

České vysoké učení technické v Praze
Fakulta elektrotechnická

Czech Technical University in Prague
Faculty of Electrical Engineering

Evoluce digitálních organismů v počítači
Computer evolution of digital organisms

Ing. Vladimír Scholtz, Ph.D.

Souhrn

S myšlenkou vytvoření vlastní evoluce a její úspěšnou implementací přišel Thomas Ray, biolog z Delawarské univerzity. Na počátku 90. let minulého století vytvořil systém zvaný Tierra (Země). Šlo o počítačový systém několika krátkých, vzájemně si konkurujících, programů. Programy se množily, v systému docházelo k mutacím a díky tomu k evoluci. V tomto systému Thomas Ray pozoroval, kromě jiného, vznik parazitismu, kdy některé programy zkrátily svůj kód a k životu používaly kód programů jiných. Dalším významným systémem byla Avida, vyvinutá na Michigan State University, kde byl pozorován vznik složitých funkcí. Zajímavý je systém Amoeba (publikován v roce 1996), ve kterém byl pozorován vznik samoreplikujících struktur z náhodného počátečního kódu. Já jsem se těmito systémy nechal inspirovat a spolu s několika spolupracovníky jsme vyvinuli v roce 2008 vlastní systém Svet, ve kterém byly také pozorovány některé prvky evoluce. V následujícím textu budou popsány systémy Tierra, Avida, Amoeba a Svet.

Summary

The idea of creating own evolution and successful implementation came from Thomas Ray, a biologist at the University of Delaware. In the early 90s of the last century he created a system called Tierra (Earth). It was a computer system of a short competing programs. Programs have reproduced and in the system occurred the mutations. Among other things, Thomas Ray observed the creation of parasitism: some programs shorten their code length and the missing parts used from other programs. Another important system Avida was developed at Michigan State University, where the creation of new complex functions was observed. There is also an interesting system Amoeba (published in 1996), in which the emergence of self-replicating structures from random initial code was observed. I was inspired by these systems and together with colleagues, in 2008 we have developed our own system Svet, where also some elements of evolution were observed. This text describes the Tierra, Avida, Amoeba and Svet systems.

Klíčová slova:

digitální evoluce, evoluce, Tierra, Avida, Amoeba, Svet, parazitismus, vznik složité funkce

Keywords:

digital evolution, evolution, Tierra, Avida, Amoeba, Svet, parasitism, complex function creation

Obsah

1	Úvod	6
2	Historie digitální evoluce	7
3	Systémy digitálního života	8
3.1	Tierra	8
3.2	Avida	10
3.3	Amoeba	12
3.4	Svet	13
4	Význam digitálního života pro studium biologické evoluce	17
	Reference	19
5	Ing. Vladimír Scholtz, Ph.D.	21

1 Úvod

Živé systémy jsou subjektem i objektem zvláštního typu evoluce, evoluce biologické, kterou popisuje evoluční teorie [1]. Pokud bychom chtěli abstrahovat, můžeme na základě evoluce biologické stanovit základní podmínky, za kterých by k dějům analogickým k biologické evoluci mohlo docházet i u systémů obecných:

Reprodukce Organismy, nebo jejich zobecněné analogie, samy vytvářejí živoucí kopie samy sebe.

Kompetice Organismy mezi sebou soutěží o zdroje (energie, čas, prostor). V evoluční biologii je hlavním procesem přirozený výběr, jak jej postuloval Charles Darwin. Dnes je však známo, že není zdaleka procesem jediným.

Mutace Organismy se náhodně mění.

Splnění těchto tří základních podmínek je možné docílit vytvořením vhodného počítačového systému, ve kterém by byly organismy reprezentovány programy, jejichž vzájemná kompetice by vyplývala z charakteru systému a v němž by docházelo k mutacím, tj. občasným náhodným změnám běžících programů. V tomto systému již může dojít k evoluci programů a vzniku vlastností typických pro živé systémy, za které jsou považovány komplexita, uspořádanost, biodiverzita a vznik účelných (nikoli účelových) vlastností. Tento proces vývoje počítačových programů je nazýván digitální evolucí.

Zatímco teorie ve fyzice, chemii a jiných exaktních vědách se zakládají na tvrzeních, podle kterých jsou schopné předpovídat chování systému který popisují, pro studium biologické evoluce tuto možnost nemáme. V exaktních vědách teorie předpovídá výsledek experimentu a tento experiment teorii buď potvrdí nebo vyvrátí. Avšak biologická evoluce probíhá okolo nás jen jedna, nedá se s ní příliš manipulovat a navíc probíhá velmi pomalu. To znamená, že evoluci můžeme sledovat (například přes fosilní záznamy nebo sekvencování DNA), ale experimentální kontrola teoretických předpovědí je přinejmenším časově náročná. Jsme sice schopni vyšlechtit (v omezené míře) organismy a podle svých potřeb potlačit, resp. posilnit, u nich nějakou vlastnost. V tomto procesu však jen manipulujeme s už existujícími organismy a vzhledem k jejich složitosti měníme jejich způsob fungování jen minimálně. Sledovat probíhající evoluci biologických systémů můžeme

pouze u živočichů s krátkou generační dobou. Nejčastěji jsou používány různé viry a bakterie, ale i některé druhy hmyzu, např. octomilky (*drosophila*). V každém případě je pěstování a chování i těch nejméně náročných živých organismů nepoměrně složitější než spouštění a sledování programů na počítači. Navíc při popisování a sledování evoluce živých objektů jsme nuceni sledovat pouze „jednu“ evoluci a bývá velice náročné rozhodnout, do jaké míry je výsledná pozorovaná vlastnost dílem náhody a do jaké míry zákonitostí. Z tohoto důvodu představuje studium digitální evoluce vhodný nástroj ke studiu obecných vlastností evoluce.

