

Spojení samoorganizace s výpočty: minimální život v moři umělých molekul

Jiří Wiedermann¹

Ústav informatiky AV ČR, v.v.i.
Pod Vodárenskou věží 2, 182 07 Praha 8
Česká republika
jiri.wiedermann@cs.cas.cz

„Maybe there's something beyond computation in the sense that we don't understand and we can't describe what's going on inside living systems using computation only... Maybe we're missing something, but what could that something be?”

Rodney Brooks, 2002 [3]

Abstrakt. Bakteriod je formální abstraktní hybridní systém, který ve své činnosti kombinuje výpočetní a nevýpočetní mechanismy. Ukážeme, že v prostředí umělých molekul, nadanými jistými samoorganizačními schopnostmi, některé bakteroidy vykazují znaky minimálního života: jsou autonomní, replikují se a mají schopnost darwinovské evoluce. Návrh bakteroidů je inspirován představami současné molekulární biologie o dnes již neexistujících (či zatím neobjevených) formách protoživota.

1. Úvod

Koncem roku 2002 získal Dr. Craig Venter, známý jako „luštitel“ lidského genomu, a Dr. Hamilton Smith, laureát Nobelovy ceny, lékař, lukrativní grant amerického Ministerstva energetiky na projekt syntézy umělého života, resp. přesněji „minimální bakterie“, která by byla schopna generovat „čistou“ energii (produkci vodíku), případně mírnit globální oteplování („požíráním“ CO₂) [9]. Tento program stále pokračuje. V současné době byl „přečten“ genetický kód stovek různých mikrobů s cílem nalézt principy, jak fungují nejjednodušší formy života a poté sestavit příslušné počítačové modely. U té příležitosti se rozhořely debaty ohledně definice „minimálního života“. I když se tyto debaty pohybovaly a pohybují především v rámci reálného života, přirozenou půdou pro takové debaty je i oblast umělého života (ALife). Jedním z hlavních cílů umělého života je totiž studium zákonitostí života, které jsou nezávislé na mediu, v němž se odehrává. Takovým mediem může být počítač. Hlavní hnací silou přesvědčení, že (také) v počítačích může povstat život, je schopnost počítačů simulovat reálný svět molekul a komplexních chemických a fyzikálních interakcí mezi nimi. Jedna z možností syntézy ALife v počítači je tedy věrná simulace všech chemických procesů a fyzikálních zákonitostí – jakéhosi

¹ Tato práce vznikla v rámci záměru AV0Z10300504 s částečnou podporou grantu NPV I: Informační společnost1ET100300419

věrného obrazu našeho vesmíru – kterým podléhají reálné molekuly při vzniku života. To se děje v rámci tzv. umělé chemie (viz. např. [6]). Zde však narážíme na naše nedostatečné znalosti odpovídajících procesů a také na nedostatečnou kapacitu počítačů pro účely takové simulace (viz např. [21]). Další možností je kvalifikovaný odhad těchto podrobností, sestavení příslušného počítačového modelu a následné počítačové ověření, jestli i za takových ne zcela potvrzených předpokladů vznikne (teď už jistě umělý) život. Na základě takových motivací vznikla a stále vzniká řada počítačových modelů, které více či méně zdařile modelují ne—li umělý život, tak alespoň některé jeho aspekty. Takovým aspektem může být např. schopnost zpracování informací ve smyslu turingovské vypočitatelnosti a míra takové schopnosti. Zde je základním modelem „praotec“ všech počítačů, Turingův stroj, který byl Turingem navržen také podle biologického vzoru člověka-počítáře. Jiný model – neuronové síť – byl inspirován přirozenými neurony a jejich (tušenou) možností učit se. Další, relativně pozdější modely, jsou celulární automaty², ještě pozdější vycházejí z různých gramatických modelů (L-systémy, eko-gramatické systémy). Tyto jsou těsně následované modely, které ještě více zdůrazňují motivaci molekulární biologií (molekulární výpočty, Watson-Crickovy automaty, spojovací systémy apod.). V poslední době jsou populární i tzv. membránové výpočty (viz např. [12]), které jsou zřetelně motivovány buněčnou molekulární biologií a prepisovacími systémy. Nejnovějším hitem jsou tzv. samosestavovací procesy (self-assembly, viz. např. [1],[17],[20]), inspirované samo-organizačními schopnostmi biomolekul. Příznačné pro teoretické studium všech zmíněných modelů je odklon od původní biologické motivace v okamžiku, kdy se ukazuje jejich „praktický význam“. Jako typický příklad mohou posloužit P-systémy, kde se zdůrazňuje, že cílem těchto modelů není modelování celulárního života, ale naopak zkoumání výpočetní síly modelů inspirovanými celulární biologií [12], anebo v případě samosestavování i využití těchto procesů v (nano)technologii. Tím se ovšem dostávají mimo rámec umělého života, protože např. výpočetní univerzalita v Turingovském smyslu, která se zdá býti svatým grálem těchto a podobných modelů, není fundamentální vlastností živých systému. Naopak, schopnost samoreprodukce a darwinovské evoluce – základních podmínek života – se v rámci těchto modelů většinou nezkoumají, i když i zde se najdou výjimky (viz např. [18]).

Náš příspěvek si klade za cíl zkoumat zejména z výpočetního hlediska přesně to, čemu se shora zmíněné modely, byť inspirované živými systémy, většinou vyhýbají: vlastnosti a projevy života, hledání resp. modelování minimálních vlastností nutných pro život. Takovému životu se říká minimální život. Je třeba zdůraznit již teď, že naším cílem také není modelování reálného života či našich představ o něm – necháme se jimi pouze inspirovat. V důsledku toho zůstaneme pevně na půdě ALife. V současné době mezi odborníky nepanuje shoda o definici života, zachycující jeho rozmanitost v plné šíři. S minimálním životem je to jiné, zde převládající a pro nás použitelná definice minimálního života, je tato [14]:

nějaký systém lze považovat za živý, když

- je autonomní (tj. nezávislý na přítomnosti produktů jiného života v prostředí),
- reprodukuje se

² Vzhledem k relativní známosti těchto i dalších v úvodě zmiňovaných modelů v komunitě odborníků umělé inteligence a umělého života, s výjimkou nejnovějších modelů neuvádíme žádné reference na příslušnou literaturu.

- a vyvíjí se v souladu s darwinovskou evolucí.

