

Artur RATAJ

Instytut Informatyki Teoretycznej i Stosowanej PAN

## PROCESY KONSTRUKCJI AUTOMATÓW POWIELAJĄCYCH SIĘ W ŚRODOWISKACH ZE ZDARZENIAMI LOSOWYMI

**Streszczenie.** W niniejszym opracowaniu przedstawiono przykład modelowania środowiska automatów, w którym zdarzenia losowe wpływają na procesy konstrukcji tych automatów. Omówione są niektóre własności takich automatów.

## PROCESSES OF CONSTRUCTION OF SELF-REPLICATING AUTOMATA IN ENVIRONMENTS WITH RANDOM EVENTS

**Summary.** In this article an example of modelling of an environment of automata is presented, in which random events influence on processes of construction and the automata. Some features of such automata are discussed.

### 1. Wstęp

Losowy wpływ na działanie automatów, w połączeniu z ograniczonością zasobów w ich środowisku, może być czynnikiem powodującym powstawanie automatów w określony sposób skonstruowanych. W dalszej części omówimy niektóre możliwe cechy takich automatów, jak na przykład optymalizacja pobierania zasobów i zdolność do wytworzenia swojej kopii.

Opisywane automaty mogą zawierać program wytworzenia kolejnych automatów. Jednym z elementów ich działania jest więc tworzenie kopii takiego programu. W opracowaniu tym omówimy niektóre ze zjawisk, jakie mogą być związane z tym procesem.

## 2. Definicja automatu

Wyróżnijmy w pewnym układzie wybrany zbiór jego parametrów i przyjmijmy, że wartości tych parametrów określają stan układu. Jeżeli zmiany tak zdefiniowanych stanów będą następować według określonych reguł, to taki układ będziemy nazywać automatem, a za jego działanie będziemy uważać przejścia między jego stanami.

Jeżeli dany automat w trakcie swojego działania wytwarza automat o identycznych jak jego własnościach, to jest on automatem powielającym się.

## 3. Przykłady automatów

W kolejnych sekcjach opisane są niektóre ze znanych z literatury przykładów automatów powielających się.

### 3.1. Uniwersalny konstruktor

W roku 1966 von Neumann przedstawił automat komórkowy, który nazwał uniwersalnym konstruktorem [1]. Maszyna tego automatu była 29-stanowa i miała reguły iteracyjne, do których było potrzebne pięciosiędziesiątstwo, to znaczy stan komórki w kolejnej iteracji był wyznaczany przez aktualny stan jej samej i jej czterech ortogonalnych sąsiadów. Automat ten umiał skonstruować dowolny inny automat komórkowy na podstawie wewnętrznego programu.

### 3.2. Pętla Langtona

W roku 1984 C. G. Langton przedstawił automat komórkowy, który był tak zaprojektowany, aby symulować pewne procesy zachodzące w żywych komórkach[2]. Miał on formę pętli. Pętla taka tworzyła ramię, które z kolei tworzyło nową pętlę. Pętla składała się z osłony, wewnątrz której znajdował się kod opisujący samą pętlę. Kod ten przesuwiał się wewnątrz pętli, a w miejscu, w którym z pętli wychodziło ramię, był kopiowany. Jedna z kopii wracała do pętli, druga przesuwiała się do ramienia, na którego końcu, według określonych reguł i kopii kodu, tworzona była nowa pętla. Dzięki temu, że jedna z kopii kodu wracała do pętli, pętla ta mogła tworzyć swoje kolejne kopie. Kod miał być odpowiednikiem genotypu, jego kopiowanie miało symulować kopiowanie genotypu, a reguły budujące nową pętlę na podstawie kodu miały modelować fenotyp.

## 4. Rola środowiska

Środowisko w dyskutowanym przypadku jest źródłem zdarzeń losowych, powodujących zmienność warunków. Środowisko dostarcza również zasoby, które mogą być ograniczone.

