Spis rzeczy

5 Uliniowienie wielu sekwencji **48**
5.1 Uliniowienie sekwencji z profilem 48
5.2 Miara typu 'suma par' (SP) 50
5.3 Uliniowienie gwiazdowe dla SP 54

5 Uliniowienie wielu sekwencji

Znajdowanie optymalnego uliniowienia dla wielu sekwencji jest bardzo ważne ze względu na liczne zastosowania w biolologii:

- Znajdowanie i reprezentowanie konserwowanych regionów w DNA i białkach.
- Reprezentowanie rodzin białek.
- Odtwarzanie ewolucyjnej historii.

5.1 Uliniowienie sekwencji z profilem

Przypuśćmy, że mamy rodzinę sekwencji uliniowioną w następujący sposób:

```
a b c - a
a b a b a
a c c b -
c b - b c
```

Powyższe ulinowienie generuje profil w następujący sposób. Dla każdej kolumny i każdego symbolu (również spacji) liczymy częstość wystąpienia tego symbolu w kolumnie. W ten sposób powstaje tablica $P = (p_{x,i})$, gdzie x przebiega po wszystkich symbolach, a i przebiega po numerach kolumn. Ta tablica nazywa się *profilem*. W naszym przykładzie mamy następujący profil:

	1	2	3	4	5
a	0.75	0	0.25	0	0.50
b	0	0.75	0	0.75	0
c	0.25	0.25	0.50	0	0.25
-	0	0	0.25	0.25	0.25

Tak więc profil jest skończonym ciągiem rozkładów prawdopodobieństwa na zbiorze symboli $\Sigma \cup \{-\}$. Mając dane słowo $S \in \Sigma^*$ oraz profil P, *uliniowienie* S z P jest to każde ustawienie symboli z S z kolumnami z P (z możliwością wstawiania spacji zarówno pomiędzy symbole z S, jak i pomiędzy kolumny z z P), przy którym spacja nie stoi na wprost spacji. Przykładem uliniowienia dla powyższego profilu P ze słowem *aabbc* jest

```
a a b - b c
1 - 2 3 4 5
```

Zatem w ogólności uliniowienie słowa S z profilem P jest para słów $S^\# \in (\Sigma \cup \{-\})^*$, $P^\# \in (\mathbb{N} \cup \{-\})^*$, taka że

1. $|S^\#| = |P^\#|$.
2. $S^\#$, po usunięciu spacji daje S.
3. $P^\#$, po usunięciu spacji, daje ciąg $1 \ldots n$.
4. Dla każdego i, $S^\#(i)$ oraz $P^\#(i)$ nie są jednocześnie spacją.

Niech $s : (\Sigma \cup \{-\})^2 \to \mathbb{R}$ będzie funkcją podobieństwa. Wartością uliniowienia $(S^\#, P^\#)$ jest liczba

$$\sum_{i=1}^{|S^\#|} s(S^\#(i), P^\#(i)),$$

gdzie dla $x \in \Sigma \cup \{-\}$ oraz dla rozkładu p_i występującego w profilu P,

$$s(x, i) = \sum_{y \in \Sigma \cup \{-\}} s(x, y) \cdot p_i(y).$$

W naszym przykładzie, jeśli funkcją podobieństwa jest

$$s(x, y) = \begin{cases} 1 & \text{jeśli } x = y, \\ -1 & \text{jeśli } x \neq y, \end{cases}$$

to wartość ulinowienia przedstawionego powyżej wynosi $0.5 - 1 + 0.5 - 0.5 + 0.5 - 0.5 = -.05$.

Problem znajdowania optymalnego uliniowienia słowa z profilem rozwiązuje się, używając metody dynamicznego programowania, w taki sam sposób jak dla przypadku dwóch słów. Jedyna różnica polega na tym, że obecnie mamy do czynienia z dwoma alfabetami, nad którymi są brane słowa: Σ oraz

zbiór rozkładów prowdopodobieństw nad $\Sigma \cup \{-\}$. Ponieważ koszt obliczania wartości podobiebieństwa $s(x,i)$ wynosi $O(|\Sigma|)$ (przyjmujemy koszt liczenia $s(x,y)$ oraz $p_i(y)$ jako jednostkowe), to całkowity czas, w którym znajdujemy optymalne uliniowienie słowa S z profilem P wynosi $O(|\Sigma| \cdot m \cdot n)$, gdzie $m = |S|$, $n = |P|$ (tzn. P jest ciągiem n rozkładów).

