CHAPTER 10: NETWORK OPTIMIZATION MODELS

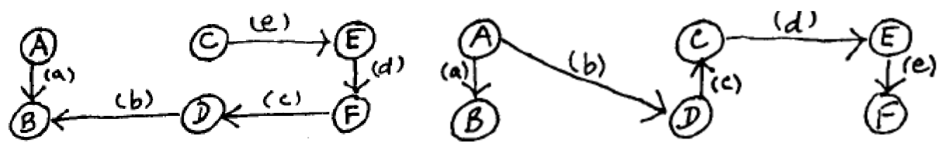
10.2-1.

(a) Directed path: AD-DC-CE-EF (A → D → C → E → F)
Undirected paths: AD-FD (A → D → F)
CA-CE-EF (A → C → E → F)
AD-ED-EF (A → D → E → F)

(b) Directed cycles: AD-DC-CA
DC-CE-ED
DC-CE-EF-FD
Undirected cycle that includes every node: CA-CE-EF-FD-DB-AB

(c) {CA, CE, DC, FD, DB} is a spanning tree.

(d)



10.3-1.

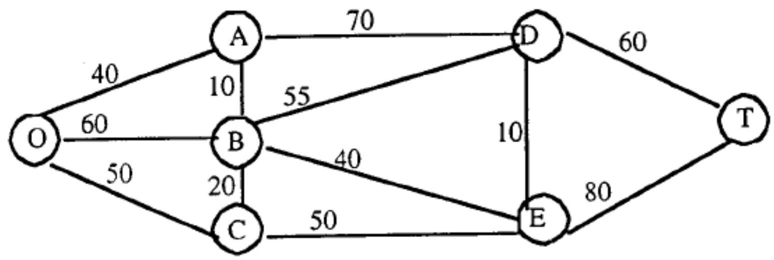
Prior to this study, Canadian Pacific Railway (CPR) used to run trains only after a sufficient level of freight was attained. This policy resulted in unreliable delivery times, so poor customer service. In order to improve customer service and utilization of available resources, CPR designed the railway operating plan called Integrated Operating Plan (IOP). "The problem of designing a railway operating plan is to satisfy a set of customer requirements expressed in terms of origin-destination traffic movements, using a blocking plan and a train plan. Thus, the primary variables are the blocks and trains. The constraints are the capacities of the lines and yards, the customer-service requirements, and the availability of various assets, such as crews and locomotives. The objective function in an abstract sense is to maximize profits" [p. 8].

Developing the blocking plan, i.e., determining the group of railcars to move together at some point during their trips, involves solving a series of shortest-path problems over a directed graph. The train plan is based on the blocking plan. It includes departure and arrival times for the trains, blocks they pick up and crew schedules. This problem is solved for each train using heuristics. Following this, simulation models and locomotive cycle plans are developed.

This study enabled CPR to save $170 million in half a year. "Total documented cost savings through the end of 2002 have exceeded half a billion dollars" [p. 12]. More savings are expected in following years. The improvements in CPR's profitability and operations can be attributed to the decrease in transit and dwelling times, lowered fuel consumption, reduction of the workforce and of the number of railcars, and balanced workloads. CPR can now schedule the trains and the crew more efficiently and provide a more reliable customer service. By allowing variability in the parameters of its plans, CPR gained flexibility and agility. It can now respond to disruptions more effectively by shifting resources quickly. These improvements earned CPR many awards and more importantly a significant competitive advantage.

10.3-2.

(a)



(b)

n	Solved Nodes Directly Connected to Unsolved Nodes	Closest Connected Unsolved Node	Total Distance Involved	*n*th Nearest Node	Minimum Distance	Last Connection
1	O	A	40	A	40	OA
2, 3	O A	C B	50 40+10 = 50	C B	50 50	OC AB
4	A B C	D E E	40+70 = 110 50+40 = 90 50+50 = 100	E	90	BE
5	A B E	D D D	40+70 = 110 50+55 = 115 90+10 = 100	D	100	ED
6	D E	T T	100+60 = 160 90+80 = 170	T	160	DT

The shortest path from the origin to the destination is O → A → B → E → D → T, with a total distance of 160 miles.

(c)

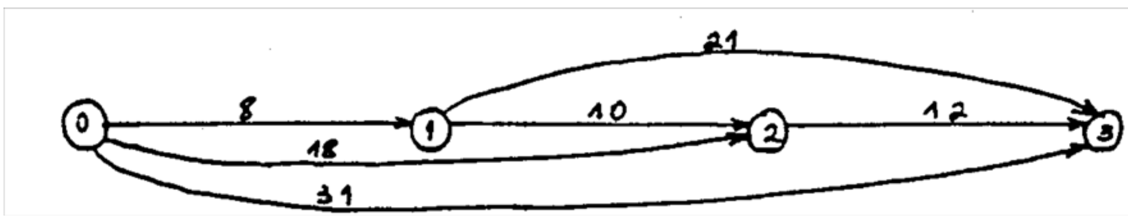
From	To	On Route	Distance (miles)		Nodes	Net Flow		Supply/Demand
Origin	A	1	40		Origin	1	=	1
Origin	B	0	60		A	0	=	0
Origin	C	0	50		B	0	=	0
A	B	1	10		C	0	=	0
A	D	0	70		D	0	=	0
B	A	0	10		E	0	=	0
B	C	0	20		Destination	-1	=	-1
B	D	0	55					
B	E	1	40					
C	B	0	20					
C	N	0	20					
C	E	0	50					
D	A	0	70					
D	B	0	55					
D	E	0	10					
D	Destination	1	60					
E	D	1	10					
E	Destination	0	80					
Total Distance (miles)		160						

(d) Yes.

(e) Yes.

10.3-3.

(a) The nodes represent the years. Let d_{ij} be the cost (in thousand dollars) of using the same tractor from the end of year i to the end of year j.



(b)

n	Solved nodes connected to unsolved nodes	its closest connected unsolved node	total distance involved	nth nearest node	its minimum distance	its last connection
1	0	1	8	1	8	01
2	0	2	18	2	18	02
	1	2	8+10=18	2		12
3	0	3	31			
	1	3	8+21=29	3	29	13
	2	3	18+12=30			

The minimum-cost strategy is to replace the tractor at the end of the first year and keep the new one until the end of the third year. This incurs a total cost of $29 thousand.

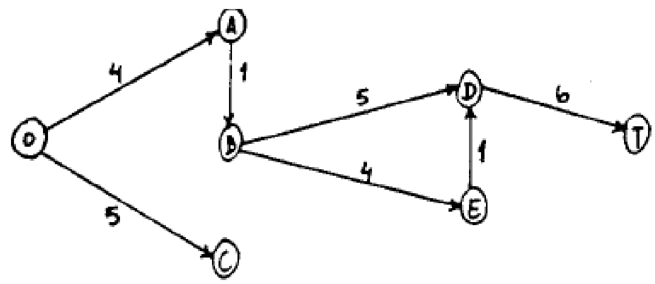
(c)

From	To	On Route	Cost		Nodes	Net Flow		Supply/Demand
Year 0	Year 1	1	$8,000		Year 0	1	=	1
Year 0	Year 2	0	$18,000		Year 1	0	=	0
Year 0	Year 3	0	$31,000		Year 2	0	=	0
Year 1	Year 2	0	$10,000		Year 3	-1	=	-1
Year 1	Year 3	1	$21,000					
Year 2	Year 3	0	$12,000					
	Total Cost	$29,000						

10.3-4.

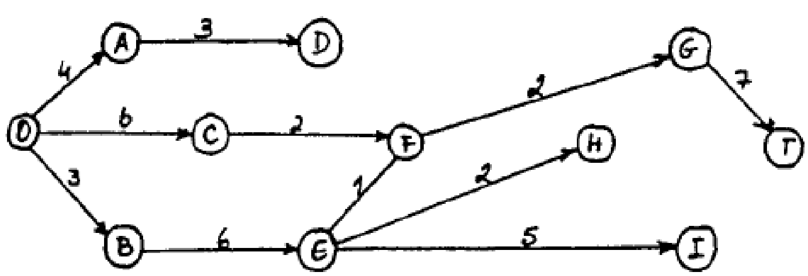
(a) Length of the shortest path: 16

0	4	5	5	10	9	16
O	A	B	C	D	E	T
OA-4	AB-1	~~BA-1~~	~~CB-2~~	~~DE-1~~	ED-1	
OC-5	~~AD-7~~	~~BC-2~~	~~CO-5~~	~~DB-5~~	~~EB-4~~	
~~OB-6~~		BE-4		DT-6	~~EC-5~~	
		BD-5		DA-7	~~ET-8~~	



(b) Length of the shortest path: 17

0	4	3	6	7	9	8	10	11	14	17
O	A	B	C	D	E	F	G	H	I	T
OB-3	AD-3	~~BC-4~~	~~CD-2~~	~~DE-2~~	~~EF-1~~	FE-1	~~GF-2~~	~~HG-2~~	~~IH-3~~	
OA-4	~~AC-5~~	BE-6	CF-2	~~DF-2~~	EH-2	~~FC-2~~	~~GH-2~~	~~HE-2~~	~~IT-4~~	
OC-6			~~CB-4~~	~~DA-3~~	EI-5	~~FD-2~~	~~GD-4~~	~~HI-3~~	~~IE-5~~	
			~~CE-5~~	~~DG-4~~	~~EC-5~~	FG-2	GT-7	~~HF-5~~		
			~~CA-5~~		~~EB-6~~	~~FH-5~~		~~HT-8~~		



10.3-5.

The shortest-path problem is a minimum cost flow problem with a unit supply at the origin and a unit demand at the destination. Label the origin as node 1 and the destination as node n. Then, the LP formulation is as follows:

$$\begin{aligned}
\text{minimize} \quad & z = \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}x_{ij} \\
\text{subject to} \quad & \sum_{j=1}^{n} x_{1j} - \sum_{j=1}^{n} x_{j1} = 1 \\
& \sum_{j=1}^{n} x_{ij} - \sum_{j=1}^{n} x_{ji} = 0, \text{ for } 2 \leq i \leq n-1 \\
& \sum_{j=1}^{n} x_{nj} - \sum_{j=1}^{n} x_{jn} = -1 \\
& 0 \leq x_{ij} \leq 1, \text{ for } 1 \leq i, j \leq n.
\end{aligned}$$

10.3-6.

(a) The flying times play the role of "distances."

(b) Shortest path: SE → C → E → LN, with total flight time 11.3

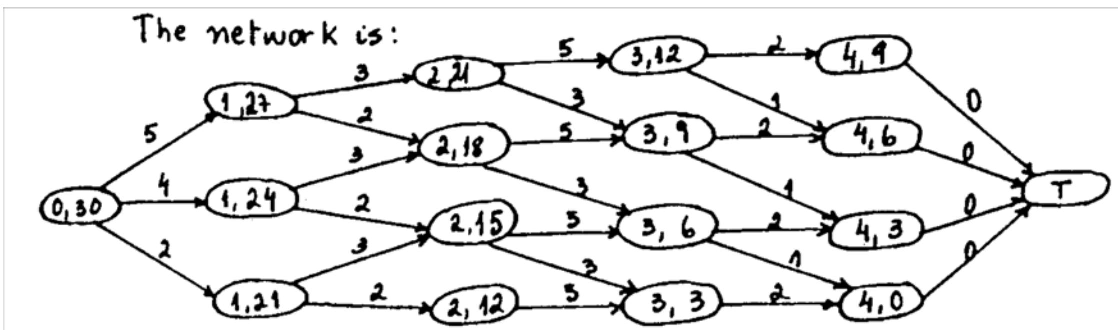
n	Solved Nodes Directly Connected to Unsolved Nodes	Closest Connected Unsolved Node	Total Distance Involved	*n*th Nearest Node	Minimum Distance	Last Connection
1	SE	C	4.2	C	4.2	SE-C
2	SE C	A F	4.6 4.2+3.4 = 7.6	A	4.6	SE-A
3	SE C A	B F E	4.7 4.2+3.4 = 7.6 4.6+3.4 = 8	B	4.7	SE-B
4	A B C	E E F	4.6+3.4 = 8 4.7+3.2 = 7.9 4.2+3.4 = 7.6	F	7.6	C-F
5	A B C F	E E E LN	4.6+3.4 = 8 4.7+3.2 = 7.9 4.2+3.5 = 7.7 7.6+3.8 = 11.4	E	7.7	C-E
6	A B F E	D D LN LN	4.6+3.5 = 8.1 4.7+3.6 = 8.3 7.6+3.8 = 11.4 7.7+3.6 = 11.3	D	8.1	A-D
7	D E F	LN LN LN	8.1+3.4 = 11.5 7.7+3.6 = 11.3 7.6+3.8 = 11.4	LN	11.3	E-LN

(c)

From	To	On Route	Time (hours)		Nodes	Net Flow		Supply/Demand
Seattle	A	0	4.6		Seattle	1	=	1
Seattle	B	0	4.7		A	0	=	0
Seattle	C	1	4.2		B	0	=	0
A	D	0	3.5		C	0	=	0
A	E	0	3.4		D	0	=	0
B	D	0	3.6		E	0	=	0
B	E	0	3.2		F	0	=	0
B	F	0	3.3		London	-1	=	-1
C	E	1	3.5					
C	F	0	3.4					
D	London	0	3.4					
E	London	1	3.6					
F	London	0	3.8					
Total Time (hours)		11.3						

10.3-7.

(a) Let node (i, j) denote phase i being completed with j million dollars left to spent and $t_{(i,j),(i+1,k)}$ be the time to complete phase $i+1$ if a cost of $(j-k)$ million dollars is spent.



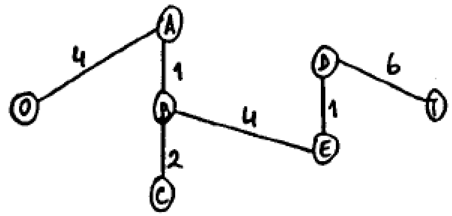
(b) Shortest path: $(0,30) \xrightarrow{2} (1,21) \xrightarrow{3} (2,15) \xrightarrow{3} (3,3) \xrightarrow{2} (4,0) \xrightarrow{0}$ T. Time= 10 months.

n	Solved nodes connected to unsolved nodes	its closest connected unsolved node	total distance involved	nth nearest node	its minimum distance	its last connection
1	(0,30)	(1,21)	2	(1,21)	2	(0,30)-(1,21)
2	(0,30)	(1,24)	4	(1,24)	4	(0,30)-(1,24)
	(1,21)	(2,12)	2+2=4	(2,12)	4	(1,21)-(2,12)
4	(0,30)	(1,27)	5	(1,27)	5	(0,30)-(1,27)
	(1,21)	(2,15)	2+3=5	(2,15)	5	(1,21)-(2,15)
	(1,24)	(2,15)	4+2=6			
	(2,12)	(3,3)	4+5=9			
6	(1,24)	(2,18)	4+3=7	(2,18)	7	(1,24)-(2,18)
	(1,27)	(2,18)	5+2=7	(2,18)	7	(1,27)-(2,18)
	(2,12)	(3,3)	4+5=9			
	(2,15)	(3,3)	5+3=8			
7	(1,27)	(2,21)	5+3=8	(2,21)	8	(1,27)-(2,21)
	(2,12)	(3,3)	4+5=9			
	(2,15)	(3,3)	5+3=8	(3,3)	8	(2,15)-(3,3)
	(2,18)	(3,6)	7+3=10			
9	(2,15)	(3,6)	5+5=10	(3,6)	10	(2,15)-(3,6)
	(2,18)	(3,6)	7+3=10	(3,6)	10	(2,18)-(3,6)
	(2,21)	(3,9)	8+3=11			
	(3,3)	(4,0)	8+2=10	(4,0)	10	(3,3)-(4,0)
11	(2,18)	(3,9)	7+5=12			
	(2,21)	(3,9)	8+3=11			
	(3,6)	(4,3)	10+2=12			
	(4,0)	T	10+0=10	T	10	(4,0)-T

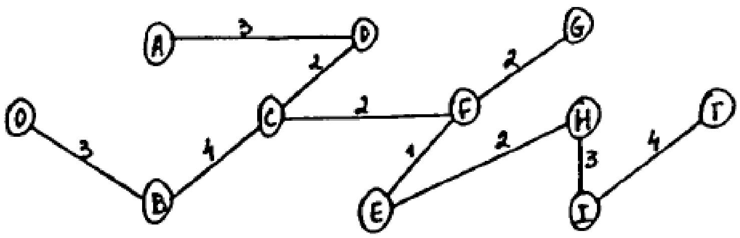
Phase	Level	Cost	Time
Research	Crash	9	2
Development	Priority	6	3
Design	Crash	12	3
Production	Priority	3	2

10.4-1.

(a) Length: 18



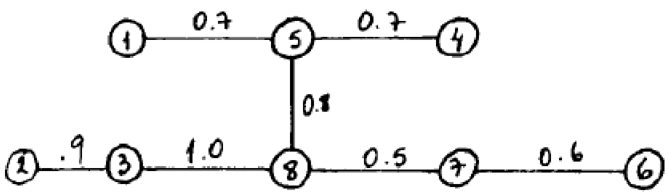
(b) Length: 26



10.4-2.

(a) The nodes represent the groves and the branches represent the roads.

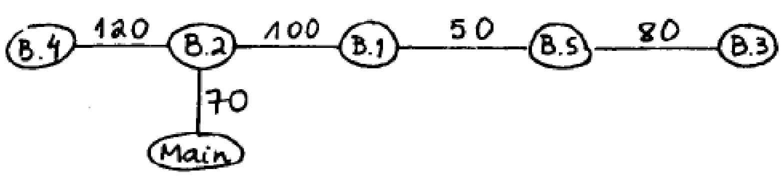
(b) Length: 5.2



10.4-3.