2 Historie digitální evoluce

Původní oblastí průniku evoluční biologie a výpočetní techniky jsou genetické algoritmy nebo jejich obměny. Všechny tyto varianty představují vhodnou metodu pro řešení některých problémů a jsou založeny na následujícím principu: (1) nejdříve vytvoříme náhodná řešení problému, (2) řešením přidělíme hodnotu reprezentující jejich kvalitu, (3) vybereme několik nej kvalitnějších řešení, (4) tato řešení podrobíme náhodným změnám a vzájemným rekombinacím a (5) celý proces opakujeme od bodu 2 do té doby, než získáme (nebo někdy nezískáme) dostatečně kvalitní řešení.

Genetické algoritmy jsou sice dobrými nástroji pro řešení některých problémů, postrádají však některé základní znaky evoluce. Ta totiž vyžaduje, aby množení organismů bylo jejich vlastností a jejich výběr vyplýval z jejich interakce mezi sebou a s prostředím.

Dalším krokem k přiblížení se evoluci je možné datovat do roku 1990, kdy se Steen Rasmussen nechal inspirovat počítačovou hrou „Core War“ [2] (a v té době také již rozšířenými počítačovými viry). V této hře soutěžily malé programy různých autorů a vyhrál ten, kterému se podařilo zastavit ostatní kompetující programy. Rasmussen si všiml, že nejúspěšnější jsou ty programy, které se snaží množit (tedy samy sebe kopírovat). Ty pak, v případě svého zničení, mohou dál soutěžit ve svých kopiích. V systému nebyly přítomné mutace a tudíž nedocházelo ke změnám kódu jednotlivých programů, a tudíž ani k evoluci. Z toho důvodu se Rasmussen rozhodl naprogramovat vlastní systém, který byl velmi podobný hře, avšak pro umožnění evoluce do něj, jako zdroj mutací, zařadil vadný příkaz kopírování, který se čas od času při kopírování spltl. V tomto systému už docházelo k evoluci analogické

evoluci biologické, nebyl ale příliš úspěšný, protože po nějaké době všechny programy vyhynuly. Stalo se tak proto, že se programy začaly navzájem přepisovat. Prvním úspěšným projektem v této oblasti byl projekt Tierra z roku 1991, jehož autorem je Thomas Ray. Aby nedošlo k problému, díky kterému neuspěl projekt S. Rasmussena, programy musely před zápisem do paměti nejprve alokovat paměť a pokud došlo k vyčerpání paměti, staré programy musely být smazány, aby poskytl místo novým. Výsledkem již byla skutečná evoluce digitálních organismů. Na tento systém navázal v roce 1993 nový systém Avida, který přinesl do prostředí digitální evoluce soutěžení o zdroje a podnítil vývin komplexních struktur organismů. Zajímavý je systém Amoeba (publikován v roce 1996), ve kterém byl pozorován vznik samo-replikujících struktur z náhodného počátečního kódu. Já jsem se těmito systémy nechal inspirovat a spolu s několika spolupracovníky jsme vyvinuli v roce 2008 vlastní systém Svet, ve kterém byly také pozorovány některé prvky evoluce. V následujícím textu budou popsány systémy Tierra, Avida, Amoeba a Svet.

3 Systémy digitálního života

3.1 Tierra

Systém Tierra [3] vyvinul Thomas S. Ray v roce 1991 inspirován předchozím systémem Steena Rasmussena, od kterého se lišil zejména ochranou paměti alokované organismy. Každý organismus byl reprezentován částí alokované paměti s uloženým kódem organismu, který se postupně vykonával paralelně pro všechny organismy přítomné v systému. Tento virtuální programovací jazyk, nazvaný „Terrian“, obsahoval 32 instrukcí RISCového typu. Zvláštností byl typ adresace paměti pomocí předlohy (address by template), kde se při skocích neadresuje přímo místo v paměti, ale posloupnost čtveřice čísel. Výsledné místo v paměti se určí nalezením obsahu paměti shodného s argumentem, což do jisté míry připomíná enzymy nebo protilátky, které se specificky navazují na jiné molekuly. Jako poslední podmínka evoluce byly do systému zapracovány mutace, a to dvojího druhu: Náhodné přepnutí některého bitu v celé paměti, nebo náhodné přepnutí bitu při vykonávání instrukce při zápisu do paměti nebo aritmetických operacích.

