Automaty z czasem

Paulina Kania, Łukasz Osipiuk

16 lutego 2004

1 Modelowanie czasu - wprowadzenie

1.1 Czas dyskretny

W tym podejściu czas modelowany jest jako monotonicznie rosnący ciąg licz naturalnych. Jest to podejście które nadaje się do modelowania niektórych specyficznych systemów (np. mikroprocesorów sterowanych zegarem)

Zalety

- prosty model
- łatwo przejść do modelu zwykłych automatów skończonych

Wady

- nienaturalny
- konieczne jest ustalenie apriori dokładności pomiarów - minimalnego czasu jaki musi wystąpić między zdarzeniami.

1.2 Czas ciągły

- Dziedziną czasu jest zbiór gęsty (np. liczby wymierne).
- Zdarzenia mogą następować po sobie w dowolnie krótkich odstępach czasu.
- W automatach z czasem realizowane jest to podejście do modelowania czasu.

2 Systemy tranzycyjne (ang.: *Transition systems*)

Systemem tranzycyjnym nazwiemy krotkę: $(Q, Q^0, \Sigma, \rightarrow)$

gdzie:

- Q - zbiór lokacji
- Q^0 - zbiór lokacji początkowych ($Q^0 \subseteq Q$)
- Σ - zbiór zdarzeń, etykiet (alfabet)
- $\rightarrow$ - relacja przejścia - zbiór krawędzi ($\rightarrow \subseteq Q \times \Sigma \times Q$)

2.1 Złożenie równoległe (ang.: *Product*)

Konstrukcja złożenia równoległego $\|$ służy do modelowania złożonych systemów. Osobne elementy modelowane są przez osobne systemy tranzycyjne natomiast cały (złożony) system przez ich złożenie.

Złożenie równoległe definiujemy następująco:

dla systemów tranzycyjnych $S_1 = (Q_1, Q_1^0, \Sigma_1, \rightarrow_1)$ oraz $S_2 = (Q_2, Q_2^0, \Sigma_2, \rightarrow_2)$

$$S_1 \parallel S_2 = (Q_1 \times Q_2, Q_1^0 \times Q_2^0, \Sigma_1 \cup \Sigma_2, \rightarrow_{\|})$$

$(q_1, q_2) \xrightarrow{a}_{\|} (q_1', q_2')$ wtedy i tylko wtedy jeśli:

- $a \in \Sigma_1 \cap \Sigma_2$ oraz $q_1 \xrightarrow{a}_1 q_1'$ i $q_2 \xrightarrow{a}_2 q_2'$
- $a \in \Sigma_1 \setminus \Sigma_2$ oraz $q_1 \xrightarrow{a}_1 q_1'$ i $q_2 = q_2'$
- $a \in \Sigma_2 \setminus \Sigma_1$ oraz $q_1 = q_1'$ i $q_2 \xrightarrow{a}_2 q_2'$

Uwaga: Zdarzenia należące do Σ_1 i Σ_2 służą do synchronizacji.

3 Systemy tranzycyjne z więzami czasowymi (ang.: *Transition systems with timing constraints*)

Aby móc modelować czas, do powyższej konstrukcji dokładamy zbiór zegarków (ang.: *Clocks*).

1. Każdy zegarek ma wartość rzeczywistą.
2. Zegarek może być resetowany (ustawiany na zero) równocześnie z jakimś zdarzeniem (przejściem zgodnie z relacją $\rightarrow$ w systemie tranzycyjnym).

3. Z każdą krawędzią wiążemy jeden z więzów czasowych (ang.: *Clock coinstraints*), który musi być spełniony, aby przejście mogło nastąpić.

4. Podobnie, z każdym stanem wiążemy jeden z więzów i pozwalamy na upływ czasu w tym stanie, tylko jeśli jest on spełniony.

3.1 Więzy czasowe

Zdefiniujmy formalnie więzy czasowe:

- X - zbiór zegarków
- $\Phi(X)$ - język więzów czasów ϕ jest zdefiniowany przez gramatykę

$$\phi := x \leq c | c \leq x | x \leq c | c \leq x | \phi_1 \wedge \phi_2$$

gdzie x jest zegarkiem, a $c \in \mathbb{Q}$.