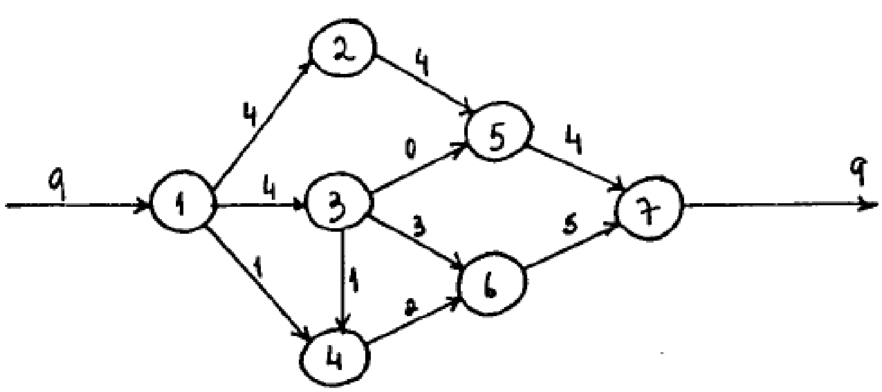
(a) The nodes are Main Office, Branch 1, Branch 2, Branch 3, Branch 4, and Branch 5. The branches are the phones lines.

(b)



10.5-1.

Maximum flow: 9



10.5-2.

Let node 1 be the source and node N be the sink.

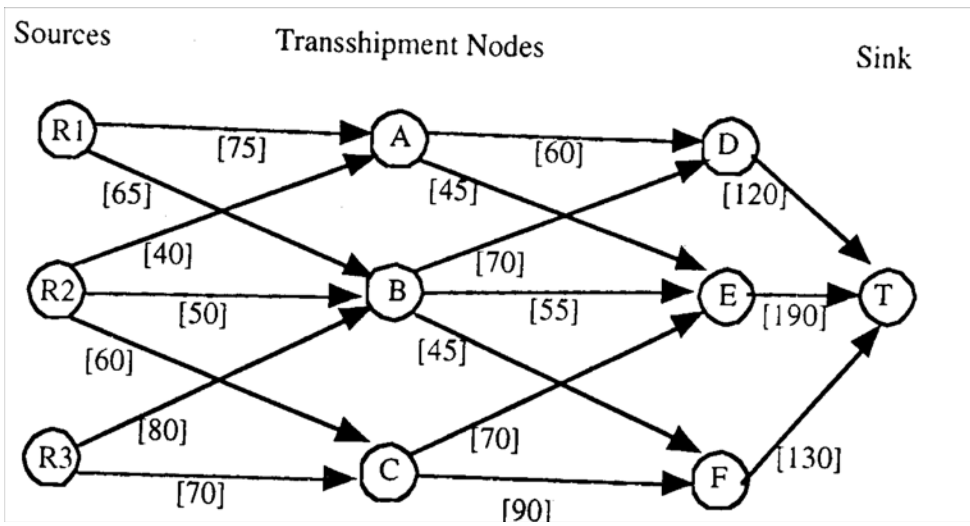
$$\text{maximize} \qquad z = \sum_{j=2}^{N} x_{1j}$$

$$\text{subject to} \qquad \sum_{j=1, j\neq i}^{N} x_{ij} - \sum_{j=1, j\neq i}^{N} x_{ji} = 0, \text{ for } i = 2, 3, \ldots, N-1$$

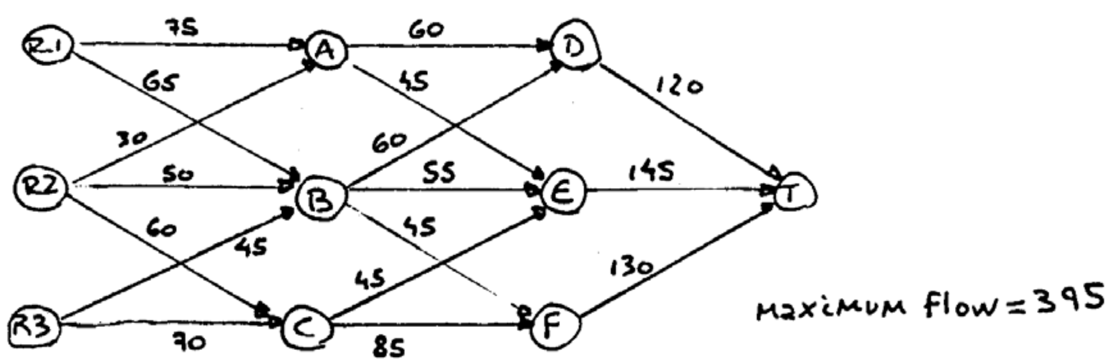
$$0 \leq x_{ij} \leq c_{ij}, \text{ where } c_{ij} = 0 \text{ if } (i, j) \text{ is not a branch.}$$

10.5-3.

(a)



(b) Maximum flow = 395

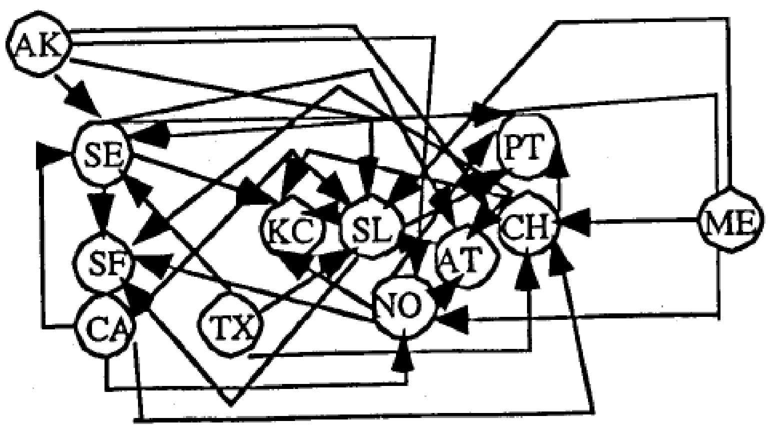


(c) Maximum flow = 395

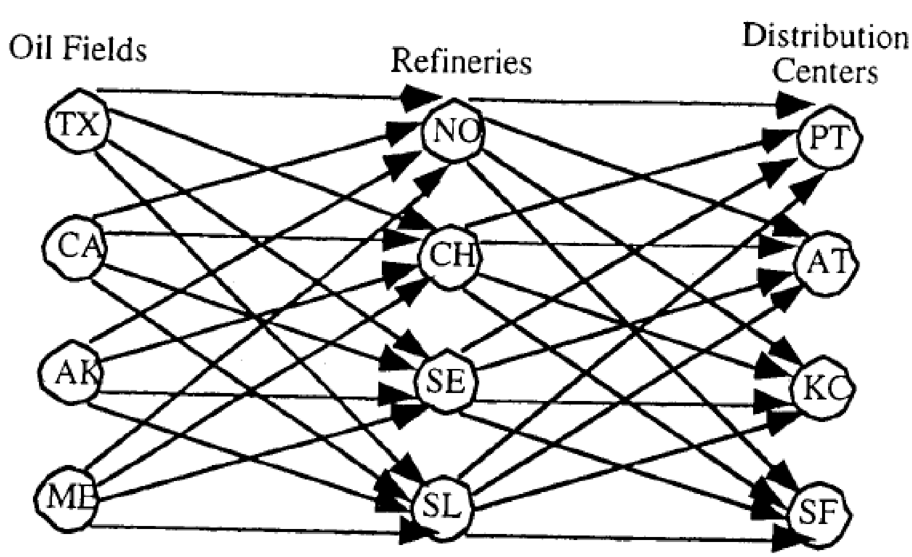
From	To	Ship		Capacity		Nodes	Net Flow		Supply/Demand
R1	A	65	<=	75		R1	95		
R1	B	30	<=	65		R2	150		
R2	A	40	<=	40		R3	150		
R2	B	50	<=	50		A	0	=	0
R2	C	60	<=	60		B	0	=	0
R3	B	80	<=	80		C	0	=	0
R3	C	70	<=	70		D	0	=	0
A	D	60	<=	60		E	0	=	0
A	E	45	<=	45		F	0	=	0
B	D	60	<=	70		T	-395		
B	E	55	<=	55					
B	F	45	<=	45					
C	E	45	<=	70					
C	F	85	<=	90					
D	T	120	<=	120					
E	T	145	<=	190					
F	T	130	<=	130					
Maximum Flow		395							

10.5-4.

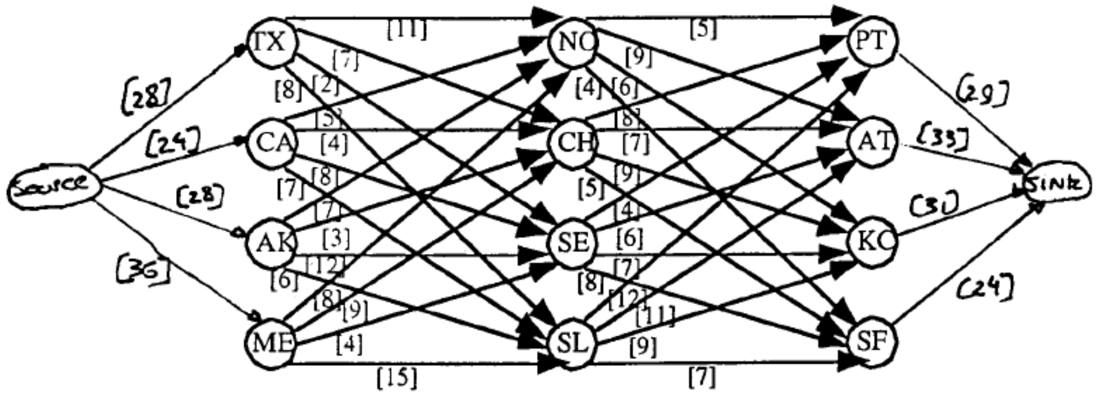
(a)



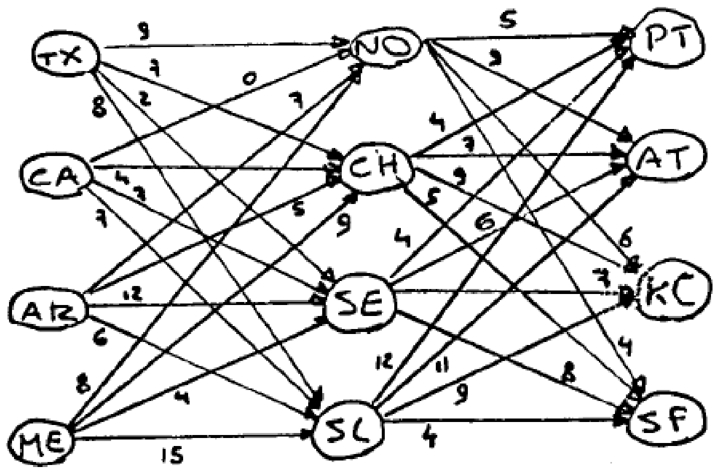
(b)



(c)


Source
TX
CA
AK
ME
NO
CH
SE
SL
PT
AT
KC
SF
Sink
[28]
[24]
[28]
[36]
[11]
[15]
[5]
[7]
[29]
[33]
[30]
[24]

(d)

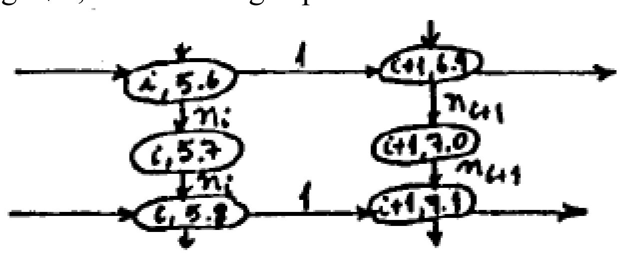

TX
CA
AK
ME
NO
CH
SE
SL
PT
AT
KC
SF
15
5

(e)

From	To	Ship		Capacity		Nodes	Net Flow		Supply/Demand
Texas	New Orleans	11	<=	11		Texas	28		
Texas	Charleston	7	<=	7		California	24		
Texas	Seattle	2	<=	2		Alaska	28		
Texas	St. Louis	8	<=	8		Middle East	28		
California	New Orleans	5	<=	5		New Orleans	0	=	0
California	Charleston	4	<=	4		Charleston	0	=	0
California	Seattle	8	<=	8		Seattle	0	=	0
California	St. Louis	7	<=	7		St. Louis	0	=	0
Alaska	New Orleans	7	<=	7		Pittsburgh	-26		
Alaska	Charleston	3	<=	3		Atlanta	-33		
Alaska	Seattle	12	<=	12		Kansas City	-25		
Alaska	St. Louis	6	<=	6		San Francisco	-24		
Middle East	New Orleans	1	<=	8					
Middle East	Charleston	9	<=	9					
Middle East	Seattle	3	<=	4					
Middle East	St. Louis	15	<=	15					
New Orleans	Pittsburgh	5	<=	5					
New Orleans	Atlanta	9	<=	9					
New Orleans	Kansas City	6	<=	6					
New Orleans	San Francisco	4	<=	4					
Charleston	Pittsburgh	8	<=	8					
Charleston	Atlanta	7	<=	7					
Charleston	Kansas City	3	<=	9					
Charleston	San Francisco	5	<=	5					
Seattle	Pittsburgh	4	<=	4					
Seattle	Atlanta	6	<=	6					
Seattle	Kansas City	7	<=	7					
Seattle	San Francisco	8	<=	8					
St. Louis	Pittsburgh	9	<=	12					
St. Louis	Atlanta	11	<=	11					
St. Louis	Kansas City	9	<=	9					
St. Louis	San Francisco	7	<=	7					
	Maximum Flow	108							

10.5-5.

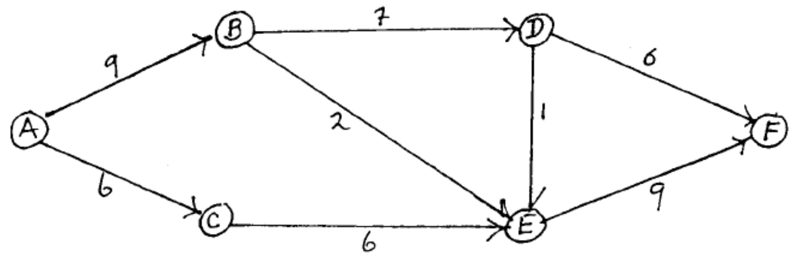
For convenience, call the Faireparc station siding 0 and the Portstown station siding $s+1$. Let node (i, j) represent siding i at time j for $i = 0, 1, \ldots, s, s+1$ and $j = 0.0, 0.1, 0.2, \ldots, 23.9$. Node $(0, 0)$ is the source and node $(s+1, 23.9)$ is the sink. Arcs with unit capacity exist between nodes (i, j) and $(i+1, j+t_i)$ if and only if a freight train leaving siding i at time j could not be overtaken by a scheduled passenger train before it reached siding $i+1$. Arcs with capacity n_i exist between nodes (i, j) and $(i, j+1)$ for $j = 0.0, 0.1, 0.2, \ldots, 23.8$. There are no other arcs. For example, if $t_i = 1.3$ and a scheduled passenger train could overtake a freight train leaving siding i at time 5.7 before it reached siding $i+1$, the following is part of the network:



The maximum flow problem in this case maximizes the number of sent freight trains.

10.5-6.

(a)



(b)

From	To	Ship		Capacity		Nodes	Net Flow		Supply/Demand
A	B	8	<=	9		A	15		
A	C	7	<=	7		B	0	=	0
B	D	7	<=	7		C	0	=	0
B	E	1	<=	2		D	0	=	0
C	D	1	<=	4		E	0	=	0
C	E	6	<=	6		F	-15		
D	E	2	<=	3					
D	F	6	<=	6					
E	F	9	<=	9					
Maximum Flow		15							

10.5.7

Answers will vary.

10.5.8

Answers will vary.

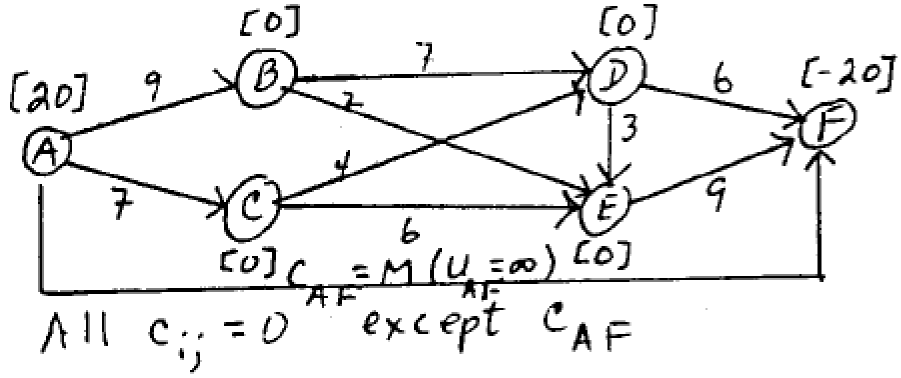
10.6-1.

In this study, flight delay and cancellation problems faced by United Airlines (UA) are modeled as minimum-cost-flow network models. The overall objective is to minimize a weighted sum of various measures related to delay. These include the total number of delay minutes for every passenger, the number of passengers affected by delays and the number of aircraft swaps. Nodes represent "arriving and departing aircraft, spare aircraft, and recovered aircraft" on a two-dimensional network, with time and airport being the two dimensions. Arcs represent "scheduled flights, connections, and aircraft substitutions" [p. 56]. Costs include the revenue loss, the costs from swapping aircraft and from delaying aircraft.

