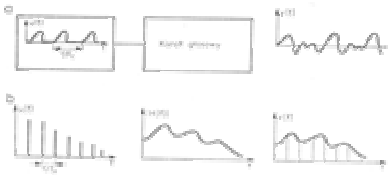
Rozpoznawanie mowy

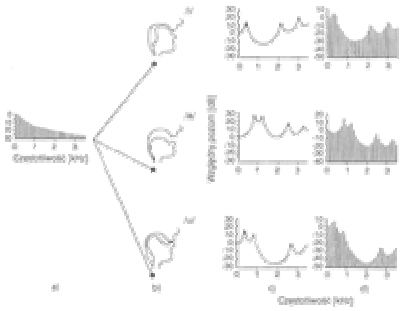
Artykulacja i percepcja mowy,
Niejawne Modele Markowa,

Wytwarzanie dźwięków mowy

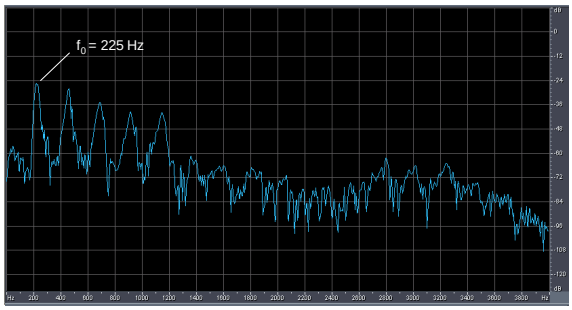
- Artykulacja bazuje na dwóch układach biologicznych:
  - źródle sygnału pobudzenia dźwiękowego (krtań z wiązadłami głosowymi - generacja podstawowego tonu krtaniowego F_0),
  - kanale głosowym (gardło, jama ustna, jama nosowa)
- Odpowiednie ustawienie języka i ust powoduje zmianę objętości poszczególnych komór oraz ich wzajemne sprzężenie, co wpływa na ich częstotliwość drgań własnych - co przejawia się lokalnymi maksimami w widmie - **formantami**