Výchozím výpočetní modelem, pomocí kterého budeme zkoumat minimální život, pro nás bude tzv. asynchronní multitransducer s přepínacími stavy. Bude to multimnožina konečných automatů (přesněji řečeno: automatů s výstupem, tj. transducerů), které mohou navzájem asynchronně komunikovat prostřednictvím stavů, které se navíc mohou přepínat z počátečních na ne-počáteční a naopak. Tato množina automatů zpracovává multimnožinu symbolů a řetězců nad nimi, které se nacházejí v prostředí, způsobem podobným, jako pracují P-systémy, tj. každý automat jako by aktivně vyhledal ten řetězec, pro zpracování kterého je naprogramován (pokud takový řetězec v okolí existuje). Automaty multitransduceru „drží pohromadě“ pomocí membrány, ve které se všechny nacházejí a která je produktem činnosti některých automatů. Membránou jsou tedy jednotlivé multitransducery, které se mohou lišit ve své činnosti a v konečném důsledku i „morfologii“, od sebe fyzicky oddělené. Multitransducer umístěný v membráně se nazývá bakteroid. Dále multitransducer zabezpečí kopírování (reprodukcí) „genetické informace“ bakteroidu, která se nachází uvnitř membrány také ve formě řetězce symbolů a popisuje vlastně přechodové (řídící) funkce jednotlivých automatů. Vlastnost genetické mutace je dosažena povolením (resp. vyžadováním) přepisovacích chyb při kopírování genetické informace. Jiné automaty mají na starosti přípravu (syntezu, modelovanou jako výpočet) stavebních kamenů různých dalších percepčních a motorických orgánů bakteroidů. Tyto orgány však multitransducer nekonstruuje – pouze připravuje jejich součásti. To znamená, že multitransducer nemá k dispozici „návod“ na konstrukci těchto orgánů, pouze „v pravý čas“ dodá součástky potřebné pro další fázi vývoje orgánu. Vlastní asambláž je považována za nevýpočetní proces (nezaměňovat s nevypočitatelným procesem, pro který neexistuje algoritmus) v tom smyslu, že není řízena multitransducerem. Kýžené orgány (receptor, motor, transpoziční kanál – jakási „ústa“ bakteroidu, atp.) pak vznikají samoorganizací, resp. „samosestavováním“, tj. pomocí vlastnosti, kterou a priori postulujeme pro některé typy objektů, které automaty zpracovávají. Výsledný mechanismus bakteroidů tudíž kombinuje ryze výpočetní mechanismus multitransduceru s „nevýpočetním“ mechanismem pro konstrukci membrány a percepčních a lokomočních orgánů, který je založen na vlastnostech samoorganizace příslušných součástí. Tento nevýpočetní mechanismus do jisté míry připomíná orákula, která se ve světě Turingových strojů používají za účelem získání výsledků, které daný mechanismus není principiálně schopen obdržet vlastními silami. Jiný pohled na věc je, že výpočetní mechanismy zabezpečují epistemickou složku práce bakteroidu, kdežto nevýpočetní mechanismy složku ontickou [7]. Poslední vlastnost bakteroidů, totiž množení, je zabezpečeno pomocí dělení membrány na dvě části, v jedné z nich se nachází původní a ve druhé zkopírovaná a případně zmutovaná genetická informace. Samotný mechanismus dělení membrány je opět nevýpočetní a pracuje na základě využití fyzikálních a chemických zákonů, panujících v prostředí, ve kterém se bakteroid nachází.

Z hlediska ALife je přínosem práce již samotný návrh bakteroidu, který striktně odděluje výpočetní aspekty jeho činnosti od nevýpočetních. Získané výsledky ukazují důležitost komunikace mezi jednotlivými automaty multitransduceru a dávají představu o typu a rozsahu epistemické práce bakteroidu. Komunikace se v našem případě modeluje pomocí přepínání stavů a zhruba odpovídá aktivaci genů v genetické informaci organismu. V práci dále ukážeme, že definice bakteroidu je „dobrá“ tím způsobem, že zkonstruuje jednoduchý bakteroid – tzv. protobakteroid – o kterém dokážeme, že vykazuje známky minimálního života. Již tento výsledek samotný umožňuje hlubší vhled do principů minimálního života a úlohy zpracování informací ve vývoji protoorganismů na straně

jedné a úlohy samoorganizace na straně druhé. Další evolučně vyšší typ bakteroidu, tzv. selektivní bakteroid, již vykazuje známky kognice – dovede rozlišovat mezi dvěma typy „potravy“ a upřednostnit výživnější stravu. Konečně nejvyvinutější typ bakteroidu, jehož konstrukci zde nastíníme, je motilní bakteroid vykazující známky inteligentního chování – pohybuje se v směru gradientu stravy. Dva posledně zmíněné bakteroidy jsou příspěvkem k pochopení vzniku a fungování kognice a inteligentního chování.

Článek pozůstává ze sedmi částí. První část obsahuje úvod do problematiky a nástin hlavních výsledků a jejich přínosů. Dále následuje definice multitransduceru, ve třetí části popis jeho „vtělení“ do membrány a popis mechanismů růstu a dělení membrány. V části 4 definujeme pojem bakteroidu a jeho speciálního případu protobakteroidu, který lze považovat za nejjednodušší bakteroid splňující podmínky minimálního života. Toto dokážeme v 5. části. Dále v 6. části popíšeme „evoluční inovace“ protobakteroidu. Základem těchto inovací je tzv. selektivní bakteroid, který je schopen preferovat kvalitnější stravu v prostředí bohatém na živiny. Tento bakteroid tudíž vykazuje elementární kognitivní schopnosti a za tím účelem musí být vybaven perceptory a mít schopnost činit rozhodnutí ve prospěch lepší alternativy. Další typ bakteroidu je již vybaven i pohybovým aparátem a dovede se pohybovat směrem ke větší koncentraci potravy. V závěrečné sedmé části jsou zhodnoceny přínosy práce a nastíněn další postup výzkumu.

Článek má předběžný referující charakter popisující problematiku ve stavu zrodu. Zatím není ustálen ani model, ani formalizmus, ani terminologie, výsledky se pouze rysují.

2. Interaktivní konečný multitransducer

Soustředme se nejprve na výpočetní aspekt zamýšleného formálního systému, který má modelovat mechanismy zpracování informací a jejich využití pro řízení činnosti umělého organismu. Pro tyto účely se jako první nabízí využití konečných automatů. Jejich modifikace pro účely interaktivního zpracování (tj. zpracování potenciálně nekonečného proudu předem neznámých vstupních dat) je poměrně přímočará záležitost a výsledné zařízení – tzv. interaktivní konečný transducer – bylo popsáno již mnohokrát (viz např. [15]). Pro naše záměry budeme potřebovat ještě jiné, byť podobné, zařízení – tzv. multitransducer. Na multitransducer lze pohlížet jako na multimnožinu Mealyho automatů konečně mnoha typů, které pracují asynchronně. To znamená, že i když je popis multitransduceru (a tudíž všech automatů dohromady) konečný, mohutnost jejich multimnožiny je prakticky neomezená a počet automatů v ní závisí od množství objektů, které automaty mají v daném chvíli k dispozici pro zpracování (viz v dalším popisu činnosti multitransduceru). Z hlediska výpočtů se tedy jedná o vysoce paralelní zařízení. Každý typ Mealyho automatu je popsán svou vlastní přechodovou funkcí a je upraven pro případ zpracování vstupů, které nejsou předem dány na pásce. Předpokládáme, že multitransducer se nachází v prostředí tvořeném multimnožinou řetězců vstupních a výstupních symbolů, kterým budeme v našem kontextu říkat objekty. Také tato multimnožina není předem daná, může se časem měnit, tj. mohou do ní přibývat nové typy objektů anebo se může měnit multiplicita výskytu jednotlivých objektů. Multitransducer pracuje tak, že soustavně transformuje předepsaným způsobem objekty, nacházející se v prostředí, na jiné objekty. Vstupní objekty jsou čteny přes tzv. vstupní porty a výstupní objekty jsou vypisovány přes výstupní porty jednotlivých automatů. Pokud je objekt strukturovaný, pozůstává z více symbolů, tak je čten sekvenčně, jako řetězec. Každý automat multitransduceru může být navržen i naprogramován i tak, že

