Markovian Modeling in the UAV system

Minjun Kim-xS3d master candidate

Monday March 28th, 2016

Tables of contents

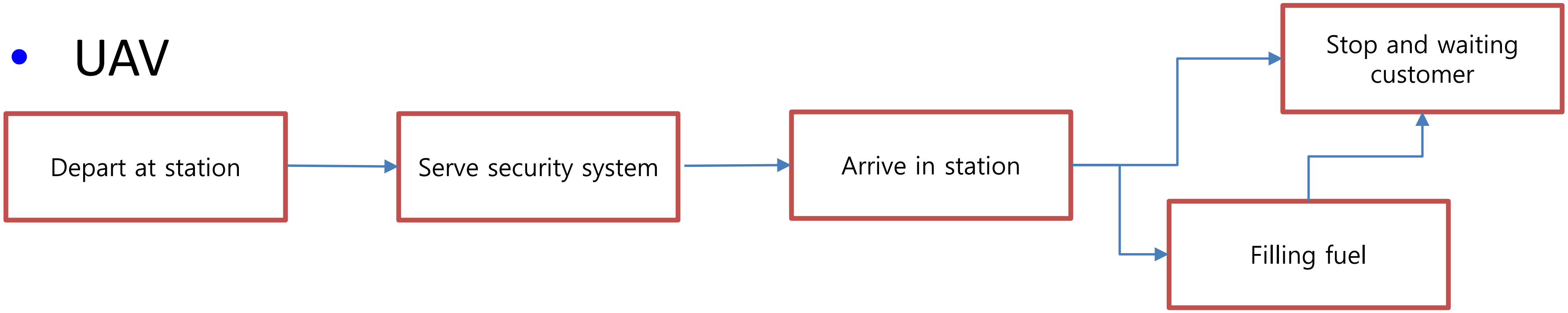
Introduction

- UAV-security system
  - There are various application of UAV system such as border patrol, security system, search and rescue. Among them, we use security system for this research
- Problem
  - In certain time demand for UAV is larger than supply. In that case, customer queue is occurred so we should efficiently allocated UAV resource.

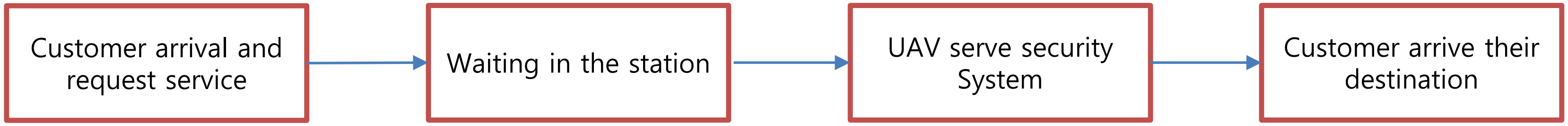
Introduction (System architecture)