3.2 Iterpretacja czasowa (ang.: *Clock interpretation*)

Interpretacja czasowa ν to mapowanie ze zbioru X (zbioru zegarków) w $\mathbb{R}$ (nieujemne liczby rzeczywiste).

Mówimy, że interpretacja czasowa spełnia ϕ (clock coinstraint) nad X wtw gdy ϕ wylicza się do prawdy po podstawieniu na każdą zmienną x, w ϕ, $\nu(x)$.

Dla $\delta \in \mathbb{R}$ przez $\nu + \delta$ oznaczmy interpretację czasową, która przyporządkowuje każdemu zegarkowi X $\nu(x) + \delta$.

Dla $Y \subseteq X$, $v[Y := 0]$ oznacza interpretację, która przyporządkowuje 0 dla każdego $x \in Y$ i to co ν dla pozostałych.

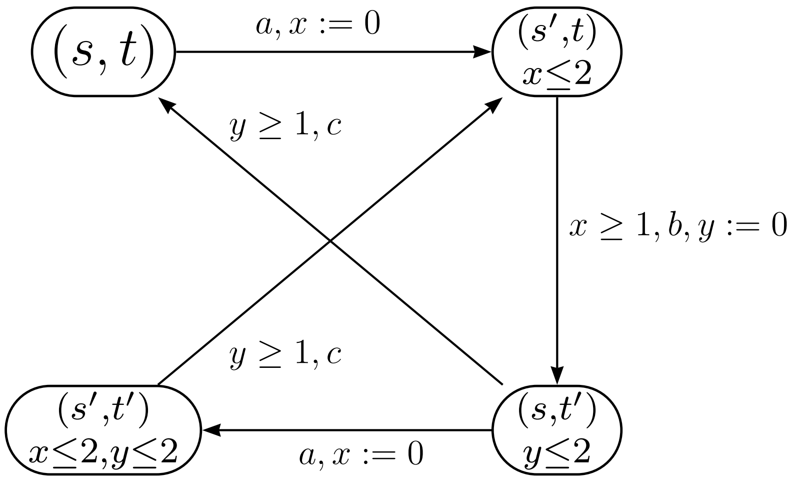
4 Automaty z czasem (ang.: *Timed automata*)

Automatem z czasem nazwiemy krotkę: $(L, L^0, \Sigma, \rightarrow, X, I)$

gdzie:

- L - zbiór lokacji
- L^0 - zbiór lokacji początkowych ($L^0 \subseteq L$)
- Σ - zbiór zdarzeń, etykiet (alfabet)
- $\rightarrow$ - relacja przejścia - zbiór krawędzi ($\rightarrow \subseteq L \times \Sigma \times \Phi(X) \times 2^X \times L$).
  Krotka $\langle l, a, \phi, \lambda, l' \rangle$ oznacza przejście z lokacji l do l' przy zdarzeniu a. ϕ oznacza, kiedy przejście jest aktywne. λ określa zbiór zegarków, które są resetowane.

Rysunek 1: Automaty $\mathcal{A}$ i $\mathcal{B}$



Rysunek 2: Złożenie równoległe automatów $\mathcal{A}$ i $\mathcal{B}$

- X - skończony zbiór zegarków
- I - funkcja przyporządkowująca każdej lokacji jeden z więzów ($I \in \Phi(X)^L$).

Złożenie równoległe jest definiowane analogicznie do złożenia bez więzów czasowych. Zainteresowanych odsyłam do [**?**].

W modelu z czasem stan definiujemy jako parę (s, ν), gdzie s jest lokacją, a ν jest interpretacją czasową.

Zmiana stanu może nastąpić na dwa sposoby:

- poprzez upływ czasu $(s, \nu) \xrightarrow{\delta} (s, \nu + \delta)$
- poprzez zmianę lokacji $(s, \nu) \xrightarrow{a} (s', \nu[\lambda := 0])$, gdzie λ - zbiór resetowanych zegarków.

5 Po co to wszystko?

Automaty z czasem są używane do modelowania systemów współbieżnych. Złożone systemy modelujemy za pomocą złożenia równoległego automatów modelujących ich komponenty.

Aby móc wnioskować o zachowaniu się systemu modelowanego przez automat potrzebujemy mechanizmów jego analizy.

Takim podstawowym mechanizmem jest algorytm znajdujący wszystkie stany osiągalne przez automat (ang.: *Reachability algorithm*). Należy zwrócic uwagę, iż jest to zadanie nietrywialne, gdyż liczba stanów, które może osiągnąć automat z czasem jest nieskończona.

