$x_1, x_2, \ldots, x_k,\ y_1, y_2, \ldots,\ y_k$. Does D have a collection of disjoint paths $P_1, P_2, \ldots, P_k$ such that P_i is an (x_i, y_i)-path, for every $i \in [k]$?

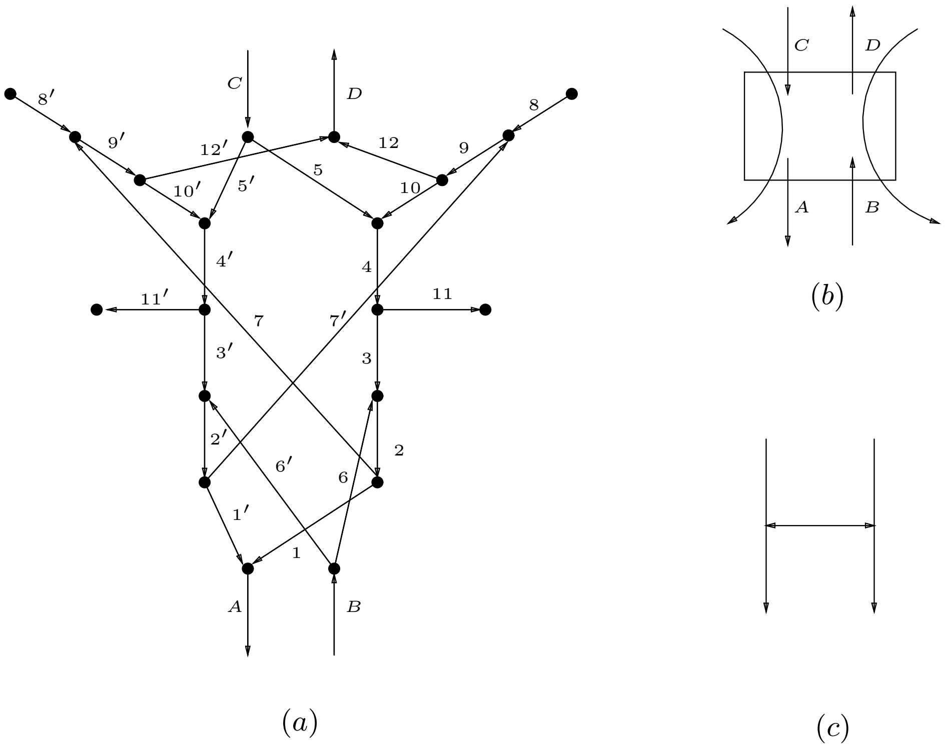
10.2 The Complexity of the k-Linkage Problem

We start with the following result by Fortune, Hopcroft and Wyllie showing that already for $k = 2$ the k-linkage problem is very difficult for general digraphs.

Theorem 10.2.1 *[332] The 2-linkage problem is $\mathcal{NP}$-complete.*

Since this theorem is very important and the gadget[2] construction used in the proof is quite illustrative, we give the proof in detail below. We follow the proof in [332].

First we need a lemma whose proof is left as Exercise 10.4.



(a)

(b)

(c)

Figure 10.1 Part (a) shows a switch S. Parts (b) and (c) show schematic pictures of a switch ([332, Fig. 1]). In (c) the two vertical arcs correspond to the paths (8,9,10,4,11), respectively, (8',9',10',4',11'). Note that for convenience, we label the arcs, rather than the vertices, in this Figure.

---

[2] Quite often $\mathcal{NP}$-completeness proofs are constructed by piecing together certain **gadgets** about which one can prove certain properties. Based on these properties one then shows that the whole construction has the desired properties. For other instances of this technique, see e.g. Chapters 6 and 16.

Lemma 10.2.2 *[332] Consider the digraph S shown in Figure 10.1(a). Suppose there are two disjoint paths P, Q passing through S such that P leaves S at A and Q enters S at B. Then P must enter S at C and Q must leave S at D. Furthermore, there exists exactly one more path R passing through S which is disjoint from P, Q and this is either*

$$(8, 9, 10, 4, 11) \qquad \textit{or} \qquad (8', 9', 10', 4', 11'),$$

depending on the actual routing of P. □

The digraph S in Figure 10.1 is called a **switch**. We can stack arbitrarily many switches on top of each other and still have the conclusion on Lemma 10.2.2 holding for each switch. The way we stack is simply by identifying the C and D arcs of one switch with the A and B arcs of the next (see Figure 10.2). A switch can be represented schematically as in Figure 10.1(c), or, when we want to indicate stacking of switches, as in Figure 10.1(b).