zpracovává více objektů současně, podobně jako klasický vícehlavý konečný automat. Přitom předpokládáme, že každý automat čte objekty z prostředí selektivně – má zvláštní schopnost „najít“ a zpracovat v prostředí ten objekt, pro který je naprogramován, pokud v prostředí takový objekt existuje. Tato vlastnost automatů může být chápána i jako vlastnost prostředí, ve kterém se multitransducer nachází – prostředí se jakoby pokouší zpracovat každý objekt každým z automatů, ze kterých se multitransducer skládá, a pokud taková dvojice automat – objekt existuje, tak se zpracování objektu automatem uskuteční. Pro prostředí má tedy potenciál pro realizaci vysoce paralelních výpočtů. V případě, kdy je možné daný objekt zpracovat dvěma různými automaty, tak se náhodně vybere jeden z nich. Pokud v prostředí takový objekt, který by mohl automat zpracovat, neexistuje, automat čeká v ve stavu, ve kterém předtím skončil. Jakmile se v prostředí znovu objeví objekt vhodný ke zpracování, automat se opět spustí z počátečního stavu. Objekt „přečtený“ na vstupním portu po této akci „zmizí“ – je transformován na nějaký výstupní objekt. Poté je port připraven číst další objekt.

Multitransducer se od „běžné“ množiny automatů liší v jednom podstatném bodě – množina počátečních stavů jednotlivých automatů není pevná, předem daná, neměnná, ale jednotlivé stavy se v závislosti na průběhu výpočtu mohou stát počátečními stavy a existující počáteční stavy mohou vlastnost „počátečnosti“ ztratit. Příkazy pro aktivaci počátečních stavů jsou součástí přechodové funkce automatů. Počáteční aktivace stavů je popsána ve specifikaci multitransduceru. Stavy, které jsou momentálně počáteční, budeme nazývat i aktivními stavy. Dále popíšeme formálně případ multitransduceru s jednonásobnými vstupními porty, tj. multitransduceru, ve kterém každý automat čte své vstupy právě prostřednictvím jednoho portu.

DEFINICE: *Interaktivní asynchronní konečný multitransducer* $T = (I, O, S, B, \delta)$ s jednonásobnými vstupními porty je pětice, kde

- I a O jsou konečné abecedy objektů, I je abeceda vstupní a O je abeceda výstupní;
- S je konečná abeceda stavů
- $B \subseteq S$ je množina aktivních stavů na počátku výpočtu
- δ je přechodová funkce tvaru $I \times S \rightarrow S \times O \times S \times \{0,1\}$, která pro objekt $v \in I$ čtený na vstupním portu zařízení a stav $s \in S$ přiřadí nový stav $r \in S$, na výstupní port pošle objekt $w \in O$ a stavu q přiřadí hodnotu aktivace 0 nebo 1 (0 znamená, že stav není počáteční, 1 opak); tomu odpovídá zápis $\delta(v,s)=(r,w,q,0)$ resp. $\delta(v,s)=(r,w,q,1)$.

Dále se zdá býti rozumným požadovat, aby multitransducer nemohl transformovat prázdný řetězec na neprázdný řetězec a naopak, tj. multitransducer nemůže „tvořit“ něco z ničeho, ani udělat z něčeho nic.

U tohoto zařízení se nepočítá s jeho evolucí – změnu jeho činnosti lze dosáhnout pouze změnou jeho přechodové funkce. Je zřejmé, že aktivita jednotlivých stavů je na začátku výpočtu daná množinou B a dále se mění v závislosti na průběhu výpočtu, tj. v závislosti na zpracovávaných objektech a jejich pořadí (nezapomínejme, že jednotlivé automaty pracují asynchronně).

3. Vtělení interaktivního transduceru

V práci [16] později i v [19] bylo navrženo rozšíření konečných kognitivních transducerů na tzv. konečné kognitivní agenty, kteří se liší od dříve zmíněných zařízení tím, že součástí modelu jsou i tzv. percepčně-motorické jednotky, které umožňují interakci agenta

s okolím a představují tak jakési první přiblížení agentova těla. Tento model však nemá žádné mechanismy pro modelování „růstu“ těla, ani agentovy evoluce, jak „tělesné“ anebo „duševní“, které se zdají nevyhnutelné, pokud chceme modelovat vznik a vývoj jednoduchých forem života. V tomto pojetí mentální evoluce znamená samozřejmě změnu mechanismů, řídicích zpracování dat agentem – tj. změnu přechodové funkce odpovídajícího transduceru. To jsme částečně dosáhli v předchozím modelu multitransduceru pomocí přídatného mechanismu aktivace stavů. V dalším k tomu přibude ještě mutace – náhodná změna přechodové funkce multitransduceru. Zbývá vyřešit problematiku „vtělení“ (ve smyslu teorie tzv. vtělené umělé inteligence – viz. např. [13]) programovatelného kognitivního transduceru (či agenta) do vhodné tělesné schránky, abychom mohli zachytit i aspekty tělesného vývoje. Tato schránka tudíž musí plnit několik základní úkolů: musí chránit řídicí a informační mechanismy agenta před vlivy vnějšího světa, musí být „nosičem“ percepčně-motorických jednotek agenta, a v neposlední řadě musí umožnit vývoj (zejména růst) a dělení (množení) agenta.

Jako model nejjednoduššího „těla“ budeme uvažovat (třírozměrnou) *membránu*. Budeme ji modelovat jako třírozměrnou strukturu, pozůstávající ze speciálních objektů, tzv. *destiček*. Tyto destičky mají za jistých fyzikálně-chemických okolností vlastnost „samoorganizace“ do kulovité struktury, kterou budeme nazývat membrána. To znamená, že za příznivých okolností jakýkoliv shluk takových destiček, které jsou dostatečně blízké jedna druhé, se spontánně uspořádá do kulovité struktury a navíc, jestli se v blízkosti takové membrány nachází jedna anebo více dalších destiček, tak se tyto inkorporují do existující membrány. Za jiných okolností, např. pokud se membrána „naplní k prasknutí“, se membrána „roztrhne“ – rozdělí na dvě přibližně stejné části, které se opět spontánně uzavrou do kulovitého tvaru. Původní obsah membrány se přitom také rozdělí přibližně napůl, malé ztráty obsahu jsou však možné a pravděpodobné. Vlastnost růstu a dělení membrány lze vidět jako důsledek fyzikálních a chemických zákonů, působících v daném prostředí na materiál membrány. Rychlost růstu membrány bude (kromě jiného) záviset na přísunu materiálu (destiček). V dalším uvidíme, že destičky mohou vznikat i za přispění automatů, jako výsledek zpracování jiných objektů, které se nacházejí v prostředí a nemají vlastnost samoorganizace. V membráně se dále mohou nacházet percepční a motorické jednotky agenta. Materiál na jejich konstrukci dodávají automaty umístěné uvnitř membrány. Přítomnost těchto jednotek a jejich počet závisí od přísunu materiálu, ze kterých jsou budovány.