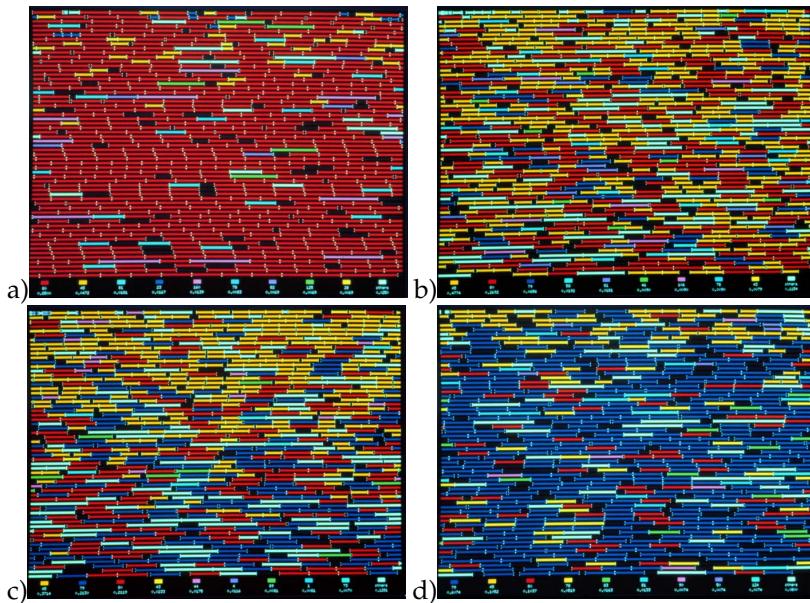
Systém Tierra nabízí organismům typicky 64 kB paměti. Každý organismus si alokuje určitou část této paměti, čímž je chráněna proti

zápisu před cizími organismy, ne však před čtením. Každý tvor může mít nejvýše dva bloky paměti chráněné proti zápisu. Jedna je mateřská a druhá dceřiná. Ve chvíli, kdy se organismus rozdělí, ztratí mateřská buňka možnost zapisovat do dceřiné buňky, ale získá možnost alokovat si novou paměť. V případě naplnění paměti jsou podle určitých pravidel, znevýhodňujících „vadné“ organismy, organismy odstraňovány z paměti a jejich paměť uvolňována.

Systém Tierra byl oživen jediným organismem dlouhým 80 instrukcí, který se dokázal sám množit (kopírovat) a jinak neměl další schopnosti. Po spuštění se tento organismus velice rychle nakopíroval do celé paměti a v momentě, kdy začala být paměť plná a došlo k usmrcování procesů, byla splněna i poslední podmínka evoluce – kompetice – a v systému byla pozorována evoluce.

Z pohledu mikroevoluce je nejzajímavějším pozorovaným jevem parazitismus, který se, za určitých podmínek, vyvinul u organismů, u kterých díky mutacím chyběla část kódu pro kopírování. Právě díky adresování paměti pomocí předlohy mohlo dojít k tomu, že program našel část kódu pro kopírování jiného organismu („hostitele“) a využil ho k vlastnímu kopírování. Vytvoření kopie organismu bez kopírovací části však proběhla rychleji (program je kratší) a tudíž se za stejný čas mohl parazit zkopírovat vícekrát a zabrat tím místo v paměti pro potomky „hostitelského“ organismu. V další evoluci pak byl pozorován vývoj jistého druhu imunity hostitelských organismů vůči parazitům a další přizpůsobování se parazitů k infekci imunních jedinců, jak je možné vidět na obr. 1. Kromě obyčejného parazitismu byl ještě pozorován jev tzv. hyperparazitismu, kdy se vyvinuly organismy schopny parazitovat na jiných parazitech pouze tím, že zmanipulovaly jejich adresování ke kopírování svého kódu (takže využily nejej jejich kód, ale i strojový čas).

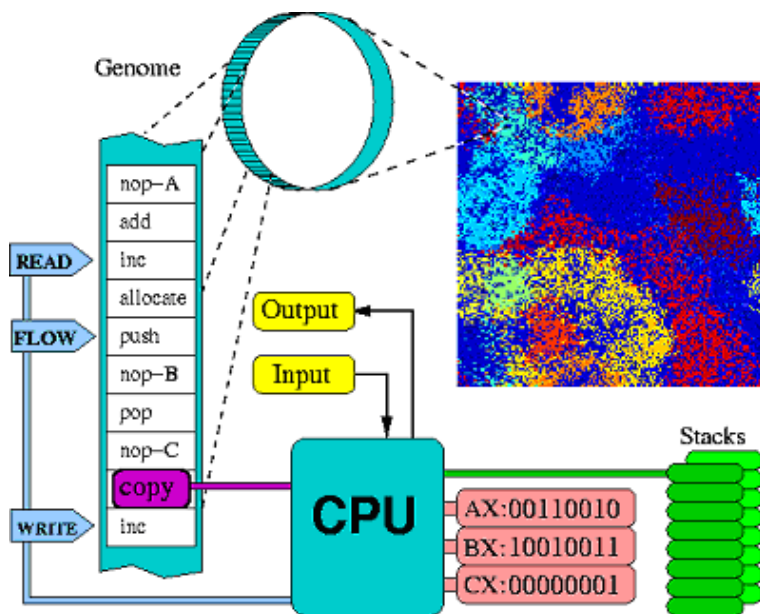
Z pohledu makroevoluce byly při různě nastavených parametrech systému, např. výběr organismu k zabití nebo nerovnoměrné přidělování strojového času, pozorovány různé evoluční odezvy. Při zvýhodnění malých organismů se rozšířily malí parazité, naopak při zvýhodnění velkých organismů přestalo být parazitické zkracování kódů zajímavé a organismy se obecně zvětšovaly, a to i za cenu vzniku tzv. intronů, části kódů nekódujících žádnou užitečnou funkci. Z dlouhodobého hlediska byla v chování systému pozorována dlouhá období evoluční stáze přerušovaná obdobími prudkých evolučních změn.



