Neomezený evoluční růst výpočetní síly sebereprodukčních automatů v globulárním vesmíru a jiné výsledky[1]

Jiří Wiedermann

Ústav informatiky Akademie věd České republiky
Pod Vodárenskou věží 2, 182 07 Praha 8, Česká republika
E-mail: jiri.wiedermann@cs.cas.cz

Abstrakt. Popíšeme původní výpočetní modely – globulární vesmír a autopoietické automaty – které zachycují podstatné výpočetní aspekty evoluce: konstrukci sebereprodukčních evolučních automatů pomocí sebesestavování a přenos algoritmicky modifikovatelné genetické informace na potomka. V tomto rámci ukážeme neomezený růst výpočetní síly automatů během evoluce a pomocí interaktivního Turingova stroje charakterizujeme výpočetní sílu rodových linií automatů.

1 Úvod

Je všeobecně známé, že von Neumann ve své stěžejní práci o sebereprodukčních automatech pomocí celulárních automatů dokázal existenci složité struktury se schopností sebereprodukce [6]. Příslušná konstrukce byla neuvěřitelně složitá – její popis zabral von Neumannovi přes 200 stran a výsledný celulární automat se skládal z cca 20 000 buněk. Proto o něco později započaly snahy o jednodušší konstrukci sebereprodukčního automatu. Protože v prostředí celulárního automatu je možné dosáhnout reprodukci jediné buňky triviálním způsobem – jednoduše se „zkopíruje" na vedlejší políčko, hledaly se podmínky, které musí splňovat reprodukovaná struktura, aby se vyloučily triviální řešení. Zdálo se, že tou pravou podmínkou je požadavek najít takovou „univerzální" konstrukci sebereprodukčního automatu, která by zaručila sebereprodukci jakéhokoliv konečného automatu. To se ale záhy ukázalo být lichým, protože už jedna jediná buňka celulárního automatu, která je sama konečným automatem, dovede „simulovat" jakýkoliv konečný automat [1]. Nyní, s odstupem času se zdá, že hlavním smyslem von Neumannovy konstrukce nebyla pouze konstrukce složitého sebereprodukčního zařízení, ale konstrukce takového zařízení, které by mělo schopnost evolučních změn vedoucích „od jednodušších ke složitějším typům", řečeno slovy von Neumanna [2]. Bohužel, von Neumann tento problém nevyřešil, pouze naznačil v závěru své práce, že příslušným evolučním mechanizmem by měly být náhodné mutace, a zařadil odpovídající problém do seznamu problémů, které je třeba řešit v budoucnosti.

Zřejmě von Neumann se svou intuicí byl na správné stopě, jak ostatně dokládají stávající teorie buněčné a evoluční biologie. Ovšem zajímavým problémem zůstává, jestli jedinou možností pro evoluci jsou pouze zcela náhodné mutace genetické informace, anebo jestli existuje i jiný mechanismus, řídící či omezující mutace tak, aby

[1] Tato práce byla vytvořena v rámci výzkumného záměru AV0Z10300504 a byla také částečně podporována grantem 1ET100300419 v rámci programu "Informační společnost".

zasahovaly pouze vybraná místa genetické informace a vedly k efektivnější evoluci. Jinou zajímavou otázkou je, jestli i odpovídající hypotetický „jiný“ mechanismus se může stát subjektem evoluce, a jak se toho dá dosáhnout. Nikoliv poslední v řadě je i otázka, jestli lze skutečně „zkonstruovat“ neomezenou evoluci, generující automaty vedoucí od jednodušších typů ke složitějším, ve smyslu jejich výpočetní síly. Jestliže ano, tak jaká je vlastně „výpočetní síla“ evoluce? A dovedeme vůbec říci, jestli nějaká evoluce bude za daných podmínek pokračovat neomezeně?

V této práci nastíníme odpovědi na shora zmíněné otázky. Přitom budeme vycházet z nedávných prací autora, které byly podobně motivovány [7][8][9][10]. Na rozdíl od citovaných prací tato práce však bude mít více přehledový a popularizující charakter a nebude se věnovat rigoróznímu formálnímu dokazování popisovaných výsledků. Je tomu tak zejména proto, že technický výklad, nutný k přesnému zformování a dokazování uvedených výsledků, vyžaduje daleko větší prostor, než je k dispozici v tomto příspěvku. Zájemce se však může obrátit k původním pramenům.