The delay problem is solved for each airport separately as a minimum-cost-flow network problem. The flow on each arc can be at most one. The solution is a set of arcs starting at a supply node and ending at a demand node, which determines flight delays due to shortage in aircraft. The cancellation model is a minimum-cost-flow network problem on the entire network. Again, the flow on each arc cannot exceed one. The solution determines which flight is canceled and what flight its aircraft is assigned to.

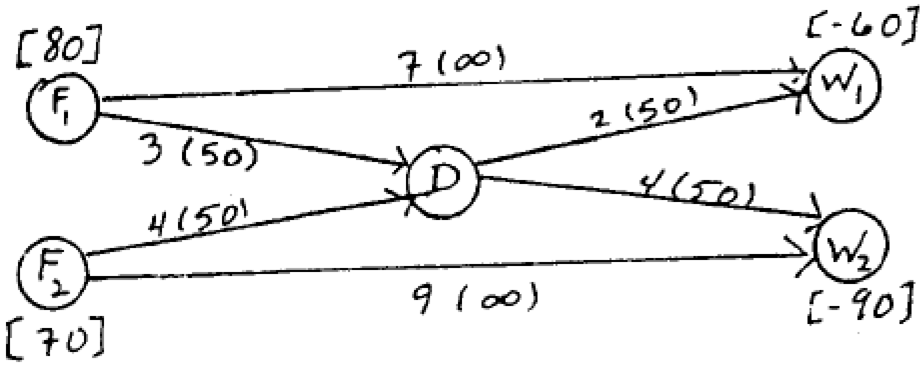
This study has saved UA over half a billion dollars in delay costs alone in less than a year. Many potential delays were prevented and hence the number of flight delays was reduced by 50%. Customer inconveniences due to delays and cancellations were reduced. Additionally, developing an efficient way of addressing these problems helped UA respond to changes in the conditions quickly.

10.6-2.



10.6-3.

(a)

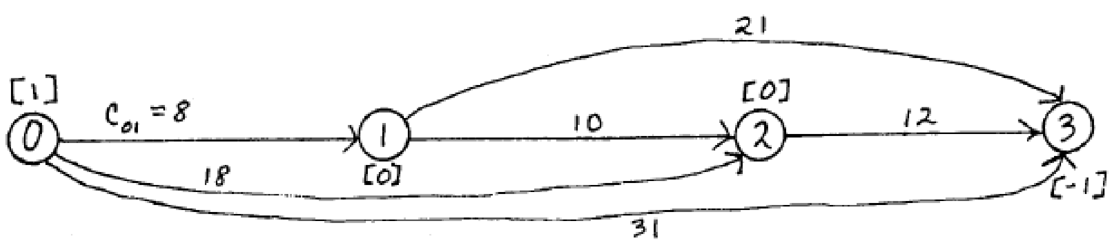


(b) minimize $7x_{F_1W_1} + 3x_{F_1D} + 2x_{DW_1} + 4x_{F_2D} + 4x_{DW_2} + 9x_{F_2W_2}$

subject to

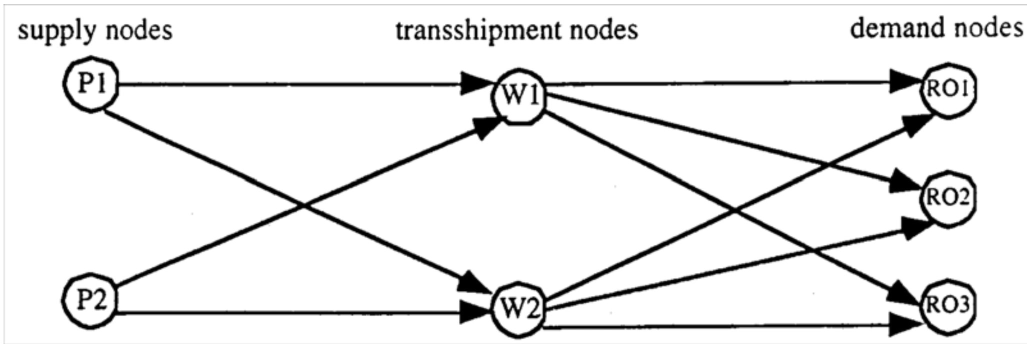
$$x_{F_1W_1} + x_{F_1D} = 80$$
$$x_{F_2D} + x_{F_2W_2} = 70$$
$$x_{F_1W_1} + x_{DW_1} = 60$$
$$x_{DW_2} + x_{F_2W_2} = 90$$
$$x_{F_1D} - x_{DW_1} + x_{F_2D} - x_{DW_2} = 0$$
$$0 \le x_{F_1D}, x_{DW_1}, x_{F_2D}, x_{DW_2} \le 50$$

10.6-4.

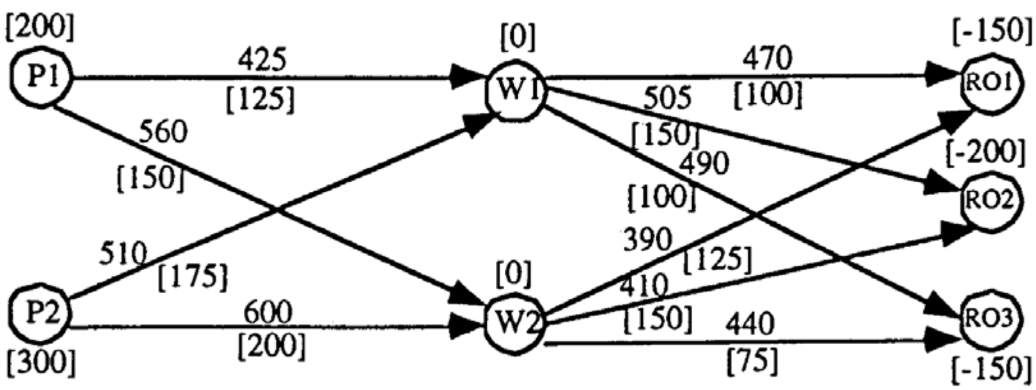


10.6-5.

(a)



(b)

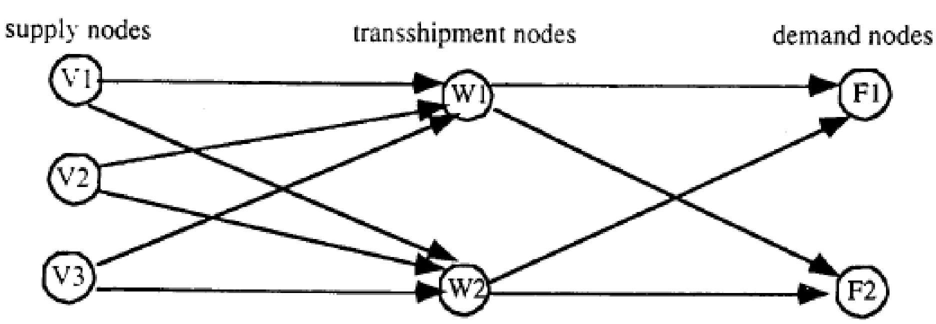


(c) Total cost: $488,125

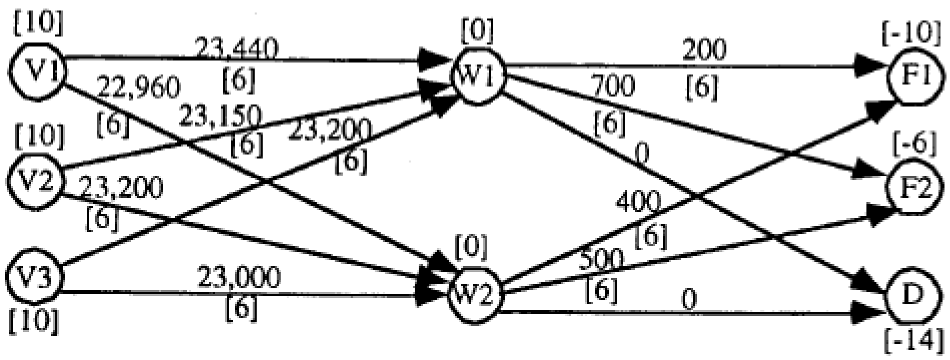
From	To	Ship		Capacity	Unit Cost		Nodes	Net Flow		Supply/Demand
P1	WH1	125	<=	125	$425		P1	200	=	200
P1	WH2	75	<=	150	$560		P2	300	=	300
P2	WH1	125	<=	175	$510		WH1	0	=	0
P2	WH2	175	<=	200	$600		WH2	0	=	0
WH1	RO1	100	<=	100	$470		RO1	-150	=	-150
WH1	RO2	50	<=	150	$505		RO2	-200	=	-200
WH1	RO3	100	<=	100	$490		RO3	-150	=	-150
WH2	RO1	50	<=	125	$390					
WH2	RO2	150	<=	150	$410					
WH2	RO3	50	<=	75	$440					
	Total Cost	$488,125								

10.6-6.

(a)



(b)

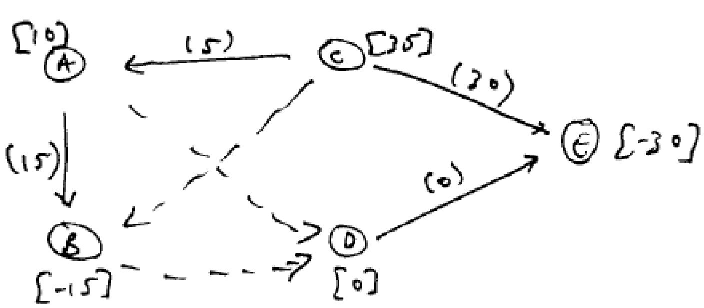


(c)

		Fixed Shipping Charge	Per Mile Charge	Miles to WH1	Miles to WH2	Total Cost to WH1	Total Cost to WH2			
Vendor	Price									
1	$22,500	$300	$0.40	1600	400	$23,440	$22,960			
2	$22,700	$200	$0.50	500	600	$23,150	$23,200			
3	$22,300	$500	$0.20	2000	1000	$23,200	$23,000			
From	To	Ship		Capacity	Unit Cost		Nodes	Net Flow		Supply/Demand
V1	WH1	0	<=	6	$23,440		V1	10	=	10
V1	WH2	6	<=	6	$22,960		V2	10	=	10
V2	WH1	6	<=	6	$23,150		V3	10	=	10
V2	WH2	0	<=	6	$23,200		WH1	0	=	0
V3	WH1	0	<=	6	$23,200		WH2	0	=	0
V3	WH2	4	<=	6	$23,000		F1	-10	=	-10
WH1	F1	6	<=	6	$200		F2	-6	=	-6
WH1	F2	0	<=	6	$700		D	-14	=	-14
WH2	F1	4	<=	6	$400					
WH2	F2	6	<=	6	$500					
V1	D	4			$0					
V2	D	4			$0					
V3	D	6			$0					
	Total Cost	$374,460								

10.7-1.

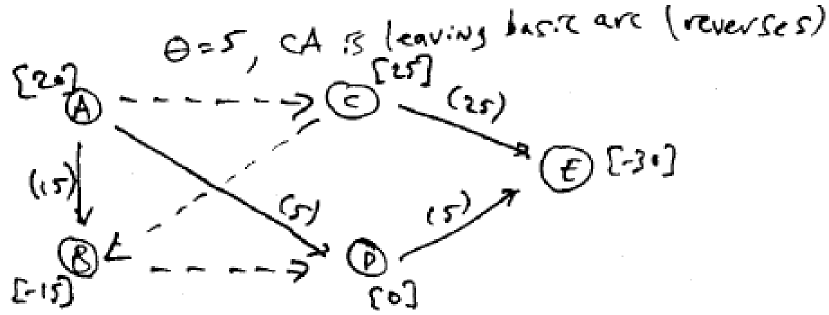
(a)



(b) Compute Δ for nonbasic arcs:

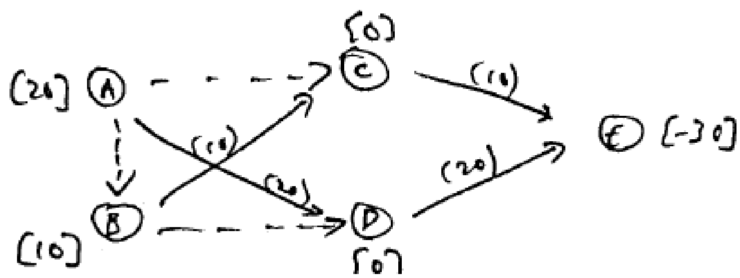
$$\Delta_{\text{BD}} = 5 + 4 - 3 + (-6) + 2 = 2$$
$$\Delta_{\text{AD}} = 5 + 4 - 3 + (-6) = 0$$
$$\Delta_{\text{CB}} = (-3) - 2 - (-6) = 1$$

All of them are nonnegative, so this solution is optimal. Since $\Delta_{\text{AD}} = 0$, multiple optima exist. Network simplex:



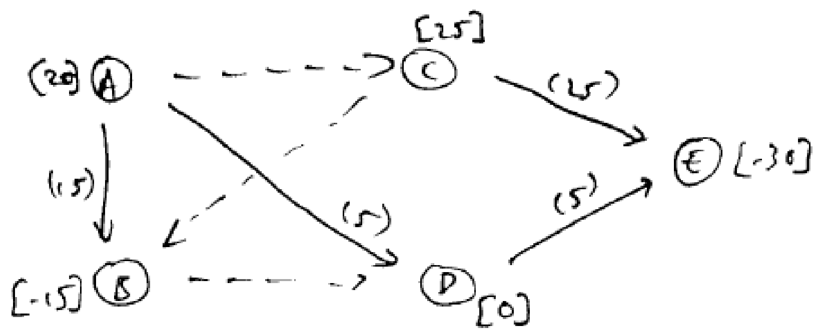
Optimal nonbasic solutions have $x_{\text{AB}} = 15$, $x_{\text{AC}} = \theta$, $x_{\text{AD}} = 5 - \theta$, $x_{\text{CE}} = 25 + \theta$, and $x_{\text{DE}} = 5 - \theta$, where $0 \leq \theta \leq 5$ and C $\rightarrow$ B and B $\rightarrow$ D are nonbasic arcs.

(c) Start with:



Network simplex: $\Delta_{\text{AC}} = 6 + 3 - 4 - 5 = 0$
$\Delta_{\text{AB}} = 2 + 3 + 3 - 4 - 5 = -1 < 0 \leftarrow$ entering arc
$\Delta_{\text{BD}} = 5 + 4 - 3 - 3 = 3$

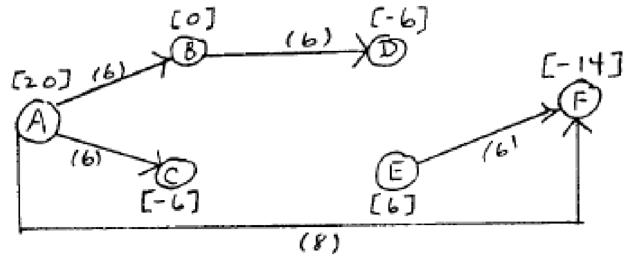
$\theta = 15$ and BC is leaving arc (reverses). The next BF solution is:



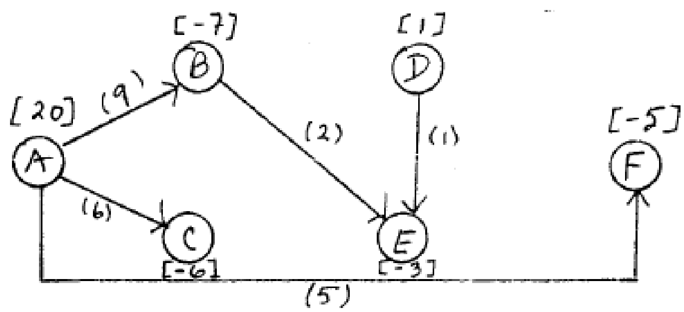
From (b), we recognize this solution as optimal.

10.7-2.

(a)



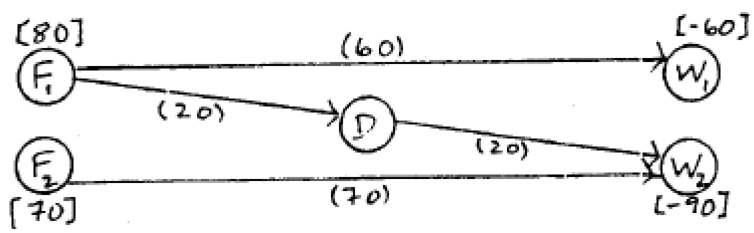
(b) The final feasible spanning tree is:



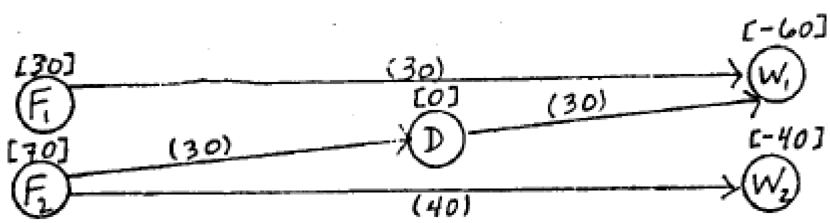
The flow to which it corresponds is the same as in Prob. 10.5-6.

10.7-3.

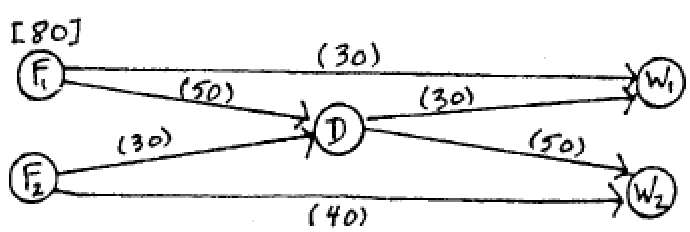
(a) There are no reverse arcs in this solution.



