WYKŁAD 10

- **Kompresja krzywych dyskretnych**

Kompresja krzywych dyskretnych

$$SK = \frac{KP}{KW}$$

SK - stopień kompresji krzywej.
KP [bajt] - obszar pamięci zajmowany przez ***kod pierwotny*** krzywej.
KW [bajt] - obszar pamięci zajmowany przez ***kod wynikowy*** krzywej

- W przypadku krzywych dyskretnych:

kodem pierwotnym krzywej jest ciąg par współrzędnych x, y kolejnych punktów krzywej: $(x_1, y_1), (x_2, y_2), \ldots\ldots, (x_n, y_n)$,

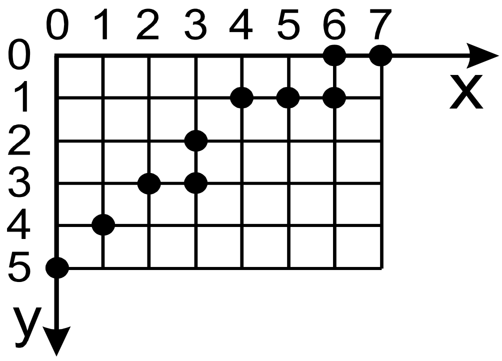
kodem wynikowym krzywej może być na przykład :
- kod łańcuchowy (chain code) o stałej długości (3 bity/punkt),
- lub różnicowy kod łańcuchowy

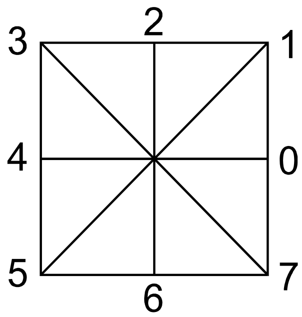
Powyższa kompresja jest bezstratna (*lossless compression)*

Krzywa dyskretna - zbiór punktów (piksli) siatki prostokątnej (rastru prostokątnego) z których każdy (oprócz punktów końcowych) posiada **nie mniej** niż 2 i **nie więcej** niż 3 sąsiadów odpowiednio skonfigurowanych (w sensie sąsiedztwa 8-mio lub 4-spójnego). Punkty końcowe: 1-2 sąsiadów.

Krzywe dzielimy na: otwarte, zamknięte.

Kierunki:





Reprezentacja krzywych

- Ciąg par współrzędnych x, y kolejnych punktów krzywej

$(x_1, y_1), (x_2, y_2), \ldots\ldots, (x_n, y_n)$,

(0,5), (1,4),...............,(7,0)   (krzywa z przykładu)

- Kod łańcuchowy (*chain code*) o stałej długości (3 bity/punkt)

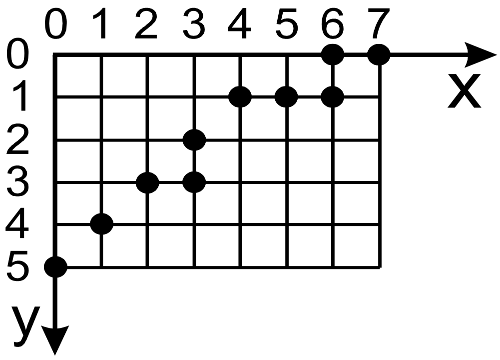
(0,5) 001001000010001000.....000

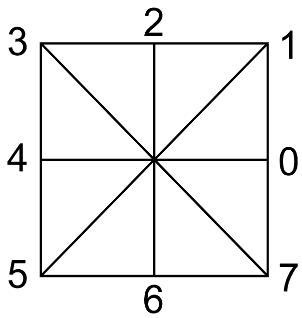
(0,5) - współrzędne punktu początkowego krzywej z przykładu

001 - kod kierunku „1”

Długość kodu **nie zależy** do kształtu krzywej (określonego zmianami kierunków pomiędzy kolejnymi punktami krzywej).

• Różnicowy kod łańcuchowy
(o zmiennej długości, średnio 2 bity / punkt, długość kodu zależy od kształtu krzywej).





Przypisania:

zmiana nachylenia	0	+1	-1	+2	-2	+3	-3	4
kod	0	01	011	0111	01111	011111	0111111	01111111

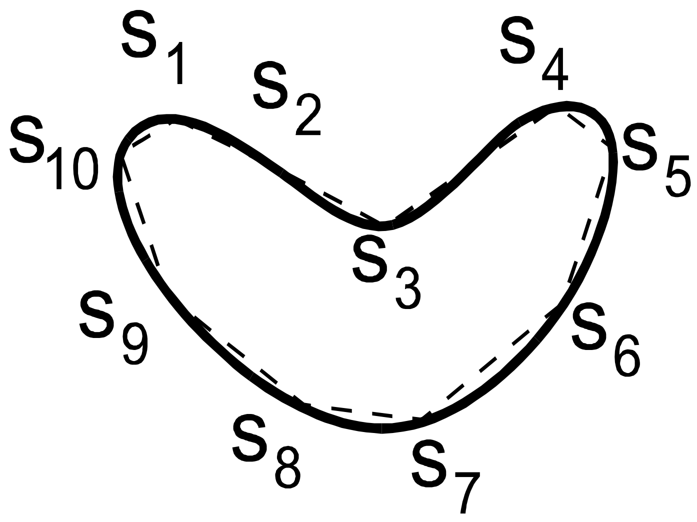
+1, +2, +3, 4 - **zmiana** nachylenia o 1,2,3,4 w kierunku dodatnim
- 1, -2, -3, 4 - **zmiana** nachylenia o 1,2,3,4 w kierunku ujemnym

<u>Krzywa z przykładu:</u>

(0,5) 001 0 01101110110110011101111

Dla jednoznacznego opisu krzywej powyższy kod musi zawierać:
(0,5) - współrzędne punktu początkowego krzywej z przykładu
001 - kod łańcuchowy
0   - różnicowy kod łańcuchowy