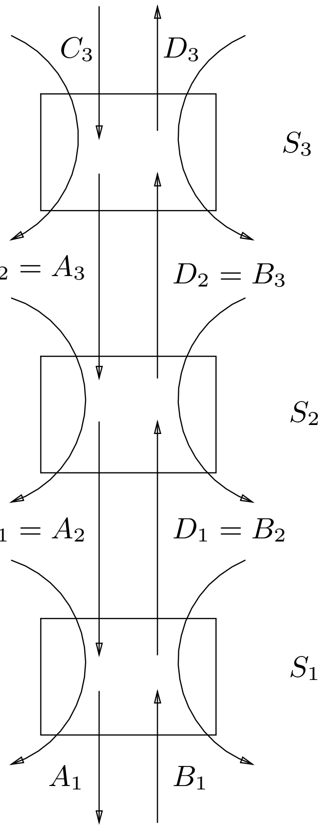


Figure 10.2 Stacking three switches on top of each other.

Proof of Theorem 10.2.1: The reduction is from 3-SAT (see the definition in Section 17.5). Let $\mathcal{F} = C_1 * C_2 * \ldots * C_r$ be an instance of 3-SAT with variables $x_1, x_2, \ldots, x_k$. For each variable x_i we let H_i be the digraph consisting of two internally disjoint (u, v)-paths of length r (the number of clauses in $\mathcal{F}$). We associate one of these paths with the literal x_i and the other with the literal $\overline{x}_i$. We are now ready to explain the construction of the digraph $D[\mathcal{F}]$ and show that it contains disjoint (u_1, v_1)-, (u_2, v_2)-paths if and only if $\mathcal{F}$ is satisfiable.

See Figure 10.3. We form a chain $H_1 \rightarrow H_2 \rightarrow \ldots \rightarrow H_k$ on the subdigraphs corresponding to each variable (see the middle of the figure, H_i corresponds to the variable x_i). With each clause C_i we associate three switches, one for each literal it contains. The left paths of these switches (that is, the paths in the left-hand part of the figure) all start at the vertex n_{i-1} and end at n_i. The right path of each switch is substituted for a (private) arc of H_i such that the arc is taken from the path which corresponds to x_i if the literal is x_i and from the path which corresponds to $\overline{x}_i$ if the literal is $\overline{x}_i$. The substitution is shown for the clause $C_i = x_1 + \overline{x}_2 + x_5$ in the figure. By the choice of the lengths of the paths in H_i we can make this substitution so that different arcs in H_i are substituted by different switches corresponding to several clauses, all of which contain the literal x_i or $\overline{x}_i$. The switches corresponding to the clause C_i are denoted $S_{i,1}, S_{i,2}, S_{i,3}$. We stack these switches in the order $S_{1,1}S_{1,2}S_{1,3} \ldots S_{r,1}S_{r,2}S_{r,3}$ as shown in the right part of the figure. A two-way arc between a clause and some H_j (shown only for C_i) indicates a switch that is substituted for these arcs[3]. Finally, we join the D arc of the switch $S_{r,3}$ to the vertex z_1 of H_1, add an arc from w_k in H_k to n_0 and choose vertices u_1, u_2, v_1, v_2 as shown (that is, u_2 is the tail of the C arc for $S_{r,3}$, u_1 is the tail of the B arc of $S_{1,1}$ and v_2 is the head of the A arc of $S_{1,1}$). This completes the description of $D[\mathcal{F}]$.

We claim that $D[\mathcal{F}]$ contains disjoint (u_1, v_1)-, (u_2, v_2)-paths if and only if $\mathcal{F}$ is satisfiable. Suppose first that $D[\mathcal{F}]$ has disjoint (u_1, v_1)-, (u_2, v_2)-paths P, Q. It follows from the definition of $D[\mathcal{F}]$ that the paths P and Q will use all the arcs that go between two switches (i.e., those arcs that are explicitly shown in the right-hand side of Figure 10.3). Hence, by Lemma 10.2.2, after removing the arcs of Q and the arcs of P from u_1 to the first vertex z_1 of H_1, the only remaining way to pass through a switch $S_{i,j}$ is to use either the right path or the left path of $S_{i,j}$ but not both! By the construction of $D[\mathcal{F}]$, P must traverse the subdigraphs corresponding to the variables in the order $H_1, H_2, \ldots, H_k$ and each time P uses precisely one of the two paths in H_i (recall again that some of the arcs in H_i in Figure 10.3 correspond to the right path of some switch). Let T be the truth assignment which sets $x_i := 1$ if P uses the path corresponding to $\overline{x}_i$ and let $x_i := 0$ in the opposite case. We show that this is a satisfying truth assignment for $\mathcal{F}$.

