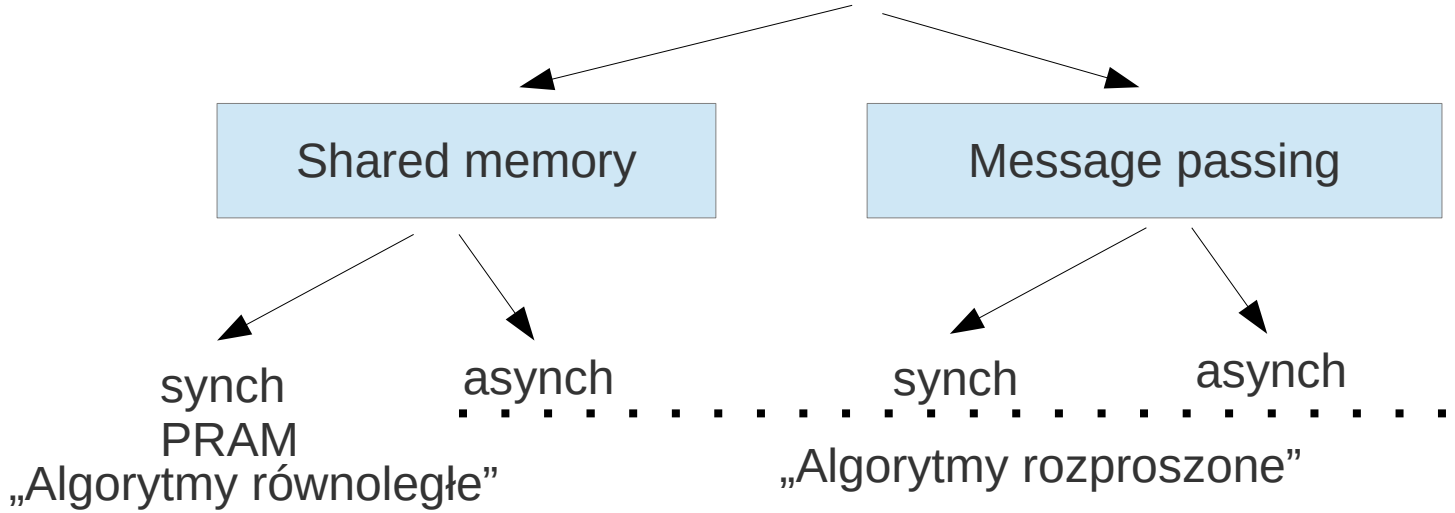
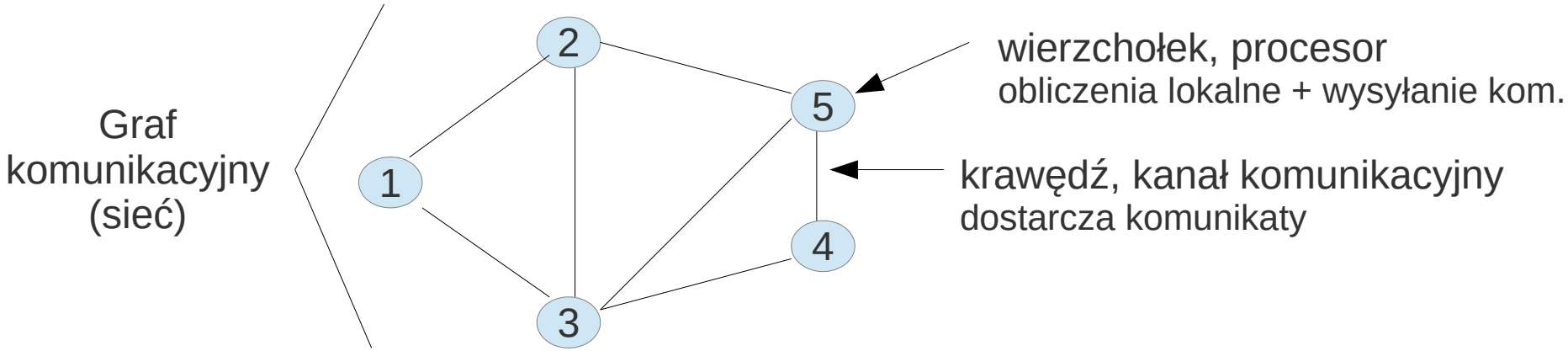
Modelujemy systemy n-procesorowe...