Interpolacja krzywych dyskretnych odcinkami linii prostych dyskretnych



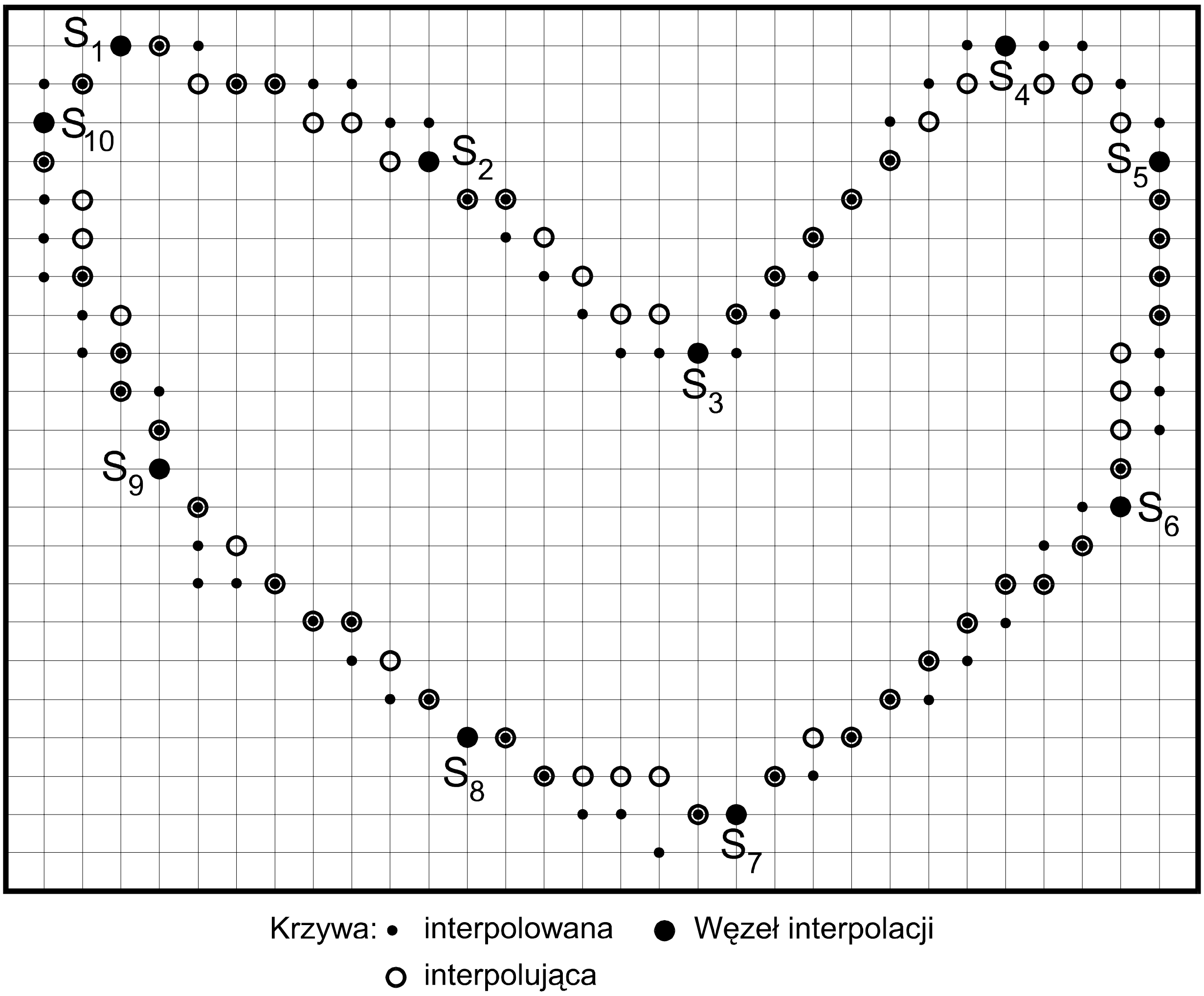
Krzywe **dyskretne**:
- krzywa **pierwotna** (*interpolowana*) ——— ,
- krzywa **odtworzona** (*zdekompresowana, interpolująca*) ---------- - jest to zbiór kolejnych odcinków linii prostej dyskretnej S_1-S_2, S_2 -S_3,....,S_9-S_{10} .
S_1,......, S_{10} - **węzły interpolacji**

KP - obszar pamięci zajmowany przez ciąg par współrzędnych x, y kolejnych punktów krzywej pierwotnej: $(x_1, y_1), (x_2, y_2),......, (x_N, y_N)$,

KW - obszar pamięci zajmowany przez ciąg par współrzędnych x, y kolejnych węzłów interpolacji: $(x_{S1},y_{S1}), (x_{S2}, y_{S2}),...,(x_{Sn},y_{Sn})$.

Generalnie rzecz biorąc metoda ta daje w efekcie kompresję stratną (*lossy*), co oznacza, że kod krzywej pierwotnej różni się od kodu krzywej odtworzonej, jednak przy pewnych technikach stosowanych w ramach tej metody można uzyskać kompresję bezstratną (*lossless).*

Miara stratności kompresji ---- ***BŁĄD INTERPOLACJI***



Parametry interpolacji

1.Długość odcinka (***IW***) krzywej interpolowanej (pierwotnej) - jest to liczba punktów tego odcinka „od węzła do węzła" wraz z jednym z węzłów interpolacji.

Uwaga: każdy z odcinków krzywej interpolującej (odtworzonej) skonstruowany zgodnie ze **schematem dyskretyzacji Freemana** linii prostej.

2. Całkowita długość krzywej interpolowanej (***IMEM***) reprezentowana przez **kod pierwotny *KP***):

$$IMEM=\sum_{i=1}^{M} IW_i$$

gdzie: IW_i - długość i - tego odcinka krzywej interpolowanej

M - liczba odcinków krzywej interpolowanej.

3. Liczba węzłów interpolacji (***IMEMS***) reprezentowana przez **kod wynikowy** (***KW***):

IMEMS=M - dla krzywych zamkniętych,

IMEMS=M+1 - dla krzywych otwartych

4. ***Współczynnik redukcji pamięci WRP*** określający **stopień kompresji *SK*** krzywej; ***WRP = SK***

