

Automaty komórkowe

[http://home.agh.edu.pl/malarz/dyd/ak/
v. 2.718281828459045235360287](http://home.agh.edu.pl/malarz/dyd/ak/v.2.718281828459045235360287)

Wprowadzenie

Krzysztof Malarz

2 maja 2024

Terry Pratchett, *Mort*, Prószyński i S-ka (Warszawa 2???)

Mam wrażenie, że nie posiada pan żadnej pożytecznej umiejętności ani talentu. Czy myślał pan, żeby się zająć nauczaniem?

Terry Pratchett, *Muzyka duszy*, Prószyński i S-ka (Warszawa 2002)

— Tak to właśnie wygląda, kiedy ludzie próbują edukować swoje dzieci — stwierdził kruk — zamiast mówić im o różnych rzeczach...

Czym są automaty komórkowe (AK)?

- alternatywną matematyką, szczególnie przydatną w obliczeniach równoległych, wolną od błędów zaokrągleń,
- narzędziem do symulacji procesów fizycznych, w których bierze udział wiele układów oddziałujących ze sobą,
- obiektami matematycznymi, interesującymi z punktu widzenia teorii procesów dynamicznych,
- zabawką,
- oraz — przedmiotem tego wykładu...

Trochę historii I

- Za twórcę automatów komórkowych uważa się Janosa von Neumanna, Węgry pracującego w Princeton. Jak wiadomo, wprowadził do swego modelu “pierwotnej zupy” dyskretny czas i przestrzeń z inspiracji Stanisława Ulama, lwowskiego matematyka, który przebywał wtedy w Los Alamos. Ulam jest też autorem określenia automatów komórkowych jako “fizyki urojonej”...
- Najstłynniejszym chyba automatem jest **Game of Life** autorstwa angielskiego matematyka Johna H. Conwaya [1]. Ten automat miał kiedyś swój fanklub wśród naukowców i studentów USA. W konkursie na konfigurację komórek, która będzie rosła nieskończenie, zwyciężyła w listopadzie 1970 grupa z MIT, publikując “działo szybowcowe” i zgarniając nagrodę w wysokości 50 dolarów amerykańskich. W *Life*

Trochę historii II

można widzieć model żyjącego środowiska; w użyciu jest terminologia “żywych” i “martwych” komórek. Ale jest to automat uniwersalny, czyli zdolny do każdej operacji logicznej.

- AK weszły do bibliotek fizyków na początku lat 80-tych. Jednym z głównych popularyzatorów tej idei był Stephen Wolfram, znany jako twórca pakietu Mathematica. Próbował on również sklasyfikować AK — gdyby się ten zamiar powiódł, można by mówić o klasyfikacji **wszystkich** dyskretnych procesów dynamicznych.

Podręczniki I

- K. Kułakowski. *Automaty Komórkowe*. Kraków: OEN AGH, 2000
- S. Wolfram. *A New Kind of Science*. Wolfram Media, 2002
- A. Ilachinski. *Cellular Automata. A Discrete Universe*. World Scientific, 2001
- B. Chopard and M. Droz. *Cellular Automata Modeling of Physical Systems*. Cambridge: Cambridge UP, 2005
- S. Wolfram. *Theory and Applications of Cellular Automata*. World Scientific, 1986
- H. Gutowitz, ed. *Cellular Automata. Theory and Experiment*. MIT Press, 1991
- A. Bunde and S. Havlin, eds. *Fractals and Disordered Systems*. Berlin: Springer, 1996

Terry Pratchett, *Ciekawe czasy*, Prószyński i S-ka (Warszawa 2003)

— Oczywiście, nadrektorze. Ale zdaje pan sobie chyba sprawę, że odnalezienie go może nam zabrać całe lata?

— Ehm... — wtrącił Myślak. — Gdybyśmy odszyfrowali jego sygnaturę thaumiczną, HEX poradziłby sobie z tym w jeden dzień. Dziekan rzucił mu gniewne spojrzenie.

— To nie jest magia — burknął. — To... mechanika!

Skala trudności

Deterministyczna maszyna Turinga

składa się:

- z **modułu sterującego** mogącego się znajdować w jednym ze skończonej ilości stanów w danej chwili,
- **głowicy** czytająco-piszącej,
- **taśmy** będącej układem pamięciowym podzielonym na jednostki i prawostronnie nieskończonym,

i może być traktowana jako **model** każdego obliczenia sekwencyjnego.