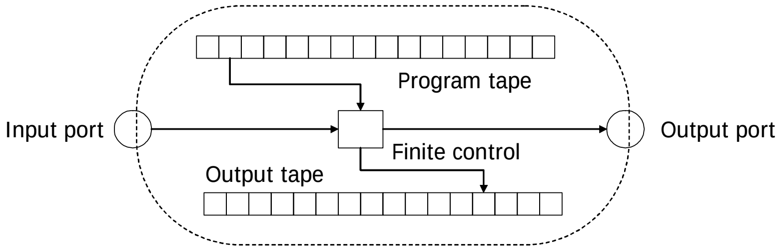
Ve 2. části práce popíšeme model globulárního vesmíru, který je východiskem pro konstrukci tzv. sebereprodukčních globulárních automatů. Ve 3. kapitole popíšeme zjednodušené modely sebereprodukčních automatů – tzv. autopoietické automaty – které mají vlastnost sebereprodukce z definice; přitom svému potomkovi odevzdají genetickou informaci, kterou ale mohou algoritmicky změnit. Ve 4. části ukážeme, že libovolný autopoietický automat lze realizovat ve vhodném globulárním vesmíru. Reprodukce se děje způsobem podobným, jakým se reprodukuje genetická informace v živých buňkách. Dále se budeme zabývat výlučně autopoietickými automaty. Tyto automaty nám umožní v 5. kapitole zodpovědět i otázku, týkající se výpočetní síly rodové linie autopoietických automatů. Ukážeme, že výpočetní sílu takové formace lze přesně charakterizovat pomocí interaktivních Turingových strojů. Šestá kapitola se zabývá problémem existence autopoietického automatu, který má vlastnost „sebezlepšování“ – vede k evoluci, která postupně generuje všechny možné sebereprodukující se automaty. Pro efektivitu celého procesu bude důležité usměrnění mutace tím způsobem, aby zachovávala „syntaxi“ genetického kódu. Pokrytí celého evolučního prostoru autopoietických automatů zaručí evoluce reprodukčního a mutačního mechanizmu. Zmíníme také problém tzv. udržitelné evoluce a ukážeme jeho nerozhodnutelnost. Závěrečná 7. kapitola stručně rekapituluje hlavní výsledky práce.

2 Globulární vesmír

Základní představu o tom, jak je globulární vesmír navržen, jakými se řídí zákony a jak funguje, dostaneme z následujícího myšlenkového experimentu. Představme si klasický dvourozměrný celulární automat a „rozřežme“ jej svislými a vodorovnými řezy na jednotlivé buňky. Na každou buňku budeme i nadále pohlížet jako na konečný automat, umístněný ve čtvercovém pouzdře. Stav automatu se řídí stavem jeho bezprostředních sousedů. Samozřejmě, že po rozřezání celulárního automatu buňky ztratily kontakty se svými původními sousedy. Představme si nyní, že buňky volně náhodně poletují prostorem a občas narážejí jedna do druhé jako při Brownově pohybu molekul. Ovšem, že v okamžiku srážky se buňka opět dostane do kontaktu s jinou buňkou a při té příležitosti může provést „výpočet" podobně jako buňka klasického celulárního automatu. Výsledkem takového výpočtu je však nejen změna stavu kolidujících buněk, ale i změna fyzikálních vlastností stran čtverců, které buňku ohraničují. Tyto vlastnosti mohou být trojího druhu: v závislosti na tom, v jakém je buňka stavu, tak určené strany se mohou přitahovat, odpuzovat, anebo být neutrální. Takže, např., jestliže buňky před srážkou byly ve stavu, odpovídající neutrálním vlastnostem stran čtverce (tj. buňky se ani nepřitahovaly, ani neodpuzovaly), tak po srážce mohou interagující strany buněk změnit své vlastnosti na vlastnost přitažlivosti (stanovu se „lepkavými") a výsledkem interakce bude spojení obou buněk podél dané strany. Anebo v jiném případě se hrany mohou stát odpudivými a buňky budou nuceny pohybovat se směrem od sebe. To znamená, že jako výsledek našeho myšlenkového experimentu dostaneme něco, co bychom mohli nazvat „programovatelnou hmotou", anebo vesmír programovatelných částic. V našem modelu neuvažujeme žádné kinetické aspekty interakce, jako je tomu např. v modelu tzv. celulárních mřížových plynů (lattice gas cellular automata) [3]: nezajímá nás ani směr letu částice, ani její hmotnost, ani rychlost před srážkou a po ní. Zajímá nás pouze fakt, že se částice srazily a že se tím vyvolala změna „přitažlivostních" vlastností částic. Další zjednodušení právě zmíněné představy dosáhneme tím, že budeme předpokládat „nedeterministický" vesmír, ve kterém se částice nacházejí: částice, kterou budeme právě „potřebovat" pro naše úvahy, „přiletí" na místo, kde ji potřebujeme a ve stavu, jaký budeme potřebovat. Tento předpoklad se podobá nedeterministickému výběru z konečného počtu alternativ. Podobně jako v nedeterministickém výpočtu se i v případě nedeterministického vesmíru realizuje ta akce, která spěje k výsledku, který požadujeme (a dovedeme ověřit). Takový přístup má velké výhody v tom, že se nemusíme zajímat o pravděpodobnosti uvažovaných jevů – jestliže je nějaký jev v našem vesmíru možný a potřebujeme, aby se stal, tak se stane. Poslední kosmetická úprava shora zmíněné představy je přeměna buněk celulárního automatu na tzv. globule. Jsou všechny stejně velké, mají tvar kuliček a na jejich povrchu na přesně (výpočetně) definovaných místech jsou tzv. kontaktní oblasti, kterých atraktivní vlastnosti jsou ovládány konečným řízením jako v případě předchozích čtvercových buněk. Takže pokud jsou globule patřičně naprogramovány a nedeterministicky přilétají na místa, kde je potřebujeme, ve stavu, jaký potřebujme a v okamžiku, kdy je potřebujeme, tak lze z nich budovat různé prostorové komplexy, které se navíc mohou spojovat při kontaktu do větších celků. Samozřejmě, že stačí takový kontakt „předpokládat" a on v našem nedeterministickém vesmíru nastane. Dále, komplexy se mohou i „rozkládat" řízeným či neřízeným způsobem. Nekonečná (multi)množina globulí, kterých vlastnosti (stav plus atraktivní vlastnosti kontaktních oblastí) jsou řízeny při interakci konečnou přechodovou relací, se nazývá globulární nedeterministický vesmír (přesnou formální definici viz v práci [9]). Je zřejmé, že ve vhodně zvoleném globulárním vesmíru můžeme pomocí nedeterministických interakcí konstruovat různé objekty. Přesněji řečeno, „my" zmíněné objekty nebudujeme, ony se budují samy, poslušny zákonům globulárního vesmíru (tj. programu, který řídí jednotlivé globule), ve kterém se objekty nacházejí. Pro vnějšího pozorovatele vypadá budování takových objektů jako emergentní proces sebesestavování. Je zřejmé, že náš model je jak zobecněním klasického modelu buněčných automatů, tak i současných modelů sebeorganizace (viz např. [4]). V dalším nás budou zajímat speciální objekty se schopností sebereprodukce a sebezlepšování.

