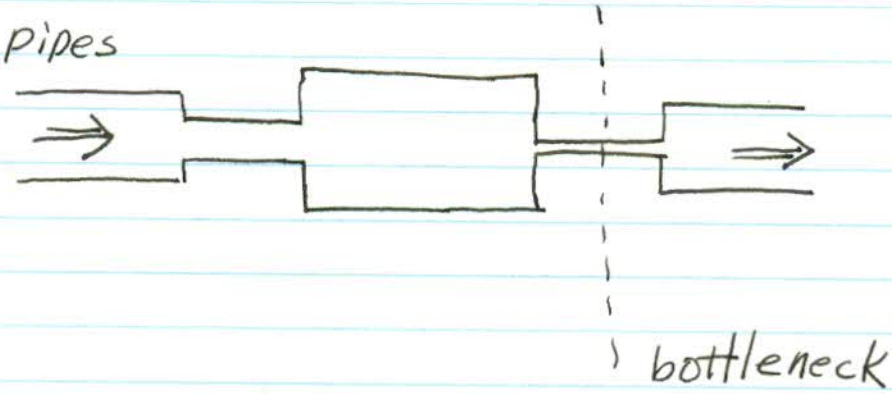
Max Flow problem

Max flow is determined by the "bottleneck"



Reverse ~~a~~ edge (flow)

$$u \xrightarrow[f_c]{} v$$

$$\downarrow$$

$$u \xrightarrow[f_c]{f} v$$

$f_c^{(u,v)}$ — capacity of the edge (u,v)

$f^{(u,v)}$ — flow pushed from u to v

$f_r^{(u,v)}$ — residual flow (how much more we can push from u to v)

$$f_r = f_c - f$$

Now Initially:

$f_c(v,u) = 0$ (cannot push from v to u)

but after a flow $f(u,v)$ has been pushed $u \to v$,

$$f_r(v,u) = f_c(v,u) - f(v,u)$$
$$= 0 - (-f(u,v))$$
$$f_r(v,u) = f(u,v)$$

That is why we got a reverse flow.