Obrázek 1: Různé situace v systému Tierra: a) původní organismus (červený) převažuje v systému, parazité (žlutý) jsou zatím vzácní; b) původní hostitel téměř ustoupil, parazité se rozšiřují, objevil se imunní (modrý) organismus; c) zvětšuje se podíl imunních organismů a klesá počet parazitů; d) imunní organismy vytlačují parazity, ty se budou muset přizpůsobit k parazitování na imunních organismech nebo vyhynout. Převzato z [4] se souhlasem autora.

3.2 Avida

Systém Avida [5, 6] byl vyvinut v roce 1993 na Michigan State University autory Charlesem Orfiou, C. Titem Brownem a Chrisem Adamim, kteří se nechali inspirovat systémem Tierra. V současné době je systém dále aktivně vyvíjen ve spolupráci s Clausem Wilkem z California Institute of Technology. Zásadní změnou bylo přidání ekvivalentu metabolismu a získávání energie. Autoři zvolili také trochu jinou architekturu virtuálního prostředí. Zatímco v systému Tierra byl strojový čas organismům přidělován podle pevných pravidel, v systému Avida si mohly jednotlivé organismy získávat prémiový strojový čas na základě úspěšného řešení problémů. Životní prostor organismů



Obrázek 2: Životní prostor v systému Avida (vpravo nahoře) osídlený organismy (barevně jsou odlišeny organismy s různými genomy) a architektura organismu (každý organismus obsahuje vlastní cyklickou paměť a virtuální CPU vykonávající instrukce). Převzato z [7] se souhlasem autora.

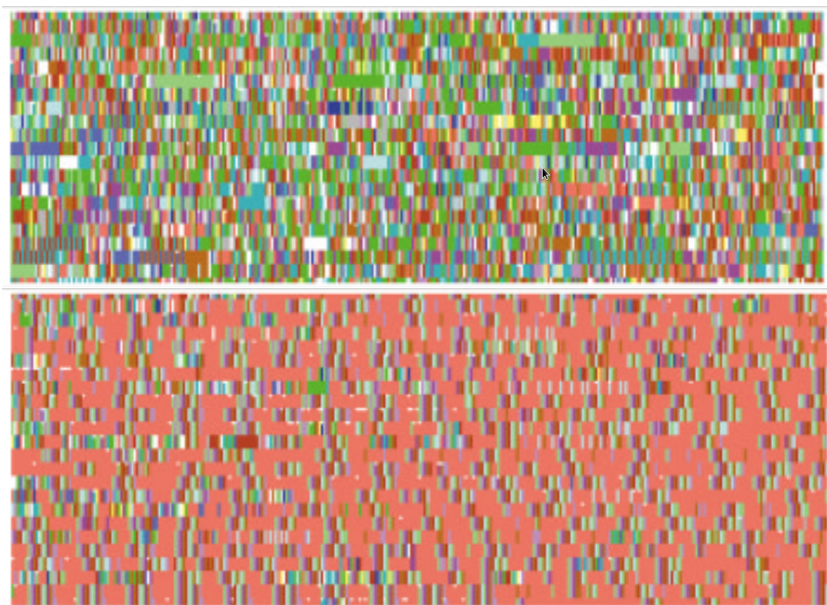
byl tvořen dvourozměrnou, cyklicky uzavřenou mřížkou, kde každý bod mřížky představoval jeden organismus (obr. 2). Každý organismus byl tvořen virtuálním počítačem s vlastním CPU, třemi registry a pamětí, ve které je uložen kód (genom) organismu. CPU každého organismu obsahuje čtyři hlavy ukazující do paměti. Instrukční hlavou je čten kód programu, čtecí a zápisovací hlava slouží ke čtení a zápisu z a do paměti při procesu kopírování, a řídicí hlava je používána při skocích a smyčkách. Avida nemá jediný genetický jazyk, ale má několik různých instrukčních sad. Ty tvoří ve specifických sadách abecedu, ze které se pak skládá genom (program) organismu. Při množení si organismus alokuje novou paměť, kam začne svůj kód kopírovat. K mutacím může dojít při kopírování, kdy se do paměti zapíše změněná hodnota a změní se tím význam instrukce; nebo se do paměti

tato hodnota nezapíše vůbec nebo vícekrát, čímž dochází ke změně délky kódu potomka. Další mutace mohou být přirovnány k mutacím v důsledku kosmického záření a jde o náhodné změny v paměti virtuálního počítače. Organismy mohou mezi sebou interagovat, ať už spoluprací nebo jako hlavní nový prvek, mohou mezi sebou soupeřit o zdroje. Každý organismus po úspěšném splnění nějakého úkolu zadaného systémem, může získat prémiový strojový čas a tím urychlit své množení. Právě tento prvek byl hlavním zdrojem nových pozorovaných výsledků evoluce.

Jedinou logickou operací v instrukční sadě systému Avida je instrukce NAND, ze které, jak je známo, je možné složit všechny ostatní logické operace. Každý organismus se může systému „zeptat“ na otázku, a pokud správně odpoví, získává prémiový strojový čas. Systém postupuje od jednoduchých otázek jako NOT, které je možné složit ze dvou operací NAND, přes složitější až po nejsložitější operaci porovnání EQU, na kterou je potřeba použít minimálně 32 operací NAND (samozřejmě ve správné kombinaci). Výsledkem tohoto odměňování byl vývoj schopnosti organismů tyto úlohy správně řešit. V případě, že byly organismy odměňovány podle vyřešení úloh postupně od nejjednodušších až po nejsložitější operace, došlo při 50 nezávisle spuštěných evolucioních k vývoji nejsložitější operace EQU ve 23 případech. Pokaždé k tomu došlo různou cestou a s jinou kombinací základních operátorů NAND. Pokud však byly programy odměňovány pouze za vyřešení nejsložitější operace EQU, nedošlo ke vzniku této operace ani v jedné z 50 paralelních evolucioní.