3 Autopoietické automaty

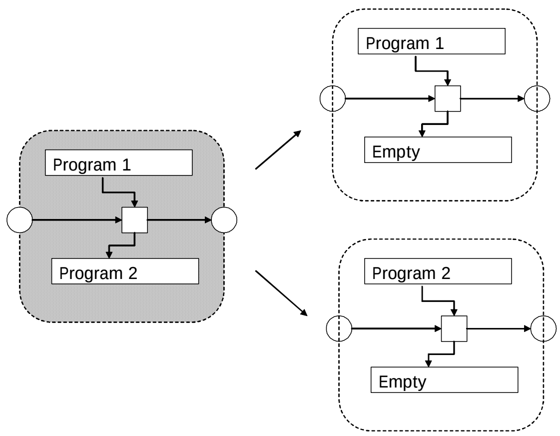
Naším dalším cílem bude konstrukce tzv. globulárního sebereprodukčního automatu, který bude vycházet ze základních principů sebereprodukce „objevených“ von Neumannem [6]: bude využívat „program“ jak pro řízení svého vlastního (výpočetního) chování tak i jako „matrici“ pro výrobu jiného programu, který bude řídit potomka tohoto automatu. V globulárním univerzu však pro „výrobu“ potomka nemůžeme používat diskrétní mřížkovou strukturu klasického celulárního automatu, pomocí které lze na zvolených, předem vypočítaných pozicích v dostatečné vzdálenosti od originálního automatu budovat jeho kopii podobně, jako tomu bylo ve von Neumannově případě. My pro tyto účely budeme využívat sebesestavovací schopnosti globulárního univerza. Dříve než se pustíme do popisu globulárního automatu popíšeme si obecnější, abstraktnější objekt – tzv. autopoietický automat, jehož implementací v patřičně navrženém globulárním vesmíru dostaneme kýžený sebereprodukční globulární automat.