$$WRP=\frac{N}{IMEMS}$$ - dla krzywych otwartych,

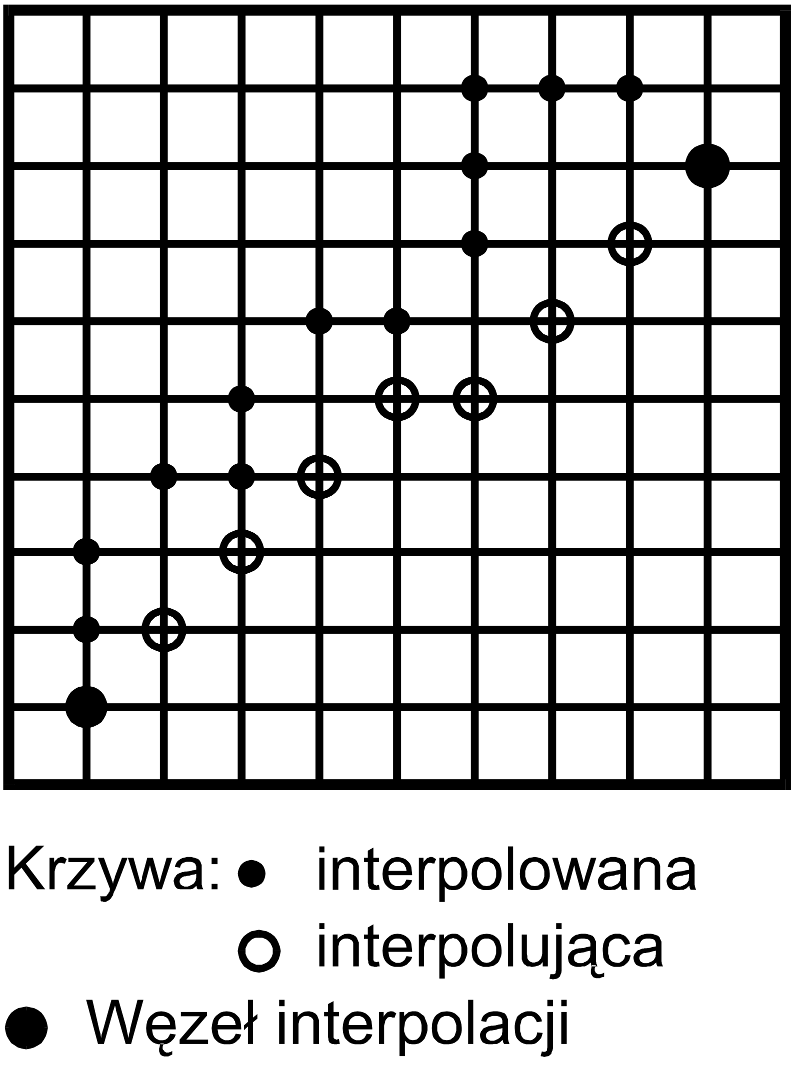
$$WRP=\frac{N+1}{IMEMS+1}$$ - dla krzywych zamkniętych, gdzie: N - liczba punktów (piksli) krzywej interpolowanej.

5. ***Błąd interpolacji*** będący miarą **stratności kompresji**

$$LD= LU-LW$$

gdzie:
LD - błąd interpolacji.
LU - liczba piksli (w tym przypadku punktów) siatki (rastru) zawartych pomiędzy krzywą **interpolowaną** (pierwotną)i **interpolującą (odtworzoną)** wraz z punktami należącymi do tych krzywych.
LW - liczba punktów wspólnych, czyli należących zarówno do krzywej interpolowanej jak i interpolującej.

Przykład:



LU = 27, *LW* = 2
błąd interpolacji:
LD = 27-2 = 25

stopień kompresji:
SK = *WRP* = 7

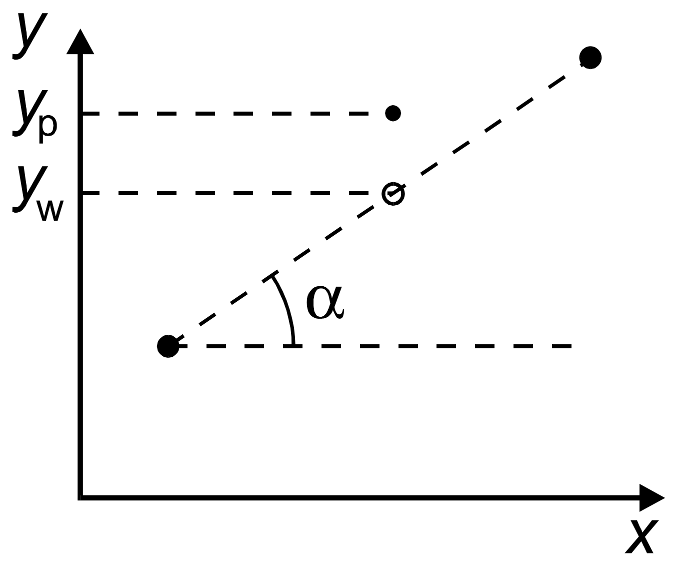
Zdefiniowana powyżej funkcja błędu interpolacji *LD* **jest nieujemna** i spełnia następujące postulaty:

1. Punkty należące jednocześnie do krzywej interpolowanej i interpolującej **nie dają przyrostu** funkcji błędu.
2. Funkcja błędu jest funkcją **symetryczną.**
3. Wartość funkcji błędu jest równa liczbie punktów krzywej interpolowanej i interpolującej nie pokrywających się ze sobą, plus liczba punktów nie należących do żadnej z tych krzywych, znajdujących się wewnątrz obszaru ograniczonego tymi krzywymi.
4. Każdy z punktów (wymienionych w punkcie 3.) wnosi wartość „1” do funkcji błędu.
5. Funkcja błędu przyjmuje wartości całkowitoliczbowe.

Na podstawie punktów 1-5 można stwierdzić, ze funkcja błędu interpolacji **(*LD*) spełnia warunki metryki.**

Zaleta tak zdefiniowanego błędu interpolacji:
Liczenie błędu z użyciem liczb całkowitych, brak dodatkowych błędów wprowadzonych przez reprezentację liczb rzeczywistych.

Zdefiniowany powyżej błąd jest *typu powierzchniowego*.
Można również zdefiniować *błąd maksymalny* (por. def. metryki Czebyszewa):



$$|\alpha| < 45^o \quad \varepsilon = \max|y_p - y_w|$$
$$|\alpha| \geq 45^o \quad \varepsilon = \max|x_p - x_w|$$

Błąd interpolacji dla krzywej składającej się z wielu odcinków:

$$ISLD = \sum_{i=1}^{M} LD_i$$

gdzie: M - liczba odcinków interpolacji
LD_i - błąd interpolacji dla i - tego odcinka krzywej.

Względny błąd interpolacji:

$$WLD = \frac{ISLD}{N}$$

gdzie: $ISLD$ - błąd interpolacji dla całej krzywej
N - liczba punktów krzywej interpolowanej.

- Zadanie metody:

dla danej krzywej dyskretnej dobrać taką technikę interpolacji, czyli **taki sposób rozmieszczenia węzłów interpolacji**, aby przy zadanej wielkości błędu interpolacji ***LD*** uzyskać maksymalny stopień kompresji ***SK*** określony wartością współczynnika redukcji pamięci ***WRP***.