Przyjmijmy, że prawdopodobieństwo  $p$  uformowania się automatu pewnego typu na skutek zdarzeń losowych występujących w pewnym środowisku jest bardzo małe. W takim razie, jeżeli zdarzenie „uformowanie się automatu tego typu” w jednym miejscu w środowisku jest niezależne od wystąpienia takiego zdarzenia w innym miejscu w środowisku, prawdopodobieństwo  $p_n$  wystąpienia  $n$  takich zdarzeń w różnych miejscach w środowisku, zgodnie z zasadami rachunku prawdopodobieństwa, może maleć silnie wykładniczo z  $n$ . Prawdopodobieństwo to może maleć w przybliżeniu w taki sposób również wtedy, jeżeli zależności pomiędzy omawianymi zdarzeniami są odpowiednio małe. Jednak, jeżeli jedno takie zdarzenie pociąga z odpowiednio dużym prawdopodobieństwem  $r$  drugie takie zdarzenie,  $p_n$  może słabo maleć wraz z  $n$ . Prawdopodobieństwo  $r$  może być duże w przypadku automatów, które się powielają. Stąd teza, że w odpowiednim środowisku automaty wytwarzające swoje kopie wystąpią w dużej liczbie z największym prawdopodobieństwem.

Automaty występujące w dużej ilości, o własności powielania się, mogą podlegać specyficznym zjawiskom. Omówimy niektóre z tych zjawisk.

### 4.1. Błędy

Z względu na szeregowość kolejnych procesów kopiowania ewentualne błędy w tych procesach mogą się kumulować w kolejnych realizacjach automatu. Dodatkowo, błędy mogą powodować inne błędy. Wskutek tego efekty błędów mogą silnie rosnać, na przykład w sposób zbliżony do wykładniczego.

### 4.2. Automaty zmienione

Jeżeli dany automat jest różny od automatu, który go wytworzył, będziemy go dalej nazywać zmienionym.

#### 4.2.1. Selekcja

Załóżmy, że w danym środowisku istnieją zasoby potrzebne automatom i że zasoby te są ograniczone. Wtedy jeden automat, pobierając zasoby, których używa lub mógłby użyć inny automat, może na przykład uniemożliwić działanie tego drugiego automatu, lub go uszkodzić. Automaty skuteczniej wykorzystujące zasoby mogą więc doprowadzić do zmniejszenia liczby

automatów mniej skutecznie wykorzystujących zasoby. Może więc nastąpić selekcja automatów ze względu na skuteczność wykorzystania zasobów. Ze względu na proces selekcji typy automatów ze zmianami poprawiającymi tę skuteczność mogą z większym prawdopodobieństwem działać w środowisku.

#### 4.2.2. Sklejanie i specjalizacja

Omówimy teraz dwa szczególne typy zmian. Niech sklejanie będzie połączeniem się automatów lub ich fragmentów w jeden automat. Niech specjalizacja będzie podziałem typów zadań wykonywanych przez automat na różne jego fragmenty.

Jeżeli automaty tworzą swoje kopie we względnie niewielkiej odległości od siebie, sklejanie może doprowadzić do konstrukcji automatów zawierających w swoim obrębie więcej identycznych lub podobnych fragmentów. Ze względu na różnorodność środowiska może nastąpić różnicowanie się takich elementów, czyli ich specjalizacja.

## 5. Przykład procesów konstrukcji automatów

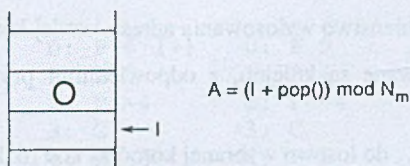
W kolejnych sekcjach wykonamy symulację działania przykładowego środowiska opisywanych automatów. Automaty te będą reprezentowane przez sekwencje rozkazów w pamięci maszyny o następującej specyfikacji.

### 5.1. Specyfikacja maszyny

Maszyna składa się z pamięci o wielkości  $N_m$  komórek, procesora, listy wątków, pamięci wspólnej wątków i pamięci na dane własne wątków.

#### 5.1.1. Wątki

W maszynie istnieje lista wątków, to znaczy procesów używających wspólnej pamięci maszyny i mających własne stosy do przechowywania wartości liczbowych. Liczba wątków w liście jest nieograniczona. Każdy wątek przechowuje swoją wartość i licznika rozkazów maszyny i swój stos. Maksymalna długość stosu wynosi  $N_{\max}^s$ . Jeżeli nastąpi niedomiar lub przepełnienie stosu, wątek zostaje zakończony.



Rys. 1. Ilustracja do opisu adresacji używanej w maszynie –  $O$  oznacza komórkę maszyny z kodem aktualnego rozkazu

Fig. 1. Illustration to the description of the addressing used in the machine –  $O$  denotes the cell with the current operation code

### 5.1.2. Rozkazy

Każdy rozkaz zajmuje jedną komórkę pamięci. Ewentualne argumenty rozkazu pobierane są ze stosu. Po pobraniu kodu rozkazu z komórki pamięci wskazywanej przez licznik rozkazów  $I$  wartość  $I$  zmieniana jest następująco:

$$I := (I + 1) \bmod N_i. \quad (1)$$

Jeżeli wartość  $n$  argumentu ma wyznaczać adres  $A$ , to jest on wyliczany następująco:

$$A = (i + n) \bmod N_m. \quad (2)$$

Operacje takie są wykonywane po zmianie  $I$  w obrębie rozkazu zgodnie z (1). Ilustruje to rys. 1.