5.2 Miara typu 'suma par' (SP)

Uliniowienie wielu sekwencji definiuje się podobnie do przypadku dwóch sekwencji (tzn. wyrównuje się długości sekwencji przez wstawienie spacji, przy dodatkowym warunku, że w żadnej kolumnie nie mogą się znajdować same spacje).

Będziemy się zajmować wersją optymalizacji problemu uliniowienia polegającą na *minimalizacji* wartości uliniowienia, zamiast maksymalizacji, tak jak to było definiowane dla podobieństwa. Ustalmy alfabet Σ i niech $s : (\Sigma \cup \{-\})^2 \to \mathbb{R}$ będzie dowolną funkcją podobieństwa. Weźmy dowolne k słów $S_1, \ldots, S_k \in \Sigma^*$ i niech $S_1^\#, \ldots, S_k^\#$ będzie ich uliniowieniem. Przypomnijmy,że wartość uliniowienia dwóch słów $S_i^\#$ oraz $S_j^\#$ jest równa

$$s(S_i^\#, S_j^\#) = \sum_{l=1}^{m} s(S_i^\#(l), S_j^\#(l)),$$

gdzie $m = |S_i^\#| = |S_j^\#|$. Zauważmy, że obecnie, dla pewnego $1 \leq l \leq m$, możliwe jest $S_i^\#(l) = S_j^\#(l) = -$. Przyjmijmy, że wartość podobieństwa dla spacji wynosi 0, tzn. $s(-,-) = 0$.

Miarą podobieństwa jaką się teraz będziemy zajmować jest tzw. miara *sumy par*. Pojęcie to zostało zaproponowane przez Carillo i Lipmana w 1988r. Dla danego uliniowienia $S_1^\#, \ldots, S_k^\#$ bierzemy sumę wartości podobieństwa po wszystkich różnych parach:

$$s(S_1^\#, \ldots, S_k^\#) = \sum_{i<j} s(S_i^\#, S_j^\#).$$

Niech $\overline{sim}(S_1, \ldots, S_k)$ oznacza minimum sum $\sum_{i<j} s(S_i^\#, S_j^\#)$, brane po wszystkich uliniowieniach $(S_1^\#, \ldots, S_k^\#)$ ciągu słów $S_1, \ldots, S_k$. Wartość $\overline{sim}(S_1^\#, \ldots, S_k^\#)$ będziemy nazywać *SP-wartością* dla ciągu $S_1, \ldots, S_k$.

Zauważmy, że naturalnym uogólnieniem funkcji *sim* z działu poświęconemu podobieństwu dwóch sekwencji jest $sim(S_1, \ldots, S_k)$ oznaczające maksimum sum $\sum_{i<j} s(S_i^\#, S_j^\#)$, brane po wszystkich uliniowieniach. Mamy prosty związek pomiędzy tymi dwoma funkcjami

$$\overline{sim}_s(S_1, \ldots, S_k) = sim_{-s}(S_1, \ldots, S_k),$$

gdzie $\overline{sim}_s$ jest określona dla funkcji podobieństwa s, a sim_{-s} jest określona dla funkcji podobieństwa $-s$ (tzn. wartość s z przeciwnym znakiem).

Pokażemy, że problem obliczania SP-wartości dla dowolnych ciągów słów $S_1, \ldots, S_k$ jest trudny obliczeniowo. Niech $\Sigma = \{0, 1, a, b\}$. Na alfabecie Σ określamy funkcję podobieństwa następującą tabelką

	0	1	a	b	–
0	2	2	1	2	1
1	2	2	2	1	1
a	1	2	0	2	1
b	2	1	2	0	1
–	1	1	1	1	0

Niech SP będzie następującym problemem decyzyjnym: dany ciąg słów $S_1, \ldots, S_k \in \Sigma^*$ oraz liczba $c \in \mathbb{N}$. Czy zachodzi $\overline{sim}(S_1, \ldots, S_k) \leq c$, gdzie $\overline{sim}$ jest określone dla w/w funkcji podobieństwa?