- System controller planning in advance based on expected demand

- UAV



- Customer



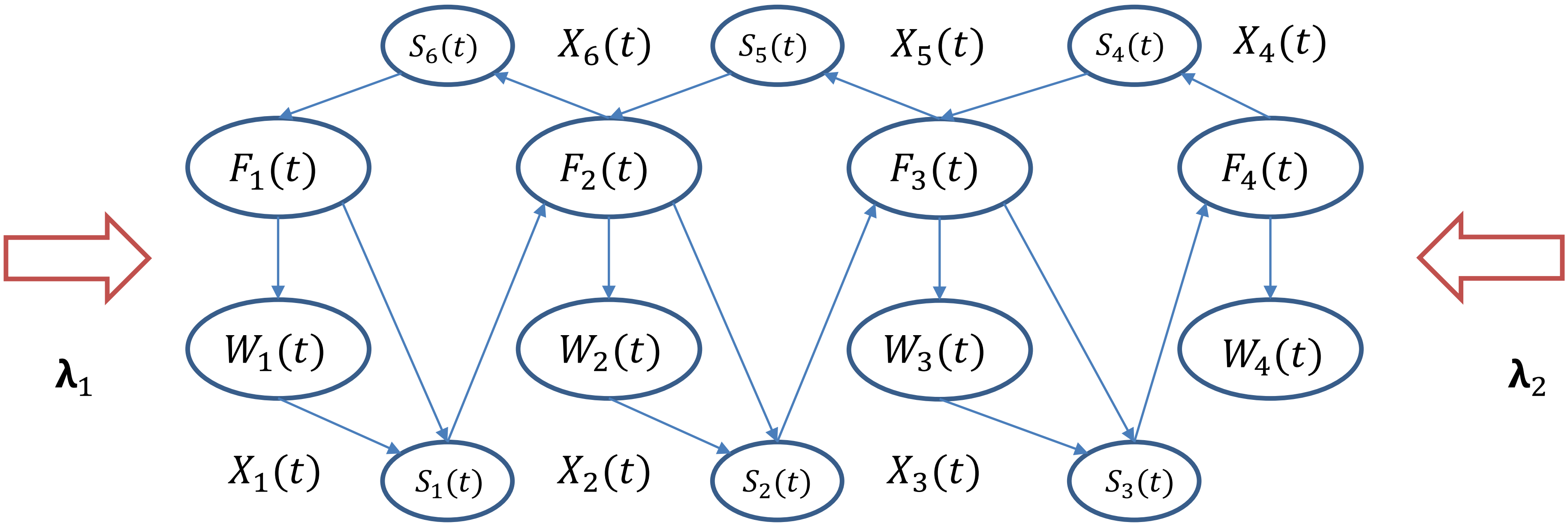
Introduction

- Markovian modeling
  - We introduce markovian modeling in this research.
  - Customer arrival follow Poisson process
  - Security job serve time follow exponential distribution
  - UAV`s fuel replenishment time follow exponential distribution
- Expectation of the research
  - Efficiently allocate UAV resources in the stochastic environment

Problem description

- Objective
  - UAV planning to maximize expected job serve in our system
- Constraints
  - UAV fuel constraint
  - Number of customer in the waiting queue
- Assumption
  - Fixed design and fixed resource in the system
  - UAV`s failure is not considered
  - In initial state UAV`s fuel is fully charged
  - When UAV move, it always contain customer
  - There are sufficient charging machine in each RSS
  - Each UAV is identical

Problem description



Problem description

- A small scale problem is provided for illustration. Consider the system diagram of UAV system with 2 customer arrival locations, 6 security tasks and 4 RSSs.

- $F_i(t)$: The number of UAV receiving replenishment service at RSS i at time t
- $S_i(t)$: The number of UAV conducting task i at time t
- $X_i(t)$: The number of waiting customers for task i at time t
- $W_i(t)$: The number of waiting UAV at RSS i at time t
- Customers arrive to station i as a Poisson process of rate λ_i

- Number of customer in the system at time t: $\sum S_i(t) + \sum X_i(t)$
- Number of UAV in the system at time t: $\sum F_i(t) + \sum S_i(t) + \sum W_i(t)$