3.3 Amoeba

Systém Amoeba byl publikován autorem Andrewem Pargellisem v roce 1996 v pracích [8, 9]. Tento systém byl speciálně navržen pro studium možností a podmínek vzniku „života“ z náhodných posloupností kódů, tedy přechod z prebiotické do biotické fáze systému. V předchozích dvou systémech byly nejkratší samoreplikující se organismy délky 80 a 50 instrukcí, což při počtu 32 a 26 možných instrukcí v systémech dává astronomický počet možných kombinací. Pravděpodobnost, že v kódu nad abecedou velikosti D o délce l instrukcí dojde k náhodnému vzniku vhodného organismu, je $P(D, l) = ND^{-l}$, kde N je počet permutací, které dávají vhodné řešení a obecně je těžké tento počet určit. Systém Amoeba byl navržen tak, aby byla velikost abecedy instrukcí nízká, konkrétně $D = 15$, a samoreplikující se organismus mi-



Obrázek 3: Znázornění obsahu paměti systému Amoeba před (nahore) a po (dole) objevení se samoreplikujícího se organismu. V prebiotické fázi je obsah paměti náhodný, po objevení se organismu je vidět její strukturalizaci kopírováním vzniklého organismu. Převzato z [10] se souhlasem nakladatele.

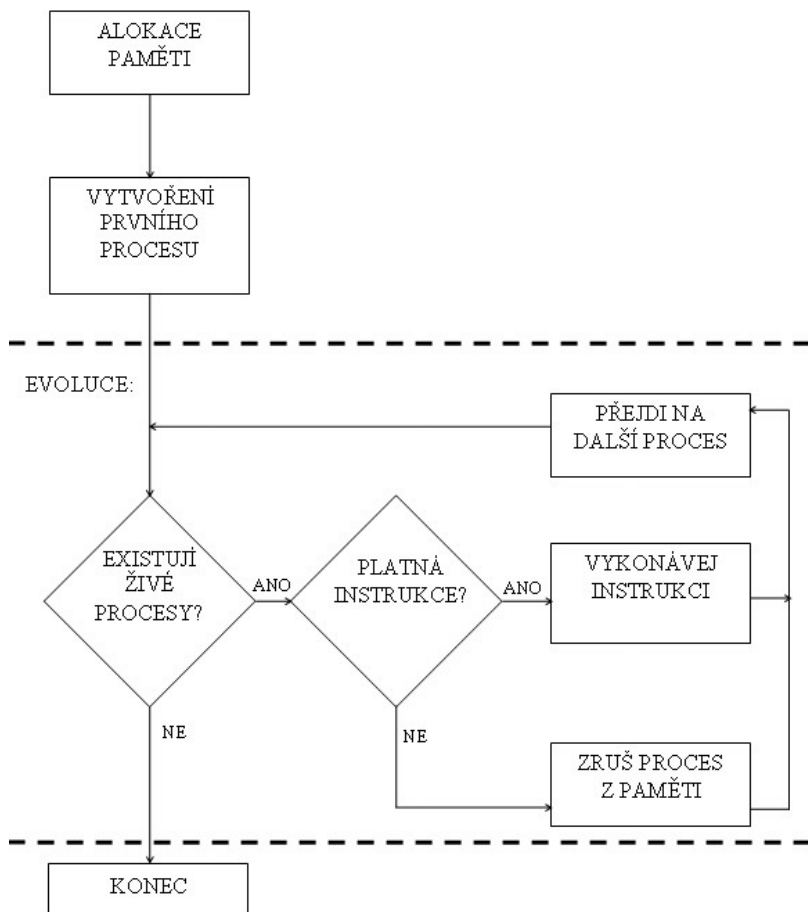
nimální délky $l = 7$ instrukcí. Počet kombinací je pak $D^l = 15^7 \sim 10^8$. V systému Amoeba bylo prostředí osídlováno náhodně vygenerovanými organismy a byla sledována jejich schopnost samoreplikace. Více méně v souladu s očekáváním byl vznik funkčního samoreplikujícího se organismu pozorován s pravděpodobností $p \sim 10^{-7}$, na základě čehož je možné odhadnout parametr $N \sim 10$. Na grafickém zobrazení obsahu paměti na obr. 3 je vidět chaotickou prebiotickou fázi a strukturovanou fázi po vzniku samoreplikujícího organismu (programu).

