Fakulta matematiky, fyziky a informatiky
Univerzity Komenského v Bratislave

Mgr. Tomáš Záthurecký

Autoreferát dizertačnej práce

Irregular and Asynchronous Cellular Automata

(Nepravidelné a Asynchrónne bunkové automaty)

na získanie vedecko-akademickej hodnosti philosophiæ doctor
v odbore doktorandského štúdia: 9.2.1. informatika

Bratislava 2010

Dizertačná práca bola vypracovaná v internej forme doktorandského štúdia na Katedre informatiky Fakulty matematiky, fyziky a informatiky Univerzity Komenského v Bratislave.

Predkladateľ: Mgr. Tomáš Záthurecký
Katedra informatiky
Fakulty matematiky, fyziky a informatiky
Univerzity Komenského
Mlynská dolina
842 48 Bratislava

Školiteľ: Prof. RNDr. Branislav Rovan, PhD.
Katedra informatiky FMFI UK
Bratislava

Oponenti: ....

....

....

....

....

Obhajoba dizertačnej práce sa koná dňa .... o .... h.
pred komisiou pre obhajobu dizertačnej práce v odbore doktorandského štúdia
vymenovanou predsedom odborovej komisie dňa ....

v študijnom odbore 9.2.1. informatika

na ....

Predseda odborovej komisie:
Prof. RNDr. Branislav Rovan, PhD.
Fakulta matematiky, fyziky a informatiky
Univerzity Komenského
Mlynská dolina
842 48 Bratislava

1 Úvod

V predkladanej práci definujeme nový výpočtový model: Nepravidelné a asynchrónne bunkové automaty. Bunkový automat je výpočtové zariadenie skladajúce sa z mnohých konečných automatov, ktoré môžu navzájom komunikovať. Tieto malé konečné automaty sa nazývajú bunky. Každá bunka je riadená programom, ktorý hovorí, v akom stave bude bunka v ďalšom kroku výpočtu v závislosti od jej stavu a stavov susedných buniek. Tento program, ktorý je pre každú bunku rovnaký voláme prechodová funkcia, alebo aj lokálne pravidlo.

Najviac sa skúmali bunkové automaty, ktoré pracujú synchrónne a v ktorých sú bunky prepojené v pravidelnej mriežke.

Bunkové automaty zaviedol John von Neumann [vN63] ako formálny model na skúmanie samoreplikujúcich sa organizmov. V [vNB66] Navrhol 29-stavový bunkový automat, v ktorom sa dajú zostrojiť samoreplikujúce sa konfigurácie. Tento jeho výsledok zjednodušil Codd v [Cod68], kde navrhol 8-stavový automat a neskôr Banks [Ban71] navrhnutím 4-stavového automatu.

Iný zaujívavý príklad bunkového automatu je Conwayova hra Life [BCG82]. Jeho motiváciou bolo navrhnúť jednoduchú sadu pravidiel na simulovanie správania makroskopických populácií. Snažil sa tieto pravidlá navrhnúť tak, aby chovanie populácie bolo ťažko predvídateľné.

Najviac skúmanou vlastnosťou bunkových automatov je ich výpočtová sila. Už von Neumannov 29-stavový automat [vNB66] je výpočtovo univerzálny. Iný prístup použil Smith [SI71a]. Ukázal, ako skonštruovať dvojrozmerný bunkový automat, ktorý by vedel simulovať ľubovoľný Turingov stroj. Taktiež sa podarilo ukázať [BCG82], že Cnowayova hra Life je výpočtovo univerzálna.

Okrem toho sa skúmali aj mnohé iné vlastnosti bunkových automatov, napríklad surjektivita globálnej prechodovej funkcie [Myh63, AP72], vzťah medzi počtom stavov a počtom susedov bunky [SI71b, But74, AG72], reverzibilita [Ric72, AP72, Kar90] a iné.

Okrem toho sa skúmali aj asynchrónne bunkové automaty, v ktorých bunky nemusia počítať synchrónne. Ukázalo sa, že napriek tomu sa bunky môžu navzájom lokálne synchronizovať [Neh04] a tak sa dajú simulovať synchrónne bunkové automaty.