- Techniki interpolacji i ich *złożoność obliczeniowa*

1. Interpolacja równomierna (INTR1)

$$T = a \cdot N\,;\ \left(O(N)\right)$$

2. Interpolacja równomierna z ruchomym węzłem początkowym wzdłuż całej krzywej (INTR3)

$$T = a \cdot N^2;\ \left(O(N^2)\right)$$

3. Interpolacja równomierna z ruchomym węzłem początkowym wzdłuż pierwszego odcinka krzywej (INTR5)

$$T = a \cdot n \cdot N\,;\ \left(O(n \cdot N)\right)$$

4. Interpolacja nierównomierna (INTN3) - metoda dołączania punktów i badania wartości błędu LD

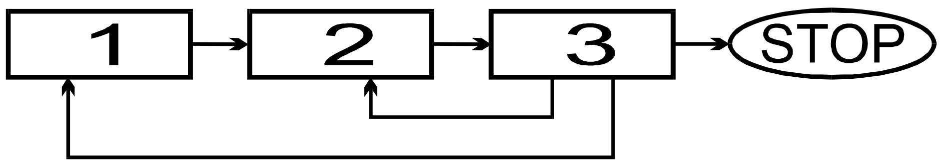
$$T = a \cdot \sum_{i=1}^{M} n_i^2;\ \left(O\left(\sum_{i=1}^{M} n_i^2\right)\right)$$

N - liczba punktów krzywej
M - liczba odcinków interpolacji
n - liczba punktów pierwszego odcinka krzywej interpolowanej
n_i - liczba punktów i - tego odcinka krzywej interpolowanej.
a - współczynnik proporcjonalności.

• Dialog (interakcja) użytkownik - system komputerowy

1. Podział krzywej na segmenty
2. Zadawanie parametrów ***WRP, WLD, T***
3. Obserwacja na ekranie:

– krzywej przed kompresją (pierwotnej) i po kompresji (węzły interpolacji) oraz odtworzonej (odcinki linii prostej dyskretnej),
– liczb określających jakość odtworzenia (błąd całkowity(*ISLD*)), błędy dla poszczególnych odcinków interpolacji (LD_i), typy maksymalne i powierzchniowe) i decyzja o dalszym sposobie postępowania.

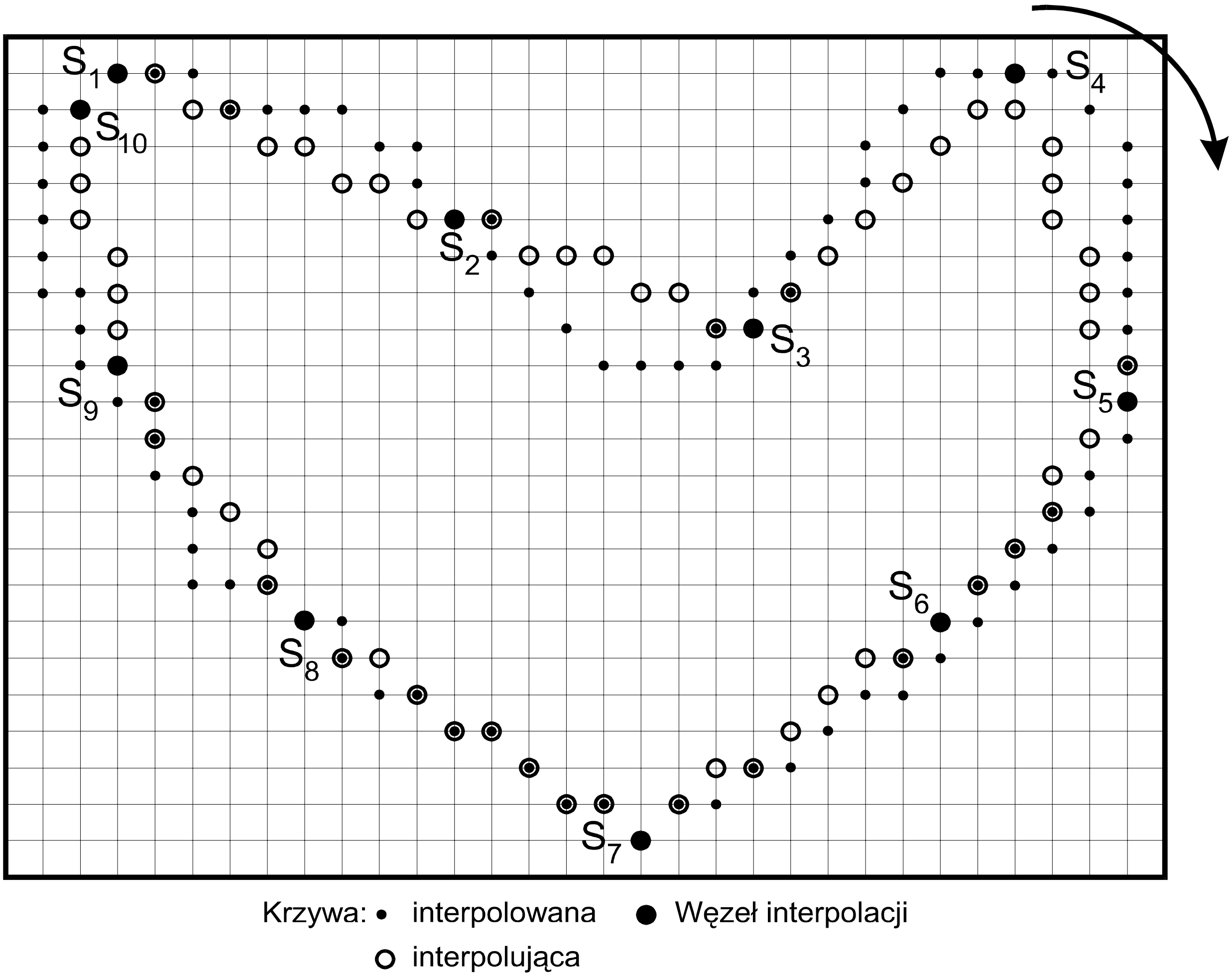


Problem: oszacować i porównać wielkości *SK* dla następujących metod kompresji:
- kodu łańcuchowego o stałej długości,
- różnicowego kodu łańcuchowego,
- interpolacji odcinkami prostych dyskretnych różnymi technikami

Przykład:

- Interpolacja równomierna (algorytm INTR1)

Węzły S1, S2, ... , S10 dzielą krzywą pierwotną na odcinki o **równej** liczbie punktów "*n*" każdy (takie same długości). Stałe położenie węzła początkowego S_1.