3.4 Svet

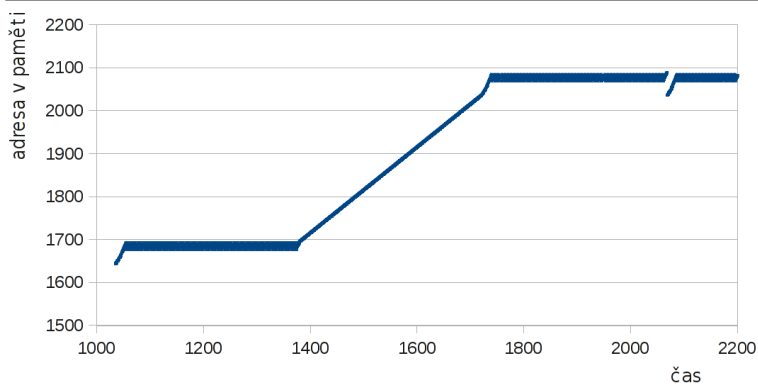
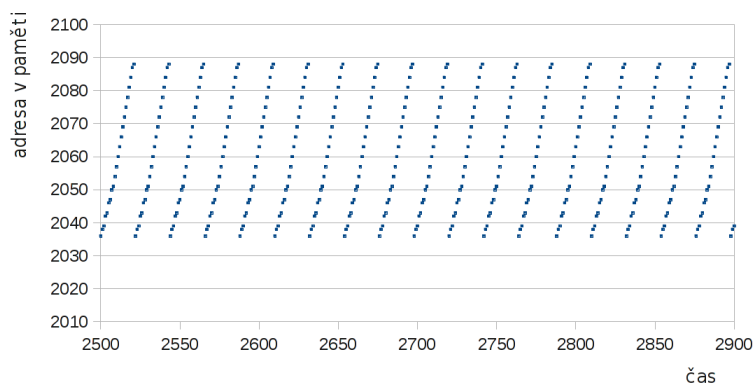
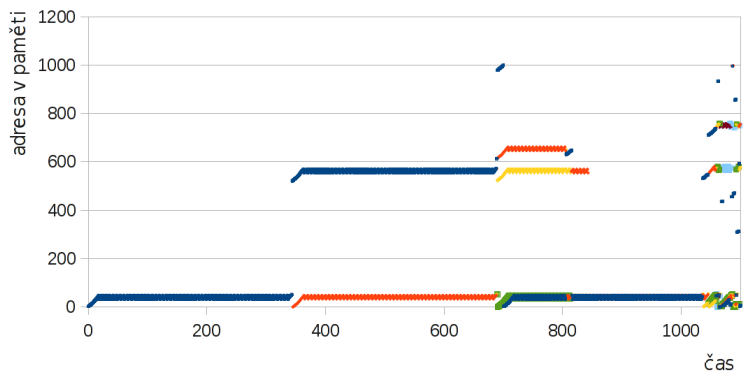
I když jsem se z různých důvodů od svého původního zaměření informatika a programátora odklonil k fyzice, můj vztah k programování tím byl pouze utužen. Když jsem se kolem roku 2005 nechal in-

spirovat při čtení nějaké populární knížky systémem Tierra a Avida (a protože se nerad věnuji pouze jedné činnosti), rozhodl jsem se, že se při svém hlavním výzkumu mikrobicidních vlastností nízkoteplotního plazmatu pokusím zmíněné systémy napodobit. Systém Amoeba jsem v té době ještě neznal. Spojil jsem se se svým bývalým studentem (dnes také úspěšným fyzikem působícím na univerzitě v Düsseldorfu), a spolu s ním sestavil hlavní prvky systému Svet. Název je analogií k názvu Tierra, což je španělsky Země. Systémy Tierra i Avida a dosažené výsledky se nám zdály úžasné, v jejich konstrukci jsme však spatřovali jistou formu násilného přizpůsobení k přiblížení se biologickým systémům, hlavně jejich celulárnímu charakteru. Buňky jsou přece jen již velice složité a vysoce organizované struktury. A tak jsme se rozhodli, že zkusíme postavit systém bez členění na buňky a jejich ochrany před zápisem – atomy a molekuly také nikdo nechrání. Byli jsme zvědaví, co se bude dít. Systém představuje virtuální počítač vycházející ze standardní architektury typu von Neumann, adresování paměti bylo přímé přes 32 bitový registr a jako instrukce jsme implementovali základní instrukce procesoru Intel 8080 s přidáním instrukce pro vygenerování náhodného čísla (pro určení místa v paměti, kde se má vytvářet potomek) a instrukce pro spuštění nového procesu. Instrukce byly kódovány osmi bity, přičemž asi jedna třetina možností zůstala nevyužita a byla chápána jako neplatná instrukce. Paměť systému byla společná pro všechny programy-organismy a každý z nich měl vlastní sadu 256 osmibitových registrů. Paměť nepatřila nikomu a každý organismus obsahoval kromě své sady registrů pouze ukazatel do paměti na právě vykonávanou instrukci. Mutace v systému nastávaly náhodnou změnou obsahu paměti, nebo náhodnou změnou při zápisu číselné hodnoty procesem do paměti. Schématické znázornění činnosti systému je na obr. 4. Jednotlivé procesy jsou seřazeny do zřetězeného uzavřeného seznamu a vykonávají se sekvenčně vždy po jedné instrukci. Proces je zabit, jestliže vykonal neplatnou instrukci. Pokud v systému nezůstává žádný živý proces, je evoluce ukončena. Na počátku je Svet osídlen jediným samoreplikujícím se organismem délky 55 instrukcí a je sledována evoluce. Systém jsme dokončili v roce 2008 a na jeho zkoumání se podílela jedna studentka v rámci diplomové práce v roce 2009 [11] a jedna studentka v rámci probíhající bakalářské práce. Uvedené výsledky jsou převzaty z těchto prací.