Všechny automaty multitransduceru se řídí programem, který se po vzoru DNA (resp. prokaryotických bakterií) nachází uvnitř membrány a má i podobný tvar ve formě (svinuté) konečné pásky. Z hlediska multitransduceru je to objekt jako kterýkoliv jiný a proto se může stát předmětem zpracování některého z automatů, které tvoří multitransducer. Je to vlastně lineární zápis specifikace multitransduceru. V prvním přiblížení si tento zápis lze představit jako posloupnost instrukcí „co dělat, když na vstupu nějakého automatu ve stavu s je objekt v “. Tento objekt je potom zpracován způsobem odpovídajícím přechodové funkci. Jednotlivé segmenty zápisu přechodové funkce odpovídají zápisu jedné instrukce v přechodové funkci, tj. zobrazení tvaru $I \times S \rightarrow S \times O \times S \times \{0,1\}$ pro konkrétní hodnoty parametrů. Na program lze nahlížet jako na „návod“ (instrukce) pro „výrobu“ různých objektů, které lze dále použít jako jakési prefabrikáty např. při stavbě membrány anebo percepčních a motorické orgánů. Dále program obsahuje instrukce pro započítí jejich konstrukce, případně pro jejich zpětné rozložení. To se děje pomocí aktivace stavů do té doby nečinných automatů anebo naopak pomocí deaktivace stavů činných automatů. Pozornosti hodným a důležitým je případ, když se aktivuje automat, jehož úkolem je vytvořit kopii objektu, který reprezentuje program

multitransduceru. Takový automat má dva vstupy – jedním čte stávající program a druhým přijímá „materiál“ (objekty), ze kterého na výstupu konstruuje kopii programu. Tj. jedná se o jednoduchý kopírovací automat. Zdá se, že někteří autoři se domnívají, že pro modelování sebe reprodukce živých organismů je nutné, aby součástí genetického popisu organismu byl kromě popisu reprodukčního zařízení pro kopírování genetické informace také „návod“, jak se „sebereprodukovat“ (to je např. esence známého von Neumannova výsledku o sebe reprodukujících se automatech). V naší analogii by to znamenalo, že součástí multitransduceru by měl být i popis, jak sestavit membránu a kdy a jak ji rozdělit, jak zabezpečit, aby v každé kopii organismu byla právě jedna kopie genetické informace, atd. To je sice myslitelné, ale pro všechny další účely pravděpodobně zbytečné a z hlediska evoluce dost složité a tudíž málo pravděpodobné. Mechanismus kopírování DNA v reálných buňkách ukazuje, že pro kopírování genetické informace stačí právě popsání kopírovacího automatu.

Při své činnosti multitransducer „spoléhá“ na samoorganizační schopnost některých objektů, jako jsme již viděli v případě destiček. Tato vlastnost objektů je inspirována vlastnostmi některých organických molekul, které se za jistých specifických fyzikálně-chemických podmínek spontánně organizují do větších, složitějších struktur, přičemž volná energie systému klesá. Molekuly přitom není potřebné nijak zvláště instruovat, jakým způsobem to mají činit. V našem případě budeme takové vlastnosti vybraných objektů pouze postulovat, podobně jako podmínky, za kterých tak činí. Nebudeme se nutně starat o to, zda takové objekty ve skutečnosti existují a jak fungují. Pokud tedy multitransducer vygeneruje z jiných objektů objekty, ze kterých je možné „poskládat“ pomocí samoorganizace nějaký percepční či motorický orgán, tak takový orgán se z těchto nově vygenerovaných objektů vytvoří „sám od sebe“ a bude umístěn v prostředí, které je *de facto* určeno vlastnostmi sebesestavovacích objektů, ze kterých se orgán skládá (např. stane se součástí membrány, jako v případě percepčních či lokomočních orgánů). Řízení procesu generace jednotlivých objektů, tj. rozhodnutí, kdy (za jakých okolností) co (jaké orgány či jejich součástky) generovat a z jakých objektů má na starosti multitransducer, popsání výše zmíněným programem, který tvoří zvláštní objekt uvnitř membrány.

Pravděpodobně bude na místě pár slov o reálnosti navrhovaných výpočetních a nevýpočetních mechanismů vtělených multitransducerů. Výpočetní mechanismy jsou vlastně podstatným zjednodušením tzv. centrálního dogmatu molekulární biologie [4], totiž že každý gen v DNA molekule nese informaci potřebnou pro konstrukci jednoho proteinu, který ve formě enzymu řídí jednu chemickou reakci v rámci buňky. V našem modelu genu odpovídá jeden příkaz přechodové funkce multitransduceru a příslušný automat odpovídá zmíněnému proteinu. Chemickým sloučeninám (biomolekulám) odpovídají objekty, reakcím odpovídají výpočty nad objekty. Objekty jsou tedy skutečnou fyzickou reprezentací dat, jsou jejich „ztělesněním“; konstrukcím složitějších objektů odpovídají výpočty anebo samoorganizace. Dále, v reálném životě je DNA molekula „navinutá“ na skupinu proteinů, tzv. histonů. Přepínání genů se ve skutečných buňkách pravděpodobně děje pomocí chemických přepínačů (viz tzv. operonovou hypotézu), které „zakryjí“ část DNA s požadovaným genem a tím jej vyřadí z činnosti. Komunikace v rámci buňky funguje pomocí chemických či jiných signálů (změnou fyzikálních vlastností molekul indukovanými změnou jejich tvaru). Samoorganizace vybraných typů biomolekul (např. amfifilních molekul do tvaru třírozměrných obálek – viz např. [3]) je skutečností, která je experimentálně ověřena. Prostupování vstupních objektů přes membránu, případně přes specializované kanály či póry, nacházející se v membráně, také odpovídá skutečnosti, v praxi je to důsledek brownova pohybu molekul a chemicko-fyzikálních vlastností zúčastněných molekul (jak v membráně, tak i těch prostupujících). Více o

samoorganizaci, její biochemické realizaci a jejím využití viz např. v přehledovém článku [20]. V teorii se zatím uvažují samosestavovací objekty čtvercového tvaru, ze kterých však nelze poskládat orgány, které potřebujeme; nicméně již tyto výsledky ukazují, že samosestavování objektů je v principu možné a že výsledný tvar složitějších objektů závisí na tvaru součástí, ze kterých se komplex skládá.

Nyní, když máme dostatečnou představu o mechanismech jednotlivých součástí vtěleného multitransduceru, můžeme přikročit k popisu tzv. bakteroidu, který představuje jednoduchý umělý živý organizmus, žijící v oceánu objektů, ze kterých lze sestavit složitější objekty se samoorganizačními schopnostmi.