Model „Message passing” - nieformalna definicja :



Co jest celem alg rozproszonego? Odp: obliczenie „czegoś” w grafie kom

Model message-passing – synch i asynch

W modelu synch mamy **rundy**...
w każdej rundzie każdy wierzchołek:
1. wykonuje obliczenia lokalne („za darmo”)
2. wymienia komunikaty z sąsiadami w grafie kom.
(dokładniej: w i+1 rudzie otrzymuje kom wysłane w i-tej rundzie przez sąsiadów)

W modelu asynch nie ma rund...
wysłane przez kraw komunikaty po jakimś czasie są dostarczane...

Warianty modelu message-passing:
1. graf kom może być: cyklem (ring), drzewem, pełnym grafem, …
2. stopien synchronizacji: synch i asynch
3. stopień symetrii, czy wierz v ma ID(v) czy też są nieodróżnialne?
4. jednorodność (uniform), niejednorodny= wierz muszą znać „n” (liczbę wierz)

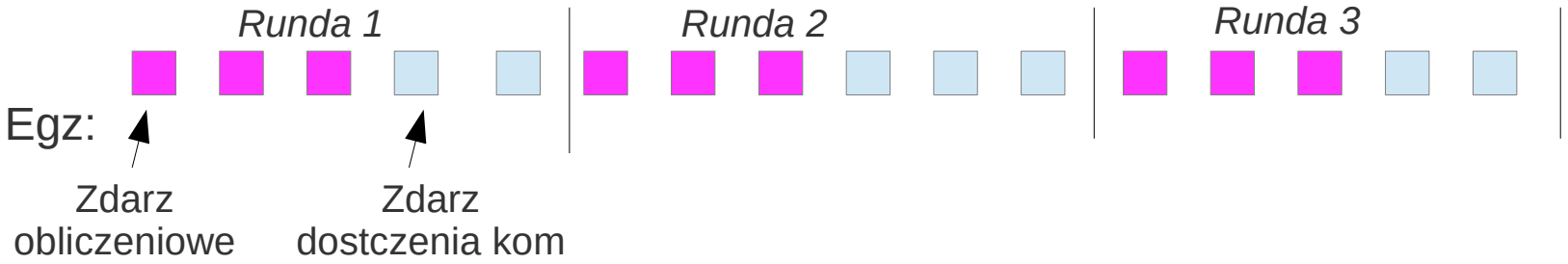
Formalna def modelu message-passing, asynch:
mamy n-procesorów, P1, …, Pn
konfiguracja, C= (q1, …, qn), stan procesorów (zmienne), także zawartość połączeń
zdarzenie, 1. obliczeniowe (obl lokalne + wysyłanie kom), 2. dostarczenia kom
egzekucja, C0,Phi0, C1, Phi1, … gdzie Ci to konfiguracja, Phi_i to zdarzenie
konfiguracja C_i+1 wynika z zastosowania zdarzenia Phi_i do konfig C_i
w przypadku alg deterministycznych wystarczy podać: C0, Phi0, Phi1, Phi2, …

Model message-passing – synch i asynch

*Co to znaczy, że dany algorytm **asynch** „A” działa ?*
Odp: działa (obl to co trzeba) dla dowolnej dopuszczalnej egzekucji
Egzekucja jest dopuszczalna gdy:
1. każdy wierz ma nieskoczenie wiele zdarzeń obliczeniowych,
2. każdy wysłany komunikat jest dostarczany (kiedy?)