Dowód poniższego twierdzenia jest zmieniony w stosunku do wersji oryginalnej, która zawiera błąd (redukcja przedstawiona w dowodzie jest niepoprawna).

Twierdzenie 5.2.1 (Wang, Jiang (1994)) *Problem SP jest NP-zupełny.*

Dowód: Oczywiście SP należy do klasy NP: wystarczy zgadnąć uliniowienie dla danego ciągu słów $S_1, \ldots, S_k$ (długość każdego słowa w takim uliniowieniu jest nie większa od $\sum_{i=1}^{k} |S_i|$).

Dla dowodu NP-trudności problemu SP pokażemy redukcję problemu *najkrótszego wspólnego superciągu (NWS)* do problemu SP. Problem NWS definiuje się następująco: dany ciąg słów $S_1, \ldots, S_k \in \{0,1\}^*$ oraz $m \in \mathbb{N}$. Czy istnieje słowo $S \in \{0,1\}^*$ takie, że $|S| \leq m$ oraz każde S_i jest *podciągiem* słowa S, tzn. istnieje rosnący ciąg $1 \leq j_1 < j_2 < \ldots < j_{n_i} \leq |S|$ (gdzie $|S_i| = n_i$) takie, że dla $p = 1, \ldots, n_i$ mamy $S(j_p) = S_i(p)$. W tej sytuacji mówimy też, że S jest *superciągiem* S_i.

Niech $S_1, \ldots, S_k \in \{0,1\}^*$oraz $m \in \mathbb{N}$ będą danymi wejściowymi dla problemu NWS. Bez zmniejszenia ogólności możemy przyjąć, że $|S_i| \leq m$, dla $i = 1, \ldots, k$. Niech $M = \sum_{i=1}^{k} |S_i|$.

Pokażemy, że $S_1, \ldots, S_k$ mają wspólny superciąg o długości co najwyżej m $\Longleftrightarrow$ istnieje $0 \leq p \leq m$ takie, że $\overline{sim}(S_1, \ldots, S_k, a^p, b^{m-p}) \leq k \cdot M + (k+1) \cdot m$.

Zanim przejdziemy do dowodu powyższej równoważności zauważmy, że dla dowolnego uliniowienia $S_1^{\#}, \ldots, S_k^{\#}, A, B$ pochodzącego od $S_1, \ldots, S_k, a^p, b^{m-p}$ (gdzie $p \leq m$ jest dowolne) mamy

$$\sum_{i<j} s(S_i^{\#}, S_j^{\#}) = (k-1) \cdot M, \tag{5.5.1}$$

oraz

$$s(A, B) = m. \tag{5.5.2}$$

Rowność (5.5.1) dowodzimy przez rutynową indukcję ze względu na k. W dowodzie wykorzystujemy użyteczną obserwację:

$$s(S_i^{\#}, S_j^{\#}) = |S_i| + |S_j|,$$

dla $1 \leq i, j \leq k$.

Równość (5.5.2) wynika natychmiast z definicji funkcji s.

Zajmiemy się teraz dowodem równoważności. Najpierw ($\Rightarrow$): niech $S \in \{0,1\}^*$ będzie superciągiem dla $S_1, \ldots, S_k$ takim, że $|S| \leq m$. Możemy przyjąć, że $|S| = m$. Niech p będzie liczbą zer w S. Zbudujemy uliniowienie $S_1^{\#}, \ldots, S_k^{\#}, A, B$ dla ciągu słów $S_1, \ldots, S_k, a^p, b^{m-p}$. Każde słowo uliniowienia będzie miało długość m. Niech $1 \leq j_1 < \ldots < j_{n_i} \leq m$ będzie ciągiem indeksów wyznaczających pozycje liter z S_i w słowie S. Wówczas dla $1 \leq l \leq m$,

$$S_i^{\#}(l) = \begin{cases} S_i(\xi) & \text{jeśli } l = j_\xi, \\ - & \text{w pozostałych przypadkach.} \end{cases}$$