Aby zrealizować ten algorytm przedstawiamy zbiór stanów w skończonej postaci.

5.1 Regiony czasowe (ang.: *Time regions*)

Lemat 1 *Zauważmy, że możemy sprowadzić wszystkie stałe występujące w więzach czasowych do wartości całkowitych mnożąc je przez najmniejszą wspólną wielokrotność n mianowników ich części ułamkowych. Gdy dla tak zmodyfikowanego automatu znajdziemy osiągalne stany, możemy wrócić do osiągalnych stanów automatu wejściowego dzieląc wartości przypisane zegarkom przez to samo n.*

Maksymalny możliwy rozrost stałych jest kwadratowy (względem długości stałych).

Dalej bez straty ogólności zakładamy, że wszystkie stałe są liczbami całkowitymi.

Pomysł polega na tym, że jeśli stany dotyczące tej samej lokacji zgadzają się co do części całkowitych wartości zegarków i uporządkowania ich części ułamkowych to zachowują się podobnie.

Dodatkowo zauważmy, że każda wartośc zegarka większa niż największa stała z jaką dany zegarek jest porównywany jest nierozróżnialna.

Formalnie: Określmy relację równoważności między interpretacjami czasowymi $\nu \cong \nu'$. $\nu \cong \nu'$ wtedy i tylko wtedy, gdy trzy poniże warunki są spełnione:

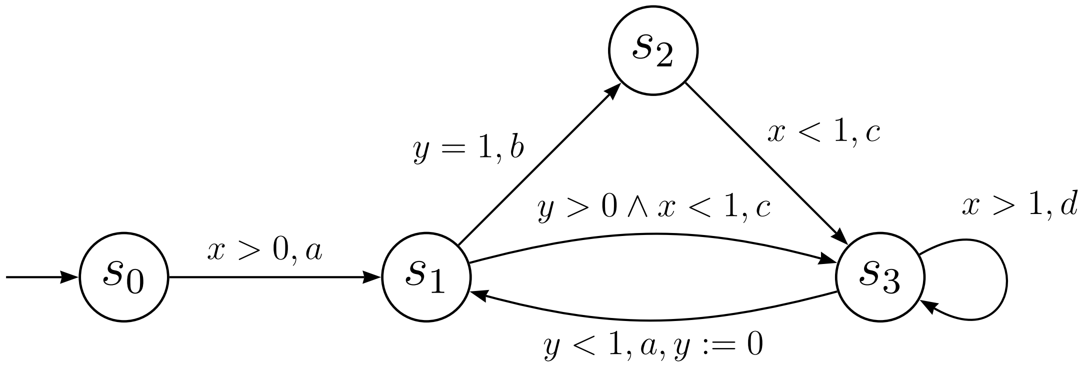
1. dla wszystkich $x \in X$ albo $\nu(x) \geq c_x$ i $\nu'(x) \geq c_x$ albo $\lfloor \nu(x) \rfloor = \lfloor \nu'(x) \rfloor$
2. dla wszystkich $x, y \in X$ takich, że $\nu(x) \leq c_x$ i $\nu(y) \leq c_x$ zachodzi $\{\nu(x)\} \leq \{\nu(y)\} \iff \{\nu'(x)\} \leq \{\nu'(y)\}$, gdzie $\{y\}$ - część ułamkowa y
3. dla wszystkich $x \in X$ takich, że $\nu(x) \leq c_x$ zachodzi $\{\nu(x)\} = 0 \iff \{\nu'(x)\} = 0$.

Klasy równoważności powyższej relacji nazwiemy regionami czasowymi.