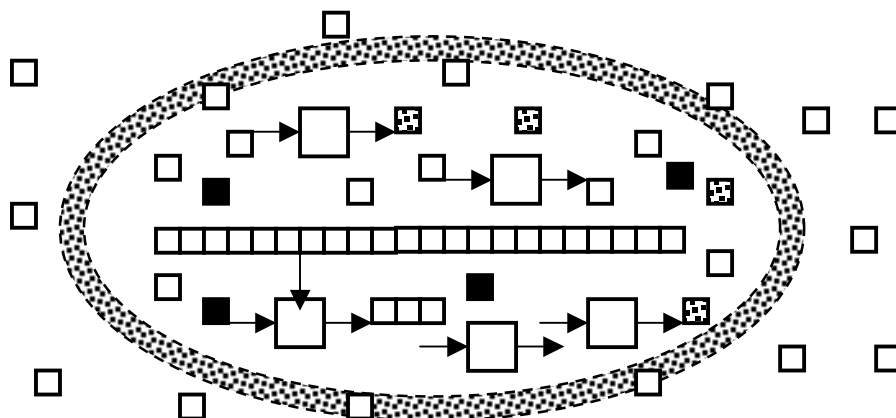
4. Bakteroid

Bakteroid je generický název pro dvojici, skládající se z multitransduceru a membrány, která multitransducer obklopuje. Podobně jako v případě skutečných bakterií se jednotlivé druhy bakteroidů liší svojí „výbavou“ – a to jak genetickou (zhruba bychom mohli říci, složitostí multitransduceru, který bakteroid řídí, anebo, řečeno jazykem evoluční biologie, genotypem), tak i fyzickou (fenotypem, tj. jak bakteroid „vypadá“, jakou má membránu, jaké má vlastnosti tato membrána, jaké má orgány, kde umístěné apod.). Nejjednodušší bakteroid v tomto směru je tzv. *protobakteroid*, který nemá žádné percepční a motorické orgány, skládá se pouze z membrány, která uvnitř obsahuje objekt, ve kterém je zakódován řídicí program multitransduceru (viz schéma bakteroidu na obr. 1). Tomuto objektu budeme říkat jednoduše páska. Důvod pro existenci takového seskupení je v tom, že multitransducer syntetizuje z objektů, které jsou vně protobakteroidu a volně prostupují membránou, destičky, které vznikají uvnitř membrány a samoorganizací se inkorporují do již existující membrány. Příslušné automaty budeme nazývat *destičkové automaty*. Tím způsobem membrána roste. Současně s tím probíhá reprodukce programu multitransduceru, pomocí *kopírovacího automatu*, také z objektů, které volně prostupují membránou. Jak již bylo zmíněno dříve, předpokládejme, že původní i nová páska jsou svinuty do dvou kulovitých klubek. Dále předpokládejme, že membrána obepínající obě klubka není napnutá, ale že „vyrostla“ více než by stačilo k těsnému obepnutí klubek. To znamená, že výsledný tvar celého bakteroidu se čím dále tím více začne podobat jakési třírozměrné osmičce a v okamžiku, kdy se vnitřní část membrány mezi oběma klubky pásky dostatečně přiblíží, zapracuje samoorganizační schopnost destiček a protobakteroid se rozdělí na dva více méně stejné protobakteroidy (viz. obr. 2)³.

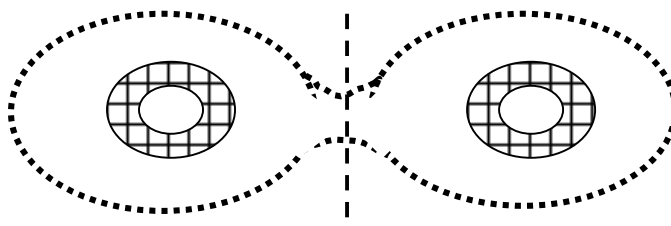
I z tohoto neformálního popisu je vidět, že protobakteroid potřebuje pro svou existenci dle výše zmíněných představ řadu objektů s různými vlastnostmi, z nichž musí být schopen generovat další objekty (např. destičky) ještě s jinými vlastnostmi, zejména

³ V práci [7] se navrhuje jiný mechanismus dělení membrány inspirovaný reálnými bakteriemi: autoři předpokládají, že původní i zkopírovaná páska s genetickou informací se „připevní“ na protilehlé strany membrány a membrána se rozdělí tak, aby se při dělení obě pásky rozmotaly a oddělily a současně aby každá zůstala ve „své“ půlce membrány, která se samozřejmě současně uzavírá. Toto nedovedeme v našem modelu napodobit – problém je, jak určit „protilehlé“ strany pomocí mechanismů konečných automatů (anebo jinak, pomocí nějakých fyzikálních vlastností) a zabezpečit dělení membrány právě mezi takto orientovanými póly.

samoorganizačními, které se opět organizují do dalších struktur (např. membrány), které mají ještě jiné vlastnosti, než její součásti a podléhají fyzikálním zákonům.



Obr. 1 Schéma bakteroidu



Obr. 2 Dělení bakteroidu

Pro úplnost se pokusme nyní ještě blíže a formálněji specifikovat náš protobakteroid, částečně ve smyslu definice multitransduceru a s přihlédnutím na další vlastnosti jednotlivých objektů, nutné pro činnost protobakteroidu dle výše zmíněných představ.

Vstupní objekty: to jsou objekty, které se budou nacházet vně i uvnitř těla bakteroidu, ale nejsou součástí membrány; z nich se budou generovat *destičky* a z nich se bude sestavovat *kopie pásky* s řídicím programem transduceru.

x,y,z: objekty pro budování destiček a elementů pásky

Výstupní objekty: jsou výsledkem zpracování vstupních objektů automaty; z výstupních objektů typu *destička* se buduje samoorganizační objekt *membrána*; z dalších se buduje kopie pásky; membrána je prostupná pro všechny vstupní objekty směrem dovnitř a neprostupná pro výstupní objekty (směrem ven).

a,g,t,c: objekty, ze kterých se skládá páska;

d: destička

Multitransducer: skládá se ze tří automatů: *destičkového*, který syntetizuje destičky d z objektů x a y, *syntetizačního* automatu, který syntetizuje elementy a,g,t,c, z objektů z,v,w a *kopírovacího* automatu, který čte objekt p, skládající se z objektů a,g,t,c, a syntetizuje jeho kopii z objektů a,g,t,c.

p: páska s popisem přechodové funkce multitransduceru

Přechodová funkce: od jejího popisu upustíme, zatím není příliš zajímavá, protože nevyžaduje žádnou změnu aktivace počátečních stavů – destičkový automat i syntetizační automat jsou jednokrokové, stále se nacházejí v počátečním stavu. Cyklus kopírovacího

automatu musí ovšem proběhnout pouze jednou; aby se páska nekopírovala podruhé, musí být dosažení okamžiku dělení synchronizováno s ukončením procesu kopírování. Za tím účelem nemá protobakterie žádné zvláštní mechanismy, tato vlastnost musí být zaručena relativními rychlostmi všech zúčastněných automatů.