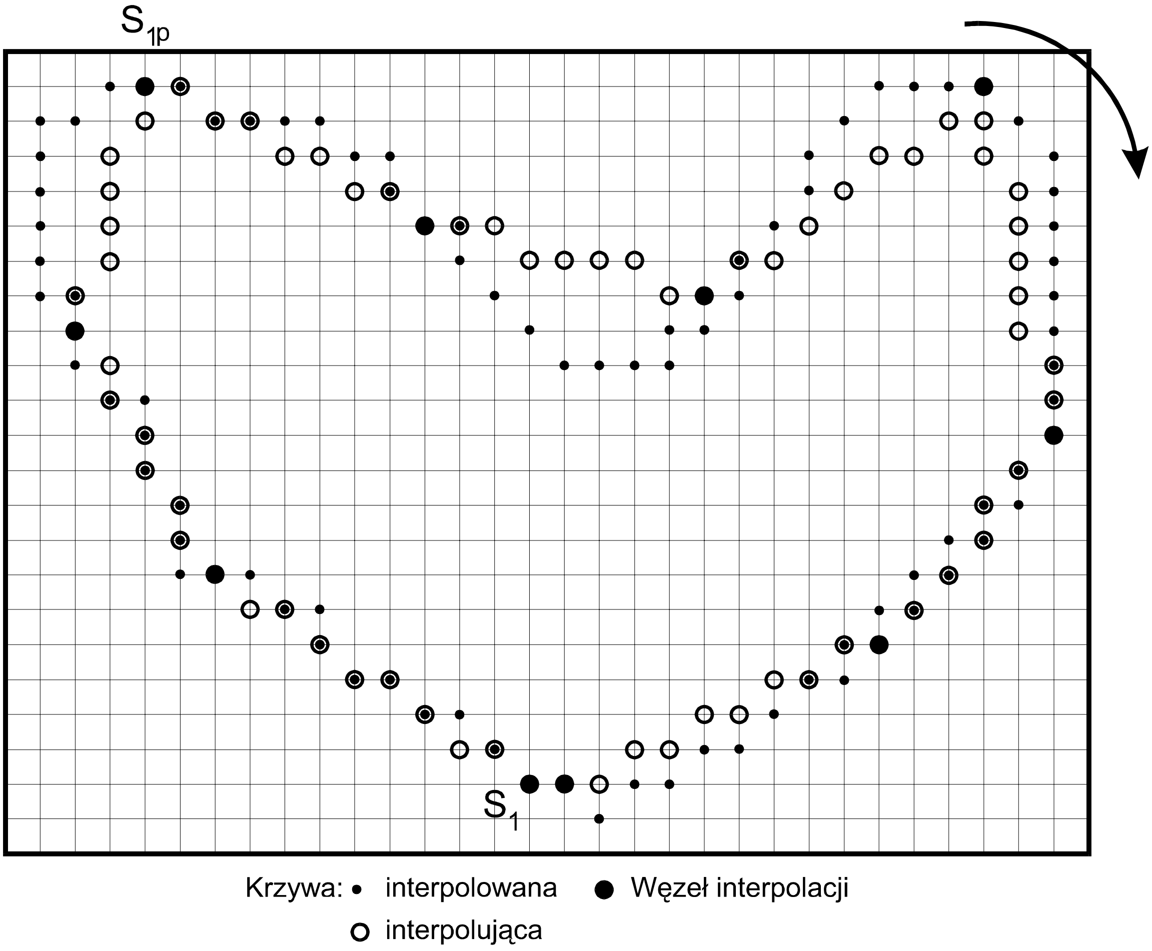
Problem: jak obliczyć *WRP* i *WLD*?

Przykład:

- Interpolacja równomierna (algorytm INTR3)

Działanie algorytmu.
Wyznaczanie węzłów interpolacji dla różnych, zadawanych kolejno we **wszystkich** punktach krzywej pierwotnej, położeń węzła początkowego S_1.
Wybór **położenia węzła S_1** , gdzie ***WLD* jest minimalne.**

Dane:
Położenie początkowe węzła początkowego S_{1P}, długość „n" odcinka krzywej.



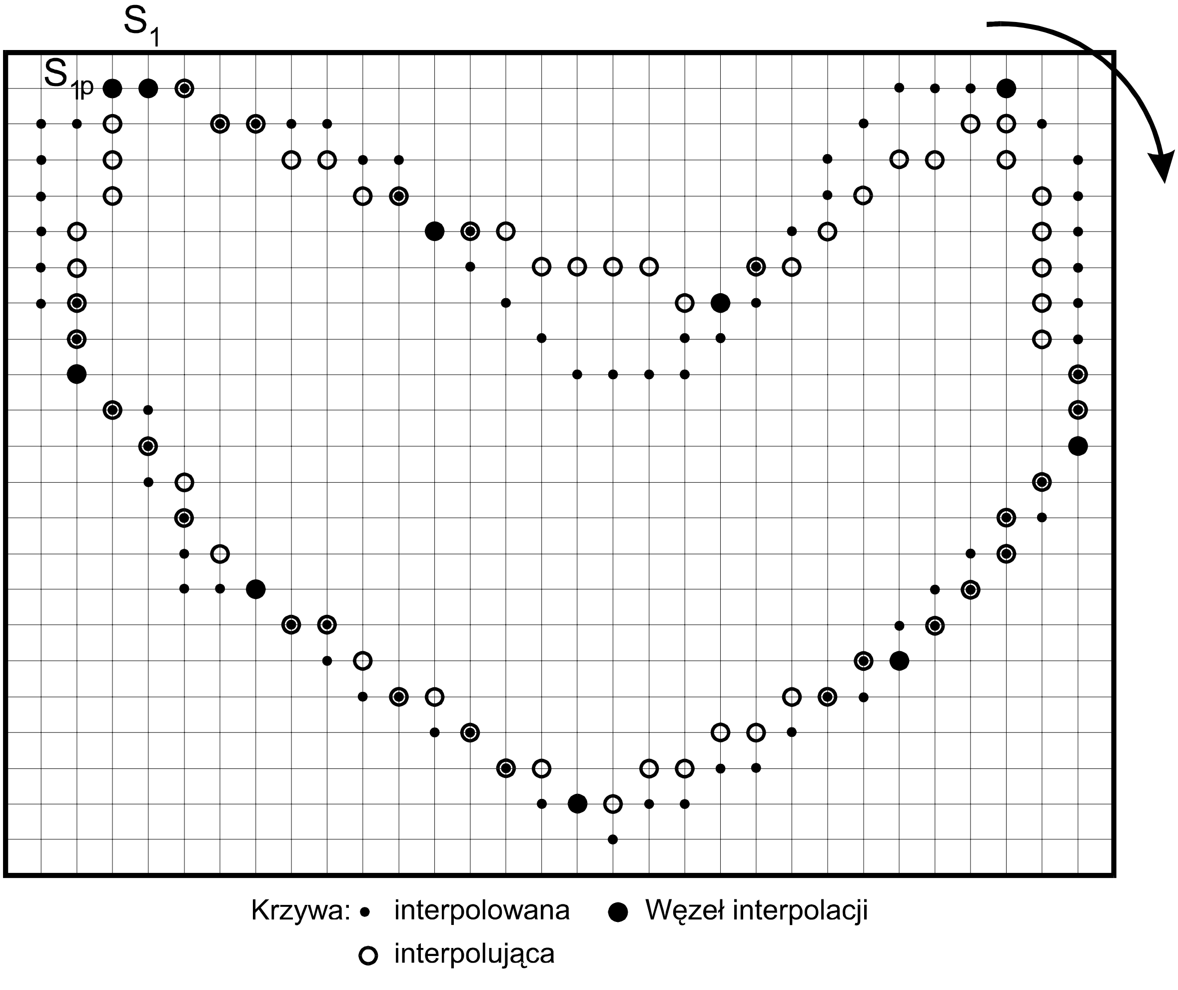
Problem: jak obliczyć *WRP* i *WLD*?
(odp: *WRP*=8,36; *WLD*=1,141)
Uwaga: jest oczywiste, że na ogół przy zadanym "*n*" iloraz WRP/WLD po zastosowaniu algorytmu INTR3 jest większy niż po zastosowaniu algorytmu INTR1.

