4. Exercise 7.1. Show that Lemma 7.3 cannot be strengthened to

$$|overlap(r,r')| < max\{wt(c), wt(c')\}.$$

<u>*Lemma 7.3:*</u>

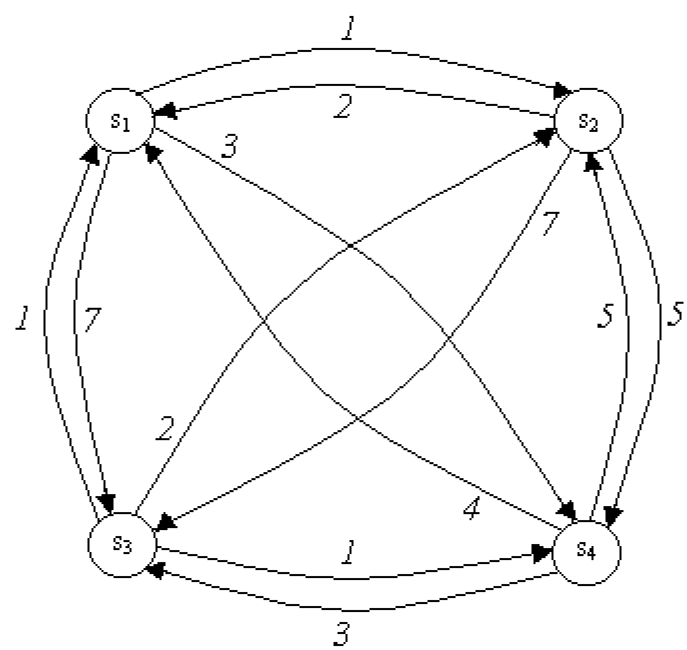
Let c and c' be two cycles in C, and let r, r' be representative strings from these cycles. Then

$$|overlap(r,r')| < wt(c) + wt(c').$$

We first give an example in which |overlap(r,r')| = max{wt(c), wt(c')}, and then attempt to generalize the example.

<u>Example</u>

Let S = { abcabca, bcabcab, aabca, abcaa }. Let s_i denote the i^{th} string in S. The prefix graph for this S is the following:



Upon examination, it becomes clear that a minimal cycle cover for this graph consists of the pair of edges between s_1 and s_2 and the pair of edges between s_3 and s_4. So, let c denote the cycle consisting of the edges between s_1 and s_2, and let c' denote the cycle consisting of the pair of edges between s_3 and s_4. Then wt(c) = 3 and wt(c') = 4. Now, suppose the representative strings for c and c' are chosen as follows: r = abcabca, r' = aabca. Then |overlap(r,r')| = 4 = max(wt(c), wt(c')).

We now show how to construct an infinite number of examples of this type:

S consists of 4 strings: s_1, s_2, s_3, and s_4.
Let ∑ denote the set of symbols from which the strings are formed.

s_1 is constructed as follows:

(1) Repeat a sequence of distinct symbols from ∑ twice. Let us denote the letters in the sequence as A, B, …, and let X denote the sequence.
(2) Append A to the end of the string so generated.

In the example given above, A = a, B = b, C = c. s_1 is ABCABCA.

s_2 is constructed as follows:

(1) s_2 is first set to equal s_1
(2) The first symbol in s_2 is removed.
(3) The new first symbol is s_2 is appended to the end of s_2.

In the example given above, these steps produce the following:

(1) s_2 = ABCABCA
(2) s_2 = BCABCA
(3) s_2 = BCABCAB

s_3 is constructed as follows:

(1) s_3 is first set to equal s_1
(2) The first occurrence of X is removed.
(3) A is appended to the beginning of s_3.