(b) The optimal BF spanning tree is:



which corresponds to a real flow of:

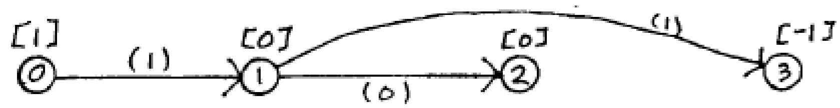


with cost 1, 100.

10.7-4.

Initial BF spanning tree:

Optimal BF spanning tree:

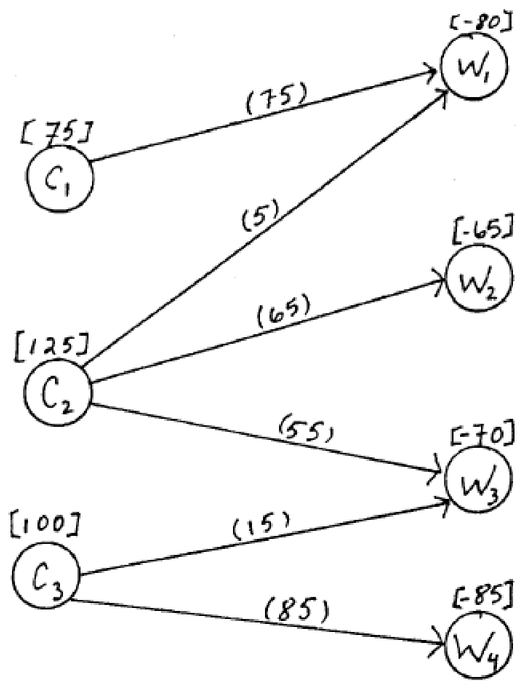


which has a real flow of:

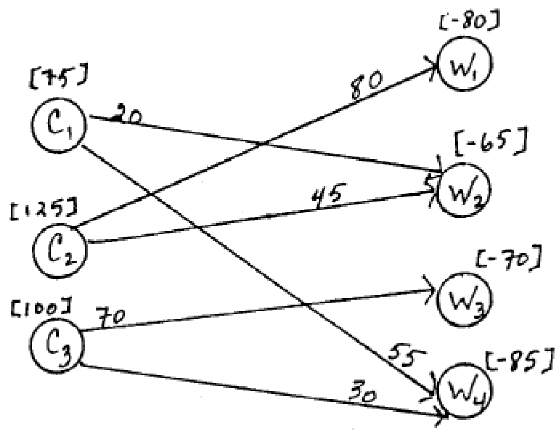
with cost 29.

10.7-5.

Initial BF spanning tree:



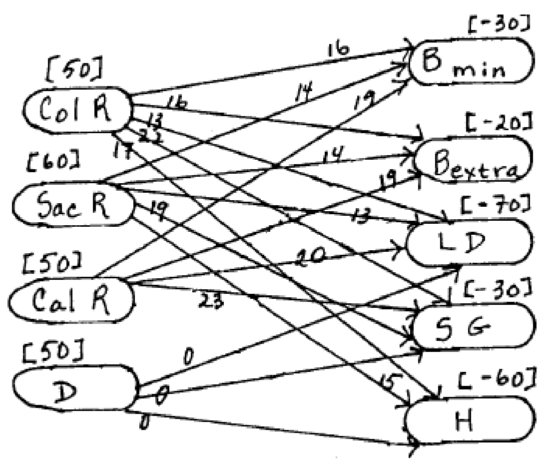
Optimal BF spanning tree:



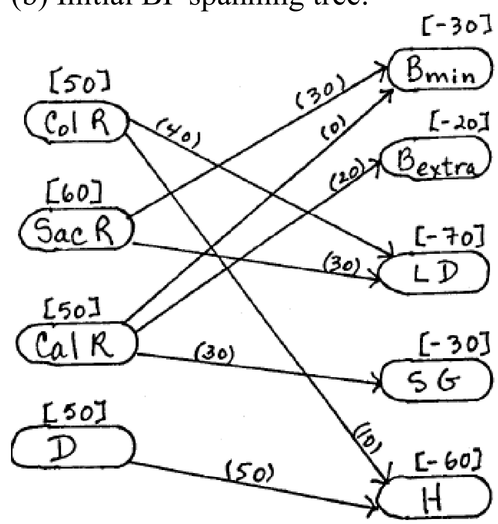
which corresponds to the optimal solution given in Sec. 8.1.

10.7-6.

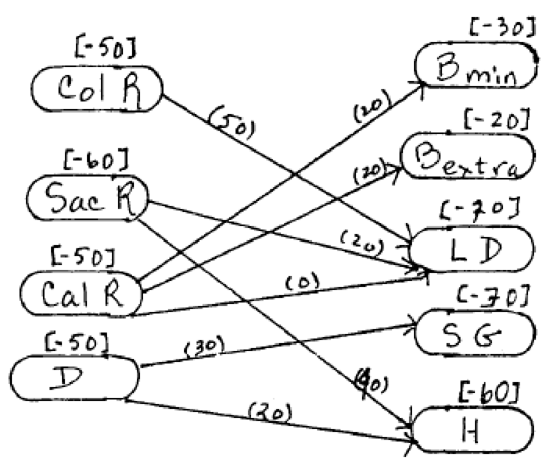
(a)



(b) Initial BF spanning tree:

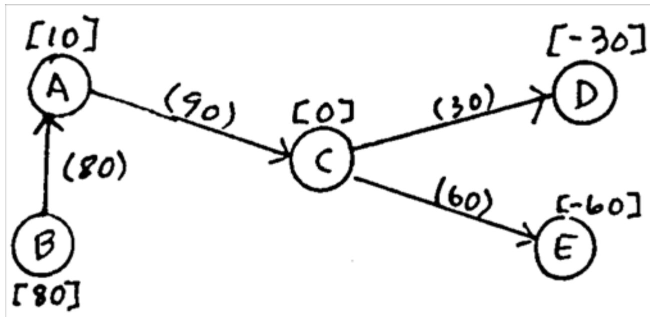


(c) Optimal BF spanning tree:

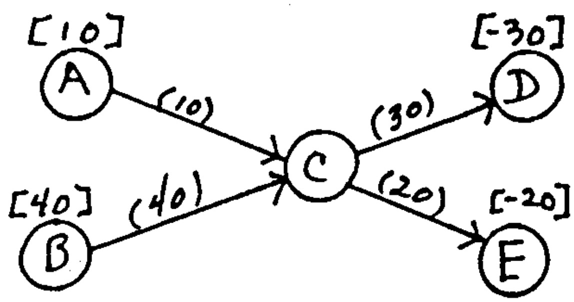


The sequence of basic feasible solutions is identical with the transportation simplex method.

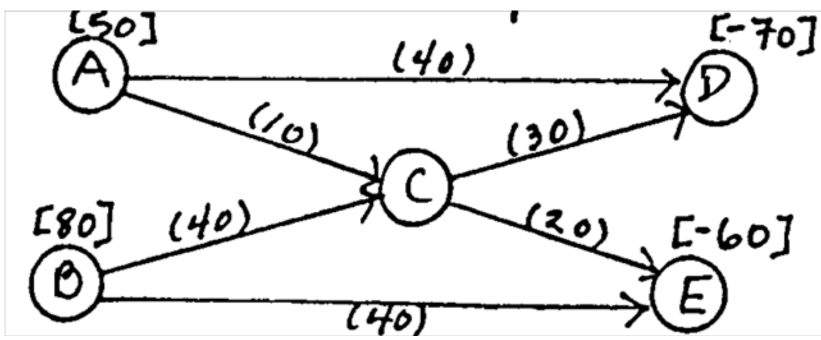
10.7-7.



Optimal BF spanning tree:



which correspond to the real flow of:



with a total cost of 750.

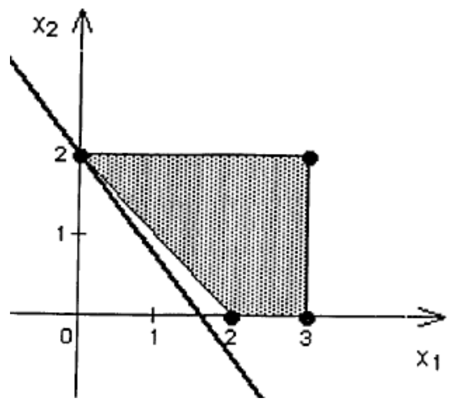
10.8-1.

		Length of Path	
Activity to Crash	Crash Cost	$A-C$	$B-D$
		14	16
B	$\$5,000$	14	15
B	$\$5,000$	14	15
D	$\$6,000$	14	14
C	$\$4,000$	13	14
D	$\$6,000$	13	13
C	$\$4,000$	12	13
D	$\$6,000$	12	12

10.8-2.

(a) Let x_A and x_C be the reduction in A and C respectively, due to crashing.

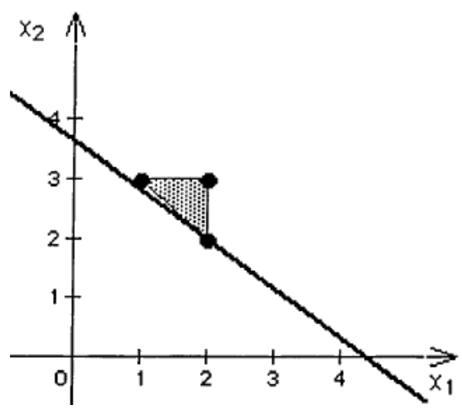
$$\begin{array}{ll} \text{minimize} & C = 5000x_A + 4000x_C \\ \text{subject to} & x_A \leq 3 \\ & x_C \leq 2 \\ & x_A + x_C \geq 2 \\ \text{and} & x_A, x_C \geq 0 \end{array}$$



Optimal Solution: $(x_A, x_C) = (0, 2)$ and $C^* = 8,000$.

(b) Let x_B and x_D be the reduction in B and D respectively, due to crashing.

$$\begin{array}{ll} \text{minimize} & C = 5000x_B + 6000x_D \\ \text{subject to} & x_B \leq 2 \\ & x_D \leq 3 \\ & x_B + x_D \geq 4 \\ \text{and} & x_B, x_D \geq 0 \end{array}$$



Optimal Solution: $(x_B, x_D) = (2, 2)$ and $C^* = 22,000$.

(c) Let x_A, x_B, x_C, and x_D be the reduction in the duration of A, B, C, and D respectively, due to crashing.

$$\begin{array}{ll}
\text{minimize} & C = 5000x_A + 5000x_B + 4000x_C + 6000x_D \\
\text{subject to} & x_A \leq 3 \\
& x_B \leq 2 \\
& x_C \leq 2 \\
& x_D \leq 3 \\
& x_A + x_C \geq 2 \\
& x_B + x_D \geq 4 \\
\text{and} & x_A, x_B, x_C, x_D \geq 0
\end{array}$$

Optimal Solution: $(x_A, x_B, x_C, x_D) = (0, 2, 2, 2)$ and $C^* = 30,000$.

(d) Let x_j be the reduction in the duration of activity j due to crashing for $j = A, B, C, D$. Also let y_j denote the start time of activity j for $j = C, D$ and y_{FINISH} the project duration.

$$\begin{array}{ll}
\text{minimize} & C = 5000x_A + 5000x_B + 4000x_C + 6000x_D \\
\text{subject to} & x_A \leq 3, x_B \leq 2, x_C \leq 2, x_D \leq 3 \\
& y_C \geq 0 + 8 - x_A \\
& y_D \geq 0 + 9 - x_B \\
& y_{\text{FINISH}} \geq y_C + 6 - x_C \\
& y_{\text{FINISH}} \geq y_D + 7 - x_D \\
& y_{\text{FINISH}} \leq 12 \\
\text{and} & x_A, x_B, x_C, x_D, y_C, y_D, y_{\text{FINISH}} \geq 0
\end{array}$$

(e)

Activity	Normal Time (months)	Crash Time (months)	Normal Cost	Crash Cost	Maximum Time Reduction (months)	Crash Cost per Month Saved	Start Time	Time Reduction	Finish Time
A	8	5	$25,000	$40,000	3	$5,000	0	0	8
B	9	7	$20,000	$30,000	2	$5,000	0	2	7
C	6	4	$16,000	$24,000	2	$4,000	8	2	12
D	7	4	$27,000	$45,000	3	$6,000	7	2	12
									Max Time
				Project Completion Time (months)			12	<=	12
						Total Cost	$118,000		

(f) The solution found using LINGO agrees with the solution in (e), i.e., it is optimal to reduce the duration of activities B, C, and D by two months. Then the entire project takes 12 months and costs $25 + 30 + 24 + (27 + 12) = 118$ thousand dollars.

```
Variable           Value        Reduced Cost
      XA        0.000000            0.000000
      XB        2.000000            0.000000
      XC        2.000000            0.000000
      XD        2.000000            0.000000

     Row    Slack or Surplus      Dual Price
       1        30000.00           -1.000000
       2        3.000000            0.000000
       3        0.000000            1000.000
       4        0.000000            1000.000
       5        1.000000            0.000000
       6        0.000000           -5000.000
       7        0.000000           -6000.000
```

(g) Deadline of 11 months

Activity	Normal Time (months)	Crash Time (months)	Normal Cost	Crash Cost	Maximum Time Reduction (months)	Crash Cost per Month Saved	Start Time	Time Reduction	Finish Time
A	8	5	$25,000	$40,000	3	$5,000	0	1	7
B	9	7	$20,000	$30,000	2	$5,000	0	2	7
C	6	4	$16,000	$24,000	2	$4,000	7	2	11
D	7	4	$27,000	$45,000	3	$6,000	7	3	11
									Max Time
					Project Completion Time (months)		11	<=	11
						Total Cost	$129,000		

Deadline of 13 months

Activity	Normal Time (months)	Crash Time (months)	Normal Cost	Crash Cost	Maximum Time Reduction (months)	Crash Cost per Month Saved	Start Time	Time Reduction	Finish Time
A	8	5	$25,000	$40,000	3	$5,000	0	0	8
B	9	7	$20,000	$30,000	2	$5,000	0	2	7
C	6	4	$16,000	$24,000	2	$4,000	8	1	13
D	7	4	$27,000	$45,000	3	$6,000	7	1	13
									Max Time
					Project Completion Time (months)		13	<=	13
						Total Cost	$108,000		

10.8-3.

(a) \$7,834 is saved by the new plan given below.

		Length of Path		
Activity to Crash	Crash Cost	$A - B - D$	$A - B - E$	$A - C - E$
		10	11	12
C	$\$1,333$	10	11	11
E	$\$2,500$	10	10	10
D & E	$\$4,000$	9	9	9
B & C	$\$4,333$	8	8	8

Activity	Duration	Cost
A	3 weeks	$\$54,000$
B	3 weeks	$\$65,000$
C	3 weeks	$\$58,666$
D	2 weeks	$\$41,500$
E	2 weeks	$\$80,000$

(b)

Activity	Normal Time (weeks)	Crash Time (weeks)	Normal Cost	Crash Cost	Maximum Time Reduction (weeks)	Crash Cost per Week Saved	Start Time	Time Reduction	Finish Time
A	3	2	\$54,000	\$60,000	1	\$6,000	0	0	3
B	4	3	\$62,000	\$65,000	1	\$3,000	4	0	8
C	5	2	\$66,000	\$70,000	3	\$1,333	3	0	8
D	3	1	\$40,000	\$43,000	2	\$1,500	9	0	12
E	4	2	\$75,000	\$80,000	2	\$2,500	8	0	12
									Max Time
						Project Completion Time (weeks)	12	<=	12
						Total Cost	\$297,000		

Activity	Normal Time (weeks)	Crash Time (weeks)	Normal Cost	Crash Cost	Maximum Time Reduction (weeks)	Crash Cost per Week Saved	Start Time	Time Reduction	Finish Time
A	3	2	\$54,000	\$60,000	1	\$6,000	0	0	3
B	4	3	\$62,000	\$65,000	1	\$3,000	3	0	7
C	5	2	\$66,000	\$70,000	3	\$1,333	3	1	7
D	3	1	\$40,000	\$43,000	2	\$1,500	8	0	11
E	4	2	\$75,000	\$80,000	2	\$2,500	7	0	11
									Max Time
						Project Completion Time (weeks)	11	<=	11
						Total Cost	\$298,333		

Activity	Normal Time (weeks)	Crash Time (weeks)	Normal Cost	Crash Cost	Maximum Time Reduction (weeks)	Crash Cost per Week Saved	Start Time	Time Reduction	Finish Time
A	3	2	$54,000	$60,000	1	$6,000	0	0	3
B	4	3	$62,000	$65,000	1	$3,000	3	0	7
C	5	2	$66,000	$70,000	3	$1,333	3	1	7
D	3	1	$40,000	$43,000	2	$1,500	7	0	10
E	4	2	$75,000	$80,000	2	$2,500	7	1	10
									Max Time
				Project Completion Time (weeks)			10	<=	10
						Total Cost	$300,833		

Activity	Normal Time (weeks)	Crash Time (weeks)	Normal Cost	Crash Cost	Maximum Time Reduction (weeks)	Crash Cost per Week Saved	Start Time	Time Reduction	Finish Time
A	3	2	$54,000	$60,000	1	$6,000	0	0	3
B	4	3	$62,000	$65,000	1	$3,000	3	0	7
C	5	2	$66,000	$70,000	3	$1,333	3	1	7
D	3	1	$40,000	$43,000	2	$1,500	7	1	9
E	4	2	$75,000	$80,000	2	$2,500	7	2	9
									Max Time
				Project Completion Time (weeks)			9	<=	9
						Total Cost	$304,833		

Activity	Normal Time (weeks)	Crash Time (weeks)	Normal Cost	Crash Cost	Maximum Time Reduction (weeks)	Crash Cost per Week Saved	Start Time	Time Reduction	Finish Time
A	3	2	$54,000	$60,000	1	$6,000	0	0	3
B	4	3	$62,000	$65,000	1	$3,000	3	1	6
C	5	2	$66,000	$70,000	3	$1,333	3	2	6
D	3	1	$40,000	$43,000	2	$1,500	6	1	8
E	4	2	$75,000	$80,000	2	$2,500	6	2	8
									Max Time
				Project Completion Time (weeks)			8	<=	8
						Total Cost	$309,167		

Activity	Normal Time (weeks)	Crash Time (weeks)	Normal Cost	Crash Cost	Maximum Time Reduction (weeks)	Crash Cost per Week Saved	Start Time	Time Reduction	Finish Time
A	3	2	$54,000	$60,000	1	$6,000	0	1	2
B	4	3	$62,000	$65,000	1	$3,000	2	1	5
C	5	2	$66,000	$70,000	3	$1,333	2	2	5
D	3	1	$40,000	$43,000	2	$1,500	5	1	7
E	4	2	$75,000	$80,000	2	$2,500	5	2	7
									Max Time
				Project Completion Time (weeks)			7	<=	7
						Total Cost	$315,167		

Crash to 8 weeks.