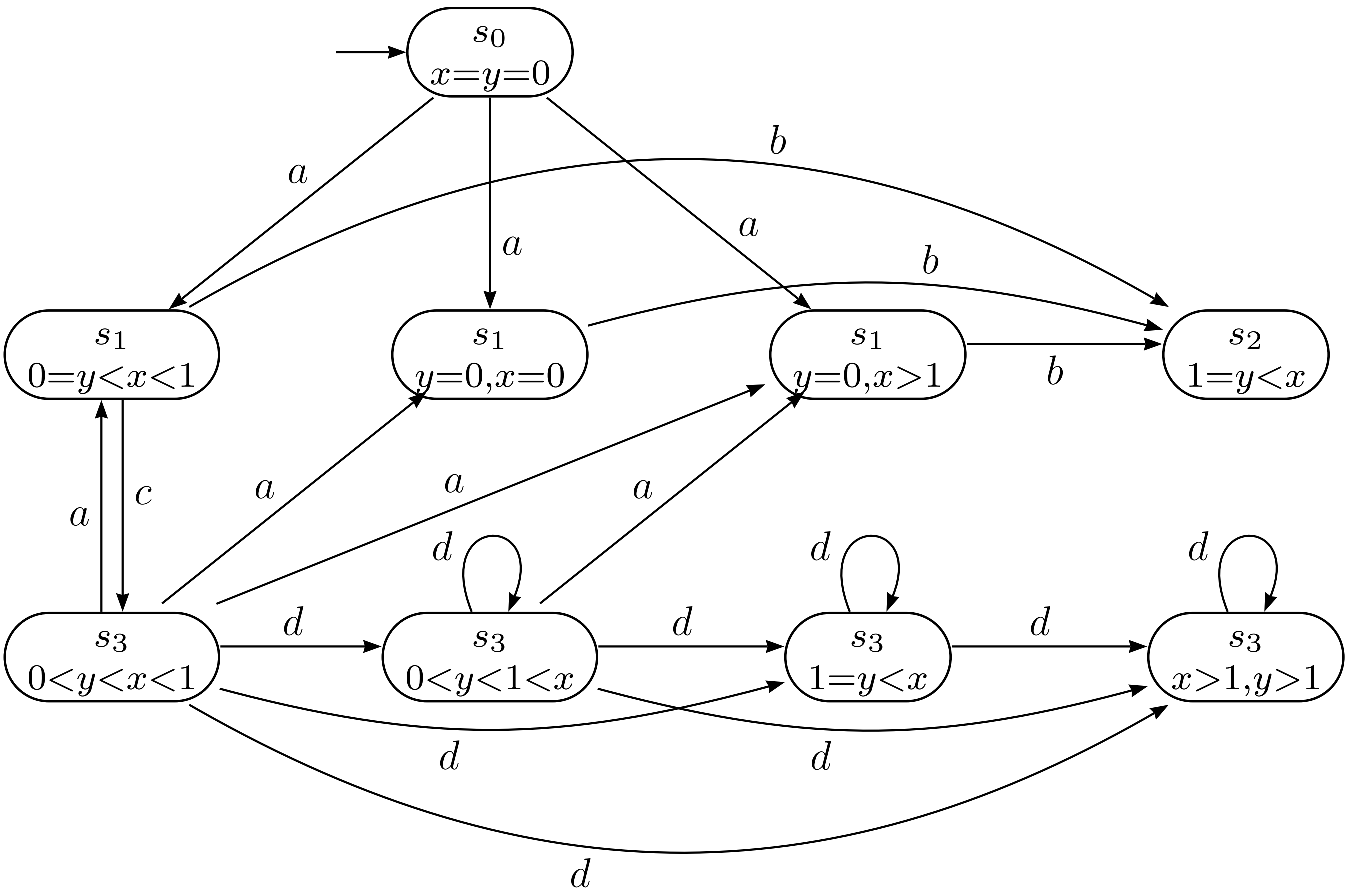
Lemat 2 *Liczba regionów czasowych jest ograniczona przez*

$$|X|! \cdot 2^{|X|} \cdot \prod_{x \in X} (2c_x + 2)$$

czyli jest wartości skończoną.



Rysunek 3: Automat z czasem $\mathcal{A}$



Rysunek 4: Automat regionów $\mathcal{R}(\mathcal{A})$

Lemat 3 *Jeśli* $\nu_1 \cong \nu_2$ *i* $(s,\nu_1) \stackrel{a}{\Longrightarrow} (s',\nu_1')$, *wtedy istnieje interpretacja czasowa* ν_2' *taka, że* $\nu_1' \cong \nu_2'$ *i* $(s,\nu_2) \stackrel{a}{\Longrightarrow} (s',\nu_2')$.

Z powyższego lematu wynika [**?**], że możemy skonstruować skończenie stanowy automat, który jest bisymulacyjnie równoważny wejściowemu automatu $\mathcal{A}$ o nieskończonej liczbie stanów.