Výpočtový model podobný tomu, ktorým sa zaoberá táto práca je amorfný počítač [AAC$^+$00]. Tiež sa skladá z mnohých menších zariadení (buniek), ale tie sú náhodne rozptýlené v priestore (resp. rovine) a často majú väčšiu výpočtový silu než konečné automaty. Aj o nich sa podarilo ukázať [WP09], že sú výpočtovo univerzálne.

Bunky v takomto zariadení môžu vytvoriť malé skupinky [NC97], prípadne sa zorganizovať do hierarchickej štruktúry. Tento algorimus sa dá ďalej rozšíriť na hľadanie maximálnej nezávislej množiny, alebo na hľadanie $\Delta + 1$ farbenia.

Taktiež môžu bunky spolupracovať na zavedení globálneho súradnicového systému [NSB03] pomocou odhadovania vzájomnej vzdialenosti.

Tieto konštrukcie ale vyžadujú bunky s väčšou výpočtovou silou (potrebujú si pamätať veľké čísla) a využívajú náhodný generátor, alebo (aspoň lokálne) identifikátory buniek.

V predkladanej práci skúmame výpočtovú silu pri extrémne voľných predpokladoch o prepojeniach medzi bunkami a časovaní výpočtu. V našom modeli bunky nemusia byť prepojené v pravidelnej mriežke, a výpočet nemusí byť synchrónny. Jednotlivé bunky sú konečnostavové stroje, ktoré nemajú k dispozícii ani náhodný generátor, lokálny identifikátor, a ani žiadny iný spôsob, ako rozlíšiť svojich susedov. Takéto zariadenie komunikuje s vonkajším svetom prostredníctvom jednej špeciálnej bunky, ktorá môže čítať vstupné slovo a signalizovať jeho akceptovanie. Navyše je možné pred spustením výpočtu niektoré bunky označiť špeciálnym stavom. Chceme, aby celé zariadenie dalo správnu odpoveď bez ohľadu na konkrétne prepojenia medzi bunkami a časovanie výpočtu. Hlavným objektom nášho záujmu teda nie sú automaty, ale jednotlivé bunky, z ktorých sa skladajú. Inak povedané, hlavná otázka, ktorú si chceme položiť je: „Ako navrhnúť bunku tak, aby ľubovoľné zariadenie zostavené z veľkého množstva takých buniek vyriešilo danú úlohu."

2 Ciele práce

Ciele práce sú nasledovné:

- Definovať nepravidelné a asynchrónne bunkové automaty a navrhnúť vhodné miery časovej a pamäťovej zložitosti tak, aby sa dali použiť aj pri asynchrónnom výpočte a na nepravidelných prepojeniach jednotlivých buniek.
- Navrhnúť vhodnú definíciu nepravidelnosti. Totiž ak by sme pripustili všetky možné vzájomné prepojenia jednotlivých buniek, prinieslo by to pravdepodobne mnohé komplikácie. Asi by bolo veľmi ťažké navrhnúť bunkový automat, ktorý by bol schopný správne fungovať pri ľubovoľnom mysliteľnom prepojení jeho buniek. Pritom však mnohé „patologické" prípady, ktoré by kazili výsledky sú v praxi aj v teórii nezaujímavé.
- Zistiť, či a za akých podmienok bunkové automaty s nepravidelnými prepojeniami buniek dokážu v týchto prepojeniach nájsť nejakú pravidelnosť, ktorá by sa dala využiť napríklad na simuláciu automatu s pravidelnými prepojeniami.
- Nájsť vzťah medzi nepravidelnými bunkovými automatmi a Turingovymi strojmi, prípadne iným univerzálnym výpočtovým modelom. To znamená zistiť, či sa Turingov stroj dá simulovať nepravidelným bunkovým automatom a nájsť čo najlepšie odhady zložitosi tejto simulácie.

3 Výsledky a ich význam

3.1 Definícia modelu

Lokálna prechodová schéma popisuje chovanie jednej bunky. Určuje množinu stavov, v ktorých môže bunka byť, počiatočný stav, akceptačné stavy a prechodovú funkciu. Prechodová funkcia určuje, ako sa pri jednom kroku výpočtu zmení stav bunky (v závislosti od jej predchádzajúceho stavu a stavov jej susedov).