Każde obliczenie można przedstawić poprzez siedem elementarnych operacji, tworzących język Turinga–Posta mogący realizować dowolne możliwe obliczenia. Np. DRUKUJ-0 — ma kod 000, DRUKUJ-1 — ma kod 001, IDŹ-W-LEWO — ma kod 010, aż STOP — ma kod 100. Tym sposobem, wszystkie możliwe algorytmy (a między nimi i dowody twierdzeń) można ustawić w ciąg i ponumerować — tworzą one zbiór przeliczalny.

Swój numer ma również sama maszyna Turinga — czyli algorytm odczytujący i wykonujący dowolny zadany algorytm.

Twierdzenie Gödla I

- Swój numer ma nawet dowód, że niektórych algorytmów **nie ma** na liście — inaczej mówiąc, nie istnieją.
- **Nie istnieją więc dowody niektórych twierdzeń!** Można je wypowiedzieć, ale niesposób udowodnić.
- Ta teza jest treścią słynnego twierdzenia Gödla, opublikowanego w 1931 roku: **“W ramach każdego formalizmu można wypowiedzieć twierdzenia, których nie można udowodnić w ramach tego formalizmu”**.
- Problem stopu: Czy dla danego programu Turinga–Posta P , można za pomocą obliczenia sprawdzić czy dla pewnej danej tego programu, program P zatrzyma się jeśli obliczenie programu P zostało rozpoczęte z tą daną?

Twierdzenie Gödla II

- Problem stopu i inne problemy tej klasy nazywamy **nierozwiązywalnymi** bądź **nierozstrzygalnymi** (undecidable).

Problemy NP i NP-zupełne I

- Inną kategorię trudności stanowią problemy w zasadzie rozwiązywalne, których czas rozwiązywania jest jednak bardzo długi.
- Problem komiwojażera: dane są położenia N miast, które komiwojażer chce odwiedzić. W jakiej kolejności powinien je odwiedzać, jeśli w każdym musi być dokładnie jeden raz a całkowita droga powinna być najkrótsza?
- Czas wykonania takiego algorytmu jest proporcjonalny do $N!$, czyli rośnie z N szybciej, niż N^α , gdzie $\alpha \in \mathbb{N}$.
- Ta zależność czasu obliczeń od rozmiaru problemu może służyć do naiwnego rozróżnienia między problemami klasy P (polynomial) a problem klasy NP (non-polynomial). Te drugie są uważane za **praktycznie nierozwiązywalne** (intractable).

Problemy NP i NP-zupełne II

- Istnieje cała grupa problemów klasy NP które da się wzajemnie na siebie przetłumaczyć (w formalnej definicji problemy mają być wielomianowo redukowalne jeden w drugi) — tworzą one grupę problemów **NP-zupełnych** (NP-com).
- Gdy zaczynaliśmy dzisiejszy wykład problem komiwojażera był na liście problemów NP-com ($P \neq NP$)...

Tabela: Porównanie czasów obliczeń problemów klasy P i klasy NP

Typ	$N = 10$	$N = 20$	$N = 40$	$N = 60$
N	10^{-5} s	$2 \cdot 10^{-5}$ s	$4 \cdot 10^{-5}$ s	$6 \cdot 10^{-5}$ s
N^2	10^{-4} s	$4 \cdot 10^{-4}$ s	0.0016 s	0.0036 s
N^3	0.001 s	0.008 s	0.064 s	0.216 s
N^4	0.1 s	3.2 s	1.7 min.	13 min.
2^N	0.001 s	1 s	12.7 dni	36600 lat
3^N	0.059 s	58 min.	385500 lat	$1.2 \cdot 10^{15}$ lat

- Złożoność problemu definiuje się jako minimalną długość algorytmu jego rozwiązania.
- O nieprzewidywalności obliczeniowej mówimy, gdy algorytmu nie można skrócić: jedyną drogą uzyskania wyniku jest wykonanie obliczeń krok po kroku.

Na podstawie: [2].

- [1] E. R. Berlekamp, J. H. Conway, and R. K. Guy. *Winning Ways for Your Mathematical Plays*. New York: Academic, 1982.
- [2] K. Kułakowski. *Automaty Komórkowe*. Kraków: OEN AGH, 2000.
- [3] S. Wolfram. *A New Kind of Science*. Wolfram Media, 2002.
- [4] A. Ilachinski. *Cellular Automata. A Discrete Universe*. World Scientific, 2001.
- [5] B. Chopard and M. Droz. *Cellular Automata Modeling of Physical Systems*. Cambridge: Cambridge UP, 2005.
- [6] S. Wolfram. *Theory and Applications of Cellular Automata*. World Scientific, 1986.
- [7] H. Gutowitz, ed. *Cellular Automata. Theory and Experiment*. MIT Press, 1991.

- [8] A. Bunde and S. Havlin, eds. *Fractals and Disordered Systems*. Berlin: Springer, 1996.