen ako čas, ale v určitom ponímaní aj ako hmotu. Teda keď je to napríklad trojrozmerná kocka, pričom jedna jej stena bude predstavovať plochu, do ktorej sa bude obsah kocky premietat' a na tejto stene zvolíme jeden bod, potom tretí rozmer, ktorý je obsiahnutý za týmto bodom predstavuje všetku hmotu (atómy, čiže 1, 0), ktorá ním pri spustení simulácie prejde daným bodom prejde. Ak by sme z nejakého dôvodu spomalili plynutie času pre daný bod, potom aj čosi ako hustota hmoty v danom bode sa zvýši – najmä pre okolitého pozorovateľa. Napokon ak by sme pre daný bod čas úplne zastavili obsahoval by v sebe úplne všetku hmotu (maximálnu informáciu) ktorá sa nachádza v treťom rozmere. A platí to aj obrátene, čím rýchlejšie pustíme plynutie času, pre daný bod, tým menej informácie na jednotku tiku hodín dostávame, tým akoby sme mali „menej“ hmoty. Ak by čas plynul maximálnou rýchlosťou (napr. by bol presne rovný perióde nášho vesmíru), potom informácia, ktorú o sebe jeden bod predstavuje sa redukuje na jediný atóm 1 / 0. V takomto chápaní hmoty a času sú čas aj hmota medzi sebou zameniteľné a obe sú relatívne. Samozrejme je tu problém, čo v mojom virtuálnom vesmíre nazvem vlastne hmotou.

A na úplný záver ešte drobné zamyslenie. Z dôvodu, že vo virtuálnom vesmíre všetko so všetkým súvisí, nasledujúci stav je s predošlým spojený exponenciálnou zložitou procedurálnych algoritmov. Ak vesmír je tiež iba niečo počítač, potom by som ho podľa klasických merítok mohol nazvať kvantovým superpočítačom – schopným vykonávať akoby obrovské množstvo paralelných operácií súčasne. V našom virtuálnom vesmíre tiež všetko so všetkým súvisí – v určitom zmysle by sme tiež mohli hovoriť o kvantovom počítači (ale to len akože), avšak je otázne ako ho zmysluplne využiť a vždy samozrejme hlavný problém je ako docieľiť dostatok pamäte na rekurziu. Ak by sa ale tento problém s pamäťou podarilo vyriešiť, neexistuje nejaký spôsob ako to celé využiť k riešeniu exponenciálnych problémov v lineárne logaritmickom čase ? Ak by napr. celý vesmír bol iba jeden superpočítač založený na rekurzii, potom vieme, že cyklus (iterácia) by mala mať podobnú vyjadrovaciu silu ako rekurzia a teda ľubovoľný proces vo vesmíre by mal byť odsimulovateľný na bežnom počítači. Pokiaľ, ale

koncept kvantového počítača vyžaduje k svojmu vzniku existenciu nietkorých fyzikálnych javov na subatomárnej úrovni, logicky sa ponúka otázka, či by tieto javy proste tiež nebolo možné odsimulovať. Alebo zostrojiť takýto počítač nejakým spôsobom, ktorý by zohľadňoval jednotu informácie a algoritmu. Toto je samozrejme vysoko špekulatívne, zatiaľ to neviem nijako logicky zdôvodniť, okrem niečoho čo by som mohol nazvať intuíciou. O tom či niečo ako kvantový počítač skutočne existuje alebo nie, alebo či je ho možné zostrojiť by sme mohli polemizovať, ale existujú napr. prípady, kedy sa ľudské myslenie v určitom zmysle správa ako kvantový počítač, občas dokážeme bleskovo z množstva možnosti vybrať práve tú jednu správnu. Pravdepodobne dobrým príkladom by mohol byť súboj šachistu človeka a počítača.

Počítač napr. dokáže vykonať za jedinú sekundu dve miliardy elementárnych operácií (sčítanie a pod) kým bežný človek by sčítanie čísel iba v hlave robil určite dlhšie ako sekundu. Počítač dokáže za jednu sekundu spočítať povedzme okolo milióna pozícií. Človek (veľmajster) možno dve – tri ? To ako postupuje človek však spočíva v hľadaní hlbších súvislostí v pozícií. Veľmajster proste „vie“, ktoré varianty k ničomu nevedú ešte pred tým, než ich začne počítať. Otázka znie – ako to vie ? Existujú samozrejme aj tu pokusy ako napodobniť ľudské myslenie, aby počítače nemuseli prehľadávať pozíciu iba metódou brute-force (aj nasadenie efektívnych heuristik je v porovnaní s tým ako myslí človek iba metódou brute-force), avšak takéto algoritmy v súčasnosti ďaleko zaostávajú. Hlavný problém je v tom, že počítače proste nie sú inteligentné. Ľubovoľný jednobunkový živočích je inteligentnejší ako najvykonnejší desktopový server súčasnosti.

To čo mám na mysli, je zostrojenie niečoho ako nelineárnej boolovskej algebry, kedy ku každej operácii si okrem informačného vektora uchovávam aj riadiaci vektor bitov (ktorý by sa tiež mohol modifikovať), čím by sa operácie mohli „nejakým“ spôsobom vykonávať paralelne, napr. algebra tretieho rádu by v tomto ponímaní mala na jeden informačný bit dva riadiace bity a teda vstupom by bola trojica bitov schopná uchovávať až osem rôznych stavov a takéto logické operácie by mohli byť schopné odhaliť na svojich vstupoch hlboké závislosti - niečo ako neurónová sieť, len ešte viac podobné spôsobu akým myslí človek. (zatiaľ je to samozrejme čisto fikcia a je možné že aj halucinácia). Jedno je isté, človek určite nemyslí spôsobom ako počítač. Teoreticky, ak by sme niekoho naučili ako interpretovať strojový kód pre architektúru x86, potom by sa tento človek pokojne mohol stať veľmajstrom, lebo programov na to ako hrať šach „dobré“ už dnes existuje veľké množstvo – stačilo by mu iba odsimulovať jednotlivé inštrukcie tak ako sa za sebou vykonávajú. Toto je však prakticky nereálne. Nie je v ľudských silách vyhodnotiť viac ako zopár takýchto inštrukcií za minútu (navyše by to bolo veľmi monotónne). Na to aby bol však takýto program bojaschopný potrebujeme aspoň milióny operácií za sekundu. Kvantový počítač by v takomto ponímaní napr. mohol byť úzko prepojený s umelou inteligenciou (mám na mysli skutočnou umelou inteligenciou, nie súčasne neurónové siete a pod., ktoré inteligenciu iba „predstierajú“).

## 5 Zdroje

1. Evolučné algoritmy – Vladimír Kvasnička, Jiří Pospíchal, Peter Tiňo
2. Ešte stručnejšia história času – S. Hawking, Leonard Mlodinow
3. [www.wikipedia.org](http://www.wikipedia.org)
4. Ale najmä veľa rozmyšľania, často krát aj dvoma alebo troma smermi oproti sebe a to súčasne + veľa času