Istnieją cztery typy rozkazów:

- P  $n$  – połóż na stos wartość  $n$ .
- S – zakończ wątek.
- C  $A_1, A_2, l$  – kopiuj zawartość obszaru wspólnej pamięci  $(A_1) \dots (A_1 + l - 1)$  do obszaru wspólnej pamięci  $(A_2) \dots (A_2 + l - 1)$ , jeżeli  $l > 0$ . Wartości w obrębie odpowiednich wartości  $i$  wyznaczających końce obszarów są brane modulo  $N_m$ . W ten sposób obszar taki może się „zawinać” na końcu pamięci wspólnej. Argumenty  $A_1, A_2$  i  $l$  są pobierane kolejno ze stosu.
- X  $A$  – utwórz nowy wątek z pustym stosiem. Ustaw jego wartość licznika rozkazów  $I$  na adres  $A$ . Argument  $A$  jest pobierany ze stosu. Wątek jest wstawiany na początek listy wątków.

### 5.1.3. Cykl maszyny

Cykl maszyny składa się z następujących faz:

- Faza  $F1$ , mająca modelować zmienność warunków środowiska poprzez dodanie „szumu” do pamięci wspólnej maszyny. W fazie tej, w przypadku wyboru losowej

komórki, prawdopodobieństwo wylosowania adresu każdej komórki wynosi  $1/N_m$ . W obrębie fazy wykonywane są kolejno, z odpowiednimi prawdopodobieństwami  $p$ , operacje:

1. Wpisz, z  $p = p_1$ , do losowo wybranej komórki kod rozkazu  $P = \text{norm}(0, -1, \dots, -P_{\max}^a) + \text{norm}(0, 1, \dots, P_{\max}^a)$ , gdzie  $\text{norm}()$  zwraca wartość, mogącą być z jednakowym prawdopodobieństwem równą każdemu z argumentów.
2. Wpisz, z  $p = p_2$ , do losowo wybranej komórki kod rozkazu  $S$ .
3. Wpisz, z  $p = p_3$ , do losowo wybranej komórki kod rozkazu  $C$ .
4. Wpisz, z  $p = p_4$ , do losowo wybranej komórki kod rozkazu  $X$ .
5. Utwórz, z  $p = p_5$ , nowy wątek, z pustym stosem i losowo wybraną wartością  $I$ .

W symulacji zostały przyjęte następujące wartości współczynników:  $P_{\max}^a = 10$ ,  $p_1 = 1$ ,  $p_2 = 1/3$ ,  $p_3 = 1/3$ ,  $p_4 = 1/3$ ,  $p_5 = 1$ .

- Faza  $F2$  – wykonania rozkazów. Wykonaj szeregowo, w obrębie każdego kolejnego wątku w liście wątków, jeden rozkaz. Jeżeli wykonanie rozkazu spowoduje powstanie nowego wątku, to wątek ten zacznie być wykonywany, jeżeli w ogóle zacznie być wykonywany, dopiero w kolejnym cyklu maszyny.
- Faza  $F3$  – ograniczenia liczby wątków. Jeżeli po fazie  $F2$  liczba wątków przekroczyła pewną wartość  $N'_{\max}$ , to kończ losowo wybrane wątki, aż ich liczba osiągnie  $N'_{\max}$ .

#### 5.1.4. Inicjalizacja maszyny

Przed rozpoczęciem wykonywania cykli maszyny do wszystkich komórek pamięci jest wpisywany kod rozkazu  $S$ , a lista wątków jest pusta.