Obrázek 1 Schéma autopoietického automatu

Autopoietické automaty jsou nedeterministická konečně–stavová zařízení modelující zpracování informace a reprodukční a evoluční schopnosti živých buněk. Technicky je autopoietický automat Mealyho automat (překladač) počítající a generující přechodovou relaci svého potomka. Autopoietický automat může pracovat ve dvou režimech. Oba režimy jsou řízeny stejnou přechodovou relací automatu. Prvním režimem je *překládací režim* ve kterém automat zpracovává vstupní symboly čtené prostřednictvím svého vstupního portu, zpracuje je způsobem, popsaným v přechodové relaci a výsledek pošle na svůj výstupní port. Druhým režimem je *reprodukční režim*. V tomto režimu automat již nezpracovává žádnou externí informaci a nic neposílá na svůj výstupní port. Namísto vstupu používá reprezentaci své vlastní přechodové relace, kterou má za tím účelem zapsanou na tzv. *programové pásce*. Tato páska je určena pouze pro čtení. Po níse v obou směrech pohybuje čtecí hlava řízená přechodovou relací, která je na pásce zapsaná. V závislosti na tom, co čte tato hlava a na tom, v jakém je automat stavu automat zapíše na druhou, tzv. *výstupní pásku*, symbol určený přechodovou relací automatu pro tuto situaci. Na výstupní pásku lze symboly pouze zapisovat, způsobem jeden za druhým, bez mezer. Výsledkem práce autopoietického automatu v reprodukčním režimu je přechodová relace potomka daného automatu, zapsaná na výstupní pásce syntakticky stejným způsobem, jako byla zapsána relace původního automatu na programové pásce. Obecně, *přechodová relace* je konečnou podmnožinou kartézského součinu $\Sigma x Q x \Sigma x Q x D$, kde Σ je uspořádaná množina vstupních a výstupních symbolů, Q je uspořádaná množina stavů a D je množina směrů pro pohub čtecí hlavy na programové pásce. Množiny Σ a Q mohou být i nekonečné. Jejich prvky na pásce jsou reprezentovány v unárním kódování, tj. např. prvek $q_i \in Q$ je zakódován jako 0^i (tj. jako řetězec skládající se z i nul). Takže prvek $(s_i,q_j,s_k,q_m,d_n) \in \Sigma x Q x \Sigma x Q x D$ je reprezentován na programové pásce jako $10^i10^j10^k10^m10^n1$. Prvek (s_i,q_j,s_k,q_m,d_n) se nazývá i segment neboli instrukce. Jeho sémantika je „automat se vstupem s_i ve stavu q_j vypíše symbol s_k, přejde do stavu q_m a posune hlavu na programové pásce ve směru d_n“, přičemž hodnota 0 proměnné d_n znamená posun hlavy doleva, 1 doprava, 2 žádný posun. Segmenty jsou zapsány na programové pásce jeden za druhým.

Formálně se práce v jednotlivých režimech řídí typem stavu, ve kterém se automat právě nachází. To znamená, že stavy automatu jsou rozděleny do dvou podmnožin: jsou zde překládací a reprodukční stavy. Pro syntaktické rozlišení těchto dvou typů stavů na programové pásce budeme používat poslední složku v pětici, reprezentující jeden segment přechodové relace. Pokud má být automat po vstupu do stavu q_m v překládacím režimu (resp. pokud je q_m překládacím stavem), tak zmíněná poslední složka bude obsahovat čtyři nuly (hodnotu 0^4); jinak, pro označení reprodukčních stavů, bude mít tato složka hodnotu d_m, která označuje směr pohybu hlavy na programové pásce. Automat začne pracovat v počátečním překládacím stavu a dále, pokud se nachází v překládacím stavu, pracuje v překládacím režimu. V tomto režimu není brán zřetel na složku instrukce udávající směr pohybu hlavy, což ostatně odpovídá i hodnotě poslední složky segmentu. Od okamžiku, kdy automat poprvé vstoupí do reprodukčního stavu, musí už zůstat pouze v reprodukčních stavech. Reprodukční režim končí dosažením tzv. koncového reprodukčního stavu. V tom okamžiku se automat z definice rozdělí na dva autopoietické automaty. První z nich „zdědí“ programovou pásku původního automatu, druhý z nich bude mít jako programovou pásku výstupní pásku původního automatu. Výstupní pásky obou automatů budou prázdné. Je však zřejmé, že programové pásky obou automatů mohou být různé. Oba automaty budou dále pracovat nezávisle, každý se svými vlastními vstupy.



Obrázek 2: Reprodukce autopoietického automatu pomocí dělení

To, že jsme připustili nekonečné množiny symbolů i stavů umožní, aby potomek autopoietického automatu pracoval s větší množinou stavů než jeho rodič. Tím vlastně otevíráme možnost pro evoluci ve smyslu od jednodušších po složitější automaty.

4 Implementace autopoietických automatů v globulárním vesmíru

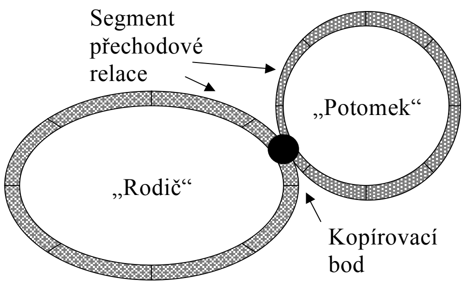
Nyní ukážeme, jak lze realizovat autopoietické automaty v globulárním vesmíru. Za tím účelem musím navrhnout specifický vesmír. Je zřejmé, že autopoietické automaty nemohou existovat ve vesmíru, který je „příliš“ jednoduchý. Např. vesmír s globulemi majícími pouze jeden stav neumožňuje žádné interakce globulí, které by vedly ke změně jejich stavu. Vesmír s neutrálními globulemi (tj. takovými, které nemohou nabýt atraktivní vlastnost) neumožňuje budování globulárních komplexů. V dalším nebudeme explicitně popisovat globulární vesmír, ve kterém ukážeme realizaci autopoietického automatu. Takový automat budeme nazývat globulárním automatem. Požadované vlastnosti vesmíru vyplynou z konstrukce globulárního automatu a bude jasné, že vesmír s takovými vlastnostmi existuje.