Definícia 1 *Lokálna prechodová schéma je šestica* $(K, \Sigma, \delta, q_0, F)$*, kde* K *je konečná množina (stavov),* Σ *je abeceda neobsahujúca špeciálne symboly* \$ *a* $\#$*,* $q_0 \in K$ *je počiatočný stav,* $F \subseteq K$ *je množina akceptačných stavov a* δ *je zobrazenie z nejakej konečnej podmnožiny* $K \times N^K \times (\Sigma \cup \{\#, \$\})$ *do* $K \times \{0, 1\}$*. Navyše musí platiť* $\delta(q_0, f, \#) = (q_0, 0)$ *pre každé* f *také, že* $f(q) = 0$ *ak* $q \neq q_0$*. Kde* f *je funkcia, ktorá každému stavu priradí počet susedov danej bunky v tom stave.*

Všeobecný bunkový automat je dvojica (G, P), kde G je graf určujúci prepojenia buniek a P je lokálna prechodová schéma.

Jeden krok výpočtu takéhoto automatu vyzerá tak, že sa vyberú niektoré bunky (nazveme ich aktívne) a tieto bunky urobia svoj lokálny krok tak, že sa pozrú na stavy svojich susedov a podľa nich a svojej prechodovej funkcie zmenia svoj stav. Postupnosť výberov aktívnych buniek v jednotlivých krokoch výpočtu nazveme *aktivačná postupnosť*.

Je viac spôsobov, ako takéto zariadenie môže prijímať vstupné slovo. V našej práci uvažujeme model 1W (a jeho modifikáciu 1WP), pri ktorom má prístup na vstupnú pásku jediná bunka. Náš model je však dostatočne všeobecný, aby sa dali použiť aj iné vsupno-výstupné modely.

Keď máme dané prepojenia buniek a aktivačnú postupnosť, môžeme jednoznačne určiť, či daná lokálna prechodová schéma akceptuje dané vstupné slovo. Ak je táto odpoveď rovnaká pre všetky prípustné prepojenia buniek a všetky prípustné aktivačné postupnosti, povieme, že tá lokálna prechodová schéma sa *chová dobre*. Potom môžeme definovať jazyk, ktorý definuje (ako množinu slov, ktoré akceptuje).

Pamäťová zložitosť konkrétneho výpočtu je počet buniek, ktoré počas neho boli aspoň raz v inom stave než počiatočnom. Časová zložitosť je počet jeho „ťahov", pričom ťah je časť výpočtu, v ktorej každá bunka dostala šancu urobiť svoj lokálny krok. Zložitosť lokálnej prechodovej schémy sa potom berie ako najhorší prípad cez všetky prípustné prepojenia buniek a aktivačné postupnoti.

Význam výsledkov: Po bunkových automatoch [Cod68], asynchrónnych bunkových automatoch [Neh04] a amorfných počítačoch [AAC$^+$00, WP07, WP09] predstavuje náš model ďalšie uvoľnenie predpokladov pre spôsob práce podobných výpočtových modelov. Umožní tak lepšie modelovať niektoré biologické a fyzikálne systémy, ktoré často netvoria previdelnú mriežku, či podobnú štruktúru. Náš model taktiež môže poslúžiť ako teoretický základ pre oblasť nanotechnológií, kde požadujeme, aby jednotlivé bunky boli extrémne jednoduché a ich uloženie do nejakej pravidelnej štruktúry je obtiažne alebo nemožné.

3.2 Definícia nepravidelnosti

Najjednoduchšia možnosť je pripustiť všetky možné prepojenia, ktoré majú ohraničený stupeň (to je zrejme nutná podmienka na to, aby takéto zariadenie s konečnými bunkami fungovalo).

Definícia 2 *Nech* d *je kladné celé číslo. Trieda* prepojení so stupňom najviac d *je množina* $\mathcal{G}_d$ *všetkých grafov* $G = (V, E)$ *takých, že:*

(i) V je spočítateľne nekonečná,

(ii) G je súvislý,

(iii) každý vrchol má najvias d susedov.

Okrem toho sme v práci definovali a študovali triedy „jednorozmerne vyzerajúcich” a „dvojrozmerne vyzerajúcich” prepojení.