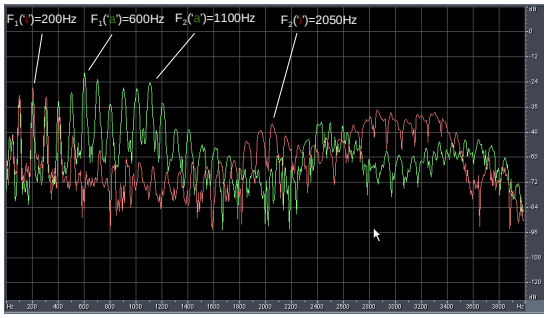
Formanty



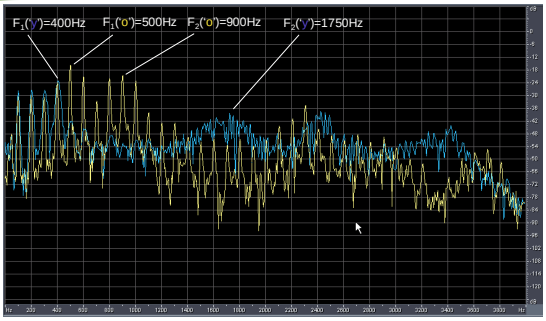
Częstotliwość krtaniowa – f_0



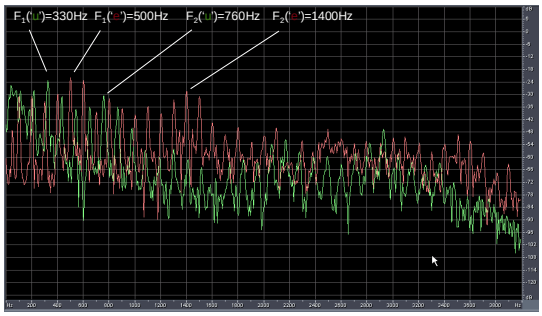
Widmo samogłosek *a* oraz *i*



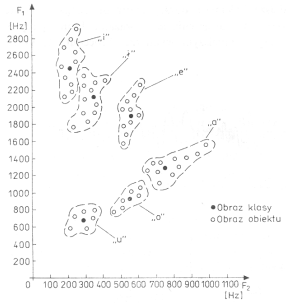
Widmo samogłosek *o* oraz *y*



Widmo samogłosek *e* oraz *u*

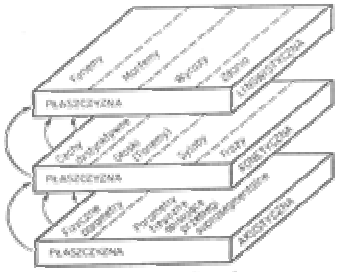


Rozpoznawanie samogłosek



Percepcja mowy

wypowiedź - zdanie - wyraz - morfem - fonem

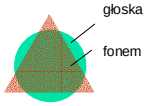


Przykładowe cechy sygnału mowy

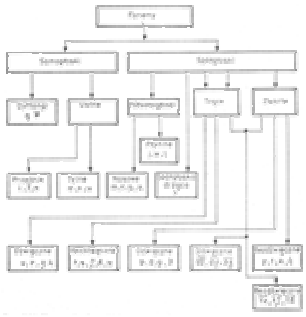
- częstotliwość podstawowa tonu krtaniowego,
- częstotliwości, stosunki amplitudowe oraz szerokość pasm formantów,
- uśrednione widmo amplitudowe,
- widmo krótkoterminowe,
- względne długości czasu wypowiedzi poszczególnych elementów fonetycznych,
- obwiednia czasowa amplitudy lub natężenie dźwięku,
- parametry analizy przejść przez zero sygnału mowy,
- parametry liniowego kodowania predykcyjnego,
- charakterystyki prozodyczne.

Głoska a fonem

- Głoska
  - najmniejsza cząstka dźwiękowa języka;
  - realizacja fonemu;
  - zawiera cechy fonemiczne (dystynktywne) oraz fonetyczne (nieistotne dla procesu komunikacji - wymiany informacji lingwistycznych - np. związane z płcią, wiekiem, emocjami,...)
- Fonem
  - model głoski,
  - abstrakcyjny symbol dźwięku występującego w danym języku,
  - minimalny segment dźwiękowy, który może odróżniać znaczenia,



Fonemy języka polskiego



Warianty fonemu

- „fura”, „trawa”, „rok”
  - „fura” - głoska „r” uderzeniowa dźwięczna,
  - „trawa” - głoska „r” uderzeniowa bezdźwięczna,
  - „rok” - drżąca dźwięczna
- Alfon - wariant fonemu związany z rodzajem kontekstu
- Dwa rodzaje zapisu dźwięków mowy (transkrypcja):
  - fonematyczna,
  - alfoniczna
- Klasyfikacje fonemów:
  - ze względu na sposób artykulacji,
  - ze względu na cechy dystynktywne,

Dyskryminacja fonemów języka polskiego

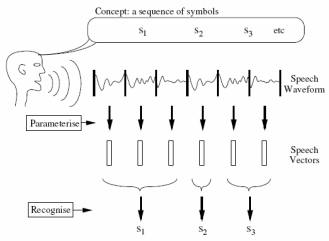
Fonem:

- **spółgłoskowy** - samogłoskowy:
  - do 3.5 kHz występują więcej niż 3 formanty,
  - widmo zawiera antyformanty,
- **ponadkrtaniowy** - krtaniowy:
  - widmo ciągłe,
  - F_1 tworzy maksimum obwiedni tylko dla widma mieszanego (prążkowo-ciągłego),
- **nosowe** - ustne:
  - przynajmniej 5 formantów do 3.2 kHz,
- **łagodne** - rozproszone:
  - wolna zmiana amplitudy < 10dB/10ms

cd. dyskryminacji fonemów PL

- **skupione** - rozproszone:
  - występowanie wysokich poziomów widma w środkowych zakresach częstotliwości,
  - dla samogłosek F_1 jest mniejsze niż 400 Hz,
  - dla spółgłosek ponadkrtaniowych - w widmie od 1..4 kHz jeden formant o amplitudzie znacznie przewyższającej pozostałe, a dla spółgłosek nosowych formanty w zakresie 1.8 - 3.0 kHz mają największe amplitudy,
- **jasne** - ciemne:
  - części widma z zakresu wyższych częstotliwości mają wyższe amplitudy,
- niskotonowe - wysokotonowe,
- długie - krótkie,
- dźwięczne - bezdźwięczne.