In the example given above, these steps produce the following:

(1) s_3 = ABCABCA
(2) s_3 = ABCA
(3) s_3 = AABCA

s_4 is constructed as follows:

(1) s_4 is first set to equal s_3
(2) A is removed from the beginning of s_4
(3) A is appended to the end s_4

In the example given above, these steps produce the following:

(1) s_4 = AABCA
(2) s_4 = ABCA
(3) s_4 = ABCAA

Note:

1) $\text{Prefix}(s_1,s_2) = A$ — $|\text{Prefix}(s_1,s_2)| = 1$
2) $\text{Prefix}(s_2,s_1) = X$ - the initial A — $|\text{Prefix}(s_2,s_1)| = |X| - 1$
3) $\text{Prefix}(s_3,s_4) = A$ — $|\text{Prefix}(s_3,s_4)| = 1$
4) $\text{Prefix}(s_4,s_3) = X$ — $|\text{Prefix}(s_4,s_3)| = |X|$

So, in a cycle cover with $c = \{s_1, s_2\}$ and $c' = \{s_3, s_4\}$:

$wt(c) = |X|$
$wt(c') = |X| + 1$
$wt(C) = 2|X| + 1$
$\max(wt(c), wt(c')) = wt(c') = |X| + 1$

5) $\text{Overlap}(s_3, s_1) = |X| + 1$.

So, if we can establish that a cycle cover with $c = \{s_1, s_2\}$ and $c' = \{s_3, s_4\}$ is a minimum cycle cover, then if we choose $r = s_1$ and $r' = s_3$, $|\text{overlap}(r,r')| = \max(wt(c), wt(c'))$. We establish that a cycle cover with $c = \{s_1, s_2\}$ and $c' = \{s_3, s_4\}$ is a minimum cycle cover, using the following additional information, below:

6) $\text{Prefix}(s_1,s_3) = XX$ — $|\text{Prefix}(s_1,s_3)| = 2|X|$
7) $\text{Prefix}(s_3,s_1) = 1$ — $|\text{Prefix}(s_3,s_1)| = 1$
8) $\text{Prefix}(s_1,s_4) = X$ — $|\text{Prefix}(s_1,s_4)| = |X|$
9) $\text{Prefix}(s_4,s_1) = XA$ — $|\text{Prefix}(s_4,s_1)| = |X| + 1$
10) $\text{Prefix}(s_2,s_3) = BCXAB$ — $|\text{Prefix}(s_2,s_3)| = 2|X| + 1$
11) $\text{Prefix}(s_3,s_2) = AA$ — $|\text{Prefix}(s_3,s_2)| = 2$
12) $\text{Prefix}(s_2,s_4) = BCX$ — $|\text{Prefix}(s_2,s_4)| = |X| + 2$
13) $\text{Prefix}(s_4,s_2) = XAA$ — $|\text{Prefix}(s_4,s_2)| = |X| + 2$

Given the information presented above, we can exhaustively examine the costs of all cycle covers to conclude that $c = \{s_1, s_2\}$ and $c' = \{s_3, s_4\}$ is a minimum cycle cover:

c	c'	wt(C)
$\{s_1,s_2\}$	$\{s_3,s_4\}$	$2\|X\| + 1$
$\{s_1,s_3\}$	$\{s_2,s_4\}$	$4\|X\| + 5$
$\{s_1,s_4\}$	$\{s_2,s_3\}$	$4\|X\| + 4$
(s_1,s_2,s_3,s_4)		$3\|X\| + 4$
(s_1,s_2,s_4,s_3)		$2\|X\| + 4$
(s_1,s_3,s_2,s_4)		$4\|X\| + 5$
(s_1,s_3,s_4,s_2)		$4\|X\| + 2$
(s_1,s_4,s_2,s_3)		$4\|X\| + 4$
(s_1,s_4,s_3,s_2)		$3\|X\| + 1$

5. Exercise 6.2. Give an approximation factor preserving reduction from the vertex cover problem to the feedback vertex set problem (thereby showing that improving the factor for the latter problem will also improve it for the former; also see Section 30.1).

We will give a polynomial time transformation from:

- an instance I of the Vertex Cover problem consisting of a graph G = (V, E) and a cost function $c: V \rightarrow \mathbf{Q}^+$

to

- an instance I' of the Feedback Vertex Set problem consisting of a graph G' = (V', E') and a cost function $c': V' \rightarrow \mathbf{Q}^+$.