Słowa A i B definiujemy następująco

$$A(l) = \begin{cases} a & \text{jeśli } S(l) = 0, \\ - & \text{jeśli } S(l) = 1. \end{cases}$$

$$B(l) = \begin{cases} b & \text{jeśli } S(l) = 1, \\ - & \text{jeśli } S(l) = 0. \end{cases}$$

Zauważmy, że w opisanym uliniowieniu w każdej kolumnie zawierającej a stoją tylko spacje lub 0. Podobnie w każdej kolumnie zawierającej b stoją tylko spacje lub 1. Zatem każde a i każde b wnoszą przez porównywanie z $S_1^{\#}, \ldots, S_k^{\#}$ wartość k. Ponadto każda spacja z A lub B wnosi wartość będącą liczbą symboli z $S_1^{\#}, \ldots, S_k^{\#}$ (nie liczymy spacji) stojących w tej kolumnie. Ponieważ a i b nie stoją w jednej kolumnie oraz dwie spacje z A i B nie stoją w jednej kolumnie, to łącznie spacje z A i B wnoszą do wartości uliniowienia liczbę wszystkich symboli z $S_1, \ldots, S_k$, czyli M. Zatem

$$\sum_{i=1}^{k} [s(A, S_i^{\#}) + s(B, S_i^{\#})] = k \cdot m + M.$$

Tak wiec, stosując (5.5.1) i (5.5.2), wnioskujemy, że wartość tego uliniowienia wynosi $k \cdot M + (k+1) \cdot m$.

Dla dowodu implikacji ($\Leftarrow$) weźmy dowolne uliniowienie $S_1^{\#}, \ldots, S_k^{\#}, A, B$ pochodzące od słów $S_1, \ldots, S_k, a^p, b^{m-p}$ (dla pewnego $p \leq m$).

Załóżmy, że w tym uliniowieniu symbole a oraz b stoją w jednej kolumnie. Niech l_0 będzie liczbą wystąpień 0 w tej kolumnie. Podobnie, niech l_1 będzie liczbą wystąpień 1. Wówczas wkład w sumę $\sum_{i=1}^{k}[s(A, S_i^{\#}) + s(B, S_i^{\#})]$ pochodzący z kolumny ξ, w której znajduje się a, b wynosi $2k + l_0 + l_1$. Zmieńmy teraz uliniowienie, zastępując tę kolumnę dwiema kolumnami: jedna zawiera a oraz wszystkie wystąpienia 0 z kolumny ξ (w pozostałych miejscach stoją spacje), druga kolumna zawiera b oraz wszystkie wystąpienia 1 pochodzące z kolumny ξ (w pozostałych miejscach stoją spacje). Wówczas wkład tych dwóch kolumn wynosi $k + l_0$ (za pierwszą kolumnę) oraz $k + l_1$ (za drugą kolumnę). Zatem wartość uliniowienia się nie zmieniła. Tak więc, bez zmniejszenia ogólności możemy przyjąć, że uliniowienie $S_1^{\#}, \ldots, S_k^{\#}, A, B$ nie zawiera kolumny, w której jednocześnie występuje a oraz b.

Dowód implikacji ($\Leftarrow$) jest oparty na następującej obserwacji.

$$\sum_{i=1}^{k}[s(A, S_i^{\#}) + s(B, S_i^{\#})] \geq k \cdot m + M. \tag{5.5.3}$$

Ponadto, równość w (5.5.3) zachodzi $\Longleftrightarrow$ a nie stoi w jednej kolumnie z 1, b nie stoi w jednej kolumnie z 0 oraz w A i B nie znajdują się dwie spacje w jednej kolumnie.