Rozpoznawanie mowy



Problemy:

1. Artykulacja **różnych** „symboli” może generować **podobnie** brzmiące dźwięki, czyli mapowanie symbol-dźwięk nie jest jednoznaczne i jednocześnie **te same** „symbole” mogą być w **różny** sposób wymawiane przez różne osoby.
2. W przypadku mowy ciągłej, nie można traktować sygnału akustycznego jako sekwencji skonkatenowanych statycznych wzorców „symboli” – nie istnieje możliwość identyfikacji granic „symboli” z samego przebiegu

Rozpoznawanie izolowanych słów

Niech każde wypowiedziane słowo będzie reprezentowane jako sekwencja wektorów cech sygnału akustycznego lub obserwacji O zdefiniowanych jako:

$$O = o_1, o_2, \ldots o_T$$

gdzie O_t to wektor cech sygnału akustycznego w chwili t. Problem rozpoznawania izolowanych słów można sprowadzić do obliczenia:

$$\arg\max_i \{P(w_i \mid O)\}$$

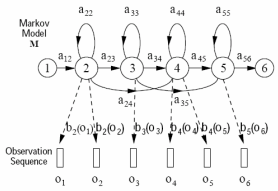
gdzie i to indeks słowa w zbiorze rozpoznawanych słów. Prawdopodobieństwo nie jest wyliczane wprost, a po zastosowaniu reguły Beysa:

$$P(w_i \mid O) = \frac{P(O \mid w_i)P(w_i)}{P(O)}$$

HMM – **H**idden **M**arkov **M**odels

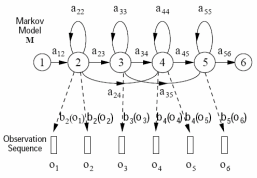
Założenia rozpoznawania mowy z wykorzystaniem *niejawnych modeli Markowa*:

1. Sekwencja obserwowanych wektorów cech akustycznych, związana z określonym słowem, generowana jest przez dany model Markowa reprezentowany przez automat skończony (n-stanowy) o określonej topologii przejść.



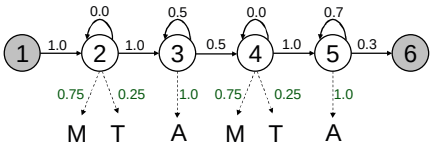
HMM cd.

2. Łączne prawdopodobieństwo wygenerowania obserwacji *O* przez model Markowa (przechodząc przez sekwencję stanów X) jest równe iloczynowi prawdopodobieństw przejść między stanami oraz prawdopodobieństw wygenerowania określonych obserwacji w danym stanie.



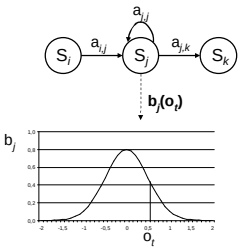
$$P(O, X|M) = a_{12}b_2(o_1)a_{22}b_2(o_2)a_{23}b_3(o_3)\ldots$$

Przykład

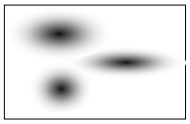


Wynik:
P(„MAMA") = 1.0*0.75*1.0*1.0*0.5*0.75*1.0*1.0*0.3 = **0.084**
P(„MAMAA") = 1.0*0.75*1.0*1.0*0.5*0.75*1.0*1.0*0.7*1.0*0.3 = **0.059**
P(„MAAMA") = 1.0*0.75*1.0*1.0*0.5*1.0*0.5*0.75*1.0*1.0*0.3 = **0.042**
P(„TATA") = 1.0*0.25*1.0*1.0*0.5*0.25*1.0*1.0*0.3 = **0.016**

Ogólna topologia HMM

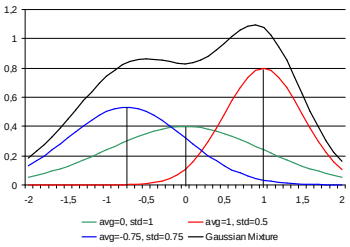


Jednomodalny rozkład gęstości prawdopodobieństwa (*Gaussian densities*) dla jednej zmiennej