Przykład:

- Interpolacja równomierna (INTR5)

Algorytm:
Wyznaczanie węzłów interpolacji dla różnych, zadawanych kolejno we **wszystkich "*n*"** punktach **pierwszego odcinka** krzywej pierwotnej, położeń węzła początkowego S_1; wybór położenia, dla którego *WLD* jest minimalne.

Dane:
Położenie początkowe węzła początkowego S_{1p}, długość „*n*" odcinka krzywej.

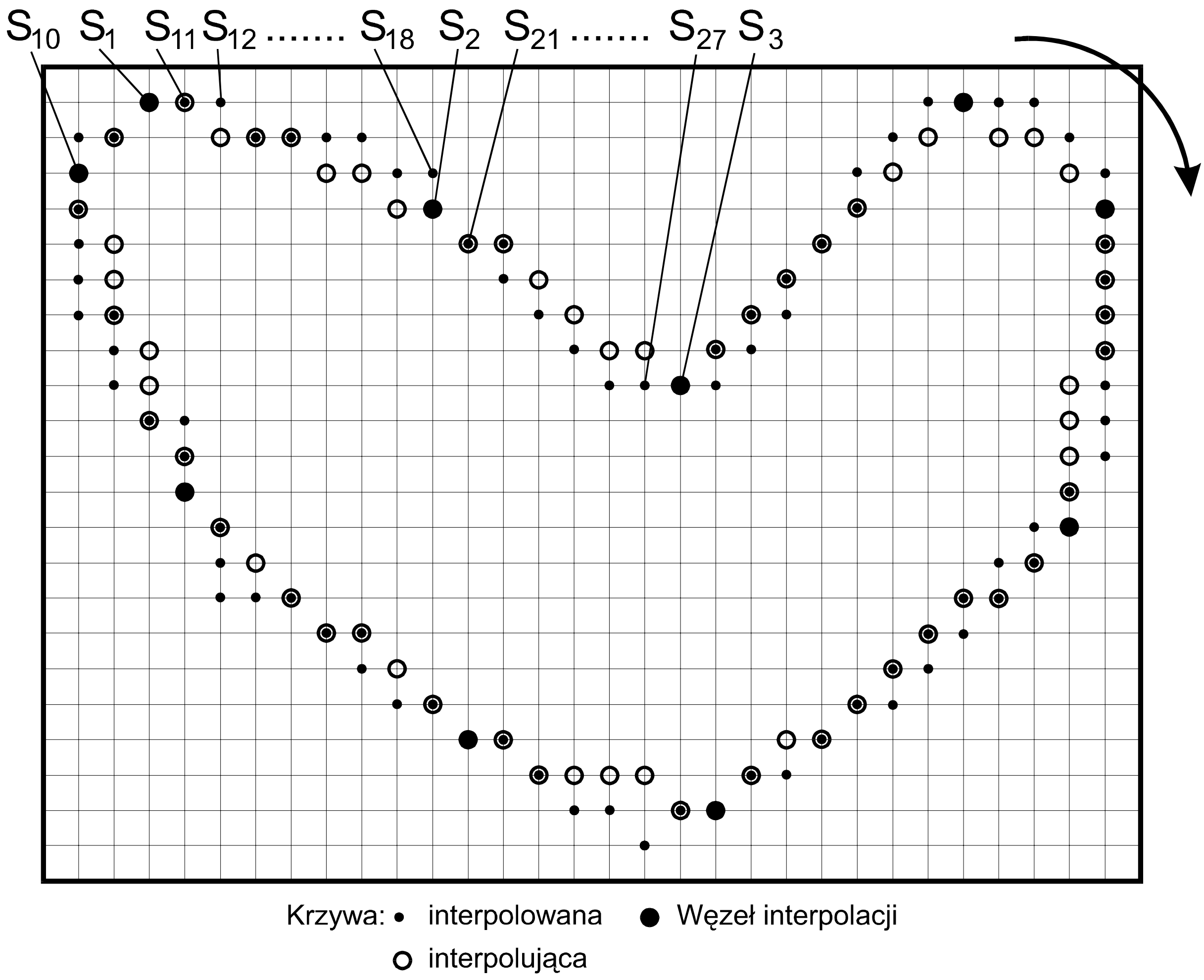


Problem:
Sposób obliczania *WRP* i *WLD*.
(odp: *WRP*=8,36; *WLD*=1,174)
Porównać z wynikami *WRP* i *WLD* dla algorytmów INTR3 i INTR1

Przykład:

- Interpolacja nierównomierna (INTN3)

Algorytm:
Zadanie położenia węzła początkowego S_1, **dołączanie** kolejnych punktów krzywej S_{11}, S_{12}, ... ; łączenie ich z punktem S_1 **odcinkiem linii prostej dyskretnej** i obliczanie błędu interpolacji *LD*. Jeżeli *LD* osiągnie ***wartość dopuszczalną,*** to końce **aktualnego** odcinka stają się **węzłami** interpolacji.
Dane:
Położenie węzła początkowego S_1, dopuszczalny błąd interpolacji *LD* dla jednego odcinka krzywej.



Problem:
Sposób obliczania *WRP* i *WLD*.
Otrzymane wyniki porównać z wynikami *WRP* i *WLD* dla algorytmów INTR5, INTR3 i INTR1