Po osídlení se původní proces začal množit na náhodná místa v paměti a spouštět své nové kopie. Na základě nastavených parametrů systému, jako je četnost mutací, velikost paměti nebo maximální po-



Obrázek 4: Vývojový diagram činnosti systému Svet. Na začátku si systém alokuje potřebnou paměť a osídlí jí prvním organismem, pak vykonává cyklicky instrukce jednotlivých procesů. Proces může být zrušen v případě, že vykoná neplatnou instrukci. Když vymře poslední proces, evoluce končí.



Obrázek 5: Typické příklady chování systému: vyhynutí (nahore), zombie (střed) a parazitismus (dole). Grafy představují adresu v paměti, ze které daný proces čte instrukce v závislosti na čase (taktech evoluce), podrobný výklad v textu.

čet procesů nastalo několik scénářů:

Vyhynutí Vyhynutí systému způsobeno zejména vzájemným přepsáním kódů programy, které následně vykonaly neplatnou instrukci. Příklad jednoho průběhu je na obr. 5. První proces na začátku vykoná několik počátečních operací, postupuje v paměti až nastane cyklus kopírování se a vytvoří nový proces (čas 0 – 340). Poté v paměti existují procesy dva, které se opět v cykle kopírují a v čase cca 680 vytvoří svoje další kopie. Dále už většina procesů svoje kopírování nedokončí a zahyne z důvodu vykonání neplatné instrukce. V čase cca 1050 zmutuje i poslední proces a po krátké chaotické fázi evoluce končí vymřením všech procesů.

Zombie Vývoj krátkých živých programů, které se však přestaly množit a sklouzly do cyklení. Příklad jednoho průběhu je na obr. 5. Proces se cyklí v paměti mezi adresami 2035 a 2089, proces tento cyklus nikdy neopustí a nevytváří nové procesy.

Parazitismus Program vlivem mutace nebo přepsáním části kódu jiným programem se typicky po dokončení cyklu kopírování a vytvoření své kopie již nevrátil na svůj začátek, ale začal vykonávat další instrukce v paměti. Pokud za tímto programem byla paměť ještě prázdná, tj. obsahovala instrukce 0 (NOP – nedělej nic), program se postupně posouval v paměti, až narazil na počátek jiného programu, který začal vykonávat. Tento jev je možné považovat za jistou formu parazitismu. Příklad jednoho průběhu je na obr. 5. Proces vykonává svůj původní kód do času 1390, pak se však nevrátí na svůj začátek, ale pokračuje ve vykonávání následných instrukcí NOP v paměti. V čase 1510 natrefí na kód jiného procesu a začne ho vykonávat.

4 Význam digitálního života pro studium biologické evoluce

Z uvedeného přehledu vyplývá, že z hlediska evoluce můžeme považovat digitální organismy za živé a studovat na nich zákonitosti biologické evoluce. V porovnání s biologickými organismy jsou organismy digitální neporovnatelně jednodušší a, i když autoři srovnávají délku jejich kódu s délkou genomu virů, co do složitosti a funkčnosti neporovnatelné. Evoluční teorie však předpokládá, že se všechny sou-

časné organismy vyvinuly z původních velice jednoduchých replikátorů a právě jejich chování je možné na digitálních organismech studovat a dedukovat obecné vlastnosti a zákonitosti evoluce. Biologická evoluce probíhá kolem nás jen jedna, a to ještě extrémně pomalu. Sledování evoluce biologického systému při větším počtu generací bývá nadlidským úkonem a dá se použít pouze u organismů s krátkou generační dobou. V počítači si však můžeme evoluci digitálních organismů spustit několikrát při stejných nebo jiných parametrech a tudíž sledovat, jak se mění chování systému při změně některých parametrů nebo naopak, které výsledky pozorování jsou do jisté míry dílem náhody, tj. při opakovaném spouštění za stejných parametrů jsou výsledky různé, nebo jsou vlastností systému, tj. při opakovaném spouštění se výsledky s nějakou pravděpodobností opakují.

K porovnání biologické a digitální evoluce může posloužit experiment biologa Richarda Lenského z Michigan State University, který po dobu dvaceti let studoval evoluci jednoho klonu bakterie *Escherichia coli* (čemuž odpovídá zhruba 30 000 generací), než se u této bakterie vyvinula nová schopnost příjmu citrátů (populárně např. [12]). Podobně jako se týmu Avida, pozorujícímu digitální evoluci, podařilo demonstrovat vznik komplexní funkce (ve smyslu evoluční biologie jde o výraz pro složitost, nikoli o funkci komplexí proměnné) porovnání dvou čísel, podařilo se Lenského týmu ukázat, že vznik komplexní funkce, v tomto případě trávení citrátů, je možný i v biologii. Můžeme tedy předpokládat, že základní principy evoluce a přirozeného výběru jsou v obou případech shodné, i když důležité a zajímavé je samozřejmě studium obou druhů evoluce. Lenski však při svých experimentech neustále bojoval s kontaminací, bakteriím musel věnovat nepřetržitou péči a k vytvoření „fosilních“ záznamů používal složité zmrazovací techniky. Naproti tomu „kultivace“ digitálních organismů (programů) je nenáročná, 30 000 generací organismů (i když podstatně jednodušších) proběhne za pár minut, vytváření „fosilních“ záznamů představuje uložení kódu na disk. Nezanedbatelná je také výhoda, že celý proces je možné v pátek přerušit, „vypnout počítač“, a pokračovat v pondělí ráno.