It follows from the construction of $D[\mathcal{F}]$ and the remark above on arcs used by Q and the first part of P from u_1 to H_1 that the path P contains all the vertices $n_0, n_1, \ldots, n_r$ in that order. Since each of the paths from n_j to

---

[3] Note that this is the same switch which is shown in the right-hand side of the figure!

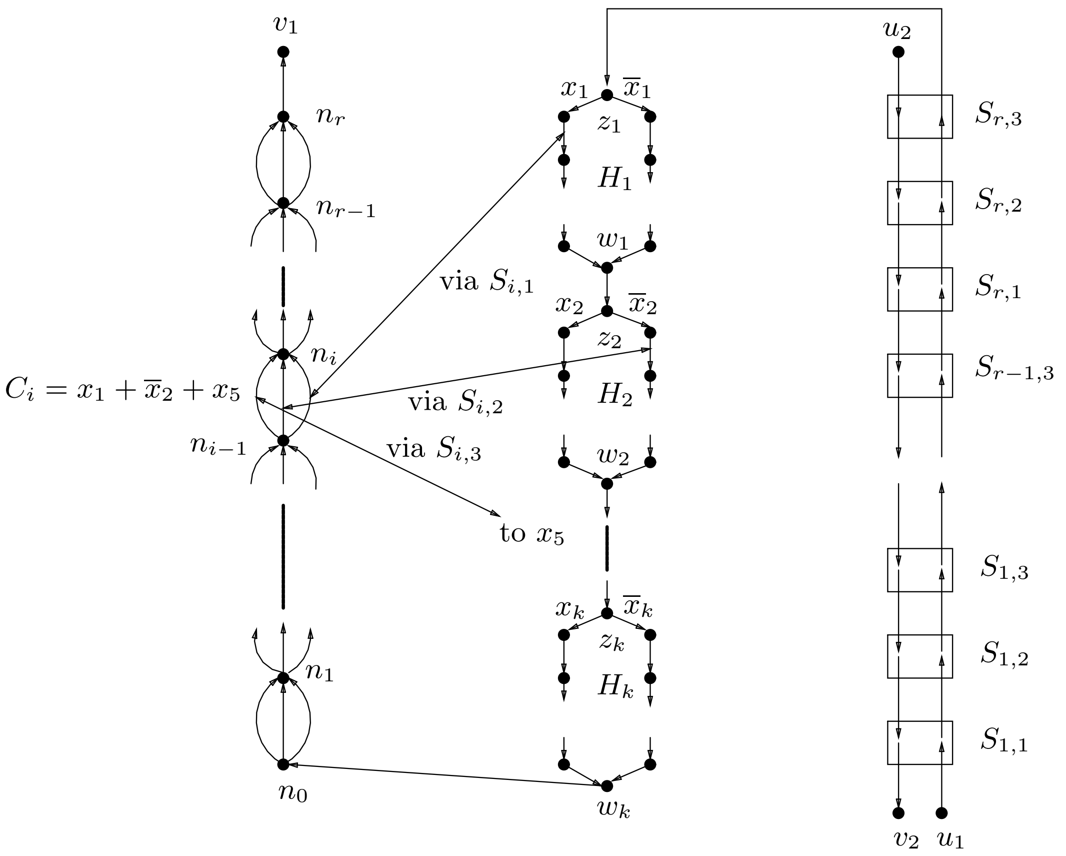


Figure 10.3 A schematic picture of the digraph $D[\mathcal{F}]$.

n_{j+1} is part of a switch for every $j = 0, 1, \ldots r-1$, we must use the left path of precisely one of these switches to go from n_j to n_{j+1}. By Lemma 10.2.2, every time we use a left path of a switch, the right path cannot also be used. From this we see that for each clause C_j, $j \in [r]$, it must be the case that at least one of the literals y (in particular the one whose left path we could use) of C_j becomes satisfied by our truth assignment. This follows because P must use the path corresponding to $\overline{y}$ in the middle. Thus we have shown that $\mathcal{F}$ is satisfiable.

Suppose now that T' is a satisfying truth assignment for $\mathcal{F}$. Then for every variable x_i which is true (false) we can use the subpath corresponding to $\overline{x}_i$ (x_i) in H_i. For each clause C_j we can fix one literal which is true and use the left path of the switch that corresponds to that literal (that path cannot be blocked by the way we chose subpaths inside the H_i's). By Lemma 10.2.2, we can find disjoint paths P, Q such that P starts in u_1 and ends in the initial vertex z_1 of H_1 and Q is a (u_2, v_2)-path in the right part of $D[\mathcal{F}]$. Furthermore, by the same lemma, after removing the vertices of P and Q, we still have the desired paths corresponding to each literal available. This shows that we can route the disjoint (u_1, v_1)-, (u_2, v_2)-paths in $D[\mathcal{F}]$. $\square$