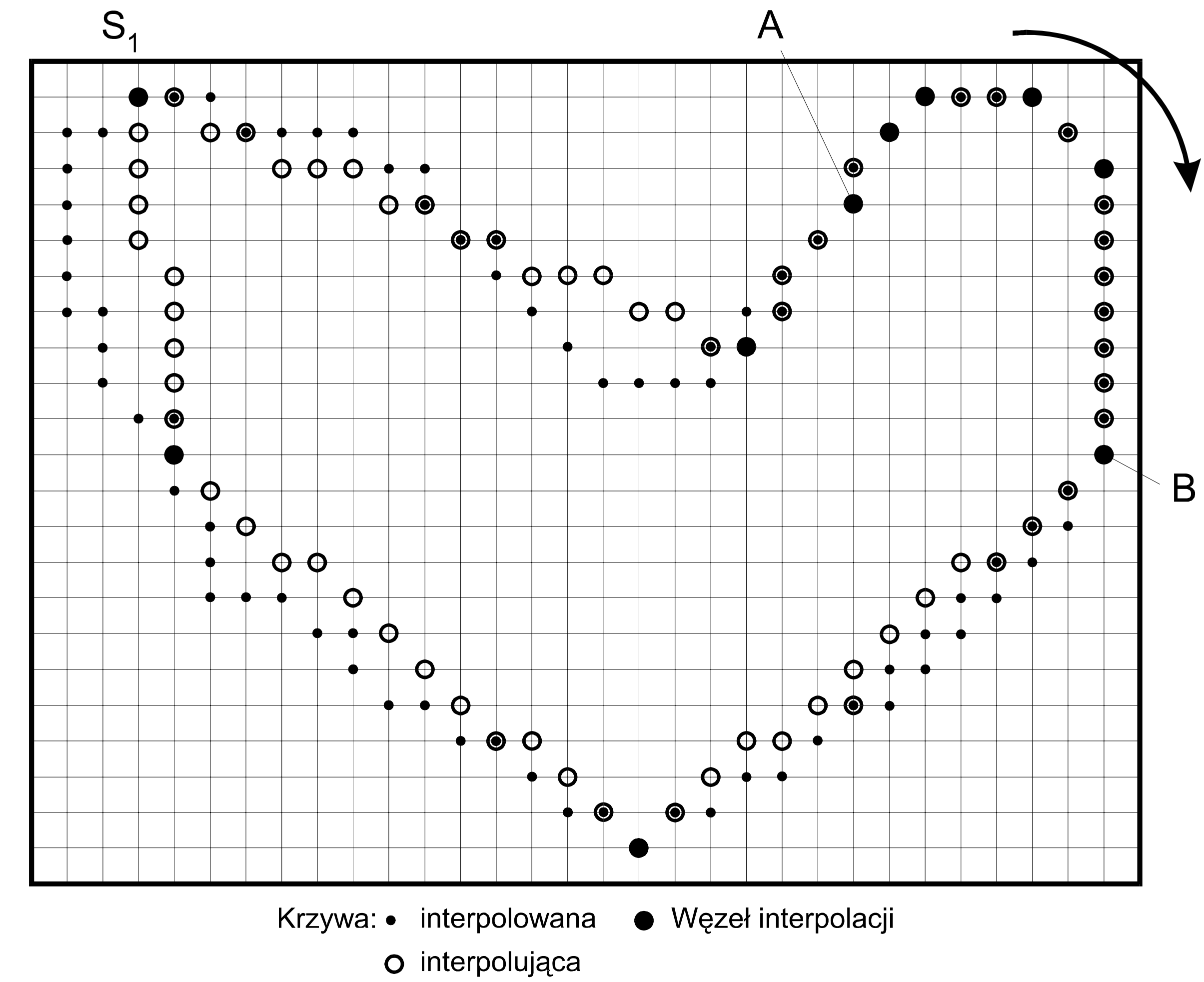
Przykład:

- Interpolacja nierównomierna (INTN3)

- z **dodatkowym podziałem** krzywej pierwotnej (interpolowanej) na segmenty o różnych dopuszczalnych wartościach błędu *LD*.

Dane:

segment AB: *LD*=0; segment BA: *LD*≠0

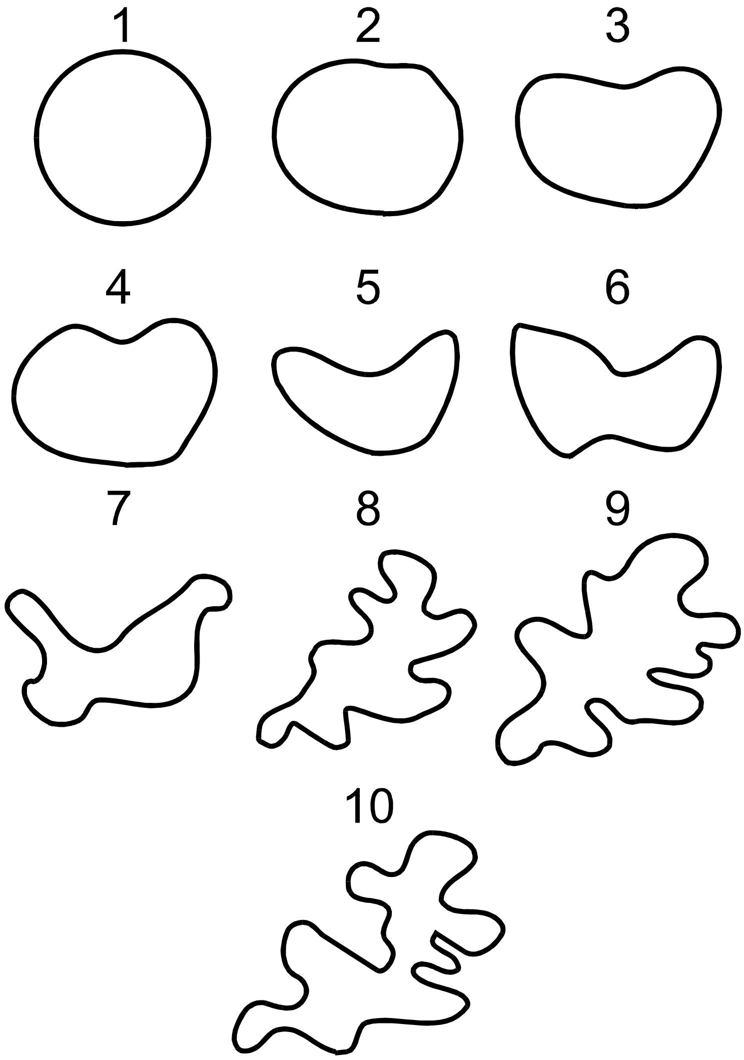


Problem:

Sposób obliczania *WRP* i *WLD*.