Formalna definicja modelu message-passing, synch:
rozpatrujemy egzekucje dopuszczalne, które dają się podzielić na rundy,
tj takie, że w każdej rundzie:
1. każdy wierz ma 1 zdarzenie obl,
2. wszystkie wysłane kom są dostarczane (w tej rundzie)

wysłane w i-tej rundzie są dostępne w (i+1)-rundzie



*Co to znaczy, że dany algorytm **synch** „A” działa ?*
Odp: działa dla każdej egz, która można podzielić na rundy...

Uwaga: alg synch jest asnych !!!

Złożoność obliczeniowa w modelu mess-pass

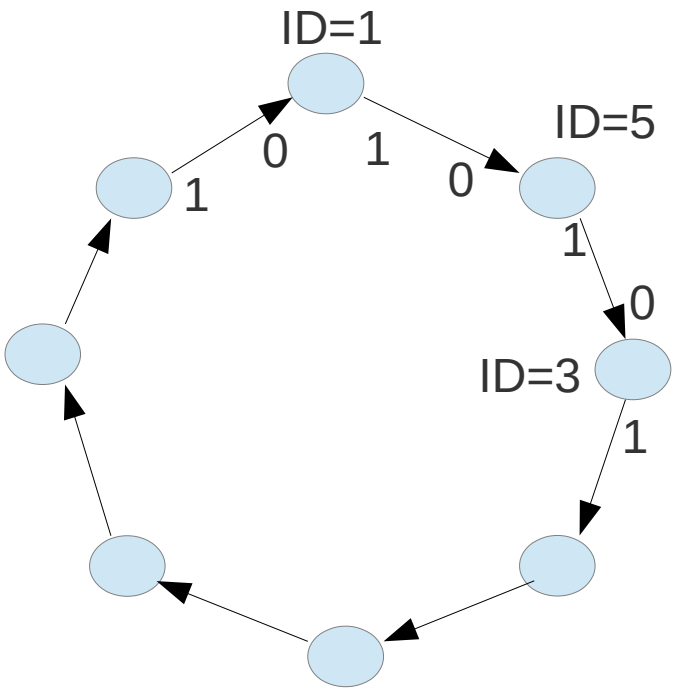
Synch:
- czas (liczba rund)
- liczba kom

Asynch:
- liczba kom
- czas (???)

Problem: leader election, **LE**, ring (zorient), asynch, są ID,
podamy alg, w którym licz kom= O(n^2)

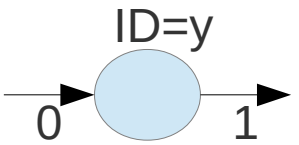
alg:

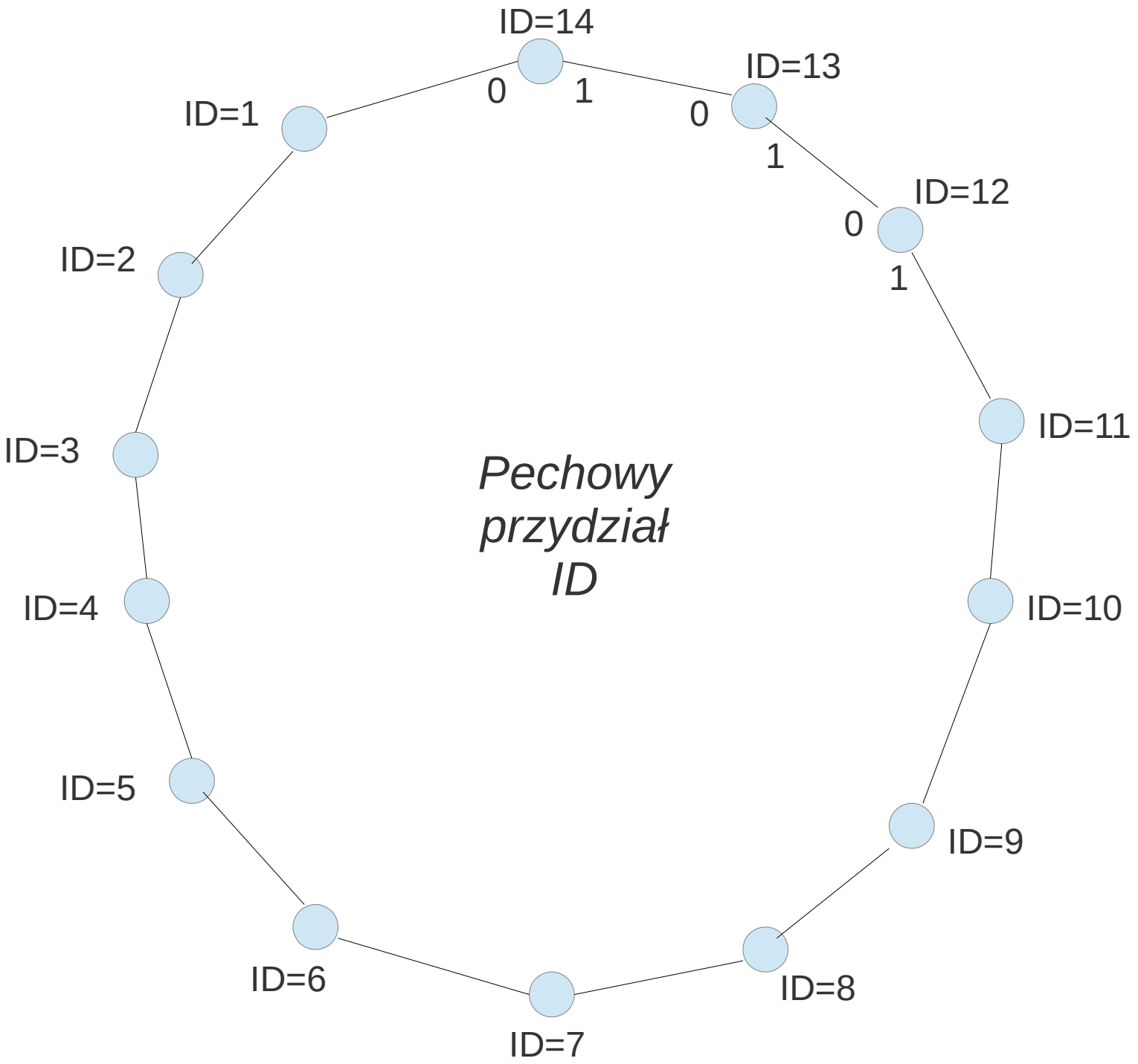
1. każdy wierz wysyła swój ID przez poł 1
2. gdy z poł 0 przyjdzie X, to:
   jeśli X>ID to wysyłamy X przez poł 1
   jeśli X==ID to ogłaszamy się liderem
   i zawiadamiamy pozostałw wierz, że nie są liderami
   jeśli X<ID to … nic nie robimy



