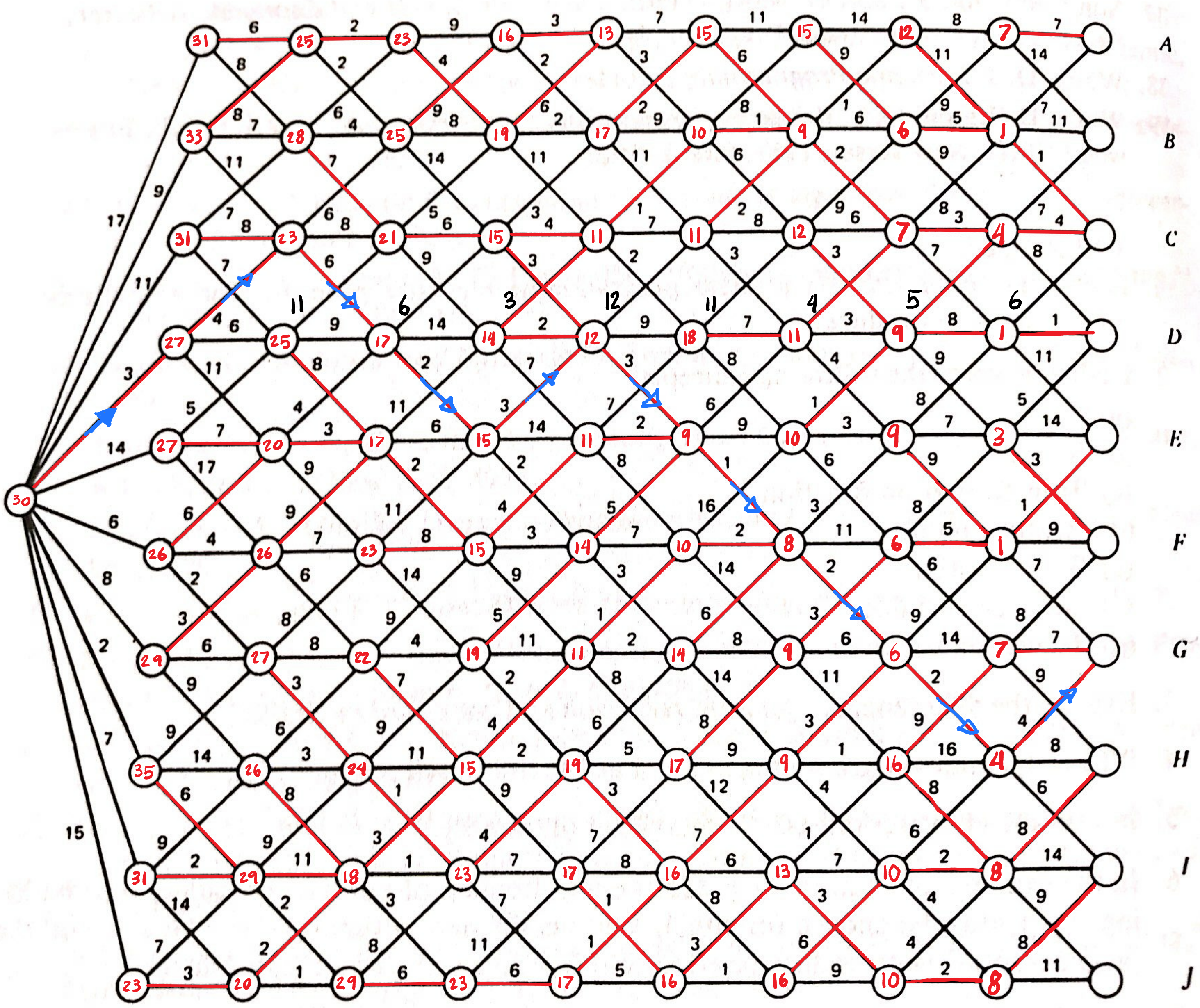