Problem description

$$F_1(t) \quad F_2(t) \quad F_3(t) \quad F_4(t)$$

$$S_1(t) \quad S_2(t) \quad S_3(t) \quad S_4(t) \quad S_5(t) \quad S_6(t)$$

$$X_1(t) \quad X_2(t) \quad X_3(t) \quad X_4(t) \quad X_5(t) \quad X_6(t)$$

$$W_1(t) \quad W_2(t) \quad W_3(t) \quad W_4(t)$$

Representation of state at time t

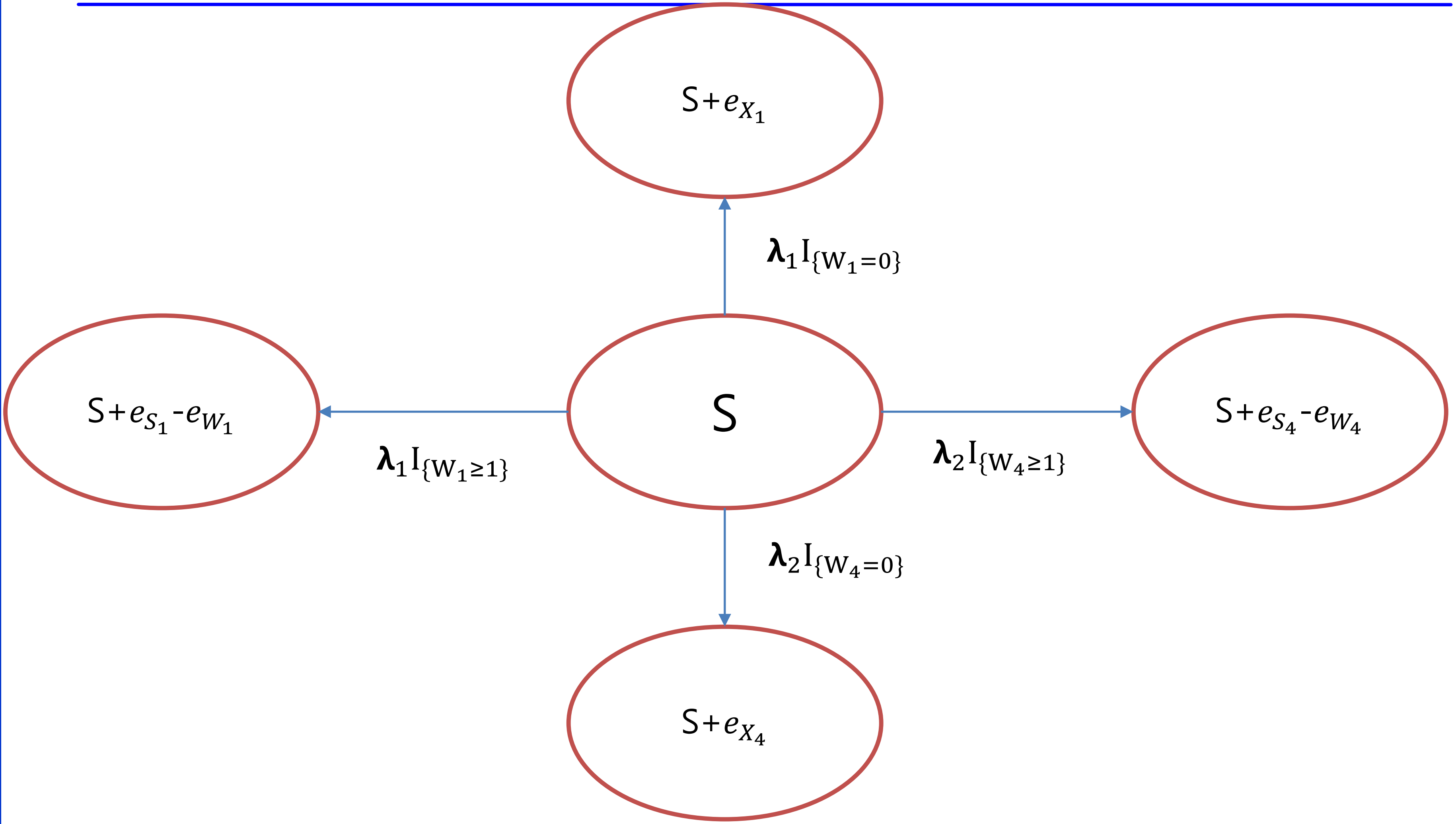
Problem description

- If event is occurred then transition is happened.

- Type of events: Customer arrival, finish fuel replenishment, finish security service
  - Overall 12 kinds of events are exist.

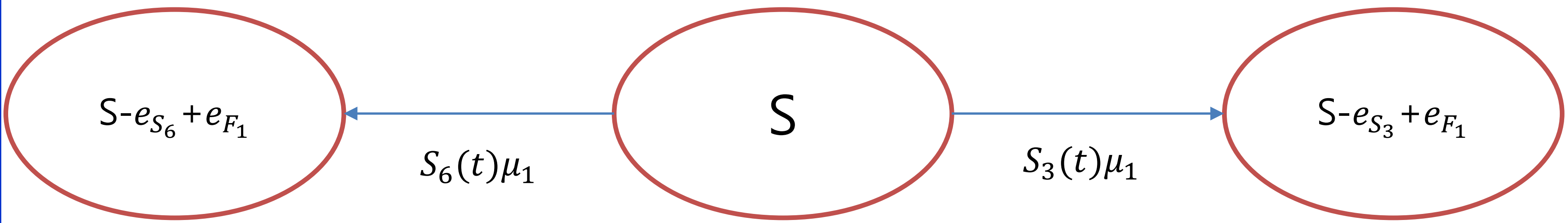
- We introduce decision free model in this research
  - Predetermined policy on the decision epoch
  - After finish F_2 and F_3 we set priority in customer on X_2 and X_3each.

Transition diagram (Customer arrival)