Note: $f(v,u) = -f(u,v)$

Residual Network G_f New flow network

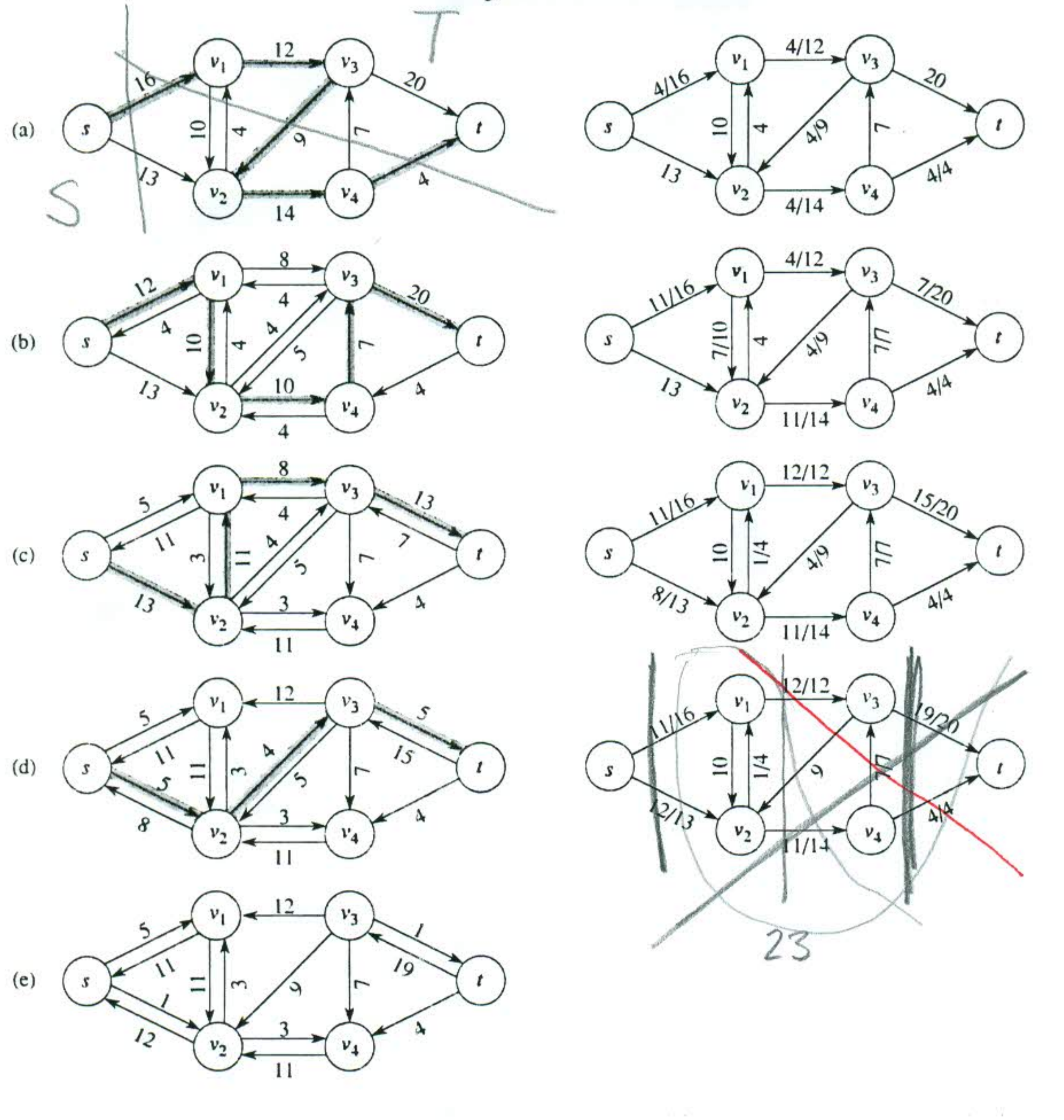


Figure 26.5 The execution of the basic Ford-Fulkerson algorithm. **(a)–(d)** Successive iterations of the **while** loop. The left side of each part shows the residual network G_f from line 4 with a shaded augmenting path p. The right side of each part shows the new flow f that results from adding f_p to f. The residual network in (a) is the input network G. **(e)** The residual network at the last **while** loop test. It has no augmenting paths, and the flow f shown in (d) is therefore a maximum flow.

<u>Note</u>: (b) Since ~~we~~ you push a flow of 4 from S → V_1, ~~we~~ you can push 16−4=12 from S → V_1 possibly. Also ~~we~~ you can push 4 from V_1 → S to cancel out 4 from S → V_1 resulting in 0 capacity from V_1 → S which is true.

HW4

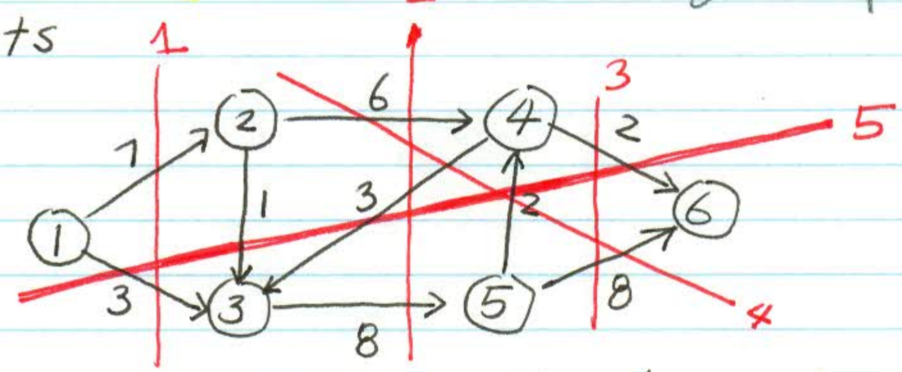
#2

Help understand cut

Side note:

Dual problem

cuts



Original graph w/ capacities

The capacity of the following cuts

Cut	capacity (from source set to sinkset)
1	$7+3=10$
2	$8+6=14$ (Maximize the capacity across the cut, we don't have to use backward arc from ④→③)
3	$2+8=10$
4	$6+2+8=16$ (the same reason)
5	$3+1+3+2=9$ (we don't use the backward arc ⑤→④)

⋮

more cuts

Cut 5 is the minimum/minimal cut because the total capacity across the cut is the minimum among all possible cuts on the graph.