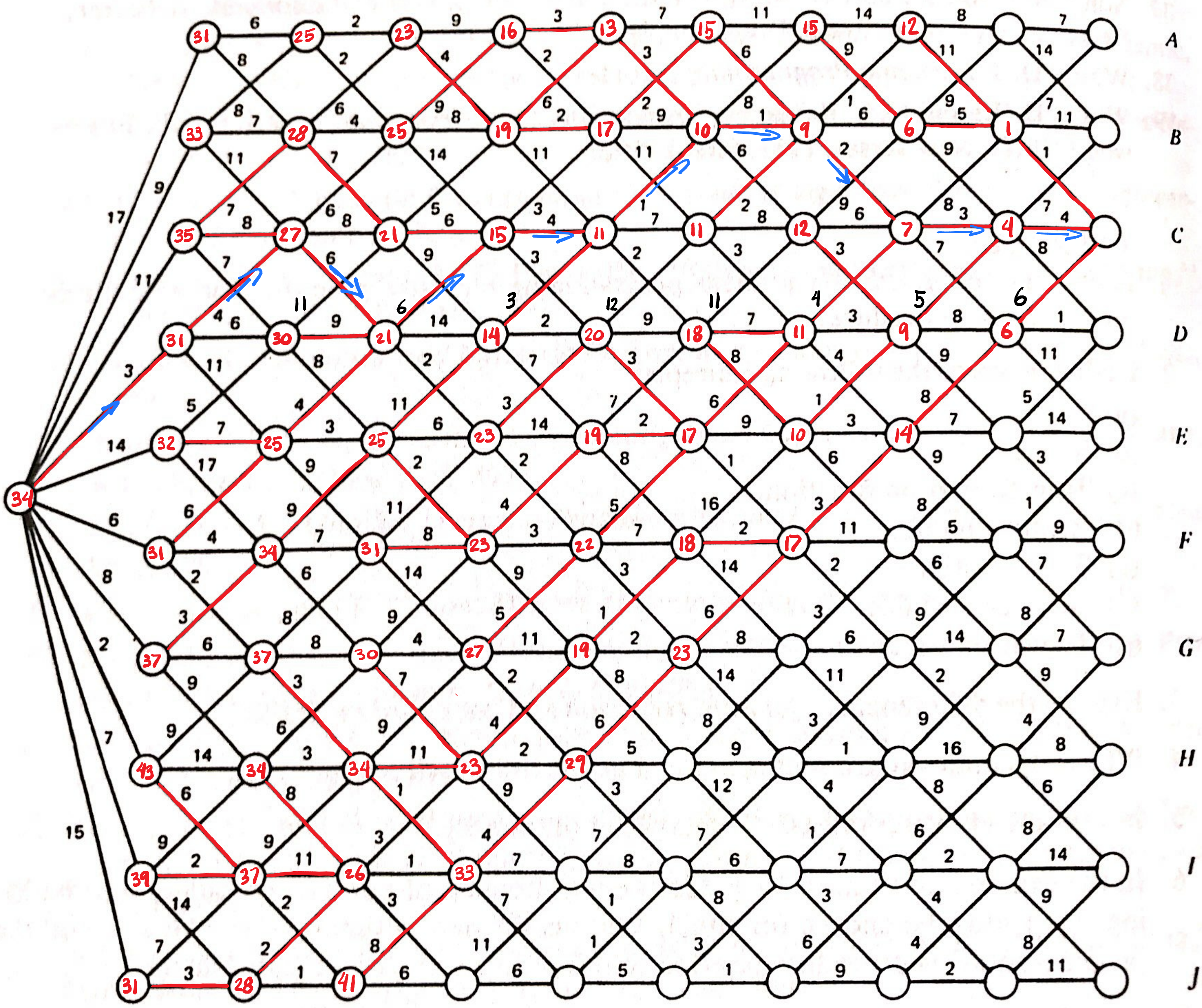
10.11
(a)