Ten automat o skończonej liczbie stanów nazywany jest automatem regionów (ang.: *Region automaton*) i oznaczany przez $\mathcal{R}(\mathcal{A})$.

Regionem (ang.: *Region*) nazywamy parę $(s,[\nu])$. Liczba regionów jest skończona.

Spróbujmy zbudować automat regionów $\mathcal{R}(\mathcal{A})$. Niech $\mathcal{A} = (S, S^0, \Sigma, \to, X, I)$ będzie automatem z czasem. Wtedy:

1. Stany $\mathcal{R}(\mathcal{A})$ są postaci $(s,[\nu])$, gdzie $s \in S$ i ν jest interpretacją czasową.
2. Stany początkowe są postaci $(s_0,[\nu])$, gdzie $s_0 \in S^0$ i $\nu(x) = 0$ dla wszystkich $x \in X$.
3. $\mathcal{R}(\mathcal{A})$ ma przejście $((s,[\nu]), a, (s',[\nu']))$ wtedy i tylko wtedy, gdy $(s,\omega) \stackrel{a}{\Longrightarrow} (s',\omega')$ dla pewnych $\omega \in [\nu]$ i $\omega' \in [\nu']$.

6 ω-jezyki i ω-automaty

ω-jezyk L nad alfabetem Σ jest to pozbiór zbioru wszystkich nieskończonych słów nad tym alfabetem ($L \subseteq \Sigma^\omega$).

6.1 Przebieg

Zdefiniujmy przebieg ω-automatu dla przypadku bez czasu. Dla automatu $\mathcal{A}$ oraz dla nieskończonego słowa $\sigma = \sigma_1\sigma_2...$ nad alfabetem Σ powiemy że: $r : s_0 \xrightarrow{\sigma_1} s_1 \xrightarrow{\sigma_2} s_2 \xrightarrow{\sigma_3} ...$ jest przebiegiem dla $\mathcal{A}$ jeśli $s_0 \in Q_0$ oraz $s_{i-1} \xrightarrow{\sigma_i} s_i$ zachodzi dla każdego $i \geq 1$

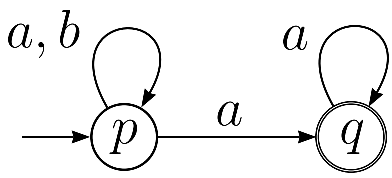
Określmy $inf(r)$ jako zbiór stanów pojawiających się w przebiegu nieskończenie wiele razy.

6.2 ω-automat Buchiego

ω-automat Buchiego definiujemy rozszerzając pojęcie systemu tranzycyjnego o zbiór stanów akceptujących.

Automat Buchiego akceptuje przebieg r jeśli $inf(r) \cap F \neq \emptyset$

ω-język generowany przez ω-automat Buchiego $\mathcal{A}$ to zbiór tych słów nad Σ^ω, dla których istnieje przebieg akceptowany przez $\mathcal{A}$.

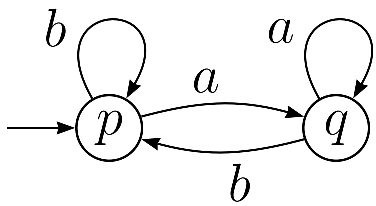


Rysunek 5: Automat Buchiego

Jezyk akceptowany przez automat na rysunku **??** to:

$$L_0 = (a + b)^* a^\omega$$

6.3 ω-automat Mullera (ang.: *Muller automaton*)



Rysunek 6: Deterministyczny automat Mullera akceptujący $(a + b)^* a^\omega$

ω-automat Mullera $\mathcal{A}$ jest to system tranzycyjny $\langle \Sigma, S, S^0, E \rangle$, rozszerzony o rodzinę zbiorów stanów akceptujących $\mathcal{F} \subseteq \mathcal{P}(S)$. Przebieg r jest przebiegiem akceptującym wtedy i tylko wtedy, gdy $inf(r) \in \mathcal{F}$. Czyli r jest przebiegiem akceptującym wtw jeśli zbiór stanów powtarzających się nieskończoną ilość razy podczas przebiegu r jest równy pewnemu zbiorowi należącemu do $\mathcal{F}$.

Język akceptowany przez $\mathcal{A}$ jest zdefiniowany tak samo jak w przypadku automatów Bucciego.