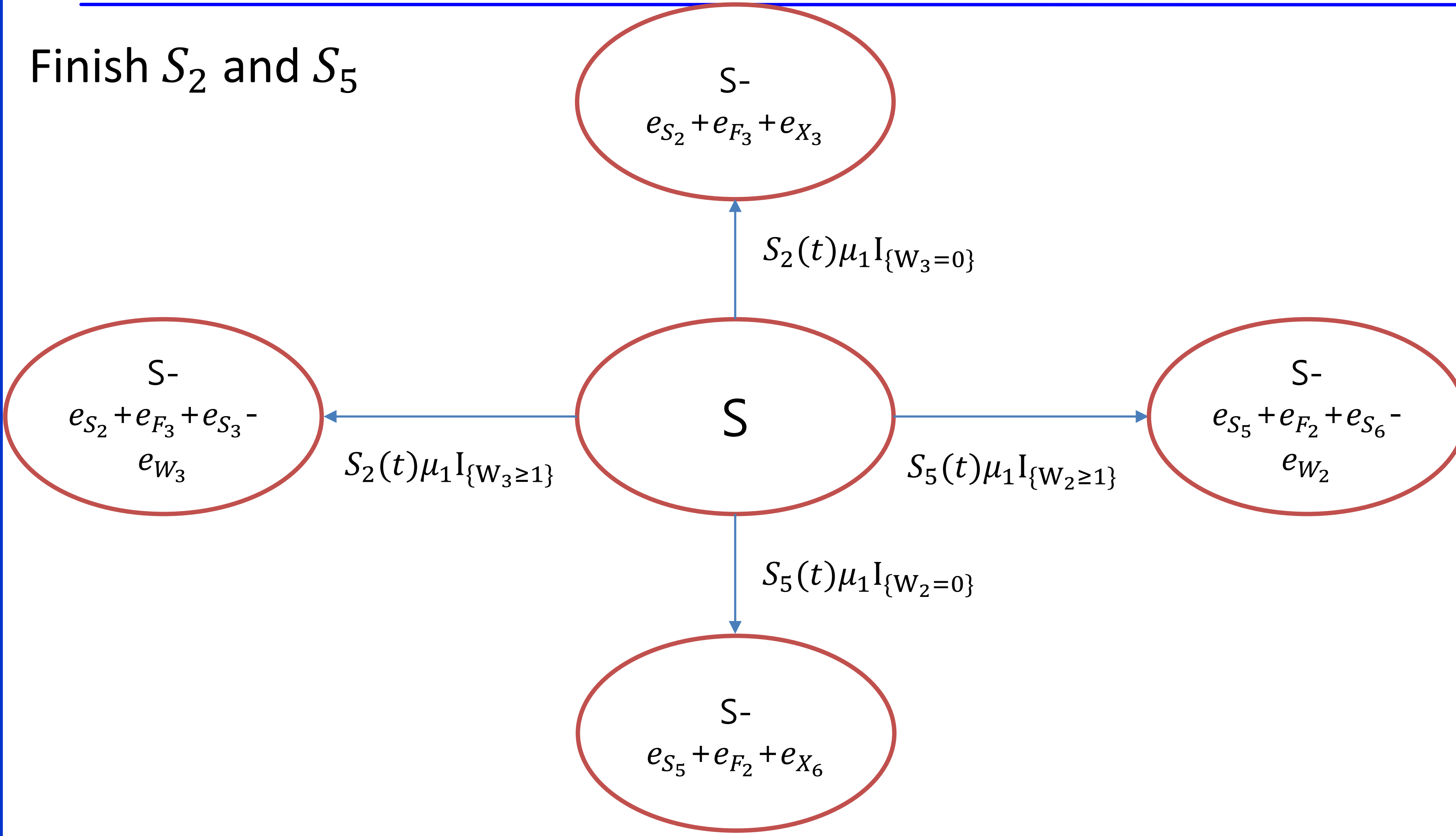
Transition diagram (Finish security service)

Finish S_3 and S_6



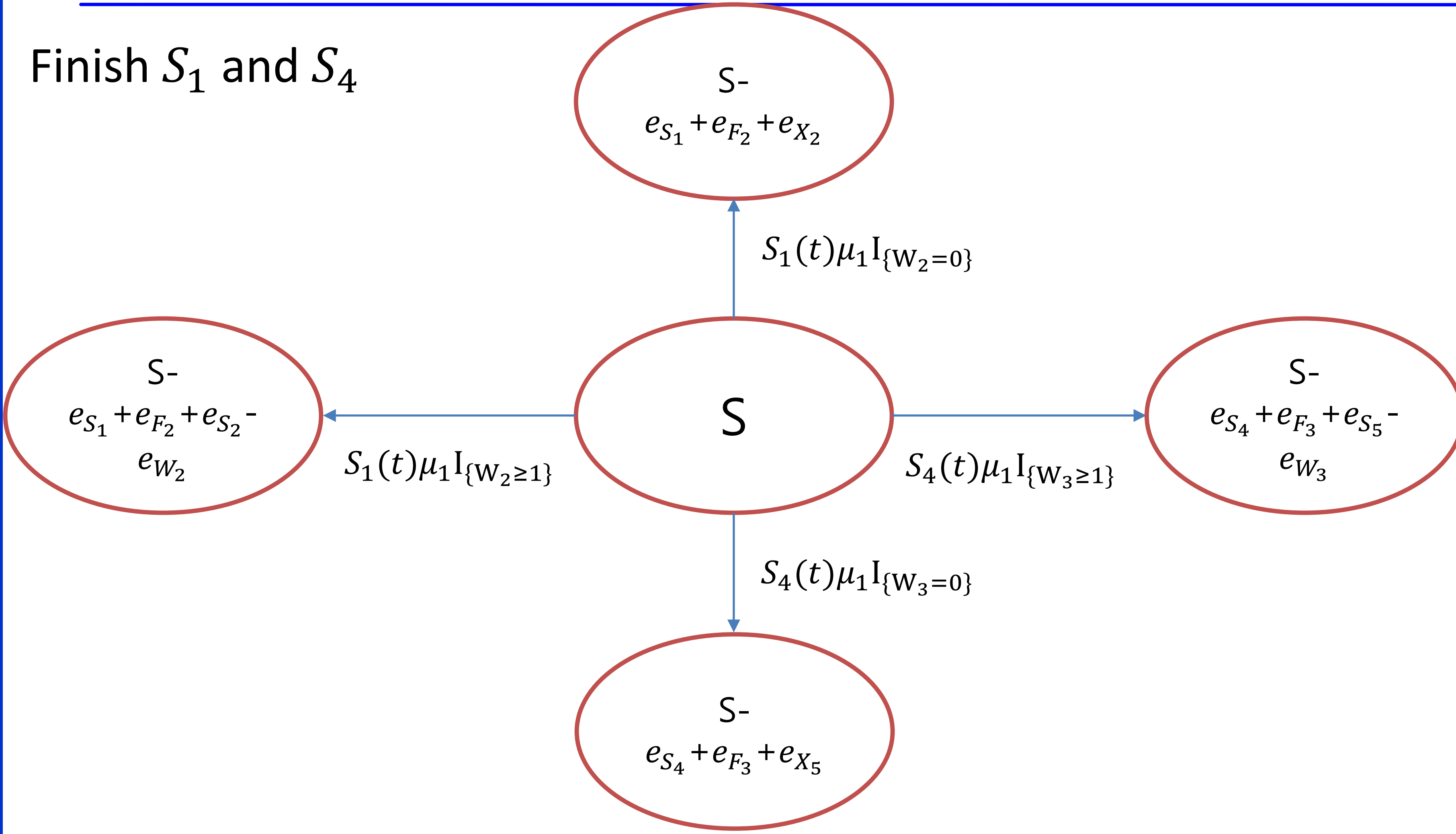
Transition diagram (Finish security service)