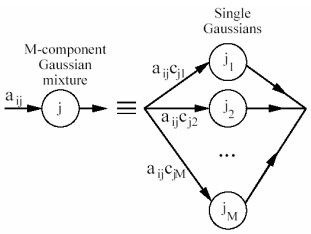
Wielomodalny gaussowski rozkład prawdopodobieństwa (*Gaussian mixture densities*) dla dwóch zmiennych

Specyfikacja wyjściowego prawdopodobieństwa – Gaussian Mixture Densities



$$b_j(o_t) = \prod_{s=1}^{S}\left[\sum_{m=1}^{M_s} c_{jsm}\mathcal{N}(o_{st}; \mu_{jsm}, \Sigma_{jsm})\right]^{\gamma_s} \qquad \mathcal{N}(o; \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^n|\Sigma|}} e^{-\frac{1}{2}(o-\mu)'\Sigma^{-1}(o-\mu)}$$

Reprezentacja wielostanowa M-component Gaussian mixture



Podsumowanie: rozpoznawanie mowy – od teorii do praktyki

$$\arg\max_i \{P(w_i \mid O)\}$$

$$P(w_i \mid O) = \frac{P(O \mid w_i)P(w_i)}{P(O)} \qquad P(O \mid w_i) = P(O \mid M_i)$$

$$P(w_i \mid O) = \frac{P(O \mid M_i)P(w_i)}{P(O)}$$

Tworzenie modeli M_i polega na ustaleniu wartości ich parametrów (prawdopodobieństw) przejść między stanami a_{ij} oraz rozkładów gęstości prawdopodobieństw $b_j(o_t)$ w oparciu o zbiór uczący, z wykorzystaniem tzw. **procedury re-estymacji Baum-Welcha**.

HMM

W praktyce, tylko obserwacje są znane, natomiast związana z nimi sekwencja przejść między stanami modelu Markowa nie jest znana (jest niejawna, niedostępna, ukryta) – stąd nazwa tego podejścia to **Niejawne** Modele Markowa (ang. ***Hidden*** *Markov Models*).

Nie znając sekwencji przejść, prawdopodobieństwo wylicza się sumując je po wszystkich możliwych kombinacjach sekwencji X = x(1), x(2), x(3)… x(T):

$$P(O|M) = \sum_X a_{x(0)x(1)} \prod_{t=1}^{T} b_{x(t)}(o_t) a_{x(t)x(t+1)}$$

gdzie x(0) – to stan początkowy (wejściowy), a x(T+1) to stan końcowy (wyjściowy) modelu. Alternatywnie, można sumować prawdopodobieństwa tylko najbardziej prawdopodobnych sekwencji:

$$\hat{P}(O|M) = \max_X \left\{ a_{x(0)x(1)} \prod_{t=1}^{T} b_{x(t)}(o_t) a_{x(t)x(t+1)} \right\}$$

W praktyce korzysta się z efektywnych procedur rekurencyjnych.

Algorytm re-estymacji Baum-Welcha

Celem algorytmu Baum-Welcha jest estymacja wartości średnich (**μ**) i wariancji (**Σ**) wszystkich pojedynczych rozkładów gaussowskich, wszystkich stanów modelu Markowa.

$$b_j(\boldsymbol{o}_t) = \frac{1}{\sqrt{(2\pi)^n |\boldsymbol{\Sigma_j}|}} e^{-\frac{1}{2}(\boldsymbol{o}_t - \boldsymbol{\mu}_j)' \boldsymbol{\Sigma}_j^{-1} (\boldsymbol{o}_t - \boldsymbol{\mu}_j)}$$

W przypadku modelu HMM o jednym stanie najlepszą estymacją **μ** oraz **Σ** są średnie:

$$\hat{\boldsymbol{\mu}}_j = \frac{1}{T} \sum_{t=1}^{T} \boldsymbol{o}_t$$

$$\boldsymbol{\Sigma}_j = \frac{1}{T} \sum_{t=1}^{T} (\boldsymbol{o}_t - \boldsymbol{\mu}_j)(\boldsymbol{o}_t - \boldsymbol{\mu}_j)'$$

Algorytm Baum-Welcha cd.