Klasa języków akceptowanych przez ω-automaty Mullera jest taka sama jak ta akceptowana przez ω-automaty Buchiego i przez deterministyczne ω-automaty Mullera. Nazywana jest ona klasą języków ω-regularnych.

Okazuje się, że języki ω-regularne mają dość dużo przyjemnych własności. Są one opisane w [**?**].

6.4 ω-automaty z czasem

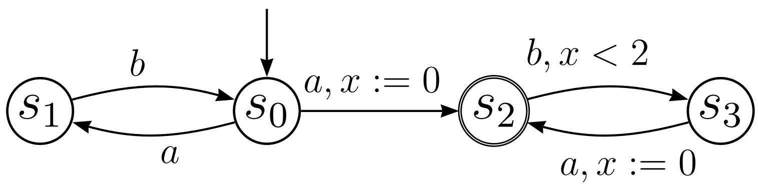
Pojęcie ω-automatu naturalnie przenosi się na przypadek, gdy rozpatrujemy model z czasem. Zamiast zwykłych nieskończonych słów rozpatrujemy wtedy słowa złożone z krotek $\langle \sigma, \tau \rangle$ gdzie $\sigma \in \Sigma$ natomiast τ jest czasem, w którym wystąpiło zdarzenie.

Naturalnej zmianie ulega również pojęcie przebiegu, gdyż musimy uwzględnić czy tranzycja jest aktywna (zależnie od stanu zegarków w automacie i więzów nałożonych na krawędzie i lokacje).

Podobnie jak w przypadku bez czasu, wprowadzone zostało pojęcie języków ω-regularnych z czasem [**?**]. Jest to klasa języków akceptowana przez ω-automaty Buchiego z czasem, jak również przez ω-automaty Mullera z czasem.

6.5 Przykładowe języki akceptowane przez automaty z czasem

6.5.1 Język L_{crt}



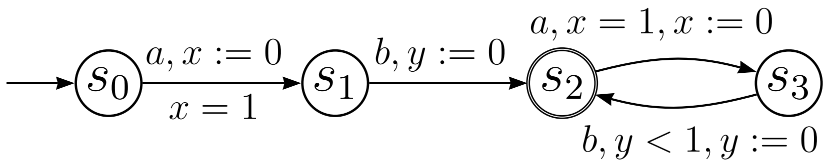
Rysunek 7: Automat z czasem Buchiego akceptujący L_{crt}

Automat widoczny na rysunku **??** akceptuje język L_{crt} nad alfabetem $\{a, b\}$.

$$L_{crt} = \{((ab)^\omega, \tau) | \exists i \forall j \geq i (\tau_{2j} < \tau_{2j-1} + 2)\}$$

Zauważmy, że po dotarciu do stanu s_2, automat zawsze (po odczytaniu a) resetuje zegarek x i ma pewność, że b zostanie odczytane przed upływem dwóch jednostek czasu.

6.5.2 Język $L_{converge}$



Rysunek 8: Automat z czasem akceptujący $L_{converge}$

Automat widoczny na rysunku **??** akceptuje język $L_{converge}$ nad alfabetem $\{a, b\}$.

$$L_{converge} = \{((ab)^{\omega}, \tau) | \forall i(\tau_{2i-1} = i \wedge (\tau_{2i} - \tau_{2i-1} > \tau_{2i+2} - \tau_{2i+1}))\}$$

Każde słowo akceptowane przez ten automat ma taką ciekawą właściwość, że ciąg różnic czasu pomiędzy kolejnymi wystąpieniami a i b jest ciągiem ściśle malejącym, np.:

$$(a, 1) \rightarrow (b, 1.5) \rightarrow (a, 2) \rightarrow (b, 2.25) \rightarrow (a, 3) \rightarrow (b, 3.125) \rightarrow \ldots$$

6.6 Przykładowe języki nie ω-regularne

1. $L_1 = \{((ab)^{\omega}, \tau) | \forall i((\tau_{2i} - \tau_{2i-1}) < (\tau_{2i+2} - \tau_{2i+1}))\}$

   Każde słowo opisane przez język L_1 ma taką własność, że ciąg różnic czasu pomiędzy kolejnymi wystąpieniami a i b jest ciągiem ściśle rosnącym.

2. $L_2 = \{(a^{\omega}, \tau) | \forall i(\tau_i = 2^i)\}$