10.11

(b)



10 - 12

Stage n $\rightarrow$ start of month 7 + 1 - n

S_n $\rightarrow$ Inventory at stage n

d_n $\rightarrow$ amount to be bought at stage n (0, 500,1000 or 1500)

D_n $\rightarrow$ demand at stage n

C_n $\rightarrow$ cost of purchasing at stage n (0 if d_n=0, 3000 if d_n=500, 5000 if d_n=1000, 75000 if d_n=1500)

Transition function:

$S_{\{n-1\}} = S_n + d_n - D_n$

Return function:

$r_n(S_n,d_n) = C_n + 0.5(S_n + d_n - D_n)$

Objective function:

$\min f_n(S_n,d_n) = r_n(S_n,d_n) + f^*_{\{n-1\}}(S_{\{n-1\}})$

Also, initial inventory is 0, so $S_7 = 0$

Stage 1:

state	d					
	0	500	1000	1500	f*(s)	d*
0	inf	inf	5200	7950	5200	1000
100	inf	3000	5250	8000	3000	500
200	inf	3050	5300	8050	3050	500
300	inf	3100	5350	8100	3100	500
400	inf	3150	5400	8150	3150	500
500	inf	3200	5450	8200	3200	500
600	0	3250	5500	8250	0	0

Stage 2:

state	d					
	0	500	1000	1500	f*(s)	d*
0	inf	inf	inf	10750	10750	1500
100	inf	inf	inf	10850	10850	1500
200	inf	inf	10200	10950	10200	1000
300	inf	inf	8050	7800	7800	1500
400	inf	inf	8150	inf	8150	1000
500	inf	inf	8250	inf	8250	1000
600	inf	inf	8350	inf	8350	1000

Stage 3:

state	d					
	0	500	1000	1500	f*(s)	d*
0	inf	inf	15900	16150	15900	1000
100	inf	inf	15300	inf	15300	1000
200	inf	inf	12950	inf	12950	1000
300	inf	inf	13350	inf	13350	1000
400	inf	13750	13500	inf	13500	1000
500	inf	13900	13650	inf	13650	1000
600	inf	13300	inf	inf	13300	500

Stage 4:

state	d					
	0	500	1000	1500	f*(s)	d*
0	inf	18350	18600	inf	18350	500
100	inf	16050	inf	inf	16050	500
200	inf	16500	inf	inf	16500	500
300	inf	16700	inf	inf	16700	500
400	15900	16900	inf	inf	15900	0
500	15350	16600	inf	inf	15350	0
600	13050	inf	inf	inf	13050	0

Stage 5:

state	d					
	0	500	1000	1500	f*(s)	d*
0	inf	inf	21600	inf	21600	1000
100	inf	inf	21850	inf	21850	1000
200	inf	inf	21100	inf	21100	1000
300	inf	21350	20600	inf	20600	1000
400	inf	19100	18350	inf	18350	1000
500	inf	19600	inf	inf	19600	500
600	inf	19850	inf	inf	19850	500

Stage 6 :

state	d					
	0	500	1000	1500	f*(s)	d*
0	inf	inf	inf	26050	26050	1500
100	inf	inf	26600	27350	26600	1000
200	inf	inf	26900	27650	26900	1000
300	inf	inf	26200	inf	26200	1000
400	inf	inf	25750	inf	25750	1000
500	inf	inf	23550	inf	23550	1000
600	inf	24600	24850	inf	24600	500

Stage 7 :

state	d					
	0	500	1000	1500	f*(s)	d*
0	inf	inf	31350	inf	31350	1000

	d1		d2		d3		d4		d5		d6		d7
optimal decisions:	[[1000]	,	[1000]	,	[1000]	,	[0]	,	[1000]	,	[1500]	,	[500]]

	s1		s2		s3		s4		s5		s6		s7
optimal states:	[[0]	,	[300]	,	[200]	,	[400]	,	[0]	,	[100]	,	[400]]

10.14 (a) This is a transportation problem:

$$\min_{x_{ij}} \sum_{i=1}^{4} \sum_{j=1}^{6} c_{ij} x_{ij}$$

s.t.

$$\sum_{j=1}^{6} x_{ij} = 1, \quad i = 1, \dots, 4$$

$$\sum_{i=1}^{4} x_{ij} \leq 1, \quad j = 1, \dots, 6$$

$$x_{ij} \in \{0, 1\}, \quad i = 1, \dots, 4$$
$$j = 1, \dots, 6$$

10 - 14

Stage n $\rightarrow$ car 5 -n (could be n, but I reversed it)

S_n $\rightarrow$ set of available markets at stage n (since we have that one market gets only one car)

d_n $\rightarrow$ market to assign at stage n (market to assign car n)

Transition function:

$S_{\{n-1\}} = S_n \setminus d_n$ (set S_n with element d_n removed from it)

Return function:

$r_n(S_n, d_n)$ = cost of assigning car n to market d_n

Objective function

$\min f_n(S_n, d_n) = r_n(S_n, d_n) + f^*_{\{n-1\}}(S_{\{n-1\}})$

Note that at stage 1, we have 3 markets left (3 have already been picked, 6 - 3 = 3), so we can pick any market in the $(6!)/((6-3)!(3!))$ possible sets of markets

Generalizing, at stage n, we have (6 - 4 + n) markets left (- 4 + n have already been picked), so we can pick any market in the $(6!)/((4-n)!(6-4+n)!)$

Stage 1:

State	d							
	1	2	3	4	5	6	f*(s)	d*
(1, 2, 3)	7	12	9	inf	inf	inf	7	1
(1, 2, 4)	7	12	inf	15	inf	inf	7	1
(1, 2, 5)	7	12	inf	inf	8	inf	7	1
(1, 2, 6)	7	12	inf	inf	inf	14	7	1
(1, 3, 4)	7	inf	9	15	inf	inf	7	1
(1, 3, 5)	7	inf	9	inf	8	inf	7	1
(1, 3, 6)	7	inf	9	inf	inf	14	7	1
(1, 4, 5)	7	inf	inf	15	8	inf	7	1
(1, 4, 6)	7	inf	inf	15	inf	14	7	1
(1, 5, 6)	7	inf	inf	inf	8	14	7	1
(2, 3, 4)	inf	12	9	15	inf	inf	9	3
(2, 3, 5)	inf	12	9	inf	8	inf	8	5
(2, 3, 6)	inf	12	9	inf	inf	14	9	3
(2, 4, 5)	inf	12	inf	15	8	inf	8	5
(2, 4, 6)	inf	12	inf	15	inf	14	12	2
(2, 5, 6)	inf	12	inf	inf	8	14	8	5
(3, 4, 5)	inf	inf	9	15	8	inf	8	5
(3, 4, 6)	inf	inf	9	15	inf	14	9	3
(3, 5, 6)	inf	inf	9	inf	8	14	8	5

State	1	2	3	4	5	6	f*(s)	d*
(4, 5, 6)	inf	inf	inf	15	8	14	8	5

Stage 2:

State	d							
	1	2	3	4	5	6	f*(s)	d*
(1, 2, 3, 4)	14	17	12	19	inf	inf	12	3
(1, 2, 3, 5)	13	17	12	inf	13	inf	12	3
(1, 2, 3, 6)	14	17	12	inf	inf	20	12	3
(1, 2, 4, 5)	13	17	inf	19	13	inf	13	1; 5
(1, 2, 4, 6)	17	17	inf	19	inf	20	17	1; 2
(1, 2, 5, 6)	13	17	inf	inf	13	20	13	1; 5
(1, 3, 4, 5)	13	inf	12	19	13	inf	12	3
(1, 3, 4, 6)	14	inf	12	19	inf	20	12	3
(1, 3, 5, 6)	13	inf	12	inf	13	20	12	3
(1, 4, 5, 6)	13	inf	inf	19	13	20	13	1; 5
(2, 3, 4, 5)	inf	18	13	20	15	inf	13	3
(2, 3, 4, 6)	inf	19	17	21	inf	22	17	3
(2, 3, 5, 6)	inf	18	13	inf	15	21	13	3
(2, 4, 5, 6)	inf	18	inf	20	18	21	18	2; 5
(3, 4, 5, 6)	inf	inf	13	20	15	21	13	3

Stage 3:

State	d							
	1	2	3	4	5	6	f*(s)	d*
(1, 2, 3, 4, 5)	21	22	20	28	19	inf	19	5
(1, 2, 3, 4, 6)	25	22	24	28	inf	24	22	2
(1, 2, 3, 5, 6)	21	22	20	inf	19	24	19	5
(1, 2, 4, 5, 6)	26	23	inf	29	24	25	23	2
(1, 3, 4, 5, 6)	21	inf	20	28	19	24	19	5
(2, 3, 4, 5, 6)	inf	23	25	29	24	25	23	2

Stage 4:

State	d							
	1	2	3	4	5	6	f*(s)	d*
(1, 2, 3, 4, 5, 6)	32	30	31	33	29	30	29	5

car 4 car 3 car 2 car 1

optimal decisions: [[5] , [2] , [3] , [1]] optimal states: [[(1, 2, 3, 4, 5, 6)] , [(1, 2, 3, 4, 6)] , [(1, 3, 4, 6)] , [(1, 4, 6)]]

10.16: Assumption, all available capital is invested every year.