Další předpoklady: páska je svinutá do klubička, kopírování je ukončeno těsně předtím, než se membrána rozdělí výše popsaným mechanismem

Výše uvedené představy o fungování protobakteroidu jsou zajisté zcela jiné než představy ryze výpočetní a zde se rýsuje odpověď na Brooksovu otázku z citátu v úvodu této práce. Tyto představy však nepřekračují hranice našeho vědění o vlastnostech biomolekul a naopak jsou plně v souladu se současnými představami o prvních formách života a o fungování živých buněk (viz např. [5][14]) a téměř v dosahu možností současných technologií [2].

5. Protobakteroid a minimální život

Dále se budeme věnovat otázce, nakolik protobakteroid odpovídá současným představám o tzv. *minimálním životě*. Je to představa živého organismu, který má minimální množinu vlastností takových, abychom ho mohli považovat za živý systém. Odebráním kterékoliv z těchto vlastností přestane organismu odpovídat našim představám o životě. Zdá se, že v současné době převažující operační definice vyžaduje splnění dvou podmínek [14]: příslušný systém musí mít v daném prostředí schopnost autonomní replikace a musí podléhat pravidlům darwinovské evoluce. Autonomní replikace znamená nepřetržitý růst a množení systému; obě tyto akce jsou závislé pouze na vstupních jednoduchých molekulách a nezávislé na produktech jiných forem života. Darwinovská evoluce vyžaduje, aby se genetická variace (mutace) projevovala ve fenotypu organismu a jeho chování a tím ovlivňovala přežívání a reprodukci organismu.

Nuže, z návrhu protobakteroidu je vidět, že tento v daném prostředí, při dostatečném přísunu vstupních objektů, přežívá a množí se. Přitom jsme předpokládali, že vstupní objekty jsou jednodušší objekty, než výstupní, protože posledně zmíněné jsou „vypočítány“ z prve zmíněných. Darwinovskou evoluci zabezpečíme „malou úpravou“ přepisovacího (kopírovacího) mechanismu: připustíme (resp. dokonce „zařídíme“), aby tento mechanismus byl nespolehlivý. To znamená, při kopírování mohou nastávat chyby nejenom „přepisovacího“ typu, ale třeba takové, že se některá část programu vůbec nekopíruje, anebo se naopak zkopíruje dvakrát. Tím samozřejmě dochází ke „genetické variaci“ – program multitransduceru mutuje. Některé změny se vůbec neprojeví (třeba duplikace některých částí programu), jiné se projeví třeba okamžitě nefunkčností protobakteroidu anebo naopak jeho zlepšenou funkčností. Právě posledně zmíněné změny jsou ty změny, které znamenají „posunutí vývoje“. Je zajímavé si uvědomit, že podobně jako u živých organismů, i v našem modelu zásadnější pokrok nemůže nastat často za sebou, s malými časovými odstupy. Je to proto, že některé změny se musejí „naakumulovat“. Např. se některá část kódu duplikuje a zatím se v životě protobakteroidu neprojevuje. Dále může proběhnout mutace, která tuto část kódu vyřadí vůbec z činnosti (přepnutím počátečního stavu do pasivního stavu). Dále v této části mohou proběhnout další náhodné mutace a když se náhodou opět tato část kódu „zapne“, začne se protobakteroid chovat v určité situaci jinak, třeba efektivněji, než jeho předchůdci. To znamená, že kromě jakési pomalé, „plíživé evoluce“, postupující v malých, sotva pozorovatelných krůčcích, existuje i radikální evoluce propukající s delšími časovými prodlevami, ale projevující se zásadnějšími změnami jak ve fenotypu, tak i v chování

protobakteroidu. Zdá se, že mechanismus, který podporuje radikální inovaci způsobem popsaným výše, lze „zabudovat“ do kopírovacího automatu, takže vlastně může vzniknout „sám od sebe“ pomocí evoluce.

V dalším si popíšeme příklad bakteroidu, který je výsledkem takové radikálnější evoluce protobakteroidu uvažovaného v 4. části.

6. Selektivní bakteroid

Protobakteroid, byť vykazoval známky minimálního života, byl poměrně jednoduchý. Hlavní „oživovací trik“ spočíval v tom, že jeho „tělesný vývoj“, představovaný růstem membrány a kopírovacím mechanismem programu, byl řízen multitransducerem. Tím se dosáhlo, že „přežití“ protobakterie záviselo na souhře mezi programem, „kvalitou“ tohoto programu, přísunem materiálu na výrobu destiček a částí kódu a jejich výrobou. Ostatní věci zabezpečila samoorganizace a fyzikální zákony. Pokud potřebná souhra nebyla dosažena, protobakterie jednoduše nefungovala a nemnožila se. V opačném případě se množila a pokud mutací vzniklo nějaké vylepšení, které rezultovalo např. v rychlejší růst protobakteroidu, tak takto modernizovaný bakteroid po čase v populaci protobakteroidů převládnul. Pro svou existenci protobakteroid nepotřeboval informace o vnějším prostředí. V dalším uvedeme příklad tzv. selektivního bakteroidu, který už bude využívat informace o prostředí a tím dosáhne v porovnání s protobakteroidem evoluční výhodu rychlejšího růstu. Tento příklad současně poslouží i jako ilustrace účinnosti mechanismu průběžné aktivace a deaktivace jednotlivých stavů, čím se zvýrazní i přínos tohoto mechanismu pro studium genetické expresivity bakteroidů.

Uvažujme bakteroid, který se nachází v prostředí nutrientů, které jsou modelovány objekty typu A a B. Bakteroid může zužitkovat oba typy potravy, avšak potrava typu A je mnohem výživnější než potrava typu B – např. bakteroid může vyprodukovat z potravy typu A několikrát více destiček a částí kódu než z potravy typu B. Protobakteroid, popsaný ve 4. části – řekněme mu $\mathcal{P}$ – může fungovat v takovém prostředí beze změny – nemusí nijak rozlišovat mezi jednotlivými nutrieny. Avšak bakteroid, který by v prostředí preferoval živiny typu A (pokud by těchto byl dostatek), by zajisté byl evolučně úspěšnější než $\mathcal{P}$. Bakteroid takového typu budeme označovat jako selektivní bakteroid.

Selektivní bakteroid – označme jej $\mathcal{S}$ – tedy musí nějakým způsobem zjistit, jestli je v současné době v okolí dostatek anebo nedostatek živin typu A a zařídit se podle toho. Jinými slovy, jestli je koncentrace živin typu A dostatečná, měl by $\mathcal{S}$ přestat přijímat živiny typu B, v opačném případě by měl akceptovat jakýkoliv typ potravy.

Problém s návrhem takového bakteroidu je v tom, že pro běžný konečný automat přijímající řetězce symbolů nad abecedou $\{a,b\}$ představuje problém rozeznat, jestli přijímá řetězce, ve kterých převažují a-čka nad b-čky. Vpravdě, takový jazyk není regulární a nelze je konečným automatem rozeznat. Není ale problém naprogramovat jeden automat $\mathcal{A}$ tak, aby akceptoval jenom a-čka, a jiný automat $\mathcal{B}$ tak, aby akceptoval a-čka i b-čka. Zřejmě $\mathcal{A}$ může být pořád aktivní (není rozumné odmítat a-čka, pokud se nějaká vyskytnou). Co má smysl je vypínat $\mathcal{B}$ v případě, že je v okolí dostatek a-ček. Uvažujme proto ještě další typ automatu – automat $\mathcal{R}$ zvaný receptor. Tento automat nedělá nic jiného než že „zkoumá“ objekt na svém vstupu a pokud je to objekt a, tak $\mathcal{R}$ „vypne“ automat $\mathcal{B}$ tím způsobem, že přepne počáteční stav automatu $\mathcal{B}$ na pasivní. Pokud

je na vstupu b , aktivizuje automat $\mathcal{B}$. Poté $\mathcal{R}$ objekt „pustí“ bez toho, že by ho nějak změnil, a zkoumá další náhodný objekt plující kolem.