Definícia 3 *Nech $G = (V, E)$ je graf a (S, ϱ) je metrický priestor. Nech r, h and H sú kladné reálne čísla. Povieme, že G je (r, h, H)-topologicky* compatibilný *s (S, ϱ) ak G je súvislý a existuje zobrazenie $\varphi\colon G \to S$ také, že:*

(i) pre každé $x \in S$ existuje vrchol $v \in V$ taký, že $\varrho(x, \varphi(v)) \leq r$,

(ii) pre každé $u, v \in V$: $h\,\mathrm{d}(u, v) \leq \varrho(\varphi(u), \varphi(v)) \leq H\,\mathrm{d}(u, v)$, kde $\mathrm{d}(u, v)$ je vzdialenosť medzi u a v v grafe G.

Definícia 4 *Trieda (r, h, H)-jednorozmerných prepojení je množina všetkých grafov, ktoré sú (r, h, H)-topologicky kompatibilné s reálnou osou.*

Definícia 5 *Trieda (r, h, H)-dvojrozmerných prepojení je množina všetkých grafov, ktoré sú (r, h, H)-topologicky kompatibilné s euklidovskou rovinou.*

Význam výsledkov: Takto definované štruktúry prepojení medzi bunkami sú všeobecnejšie, než v prípade klasických bunkových automatov. Umožňujú nám tak presnejšie modelovať biologické a fyzikálne systémy, ktoré sa vyskytujú v prírode, prípadne systémy vzniknuté technologickými procesmi, ktoré sú najčastejšie usporiadané v nejakej dvojrozmernej štruktúre.

3.3 Výpočtová sila

Ukázali sme že, nepravidelné a asynchrónne bunkové automaty dokážu simulovať ľubovoľné iteratívne pole aj pri tých najslabších predpokladoch o ich medzibunkových prepojeniach.

Veta 1 *Pre každé celé $d > 1$ a každé iteratívne pole A existuje lokálna prechodová schéma P, ktorá sa chová dobre na prepojeniach stupňa najviac d a simuluje A v priestorovej zložitosti $O((d-1)^{3S(n)})$, kde $S(n)$ je priestorová zložitosť ieratívneho poľa A.*

Zložitosť tejto simulácie je však exponenciálne horšia a nedá sa zlepšiť.

Veta 2 *Nech $d > 2$ je celé číslo a P je lokálna prechodová schéma, ktorá sa chová dobre na prepojeniach stupňa najviac d.*

Existuje iteratívne pole A také, že $L(A) = L_{1W,\mathcal{G}_d,\mathcal{A}_f}(P)$ a jeho priestorová zložitosť je $O(\log S(n))$, kde $S(n)$ je priestorová zložitosť P na prepojeniach stupňa najviac d.

Keď sú prepojenia medzi bunkami jednorozmerné, funguje táto simulácia bez zhoršenia zložitosti.

Veta 3 *Nech r, h and H sú kladné reálne čísla a A je 1D iteratívne pole. Existuje lokálna prechodová schéma P, ktorá sa chová dobre na (r,h,H)-jednorozmerných prepojeniach a simuluje A v priestorovej zložitosti $O(S(n))$, kde $S(n)$ je priestorová zložitosť ieratívneho poľa A.*

V prípade dvojrozmerných prepojení je priestorová zložitosť tejto simulácie len kvadraticky horšia.

Veta 4 *Nech r, h and H sú kladné reálne čísla a A je 1D iteratívne pole. Existuje lokálna prechodová schéma P, ktorá sa chová dobre na (r,h,H)-dvojrozmerných prepojeniach a simuluje A v priestorovej zložitosti $O(S^2(n))$, kde $S(n)$ je priestorová zložitosť ieratívneho poľa A.*

Veta 5 *Nech $r \geq \frac{\pi}{3} + \frac{1}{2}$, $h \leq \frac{1}{\sqrt{8}}$, $H \geq \frac{2\pi}{3}$ a P je lokálna prechodová schéma, ktorá sa chová dobre na (r,h,H)-dvojrozmerných prepojeniach.*

Existuje iteratívne pole A také, že $L(A) = L_{1W,\mathcal{G}^{E_2}_{(r,h,H)},\mathcal{A}_f}(P)$ a jeho priestorová zložitosť je $O(\sqrt{S(n)})$, kde $S(n)$ je priestorová zložitosť P na (r,h,H)-dvojrozmerných prepojeniach.