Finish S_2 and S_5



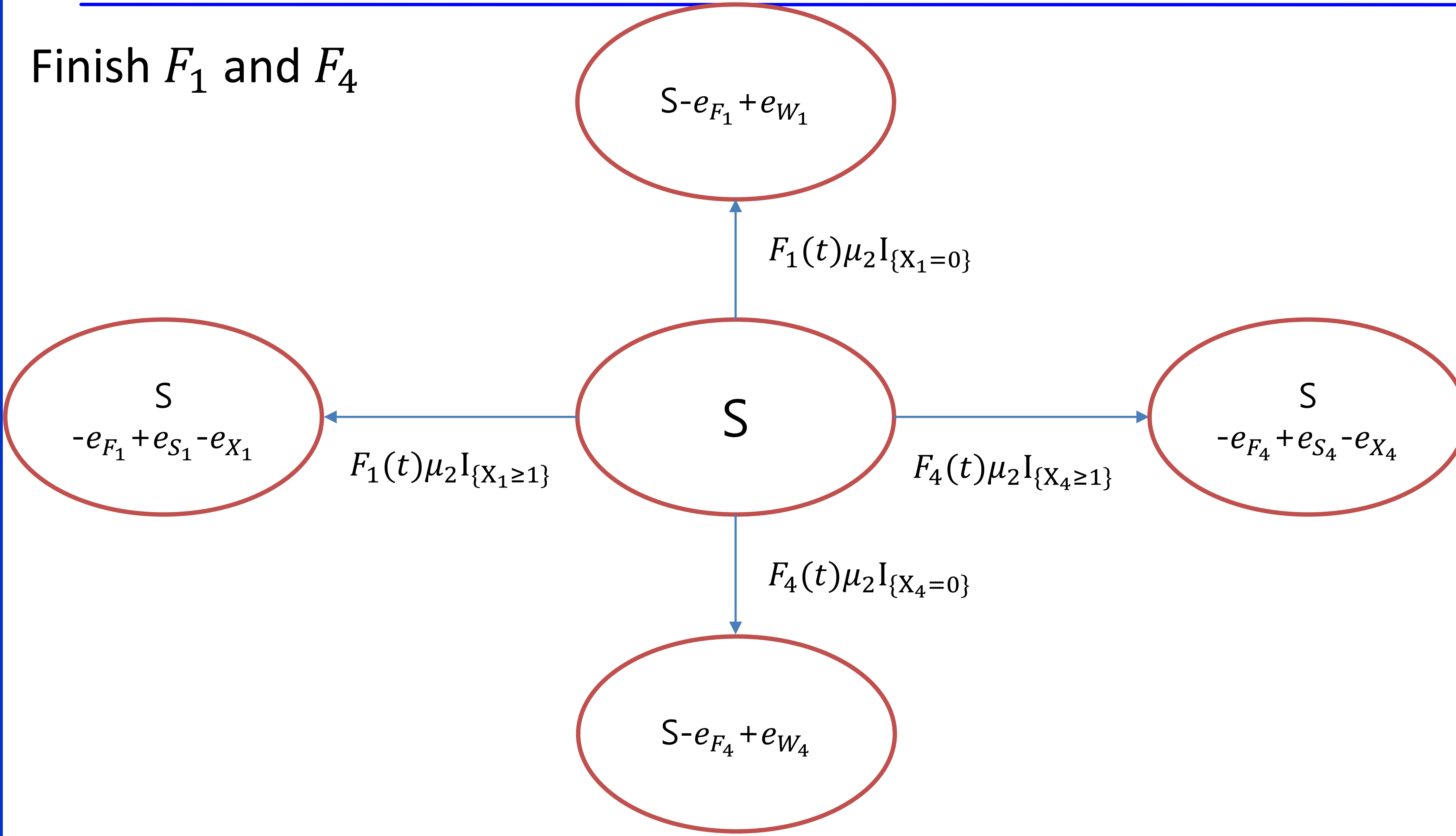
Transition diagram (Finish security service)