10.8-4.

(a) Let x_j be the reduction in the duration of activity j and y_j be the start time of activity j.

minimize $\quad C = 5x_A + 10x_B + 4x_C + 6x_D + 8x_E + 6x_F + 5x_G + 7x_H$

subject to
$$0 \le x_A \le 2 \quad 0 \le x_B \le 1 \quad 0 \le x_C \le 2 \quad 0 \le x_D \le 3$$
$$0 \le x_E \le 1 \quad 0 \le x_F \le 3 \quad 0 \le x_G \le 4 \quad 0 \le x_H \le 2$$
$$y_A + 5 - x_A \le y_C \qquad y_A + 5 - x_A \le y_D$$
$$y_B + 3 - x_B \le y_E \qquad y_B + 3 - x_B \le y_F$$
$$y_C + 4 - x_C \le y_G \qquad y_D + 6 - x_D \le y_H$$
$$y_E + 5 - x_E \le y_G \qquad y_F + 7 - x_F \le y_H$$
$$y_G + 9 - x_G \le y_{\text{FINISH}} \qquad y_H + 8 - x_H \le y_{\text{FINISH}}$$
$$0 \le y_{\text{FINISH}} \le 15$$
$$y_j \ge 0$$

(b) Finish Time: 15 weeks, total cost: \$217 million.

Activity	Normal Time (weeks)	Crash Time (weeks)	Normal Cost ($million)	Crash Cost ($million)	Maximum Time Reduction (weeks)	Crash Cost per Week Saved ($million)	Start Time	Time Reduction	Finish Time
A	5	3	20	30	2	5	0	2	3
B	3	2	10	20	1	10	0	1	2
C	4	2	16	24	2	4	3	0	7
D	6	3	25	43	3	6	3	0	9
E	5	4	22	30	1	8	2	0	7
F	7	4	30	48	3	6	2	0	9
G	9	5	25	45	4	5	7	1	15
H	8	6	30	44	2	7	9	2	15
									Max Time
					Project Completion Time (weeks)		15	<=	15
					Total Cost ($million)		217		

10.8-5.

(a) Let x_j be the reduction in the duration of activity j and y_j be the start time of activity j.

minimize $\quad C = 5x_A + 7x_B + 8x_C + 4x_D + 5x_E + 6x_F + 3x_G + 4x_H + 9x_I + 2x_J$

subject to
$$0 \le x_A \le 4 \quad 0 \le x_B \le 3 \quad 0 \le x_C \le 5 \quad 0 \le x_D \le 3 \quad 0 \le x_E \le 5$$
$$0 \le x_F \le 7 \quad 0 \le x_G \le 2 \quad 0 \le x_H \le 3 \quad 0 \le x_I \le 4 \quad 0 \le x_J \le 2$$
$$y_A + 32 - x_A \le y_C \qquad y_B + 28 - x_B \le y_D$$
$$y_B + 28 - x_B \le y_E \qquad y_B + 28 - x_B \le y_F$$
$$y_C + 36 - x_C \le y_J \qquad y_D + 16 - x_D \le y_G$$
$$y_E + 32 - x_E \le y_H \qquad y_E + 32 - x_E \le y_I$$
$$y_F + 54 - x_F \le y_J \qquad y_G + 17 - x_G \le y_H$$
$$y_G + 17 - x_G \le y_I \qquad y_H + 20 - x_H \le y_{\text{FINISH}}$$
$$y_I + 34 - x_I \le y_{\text{FINISH}} \qquad y_J + 18 - x_J \le y_{\text{FINISH}}$$
$$0 \le y_{\text{FINISH}} \le 92$$
$$y_j \ge 0$$

(b) Finish Time: 92 weeks, total crashing cost: $43 million, total cost: $1.388 billion.

Activity	Normal Time (weeks)	Crash Time (weeks)	Normal Cost ($million)	Crash Cost ($million)	Maximum Time Reduction (weeks)	Crash Cost per Week Saved ($million)	Start Time	Time Reduction	Finish Time
A	32	28	160	180	4	5	0	0	32
B	28	25	125	146	3	7	0	3	25
C	36	31	170	210	5	8	32	0	68
D	16	13	60	72	3	4	25	0	41
E	32	27	135	160	5	5	26	0	58
F	54	47	215	257	7	6	25	3	76
G	17	15	90	96	2	3	41	0	58
H	20	17	120	132	3	4	58	0	78
I	34	30	190	226	4	9	58	0	92
J	18	16	80	84	2	2	76	2	92
									Max Time
				Project Completion Time (weeks)			92	<=	92
				Total Cost ($million)			1,388		

10.9-1.

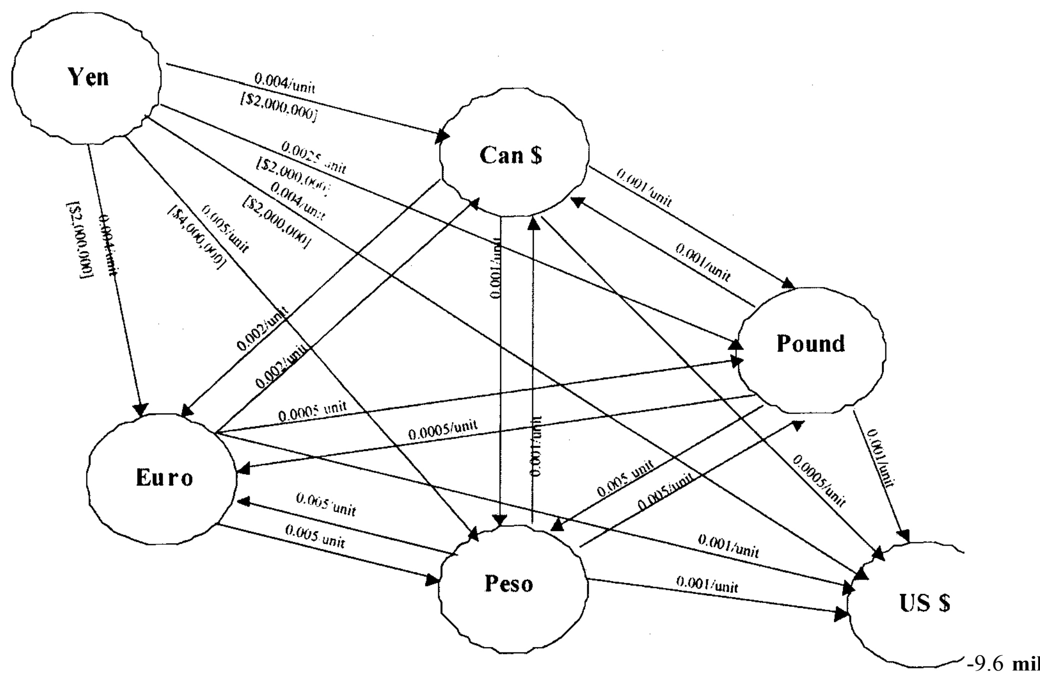
Answers will vary.

10.9-2.

Answers will vary.

Case 10.1

a) There are three supply nodes – the Yen node, the Rupiah node, and the Ringgit node. There is one demand node – the US$ node. Below, we draw the network originating from only the Yen supply node to illustrate the overall design of the network. In this network, we exclude both the Rupiah and Ringgit nodes for simplicity.



b) Since all transaction limits are given in the equivalent of $1000 we define the flow variables as the amount in thousands of dollars that Jake converts from one currency into another one. His total holdings in Yen, Rupiah, and Ringgit are equivalent to $9.6 million, $1.68 million, and $5.6 million, respectively (as calculated in cells I16:K18 in the spreadsheet). So, the supplies at the supply nodes Yen, Rupiah, and Ringgit are -$9.6 million, -$1.68 million, and -$5.6 million, respectively. The demand at the only demand node US$ equals $16.88 million (the sum of the outflows from the source nodes). The transaction limits are capacity constraints for all arcs leaving from the nodes Yen, Rupiah, and Ringgit. The unit cost for every arc is given by the transaction cost for the currency conversion.

	A	B	C	D	E	F	G	H	I	J	K
1					Transaction						
2			Convert		Limit	Unit			Net Flow		Supply/Demand
3	From	To	($thousands)		($thousands)	Cost		Nodes	($thousands)		($thousands)
4	Yen	Rupiah	0	<=	5,000	0.50%		Yen	9,600	=	9,600
5	Yen	Ringit	0	<=	5,000	0.50%		Rupiah	1,680	=	1,680
6	Yen	US$	2,000	<=	2,000	0.40%		Ringit	5,600	=	5,600
7	Yen	Can$	2,000	<=	2,000	0.40%		Can$	0	=	0
8	Yen	Euro	2,000	<=	2,000	0.40%		Euro	0	=	0
9	Yen	Pound	2,000	<=	2,000	0.25%		Pound	0	=	0
10	Yen	Peso	1,600	<=	4,000	0.50%		Peso	0	=	0
11	Rupiah	Yen	0	<=	5,000	0.50%		US$	-16,880	=	-16,880
12	Rupiah	Ringit	0	<=	2,000	0.70%					
13	Rupiah	US$	200	<=	200	0.50%			Starting		
14	Rupiah	Can$	200	<=	200	0.30%			Supply	Conversion	Starting
15	Rupiah	Euro	1,000	<=	1,000	0.30%			(thousands)	($ per)	($thousands)
16	Rupiah	Pound	280	<=	500	0.75%		Yen	1,200,000	0.008	9,600
17	Rupiah	Peso	0	<=	200	0.75%		Rupiah	10,500,000	0.00016	1,680
18	Ringit	Yen	0	<=	3,000	0.50%		Ringgit	28,000	0.2	5,600
19	Ringit	Rupiah	0	<=	4,500	0.70%					
20	Ringit	US$	1,100	<=	1,500	0.70%					
21	Ringit	Can$	0	<=	1,500	0.70%					
22	Ringit	Euro	2,500	<=	2,500	0.40%					
23	Ringit	Pound	1,000	<=	1,000	0.45%					
24	Ringit	Peso	1,000	<=	1,000	0.50%					
25	Can$	US$	2,200			0.05%					
26	Can$	Euro	0			0.20%					
27	Can$	Pound	0			0.10%					
28	Can$	Peso	0			0.10%					
29	Euro	US$	5,500			0.10%					
30	Euro	Can$	0			0.20%					
31	Euro	Pound	0			0.05%					
32	Euro	Peso	0			0.50%					
33	Pound	US$	3,280			0.10%					
34	Pound	Can$	0			0.10%					
35	Pound	Euro	0			0.05%					
36	Pound	Peso	0			0.50%					
37	Peso	US$	2,600			0.10%					
38	Peso	Can$	0			0.10%					
39	Peso	Euro	0			0.50%					
40	Peso	Pound	0			0.50%					
41											
42		Total Cost	83.38								

Range Name	Cells
Conversion	J16:J18
Convert	C4:C40
From	A4:A40
NetFlow	I4:I11
Nodes	H4:H11
StartingSupply	I16:I18
SupplyDemand	K4:K11
To	B4:B40
TotalCost	C42
TransactionLimit	E4:E24
UnitCost	F4:F40

	I	J	K
2	**Net Flow**		**Supply/Demand**
3	**($thousands)**		**($thousands)**
4	=SUMIF(From,Nodes,Convert)-SUMIF(To,Nodes,Convert)	=	=K16
5	=SUMIF(From,Nodes,Convert)-SUMIF(To,Nodes,Convert)	=	=K17
6	=SUMIF(From,Nodes,Convert)-SUMIF(To,Nodes,Convert)	=	=K18
7	=SUMIF(From,Nodes,Convert)-SUMIF(To,Nodes,Convert)	=	0
8	=SUMIF(From,Nodes,Convert)-SUMIF(To,Nodes,Convert)	=	0
9	=SUMIF(From,Nodes,Convert)-SUMIF(To,Nodes,Convert)	=	0
10	=SUMIF(From,Nodes,Convert)-SUMIF(To,Nodes,Convert)	=	0
11	=SUMIF(From,Nodes,Convert)-SUMIF(To,Nodes,Convert)	=	=-SUM(K16:K18)

	H	I	J	K
13		Starting		
14		Supply	Conversion	Starting
15		(thousands)	($ per)	($thousands)
16	Yen	1200000	0.008	=StartingSupply*Conversion
17	Rupiah	10500000	0.00016	=StartingSupply*Conversion
18	Ringgit	28000	0.2	=StartingSupply*Conversion

	B	C
42	Total Cost	=SUMPRODUCT(UnitCost,Convert)

Jake should convert the equivalent of $2 million from Yen to each US$, Can$, Euro, and Pound. He should convert $1.6 million from Yen to Peso. Moreover, he should convert the equivalent of $200,000 from Rupiah to each US$, Can$, and Peso, $1 million from Rupiah to Euro, and $80,000 from Rupiah to Pound. Furthermore, Jake should convert the equivalent of $1.1 million from Ringgit to US$, $2.5 million from Ringgit to Euro, and $1 million from Ringgit to each Pound and Peso. Finally, he should convert all the money he converted into Can$, Euro, Pound, and Peso directly into US$. Specifically, he needs to convert into US$ the equivalent of $2.2 million, $5.5 million, $3.08 million, and $2.8 million Can$, Euro, Pound, and Peso, respectively. Assuming Jake pays for the total transaction costs of $83,380 directly from his American bank accounts he will have $16,880,000 dollars to invest in the US.

c) We eliminate all capacity restrictions on the arcs.

	A	B	C	D	E	F	G	H	I	J
1										
2			**Convert**		**Unit**			**Net Flow**		**Supply/Demand**
3	**From**	**To**	**($thousands)**		**Cost**		**Nodes**	**($thousands)**		**($thousands)**
4	Yen	Rupiah	0		0.50%		Yen	9,600	=	9,600
5	Yen	Ringit	0		0.50%		Rupiah	1,680	=	1,680
6	Yen	US$	0		0.40%		Ringit	5,600	=	5,600
7	Yen	Can$	0		0.40%		Can$	0	=	0
8	Yen	Euro	0		0.40%		Euro	0	=	0
9	Yen	Pound	9,600		0.25%		Pound	0	=	0
10	Yen	Peso	0		0.50%		Peso	0	=	0
11	Rupiah	Yen	0		0.50%		US$	-16,880	=	-16,880
12	Rupiah	Ringit	0		0.70%					
13	Rupiah	US$	0		0.50%			Starting		
14	Rupiah	Can$	1,680		0.30%			Supply	Conversion	Starting
15	Rupiah	Euro	0		0.30%			(thousands)	($ per)	($thousands)
16	Rupiah	Pound	0		0.75%		Yen	1,200,000	0.008	9,600
17	Rupiah	Peso	0		0.75%		Rupiah	10,500,000	0.00016	1,680
18	Ringit	Yen	0		0.50%		Ringgit	28,000	0.2	5,600
19	Ringit	Rupiah	0		0.70%					
20	Ringit	US$	0		0.70%					
21	Ringit	Can$	0		0.70%					
22	Ringit	Euro	5,600		0.40%					
23	Ringit	Pound	0		0.45%					
24	Ringit	Peso	0		0.50%					
25	Can$	US$	1,680		0.05%					
26	Can$	Euro	0		0.20%					
27	Can$	Pound	0		0.10%					
28	Can$	Peso	0		0.10%					
29	Euro	US$	5,600		0.10%					
30	Euro	Can$	0		0.20%					
31	Euro	Pound	0		0.05%					
32	Euro	Peso	0		0.50%					
33	Pound	US$	9,600		0.10%					
34	Pound	Can$	0		0.10%					
35	Pound	Euro	0		0.05%					
36	Pound	Peso	0		0.50%					
37	Peso	US$	0		0.10%					
38	Peso	Can$	0		0.10%					
39	Peso	Euro	0		0.50%					
40	Peso	Pound	0		0.50%					
41										
42		Total Cost	67.48							

Jake should convert the entire holdings in Japan from Yen into Pounds and then into US$, the entire holdings in Indonesia from Rupiah into Can$ and then into US$, and the entire holdings in Malaysia from Ringgit into Euro and then into US$. Without the capacity limits the transaction costs are reduced to $67,480.

d) We multiply all unit cost for Rupiah by 6.