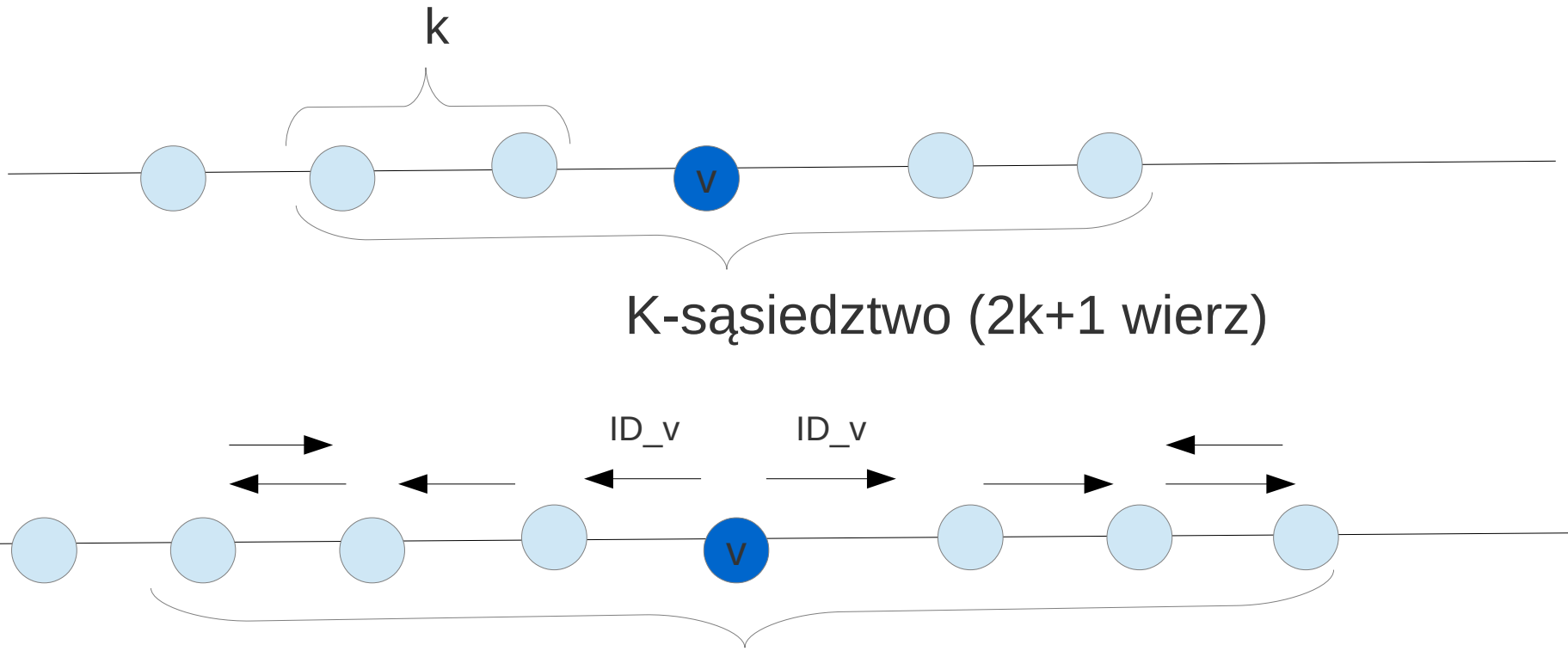
Pytania:

1. dlaczego alg wybiera lidera?
2. Ile kom się wysyła?




ID=14
ID=13
ID=12
ID=11
ID=10
ID=9
ID=8
ID=7
ID=6
ID=5
ID=4
ID=3
ID=2
ID=1
0
1
0
1
0
1
Pechowy przydział ID

Problem: leader election, **LE**, ring (zorient), asynch, są ID,
podamy alg, w którym licz kom= O(n logn)



Mamy **fazy**: 0, 1, 2, …

Wierz uczestniczące w L-tej fazie wykonują *powyższe* w 2^L- sąsiedztwie

wierz v wysyła ID_v w obie strony **+ zasady jak z LE O(n^2)**

Jeśli wierz otrzyma odpowiedź z obu stron to ogłasza się:

„tymczasowym liderem L-tej fazy”

W fazie L+1 uczestniczą tymczasowi liderzy L-tej fazy

LE, O(n logn) kom, c.d.

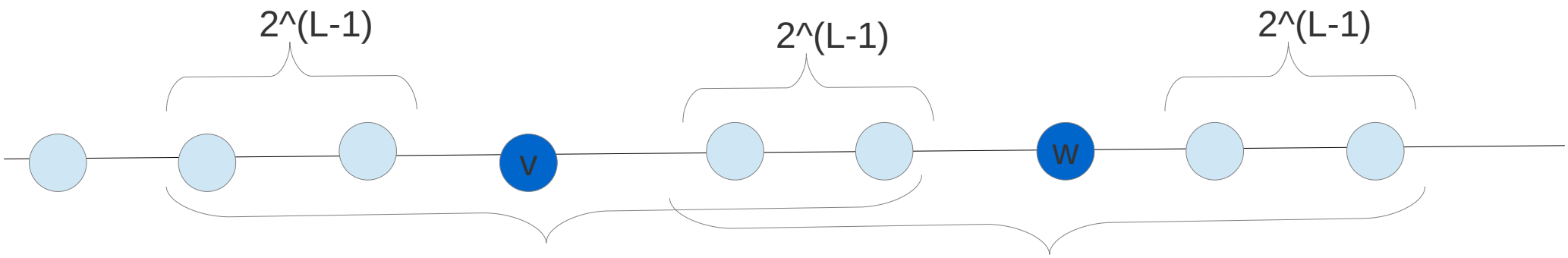
Lemat 2.3.1:

Dla L>=1, liczba wierz uczestn w L-tej fazie < n/2^(L-1)

Dowód:

jeśli wierz uczestniczy w L-tej fazie to znaczy że był tymczas liderem (L-1)- fazy,

tj ID_v jest max w 2^(L-1) -sąsiedztwie „v'



Liderzy nie mogą być blisko siebie !!!

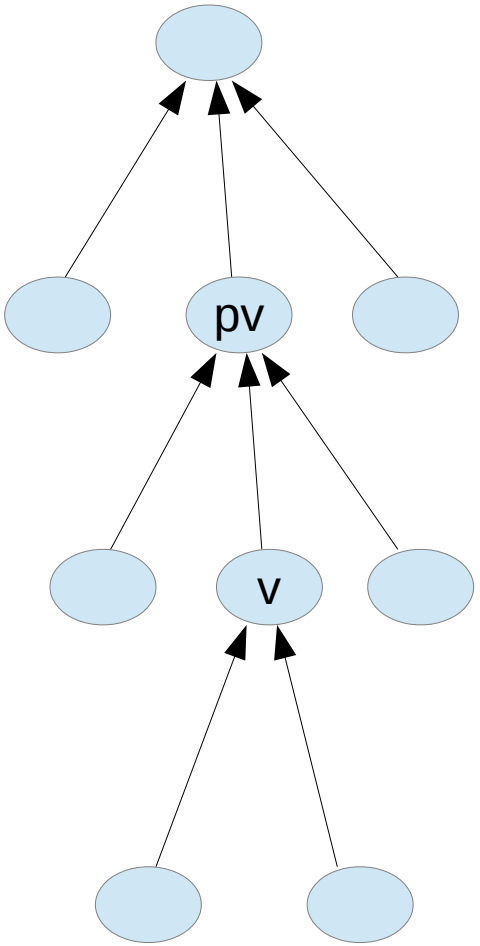
Skoro każdy wierz uczestn w fazie L wysyła 4*2^L kom,

to w L-tej fazie łącznie wysyła się 4*2^L * n/2^(L-1) = 8n kom.

Liczba faz wynosi logn (dlaczego ??)

stąd złożoność komunikatowa algortymu to O(n logn)

8-kolorowanie wierz drzewa ukorzenionego Cole & Vishkin, synch, czas= O(log^*n)



log^*n = „ile razy trzeba zlogarytmować n aby zejść poniżej 1”
chodzi o **prawidłowe** kolorowanie wierz,
czyli wierz połaczone kraw mają RÓŹNE kolory...

Algorytm:

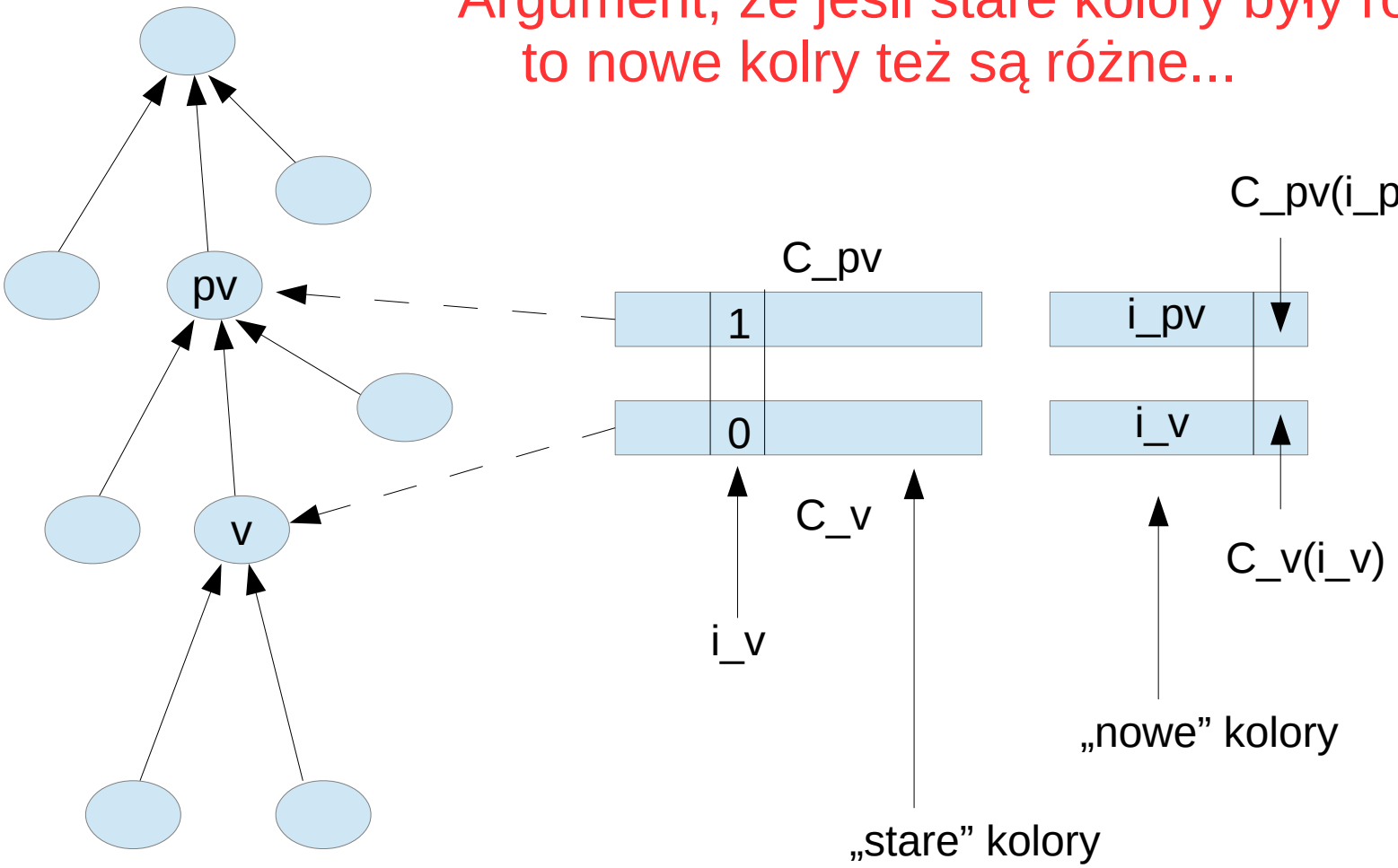
1. C_v : = ID_v
2. powtórz O(log^*n) razy:
   i_v:= pozycja bitu, na którym różnia się C_v i C_pv
   # gdzie C_pv to kolor parenta wierz v
   C_v:= (bity(i_v), C_v(i_v))
   # bity(x) = reprezentacja dwójkowa liczby x
   # C_v(k) = k-ty bit C_v

L_k = liczba bitów C_v w k-tej iteracji
L_0 = O(logn)
L_{k+1} = sufit(log_2 L_k) + 1
Jest to ciąg malejący zbieżny do 3.
Jeśli kolor jest zpisany na 3 bitach, to liczba kolorów=8

Dlaczego po zakończeniu algrytmu mamy
prawidłowe kolorowanie wierz ? (8-ma kolorami)

8-kolorowanie wierz drzewa ukorzenionego Cole & Vishkin, synch, czas= O(log^*n)

Argument, że jeśli stare kolory były różne,
to nowe kolry też są różne...