Závěrem je tedy možné konstatovat, že u evoluce digitálních organismů můžeme sledovat evoluci sice jednodušších organismů, ale z mnohem obecnějšího pohledu a v řádově kratších časech. O tom, že jde o aktuální a ceněnou oblast vědění, svědčí i skutečnost, že se digitální evolucí zabývají také biologové. Ti na popsanych systémech provádějí další a složitější experimenty, a to jak v oblasti mikro tak

i makroevoluce. To však jde za hranici poslání tohoto textu. Na závěr uvedu několik aktuálních příkladů prací, které se zabývají digitální evolucí: V článku Clune et al., 2012 [13] jsou studovány tendence ontogeneze (vývoje organismu po dobu jeho života) sledovat fylogenezi (vývojovou linii) na digitálních organismech; článek Kawecki et al. 2012 [14] představuje přehledový článek o experimentální evoluci, sledující a porovnávající vývoj jak biologických tak i digitálních organismů; článek Anderson and Harmon 2014 [15] na digitálních organismech studuje a porovnává vliv ekologických a mutačních selekčních tlaků na speciaci, tj. vytváření nových druhů.

Reference

- [1] J. Flégr, *Evoluční biologie*. Academia Praha, 2009.
- [2] A. Dewdney, "In a game called core war hostile programs engage in a battle of bits," *Scientific American*, vol. 250, no. 4, pp. 14–22, 1984.
- [3] T. Ray, "Evolution, ecology and optimization of digital organisms," *Report 92-08-942 of the Santa Fe Institute*, 1992.
- [4] T. Ray, "Tierra photoessay." <http://life.ou.edu/pubs/images/>, 2014.
- [5] C. Ofria and C. Wilke, "Avida: A software platform for research in computational evolutionary biology," *Artificial Life*, vol. 10, no. 2, pp. 191–229, 2004.
- [6] R. Lenski, C. Ofria, R. Pennock, and C. Adami, "The evolutionary origin of complex features," *Nature*, vol. 423, no. 6936, pp. 139–144, 2003.
- [7] B. D. Kenyon College, "The structure of an avida critter." <http://biology.kenyon.edu/courses/biol391/critter.htm>, 2014.
- [8] A. Pargellis, "The evolution of self-replicating computer organisms," *Physica D: Nonlinear Phenomena*, vol. 98, no. 1, pp. 111–127, 1996.
- [9] A. Pargellis, "The evolution of self-replicating computer organisms," *Physica D: Nonlinear Phenomena*, vol. 98, no. 1, pp. 111–127, 1996.

- [10] A. Pargellis, "Self-organizing genetic codes and the emergence of digital life," *Complexity*, vol. 8, no. 4 SPEC. ISS., pp. 69–78, 2003.
- [11] L. Vláčilová, "Digitální evoluce," *Diplomová práce, VŠCHT Praha*, 2009.
- [12] V. Scholtz, "Dvadsať rokov trvajúca evolúcia," *Aldebaran bulletin*, vol. 7, no. 19, 2009.
- [13] J. Clune, R. Pennock, C. Ofria, and R. Lenski, "Ontogeny tends to recapitulate phylogeny in digital organisms," *American Naturalist*, vol. 180, no. 3, pp. E54–E63, 2012.
- [14] T. Kawecki, R. Lenski, D. Ebert, B. Hollis, I. Olivieri, and M. Whitlock, "Experimental evolution," *Trends in Ecology and Evolution*, vol. 27, no. 10, pp. 547–560, 2012.
- [15] C. Anderson and L. Harmon, "Ecological and mutation-order speciation in digital organisms," *American Naturalist*, vol. 183, no. 2, pp. 257–268, 2014.

5 Ing. Vladimír Scholtz, Ph.D.

Narozen 1977 v Bratislavě, Slovensko.

1996 – 2000 bakalářské a magisterské studium: Slovenská technická univerzita v Bratislave, Fakulta elektrotechniky a informatiky, obor „Informatika“ (Bc.) a „Elektromateriálové inžinierstvo“ (Ing.).

2003 – 2006 doktorské studium: České vysoké učení technické v Praze, Fakulta elektrotechnická, obor Fyzika plazmatu.

2006 – 2007 odborný asistent na Katedre fyziky, FEI STU v Bratislavě.

od 2007 asistent a odborný asistent na Ústavu fyziky a měřicí techniky Vysoké školy chemicko-technologické v Praze.

Výuka: Vedení přednášek a seminářů předmětů Fyzika I, Physics I, Physics II, Laboratoř fyziky, Úvod do fyziky plazmatu a Biofyzika.

Účast na grantech:

GAČR 202/03/H162 Pokročilé směry ve fyzice a chemii plazmatu.

MSM ČR 212300016 Tvorba a monitorování životního prostředí.

COST 527.120 Plasma Polymers Modification by Atmospheric Corona Discharge.

MSM ČR 0021620806 Molekulární biologie a patologie buňky za normy a u vybraných klinicky závažných patologických procesů.

MSM ČR 6046137306 Reakčně-transportní chemicko-inženýrské systémy a jejich dynamické chování, modelování a řízení chemických a biotechnologických procesů.

Nadační fond na podporu vědy Neuron. Možnosti uplatnění nízkoteplotního plazmatu při léčbě kožních mykóz.

Publikační aktivity:

WOS 15 publikací; celkový počet citací: 64; bez autocitací: 44; h-index: 5

Scopus 22 publikací; celkový počet citací: 114; bez autocitací: 50; h-index: 6