Ak však vyznačíme ešte niekoľko buniek navyše (okrem vstupnej bunky), môžeme pri dvojrozmerných prepojeniach priestorovú zložitosť zlepšiť.

Veta 6 *Nech r, h a H sú kladné reálne čísla a A je 1D iteratívne pole s priestorovou zložitosťou $S(n)$.*

Existuje číslo l_0 a lokálna prechodová schéma P, ktorá sa chová dobre a simuluje A na dvojrozmerných prepojeniach, v ktorých je vyznačená nejaká najkratšia cesta dĺžky aspoň l_0. Navyše priestorová zložitosť tejto lokálnej prechodovej schémy (na týchto prepojeniach) je $O(S(n))$, kde $S(n)$ je priestorová zložitosť ieratívneho poľa A.

Význam výsledkov: Ukázali sme, že nepravidelné a asynchrónne bunkové automaty dokážu simulovať ľubovoľné jednorozmerné iteratívne pole a teda sú Turingovsky úplné. V prípade jednorozmerných prepojení je dokonca aj priestorová zložitosť tejto simulácie rovnaká ako priestorová zložitosť simulovaného iteratívneho poľa. Pri dvojrozmerných prepojeniach potrebujeme na dosiahnutie takejto zložitosti vyznačiť niekoľko buniek navyše. Vo všeobecnosti (keď jediné obmedzenie na štruktúru prepojení je maximálny stupeň) však dochádza k exponenciálnemu zhoršeniu priestorovej zložitosti.

Literatúra

[AAC+00] Harold Abelson, Don Allen, Daniel Coore, Chris Hanson, George Homsy, Thomas F. Knight, Jr., Radhika Nagpal, Erik Rauch, Gerald Jay Sussman, and Ron Weiss. Amorphous computing. *Commun. ACM*, 43(5):74–82, 2000.

[AG72] S. Amoroso and R. Guilfoyle. Some comments on neighbourhood size for tesselation automata. *Information and Control*, 21:48–55, 1972.

[AP72] S. Amoroso and Y.N. Patt. Decision Procedures for Surjectivity and Injectivity of Parallel Maps for Tessellation Structures. *JCSS*, 6(5):448–464, 1972.

[Ban71] E. R. Banks. Information processing and transmission in cellular automata. Technical report, Cambridge, MA, USA, 1971.

[BCG82] E.R. Berlekamp, J.H. Conway, and R.K. Guy. *Winning ways for your mathematical plays. Volume 2*. 1982.

[But74] Jon T. Butler. A note on cellular automata simulations. *Information and Control*, 26(3):286–295, 1974.

[CD91] Karel Culik II and Simant Dube. An efficient solution of the firing mob problem. *Theor. Comput. Sci.*, 91(1):57–69, 1991.

[Cod68] E. F. Codd. *Cellular Automata*. Academic Press, Inc., Orlando, FL, USA, 1968.

[Col69] S. N. Cole. Real-time computation by n-dimensional iterative arrays of finite-state machines. *IEEE Trans. Comput.*, 18(4):349–365, 1969.

[Dub95] Jean-Christophe Dubacq. How to simulate turing machines by invertible one-dimensional cellular automata. *International Journal of Foundations of Computer Science*, 6(4):395–402, 1995.

[Fis65] Patrick C. Fischer. Generation of primes by a one-dimensional real-time iterative array. *J. ACM*, 12(3):388–394, 1965.

[Ham71] V.C. Hamacher. Machine complexity versus interconnection complexity in iterative arrays. *Computers, IEEE Transactions on*, C-20(3):321–323, March 1971.

[JK74] J. Jump and J. Kirtane. On the interconnection structure of cellular networks. *Inf. Control*, 24:74–91, 1974.

[Kar90] J. Kari. Reversibility of 2d cellular automata is undecidable. *Phys. D*, 45(1-3):386–395, 1990.

[Kar94] Jarkko Kari. Reversibility and surjectivity problems of cellular automata. *J. Comput. Syst. Sci.*, 48(1):149–182, 1994.

[Maz87] Jacques Mazoyer. A six-state minimal time solution to the firing squad synchronization problem. *Theoretical Computer Science*, 50(2):183–238, 1987.

[ML68] FR Moore and GG Langdon. A generalized firing squad problem. *Information and Control*, 12:212–220, 1968.