Rozważmy jedną kolumnę uliniowienia. Niech l_0 oraz l_1 oznaczają liczbę wystąpień 0, odpowiednio 1, w tej kolumnie. Jeśli symbole stojące w A, B są a oraz $-$, to wkład takiej kolumny w wartość sumy w (5.5.3) wynosi $k + l_1 + l_0 + l_1 = k + l_0 + 2 \cdot l_1$. Podobnie, jeśli symbole stojące w A, B są b oraz $-$ wówczas wkład wynosi $k + 2 \cdot l_0 + l_1$. Jeśli natomiast w A, B znajdują się dwie spacje, to wkład takiej kolumny wynosi $2 \cdot (l_0 + l_1)$. Tak więc łącznie wkład w wartość sumy (5.5.3) wynosi co najmniej $k \cdot m + M$. To dowodzi (5.5.3). Z powyższej analizy wynika[1], że w (5.5.3) zachodzi równość $\Longleftrightarrow$ liczba jedynek stojących w jednej kolumnie z a jest zero, liczba zer stojących w jednej kolumnie z b jest zero oraz nie ma symboli (różnych od spacji) w kolumnie zawierającej dwie spacje z A, B. To dowodzi uwagi znajdującej się bezpośrednio po (5.5.3).

Zatem jeśli $S_1^{\#}, \ldots, S_k^{\#}, A, B$ jest uliniowieniem o wartości nie większej

[1]Przypomnijmy, że zakładamy że a oraz b nie znjadują się w jednej kolumnie.

niż $k \cdot M + (k+1) \cdot m$, to z (5.5.1) i (5.5.2) wynika, że

$$\sum_{i=1}^{k} [s(A, S_i^{\#}) + s(B, S_i^{\#})] \leq k \cdot m + M.$$

Zatem z (5.5.3) i z uwagi znajdującej się bezpośrednio po (5.5.3) wynika, że poniższa definicja słowa S jest poprawna.

$$S(l) = \begin{cases} 0 & \text{gdy } A(l) = a, \\ 1 & \text{gdy } B(l) = b. \end{cases}$$

Długość słowa S jest równa m oraz każde S_i jest podciągiem S. To kończy dowód twierdzenia. ∎

Stosując metodę dynamicznego programowania możemy obliczyć wartość $\overline{sim}(S_1, \ldots, S_k)$ dokładnie. Niestety czas potrzebny do wykonania obliczeń zależy wykładniczo od liczby słów k.

Twierdzenie 5.2.2 *Stosując metodę dynamicznego rpogramowania możemy obliczyć* $\overline{sim}(S_1, \ldots, S_k)$ *w czasie* $O(n^k)$*, gdzie* n *spełnia warunek* $|S_i| \leq n$*, dla* $i = 1, \ldots, k$.

Zadanie 5.2.1 Udowodnić Twierdzenie 5.2.2.

5.3 Uliniowienie gwiazdowe dla SP

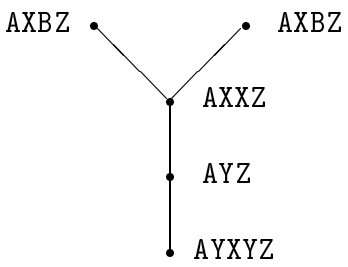
Drzewo jest to spójny graf (nie zorientowany) bez pętli. Będziemy rozpatrywać skończone drzewa, których wierzchołki są etykietowane słowami nad pewnym alfabetem Σ. Powiemy, że dwa wierzchołki są *sąsiadujące*, gdy są one połączone krawędzią.

Niech T będzie drzewem o k wierzchołkach, którego wierzchołki są etykietowane sławami $S_1, \ldots, S_k \in \Sigma^*$. Niech $S_1^{\#}, \ldots, S_k^{\#}$ będzie pewnym uliniowieniem dla $S_1, \ldots, S_k$. Niech s będzie pewną funkcją podobieństwa. Funkcja ta wyznacza podobieństwo $\overline{sim}(S_1, \ldots, S_k)$ liczone według miary SP. Powiemy, że uliniowienie $S_1^{\#}, \ldots, S_k^{\#}$ jest *zgodne* z T, gdy dla każdej pary sąsiadujących wierzchołków, jeśli S_i oraz S_j są etykietami tych wierzchołków, to $s(S_i^{\#}, S_j^{\#}) = \overline{sim}(S_i, S_j)$.