Věta 1: Existuje nedeterministický globulární vesmír takový, že v něm pro libovolný autopoietický automat existuje jeho implementace v podobě globulárního automatu.

Nástin důkazu: Nechť A je autopoietický automat a uvažujme jeho programovou pásku, na které je reprezentovaná přechodová relace automatu A jako posloupnost segmentů ve tvaru popsaném v předchozí kapitole. Tuto pásku budeme v našem vesmíru reprezentovat jako řetězec globulí stejné délky, jako je délka pásky. Každá globule řetězce je ve stavu, který jednojednoznačně odpovídá symbolu, který je zapsán v odpovídajícím políčku programové pásky. Pro jednoduchost budeme zatím předpokládat, že globule v našem vesmíru mají dost stavů na to, aby mohly reprezentovat stavy a symboly automatu A a ještě nějaké stavy navíc, které budou sloužit jako pomocné stavy. V dalším budeme pro jednoduchost pojmenovávat globule stavem, ve kterém se právě nacházejí. Globule jsou navrženy tak, aby na svém „rovníku“ měly čtyři ekvidistatní póly. Jedna dvojice protilehlých pólů na každé globuli je ve stavu „lepivém“, tzn., že globule se spojí v posloupnost a navíc můžeme předpokládat, že i první a poslední globule se slepí, takže vznikne prstenec reprezentující přechodovou relaci automatu A. Připomeňme si, že jeden segment přechodové relace má tvar (s_i,q_j,s_k,q_m,d_n). Nazvěme stav současným stavem a stav q_m novým stavem. Na právě popsaný tzv. základní prstenec je připojen další tzv. pomocný prstenec se stejným počtem globulí. Globule pomocného prstence, které odpovídají jedničkám v základním prstenci, jsou ve stejném stavu odpovídajícím jedničce. Ostatní globule jsou ve stavu n označujícím „žádná zajímavá informace“, kromě jedné výjimky. Nechť q_m=q je okamžitý stav automatu A po vykonání instrukce (s_i,q_j,s_k,q_m,d_n). V takovém případě se stav q zkopíruje do odpovídající globule v pomocném prstenci. Právě naznačená struktura, pozůstávající ze dvou prstenců, reprezentuje základ globulárního automatu, který označíme jako G.

Nyní popíšeme, jak globulární automat realizuje jeden krok autopoietického automatu v překladovém režimu. Nechť s je vstupní symbol čtený automatem A ve stavu q. Budeme předpokládat, že G čte vstupy „celým povrchem těla“, tj. globule ve stavu s „přiletí“ a každá z nich se „přilepí“ na jednu globuli pomocného prstence. Nyní se ty globule s, které jsou s globulemi odpovídajícím vstupním symbolům v instrukcích, „okopírují“ do příslušných pomocných globulí. Dále potřebujme distribuovat informaci o současném stavu q do všech segmentů. To se naprogramuje tak, že globule q interaguje se sousední globulí a „předá“ ji informaci o svém stavu, atd., až tato informace „oběhne“ celý prstenec. Nyní se identifikují ty segmenty, které mají dvojici (s,q) na prvních dvou pozicích. To jsou instrukce, které přicházejí do úvahy jako další krok automatu A. Tato identifikace se děje tak, že globule v pomocném prstenci začnou interagovat se svými horními sousedy v základním prstenci a ty páry, které se shodují v základním a pomocném prstenci, přejdou do stavu d, který je superponovaný na stávající stavy a označuje „další krok“. Globulí ve stavu d může být několik, protože transitivní relace je nedeterministická. Takže v dalším kroku se nedeterministicky vybere jedna instrukce a ta se realizuje následujícím způsobem. Okamžitým stavem se stane nový stav q_m a výstupní symbol bude s_k . Za tím účelem se vyšle okružní signál, který zruší označení stavu q_j jako současného stavu. Dále se výstupní symbol postupným „oběhnutím“ celého prstence zkopíruje do globulí, které reprezentovaly vstupní (čtené) symboly. Tyto globule se tím pádem „transformují“ na výstupní symboly a jako takové jsou „vypuštěny“ do prostředí tak, že se zruší jejich vazby s globulemi pomocného prstence. Takže lze s jistou nadsázkou říci, že i výstup z globulárního automatu se děje „celým povrchem těla“. Pokud nový stav q'_m není reprodukčním stavem, je vše připraveno pro vykonání dalšího překladového kroku. Jestliže q_m je reprodukčním stavem, globulární automat zahájí svou vlastní reprodukci.