W praktyce modele Markowa są wielostanowe, a jako że sekwencja przejść między stanami nie jest znana, nie można wprost przydzielić obserwacji do poszczególnych stanów i na tej podstawie wyliczyć wartości parametrów modelu. Stąd algorytm Bauma-Welcha ma charakter iteracyjny. W pierwszym kroku następuje przybliżone (równomierne) przydzielenie obserwacji do poszczególnych stanów, co daje pierwsze przybliżenie wartości **μ** oraz **Σ**. Natomiast w kolejnych iteracjach algorytmu, poszukiwane są najbardziej prawdopodobne sekwencje przejść i wyznaczane parametry HMM są re-estymowane. Jako, że obliczanie prawdopodobieństwa danej sekwencji obserwacji polega na sumowaniu prawdopodobieństw wszystkich możliwych sekwencji przejść między stanami modelu HMM, każdy wektor obserwacji o_t ma udział w obliczaniu parametrów **μ** oraz **Σ** każdego stanu *j*. Stąd zamiast poszukiwać przydziału każdego wektora obserwacji do określonego stanu, każda obserwacja jest przydzielana do każdego stanu z wagą równą prawdopodobieństwu przebywania w danym stanie, podczas jej obserwowania. I tak niech $L_j(t)$ oznacza prawdopodobieństwo przebywania w stanie *j* w chwili *t*. Estymowane wartości **μ** oraz **Σ** wyniosą odpowiednio:

$$\hat{\boldsymbol{\mu}}_j = \frac{\sum_{t=1}^{T} L_j(t) \boldsymbol{o}_t}{\sum_{t=1}^{T} L_j(t)}$$

$$\hat{\boldsymbol{\Sigma}}_j = \frac{\sum_{t=1}^{T} L_j(t)(\boldsymbol{o}_t - \boldsymbol{\mu}_j)(\boldsymbol{o}_t - \boldsymbol{\mu}_j)'}{\sum_{t=1}^{T} L_j(t)}$$

Forward-backward algorythm

W celu obliczenia wartości $L_j(t)$ stosuje się tzw. algorytm „*forward-backward*". Prawdopodobieństwo „*forward*" $\alpha_j(t)$ dla N-stanowego modelu M jest zdefiniowane następująco:

$$\alpha_j(t) = P(\boldsymbol{o}_1, \ldots, \boldsymbol{o}_t, x(t) = j|M).$$

$\alpha_j(t)$ jest prawdopodobieństwem zaobserwowania pierwszych *t* wektorów mowy i osiągnięciem stanu *j* w chwili *t*. Prawdopodobieństwo to jest wyliczane na podstawie rekurencyjnej formuły:

$$\alpha_j(t) = \left[\sum_{i=2}^{N-1} \alpha_i(t-1) a_{ij} \right] b_j(\boldsymbol{o}_t).$$

Prawdopodobieństwo „*forward*"

Warunki brzegowe:

$$\alpha_1(1) = 1$$

$$\alpha_j(1) = a_{1j} b_j(\boldsymbol{o}_1) \quad , \quad 1 < j < N$$

$$\alpha_N(T) = \sum_{i=2}^{N-1} \alpha_i(T) a_{iN}.$$

Z definicji $\alpha_j(t)$:

$$P(\boldsymbol{O}|M) = \alpha_N(T).$$

Prawdopodobieństwo „*backward*"

Prawdopodobieństwo „*backward*" $\alpha_j(t)$ dla N-stanowego modelu M jest zdefiniowane następująco:

$$\beta_j(t) = P(\boldsymbol{o}_{t+1}, \ldots, \boldsymbol{o}_T | x(t) = j, M).$$

Prawdopodobieństwo to jest wyliczane na podstawie rekurencyjnej formuły:

$$\beta_i(t) = \sum_{j=2}^{N-1} a_{ij} b_j(\boldsymbol{o}_{t+1}) \beta_j(t+1)$$

Warunki brzegowe:

$$\beta_i(T) = a_{iN} \quad , \quad 1 < i < N$$

$$\beta_1(1) = \sum_{j=2}^{N-1} a_{1j} b_j(\boldsymbol{o}_1) \beta_j(1).$$

Prawdopodobieństwo $L_j(t)$

Z definicji prawdopodobieństwa „forward" i „backward":

$$\alpha_j(t)\beta_j(t) = P(\boldsymbol{O}, x(t) = j|M).$$

Stąd poszukiwane prawdopodobieństwo przebywania w stanie *j* w chwili *t* wynosi:

$$\begin{aligned} L_j(t) &= P(x(t) = j|\boldsymbol{O}, M) \\ &= \frac{P(\boldsymbol{O}, x(t) = j|M)}{P(\boldsymbol{O}|M)} \\ &= \frac{1}{P}\alpha_j(t)\beta_j(t) \end{aligned}$$

gdzie:

$$P = P(\boldsymbol{O}|M).$$

Algorytm Baum-Welcha

1. Dla każdego estymowanego parametru modelu Markowa zaalokuj pamięć (tzw. akumulator) na wartości mianownika i licznika równań:

$$\hat{\boldsymbol{\mu}}_j = \frac{\sum_{t=1}^{T} L_j(t)\boldsymbol{o}_t}{\sum_{t=1}^{T} L_j(t)}$$

$$\hat{\boldsymbol{\Sigma}}_j = \frac{\sum_{t=1}^{T} L_j(t)(\boldsymbol{o}_t - \boldsymbol{\mu}_j)(\boldsymbol{o}_t - \boldsymbol{\mu}_j)'}{\sum_{t=1}^{T} L_j(t)}$$

2. Oblicz prawdopodobieństwo „forward" i „backward" dla każdego stanu *j* w chwili *t*.
3. Dla każdego stanu *j* oraz chwili *t*, korzystając z $L_j(t)$ i aktualnego wektora obserwacji o_t, aktualizuj akumulatory tego stanu.
4. Wykorzystaj otrzymane wartości akumulatorów do wyznaczenia nowych wartości parametrów modelu.
5. Jeżeli wartość $P = P(O|M)$ dla tej iteracji nie jest większa niż wartość P z poprzedniej iteracji zakończ działanie algorytmu, w przeciwnym razie powtarzaj powyższe kroki algorytmu (2 – 5) wykorzystując nowe wartości re-estymowanych parametrów modelu.

Faza rozpoznawania - Viterbi decoding

$$\arg\max_i \{P(w_i|O)\}$$

$$P(w_i|O) = \frac{P(O|M_i)P(w_i)}{P(O)}$$

Stąd teoretycznie do rozpoznawania można wykorzystać prawdopodobieństwo „forward" – $\alpha_N(T)$, gdyż:

$$\alpha_N(T) = P(O|M)$$

Jednak w praktyce rozpoznawanie oparte jest na poszukiwaniu najbardziej prawdopodobnej sekwencji stanów modelu Markowa, co umożliwia łatwe przejście do rozpoznawania mowy ciągłej. Obliczenia są analogiczne jak w przypadku liczenia prawdopodobieństwa „forward", z tą różnicą, że suma jest zastąpiona operatorem *max*:

$$\phi_j(t) = \max_i \{\phi_i(t-1)a_{ij}\}\, b_j(\boldsymbol{o}_t).$$

Viterbi decoding cd.

Warunki brzegowe:

$$\phi_1(1) = 1$$

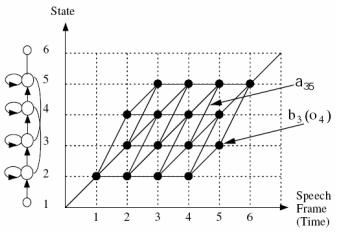
$$\phi_j(1) = a_{1j}b_j(\boldsymbol{o}_1) \quad , \quad 1 < i < N$$

$$\phi_N(T) = \max_i \{\phi_i(T)a_{iN}\}$$

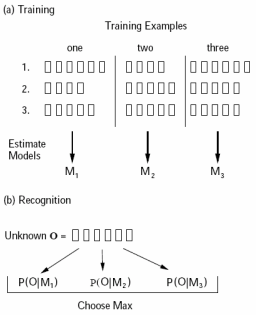
Bezpośrednie obliczanie prawdopodobieństwa, ze względu na wielokrotne mnożenie wartości mniejszych od jedności, prowadzi do błędów reprezentacji typu „underflow". Stąd równanie rekurencyjne Viterbiego zawiera logarytmy:

$$\psi_j(t) = \max_i \{\psi_i(t-1) + log(a_{ij})\} + log(b_j(\boldsymbol{o}_t)).$$

Reprezentacja graficzna algorytmu Viterbiego



Podsumowanie: wykorzystanie HMM do rozpoznawania izolowanych słów



Rozpoznawanie mowy ciągłej