#### 5.2. Przebieg symulacji

Symulacja została przeprowadzona przy następujących współczynnikach:  $N_m = 10000$ ,  $N_{\max}^s = 10$ ,  $N'_{\max} = 1000$ . Niech  $R_n$  oznacza proporcję w procentach komórek pamięci wspólnej, które należą do wzorca automatu  $Pn$ , znajdującego się w tej pamięci, do liczby wszystkich komórek pamięci wspólnej  $N_m$ . Na rys. 2 pokazane są wzorce automatów,

0: P 4 (+)	0: P 6 (+)	0: P 9	0: P 9
1: P 0	1: P 2	1: P 4	1: P 4
2: P -4	2: P -4	2: P -4	2: P -4
3: C	3: C	3: C	3: X
	4: P -6	4: P 9	4: X
	5: X	5: P 4	5: P 4
		6: P -4	6: P -4
		7: X	7: C
			8: P 9
			9: P 4
			10: P -4
			11: X

Rys. 2. Przykłady wzorców automatów

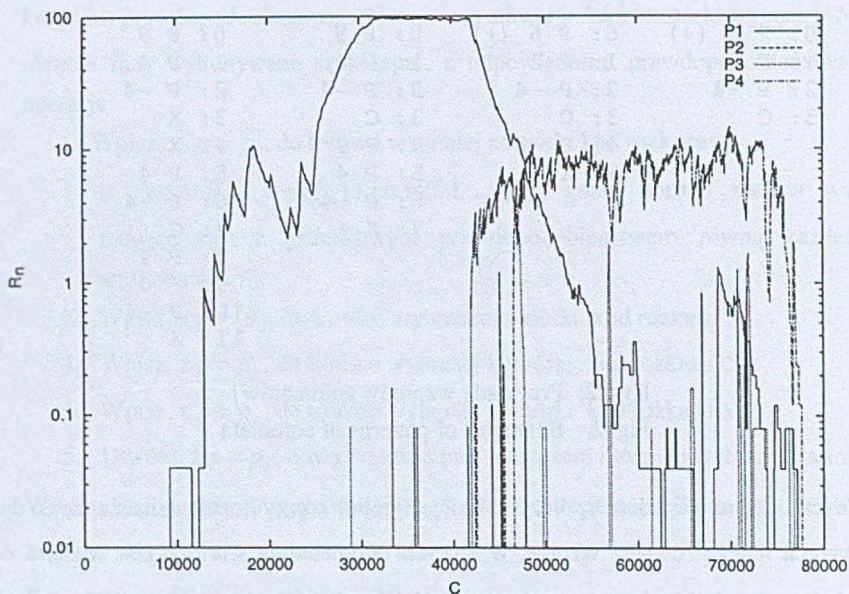
Fig. 2. Examples of patterns of automata

dla których liczono wartości współczynnika  $R_n$ . Symbol + przy liczbie oznacza, że do danego typu wzorca należą również wzorce, w których odpowiednia wartość jest większa od tej liczby.

Automat P1 wykonuje swoją kopię w obszarze pamięci bezpośrednio po swoim ostatnim rozkazie, po czym przekazuje sterowanie do utworzonej kopii. Ze względu na to, że składa się tylko z czterech rozkazów, można przypuszczać, że pojawi się podczas symulacji ze względnie dużym prawdopodobieństwem. Automat P2 dodatkowo uruchamia nowy wątek od swojego pierwszego rozkazu. Sekwencje rozkazów automatów P3 i P4 zostały dodane po stwierdzeniu, że przy użytym generatorze liczb pseudolosowych i użytej wartości początkowej tego generatora wystąpiły podczas symulacji we względnie dużej ilości. Wyniki obliczeń wartości  $R_n$  wzorców dla pierwszych 80000 cykli maszyny są przedstawione na rys. 3.

Podczas przebiegu symulacji można było zaobserwować następujące zjawiska:

1. Początkowo żaden z obserwowanych wzorców nie zajął dużej części pamięci. Wynika to z tego, że z dużym prawdopodobieństwem na początku symulacji duża ilość wątków natrafiała na rozkaz S i że było względnie mało cykli maszyny na uformowanie się automatu powielającego.
2. Około 10000 cyklu maszyny pojawiły się pierwsze sekwencje rozkazów automatów typu P1, jednak po pewnym czasie zniknęły.
3. Około cyklu 12000 znowu pojawiają się sekwencje automatów typu P1. Około cyklu 30000 zajmują ponad 90% pamięci.