	A	B	C	D	E	F	G	H	I	J
1										
2			**Convert**		**Unit**			**Net Flow**		**Supply/Demand**
3	**From**	**To**	**($thousands)**		**Cost**		**Nodes**	**($thousands)**		**($thousands)**
4	Yen	Rupiah	0		0.50%		Yen	9,600	=	9,600
5	Yen	Ringit	0		0.50%		Rupiah	1,680	=	1,680
6	Yen	US$	0		0.40%		Ringit	5,600	=	5,600
7	Yen	Can$	0		0.40%		Can$	0	=	0
8	Yen	Euro	0		0.40%		Euro	0	=	0
9	Yen	Pound	9,600		0.25%		Pound	0	=	0
10	Yen	Peso	0		0.50%		Peso	0	=	0
11	Rupiah	Yen	0		3.00%		US$	-16,880	=	-16,880
12	Rupiah	Ringit	0		4.20%					
13	Rupiah	US$	0		3.00%			Starting		
14	Rupiah	Can$	1,680		1.80%			Supply	Conversion	Starting
15	Rupiah	Euro	0		1.80%			(thousands)	($ per)	($thousands)
16	Rupiah	Pound	0		4.50%		Yen	1,200,000	0.008	9,600
17	Rupiah	Peso	0		4.50%		Rupiah	10,500,000	0.00016	1,680
18	Ringit	Yen	0		0.50%		Ringgit	28,000	0.2	5,600
19	Ringit	Rupiah	0		0.70%					
20	Ringit	US$	0		0.70%					
21	Ringit	Can$	0		0.70%					
22	Ringit	Euro	5,600		0.40%					
23	Ringit	Pound	0		0.45%					
24	Ringit	Peso	0		0.50%					
25	Can$	US$	1,680		0.05%					
26	Can$	Euro	0		0.20%					
27	Can$	Pound	0		0.10%					
28	Can$	Peso	0		0.10%					
29	Euro	US$	5,600		0.10%					
30	Euro	Can$	0		0.20%					
31	Euro	Pound	0		0.05%					
32	Euro	Peso	0		0.50%					
33	Pound	US$	9,600		0.10%					
34	Pound	Can$	0		0.10%					
35	Pound	Euro	0		0.05%					
36	Pound	Peso	0		0.50%					
37	Peso	US$	0		0.10%					
38	Peso	Can$	0		0.10%					
39	Peso	Euro	0		0.50%					
40	Peso	Pound	0		0.50%					
41										
42		Total Cost	92.68							

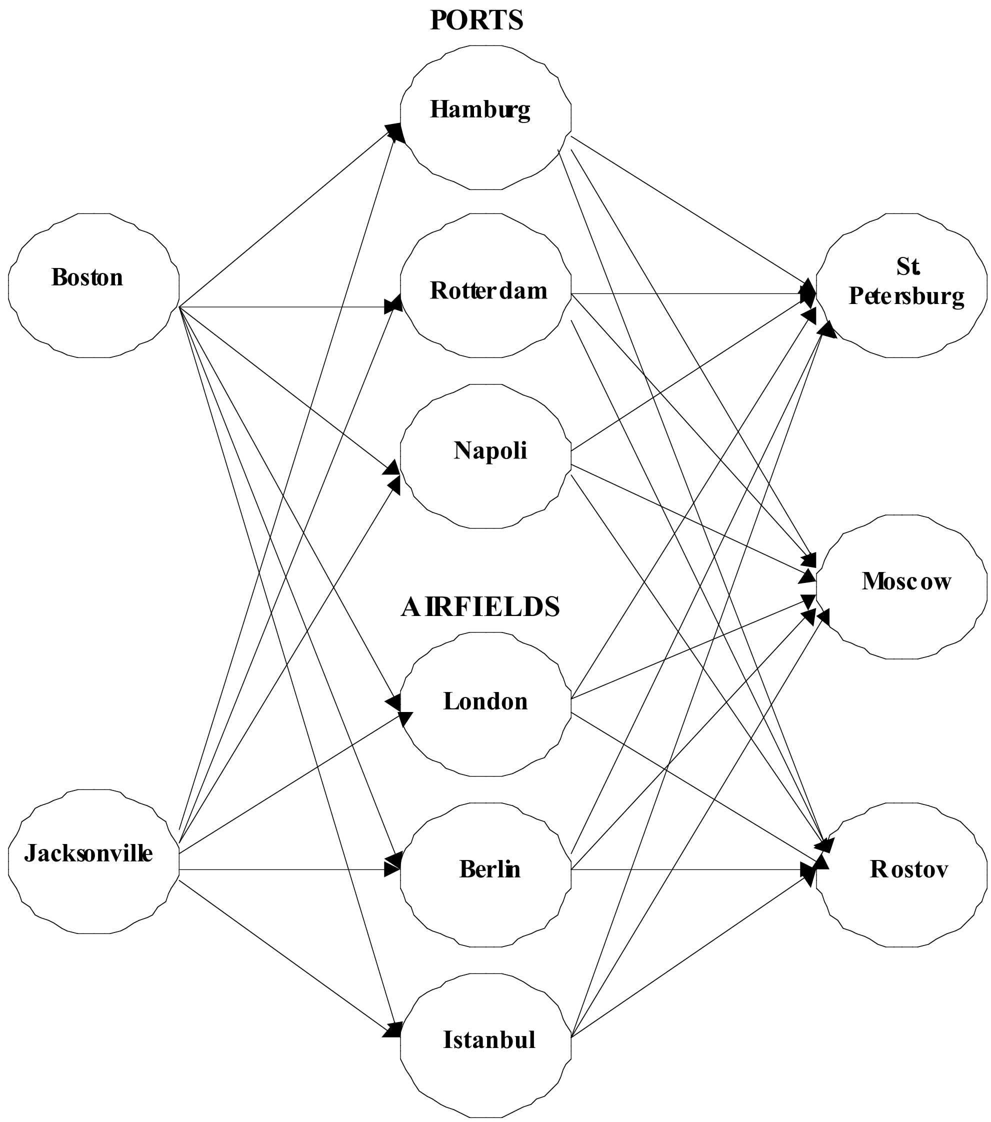
The optimal routing for the money doesn't change, but the total transaction costs are now increased to $92,680.

e) In the described crisis situation the currency exchange rates might change every minute. Jake should carefully check the exchange rates again when he performs the transactions.

The European economies might be more insulated from the Asian financial collapse than the US economy. To impress his boss Jake might want to explore other investment opportunities in safer European economies that provide higher rates of return than US bonds.

Case 10.2

a) The network showing the different routes troops and supplies may follow to reach the Russian Federation appears below.



b) The President is only concerned about how to most quickly move troops and supplies from the United States to the three strategic Russian cities. Obviously, the best way to achieve this goal is to find the fastest connection between the US and the three cities. We therefore need to find the shortest path between the US cities and each of the three Russian cities.

The President only cares about the time it takes to get the troops and supplies to Russia. It does not matter how great a distance the troops and supplies cover. Therefore we define the arc length between two nodes in the network to be the time it takes to travel between the respective cities. For example, the distance between Boston and London equals 6,200 km. The mode of transportation between the cities is a Starlifter traveling at a speed of 400 miles per hour * 1.609 km per mile = 643.6 km per hour. The time is takes to bring troops and supplies from Boston to London equals 6,200 km / 643.6 km per hour = 9.6333 hours. Using this approach we can compute the time of travel along all arcs in the network.

By simple inspection and common sense it is apparent that the fastest transportation involves using only airplanes. We therefore can restrict ourselves to only those arcs in the network where the mode of transportation is air travel. We can omit the three port cities and all arcs entering and leaving these nodes.

The following six spreadsheets find the shortest path between each US city (Boston and Jacksonville) and each Russian city (St. Petersburg, Moscow, and Rostov).

Boston to St. Petersburg:

	A	B	C	D	E	F	G	H	I	J	K
1	**From**	**To**	**On Route**		**Distance (km)**	**Time (hours)**		**Node**	**Net Flow**		**Supply/Demand**
2	Boston	London	1		6,200	9.63		Boston	1	=	1
3	Boston	Berlin	0		7,250	11.26		Jacksonville	0	=	0
4	Boston	Istanbul	0		8,300	12.90		London	0	=	0
5	Jacksonville	London	0		7,900	12.27		Berlin	0	=	0
6	Jacksonville	Berlin	0		9,200	14.29		Istanbul	0	=	0
7	Jacksonville	Istanbul	0		10,100	15.69		St. Petersburg	-1	=	-1
8	London	St. Petersburg	1		1,980	3.08		Moscow	0	=	0
9	London	Moscow	0		2,300	3.57		Rostov	0	=	0
10	London	Rostov	0		2,860	4.44					
11	Berlin	St. Petersburg	0		1,280	1.99					
12	Berlin	Moscow	0		1,600	2.49					
13	Berlin	Rostov	0		1,730	2.69					
14	Istanbul	St. Petersburg	0		2,040	3.17		Travel Speed (mph)	400		
15	Istanbul	Moscow	0		1,700	2.64		km/hr	1.609		
16	Istanbul	Rostov	0		990	1.54		Travel Speed (km/hr)	643.6		
17											
18		Total Time	12.71								

Boston to Moscow:

	A	B	C	D	E	F	G	H	I	J	K
1	From	To	On Route		Distance (km)	Time (hours)		Node	Net Flow		Supply/Demand
2	Boston	London	1		6,200	9.63		Boston	1	=	1
3	Boston	Berlin	0		7,250	11.26		Jacksonville	0	=	0
4	Boston	Istanbul	0		8,300	12.90		London	0	=	0
5	Jacksonville	London	0		7,900	12.27		Berlin	0	=	0
6	Jacksonville	Berlin	0		9,200	14.29		Istanbul	0	=	0
7	Jacksonville	Istanbul	0		10,100	15.69		St. Petersburg	0	=	0
8	London	St. Petersburg	0		1,980	3.08		Moscow	-1	=	-1
9	London	Moscow	1		2,300	3.57		Rostov	0	=	0
10	London	Rostov	0		2,860	4.44					
11	Berlin	St. Petersburg	0		1,280	1.99					
12	Berlin	Moscow	0		1,600	2.49					
13	Berlin	Rostov	0		1,730	2.69					
14	Istanbul	St. Petersburg	0		2,040	3.17		Travel Speed (mph)	400		
15	Istanbul	Moscow	0		1,700	2.64		km/hr	1.609		
16	Istanbul	Rostov	0		990	1.54		Travel Speed (km/hr)	643.6		
17											
18		Total Time	13.21								

Boston to Rostov:

	A	B	C	D	E	F	G	H	I	J	K
1	From	To	On Route		Distance (km)	Time (hours)		Node	Net Flow		Supply/Demand
2	Boston	London	0		6,200	9.63		Boston	1	=	1
3	Boston	Berlin	1		7,250	11.26		Jacksonville	0	=	0
4	Boston	Istanbul	0		8,300	12.90		London	0	=	0
5	Jacksonville	London	0		7,900	12.27		Berlin	0	=	0
6	Jacksonville	Berlin	0		9,200	14.29		Istanbul	0	=	0
7	Jacksonville	Istanbul	0		10,100	15.69		St. Petersburg	0	=	0
8	London	St. Petersburg	0		1,980	3.08		Moscow	0	=	0
9	London	Moscow	0		2,300	3.57		Rostov	-1	=	-1
10	London	Rostov	0		2,860	4.44					
11	Berlin	St. Petersburg	0		1,280	1.99					
12	Berlin	Moscow	0		1,600	2.49					
13	Berlin	Rostov	1		1,730	2.69					
14	Istanbul	St. Petersburg	0		2,040	3.17		Travel Speed (mph)	400		
15	Istanbul	Moscow	0		1,700	2.64		km/hr	1.609		
16	Istanbul	Rostov	0		990	1.54		Travel Speed (km/hr)	643.6		
17											
18		Total Time	13.95								

Jacksonville to St. Petersburg:

	A	B	C	D	E	F	G	H	I	J	K
1	From	To	On Route		Distance (km)	Time (hours)		Node	Net Flow		Supply/Demand
2	Boston	London	0		6,200	9.63		Boston	0	=	0
3	Boston	Berlin	0		7,250	11.26		Jacksonville	1	=	1
4	Boston	Istanbul	0		8,300	12.90		London	0	=	0
5	Jacksonville	London	1		7,900	12.27		Berlin	0	=	0
6	Jacksonville	Berlin	0		9,200	14.29		Istanbul	0	=	0
7	Jacksonville	Istanbul	0		10,100	15.69		St. Petersburg	-1	=	-1
8	London	St. Petersburg	1		1,980	3.08		Moscow	0	=	0
9	London	Moscow	0		2,300	3.57		Rostov	0	=	0
10	London	Rostov	0		2,860	4.44					
11	Berlin	St. Petersburg	0		1,280	1.99					
12	Berlin	Moscow	0		1,600	2.49					
13	Berlin	Rostov	0		1,730	2.69					
14	Istanbul	St. Petersburg	0		2,040	3.17		Travel Speed (mph)	400		
15	Istanbul	Moscow	0		1,700	2.64		km/hr	1.609		
16	Istanbul	Rostov	0		990	1.54		Travel Speed (km/hr)	643.6		
17											
18		Total Time	15.35								

Jacksonville to Moscow:

	A	B	C	D	E	F	G	H	I	J	K
1	**From**	**To**	**On Route**		**Distance (km)**	**Time (hours)**		**Node**	**Net Flow**		**Supply/Demand**
2	Boston	London	0		6,200	9.63		Boston	0	=	0
3	Boston	Berlin	0		7,250	11.26		Jacksonville	1	=	1
4	Boston	Istanbul	0		8,300	12.90		London	0	=	0
5	Jacksonville	London	1		7,900	12.27		Berlin	0	=	0
6	Jacksonville	Berlin	0		9,200	14.29		Istanbul	0	=	0
7	Jacksonville	Istanbul	0		10,100	15.69		St. Petersburg	0	=	0
8	London	St. Petersburg	0		1,980	3.08		Moscow	-1	=	-1
9	London	Moscow	1		2,300	3.57		Rostov	0	=	0
10	London	Rostov	0		2,860	4.44					
11	Berlin	St. Petersburg	0		1,280	1.99					
12	Berlin	Moscow	0		1,600	2.49					
13	Berlin	Rostov	0		1,730	2.69					
14	Istanbul	St. Petersburg	0		2,040	3.17		Travel Speed (mph)	400		
15	Istanbul	Moscow	0		1,700	2.64		km/hr	1.609		
16	Istanbul	Rostov	0		990	1.54		Travel Speed (km/hr)	643.6		
17											
18		Total Time	15.85								

Jacksonville to Rostov:

	A	B	C	D	E	F	G	H	I	J	K
1	**From**	**To**	**On Route**		**Distance (km)**	**Time (hours)**		**Node**	**Net Flow**		**Supply/Demand**
2	Boston	London	0		6,200	9.63		Boston	0	=	0
3	Boston	Berlin	0		7,250	11.26		Jacksonville	1	=	1
4	Boston	Istanbul	0		8,300	12.90		London	0	=	0
5	Jacksonville	London	1		7,900	12.27		Berlin	0	=	0
6	Jacksonville	Berlin	0		9,200	14.29		Istanbul	0	=	0
7	Jacksonville	Istanbul	0		10,100	15.69		St. Petersburg	0	=	0
8	London	St. Petersburg	0		1,980	3.08		Moscow	0	=	0
9	London	Moscow	0		2,300	3.57		Rostov	-1	=	-1
10	London	Rostov	1		2,860	4.44					
11	Berlin	St. Petersburg	0		1,280	1.99					
12	Berlin	Moscow	0		1,600	2.49					
13	Berlin	Rostov	0		1,730	2.69					
14	Istanbul	St. Petersburg	0		2,040	3.17		Travel Speed (mph)	400		
15	Istanbul	Moscow	0		1,700	2.64		km/hr	1.609		
16	Istanbul	Rostov	0		990	1.54		Travel Speed (km/hr)	643.6		
17											
18		Total Time	16.72								

The spreadsheets contain the following formulas:

Range Name	Cells
Distance	E2:E16
From	A2:A16
NetFlow	I2:I9
Node	H2:H9
OnRoute	C2:C16
SupplyDemand	K2:K9
Time	F2:F16
To	B2:B16
TotalTime	C18
TravelSpeed	I16

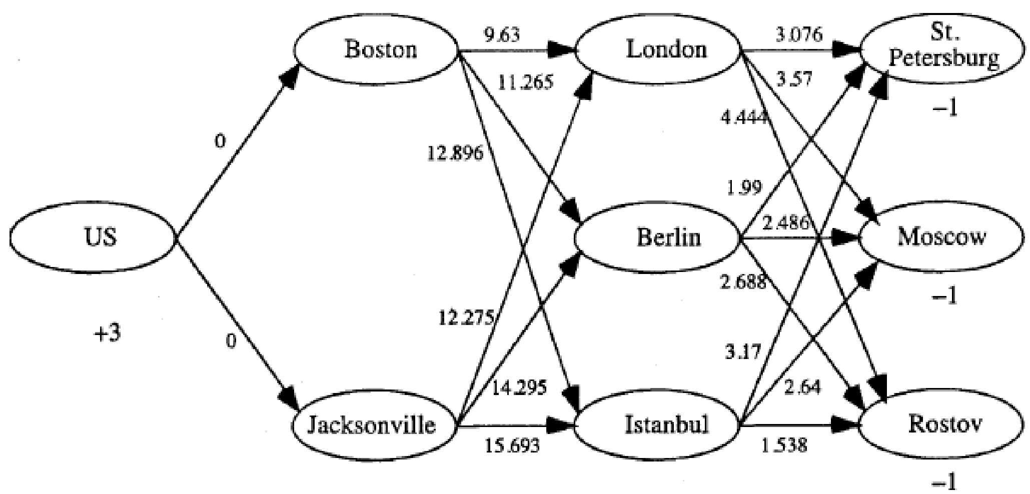
	H	I
14	Travel Speed (mph)	400
15	km/hr	1.609
16	Travel Speed (km/hr)	=I14*I15