Předpokládejme, že kódy automatů $\mathcal{A}$, $\mathcal{B}$, a $\mathcal{R}$ jsou součástí genetického kódu (řídícího programu) bakteroidu $\mathcal{S}$ a že kromě toho jsou zde přítomny i kódy pro kopírování programu, výrobu deštiček a součástí genetického kódu podobně, jako tomu bylo v případě protobakteroidu $\mathcal{P}$. Automaty $\mathcal{A}$, a $\mathcal{B}$ budeme nazývat póry, anebo kanály a budeme předpokládat, že společně s receptory $\mathcal{R}$ jsou umístěné v membráně. Jak vzniknou a jak se dostanou do membrány? Zde opět budeme spoléhat na samoorganizaci – $\mathcal{S}$ bude obsahovat automaty na výroby příslušných součástí, které se dále v membráně samoorganizací uspořádají do tvaru pórů a receptorů. Je to pouze jakýsi jiný typ deštiček, ze kterých byla konstruována původní membrána. Počet pórů a receptorů je dán množstvím živin (objektů typu A a B), které bakteroid „pozře“ a případně velikostí povrchu membrány. To vše jsou veličiny, které výpočetní mechanismus bakteroidu nesleduje (a ani nemůže sledovat, protože je to konečný mechanismus a nechceme, aby jeho velikost závisela na hodnotě těchto veličin).

Výsledný systém, bakteroid $\mathcal{S}$, funguje následujícím způsobem. Póry typu A procházejí selektivně pouze objekty typu A , pokud se takové v blízkosti póru vyskytnou. Receptory asynchronně zkoumají objekty na svém vstupu a aktivují či deaktivují kanály typu B . Je důležité si uvědomit, že každý receptor zavře/otevře všechny kanály a že receptory pracují asynchronně. Takže, pokud je v okolí protobakteroidu v daném časovém intervalu přítomna většina a -ček, tak většina receptorů zavře kanály typu B a tyto zůstanou prakticky po celý interval uzavřeny, protože ojedinělé signály k jejich otevření (od receptorů, které „viděly“ b -čka) jsou okamžitě „přepsány“ signály pro zavření kanálů. V opačném případě převáží signály k otevření kanálů typu B a bakteroid „akceptuje“ potravu obou typů. Nemá cenu se zabývat podrobným „vyladěním“ celého systému – kdy nastane přepnutí z jednoho režimu do druhého – to zřejmě závisí od mnoha parametrů. Důležité ale je, že o to vše se postará evoluce, ve které zvítězí protobakterie s nejlépe fungující selektivní strategií. Všimneme si, že bakteroid $\mathcal{S}$ vykazuje známky kognitivního chování – rozlišuje mezi dobrým a nedobрым.

Právě popsaný selektivní mechanismus chování bakteroidu je základem mnoha dalších typu chování, která pozorujeme u skutečných bakterií (např. u *Escherichia coli*). Uvažujme např. bakteroid $\mathcal{M}$ opatřený flagelárním motorem. Tento motor lze zapnout tak, aby způsobil pohyb bakteroidu směrem vpřed, anebo tak, aby způsobil náhodné natočení směru plavby bakteroidu v třírozměrném moři objektů. Konstrukci stroje opět přenecháme samoorganizaci, bakteroid musí pro něj připravit pouze vhodné součásti. Zájemce o konstrukci a evoluci takových motorů ze sekrečních pórů odkazujeme např. na práci [10]. Pro naše další úvahy bude důležitý pouze předpoklad, že motor je vlastně elektromotorem, který čerpá energii z rozdílů elektrických potenciálů vně a uvnitř membrány, tj. spotřeba motoru nesouvisí s přijímanou potravou. Platí ale, že množství potravy zkonsumované bakteroidem roste s délkou dráhy, kterou bakteroid v prostředí bohatém na potravu překoná. Budeme tedy chtít, aby se $\mathcal{M}$ „pásl“ na místech s dostatkem stravy co nejdéle a v případě, že z této oblasti vyjede, aby se do ní opět vrátil.

Uvažujme proto podobný selektivní mechanismus jako v případě bakteroidu $\mathcal{S}$, pouze s tím rozdílem, že strava typu A je „dobrá“, strava typu B „nechutná“. Naprogramujeme řízení $\mathcal{M}$ takto: pokud receptory hlásí „dobrá strava“, nechajme bakteroid „pást se“ a současně plout jedním směrem. Pokud bakteroid vyjede ze zóny dobré potravy, což jeho

receptory zjistí, přepne na okamžik svůj motor do druhého režimu, co způsobí jeho náhodné natočení v prostoru a vzápětí přepne opět, na krátkou dobu, na „plnou parou vpřed“. Pokud se dostane opět do zóny dobré stravy, tak pokračuje v plavbě daným směrem, jinak se opět pokusí náhodně natočit a krátkým posunem „najít“ oblast dobré potravy, atd. Zbývá pouze vyřešit problém, jak bakteroid pozná, že uplynul „okamžik“ anebo „krátká doba“. Za tím účelem využívá jakési „minutky“ – specializované automaty, jejichž činnost trvá potřebnou dobu. Příslušný automat nejprve „vypočítá“ tj. zkonstruuje nějaký objekt a vzápětí jiný automat ho začne „rozebírat“. Velikost objektů je volena tak, aby celý proces konstrukce a dekonstrukce trval potřebnou dobu. Samozřejmě, v rámci konečných automatů není na místě používat čítače, proto je velikost objektů, které minutkový automat konstruuje, daná externě – velikostí prostoru, který objekt vyplní. Toto řešení, jakkoliv uměle vypadá, odpovídá do jisté míry skutečnosti (viz. např. [8]). Pro zajímavost, mechanismus zapínání a vypínání flagelárního motoru u bakterie *Escherichia coli* pracuje přibližně takto. Na motoru je jakýsi přepínač z jednoho režimu do druhého, který lze v dané poloze fixovat tím způsobem, že se „překryje“ jistým množstvím látky, které je vymezené, zdá se, ohraničeným prostorem jistého objemu v okolí přepínače. Tato látka zamezí přístupu signálů k přepnutí přepínače. Látka se však postupně rozkládá a po nějaké době (řádově vteřiny až minuty) je opět přepínač volný. Právě popsané chování bakteroidu, zvané chemotaxe, zabezpečí, že bakteroid nevyjíždí často z oblasti dobré potravy a pokud se tak stane, má šanci se tam opět vrátit. Navenek se takové chování jeví jako inteligentní, avšak, pokud známe mechanismus v pozadí tohoto jevu – je to intelligence? Anebo opačně – není v pozadí jakékoliv domnělé intelligence mechanismus, který neznáme, a proto se nám jeví příslušné chování jako inteligentní?