Przykład 5.3.1 Rozważmy uliniowienie

```
A X X - Z
A X B - Z
A - X - Z
A Y - - Z
A Y X Y Z
```

oraz drzewo T



Powyższe uliniowienie jest zgodne z T przy następującej funkcji podobieństwa

$$s(x,y) = \begin{cases} 0 & \text{gdy } x = y \\ 1 & \text{gdy } x \neq y. \end{cases}$$

□

Twierdzenie 5.3.1 *Dla każdego ciągu słów $S_1, \ldots, S_k$ i dla każdego drzewa T, którego wierzchołki są etykietowane słowami $S_1, \ldots, S_k$ istnieje uliniowienie $S_1^{\#}, \ldots, S_k^{\#}$ zgodne z T.*

Dowód: Indukcja ze względu na $k \geq 1$. Dla $k = 1$ twierdzenie jest oczywiste. Załóżmy, że twierdzenie zachodzi dla k i niech $S_1, \ldots, S_k, S_{k+1}$będzie dowolnym ciągiem słów oraz niech T będzie dowolnym drzewem o $k+1$ wierzchołkach, etykietowanych słowami $S_1, \ldots, S_k, S_{k+1}$. Bez zmniejszenia ogólności możemy przyjąć, że S_{k+1} jest etykietą pewnego liścia v (tzn. wierzchołka, z którego wychodzi dokładnie jedna krawędź). Niech T' będzie drzewem T bez liścia v. Na mocy założenia indukcyjnego znajdujemy uliniowienie $S_1^{\#}, \ldots, S_k^{\#}$, zgodne z T'. Niech S_i będzie etykietą wierzchołka sąsiadującego z v. Niech $S_i^{\#\#}, S_{k+1}^{\#}$ będzie optymalnym uliniowieniem dla $S_i^{\#}, S_{k+1}$. Niech

$S_1^{\#\#}, \ldots, S_k^{\#\#}$ powstaje z $S_1^{\#}, \ldots, S_k^{\#}$ przez wstawienie spacji w taki sam sposób jak $S_i^{\#\#}$ powstaje z $S_i^{\#}$. Wówczas $S_1^{\#\#}, \ldots, S_k^{\#\#}, S_{k+1}^{\#}$ jest uliniowieniem zgodnym z T. *Uwaga:* w powyższym dowodzie w istotny sposób korzystaliśmy z założenia, że $s(-,-) = 0$. ∎

Zauważmy, że jeśli $T_k(n)$ oznacza czas potrzebny do znalezienia uliniowienia zgodnego z danym drzewem dla k słów, (każde o długości $O(n)$), to mamy zależność

$$T_{k+1}(n) = T_k(n) + O(k \cdot n^2).$$

Zatem czas potrzebny na znalezienie takiego uliniowienia wynosi $O(k^2 \cdot n^2)$.

Niech $S_1, \ldots, S_k$ będzie dowolnym ciągiem słów. Słowo $S_c \in \{S_1, \ldots, S_k\}$ nazwiemy *centrum* ciągu $S_1, \ldots, S_k$, gdy $\sum_{j=1}^{k} \overline{sim}(S_j, S_c)$ jest minimalną wartością spośród wszystkich sum $\sum_{j=1}^{k} \overline{sim}(S_j, S_i)$, gdzie $i = 1, \ldots, k$.

Niech S_c będzie centrum ciągu $S_1, \ldots, S_k$. *Gwiazdą* (z centrum S_c) dla $S_1, \ldots, S_k$ nazwiemy drzewo G_c o wierzchołkach $v_1, \ldots, v_k$, takie że dla $i \neq c$ istnieje dokładnie jedna krawędź łącząca v_c z v_i oraz G_c nie ma innych krawędzi. Etykietą wierzchołka v_i jest S_i.