8-kolorowanie wierz drzewa ukorzenionego Cole & Vishkin, synch, czas= O(log^*n)

Można w prosty sposób
zmniejszyć liczbę kolorów
z 8 do 3 ...

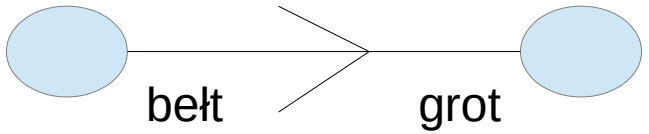
W jaki sposób uogólnic
algorytm C&V
na inne grafy ???
(np. dla cyklu niezorient ??)

(Delta+1)-kolorowanie wierz, w grafie dowolnym, synch, czas= O(Delta^2 + log*n)

Def „forest decomposition”:
Mamy graf G=(V,E)
1. orientujemy kraw wg ID
2. każdy wierz numeruje sąsiadów
3. każdej kraw przypisujemy nr
   na „bełcie strzały”

Otrzymujemy podział:
E=E_1 u … u E_Delta
E_i = zb kraw którym przyp „i”

Fakt: (V,E_i) jest lasem ukorz drzew
Dowód:
1. z wierz nie wychodzi >1 strzał
2. nie ma skierowanych cykli



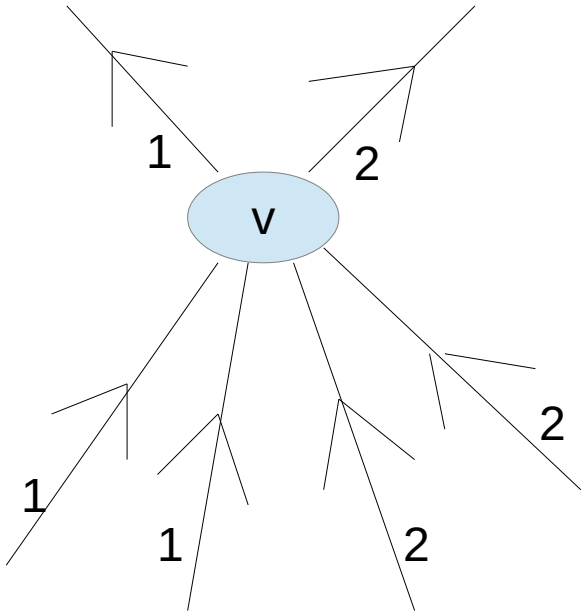
Algorytm:
1. oblicz forest decomp
2. w drzewach ukorze wsz E_i
 oblicz 8-kol-wierz metodą C&V
3. A:= puste
 For i=1 to Delta do
  (1) w G[A u E_i] mamy
  8*(Delta+1)-kol-wierz
  pochodzące z połączenia
   (Delta+1)-kol-wierz w A
   z 8-kol-wierz w E_i
  (2) redukujemy je do
  (Delta+1)-kol-wierz
  (3) A:=A u E_i

Po zakonczeniu pkt 3,
 mamy w calym grafie
  (Delta+1)-kol-wierz...

(Delta+1)-kolorowanie wierz, w grafie dowolnym, synch, czas= O(Delta^2 + log*n)

Co jeszcze trzeba wyjaśnić w tym algorytmie ???

Łączenie kolorowania
w 2 podgrafach rozł kraw



W E_1 mamy kolor C^1_v
W E_2 mamy kolor C^2_v
w całym grafie mamy kolor
(C^1_v, C^2_v)

Redukcja liczby kolorów
do Delta+1

W palecie Delta+1 kolorów
zawsze jest 1 kolor wolny...

Można iterować po „starych kolorach”;
Dla wsz wierz bieżącego koloru,
  obliczać nowe kolor,
    z palety Delta+1 kol...