Abychom si učinili představu o složitosti reálných bakterií, tyto se skládají z desítek miliónů biologických molekul, mají v průměru kolem 2000 genů, v jednodušších případech několik stovek genů, převyšující 300 genů. *Escherichia coli* má 4288 genů, používá asi 20 různých „přenosových tras“ (signal pathways) pro přenos informací uvnitř buňky. Dále má na výběr cca 30 typů chemoreceptorů, které nejsou vždy všechny přítomny; bakterie je ovšem může dle potřeby „nechat narůst“. V průměru se na povrchu membrány nachází přibližně 5 typů chemoreceptorů v celkovém množství kolem 1500 [11].

7. Závěr

V článku jsme navrhli abstraktní ztělesněný hybridní systém – bakteroid, který spojuje výpočetní mechanismy s mechanismy samoorganizace. Pokud je nám známo, je to první systém svého druhu v oblasti ALife, o kterém lze formálně dokázat, že ve vhodně navrženém vesmíru, který je ovládán zákony, které dovolují existenci a konstrukci objektů s požadovanou samoorganizační vlastností, vykazuje známky minimálního života. Pochopení a matematické modelování sebesestavovacích procesů je v současné době předmětem intenzivního výzkumu (viz např. práci [17] která navazuje na stávající práci). Popisný rámec bakteroidu umožňuje formulaci a řešení problémů (jakými je např. problém nutriční selektivity či chemotaxe za potravou), které v jiných formalizmech nelze přirozeně formulovat ani řešit. Problémů inspirovaných schopnostmi reálných bakterií existuje celá řada (viz např. [8]) a v příštím období bude jistě zajímavé pokusit se nalézt jejich řešení. Další zajímavou výzvou do budoucna je rozšíření záběr našeho modelování z jednobuněčných organizmů na mnohobuněčné. Podobné projekty mohou vrhnout nové světlo na složitost (ve výpočetním smyslu toho slova) konstrukce a epistemické práce bakteroidů a přeneseně i na syntezu *de novo* reálných umělých organismů. Craig Venter a

Hamilton Smith začali svůj projekt, zmíněný v úvodu této práce, také počítačovými simulacemi; jejich cílem bylo zjistit, jaký genetický materiál by mohl být použitelný pro minimální život.

Literatura

- [1] Adleman, L.: *Toward a mathematical theory of self-assembly*. Tech. Rep. 00-722, Dept. of Computer Science, University of Southern California, 2000.
- [2] Benenson, Y., Paz-Elizur, T., Adar, R., Keinan, E., Livneh, Z., Shapiro, E.: Programmable and autonomous computing machines made of biomolecules. *Nature* **414**, 2001, s. 430-434.
- [3] Brooks, R.: Beyond Computation. *EDGE* **132**, June (2002) (http://www.edge.org/3rd_culture/brooks_beyond/beyond_index.html).
- [4] *Central dogma of molecular biology*: viz např. <http://www.bartleby.com/59/21/centraldogma.html>.
- [5] Hanczyc, M. M., Fukijawa, S. M., Szostak, J. W.: Experimental Models of Primitive Cellular Compartments: *Encapsulation, Growth, and Division*. *Science* **302**, 2003, s. 618-622.
- [6] Hutton, T. J.: Simulating Evolution's First Steps. In: *Advances in Artificial Life. Proceedings of the 7th European Conference, ECAL2003*. Banzhaf W., Christaller T., Dittrich P., Kim J.T., Ziegler J. (Eds.), Dortmund, Germany, 14-17th September 2003. LNAI 2801, Springer 2003, 51-58.
- [7] Kováč, L., Nosek, J., Tomáška, L.: An Overlooked Riddle of Life's Origin: Energy Dependent Nucleic Acid Unzipping. *J. Mol. Evol.* In Press 2004.
- [8] Lengeler, J. W., Müller, B. S., di Primio, F.: Neubewertung kognitiver Leistungen im Lichte der Fähigkeiten einzelliger Lebewesen. *Kognitionswissenschaft* **8**, 2000, s.160-178.
- [9] Mullen, L.: Life from scratch? *Astrobiology Magazine*, 2004 (<http://www.astrobio.net/news/article319.html>).
- [10] Musgrave, I.: Evolution of the Bacterial Flagellum. In: *Why Intelligent Design Fails: A Scientific Critique of the Neocreationism*. Young, M., and Edis, T. (Eds.), Rutgers University Press, Piscataway, N.J. 2005, 240 s..
- [11] Müller, B.S., di Primio, F., Lengeler, J.W.: Contributions of Minimal Cognition to Flexibility. In: *SCI 2001, Proceedings of the 5th World Multi-Conference on Systemics, Cybernetics and Informatics*. N. Callaos, W. Badawy and S. Bozinovski (Eds), Volume XV, Industrial Systems: Part II, International Institute of Informatics and Systemics 2001, 93-98.
- [12] Paun, G., Rozenberg, G.: A Guide to Membrane Computing. *Theoretical Computer Science* **287**, 2002, s. 73–100.
- [13] Pfeifer, R., Scheier, C.: *Understanding Intelligence*. The MIT Press, Cambridge, 1999, 697 s.
- [14] Szostak, J.W., Bartel, D.P., Luisi, P.L.: Synthesizing Life. *Nature* **409**, 2001, s. 389-390.
- [15] Wiedermann, J., van Leeuwen, J.: The Emergent Computational Potential of Evolving Artificial Living Systems. *Ai Communications* **15**, 4 , 2002, s. 205-216.
- [16] Wiedermann, J.: Building a Bridge between Mirror Neurons and Theory of Embodied Cognition. In: *Proc. SOFSEM 2004*, P. van Emde Boas et al. (Eds.), LNCS 2932, Springer, 2004, s.361-372.

- [17] Wiedermann, J.: Globular Universe and Autopoietic Automata: A Framework for Artificial Life. In: *Advances in Artificial Life*. (Ed.: Capcarrere M., Freitas A.A., Bentley P.J., Johnson C.G., Timmis J.) - Berlin, LNCS Vol. 3630, Springer-Verlag 2005, s. 21-30
- [18] Wiedermann, J.: Computing by Self-Reproduction: Autopoietic Automata. In: *Unconventional Computation*. (Ed.: Calude C.S., Dinnen M.J., Paun G., Perez-Jimenez M.J., Rozenberg G.) - Berlin, LNCS 3699, Springer-Verlag 2005, s. 224-236
- [19] Wiedermann, J.: HUGO: A Cognitive Architecture with an Incorporated World Model. *Proc. of the European Conference on Complex Systems ECCS'06*, Saïd Business School, Oxford University, 2006
- [20] Winfree, E.: DNA Computing by Self-Assembly. *National Academy of Engineering Website, The Bridge* **33**, 4, 2003, s. 31-38.
- [21] Zimmer, C.: Building a Virtual Microbe, Gene by Gene by Gene. *The New York Times*, August 16, 2005