V reprodukčním režimu automat „čte“ svou vlastní programovou pásku a nadále vykonává instrukce, které jsou na ní napsány. Automat otrocky interpretuje instrukce podobně jako předtím pouze s tím rozdílem, že vstupní symboly jsou čteny přímo z pásky (za tím účelem si automat musí označit pozici čteného symbolu) a výstupní symboly, které se „vyrábějí“ podle potřeby z přiletěvších globulí (nezapomeňme, že jsme v nedeterministickém univerzu), ale nejsou „vypouštěny“ do okolí, nýbrž se postupně, tak jako vznikají, „lepí“ jedna za druhou a uspořádávají se do dalšího prstence, který tímto způsobem roste. Tento další prstenec tudíž reprezentuje výstupní pásku autopoietického automatu. Vznikající „výstupní prstenec“ se dotýká původních prstenců na místě, kde se naházela čtecí hlava při vstupu do reprodukčního režimu. Souběžně s výstupním prstencem se buduje i pomocný prstenec v podobné struktuře, jakou má pomocný prstenec u rodičovského automatu. Jakmile globulární automat dosáhne finálního reprodukčního stavu, nově vybudovaný dvojitý prstence se oddělí od původního, každý automat přejde do počátečního stavu a dále budu fungovat samostatně.



Obrázek 3 Reprodukce globulárního automatu

Právě popsaná konstrukce funguje v univerzu s dostatečným množstvím stavů, jak bylo zdůrazněno v úvodu „důkazu“. Pokud je stavů málo, musíme přejít v reprezentaci přechodové relace na unární kódování stavů i symbolů, tak jak to vyžaduje definice autopoietického automatu. Tím se proces simulace stane složitějším, nicméně v hrubých rysech zůstává stejný.

□

I z předcházejícího nástinu konstrukce globulárního automatu je zřejmé, že proto, aby vůbec bylo možné automat zkonstruovat, potřebujeme nedeterministické univerzum s globulemi, které mají jistý počet stavů. Horní odhad tohoto počtu by se dal odvodit z konstrukce konkrétního globulárního automatu. Na druhé straně je zřejmé, ve vesmírech s příliš malým počtem stavů nemusí být vůbec principiálně možné globulární automat sestrojit.

5 Výpočetní síla autopoietických automatů

Nyní, když víme, že každý autopoietický automat lze implementovat ve vhodném vesmíru pomocí globulárních automatů, můžeme studovat výpočetní sílu sebereprodukčních globulárních automatů prostřednictvím autopoietických automatů bez toho, že bychom se museli zabývat detaily jejich implementace. Následující věta přesně charakterizuje výpočetní sílu rodové linie autopoietických automatů pomocí interaktivního Turingova stroje. *Rodová linie autopoietických automatů* je potenciálně nekonečná posloupnost takových automatů, kde každý následující je potomkem svého předchůdce. Dostaneme ji tedy tím způsobem, že uvažujeme „genealogický“ strom všech možných potomků vybraného nedeterministického autopoietického automatu a v tomto stromě zvolíme jednu z cest začínající ve vrcholu stromu. Strom uvažujeme proto, že nedeterministický autopoietický automat může mít několik různých potomků. Který z nich se skutečně „narodí“ závisí na posloupnosti nedeterministických instrukcí, které se realizují během výpočtu, a na vstupu, který je automatu „předložen“. Interaktivní Turingův stroj (ITS) [5] je Turingův stroj, který čte své vstupy prostřednictvím vstupního portu a výstupy posílá na výstupní port. Jak u autopoietických automatů tak i u interaktivních Turingových strojů povolíme i tzv. prázdné vstupy, které vlastně znamenají, že na vstupním či výstupním portu se nevyskytuje žádný „reálný“ symbol. Budeme říkat, *že ITS simuluje autopoietický automat* (či naopak) právě když obě zařízení počítají stejný překlad (zobrazení) posloupnosti vstupních symbolů na posloupnost výstupních symbolů, přičemž v obou posloupnostech „vyškrtáme“ prázdné symboly.

Věta 2: Výpočetní síla rodové linie (nedeterministických) autopoietických automatů je rovna výpočetní síle nedeterministického interaktivního Turingova stroje.

Nástin důkazu: Nejprve ukážeme, že nedeterministický ITS S dovede simulovat danou rodovou linii $L=\{A_1,A_2,\ldots\}$ autopoietických automatů. Stroj S je univerzální stroj, který má na své pracovní pásce zapsánu přechodovou relaci právě simulovaného automatu A_i. Pokud je tento automat v překladovém režimu S interpretuje program A_i a tudíž realizuje stejný překlad. Jakmile A_i přejde do reprodukčního režimu, S čte své vstupy z pásky a výstupy zapisuje na další pracovní pásku. Když A_i vstoupí do závěrečného reprodukčního stavu, S jednoduše zamění své dvě pásky: druhá (výstupní) páska zřejmě odpovídá programové pásce automatu A_{i+1} a původní programová páska se vymaže, aby později použita jako výstupní páska. Je zřejmé, že pokud taková simulace začne automatem A_1 tak S realizuje stejný překlad jako linie L.