	F
1	**Time (hours)**
2	=Distance/TravelSpeed
3	=Distance/TravelSpeed
4	=Distance/TravelSpeed
5	=Distance/TravelSpeed
6	=Distance/TravelSpeed
7	=Distance/TravelSpeed
8	=Distance/TravelSpeed
9	=Distance/TravelSpeed

	I
1	**Net Flow**
2	=SUMIF(From,Node,OnRoute)-SUMIF(To,Node,OnRoute)
3	=SUMIF(From,Node,OnRoute)-SUMIF(To,Node,OnRoute)
4	=SUMIF(From,Node,OnRoute)-SUMIF(To,Node,OnRoute)
5	=SUMIF(From,Node,OnRoute)-SUMIF(To,Node,OnRoute)
6	=SUMIF(From,Node,OnRoute)-SUMIF(To,Node,OnRoute)
7	=SUMIF(From,Node,OnRoute)-SUMIF(To,Node,OnRoute)
8	=SUMIF(From,Node,OnRoute)-SUMIF(To,Node,OnRoute)
9	=SUMIF(From,Node,OnRoute)-SUMIF(To,Node,OnRoute)

	B	C
18	Total Time	=SUMPRODUCT(OnRoute,Time)

Comparing all six solutions we see that the shortest path from the US to Saint Petersburg is Boston → London → Saint Petersburg with a total travel time of 12.71 hours. The shortest path from the US to Moscow is Boston → London → Moscow with a total travel time of 13.21 hours. The shortest path from the US to Rostov is Boston → Berlin → Rostov with a total travel time of 13.95 hours. The following network diagram highlights these shortest paths.



c) The President must satisfy each Russian city's military requirements at minimum cost. Therefore, this problem can be solved as a minimum-cost network flow problem. The two nodes representing US cities are supply nodes with a supply of 500 each (we measure all weights in 1000 tons). The three nodes representing Saint Petersburg, Moscow, and Rostov are demand nodes with demands of –320, -440, and –240, respectively. All nodes representing European airfields and ports are transshipment nodes. We measure the flow along the arcs in 1000 tons. For some arcs, capacity constraints are given. All arcs from the European ports into Saint Petersburg have zero capacity. All truck routes from the European ports into Rostov have a transportation limit of 2,500*16 = 40,000 tons. Since we measure the arc flows in 1000 tons, the corresponding arc capacities equal 40. An analogous computation yields arc capacities of 30 for both the arcs connecting the nodes London and Berlin to Rostov. For all other nodes we determine natural arc capacities based on the supplies and demands at the nodes. We define the unit costs along the arcs in the network in $1000 per 1000 tons (or, equivalently, $/ton). For example, the cost of transporting 1 ton of material from Boston to Hamburg equals $30,000 / 240 = $125, so the costs of transporting 1000 tons from Boston to Hamburg equals $125,000.

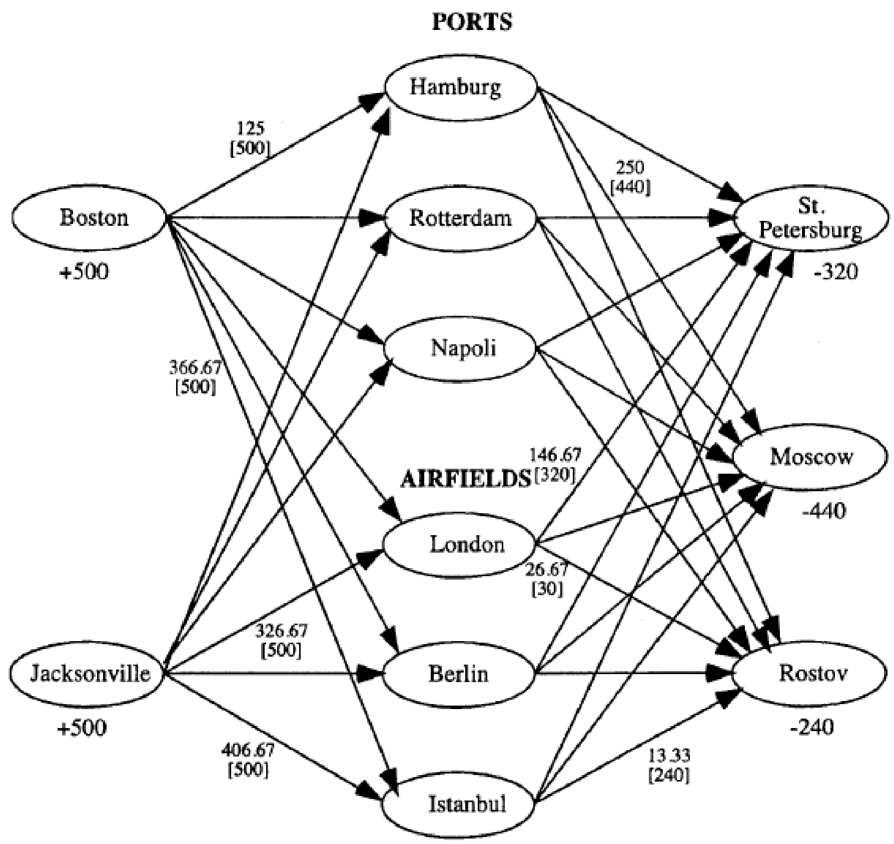
The objective is to satisfy all demands in the network at minimum cost. The following spreadsheet shows the entire linear programming model.

	A	B	C	D	E	F	G	H	I	J
1						Cost per			Vehicle	Unit
2			Ship		Capacity	Vehicle		Maximum	Capacity	Cost
3	From	To	(thousand tons)		(thousand tons)	($thousand)	Vehicle	Vehicles	(tons)	($/ton)
4	Boston	Berlin	0			50	Starlifter		150	$333.33
5	Boston	Hamburg	440			30	Transport		240	$125.00
6	Boston	Istanbul	60			55	Starlifter		150	$366.67
7	Boston	London	0			45	Starlifter		150	$300.00
8	Boston	Rotterdam	0			30	Transport		240	$125.00
9	Boston	Napoli	0			32	Transport		240	$133.33
10	Jacksonville	Berlin	0			57	Starlifter		150	$380.00
11	Jacksonville	Hamburg	0			48	Transport		240	$200.00
12	Jacksonville	Istanbul	150			61	Starlifter		150	$406.67
13	Jacksonville	London	350			49	Starlifter		150	$326.67
14	Jacksonville	Rotterdam	0			44	Transport		240	$183.33
15	Jacksonville	Napoli	0			56	Transport		240	$233.33
16	Berlin	St. Petersburg	0			24	Starlifter		150	$160.00
17	Hamburg	St. Petersburg	0	<=	0	3	Truck	0	16	$187.50
18	Istanbul	St. Petersburg	0			28	Starlifter		150	$186.67
19	London	St. Petersburg	320			22	Starlifter		150	$146.67
20	Rotterdam	St. Petersburg	0	<=	0	3	Truck	0	16	$187.50
21	Napoli	St. Petersburg	0	<=	0	5	Truck	0	16	$312.50
22	Berlin	Moscow	0			22	Starlifter		150	$146.67
23	Hamburg	Moscow	440			4	Truck		16	$250.00
24	Istanbul	Moscow	0			25	Starlifter		150	$166.67
25	London	Moscow	0			19	Starlifter		150	$126.67
26	Rotterdam	Moscow	0			5	Truck		16	$312.50
27	Napoli	Moscow	0			5	Truck		16	$312.50
28	Berlin	Rostov	0	<=	30	23	Starlifter	200	150	$153.33
29	Hamburg	Rostov	0	<=	40	7	Truck	2,500	16	$437.50
30	Istanbul	Rostov	210			2	Starlifter		150	$13.33
31	London	Rostov	30	<=	30	4	Starlifter	200	150	$26.67
32	Rotterdam	Rostov	0	<=	40	8	Truck	2,500	16	$500.00
33	Napoli	Rostov	0	<=	40	9	Truck	2,500	16	$562.50
34										
35		Total Cost ($thousand)	412,867							

	L	M	N	O
2		Net Flow		Supply/Demand
3	Node	(thousand tons)		(thousand tons)
4	Boston	500	=	500
5	Jacksonville	500	=	500
6	Berlin	0	=	0
7	Hamburg	0	=	0
8	Istanbul	0	=	0
9	London	0	=	0
10	Rotterdam	0	=	0
11	Napoli	0	=	0
12	St. Petersburg	-320	=	-320
13	Moscow	-440	=	-440
14	Rostov	-240	=	-240
15				
16				
17				
18				
19				
20				
21				
22		Capacity		
23		(tons)		
24	Starlifter	150		
25	Transport	240		
26	Truck	16		

	I	J
1	Vehicle	Unit
2	Capacity	Cost
3	(tons)	($/ton)
4	=VLOOKUP(G4,$L$24:$M$26,2)	=1000*F4/I4
5	=VLOOKUP(G5,$L$24:$M$26,2)	=1000*F5/I5
6	=VLOOKUP(G6,$L$24:$M$26,2)	=1000*F6/I6
7	=VLOOKUP(G7,$L$24:$M$26,2)	=1000*F7/I7

The total cost of the operation equals $412.867 million. The entire supply for Saint Petersburg is supplied from Jacksonville via London. The entire supply for Moscow is supplied from Boston via Hamburg. Of the 240 (= 240,000 tons) demanded by Rostov, 60 are shipped from Boston via Istanbul, 150 are shipped from Jacksonville via Istanbul, and 30 are shipped from Jacksonville via London. The paths used to ship supplies to Saint Petersburg, Moscow, and Rostov are highlighted on the following network diagram.



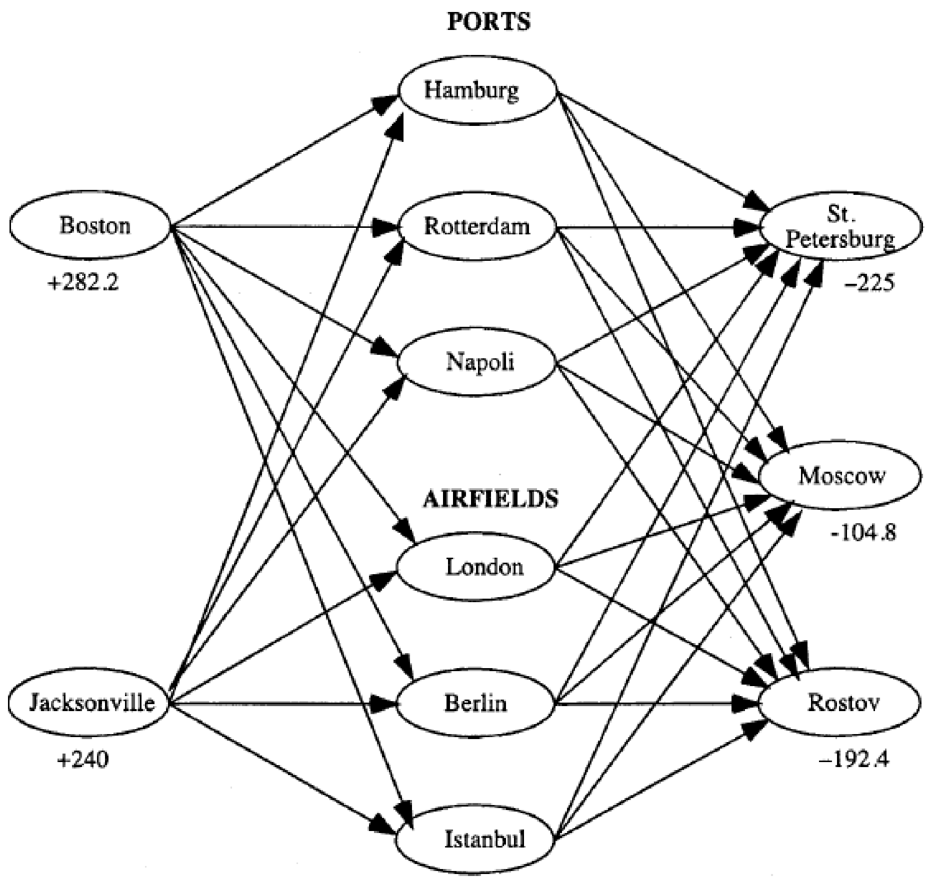
d) Now the President wants to maximize the amount of cargo transported from the US to the Russian cities. In other words, the President wants to maximize the flow from the two US cities to the three Russian cities. All the nodes representing the European ports and airfields are once again transshipment nodes. The flow along an arc is again measured in thousands of tons. The new restrictions can be transformed into arc capacities using the same approach that was used in part (c). The objective is now to maximize the combined flow into the three Russian cities.

The linear programming spreadsheet model describing the maximum flow problem appears as follows.

	A	B	C	D	E	F	G	H
1								**Vehicle**
2			**Ship**		**Capacity**		**Maximum**	**Capacity**
3	**From**	**To**	**(thousand tons)**		**(thousand tons)**	**Vehicle**	**Vehicles**	**(tons)**
4	Boston	Berlin	45	<=	45	Starlifter	300	150
5	Boston	Hamburg	19.2			Transport		240
6	Boston	Istanbul	45	<=	75	Starlifter	500	150
7	Boston	London	75	<=	75	Starlifter	500	150
8	Boston	Rotterdam	21.6			Transport		240
9	Boston	Napoli	46.4			Transport		240
10	Jacksonville	Berlin	75	<=	75	Starlifter	500	150
11	Jacksonville	Hamburg	0			Transport		240
12	Jacksonville	Istanbul	105	<=	105	Starlifter	700	150
13	Jacksonville	London	90	<=	90	Starlifter	600	150
14	Jacksonville	Rotterdam	0			Transport		240
15	Jacksonville	Napoli	0			Transport		240
16	Berlin	St. Petersburg	75	<=	75	Starlifter	500	150
17	Hamburg	St. Petersburg	0	<=	0	Truck	0	16
18	Istanbul	St. Petersburg	0	<=	0	Starlifter	0	150
19	London	St. Petersburg	150	<=	150	Starlifter	1,000	150
20	Rotterdam	St. Petersburg	0	<=	0	Truck	0	16
21	Napoli	St. Petersburg	0	<=	0	Truck	0	16
22	Berlin	Moscow	45	<=	45	Starlifter	300	150
23	Hamburg	Moscow	11.2	<=	11.2	Truck	700	16
24	Istanbul	Moscow	15	<=	15	Starlifter	100	150
25	London	Moscow	0	<=	30	Starlifter	200	150
26	Rotterdam	Moscow	9.6	<=	9.6	Truck	600	16
27	Napoli	Moscow	24	<=	24	Truck	1,500	16
28	Berlin	Rostov	0	<=	0	Starlifter	0	150
29	Hamburg	Rostov	8	<=	8	Truck	500	16
30	Istanbul	Rostov	135	<=	135	Starlifter	900	150
31	London	Rostov	15	<=	15	Starlifter	100	150
32	Rotterdam	Rostov	12	<=	12	Truck	750	16
33	Napoli	Rostov	22.4	<=	22.4	Truck	1,400	16
34								
35		Maximum Shipment	522.2					

	J	K	L	M
1				
2		**Net Flow**		**Supply/Demand**
3	**Node**	**(thousand tons)**		**(thousand tons)**
4	Boston	252.2		
5	Jacksonville	270		
6	Berlin	0	=	0
7	Hamburg	0	=	0
8	Istanbul	0	=	0
9	London	0	=	0
10	Rotterdam	0	=	0
11	Napoli	0	=	0
12	St. Petersburg	-225		
13	Moscow	-104.8		
14	Rostov	-192.4		
15				
16				
17				
18		Capacity		
19		(tons)		
20	Starlifter	150		
21	Transport	240		
22	Truck	16		

The spreadsheet shows all the amounts that are shipped between the various cities. The total supply for Saint Petersburg, Moscow, and Rostov equals 225,000 tons, 104,800 tons, and 192,400 tons, respectively. The following network diagram highlights the paths used to ship supplies between the US and the Russian Federation.



e) The creation of the new communication network is a minimum spanning tree problem. As usual, a greedy algorithm solves this type of problem.