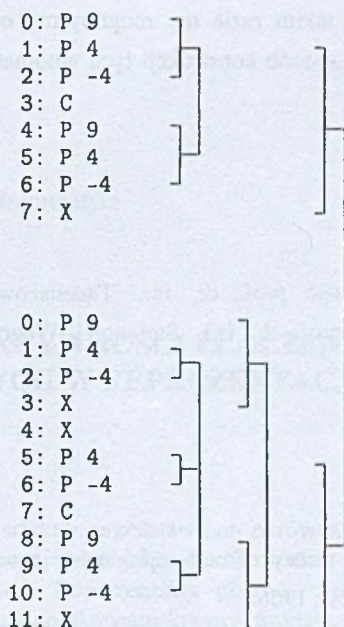


Rys. 3. Wyniki obliczeń współczynników  $R_n$  automatów. C jest numerem cyklu maszyny

Fig. 3. Computation results of the coefficients  $R_n$  of the automata. C is the machine cycle number

4. Około 43000 cyklu maszyny pojawiają się pierwsze sekwencje rozkazów automatów P3, a równocześnie zaczyna szybko maleć liczebność wzorców P1. Można przypuszczać, że zjawiska te są związane i wynikają z tego, że automat P3 ma przewagę nad automatem P1 pod względem efektywności wykorzystania zasobów przyjętej maszyny.
5. Po kilku tysiącach cykli pojawiają się sekwencje automatów P4. Być może powtórzyła się sytuacja z poprzedniego punktu, tylko w odniesieniu do automatów P3 i P4. Widoczne jest podobieństwo wzorców automatów P3 i P4. Być może wynika ono z tego, że P4 jest zmienionym automatem P3. Być może podczas konstrukcji P4 wystąpiły zjawiska sklejania i specjalizacji, tak jak jest to zilustrowane na rys. 4.
6. Przed 80000 cyklem maszyny zanikają sekwencje rozkazów automatów P4. Być może przyczyną są wątki oparte na jakichś innych sekwencjach, lub szum środowiska.





Rys. 4. Ilustracja zjawisk sklejania i specjalizacji, na przykładzie automatów P3 i P4. Oznaczone są identyczne sekwencje rozkazów

Fig. 4. Illustration of the gluing and specialization phenomena, on the example of automata P3 and P4. Identical sequences of operations are marked

## 6. Wnioski

Wyniki modelowania wskazują, że już w przyjętym, prostym modelu można zaobserwować niektóre z właściwości wykazywanych przez opisywane automaty.

W artykule używane było wyrażenie „konstrukcja automatów”, ale oczywiście w przypadku środowiska deterministycznego można by użyć określeń, takich jak „rozwiązanie równania”. Stan początkowy środowiska mógłby być traktowany jako warunki początkowe równania, a procesy zachodzące w tym środowisku jako, na przykład iteracyjne, rozwiązywanie tego równania. Moglibyśmy powiedzieć, że automaty były w środowisku od jego początku, w formie którą można by arbitralnie nazwać *niejawną*, i po pewnej liczbie iteracji uzyskały, powiedzmy, *jawną* formę. Wtedy moglibyśmy powiedzieć, że automaty te były skonstruowane przed tym, jak środowisko zaczęło działać.

Przedyskutujmy teraz przypadek, w którym zdarzenia losowe byłyby nieprzewidywalne. W takim przypadku moglibyśmy raczej powiedzieć, że zdarzenia te wydają się być przypadkowe w zakresie przeprowadzonych obserwacji, ale nie moglibyśmy sformułować

tezy, że są przypadkowe. W takim razie nie moglibyśmy określić roli, jaką mają przy konstrukcji automatów, a stąd sposób konstrukcji tych automatów również nie mógłby być określony.

## Podziękowania

Autor chciałby podziękować prof. dr. inż. Tadeuszowi Czachórskiemu, dr. inż. Przemysławowi Szmali i prof. dr. inż. Stefanowi Węgrzynowi za pomocne uwagi dotyczące tego artykułu.

## LITERATURA

1. von Neumann J.: The theory of self-reproducing automata. Edytor A. W. Burks, University of Illinois Press, 1966.
2. Langton C. G.: Studying artificial life with cellular automata. Physics D, t. 22, str. 120–149, 1986.

Recenzent: Dr inż. Przemysław Szmal

Wpłynęło do Redakcji 5 stycznia 2000 r.

## Abstract

In this article, example of modelling automata construction and self-replication using random events is presented. Some features of the presented automata, like their interaction with environment and copying of their replication program, are discussed. Sec. 1 is an introduction. In Sec. 2, basic notations are defined. In Sec. 3, some examples of self-replicating automata are presented. In Sec. 4, the role of environment and phenomena like alteration, gluing and specialization are discussed. In Sec. 5, example simulation is shown. Fig. 1 is an illustration to an example machine addressing mode. In Fig. 2 some patterns of discussed automata are presented, and in Fig. 3 the simulation results are shown. Fig. 4 is an illustration to the gluing and specialization phenomena on the example of automata in the simulation. Sec. 6 contains some conclusions.