Twierdzenie 5.3.2 *Niech funkcja podobieństwa s spełnia warunki odległości. Niech S_c będzie centrum ciągu $S_1, \ldots, S_k$ i niech $S_1^{\#}, \ldots, S_k^{\#}$ będzie uliniowieniem zgodnym z gwiazdą G_c. Wówczas*

$$\frac{s(S_1^{\#}, \ldots, S_k^{\#})}{\overline{sim}(S_1, \ldots, S_k)} \leq 2 - 2/k.$$

Dowód: Na mocy warunku trójkąta mamy

$$s(S_1^{\#}, \ldots, S_k^{\#}) = \sum_{i<j} s(S_i^{\#}, S_j^{\#}) \leq \sum_{i<j} [s(S_i^{\#}, S_c^{\#}) + s(S_c^{\#}, S_j^{\#})]. \qquad (5.5.4)$$

Ze zgodności uliniowienia $S_1^{\#}, \ldots, S_k^{\#}$ z G_c wynika, że

$$s(S_i^{\#}, S_c^{\#}) = \overline{sim}(S_i, S_c),$$

dla $i \neq c$. Ponieważ s jest odległością to powyższa równość zachodzi również dla $i = c$. Dla każdego i, wyrażenie $s(S_i^{\#}, S_c^{\#})$ występuje w sumie (5.5.4) dokładnie $k-1$ razy. Zatem mamy

$$s(S_1^{\#}, \ldots, S_k^{\#}) \leq (k-1) \cdot \sum_{i=1}^{k} s(S_i^{\#}, S_c^{\#}) = (k-1) \cdot \sum_{i=1}^{k} \overline{sim}(S_i, S_c). \quad (5.5.5)$$

Niech $W_1, \ldots, W_k$ będzie optymalnym uliniowieniem dla $S_1, \ldots, S_k$ (w sensie miary SP). Wówczas mamy

$$\overline{sim}(S_1, \ldots, S_k) = \sum_{i<j} s(W_i, W_j) \geq \sum_{i<j} \overline{sim}(S_i, S_j) = (1/2) \cdot \sum_{i,j} \overline{sim}(S_i, S_j). \tag{5.5.6}$$

Ostatnia z powyższych równości wynika z faktu, że zbiór wszystkich par (i, j) rozbija się na rozłączne zbiory par spełniających: $i < j$, $i > j$, $i = j$. Ponadto $\overline{sim}(S, S) = 0$, bo s jest odległością.

Opierając się na definicji centrum mamy

$$\sum_{i,j} \overline{sim}(S_i, S_j) \geq k \cdot \sum_{j=1}^{k} \overline{sim}(S_j, S_c).$$

Zatem z (5.5.6) otrzymujemy

$$\overline{sim}(S_1, \ldots, S_k) \geq (k/2) \cdot \sum_{j+1}^{k} \overline{sim}(S_j, S_c).$$

Z (5.5.5) i z powyższej nierówności otzrymujemy

$$\frac{s(S_1^{\#}, \ldots, S_k^{\#})}{\overline{sim}(S_1, \ldots, S_k)} \leq \frac{k-1}{k} \cdot 2 = 2 - 2/k.$$

To kończy dowód ∎

Zauważmy, że Twierdzenie 5.3.2 sugeruje następujący algorytm aproksymacyjny.
Dane słowa $S_1, \ldots, S_k$

1. Znajdujemy centrum S_c. [Ten krok zajmuje $O(k^2 \cdot n^2)$ czasu.]
2. Znajdujemy uliniowienie $S_1^{\#}, \ldots, S_k^{\#}$ zgodne z gwiazdą G_c. [Ten krok zajmuje $O(k^2 \cdot n^2)$ czasu.]
3. Jako wynik algorytmu podajemy $s(S_1^{\#}, \ldots, S_k^{\#})$.

Mamy więc

Twierdzenie 5.3.3 *Wynik uliniowienia dla $S_1, \ldots, S_k$ można znaleźć w czasie $O(k^2 \cdot n^2)$. Wynik ten jest co najwyżej $(2 - 2/k)$ razy gorszy (większy) od wartości optymalnej $\overline{sim}(S_1, \ldots, S_k)$.*