A: 1000 ⟶ 2 calves + 500 per year

B: 1000 ⟶ 3 calves + 200 per year

Define

Stage 1: beginning of year 4 (n ↙ Stage 1)

Stage 2: beginning of year 3

Stage 3: beginning of year 2

Stage 4: beginning of year 1

S_n: Available capital at stage n

d_n: Capital invested in cattle A at stage n. Note that, consequently, $S_n - d_n$ is the amount invested in cattle B at stage n.

r_n: # of cattle at the end of the 4th year due to investment made at stage n:

$$r_n = n\left(\frac{2d_n}{1000} + \frac{3(S_n - d_n)}{1000}\right) = \frac{n}{1000}\left[3S_n - d_n\right]$$

(per year) ↗ n; A ↙ $\frac{2d_n}{1000}$; B ↓ $\frac{3(S_n-d_n)}{1000}$

State transition function $T(S_n, d_n)$:

$$S_{n-1} = T(S_n, d_n) = 0.5 d_n + 0.2(S_n - d_n)$$

T represents the yearly capital rollover

Objective function: $\max f_n(S_n) = \max \ r_n(d_n) + f^*_{n-1}(S_{n-1})$

Stage 1:

$$f_1(S_1) = \frac{1}{1000}(3S_1 - d_1)$$

Then, $d_1^* = 0$, $f_1^*(S_1) = 3S_1/1000$

Stage 2:

$$S_1 = 0.5 d_2 + 0.2(S_2 - d_2)$$

$$f_2(S_2) = \frac{2}{1000}(3S_2 - d_2) + \frac{3}{1000}(0.5d_2 + 0.2(S_2 - d_2))$$

$$= (6.6 S_2 - 1.1 d_2)/1000$$

Then, $d_2^* = 0$, $f_2^*(S_2) = 6.6 S_2/1000$

Stage 3:

$$S_2 = 0.2 S_3 + 0.3 d_3$$

$$f_3(S_3) = \frac{3}{1000}(3S_3 - d_3) + \frac{6.6}{1000}(0.2 S_3 + 0.3 d_3)$$

$$= (10.32 S_3 - 1.02 d_2)/1000$$

Then, $d_3^* = 0$, $f_3^*(S_3) = 10.32 S_3/1000$

$S_3 = 0.2\, S_4 + 0.3\, d_4$

$$f_4(S_4) = \frac{4}{1000}(3S_4 - d_4) + \frac{10.32}{1000}(0.2S_4 + 0.3d_4)$$

$$= (14.064\, S_4 - 0.904\, d_4)/1000$$

Then, $d_4^* = 0$, $f_4^*(S_4) = 14.064 S_4 / 1000$

$S_4 = 5000$ (initial investment)

$\Rightarrow f_4^*(5000) = 70.32$ calves

Since $d_n^* = 0$ for all n in $\{1, 2, 3, 4\}$, all the money should be allocated on strain B.

10-25

$Stage n \rightarrow type\ of\ \ subsystem\ (n=1,...,5)$

$d_n \rightarrow option: a\ or\ b$

$C_n(d_n) \rightarrow cost\ of\ \ decision\ d_n$

$S_n \rightarrow total\ cost: \Sigma_{i=1}^{n} C_i\ d_i (between\ 0\ and\ 100k)$

$E_n(d_n) \rightarrow effectiveness\ of\ \ decision\ d_n\ at\ stage\ n$

$Transition\ \ function:$

$S_{n-1} = S_n - C_n(d_n)$

$Return\ \ function:$

$r_n(S_n, d_n) = E_n(d_n)$

$Objective\ \ function:$

$max\ f_n(S_n, d_n) = (r_n(S_n, d_n))(f*_{n-1}(S_{n-1}))$

$Note\ that\ in\ this\ \ problem, we\ accumulate\ returns\ with\ multiplication.$