[Moo64] Edward F. Moore, editor. *Sequential Machines: Selected Papers*. Addison-Wesley Longman Ltd., Essex, UK, UK, 1964.

[Mor95] Kenichi Morita. Reversible simulation of one-dimensional irreversible cellular automata. *Theor. Comput. Sci.*, 148(1):157–163, 1995.

[Myh63] J. Myhill. The converse of moore's garden-of-eden theorem. *Proc. Am. Math. Soc.*, 14:685–686, 1963.

[NC97] R. Nagpal and D. Coore. An algorithm for group formation and maximal independent set in an amorphous computer. *AI Memo 1626, MIT*, 1997.

[Neh04] C.L. Nehaniv. Asynchronous Automata Networks Can Emulate Any Synchronous Automata Network. *International Journal of Algebra and Computation*, 14(5):719–739, 2004.

[NH74] H.B. Nguyen and V.C. Hamacher. Pattern Synchronization in Two-Dimensional Cellular Spaces. *Information and Control*, 26:12–23, 1974.

[NSB03] R. Nagpal, H. Shrobe, and J. Bachrach. Organizing a global coordinate system from local information on an ad hoc sensor network. In *Information Processing in Sensor Networks*, pages 553–553. Springer, 2003.

[Ric72] D. Richardson. Tessellations with local transformations. *Journal of Computer and System Sciences*, 6(5):373–388, 1972.

[San94] P. Sanders. Massively parallel search for transition-tables of polyautomata. *Parcella*, 94:99–108, 1994.

[Shi74] Ilka Shinahr. Two-and Three-Dimensional Firing-Squad Synchronization Problems. *Information and Control*, 24:163–180, 1974.

[SI71a] Alvy Ray Smith III. Simple computation-universal cellular spaces. *J. ACM*, 18(3):339–353, 1971.

[SI71b] A.R. Smith III. Cellular automata complexity trade-offs. *Information and Control*, 18(5):466–482, 1971.

[TM91] T. Toffoli and N.H. Margolus. Invertible cellular automata: a review. *Cellular automata: theory and experiment table of contents*, pages 229–253, 1991.

[Tof77] T. Toffoli. Computation and construction universality of reversible cellular automata. *Journal of Computer and System Sciences*, 15(2):213–231, 1977.

[vN63] J. von Neumann. The general and logical theory of automata. In A. Taub, editor, *J. von Neumann Collected Works*. 1963.

[vNB66] J. von Neumann and A. Burks, editors. *Theory of Self-Reproducing Automata*. University of Illinois Press, Champaign, IL, 1966.

[Wak86] A. Waksman. An Optimum Solution to the Firing Squad Synchronization Problem. *Information and Control*, 9:66–78, 1986.

[WP07] J. Wiedermann and L. Petru. Computability in Amorphous Structures. *Lecture Notes in Computer Science*, 4497:781, 2007.

[WP09] J. Wiedermann and L. Petru. On the universal computing power of amorphous computing systems. *Theory of Computing Systems*, 45(4):995–1010, 2009.

4 Summary

We define a new computational model of asynchronous and irregular cellular automata. This is a variant of cellular automata, in which the cells need not be connected in a regular grid and the computation need not be synchronous. The cells are finite-state and are not equipped with a random generator, local identification nor any means to tell apart their neighbours. The communication with the outside world is done through one special cell which has access to input and is able to signalize acceptance. Optionally some other cells may be marked by a special state prior to computation. We require the whole device to give the correct answer for any connections between cells and any computation timing. For this reason the cells themselves, not the automata composed of them are of our main interest. In other words, the main question we would like to ask is "How to design a cell so that any device composed of many copies of such cell solves our main problem".

We provide the definition of this computational model, the way it works and its complexity measures. We introduce some basic techniques to deal with such device and study its computational power. We shall see that even under such loose conditions it is able to simulate any iterative array and thus is Turing complete. However its space complexity may be exponentially worse than that of the simulated iterative array. We show that if we assume a reasonably restricted class of cellular interconnection structures (1D- and 2D-like) the space complexity of the simulation is better (linear and quardatic respectively). We show that even in the case of 2D-like connections the complexity can be made linear if we mark a constant number of cells prior to the computation.