Otrzymane wyniki porównać z wynikami *WRP* i *WLD* dla algorytmów INTN3, INTR5, INTR3 i INTR1

Przykład:
Zbadanie i ocena **efektywności** metody interpolacji krzywych dyskretnych odcinkami linii prostych dyskretnych.
Miara efektywności: przebieg zależności *WLD(WRP)*
gdzie: WLD - błąd interpolacji, WRP - współczynnik redukcji pamięci (stopień kompresji). *WRP = SK = KP/KW*
Dane: zbiór podzbiorów krzywych dyskretnych o **różnych kształtach** wyrażających się różnymi stopniami **zmienności krzywizny**

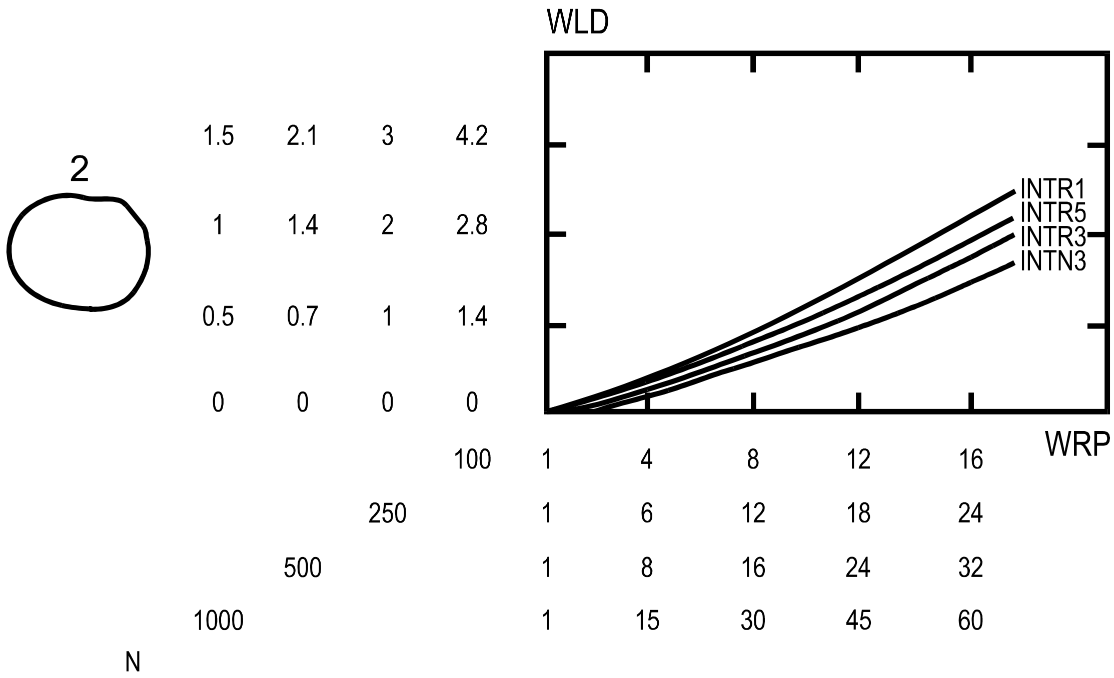


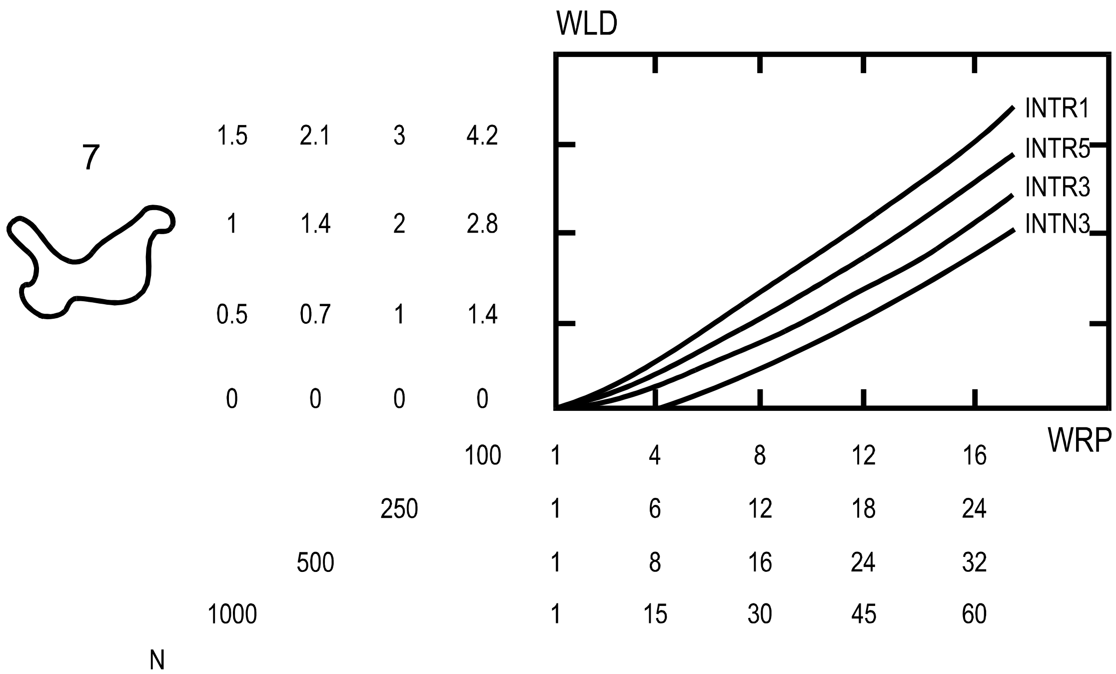
Ocena efektywności na podstawie przebiegów dla **czteroelementowych** podzbiorów krzywych o kształtach 2 i 7.

1. Krzywe o **mało zmiennej krzywiźnie**: **małe różnice** między wartościami błędu dla różnych algorytmów. Pozwala to na wybranie algorytmu o najmniejszej złożoności obliczeniowej (INTR1).
2. Krzywe o **bardziej zmiennej krzywiźnie**: **duża różnica** między wartościami błędu dla różnych algorytmów. Należy wybrać algorytm INTN3.

Uwaga: Z obu wykresów wynika, że przy **dostatecznie małej** wartości WRP omawiana metoda interpolacji realizuje kompresję typu *lossless*
Przykład:

Przebiegi zależności *WLD(WRP)* dla algorytmów INTR1, INTR3, INTR5, INTN3 dla 2 wybranych kształtów krzywych zamkniętych o różnych stopniach zmienności krzywizny. Każdy z kształtów reprezentowany jest przez **4 krzywe** (**czteroelementowy** podzbiór krzywych)o różnych długościach





Przykład

Dana jest siatka prostokątna o rozmiarach 16x16 węzłów.

a) Podać przykład takiej dyskretnej krzywej otwartej (piksle umiejscowione w węzłach siatki) o długości IMEM=20, dla której po zastosowaniu algorytmu INTN3 z parametrem LD=0 uzyskuje się wartość WRP=SK > 1.

Jaki rodzaj kompresji (stratna, bezstratna) przeprowadzono?

b) Porównać uzyskane wartości WRP=SK po zastosowaniu tego samego algorytmu dla okręgu dyskretnego o średnicy d=11 piksli (węzłów siatki) tzn. o promieniu r=5 dla 2 przypadków: 1) LD=0, 2) LD=2.

Dla przypadku okręgu dyskretnego zaproponować metodę kompresji bezstratnej dającej w efekcie dalsze zwiększenie stopnia kompresji SK.

Uwaga: okrąg dyskretny należy utworzyć stosując dyskretyzację Freemana okręgu wykluczając istnienie punktów niejednoznaczności.