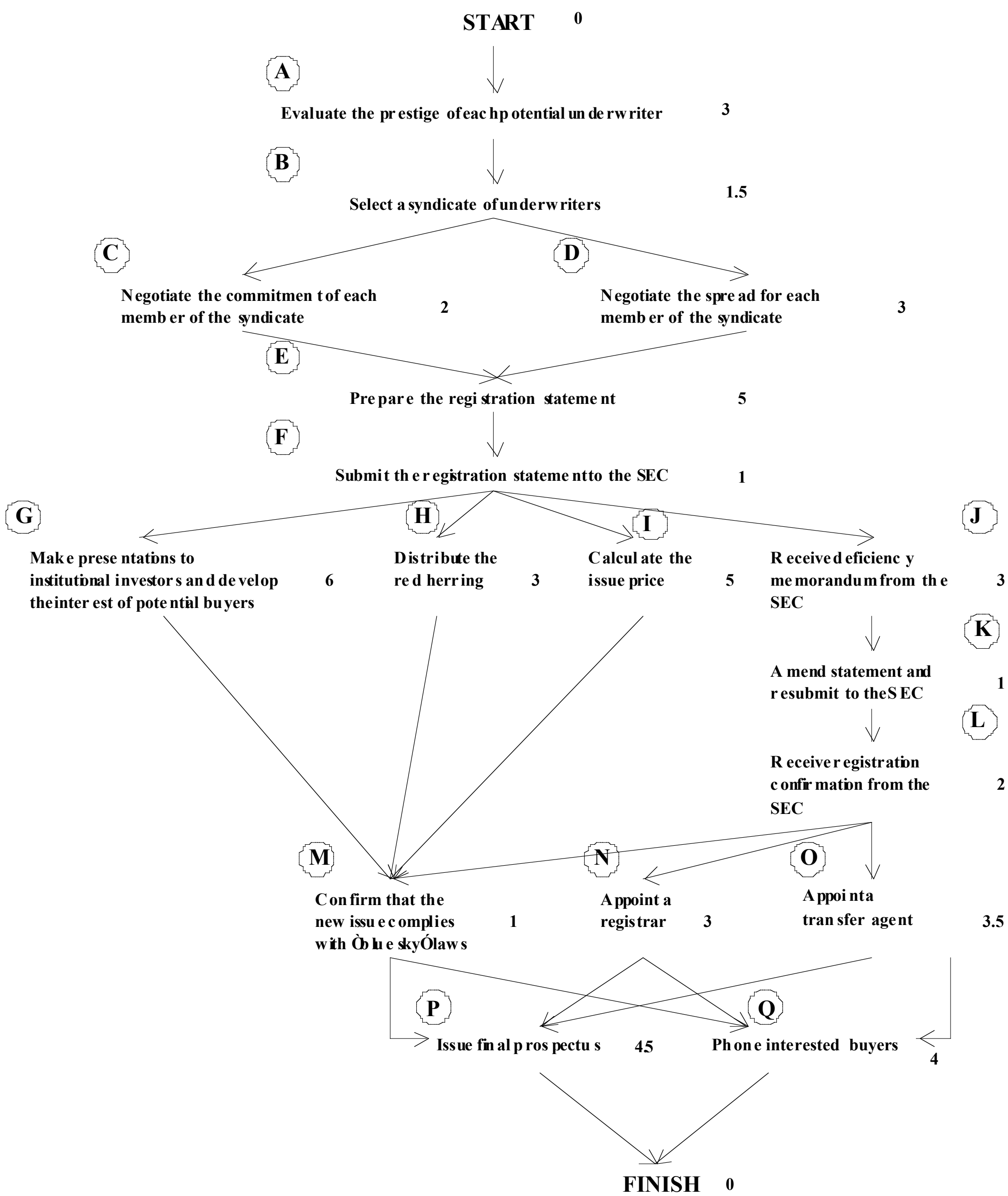
Arcs are added to the network in the following order (one of several optimal solutions):

Rostov – Orenburg	120
Ufa – Orenburg	75
Saratov – Orenburg	95
Saratov – Samara	100
Samara – Kazan	95
Ufa – Yekaterinburg	125
Perm – Yekaterinburg	85

The minimum cost of reestablishing the communication lines is $695 thousand.

Case 10.3

a) A diagram of the project network appears below.



To determine the project schedule and which activities are critical, we calculate the early start, late start, early finish, late finish, and slack below.

	A	B	C	D	E	F	G	H	I
1			**Time**		Week			**Slack**	
2	**Activity**	**Description**	(weeks)	**ES**	**EF**	**LS**	**LF**	(weeks)	**Critical?**
3	A	Evaluate prestige	3	0	3	0	3	0	Yes
4	B	Select syndicate	1.5	3	4.5	3	4.5	0	Yes
5	C	Negotiate commitment	2	4.5	6.5	5.5	7.5	1	No
6	D	Negotiate spread	3	4.5	7.5	4.5	7.5	0	Yes
7	E	Prepare registration	5	7.5	12.5	7.5	12.5	0	Yes
8	F	Submit registration	1	12.5	13.5	12.5	13.5	0	Yes
9	G	Present	6	13.5	19.5	16	22	2.5	No
10	H	Distribute red herring	3	13.5	16.5	19	22	5.5	No
11	I	Calculate price	5	13.5	18.5	17	22	3.5	No
12	J	Receive deficiency	3	13.5	16.5	13.5	16.5	0	Yes
13	K	Amend statement	1	16.5	17.5	16.5	17.5	0	Yes
14	L	Receive registration	2	17.5	19.5	17.5	19.5	0	Yes
15	M	Confirm blue sky	1	19.5	20.5	22	23	2.5	No
16	N	Appoint registrar	3	19.5	22.5	20	23	0.5	No
17	O	Appoint transfer	3.5	19.5	23	19.5	23	0	Yes
18	P	Issue prospectus	4.5	23	27.5	23	27.5	0	Yes
19	Q	Phone buyers	4	23	27	23.5	27.5	0.5	No
20									
21				**Project Duration**	27.5				

	A	B	C	D	E	F	G	H	I
1			**Time**			Week		**Slack**	
2	**Activity**	**Description**	(weeks)	**ES**	**EF**	**LS**	**LF**	(weeks)	**Critical?**
3	A	Evaluate prestige	3	0	=ES+Time	=LF-Time	=MIN(F4)	=LF-EF	=IF(Slack=0,"Yes","No")
4	B	Select syndicate	1.5	=MAX(E3)	=ES+Time	=LF-Time	=MIN(F5,F6)	=LF-EF	=IF(Slack=0,"Yes","No")
5	C	Negotiate commitment	2	=MAX(E4)	=ES+Time	=LF-Time	=MIN(F7)	=LF-EF	=IF(Slack=0,"Yes","No")
6	D	Negotiate spread	3	=MAX(E4)	=ES+Time	=LF-Time	=MIN(F7)	=LF-EF	=IF(Slack=0,"Yes","No")
7	E	Prepare registration	5	=MAX(E5,E6)	=ES+Time	=LF-Time	=MIN(F8)	=LF-EF	=IF(Slack=0,"Yes","No")
8	F	Submit registration	1	=MAX(E7)	=ES+Time	=LF-Time	=MIN(F9,F10,F11,F12)	=LF-EF	=IF(Slack=0,"Yes","No")
9	G	Present	6	=MAX(E8)	=ES+Time	=LF-Time	=MIN(F15)	=LF-EF	=IF(Slack=0,"Yes","No")
10	H	Distribute red herring	3	=MAX(E8)	=ES+Time	=LF-Time	=MIN(F15)	=LF-EF	=IF(Slack=0,"Yes","No")
11	I	Calculate price	5	=MAX(E8)	=ES+Time	=LF-Time	=MIN(F15)	=LF-EF	=IF(Slack=0,"Yes","No")
12	J	Receive deficiency	3	=MAX(E8)	=ES+Time	=LF-Time	=MIN(F13)	=LF-EF	=IF(Slack=0,"Yes","No")
13	K	Amend statement	1	=MAX(E12)	=ES+Time	=LF-Time	=MIN(F14)	=LF-EF	=IF(Slack=0,"Yes","No")
14	L	Receive registration	2	=MAX(E13)	=ES+Time	=LF-Time	=MIN(F15,F16,F17)	=LF-EF	=IF(Slack=0,"Yes","No")
15	M	Confirm blue sky	1	=MAX(E9,E10,E11,E14)	=ES+Time	=LF-Time	=MIN(F18,F19)	=LF-EF	=IF(Slack=0,"Yes","No")
16	N	Appoint registrar	3	=MAX(E14)	=ES+Time	=LF-Time	=MIN(F18,F19)	=LF-EF	=IF(Slack=0,"Yes","No")
17	O	Appoint transfer	3.5	=MAX(E14)	=ES+Time	=LF-Time	=MIN(F18,F19)	=LF-EF	=IF(Slack=0,"Yes","No")
18	P	Issue prospectus	4.5	=MAX(E15,E16,E17)	=ES+Time	=LF-Time	=ProjectDuration	=LF-EF	=IF(Slack=0,"Yes","No")
19	Q	Phone buyers	4	=MAX(E15,E16,E17)	=ES+Time	=LF-Time	=ProjectDuration	=LF-EF	=IF(Slack=0,"Yes","No")
20									
21				**Project Duration**	=MAX(EF)				

Range Name	**Cells**
Activity	A3:A19
Critical?	I3:I19
Description	B3:B19
EF	E3:E19
ES	D3:D19
LF	G3:G19
LS	F3:F19
ProjectDuration	E21
Slack	H3:H19
Time	C3:C19

The initial public offering process is 27.5 weeks long. The critical path is:
START $\rightarrow$ A $\rightarrow$ B $\rightarrow$ D $\rightarrow$ E $\rightarrow$ F $\rightarrow$ J $\rightarrow$ K $\rightarrow$ L $\rightarrow$ O $\rightarrow$ P $\rightarrow$ FINISH

b)

	A	B	C	D	E	F	G	H	I	J	K
1			Time		Cost		Maximum	Crash Cost	Start	Time	Finish
2			(weeks)		($thousand)		Time Reduction	per Week Saved	Time	Reduction	Time
3	Activity	Description	Normal	Crash	Normal	Crash	(weeks)	($thousand)	(week)	(weeks)	(week)
4	A	Evaluate prestige	3	1.5	8	14	1.5	4	0	1.5	1.5
5	B	Select syndicate	1.5	0.5	4.5	8	1	3.5	1.5	1	2
6	C	Negotiate commitment	2	2	9	9	0	0	3	0	5
7	D	Negotiate spread	3	3	12	12	0	0	2	0	5
8	E	Prepare registration	5	4	50	95	1	45	5	0	10
9	F	Submit registration	1	1	1	1	0	0	10	0	11
10	G	Present	6	4	25	60	2	17.5	11	0	17
11	H	Distribute red herring	3	2	15	22	1	7	14	0	17
12	I	Calculate price	5	3.5	12	31	1.5	12.67	12	0	17
13	J	Receive deficiency	3	3	0	3	0	0	11	0	14
14	K	Amend statement	1	0.5	6	9	0.5	6	14	0.5	14.5
15	L	Receive registration	2	2	0	0	0	0	14.5	0	16.5
16	M	Confirm blue sky	1	0.5	5	8.3	0.5	6.6	17	0	18
17	N	Appoint registrar	3	1.5	12	19	1.5	4.67	16.5	1.5	18
18	O	Appoint transfer	3.5	1.5	13	21	2	4	16.5	2	18
19	P	Issue prospectus	4.5	2	40	99	2.5	23.6	18	0.5	22
20	Q	Phone buyers	4	1.5	9	20	2.5	4.4	18	0	22
21											
22											Max Time
23								Project Finish Time (week)	22	<=	22
24											
25								Total Cost	260.8		

	G	H	I	J	K
1	Maximum	Crash Cost	Start	Time	Finish
2	Time Reduction	per Week Saved	Time	Reduction	Time
3	(weeks)	($thousand)	(week)	(weeks)	(week)
4	=NormalTime-CrashTime	=(CrashCost-NormalCost)/MaxTimeReduction	0	1.5	=StartTime+NormalTime-TimeReduction
5	=NormalTime-CrashTime	=(CrashCost-NormalCost)/MaxTimeReduction	1.5	1	=StartTime+NormalTime-TimeReduction
6	=NormalTime-CrashTime	0	3	0	=StartTime+NormalTime-TimeReduction
7	=NormalTime-CrashTime	0	2	0	=StartTime+NormalTime-TimeReduction
8	=NormalTime-CrashTime	=(CrashCost-NormalCost)/MaxTimeReduction	5	0	=StartTime+NormalTime-TimeReduction
9	=NormalTime-CrashTime	0	10	0	=StartTime+NormalTime-TimeReduction
10	=NormalTime-CrashTime	=(CrashCost-NormalCost)/MaxTimeReduction	11	0	=StartTime+NormalTime-TimeReduction
11	=NormalTime-CrashTime	=(CrashCost-NormalCost)/MaxTimeReduction	14	0	=StartTime+NormalTime-TimeReduction
12	=NormalTime-CrashTime	=(CrashCost-NormalCost)/MaxTimeReduction	12	0	=StartTime+NormalTime-TimeReduction
13	=NormalTime-CrashTime	0	11	0	=StartTime+NormalTime-TimeReduction
14	=NormalTime-CrashTime	=(CrashCost-NormalCost)/MaxTimeReduction	14	0.5	=StartTime+NormalTime-TimeReduction
15	=NormalTime-CrashTime	0	14.5	0	=StartTime+NormalTime-TimeReduction
16	=NormalTime-CrashTime	=(CrashCost-NormalCost)/MaxTimeReduction	17	0	=StartTime+NormalTime-TimeReduction
17	=NormalTime-CrashTime	=(CrashCost-NormalCost)/MaxTimeReduction	16.5	1.5	=StartTime+NormalTime-TimeReduction
18	=NormalTime-CrashTime	=(CrashCost-NormalCost)/MaxTimeReduction	16.5	2	=StartTime+NormalTime-TimeReduction
19	=NormalTime-CrashTime	=(CrashCost-NormalCost)/MaxTimeReduction	18	0.5	=StartTime+NormalTime-TimeReduction
20	=NormalTime-CrashTime	=(CrashCost-NormalCost)/MaxTimeReduction	18	0	=StartTime+NormalTime-TimeReduction

	H	I
25	Total Cost	=SUM(NormalCost)+SUMPRODUCT(CrashCostPerWeekSaved,TimeReduction)

Range Name	Cells
Activity	A4:A20
CrashCost	F4:F20
CrashCostPerWeekSaved	H4:H20
CrashTime	D4:D20
Description	B4:B20
FinishTime	K4:K20
MaxTime	K23
MaxTimeReduction	G4:G20
NormalCost	E4:E20
NormalTime	C4:C20
ProjectFinishTime	I23
StartTime	I4:I20
TimeReduction	J4:J20
TotalCost	I25

The constraints in the linear programming spreadsheet model were as follows:

TimeReduction ≤ MaxTimeReduction
ProjectFinishTime ≤ MaxTime
BStart ≥ AFinish
CStart ≥ BFinish
DStart ≥ BFinish
EStart ≥ CFinish
EStart ≥ DFinish
FStart ≥ EFinish
GStart ≥ FFinish
HStart ≥ FFinish
IStart ≥ FFinish
JStart ≥ FFinish
KStart ≥ JFinish
LStart ≥ KFinish
MStart ≥ GFinish
MStart ≥ HFinish
MStart ≥ IFinish
MStart ≥ LFinish
NStart ≥ LFinish
OStart ≥ LFinish
PStart ≥ MFinish
PStart ≥ NFinish
PStart ≥ OFinish
QStart ≥ MFinish
QStart ≥ NFinish
QStart ≥ OFinish
ProjectFinishTime ≥ PFinish
ProjectFinishTime ≥ QFinish

Janet and Gilbert should reduce the time for step A (evaluating the prestige of each potential underwriter) by 1.5 weeks, the time for step B (selecting a syndicate of underwriters) by 1 week, the time for step K (amending statement and resubmitting it to the SEC) by 0.5 weeks, the time for step N (appointing a registrar) by 1.5 weeks, the time for step O (appointing a transfer agent) by two weeks, and the time for step P (issuing final prospectus) by 0.5 weeks. Janet and Gilbert can now meet the new deadline of 22 weeks at a total cost of $260,800.

c) We use the same model formulation that was used in part (c). We change one constraint, however. The project duration now has to be less-than-or-equal to 24 weeks instead of 22 weeks. We obtain the following solution.

	A	B	C	D	E	F	G	H	I	J	K
1			Time		Cost		Maximum	Crash Cost	Start	Time	Finish
2			(weeks)		($thousand)		Time Reduction	per Week Saved	Time	Reduction	Time
3	Activity	Description	Normal	Crash	Normal	Crash	(weeks)	($thousand)	(week)	(weeks)	(week)
4	A	Evaluate prestige	3	1.5	8	14	1.5	4	0	1.5	1.5
5	B	Select syndicate	1.5	0.5	4.5	8	1	3.5	1.5	1	2
6	C	Negotiate commitment	2	2	9	9	0	0	3	0	5
7	D	Negotiate spread	3	3	12	12	0	0	2	0	5
8	E	Prepare registration	5	4	50	95	1	45	5	0	10
9	F	Submit registration	1	1	1	1	0	0	10	0	11
10	G	Present	6	4	25	60	2	17.5	12.5	0	18.5
11	H	Distribute red herring	3	2	15	22	1	7	11	0	14
12	I	Calculate price	5	3.5	12	31	1.5	12.67	13.5	0	18.5
13	J	Receive deficiency	3	3	0	3	0	0	11	0	14
14	K	Amend statement	1	0.5	6	9	0.5	6	14	0.5	14.5
15	L	Receive registration	2	2	0	0	0	0	14.5	0	16.5
16	M	Confirm blue sky	1	0.5	5	8.3	0.5	6.6	18.5	0	19.5
17	N	Appoint registrar	3	1.5	12	19	1.5	4.67	16.5	0	19.5
18	O	Appoint transfer	3.5	1.5	13	21	2	4	16.5	0.5	19.5
19	P	Issue prospectus	4.5	2	40	99	2.5	23.6	19.5	0	24
20	Q	Phone buyers	4	1.5	9	20	2.5	4.4	20	0	24
21											
22											Max Time
23								Project Finish Time (week)	24	<=	24
24											
25								Total Cost	236		

Janet and Gilbert should reduce the time for step A (evaluating the prestige of each potential underwriter) by 1.5 weeks, the time for step B (selecting a syndicate of underwriters) by 1 week, the time for step K (amending statement and resubmitting it to the SEC) by 0.5 weeks, and the time for step O (appointing a transfer agent) by 0.5 weeks. Janet and Gilbert can now meet the new deadline of 24 weeks at a total cost of $236,000.