Finish S_1 and S_4



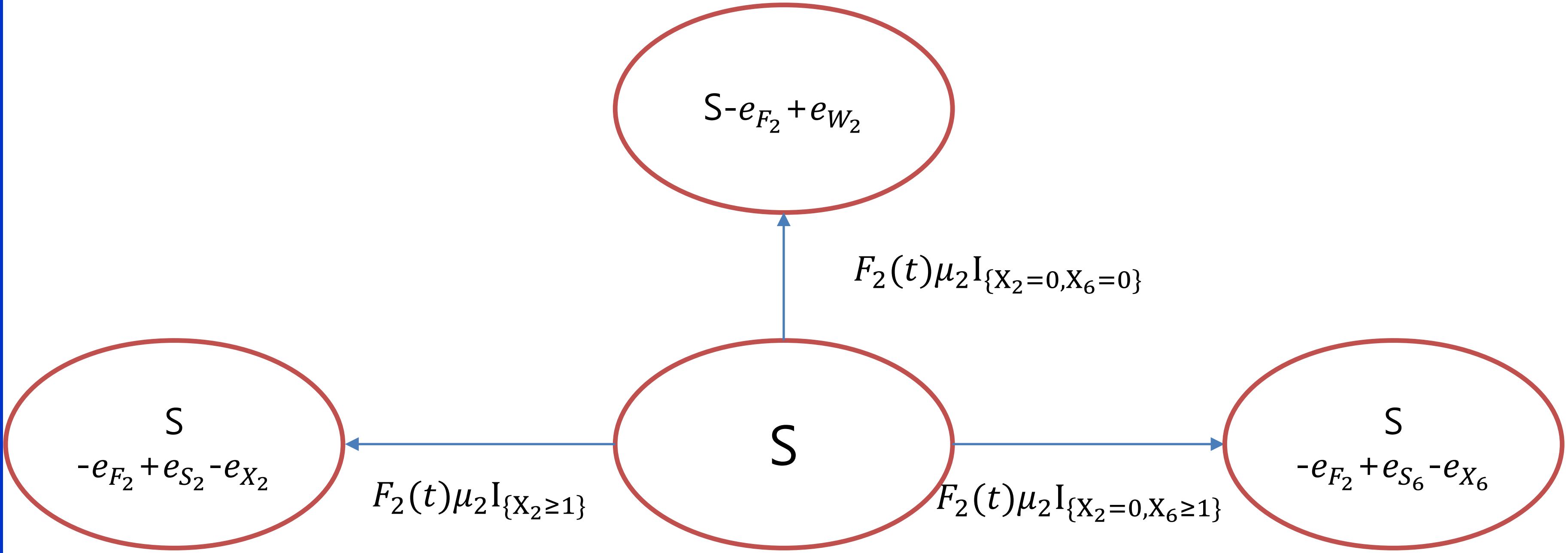
Transition diagram (Finish fuel replenishment)

Finish F_1 and F_4



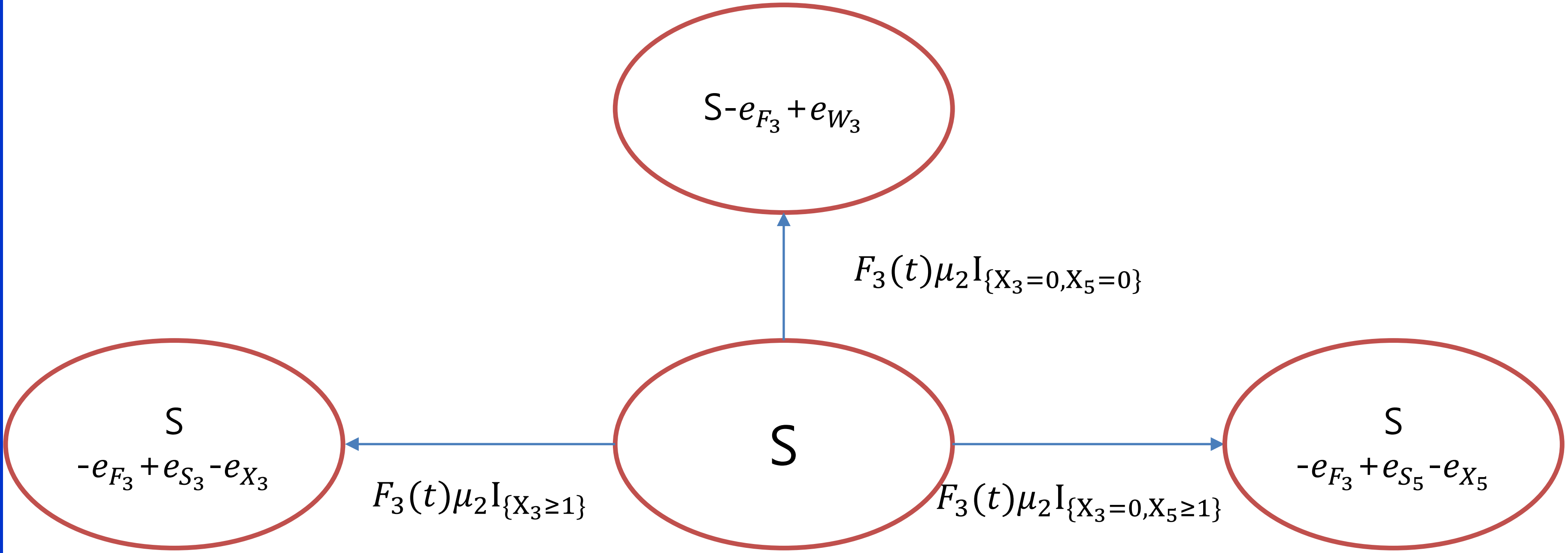
Transition diagram (Finish fuel replenishment)