We construct G' in two stages. First, we add V and E to G', maintaining the same costs for the vertices in V. Next, we add a set of vertices V_P to G', where:

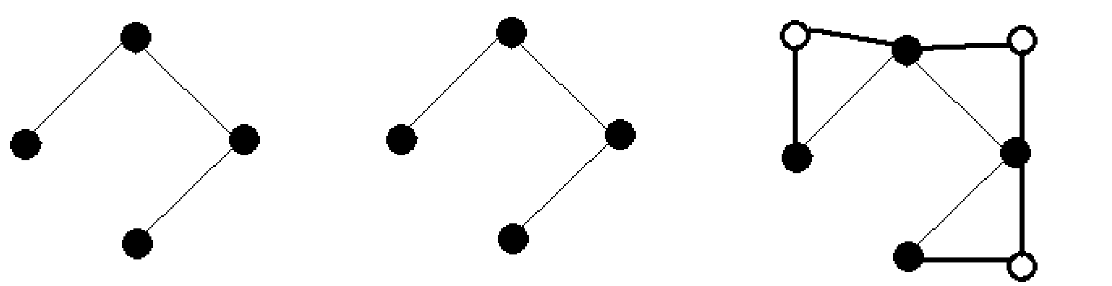
(1) $|V_P| = |E|$

(2) For all v_P in V_P, $c'(v_P) = 1 + \sum_{v \text{ in } V} c(v)$

That is, V_P contains one vertex for each edge in E, and the cost of each vertex in V_P is one greater than the sum of the costs of all vertices in V. We use Sum_V to denote the sum of the costs of all vertices in V.

We next add a set of edges connecting each of the vertices in V_P to exactly two vertices in $V' - V_P$. More specifically, we associate with each vertex v_P in V_P a unique edge e in E, and add an edge between v_P and both endpoints of e.

We illustrate the construction described on a small example below:



G   First Stage of G'   G' (Second Stage of G')

In the illustration, the hollow vertices are the vertices in V_P.

It is obvious that the construction requires time polynomial in |I|.

We first show that the cost of an optimal feedback vertex set in G' is no greater than the cost of an optimal vertex cover in G. To establish this, note that given any vertex cover VC for G, the corresponding vertices in G' make up a feedback vertex set for G'.

Aside

> The preceding statement is valid because every cycle in G' contains at least one edge between two vertices in $V' - V_P$. By removing all of the vertices corresponding to those in VC in building a feedback vertex set, we remove all of the edges between two vertices in $V' - V_P$, thereby breaking any cycles.

So, there is a feedback vertex set for G' with cost equal to the cost of an optimal vertex cover for G. Therefore, the cost of the *optimal* feedback vertex set for G' is no greater than the cost of an optimal vertex cover for G.

We now show that given a feedback vertex set FVS for G', we can obtain a vertex cover VC for G of at most the same cost. We consider separately two partitioning cases:

(1) FVS contains at least one vertex in V_P
(2) FVS contains no vertices in V_P

First, suppose that FVS contains at least one vertiex in V_P. Then the cost of FVS is at least 1 + Sum_V (defined above). So, in this case, simply choosing every vertex in $V' - V_P$ provides a vertex cover for G with a cost less than the cost of FVS.

Next, suppose that FVS contains no vertices in V_P. Then we claim that the vertices corresponding to those in FVS comprise a vertex cover for G.

Aside

> To see this, suppose for contradiction that the vertices corresponding to those in FVS do not comprise a vertex cover for G. Then there is an edge e between two vertices v_1 and v_2 in $V' - V_P$ where v_1 and v_2 are not in FVS. But then there is also a vertex v in V_P with edges to both v_1 and v_2. So, since FVS does not include v (by hypothesis), there exists a cycle ($v - v_1 - v_2$) in the graph induced by removing the vertices in FVS. But this is a contradiction and we conclude that the vertices corresponding to those in FVS comprise a vertex cover for G.

So, in this case, choosing every vertex in FVS provides a vertex cover for G with a cost equal to the cost of FVS.