Opačná simulace je složitější. Na výpočet stroje S se budeme dívat jako na výpočet linie automatů, přičemž i-tý automat simuluje stroj S, který používá prvních i políček své pásky. Zřejmě takový stroj lze simulovat pomocí autopoietického automatu A_i,A_i který má ve svých stavech zapamatovány konfigurace stroje S s páskou délky i a příslušný program má na své programové pásce. Začneme automatem, který simuluje S s páskou délky i pro nějakou konstantu c=i>1. Jakmile potřebuje stroj S přejít na (i+1)-vé políčko, automat A_i vygeneruje konečný program, který odpovídá všem možným přechodům mezi konfiguracemi stroje S délky (i+1). To může vhodně naprogramovaný automat A_i udělat, protože takový program se „moc neliší" od stávajícího programu automatu A_i. Rozdíl je v tom, že nyní jsou konfigurace o jedno políčko delší a máme jich k-krát více než předtím, přičemž konstanta k závisí pouze na stroji S a nikoliv na délce i jeho pásky. Nové konfigurace tudíž musí automat vygenerovat a zapsat na svou výstupní pásku pro všechny možné symboly, které může S zapsat na (i+1)-vní políčko, a stavy a pozice hlavy na tomto políčku. To lze udělat v několika přechodech nad stávající programovou páskou, kopírováním stávajících konfigurací a jejich lokální modifikací na tom konci, kde je prodlužujeme. Poté A_i vygeneruje A_{i+1} a celý proces simulace pokračuje podobně dále. Všimněme si, že pro všechny automaty v linii jsou instrukce, zabezpečující „rozmnožování" automatu, tj. generování programu pro potomka A_{i+1} z programu automatu A_i, stejné. Opět je zřejmé, že takto konstruovaná linie bude simulovat S. Pro podrobnosti viz původní práci [10].

□

6 Neohraničená evoluce

Nyní si položme následující otázku: existuje autopoietický automat, který by generoval všechny možné autopoietické automaty? Ukážeme, že ve vhodném nedeterministickém vesmíru je odpověď na předchozí otázku kladná.

Věta 3: Existuje nedeterministický autopoietický automat, který v nedeterministickém vesmíru generuje všechny možné autopoietické automaty.

Nástin důkazu: Začneme automatem, který má pouze „rozmnožovací orgány", tj. pouze instrukce pro práci v reprodukčním režimu. Tyto jsou navrženy tak, aby postupně, v nekonečném počtu generací, nedeterministicky modifikovaly stávající přechodovou relaci všemi možnými způsoby při zachování její syntaxe. Tento automat se stane „praotcem" genealogického stromu, o kterém dokážeme, že bude obsahovat všechny možné autopoietické automaty. Konkrétně vypadá náš automat takto. Jeho přechodová relace se skládá ze segmentů, které obsahují pouze reprodukční stavy. Jsou zde instrukce, které čtou stávající segmenty z programové pásky a přepisují je na výstupní pásku a přitom je i modifikují. Jediné, co nepodléhá modifikaci, je syntaxe segmentů: oddělovače (jedničky) mezi segmenty a také počet sektorů v segmentu (každý segment se skládá z pěti sektorů, každý a nich obsahuje samé nuly, nejméně jednu, sektory jsou odděleny jedničkami, poslední sektor může mít nejvýše čtyři nuly) zůstávají zachovány. Modifikace jsou trojího druhu:

- Změna v rámci sektoru: při přepisování se nedeterministicky přidá anebo ubere jedna nula (to má za následek, že „potomek" automatu pracuje s více či s jinými symboly anebo stavy než jeho rodič);

- Přidání dalšího segmentu, kterého „náplň“ se generuje nedeterministicky (tím vznikají potenciálně „složitější“ automaty);
- Vynechání segmentu (automaty se „zjednodušují“).

Všimněme si, že dle předchozího lze sektor změnit (prodloužit anebo zkrátit) pouze o jeden symbol a počet segmentů lze také změnit nejvýše o jednotku. Z toho je zřejmé, že tímto způsobem vygenerujeme z právě popsaného automatu autopoietické automaty se všemi možnými přechodovými relacemi. Ty z nich, které mohou na nějaký vstup „přežít“, tj. dosáhnout reprodukčního stavu, takový vstup skutečně dostanou, díky předpokladu o nedeterministickém vesmíru. Ostatní se nereprodukují.

□

Je důležité si že právě naznačená konstrukce umožňuje evoluci mechanizmu algoritmického přepisování genetické informace – tj. v jistém smyslu zde máme mechanizmus evoluce evoluce. Řečeno ještě jinak – evoluce se v našem posledním modelu autopoietického automatu neděje podle předem daných pevných pravidel, ale tato pravidla se průběžně vyvíjejí. Všimněme si, že automat konstruovaný v důkazu Věty 2 tuto vlastnost neměl. Zdůrazněme ještě, že pro pokrytí celého evolučního prostoru autopoietických automatů je důležité, aby evoluci skutečně podléhaly obě složky řízení automatů – jak překladová, tak i reprodukční.