Finish F_2



Transition diagram (Finish fuel replenishment)

Finish F_3



Generating state

- Use Matlab 2014 generate all possible state based on transition diagram and constraints

- Initial state and input variable
  - Initial state = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 2, 2, 1}
  - $\boldsymbol{\lambda}_1$ (Customer arrival rate for right direction): 0.2
  - $\boldsymbol{\lambda}_2$ (Customer arrival rate for left direction): 0.3
  - μ_1 (service rate for split job): 0.1
  - μ_2 (service rate for fuel replenishment): 0.08

Logic of generating state and transition probability

- 1. Generate gstate function which represent the transition from current status it return possible next state and transition probability
- 2. Set stateset(set of all state) and initial state, transition probability matrix
- 3. From initial state, update the stateset use gstate function , when operate gstate function check the generating state is exist in the stateset. If it is not add to the stateset
- 4. Also when update stateset, update the transition probability matrix
- 5. Repeat above procedure(3,4) until stateset is not updated

Example of gstate

- Example of gstate
  - Input state {0 1 1 0 0 0 1 0 0 0 1 0 0 0 0 0 0 1 1 1}
  - Result

0	1	1	0	0	0	1	0	0	0	2	0	0	0	0	0	0	1	1	1	0.263
0	1	1	0	0	0	1	1	0	0	1	0	0	0	0	0	0	1	1	0	0.394
0	1	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1	1	1	0
0	1	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1	1	1	0
0	1	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	1	1	1	0.132
0	1	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1	1	1	0
0	1	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1	1	1	0
0	1	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1	1	1	0
0	1	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1	1	1	0
0	0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	2	1	1	0.105
0	1	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1	2	1	0.105
0	1	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1	1	1	0

Example of generating state and transition probability

- Example of generating state and transition probability
  - Constraints: $\sum \mathrm{X_i}(\mathrm{t}) \leq 6, \mathrm{X_1}(t) \leq 4, \mathrm{X_4}(t) \leq 4$
  - Changes that number of states in the stateset

1	3	8	20	46	98	188	330	538	822	1186	1624	2127	2685	3291	3935	4612	5316	6034	6748	7434	8074	8640	9120	9507	9819	10071	10271	10417	10523	10593	10637	10657	10663	10663

  - Total state: 10663
  - Generating state time: 1546.73
  - Generating state and transition probability matrix time: 1935.37

ch	1
h	10663
i	12
in	*1x1000000 cell*
ini	*1x20 double*
j	10663
p	*1x35 double*
q	10663
s	*1x10663 double*
stateset	*10663x21 double*
t2	1.9354e+03
tp	*10663x10663 double*
x	33

Result of generating state and transition probability

- S: generating state, T: transition probability matrix

# of states	CPUtime(S)	CPUtime(S+T)	Gap
1707	40.72	43.08	2.36
2149	63.97	67.90	3.93
3219	140.39	155.28	14.89
4103	223.72	259.61	35.89
4545	283.94	321.69	37.75
6499	565.87	661.08	95.21
7383	763.70	946.57	152.87
9337	1370.19	1612.01	241.82
10663	1546.73	1935.37	388.64
17037	3837.54	5929.87	2092.33

Simulation

- Use matlab 2014b simulate the system
- Initial state and input variable
  - Initial state = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 2, 2, 1}
  - $\boldsymbol{\lambda}_1$ (Customer arrival rate for right direction): 1/60
  - $\boldsymbol{\lambda}_2$ (Customer arrival rate for left direction): 1/50
  - μ_1 (service rate for split job): 1/10
  - μ_2 (service rate for fuel replenishment): 1/40
  - Total service time: 0~720, Customer entering time:0 ~ 660
- Output variable
  - T_i: Customer arrival time for direction i
  - S_i: First row imply Security job i`s start time and Second row imply security job i`s complete time
  - F_i: First row imply Fuel replenishment start time and Second row imply fuel replenishment finish time