Stage 1:

state	d			
	a	b	f*(s)	d*
0	0	0	0	a ; b
5	0	0	0	a ; b
10	0	0	0	a ; b
15	0	0.9	0.9	b
20	0.95	0.9	0.95	a
25	0.95	0.9	0.95	a
30	0.95	0.9	0.95	a
35	0.95	0.9	0.95	a
40	0.95	0.9	0.95	a
45	0.95	0.9	0.95	a
50	0.95	0.9	0.95	a
55	0.95	0.9	0.95	a
60	0.95	0.9	0.95	a

65	0.95	0.9	0.95	a
70	0.95	0.9	0.95	a
75	0.95	0.9	0.95	a
80	0.95	0.9	0.95	a
85	0.95	0.9	0.95	a
90	0.95	0.9	0.95	a
95	0.95	0.9	0.95	a
100	0.95	0.9	0.95	a

Stage 2:

state	d			
	a	b	f*(s)	d*
0	0	0	0	a ; b
5	0	0	0	a ; b
10	0	0	0	a ; b
15	0	0	0	a ; b
20	0	0	0	a ; b
25	0	0.765	0.765	b
30	0.81	0.8075	0.81	a
35	0.855	0.8075	0.855	a
40	0.855	0.8075	0.855	a
45	0.855	0.8075	0.855	a
50	0.855	0.8075	0.855	a
55	0.855	0.8075	0.855	a
60	0.855	0.8075	0.855	a
65	0.855	0.8075	0.855	a
70	0.855	0.8075	0.855	a
75	0.855	0.8075	0.855	a
80	0.855	0.8075	0.855	a
85	0.855	0.8075	0.855	a
90	0.855	0.8075	0.855	a
95	0.855	0.8075	0.855	a
100	0.855	0.8075	0.855	a

Stage 3:

state	d			
	a	b	f*(s)	d*

0	0	0	0	a ;b
5	0	0	0	a ;b
10	0	0	0	a ;b
15	0	0	0	a ;b
20	0	0	0	a ;b
25	0	0	0	a ;b
30	0	0	0	a ;b
35	0	0	0	a ;b
40	0	0	0	a ;b
45	0	0	0	a ;b
50	0	0.6885	0.6885	b
55	0.72675	0.729	0.729	b
60	0.7695	0.7695	0.7695	a ;b
65	0.81225	0.7695	0.81225	a
70	0.81225	0.7695	0.81225	a
75	0.81225	0.7695	0.81225	a
80	0.81225	0.7695	0.81225	a
85	0.81225	0.7695	0.81225	a
90	0.81225	0.7695	0.81225	a
95	0.81225	0.7695	0.81225	a
100	0.81225	0.7695	0.81225	a

Stage 4：

state	d			
	a	b	f*(s)	d*
0	0	0	0	a ;b
5	0	0	0	a ;b
10	0	0	0	a ;b
15	0	0	0	a ;b

20	0	0	0	a ;b
25	0	0	0	a ;b
30	0	0	0	a ;b
35	0	0	0	a ;b
40	0	0	0	a ;b
45	0	0	0	a ;b
50	0	0	0	a ;b
55	0	0	0	a ;b
60	0	0	0	a ;b
65	0	0	0	a ;b
70	0	0	0	a ;b
75	0	0	0	a ;b
80	0	0.585225	0.585225	b
85	0	0.61965	0.61965	b
90	0.61965	0.654075	0.654075	b
95	0.6561	0.690412	0.690412	b
100	0.69255	0.690412	0.69255	a

Stage 5:

state	d			
	a	b	f*(s)	d*
0	0	0	0	a ;b
5	0	0	0	a ;b
10	0	0	0	a ;b
15	0	0	0	a ;b
20	0	0	0	a ;b
25	0	0	0	a ;b
30	0	0	0	a ;b
35	0	0	0	a ;b
40	0	0	0	a ;b
45	0	0	0	a ;b
50	0	0	0	a ;b
55	0	0	0	a ;b
60	0	0	0	a ;b
65	0	0	0	a ;b

70	0	0	0	a ;b
75	0	0	0	a ;b
80	0	0	0	a ;b
85	0	0	0	a ;b
90	0	0.46818	0.46818	b
95	0	0.49572	0.49572	b
100	0.555964	0.52326	0.555964	a

	Navigation System	Radar	WPNS Delivery	TACAN	UHF Radio
optimal decisions:	[['a'],	['b'],	['b'],	['b']	['b']]
optimal state:	[[100],	[80],	[50],	[25],	[15]]