Právě dosažený výsledek dává odpověď na von Neumannovu otázku, jestli existuje evoluce postupující od sebereprodukčních automatů jednodušších typů směrem ke složitějším, přičemž složitost je chápána skutečně ve výpočetním smyslu – složitější automaty mají složitější chování. Pro dosažení tohoto výsledku byl důležitý předpoklad, že automaty dostávaly na vstupu data, které je „udržely při životě“. Takže vlastně „prostředí“ se přizpůsobovalo potřebám automatů. Tak tomu ovšem v reálném světě není. Položme si proto otázku, jestli lze pro daný autopoietický automat a předem danou nekonečnou postoupnost vstupů rozhodnout, jestli automat bude generovat nekonečnou rodovou linii automatů, které budou zpracovávat daný vstup. Tomuto problému budeme říkat *problém udržitelné evoluce*.

Věta 4: Problém udržitelné evoluce je algoritmicky nerozhodnutelný.

Nástin důkazu: Podle Věty 2 je výpočet libovolné rodové linie autopoietických automatů ekvivalentní výpočtům interaktivního Turingova stroje. To znamená, že problém udržitelné evoluce je roven problému zastavení zmíněného stroje, o kterém je známo, že je nerozhodnutelný.

□

7 Závěr

V článku jsme přehledovým způsobem prezentovali několik fundamentálních výsledků, týkajících se sebereprodukčních automatů. Předně jsme definovali globulární vesmír a ukázali jsme, že v něm existuje implementace tzv. autopoietického automatu. Pojem autopoietického automatu je abstrakcí podstatných výpočetních vlastností sebereprodukčních automatů, a sice algoritmické modifikace genetické informace při jejím přenosu z rodičovského automatu na potomky. Implementace autopoietického automatu v globulárním vesmíru je zajímavá proto, že ukazuje, že reprodukce automatů se může dít i jiným způsobem než doposud studovaná reprodukce v celulárních automatech. Tento „nový“ způsob reprodukce je vysokou abstrakcí reprodukce živých buněk a v podstatě odpovídá replikaci (a modifikaci) DNA řetězců. Dále jsme ukázali, že výpočetní síla rodové linie autopoietických automatů je stejná jako výpočetní síla interaktivních Turingových strojů, že může existovat neomezená evoluce autopoietických automatů a že problém udržitelné evoluce je pro tyto automaty nerozhodnutelný. Uvedené výsledky naznačují životaschopnost příslušných modelů a současně jsou zajímavé jak z hlediska výpočetní teorie tak např. z hlediska umělého života. Zajímavým a důležitým problémem do budoucna je otázka, jestli v rámci našich modelů lze nějakým způsobem dosáhnout evoluční (či nějaký jiný věrohodný) vznik sebereprodukčních globulárních automatů. To by pomohlo odhalit tajemství vzniku života, i když zatím pouze umělého.

Literatura

[1] Herman, G. T.: On universal computer constructor. Information Processing Letters, Vol. 2, s. 61–64, 1973

[2] McMullin, B.: John von Neumann and the Evolutionary Growth of Complexity: Looking Backwards, Looking Forwards... Artificial Life, Vol 6. Issue 4, Fall 2000, s. 347-361

[3] Rothman, D., Zaleski, S.: Lattice-Gas Cellular Automata: Simple Models of Complex Hydrodynamics. Collection Aléa Saclay, Vol. 5, Cambridge University Press, 1997

[4] Rothemund, P., Winfree, E.: The program−size complexity of self−assembled squares (extended abstract). In Proceedings of the thirty-second annual ACM symposium on Theory of computing, s. 459–468. ACM Press, 2000.

[5] van Leeuwen, J., Wiedermann, J.: The Turing machine paradigm in contemporary computing, in: B. Enquist and W. Schmidt (Eds), {\em Mathematics Unlimited - 2001 and Beyond}, Springer-Verlag, Berlin, 2001, pp.\ 1139-1155.

[6] von Neumann, J.: Theory of Selfreproducing Automata. A. Burks (Ed.), University of Illinois Press, Urbana and London, 1966

[7] Wiedermann, J.: Spojení samoroganizace s výpočty: minimální život v moři umělých molekul. In: Sborník 4. česko—slovenské konference Kognice a umělý život 4, CAL 2004

[8] Wiedermann, J.: Coupling computational and non−computational processes: minimal artificial. Pre−proceedings of the Fifth Workshop on Membrane Computing (WMC5), G. Mauri, Gh. Paun, C. Zandroni (Eds.), Dept. of Comp. Sci., University of Milan, Bicocca, Italy, June 16–19, 2004, 444 s.

[9] Wiedermann, J.: Self−reproducing self—assembling evolutionary automata. Proc. of the Workshop on Tilings and Cellular Automata, WTCA'04, CDMTCS Rechnical Report, University of Auckland, Dec. 2004

[10] Wiedermann, J.: Autopoietic automata. Tech. Rep. V−929, Institute of Computer Science AS CR, Prague, February 2005