Example of simulation result

T1	27.11829	41.3921	78.54022	96.47124	114.5602	163.1688	172.891	215.1392	218.3333	231.888	352.9123	439.0994	562.5727	600.2008				
T2	7.656081	45.55904	46.43223	65.5523	72.46366	107.5014	169.7998	254.708	256.6857	270.6541	273.2001	293.1296	377.2153	439.1363	581.3231			
S1	27.11829	44.01963	140.3127	171.0521	244.3572	353.706	375.912	506.3967	562.5727									
	50.1283	51.60327	144.8769	178.136	244.6958	361.2944	379.8569	510.26	564.696									
S2	50.1283	51.60327	144.8769	178.136	244.6958	361.2944	379.8569	515.3999	564.696									
	53.93975	55.76839	154.9415	226.7945	250.5049	365.7344	380.5813	557.4705	568.758									
S3	53.93975	72.20416	154.9415	232.6336	250.5049	365.7344	380.5813	557.4705	568.758									
	60.38054	75.10327	175.0432	239.1383	285.5272	369.8333	397.8354	592.3199	570.952									
S4	7.656081	87.16511	109.8198	186.6458	270.6718	307.2214	399.9191	453.0503	611.9919									
	20.40151	90.21028	146.2489	191.4971	275.0992	308.5947	416.0323	465.0692	627.1976									
S5	20.40151	91.16943	146.2489	191.4971	275.0992	308.5947	416.0323	465.0692	627.1976									
	24.30086	91.62227	153.9665	195.8229	284.3845	332.1747	463.6834	481.8676	633.1956									
S6	24.30086	91.62227	153.9665	200.2451	284.3845	332.1747	463.6834	481.8676	633.1956									
	34.74246	102.3429	168.8537	223.1366	309.0493	335.2887	469.0201	483.7032	636.2224									
F1	34.74246	102.3429	168.8537	223.1366	309.0493	335.2887	469.0201	483.7032	636.2224									
	44.01963	140.3127	171.0521	244.3572	375.912	353.706	506.3967	559.6541	727.0542									
F2	24.30086	50.1283	51.60327	91.62227	144.8769	153.9665	178.136	195.8229	244.6958	284.3845	332.1747	361.2944	379.8569	463.6834	481.8676	510.26	564.696	633.1956
	28.40796	65.12528	86.16677	138.7406	167.6963	200.2451	224.8629	202.5134	312.7784	317.466	348.3231	407.5828	395.2651	515.3999	556.5458	521.7625	686.8763	638.7802
F3	20.40151	53.93975	55.76839	90.21028	146.2489	154.9415	191.4971	226.7945	250.5049	275.0992	308.5947	365.7344	380.5813	416.0323	465.0692	557.4705	568.758	627.1976
	91.16943	72.20416	121.9288	105.2581	180.8103	241.1299	232.6336	248.4197	269.2108	282.161	371.6142	388.7826	388.0914	477.0214	502.067	635.4783	600.1656	640.8457
F4	60.38054	75.10327	175.0432	239.1383	285.5272	369.8333	397.8354	570.952	592.3199									
	87.16511	109.8198	186.6458	270.6718	307.2214	453.0503	399.9191	611.9919	739.3441									

Job serve rate for right direction: length of S3/length of T10 = 9/14=0.643

Job serve rate for left direction: length of S6/length of T2 = 9/15=0.6

Result of simulation

- Simulate 30 times

Experiment	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
# of served Customer(S_3)	9	7	4	7	7	8	6	8	6	10	10	5	9	9	8	10	7	7	6	10	7	6	8	8	7	8	8	8	9	9
# of arrival Customer(T_1)	14	7	16	8	10	13	6	11	9	14	11	5	12	11	9	18	18	8	9	11	12	6	13	14	11	10	12	14	9	10
# of served Customer(S_6)	9	7	4	6	8	8	7	7	6	11	9	6	9	9	7	10	7	6	6	10	7	7	8	9	6	8	8	8	9	8
# of arrival Customer(T_1)	15	9	5	11	9	9	12	13	10	12	17	21	13	10	15	24	10	9	10	21	7	15	17	20	14	16	14	19	12	18

- Mean of served customer in right direction: 7.7
- Mean of arrival customer in right direction: 11.03
- Mean of served customer in left direction: 7.67
- Mean of arrival customer in left direction: 13.57
- Total served rate for right direction: 231/331=0.698
- Total served rate for left direction: 230/407=0.565

Analysis of the result

- Stochastic scheduling suffer from complexity issue
- In case of generating state, number of state`s square is proportional to CPUtime
- In case of generating state and transition probability matrix, number of state`s cube is proportional to CPU time for (transition probability matrix + state) – generating state
- In the simulation, right direction served rate is higher than left direction
- In the simulation, the gap between customer in two direction is less or equal than 1
- In the simulation, Variation of result is high

Future research plan

- Research more complex model
- Get more data from the simulation and do Regression analysis
- Markovian analysis in the model
- Research in the decision model
- In order to reduce the complexity suggest approximate dynamic programming algorithm
- Including priority in customer in the system
- Use different measure such as customer waiting time